

Kyiv University of Aviation and
Information Technologies (Kyiv, Ukraine)
Publishing House Education and Science (Prague, Czech Republic)
Newcastle University Business School (Newcastle, Great Britain)
Berlin University of Economics and Law (Berlin, Germany)
Public organization "Association of Scientists
of Ukraine" (Kyiv, Ukraine)

Introduction to Business-Analytics

Edited by V. Zaluzhnyi

Praha, České republika 2025

**Kyiv University of Aviation and
Information Technologies (Kyiv, Ukraine)
Publishing House Education and Science (Prague, Czech Republic)
Newcastle University Business School (Newcastle, Great Britain)
Berlin University of Economics and Law (Berlin, Germany)
Public organization "Association of Scientists
of Ukraine" (Kyiv, Ukraine)**

Introduction to Business-Analytics

Edited by V. Zaluzhnyi

Praha, České republika 2025

ISBN 978-80-907845-9-8

UDK 005.511(075.8)

Recommended for publication by the Academic Council of the Kyiv University of Aviation and Information Technologies (Protocol No. 5 of December 12, 2025)

Vydavatel:

Publishing house Education and Science s.r.o. IČO : 271 56 877. Frýdlanská
15/1314 , Praha 8. MS v Praze , oddíl C, vložka 100614

Reviewers:

Dankevych V., Doctor of Economics, Professor, Dean of the Faculty of Law,
Public Administration and National Security of the Polesie National University;

Andryushchenko K., Doctor of Economics, Professor, Professor of the
Department of Business Economics and Entrepreneurship of the Vadym Hetman
Kyiv National Economic University.

Introduction to business-analytics. Edited by V. Zaluzhnyi. Textbook. Prague:
Publishing house education and science s.r.o., 2025. 307 p.

© Publishing house Education and Science s.r.o., Česká republika, 2025

© Public Organization "Association of Scientists of Ukraine", Ukraine, 2025

© autoři článků, 2025

ISBN 978-80-907845-9-8

УДК 005.511(075.8)

Рекомендовано до друку Вченою радою Київського університету авіаційних та інформаційних технологій (протокол № 5 від 12 грудня 2025 р.)

Рецензенти:

Данкевич В.Є., д.е.н., професор, декан факультету права, публічного управління та національної безпеки Поліського національного університету;

Андрющенко К.А., д.е.н., професор, професор кафедри бізнес-економіки та підприємництва Київського національного економічного університету імені Вадима Гетьмана.

Вступ до бізнес-аналітики. За заг. редакцією В.Ф. Залужного. Підручник. Прага: Publishing house Education and Science s.r.o., 2025. 307 с.

Цей підручник є фундаментальним виданням, що пропонує системний підхід до опанування професії бізнес-аналітика в сучасних умовах. Навчальний матеріал поєднує класичні методології розробки програмного забезпечення з інструментами математичної статистики та галузевої аналітики, що робить його універсальним посібником для фахівців різного профілю.

У першій частині видання детально розглядається роль бізнес-аналітики в IT-ландшафті, життєвий цикл проєкту та моделі розробки (від традиційних до RUP). Значна увага приділена гнучким методологіям Agile, Scrum та Kanban, а також концепціям професійного вдосконалення та впровадження змін.

Другий великий блок присвячено інженерії вимог та візуальному моделюванню. Видання пропонує покрокове освоєння ключових нотацій: від структурного аналізу (IDEF0) та подієвого моделювання у контексті ARIS (eEPC) до сучасних виконуваних моделей у нотації BPMN. Окремо висвітлено методи виявлення вимог, ідентифікацію стейкхолдерів та роботу з варіантами використання (Use Cases).

Унікальною особливістю підручника є включення розділів з кількісного аналізу даних, де розглядаються стаціонарні часові ряди, авторегресійні моделі (AR, MA) та прогнозування трендів. Завершується видання оглядом практичного застосування бізнес-аналітики у спеціалізованих доменах: фінансах (оцінка ризиків), маркетингу (сегментація споживачів) та HR (управління талантами).

Підручник призначений для студентів закладів вищої освіти, слухачів курсів з бізнес-аналізу, а також для початківців-практиків, які прагнуть систематизувати знання та навчитися приймати рішення на основі даних у різних сферах бізнесу.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1. Бізнес-аналітика в контексті методологій розробки програмного забезпечення	
(Залужний В.Ф.)	10
Тема 1. Роль та місце бізнес-аналітики в ІТ-сфері	10
Тема 2. Життєвий цикл проєкту	17
Тема 3. Моделі розробки програмного забезпечення	26
Тема 4. RUP (rational unified process)	33
Тема 5. Гнучкі моделі управління командою: AGILE	39
РОЗДІЛ 2. Ключові AGILE-методології: SCRUM та KANBAN	
(Коваль В.В.)	46
Тема 1. KANBAN	46
Тема 2. Ролі в KANBAN та СТАТИК (S.T.A.T.I.K.)	55
Тема 3. Гнучкі моделі управління командою: SCRUM	62
Тема 4. Події та артефакти в SCRUM	70
Тема 5. Принцип СЮ-ХА-РІ (SHU-HA-Rİ) в освоєнні майстерності	76
РОЗДІЛ 3. Робота з вимогами: класифікація та методи виявлення	
(Романенко Є.О.)	82
Тема 1. Основи розробки вимог	82
Тема 2. Користувачі та класифікація	90
Тема 3. Організаційна модель компанії	82
Тема 4. Вимоги та їх класифікація	102
Тема 5. Збір та виявлення вимог	109
Тема 6. Інтерв'ювання та аналіз вимог	114
РОЗДІЛ 4. Моделювання вимог та бізнес-процесів	
(Скуріневська Л.В.)	120
Тема 1. Деталізація варіантів використання: альтернативи та	

виключення	120
Тема 2. Опис вимог за допомогою процесного підходу	125
Тема 3. Види бізнес-процесів та нотації моделювання	131
Тема 4. Структурні моделі: нотація IDEF0	135
Тема 5. Деталізація та додаткові елементи нотації IDEF0	140
Тема 6. Моделі потоків робіт: нотація eEPC	146
Тема 7. Чередування «подія – функція» та використання операторів у нотації eEPC	152
РОЗДІЛ 5. Перехід до виконуваних моделей: поглиблення eEPC та введення в BPMN	
(Дацій О.І.)	157
Тема 1. Доповнення моделі процесу деталями у нотації eEPC	157
Тема 2. Використання інтерфейсів для зв'язування процесів та ієрархія ARIS	162
Тема 3. Особливості побудови моделі eEPC та рекомендації з оптимізації	166
Тема 4. Порівняльний аналіз нотацій IDEF0 та eEPC	170
Тема 5. Сильні та слабкі сторони нотації eEPC та типові помилки	174
Тема 6. Виконувані моделі процесів: нотація BPMN	179
Тема 7. Типи зв'язків між елементами діаграми BPMN	186
РОЗДІЛ 6. Стаціонарні часові ряди: поняття, діагностика та реалізація	
(Квашук Д.М.)	193
Тема 1. Поняття стаціонарності	193
Тема 2. Авторегресійні моделі AR(P)	214
Тема 3. Моделі ковзного середнього MA(Q)	220
Тема 4. Аналіз часових рядів: лінійна регресія, TS-ряди, тренд і сезонність	226

РОЗДІЛ 7. Бізнес-аналітика у фінансах, маркетингу та HR

(Литвин О.Є.) 237

Тема 1. Аналітика у фінансах: моделі прогнозування, оцінка ризиків
та управління капіталом 237

Тема 2. Маркетингова аналітика: поведінка споживачів, сегментація
та цифрові канали 252

Тема 3. Аналітика в HR: управління талантами, ефективність
персоналу та HR-передбачення 272

ВСТУП

Стрімка цифрова трансформація глобальної економіки та ускладнення архітектури інформаційних систем зумовили виникнення гострої потреби у фахівцях, здатних забезпечити ефективну комунікацію між стратегічними цілями бізнесу та технологічними можливостями розробки. Бізнес-аналітик сьогодні виступає не лише ретранслятором вимог, а стратегічним посередником, архітектором змін та гарантом створення цінності для стейкхолдерів. Підручник «Вступ до бізнес-аналітики» за загальною редакцією В.Ф. Залужного розроблений як фундаментальна база для формування системного аналітичного мислення та професійних компетенцій у сфері ІТ.

Навчальний матеріал розпочинається з концептуального визначення ролі бізнес-аналітики в структурі сучасних організацій. У першому та другому розділах детально аналізується місце аналітика в різних моделях розробки програмного забезпечення (SDLC) — від класичних каскадних підходів та ітеративного процесу RUP (Rational Unified Process) до панівних сьогодні гнучких методологій.

Особливий акцент зроблено на вивченні екосистеми Agile, зокрема фреймворків Scrum та Kanban. Автори не обмежуються лише описом подій та артефактів, а заглиблюються у філософські та практичні аспекти впровадження змін, такі як метод S.T.A.T.I.K. для еволюційного дизайну систем та концепція SHU-NA-RI, що описує етапи професійного становлення фахівця від слідування правилам до творчої адаптації методик.

У третьому розділі розглядається основа професійної діяльності — робота з вимогами. Автори пропонують системний підхід до класифікації вимог (бізнес-вимоги, користувацькі, функціональні та нефункціональні), а також детально описують методи їхнього виявлення. Теоретичний опис технік інтерв'ювання та аналізу організаційних моделей компанії

супроводжується рекомендаціями щодо ідентифікації стейкхолдерів, що є критично важливим для мінімізації ризиків на ранніх етапах проєкту.

Четвертий та п'ятий розділи присвячені інструментарію візуалізації бізнес-логіки. Підручник пропонує послідовне освоєння трьох фундаментальних рівнів моделювання: структурний рівень (IDEF0): вивчення функціональної декомпозиції систем, управління входами, виходами, механізмами та управлінням; подієвий рівень (eEPC): аналіз логічної послідовності подій та функцій у межах архітектури ARIS, що дозволяє детально описати операційну діяльність підприємства; виконуваний рівень (BPMN): перехід до сучасних стандартів опису процесів, які є зрозумілими як бізнесу, так і технічним спеціалістам, і можуть слугувати основою для автоматизації в BPMS-системах.

Порівняльний аналіз цих нотацій дозволяє студентам свідомо обирати інструментарій залежно від мети аналізу та цільової аудиторії документації.

Новацією даного видання є включення шостого розділу, присвяченого математичним методам аналізу даних. В умовах Big Data бізнес-аналітик повинен володіти апаратом стаціонарних часових рядів, авторегресійними моделями (AR) та моделями ковзного середнього (MA). Це забезпечує перехід від описового аналізу («що сталося») до предиктивного («що станеться»).

Завершальний сьомий розділ розкриває прикладний характер бізнес-аналітики у фінансовому секторі, маркетингу та управлінні людським капіталом (HR). Це дозволяє студенту побачити універсальність аналітичних методів та специфіку їхнього застосування для оцінки ризиків, сегментації споживачів або прогнозування ефективності персоналу.

Цей підручник розроблений відповідно до вимог освітніх стандартів для підготовки бакалаврів та магістрів спеціальностей «Комп'ютерні науки», «Інженерія програмного забезпечення» та «Менеджмент». Він також стане настільною книгою для практикуючих фахівців, які прагнуть

систематизувати знання та підготуватися до сертифікації за міжнародними стандартами.

Мета авторів — надати читачеві не лише теоретичні знання, а й практичну впевненість у використанні методологічного інструментарію бізнес-аналізу для розв'язання складних бізнес-проблем сучасності.

РОЗДІЛ 1. БІЗНЕС-АНАЛІТИКА В КОНТЕКСТІ МЕТОДОЛОГІЙ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

**(Залужний В.Ф., доктор філософії в галузі права, Інститут проблем
реєстрації інформації Національної академії наук України)**

Тема 1. Роль та місце бізнес-аналітики в ІТ-сфері

Існує два види бізнес-аналітиків:

Звичайний (не-ІТ) бізнес-аналітик – той, хто працює в контексті бізнесу в цілому. Такий фахівець бере участь у вдосконаленні процесів, оптимізації витрат тощо.

Бізнес-аналітик в ІТ – той, чия робота відбувається в межах окремих ІТ-проектів (закупівлі, впровадження, внесення змін тощо). Його завдання – взаємодія із зацікавленими сторонами з боку замовника (бізнес-користувачі, технічний персонал тощо) для збору вимог, що виходять від бізнесу.

При цьому ІТ – вкрай снобістська сфера, і за межами великих корпорацій про бізнес-аналітиків, які не працюють у ІТ, зазвичай ніхто не знає.

Водночас у галузі немає чіткого поділу на «бізнес-» і «системних» аналітиків. У кожній компанії (а часом і в окремих командах усередині однієї компанії) межі їхніх компетенцій сильно розмиті, аж до того, що часто на проєкті взагалі немає системного аналітика, а його обов'язки розподілені між бізнес-аналітиком і розробниками. І це не єдина особливість: у світі також немає єдиного однозначного визначення, достатньо почитати формулювання з «біблії» бізнес-аналітиків – BABOK (A Guide to the Business Analysis Body of Knowledge).

Бізнес-аналітик – це не та професія, в якій чітко сказано, що «потрібно копати звідси й до сюди». Ви постійно будете стикатися з вимогами, які часто суперечать не лише одна одній, а й здоровому глузду; замовники самі

не знатимуть, чого хочуть, а ви нерідко не знатимете, чи можливо реалізувати їхні бажання насправді. Але разом із тим, бізнес-аналітик – одна з найбільш «інженерних» професій, бо всі інші члени команди реалізують те, що ви вигадали та написали/намалювали.

Кажуть, девізом IBM у давні часи було твердження: «Ми продаємо нашим замовникам не те, що вони хочуть, а те, що їм потрібно». Не знаю, наскільки це правда, але формулювання дуже показове. З часом, коли ви наберетеся досвіду, вам потрібно буде дотримуватися цього девізу.

Атмосферна частина закінчилася, далі інформація в книзі буде набагато більш структурованою. Ну а щоб ви знали, що відповісти на співбесіді на це питання, просто вивчіть визначення з BABOK, який не просто так називається «Посібник зі зведення знань з бізнес-аналізу»®.

Основи розробки ПЗ: Ролі в команді

Менеджер проєкту (Project manager)

- Проєкт – це тимчасове підприємство, спрямоване на створення унікального продукту, послуги або результату.
- Менеджер проєкту – керівник проєктної команди, відповідальний за управління проєктом, досягнення цілі проєкту в рамках бюджету, в строк і з заданим рівнем якості.

Менеджер релізів – у реаліях Enterprise-компаній так називають керівників команд, які управляють процесом, коли ІТ-продукт із проєкту вже перетворився на сервіс, але перебуває на доопрацюванні відповідно до запитів на зміни (33).

Менеджер продукту (Product manager)

- Продукт – кінцевий відчутний результат (продукт, сервіс, послуга), що надається клієнту.
- Менеджер продукту задає стратегію розвитку продукту. Він знає все про бажання споживача, передбачає, що буде з ринком (проводить аналіз

ринку) тощо, добре уявляє продукт з точки зору споживача, але не заглиблюється в технічний контекст.

Ключова відмінність: Product Manager вирішує, що зробити (нові функції, стратегія), а Project Manager – коли і як (реалізація, терміни, бюджет).

Архітектура ІС – це набір структурних елементів, організованих певними способами для взаємодії одне з одним, які складаються в єдиний програмно-апаратний комплекс, побудований для досягнення конкретних бізнес-цілей (таблиця 1.1).

Таблиця 1.1

Архітектура ІС

Вид Архітектури	Опис	Відповідальність Архітектора
Архітектура підприємства (Enterprise Architecture)	Високорівнева архітектура всього підприємства, що покриває бізнес-потреби ІТ-можливостями.	Enterprise Architect (EA): Відповідає за узгодження бізнес-стратегії компанії з технологічною архітектурою. Приймає концептуальні рішення.
Бізнес-архітектура (Business Architecture)	Описує всі бізнес-процеси, бізнес-актори, бізнес-сутності та бізнес-правила з точки зору бізнесу. Не залежить від технологій.	EA / Бізнес-аналітик (на нижчому рівні).
Архітектура рішення (Solution Architecture)	Архітектура ПЗ, яке реалізує функції бізнес-архітектури.	Solution Architect (SA): Якщо EA вирішує, що робити, то SA – як робити. Розробляє архітектуру одного або декількох додатків.
Технічний/Програмний Архітектор (Technical/Software Architect)	Технічний експерт. Займається рівнем реалізації конкретної системи. Визначає стек технологій, сервіси, підходи до розробки, обирає БД.	Technical Architect (TA): Часто є техлідом. Вирішує проблеми, поставлені SA, і ставить задачі розробникам.

Таблиця 1.2

Роль та завдання аналітика

Роль	Основне завдання
Бізнес-аналітик (Business Analyst)	Описує бізнес-процеси "як є" (as is) і формулює, якими вони мають бути "як буде" (to be) для досягнення цілей бізнесу. Фіксує бізнес-вимоги (схеми процесів, use case).
Системний аналітик (Systems Analyst)	Сполучна ланка між впровадженням і розробкою. Збирає функціональні вимоги, обговорює їх із командою розробки і описує постановку задачі (технічне завдання).
Продуктовий аналітик (Product Analyst)	Місток між бізнесом і даними. Працює з менеджером продукту. Збирає та інтерпретує дані (метрики, воронки) для розуміння поведінки користувачів і прийняття рішень щодо розвитку продукту.

Дизайнер (Designer)

- User Experience (UX): Шлях клієнта від точки входу до точки виходу. Відповідає за функціональність (наскільки зручно заповнити заявку).
- User Interface (UI): Вигляд інтерфейсу. Відповідає за статику (кольорова палітра, розташування об'єктів).
- UX-дизайнер: Проектувальник, вивчає потреби, будує логічні схеми роботи інтерфейсу, тестує прототипи.
- UI-дизайнер: Визначає візуальні елементи (кольори, форми, кнопки). Візуалізує робочий прототип, створює макет.

Інші ключові ролі відображені у таблиці 1.3

Життєвий цикл ПЗ (SDLC, Software Development Life Cycle)

Життєвий цикл ПЗ (SDLC) – це період часу, який починається з моменту ухвалення рішення про необхідність створення програмного продукту і закінчується в момент його повного видалення з експлуатації.

SDLC – це основні етапи, через які проходить будь-який програмний продукт. Плюс-мінус завжди мають місце такі етапи:

1. Фаза планування.
2. Фаза збору вимог / аналізу.

3. Фаза дизайну / проєктування.
4. Фаза розробки.
5. Фаза тестування.
6. Фаза впровадження.
7. Фаза підтримки.

Таблиця 1.3

Інші ключові ролі

Роль	Опис
Спеціаліст з інформаційної безпеки (Application Security)	Аналізує додаток з точки зору ІБ, визначає вимоги до безпеки, займається пошуком та аналізом вразливостей.
Розробник (Developer)	Frontend-розробник (клієнтська сторона, інтерфейс); Backend-розробник (серверна розробка, бізнес-логіка); Fullstack (поєднує обидва обов'язки).
Тестувальник (QA Engineer)	Відповідає за якість ПЗ, перевіряючи його відповідно до вимог. Виконує функціональне, безпекове, продуктивне тестування тощо (залежно від спеціалізації).
Технічна підтримка (Tech Support)	Підрозділяється на рівні: Level 1 (прийом повідомлень, ідентифікація, типові відповіді); Level 2 (систематизація, аналіз, вирішення складніших проблем); Level 3 (зазвичай команда розробки, вирішення унікальних проблем).

Важливо: ЖЦ Проєкту лежить всередині ЖЦ Продукту. ЖЦ проєкту закінчується, коли продукт передається в операційну діяльність, а ЖЦ продукту триває до його списання, включаючи всі наступні доопрацювання та нові проєкти змін.

Питання для самоперевірки, тести для самоконтролю

1. Чим «Бізнес-аналітика» відрізняється від «Бізнес-аналізу (аудиту)»?
2. Опишіть ключову відмінність між не-ІТ бізнес-аналітиком та ІТ бізнес-аналітиком.

3. Поясніть, чому межі компетенцій бізнес- та системного аналітика часто розмиті в компаніях.
4. Сформулюйте девіз IBM, який рекомендується взяти за основу бізнес-аналітику, що набрався досвіду.
5. Які основні відмінності в ролях та завданнях між Product Manager і Project Manager?
6. Дайте визначення поняттю «Архітектура інформаційної системи» (IC).
7. Які основні види архітектури розглядаються в контексті Enterprise Architecture (Корпоративної архітектури)?
8. Яка роль архітектора (EA, SA, TA) вирішує, що робити, а яка – як робити?
9. У чому полягає основна відмінність між завданнями Бізнес-аналітика та Системного аналітика?
10. Яка роль відповідає за функціональність інтерфейсу (шлях клієнта), а яка – за його вигляд (статичу)?
11. Назвіть усі фази Життєвого циклу програмного забезпечення (SDLC).
12. Поясніть, чому Життєвий цикл проєкту лежить усередині Життєвого циклу продукту.

Обов'язкові та додаткові задачі, приклади

Обов'язкова задача 1: Визначення ролі

Ви – керівник команди, яка створює мобільний додаток для замовлення доставки їжі. Сформулюйте 3 ключові питання для кожного з фахівців, щоб визначити сферу його відповідальності:

- Product Manager
- Project Manager
- Solution Architect
- Business Analyst

Приклад розв'язання (фрагмент):

- Product Manager: 1) Яку нову функцію (наприклад, "групове замовлення") ми додаємо наступного кварталу і чому? 2) Які ключові метрики успіху цієї функції?
- Project Manager: 1) Скільки часу займе реалізація функції "групове замовлення" і який її бюджет? 2) Які ризики зриву термінів?
- Solution Architect: 1) Які компоненти системи потрібно змінити для підтримки "групового замовлення"? 2) Чи потрібно інтегрувати нову сторонню службу?
- Business Analyst: 1) Які бізнес-вимоги до функції "групове замовлення" (наприклад, "можливість додавати більше 5 учасників")? 2) Опишіть поточний процес замовлення (as is) та запропонуйте, яким він має бути (to be) з цією функцією.

Додаткова задача 2: Класифікація вимог

Визначте, чи є наведені нижче вимоги: Бізнес-вимоги (BA), Функціональні вимоги (SA), чи Архітектурні/Технічні вимоги (TA/SA):

1. Кнопка "Оформити замовлення" повинна бути синього кольору.
2. Час відповіді системи на запит пошуку товару не повинен перевищувати 300 мс.
3. Клієнт повинен мати можливість редагувати свою адресу доставки після її збереження.
4. Мета проєкту – збільшити середній чек на 15% протягом 6 місяців.
5. Система має бути розроблена з використанням мікросервісної архітектури на платформі Java Spring.
6. Для автентифікації користувачів необхідно використовувати протокол OAuth 2.0.

Відповідь:

1. UI (Дизайнер)

2. Архітектурні/Технічні (TA/SA) / Вимоги до Продуктивності (QA)
3. Функціональні (SA)
4. Бізнес-вимоги (BA)
5. Архітектурні/Технічні (TA/SA)
6. Архітектурні/Технічні (TA/SA)

Довідково-інформаційні дані для розв'язування задач (таблиці, схеми тощо)

1. Схема розмежування відповідальності

Сфера	Що? (Стратегія)	Як? (Проектування)	Коли/Скільки? (Управління)	Реалізація/Якість
Бізнес/Продукт	Product Manager, Enterprise Architect	Business Analyst	Project Manager	-
Система/ІТ	Solution Architect	Systems Analyst	Project Manager	Developer, QA

Бібліографічний список

1. ІІБА. A Guide to the Business Analysis Body of Knowledge® (BABOK® Guide). International Institute of Business Analysis, 2015. (Основне джерело визначень ролей та завдань бізнес-аналізу).
2. Pressman, R. S. Software Engineering: A Practitioner's Approach. McGraw-Hill Education. (Для розуміння Життєвого циклу ПЗ).
3. The Open Group. The TOGAF® Standard (Для розуміння Корпоративної Архітектури та її видів).

Тема 2. Життєвий цикл проекту

Будь-який Життєвий Цикл (ЖЦ) проекту включає в себе чотири основні фази:

1. Початкова. Фаза переходу ідеї в проєкт. На цій стадії зазвичай пишеться статут проєкту, визначаються первинні критерії успіху проєкту. Також на цій фазі формуються ключові ролі: менеджер проєкту, власник, спонсор.

2. Підготовка та організація. На цьому етапі виділяються ресурси для проєкту, створюється команда, складається план проєкту, визначається бюджет, а також шукаються постачальники послуг, якщо вони необхідні.

3. Виконання робіт. Це фаза створення продукту. Менеджер проєкту отримує звіти про хід виконання і контролює процес. Далі відбувається фінальне схвалення продукту перед підписанням акта приймання робіт.

4. Фінальна стадія. Основна суть цієї фази – підвести підсумки після завершення проєкту, підписати акт приймання-передачі та передати продукт в операційну діяльність.

Вплив стадій на показники проєкту, як залежно від стадії проєкту змінюються ключові показники:

1. Вартість проєкту та чисельність персоналу (залученість ресурсів)
2. Вплив зацікавлених сторін, невизначеність, можливість ризиків
3. Вартість змін у проєкті

4. Вартість змін: Найнижча вартість змін – на початку. На цьому етапі найлегше вплинути на зміну проєкту, оскільки саме тут визначаються цілі. Якщо цілі зміняться пізніше, обране рішення може виявитися невідповідним, і витрати на переробку будуть дуже високими.

Приклад: Ви вирішили побудувати каркасний будинок на пальовому фундаменті. Під кінець будівництва ви хочете камін. Каркасні будинки не розраховані на таку вагу. Доведеться розбирати підлогу і заливати окремий фундамент, що значно збільшить вартість і складність робіт.

- Затрати ресурсів: Максимальні витрати ресурсів спостерігаються в період виконання робіт (ближче до кінця). Це пов'язано з тим, що більшість

спірних моментів прояснилася, команда активно працює, і часто підключаються додаткові ресурси, щоб вкластися в терміни.

- Вплив зацікавлених сторін / Невизначеність: Максимальна невизначеність та вплив на початку проєкту, коли рішення ще не зафіксовані.

Деталізація етапів ЖЦ проєкту (SDLC)

1. Етап планування

На цій стадії потрібно максимально знизити невизначеність. Це найбільш критичний етап для створення успішної системи.

Ключові дії:

- Визначити проблеми, цілі, завдання та ресурси (персонал, витрати, інфраструктура).
- Розглянути альтернативні рішення (консультації з клієнтами, постачальниками).
- Провести аналіз здійсненності (feasibility study) – вивчення можливості створення продукту відповідно до вимог та ресурсів.

Роль аналітика: Бізнес-аналітик (або Product Manager) відіграє провідну роль, працюючи з інвестором/замовником для вироблення єдиного візії майбутнього продукту. Створюється образ продукту, визначаються межі проєкту, орієнтовні терміни та бюджет.

Вхідні документи	Вихідні документи
Ідея, концепція, економічне обґрунтування.	Статут проєкту, Product Vision Document (PVD) або Marketing Requirement Document (MRD), дорожня карта, план управління ризиками, список зацікавлених сторін, ресурсний план.

2. Етап аналізу вимог

На цьому етапі потрібно чітко визначити, задокументувати та затвердити вимоги до продукту.

Вхідні документи	Вихідні документи
Реєстр зацікавлених осіб, дорожня карта.	Технічне завдання (ТЗ).

3. Етап проєктування

Ця фаза настає після того, як вимоги чітко зрозумілі. На ній визначаються елементи, компоненти, архітектура, інтерфейси та типи даних системи.

Ключові дії: Визначення набору технологій, фреймворків, вимог до зберігання інформації (бази даних), конкретної архітектури (з точки зору компонентів, взаємозв'язків та потоків даних).

Вхідні документи	Вихідні документи
Технічне завдання (ТЗ).	Архітектурне рішення, специфікація вимог до ПЗ, дизайн-макети, тест-кейси.

4. Етап розробки

Це власне процес створення системи, коли дизайн вже завершений і представлений. На цьому етапі пишеться код.

Вхідні документи	Вихідні документи
Архітектурне рішення, специфікація вимог до ПЗ, дизайн-макети, тест-кейси.	Працююча версія ПЗ, готова до тестування.

5. Етап тестування

На цьому етапі перевіряється працездатність системи.

Важливий момент: Пошук, реєстрація, відстеження, виправлення та повторне тестування дефектів тривають доти, доки не будуть виправлені всі дефекти певної критичності або доки продукт не досягне стандартів якості, обумовлених на етапі аналізу.

Вхідні документи	Вихідні документи
Тест-кейси, версія ПЗ.	Звіт про тестування, баг-репорти.

6. Етап впровадження та підтримки

Продукт офіційно випускається на ринок. Часто розгортання відбувається поетапно. На етапі підтримки команда виправляє проблеми, впроваджує нові функції та доопрацьовує продукт відповідно до нових вимог.

Вхідні дані	Вихідні дані
Консультавання користувачів; виправлення дефектів; доопрацювання.	Стратегія підтримки користувачів, користувацька документація.

Стандарти Життєвого циклу ПЗ

Стандарт визначає загальну структуру ЖЦ ПЗ у вигляді триступеневої моделі, що складається з процесів, видів діяльності та задач.

Стандарт визначає процеси, що використовуються при придбанні, постачанні, розробці, застосуванні, супроводженні та припиненні застосування програмних продуктів.

Категорія Процесів	Призначення	Приклади Процесів
Процеси угоди	Вироблення угод між двома організаціями (придбання або постачання продукту).	Процес придбання, Процес постачання.
Процеси організаційного забезпечення проєкту	Менеджмент можливостей організації (підтримка та управління проєктами, забезпечення ресурсами та інфраструктурою).	Менеджмент моделі ЖЦ, Менеджмент інфраструктури, Менеджмент якості.
Процеси проєкту	Планування, оцінка та управління проєктом.	Планування проєкту, Управління проєктом та його оцінка, Менеджмент ризиків, Менеджмент конфігурації.
Технічні процеси	Визначення вимог, перетворення їх на продукт, інсталяція, функціонування, супроводження.	Аналіз системних вимог, Проєктування архітектури, Реалізація, Кваліфікаційне тестування.

Обмеження: Не деталізує процеси в розрізі методів, не встановлює вимог до документації, не приписує чітких схем побудови ЖЦ. Організація сама несе відповідальність за вибір моделі.

Наступний стандарт передбачає наступні стадії та етапи створення автоматизованої системи (АС):

1. Формування вимог до АС (Обстеження, формування вимог користувача).
 2. Розробка концепції АС.
 3. Технічне завдання (Розробка та затвердження ТЗ).
 4. Ескізний проєкт (Розробка попередніх рішень).
 5. Технічний проєкт (Розробка детальних проєктних рішень).
 6. Робоча документація (Розробка робочої документації та адаптація програм).
 7. Введення в дію (Підготовка персоналу, монтаж, пусконаладжувальні роботи, випробування).
 8. Тестування АС.
 9. Супроводження АС (Гарантійне та післягарантійне обслуговування).
- Гнучкість: Допускається виключати стадію "Ескізний проєкт", об'єднувати "Технічний проєкт" і "Робочу документацію" у "Техноробочий проєкт", а також паралельно виконувати окремі етапи.

Питання для самоперевірки, тести для самоконтролю

1. Назвіть чотири основні фази Життєвого циклу проєкту. Яка фаза включає підписання акта приймання-передачі?
2. На якій стадії проєкту вартість змін є найнижчою і чому?
3. На якій стадії спостерігаються максимальні затрати ресурсів і чому?
4. Які ключові документи є виходом етапу Планування для продукту "під замовлення" і "для відкритого ринку"?
5. Яка основна мета етапу Аналізу вимог?
6. Які вихідні документи отримують після етапу Проєктування?
7. У чому полягає важливий момент етапу Тестування, пов'язаний із дефектами?
8. Назвіть три основні групи процесів.

9. Що визначає стандарт? Які стадії він передбачає для створення автоматизованої системи (АС)?

10. У чому полягає гнучкість стандарту щодо стадій "Ескізний проєкт" та "Технічний проєкт"?

Обов'язкові та додаткові задачі, приклади

Обов'язкова задача 1: Аналіз вартості змін

Проєкт зі створення нового банківського мобільного додатку перебуває на етапі Розробки (Фаза 3). Замовник вимагає додати функцію "переказу коштів за номером телефону" (вимога, яка не була зафіксована раніше).

1. Обґрунтуйте, на якій кривій (1, 2 чи 3) ви помітите найбільший вплив цієї зміни.

2. Поясніть, чому вартість цієї зміни буде високою.

Відповідь:

1. Найбільший вплив буде на Кривій 3 (Вартість змін).

2. Вартість буде високою, оскільки зміна вноситься на пізньому етапі.

Для реалізації цієї функції необхідно:

- Повернутися на Етап Аналізу (додати вимоги) та Проєктування (змінити архітектурне рішення, додати інтерфейси).

- Змінити вже написаний код на Етапі Розробки.

- Переписати Тест-кейси та провести додаткове Тестування.

- Це спричинить додаткові витрати ресурсів (Крива 1) та затримку термінів.

Обов'язкова задача 2: Класифікація виходів

Класифікуйте наведені вихідні документи за відповідною фазою ЖЦ проєкту (Начальна, Підготовка/Організація, Виконання робіт, Фінальна стадія):

1. Звіт про тестування.

2. Статут проєкту.

3. Акт приймання-передачі.
4. План проєкту.
5. Працююча версія ПЗ.
6. Список зацікавлених сторін.

Відповідь:

1. Виконання робіт (або Тестування).
2. Начальна.
3. Фінальна стадія.
4. Підготовка та організація.
5. Виконання робіт (або Розробка).
6. Начальна.

Довідково-інформаційні дані для розв'язування задач (таблиці, схеми тощо)

1. Зведення етапів ЖЦ Проєкту та їхніх Входів/Виходів

Фаза ЖЦ	Етап SDLC	Основний Вхід	Основний Вихід
Початкова	Планування	Ідея, концепція	Статут проєкту
Підготовка/Організація	Аналіз Вимог	Статут проєкту	Технічне завдання (ТЗ)
Виконання робіт	Проєктування	ТЗ	Архітектурне рішення, Дизайн
Виконання робіт	Розробка	Архітектурне рішення	Працююча версія ПЗ
Виконання робіт	Тестування	Працююча версія ПЗ	Звіт про тестування
Фінальна стадія	Впровадження/Підтримка	Протестований продукт	Акт приймання-передачі, Документація

Бібліографічний список:

1. Project Management Institute (PMI). A Guide to the Project Management Body of Knowledge (PMBOK® Guide) – Seventh Edition. PMI, 2021. (Ключове джерело знань з управління проєктами, яке детально описує фази, процеси та

області знань, пов'язані з ЖЦП, включаючи етапи ініціації, планування, виконання та завершення).

2. International Institute of Business Analysis (IIBA). A Guide to the Business Analysis Body of Knowledge® (BABOK® Guide) – Version 3. IIBA, 2015. (Основний посібник для бізнес-аналітиків, що визначає завдання бізнес-аналізу на різних стадіях ЖЦП, особливо на критичних етапах планування та аналізу вимог).

3. ISO/IEC/IEEE 12207:2017. Systems and software engineering — Software life cycle processes. Geneva: International Organization for Standardization. (Міжнародний стандарт, що визначає загальну структуру ЖЦ ПЗ та необхідні процеси, види діяльності та завдання для забезпечення якості та системності розробки).

4. ISO 21500:2021. Guidance on project management. Geneva: International Organization for Standardization. (Міжнародний стандарт, що надає високорівневе керівництво та загальні поняття щодо управління проектами).

5. The Open Group. The TOGAF® Standard, 10th Edition. (Посібник, що дає уявлення про Архітектуру підприємства (Enterprise Architecture), яка слугує основою для прийняття рішень на початковій фазі проекту, забезпечуючи узгодженість ІТ-рішень із бізнес-стратегією).

6. Pressman, Roger S. Software Engineering: A Practitioner's Approach. McGraw-Hill Education. (Класичний підручник з інженерії програмного забезпечення, що детально розглядає різні моделі ЖЦПЗ — від Каскадної до Гнучкої (Agile) — та їхні етапи).

7. IEEE Standard 1074-2017. Standard for Developing Software Life Cycle Processes. IEEE. (Стандарт, що описує гнучкі рамки для створення, вибору та управління процесами ЖЦПЗ).

Тема 3. Моделі розробки програмного забезпечення

Модель розробки ПЗ описує стадії його життєвого циклу та всі події, що відбуваються на кожній із них. У цьому розділі розглянуто найбільш популярні та основні моделі розробки.

Класичні Моделі Розробки

1. Модель кодування та усунення помилок

Ця модель є найпростішою і часто застосовується лише в навчальному процесі (наприклад, для лабораторних робіт).

Алгоритм:

1. Постановка завдання.
2. Створення програми (кодування).
3. Тестування.
4. Аналіз результатів тестування та можливий перехід до кроку 1.

Ця модель неактуальна для професійної розробки ПЗ через надмірну простоту, яка не дозволяє конкурувати з іншими моделями.

2. Каскадна модель, або Waterfall (Водоспад)

В цій моделі розробка здійснюється поетапно: кожна наступна стадія починається строго після закінчення попередньої, і рух можливий лише вперед. При правильному використанні «Водоспад» вважається однією з найбільш швидких і простих моделей.

Етапи:

1. Збір вимог. Збираються вимоги до продукту (визначаємо вимоги замовника).
2. Аналіз вимог. Вимоги аналізуються, деталізуються, уточнюються, обговорюються, документуються (формуємо вимоги до ПЗ).
3. Проектування. Проектуються та узгоджуються логіка роботи ПЗ і вимоги до дизайну; обираються інструменти. На виході — детальна документація для розробників.

4. Розробка (Реалізація).
5. Тестування (Перевірка).
6. Впровадження, підтримка. Виведення в промислову експлуатацію та переведення на підтримку.

Застосування:

- Для середніх і великих проєктів із залученням багатьох команд.
- Для проєктів, у яких вимоги та межі прозорі й точно відомі на початку ЖЦП (проєкт зрозумілий і простий).
- Наприклад, у медичній та космічній галузях, де є велика база нормативних документів, на основі яких можна скласти вимоги.
- Будь-яке будівництво є прикладом «Водоспаду».

Проблема: При виявленні недоробок чи помилок на пізніх етапах, повернення на ранні стадії ускладнене та дороге.

Розширення моделі:

- Каскадна модель зі зворотним зв'язком: Додає цикли зворотного зв'язку для повернення на попередню стадію, якщо вимоги змінилися або виявлено помилки.
- Каскадна модель з проміжним контролем: Збільшує час на розробку за рахунок проведення проміжних коригувань між фазами. Це дозволяє знизити ризики та підвищити надійність системи.

3. Інкрементна (Інкрементальна) модель розробки

Дає можливість створювати продукт частинами (інкрементами). Кожна частина – готовий фрагмент підсумкового продукту, який ідеально не потребує значних змін.

Процедура розробки передбачає на першому етапі випуск продукту з базовою функціональністю, а потім послідовне додавання нових функцій (інкрементів). Процес триває до створення повної системи.

Фактично, ми отримуємо кілька циклів розробки – своєрідний «мультиводоспад», де кожен цикл ділиться на фази: визначення вимог, проєктування, написання коду, впровадження, тестування.

Застосування:

- Для проєктів, у яких точне технічне завдання прописано вже на старті.
- Коли продукт повинен швидко вийти на ринок (базова версія).

Гнучкі та Адаптивні Моделі Розробки

4. Ітеративна модель розробки

Це модель, за якої замовник не зобов'язаний одразу прописувати докладне технічне завдання і може не мати повного розуміння кінцевого продукту.

Створення починається з реалізації частини функціоналу, який стає базою для визначення подальших вимог. У результаті отримуємо мінімально життєздатний продукт (Minimum Viable Product, MVP). Процес повторюється (ітерується).

Ключові характеристики:

- Не вимагає повної специфікації вимог на старті.
- Версія може бути неідеальною, але роботоздатною.
- Кожен крок є результативним, кожна версія – працездатною.

Приклад (візуальний): Ітераційна «розробка» Мони Лізи. У першій ітерації є лише начерк, у другій – кольори, у третій – деталі. Продукт стає кращим, але завжди є цілісним малюнком.

Застосування:

- У великих довгострокових проєктах із відсутністю чітких вимог або ймовірністю їхньої динамічної зміни.
- Коли важливий аналіз ризиків та витрат.
- При розробці нової лінійки продуктів.

5. V-модель розробки (Verification and Validation model)

Це вдосконалена каскадна модель, яка називається моделлю Верифікації та Валідації. Вона успадкувала структуру «крок за кроком» від «Водоспаду».

Ключова відмінність: Розробка кожного кроку безпосередньо пов'язана з етапом тестування, і вони йдуть паралельно. На ранніх стадіях розробки замовник і команда одночасно складають вимоги до системи та описують, як будуть її тестувати на кожному етапі.

Принцип: Кожна дія з розробки тестується, що забезпечує ретельну перевірку на ранніх стадіях.

6. Спіральна модель (Spiral Model)

Втілює переваги каскадної моделі, але включає аналіз ризиків та передбачає розробку проєкту за допомогою прототипування.

ЖЦПЗ зображується у вигляді спіралі, кожен виток якої відповідає створенню фрагмента або версії ПЗ. У ході процесу проєкт проходить чотири повторювані фази по кілька разів.

Головні фази (повторюються у кожному витку):

1. Визначення цілей, альтернатив, обмежень.
2. Аналіз, визначення та розв'язання ризиків.
3. Фаза розробки та тестування.
4. Планування нової ітерації (наступного витка).

Особливість: Головна особливість — концентрація на можливих ризиках. На виході з кожного витка отримуємо готовий протестований прототип.

Застосування:

- У дослідницьких проєктах та проєктах із високими ризиками.
- Дозволяє послідовно конкретизувати деталі проєкту та обирати оптимальний варіант для реалізації.

Питання для самоперевірки, тести для самоконтролю

1. Яка класична модель розробки не є актуальною для професійної сфери і чому?
2. Які дві основні умови потрібні для ефективного використання Каскадної моделі (Waterfall)?
3. Чому з'явилися розширення Каскадної моделі (з зворотним зв'язком та проміжним контролем)?
4. У чому полягає принципова відмінність Інкрементної моделі від Каскадної?
5. Поясніть ключову відмінність між Інкрементною та Ітеративною моделями.
6. Що таке Minimum Viable Product (MVP) і в якій моделі розробки це поняття є базовим?
7. Яка модель, що є вдосконаленням «Водоспаду», передбачає паралельне тестування та розробку? Як вона називається і що означає її назва?
8. На чому концентрується Спіральна модель? Скільки фаз вона має у кожному витку?
9. Яка модель найбільше підходить для проєктів, де вимоги можуть динамічно змінюватися?
10. Яка модель найкраще підходить для проєктів із високими ризиками та дослідницьким характером?

Обов'язкові та додаткові задачі, приклади

Обов'язкова задача 1: Вибір моделі

Ви – бізнес-аналітик у компанії. Вам потрібно вибрати оптимальну модель розробки для наступних проєктів:

1. Проєкт А: Створення системи управління польотами для нового космічного апарату. Вимоги суворі, фіксовані та повинні бути задокументовані з максимальною деталізацією перед початком кодування.

2. Проєкт Б: Розробка нового мобільного застосунку для соціальних мереж. Вимоги ринку постійно змінюються, необхідно швидко випускати робочі версії для збору зворотного зв'язку від користувачів.

3. Проєкт В: Створення системи електронного документообігу для державної установи. Вимоги відомі, але продукт повинен поступово нарощувати функціонал, починаючи з базових модулів.

Відповідь:

1. Проєкт А (Космос): Каскадна (Waterfall) або V-модель (через високу потребу у фіксованих вимогах, ретельній документації та паралельній верифікації/валідації).

2. Проєкт Б (Мобільний застосунок): Ітеративна або Спіральна (через високу невизначеність, динамічні зміни та потребу у швидкому MVP).

3. Проєкт В (Документообіг): Інкрементна (вимоги відомі, але функціонал нарощується частинами, кожен інкремент — це окремий робочий модуль).

Обов'язкова задача 2: Порівняння інкременту та ітерації

Поясніть різницю між інкрементом та ітерацією на прикладі створення автомобіля:

- Інкрементний підхід:
- Ітеративний підхід:

Відповідь:

- Інкрементний підхід: У першій версії ви створюєте повний автомобіль, але тільки з базовими функціями (двигун, колеса, кермо). У наступному інкременті ви додаєте нову частину функціоналу, наприклад, систему кондиціонування. Продукт завжди цілісний, але неповний.

- Ітеративний підхід: У першій ітерації ви створюєте дуже простий, але робочий транспортний засіб (наприклад, скутер). У другій ітерації ви його вдосконалюєте та розширюєте до мотоцикла, а в третій — до автомобіля.

Продукт щоразу вдосконалюється та еволюціонує на основі попередньої версії, аж до досягнення кінцевої цілі.

Довідково-інформаційні дані для розв'язування задач (таблиці, схеми тощо)

Порівняння основних моделей розробки

Модель	Основний принцип	Коли застосовується	Переваги
Каскадна (Waterfall)	Послідовне виконання фаз, рух лише вперед.	Вимоги фіксовані, проєкт зрозумілий, висока потреба в документації.	Простота, швидкість виконання при фіксованих вимогах.
Інкрементна	Продукт створюється частинами (інкрементами), кожен з яких є робочим.	Вимоги відомі, потрібний швидкий вихід на ринок базової функціональності.	Швидкий реліз базової версії, поступове нарощування функціоналу.
Ітеративна	Розробка повторюється циклами (ітераціями), кожна покращує попередню версію.	Вимоги нечіткі або часто змінюються; висока невизначеність.	Висока гнучкість, зниження ризиків, швидке отримання MVP.
Спіральна	Повторювані цикли (спіралі) з акцентом на аналізі ризиків і прототипуванні.	Високі ризики, науково-дослідні проєкти, інновації.	Фокус на ризиках, висока адаптивність.

Бібліографічний список

1. Project Management Institute (PMI). A Guide to the Project Management Body of Knowledge (PMBOK® Guide) – Seventh Edition. PMI, 2021. (Керівництво, яке обговорює різні підходи до життєвого циклу проєкту, включаючи прогнозовані (наприклад, Каскадна) та адаптивні (Ітеративна, Інкрементна) моделі).

2. International Institute of Business Analysis (IIBA). A Guide to the Business Analysis Body of Knowledge® (BABOK® Guide) – Version 3. IIBA, 2015.

(Джерело, що визначає, як діяльність бізнес-аналізу інтегрується з різними моделями життєвого циклу ПЗ).

3. ISO/IEC/IEEE 12207:2017. Systems and software engineering — Software life cycle processes. Geneva: International Organization for Standardization. (Міжнародний стандарт, що надає загальні рамки для процесів життєвого циклу ПЗ, до яких можуть бути адаптовані конкретні моделі розробки).

4. ISO/IEC/IEEE 15288:2023. Systems and software engineering — System life cycle processes. Geneva: International Organization for Standardization. (Стандарт, що описує процеси життєвого циклу систем, застосовний також до ЖЦПЗ).

5. Royce, Walker W. Managing the Development of Large Software Systems: Concepts and Techniques. 1970. (Стаття, яка часто розглядається як перше систематичне формулювання Каскадної моделі (Waterfall)).

6. Boehm, Barry W. A Spiral Model of Software Development and Enhancement. IEEE Computer, 1988, Vol. 21, No. 5, pp. 61–72. (Оригінальна робота, що представляє Спіральну модель, яка фокусується на управлінні ризиками та прототипуванні).

7. Pressman, Roger S. Software Engineering: A Practitioner's Approach. McGraw-Hill Education. (Підручник, що детально розглядає архітектуру, проектування та різні моделі життєвого циклу розробки ПЗ, включаючи класичні та еволюційні підходи).

Тема 4. RUP (RATIONAL UNIFIED PROCESS)

RUP (Rational Unified Process) – це вже не модель, а методологія розробки.

Модель розробки ПЗ описує, які стадії життєвого циклу проходить продукт.

Методологія включає набір методів з управління розробкою: правила, техніки та принципи, які роблять її ефективнішою, надаючи практичний опис циклів.

RUP використовує ітеративну модель розробки. В кінці кожної ітерації (яка в ідеалі триває 2–6 тижнів) команда має досягти запланованих цілей, доопрацювати проєктні артефакти й отримати проміжну, але функціональну версію кінцевого продукту.

Переваги ітеративної розробки в RUP: швидке реагування на зміни вимог, виявлення та усунення ризиків на ранніх стадіях, ефективний контроль якості.

Фази Життєвого Циклу в RUP

Повний життєвий цикл розробки продукту за RUP складається з чотирьох послідовних фаз, кожна з яких включає одну або кілька ітерацій.

Фаза	Назва (Українською)	Основна Мета	Фокус
1. Inception	Початкова стадія	Розуміння того, що ми створюємо.	Збір інформації, аналіз вимог, визначення образу проєкту в цілому (Project Vision).
2. Elaboration	Уточнення	Розуміння того, як ми це створюємо.	Аналіз вимог, проєктування системи, планування ресурсів, специфікація функцій та дизайну.
3. Construction	Конструювання	Створення бета-версії продукту.	Основна фаза розробки та кодування, побудова продукту як послідовності ітерацій (версій коду).
4. Transition	Впровадження	Створення кінцевої версії продукту.	Фаза впровадження, постачання продукту користувачеві, приймально-здавальні випробування.

RUP проти Waterfall: На відміну від Каскадної моделі, кожна з цих фаз може повторюватися по кілька разів у циклі ітерацій, відображаючи зміну вимог замовника.

Основні Потоки (Workflow) RUP

Методологія RUP ґрунтується на дев'яти основних потоках (робочих процесах), які виконуються протягом кожної ітерації життєвого циклу. Їх ділять на основні (Core Engineering Workflows) та підтримувальні (Core Supporting Workflows).

Core Engineering Workflows (Основні інженерні потоки)

Потік	Суть	Результат / Артефакт
Business Modeling (Бізнес-аналіз)	Аналіз вимог на ітерації, визначення бажаних параметрів системи та потреб користувачів.	Бізнес-моделі, опис процесів.
Requirements (Вимоги)	Формалізація образу системи, збір та управління вимогами, переведення в функціональні специфікації.	Документи рівня менеджменту, Use Cases (прецеденти) в UML.
Analysis and Design (Аналіз і моделювання)	Переведення зібраних вимог у формалізовану програмну модель. Використовується UML.	Опис системи на етапі реалізації (Технічний проєкт), моделі проєкту.
Implementation (Реалізація, кодування)	Власне написання коду.	Працюючий код, виконані модулі.
Test (Тестування)	Тестування продукту на даній ітерації.	Звіти про тестування.
Deployment (Впровадження)	Установка продукту, підготовка персоналу, запуск, приймально-здавальні випробування.	Підготовлений продукт для користувача, стандарти упаковки.

Акценти RUP: На початкових етапах (Inception, Elaboration) основна увага приділяється потокам Business Modeling, Requirements та Analysis and Design. Це спрямовано на скорочення комерційних ризиків і зниження вартості помилок дизайну. У фазі Construction домінують Implementation та Test.

Core Supporting Workflows (Основні підтримувальні потоки)

Потік	Суть	Значення
Management (Управління проектом)	Набір адміністративних дій з управління проектом відповідно до ідеології RUP.	Забезпечення виконання проекту в межах бюджету та термінів.
Configuration Management (Управління конфігурацією та змінами)	Управління версіями продукту, контроль вихідного коду (моделей, модулів, документації), відстеження змін та помилок (bug tracking).	Забезпечення цілісності та відстеження історії продукту.
Environment (Оточення)	Створення та підтримка засобів для аналізу, проєктування, розробки та тестування (як програмне, так і апаратне забезпечення).	Надання необхідної інфраструктури та інструментів для команди.

Артефакти та Відповідальність

У RUP увесь процес розробки розглядається як процес створення артефактів – від початкових документів аналізу до виконуваних модулів, посібників користувача тощо.

RUP чітко визначає обов'язки кожного члена групи розробки на тому чи іншому етапі (коли і хто повинен створити той чи інший артефакт).

Питання для самоперевірки, тести для самоконтролю

1. У чому полягає ключова відмінність між моделлю та методологією розробки ПЗ?
2. Яку модель розробки використовує RUP? Скільки триває ітерація в RUP в ідеалі?
3. Назвіть чотири фази повного життєвого циклу продукту за RUP. Яка фаза відповідає за створення бета-версії?
4. Яка фаза RUP фокусується на "розумінні того, як ми це створюємо"?
5. Скільки основних потоків (workflow) має методологія RUP?
6. Який потік RUP відповідає за формалізацію образу системи та переведення вимог у функціональні специфікації?

7. Який потік RUP використовує Unified Modelling Language (UML) як основний інструмент формалізації?
8. Які два потоки RUP є домінуючими на початкових етапах (Inception, Elaboration) і чому?
9. Назвіть три підтримувальні потоки (Core Supporting Workflows) RUP.
10. Яку роль відіграють артефакти в RUP?

Обов'язкові та додаткові задачі, приклади

Обов'язкова задача 1: Співвідношення фаз і потоків

Визначте, які потоки RUP (за номером) матимуть найбільший акцент у кожній із наведених фаз:

1. Фаза Elaboration (Уточнення):
2. Фаза Construction (Конструювання):

Відповідь:

1. Elaboration: 1. Business modeling; 2. Requirements; 3. Analysis and design.
2. Construction: 4. Implementation; 5. Test.

Обов'язкова задача 2: Артефакт RUP

Команда розробляє нову систему керування запасами (СУЗ). Визначте, який фахівець і в рамках якого потоку створить наступні артефакти:

- Артефакт А: Діаграма прецедентів (Use Case Diagram), що відображає взаємодію "Менеджер складу" із СУЗ.

- Артефакт Б: Керівництво користувача для кінцевого замовника.
- Артефакт В: План ітерацій та графік засідань команди.

Відповідь:

- Артефакт А: Бізнес-аналітик / Потік Requirements (Вимоги).
- Артефакт Б: Технічний письменник / Потік Deployment (Впровадження).
- Артефакт В: Менеджер проєкту / Потік Management (Управління проєктом).

Довідково-інформаційні дані для розв'язування задач (таблиці, схеми тощо)

1. Візуалізація RUP (Фази та Потоки)

2. Призначення фаз RUP

Фаза	Запитання, на яке відповідає	Комерційний Ризик, що знижується
Inception	Що потрібно зробити?	Ризик створення не того продукту.
Elaboration	Як ми це зробимо?	Ризик створення неправильної архітектури.
Construction	Ми це робимо?	Ризик неефективності розробки.
Transition	Чи готовий користувач до продукту?	Ризик неприйняття продукту користувачем.

Бібліографічний список

1. Kruchten, Philippe. The Rational Unified Process: An Introduction. Addison-Wesley Professional, 2003. (Класичний вступний посібник, написаний "батьком" RUP, який чітко описує архітектуру, фази, ітерації та робочі процеси RUP).

2. Jacobson, Ivar; Booch, Grady; Rumbaugh, James. The Unified Software Development Process. Addison-Wesley Professional, 1999. (Фундаментальна праця, яка заклала теоретичну основу RUP, детально описуючи принципи уніфікованого процесу розробки).

3. Scott, Kendal. UML Explained. Addison-Wesley, 2001. (Книга для розуміння Unified Modeling Language (UML), який є основним інструментом формалізації та моделювання в потоці Analysis and Design RUP).

4. International Institute of Business Analysis (IIBA). A Guide to the Business Analysis Body of Knowledge® (BABOK® Guide) – Version 3. IIBA, 2015. (RUP інтегрує діяльність бізнес-аналізу (потік Business Modeling та Requirements), і цей посібник є стандартом для опису цих завдань).

5. ISO/IEC/IEEE 12207:2017. Systems and software engineering — Software life cycle processes. Geneva: International Organization for Standardization.

(Міжнародний стандарт, який описує процеси життєвого циклу, що можуть бути реалізовані за допомогою ітеративного підходу, як у RUP).

Тема 5. Гнучкі моделі управління командою: AGILE

Agile (з англ. — «рухливий, швидкий, спритний») — це набір ідей, принципів та філософія організації проєктної діяльності, а не конкретний метод розробки ПЗ. У IT-лексичі за цією групою методологій закріпилося визначення «гнучкі».

В Agile проєкт постійно адаптується до нових обставин і вимог. Це відрізняє його від Каскадної моделі (Waterfall), де кінцевий продукт має ідеально відповідати вимогам, сформульованим до початку розробки, а внесення змін — це надзвичайна ситуація.

Для Agile нормально постійно вносити зміни до технічного завдання (ТЗ), навіть у розпал розробки, оскільки в пріоритеті — актуальні потреби користувача, а не вихідні установки.

Розробка відбувається через низку циклів — ітерацій. Кожна ітерація — це фактично окремий проєкт, де розробляють фрагмент програми, покращують функціональність або додають нові можливості.

Ідеї та Принципи Agile

Agile базується на чотирьох центральних ідеях (Цінностях Agile), які визначають пріоритети в роботі:

1. Люди та взаємодія між ними важливіші, ніж процеси та інструменти.
2. Працююче ПЗ важливіше, ніж вичерпна документація.
3. Співпраця із замовником важливіша, ніж узгодження умов контракту.
4. Готовність до змін важливіша, ніж дотримання початкового плану.

Важливе уточнення: Слово "важливіше" у перекладі не означає, що права частина твердження (документація, процеси, контракт, план) ігнорується. Ідеї Agile стверджують, що, не заперечуючи важливості правої

частини, ми більше цінуємо ліву. Фокус має бути на лівій частині, але увага приділяється обом.

12 Принципів Agile

1. Найвищий пріоритет – задоволення потреб клієнта завдяки регулярній та ранній поставці цінного ПЗ.

2. Працюючий продукт слід випускати якомога частіше, з періодичністю від пари тижнів до пари місяців.

3. Зміна вимог вітається навіть на пізніх стадіях розробки, оскільки це дає замовнику конкурентні переваги.

4. Розробники та представники бізнесу повинні щоденно працювати разом для оперативного обміну інформацією.

5. Над продуктом мають працювати вмотивовані професіонали. Створіть для них умови та повністю довіртеся їм.

6. Особисте спілкування – найбільш практичний та ефективний спосіб обміну інформацією (допомагає команді швидше досягати єдиної думки).

7. Працюючий продукт – основний показник прогресу. Демонструючи результати щоітераційно, команда інформує всіх учасників про стан проєкту.

8. Інвестори, розробники та користувачі повинні мати можливість нескінченно підтримувати постійний ритм (темп). Це дозволяє давати правдиві обіцянки щодо термінів.

9. Постійна увага до технічної досконалості та якості проєктування підвищує гнучкість (знижує технічний борг).

10. Простота – це мистецтво не робити зайвої роботи (фокус лише на тих функціях, які несуть максимальну цінність клієнту).

11. Найкращі вимоги, архітектурні та технічні рішення народжуються у самоорганізованих командах.

12. Команда має систематично аналізувати можливі способи поліпшення ефективності та відповідно коригувати стиль своєї роботи (постійний контроль та адаптація).

Застосовність Agile до Проєкту

Для оцінки застосовності Agile до проєкту використовують Матрицю Стейсі .

Матриця Стейсі (Матриця узгодженості та визначеності) класифікує проєкти на основі двох ступенів невизначеності:

- Горизонтальна вісь: Рівень технічної невизначеності (Знаємо, ЯК робити?).
- Вертикальна вісь: Рівень невизначеності вимог (Знаємо, ЩО робити?).

Домени Невизначеності

Виділені області матриці відповідають доменам з Cynefin Framework (Теорія заплутаності):

Домен (Область)	Характеристики Невизначеності	Типовий Метод Розробки
Простий (Simple)	Вимоги та Технології високо визначені (відомо, ЩО і ЯК).	Будь-який метод, включаючи Waterfall (Водоспад).
Складний (Complicated)	Вимоги відомі, Технології зрозумілі, але багато змінних.	Потрібні професіонали для вирішення.
Комплексний (Complex)	Причинно-наслідкові зв'язки незрозумілі (невідомо, ЩО або ЯК робити).	Agile (потрібні гіпотези, експерименти, ітерації).
Хаотичний (Chaotic)	Невизначеність у «кубі». Багато факторів, причинно-наслідкові зв'язки здебільшого невідомі.	Agile (потрібні швидкі дії для стабілізації).

Висновки щодо застосовності:

- Якщо ступінь визначеності дуже високий (ви можете чітко описати завдання, знаєте, як його реалізувати, і можете зафіксувати це в документах), підійдуть будь-які методи, включно з Каскадною моделлю;
- В інших випадках (коли присутня значна невизначеність щодо вимог або технологій) краще використовувати Agile та її практики, що ґрунтуються на ітераціях та адаптації.

Питання для самоперевірки, тести для самоконтролю

1. Як перекладається термін "Agile" і яке визначення закріпилося за цією групою методологій в ІТ-лексичі?
2. Назвіть чотири центральні цінності (ідеї) Agile. Поясніть, що означає пріоритет лівої частини твердження "над" (over) правою.
3. Чому, згідно з принципами Agile, вітається зміна вимог навіть на пізніх стадіях розробки?
4. Який показник, згідно з 12 принципами Agile, є основним показником прогресу?
5. Сформулюйте Принцип №10 Agile, пов'язаний із "Простотою".
6. Яку функцію виконує Матриця Стейсі? Які дві осі вона використовує для оцінки проєкту?
7. Визначте, в якому домені Матриці Стейсі (Simple, Complicated, Complex, Chaotic) Каскадна модель (Waterfall) може бути використана ефективно.
8. Поясніть, чому Agile є найкращим вибором для проєктів, що потрапляють у домен "Комплексний (Complex)".
9. Який з 12 принципів Agile рекомендує використовувати постійний темп роботи і чому?
10. Як Agile-команди забезпечують постійне покращення своєї роботи (Принцип №12)?

Обов'язкові та додаткові задачі, приклади

Обов'язкова задача 1: Застосування цінностей Agile

Команда розробників ПЗ стикається з такою проблемою: замовник постійно вимагає детальних і вичерпних звітів про хід роботи, хоча команда щотижня надає робочу версію продукту з новим функціоналом.

1. Яку з чотирьох центральних ідей Agile порушує замовник, вимагаючи вичерпної документації?

2. Як має відреагувати команда, керуючись філософією Agile?

Відповідь:

1. Порушується ідея: "Працююче ПЗ важливіше, ніж вичерпна документація".

2. Команда має наголосити на тому, що працюючий продукт є основним показником прогресу (Принцип №7). Потрібно пояснити замовнику, що надмірне створення документації знижує швидкість розробки, і запропонувати компроміс: мінімально необхідну документацію або демонстрацію функціоналу як основний звіт.

Обов'язкова задача 2: Аналіз невизначеності (Матриця Стейсі)

Проаналізуйте наступні проекти та визначте їхнє місце в Матриці Стейсі, а також рекомендований підхід до управління:

1. Проект А: Створення стандартного корпоративного вебсайту за чітким ТЗ. Технології добре відомі (типовий стек LAMP/MERN). Вимоги змінюватися не будуть.

2. Проект Б: Розробка системи діагностики на основі штучного інтелекту для раннього виявлення рідкісних хвороб. Невідомо, чи зможе обраний алгоритм забезпечити необхідну точність (технологічна невизначеність висока). Вимоги до кінцевої точності також можуть змінитися в процесі дослідження.

Відповідь:

1. Проект А (Стандартний вебсайт):

- Невизначеність вимог: Низька (вимоги чіткі).
- Технічна невизначеність: Низька (технології відомі).
- Домен: Простий (Simple).
- Рекомендований підхід: Waterfall або будь-який інший, але Agile також підійде.

2. Проект Б (Система ІІІ):

- Невизначеність вимог: Висока.

- Технічна невизначеність: Висока.
- Домен: Комплексний (Complex) або Хаотичний (Chaotic).
- Рекомендований підхід: Agile (Ітеративний, Спіральний, Scrum) — вимагає постійного тестування гіпотез та швидкої адаптації.

Довідково-інформаційні дані для розв'язування задач (таблиці, схеми тощо)

1. Зведення Чотирьох Цінностей Agile (Agile Manifesto)

Пріоритет (Ліва частина)	Над (over)	Менш Пріоритетно (Права частина)
Люди та взаємодія	over	Процеси та інструменти
Працююче ПЗ	over	Вичерпна документація
Співпраця із замовником	over	Узгодження умов контракту
Готовність до змін	over	Слідування первісному плану

2. Матриця Стейсі (Stacey Matrix)

Домен (За Synefin)	Невизначеність Вимог (ЩО?)	Невизначеність Технологій (ЯК?)	Рекомендований Підхід
Простий	Близька до узгодження	Близька до визначеності	Waterfall, Прогнозовані
Складний	Близька до узгодження	Далека від визначеності	Експертні методи
Комплексний	Далека від узгодження	Далека від визначеності	Agile, Ітеративні, Scrum
Хаотичний	Дуже далека	Дуже далека	Agile, Адаптивні

3. Акценти 12 Принципів Agile

Фокус	Принципи	Суть
Цінність для Клієнта	1, 3, 10	Регулярна поставка, вітання змін, простота (не робити зайвого).
Ритм та Швидкість	2, 8	Частий випуск продукту (що два тижні), підтримка постійного темпу роботи.
Комунікація та Довіра	4, 5, 6	Щоденна співпраця бізнесу та розробки, особисте спілкування, довіра вмотивованим командам.
Якість та Вдосконалення	7, 9, 11, 12	Працюючий продукт як показник прогресу, технічна досконалість, самоорганізація, постійна адаптація.

Бібліографічний список

1. Beck, Kent et al. Manifesto for Agile Software Development. 2001. [Електронний ресурс]. (Оригінальний документ, що містить чотири основні цінності та дванадцять принципів Agile, на яких базуються всі гнучкі методології).
2. Highsmith, Jim. Agile Software Development Ecosystems. Addison-Wesley Professional, 2002. (Одна з перших книг, що пояснює суть гнучких методологій та їхній контекст у розробці ПЗ).
3. Cohn, Mike. Agile Estimating and Planning. Prentice Hall, 2006. (Джерело, що описує практичні методи планування та оцінки, які відповідають принципам Agile).
4. Stacey, Ralph D. Strategic Management and Organisational Dynamics: The Challenge of Complexity. Financial Times Prentice Hall, 2007. (Книга, що описує Матрицю Стейсі, яка використовується для визначення приналежності проєкту до доменів складності та застосовності гнучких підходів).
5. Kurtz, Cynthia F., and Snowden, David J. The New Dynamics of Strategy: Sense-Making in a Complex and Complicated World. IBM Systems Journal, 2003, Vol. 42, No. 3, pp. 462-483. (Публікація, що описує Cynefin Framework, на основі якого сформована логіка застосовності Agile у Матриці Стейсі).

РОЗДІЛ 2. КЛЮЧОВІ AGILE-МЕТОДОЛОГІЇ: SCRUM ТА KANBAN

(Коваль В.В., кандидат військових наук, старший науковий співробітник, генерал-майор Збройних сил України)

Тема 1. KANBAN

Походження Kanban

Систему Kanban (з яп. «сигнальна картка», «бирка», «знак») винайшли в Японії. Вона сформувалася до 1950-х років на виробництві автомобілів «Тойота» (Toyota) під керівництвом президента компанії Таїті Оно (Taiichi Ohno).

Основною метою було:

- випускати автомобілі з максимальною швидкістю;
- мінімізувати складські запаси (принцип Just-in-Time, "точно вчасно").

Таїті Оно надихнувся роботою американських супермаркетів, де покупець сам вибирав необхідні товари, витягуючи їх зі стелажів. Роль «супермаркету» на Toyota виконував склад.

Японці запровадили цей «витягувальний» (pull) принцип:

- деталі для збирання почали постачати в цехи невеликими партіями в міру необхідності;
- кожен цех замовляв іншому рівно стільки деталей, скільки потрібно для збірки на найближчий період.

Суть підходу: Будь-яка запчастина мала виготовлятися не раніше і не пізніше, ніж у ній виникне необхідність. Це дозволило відмовитися від великих складських запасів, що значно знизило поточні витрати.

Використання Карток («Бірок») на Toyota

Картки (канбани) кріпилися до тари з деталями. На них вказувалася інформація:

- номер та кількість деталей;
- відділ, що відправляє, і відділ-отримувач.

1. Працівник монтажної лінії (нижній потік) забирав деталі. Картка знімалася і разом із порожнім ящиком передавалася на склад.

2. На складі працівник готував нову тару і кріпив на неї виробничий «канбан» (інформація про вироблені запчастини).

3. Виробничий «канбан» замінювався на «канбан» із запитом і відправлявся на виробничу лінію запчастин (верхній потік).

4. Це призводило до виробництва саме тієї кількості деталей, що була вказана в картці.

Таким чином, візуалізація та обмеження кількості матеріалів у роботі стали основними принципами системи, яка згодом проникла у сферу ІТ.

Kanban — це метод поліпшення процесів розробки шляхом візуалізації та активної роботи над незавершеними завданнями. Це частина філософії Agile, але не є повноцінною методологією, а радше гнучким інструментом, який можна використовувати з будь-яким підходом до розробки.

Kanban робить роботу над проєктом наочнішою, дозволяє відстежувати виконання завдань і контролювати навантаження фахівців.

Цінності Kanban

Kanban базується на дев'яти основних цінностях:

1. Прозорість: Візуалізація задач на дошці та відкритий обмін інформацією. Створює безпечне середовище;

2. Баланс: Обмеження обсягу незавершеної роботи (WIP) для запобігання перевантаженню та встановлення сталого, передбачуваного ритму роботи;

3. Співпраця: Спільна робота членів команди та співробітництво із замовником для постійного вдосконалення;

4. Клієнтоорієнтованість: Своєчасне завершення задач і плавне нарощування цінності для клієнта;

5. Потік: Розуміння роботи як потоку створення цінності (послідовності етапів). Управління потоком — пошук та усунення слабких місць;

6. Лідерство: Заохочення ініціативи на всіх рівнях. Лідер створює гарні системи роботи та сильну культуру;

7. Розуміння: Розуміння поточних ролей, процесів і причин, чому все влаштовано саме так;

8. Згода: Уміння домовлятися, навіть за наявності розбіжностей, на шляху до спільної мети;

9. Повага: Повага до колег та спільної роботи для комфортної та продуктивної діяльності.

Шість основоположних принципів Kanban діляться на дві групи.

Принципи Управління Змінами

1. Почніть з того, що є зараз. Не потрібно різко змінювати структуру, процеси та ролі. Kanban передбачає послідовні та плавні еволюційні поліпшення. Це знижує опір команди.

2. Домовтеся про еволюційний розвиток. Команда має прийняти, що постійні, невеликі, безпечні експерименти — це спосіб покращити процес. Еволюційний підхід шукає найближчу "точку незадоволеності", усуває її, і закріплює поліпшення.

3. Заохочуйте розвиток лідерства на всіх рівнях. Приймаються ініціативи кожного члена команди, важливою є толерантність до помилок.

Принципи Надання Сервісів

Розглядаючи організацію як систему взаємозалежних сервісів, важливо:

1. З'ясуйте потреби та очікування замовника. Зосередьтеся на їх задоволенні.

2. Управляйте роботою, а не людьми. Дайте людям можливість самоорганізуватися навколо роботи.

3. Розвивайте правила, щоб поліпшити показники та задоволеність замовника. Регулярно переглядайте систему та правила для поліпшення практик Kanban.

Практики Kanban

1. Візуалізуйте

Процес роботи візуалізується на Kanban-дошці, розділеній на сектори (стовпці). Завдання записуються на картках.

- Дошка: Може бути фізичною (стіна зі стікерами) або віртуальною (наприклад, у Jira).
- Стовпці: Відображають основні етапи роботи в потоці (наприклад, «В очікуванні» / «Скласти», «В роботі» / «Обрано», «Виконано» / «Готово»).
- Мета: Візуалізація добре показує вузькі місця, черги, відхилення та втрати, що негативно впливають на продуктивність.

Робота з дошкою:

- «В очікуванні» («Скласти»): Задачі, що надійшли від замовника. Призначається пріоритет.
- Точка прийняття зобов'язань: Замовник підтвердив бажання отримати елемент, команда підтвердила можливість виконання.
- «В роботі» («Вибрано»): Задача, над якою працює команда.
- Точка поставки: Замовник приймає виконаний робочий елемент.

2. Обмежуйте кількість незавершеної роботи (WIP)

WIP (Work In Progress) — це кількість задач, що перебувають у роботі між двома точками: «зобов'язання» і «поставка».

- Обмеження: Кількість задач-листків у роботі на одній стадії має бути обмежена (встановлюється виходячи з можливостей команди).
- Ефект: Стабільно виконувати невелику кількість задач краще, ніж почати багато і не закінчити жодної. Обмеження WIP стимулює не кидати почате і виявляє проблеми, оскільки будь-яка затримка одразу помітна всій команді.

- Стимул до співпраці: Якщо розробник заблокований, проблема стає проблемою команди, що стимулює співпрацю для швидшого завершення задачі.

3. Управляйте потоком роботи

Мета: Зробити потік роботи рівномірним і передбачуваним. Фокус на результаті, а не на утилізації ресурсів.

- Kanban допомагає бачити, де саме виникають проблеми, щоб оперативно їх усунути.

- Розширена формулювання: «Управляйте потоком роботи, а не людьми. Дайте людям самим організуватися навколо неї».

4. Використовуйте явні правила

Мета: Зробити прозорими домовленості та знизити залежність від людського фактора, тобто визначити правила гри.

- Приклади: Підрахунок WIP; Критерії готовності (Definition of Done) для переміщення роботи на наступний етап (наприклад, "задача повинна пройти навантажувальне тестування, щоб перейти в 'Приймання'").

5. Вводьте цикли зворотного зв'язку (Каденції)

Каденція (термін з музики/ритму) — це регулярні зустрічі, які є петлями зворотного зв'язку та призводять до еволюційних змін.

Сім способів отримання зворотного зв'язку (Каденції):

1. Канбан-мітинг (Щоденний): Обговорення статусу задач, фокус на заблокованих задачах. Схожий на стендап.

2. Зустріч з наповнення черги (1-2 тижні): Планування, які продукти/задачі будуть реалізовуватися (прийняття зобов'язань).

3. Зустріч з планування поставки (за частотою поставки): Обговорення продукту, який буде впроваджуватися (повернення виконаних зобов'язань).

4. Зустріч з огляду сервісу поставки (1-2 тижні): Аналіз та вдосконалення ефективності команди (якість, своєчасність).

5. Операційна зустріч (Місячний): Обговорення якості взаємодії пов'язаних сервісів (наприклад, з техпідтримкою).

6. Зустріч з огляду ризиків (Місячний): Вивчення ризиків, пов'язаних з блокуваннями, та їх мінімізація.

7. Зустріч з огляду стратегії (Квартальний): Аналіз метрик, обговорення змін у стратегії/концепції розробки.

6. Покращуйтесь та еволюціонуйте

Мета: Керувати змінами команди.

- Підхід: Замість грандіозного плану, шукаються джерела незадоволеності (в роботі, результатах, обслуговуванні клієнтів) і формулюються гіпотези для їх усунення.

- Експерименти: Повинні бути невеликими та безпечними, але мати реальний ефект. Результат закріплюється, або робиться крок назад.

Класи Обслуговування

Kanban використовує класи обслуговування для управління пріоритетом робіт, виходячи з вартості затримки (недоотриманий прибуток/витрати через несвоєчасне надання послуг).

1. Прискорений клас (Expedite): Невідкладна допомога. Немає часу на відкладання. Приклад: Блокуючі дефекти.

2. Клас з фіксованою датою (Fixed Date): Вартість затримки різко зростає після певної дати. Приклад: Законодавчий проєкт з фіксованою датою набрання чинності.

3. Стандартний клас (Standard): Вартість затримки зростає пропорційно часу.

4. Нематеріальний клас (Intangible): Явної та швидкої прибутку не приносить, вартість затримки зростає повільно. Приклад: Регулярне прибирання коду (рефакторинг).

Питання для самоперевірки, тести для самоконтролю

1. Як перекладається слово «Kanban» з японської? Яку роль відігравали ці картки на виробництві Toyota?
2. Назвіть двох засновників системи Kanban на виробництві Toyota.
3. Чому Kanban вважається методом та інструментом, а не повноцінною методологією розробки?
4. Яка з цінностей Kanban (за номером) заявляє, що обсяг незавершеної роботи необхідно обмежувати?
5. Сформулюйте головну ідею Принципу управління змінами "Почніть з того, що є зараз".
6. Який принцип надання сервісів Kanban рекомендує "Управляйте роботою, а не людьми"?
7. Що таке WIP (Work In Progress) у контексті Kanban? Між якими двома точками на дошці знаходиться WIP?
8. Яка з практик Kanban допомагає усунути вузькі місця, черги та втрати?
9. Що таке Критерії готовності (Definition of Done) і до якої практики Kanban вони належать?
10. Яка Каденція Kanban (зустріч) відбувається щоденно та фокусується на обговоренні статусу задач та блокувань?
11. Який Клас обслуговування Kanban слід застосувати до помилки, що повністю блокує використання програми клієнтом? Чому?

Обов'язкові та додаткові задачі, приклади

Обов'язкова задача 1: Аналіз WIP

Команда розробників ПЗ встановила ліміт WIP 3 для етапу «Кодування». На дошці є 5 задач, які очікують на кодування, але 3 задачі вже перебувають у цьому стовпці.

1. Чи може розробник взяти в роботу нову задачу зі стовпця «В очікуванні»?

2. Яку дію має зробити команда, керуючись принципами Kanban, щоб якнайшвидше просунути роботу?

Відповідь:

1. Ні, не може. Ліміт WIP для стовпця «Кодування» вже вичерпано (3 з 3).
2. Команда має співпрацювати (Цінність 3), щоб допомогти розробникам, які працюють над поточними 3 задачами, завершити їх. Це забезпечує Баланс (Цінність 2) і дозволяє Потоку (Цінність 5) рухатися вперед, не створюючи "заторів".

Обов'язкова задача 2: Визначення класу обслуговування

Визначте відповідний Клас обслуговування Kanban для наступних типів робіт:

1. виправлення критичної вразливості системи безпеки, яка призвела до витоку даних.
2. Розробка функції для обов'язкового щорічного звіту, який має бути поданий в державну установу 31 грудня.
3. Поточна розробка нової функціональності, яка не має фіксованого терміну, але принесе прибуток після впровадження.
4. Наведення ладу в старих внутрішніх бібліотеках коду (рефакторинг), що не приносить негайної вигоди клієнту.

Відповідь:

1. Прискорений клас (Expedite) — невідкладна реанімація.
2. Клас з фіксованою датою (Fixed Date) — вартість затримки різко зросте після 31 грудня.
3. Стандартний клас (Standard) — вартість затримки зростає пропорційно часу (чим довше робимо, тим пізніше отримуємо прибуток).
4. Нематеріальний клас (Intangible) — явна цінність зростає повільно, але відкладання призведе до великих витрат у майбутньому.

Довідково-інформаційні дані для розв'язування задач (таблиці, схеми тощо)

1. Карта RIK (Rational Unified Process) та Kanban

Колонка (Етап)	Потік (Kanban-термін)	Точка на Дощці
Backlog	Черга (Очікування)	До Точки прийняття зобов'язань
In Progress	Робота (WIP)	Між Точкою прийняття зобов'язань та Точкою поставки
Done	Завершено	Після Точки поставки

2. Принципи Kanban у Співвідношенні

Група Принципів	Фокус	Принципи
Управління Змінами	Мінімізація опору команди	1. Почніть з того, що є зараз. 2. Домовтеся про еволюційний розвиток. 3. Заохочуйте розвиток лідерства на всіх рівнях.
Надання Сервісів	Клієнтоорієнтованість та Потік	1. З'ясуйте потреби та очікування замовника. 2. Управляйте роботою, а не людьми. 3. Розвивайте правила для поліпшення сервісу.

Бібліографічний список

1. Ohno, Taiichi. Toyota Production System: Beyond Large-Scale Production. Productivity Press, 1988. (Оригінальна робота Таїті Оно, яка описує систему виробництва Toyota, включаючи принципи Just-in-Time та Kanban).

2. Anderson, David J. Kanban: Successful Evolutionary Change for Your Technology Business. Blue Hole Press, 2010. (Одна з основних книг, що адаптувала Kanban для ІТ-індустрії та описала його шість принципів та практики).

3. Ladas, Jim. Scrumban: Essays on Kanban Systems for Lean Software Development. Modus Cooperandi, 2009. (Джерело, що розглядає інтеграцію Kanban з іншими гнучкими методологіями, зокрема Scrum).

4. Hines, Peter, and Rich, Nick. The Seven Value Streams. Journal of Business Process Reengineering and Management, 1997. (Роботи з Lean-менеджменту, які заклали основи концепції потоку та цінності в Kanban).

5. Kniberg, Henrik, and Skarin, Mattias. Kanban and Scrum: Making the Most of Both. Lulu.com, 2010. (Посібник, що порівнює та показує, як ефективно використовувати Kanban та Scrum разом).

Тема 2. Ролі в KANBAN та СТАТІК (S.T.A.T.I.K.)

Ролі в Kanban

У методі Kanban, оскільки він фокусується на управлінні потоком роботи, а не людьми, виділяють лише три ключові ролі, які відповідають за функціонування сервісу.

1. Менеджер сервісу запитів (Service Request Manager, SRM):

- Також відомий як Менеджер продукту або Власник продукту.
- Відповідальність: Вивчення потреб та очікувань замовників. Сприяє вибору та пріоритетизації робочих елементів на зборах з поповнення черги (Kanban Replenishment Meeting).

2. Менеджер сервісу поставки (Service Delivery Manager, SDM):

- Також відомий як Менеджер потоку або Менеджер поставок.
- Відповідальність: Управління потоком роботи (Work Flow), завдяки якому вибрані елементи постачаються замовникам. Проводить Kanban-мітинги та збори планування поставки.

3. Команда розробки (Development Team):

- Відповідає за виконання технічних завдань і створення цінності.

S.T.A.T.I.K. (Системний Підхід до Впровадження Kanban)

S.T.A.T.I.K. розшифровується як System Thinking Approach for Introducing Kanban («Системний підхід до впровадження Kanban»).

Це структурований набір кроків, які необхідно послідовно виконати для успішного впровадження Kanban-методу. Він допомагає подивитися на робочу систему як на єдине ціле, виявити її недоліки та визначити шляхи еволюційного покращення.

Вісім Ключових Кроків S.T.A.T.I.K.

1. Визначити споживачів сервісу. (Клієнти, для яких ми створюємо цінність).
2. Виявити джерела незадоволеності. (Внутрішні та зовнішні проблеми, які гальмують роботу).
3. Проаналізувати попит. (Які типи задач та з якою частотою до нас надходять).
4. Проаналізувати можливості. (Яку кількість роботи та з якою швидкістю ми здатні виконувати).
5. Побудувати модель робочого процесу. (Візуалізувати поточний процес "як є").
6. Визначити класи сервісу. (Правила пріоритезації на основі вартості затримки).
7. Створити проєкт візуалізації Kanban-системи. (Начерк Kanban-дошки з лімітами WIP).
8. Представити систему. (Презентувати нову систему команді та замовникам).

Деталізація Кроків S.T.A.T.I.K.

0/1. Сервіс та Менеджер

Тут під «сервісом» розуміється ваша діяльність, яку ви надаєте клієнтам (замовникам) як послугу.

- Допоміжні питання: Який саме сервіс ви надаєте? Хто є Менеджером сервісу поставки (SDM), який відповідає за те, щоби послуга була надана вчасно та в належній якості?

1/2. Джерела Незадоволеності

Важливо відобразити внутрішні та зовнішні (з точки зору клієнта) проблеми.

- Внутрішні джерела незадоволеності (Команда): Те, чим незадоволена сама команда сервісу.
- Приклади: Перевантаження, переробки, часте перемикання між задачами, часті простої, блоки, конфлікти з колегами.
- Зовнішні джерела незадоволеності (Замовники/Суміжні сервіси): Те, чим незадоволені споживачі сервісу.
- Приклади: Занадто тривалі терміни, порушення обіцянок щодо термінів, непрозорість робочих процесів, конфлікти із виконавцями.

2/3. Аналіз Запитів (Попиту)

Збір інформації про типи робочих елементів, що надходять, їхні джерела, характер та очікування клієнта.

Для кожного типу робочого елемента збирається інформація:

Параметр	Опис	Приклад
Джерело запиту	Звідки приходять запити на роботу.	Користувачка історія, технічний борг, інцидент, договір.
Місце призначення	Куди поставляються результати роботи сервісу.	Повертаються замовнику, передаються в маркетинг, іншому сервісу.
Частота виникнення	Кількість запитів на одиницю часу.	"Два рази на тиждень", "щоденно".
Характер запиту	Будь-які закономірності чи особливості (часто містить приховану інформацію про ризики).	"Прибігає в останню хвилину", "посилається на керівника", "спокійний, можна домовитися".
Очікування клієнта	Уявлення замовника про час поставки елемента.	"Через тиждень", "сьогодні", "по готовності".

3. Побудова Моделі Робочого Процесу

Необхідно візуально зобразити робочий процес (потік) для декількох ключових типів робочих елементів. Це допомагає виявити спільні риси, відмінності та неупорядковані або паралельні дії.

- Методи візуалізації: Блок-схеми, послідовність етапів на стікерах тощо.

4. Визначення Класів Обслуговування

Класи обслуговування визначаються для кожного типу робочого елемента на основі вартості затримки (Cost of Delay).

Питання: «Яку суму грошей ми втратимо, якщо не почнемо робити цю роботу прямо зараз?»

Архетип Класу	Втрата при Затримці	Рекомендована Дія
1. Прискорений	Втрата буде величезною і негайною.	Необхідно розпочати в терміновому порядку (Expedite).
2. З фіксованою датою	Нічого не втратимо до певного дедлайну, після чого втрата буде різкою.	Роботу потрібно зробити до визначеної дати (Fixed Date).
3. Стандартний	Будемо втрачати поступово, якщо не зробимо роботу.	Потік роботи має бути рівномірним (Standard).
4. Нематеріальний	Дуже довго нічого не будемо втрачати, але накопичення призведе до зупинки бізнесу.	Завдання, що покращують, автоматизують, знижують технічний борг (Intangible).

5. Каденції з Поповнення та Поставки

Визначення частоти вхідної та вихідної каденцій для кожного типу робочого елемента.

- Вхідна каденція (Поповнення): Як часто ми беремо елемент у роботу? (Наприклад, раз на тиждень, або по мірі надходження для прискореного класу).

- Вихідна каденція (Поставка): Як часто ми плануємо поставку? (Календарна дата/час).

6. Візуалізація Kanban-системи

Створення начерку (прототипу) майбутньої Kanban-дошки . Визначення основних елементів:

- Колонки (етапи потоку).
- Ліміти WIP (кількість незавершеної роботи).
- Дорожки (Swimlanes): Виділення окремих горизонтальних ліній для різних Класів обслуговування або типів задач.

- Колір: Використання кольору для карток для швидкої ідентифікації.

Питання для самоперевірки, тести для самоконтролю

1. Назвіть три основні ролі, які виділяють у Kanban.
2. Яка роль (за Kanban) відповідає за пріоритетизацію робочих елементів та проводить збори з поповнення черги?
3. Яка роль відповідає за управління потоком роботи та його ефективну доставку до замовника?
4. Як розшифровується аббревіатура S.T.A.T.I.K. і яка його основна мета?
5. Назвіть перші чотири ключові кроки S.T.A.T.I.K.
6. Поясніть, у чому різниця між внутрішніми та зовнішніми джерелами незадоволеності (крок S.T.A.T.I.K. 1/2). Наведіть по одному прикладу.
7. Які дві ключові точки на Kanban-дошці (точки прийняття зобов'язань та поставки) є важливими для визначення WIP (Work In Progress)?
8. Яке основне питання необхідно поставити для визначення відповідного Класу обслуговування (крок S.T.A.T.I.K. 4)?
9. Опишіть, чим відрізняється Прискорений клас від Класу з фіксованою датою за критерієм вартості затримки.
10. Що таке дорожки (Swimlanes) у візуалізації Kanban-системи (крок S.T.A.T.I.K. 6) і для чого вони можуть використовуватися?

Обов'язкові та додаткові задачі, приклади

Обов'язкова задача 1: Аналіз джерел незадоволеності

Команда розробки стикається з наступними проблемами:

1. Розробники часто змушені перемикатися між трьома різними проєктами, що знижує їхню концентрацію.
2. Замовник скаржиться, що не знає, на якому етапі знаходиться його запит, і просить постійних оновлень.

Визначте, до якого типу джерела незадоволеності (внутрішнє чи зовнішнє) належить кожна проблема, та запропонуйте, як Kanban може це вирішити.

Відповідь:

1. Проблема 1 (Перемикання між проєктами): Внутрішнє джерело незадоволеності (перевантаження, часте перемикання).

- Рішення Kanban: Введення лімітів WIP (практика 2) для обмеження кількості одночасних проєктів/задач, що виконуються.

2. Проблема 2 (Непрозорість): Зовнішнє джерело незадоволеності (непрозорість робочих процесів).

- Рішення Kanban: Візуалізація (практика 1) на Kanban-дошці, що забезпечує Прозорість (Цінність 1) для всіх учасників процесу.

Обов'язкова задача 2: Проєктування Kanban-дошки (Крок S.T.A.T.I.K. 6)

Ви — SDM (Менеджер сервісу поставки) проєкту, який обслуговує запити від трьох джерел:

- Інциденти (Баги): Виправляються негайно (Прискорений клас).
- Нові функції: Стандартний клас.
- Технічний борг: Нематеріальний клас.

Створіть начерк Kanban-дошки, який відображатиме:

- Мінімум чотири етапи потоку (стовпці).
- Використання дорожок для класів обслуговування.
- Приклад ліміту WIP.

Відповідь (Начерк Дошки):

Етап Потоку (Колонка) →	Аналіз	Кодування (WIP ≤ 3)	Тестування	Готово
Дорожка А: Прискорений		[Карточка: Блокуючий Баг]		
Дорожка В: Нові Функції (Стандартний)	[Карточка: Функція X]	[Карточка: Функція Y]		
Дорожка С: Технічний борг (Нематеріальний)		[Карточка: Рефакторинг]		

Пояснення: Дорожка А (Прискорений) часто має найменший ліміт WIP або його відсутність, оскільки це пріоритетна "швидка допомога". Дорожки В і С конкурують за загальний ліміт WIP у стовпці "Кодування" (наприклад, № 3).

Довідково-інформаційні дані для розв'язування задач (таблиці, схеми тощо)

1. Ролі Kanban та їхні обов'язки

Роль	Призначення	Відповідальність	Ключові Каденції
SRM (Менеджер запитів)	Що робити?	Пріоритетизація, вивчення очікувань клієнтів.	Зустріч по наповненню черги.
SDM (Менеджер поставки)	Як робити? (Потік)	Управління потоком, контроль WIP та часу циклу.	Kanban-мітинг, Планування поставки.
Команда розробки	Виконання	Створення цінності, самоорганізація навколо роботи.	Kanban-мітинг.

2. Вісім кроків S.T.A.T.I.K. та їхнє призначення

Крок	Мета	Kanban-Принцип, що підтримується
1/2. Джерела Незадоволеності	Виявлення точки незадоволеності (для еволюції).	Еволюційний розвиток (Принцип 2).
2/3. Аналіз Запитів (Попиту)	Розуміння попиту та очікувань клієнта.	З'ясування потреб та очікувань замовника (Принцип 4).
3. Модель Робочого Процесу	Візуалізація поточного процесу.	Почніть з того, що є зараз (Принцип 1).
4. Класи Обслуговування	Встановлення явних правил пріоритетизації та політик.	Розвивайте правила (Принцип 6).
6. Візуалізація Kanban-системи	Створення інструменту для Управління потоком.	Управляйте роботою, а не людьми (Принцип 5).

3. Класи Обслуговування Kanban

Клас Обслуговування	Характеристика вартості затримки	Політика
Прискорений (Expedite)	Критична, негайна	Обхід черги, негайне виконання.
З фіксованою датою (Fixed Date)	Різкий стрибок після дедлайну	Робота береться в обробку відповідно до терміну.
Стандартний (Standard)	Пропорційне зростання	Обробка за порядком черги (FIFO), відповідно до WIP.
Нематеріальний (Intangible)	Повільне, відкладене зростання	Робота виконується, коли є можливість, або за накопиченням.

Бібліографічний список

1. Anderson, David J. Kanban: Successful Evolutionary Change for Your Technology Business. Blue Hole Press, 2010. (Основоположна праця, що описує методологію Kanban, її принципи, а також детально вводить ролі та інструмент S.T.A.T.I.K. як системний підхід до впровадження).

2. Anderson, David J., and Carmichael, Andy. Kanban: Changing the Way You Manage Work. Blue Hole Press, 2018. (Оновлене керівництво, яке посилює фокус на сервісній парадигмі, управлінні потоком та ролях SDM і SRM).

3. Hines, Peter, and Rich, Nick. The Seven Value Streams. Journal of Business Process Reengineering and Management, 1997. (Джерела з Lean-менеджменту, які є концептуальною основою для аналізу потоку та джерел незадоволеності в S.T.A.T.I.K.).

4. Ladas, Jim. Kanban from the Inside: Understand the Kanban Method to Deepen Your Agility. Blue Hole Press, 2014. (Книга, що надає глибше розуміння цінностей Kanban і ролей, особливостей управління сервісами).

Тема 3. Гнучкі моделі управління командою: SCRUM

Scrum — це процесний фреймворк для розробки, поставки та супроводу складних продуктів. Він використовує ітеративно-інкрементальний підхід і базується на теорії емпіричного управління.

Термін "Scrum" походить із регбі й перекладається як «схватка». Це підкреслює необхідність слагодженості дій та командної роботи, чого, на думку його творців (Джеффа Сазерленда та Кена Швабера), бракувало розробникам ПЗ.

Scrum-гайд — це документ, який описує набір ролей, подій та артефактів. Жоден з цих елементів не може бути проігнорований, інакше це вже не Scrum.

- Емпіричне управління: Метод проб і помилок. Керівник діє дослідним шляхом, аналізує результат і вибудовує подальшу роботу, ґрунтуючись на цьому аналізі.

- Ітеративний підхід: Повторення однієї й тієї ж послідовності дій (замкнутий цикл). В Scrum ітерацією є спринт (тривалість 1–4 тижні).

- Інкрементальний підхід: Часткова реалізація та повільне нарощування функціональності («приріст»). Кожен виток ітерації (спринт) дає інкремент — потенційно готовий до випуску шматок продукту.

Три Принципи Scrum («Три Кити»)

В основі Scrum лежать три принципи, які дозволяють реалізовувати гнучкий підхід:

1. Прозорість (Transparency)

Полягає в тому, що всі аспекти процесу та продукту мають бути відкритими та зрозумілими для всіх учасників. Це запобігає помилкам та забезпечує можливість необхідних змін.

- Приклади: Спільна термінологія; спільні Критерії готовності (Definition of Done) для тих, хто виконує роботу і тих, хто її приймає.

2. Інспекція (Inspection)

Це оперативне відстеження всіх змін у процесі роботи, в продукті та в навколишньому середовищі (ринок, технології). Інспекція необхідна для виявлення небажаних відхилень або перешкод.

- Приклад: Щоденні збори (Daily Scrum) для виявлення блокерів та обміну інформацією.

3. Адаптація (Adaptation)

Це оперативне внесення змін у систему при виявленні небажаних відхилень або перешкод (завдяки інспекції). Адаптація забезпечує гнучкість команди та продукту.

- Приклад: Зміна планів у ході Daily Scrum, щоб швидше досягти мети спринту.

Цінності Scrum

П'ять цінностей, які слугують основою для роботи команди та підтримки «Трьох китів» Scrum:

1. Сфокусованість (Focus): Кожен учасник зосереджується на обмеженій кількості задач та на Меті Спринту. Це підвищує продуктивність і ймовірність реалізації інкременту.

2. Сміливість (Courage): Необхідна для відкритого обговорення складнощів, перешкод, власних помилок, а також для пропонування нових творчих ідей.

3. Відкритість (Openness): Відкритість до проблемних моментів у продукті чи процесі, до нових рішень та до нової інформації.

4. Зобов'язання (Commitment): Кожен член команди свідомо приймає та бере на себе особисту відповідальність за досягнення Меті Спринту та виконання задач.

5. Повага (Respect): Повага до членів команди, до продукту, до замовника та кінцевого користувача. Є необхідною умовою для продуктивної роботи.

Ролі в Scrum

Scrum-команда є самоорганізованою та крос-функціональною і складається з трьох ключових ролей:

1. Власник Продукту (Product Owner, PO)

Власник продукту — це лідер, відповідальний за максимізацію цінності продуктів, які створює Scrum-команда. Він є єдиним голосом клієнта та бізнесу.

- Формує Візію продукту: Розуміє цільову аудиторію, її проблеми, потреби та бажання.
- Керує Product Backlog: Це єдиний впорядкований перелік усіх ідей та побажань. PO описує задачі, визначає їхній пріоритет та забезпечує прозорість цього беклогу.
- Максимізує цінність: Визначає, яку цінність буде створено в поточному спринті, а яку відкладено.
- Залучає стейкхолдерів: Запрошує замовників та зацікавлених осіб для участі в Оглядах Спринту (Sprint Review) та управляє їхніми очікуваннями.

2. Scrum-Майстер (Scrum Master, SM)

Scrum-майстер — це лідер-слуга (servant-leader), який відповідає за розуміння та застосування Scrum як всередині команди, так і за її межами.

- Коуч і фасилітатор: Допомагає команді самоорганізовуватися, залишатися продуктивною та ефективно проводити зустрічі.
- Усуває перешкоди (блокери): Захищає команду від внутрішніх та зовнішніх "подразників".
- Консультує: Налагоджує процеси, допомагає команді, Власнику продукту та організації в цілому вибудовувати Scrum.
- Не управляє командою: SM не розподіляє обов'язки і не вказує команді, як створювати інкремент, але контролює дотримання принципів Scrum.

3. Scrum-Команда (Команда Розробки)

Команда розробки — це невелика група крос-функціональних професіоналів, які самоорганізуються для створення інкременту продукту.

- Крос-функціональність: Команда має всі необхідні знання та вміння для створення кожного інкременту.

- Самоорганізація: Команда сама вирішує, як вона виконуватиме роботу. Ніхто (навіть Scrum-майстер) не може вказувати їй, як створювати інкременти.
- Відповідальність: Члени команди відповідають за прогнозування обсягу роботи в спринті, виконання задач та якість своєї роботи.
- Розмір: Оптимально 7 \pm 2 людини (зазвичай 3–9 розробників), щоб залишатися керованою, але здатною виконати значний обсяг роботи.

Питання для самоперевірки, тести для самоконтролю

1. Як Scrum-фреймворк базується на теорії емпіричного управління?
2. Назвіть три основні принципи (три "кити") Scrum. Поясніть, як один із них забезпечує можливість оперативного внесення змін.
3. Сформулюйте п'ять основних цінностей Scrum. Яка з них підкреслює необхідність відкрито говорити про проблеми та помилки?
4. Що таке інкрементальний підхід у Scrum? Коли команда створює новий інкремент?
5. Хто є єдиною особою, відповідальною за максимізацію цінності продукту та управління Product Backlog?
6. Чим відрізняється роль Scrum-майстра від традиційного менеджера проєкту? Яку основну функцію він виконує щодо команди?
7. Чому Scrum вимагає, щоб команда розробки була крос-функціональною?
8. Хто в Scrum-команді приймає рішення щодо того, як саме буде створюватися інкремент продукту?
9. Скільки часу (мінімум і максимум) може тривати спринт згідно зі Scrum-гайдом?
10. Як цінність Зобов'язання (Commitment) пов'язана з плануванням Спринту?

Обов'язкові та додаткові задачі, приклади

Обов'язкова задача 1: Аналіз порушення принципів Scrum

Команда працює над створенням мобільного додатку. Власник продукту (PO) регулярно додає нові, невеликі, але важливі задачі в Sprint Backlog після того, як Спринт розпочався, пояснюючи це тим, що "задачі дуже важливі для бізнесу".

1. Який основний принцип Scrum (з "трьох китів") порушується в цій ситуації?
2. Яка цінність Scrum порушується?
3. Як Scrum-майстер має вирішити цю ситуацію, керуючись своєю роллю?

Відповідь:

1. Порушується принцип Адаптації/Прозорості. Різке додавання задач без належної інспекції та узгодження підриває цілісність Спринту. Якщо PO постійно змінює план, команда не може адаптуватися планово, а змушена реагувати, що порушує фокусування.

2. Порушується цінність Сфокусованості. Команда відволікається від Мети Спринту та не може сконцентруватися на завершенні початкового набору задач. Також порушується Зобов'язання команди, оскільки її план змінюється.

3. Scrum-майстер має виступити як коуч та захисник команди, нагадавши PO та команді про важливість Мети Спринту та необхідність захищати Спринт. Він має пояснити PO, що нові важливі задачі повинні бути додані до Product Backlog і обговорені на наступному Плануванні Спринту (Sprint Planning).

Обов'язкова задача 2: Оцінка розміру команди

Команда розробки складається з 12 досвідчених фахівців. Вони успішно доставляють інкременти, але Scrum-майстер помічає, що

комунікація займає занадто багато часу, а на Daily Scrum деякі розробники не встигають висловитися.

1. Чи відповідає розмір цієї команди рекомендаціям Scrum-гайду? Якщо ні, то чому?

2. Яку цінність Scrum порушує занадто великий розмір команди?

3. Що може зробити команда, щоб покращити цю ситуацію (окрім поділу)?

Відповідь:

1. Ні, не відповідає. Scrum-гайд рекомендує розмір команди (розробки) 7 \pm 2 людини (зазвичай 3–9). Команда у 12 осіб є занадто великою для ефективної самоорганізації та комунікації.

2. Занадто великий розмір порушує Сфокусованість (важко синхронізуватись) та Прозорість (складно ефективно обмінюватись інформацією).

3. Щоб покращити ситуацію, команда може:

- Поділитися на дві менші Scrum-команди (оптимальне рішення).
- Встановити чіткіший часовий ліміт на Daily Scrum та використовувати короткий формат обговорення блокерів.
- Посилити особисте спілкування між окремими парами розробників, щоб не виносити всі деталі на загальну зустріч.

Довідково-інформаційні дані для розв'язування задач (таблиці, схеми тощо)

1. Scrum-Команда та Її Фокус

Роль	Основна відповідальність	Фокус
Product Owner (PO)	ЩО робити (Максимізація цінності)	Product Backlog, Клієнти, Ринок.
Scrum Master (SM)	ЯК працювати (Дотримання правил)	Процес, Принципи Scrum, Усунення блокерів.
Команда розробки	Створення (Якість та доставка)	Sprint Backlog, Інкремент, Самоорганізація.

\

2. Принципи та Цінності Scrum

Принципи (Емпіризм)	Цінності (Людський фактор)
Прозорість	Сміливість
Інспекція	Сфокусованість
Адаптація	Зобов'язання
	Повага
	Відкритість

3. Основні Концепції Спринту

- Тривалість Спринту: Фіксована (Time-Boxed), від 1 до 4 тижнів.
- Мета Спринту: Єдина мета, яку команда зобов'язується досягти протягом Спринту.
 - Інкремент: Потенційно готовий до випуску (usable), перевірений шматок продукту, створений наприкінці Спринту.
 - Definition of Done (DoD): Спільний набір критеріїв, що визначає, коли робота вважається завершеною та відповідає якості.

Бібліографічний список

1. Schwaber, Ken, and Sutherland, Jeff. The Scrum Guide™. 2020. [Електронний ресурс]. (Офіційний, канонічний документ, що містить визначення, ролі, події та артефакти Scrum-фреймворку. Є обов'язковим для вивчення).
2. Schwaber, Ken. Agile Software Development with Scrum. Prentice Hall, 2002. (Одна з перших книг, що детально описує Scrum, його філософію та емпіричні принципи).
3. Sutherland, Jeff. Scrum: The Art of Doing Twice the Work in Half the Time. Crown Business, 2014. (Книга від співзасновника Scrum, що описує історію фреймворку, принципи та практичні поради щодо підвищення продуктивності).
4. Cohn, Mike. Succeeding with Agile: Software Development Using Scrum. Addison-Wesley Professional, 2009. (Популярний посібник для практичного застосування Scrum, що фокусується на техніках оцінки та планування).

5. Rising, Linda, and Janoff, Norman. The Scrum Master as Servant-Leader. IEEE Software, 2010. (Стаття, що детально розкриває роль Scrum-майстра та його функції як лідера-слуги, захисника команди).

Тема 4. Події та артефакти в SCRUM

У Scrum використовується п'ять видів активностей/подій (Events). Вони застосовуються для підтримки принципів Інспекції та Адаптації, а також для створення регулярності (Каденції).

Події в Scrum (Scrum Events)

1. Спринт (Sprint)

- **Контейнер:** Спринт є контейнером для всіх інших подій Scrum. Це фіксований часовий відрізок (Time-Boxed), протягом якого створюється інкремент продукту.
- **Тривалість:** Зазвичай місяць або менше. Рекомендується підтримувати постійну тривалість для полегшення планування та оцінки продуктивності.
- **Мета Спринту (Sprint Goal):** Визначає, навіщо створюється інкремент і які задачі потрібно вирішити в першу чергу. Надає команді гнучкість для коригування вмісту Спринту.
- **Ритм:** Наступний Спринт починається одразу після закінчення попереднього.

2. Планування Спринту (Sprint Planning)

- **Коли:** Перша подія Спринту.
- **Учасники:** Вся Scrum-команда (PO, SM, Команда розробки).
- **Мета:** Визначити Мету Спринту та вибрати список задач (Sprint Backlog), які необхідно виконати. Також обговорюється спосіб реалізації намічених задач.

- Хто вирішує: Власник продукту представляє пріоритетні задачі з Product Backlog, а Команда розробки сама визначає кількість роботи, яку вона може виконати, ґрунтуючись на своїй продуктивності.

- Обмеження за часом (Time-box): Рекомендується не більше 8 годин для місячного Спринту (пропорційно менше для коротших Спринтів, наприклад, 4 години для двотижневого).

3. Щоденний Scrum (Daily Scrum, Стендап)

- Коли: Щоденно (5 днів на тиждень), в один і той самий час та в одному і тому ж місці.

- Учасники: Команда розробки (обов'язково). РО та SM можуть бути присутніми.

- Тривалість: Жорстко обмежена — 15 хвилин.

- Мета: Інспектувати прогрес у напрямку Мети Спринту та створити план роботи на найближчі 24 години.

- Фокус: Синхронізація дій, виявлення блокерів (перешкод). Пошук рішень блокерів виноситься за рамки цієї 15-хвилинної зустрічі.

- Принцип: Ключова подія для Інспекції та Адаптації на мікрорівні.

4. Огляд Спринту (Sprint Review)

- Коли: В кінці Спринту.

- Учасники: Scrum-команда, Зацікавлені особи (Стейкхолдери) та Замовники.

- Мета: Інспекція Інкременту та Адаптація Product Backlog. Команда демонструє стейкхолдерам результати Спринту.

- Результат: Переглянутий та, за потреби, скоригований Product Backlog на основі зворотного зв'язку від зацікавлених осіб.

- Обмеження за часом (Time-box): Рекомендується не більше 4 годин для місячного Спринту.

5. Ретроспектива Спринту (Sprint Retrospective)

- Коли: Відбувається після Огляду Спринту та перед Плануванням наступного Спринту.
- Учасники: Scrum-команда (PO, SM, Команда розробки).
- Мета: Інспекція самої команди та її процесу. Обговорення: "Що вийшло добре?", "Які виникли труднощі?", "Що потрібно покращити?".
- Результат: План покращень (дій), які команда впровадить у наступному Спринті (пряма Адаптація процесу).
- Обмеження за часом (Time-box): Рекомендується не більше 3 годин для місячного Спринту.

Артефакти в Scrum (Scrum Artifacts)

Scrum включає три ключові артефакти, що представляють роботу або цінність.

1. Бэклог Продукту (Product Backlog)

- Що це: Упорядкований список усіх вимог, ідей та побажань до продукту. Це єдине джерело роботи для Scrum-команди.
- Власник: Власник Продукту (PO) є єдиною особою, що керує його вмістом, доступністю та пріоритетом.
- Деталізація: Вимоги з високим пріоритетом, які будуть взяті в роботу найближчим часом, мають бути деталізовані та мати такий розмір, щоб їх можна було виконати за один Спринт.

2. Бэклог Спринту (Sprint Backlog)

- Що це: Набір елементів, вибраних із Product Backlog для поточного Спринту, плюс план команди щодо їх реалізації.
- Власник: Команда розробки володіє Sprint Backlog.
- Фіксація: У ідеалі, Sprint Backlog є фіксованим на час Спринту. Будь-які зміни мають обговорюватися з PO та командою, щоб не порушити Мети Спринту.

3. Інкремент (Increment)

- Що це: Сума всіх виконаних вимог Product Backlog (за поточний Спринт) плюс цінність усіх попередніх Спринтів. Це потенційно готовий до випуску (usable) шматок продукту.

- Критерій "Готовий" (Definition of Done, DoD):
- Це спільне розуміння того, що означає "готова" робота.
- Встановлює мінімальні критерії якості (тестування, документація, review), яким має відповідати кожен елемент. Без виконання DoD робота не може вважатися частиною Інкременту.

Додаткові терміни:

- Критерій готовності до взяття в роботу (Definition of Ready, DoR): Хоча це не офіційний артефакт Scrum-гайду, багато команд використовують DoR як критерій, якому має відповідати задача з Product Backlog, щоб її можна було взяти в Sprint Planning (наприклад, "наявність оцінки", "чітке формулювання").

Питання для самоперевірки, тести для самоконтролю

1. Яка з п'яти подій Scrum є "контейнером" для всіх інших?
2. Які два принципи Scrum підтримуються за допомогою регулярного проведення всіх п'яти подій?
3. Хто входить до складу учасників Daily Scrum? Яке його обмеження за часом?
4. Яка подія Scrum має на меті інспекцію інкременту та адаптацію Product Backlog?
5. Яка подія Scrum має на меті інспекцію самого процесу роботи команди та її взаємодії?
6. Що є основним результатом Огляду Спринту?
7. Що є основним результатом Ретроспективи Спринту?
8. Який із трьох артефактів Scrum є єдиним джерелом роботи для команди? Хто ним володіє?

9. Що таке Інкремент? Яке ключове правило стосується його готовності?
10. Поясніть роль Критерію готовності (DoD) у забезпеченні Прозорості Scrum.

Обов'язкові та додаткові задачі, приклади

Обов'язкова задача 1: Аналіз часових обмежень

Команда проводить Планування Спринту, який триває 4 тижні. На обговорення РО виділяє 3 години, а на Ретроспективу — лише 1 годину.

1. Чи відповідають ці часові обмеження рекомендаціям Scrum-гайду?
2. Які ризики для Принципів Scrum виникають через надто коротку Ретроспективу?

Відповідь:

1. Планування Спринту (4 тижні): Рекомендація до 8 годин. 3 години — це замало, що може призвести до нечіткого Sprint Backlog та відсутності Сфокусованості.

Ретроспектива Спринту (4 тижні): Рекомендація до 3 годин. 1 година — замало. Це загрожує неможливістю провести належну Інспекцію та розробити ефективний План покращень.

2. Ризики Ретроспективи: Порушується принцип Інспекції та Адаптації. Команда не матиме достатньо часу для чесного обговорення проблем (порушується Сміливість) та не зможе сформувавши ефективний План покращень, що призведе до повторення тих самих помилок у наступних Спринтах.

Обов'язкова задача 2: Захист Критерію Готовності (DoD)

Наприкінці Спринту команда завершила задачу, яка була закодована, але не пройшла інтеграційне тестування. Власник продукту (PO) наполягає, що цю задачу необхідно включити в Інкремент і показати на Огляді Спринту.

Критерій готовності (DoD) команди: "Код протестовано, функціональність перевірена на інтеграційному середовищі".

1. Чи може ця задача бути включена в Інкремент?
2. Який ризик створює РО, порушуючи DoD?
3. Хто має захистити DoD у цій ситуації?

Відповідь:

1. Ні, не може. За правилами Scrum, якщо задача не відповідає Критерію готовності (DoD), вона не є частиною Інкременту.

2. Ризик: Порушується принцип Прозорості. Якщо елементи включаються в Інкремент без дотримання якості, це створює технічний борг та спотворює реальний стан продукту для стейкхолдерів.

3. Захисник: Scrum-майстер має захистити команду, пояснюючи, що DoD — це спільне зобов'язання, яке забезпечує якість та Прозорість. Задача повинна повернутися в Product Backlog (або Sprint Backlog наступного Спринту) для завершення тестування.

Довідково-інформаційні дані для розв'язування задач (таблиці, схеми тощо)

1. Каденції Scrum та їхній Фокус (за місячний Спринт)

Подія	Time-Box (макс.)	Основна Мета	Принцип Scrum
Спринт	1 місяць	Створення Інкременту.	Емпіризм, Сфокусованість
Планування Спринту	8 годин	Що і як буде зроблено в Спринті (Sprint Backlog).	Прозорість, Зобов'язання
Daily Scrum	15 хвилин	Синхронізація та план на 24 години.	Інспекція, Адаптація
Огляд Спринту	4 години	Інспекція Інкременту та оновлення Product Backlog.	Інспекція, Адаптація
Ретроспектива Спринту	3 години	Інспекція Процесу та План покращень.	Інспекція, Адаптація

2. Артефакти та Їхні "Комітменти" (Зобов'язання)

Артефакт	Власник	Зобов'язання (Commitment)
Product Backlog	Власник Продукту	Візія Продукту (Product Goal)
Sprint Backlog	Команда розробки	Мета Спринту (Sprint Goal)
Інкремент	Вся Scrum-команда	Критерій Готовності (Definition of Done)

Бібліографічний список

1. Schwaber, Ken, and Sutherland, Jeff. The Scrum Guide™. 2020. [Електронний ресурс]. (Офіційний, канонічний документ, що детально описує п'ять подій (Events) та три артефакти (Artifacts) Scrum-фреймворку).
2. Cohn, Mike. Succeeding with Agile: Software Development Using Scrum. Addison-Wesley Professional, 2009. (Посібник, що детально описує, як на практиці проводяться події Scrum та як ефективно використовувати артефакти).
3. Kerievsky, Joshua. Diving into the Scrum Retrospective. Industrial Logic, 2006. (Джерело, що надає глибокий огляд Ретроспективи як ключової події для еволюційної адаптації).
4. Cohn, Mike. User Stories Applied: For Agile Software Development. Addison-Wesley Professional, 2004. (Описує техніку "User Story" як основний формат елементів Product Backlog).
5. Wiegers, Karl E. Software Requirements, 3rd Edition. Microsoft Press, 2013. (Хоча не є виключно Scrum-джерелом, пояснює принципи розробки вимог, необхідні для ефективного керування Product Backlog).

Тема 10. Принцип СЮ-ХА-РІ (SHU-HA-RI) в освоєнні майстерності

Сю-Ха-Рі (яп. 守破離, Shu-Ha-Ri) — це принцип освоєння нової сфери знань чи навичок поетапно, слідуючи якому можна максимально швидко стати майстром у новій техніці. Цей принцип початково був сформульований

в японських бойових мистецтвах, зокрема айкідо, але його можна використовувати і в особистому житті, і в роботі (наприклад, у методології Scrum).

Назва відображає три етапи освоєння:

Слідування правилам.

Невеликі експерименти.

Слідування інтуїції (самостійне творення).

1. Стадія Сю (守) – «Слідуй правилу» / «Дотримуйся»

На цьому етапі необхідно слідувати тому, що показує інша людина, яка вже досягла майстерності (вчитель, наставник). Учень вивчає та виконує всі дії у суворій відповідності до правил, інструкцій та шаблонів.

В айкідо учневі потрібно багато років тренуватися, інакше у нього не буде бази для переходу на наступний ступінь. На цій стадії напрацьовується основна база, учень поступово звикає до механіки дій і через деякий час стає готовим перейти на наступну стадію.

2. Стадія Ха (破) – «Зламай правила» / «Відділяйся»

На цьому етапі слід відійти від правил і розпочати експериментувати, пробувати нове, шукати кращі способи роботи, аналізувати та практикувати. Учень починає ставити під сумнів встановлені правила.

В айкідо це друга ступінь, на якій учень обмірковує свої дії, змінює правила, пробує їх порушувати, будує нову систему власних правил. Також він освоює нові прийоми в інших учителів, розширюючи свій кругозір.

3. Стадія Рі (離) – «Відділення від правил» / «Перевершуй»

Суть правил освоєна, вони більше не потрібні, можна діяти природним чином. На цьому етапі слід, відірвавшись від звичних правил і шаблонів, практикувати нове у власному стилі, створюючи інновації.

В айкідо це третя ступінь, згідно з якою потрібно звільнитися від правил: їх більше немає, є лише природний хід речей (Дао). «Рі» означає піднятися над усім, що вивчалось раніше, створити більш високі й загальні

принципи. На цьому ступені вже формується органічна та природна поведінка на рівні звичок.

Принцип Сю-Ха-Рі та Scrum (Скрам)

Принцип освоєння Scrum (гнучкої методології управління проєктами) дуже схожий на методику Сю-Ха-Рі:

Сю (Shu): Навчаючись Scrum, спочатку необхідно добре освоїти всі правила та ролі (Скрам-Гайд).

Ха (Ha): Після освоєння базових правил команда починає адаптувати їх до своїх умов, експериментувати з різними практиками та інструментами.

Рі (Ri): Команда переходить на рівень, коли вона діє органічно, і правила Скраму стають природною частиною робочого процесу, дозволяючи створювати нові, більш досконалі практики.

Скрам заснований на постійному поповненні досвіду, вимагає особливої уважності та докладання великих зусиль.

Питання для самоперевірки, тести для самоконтролю

Назвіть усі три етапи принципу Сю-Ха-Рі та поясніть їхні японські відповідники.

Яка основна мета етапу Сю?

Чим відрізняється діяльність учня на етапі Ха від етапу Сю?

Що означає «звільнитися від правил» на етапі Рі?

Як застосовується принцип Сю-Ха-Рі в контексті освоєння методології Scrum?

Тест (Виберіть одну правильну відповідь):

1. На якому етапі учень починає шукати кращі способи роботи, порушуючи встановлені правила?

а) Рі

б) Сю

в) Ха

2. Яка стадія відповідає слідуванню інтуїції та формуванню власних високих принципів?

а) Сю

б) Рі

в) Ха

3. З якого японського бойового мистецтва походить принцип Сю-Ха-Рі?

а) Карате

б) Дзюдо

в) Айкідо

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача: Ви — новий співробітник на посаді тестувальника ПЗ. Вам надали чіткий, покроковий протокол тестування. Як ви застосуєте стадію Сю до цієї ситуації?

Відповідь-приклад: Суворо дотримуватимусь наданого протоколу, не пропускаючи жодного кроку та не намагаючись змінити послідовність, щоб напрацювати базову навичку та зрозуміти логіку процесу.

Задача: Ви, будучи досвідченим інженером (Ха), помічаєте, що стандартний інструктаж із запуску обладнання містить зайвий крок, який сповільнює процес. Що ви зробите згідно з принципом?

Відповідь-приклад: Проведу експеримент: спробую запустити обладнання без цього кроку. Якщо результат буде позитивним, проаналізую це, задокументую та запропоную зміну до стандартного протоколу.

Додаткові задачі (Приклади з професійного життя)

Приклад (Рі): Ви — архітектор ПЗ, який створив нову, більш ефективну модель розробки програмного забезпечення, що об'єднує елементи Scrum, Kanban та XP, але при цьому виходить за рамки кожного з них, базуючись на глибокому розумінні принципів. Поясніть, як це ілюструє стадію Рі.

Приклад (Перехід Сю to Ха): Студент вчиться грати на гітарі. Спочатку він суворо дотримується акордів, які показує вчитель, та грає лише за нотами. Що має статися, щоб він перейшов до стадії Ха?

Довідково-інформаційні дані для розв'язування задач

Стадія (Японська назва)	Переклад (Значення)	Характерні дії / Фокус	Результат / Мета	Приклад (Айкідо / Scrum)
Сю (守)	Дотримуйся, Слідуй	Суворе дотримання правил, копіювання майстра, наслідування інструкцій.	Напрацювання базових навичок, формування основи.	Вивчення всіх правил Скрам- Гайда без відхилень.
Ха (破)	Зламай, Відділяйся	Експерименти, аналіз правил, пошук кращих методів, адаптація, вихід за рамки.	Розуміння суті правил, створення власної системи.	Адаптація Скраму до особливостей команди, пошук нових інструментів.
Pi (離)	Перевершуй, Створюй	Діяти природно, інтуїтивно, створення нових, вищих принципів, вільна творчість.	Створення майстерності, досягнення природної поведінки (Дао).	Створення власної, гібридної, високоєфективної методології.

Бібліографічний список

1. Sutherland, Jeff. Scrum: The Art of Doing Twice the Work in Half the Time. Crown Business, 2014. (Книга від співзасновника Scrum, яка часто посиляється на принцип Сю-Ха-Pi як на основну філософію навчання та майстерності у фреймворку Scrum.)

2. Liker, Jeffrey K. The Toyota Way: 14 Management Principles from the World's Greatest Manufacturer. McGraw-Hill, 2004. (Описує принципи

навчання та вдосконалення, які лежать в основі японської філософії управління, де принцип Сю-Ха-Рі є невід'ємною частиною культури постійного вдосконалення (Кайдзен).)

3. Stevens, John. Three Budo Masters: Jigoro Kano, Gichin Funakoshi, Kenwa Mabuni. Kodansha International, 1995. (Джерело, що надає історичний та філософський контекст японських бойових мистецтв (Будо), звідки походить принцип Сю-Ха-Рі, допомагаючи зрозуміти його оригінальне значення.)

4. Albon, Rob. Shu-Ha-Ri: Learn, Detach, Transcend. 2017. [Електронний ресурс]. (Сучасний огляд принципу Сю-Ха-Рі з акцентом на його застосуванні в процесах навчання, професійного розвитку та гнучких методологіях (Agile) у сфері IT.)

5. Sensei, K. M. (Ed.) The Concepts of Shu-Ha-Ri in Japanese Martial Arts. Journal of Budo Studies, Vol. 15, No. 2, 2008. (Академічна стаття, що глибоко аналізує етапи Сю, Ха та Рі в контексті бойових мистецтв, надаючи канонічне тлумачення для навчальних цілей.)

РОЗДІЛ 3. РОБОТА З ВИМОГАМИ: КЛАСИФІКАЦІЯ ТА МЕТОДИ ВИЯВЛЕННЯ

(Романенко Є.О., д.держ.упр., професор, заслужений юрист України,
начальник управління Центрального науково-дослідного інституту Збройних
Сил України)

Тема 1. Основи розробки вимог

Зацікавлені в проєкті особи (суб'єкт, стейкхолдер, причетна сторона, англ. Stakeholders) – це активно залучені в проєкт окремі особи, групи або організації, на яких впливає результат проєкту і які самі можуть впливати на цей результат.

Стейкхолдери забезпечують можливості для системи та висувують вимоги до неї. У контексті процесу прийняття рішень їх також можна розглядати як людей або організації, які залежать від результатів прийнятих рішень.

Необхідно заздалегідь визначити, хто зацікавлений у прийнятті рішень. Це часто ігнорують. Однак, щойно рішення буде оголошено або реалізовано, всі, кого це рішення хоч якось торкнулося, висловлять свою думку.

Нижче наведені деякі визначення терміна «стейкхолдер» із міжнародних стандартів, які можуть бути корисними для співбесід:

Особи або організації, що мають права, частку, вимоги чи інтереси щодо системи або її властивостей, які задовольняють їхні потреби та очікування (ISO/IEC/IEEE 15288:2015).

Індивідуум, команда, організація або їхні групи, які мають інтерес у системі (ISO/IEC 42010).

Люди, групи або організації, які можуть впливати на систему або на яких може вплинути система (OMG Essence).

Особа, група чи організація, яка може впливати, зазнавати впливу чи сприймати себе як таку, що зазнала впливу рішення, операції чи результату проєкту (PMBOK).

Особа або організація, яка може впливати на здійснення діяльності чи прийняття рішення, бути підданою їхньому впливу або сприймати себе як піддану впливу (ISO 9000:2015).

Ключові поняття в розробці вимог

Термін	Визначення	Приклади
Проблеми (Problems)	Невідповідності у бізнес-процесі, які заважають або не сприяють досягненню бізнес-цілей організації.	Низька рентабельність; Вірогідність втрати частки ринку; Великий відсоток простоїв обладнання/серверів.
Цілі (Goals)	Вигоди для організації, які вона отримає після впровадження нового рішення. Вигода не обов'язково повинна бути фінансовою.	Відділ замовлень працює без збоїв; Маржинальність становить мінімум 15%; Скорочення часу очікування відповіді клієнта.
Очікування (Expectations)	Неявні (приховані) уявлення зацікавлених осіб про характеристики цільової ситуації, продукту, системи, результату проєкту. Мають прямий зв'язок з інтересами цих осіб.	Система буде інтуїтивно зрозумілою; Супровід системи не потребуватиме додаткових витрат; Підрядник дотримуватиметься бюджету і термінів.
Рішення (Solutions)	Діапазон можливих шляхів усунення виявлених бізнес-проблем.	Зміна організаційної структури; Оптимізація існуючих бізнес-процесів; Розробка нової інформаційної системи (IC).

Типи зацікавлених осіб

Вичерпного списку типів або груп стейкхолдерів не існує, оскільки для різних цільових систем вони можуть суттєво різнитися. Нижче наведено

приклади найбільш поширених груп, які згадуються у міжнародних стандартах (наприклад, ISO/IEC 29148:2011).

Придбальна сторона, або Покупець (Acquirer): Придбає або отримує продукт чи послугу від постачальника. Це може бути кінцевий Замовник або власник продукту.

Замовник, або Клієнт (Customer): Отримує продукт або послугу.

Розробник (Developer): Виконує завдання розробки, включаючи аналіз вимог, проєктування, тестування.

Постачальник (Supplier): Надає продукт або послугу.

Користувач (User): Безпосередньо використовує систему та отримує вигоду в процесі її застосування.

Виробник (Producer): Відповідає за виконання роботи, графік, бюджет і розподіл ресурсів.

Сторона, що супроводжує (Maintainer): Здійснює підтримку системи на одному або кількох етапах життєвого циклу.

Ліквідатор (Disposer): Виконує виведення з експлуатації та списання системи.

Акредитор, або Інспектор (Accreditor): Перевіряє систему на відповідність вимогам під час здачі її в експлуатацію.

Регулювальний орган (Regulatory bodies): Перевіряє систему на відповідність державним вимогам і стандартам під час експлуатації (наприклад, Держспецзв'язку в Україні).

Інші: Персонал підтримки (Supporters), інструктори (Trainers), оператори (Operators) тощо.

Ролі зацікавлених осіб у Проєкті

Хід проєкту суттєво залежить від намірів, дій, інформування та підтримки потреб його учасників. В рамках кожного процесу зацікавлені сторони можуть мати декілька ролей.

Основні ролі:

Інвестор (Investor): Надає ресурси для проєкту або формує інвестиційно привабливу ситуацію. Важливо визначити його до початку проєкту для вирішення всіх питань забезпечення ресурсами.

Менеджер Проєкту (Project Manager, PM): Відповідає за управління проєктом, тобто за забезпечення найкращих його результатів у межах заданих обмежень. У класичному розумінні виконує адміністративні обов'язки, не обов'язково будучи лідером.

Лідер Проєкту (Project Lead): Найбільш авторитетна особа в команді. До його думки прислухаються; він приймає більшість технічних рішень. Часто може суміщати роль Менеджера Проєкту.

Замовник (Customer/Client): Отримує результати проєкту у власність того чи іншого виду.

Користувачі (Users): Безпосередньо використовують результати проєкту у своїй діяльності.

Організація-Виконавець (Performing Organization): Виконує проєкт і несе відповідальність перед Замовником.

Проєктна Команда (Project Team): Група спеціалістів, об'єднаних спільною метою, які працюють над завданнями протягом життєвого циклу проєкту.

Профілі зацікавлених осіб (Профілі Користувачів)

Профілі зацікавлених осіб – це категорії клієнтів та інших ключових фігур, які зацікавлені в проєкті. Сферою інтересу повинні стати різні групи клієнтів, цільові ринкові сегменти та класи користувачів (наприклад, Персони в UX-дизайні).

Інформація, що включається у профіль (цінності):

Передусім до профілю включається основна цінність або перевага, яку продукт принесе зацікавленій особі, і те, як він задовольнить їхні потреби.

Що може бути цінністю для стейкхолдера:

Покращена продуктивність;

Менша кількість переробок;

Зниження собівартості;

Прискорення бізнес-процесів;

Автоматизація ручних завдань;

Можливість розв'язувати абсолютно нові завдання;

Відповідність стандартам і законам України (наприклад, щодо захисту персональних даних);

Легкість і простота використання (зручний інтерфейс).

Джерела впливу:

Це особи, групи або організації, які напряду не пов'язані з проведенням проєкту, але мають свої інтереси і здатні на нього вплинути. Це можуть бути державні та громадські організації, компанії-конкуренти, ЗМІ, окремі особи, чиє життя зачіпає проєкт.

Проектна команда повинна ідентифікувати зацікавлені сторони, визначити їхні вимоги та очікування і, по можливості, відрегулювати їхній вплив на проєкт, щоб забезпечити його успішність.

Питання для самоперевірки, тести для самоконтролю

Дайте визначення терміна «стейкхолдер» і наведіть приклад із реального ІТ-проєкту (наприклад, розробка мобільного банкінгу).

Назвіть різницю між Проблемою та Ціллю в контексті розробки вимог. Наведіть власний приклад.

Чому так важливо виявити Очікування зацікавлених осіб, якщо вони не є явними вимогами?

Опишіть, як роль Менеджера Проєкту відрізняється від ролі Лідера Проєкту.

Яка інформація є ключовою при формуванні Профілю зацікавленої особи?

Тест (Виберіть одну правильну відповідь):

1. Хто з перелічених осіб найбільше відповідає за суворе дотримання бюджету і термінів у рамках проєкту?

- а) Замовник
- б) Користувач
- в) Менеджер Проєкту

2. Який тип стейкхолдера в українських реаліях може перевіряти систему на відповідність законодавству?

- а) Розробник
- б) Регулювальний орган
- в) Ліквідатор

3. Що з переліченого є типовим прикладом бізнес-ЦІЛІ?

- а) Система буде швидко працювати.
- б) Збільшення конверсії продажів на 5%.
- в) Інтерфейс буде у фірмових кольорах.

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача: Ви — бізнес-аналітик у проєкті розробки застосунку для онлайн-запису до лікаря. Ідентифікуйте мінімум 5 різних типів стейкхолдерів і сформулюйте їхню ключову ЦІННІСТЬ (вигоду) від системи.

Відповідь-приклад: 1) Лікарі (Користувачі): Цінність — зменшення рутинної роботи з ведення графіків, автоматичне нагадування пацієнтам. 2) Власник клініки (Придбальна сторона/Інвестор): Цінність — збільшення завантаженості клініки та прибутку. 3) Реєстратор (Оператор): Цінність — швидкий пошук інформації про пацієнта.

Задача: Директор компанії (Замовник) висловлює Очікування: «Система має бути красивою». Запропонуйте, як перевести це неявне очікування у вимірювану вимогу.

Відповідь-приклад: Провести сесію з Замовником, щоб визначити, що він вкладає в поняття «красива». Наприклад: «Система має відповідати корпоративному стилю (надати гайдлайни)» або «Система має отримати оцінку юзабіліті не менше 4/5 від 90% тестової групи користувачів».

Додаткові задачі (Приклади)

Приклад: Успішний запуск державної послуги в застосунку «Дія». Проаналізуйте, хто виступав у ролях Замовника, Організації-Виконавця, Регулювального Органу та Користувачів у цьому проєкті.

Приклад: Визначте, чим Проблема «Висока плинність кадрів» відрізняється від Цілі «Зниження плинності кадрів на 15% протягом року». Сформулюйте для цієї ситуації 2-3 можливих Рішення (не тільки ІТ-рішення).

Довідково-інформаційні дані для розв'язування задач (таблиці, схеми тощо)

Матриця влади та інтересу Стейкхолдерів (модель Ментелоу)

Ця модель допомагає Менеджеру Проєкту визначити пріоритетність взаємодії зі стейкхолдерами.

	Висока ВЛАДА (Вплив)	Низька ВЛАДА (Вплив)
Високий ІНТЕРЕС	Керувати тісно (Manage Closely) (Замовник, Інвестор)	Тримати в інформуванні (Keep Informed) (Лідер Проєкту, Ключові Користувачі)
Низький ІНТЕРЕС	Задовольнити (Keep Satisfied) (Регулювальний Орган, Вище Керівництво)	Мінімізувати зусилля (Monitor/Minimal Effort) (Персонал Підтримки, Широкі Користувачі)

Пояснення:

ВЛАДА (Power): Здатність стейкхолдера впливати на проєкт та його результати.

ІНТЕРЕС (Interest): Ступінь зацікавленості стейкхолдера у результатах проєкту.

Керувати тісно: Необхідна постійна комунікація та залучення до прийняття ключових рішень.

Задовольнити: Важливо, щоб ці стейкхолдери не були незадоволені, але їхнє активне залучення не потрібне.

Бібліографічний список

1. Wiegers, Karl E., and Joyner, Laura. Software Requirements. 3rd Edition. Microsoft Press, 2013. (Класичний посібник, що детально описує процес розробки вимог, включно з ідентифікацією стейкхолдерів, виявленням їхніх потреб та керуванням вимогами.)

2. ISO/IEC/IEEE 15288:2015. Systems and software engineering — System life cycle processes. (Міжнародний стандарт, що описує процеси життєвого циклу системної інженерії, містить канонічні визначення зацікавлених осіб (стейкхолдерів) та їхньої ролі у процесі розробки вимог, зокрема в процесі визначення потреб стейкхолдерів.)

3. ISO/IEC/IEEE 29148:2018. Systems and software engineering — Life cycle processes — Requirements engineering. (Спеціалізований міжнародний стандарт, присвячений виключно інженерії вимог, який деталізує діяльність, пов'язану з виявленням, аналізом, специфікацією, валідацією та керуванням вимогами, отриманими від стейкхолдерів.)

4. PMI. A Guide to the Project Management Body of Knowledge (PMBOK® Guide) – Seventh Edition. Project Management Institute, 2021. (Офіційний звід знань з управління проєктами, містить розділи про ідентифікацію, аналіз та залучення стейкхолдерів, а також їхній вплив на цілі та результати проєкту.)

5. Robertson, Suzanne, and Robertson, James. Mastering the Requirements Process: Getting Requirements Right. 3rd Edition. Addison-Wesley Professional, 2013. (Практичний посібник із фокусом на методах виявлення справжніх потреб бізнесу та стейкхолдерів, а також на техніках структурування та моделювання вимог.)

Тема 2. Користувачі та класифікація

Користувачі – це окрема група зацікавлених осіб і учасників проекту, які будуть безпосередньо працювати з системою.

Користувач (User) – особа або організація, яка використовує діючу систему для виконання конкретної функції.

На самому початку роботи над проектом необхідно визначити та охарактеризувати різні класи користувачів, щоб дізнатися про вимоги представників усіх важливих класів.

Класифікація Користувачів

Користувачів продукту можна поділяти за такими ознаками:

Частота використання продукту (наприклад, щоденно, щомісячно);

Досвід у предметній області та досвід роботи з комп'ютерними системами (Новачок проти Експерта);

Функціональність, яка їм потрібна (наприклад, лише перегляд звітів чи повне редагування);

Завдання, які їм доводиться розв'язувати;

Права доступу до системи (наприклад, «звичайний користувач», «гість», «адміністратор»).

На основі цих ознак формуються класи користувачів. Деяких співробітників можна віднести до декількох класів. Наприклад, адміністратор іноді працює з системою як рядовий користувач.

Приклади впливу досвіду на вимоги:

Новачків хвилює, наскільки легко навчитися працювати з продуктом. Вони віддають перевагу графічним інтерфейсам, упорядкованому поданню інформації та докладним підказкам.

Досвідчених користувачів більше цікавить легкість використання та ефективність роботи. Вони цінують клавіатурні скорочення, макроси, можливості налаштування та інтерфейс командного рядка замість графічного.

Типи Класів Користувачів

При прийнятті рішення про пріоритети або пошуку компромісу між вимогами, висунутими різними класами користувачів, думка привілейованих класів має першорядне значення.

Привілейовані Класи: Групи користувачів, робота яких із продуктом визначає, чи досягне він заявлених бізнес-цілей. Їхні вимоги мають найвищий пріоритет.

Непривілейовані Класи: Користувачі, які з причин безпеки, конфіденційності або правових причин не працюють з продуктом або працюють з ним мінімально (наприклад, аудитори).

Ігноровані Класи: Решта класів користувачів, інтереси яких можна проігнорувати. Вони отримають те, що вдасться створити; під час розробки системи не потрібно враховувати їхні інтереси.

Інші Класи: Користувачі, думки яких при визначенні вимог мають приблизно однакове значення.

Важливо пам'ятати про класи непрямих, або вторинних, користувачів. Вони можуть не мати прямого доступу до застосунку, а працювати з його даними та сервісами через інші застосунки або звіти. Проте, навіть опосередкований клієнт все одно залишається клієнтом.

Класи користувачів не обов'язково складаються лише з людей. Сторонні застосунки та апаратні компоненти можна розглядати як додаткові класи користувачів, з якими взаємодіє система.

Приклад: Система впорскування палива в автомобілі є користувацьким класом для програмного забезпечення, вбудованого в систему керування двигуном. Вимоги цього "користувача" (наприклад, щодо часу відгуку) має з'ясувати аналітик в інженера-розробника цієї системи.

Метод Виявлення Класів Користувачів

Один із корисних способів визначення класів називається «від розширення до стиснення» (expand then contract).

Розширення: Спочатку потрібно придумати якомога більше класів користувачів (навіть занадто багато). Важливо не пропустити жодного класу, інакше це може призвести до проблем пізніше.

Стиснення: Наступний етап — виявити групи зі схожими потребами. Їх можна об'єднати в один клас або розглядати як кілька підкласів. У підсумковому списку окремих класів не повинно бути більше 10–15 пунктів.

Корисні запитання для виявлення класів:

Хто є користувачем системи (той, хто працює з нею)?

Хто є замовником/економічним покупцем системи?

На кого ще вплинуть результати роботи системи?

Хто буде оцінювати та приймати систему під час її розгортання?

Чи існують інші внутрішні чи зовнішні користувачі, чії потреби необхідно врахувати?

Хто буде займатися супроводом нової системи?

Чи не забули ми когось? (Це клас користувачів, думка яких при визначенні вимог до продукту мають приблизно однакове значення)

Для чого потрібна класифікація?

Класифікація користувачів за схожими потребами значно спрощує збір вимог та їхнє пріоритезування, оскільки замість опису потреб 65 окремих клієнтів, аналітик працює з потребами 6-10 узагальнених класів.

Запам'ятайте: Клас користувачів — це група людей, яким слід надати можливість висловити свою думку про створюваний продукт.

Питання для самоперевірки, тести для самоконтролю

У чому полягає основна відмінність між Зацікавленою особою (Стейкхолдером) і Користувачем?

Наведіть три критерії, за якими можна розділити користувачів на класи.

Чому вимоги від Привілейованих класів мають вищий пріоритет, ніж від Інших класів?

Поясніть сутність методу виявлення класів користувачів «від розширення до стиснення».

Наведіть приклад Нелюдського Користувача для системи керування розумним будинком.

Тест (Виберіть одну правильну відповідь):

1. Який клас користувачів найбільше цінує клавіатурні скорочення та можливості налаштування?

- а) Новачки
- б) Досвідчені користувачі
- в) Непрямі користувачі

2. Яка дія є частиною фази «Стиснення» методу «від розширення до стиснення»?

- а) Генерація максимальної кількості потенційних класів.
- б) Об'єднання груп зі схожими потребами в один клас.
- в) Визначення функціональних вимог системи.

3. Хто з перелічених осіб є типовим «Ігнорованим Класом» у процесі збору вимог?

- а) Спонсор проєкту.
- б) Користувачі, яких не потрібно враховувати, оскільки їхні інтереси не впливають на бізнес-цілі.
- в) Адміністратор бази даних.

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача: Уявіть, що ви розробляєте нову систему електронного документообігу (ЕДО) для середньої української компанії. Використовуючи метод «від розширення до стиснення», спочатку розширте список потенційних користувачів (мін. 8), а потім стисніть їх у 3-4 ключові класи.

Розширення: Керівник відділу, секретар, бухгалтер, ІТ-адміністратор, звичайний співробітник, юрист, зовнішній аудитор, архівіст.

Стиснення: 1) Оператори (Секретарі, Співробітники): Використовують щоденно для створення/підписання документів. 2) Контролери (Керівники, Юристи): Використовують періодично для затвердження та перевірки. 3) Технічна підтримка (ІТ-адміністратор): Використовує для налаштування та обслуговування.

Задача: Сформулюйте по одному нефункціональному вимогу для Новачка та для Досвідченого користувача системи ЕДО.

Відповідь-приклад: Новачок: Система повинна мати вбудований «Майстер першого запуску» з покроковими інструкціями. Досвідчений користувач: Система повинна підтримувати налаштування власних клавіатурних скорочень для швидкої навігації.

Додаткові задачі (Приклади)

Приклад (Непрямий Користувач): Система формування звітів про доходи компанії не використовується безпосередньо Генеральним директором. Директор отримує ці дані лише через PDF-звіт, сформований іншим підрозділом. Поясніть, чому Генеральний директор все ще є важливим непрямим користувачем і як це впливає на вимоги (наприклад, до формату та змісту PDF).

Приклад (Привілейовані Класи): У проєкті розробки ІТ-системи для керування залізничним рухом визначте, хто буде Привілейованим Класом користувачів і чому (з огляду на бізнес-цілі та безпеку).

Довідково-інформаційні дані для розв'язування задач

Ознака	Новачок (Початківець)	Досвідчений Користувач (Експерт)
Ключовий інтерес	Легкість навчання / Запам'ятовування	Ефективність / Швидкість роботи
Бажаний Інтерфейс	Графічний інтерфейс (GUI), Wizards (Майстри)	Клавіатурні скорочення, Командний рядок (CLI)
Нефункціональні Вимоги	Послідовність, Підказки, Висока Толерантність до помилок	Можливість налаштування, Гнучкість, Багатозадачність
Вплив на дизайн	Чітка структура, мінімум функцій на екрані	Висока інформаційна щільність, багато елементів керування

Бібліографічний список

1. Wiegers, Karl E., and Joyner, Laura. Software Requirements. 3rd Edition. Microsoft Press, 2013. (Містить детальні методики для ідентифікації та класифікації користувачів, а також вплив різних класів на пріоритезацію функціональних та нефункціональних вимог.)
2. Robertson, Suzanne, and Robertson, James. Mastering the Requirements Process: Getting Requirements Right. 3rd Edition. Addison-Wesley Professional, 2013. (Описує техніку «від розширення до стиснення» для ефективного виявлення класів стейкхолдерів та користувачів.)
3. ISO/IEC/IEEE 29148:2018. Systems and software engineering — Life cycle processes — Requirements engineering. (Міжнародний стандарт, який визначає термін «користувач» та його роль у процесі інженерії вимог, необхідний для академічної точності.)
4. Cooper, Alan, Reimann, Robert, Cronin, Dave, and Maartje, Heijne. About Face: The Essentials of Interaction Design. 4th Edition. Wiley, 2014. (Ключове джерело з дизайну взаємодії, яке пояснює різницю між вимогами Новачків та Експертів і те, як це впливає на проектування користувацького інтерфейсу.)

Тема 3. Організаційна модель компанії

Ця тема має ознайомчий характер і корисний для розуміння того, як організовані внутрішні процеси у великих компаніях. Розуміння організаційної моделі допоможе бізнес-аналітику ідентифікувати стейкхолдерів та ефективно з ними комунікувати в процесі збору вимог.

Організаційна модель – це принципи формування підрозділів компанії, делегування повноважень та наділення відповідальністю. По суті, вона показує, як структуруються підрозділи в організації.

На практиці застосовують такі основні принципи організаційних моделей:

Функціональна модель: Один підрозділ = одна функція (наприклад, бухгалтерія, відділ маркетингу).

Процесна модель: Один підрозділ = один процес (наприклад, процес закупівлі, процес обслуговування клієнтів).

Матрична модель: Один проєкт = група співробітників із різних функціональних підрозділів.

Модель, орієнтована на контрагента: Один підрозділ = один контрагент (клієнт, постачальник, підрядник).

В більшості випадків використовуються функціональна та процесна моделі, а також їхні модифікації.

1. Функціональна Модель (Ієрархічна / Бюрократична)

Структури управління на багатьох сучасних підприємствах побудовані відповідно до функціональної моделі, в основі якої лежать принципи, сформульовані Максом Вебером (концепція раціональної бюрократії).

Принципи функціональної моделі:

Принцип ієрархічності: Кожен нижчий рівень контролюється та підпорядковується вищому.

Принцип відповідності: Повноваження та відповідальність працівників відповідають їхньому місцю в ієрархії.

Принцип розділення праці: Поділ на окремі функції та спеціалізація працівників за виконуваними функціями.

Принцип формалізації та стандартизації: Діяльність регламентується стандартами та інструкціями для забезпечення однорідності виконання обов'язків.

Принцип безособовості: Виконання працівниками своїх функцій є обезличеним.

Принцип кваліфікаційного відбору: Наймання та звільнення здійснюється відповідно до кваліфікаційних вимог.

Найбільш поширений тип такої структури — лінійно-функціональна (лінійна) структура.

2. Процесна Модель

Процесні системи ґрунтуються на концепції управління процесами, запропонованій Фредеріком Тейлором та розвиненій Анрі Файолем (реінжиніринг).

Реінжиніринг — це здійснення діяльності відповідно до поставлених завдань шляхом отримання оптимальної переваги з усіх доступних ресурсів.

Принципи процесної моделі:

Принцип об'єднання процедур: Операції, які виконували різні співробітники, інтегруються в одну (горизонтальне стиснення процесу). Якщо це неможливо, створюється команда, відповідальна за весь процес.

Принцип нерозривної послідовності: Кроки процесу виконуються в природному порядку. Робота виконується силами змішаних груп (різних спеціалізацій).

Принцип власника процесу: Уповноважений менеджер забезпечує єдину точку контакту для замовника, виступаючи буфером між складним процесом та клієнтом.

Принцип самостійності вибору: Виконавці самостійно приймають рішення та несуть відповідальність за кінцевий результат.

Принцип горизонтального контролю: Якість результату перевіряється його споживачем – наступним елементом процесного ланцюжка.

Принцип системності (цілісності) управління: Управління витратами відбувається за місцем їх виникнення. Діє правило: «один процес – один підрозділ – один бюджет».

3. Матрична Модель

Матричні моделі поєднують принципи побудови функціональних і процесних систем. У цих структурах співіснують жорстко регламентовані функції та проєкти/процеси.

Принципи матричної моделі:

Подвійне підпорядкування: Працівники здійснюють свою діяльність під оперативним керівництвом менеджера процесу/проєкту та адміністративним контролем керівника функціонального підрозділу («функціонального колодязя»).

Роль менеджера процесу: Полягає у координації дій та ресурсів всередині процесу чи проєкту.

Особливості:

Матрична модель часто призводить до двовладдя процесів та функцій, тому вона не повністю усуває недоліки функціональної системи. На практиці матричні структури добре застосовні для організації управління проєктною діяльністю, але менш придатні для регулярного менеджменту.

4. Змішані Моделі

Змішані моделі передбачають використання різних організаційних субмоделей у межах окремих бізнес-процесів, тоді як для організації в цілому може застосовуватися, наприклад, процесна модель. Це дозволяє використовувати переваги тієї чи іншої моделі для конкретного підрозділу.

Приклади застосування змішаних моделей:

Для структурного блоку, що реалізує бізнес-процес розробки нових продуктів (де потрібна гнучкість і міжфункціональна взаємодія), доцільно використовувати матричну модель.

Для фінансових служб або відділу кадрів, де потрібна висока стандартизація та спеціалізація, часто використовують функціональну модель.

Для підрозділів, що працюють із обмеженими клієнтськими групами або монополістами-постачальниками, підходять моделі, орієнтовані на контрагента.

Вибір тих чи інших субмоделей залежить від специфіки бізнесу та його стратегічних цілей.

Питання для самоперевірки, тести для самоконтролю

У чому полягає основна відмінність між Функціональною та Процесною організаційними моделями?

Назвіть, який принцип функціональної моделі сформулював Макс Вебер.

Яка організаційна модель найбільш підходить для управління проєктною діяльністю і чому?

Поясніть, що означає «двовладдя» в контексті матричної моделі.

Яка модель, ймовірно, буде використовуватися для відділу, що займається обслуговуванням лише одного ключового корпоративного клієнта?

Тест (Виберіть одну правильну відповідь):

1. Принцип «Один підрозділ = одна функція» відповідає:

- а) Процесній моделі
- б) Матричній моделі
- в) Функціональній моделі

2. Ключова роль «Власника процесу» (Principle of Process Owner) є характерною для:

- а) Функціональної моделі
- б) Процесної моделі
- в) Моделі, орієнтованої на контрагента

3. Яка модель передбачає, що працівники підпорядковуються одночасно двом керівникам (функціональному та проєктному/процесному)?

- а) Лінійно-функціональна
- б) Матрична
- в) Процесна

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача: Ви — бізнес-аналітик у великому банку. Вам потрібно зібрати вимоги для вдосконалення процесу видачі кредитів малому бізнесу. Припустимо, банк працює за Функціональною моделлю. Ідентифікуйте 3 різні функціональні підрозділи, які обов'язково будуть залучені до цього процесу.

Відповідь-приклад: 1) Кредитний відділ (аналіз платоспроможності), 2) Юридичний відділ (підготовка договору), 3) Бухгалтерія/Фінансовий відділ (видача та контроль коштів).

Задача: Команда розробки має суворе подвійне підпорядкування: функціонально (за якість коду) — Технічному директору, а оперативно (за терміни та функціональність) — Менеджеру Проєкту. Яка організаційна модель застосовується до цієї команди? Опишіть потенційний ризик такої структури.

Відповідь-приклад: Застосовується Матрична модель. Ризик — конфлікт пріоритетів та двовладдя, коли технічний директор вимагає високої якості, а менеджер проєкту — дотримання стислих термінів.

Додаткові задачі (Приклади)

Приклад (Смішана модель): В ІТ-компанії відділ розробки застосовує Матричну модель (співробітники працюють над різними проєктами), а відділ продажів — Модель, орієнтовану на клієнта (кожен менеджер відповідає за певну клієнтську групу). Поясніть, які переваги отримує компанія, використовуючи таку змішану модель.

Приклад (Процесна модель): Сформулюйте, як виглядав би процес «Обробка скарги клієнта» згідно з принципами Процесної моделі (враховуючи принцип власника процесу та горизонтального контролю).

Довідково-інформаційні дані для розв'язування задач

Організаційна Модель	Основний принцип формування	Переваги	Недоліки / Ризики
Функціональна	Одна функція = Один підрозділ	Висока спеціалізація, економія за рахунок масштабу, чітка ієрархія.	Складна комунікація між відділами (бар'єри), фокус на функціях, а не на результаті.
Процесна	Один процес = Один підрозділ	Фокус на клієнті та кінцевому результаті, швидка адаптація, гнучкість.	Високі витрати на дублювання функцій, складна підтримка глибокої спеціалізації.
Матрична	Комбінація функцій та процесів/проєктів	Гнучке використання ресурсів, орієнтація на проєкт, збереження функціональної спеціалізації.	Двовладдя (конфлікт пріоритетів), висока накладність на адміністрування.

Бібліографічний список

1. Robbins, Stephen P., and Judge, Timothy A. Organizational Behavior. 18th Edition. Pearson Education, 2019. (Класичний підручник з організаційної

поведінки, який докладно описує принципи функціональної (бюрократичної) та матричної організаційних структур, включаючи концепції Макса Вебера.)

2. Hammer, Michael, and Champy, James. Reengineering the Corporation: A Manifesto for Business Revolution. HarperBusiness, 1993. (Ключове джерело з концепції реінжинірингу бізнес-процесів, яке лягло в основу Процесної моделі управління організацією.)

3. PMI. A Guide to the Project Management Body of Knowledge (PMBOK® Guide) – Seventh Edition. Project Management Institute, 2021. (Офіційний звіт знань з управління проєктами, містить розділи про вплив організаційних структур (включно з матричною та функціональною) на управління проєктами.)

4. Daft, Richard L. Organization Theory and Design. 13th Edition. Cengage Learning, 2019. (Навчальний посібник, що надає глибокий аналіз різних організаційних моделей, їхніх переваг, недоліків та умов застосування, включно зі змішаними моделями та моделями, орієнтованими на клієнта.)

5. Davenport, Thomas H. Process Innovation: Reengineering Work Through Information Technology. Harvard Business School Press, 1993. (Робота, що розвиває ідеї процесного управління, пояснюючи, як інформаційні технології можуть підтримувати принцип «один процес – один підрозділ».)

Тема 4. Вимоги та їх класифікація

Вимога – це будь-яка умова, якій має відповідати система, що розробляється, або програмний засіб. Вимога може бути можливістю, якою повинна володіти система, або обмеженням, якому вона повинна задовольняти. Простіше кажучи, це придатний для реалізації опис того, що має бути зроблено.

Існує три основні види вимог, з якими працює бізнес-аналітик. Вони формують ієрархію або «матрьошку», де високорівневі вимоги розпадаються на детальніші:

Бізнес-вимоги (Business Requirements)

Користувацькі вимоги (User Requirements)

Функціональні вимоги (System/Functional Requirements)

1. Бізнес-вимоги (Business Requirements)

Це вимоги, що описують високорівневі бізнес-цілі організації, які вона прагне досягти за допомогою проєкту чи впровадження нової системи. Вони пояснюють, навіщо організації потрібні ці інформаційні системи.

Хто висуває: Замовник проєкту, ключові стейкхолдери.

Приклади:

Зменшити середній час реалізації заявок на 30%.

Збільшити кількість виробленої продукції на 10%.

Підвищити задоволеність клієнтів (Customer Satisfaction Index) на 15%.

Перш ніж створювати будь-яку нову систему, потрібно зрозуміти, яких бізнес-цілей ми прагнемо досягти.

2. Користувацькі вимоги (User Requirements)

Це вимоги, що описують завдання, які певні класи користувачів повинні мати можливість виконувати в системі для досягнення бізнес-цілей.

Користувацькі вимоги часто формуються у вигляді User Stories (Користувацьких історій) за шаблоном:

ЯК <роль користувача>

Я ХОЧУ <досягти певної мети>

ЩОБ <отримати бізнес-вигоду>

Приклад:

ЯК співробітник одного з відділів, Я ХОЧУ мати можливість формувати та подавати заявки в електронному вигляді, ЩОБ скоротити час на пересування між кабінетами, прискорити створення продукту та виключити помилки.

Приклад залежності від Бізнес-вимоги:

Бізнес-вимога: Зменшити середній час реалізації заявок на 30%.

Користувачка вимога: Як співробітник обслуговуючого підрозділу, Я ХОЧУ мати можливість бачити список вхідних заявок у вигляді таблиці, ЩОБ було зручно фільтрувати, відстежувати заявки та запобігати їхній втраті.

3. Системні / Функціональні Вимоги (System Requirements)

Це кінцевий, найбільш деталізований вид вимог, що описує поведінку системи в певних умовах. З цими вимогами безпосередньо працюють програмісти.

Приклад (деталізація Користувачської вимоги):

Необхідно реалізувати форму подачі заявки з такими полями: ...

Поле «Тип заявки» має бути реалізоване як випадаючий список з можливістю пошуку за введеними символами.

Варіанти значення випадаючого списку: «Технічна підтримка», «Запит на відпустку», «Новий продукт».

Поле «Тип заявки» є обов'язковим до заповнення.

Класифікація Вимог за Вігерсом (Розширена Класифікація)

Хоча бізнес-аналітик переважно працює з трьома основними видами, корисно мати уявлення про розширену класифікацію (за Вігерсом), яка поділяє системні вимоги на Функціональні та Нефункціональні.

Функціональні та Нефункціональні Вимоги

Функціональні вимоги (Functional Requirements): Описують, ЩО система повинна робити (її поведінку, функції, послуги).

Приклад: Система повинна дозволяти користувачеві входити в обліковий запис, використовуючи двофакторну автентифікацію.

Нефункціональні вимоги (Non-functional Requirements): Описують, Як повинна працювати система. Це особливі властивості системи або обмеження, які на неї накладаються. Вони часто пов'язані з атрибутами якості.

Приклади: Продуктивність, безпека, надійність, масштабованість, зручність використання (Usability).

Додаткові Класифікації

Системні Вимоги (System Requirements) (в контексті Вігерса): Вимоги до системи в цілому — до ПЗ та обладнання, а також до структури даних та алгоритмів.

Приклади: Масштабованість, відмовостійкість.

Бізнес-правила (Business Rules): Корпоративні політики, урядові постанови, промислові стандарти та обчислювальні алгоритми. Це формалізовані або неформалізовані правила, які використовуються в роботі та накладають обмеження на бізнес-вимоги.

Бізнес-правила не є вимогами до ПЗ, оскільки знаходяться поза межами будь-якої системи ПЗ, але безпосередньо впливають на них.

Приклад: SMS клієнтам можуть відправлятися з 9:00 до 21:00 за часовим поясом клієнта.

Обмеження (Constraints): Вимоги, що визначають, які дії система не виконуватиме, або вимоги до застосування тих чи інших технічних рішень.

Приклади:

Система не зможе збирати персональні дані користувачів, оскільки це обмежено законодавством України (наприклад, Законом про захист персональних даних).

Для здешевлення підтримки система має бути написана мовою програмування Java, оскільки замовник має багато ПЗ, написаного на цій мові.

Важливо: Необхідно розібратися в обґрунтованості обмежень, оскільки іноді вони суперечать бізнес-цілям і є наслідком надмірної перестраховки стейкхолдерів.

Питання для самоперевірки, тести для самоконтролю

Поясніть, що таке Вимога та як її можна спрощено описати?

Назвіть та опишіть три основні види вимог, які формують ієрархію «матрьошки».

Наведіть приклад Бізнес-вимоги, що стосується фінансових показників.

Сформулюйте Користувацьку вимогу у форматі User Story для системи пошуку роботи.

Чому Бізнес-правила не вважаються вимогами до ПЗ, але мають бути відображені в документації?

Тест (Виберіть одну правильну відповідь):

1. Вимоги, що описують високоуровневі бізнес-цілі організації (наприклад, «Збільшити частку ринку»), називаються:

- а) Системні вимоги
- б) Користувацькі вимоги
- в) Бізнес-вимоги

2. Вимога «Система повинна завантажувати сторінку не довше ніж за 3 секунди» є прикладом:

- а) Функціональної вимоги
- б) Нефункціональної вимоги (Продуктивність)
- в) Обмеження

3. Принцип, згідно з яким SMS клієнтам не можуть відправлятися вночі, є прикладом:

- а) Користувацької вимоги
- б) Бізнес-правила
- в) Обмеження (Constraint)

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача: Для такого Бізнес-вимоги: «Збільшити кількість успішних транзакцій на 20%» сформулюйте:

Одну Користувацьку вимогу.

Одну Функціональну вимогу (Системну).

Відповідь-приклад: Користувацька: ЯК клієнт, Я ХОЧУ отримувати миттєве підтвердження транзакції на екрані, ЩОБ бути впевненим у її успішному завершенні. Функціональна: Система повинна надсилати код відповіді 200 ОК протягом 0.5 секунди після обробки транзакції.

Задача: Визначте, чим є: «Система має бути розроблена на платформі .NET Core».

Відповідь-приклад: Це Обмеження (Constraint), оскільки воно стосується технічного рішення, яке накладається зовнішніми чинниками (наявність експертизи/інфраструктури у Замовника).

Додаткові задачі (Приклади)

Приклад (Бізнес-правило): Уявіть, що ви розробляєте внутрішню ІТ-систему для страхової компанії. Сформулюйте одне Бізнес-правило, що регулює процес виплати страхового відшкодування. Поясніть, як це правило вплине на функціональну вимогу.

Приклад (Розмежування): Розмежуйте та поясніть різницю між Нефункціональною вимогою «Система повинна бути високодоступною (працювати 99.9% часу)» та Функціональною вимогою «Система повинна мати функцію автоматичного резервного копіювання даних щоночі».

Довідково-інформаційні дані для розв'язування задач

Рівень	Тип Вимоги	Фокус	Формат / Приклад
Найвищий	Бізнес-вимоги	Цілі, Вигоди для Організації	Зменшити операційні витрати на 15%.
Середній	Користувацькі вимоги	Завдання, які виконують Користувачі	User Story: ЯК менеджер, Я ХОЧУ фільтрувати замовлення...
Деталізований	Функціональні вимоги	Поведінка, Функції Системи	Система повинна відображати 5 вкладок у головному меню.

Основні Атрибути Якості (Нефункціональні Вимоги)

Атрибут	Приклад
Продуктивність	Час відгуку системи не більше 2 секунд при 1000 користувачах.
Безпека	Всі дані користувачів мають бути зашифровані за стандартом AES-256.
Надійність	Середній час між збоями (MTBF) має перевищувати 1000 годин.
Масштабованість	Система повинна підтримувати зростання кількості транзакцій на 30% на рік.
Юзабіліті (Зручність)	Нові користувачі повинні виконати перше завдання без допомоги за 5 хвилин.

Бібліографічний список

1. Wiegers, Karl E., and Joyner, Laura. Software Requirements. 3rd Edition. Microsoft Press, 2013. (Еталонне джерело, яке надає деталізовану класифікацію вимог, включно з функціональними, нефункціональними, бізнес-правилами та обмеженнями, необхідне для академічної повноти.)
2. IIBA. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute of Business Analysis, 2015. (Ключовий документ для бізнес-аналітиків, що визначає ієрархію вимог: Бізнес-вимоги, Стейкхолдерські вимоги (Користувацькі), Рішення (Функціональні та Нефункціональні) та Вимоги переходу.)

3. Cohn, Mike. User Stories Applied: For Agile Software Development. Addison-Wesley Professional, 2004. (Описує техніку Користувацьких історій (User Stories) як основний формат для документування Користувацьких вимог в Agile-контексті.)

4. ISO/IEC/IEEE 29148:2018. Systems and software engineering — Life cycle processes — Requirements engineering. (Міжнародний стандарт, який встановлює процеси та інформаційні елементи для інженерії вимог, включаючи їхню класифікацію та зв'язок між різними рівнями.)

Тема 5. Збір та виявлення вимог

Виявлення та збір вимог — це комплекс заходів, спрямованих на отримання й аналіз потрібної інформації від зацікавлених осіб (стейкхолдерів). Отримані висновки стануть основою для функціональних вимог до майбутньої системи або її модифікації.

Народна мудрість: «Починайте з кінця, щоб не зробити помилку на початку».

Ця мудрість означає, що слід почати з визначення кінцевої мети. Підсумком роботи бізнес-аналітика є функціональні вимоги, реалізація яких:

Дасть користувачам можливість виконувати необхідні завдання в системі, задовольняючи їхні потреби.

Допоможе досягти поставлених бізнес-цілей організації.

Усталений термін «збір вимог» не зовсім коректний. Він передбачає, що у користувача вже є готові вимоги, і вам необхідно лише зібрати їх, що найчастіше хибно. У найкращому випадку у замовника є побажання, які часто є неформалізованими. Таким чином, під цим терміном мається на увазі отримання інформації, на основі якої аналітик буде створювати, виявляти та формалізувати вимоги.

Основні Методи Виявлення Вимог

Існує багато методів виявлення вимог. Нижче наведено ті, які найчастіше застосовуються на практиці бізнес-аналітиками:

Метод	Опис	Тип
1. Інтерв'ювання	Пряма розмова із зацікавленими особами для виявлення їхніх потреб, проблем та очікувань.	Первинний
2. Аналіз документації	Вивчення існуючих внутрішніх документів, нормативних актів, звітів, інструкцій (наприклад, українського законодавства).	Вторинний
3. Моделювання процесів	Створення графічних моделей «As-Is» (як є) та «To-Be» (як має бути) для виявлення невідповідностей та прогалин.	Вторинний
4. Аналіз існуючих рішень	Вивчення зовнішніх конкурентних або аналогічних систем (бенчмаркінг).	Вторинний
5. Прототипування	Швидке створення макетів інтерфейсу (мокапів) або функціональних прототипів для отримання раннього зворотного зв'язку від користувачів.	Первинний/Вторинний
6. Нагляд	Спостереження за роботою користувачів у їхньому природному робочому середовищі.	Первинний
7. Аналіз даних	Вивчення наявних даних (наприклад, статистики відмов, журналів подій, Google Analytics) для виявлення проблемних місць.	Вторинний
8. Аналіз поточних систем	Вивчення архітектури та функціональності систем, які будуть інтегровані або замінені.	Вторинний

Ефективність Комунікації

Комунікація – це процес передачі інформації між людьми, і вона є основою роботи аналітика.

Ефективність комунікації залежить від насиченості (багатства) каналу зв'язку, тобто від кількості доступних невербальних сигналів та можливості отримати зворотний зв'язок.

Нижня ліва дуга (Документальні способи): Передача інформації, де немає можливості отримати швидкий зворотний зв'язок.

Приклади: Записи на папері, аудіо- та відеозаписи.

Верхня права дуга (Інтерактивні способи): Способи комунікації, за допомогою яких можна отримати зворотний зв'язок.

За зростанням ефективності: Переписка по e-mail \to Телефонна розмова \to Розмова по відеозв'язку (відеоконференція) \to Жива зустріч \to Жива зустріч із застосуванням дошки/моделі (для фіксації пропозицій і рішень).

Висновок Алістера Коберна: Найбільш ефективна комунікація – це жива зустріч з використанням дошки для записів та моделювання.

Практичне застосування:

Потрібно прагнути обирати способи найефективнішої комунікації.

Якщо команда знаходиться в одному офісі, найефективнішою буде жива зустріч з використанням дошки.

Якщо команда розподілена по різних офісах (віддалена робота), найефективнішою буде відеоконференція з використанням інструментів для спільного моделювання (віртуальної дошки).

Питання для самоперевірки, тести для самоконтролю

Поясніть, чому термін «збір вимог» є некоректним і чому більш точним є термін «виявлення вимог».

Назвіть, які дві кінцеві цілі повинні бути досягнуті завдяки реалізації функціональних вимог.

Наведіть приклад первинного та вторинного методів виявлення вимог.

Який спосіб комунікації Алістер Коберн вважав найбільш ефективним і чому?

Як впливає «насиченість каналу зв'язку» на ефективність комунікації?

Тест (Виберіть одну правильну відповідь):

1. До якого типу методів виявлення вимог належить «Нагляд за роботою користувача»?

- а) Вторинний
- б) Первинний
- в) Аналітичний

2. Який канал комунікації є менш насиченим, ніж телефонна розмова?

- а) Відеоконференція
- б) Переписка по e-mail
- в) Жива зустріч із моделлю

3. З чого слід починати роботу над проєктом згідно з принципом «Починайте з кінця...»?

- а) З деталізації функціональних вимог
- б) З визначення кінцевої бізнес-цілі
- в) З аналізу поточної документації

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача: Ви працюєте над проєктом, де Бізнес-ціллю є «Зниження кількості помилок у процесі оформлення замовлень на 20%». Виберіть найбільш ефективний вторинний метод виявлення вимог для цієї цілі та поясніть свій вибір.

Відповідь-приклад: Аналіз даних. Необхідно проаналізувати логі системи та статистику помилок за останні 6–12 місяців, щоб точно визначити, на якому кроці та з якою частотою виникають помилки, перш ніж пропонувати рішення.

Задача: У вас запланована зустріч із ключовим стейкхолдером (Власником Продукту), який працює в іншому місті. Запропонуйте найбільш ефективний спосіб комунікації, щоб обговорити можливі структурні зміни у системі, і вкажіть необхідний інструмент.

Відповідь-приклад: Відеоконференція (Skype, Zoom, Google Meet) із застосуванням віртуальної дошки (Miro, FigJam) для спільного моделювання та фіксації рішень.

Додаткові задачі (Приклади)

Приклад (Інтерв'ювання): Сформулюйте три ключові питання, які бізнес-аналітик має поставити Власнику Продукту на першій зустрічі (Інтерв'ю), щоб з'ясувати Бізнес-вимоги до нової системи.

Приклад (Прототипування): Поясніть, як використання Прототипу (первинний/вторинний метод) допомагає мінімізувати ризик виявлення критичної помилки на пізніх етапах розробки.

Довідково-інформаційні дані для розв'язування задач

Фаза Життєвого Циклу	Виявлення Бізнес-вимог (Високий рівень)	Виявлення Детальних функціональних (Низький рівень)
Ключові методи	Інтерв'ювання, Аналіз документації, Моделювання процесів	Прототипування, Нагляд, Аналіз даних, Робочі сесії з командою

Бібліографічний список

1. Cockburn, Alistair. Agile Software Development: The Cooperative Game. Addison-Wesley Professional, 2004. (Книга, в якій описано концепцію насиченості каналу зв'язку та її вплив на ефективність комунікації, що є фундаментальним для роботи аналітика.)
2. ИБА. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute of Business Analysis, 2015. (Ключовий

документ, що містить повний перелік технік виявлення вимог та їхній детальний опис, включаючи Інтерв'ювання, Нагляд та Моделювання.)

3. Wiegers, Karl E., and Joyner, Laura. Software Requirements. 3rd Edition. Microsoft Press, 2013. (Описує важливість початку з кінця (визначення цілей) та містить практичні поради щодо використання первинних та вторинних джерел інформації.)

4. Hohmann, Luke. Beyond Software Architecture: Creating and Sustaining Winning Solutions. Addison-Wesley Professional, 2003. (Розширює концепцію комунікації та групової динаміки, необхідної для ефективного виявлення вимог у складних проєктах.)

Тема 6. Інтерв'ювання та аналіз вимог

Інтерв'ювання: Підготовка, Проведення, Дії після

Інтерв'ювання – найпоширеніший метод виявлення вимог, на який припадає до 60% виявленої інформації. Це складний процес, що вимагає ретельної підготовки та професіоналізму.

Етап 1. Підготовка

Зрозуміти тип інтерв'юваного та очікувані вимоги:

Замовник: Мета — виявити Бізнес-вимоги (бізнес-цілі).

Адміністратор/Ключовий користувач: Мета — виявити Користувацькі або Функціональні вимоги.

Тип інтерв'юваного визначає цілі та всю гілку запитань.

Експрес-аналіз контексту: Зрозуміти процес, метрики, його важливість.

Припущення про бажані цілі: Сформулювати гіпотезу про те, яких бізнес-цілей вас попросять досягти.

Припущення про функції та інтеграцію: Припустити, які функції матиме система і як вона інтегрується в поточну ІТ-екосистему (щоб уникнути дублювання інформації).

Експрес-аналіз подібних рішень: Подивитися на популярні рішення на ринку (бенчмаркінг).

Складання списку запитань: Підготувати детальний план інтерв'ю.

Етап 2. Проведення Зустрічі

Підготувати матеріали та технічне оснащення: Роздрукувати список запитань, перевірити презентацію, наявність великого екрана, ручок, блокнотів. Усе перевірити заздалегідь.

Прийти вчасно: Педантичність у цій професії — ознака професіоналізму.

Обозначити порядок денний (Повістку):

Уточнити тривалість зустрічі (наприклад, запропонувати 5-хвилинну перерву кожні 40 хвилин).

Озвучити план зустрічі та регламент (наприклад, просити уточнювати незрозумілі моменти одразу).

Представити учасників та визначити, хто веде бесіду.

Обов'язково попередити клієнта, якщо бесіда буде записуватися (на диктофон/відео).

Йти за питаннями, поглиблюючись у відповіді: Дотримуватися плану, але бути гнучким, щоб ставити додаткові запитання типу «чому?» або «як саме?».

Фіксувати (письмово або записувати на диктофон/відео).

Зачитати основні підсумки та закрити прогалини: Цей крок дозволяє уникнути багатьох додаткових зустрічей для уточнень.

Призначити наступний крок (за потреби).

Етап 3. Дії після Проведення Інтерв'ю

Оформити чистовик документа з підсумками зустрічі: Зробити це того ж дня і надіслати учасникам для підтверджень та уточнень. Додати в копію свого керівника.

Проаналізувати інформацію та скласти список уточнюючих/нових питань: Підготовлений список часто виявляється неповним.

При необхідності надіслати запит на уточнення.

Доповнити основні документи (чернетки) з вимогами.

Важливо: Не бійтеся вимагати письмового підтвердження від клієнтів щодо підсумків зустрічі. Це перестраховка для обох сторін, яка гарантує, що узгоджена інформація буде внесена в документи.

Аналіз Вимог

Аналіз вимог слідує за етапом виявлення. На цьому етапі зібрана інформація про вимоги є несистематизованою та частково дублюється.

Основна мета аналізу – систематизувати вимоги та позбутися дублюючих даних.

Для великих продуктів етап аналізу є критично важливим, оскільки його ігнорування призводить до великої кількості помилок у проєктуванні, що збільшує бюджет і терміни розробки.

Виділення Користувацьких Історій в Окремі Пакети

Перший етап аналізу – виділення користувацьких історій (User Stories) в окремі пакети вимог (за функціональною ознакою).

Головна мета: Спростити доступ до потрібних даних і керування ними. Кожен пакет містить усі варіанти використання, що стосуються певної функціональності.

Організація: Пакети спрощують редагування (автору не потрібно блокувати весь документ) та інформування користувачів про зміни (через секцію «Історія змін»).

Принцип формування: Пакети формуються з історій, які описують схожу діяльність або спосіб досягнення схожого результату, і зазвичай підпорядковані одному бізнес-вимозі.

Роботу над аналізом можна вважати закінченою, коли ви виділили всі користувачькі історії в окремі пакети, кожен з яких має зрозумілу назву та критерії включення.

Опис Вимог за Допомогою Варіантів Використання (Use Cases)

Концепція Варіантів використання (Use Cases), яка прийшла зі світу об'єктно-орієнтованого програмування, залишається фундаментальною для бізнес-аналізу.

Варіант використання (Use case) продукту описує послідовність взаємодії системи та зовнішньої дійової особи.

Дійова особа (Actor): Людина, інша система ПЗ або апаратний пристрій, що взаємодіє із системою для досягнення певної мети. Це роль користувача.

Приклади:

Я як слухач музики хочу відмітити трек як улюблений, щоб швидко його знайти.

Я як власник компанії хочу розмістити логотип на кожній сторінці сайту.

Діаграми Варіантів Використання

Діаграми варіантів використання (Use Case Diagrams) (у нотації UML) дозволяють отримати візуальне подання про вимоги користувачів.

Прямокутник показує межі системи. Лінії з'єднують кожну дійову особу (людинка) з варіантами використання (еліпсами), з якими ця особа взаємодіє.

Мета підходу Use Case:

Замість того, щоб питати користувачів, що повинна робити система, аналітик з'ясовує, які завдання користувач збирається розв'язувати за її допомогою. У кінцевий набір варіантів використання має входити вся бажана функціональність, що не виходить за межі проєкту.

Сценарій використання (Scenario): Окрема незалежна дія, яку дійова особа може виконати для отримання певного значимого результату.

Один Варіант використання може охоплювати кілька схожих Сценаріїв використання. Сценарій — це окремий приклад Варіанта використання.

Важливі Елементи Опису Варіанта Використання

Унікальний ідентифікатор (наприклад, UC-001).

Ім'я: Коротки опис у форматі «дієслово + об'єкт» (наприклад, «Розмістити замовлення»).

Короткий текстовий опис природною мовою.

Список попередніх умов: Що має бути виконано до початку розробки варіанта використання.

Вихідні умови: Стан системи після успішного завершення розробки.

Пронумерований список дій: Ілюстрація послідовності взаємодії Дійової особи та Системи.

Нормальний напрямок розвитку (Normal course): Це основний, головний успішний сценарій (сприятливий шлях). Наприклад, для варіанта використання «Запит хімікату» нормальний напрямок – запит хімікату, який є на складі.

Бібліографічний список

1. Wiegers, Karl E., and Joyner, Laura. Software Requirements. 3rd Edition. Microsoft Press, 2013. (Детально описує процес інтерв'ювання, включаючи підготовку, а також необхідність письмової фіксації та підтвердження вимог. Також є основним джерелом для опису аналізу вимог та їхньої систематизації.)

2. Cockburn, Alistair. Agile Software Development: The Cooperative Game. Addison-Wesley Professional, 2004. (Встановлює концепцію ефективності комунікації та роль живої зустрічі з дошкою у виявленні вимог.)

3. Kulak, Donna, and Guiney, Eamonn. Use Cases: Requirements in Context. 2nd Edition. Addison-Wesley Professional, 2012. (Класичне джерело для вивчення Варіантів використання, їхньої структури, опису елементів (нормальний напрямок, актори) та ролі в зборі вимог.)
4. Rosenberg, Doug. Applied UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. 3rd Edition. Pearson Education, 2009. (Навчальний посібник, що пояснює використання UML-діаграм Варіантів Використання для візуалізації вимог у контексті об'єктно-орієнтованого аналізу.)
5. ИБА. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute of Business Analysis, 2015. (Містить розділи про техніки виявлення та аналізу вимог, включаючи інтерв'ювання, документування та використання моделей.)

РОЗДІЛ 4. МОДЕЛЮВАННЯ ВИМОГ ТА БІЗНЕС-ПРОЦЕСІВ

(Скуріневська Л.В., д.е.н., старший дослідник, заступник начальника
Навчально-наукового центру оборонного менеджменту Національного
університету оборони України)

Тема 1. Деталізація варіантів використання: альтернативи та виключення

Альтернативні Напрямки Варіанта Використання

Альтернативний напрямок (Alternative course) – це інший допустимий сценарій з варіанта використання. Його також можуть називати вторинним сценарієм (secondary scenario).

Вторинні сценарії також задовольняють вихідним умовам варіанта використання і можуть призвести до успішного виконання завдання.

Це варіації розв'язання задачі або діалогової послідовності.

У певній точці нормальне напрямок може перейти в альтернативне, а потім повернутися в нормальне.

Блок-схема або UML-діаграма дозволяє візуально представити логічний розвиток складного варіанта використання, показуючи точку прийняття рішень та умови переходу.

Приклад: Альтернативний напрямок для варіанта використання «Замовити хімікат» – це «Замовити хімікат у постачальника». Кінцева мета одна — запит хімікату.

Розширення Варіантів Використання

Іноді альтернативний напрямок є настільки автономним, що може бути сам по собі окремим варіантом використання.

Якщо альтернативне напрямок є окремим варіантом використання, то нормальне напрямок можна розширити (використовується термін «extend»), включивши цей окремий варіант у нормальний потік. Наприклад, «Запросити

хімікат» може бути розширений варіантом використання «Виконати пошук по каталогах постачальника».

Якщо кілька варіантів використання мають спільні набори етапів (функціональність), щоб уникнути їх повторення, слід виділити окремий варіант із загальною функціональністю і вказати, що він включений (використовується термін «include») в інші варіанти як підваріант. Ця процедура аналогічна виклику спільної підпрограми в комп'ютерній програмі.

Виключення Варіантів Використання (Exceptions)

Виключення (Exception) — це умова, яка перешкоджає успішному завершенню варіанта використання.

Приклад: Для варіанта використання «Запросити хімікат» виключення — «Хімікату немає у продажу».

Якщо в процесі збору інформації не вказано, як обробляти виключення, можливі два шляхи:

Розробники пропонують найкращий, на їхню думку, спосіб обробки.

При виникненні непередбаченої умови відбудеться збій системи.

Важливо: Хоча Альтернативні напрямки можна відкласти до наступного випуску, крайньо важливо реалізувати виключення, оскільки вони запобігають збоєм системи. Багато недоліків кінцевого продукту пов'язані саме з відсутністю або некоректною роботою обробників виключень.

Указати умови виключень під час збору вимог — вірний спосіб створити стійкі до збоїв продукти.

Послідовність Варіантів Використання (Chaining)

Система повинна бути в такому стані після кожного варіанта використання, щоб користувач міг одразу ж приступити до наступного. Це означає, що вихідні умови одного варіанта використання повинні задовольняти попереднім умовам наступного за ним варіанта.

Приклад: На комерційному веб-сайті варіанти використання «Пошук по каталогу», «Додати товар у кошик» та «Оплатити товари в кошику» можуть виконуватися окремо (незалежні), але їх також можна об'єднати в один великий варіант використання «Придбати товар» (Chaining).

Попередні умови та вихідні умови визначають межі окремих варіантів, які можна пов'язати в ланцюжок для виконання об'ємної задачі.

Виявлення Варіантів Використання

Можна виявити варіанти використання кількома способами:

Спочатку визначити дійових осіб (Actors), а потім бізнес-процеси, в яких бере участь кожна особа.

Виразити бізнес-процеси в термінах сценаріїв, узагальнити сценарії у варіанти використання та визначити дійових осіб.

Визначити зовнішні події, на які система повинна реагувати, і співвіднести їх із дійовими особами та варіантами.

Визначити ймовірні варіанти використання на основі функціональних вимог.

Якщо будь-які вимоги неможливо простежити до якогось варіанта використання, слід визначити, чи потрібні вони взагалі.

Назва варіанта використання повинна вказувати на розв'язувану задачу, тому при формулюванні назв необхідно використовувати дієслово (наприклад, «проглянути», «надрукувати», «замовити»).

Фокус на меті, а не на дії: Якщо користувач пропонує варіант «сканувати штрихкод», аналітик має запитати: «Для чого ви скануєте штрихкод?». Справжній варіант використання може бути «Відправити хімікат у лабораторію», а сканування штрихкоду – лише один із його етапів.

Переваги Застосування Варіантів Використання

Пріоритезація: Полегшує визначення пріоритетів вимог. Вищий пріоритет мають ті функціональні вимоги, які створені на основі варіантів використання з вищим пріоритетом.

Вищий пріоритет призначається, якщо:
варіант описує основний бізнес-процес;
до нього часто звертаються користувачі;
його запросив привілейований клас користувачів;
він надає можливості, необхідні для відповідності законодавству (Compliance).

Об'єктна модель: Допомагає виявити важливі об'єкти предметної області та взаємозв'язки між ними. Розробники можуть перетворити варіанти використання в об'єктні моделі (діаграми класів, діаграми послідовностей).

Трасування: Дозволяє легко відстежити функціональні вимоги, дизайн, код та тести аж до їхніх витоків — побажань клієнта.

Порада: Не варто витратити багато часу на обговорення деталей варіантів використання, які не будуть реалізовані в найближчі місяці або роки, оскільки вони, ймовірно, зміняться через зміни в бізнес-процесах.

Застосування Варіантів Використання (Use Cases) є потужним інструментом, але вимагає дисципліни, щоб уникнути поширених пасток, які можуть призвести до плутанини та помилок у проектуванні системи.

Дуже важливо дотримуватися відповідного рівня абстракції при описі варіантів використання.

Варіант використання повинен залишатися на рівні «ЯК користувач, Я ХОЧУ досягти мети», тоді як деталі про кнопки, поля та конкретні технології належать до Функціональних вимог або Прототипів.

Наприклад, правильно сформульований елемент у Use Case: «Система пропонує дійовій особі перелік доступних опцій». Неправильно: «Система відображає випадаючий список з елементами "А", "Б" і "В"».

Ризик / Помилка	Опис Проблеми	Рекомендоване Рішення
1. Надто багато варіантів використання	Створення окремого варіанта для кожного можливого сценарію. Це ускладнює розуміння та керування вимогами.	Об'єднувати Нормальне Напрямок, Альтернативні Напрямки та Виключення в один варіант використання у вигляді сценаріїв. Кількість Use Cases має бути меншою, ніж кількість функціональних вимог.
2. Надто складні варіанти використання	Створення довгих, заплутаних описів, що охоплюють безліч логічних розгалужень.	Виділити один успішний шлях як Нормальне Напрямок. Інші успішні шляхи визначити як Альтернативні напрямки, а невдалі — як Виключення. Альтернативи мають бути короткими та зрозумілими.
3. Включення деталей інтерфейсу	Опис того, ЯК це буде виглядати на екрані, а не ЩО система повинна робити.	Варіанти використання мають описувати дії, а не дизайн. Використовуйте формулювання «Система надає вибір», а не «Система відображає розкритий список». Візуалізацію (начерки екранів) слід використовувати окремо.
4. Включення визначень даних	Вбудовування визначень елементів даних та структур безпосередньо в опис Use Cases.	Це призводить до повторів і рассинхронізації даних. Визначення даних слід збирати в єдиний Словник Даних (Data Dictionary) для всього проєкту.
5. Варіанти використання, незрозумілі користувачам	Використання занадто технічної мови або відсутність зв'язку з реальними бізнес-задачами користувача.	Слід складати вимоги з точки зору клієнтів, а не системи. Попросіть клієнтів перевірити опис. Складні Use Cases, ймовірно, зайві.
6. Нові бізнес-процеси	Спроба описати вимоги для системи, яка підтримуватиме новий процес, якого ще не існує.	Існує високий ризик створити неефективний процес. Необхідно спочатку виконати модернізацію (реінжиніринг) бізнес-процесу та затвердити його логіку, а вже потім описувати нову інформаційну систему.

Бібліографічний список

1. Kulak, Donna, and Guiney, Eamonn. Use Cases: Requirements in Context. 2nd Edition. Addison-Wesley Professional, 2012. (Основне джерело, яке детально описує структуру Варіанта Використання, включаючи Альтернативні напрямки та Виключення, а також концепції include та extend.)
2. Wiegers, Karl E., and Joyner, Laura. Software Requirements. 3rd Edition. Microsoft Press, 2013. (Пояснює важливість обробки виключень для створення стійких до збоїв продуктів, а також методи виявлення варіантів використання через дійових осіб та бізнес-процеси.)
3. Jacobson, Ivar, Christerson, Magnus, Jonsson, Patrik, and Övergaard, Gunnar. Object-Oriented Software Engineering: A Use Case Driven Approach. Addison-Wesley Professional, 1992. (Фундаментальна робота, що ввела концепцію Варіантів Використання в об'єктно-орієнтовану розробку.)
4. IIBA. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute of Business Analysis, 2015. (Описує техніку Моделювання Варіантів Використання (Use Case Modeling) та її застосування для трасування та пріоритезації вимог.)

Тема 2. Опис вимог за допомогою процесного підходу

Процесний підхід до управління передбачає об'єднання функцій окремих підрозділів, які пов'язані з конкретними бізнес-процесами, у цілісний ланцюжок дій. Це дозволяє керівництву перерозподілити функції та делегувати повноваження щодо прийняття оперативних рішень власнику процесу.

Процес у загальному сенсі – це організована діяльність, або комплекс видів діяльності, у якому використовуються ресурси для перетворення входів у виходи.

Для здійснення процесу необхідно:

Організувати процес – побудувати його в просторі та часі.

Забезпечити виконання процесу ресурсами.

Визначити виконавця процесу.

Задokumentувати процес: скласти технологію, робочі та користувацькі інструкції.

Для забезпечення узгодженості входів та виходів процесів, а також їхньої ідентифікації, необхідно організувати управління, яке робить можливою взаємодію між процесами.

Основні Поняття Процесного Підходу

Бізнес-процес – це сукупна послідовність дій (підпроцесів), яка перетворює входи у виходи. Вона спрямована на отримання заданого результату, цінного для організації (продукту або послуги для споживачів). Це регулярно повторювана послідовність взаємопов'язаних заходів (операцій, процедур, дій), при виконанні яких створюється цінність для споживача.

Власник Бізнес-процесу – посадова особа, яка має в своєму розпорядженні ресурси, необхідні для виконання процесу, та несе відповідальність за його результат. Власник керує процесом і є його невід'ємною частиною.

Вхід (Input) – матеріальний або інформаційний продукт, який у ході виконання процесу перетворюється на вихід. Вхід завжди повинен мати свого постачальника.

Приклади входів: Сировина, матеріали, документація, заявка від клієнта.

Вихід (Output) – матеріальний або інформаційний продукт чи послуга, яку споживають клієнти (зовнішні або внутрішні). Вихід процесу повинен задовольняти заданим вимогам і мати споживача, заради якого процес виконується.

Приклади виходів: Готова продукція, звітна документація, оброблена заявка.

Споживач – може бути зовнішнім (не входить до складу організації) або внутрішнім (перебуває в рамках організації). Споживач задає вимоги до процесу.

Ресурс Бізнес-процесу – матеріальний або інформаційний об'єкт, який постійно використовується для виконання процесу, але не виступає як вхід (не перетворюється у вихід).

Приклади ресурсів: Обладнання, програмне забезпечення (ПЗ), персонал, інфраструктура, транспорт, зв'язок.

Примітка: Входи, виходи та ресурси повинні позначатися іменниками, оскільки це матеріальні або інформаційні об'єкти.

Виконавці Бізнес-процесу – команда фахівців із різних функціональних областей (крос-функціональна команда), що виконує дії процесу. Вони більш орієнтовані на кінцевий результат, ніж виконавці окремих функцій у функціональному підході, оскільки їхня мотивація (бонуси) прив'язана до отримання кінцевого результату.

Важливо: Бізнес-процеси використовуються не лише для комерційних цілей. Процеси взаємодії персоналу всередині компанії (наприклад, оформлення відпустки, прийом на роботу) також є бізнес-процесами.

Способи Опису Бізнес-процесів

За форматом подання засоби для опису бізнес-процесів поділяються на текстові, табличні та графічні.

Текстовий Опис:

Недолік: Слабка формалізація, відсутність єдиних стандартів.

Приклад (Фрагмент оформлення замовлення):

Старт процесу – дзвінок клієнта. Знайти клієнта у базі.

Якщо клієнта немає, зареєструвати нового.

Уточнити потребу та перевірити наявність товару.

Табличний Формат:

Дозволяє структурувати опис та виявити основні кроки, входи, виходи та відповідальних виконавців для кожної операції.

Найкраще використовувати для опису простих лінійних процесів або для збору інформації перед графічним моделюванням.

Графічні Описи (Діаграми):

Користуються найбільшою популярністю завдяки наочності та зручності.

Використовуються різні нотації (мови) графічного опису, наприклад BPMN (Business Process Modeling Notation).

Блок-схема вважається найпопулярнішим способом, оскільки вона наочна та дозволяє швидко змінювати інформацію.

Питання для самоперевірки, тести для самоконтролю

У чому полягає основна перевага процесного підходу порівняно з функціональним у контексті виконання завдань?

Назвіть, чим відрізняється Вхід бізнес-процесу від його Ресурсу. Наведіть приклад для процесу «Приготування кави».

Яку роль відіграє Власник Бізнес-процесу і за що він несе відповідальність?

Який спосіб опису бізнес-процесів є найбільш популярним і чому?

Чому опис Входів, Виходів та Ресурсів має бути іменниками?

Тест (Виберіть одну правильну відповідь):

1. До якого елементу бізнес-процесу належить «Заявка на відпустку», що подається співробітником у відділ кадрів?

а) Ресурс

б) Вхід

в) Вихід

2. Яка нотація найчастіше використовується для графічного опису бізнес-процесів?

а) UML (Use Case)

б) BPMN

в) DFD

3. Головна мета бізнес-процесу — це:

а) Оптимізація використання ресурсів.

б) Підвищення кваліфікації виконавців.

в) Отримання результату, цінного для споживача.

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача: Визначте Вхід, Вихід та Ресурс для бізнес-процесу «Оформлення кредиту фізичній особі» в банку.

Відповідь-приклад: Вхід: Заявка клієнта на кредит + необхідні документи. Вихід: Укладений кредитний договір (або відмова). Ресурс: Програмне забезпечення для скорингу, кредитний менеджер.

Задача: Сформулюйте один крок у табличному форматі для процесу «Реєстрація нового клієнта», вказавши в ньому виконавця та умову.

Відповідь-приклад (Крок 2): Операція: Перевірка унікальності даних клієнта. Виконавець: Менеджер з продажу. Вхід: Дані клієнта. Вихід: Статус «Клієнт унікальний» (або «Дублікат»). Умова: Якщо клієнт знайдений — перейти до процесу «Оновлення даних».

Додаткові задачі (Приклади)

Приклад (Власник процесу): У процесі «Закупівля нового ІТ-обладнання» візьмуть участь: Ініціатор (співробітник), Відділ закупівель, Бухгалтерія та ІТ-відділ. Поясніть, хто з них, найімовірніше, буде Власником процесу і чому.

Приклад (Текстовий опис): Переробіть наданий у тексті приклад текстового опису процесу оформлення замовлення у формат, який можна легко перетворити на табличний опис (додавши стовпці: Виконавець, Рішення).

Довідково-інформаційні дані для розв'язування задач

Характеристика	Процес (Business Process)	Процедура (Procedure)
Масштаб	Охоплює діяльність кількох підрозділів (крос-функціональний)	Детальна послідовність дій всередині одного процесу/відділу
Мета	Створення кінцевої цінності для споживача	Виконання конкретної операції
Відповідальність	Власник процесу	Керівник підрозділу/Виконавець
Приклад	Процес «Відвантаження товару клієнту»	Процедура «Оформлення накладної у системі»

Елементи Табличного Опису Бізнес-процесу

Назва Операції	Виконавець	Вхід (Документ/Інфо)	Вихід (Документ/Інфо)	Умова переходу
Звернення клієнта	Клієнт	Запит (усно/письмово)	Запит зареєстрований	Зареєстровано
Перевірка наявності	Менеджер	Запит, Складські дані	Підтвердження наявності	Наявність > 0

Бібліографічний список

1. Hammer, Michael, and Champy, James. Reengineering the Corporation: A Manifesto for Business Revolution. HarperBusiness, 1993. (Класична робота, що утвердила важливість процесного підходу та концепції реінжинірингу.)
2. Rummler, Geary A., and Brache, Alan P. Improving Performance: How to Manage the White Space in the Organization Chart. Jossey-Bass, 1995. (Описує необхідність управління «білим простором» між функціональними підрозділами за допомогою бізнес-процесів.)
3. BPMN 2.0.2: Business Process Model and Notation. OMG, 2013. (Офіційна специфікація найбільш популярної графічної нотації для моделювання бізнес-процесів.)

4. ИБА. А Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute of Business Analysis, 2015. (Описує техніки моделювання процесів та визначає ключові елементи: входи, виходи, механізми (ресурси), учасники та контрольні точки.)

Тема 3. Види бізнес-процесів та нотації моделювання

Класифікація Бізнес-процесів за Функціональним Навантаженням

Бізнес-процеси найчастіше класифікують, ґрунтуючись на функціональному навантаженні та їхньому місці в організаційній структурі компанії. Найпоширеніша класифікація поділяє процеси на три основні види:

Основні процеси (Core/Primary Processes):

Суть: Безпосередньо пов'язані з виробництвом головних продуктів (товарів чи послуг) компанії. Це процеси, які додають продукту вартість, створюють продукт, за який клієнт готовий платити, і генерують доходи.

Стратегічна важливість: Вони є стратегічно важливими, специфічними та унікальними (їх складно скопіювати).

Приклади: Обробка та виконання замовлення; розробка, проєктування та дизайн продукту; виробництво, монтаж.

Допоміжні або Обслуговуючі процеси (Support Processes):

Суть: Забезпечують виконання необхідних робіт для підтримки ключових (основних) процесів, але не представляють безпосередньої цінності для клієнта. Це адміністративна та господарська підтримка.

Приклади: Обробка даних; технічне обслуговування (ТО); управління логістикою (внутрішня); адміністративні процеси; управління ІТ-інфраструктурою.

Керуючі процеси (Management Processes):

Суть: Містять завдання та діяльність, спрямовані на довгостроковий розвиток компанії, реалізацію її цілей та стратегічне управління.

Приклади: Стратегічне планування та розвиток; довгострокове та середньострокове планування; корпоративне управління; управління персоналом (HR); управління якістю; інвестиційне планування; мотивація персоналу.

Нотації для Моделювання Бізнес-процесів

Для побудови моделей бізнес-процесів використовуються спеціальні методики та мови моделювання – нотації.

Модель Процесу – це візуальне представлення послідовного потоку та логіки управління набором взаємопов'язаних заходів або дій.

Нотація – це система умовних позначень (знаків) і правил їхнього використання для побудови моделей, прийнята в конкретній методології.

Залежно від типу моделі (наприклад, фокус на потоці робіт, організаційній структурі або даних), яку необхідно створити, використовуються відповідні нотації.

BPMN (Business Process Model and Notation) є однією з найпоширеніших нотацій для блок-схемного опису бізнес-процесів, оскільки вона наочна, масштабована та міжнародно стандартизована.

Питання для самоперевірки, тести для самоконтролю

Назвіть три основні види бізнес-процесів, класифікованих за функціональним навантаженням.

Яка ключова ознака відрізняє Основний процес від Допоміжного?

Наведіть приклад Керуючого процесу.

Що таке Нотація у контексті моделювання бізнес-процесів?

Чому процес «Управління якістю» належить до Керуючих, а не до Основних?

Тест (Виберіть одну правильну відповідь):

1. До якого виду процесів належить процес «Нарахування заробітної плати»?

- а) Основний
- б) Допоміжний (Обслуговуючий)
- в) Керуючий

2. Яка характеристика є ключовою для Основних процесів?

- а) Забезпечують внутрішню адміністративну підтримку.
- б) Додають продукту вартість, яку клієнт готовий оплачувати.
- в) Визначають довгострокову стратегію компанії.

3. Що таке Модель Процесу?

- а) Тільки текстовий опис послідовності дій.
- б) Візуальне представлення послідовного потоку та логіки управління діями.
- в) Набір функціональних вимог, згрупованих за відділами.

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача: Компанія займається продажем програмного забезпечення. Визначте вид наступних процесів: 1) Виставлення рахунку клієнту; 2) Розробка нового функціоналу ПЗ; 3) Щорічне бюджетування.

Відповідь-приклад: 1) Допоміжний (Фінансова підтримка). 2) Основний (Створення продукту). 3) Керуючий (Стратегічне планування).

Задача: Поясніть, чому «Управління персоналом (HR)» може мати елементи як Допоміжного, так і Керуючого процесів.

Відповідь-приклад: Керуючий: Розробка стратегії найму та мотивації (довгостроковий розвиток). Допоміжний: Обробка відпусток та ведення кадрової документації (адміністративна підтримка).

Додаткові задачі (Приклади)

Приклад (Нотації): Поясніть, чому BPMN краще підходить для моделювання крос-функціональних Основних процесів, ніж звичайна блок-схема (Flowchart).

Приклад (Цінність): Сформулюйте, яку цінність для зовнішнього клієнта створює процес «Обробка даних» (Допоміжний процес) у банку, навіть якщо клієнт за нього безпосередньо не платить.

Довідково-інформаційні дані для розв'язування задач

Вид Процесу	Взаємодія з Клієнтом	Створення Прямої Цінності	Приклади Діяльності
Основний	Висока/Пряма	Так	Виготовлення товару, Обслуговування клієнта
Допоміжний	Низька/Непряма	Ні (забезпечує ефективність)	Бухгалтерія, ІТ-підтримка, Склад
Керуючий	Відсутня/Стратегічна	Ні (забезпечує розвиток)	Планування, Аудит, Корпоративне управління

Популярні Нотації Моделювання

Нотація	Тип Фокусу	Призначення
BPMN (Business Process Model and Notation)	Потік робіт та події	Моделювання та аналіз бізнес-процесів (стандарт де-факто)
UML (Activity Diagram)	Послідовність дій	Деталізація логіки виконання функціональних вимог
EPC (Event-driven Process Chain)	Події та функції	Моделювання у методології ARIS, часто використовується в SAP-середовищі

Бібліографічний список

1. Hammer, Michael. Beyond Reengineering: How the Process-Centered Organization is Changing Our Work and Our Lives. HarperBusiness, 1996. (Розширений аналіз класифікації процесів на Основні, Допоміжні та Керуючі, орієнтований на створення організаційної структури навколо процесів.)
2. Harmon, Paul. Business Process Change: A Guide for Business Process Management. 3rd Edition. Morgan Kaufmann, 2014. (Детальне керівництво з

класифікації бізнес-процесів та їхньої важливості для корпоративного управління.)

3. OMG. Business Process Model and Notation (BPMN) Version 2.0.2. OMG, 2013. (Офіційна специфікація нотації BPMN, яка є ключовим інструментом для візуального опису процесів.)

4. ИБА. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute of Business Analysis, 2015. (Описує класифікацію вимог та процесів, що використовуються бізнес-аналітиками для контекстуалізації функціональних вимог.)

Тема 4. Структурні моделі: нотація IDEF0

Модель Складу та Нотація IDEF

Модель складу представляє інформацію про внутрішній зміст системи, описуючи, з яких підсистем та елементів вона складається. Побудова моделі відбувається поетапно, через деталізацію системи від найбільших підсистем до їхніх функціональних елементів. Розбиття системи на частини відповідає обраній точці зору та меті використання моделі.

Для побудови структурних моделей найкраще підходить нотація IDEF (скорочення від англ. Integrated DEFinition) — стандартизована система бізнес-моделювання, представлена цілим сімейством методик. Найбільш поширеною є IDEF0.

Методологія та Графічна Нотація IDEF0

IDEF0 — це методологія та графічна нотація для опису функцій системи. Вона визначає одиницю діяльності як функцію, що характеризується чотирма основними атрибутами (модель ICOM):

Вхід (Input): Ресурси, дані та інші складові, які надходять до функції та необхідні для її успішного виконання.

Управління (Control): Правила, обмеження та ліміти, в яких функція існує, або правила застосування функції перетворення входів у виходи.

Механізм (Mechanism): Деякий виконуючий ресурс (персонал, обладнання, ПЗ). Те, що фактично застосовує функцію до входу для отримання виходу.

Вихід (Output): Результати виконання функції.

Основне призначення IDEF0 — формалізація та опис бізнес-процесів організації як логічних співвідношень між роботами з акцентом на співпідпорядкованість об'єктів, а не на часову послідовність (як у BPMN). IDEF0 розглядає головним чином склади функцій та їхній взаємозв'язок.

Ключові Особливості Моделей IDEF0

Мінімалістичний набір елементів: Дозволяє описати функцію як «чорний ящик» (оцінка входів, виходів, управління та механізмів), відокремлюючи важливі характеристики від деталей реалізації.

Контекст та Точка Зору: Методологія дозволяє наочно представити систему як єдине ціле на діаграмі верхнього рівня — контекстній діаграмі (рівень A-0).

Контекстна діаграма фіксує контекст і точку зору на систему.

Мета і Точка зору є важливими: модель для розробки ІС буде відрізнятися від моделі для оптимізації процесів.

Ієрархія та Наскрізна Збереженість: Зручна можливість представити ієрархію процесів на будь-якій кількості рівнів деталізації за допомогою декомпозиції. Цей підхід забезпечує наскрізну появу входів, механізмів та управління на різних рівнях, зберігаючи цілісне сприйняття потоку перетворення.

Символи Нотації IDEF0

Як і належить графічній нотації, IDEF0 має простий, але ємний набір символів.

Функціональний Блок (Function/Process):

Позначається прямокутним блоком.

Всередині розміщується найменування, виражене активним дієсловом, дієслівним зворотом або віддієслівним іменником (наприклад, «Оформити замовлення»).

Може мати номер для ідентифікації.

Стрілка (Інтерфейсна Дуга): Стрілка, яка несе стандартне значення залежно від сторони блоку, до якої вона приєднана:

Входить у ліву сторону — Входи (Input).

Входить у блок зверху — Управління (Control) (Правила).

Залишає процес справа — Виходи (Output).

Підключена до нижньої сторони — Механізми (Mechanism) (Виконавці/Ресурси).

Декомпозиція

Декомпозиція – це ключовий прийом в IDEF0, що дозволяє розділити ціле на частини та замінити розв'язання однієї великої задачі серією менших.

Декомпозиція дозволяє «прокачати» будь-яку частину процесу до будь-якого рівня деталізації, необхідного для аналізу.

Типи Діаграм в IDEF0 (з точки зору декомпозиції):

Контекстна діаграма (A-0): Діаграма з одним функціональним блоком, що представляє найбільш укрупнене бачення досліджуваної системи.

Діаграми Декомпозиції: Дочірні діаграми, які є більш деталізованим розрізом контекстної діаграми, де розглядаються складові процесу.

Питання для самоперевірки, тести для самоконтролю

Назвіть чотири основні атрибути, які характеризують функцію в нотації IDEF0 (модель ICOM).

У чому полягає основна відмінність фокусу IDEF0 від, наприклад, BPMN?

Поясніть, що таке Контекстна діаграма в IDEF0 і як вона позначається.

Якою частиною функціонального блоку в IDEF0 є Правила чи Обмеження?

Чому назва функціонального блоку в IDEF0 має бути виражена активним дієсловом?

Тест (Виберіть одну правильну відповідь):

1. До якої сторони функціонального блоку IDEF0 приєднуються стрілки, що представляють персонал або програмне забезпечення?

- а) Ліва (Вхід)
- б) Верхня (Управління)
- в) Нижня (Механізм)

2. Яка мета є ключовою при фіксації «Точки зору» та «Мети» моделювання в IDEF0?

- а) Збільшити кількість функціональних блоків.
- б) Забезпечити наявність усіх можливих сценаріїв.
- в) Визначити відповідні області та рівень необхідної деталізації.

3. Що описує IDEF0 насамперед?

- а) Часову послідовність дій.
- б) Склад функцій організації та їхній взаємозв'язок.
- в) Об'єктно-орієнтовані класи системи.

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача: Для функції «Обробка скарги клієнта» визначте та розподіліть такі елементи за ICOM-атрибутами IDEF0: Оператор кол-центру, Регламент розгляду скарг, Відповідь клієнту, Зареєстрована скарга.

Відповідь-приклад: Вхід: Зареєстрована скарга. Управління: Регламент розгляду скарг. Механізм: Оператор кол-центру. Вихід: Відповідь клієнту.

Задача: Ви пояснюєте розробнику, що таке «Механізм» в IDEF0. Наведіть аналогію «чорного ящика» (функціонального блоку), що перетворює «Вхід» у «Вихід».

Відповідь-приклад: Якщо функція – це «Приготування торта», то Вхід – це інгредієнти, Вихід – готовий торт. Механізм – це кухар, духовка, рецепт. Але в IDEF0 рецепт був би Управлінням.

Додаткові задачі (Приклади)

Приклад (Точка Зору): Поясніть, як буде відрізнятися Контекстна діаграма для підприємства, якщо мета моделювання – «Оптимізація виробничого циклу» проти «Розробка системи мотивації персоналу». Які Входи та Виходи будуть ключовими у кожному випадку?

Приклад (Декомпозиція): Наведіть приклад, як збереження наскрізного Управління (наприклад, «Законодавство про захист даних») на різних рівнях декомпозиції допомагає бізнес-аналітику уникнути помилок.

Довідково-інформаційні дані для розв'язування задач

Елемент	Приєднаний до...	Роль та Значення	Приклад
I (Input)	Ліва сторона	Ресурси, що перетворюються на вихід.	Заявка, Сировина
C (Control)	Верхня сторона	Правила, Обмеження, Ліміти, що керують перетворенням.	Регламент, Бюджет, Законодавство
O (Output)	Права сторона	Результат виконання функції.	Оброблена заявка, Готовий продукт
M (Mechanism)	Нижня сторона	Виконуючі ресурси (хто/що виконує функцію).	Співробітник, ПЗ, Обладнання

Зв'язок BPMN та IDEF0

Характеристика	BPMN (Business Process Model and Notation)	IDEF0 (Integrated DEFinition)
Основний фокус	Послідовність дій, час, події	Склад функцій та їхні логічні зв'язки (ієрархія)
Ключовий елемент	Діяльність, Події, Шлюзи	Функціональний блок (ICOM)
Найкраще підходить для	Деталізації потоку робіт, автоматизації	Структурного аналізу, високорівневої архітектури

Бібліографічний список

1. NIST. Integration Definition for Function Modeling (IDEF0). Federal Information Processing Standards Publication (FIPS PUB) 183, 1993. (Офіційний стандарт нотації IDEF0, що детально описує її методологію та елементи ICOM.)
2. Rumbaugh, James, et al. Object-Oriented Modeling and Design. Prentice Hall, 1991. (Хоча це не основне джерело IDEF0, воно ілюструє важливість структурного та функціонального моделювання, де IDEF0 є ключовим інструментом для опису функціональної ієрархії.)
3. Scheer, August-Wilhelm. ARIS – Business Process Modeling. Springer, 2000. (Описує інтеграцію структурних моделей, подібних до IDEF0, у ширші фреймворки моделювання, підкреслюючи важливість ICOM-атрибутів.)
4. ИБА. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute for Business Analysis, 2015. (Описує використання структурного моделювання та його роль у виявленні вимог та архітектури рішення.)

Тема 5. Деталізація та додаткові елементи нотації IDEF0

Правила Декомпозиції та Структури Діаграм IDEF0

Діаграми IDEF0 різних рівнів деталізації повинні бути послідовними та вкладеними (як матрешка). При переході на новий рівень глибина деталізації має сприяти розумінню складності об'єкта.

Рекомендації щодо структури діаграми:

Кількість Функціональних Блоків (ФБ): На одній діаграмі рекомендується розміщувати не більше 5-7 ФБ. Якщо потрібно більше, слід повернутися до батьківської діаграми та створити проміжну діаграму, щоб розділити блоки на дві або більше дочірніх діаграм.

Кількість Інтерфейсних Дуг (Стрілок): До кожного ФБ рекомендується приводити не більше 2-3 Входів, Виходів, Механізмів та Управлінь. Інакше необхідно перевірити рівень деталізації дуг або створити додатковий «шар» моделі.

Розташування Блоків: На дочірніх діаграмах підпроцеси мають бути пов'язані (наприклад, вихід одного ФБ є входом іншого). Найважливіші блоки розміщують ближче до лівого верхнього кута, а менш значущі — ближче до правого нижнього кута (за спаданням значущості).

Збереження Зв'язків: На дочірній діаграмі декомпозиції необхідно використовувати ті самі стрілки Входів, Виходів, Механізмів та Управлінь, які були згадані на батьківській діаграмі.

Розгалуження Стрілок: Щоб використати одну стрілку (Вхід, Управління, Механізм) для кількох ФБ, її необхідно розгалужувати (ділити на стільки закінчень, скільки блоків її використовують).

Споживач або Виробник: Усі інтерфейсні дуги повинні мати свого споживача або виробника, тобто з'єднуватися з функціональним блоком.

Етапи Побудови Моделі IDEF0

Визначення Контексту та Меті: Моделювання починається з визначення контексту та мети, з якою буде використана діаграма.

Перший Рівень (А-0): Контекстна Діаграма: Зображується діаграма з одним-єдиним ФБ, що описує процес у цілому. На цьому рівні наводяться

основні вхідні ресурси, механізми, контролю та вихідні результати, важливі для аналізу.

Другий Рівень (An): Декомпозиція: Створюється діаграма, що декомпозує основний процес на кілька ФБ (5–7), які описують загальні напрямки процесу. Допустимо залишити лише один ФБ, якщо в деталях потрібно описати лише один напрямок.

Подальші Рівні (An.m): Деталізація: Для сфер процесу, що залишаються у фокусі, створюються окремі діаграми третього та наступних рівнів. Крок декомпозиції має бути рівномірним, щоб уникнути дисонансу (не варто намагатися описати одну сферу відразу в розрізі всіх найдрібніших задач).

Завершення: Побудова діаграм продовжується доти, доки отриманий набір не покриє всю сферу дослідження в достатній мірі для досягнення мети побудови моделі.

Порада: При створенні моделі керуйтеся не лише правилами нотації, а й логікою, кращими практиками та потребами цільової аудиторії, яка буде читати діаграми.

Додаткові Елементи IDEF0

Крім основного набору символів, в IDEF0 є додаткові елементи, які підвищують гнучкість моделювання:

Тунельована Стрілка (Tunnelling): Стрілка-вхід або вихід, що означає, що дані, які передаються нею, не розглядаються на батьківській та/або дочірній діаграмі.

Стрілка в тунелі при приєднанні до блоку: дані не обов'язкові для наступного рівня декомпозиції.

Стрілка в тунелі на вільному кінці: дані відсутні на батьківській діаграмі (вперше з'являються тут).

Зовнішнє Посилання (External Reference): Позначає місце, сутність або суб'єкт, що знаходяться за межами моделі (джерело або приймач стрілки). Зображується у вигляді квадрата з найменуванням.

Міждіаграмне Посилання (Inter-Diagram Reference): Позначає іншу діаграму (не дочірню/батьківську). Використовується для позначення переходу стрілки на діаграму іншого процесу без відображення стрілки на вищерозміщеній діаграмі.

Процес-Посилання (Process Link): Веде на типову модель процесу (найбільш часто повторювані дії).

Сноска або Коментар (Footnote/Comment): Виносний елемент для додавання коментарів у вільному тексті, пов'язаних з конкретним елементом.

Текст або Опис (Text/Description): Вільний текстовий коментар до елемента або діаграми в цілому без сноски.

Застереження: Використовуючи додаткові елементи, не перестарайтеся. Мінімалізм IDEF0 є її перевагою; чим прозорішою і читабельнішою буде модель, тим успішнішою буде ваша робота.

Питання для самоперевірки, тести для самоконтролю

Скільки функціональних блоків рекомендується розміщувати на одній діаграмі IDEF0? Що робити, якщо потрібно більше?

Поясніть, що означає тунельована стрілка, яка приєднана до блоку.

Опишіть, як мають розташовуватися функціональні блоки на дочірній діаграмі декомпозиції за їхньою значущістю.

Яку роль відіграє Зовнішнє Посилання?

Як вирішується питання використання однієї стрілки Входу для кількох функціональних блоків на одній діаграмі?

Тест (Виберіть одну правильну відповідь):

1. Максимальна кількість Входів, Виходів, Механізмів та Управлінь, рекомендована до одного функціонального блоку:

- а) Не більше 7
- б) Не більше 2–3
- в) Не обмежена

2. Якщо на діаграмі другого рівня (декомпозиції) не використовується стрілка Управління, яка була на батьківській діаграмі, то на вільному кінці цієї стрілки на батьківській діаграмі слід проставити:

- а) Міждіаграмне посилення.
- б) Зовнішнє посилення.
- в) Тунель (на вільному кінці).

3. Який елемент IDEF0 використовується для позначення типових, часто повторюваних дій?

- а) Тунельована стрілка
- б) Сноска
- в) Процес-посилення

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача: У процесі «Оформлення відпустки» є блок «Перевірка залишку днів відпустки». На батьківській діаграмі є Управління «Трудовий кодекс». Поясніть, чи потрібно це Управління показувати на дочірній діаграмі, якщо воно не стосується всіх функціональних блоків?

Відповідь-приклад: Так, потрібно. На дочірній діаграмі «Трудовий кодекс» має бути відображений і розгалужений лише до тих ФБ, де він дійсно є керуючим чинником (наприклад, до блоку «Перевірка залишку днів»), відповідно до правила наскрізної збереженості зв'язків.

Задача: Ви створюєте діаграму декомпозиції процесу «Продаж товару», який складається з 8 підпроцесів. Як вам слід розподілити їх по діаграмах, дотримуючись рекомендацій?

Відповідь-приклад: Слід переглянути батьківську діаграму. Створити проміжну діаграму (наприклад, рівень A1) з 3-4 укрупненими блоками (наприклад, «Залучення клієнта», «Укладання угоди», «Відвантаження»). Потім декомпонувати кожен із цих 3-4 блоків на окремі дочірні діаграми, дотримуючись ліміту 5-7 ФБ.

Додаткові задачі (Приклади)

Приклад (Читабельність): Поясніть, чому діаграма з 15 функціональними блоками стає нечитабельною, навіть якщо всі стрілки ISOM логічно коректні.

Приклад (Тунелювання): Поясніть, як можна використовувати тунельовану стрілку (на вільному кінці), щоб не перевантажувати контекстну діаграму A-0 несуттєвою інформацією, яка важлива лише на рівні деталізації.

Довідково-інформаційні дані для розв'язування задач

Принцип	Опис	Призначення
I-O Зв'язок	Вихід одного ФБ повинен бути Входом іншого ФБ.	Створення ланцюжка додавання вартості та залежності.
ISOM Ліміт	I, C, O, M до одного ФБ.	Забезпечення читабельності та рівномірної деталізації.
Ліміт ФБ	Кількість ФБ на одній діаграмі.	Збереження зрозумілості та запобігання перевантаженню.
Тунелювання	Позначає, що стрілка ігнорується на одному з рівнів ієрархії.	Запобігання перевантаженню моделі деталями, несуттєвими для рівня.

Бібліографічний список

1. NIST. Integration Definition for Function Modeling (IDEF0). Federal Information Processing Standards Publication (FIPS PUB) 183, 1993. (Офіційний стандарт, що містить детальні рекомендації щодо кількості блоків, стрілок та правил декомпозиції, включаючи тунелювання.)

2. Marca, David A., and McGowan, Clement L. IDEF0: Business Process and Enterprise Modeling. The Business Process & Enterprise Modeling Institute, 1993.

(Ключове джерело, що пояснює практичні аспекти використання IDEF0, включаючи необхідність дотримання лімітів ієрархії та правила наскрізної збереженості (parent-child connectivity) інтерфейсних дуг.)

3. Scheer, August-Wilhelm. ARIS – Business Process Modeling. Springer, 2000. (Описує, як принцип декомпозиції IDEF0 інтегрується у фреймворки архітектури підприємства.)

4. ИБА. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute for Business Analysis, 2015. (Описує використання структурного моделювання та його роль у підтримці читабельності та узгодженості між рівнями деталізації.)

Тема 6. Моделі потоків робіт: нотація eEPC

Сутність та Призначення Нотації eEPC

eEPC (extended Event-driven Process Chain — «розширена подієва процесний ланцюжок») — це одна з частин загальної методології ARIS (Architecture of Integrated Information Systems), що використовується для представлення організації з позиції її бізнес-процесів.

Нотація eEPC базується на використанні подій як станів процесу і функцій як кроків процесу.

Призначення: Деталізований опис бізнес-процесу низького рівня у форматі workflow (послідовність задач), що виконується в рамках одного або кількох підрозділів.

Мета: Відображати процес з усіма варіаціями його виконання для реалізації імітаційного аналізу (симуляції) та надання ширших можливостей для оптимізації та реінжинірингу процесів.

Глобальна Семантика: eEPC дозволяє описувати процеси так, щоб виконання функцій могло бути залежним від стану інших, віддалених вузлів діаграми.

Взаємозв'язки: Дозволяє виявляти взаємозв'язки між організаційною, функціональною моделями та моделлю потоків даних, якщо вони створюються в методології ARIS.

Методичний Фільтр: Через велику кількість графічних елементів, які може містити еЕРС, часто створюється методичний фільтр — фіксований, обмежений набір елементів, якими аналітики користуються для забезпечення узгодженості та читабельності схем.

Базові Елементи Нотації еЕРС

Функція (Function):

Елемент, що описує набір дій (процедур, робіт), які виконуються підрозділами або співробітниками.

Веде до отримання проміжного або фінального результату чи події.

Найменування функції записується в початковій формі дієслова (наприклад, «Перевірити наявність»).

Послідовність виконання задається розташуванням на діаграмі (зверху вниз).

Подія (Event):

Стан, який є суттєвим і впливає на подальший хід бізнес-процесу або контролює його.

Події активізують функції або породжуються функціями.

Назва події поміщається всередині блоку (наприклад, «Замовлення прийнято»).

Стрілка (Зв'язок):

Відображає зв'язки елементів діаграми.

Може бути спрямованою (стрілка з трикутником) або неспрямованою (просто сполучна дуга), залежно від елементів.

Логічні Оператори (Connectors): Використовуються для позначення злиття (збіжності) або розгалуження (дивергенції) потоку процесу.

Оператор «І» (AND): Використовується для позначення ситуації, коли для подальшого виконання потрібне одночасне завершення кількох функцій (злиття) або коли завершення функції ініціює одночасно кілька подій/функцій (розгалуження).

Елемент	Зображення (зазвичай)	Призначення	Колір (традиційний)
Суб'єкт (Орг. одиниця)	Прямокутник/Овал	Виконавці, власники, учасники функції (посади, підрозділи).	Жовтий
Документ	Папірець/Карта	Документи, що супроводжують виконання функції (паперові чи електронні).	Сірий
Інформаційна система	Прямокутник з лініями	ІС, модуль чи функція, що підтримує виконання функції.	Помаранчевий/Коричневий
База даних	Цилиндр	Конкретна БД, що супроводжує виконання функції.	Сірий
Інтерфейс	Прямокутник із позначкою	Зовнішній (по відношенню до діаграми) бізнес-процес або функція.	-

Оператор «АБО» (OR): Використовується для позначення ситуації, коли завершення функції може ініціювати одне або відразу кілька подій/функцій.

Оператор «Виключне АБО» (XOR): Вказує, що при виконанні процесу може бути використана тільки одна гілка (наприклад, тільки одна з подій).

Розширені Елементи (Extended Elements)

Приставка e (extended) в нотації eEPC означає можливість використання безлічі додаткових елементів, що пояснюють та деталізують функції і події.

Нотація eEPC є більш складною і вимагає знання правил коректного побудови моделі. Проте, при використанні розширеного набору елементів, вона надає широкі можливості для докладного зображення бізнес-процесу та уточнення важливих деталей його виконання на кожному кроці.

Питання для самоперевірки, тести для самоконтролю

Як розшифровується аббревіатура eEPC і які два базові елементи лежать в її основі?

Яка головна мета створення моделі в нотації eEPC, що робить її особливо корисною для реінжинірингу?

Чим відрізняється використання логічного оператора «І» (AND) від оператора «Виключне ІЛІ» (XOR)?

Що таке «Методичний фільтр» і чому його використовують при роботі з eEPC?

Який елемент нотації eEPC використовується для позначення виконавців функції?

Тест (Виберіть одну правильну відповідь):

1. На діаграмі eEPC Подія описує:

- а) Набір дій, які потрібно виконати.
- б) Стан, який активізує або породжується функцією.
- в) Організаційну одиницю, відповідальну за процес.

2. Яка характеристика є головною для eEPC у порівнянні з IDEF0?

- а) Фокус на workflow (послідовності) та деталізації станів.

б) Фокус на функціональній ієрархії та ІСОМ-зв'язках.

в) Призначена лише для моделювання високорівневих стратегічних процесів.

3. Елементи, що містять інформацію (наприклад, Документ або База даних), у нотації eEPC зазвичай позначаються елементами якого кольору?

а) Жовтого

б) Оранжевого

в) Сірого

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача: Побудуйте простий ланцюжок eEPC для процесу «Відправка комерційної пропозиції», використовуючи одну функцію, дві події та один логічний оператор.

Відповідь-приклад: Подія «Клієнт запитав ціну» \to Функція «Підготувати пропозицію» \to Оператор XOR (розгалуження) \to Подія «Пропозиція відправлена електронною поштою» АБО Подія «Пропозиція відправлена друковано».

Задача: Визначте, які додаткові елементи eEPC необхідні для деталізації функції «Провести аудит складу», якщо її виконує Аудитор, використовує Програмне забезпечення «1С» та Звіт про залишки.

Відповідь-приклад: Суб'єкт: Аудитор. Інформаційна система: ПЗ «1С». Документ: Звіт про залишки.

Додаткові задачі (Приклади)

Приклад (Імітаційний Аналіз): Поясніть, як наявність операторів OR та XOR у моделі eEPC дозволяє проводити ефективний імітаційний аналіз процесу.

Приклад (Локальна Методологія): Вирішіть, які базові елементи eEPC (Функція, Подія, AND, OR, XOR) та який один додатковий елемент (з

Суб'єкт, Документ, ІС) ви б обрали для «методичного фільтра» у проєкті з опису HR-процесів і чому.

Довідково-інформаційні дані для розв'язування задач

Оператор	Символ	Розгалуження (Вихід з Функції)	Злиття (Вхід у Функцію/Подію)
AND (І)	[AND Connector Image]	Усі наступні гілки виконуються одночасно.	Усі попередні гілки мають бути завершені.
OR (АБО)	[OR Connector Image]	Одна або кілька наступних гілок можуть виконуватися.	Одна або кілька попередніх гілок можуть бути завершені.
XOR (Виключне АБО)	[XOR Connector Image]	Тільки одна з наступних гілок виконується.	Тільки одна з попередніх гілок може бути завершена.

Бібліографічний список

1. Scheer, August-Wilhelm. ARIS – Business Process Modeling. 2nd Edition. Springer, 2000. (Фундаментальна робота, що детально описує методологію ARIS, включаючи нотацію eEPC, її синтаксис та семантику.)
2. Harmon, Paul. Business Process Change: A Guide for Business Process Management. 3rd Edition. Morgan Kaufmann, 2014. (Порівнює eEPC з іншими нотаціями (BPMN, Flowcharts) і пояснює її використання для моделювання низькорівневих workflows та імітаційного аналізу.)
3. Davis, Alan M. Just Enough Requirements Management. Dorset House Publishing, 2005. (Обговорює важливість подієво-керованих моделей для виявлення вимог та управління станами процесів.)
4. ІБА. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute for Business Analysis, 2015. (Описує техніки моделювання процесів та їхню роль у розумінні детального виконання бізнес-завдань.)

Тема 7. Чередування «подія – функція» та використання операторів у нотації eEPC

Принцип Чередування «Подія – Функція»

В основі моделювання в нотації eEPC лежить опис послідовності дій (функцій), які виконують користувачі, і станів (подій), які ці дії активізують або є їхнім результатом.

Подія (Event): Стан процесу, при якому має бути виконана дія. Подія або комбінація подій — це необхідна і достатня умова виконання деякої дії (функції). Якщо подія не настала, функція не може бути виконана.

Приклади: Дзвінок клієнта, отримання листа, спрацьовування таймера.

Функція (Function): Дія, яку виконують користувачі в системі.

Ключове Правило eEPC: Подія \to Функція \to Подія

Кожній функції має передувати подія.

Кожна функція повинна завершуватися подією, яка вказує новий стан системи (тобто результат виконання функції).

Використання Логічних Операторів (Connectors)

Логічні оператори (І, ІЛІ, Виключне ІЛІ) використовуються для позначення злиття (збіжності) або розгалуження (дивергенції) функцій та подій у потоці процесу.

Оператор «І» (AND)

Випадок	Місце Оператора	Семантика (Значення)
Розгалуження	Після функції \to перед подіями	Завершення функції ініціює одночасно кілька подій.
Злиття	Після функцій \to перед подією	Подія настає лише після обов'язкового завершення кількох функцій.
Злиття	Після подій \to перед функцією	Функція може початися лише після того, як відбудуться всі необхідні події.
Розгалуження	Після події \to перед функціями	Одна подія ініціює одночасне виконання кількох функцій.

Оператор «І» позначає одночасне виконання або обов'язкову залежність від кількох факторів. Він використовується для злиття або розгалуження функцій та подій.

Оператор «ІЛІ» (OR)

Оператор «ІЛІ» позначає можливість вибору однієї або кількох гілок. Він використовується для злиття або розгалуження функцій, але **ТІЛЬКИ** для злиття подій.

Випадок	Місце Оператора	Семантика (Значення)
Розгалуження	Після функції \to перед подіями	Завершення функції може ініціювати одне або кілька подій.
Злиття	Після функцій \to перед подією	Подія настає після завершення однієї або кількох функцій.
Злиття	Після подій \to перед функцією	Функція може почати виконуватися після того, як відбудеться одна або кілька подій.

Обмеження: Оператор «ІЛІ» не може слідувати після одиночної події.

Оператор «Виключне ІЛІ» (XOR)

Оператор «Виключне ІЛІ» використовується для позначення вибору лише однієї з можливих гілок. Він використовується для злиття або розгалуження функцій, але **ТІЛЬКИ** для злиття подій.

Випадок	Місце Оператора	Семантика (Значення)
Розгалуження	Після функції \to перед подіями	Завершення функції може ініціювати ТІЛЬКИ ОДНУ з подій (залежно від умови).
Злиття	Після функцій \to перед подією	Подія настає одразу після завершення АБО однієї функції, АБО іншої (не обох).
Злиття	Після подій \to перед функцією	Функція може почати виконуватися одразу після того, як відбудеться ТІЛЬКИ ОДНА подія.

Обмеження: Оператор «Виключне ІЛІ» не може слідувати після одиночної події.

Питання для самоперевірки, тести для самоконтролю

Як у нотації eEPC визначається Подія? Чому вона є необхідною та достатньою умовою для виконання функції?

Сформулюйте основне правило моделювання потоків робіт у eEPC.

Чи може оператор «ІПІ» (OR) слідувати після одиночної події? Поясніть чому.

Наведіть приклад розгалуження, коли завершення однієї функції ініціює одночасне виконання двох наступних функцій. Який оператор слід використовувати?

Поясніть різницю в семантиці (значенні) між злиттям за допомогою оператора «ІПІ» (OR) та оператора «Виключне ІПІ» (XOR).

Тест (Виберіть одну правильну відповідь):

1. У коректному ланцюжку eEPC функції завжди мають бути оточені:
 - а) Іншими функціями.
 - б) Подіями.
 - в) Логічними операторами.
2. Оператор «І» (AND) використовується для злиття функцій перед наступною подією. Це означає, що подія настане, коли:
 - а) Завершиться хоча б одна з попередніх функцій.
 - б) Завершаться усі попередні функції.
 - в) Завершиться лише одна, обрана за умовою, функція.
3. Який оператор дозволяє ініціювати лише одну з двох можливих гілок процесу після завершення функції?
 - а) Оператор «І» (AND)
 - б) Оператор «ІПІ» (OR)
 - в) Оператор «Виключне ІПІ» (XOR)

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача (Створення ланцюжка): Створіть коректний ланцюжок eEPC для такої ситуації: Подія «Платіж отримано» \to має розпочатися одночасне виконання двох дій: «Сформувати рахунок-фактуру» та «Відвантажити товар».

Відповідь-приклад: Подія «Платіж отримано» \to Оператор І (AND) (розгалуження) \to Функція «Сформувати рахунок-фактуру» та Функція «Відвантажити товар».

Задача (Злиття): У процесі «Схвалення кредиту» кінцева подія «Кредит схвалено» може настати після того, як завершиться АБО функція «Автоматична перевірка», АБО функція «Ручна перевірка менеджером». Який логічний оператор слід використовувати для злиття цих функцій перед подією?

Відповідь-приклад: Оператор «Виключне ІЛІ» (XOR). Оскільки схваленням займається лише одна гілка, це виключає одночасне виконання чи можливість множинного шляху.

Додаткові задачі (Приклади)

Приклад (Некоректний Ланцюжок): Поясніть, чому ланцюжок «Функція А \to Подія Х \to Оператор OR \to Функція В» є некоректним згідно з правилами eEPC.

Приклад (Глобальна Семантика): Опишіть ситуацію, коли Подія «Замовлення перевищує 100 000 грн» (результат однієї функції) має активізувати функцію «Погодити додаткові умови», яка знаходиться в іншій частині діаграми. Поясніть, як це ілюструє поняття глобальної семантики в eEPC.

Довідково-інформаційні дані для розв'язування задач

Оператор	Розгалуження (Після Функції)	Злиття (Перед Подією)
І (AND)	Function \to AND \to Event + Event (Обидві події настають одночасно)	Function + Function \to AND \to Event (Подія настає після обох функцій)
ІЛІ (OR)	Function \to OR \to Event + Event (Настає одна або обидві події)	Function + Function \to OR \to Event (Подія настає після однієї або обох функцій)
Виключне ІЛІ (XOR)	Function \to XOR \to Event + Event (Настає ТІЛЬКИ ОДНА подія)	Function + Function \to XOR \to Event (Подія настає після ТІЛЬКИ ОДНІЄЇ функції)

Бібліографічний список

1. Scheer, August-Wilhelm. ARIS – Business Process Modeling. 2nd Edition. Springer, 2000. (Містить повний опис синтаксичних та семантичних правил eEPC, на яких ґрунтується використання логічних операторів та принцип чергування.)
2. Harmon, Paul. Business Process Change: A Guide for Business Process Management. 3rd Edition. Morgan Kaufmann, 2014. (Деталізує, як логічні оператори eEPC використовуються для відображення розгалужень та збіжностей у реальних бізнес-процесах для подальшого аналізу.)
3. ИБА. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute for Business Analysis, 2015. (Описує техніку моделювання процесів та необхідність чіткого визначення тригерів (подій) та результатів дій.)

РОЗДІЛ 5. ПЕРЕХІД ДО ВИКОНУВАНИХ МОДЕЛЕЙ:

ПОГЛИБЛЕННЯ ЕЕРС ТА ВВЕДЕННЯ В VRMN

(Даций О.І., д.е.н., професор, заслужений працівник освіти України,
президент Громадської організації «Асоціація науковців України»)

Тема 1. Доповнення моделі процесу деталями у нотації eEPC

Нотація eEPC (розширена подієва процесний ланцюжок) дозволяє збагачувати базову модель «Подія – Функція» додатковими елементами, що відображають організаційний та інформаційний аспекти процесу.

I. Виконавці (Суб'єкти)

Для позначення виконавця — департаменту, команди, або конкретної посади (ролі) — використовуються елементи типу «Суб'єкт» (Subjekt/Organizational Unit).

Правила використання Суб'єктів:

Зв'язок: Елементи суб'єктів повинні співвідноситися з конкретною функцією, яку вони виконують, за допомогою неспрямованого відношення (дуги). Якщо програмне забезпечення дозволяє, обирається зв'язок типу «Виконує».

Відображення: Кожній функції, яку вони виконують, присвоюється суб'єкт. Якщо один виконавець виконує кілька функцій, він відображається окремо як суб'єкт для кожної з цих функцій.

Розташування: Прийнято зображати елементи суб'єктів-виконавців праворуч від процесного ланцюжка (виконуваних функцій).

Фокус: На діаграмі eEPC не повинні зображатися інші відносини між суб'єктами (обмін документами, підпорядкування, ієрархія).

Узгодженість: У випадку моделювання всієї організації, в моделі процесу мають використовуватися лише ті суб'єкти, які згадані в

організаційній структурі підприємства. Це забезпечує зв'язок організаційного та функціонального аспектів.

II. Дані та Документи

Дані, документи та інформація використовуються функцією як вхідний ресурс та/або є результуючими артефактами (виходом) її виконання.

Правила використання Даних/Документів:

Вхідні Дані: Якщо дані потрібні для виконання функції (вхідний ресурс), елемент даних з'єднується з функцією спрямованою стрілкою (напрямок: від елемента даних до функції). Розташовується ліворуч або вище функції.

Вихідні Дані: Якщо дані створюються функцією (вихідний артефакт), елемент з'єднується з функцією спрямованою стрілкою (напрямок: від функції до елемента даних). Розташовується ліворуч або нижче функції і може використовуватися як вхід для наступної функції.

Напрямок Зв'язку: Документ або інформація завжди пов'язані з функцією спрямованою стрілкою, оскільки вони або використовуються, або створюються/доповнюються.

Додана Вартість: Якщо на вхід функції надходить оброблений документ, і він лише доповнюється інформацією, різниця (додана цінність) має бути відображена у назві результуючого документа (наприклад, «Замовлення клієнта» \to «Оформлене замовлення клієнта»).

Тип Носія: Слід обирати елемент, що однозначно визначає тип носія (паперовий, електронний, БД).

Потік Документів: Потік документів відображається на діаграмі окремо від потоків робіт. Це дозволяє цілісно описати причинно-наслідкові зв'язки та сам документообіг в одній схемі.

III. Інші Додаткові Елементи

Інформаційні Системи (ІС), Модулі та Функції:

Зображуються ліворуч від потоку робіт.

З'єднуються з функцією, яку реалізують, неспрямованою дугою.

Бази Даних (БД):

Зображуються ліворуч або праворуч від функції.

Використання даних БД зазвичай двонаправлене (читання та запис).

Відношення БД до функції описується неспрямованою дугою.

Інше: Може зображуватися як ліворуч, так і праворуч, залежно від значення елемента для бізнес-процесу.

Питання для самоперевірки, тести для самоконтролю

Яке відношення (тип зв'язку) використовується для з'єднання Функції та Суб'єкта у eEPC?

Як на діаграмі eEPC відображаються вхідні дані, необхідні для виконання функції? Вкажіть напрямки стрілки.

Яким чином рекомендується розташовувати елементи Суб'єктів відносно основного потоку робіт?

Чому відношення між Функцією та Базою Даних часто позначають неспрямованою дугою?

Як необхідно відображати документ, який обробляється функцією та доповнюється новою інформацією?

Тест (Виберіть одну правильну відповідь):

1. Напрямок стрілки, що з'єднує функцію «Скласти звіт» з елементом «Готовий звіт»:

- а) Від елемента даних до функції.
- б) Від функції до елемента даних.
- в) Неспрямована дуга.

2. Якщо один менеджер виконує три різні функції в одному процесі, то на діаграмі eEPC елемент «Менеджер» буде відображений:

- а) Один раз, і з'єднаний з усіма трьома функціями.
- б) Тричі, окремо для кожної функції.

в) Лише один раз, на контекстній діаграмі.

3. Згідно з правилами eEPC, де зазвичай розташовуються елементи Інформаційних Систем?

а) Праворуч від функцій.

б) Ліворуч від потоку робіт.

в) Під функціональними блоками.

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача (Моделювання документообігу): Опишіть, використовуючи елементи eEPC, як функція «Перевірити заявку» використовує «Заявку клієнта» (вхідний документ) та створює «Позначену заявку клієнта» (вихідний документ). Вкажіть напрямки зв'язків.

Відповідь-приклад: 1) «Заявка клієнта» \to Спрямована стрілка \to Функція «Перевірити заявку». 2) Функція «Перевірити заявку» \to Спрямована стрілка \to «Позначена заявка клієнта» (додана цінність у назві).

Задача (Суб'єкт): Функція «Зареєструвати новий договір» виконується Менеджером відділу продажів та підтримується Системою CRM. Зобразіть, як ці два додаткові елементи будуть пов'язані з функцією.

Відповідь-приклад: Суб'єкт (Менеджер) з'єднується з функцією неспрямованою дугою (з правого боку). ІС (Система CRM) з'єднується з функцією неспрямованою дугою (з лівого боку).

Додаткові задачі (Приклади)

Приклад (Адаптація до українських реалій): При моделюванні процесу «Подання податкової звітності» які елементи слід позначити як Зовнішні Суб'єкти та Зовнішні Документи, що відображають взаємодію з ДПС (Державна податкова служба)?

Приклад (Коректність): Поясніть, чому відображення спрямованої стрілки від Бази Даних до Функції може бути семантично неточним, навіть якщо функція тільки читає дані.

Довідково-інформаційні дані для розв'язування задач

Елемент	Зв'язок	Напрямок	Тип ролі	Рекомендоване Розташування
Суб'єкт (Виконавець)	Неспрямована дуга	N/A	Виконує	Праворуч
Інформаційна Система	Неспрямована дуга	N/A	Підтримує	Ліворуч
База Даних	Неспрямована дуга	N/A	Читання/Запис	Ліворуч або Праворуч
Вхідні Дані/Документ	Спрямована стрілка	До \to Функції	Використовується як ресурс	Ліворуч/Вище
Вихідні Дані/Документ	Спрямована стрілка	Від \to Функції	Створюється/Доповнюється	Ліворуч/Нижче

Бібліографічний список

1. Scheer, August-Wilhelm. ARIS – Business Process Modeling. 2nd Edition. Springer, 2000. (Містить детальні правила для інтеграції організаційних одиниць (Суб'єктів) та інформаційних артефактів (Даних, ІС, БД) у модель eEPC, що робить її «розширеною».)
2. Harmon, Paul. Business Process Change: A Guide for Business Process Management. 3rd Edition. Morgan Kaufmann, 2014. (Пояснює, як eEPC використовується для створення цілісного опису процесу, що поєднує логіку потоку робіт, організаційну структуру та інформаційні ресурси.)
3. IIBA. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute for Business Analysis, 2015. (Описує важливість моделювання контексту, включаючи суб'єктів та ресурси, як частину аналізу вимог.)

Тема 2. Використання інтерфейсів для зв'язування процесів та ієрархія ARIS

Елемент «Інтерфейс» (Interface) в нотації eEPC використовується для вказівки взаємозв'язку між різними процесами. Він відображає на поточній діаграмі інший процес, відмінний від розглянутого.

Правила використання елемента «Інтерфейс»:

Горизонтальний Зв'язок: Використовується як позначення попереднього або наступного процесу по відношенню до розглянутого. Це вказує на горизонтальну зв'язок між поточним процесом та іншими процесами, що формують повний шлях перетворення вхідних даних у кінцевий продукт.

Об'єктний Зв'язок: Використовується як позначення процесу, звідки надійшов або куди передається певний об'єкт (ресурс, дані), що має значення в поточному процесі.

Використання інтерфейсів дозволяє дробити процеси на менші, більш керовані діаграми, зберігаючи при цьому необхідну наочність. Це також дає можливість сконцентруватися на моделюванні лише найважливіших процесів, представляючи всі інші, пов'язані з ними, за допомогою елемента «Інтерфейс», не розробляючи їхні детальні моделі.

Зв'язок Між Бізнес-процесами в Моделі

У моделюванні бізнес-процесів за методологією ARIS, крім горизонтального зв'язку, існує вертикальний зв'язок або ієрархія, що дозволяє описати організацію як єдине ціле.

Методологія ARIS та Вертикальна Декомпозиція:

Для структурного опису всіх функцій організації та повноцінного представлення її функціоналу, методологія ARIS передбачає послідовну декомпозицію функцій (аналогічно IDEF0), але за допомогою трьох типів діаграм, що забезпечує вертикальний зв'язок:

Діаграма Процесів Верхнього Рівня: Представляє класифікацію бізнес-процесів та їхнє об'єднання у великі групи (наприклад, Основні, Керуючі, Допоміжні).

Діаграма Функціонального Дерева (Ієрархічна Структура): Розбиває групи на рівні переліків бізнес-процесів як таких (Ієрархія функцій/процесів).

Детальний Опис (eEPC): Завершує декомпозицію детальним описом кожного бізнес-процесу за допомогою нотації eEPC, фіксуючи логіку workflow (послідовність «Подія – Функція»).

Таким чином, ARIS передбачає повноцінний опис діяльності організації, послідовно поєднуючи функціональний аспект (що робимо) з процесно-подієвим аспектом (як робимо).

Питання для самоперевірки, тести для самоконтролю

У яких двох основних випадках використовується елемент «Інтерфейс» у eEPC для зв'язування процесів?

Яку функцію виконує «Діаграма Процесів Верхнього Рівня» у вертикальній ієрархії ARIS?

Чим відрізняється горизонтальний зв'язок між процесами від вертикального зв'язку в методології ARIS?

Яка перевага використання елемента «Інтерфейс» для аналітика при моделюванні дуже великих процесів?

Тест (Виберіть одну правильну відповідь):

1. Елемент «Інтерфейс» у eEPC використовується для позначення:

- а) Зв'язку функції з інформаційною системою.
- б) Іншого бізнес-процесу (попереднього чи наступного).
- в) Обміну даними між двома функціями в рамках однієї діаграми.

2. Яка нотація використовується для детального опису бізнес-процесу на найнижчому рівні в ієрархії ARIS?

а) IDEF0

б) eEPC

в) BPMN

3. «Діаграма Функціонального Дерева» в ARIS відповідає за:

а) Визначення виконавців та їхніх ролей.

б) Розбивку груп процесів на переліки конкретних бізнес-процесів.

в) Опис послідовності «Подія – Функція» для окремого workflow.

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача (Горизонтальний зв'язок): Вам потрібно з'єднати Подію «Товар відвантажено» (яка завершує поточний процес) з початком наступного процесу «Облік реалізації». Який елемент слід використати між цією Подією та початковою Подією наступного процесу?

Відповідь-приклад: Слід використати елемент «Інтерфейс» після Події «Товар відвантажено», щоб позначити перехід до процесу «Облік реалізації».

Задача (Вертикальна Декомпозиція): Процес «Обробка замовлення» знаходиться на другому рівні Діаграми Функціонального Дерева. Яка діаграма буде знаходитися вище (на першому рівні) і міститиме цей процес?

Відповідь-приклад: Діаграма Процесів Верхнього Рівня, де «Обробка замовлення» буде включена до групи «Основні Процеси» (або «Процеси Виконання Замовлень»).

Додаткові задачі (Приклади)

Приклад (Ієрархія): Якщо процес «Управління якістю» знаходиться на Діаграмі Процесів Верхнього Рівня (Керуючі Процеси), які назви функцій можуть бути присутніми у його Діаграмі Функціонального Дерева? (Назвіть 3 приклади).

Приклад (Інтерфейс): Поясніть, як елемент «Інтерфейс» може допомогти бізнес-аналітику, якщо він моделює лише «Процес повернення

товару», але йому необхідно посилатися на «Процес обробки платежів» для повернення коштів.

Довідково-інформаційні дані для розв'язування задач

Рівень (Призначення)	Нотація/Діаграма	Фокус	Відношення до eEPC
Вищий (Класифікація)	Діаграма Процесів Верхнього Рівня	Групи Процесів (Основні, Керуючі)	Об'єднує всі процеси у категорії.
Середній (Структура)	Діаграма Функціонального Дерева	Ієрархія конкретних процесів	Дерево функцій, яке декомпозується на eEPC.
Нижчий (Деталізація)	eEPC	Послідовність «Подія – Функція»	Детальний опис кожного процесу.

Бібліографічний список

1. Scheer, August-Wilhelm. ARIS – Business Process Modeling. 2nd Edition. Springer, 2000. (Описує концепцію трьох рівнів моделювання в ARIS: Діаграма Процесів Верхнього Рівня, Функціональне Дерево та eEPC, і як ці моделі взаємопов'язані вертикально.)
2. Harmon, Paul. Business Process Change: A Guide for Business Process Management. 3rd Edition. Morgan Kaufmann, 2014. (Надає аналіз необхідності використання інтерфейсів для підтримки горизонтального зв'язку між процесами в складних корпоративних архітектурах.)
3. IIBA. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute for Business Analysis, 2015. (Обговорює важливість створення ієрархічних моделей та контекстуалізації процесів у ширшій структурі підприємства.)

Тема 3. Особливості побудови моделі eEPC та рекомендації з оптимізації

Нотація eEPC (extended Event-driven Process Chain) вважається досить складною для моделювання та подальшого аналізу бізнес-процесів. Це відбувається через поєднання кількох факторів, які можуть призвести до створення формально правильної, але громіздкої та важко сприйманої схеми:

- Використання великої кількості додаткових елементів (Суб'єкти, ІС, Документи тощо).

- Відсутність коментарів або опису діаграми з легендою.

- Велика кількість розгалужень та складних потоків робіт (Workflow).

- Необхідність формального використання логічних операторів (AND, OR, XOR) для дотримання суворих правил нотації.

Ця проблема є типовою для більшості нотацій типу Workflow. Щоб уникнути створення схеми, позбавленої сенсу, та досягти головної мети моделювання (прозорості та доступності для аналізу), фахівцям рекомендується дотримуватись низки правил:

Рекомендації для Створення Читабельної Моделі eEPC

Визначення Набору Елементів (Методичний Фільтр): Залежно від задач моделювання, необхідно заздалегідь визначити та обмежити (фільтрувати) набір елементів, які будуть використовуватися. Цей набір має містити мінімум елементів, але водночас повно відображати всі необхідні особливості процесу.

Документування Легенди та Історії Змін: При створенні портфоліо бізнес-процесів організації слід формувати пояснювальні документи або хоча б короткі переліки діаграм з історією їхніх змін, а також з описом «легенди» — набору використовуваних елементів та їхнього значення.

Вибірковість Деталізації: Деталізований опис (еЕРС) слід створювати лише для тих бізнес-процесів, які потрібно аналізувати та оптимізувати в подальшому.

Високорівневе Моделювання: Для відображення загальної картини діяльності організації слід створювати Діаграми Процесів Верхнього Рівня та Функціонального Дерева для функцій, де важливо відобразити склад процесів, а не їхню логіку.

Розподіл Розгалужень: При моделюванні складного процесу з великою кількістю гілок, слід розподіляти гілки на діаграмі таким чином, щоб уникнути перетинів та надто близького розташування елементів, особливо логічних операторів.

Узгодженість Кольорів: Для одного типу додаткових елементів (наприклад, для всіх документів) слід використовувати один рекомендований колір (наприклад, сірий).

Узгодженість Найменувань: Елемент, що повторюється на діаграмі (чи на різних діаграмах моделі), повинен мати однакове найменування. Різниця в найменуванні завжди інтерпретується як різниця в суті самого елемента.

Питання для самоперевірки, тести для самоконтролю

Назвіть принаймні три фактори, які роблять нотацію еЕРС складною для моделювання.

Яку функцію виконує «легенда» при описі бізнес-процесів?

Чому при моделюванні рекомендується створювати лише мінімально необхідний набір додаткових елементів?

Як слід інтерпретувати наявність двох елементів з різними назвами, але схожою суттю (наприклад, «Замовлення» та «Заявка») на одній діаграмі?

Тест (Виберіть одну правильну відповідь):

1. Якщо аналітику потрібно відобразити загальну картину діяльності компанії без деталізації workflow, він повинен використовувати:

- а) Детальну діаграму eEPC з усіма додатковими елементами.
- б) Діаграми процесів верхнього рівня та функціонального дерева.
- в) Лише один блок на контекстній діаграмі eEPC.

2. Яка рекомендація стосується найменування елементів у моделі eEPC?

- а) Допускається невелика варіація назв для підвищення унікальності.
- б) Один і той же елемент повинен мати одне і те ж назву в усіх місцях.
- в) Найменування елементів не впливає на їхню сутність.

3. Яка мета є ключовою для рекомендації щодо уникнення перетинів та близького розташування елементів на схемі eEPC?

- а) Зменшення кількості логічних операторів.
- б) Спрощення імітаційного аналізу.
- в) Забезпечення читабельності та сприйняття схеми.

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача (Узгодженість): Ви моделюєте процес «Реєстрація клієнта». Ви використовуєте елемент «Суб'єкт» з назвою «Менеджер з роботи з клієнтами». У наступному процесі «Укладання угоди» ви використовуєте елемент «Менеджер клієнтів». Поясніть, що потрібно зробити, керуючись правилами узгодженості.

Відповідь-приклад: Необхідно уніфікувати назву елемента. Якщо це одна й та сама посада, назва має бути ідентичною (наприклад, «Менеджер з роботи з клієнтами») у всіх процесах, щоб уникнути помилкової інтерпретації, що це різні виконавці.

Задача (Складний Процес): Ви моделюєте функцію, яка має 5 розгалужень XOR та 3 розгалуження OR, а також використовує 15 додаткових елементів. Яку рекомендацію слід застосувати в першу чергу?

Відповідь-приклад: Слід переглянути мету моделювання та застосувати рекомендацію 1 (визначити набір елементів) та рекомендацію 3 (створювати деталізацію лише для потрібних частин). Також варто подумати про розбиття складної функції на кілька менших, щоб зменшити кількість розгалужень на одній діаграмі.

Додаткові задачі (Приклади)

Приклад (Кольори): Вирішіть, які кольори ви б призначили для Інформаційної Системи та Баз Даних у вашій локальній «легенді», якщо ви вибрали блакитний для функції та сірий для документа. Обґрунтуйте вибір.

Приклад (Громіздкість): Поясніть, чому діаграма eEPC, яка є формально правильною, але має велику кількість перетинів стрілок, все одно вважається неефективною для цілей аналізу.

Довідково-інформаційні дані для розв'язування задач

Рекомендації Щодо Оптимізації Моделі eEPC

Рекомендація	Мета	Приклад Порухення
Обмеження елементів (Метод. фільтр)	Запобігання громіздкості	Додавання 10+ типів додаткових елементів (шрифти, ІС, інше обладнання, ролі).
Узгодженість найменувань	Забезпечення однозначності	Назви «База клієнтів» та «БД клієнтів» на різних діаграмах.
Вибірковість деталізації	Фокус на меті аналізу	Детальне моделювання процесу «Видача ручок» при меті оптимізації «Продаж».
Чітке розташування	Покращення читабельності	Перетини стрілок, оператори OR/XOR розташовані надто близько.

Бібліографічний список

1. Scheer, August-Wilhelm. ARIS – Business Process Modeling. 2nd Edition. Springer, 2000. (Надає методичні рекомендації щодо створення зрозумілих моделей eEPC, включаючи необхідність узгодження термінології та візуальної чистоти діаграм.)

2. Harmon, Paul. Business Process Change: A Guide for Business Process Management. 3rd Edition. Morgan Kaufmann, 2014. (Обговорює загальні проблеми, пов'язані з моделями workflow (включаючи eEPC), які стають громіздкими, та пропонує стратегії мінімізації складності.)

3. ПБА. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute for Business Analysis, 2015. (Підкреслює важливість створення моделей, які є прозорими та доступними для аналізу, що є ключовою метою використання методичного фільтру.)

Тема 4. Порівняльний аналіз нотацій IDEF0 та eEPC

Порівняння нотацій IDEF0 (функціональна декомпозиція) та eEPC (подієва процесний ланцюжок) показує, що вони вирішують схожі завдання (опис процесів організації), але мають кардинально різні підходи та сфери застосування.

Ключові Відмінності та Обмеження

Критерій	IDEF0	eEPC
Основний Фокус	Функціональна структура та логіка ICOM (Входи, Управління, Механізми). Що робиться?	Послідовність виконання (workflow) та стани (події). Коли і як робиться?
Управляюча Дія	Обов'язкова (Стрілка Управління зверху).	Необов'язкова, може бути прихована в документах або в попередній події.
Логіка Виконання	Відсутня (Немає операторів AND, OR, XOR). Процес виглядає «плоско».	Сильна (Є оператори AND, OR, XOR). Дозволяє відобразити складні розгалуження та умови.
Послідовність	Неочевидна. Розташування блоків обмежується правилами.	Очевидна (зверху вниз). Легко читається як описовий текст.
Деталізація	Синтаксис зобов'язує до наповнення моделі (С, І, М, О).	На розсуд аналітика. Можливе надмірне спрощення, якщо не використовувати додаткові елементи.
Можливість Імітації	Неможлива через відсутність чіткої послідовності та логіки.	Можлива завдяки послідовності та логічним операторам.

Методологічний Підхід та Ієрархія

IDEF0: Використовує одну нотацію для опису всіх рівнів деталізації (від контекстної діаграми до детальної декомпозиції). Це спрощує читання та знижує кількість помилок.

ARIS (eEPC): Використовує мінімум три нотації/підходи (Діаграма Верхнього Рівня, Функціональне Дерево, eEPC). eEPC використовується лише на етапі максимальної деталізації процесу.

Хоча обидві методології ефективні для багат шарового опису, IDEF0 забезпечує єдиний підхід на всіх рівнях, тоді як ARIS/eEPC вимагає знання кількох нотацій.

Вибір Нотації для Конкретних Завдань

Вибір нотації залежить від мети аналізу та рівня досвіду аналітика:

Завдання Аналізу	Рекомендована Нотація	Обґрунтування
Детальна Проробка складних, розгалужених процесів.	eEPC	Чітка послідовність, логічні оператори, можливість імітації.
Опис загальної структури процесів для топ-менеджменту (максимальна прозорість).	IDEF0	Фокус на функції, обов'язкове відображення управлінь та механізмів.
Опис кількох непов'язаних процесів, але відразу детально.	eEPC	Швидко відображає workflow без необхідності будувати ієрархію.
Створення загального уявлення про організацію як систему та її функціональні напрямки.	IDEF0	Принцип ICOM та ієрархічна декомпозиція підходять для системного аналізу.
Моделювання аналітиком без досвіду або глибокого розуміння сфери.	IDEF0	Синтаксис є простішим на перший погляд, і єдиний підхід на всіх рівнях.
Моделювання досвідченим аналітиком, уважним до деталей та готовим до складності.	eEPC	Дозволяє зобразити кожен складову компоненту процесу, необхідну для оптимізації.

Питання для самоперевірки, тести для самоконтролю

Яке ключове обмеження IDEF0 у порівнянні з eEPC робить неможливим імітаційне моделювання?

Яким чином у нотації eEPC може бути відображене управляюче завдання, якщо не використовується спеціальний елемент контролю?

Опишіть головний ризик при використанні eEPC, пов'язаний із самостійним рішенням аналітика щодо деталізації.

Яка з нотацій (IDEF0 чи eEPC) вимагає знання меншої кількості підходів/діаграм для опису всіх рівнів ієрархії процесів?

Тест (Виберіть одну правильну відповідь):

1. Головна перевага IDEF0 перед eEPC полягає в:

а) Можливості чіткого відображення логіки виконання за допомогою операторів.

б) Легкості читання послідовності зверху вниз.

в) Обов'язковості відображення керуючих впливів (стрілка Управління).

2. Якщо завданням є максимально детальна проробка складного процесу з багатьма умовами («якщо... то...»), слід обрати:

а) IDEF0

б) eEPC

в) Функціональне Дерево

3. Нотація, в якій не передбачено використання символів для логіки виконання процесу (AND, OR, XOR):

а) eEPC

б) IDEF0

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача (Синтаксис): Поясніть, що станеться з моделлю процесу в eEPC, якщо аналітик вирішить не використовувати додаткові елементи, щоб зробити модель максимально простою.

Відповідь-приклад: Модель буде поверхневою і непридатною для достатнього аналізу, оскільки eEPC не зобов'язує до наповнення деталями (на відміну від IDEF0), і є ризик надмірного спрощення процесу.

Задача (Читабельність): Поясніть, чому процес в IDEF0, навіть при правильному моделюванні, часто «важко читається» з точки зору послідовності.

Відповідь-приклад: Через відсутність очевидної послідовності (відсутність стрілок workflow), розташування блоків обмежене, а множинні контролю та механізми (стрілки зверху та знизу) створюють складну сітку зв'язків, яка не відображає часовий потік.

Додаткові задачі (Приклади)

Приклад (Точка Зору): Ви — системний архітектор, який розробляє загальну структуру нової ІС. Яку нотацію ви, ймовірно, оберете для опису функціональних вимог верхнього рівня? Обґрунтуйте вибір.

Приклад (Оптимізація): Бізнес-аналітик має оптимізувати час виконання процесу. Чому наявність логічних операторів у eEPC є критично важливою для цього завдання?

Довідково-інформаційні дані для розв'язування задач

Характеристика	IDEF0	eEPC
Основний Елемент	Функціональний Блок (І-С-О-М)	Функція, Подія, Логічний Оператор
Відображення Часу	Ніяк (не фокусується)	Пріоритетне (зверху вниз)
Логіка/Розгалуження	Ні (тільки розгалуження стрілок)	Так (AND, OR, XOR)
Модель Управління	Активна (Обов'язкова стрілка Control)	Пасивна (Через попередні Події або Документи)
Сфера Застосування	Структурний/Функціональний Аналіз	Опис Workflow/Імітаційне моделювання

Бібліографічний список

1. Scheer, August-Wilhelm. ARIS – Business Process Modeling. 2nd Edition. Springer, 2000. (Один із ключових текстів, що містить порівняння IDEF0 та ARIS/eEPC, підкреслюючи їхні різні фокуси: структура проти потоку.)
2. Harmon, Paul. Business Process Change: A Guide for Business Process Management. 3rd Edition. Morgan Kaufmann, 2014. (Надає практичний порівняльний аналіз, зосереджуючись на тому, яка нотація краще підходить для оптимізації (eEPC) та системного аналізу (IDEF0).)
3. ПБА. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute for Business Analysis, 2015. (Описує обидві нотації як інструменти моделювання процесів, які використовуються для різних цілей бізнес-аналізу.)

Тема 5. Сильні та слабкі сторони нотації eEPC та типові помилки

Нотація eEPC (extended Event-driven Process Chain) є потужним інструментом для детального моделювання workflow (потіку робіт), але має низку особливостей та недоліків, які вимагають уваги аналітика.

Переваги Нотації eEPC (Сильні Сторони)

Зручність для Workflow: Належить до класу нотацій Workflow (потік робіт), що робить її простою та зручною для зображення довгого, багатокрокового процесу від початку до кінця.

Гнучкість та Розширюваність: Дозволяє додавати власні авторські елементи та визначати закріплені набір елементів (методичний фільтр). Відсутність жорсткого набору елементів і правил використання більшості з них дозволяє розширити модель від мінімалістичного вигляду до повної картини подій.

Ефективне Відображення Логіки: Пропонує використання логічних операторів (І, ІЛІ, Виключне ІЛІ), які дозволяють ефективно та прозоро відобразити всі варіації виконання процесу та його розгалуження.

Детальний Опис Контексту: Спрямована на детальне та точне описання бізнес-процесу, відображаючи в графічному вигляді всіх виконавців, ресурси, об'єкти та ІС.

Недоліки Нотації eEPC (Слабкі Сторони)

Високі Вимоги до Команди: Використання широкої бібліотеки елементів та можливість вводити авторські елементи вимагає попередньої підготовки команди, досягнення внутрішніх домовленостей щодо набору елементів (легенди) та додаткового навчання.

Розбіжність Деталізації: Розуміння необхідної деталізації у аналітика та замовника може не збігатися, що ускладнює погодження моделі з учасниками, не знайомими з нотацією.

Обмеження Часу/Діагностики: Немоżliвість вказати тривалість процесу. Модель відображає лише логічну послідовність, тому за нею не можна виявити такі проблеми, як перевантаження співробітника або неможливість виконати обсяг робіт за заданий інтервал часу. (Для цього потрібна діаграма Ганта або BPMN з атрибутами часу).

Втрата Керуючих Впливів: Проста модель типу Workflow може не відображати багатьох керуючих впливів (контролів), що призводить до втрати реальних процесів контролю і, як наслідок, до нездатності коректно визначити можливості процесу.

Типові Помилки При Побудові Моделі eEPC

Через гнучкість eEPC та одночасну наявність суворих правил, ризик виникнення синтаксичних та смислових помилок високий.

А. Некоректне Зображення Самого Ланцюжка Робіт

Порушення Чергування: Використання двох функцій або двох подій поспіль (наприклад, `\text{Функція} \to \text{Функція}`). Це порушує основне правило синтаксису: кожна дія має закінчуватися станом (`\text{Функція} \to \text{Подія}`).

Узагальнення Функцій: Перерахування кількох окремих дій в одній функції, які насправді виконуються окремо. Кожна принципово окрема активність повинна бути зображена як окрема функція зі своїм результуючим станом. Це необхідно для виявлення дублювань, відсутності залежностей та втрачених кроків при аналізі.

Б. Некоректне Використання Операторів

Оператори OR/XOR після Події: Заборонено використовувати оператори «ІІІ» та «Виключне ІІІ» після події, оскільки подія сама по собі не приймає рішення. В такому разі слід використовувати оператор «І» або додати дві окремі вихідні події.

Пропуск Операторів: Обов'язкове використання одного з операторів (І, ІІІ, XOR) для будь-яких вхідних чи вихідних зв'язків, коли їх більше однієї.

Виняток: Якщо подія має дві вихідні зв'язки на функції, допустимо використання лише оператора «І».

Порушення Циклів (Зворотний Зв'язок): При моделюванні циклу (повернення з кроку $n+k$ на крок n) для злиття зв'язків перед функцією n допустимо використання ТІЛЬКИ оператора «Виключне ІІІ».

В. Некоректне Використання Додаткових Елементів

Втрата Документообігу: Відсутність або часткове зображення потоків документів, що призводить до втрати керуючих контролів. Також помилкою є не відокремлення документообігу від потоку робіт (коли документи підписуються на стрілках між функцією та подією).

Сплутування Стану та Документа: Смішування понять результуючого стану (події) та вихідного документа. У кожній функції завжди має бути подія на виході (стан процесу), а документ/інформація відображається окремо від самого потоку.

Перевантаження Діаграми: Відображення усіх умов, обмежень, документів, ІС та модулів на одній діаграмі. Це призводить до надмірної складності, громіздкості та втрати ефективності моделі.

Питання для самоперевірки, тести для самоконтролю

Яке ключове обмеження eEPC заважає використовувати її для виявлення проблем з навантаженням співробітників?

Який логічний оператор є єдино допустимим при розгалуженні події на дві наступні функції?

Яка поширена помилка при використанні додаткових елементів призводить до втрати контролю над процесом?

Поясніть, чому використання логічного оператора «ІЛІ» (OR) після події є синтаксичною помилкою.

Тест (Виберіть одну правильну відповідь):

1. Типова помилка eEPC — це використання ланцюжка:

- а) \text{Подія} \to \text{Функція} \to \text{Подія}
- б) \text{Подія} \to \text{AND} \to \text{Функція}
- в) \text{Подія} \to \text{OR} \to \text{Функція}

2. Яка з перерахованих переваг eEPC дозволяє проводити імітаційне моделювання?

- а) Можливість додавати авторські елементи.
- б) Належність до нотацій Workflow.
- в) Використання логічних операторів.

3. Якщо функція містить п'ять окремих кроків, які аналітик об'єднав в одну функцію, це може призвести до:

- а) Неможливості виявити непотрібні кроки або відсутні залежності.
- б) Автоматичного спрощення логіки виконання.
- в) Необхідності обов'язково використовувати оператор AND.

Обов'язкові та додаткові задачі, приклади

Обов'язкові задачі

Задача (Синтаксична Помилка): Виправте некоректний ланцюжок:
\text{Подія «Запит надіслано»} \to \text{Функція «Обробити запит»} \to
\text{Функція «Сформувати відповідь»}.

Відповідь-приклад: \text{Подія «Запит надіслано»} \to \text{Функція
«Обробити запит»} \to \text{Подія «Запит оброблено»} \to \text{Функція
«Сформувати відповідь»} \to \text{Подія «Відповідь сформовано»}.

Задача (Логічна Помилка): У процесі є функція «Перевірити наявність товару». Якщо товару немає, запускається Подія «Потрібно замовити». Якщо товар є, запускається Подія «Товар готовий до відвантаження». Для розгалуження ви використали Оператор OR. Поясніть, чому це помилка, і як її виправити.

Відповідь-приклад: Це помилка, оскільки OR дозволяє одночасне настання обох подій або однієї з них, тоді як насправді відбувається ТІЛЬКИ ОДНА подія. Виправити: замінити Оператор OR на Оператор «Виключне ІЛІ» (XOR), оскільки наявність/відсутність товару є взаємовиключною умовою.

Додаткові задачі (Приклади)

Приклад (Команда): Команда аналітиків розпочала моделювання eEPC без попереднього узгодження методичного фільтра. Які проблеми можуть виникнути при читанні та агрегації їхніх діаграм?

Приклад (Контроль): В IDEF0 є обов'язкова стрілка Управління. Як аналітик, що моделює в eEPC, ви можете компенсувати ризик втрати контролю, якщо власник процесу наполягає на мінімалістичній моделі?

Бібліографічний список

1. Scheer, August-Wilhelm. ARIS – Business Process Modeling. 2nd Edition. Springer, 2000. (Містить детальний опис правил eEPC та обговорює типові помилки, особливо ті, що стосуються порушення чергування та коректного використання логічних операторів.)

2. Harmon, Paul. Business Process Change: A Guide for Business Process Management. 3rd Edition. Morgan Kaufmann, 2014. (Аналізує сильні та слабкі сторони eEPC, підкреслюючи, що вона є чудовим інструментом для workflow та імітації, але має обмеження в часовому аналізі.)

3. ПБА. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 3.0. International Institute for Business Analysis, 2015. (Наголошує на важливості узгодженості та прозорості моделі, що є ключовим для подолання недоліків, пов'язаних із гнучкістю нотації eEPC.)

Тема 6. Виконувані моделі процесів: нотація BPMN

Методологія BPM (Business Process Management – Управління бізнес-процесами) та нотація BPMN (Business Process Model and Notation – Модель і Нотація Бізнес-процесів) базуються на понятті бізнес-процесу та процесному підході до управління організацією.

Управління процесом у методології BPM — це управління цілим (організацією) через управління його частинами (процесами).

Щоб уникнути плутанини, важливо розрізняти ключові терміни:

Методологія BPM — концепція процесного управління організацією за допомогою управління бізнес-процесами, свого роду світогляд і підхід.

Нотація BPMN — система умовних позначень для побудови моделей процесів на основі методології BPM, які також можуть бути виконуваними за допомогою систем BPMS. Це методи та алгоритми для вирішення конкретних завдань.

Система BPMS (Business Process Management System) — IT-система для моделювання в методології BPM та автоматизації виконуваних екземплярів процесів на основі побудованих BPMN-моделей. Це готові прикладні рішення, що можна втілити.

Можливості BPMS-систем

BPMS-системи реалізують низку важливих можливостей для управління бізнес-процесами:

Створення, зміна, відтворення моделей процесів організації.

Швидке та легке управління послідовністю дій усередині процесу та їх оптимізація.

Управління взаємодіями між різними етапами та операціями.

Повноцінний реінжиніринг процесів в організації за потреби докорінних змін.

Збір статистики з виконання процесів для подальшого аналізу.

Виявлення відхилень та затримок під час виконання процесів.

Аналіз поточної діяльності організації відповідно до встановлених KPI (Ключових показників ефективності).

Відмінності Процесного та Функціонального Підходів

Управління організацією може здійснюватися на основі двох основних підходів:

Критерій	Функціональний Підхід (наприклад, IDEF0)	Процесний Підхід (BPM)
Орієнтація	Стратегічна, на бажаний результат ("Що саме ми хочемо отримати?")	Тактична, оперативна, на спосіб виконання ("Як це краще виконати?")
Фокус	Моделювання функцій ("чорних скриньок") та їх взаємодія	Послідовність дій всередині функцій, їх швидкість, трудомісткість та взаємодія
Мета	Стратегічне планування, оцінка роботи компанії в цілому	Управління послідовністю дій та оптимізація того, що відбувається всередині функцій
Зміни	Виходячи з цілого, змінюються об'єкти на вході/виході	Виходячи з частини (нижніх рівнів деталізації), змінюються елементи між "входом" і "виходом"

Створення опису бізнес-процесу починається "в цілому" (з верхнього рівня), після чого кожен процес ділиться на підпроцеси та деталізується.

Зміна бізнес-процесу, навпаки, починається з нижніх рівнів (максимальної деталізації).

При роботі з бізнес-процесами використовується багаторівнева декомпозиція, коли деталізація кожного функціонального блоку ("чорного ящика") виділяється в окремий процес. При цьому необхідно визначити оптимальний рівень деталізації: не надто загальний, але й не надто деталізований до дій кожного співробітника.

Основні Елементи Нотації BPMN

У нотації BPMN виділяють п'ять головних категорій елементів:

Основні елементи потоку:

Події (Events): початок, завершення або проміжні події в процесі (коло).

Процеси/Задачі (Activities): робота, яка виконується (прямокутник із заокругленими кутами).

Шлюзи (Gateways): точки розгалуження та злиття потоків (ромб).

З'єднувальні елементи:

Потоки управління (Sequence Flow): послідовність виконання задач (суцільна стрілка).

Потоки повідомлень (Message Flow): обмін інформацією між пулами (пунктирна стрілка).

Асоціації (Association): зв'язок елементів потоку з даними або артефактами (пунктирна лінія).

Дані:

Об'єкти даних (Data Objects).

Бази даних (Data Stores).

Зони відповідальності:

Пули (Pools): представляють учасників процесу або організації.

Доріжки (Lanes): підрозділи пулу, відповідальність певного підрозділу чи ролі.

Артефакти:

Виноски/Примітки (Text Annotations).

Види Діаграм BPMN

Нотація BPMN включає чотири основні види діаграм, з яких найчастіше використовуються перші два:

Процес (Process Diagram):

Призначення: Описує зміст і логіку одного бізнес-процесу у вигляді потоку задач, умов та подій. Це найпоширеніший вид.

Взаємодія (Collaboration Diagram):

Призначення: Моделює взаємодію (обмін даними) між двома або більше бізнес-процесами (Пулами). Для відображення використовуються Потоки повідомлень (Message Flow).

Хореографія (Choreography Diagram):

Призначення: Описує послідовність взаємодій учасників при виконанні бізнес-процесів, коли діаграми "Взаємодія" стають занадто складними.

Діалог (Conversation Diagram):

Призначення: Візуалізація взаємодій верхнього рівня (процесний ландшафт) між залученими сторонами. З'явився лише у BPMN 2.0.

Питання для Самоперевірки та Тести для Самоконтролю

Питання для Самоперевірки

У чому полягає основна відмінність між методологією BPM та нотацією BPMN?

Яку роль виконує система BPMS у концепції управління бізнес-процесами?

Назвіть три ключові можливості, які реалізують BPMS-системи.

Сформулюйте основну відмінність між функціональним та процесним підходами до управління організацією.

Як відрізняється послідовність дій при створенні опису бізнес-процесу та при його зміні (деталізація/узагальнення)?

Перерахуйте три основні категорії елементів потоку у нотації BPMN.

Які елементи нотації використовуються для відображення обміну інформацією між двома різними бізнес-процесами?

Який вид діаграм BPMN є найпоширенішим і описує логіку одного бізнес-процесу?

Тести для Самоконтролю

1. Що є основою для побудови моделей процесів у методології BPM?

- а) Функціональні блоки
- б) Бізнес-процес
- в) Організаційна структура
- г) Показники КРІ

2. Який з термінів відповідає системі умовних позначень для моделювання?

- а) BPM
- б) BPMS
- в) BPMN
- г) IDEF0

3. Який підхід до управління концентрується на відповіді "Як це краще виконати?"

- а) Функціональний
- б) Ієрархічний
- в) Процесний
- г) Стратегічний

4. Що представляють собою "Пули" (Pools) у нотації BPMN?

- а) Послідовність дій
- б) Точки розгалуження
- в) Зони відповідальності/Учасники процесу
- г) Об'єкти даних

5. Який вид діаграм BPMN використовується для моделювання взаємодії (обміну даними) між двома або більше процесами?

- а) Процес (Process Diagram)
- б) Хореографія (Choreography Diagram)
- в) Взаємодія (Collaboration Diagram)
- г) Діалог (Conversation Diagram)

Обов'язкові та Додаткові Задачі, Приклади

Обов'язкові Задачі

Аналіз термінології: Наведіть реальний приклад для ілюстрації відмінностей між BPM, BPMN та BPMS, використовуючи аналогію з будівництвом (наприклад, BPM — філософія управління будівництвом, BPMN — креслення та умовні позначення, BPMS — спеціалізоване програмне забезпечення для управління проектом).

Застосування підходів: Для процесу "Обробка замовлення клієнта" визначте, які аспекти будуть ключовими при його аналізі з точки зору:

Функціонального підходу (що важливо на вході/виході).

Процесного підходу (що важливо всередині "чорних скриньок").

Визначення елементів: Намалюйте або опишіть словесно, як виглядатимуть на діаграмі BPMN:

Старт процесу.

Діяльність "Перевірка наявності товару".

Шлюз, що визначає, чи є товар на складі, та розгалужує потік.

Потік повідомлень між відділом продажів та відділом складу.

Додаткові Задачі

Декомпозиція: Запропонуйте багаторівневу декомпозицію (мінімум 3 рівні) для бізнес-процесу "Найм нового співробітника", починаючи від верхнього рівня ("Управління персоналом") до деталізованих задач (наприклад, "Проведення співбесіди").

Порівняння діаграм: Поясніть, у яких випадках доцільніше використовувати діаграму "Взаємодія", а коли може знадобитися

"Хореографія". Наведіть приклад, що вимагає "Хореографії" (наприклад, послідовна взаємодія між клієнтом, банком та страховою компанією).

Довідково-інформаційні Дані

Таблиця Основних Елементів BPMN

Категорія	Назва Елемента	Символ	Опис
Елементи потоку	Подія (Event)	\bigcirc	Вказує на те, що відбувається (початок, кінець, проміжний стан)
	Задача/Діяльність (Task/Activity)	\square (заокруглені кути)	Вказує на роботу, яка виконується
	Шлюз (Gateway)	\diamond	Визначає точки розгалуження та злиття потоків (умовність, паралельність)
З'єднувальні елементи	Потік управління (Sequence Flow)	\longrightarrow (суцільна стрілка)	Показує послідовність виконання задач в одному процесі
	Потік повідомлень (Message Flow)	\text{-- - - } \text{\textcircled{\text{O}}} \longrightarrow (пунктирна стрілка з колом та стрілкою)	Показує обмін інформацією між різними Пулами
	Асоціація (Association)	\text{-- - -} (пунктирна лінія)	Зв'язок між елементами потоку та артефактами/даними
Зони відповідальності	Пул (Pool)	\square (велика рамка)	Учасник бізнес-процесу (організація, відділ)
	Доріжка (Lane)	\square (підрозділ Пулу)	Роль або підрозділ усередині Пулу

Бібліографічний Список

1. OMG. Business Process Model and Notation (BPMN) Version 2.0.2. Formal Specification.
2. Романова, А. В. (2018). Моделювання бізнес-процесів: Навчальний посібник. Київ: ЦУЛ.
3. Девіденко, О. В. (2020). Управління бізнес-процесами: Навчально-методичний посібник. Львів: Видавництво Львівської політехніки.

Тема 7. Типи зв'язків між елементами діаграми BPMN

Типи Зв'язків

У таблиці BPMN використовуються різні типи з'єднувальних елементів для встановлення логіки та взаємодії.

Категорія	Назва Зв'язку	Символ	Правила використання
Послідовність	Потік Управління (Sequence Flow)	<code>\longrightarrow</code> (суцільна стрілка)	Використовується тільки всередині одного Пулу або Доріжки. Визначає послідовність дій.
Обмін даними	Потік Повідомлень (Message Flow)	<code>\text{-- - -}</code> <code>\text{\textcircled{\text{O}}}</code> <code>\longrightarrow</code> (пунктирна стрілка з колом)	Використовується тільки між різними Пулами (процесами). Відображає обмін інформацією.
Асоціація	Асоціація (Association)	<code>\text{-- - -}</code> (пунктирна лінія)	Використовується для зв'язку елементів потоку з даними або артефактами.

Декомпозиція та Підпроцес: Ключові Відмінності

У професійному середовищі важливо чітко розрізняти поняття декомпозиції та підпроцесу.

Декомпозиція — це загальне поняття, що означає розкладання задачі на простіші елементи. Вона може використовуватися як у функціональному

(наприклад, IDEF0), так і в процесному моделюванні. Декомпозиція дозволяє поєднувати різні підходи до вивчення бізнесу, коли для розуміння цілого достатньо "чорного ящика" з назвою функції, а деталізація виконується окремо, можливо, іншими інструментами.

Підпроцес (Sub-Process) — це окремий процес усередині іншого процесу, невід'ємна частина BPMN-нотації. Його графічно малюють на тій самій діаграмі, але всередині очерчених меж. Для підпроцесу жорстко задані всі точки входу, виходу, виконавці та інструменти в рамках нотації BPMN.

Характеристика	Декомпозиція	Підпроцес (BPMN)
Використання	Загальний принцип, може поєднувати підходи (наприклад, IDEF0 + BPMN)	Елемент виключно нотації BPMN
Обмеження	Допускає більше свобод у виборі інструментів та нотацій для деталізації	Жорстко задані вхід/вихід/елементи відповідно до правил BPMN
Графіка	Виноситься на окрему схему/документ	Зображується на тій самій діаграмі (згорнутий, розгорнутий)

Використання Підпроцесів у BPMN

Підпроцеси використовуються в основному для виконуваних процесів та мають дві ключові цілі:

Декомпозиція та Читабельність: Не перевантажувати діаграму верхнього рівня деталями, полегшуючи її сприйняття (найчастіше використовується згорнутий підпроцес — прямокутник з маркером "+").

Повторювані Дії: Одноразово описаний підпроцес може багаторазово викликатися (використовуватися) всередині різних процесів.

Переваги підпроцесів:

Локалізація змін: Зміни вносяться в рамках окремого підпроцесу, не порушуючи логіку основного процесу.

Контроль нотації: Система BPMS допомагає уникнути помилок, оскільки робота ведеться всередині жорстко заданих правил BPMN.

Приклад декомпозиції процесу «Набір персоналу»:

Процес верхнього рівня може включати підпроцеси:

Пошук кандидатів на вакансію (Підпроцес 1)

Оформлення документів нового співробітника (Підпроцес 2)

Навчання нового співробітника (Підпроцес 3)

Правила Побудови Моделі (Приклад: Забезпечення Заявки на Потребу)

При моделюванні необхідно:

Визначити межі: Чітко окреслити точку входу (початок процесу, наприклад, "Отримання заявки") та точку виходу (завершення, наприклад, "Отримання замовлених товарів співробітником").

Регулювати деталізацію: Аналітик сам визначає глибину деталізації. Деякі менш важливі або допоміжні дії (наприклад, оплата товару, узгодження ціни) можуть бути опущені або винесені у згорнуті підпроцеси.

Використання елементів: Застосовувати шлюзи для розгалужень ("Якщо потрібність підтверджена...") та підпроцеси для комплексних дій ("Зарезервувати товар") з метою підвищення читабельності.

Переваги такого опису:

Наочність: Демонстрація функціонального зв'язку між підрозділами.

Інформативність: На схемі видно послідовність, джерела інформації та необхідні права доступу користувача.

Поширені Помилки При Моделюванні в BPMN

Помилки поділяються на три види: формальні (не відповідають правилам нотації), стилю (незручно читати/редагувати) та логіки (невідповідність суті процесу/реальній ситуації).

Найпоширеніші помилки:

Неправильне використання подій: Слід переконатися, що символ події (наприклад, конверт для обміну повідомленнями) правильно відображає його функцію, а не просто задачу оповіщення (email).

Елементи нічим не закінчуються: Хоча формально "покинуті" елементи (без виходу) можливі, вони викликають плутанину. Краще, щоб із задачі завжди був вихід або перехід до наступного блоку, що завершує потік.

Переплутані типи потоків:

Потоки управління (→) — ТІЛЬКИ ВНУТРІШНЬОПУЛЬНІ. Не можуть виходити за межі Пулу.

Потоки повідомлень ($\text{--} \quad \text{---} \quad \text{---}$) — ТІЛЬКИ МІЖ ПУЛАМИ. Не можуть бути всередині одного Пулу.

Не сходиться кількість потоків на вході та на виході: Наприклад, якщо ексклюзивний шлюз (XOR) має два виходи, але лише один вхід, або навпаки. Це може виправити додавання відповідного шлюзу.

Не враховується швидкість виконання залежних процесів: Якщо один процес залежить від повідомлення іншого, але невідомо, чи готовий верхній процес прийняти повідомлення від нижнього, необхідно явно позначити умову виконання або використовувати проміжні події.

Різний рівень задач у процесі: На одній схемі мають бути задачі приблизно одного рівня деталізації для збереження стилю та читабельності.

Питання для Самоперевірки та Тести для Самоконтролю

Питання для Самоперевірки

Яке основне правило використання Потoku Управління (Sequence Flow) щодо меж Пулу?

Яке основне правило використання Потoku Повідомлень (Message Flow) щодо меж Пулу?

Поясніть, чому для виконуваних процесів рекомендується використовувати підпроцес, а для не виконуваних — декомпозицію.

Назвіть графічне зображення та функцію згорнутого підпроцесу.

Наведіть приклад формальної помилки в нотації BPMN, пов'язаної з потоками.

Чому вважається помилкою, коли елемент на діаграмі нічим не закінчується?

Тести для Самоконтролю

1. Що є основною відмінністю між Декомпозицією та Підпроцесом у контексті BPMN?

а) Декомпозиція — це завжди функціональний підхід, підпроцес — процесний.

б) Підпроцес — це окремий процес на іншій сторінці, декомпозиція — на цій самій.

в) Підпроцес — це елемент BPMN-нотації, жорстко заданий її правилами, а Декомпозиція — загальний принцип, що допускає свободу.

г) Декомпозиція використовується лише для складних, а підпроцес — для простих задач.

2. Який тип потоку використовується для зв'язку між двома різними Пулами (Учасниками)?

а) Потік Управління

б) Потік Повідомлень

в) Потік Асоціацій

г) Умовний Потік

3. Як графічно позначається згорнутий підпроцес?

а) Прямокутник із позначкою "S"

б) Коло із позначкою "P"

в) Прямокутник із маркером "+"

г) Ромб із позначкою "Sub"

4. Якщо на одній діаграмі знаходяться задачі типу "Заповнити форму" та "Розробити стратегію маркетингу", про яку поширену помилку йдеться?

а) Помилка логіки

- б) Формальна помилка
- в) Помилка стилю/Різний рівень задач
- г) Помилка використання подій

Обов'язкові та Додаткові Задачі, Приклади

Обов'язкові Задачі

Розмежування потоків: Намалюйте умовну схему з двома Пулами ("Відділ Продажів" та "Склад") та зобразіть на ній:

Потік управління, що веде до задачі "Скласти рахунок" (всередині "Відділу Продажів").

Потік повідомлень, що ініціюється "Відділом Продажів" до "Складу" ("Запит на наявність").

Визначення помилки: Наведіть приклад ситуації, коли Потік Управління використовується для зв'язку між двома різними Пулами. Визначте тип помилки (формальна, стилю, логіки) та поясніть, як її виправити.

Підпроцес vs Декомпозиція: Запропонуйте, як можна було б використати декомпозицію замість підпроцесу "Зарезервувати товар" у наведеному прикладі процесу "Забезпечення заявки на потребу" і поясніть, чому це ускладнить роботу, якщо процес має бути виконуваним.

Додаткові Задачі

Аналіз залежності швидкості: Розгляньте приклад залежних процесів: Процес А надсилає повідомлення "Готовність до переговорів", а Процес Б повинен його прийняти. Як можна використати проміжні події (Intermediate Events), щоб позначити обов'язкову умову виконання і уникнути помилки швидкості виконання?

Редагування стилю: Перемалюйте (опишіть) частину процесу з великою кількістю вхідних/вихідних потоків, щоб продемонструвати, як

використання ексклюзивного шлюзу (Exclusive Gateway) може покращити читабельність та усунути потенційну логічну помилку.

Довідково-інформаційні Дані

Схема Поширених Помилоч Потоків

Зображення	Помилка	Коректне Рішення
Потік Управління виходить за межі Пулу	Формальна помилка. Потік управління виходить за межі Пулу.	Використовувати Потік Повідомлень для зв'язку між Пулами.
Потік Повідомлень використовується всередині Доріжки	Формальна помилка. Потік повідомлень використовується всередині Пулу/Доріжки.	Використовувати Потік Управління всередині Пулу/Доріжки.
Завдання має 2 входи, але 3 виходи без Шлюзу	Формальна/Логічна помилка. Не сходиться кількість потоків.	Додати Шлюз (Exclusive, Parallel, Inclusive) перед виходами для визначення логіки розгалуження.

Бібліографічний список

1. OMG. (2011). Business Process Model and Notation (BPMN) Version 2.0.2. Formal Specification. Object Management Group. (Офіційний міжнародний стандарт нотації BPMN).
2. Романова, А. В. (2018). Моделювання бізнес-процесів: Навчальний посібник. Київ: Центр учбової літератури (ЦУЛ).
3. Девіденко, О. В. (2020). Управління бізнес-процесами: Навчально-методичний посібник. Львів: Видавництво Львівської політехніки.
4. White, S., & Miers, D. (2008). BPMN Method and Style: A common business language for process modeling. Future Strategies Inc. (Одна з класичних робіт, присвячених практиці використання BPMN).
5. Harmon, P. (2019). Business Process Change: A Guide for Business and IT Leaders. Elsevier. (Широке охоплення методології BPM та її реалізації).

РОЗДІЛ 6. СТАЦІОНАРНІ ЧАСОВІ РЯДИ: ПОНЯТТЯ, ДІАГНОСТИКА ТА РЕАЛІЗАЦІЯ

(Квашук Д.М., к.е.н., доцент, доцент кафедри електричної інженерії,
енергоменеджменту та мехатроніки Київського авіаційного Інституту)

Тема 1. Поняття стаціонарності

У задачах аналізу часових рядів важливо розрізняти процеси зі **сталими статистичними властивостями** та процеси, характеристики яких **змінюються з часом** (через тренд, сезонність, структурні зміни, змінну дисперсію). Поняття *стаціонарності у широкому сенсі* (weak / covariance stationarity) формалізує “сталість” ряду на рівні першого та другого моментів і є базовим припущенням для моделей AR/MA/ARMA.

Стаціонарність (у широкому сенсі) — це властивість часового ряду $\{x_t\}$, за якої: - математичне сподівання є сталим у часі; - дисперсія є сталою у часі; - коваріація між x_t та x_{t+h} залежить тільки від лага h , а не від конкретного моменту t .

На рисунку 6.1 зображено реалізацію стаціонарного часового ряду $\{x_t\}$, значення якого коливаються навколо сталого математичного сподівання $E(x_t) = m = \text{const}$. Горизонтальна лінія відповідає середньому рівню ряду. Видно, що середній рівень не змінюється з часом, а амплітуда коливань залишається сталою на всьому інтервалі спостережень.

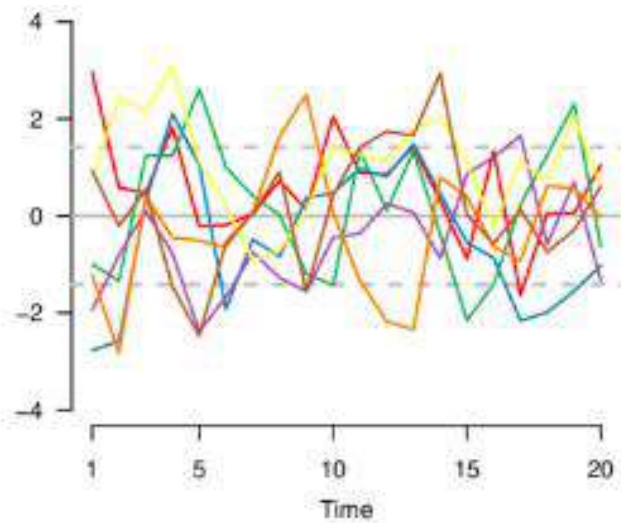


Рисунок 6.1 — Стаціонарний часовий ряд зі сталим математичним сподіванням і дисперсією

На графіку (Рис. 6.2) подано часовий ряд разом із середнім значенням та умовним коридором розсіювання. Сталість ширини коридору свідчить про виконання умови $\text{Var}(x_t) = \text{const}$.

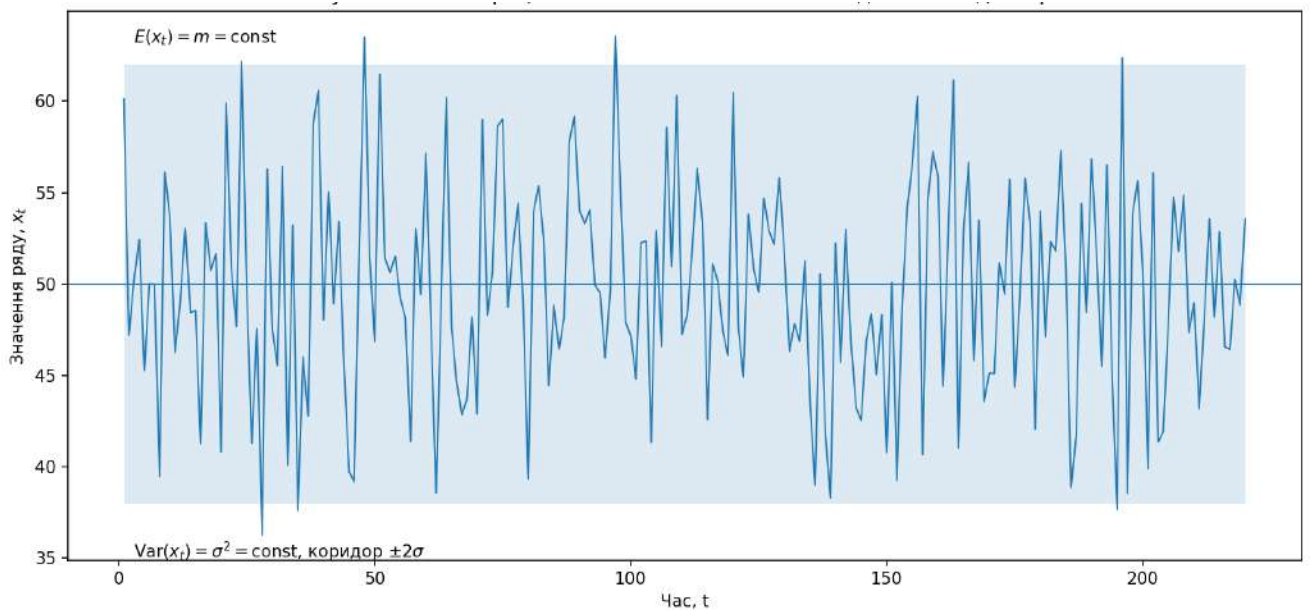


Рисунок 6.2 — Ілюстрація сталості математичного сподівання та дисперсії

Відсутність розширення або звуження коридору підтверджує відсутність гетероскедастичності та наявність стаціонарності у широкому сенсі.

На рисунку 6.3 наведено автоковаріаційну функцію часового ряду $\text{cov}(x_t, x_{t+h}) = g(h)$, де h — лаг між спостереженнями. Значення коваріації залежать лише від лага h і не залежать від конкретного моменту часу t , що є визначальною умовою стаціонарності.

На Рис. 6.4 подано автокореляційну функцію стаціонарного часового ряду. Значущі значення автокореляції спостерігаються лише для малих лагів, після чого кореляція швидко затухає. По осі абсцис відкладено лаг h , по осі ординат — автокореляцію $\rho(h)$. Значущі значення автокореляції спостерігаються для малих лагів, після чого відбувається швидке затухання.

Така форма ACF є типовою для стаціонарних процесів (зокрема AR-процесів) і використовується для: діагностики стаціонарності; попереднього вибору структури моделей класу ARMA.

Характер ACF є типовим для стаціонарних процесів і використовується для діагностики стаціонарності та попереднього вибору моделей класу ARMA.

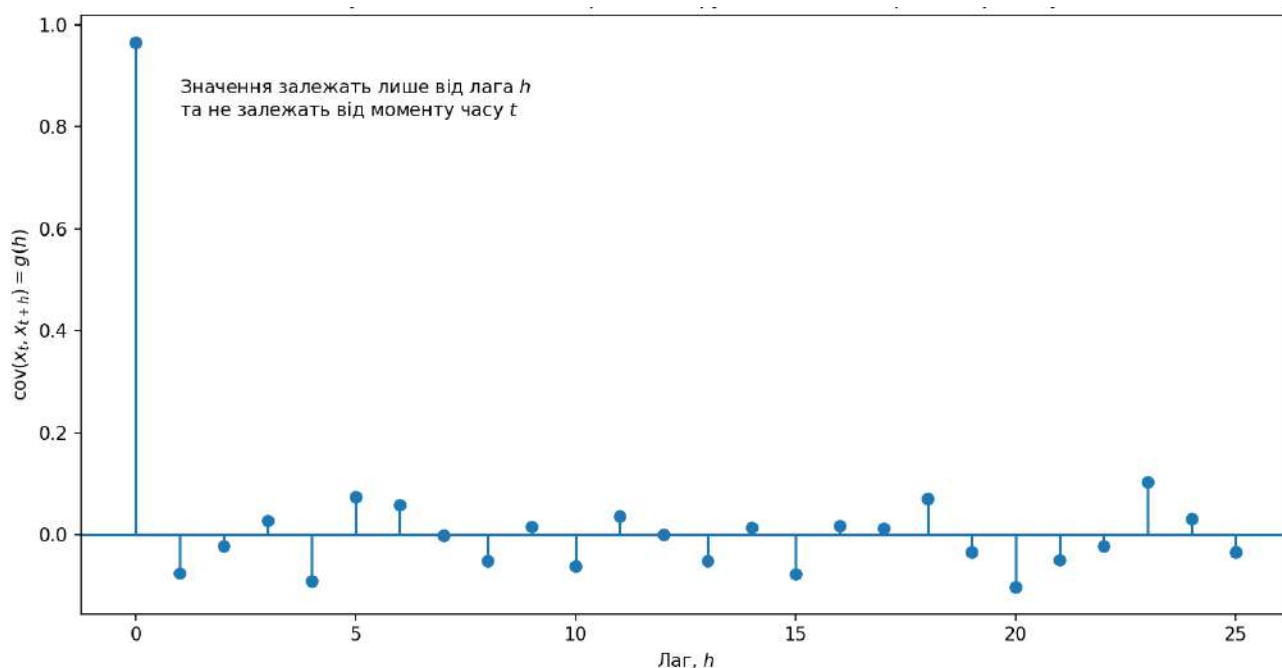


Рисунок 6.3 — Автоковаріаційна функція стаціонарного процесу

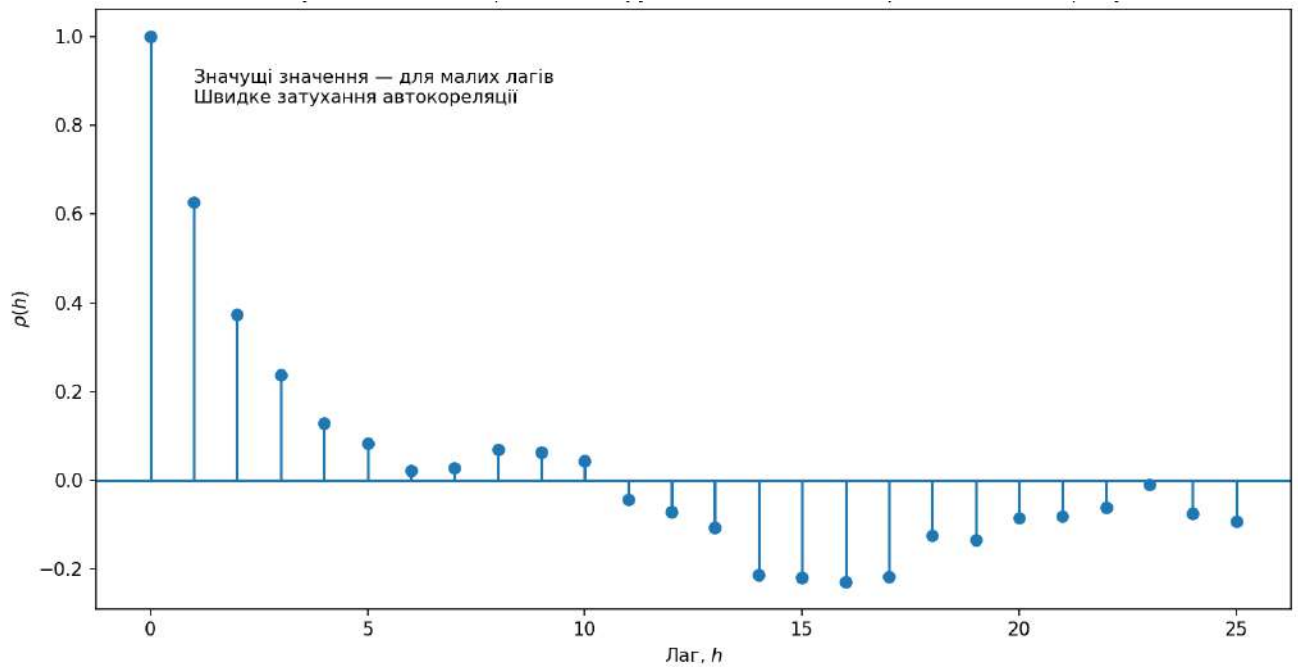


Рисунок 6.4 — Автокореляційна функція (ACF) стаціонарного часового ряду

Сукупність наведених графіків наочно демонструє виконання умов стаціонарності: сталість математичного сподівання, сталість дисперсії та залежність коваріації виключно від лага.

Розглянемо певні умови.

Умова 1: сталість математичного сподівання

$$E(x_t) = m, \quad m = \text{const.}$$

де: - x_t — значення часового ряду в момент часу t ; - $E(x_t)$ — оператор математичного сподівання; - m — сталий середній рівень ряду (не залежить від t).

Умова 2: коваріація залежить тільки від лага

$$\text{cov}(x_t, x_{t+h}) = g(h).$$

де: $\text{cov}(x_t, x_{t+h})$ — коваріація двох випадкових величин; - h — лаг (ціле число, може бути додатним, нульовим або від'ємним); - $g(h)$ — автоковаріаційна функція (залежить лише від лага).

Наслідок: сталість дисперсії

Оскільки дисперсія — це коваріація змінної із самою собою, для $h = 0$ маємо:

$$\text{Var}(x_t) = \text{cov}(x_t, x_t) = g(0) = \text{const.}$$

де: $\text{Var}(x_t)$ — дисперсія ряду у момент часу t ; $g(0)$ — значення автоковаріаційної функції при нульовому лагу.

Функцію $g(h)$ називають автоковаріаційною функцією. Вона визначена для додатних і від'ємних лагів та є парною:

$$g(-h) = g(h).$$

де: $g(-h)$ — коваріація між x_t та x_{t-h} ; $g(h)$ — коваріація між x_t та x_{t+h} .

Для довільних моментів t, s коваріація залежить тільки від різниці $t - s$:

$$\text{cov}(x_t, x_s) = g(t - s).$$

де: t, s — моменти часу; $t - s$ — лаг між ними (із урахуванням знака).

Автокореляція, ACF, PACF та «пам'ять» ряду

У практиці ідентифікації моделей (наприклад, AR/MA/ARMA) центральну роль відіграє аналіз залежностей між значеннями ряду на різних лагах. Для стаціонарного ряду ці залежності зручно описувати через **автокореляцію** та функції **ACF/PACF**.

Для стаціонарного ряду коефіцієнт кореляції між x_t та x_{t+h} має вигляд:

$$\text{corr}(x_t, x_{t+h}) = \frac{\text{cov}(x_t, x_{t+h})}{\sqrt{\text{Var}(x_t)\text{Var}(x_{t+h})}} = \frac{g(h)}{g(0)}.$$

де: $\text{corr}(x_t, x_{t+h})$ — кореляція; $g(h)$ — автоковаріація на лагу h ; $g(0)$ — дисперсія (автоковаріація на лагу 0).

Позначимо автокореляційну функцію:

$$r(h) = \text{corr}(x_t, x_{t+h}) = \frac{g(h)}{g(0)}.$$

де: $r(h)$ — значення ACF на лагу h .

Звідси випливає: - $r(0) = 1$; - для будь-яких s, t :

$$\text{corr}(x_s, x_t) = r(s - t).$$

Корелограма — це графік значень $r(h)$ для набору лагів $h = 0, 1, \dots, H$.

Гранична властивість ACF (інтерпретація «забування»)

Для багатьох стаціонарних процесів виконується умова:

$$\lim_{h \rightarrow \pm\infty} r(h) = 0.$$

де: - $r(h)$ — автокореляція на лагу h .

Інтерпретація: зі зростанням лага залежність між значеннями ряду слабшає, тобто ряд “забуває” минуле.

Часткова автокореляція (PACF)

Часткова автокореляція на лагу h — це кореляція між x_t та x_{t+h} після вилучення впливу проміжних значень $x_{t+1}, \dots, x_{t+h-1}$:

$$r_{\text{part}}(h) = \text{corr}(x_t, x_{t+h} \mid x_{t+1}, \dots, x_{t+h-1}).$$

де: - $r_{\text{part}}(h)$ — значення PACF на лагу h .

Зокрема: - $r_{\text{part}}(0) = 1$, - $r_{\text{part}}(1) = r(1)$, - часто також:

$$\lim_{h \rightarrow \infty} r_{\text{part}}(h) = 0.$$

PACF є ключовим інструментом для визначення порядку AR-моделі.

Білий шум як базова стаціонарна модель

Побудову складніших моделей стаціонарних рядів зручно починати з найпростішої моделі випадкових коливань — **білого шуму**. Білий шум часто інтерпретують як *інновації* або *шоки*.

Визначення (білий шум).

Ряд $\{u_t\}$ називають білим шумом, якщо:

$$E(u_t) = 0, \quad \text{Var}(u_t) = E(u_t^2) = \sigma_u^2, \quad \text{cov}(u_t, u_{t+h}) = 0 \text{ для } h \neq 0.$$

де: - u_t — інновація (випадковий шок) у момент часу t ; - $E(u_t)$ — математичне сподівання (для білого шуму дорівнює нулю); - σ_u^2 — дисперсія інновацій; - $\text{cov}(u_t, u_{t+h})$ — коваріація інновацій на різних лагах.

Позначення: $u_t \sim WN(0, \sigma_u^2)$.

Для білого шуму автоковаріація та ACF мають вигляд:

$$g_u(h) = \begin{cases} \sigma_u^2, & h = 0, \\ 0, & h \neq 0, \end{cases} \quad r_u(h) = \begin{cases} 1, & h = 0, \\ 0, & h \neq 0. \end{cases}$$

де: - $g_u(h)$ — автоковаріаційна функція білого шуму; - $r_u(h)$ — автокореляційна функція білого шуму.

Інтерпретація: білий шум “миттєво забуває” минулі значення.

Моделі стаціонарних рядів: AR, MA, ARMA

Після введення стаціонарності та білого шуму природним наступним кроком є опис класів моделей, які відтворюють типові структури залежностей у стаціонарних рядах. Класичними є **AR**, **MA** та **ARMA**. Ключова ідея: стаціонарний ряд можна описати як взаємодію: - **інерційної компоненти** (вплив власного минулого); - **інновацій** u_t (білий шум) та їх лагів.

Лаговий оператор

Для компактного запису моделей вводять лаговий оператор L :

$$L(x_t) = x_{t-1}, \quad L^k(x_t) = x_{t-k}, \quad L^0(x_t) = x_t.$$

де: - L — оператор зсуву на один крок назад у часі; - k — порядок лага.

Загальний вигляд ARMA(p, q)

Модель ARMA(p, q) задається рівнянням:

$$x_t = c + \sum_{j=1}^p a_j x_{t-j} + u_t + \sum_{s=1}^q b_s u_{t-s}, \quad u_t \sim WN(0, \sigma_u^2).$$

де: - x_t — значення стаціонарного ряду у момент t ; - c — стала (константа) моделі; - p — порядок авторегресійної частини (AR); - a_j — AR-коефіцієнти (вплив x_{t-j} на x_t); - q — порядок ковзного середнього (MA); - b_s — МА-коефіцієнти (вплив минулих шоків u_{t-s}); - u_t — інновація (білий шум); - σ_u^2 — дисперсія інновацій.

Інтерпретація: поточне значення x_t залежить від p попередніх значень ряду та від поточного/попередніх шоків u_t до лага q .

Умова стаціонарності ARMA через корені AR-полінома

Нехай авторегресійний поліном буде:

$$a(z) = 1 - a_1 z - \dots - a_p z^p.$$

де: - z — комплексна змінна; - $a(z)$ — AR-поліном, що визначає властивості AR-частини.

Умова стаціонарності: модель ARMA задає стаціонарний ряд тоді і тільки тоді, коли всі корені рівняння $a(z) = 0$ за модулем більші за 1.

Окремий випадок ARMA(1, q):

$$x_t = c + a x_{t-1} + u_t + \sum_{s=1}^q b_s u_{t-s} \Rightarrow |a| < 1.$$

де: - a — AR(1)-коефіцієнт.

Середнє стаціонарної ARMA

Для стаціонарної ARMA (враховуючи $E(u_t) = 0$ та сталість $E(x_t)$) маємо:

$$(1 - a_1 - \dots - a_p) E(x_t) = c, \quad E(x_t) = \frac{c}{1 - \sum_{j=1}^p a_j} = \frac{c}{a(1)}.$$

де: - $E(x_t)$ — середнє стаціонарного ряду; - $a(1) = 1 - a_1 - \dots - a_p$ — значення AR-полінома при $z = 1$.

Діагностика стаціонарності та адекватності моделі

Перед оцінюванням моделі важливо перевірити, чи узгоджуються дані з припущеннями стаціонарності (або чи потрібні перетворення). Після оцінювання — чи “вичерпала” модель залежності в даних.

Візуальна діагностика

На етапі первинного аналізу доцільно оцінити: - наявність/відсутність тренду (стабільність середнього); - “розкид” у часі (стабільність дисперсії); - характер автозалежностей (ACF/PACF).

Перевірка залишків: тест Льюнга–Бокса (Ljung–Box)

Після підбору AR/MA/ARMA важливо, щоб залишки $\hat{u}_t$ поведилися як білий шум. Тест Льюнга–Бокса перевіряє гіпотезу відсутності серійної кореляції в залишках:

$$H_0: r_{\hat{u}}(1) = \dots = r_{\hat{u}}(K) = 0.$$

де: - H_0 — нульова гіпотеза; - $r_{\hat{u}}(h)$ — автокореляція залишків на лагу h ; - K — максимальний лаг перевірки.

Тестова статистика:

$$Q = n(n+2) \sum_{h=1}^K \frac{\hat{r}_{\hat{u}}(h)^2}{n-h}.$$

де: - Q — статистика тесту Ljung–Box; - n — довжина вибірки; - $\hat{r}_{\hat{u}}(h)$ — оцінка ACF для залишків на лагу h .

У прикладних пакетах зазвичай повертають p -value, що дозволяє приймати/відхиляти H_0 на заданому рівні значущості.

Реалізація в Python (мінімальний практичний шаблон)

Нижче наведено типовий каркас: завантаження ряду, базова підготовка, візуальна діагностика, ACF/PACF, оцінювання ARMA та тест Ljung–Box для залишків. Вміст датасету знаходиться за даним посиланням [11].

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import os
os.chdir(r"C:\Users\1\Downloads\Telegram Desktop")
# -----
# 1) Завантаження, індексація, частота
# -----
df = pd.read_csv("series.csv", parse_dates=["date"])
```

```

df = df.sort_values("date").set_index("date")
# Виберіть частоту: "D" (дні), "M" (місяці), "H" (години) тощо
x = df["x"].asfreq("D")
# -----
# 2) Базова обробка пропусків (за потреби)
# -----
x = x.interpolate("time")
# -----
# 3) Візуальна діагностика
# -----
ax = x.plot(title="x_t")
ax.set_xlabel("t")
ax.set_ylabel("x")
plt.show()
# -----
# 4) Ковзні оцінки середнього/дисперсії
# -----
w = 30 # ширина вікна (наприклад, 30 днів)
roll_mean = x.rolling(window=w).mean()
roll_var = x.rolling(window=w).var()
plt.figure()
plt.plot(x.index, x.values, label="x")
plt.plot(roll_mean.index, roll_mean.values, label="rolling mean")
plt.legend()
plt.title("Mean stability check")
plt.show()
plt.figure()
plt.plot(roll_var.index, roll_var.values, label="rolling var")
plt.legend()

```

```

plt.title("Variance stability check")
plt.show()

# -----
# 5) ACF/PACF (statsmodels)
# -----

from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
plot_acf(x.dropna(), lags=40)
plt.title("ACF")
plt.show()
plot_pacf(x.dropna(), lags=40, method="ywm")
plt.title("PACF")
plt.show()

# -----
# 6) ARMA через ARIMA(p,0,q) і перевірка залишків Ljung–Box
# -----

from statsmodels.tsa.arima.model import ARIMA
from statsmodels.stats.diagnostic import acorr_ljungbox
# ARMA(1,1) = ARIMA(1,0,1)
model = ARIMA(x, order=(1, 0, 1))
res = model.fit()
u_hat = res.resid.dropna()
# Ljung–Box тест для залишків
lb = acorr_ljungbox(u_hat, lags=[10, 20], return_df=True)
print(lb)
# Графік залишків
u_hat.plot(title="Residuals")
plt.show()

```

Примітка. Якщо ACF повільно спадає та/або ковзні оцінки середнього чи дисперсії помітно змінюються у часі, то ряд, імовірно, **не є стаціонарним**

у рівнях. У такому випадку доцільно застосувати перетворення (логарифмування, різницювання, сезонне різницювання) або перейти до іншого класу моделей, що явно враховують тренд/несталість.

Зауваження: стаціонарність відносно тренду (орієнтир для діагностики)

У прикладних задачах часто трапляється ситуація, коли часовий ряд не є стаціонарним у рівнях через наявність *детермінованого тренду*. Водночас після вилучення тренду залишкова компонента може поводитися як стаціонарний процес. Такий випадок описують як *стаціонарність відносно тренду* (trend-stationarity).

Модель “тренд + стаціонарний залишок”

$$x_t = f(t) + v_t.$$

де: - x_t — значення часового ряду в момент часу t ; - $f(t)$ — детермінована функція часу (тренд), наприклад лінійний тренд $f(t) = \beta_0 + \beta_1 t$ або інша гладка функція; - v_t — стаціонарна випадкова компонента (залишок) з нульовим середнім.

Наслідок для математичного сподівання

Оскільки $f(t)$ змінюється з часом, то й математичне сподівання ряду залежить від t :

$$E(x_t) = f(t).$$

де: - $E(x_t)$ — математичне сподівання (середнє значення) ряду у момент t ; - $f(t)$ — трендова складова, що визначає зміну середнього в часі.

Наслідок для дисперсії

Якщо v_t є стаціонарним і $f(t)$ є детермінованою складовою, то дисперсія ряду може залишатися сталою та дорівнювати дисперсії стаціонарного залишку:

$$\text{Var}(x_t) = \text{Var}(v_t) = \text{const.}$$

де: - $\text{Var}(x_t)$ — дисперсія часового ряду у момент t ; - $\text{Var}(v_t)$ — стала дисперсія стаціонарної компоненти v_t .

Практичний висновок для діагностики та моделювання

Якщо візуально або за ковзними оцінками видно, що *середнє “пливе”*, але *розкид (дисперсія) є відносно стабільним*, доцільно:

- 1) спершу *вилучити тренд* (детрендувати: оцінити $f(t)$ і перейти до залишків $v_t = x_t - \hat{f}(t)$), або,
- 2) застосувати *різницювання* (як спосіб прибрати тренд), а вже після цього будувати стаціонарні моделі (AR/MA/ARMA) для отриманого стаціонарного компонента.

Прикладні практичні задачі

Дані: series_1.csv (колонки date, x) [12]. Для прикладних кейсів інтерпретуйте x_t як один із показників: - щоденний **попит/продажі** (шт. або грн), - **виручка**, - **курс** (умовно), - **ціна**, - **трафік** / кількість транзакцій, - **операційні витрати**.

У кожній задачі: побудуйте графіки, зробіть висновок про стаціонарність/перетворення, застосуйте ACF/PACF, оцініть модель (ARMA/ARIMA($\cdot$, 0, $\cdot$)), перевірте залишки (Ljung–Box) і дайте економічне рішення.

Завантаження даних (спільна для всіх задач)

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
df = pd.read_csv("series.csv",
parse_dates=["date"]).sort_values("date").set_index("date")
x = df["x"].asfreq("D").interpolate("time")
```

Задача 1. KPI-моніторинг: стабільність середнього та “зміна режиму” (контроль управління)

Ситуація

Компанія відстежує KPI (наприклад, виручку). Потрібно швидко зрозуміти: чи процес стабільний, чи відбулася зміна режиму (тренд/зміна волатильності).

Потрібно

- Побудувати графік x_t , ковзне середнє і ковзну дисперсію (вікна 14, 30, 60).
- Зробити висновок: чи можна вважати процес стаціонарним у рівнях?
- Якщо ні — запропонувати трансформацію (лог/різницювання) і повторити аналіз.

Результат

Виявлення нестабільного KPI → перегляд плану, бюджету, політик контролю.

КОД:

```
windows = [14, 30, 60]
plt.figure()
plt.plot(x.index, x.values)
plt.title("KPI  $x_t$ "); plt.xlabel("t"); plt.ylabel("x"); plt.show()
for w in windows:
    rm = x.rolling(w).mean()
    rv = x.rolling(w).var()
    plt.figure()
    plt.plot(x.index, x.values, label="x")
    plt.plot(rm.index, rm.values, label=f"rolling mean (w={w})")
    plt.legend(); plt.title("Mean stability"); plt.show()
```

```
plt.figure()  
plt.plot(rv.index, rv.values, label=f"rolling var (w={w})")  
plt.legend(); plt.title("Variance stability"); plt.show()
```

Задача 2. Управління запасами: “попит має пам’ять чи це шоки?” (ACF/PACF → політика запасів)

Ситуація

x_t — щоденний попит. Від “пам’яті” залежить стратегія запасів: інерційний попит дозволяє планувати, “шоковий” — вимагає більших буферів.

Потрібно

- Побудувати ACF/PACF для x_t та (за потреби) для Δx_t .
- За формою ACF/PACF запропонувати кандидата моделі: AR(1), AR(2), MA(1), ARMA(1,1) тощо.
- Пояснити, як форма ACF/PACF впливає на рішення про страховий запас.

Результат

Якщо інерція висока → планування закупівель “плавніше”.

Якщо домінують шоки → більший страховий запас.

КОД:

```
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf  
plot_acf(x.dropna(), lags=40); plt.title("ACF (levels)"); plt.show()  
plot_pacf(x.dropna(), lags=40, method="ywm"); plt.title("PACF (levels)");  
plt.show()  
dx = x.diff().dropna()  
plot_acf(dx, lags=40); plt.title("ACF (diff)"); plt.show()  
plot_pacf(dx, lags=40, method="ywm"); plt.title("PACF (diff)"); plt.show()
```


Задача 3. Планування бюджету: прогноз на 30 днів і бюджетна “подушка” (ARMA + інтервали)

Ситуація

x_t — щоденна виручка або витрати. Потрібно запланувати бюджет на 30 днів і оцінити ризик недофінансування.

Потрібно

- Оцінити модель ARMA(1,1) або кращу за AIC/BIC (але лише для стаціонарного ряду).
- Побудувати прогноз на 30 кроків вперед і довірчі інтервали.
- Розрахувати бюджетну “подушку” як квантиль ризику.

Приклад правила “подушки”

$$B = \sum_{i=1}^H \hat{x}_{T+i} + z_{\alpha} \sqrt{\sum_{i=1}^H \hat{\sigma}_{T+i}^2},$$

де: - H — горизонт планування; - $\hat{x}_{T+i}$ — прогноз; - $\hat{\sigma}_{T+i}$ — прогнозна стандартна похибка; - z_{α} — квантиль нормального розподілу.

КОД:

```
from statsmodels.tsa.arima.model import ARIMA
series = x # або dx, якщо різницювали
res = ARIMA(series, order=(1, 0, 1)).fit()
fc = res.get_forecast(steps=30)
mean = fc.predicted_mean
ci = fc.conf_int()
plt.figure()
plt.plot(series.index, series.values, label="history")
plt.plot(mean.index, mean.values, label="forecast")
plt.fill_between(ci.index, ci.iloc[:,0].values, ci.iloc[:,1].values, alpha=0.2,
```

```
label="CI")  
plt.legend(); plt.title("30-day forecast + CI"); plt.show()
```

Задача 4. Оцінка ефекту маркетингової акції: перетворення до стаціонарності + порівняння режимів

Ситуація

У певні дні була акція (*promo*). Треба відокремити “звичайну динаміку” від ефекту акції. У вас немає *promo* в даних, тому для навчання створіть її штучно.

Потрібно

- Створити бінарну змінну $promo_t$ (наприклад, 10 випадкових днів = 1, решта = 0).
- Оцінити, як $promo_t$ впливає на ряд у стаціонарній формі (наприклад, на $\Delta \log(x_t)$).
- Пояснити економічний ефект: “на скільки % зростає показник у день акції”.

Формула (еластичність при лог-моделі)

$$\Delta \log(x_t) = \beta_0 + \beta_1 promo_t + \varepsilon_t, \quad \Rightarrow \quad \text{ефект} \approx 100 \cdot \beta_1 \%$$

КОД:

```
import statsmodels.api as sm  
x_pos = x.clip(lower=1e-9)  
y = np.log(x_pos).diff().dropna() #  $\Delta \log(x_t)$   
promo = pd.Series(0, index=x.index)  
promo_days = np.random.choice(x.index[20:-20], size=10, replace=False)  
promo.loc[promo_days] = 1  
promo = promo.loc[y.index]  
X = sm.add_constant(promo)
```

```
ols = sm.OLS(y, X).fit()
print(ols.summary())
```

Задача 5. Управління валютним ризиком: “шоковість” vs “інерція” курсу

Ситуація

x_t — курс або індекс (умовно). Компанія має валютні платежі і хоче зрозуміти: чи можна прогнозувати зміни (інерція), чи це майже білий шум.

Потрібно

- Перейти до змін: $r_t = \Delta \log(x_t)$ (лог-доходність).
- Перевірити ACF r_t (часто близька до нуля).
- Порівняти з білим шумом і зробити висновок щодо можливості “короткого” прогнозу.
- Сформулювати просте правило хеджування: якщо ризик високий — фіксуємо курс частіше.

КОД:

```
from statsmodels.graphics.tsaplots import plot_acf
x_pos = x.clip(lower=1e-9)
r = np.log(x_pos).diff().dropna()
plot_acf(r, lags=40); plt.title("ACF of returns"); plt.show()
```

Задача 6. Виявлення аномалій у продажах: “викиди” та їх вплив на стаціонарність

Ситуація

У продажах бувають аномальні дні (помилки обліку, форс-мажор, великі разові угоди). Викиди спотворюють ACF/PACF та оцінки моделі.

Потрібно

- Виявити аномалії за робастним правилом (MAD або z-score від різниць).
- Порівняти ACF/PACF “до” і “після” обробки аномалій.
- Дати управлінську інтерпретацію: які аномалії є “операційними”, а які — “ринковими”.

Правило (Median Absolute Deviation)

$$z_t = \frac{x_t - \text{median}(x)}{1.4826 \cdot \text{MAD}(x)}, \quad \text{аномалія якщо } |z_t| > 3.5.$$

КОД:

```
med = x.median()
mad = (x - med).abs().median()
z = (x - med) / (1.4826 * mad + 1e-9)
anoms = z.abs() > 3.5
plt.figure()
plt.plot(x.index, x.values, label="x")
plt.scatter(x.index[anoms], x[anoms], label="anomalies")
plt.legend(); plt.title("Anomaly candidates"); plt.show()
x_clean = x.copy()
x_clean[anoms] = np.nan
x_clean = x_clean.interpolate("time")
```

Задача 7. Контроль якості моделі для прогнозу попиту: Ljung–Box → “чи можна довіряти прогнозу?”

Ситуація

Модель потрібна для рішень (закупівлі/план). Якщо залишки корельовані — модель “не забрала структуру”, прогноз систематично помиляється.

Потрібно

- Оцінити ARMA(1,1) (або іншу) на стаціонарному ряді.
- Побудувати залишки, ACF залишків.
- Виконати Ljung–Box (lags 10, 20).
- Якщо p -value мале — запропонувати дію: змінити порядок або трансформацію.

КОД:

```
from statsmodels.tsa.arima.model import ARIMA
from statsmodels.stats.diagnostic import acorr_ljungbox
from statsmodels.graphics.tsaplots import plot_acf
res = ARIMA(x, order=(1,0,1)).fit()
u = res.resid.dropna()
plot_acf(u, lags=40); plt.title("ACF residuals")
plt.show()
print(acorr_ljungbox(u, lags=[10,20], return_df=True))
```

Задача 8. Rolling backtest: оцінка точності прогнозу для планування (MAE/RMSE/sMAPE)

Ситуація

В економіці важливо не “один раз спрогнозувати”, а перевірити прогноз у режимі “як було б у реальності” — з послідовним донавчанням.

Потрібно

- Зробити rolling-origin backtest для горизонту 1 та 7 днів.
- Порахувати MAE, RMSE, sMAPE.
- Зробити висновок: чи достатня точність для рішення (наприклад, план закупівель)?

КОД:

```
from statsmodels.tsa.arima.model import ARIMA
def smape(y_true, y_pred):
    y_true = np.asarray(y_true); y_pred = np.asarray(y_pred)
    denom = (np.abs(y_true) + np.abs(y_pred)) + 1e-9
    return float(np.mean(2.0np.abs(y_pred - y_true)/denom))
def rolling_forecast(y, order=(1,0,1), start=200, horizon=1):
    preds, actual = [], []
    for t in range(start, len(y) - horizon + 1):
        train = y.iloc[:t]
        test = y.iloc[t:t+horizon]
        res = ARIMA(train, order=order).fit()
        fc = res.forecast(steps=horizon)
        preds.append(fc.iloc[-1])
        actual.append(test.iloc[-1])
    return np.array(actual), np.array(preds)
y_true, y_hat = rolling_forecast(x.dropna(), order=(1,0,1), horizon=1)
print("MAE:", np.mean(np.abs(y_true - y_hat)))
print("RMSE:", np.sqrt(np.mean((y_true - y_hat)2)))
print("sMAPE:", smape(y_true, y_hat))
```

Задача 9. Мініпроєкт: “прогноз → правило керування”

Ситуація

x_t — попит. Компанія хоче встановити правило поповнення запасів (rolling horizon).

Потрібно

- Побудувати прогноз попиту на 14 днів.
- Ввести правило керування запасом (safety stock) на основі невизначеності прогнозу.

- Порівняти два правила: “без прогнозу” vs “з прогнозом” (на історичних даних через backtest).

Приклад правила

$$S_t = \widehat{D}_{t:t+H} + k \cdot \widehat{\sigma}_{t:t+H},$$

де: - S_t — цільовий запас; - $\widehat{D}_{t:t+H}$ — прогноз сумарного попиту на горизонт H ; - $\widehat{\sigma}_{t:t+H}$ — прогнозна невизначеність; - k — параметр сервіс-рівня (наприклад, 1.65 для $\approx 95\%$).

Результат

Короткий текст: “який виграш у дефіцитах/надлишках” (навіть простими проксі-метриками).

Тема 2. Авторегресійні моделі AR(P)

Авторегресійні моделі є одним із базових класів моделей аналізу часових рядів, у яких поточне значення процесу пояснюється його власними минулими значеннями. Такі моделі широко застосовуються в економіці, фінансах, енергетиці та технічних системах для опису інерційної динаміки та прогнозування.

Визначення та інтерпретація

Модель авторегресії порядку p задається рівнянням:

$$x_t = c + \sum_{j=1}^p a_j x_{t-j} + u_t, \quad u_t \sim WN(0, \sigma_u^2).$$

де: - x_t — значення часового ряду в момент часу t ; - $x_{t-1}, \dots, x_{t-p}$ — попередні значення ряду; - a_j — авторегресійні коефіцієнти; - c — стала (зсув, або дрейф); - u_t — білий шум (інновація).

AR(p)-модель можна трактувати як дискретну динамічну систему з пам'яттю, у якій коефіцієнти a_j визначають інерційність процесу, а шум u_t формує його стохастичну природу. Графічно така поведінка проявляється у коливаннях навколо сталого середнього.

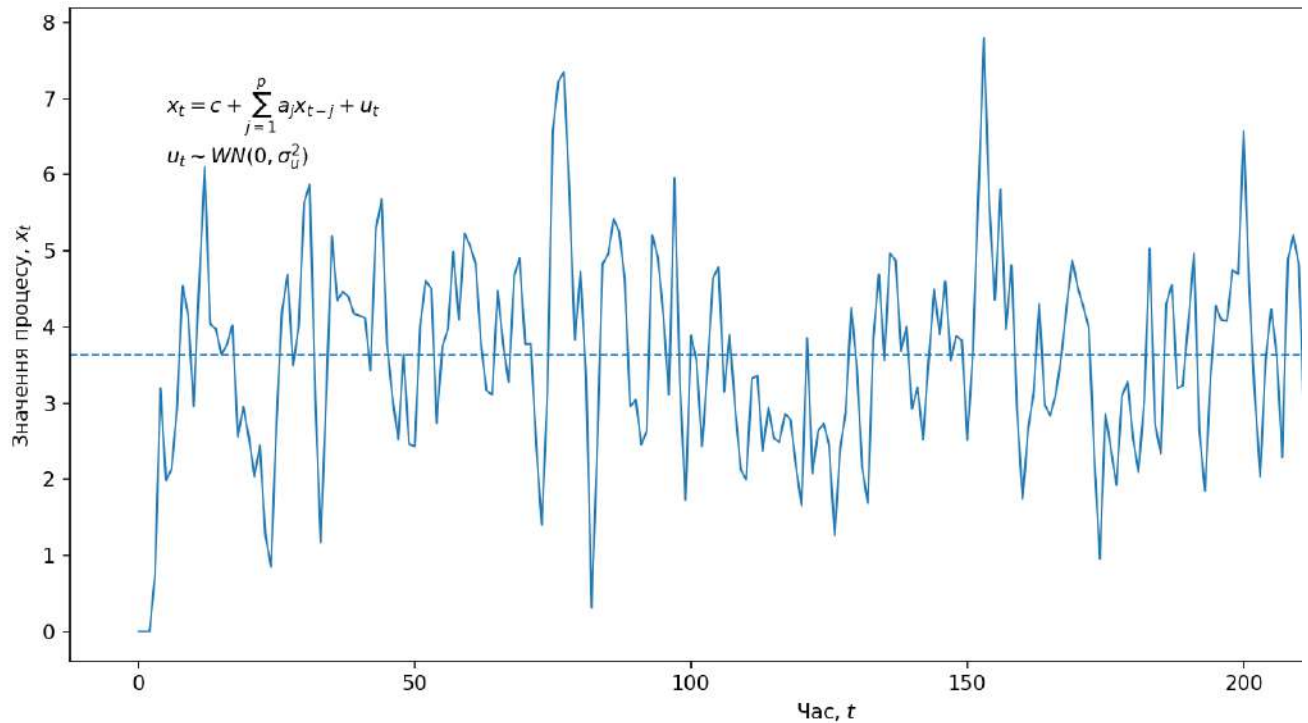


Рисунок 6.5 Коливання AR(p)-процесу навколо сталого середнього значення

AR(1) як базовий приклад

Найпростішим випадком є модель першого порядку:

$$x_t = c + ax_{t-1} + u_t.$$

Модель AR(1) часто використовується як базова для аналізу стаціонарності, кореляційної структури та прогнозування. Для неї характерне поступове затухання автокореляції, що добре ілюструється на *Рисунку 6.6 — ACF стаціонарного часового ряду.*

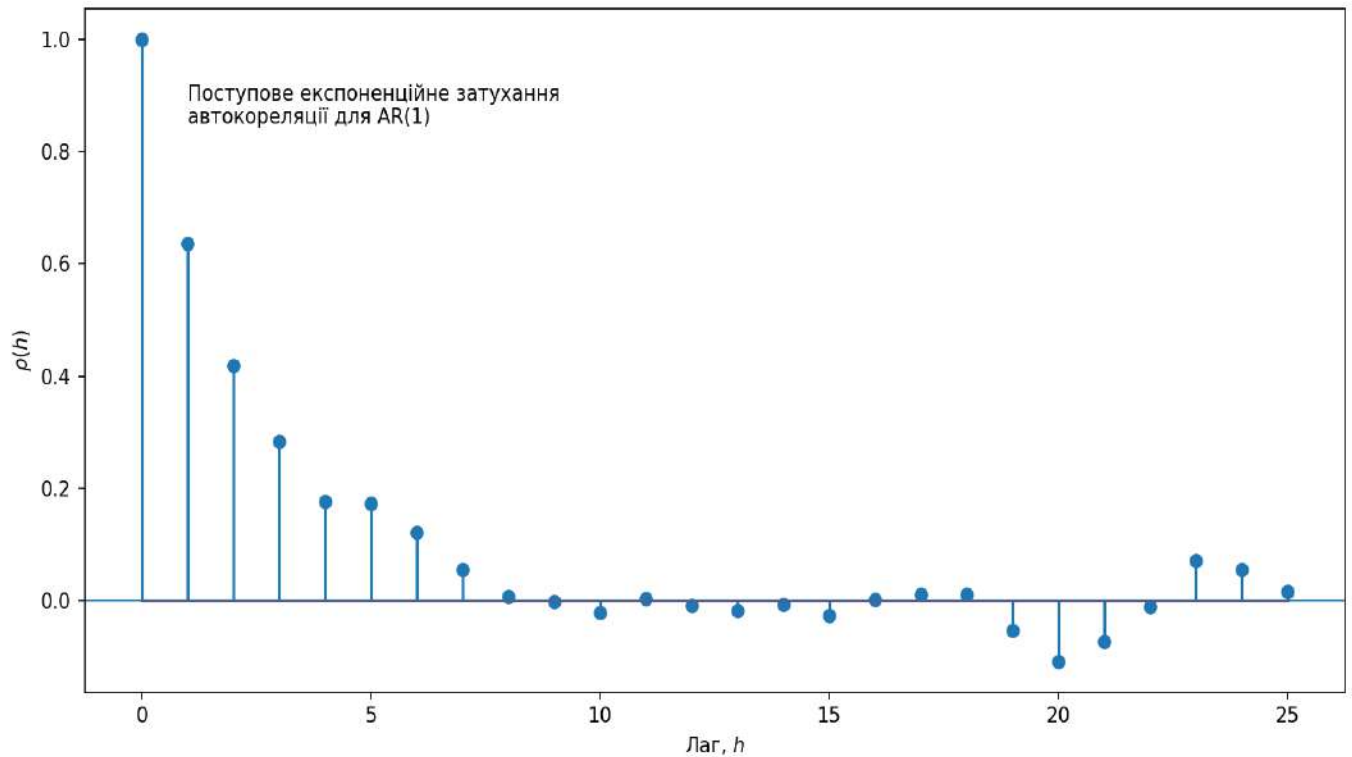


Рисунок 6.6 - ACF стаціонарного часового ряду (модель AR(1))

Стаціонарність AR(p)

AR-модель породжує стаціонарний часовий ряд лише за умови стабільності авторегресійної частини. Для цього вводиться характеристичний поліном:

$$a(z) = 1 - a_1 z - a_2 z^2 - \dots - a_p z^p.$$

Умова стаціонарності:

- усі корені рівняння $a(z) = 0$
- мають модуль більший за одиницю: $|z_i| > 1$.

Для моделі AR(1) ця умова спрощується до: $|a| < 1$.

Якщо умова стаціонарності не виконується, процес не повертається до сталого режиму, а його статистичні характеристики змінюються з часом.

Середнє стаціонарного AR(p)-процесу

Для стаціонарного процесу математичне сподівання є сталою величиною:

$$E(x_t) = m = \text{const.}$$

Усереднюючи рівняння $AR(p)$ та враховуючи $E(u_t) = 0$, отримуємо:

$$m = c + \sum_{j=1}^p a_j m.$$

Звідси:

$$\left(1 - \sum_{j=1}^p a_j\right) m = c,$$
$$m = \frac{c}{1 - \sum_{j=1}^p a_j} = \frac{c}{a(1)}.$$

Таким чином, параметр c не є безпосереднім середнім рівнем процесу — середнє значення визначається всією авторегресійною структурою.

Кореляційна структура $AR(p)$

Для стаціонарної $AR(p)$ -моделі характерні такі властивості:

- автокореляційна функція $ACF(h)$ затухає поступово (експоненційно або хвилеподібно);
- часткова автокореляційна функція $PACF(h)$ має чіткий «обрив» після лага p .

Ці закономірності використовуються для ідентифікації порядку AR -моделі можна спостерігати графічно (див. *Рисунок 6.3 — автоковаріаційна функція* та *Рисунок 6.4 — ACF*).

Практичні задачі з промисловості та економіки ($AR(p)$, $AR(1)$)

Нижче наведено три прикладні задачі, що демонструють застосування авторегресійних моделей $AR(p)$ для аналізу, діагностики та прогнозування часових рядів у промислових та економічних системах.

Задача 1. Аналіз інерційності попиту на промислову продукцію ($AR(1)$)

Умова.

Є часовий ряд щомісячного попиту на продукцію промислового підприємства x_t (шт.). Необхідно перевірити, чи має попит інерційний характер і чи може бути адекватно описаний моделлю $AR(1)$:

$$x_t = c + ax_{t-1} + u_t.$$

Економічна інтерпретація. - коефіцієнт a відображає силу «пам'яті» попиту; - якщо $|a| < 1$, попит є стаціонарним і повертається до середнього рівня; - при великому a попит сильно залежить від попереднього місяця.

Завдання.

1. Побудувати графік часового ряду.
2. Оцінити модель AR(1).
3. Проаналізувати значення коефіцієнта a .
4. Побудувати ACF та перевірити експоненційне згасання.

Приклад реалізації (Python).

```
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.tsa.ar_model import AutoReg
from statsmodels.graphics.tsaplots import plot_acf
x = pd.read_csv("series.csv")["x"]
model = AutoReg(x, lags=1, trend="c").fit()
print(model.summary())
plot_acf(x, lags=20)
plt.show()
```

Очікуваний висновок.

Якщо $0 < \hat{a} < 1$, попит має інерційний, але стабільний характер, що узгоджується з моделлю AR(1).

Задача 2. Моделювання енергоспоживання обладнання (AR(p))

Умова.

Є часовий ряд добового електроспоживання промислової установки x_t (кВт·год). Припускається, що поточне споживання залежить від кількох попередніх днів:

$$x_t = c + \sum_{j=1}^p a_j x_{t-j} + u_t.$$

Інженерна інтерпретація. - коефіцієнти a_j відображають теплову та технологічну інерцію; - більший порядок p означає довшу «пам'ять» системи.

Завдання.

1. Побудувати ACF та PACF.
2. За PACF визначити порядок p .
3. Оцінити AR(p)-модель. 4.

Перевірити стаціонарність (умову $|z_i| > 1$).

Приклад реалізації (Python).

```
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from statsmodels.tsa.ar_model import AutoReg
plot_acf(x, lags=30)
plot_pacf(x, lags=30)
plt.show()
model = AutoReg(x, lags=3, trend="c").fit()
print(model.summary())
```

Очікуваний висновок.

PACF має обрив після лага p , що підтверджує коректність вибору AR(p).

Задача 3. Прогнозування економічного показника (виручка підприємства)

Умова.

Є часовий ряд місячної виручки підприємства x_t . Необхідно побудувати короткостроковий прогноз на основі AR-моделі.

Економічна інтерпретація. - AR-модель описує автозалежність фінансових потоків; - прогноз базується на внутрішній динаміці, без зовнішніх факторів.

Завдання.

1. Оцінити стаціонарну AR(p)-модель.
2. Побудувати прогноз на $h = 6$ періодів.
3. Інтерпретувати прогнозні значення.

Приклад реалізації (Python).

```
model = AutoReg(x, lags=2, trend="c").fit()
forecast = model.predict(
    start=len(x),
    end=len(x)+5,
    dynamic=False
)
print(forecast)
```

Очікуваний висновок.

Прогнозні значення коливаються навколо стаціонарного середнього $m =$

$$\frac{c}{1 - \sum_{j=1}^p a_j},$$

що узгоджується з теоретичними властивостями AR(p)-процесу.

Тема 3. Моделі ковзного середнього MA(Q)

Визначення та інтерпретація

Модель ковзного середнього порядку q описує часовий ряд x_t як лінійну комбінацію поточної інновації та її лагів:

$$x_t = c + u_t + \sum_{s=1}^q b_s u_{t-s}, \quad u_t \sim WN(0, \sigma_u^2).$$

де: - x_t — значення часового ряду в момент часу t ; - c — стала (зсув); - u_t — випадкова інновація (білий шум); - b_s — коефіцієнти моделі ковзного середнього; - q — порядок MA-моделі.

Модель MA(q) не містить лагів змінної x_t у структурному рівнянні, тобто поточне значення процесу не залежить безпосередньо від власних минулих значень. Залежність у часі формується опосередковано — через поточну та минулі інновації u_t .

Інтерпретаційно MA(q)-процес можна розглядати як систему з обмеженою пам'яттю збурень: - кожен випадковий шок u_t впливає на

процес протягом скінченної кількості періодів; - тривалість впливу дорівнює порядку моделі q ; - після лага q вплив відповідної інновації повністю зникає.

Таким чином, $MA(q)$ є моделлю розподіленого в часі ефекту випадкових збурень, що принципово відрізняє її від AR-моделей, у яких вплив шоку затухає поступово, але формально нескінченно довго.

Графічно ця властивість проявляється у характерній кореляційній структурі МА-процесу: - автокореляційна функція має *чіткий обрив* після лага q ; - для лагів $h > q$ автокореляція дорівнює нулю.

Ці особливості наочно ілюструються на Рисунку 6.7 — Автокореляційна функція $MA(q)$ -процесу та Рисунку 6.8 — Реалізація $MA(q)$ -процесу у часовій області.

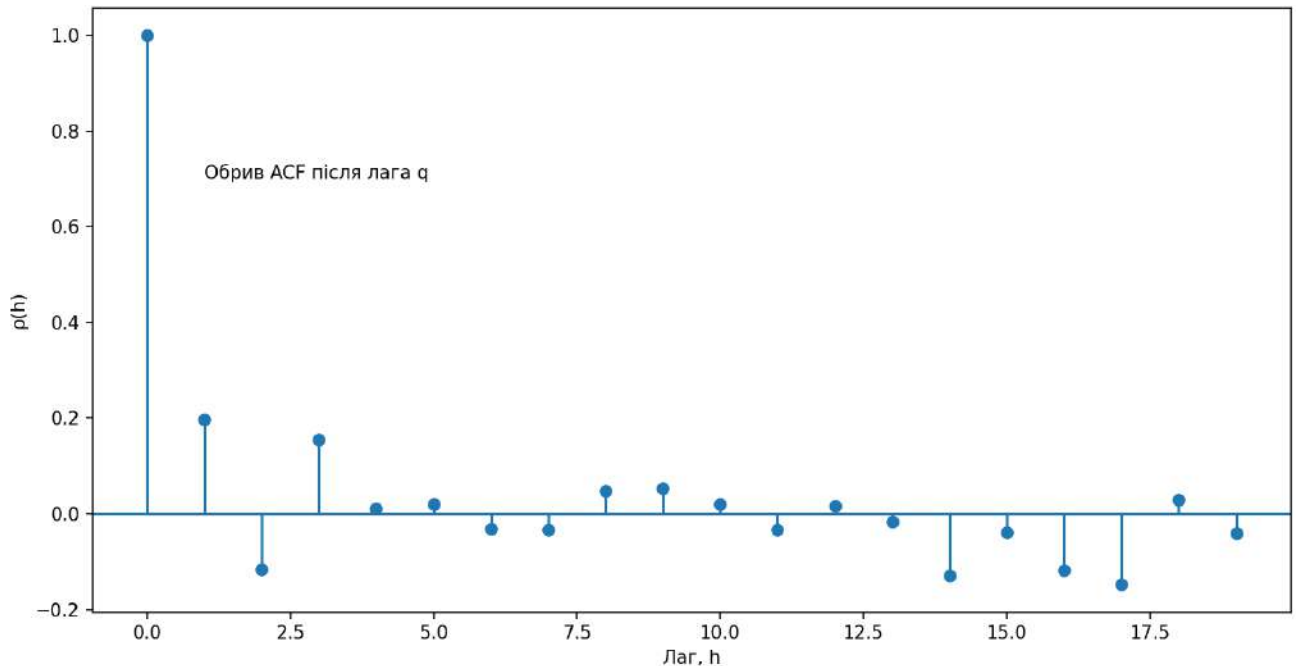


Рисунок 6.7 — Автокореляційна функція $MA(q)$ -процесу

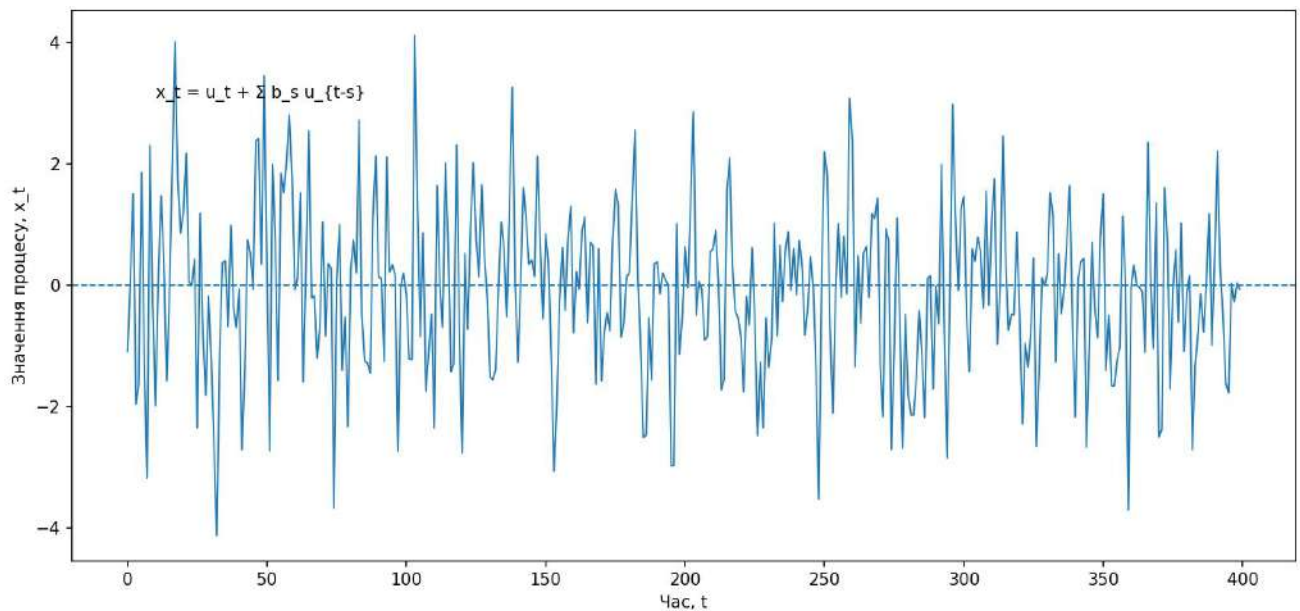


Рисунок 6.8 — Реалізація МА(q)-процесу у часовій області

На практиці МА-моделі широко застосовуються для опису процесів, у яких випадкові збурення мають короткотривалий, але розподілений у часі вплив, зокрема:

- у фінансовій економетриці (вплив новин);
- у промислових системах керування (імпульсні збурення);
- у цифровій обробці сигналів.

Стаціонарність МА(q)

МА(q) є стаціонарною у широкому сенсі за стандартних умов на u_t (білий шум) і кінцевості коефіцієнтів b_s .

Інтуїтивно: x_t є сумою скінченної кількості стаціонарних компонент, а тому також стаціонарний.

Середнє МА(q)

Оскільки $E(u_t) = 0$, маємо: $E(x_t) = c$.

На відміну від AR(p), тут середнє безпосередньо дорівнює константі.

Кореляційна структура МА(q)

Для МА(q) характерно: - ACF має «обрив» після лага q (для $h > q$ кореляції близькі до нуля); - PACF зазвичай **затухає поступово**.

Це робить АСF ключовим інструментом ідентифікації порядку МА.

Практичні задачі для МА(q)

Нижче наведено три практичні задачі, що ілюструють застосування моделей ковзного середнього МА(q) у фінансовій економетриці, промислових системах та цифровій обробці сигналів. Кожна задача супроводжується візуалізацією часових рядів та автокореляційної структури.

Задача 1. Фінансова економетрика: «ефект новин» на денні зміни курсу (МА(q))

Умова.

Нехай x_t — денні зміни курсу (або доходність фінансового активу), на які впливають випадкові новини. Передбачається, що вплив новин є короткотривалим, але розподіленим у часі, що відповідає моделі МА(q):

$$x_t = c + u_t + \sum_{s=1}^q b_s u_{t-s}, \quad u_t \sim WN(0, \sigma_u^2).$$

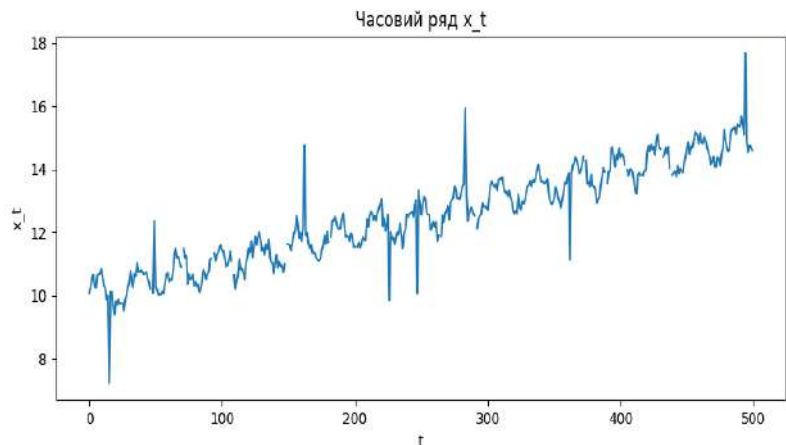
Завдання. 1. Побудувати графік часового ряду x_t (аналог Рис. 6.8).

2. Побудувати автокореляційну функцію та визначити порядок q за характерним обривом (аналог Рис. 6.7).

3. Оцінити МА(q)-модель та проаналізувати залишки.

ПРИКЛАД РЕАЛІЗАЦІЇ

```
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.graphics.tsaplots
import plot_acf
from statsmodels.tsa.arima.model
import ARIMA
df = pd.read_csv("series.csv")
x = df["x"]
plt.figure(figsize=(10,4))
plt.plot(x)
plt.title("Часовий ряд x_t")
plt.xlabel("t")
plt.ylabel("x_t")
plt.show()
plot_acf(x, lags=30)
plt.show()
model = ARIMA(x,
```




```
order=(0,0,2)).fit()
print(model.summary())
```

Задача 2. Промисловість: імпульсні збурення у технологічному процесі

Умова.

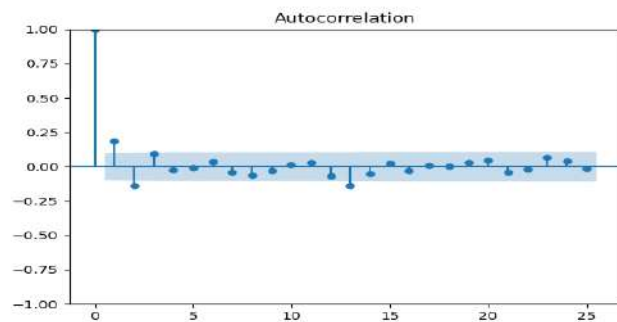
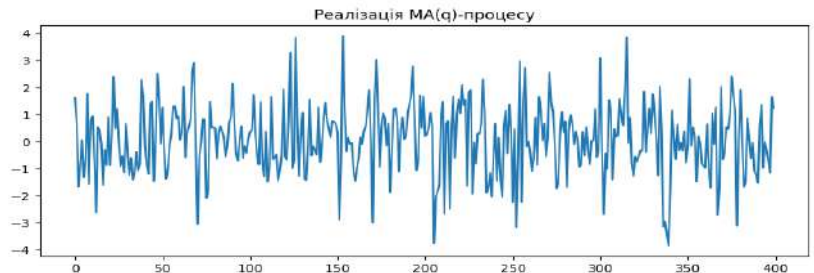
Нехай x_t — відхилення технологічного параметра (температури, тиску) від номіналу. Імпульсні збурення впливають на систему протягом кількох тактів, що моделюється $MA(q)$ -процесом.

Завдання.

1. Згенерувати $MA(q)$ -процес із заданими коефіцієнтами.
2. Побудувати часову реалізацію процесу (аналог Рис. 6.8).
3. Побудувати ACF та підтвердити обрив після лага q (аналог Рис. 6.7).
4. Оцінити параметри $MA(q)$ та порівняти їх з теоретичними.

ПРИКЛАД РЕАЛІЗАЦІЇ

```
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.graphics.tsaplots import plot_acf
from statsmodels.tsa.arima.model import ARIMA
np.random.seed(1)
n = 400
u = np.random.normal(0, 1, n)
x = np.zeros(n)
b = [0.8, -0.4, 0.25]
for t in range(n):
    x[t] = u[t]
    for j in range(1, 4):
        if t-j >= 0:
            x[t] += b[j-1]*u[t-j]
plt.figure(figsize=(10,4))
plt.plot(x)
plt.title("Реалізація MA(q)-процесу")
plt.show()
plot_acf(x, lags=25)
plt.show()
model = ARIMA(x,
order=(0,0,3)).fit()
print(model.summary())
```



Задача 3. Цифрова обробка сигналів: корельований шум вимірювань

Умова.

Нехай x_t — шум вимірювань сенсора, який має короткотривалу кореляцію через інерційність вимірювального тракту. Модель $MA(q)$ дозволяє описати таку структуру шуму.

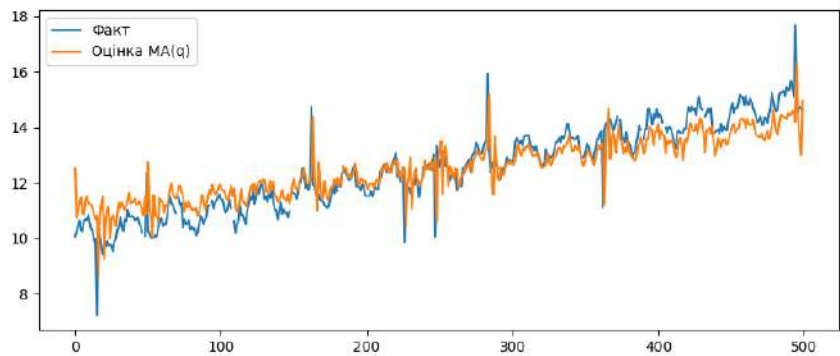
Завдання. 1. Побудувати ACF шуму та визначити порядок q .

2. Оцінити $MA(q)$ -модель.

3. Побудувати однокрокову оцінку (прогноз) та порівняти з фактичними значеннями.

ПРИКЛАД РЕАЛІЗАЦІЇ

```
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.graphics.tsaplots
import plot_acf
from statsmodels.tsa.arima.model
import ARIMA
df = pd.read_csv("series.csv")
x = df["x"]
plot_acf(x, lags=30)
plt.show()
model = ARIMA(x,
order=(0,0,3)).fit()
pred = model.predict(start=0,
end=len(x)-1)
plt.figure(figsize=(10,4))
plt.plot(x.values, label="Факт")
plt.plot(pred.values,
label="Оцінка MA(q)")
plt.legend()
plt.show()
```



Розглянуті задачі демонструють, що моделі $MA(q)$:

- адекватно описують процеси з короткотривалим, але розподіленим у часі впливом випадкових збурень;
- мають характерну автокореляційну структуру з обривом після лага q ;
- широко застосовуються в економіці.

Тема 4. Аналіз часових рядів: лінійна регресія, TS-ряди, тренд і сезонність

Лінійна регресія для стаціонарних часових рядів

Лінійна регресія для стаціонарних часових рядів використовується для опису взаємозв'язків між змінними, які **не мають тренду та сезонності**, а їхні статистичні характеристики (середнє, дисперсія, коваріація) не змінюються з часом.

Загальний вигляд лінійної регресійної моделі:

$$y_t = \beta_0 + \beta_1 x_{1,t} + \beta_2 x_{2,t} + \dots + \beta_k x_{k,t} + u_t,$$

де: - y_t — залежна змінна; - $x_{i,t}$ — пояснювальні змінні; - u_t — стаціонарний випадковий шум.

Ключова умова застосовності — стаціонарність усіх змінних або їхніх перетворень.

Модель FDL (Finite Distributed Lag)

FDL-модель описує вплив пояснювальної змінної не лише в поточний момент, а й протягом обмеженої кількості попередніх періодів (Рис. 4.9):

$$y_t = \alpha + \sum_{j=0}^q \beta_j x_{t-j} + u_t.$$

Інтерпретація:

- кожне значення β_j показує **відкладений ефект** x через j періодів;
- загальний ефект зміни x : $\sum_{j=0}^q \beta_j$;
- модель застосовується, коли реакція системи є **швидкою, але не миттєвою**.

Типові приклади застосування:

- вплив реклами на продажі;
- вплив цін на попит;
- реакція виробництва на зміну енерготарифів.

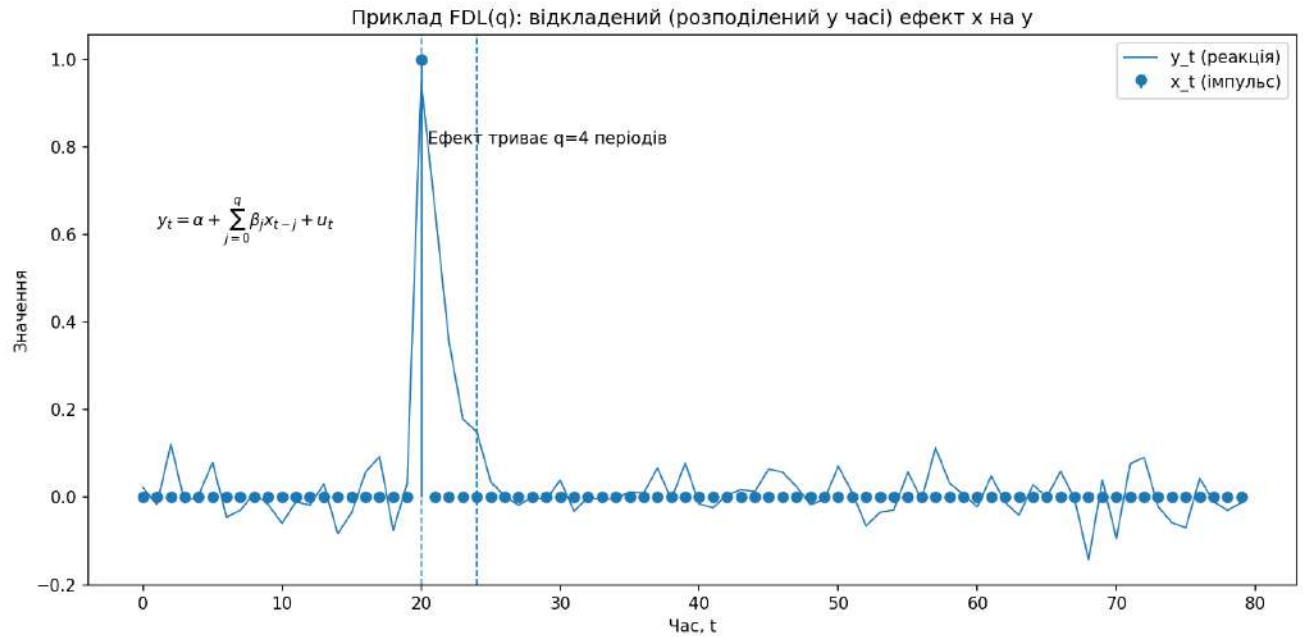


Рисунок 6.9 - Відкладений (розподілений у часі) ефект пояснювальної змінної в FDL-моделі

Модель ADL (Autoregressive Distributed Lag) (Рис. 6.10)

ADL-модель поєднує:

- лаги пояснювальної змінної;
- лаги самої залежної змінної.

$$y_t = \alpha + \sum_{i=1}^p \phi_i y_{t-i} + \sum_{j=0}^q \beta_j x_{t-j} + u_t.$$

Особливості ADL:

- враховує **інерційність** процесу;
- дозволяє моделювати як коротко-, так і довгострокові ефекти;
- широко застосовується в макроекономіці та фінансовому аналізі.

Довгостроковий ефект: $\frac{\sum_{j=0}^q \beta_j}{1 - \sum_{i=1}^p \phi_i}$.

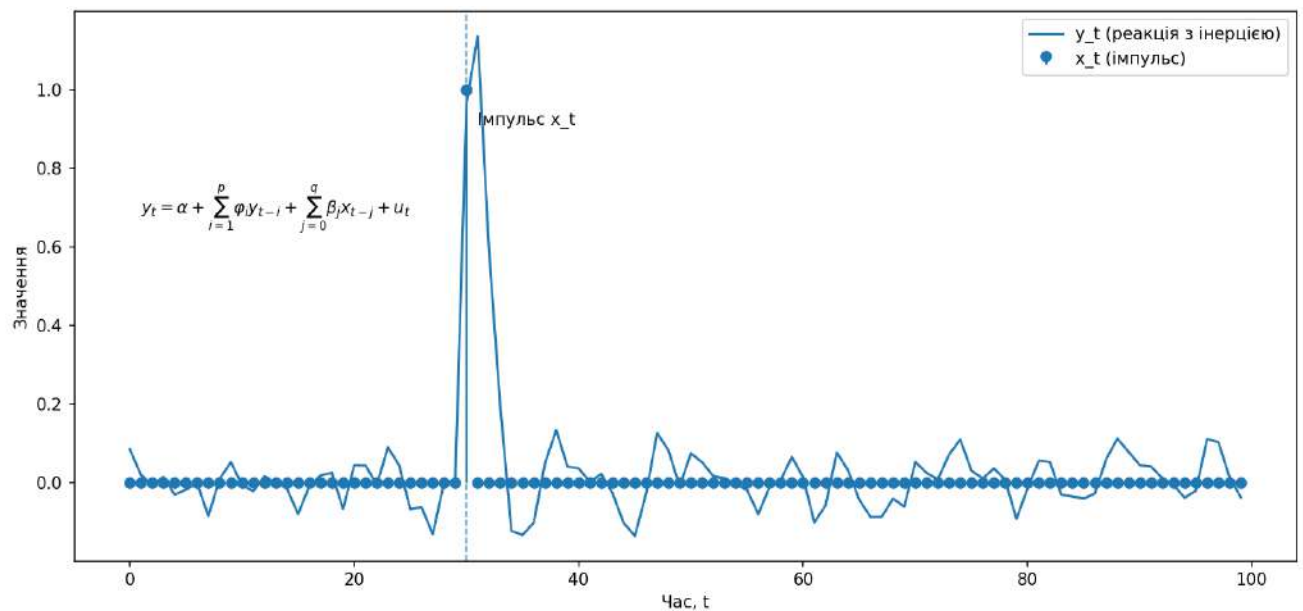


Рисунок 6.10 - ADL(p,q). Поєднання інерційності та розподіленого впливу фактору x

TS-ряди (Time Series)

Часовий ряд (TS-ряд) — це послідовність спостережень, упорядкованих у часі (Рис. 6.11):

$$\{x_t\}_{t=1}^T.$$

TS-ряди відрізняються від перехресних вибірок тим, що:

- спостереження **корельовані в часі**;
- порядок даних має принципове значення;
- можливі тренд, сезонність і структурні зміни.

Основні компоненти TS-ряду:

- тренд;
- сезонність;
- циклічність;
- випадкова складова.

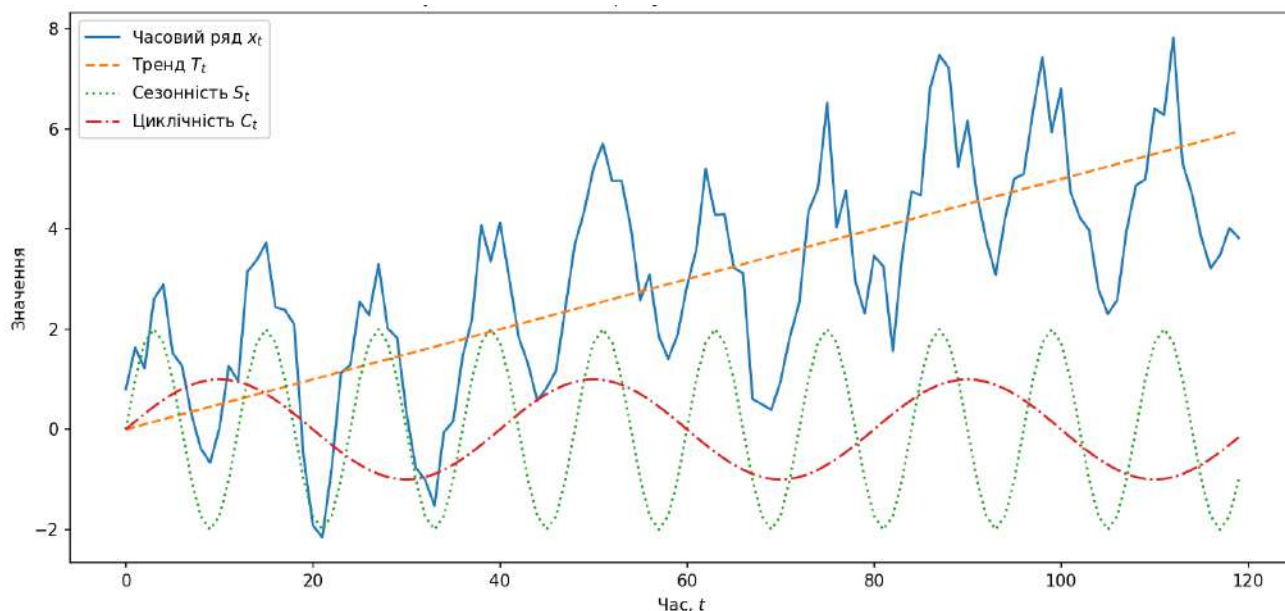


Рисунок 6.11 – Часовий ряд та його основні компоненти: тренд, сезонність, циклічність і випадкова складова

Оцінювання та статистичні висновки

Оцінювання параметрів методом найменших квадратів (OLS)

Нехай часовий ряд має вигляд:

$$x_t = f(t; \theta) + u_t,$$

де $f(t; \theta)$ — детермінована компонента (тренд, сезонність), а u_t — випадкова складова.

Метод найменших квадратів знаходить оцінки параметрів шляхом мінімізації:

$$\min_{\theta} \sum_{t=1}^T (x_t - f(t; \theta))^2.$$

Оцінювання тренду методом OLS

Лінійний тренд

Найпоширенішою моделлю є лінійний тренд:

$$x_t = \beta_0 + \beta_1 t + u_t.$$

Параметр β_1 інтерпретується як середня швидкість зміни показника.

Поліноміальний тренд

Для складнішої динаміки:

$$x_t = \beta_0 + \beta_1 t + \beta_2 t^2 + u_t.$$

Оцінювання сезонності

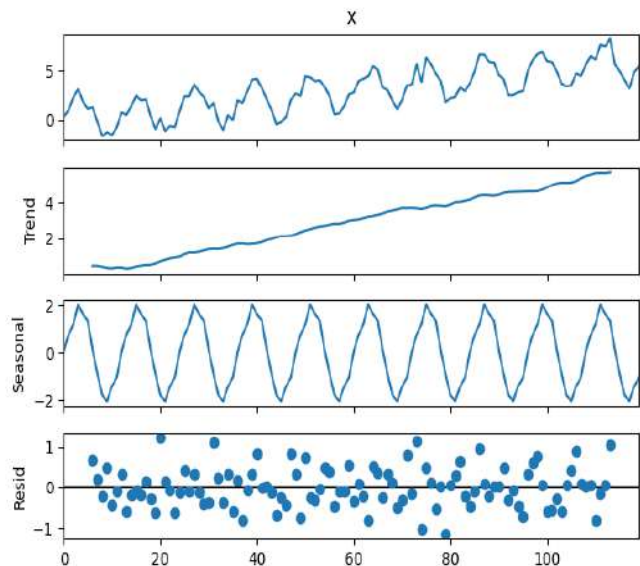
Сезонність оцінюється за допомогою фіктивних змінних:

$$x_t = \beta_0 + \beta_1 t + \sum_{j=1}^{s-1} \gamma_j D_{j,t} + u_t.$$

Такий підхід дозволяє одночасно оцінити тренд і сезонні ефекти.

ПРИКЛАД

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import statsmodels.api as sm
np.random.seed(42)
T = 120
t = np.arange(T)
trend = 0.05 * t
season = 2 * np.sin(2 * np.pi * t / 12)
noise = np.random.normal(0, 0.6, T)
x = trend + season + noise
df = pd.DataFrame({"x": x, "t": t})
# Оцінювання тренду методом OLS
X = sm.add_constant(df["t"])
model_trend = sm.OLS(df["x"], X).fit()
print(model_trend.summary())
df["trend_hat"] = model_trend.predict(X)
# Детрендування
df["x_detrended"] = df["x"] - df["trend_hat"]
# Сезонна декомпозиція
from statsmodels.tsa.seasonal import seasonal_decompose
decomp = seasonal_decompose(df["x"],
                             model="additive", period=12)
decomp.plot()
plt.show()
```



Статистичні висновки

Перевірка значущості

Для кожного параметра OLS використовується t-тест:

$$t_i = \frac{\hat{\beta}_i}{se(\hat{\beta}_i)}.$$

Аналіз залишків

Залишки:

$$\hat{u}_t = x_t - \hat{T}_t - \hat{S}_t$$

перевіряють на:

- автокореляцію (ACF, Ljung–Box);
- стаціонарність (ADF, KPSS).

Підсумкові висновки:

- OLS є базовим методом оцінювання тренду та сезонності.
- Детрендування є необхідним перед AR/MA-моделюванням.
- Після коректної декомпозиції залишки мають бути стаціонарними.

Практичні задачі (*Аналіз часових рядів: регресія, FDL, ADL, тренд і сезонність*)

Задача 1. Лінійна регресія для стаціонарного економічного часового ряду

Умова

Є часовий ряд **місячної виручки підприємства** y_t та два стаціонарні фактори: - $x_{1,t}$ — індекс ділової активності; - $x_{2,t}$ — реальна ставка кредитування.

Модель:

$$y_t = \beta_0 + \beta_1 x_{1,t} + \beta_2 x_{2,t} + u_t.$$

Завдання

- Побудувати графіки y_t , $x_{1,t}$, $x_{2,t}$.
- Перевірити стаціонарність рядів (ADF).
- Оцінити модель лінійної регресії.
- Інтерпретувати коефіцієнти.
- Проаналізувати залишки.

Очікуваний висновок: регресія дозволяє визначити, які фактори статистично значуще впливають на виручку.

Задача 2. Вплив реклами на продажі: FDL-модель

$$y_t = \alpha + \sum_{j=0}^q \beta_j x_{t-j} + u_t.$$

- y_t — тижневі обсяги продажів;
- x_t — витрати на рекламу.

Завдання

- Побудувати графіки x_t та y_t .
- Оцінити FDL-модель для $q = 3$.
- Побудувати графік коефіцієнтів β_j .
- Обчислити загальний ефект реклами: $\sum_{j=0}^q \beta_j$.

Очікуваний висновок: реклама має відкладений у часі вплив, який зникає після кількох періодів.

Задача 3. Інерційність попиту: ADL-модель

Модель:

$$y_t = \alpha + \sum_{i=1}^p \varphi_i y_{t-i} + \sum_{j=0}^q \beta_j x_{t-j} + u_t.$$

- y_t — квартальний попит на електроенергію;
- x_t — тариф на електроенергію.

Завдання

- Оцінити ADL(1,2)-модель.
- Перевірити значущість коефіцієнтів.
- Обчислити довгостроковий ефект тарифу: $\frac{\sum_{j=0}^q \beta_j}{1 - \sum_{i=1}^p \varphi_i}$.
- Побудувати фактичні та модельні значення.

Очікуваний висновок: попит характеризується інерційністю та відкладеною реакцією на тариф.

Задача 4. Аналіз TS-ряду з трендом і сезонністю

Умова

Є часовий ряд **місячних податкових надходжень**.

Завдання

- Побудувати часовий ряд x_t .
- Виділити тренд, сезонність і циклічність.
- Застосувати адитивну декомпозицію:

$$x_t = T_t + S_t + u_t.$$

- Інтерпретувати економічний зміст компонент.

Очікуваний висновок: податкові надходження мають сезонні та довгострокові закономірності.

Задача 5. Оцінювання тренду та сезонності методом OLS

Умова для показника x_t :

$$x_t = \beta_0 + \beta_1 t + \sum_{j=1}^{s-1} \gamma_j D_{j,t} + u_t$$

Необхідно оцінити тренд і сезонні ефекти.

Завдання

- Оцінити тренд методом OLS.
- Побудувати детрендований ряд.
- Оцінити сезонність через dummy-змінні.
- Перевірити залишки (ACF, Ljung–Box, ADF/KPSS).

Очікуваний висновок: після коректного усунення тренду та сезонності залишки є стаціонарними.

Запропоновані практичні задачі демонструють застосування:

- лінійної регресії для стаціонарних TS-рядів;
- FDL для короткострокових відкладених ефектів;
- ADL для інерційних економічних процесів;
- OLS як базового методу оцінювання тренду та сезонності.

Питання для самооцінювання

1. Що таке стаціонарний часовий ряд і чому це важливо для лінійної регресії?
2. Напишіть загальний вигляд лінійної регресійної моделі для TS-ряду.
3. Чому змінні повинні бути стаціонарними перед застосуванням лінійної регресії?
4. Як перевірити стаціонарність ряду перед включенням до регресійної моделі?
5. Як інтерпретуються коефіцієнти β_0 та β_k в лінійній регресії TS-ряду?
6. Як формулюється модель FDL(q) для часових рядів?
7. Що означає кожен лаг β_j в моделі FDL?
8. Як інтерпретувати загальний ефект зміни x в моделі FDL?
9. У яких економічних ситуаціях може бути корисною модель FDL?
10. Як вибрати порядок q для моделі FDL?
11. Чим ADL (p,q) відрізняється від моделі FDL(q)?
12. Напишіть математичний запис ADL(p,q).
13. Як у моделі ADL враховується інерційність процесу?
14. Як обчислити довгостроковий ефект пояснювальної змінної в ADL?
15. Чому ADL-моделі широко застосовуються в макроекономіці?
16. Що таке часовий ряд? Які його ключові характеристики?
17. Назвіть і коротко опишіть основні компоненти TS-ряду.
18. Як тренд впливає на аналіз часових рядів?
19. Що таке сезонність, і які існують підходи до її моделювання?
20. У якій ситуації потрібне тестування на структурні злами?
21. Що таке метод найменших квадратів (OLS) у контексті TS-рядів?
22. Як побудувати модель лінійного тренду за допомогою OLS?
23. Що таке детрендування, і навіщо воно проводиться?

24. Як за допомогою dummy-змінних моделювати сезонність?
25. Які основні тести варто застосувати до залишків після оцінювання моделі?
26. Чому важливо перевіряти наявність автокореляції в залишках?
27. За допомогою яких тестів перевіряють стаціонарність залишків після декомпозиції?
28. Як інтерпретувати t-тест для оцінки значущості параметрів OLS?
29. Чому оцінювання тренду та сезонності є обов'язковим перед AR/MA-аналізом?
30. Які можливі наслідки неправильного усунення тренду чи сезонності перед моделюванням ARIMA?

Бібліографічний список

1. Яровий А. Т. Аналіз часових рядів : навч. посібн. / А. Т. Яровий. — Одеса: ОНУ, 2018. — 112 с. — Електронний ресурс. — Доступ: https://onu.edu.ua/pub/bank/userfiles/files/fmfit/nachalni_materiali/navch_mat_11_1_mag/analiz_chasovykh_ryadiv.pdf
2. Бідюк П. І., Романенко В. Д., Тимошук О. Л. Аналіз часових рядів [Електронний ресурс] / НТУУ «КПІ». — Київ : КПІ, 2010. — Електронний ресурс. — Доступ: <https://ela.kpi.ua/handle/123456789/862>
3. Юрченко М. Є. Прогнозування та аналіз часових рядів : метод. вказівки / М. Є. Юрченко. — Чернігів : ЧНТУ, 2018. — 88 с. — Доступ: <https://ir.stu.cn.ua/bitstreams/64ad3c4f-b8f2-4cd9-9ae2-ab2612f9c09f/download>
4. Hyndman R. J., Athanasopoulos G. Forecasting: Principles and Practice / 3rd ed. — OTexts, 2021. — Електронний ресурс. — Доступ: <https://otexts.com/fpp3/>
5. Franses P. H., van Dijk D., Opschoor A. Time Series Models for Business and Economic Forecasting / Cambridge Univ. Press, 2014. —

Електронний ресурс. — Доступ:

<https://ideas.repec.org/b/cup/cbooks/9780521817707.html>

6. Introduction to Time Series Regression and Forecasting [Електронний ресурс]. — Доступ: <https://homepage.ntu.edu.tw/~luohm/metrics2022s/time.pdf>

7. Wooldridge J. Time Series — Distributed Lag and ADL Models [Електронний ресурс]. — Доступ: https://www.fsb.miamioh.edu/lij14/311_slide_ch1011.pdf

8. Андрусенко Ю. О. Аналіз основних моделей прогнозування часових рядів / Журнал НУПС, 2020. — Доступ: <https://journal-hnups.com.ua/index.php/zhups/article/view/358>

9. Introduction To Time Series Regression & Forecasting [Електронний ресурс]. — Доступ: [https://www.scribd.com/document/812119122/Introduction-to-Time-Series-Regression-Forecasting :contentReference\[oaicite:18\]{index=18}](https://www.scribd.com/document/812119122/Introduction-to-Time-Series-Regression-Forecasting :contentReference[oaicite:18]{index=18})

10. Козачок Л. М., Козачок А. Є. Аналіз та прогнозування часових рядів [Електронний ресурс]. — Доступ: https://rcf.khadi.kharkov.ua/fileadmin/F-HIGHWAY/Інформа.../veit_2017_12_28.pdf

11. Квашук Д. series.csv [електронний ресурс]. — Доступ з: <https://drive.google.com/file/d/1rZgCYsFw8GJXJtKnyvQrDtLzi-uBEEBx/view?usp=sharing>

12. Квашук Д. series_1.csv (2025) [електронний ресурс]. — Доступ з: https://drive.google.com/file/d/14edvieat5VcIaR1oLyjThx5pqu6NtB4_/view?usp=drive_link

РОЗДІЛ 7. БІЗНЕС-АНАЛІТИКА У ФІНАНСАХ, МАРКЕТИНГУ ТА HR

(Литвин О.Є., к.е.н., доцент, доцент кафедри міжнародних фінансів
Київського національного університету імені Тараса Шевченка)

Тема 1. Аналітика у фінансах: моделі прогнозування, оцінка ризиків та управління капіталом

Мета: сформувати у студентів системне розуміння інструментів фінансової аналітики, методів прогнозування, моделей оцінки ризиків та принципів управління капіталом, що забезпечує ефективне прийняття стратегічних рішень у бізнесі.

Завдання:

- Опанувати побудову фінансових прогнозів у стратегічному управлінні.
- Навчитися оцінювати фінансові ризики та застосовувати інструменти їх управління.
- Засвоїти управління капіталом та інвестиційними портфелями, включно з оптимізацією структури та аналітичними панелями.
- Розвинути навички прийняття рішень на основі даних для забезпечення фінансової стійкості та конкурентоспроможності.

7.1.1. Роль фінансової аналітики у стратегічному управлінні.

Фінансова аналітика включає моделі прогнозування, оцінку ризиків та управління капіталом, що дозволяє прогнозувати майбутні фінансові показники, виявляти та мінімізувати потенційні загрози, а також ефективно розподіляти й використовувати кошти. Моделі прогнозування аналізують фінансові результати, а оцінка ризиків виявляє загрози. Наприклад, інвестиційні, валютні, кредитні), а управління капіталом передбачає контроль та фінансування ризиків, диверсифікацію та хеджування для досягнення фінансової стійкості.

Фінансова аналітика відіграє ключову роль у стратегічному управлінні, надаючи керівництву інструменти для аналізу ринкових тенденцій, оцінки ефективності бізнесу та прогнозування майбутніх ризиків. Вона дозволяє розробляти стратегії розвитку, що ґрунтуються на об'єктивних даних, забезпечуючи прийняття обґрунтованих рішень щодо розподілу ресурсів, інвестицій та покращення прибутковості.

Ключові аспекти ролі фінансової аналітики:

- Аналіз ринкових тенденцій дозволяє компанії адаптуватися до змін ринку та конкурентного середовища.
- Оцінка ефективності надає можливість вимірювати ефективність маркетингових кампаній та інших стратегічних ініціатив.
- Прогнозування майбутнього допомагає передбачати майбутні тренди, можливості та ризики, що дозволяє компанії бути проактивною.
- Прийняття рішень забезпечує основу для обґрунтованих рішень щодо інвестицій, ціноутворення, розширення або реструктуризації бізнесу.
- Визначення потреб клієнтів допомагає зрозуміти потреби та переваги клієнтів, що є критично важливим для розробки продуктової та маркетингової стратегії.

Значення даних для прийняття рішень передбачає:

1. *Інформаційну базу для стратегій:* Фінансові дані допомагають оцінити поточний стан бізнесу, визначити прибуткові та збиткові напрями діяльності, а також оптимізувати ресурсне забезпечення.
2. *Прогнозування та планування:* Використання історичних фінансових показників дозволяє моделювати майбутні сценарії розвитку та оцінювати потенційні ризики.
3. *Підвищення якості рішень:* Керівники отримують можливість приймати обґрунтовані стратегічні рішення на основі аналітичних даних, знижуючи ймовірність помилок та необґрунтованих витрат.

Інтеграція фінансових аналітичних систем у корпоративне управління передбачає:

1. *Автоматизацію збору та обробки даних:* Впровадження сучасних фінансових інформаційних систем дозволяє отримувати актуальні показники в режимі реального часу.
2. *Єдину аналітичну платформу:* Інтегровані системи забезпечують централізовану обробку даних, полегшуючи координацію між різними підрозділами та керівництвом компанії.
3. *Підтримку стратегічного контролю:* Системи аналітики дозволяють відстежувати ключові фінансові індикатори (KPI), оцінювати ефективність стратегій та коригувати їх у режимі реального часу.

Взаємозв'язок фінансової аналітики з іншими бізнес-функціями відбувається через:

1. *Маркетинг та продажі:* Фінансові дані допомагають оцінювати рентабельність продуктів, сегментів ринку та каналів збуту, що дозволяє планувати маркетингові стратегії.
2. *Виробництво та операційну діяльність:* Аналітика забезпечує контроль за витратами, ефективністю використання ресурсів та оптимізацією виробничих процесів.
3. *HR та управління персоналом:* Фінансові показники допомагають планувати бюджет на персонал, оцінювати продуктивність працівників та формувати стимули для підвищення ефективності.
4. *Стратегічне управління ризиками:* Взаємодія фінансової аналітики з усіма бізнес-функціями дозволяє своєчасно ідентифікувати ризики та визначати заходи їх мінімізації.



Виробництво, маркетинг й продажі, HR та IT це – джерела первинних даних про операційну діяльність, витрати, продажі, продуктивність та IT-підтримку. Фінансовий відділ централізує та аналізує дані, формує бюджети, фінансові звіти та KPI. Ризики та контроль оцінюють фінансові ризики, проводить аудит та моніторинг показників. Стратегічне управління отримує інтегровану інформацію та приймає рішення щодо стратегії, інвестицій та розвитку компанії.

7.1.2. Методи та моделі фінансового прогнозування.

Фінансове прогнозування є ключовим інструментом стратегічного управління, який дозволяє компаніям планувати розвиток, оцінювати інвестиційні проєкти та мінімізувати ризики. Основою прогнозування є використання як кількісних, так і якісних методів, які дозволяють передбачати майбутні фінансові показники на основі історичних даних та аналітичних моделей.

Грошові потоки є центральним показником фінансового здоров'я компанії. Основні методи прогнозування включають:

- DCF (Discounted Cash Flow) – метод дисконтованих грошових потоків, що використовується для оцінки вартості компанії або проєкту на основі майбутніх грошових потоків.
- FCFF (Free Cash Flow to Firm) – вільний грошовий потік компанії, що рахує всі операційні потоки до виплати боргу та податків й використовується для оцінки вартості компанії з точки зору всіх інвесторів.
- FCFE (Free Cash Flow to Equity) – вільний грошовий потік акціонерів, який враховує потоки після сплати боргу та процентів, доступні для акціонерів та використовується для оцінки власного капіталу компанії.

Прогнозування фінансових показників часто базується на аналізі часових рядів:

- ARIMA (AutoRegressive Integrated Moving Average), що комбінує авторегресію, інтегрування та ковзне середнє та ефективним для стаціонарних рядів та короткострокових прогнозів.
- SARIMA (Seasonal ARIMA) – це розширена модель ARIMA з урахуванням сезонності, що застосовується для фінансових показників із періодичними коливаннями.
- Prophet є автоматизованою моделлю для прогнозування часових рядів з трендом та сезонністю, яка розроблена компанією Facebook. Вона досить зручна для бізнес-аналітики і дозволяє враховувати святкові дні, події та аномалії.

Оцінка компанії є ключовою для інвесторів та стратегічного управління. Основні підходи передбачають застосування: моделі дисконтованих грошових потоків (DCF), тобто оцінку вартості компанії на основі прогнозованих майбутніх грошових потоків; мультиплікаторів (Multiples / Comparable Company Analysis), а саме: порівняння фінансових показників компанії з аналогами на ринку (наприклад, P/E, EV/EBITDA, P/B) та моделі оцінки

активів (Asset-based models), тобто визначення вартості компанії через суму ринкової або балансової вартості її активів.

Сценарне моделювання та аналіз чутливості є ключовими інструментами управління невизначеністю у фінансовому прогнозуванні. Воно передбачає формування кількох можливих варіантів розвитку бізнесу від оптимістичного, базового до песимістичного, що дозволяє оцінити, як різні зовнішні та внутрішні фактори можуть вплинути на фінансові результати, тоді як аналіз чутливості визначає, як зміна окремих ключових параметрів (наприклад, вартості продажу, рівня витрат чи ставки дисконту) змінює кінцевий прогноз, що дає змогу виявити найбільш критичні фактори ризику і підвищити стійкість стратегічних рішень.

7.1.3. Аналіз фінансових ризиків.

Аналіз фінансових ризиків є невід'ємною складовою стратегічного управління та забезпечує компанії можливість передбачати потенційні загрози, оцінювати їхній вплив на діяльність та розробляти ефективні механізми реагування. Він охоплює системне виявлення внутрішніх і зовнішніх ризиків, таких як: коливання валютних курсів, зміни процентних ставок, нестабільність ринкового попиту, кредитні ризики, ліквідність та операційні загрози, що можуть порушити фінансову рівновагу підприємства.



Процес аналізу включає збір та інтерпретацію даних, моделювання можливих сценаріїв розвитку подій, оцінку ймовірності кожного ризику та розрахунок їхнього потенційного впливу на фінансові показники. Використовуються різні методи, від статистичних моделей, таких як Value at Risk (VaR), Stress Testing і Monte Carlo Simulation, до якісних експертних оцінок. Результати аналізу дозволяють сформувати карту ризиків, встановити їхні пріоритети та визначити стратегії мінімізації, а саме: диверсифікацію активів, хеджування, створення резервів, оптимізацію структури капіталу чи вдосконалення внутрішнього контролю. Завдяки цьому компанія отримує не лише інструменти для зниження можливих збитків, але й підвищує свою стійкість, адаптивність і конкурентоспроможність у мінливому економічному середовищі.

Кредитний ризик охоплює імовірність того, що позичальник не виконає своїх зобов'язань, що може призвести до фінансових втрат для кредитора. Для його оцінювання застосовуються скорингові моделі та підхід PD-LGD-EAD, які забезпечують кількісну оцінку ризику. Скорингові моделі використовують статистичні та машинні алгоритми для прогнозування платоспроможності клієнтів, аналізуючи фінансові показники, кредитну історію, демографічні

характеристики та поведінкові дані з метою присвоєння кожному позичальнику рейтингу ризику.

У рамках підходу PD (Probability of Default) визначається ймовірність дефолту клієнта протягом певного періоду. LGD (Loss Given Default) показує, яку частку заборгованості кредитор втратить у разі дефолту. EAD (Exposure at Default) оцінює обсяг кредитної заборгованості, що буде непогашеною на момент дефолту. Комбінування цих трьох компонентів дозволяє точно розрахувати очікувані кредитні збитки (Expected Credit Loss) та забезпечує основу для управління портфелем кредитів, формування резервів, встановлення процентних ставок і прийняття ефективних рішень у системі ризик-менеджменту.

Ринкові ризики виникають унаслідок коливань цін фінансових інструментів, процентних ставок, валютних курсів і вартості товарів, тому їх аналіз спрямований на вимірювання потенційних збитків, які можуть виникнути через несприятливі зміни ринкової кон'юнктури. Одними з найпоширеніших методів оцінювання є VaR (Value at Risk), CVaR (Conditional Value at Risk) та Stress Testing.

Модель VaR визначає максимальні можливі втрати портфеля за певного рівня довіри та часовим горизонтом, використовуючи історичні дані, варіаційно-коваріаційний підхід або метод Монте-Карло. Вона дозволяє банкам та інвесторам оцінювати ризик у нормальних ринкових умовах. Проте, оскільки VaR не враховує крайніх, рідкісних подій, застосовується CVaR, який показує середні втрати за умов перевищення VaR і таким чином дає більш консервативну оцінку ризику. Stress Testing доповнює ці методи, моделюючи екстремальні, але можливі ринкові ситуації, а саме: різкі девальвації, обвали ринку, зростання процентних ставок чи геополітичні шоки, щоб визначити, як портфель або компанія поводитимуться у кризових умовах. Разом ці інструменти формують комплексний підхід до управління ринковими ризиками, підвищуючи фінансову стійкість та готовність до нестабільності.

Операційний ризик та ризики шахрайства охоплюють загрозу фінансових втрат, що виникають унаслідок недоліків внутрішніх процесів, збоїв у роботі систем, помилок персоналу або зовнішніх подій, а також навмисних дій, спрямованих на отримання неправомірної вигоди. Операційний ризик включає широкий спектр проблем, від технічних збоїв, кібератак і неякісних управлінських рішень до порушень процедур, помилкових транзакцій чи неефективного контролю.

Ризики шахрайства формуються як з боку працівників (наприклад, внутрішнє шахрайство: підробка документів, зловживання повноваженнями, викривлення даних), так і з боку клієнтів або сторонніх осіб (наприклад, зовнішнє шахрайство: кіберзлом, фальшиві заявки, схемні операції). Для їх виявлення та мінімізації застосовуються системи внутрішнього контролю, сегрегація обов'язків, автоматизовані інструменти моніторингу операцій, алгоритми виявлення аномалій і поведінкові моделі, що допомагають вчасно визначати підозрілі дії. Крім того, важливими є культура комплаєнсу, регулярні аудити, навчання персоналу та впровадження систем управління інформаційною безпекою. Комплексний підхід до управління операційним ризиком і запобігання шахрайству забезпечує стабільність бізнес-процесів, знижує ймовірність фінансових втрат і зміцнює довіру до організації

7.1.4. Управління капіталом на основі аналітики

Управління капіталом на основі аналітики передбачає системний підхід до оптимізації структури, вартості та ефективності використання капіталу підприємства шляхом застосування кількісних моделей, прогнозних алгоритмів і даних у реальному часі. Аналітика дозволяє оцінювати потребу в капіталі з урахуванням стратегічних цілей, інвестиційних планів та ризикового профілю компанії, визначати оптимальне співвідношення власного й позикового капіталу, враховуючи вартість обслуговування боргу, доступність фінансування та вплив фінансового левереджу на рентабельність.



Використання інструментів, таких як: WACC, EVA, моделі прогнозування грошових потоків, стрес-тестування та аналіз чутливості, забезпечує обґрунтоване прийняття рішень щодо фінансування, дивідендної політики та інвестування. Аналітичні панелі (dashboards) дозволяють відстежувати ключові показники капіталу та ризику, а саме: ліквідність, коефіцієнти покриття, кредитоспроможність, вартість капіталу і швидко виявляти відхилення, що можуть вплинути на фінансову стійкість.

Завдяки сучасним технологіям, таким як: machine learning, big data, автоматизовані системи планування, компанії можуть моделювати різні сценарії розвитку, прогнозувати потребу в капіталі, оптимізувати розподіл ресурсів і забезпечувати найбільш ефективне використання інвестицій. Таким чином, управління капіталом, підсилене аналітикою, стає не лише інструментом контролю, а й ключовим драйвером стратегічного зростання, підвищення конкурентоспроможності та довгострокової фінансової стійкості бізнесу.

Оптимізація структури капіталу полягає у визначенні такого співвідношення власного та позикового капіталу, яке мінімізує загальну вартість фінансування компанії та водночас забезпечує її фінансову стійкість і стратегічну гнучкість. Аналітичні інструменти допомагають моделювати вплив різних варіантів структури капіталу на ключові показники, таких як: вартість

капіталу (WACC), рентабельність власного капіталу (ROE), рівень фінансового левереджу, ліквідність та кредитний рейтинг. Дані про ринкові умови, динаміку процентних ставок, доступність фінансування та ризиковий профіль підприємства інтегруються в прогнозні моделі, що дозволяє оцінити оптимальний рівень боргового навантаження й визначити, коли вигідніше залучати позикові кошти, а коли зміцнювати власний капітал.

Крім того, оптимізація включає аналіз сценаріїв (наприклад, як зміна ставок, падіння виручки або коливання вартості активів) позначається на здатності компанії обслуговувати борг. У результаті бізнес отримує можливість не лише зменшити фінансові ризики й підвищити вартість компанії, а й забезпечити стабільну основу для реалізації інвестиційних і стратегічних проєктів. Оцінка інвестиційних проєктів на основі даних передбачає використання кількісних моделей, глибокої аналітики та сучасних технологій для прийняття обґрунтованих інвестиційних рішень, мінімізації ризиків і максимізації вартості бізнесу. У цьому процесі застосовуються як класичні фінансові інструменти, а саме: NPV, IRR, PI, період окупності, аналіз дисконтованих грошових потоків, так і розширені методи, включно з прогновною аналітикою, моделями машинного навчання, бенчмаркінгом та оцінкою ризику на рівні проєкту.

Дані збираються з внутрішніх і зовнішніх джерел, а саме: історичні фінансові показники, ринкова кон'юнктура, макроекономічні індикатори, поведінка споживачів, операційні витрати та можливі сценарії розвитку середовища. На їх основі будуються моделі прогнозування доходів і витрат, визначаються потенційні точки зростання та ризики. Інструменти аналізу чутливості дозволяють оцінити, як зміни ключових параметрів, а саме: ціни, обсягу продажу, собівартості, капітальних витрат чи ставки дисконту впливають на ефективність проєкту. Сценарне моделювання та стрес-тестування дають змогу оцінити поведінку проєкту в умовах невизначеності або кризових ситуацій. У підсумку підхід, заснований на даних, забезпечує

комплексну, точну та стратегічно орієнтовану оцінку інвестицій, що дає можливість підприємству спрямовувати капітал у найбільш перспективні й стійкі ініціативи.

Алгоритмічне управління портфелем – це підхід, за якого рішення щодо формування, ребалансування та оптимізації інвестиційного портфеля приймаються на основі математичних моделей, статистичних методів та алгоритмів машинного навчання, що працюють з великими масивами ринкових і фундаментальних даних. Алгоритми враховують динаміку цін активів, волатильність, кореляції, ліквідність, макроекономічні фактори та ризикові показники, що дозволяє автоматично будувати портфелі з оптимальним співвідношенням ризику й доходності. Для цього застосовуються такі моделі, як Mean-Variance Optimization Марковіца, Black-Litterman, факторні моделі, а також прогнознi підходи (ARIMA, ML-регресія, нейронні мережі) з метою передбачення поведінки активів.

Алгоритмічні стратегії можуть бути як пасивними (автоматична ребалансування, Smart Beta), так і активними (тренд-фолловінг, арбітражні моделі, high-frequency trading). Крім того, у портфель інтегруються модулі контролю ризиків, а саме: VaR, CVaR, стрес-тести, ліміти волатильності, які забезпечують оперативне реагування на ринкові шоки. Такий підхід дозволяє зменшити людський фактор, підвищити швидкість ухвалення рішень, збільшити точність оцінок та забезпечити стійке виконання інвестиційної стратегії навіть у мінливих ринкових умовах.

Отже, аналітика у фінансах є фундаментальним інструментом для прийняття обґрунтованих управлінських рішень та забезпечення фінансової стабільності підприємств. Використання моделей прогнозування, таких як DCF, FCFF, FCFE та аналіз часових рядів, дозволяє точно оцінювати майбутні грошові потоки та фінансові результати. Одночасно ефективна оцінка ризиків забезпечує зниження фінансових втрат і дозволяє будувати стратегії хеджування та диверсифікації. Інтеграція аналітичних даних у процес

управління капіталом сприяє оптимізації ресурсів, підвищенню рентабельності та зміцненню конкурентних позицій компанії. Фінансова аналітика дозволяє поєднати точність прогнозів з контролем ризиків, що робить її ключовим чинником стратегічного планування та стабільного розвитку бізнесу.

Питання для самоперевірки:

1. Яку роль відіграє фінансова аналітика у стратегічному управлінні компанією?
2. Які ключові аспекти використання фінансової аналітики для прийняття рішень?
3. Які бізнес-функції взаємодіють із фінансовою аналітикою та як це впливає на прийняття рішень?
4. У чому полягає відмінність між кількісними та якісними методами фінансового прогнозування?
5. Що таке сценарне моделювання та аналіз чутливості і як вони допомагають управлінню ризиками?
6. Які переваги та обмеження мають моделі прогнозування для стратегічного управління?
7. Які основні види фінансових ризиків існують і чим вони відрізняються?
8. Які інструменти застосовуються для управління операційними ризиками та шахрайством?
9. Як визначається оптимальна структура капіталу з точки зору власного та позикового фінансування?
10. Яку роль відіграють стрес-тестування та аналіз чутливості у прийнятті рішень щодо капіталу та інвестицій?

Тести для самоконтролю:

1. Фінансова аналітика в стратегічному управлінні потрібна для:
 - a) Визначення корпоративного бренду;
 - b) Прогнозування фінансових показників та оцінки ризиків; 
 - c) Розробки дизайну продукту;
 - d) Найму персоналу.

2. Що забезпечує використання фінансової аналітики у прийнятті рішень?
- a) Випадкові рішення;
 - b) Обґрунтовані рішення на основі даних; ✓
 - c) Лише короткострокову вигоду;
 - d) Ігнорування ринкових тенденцій.
3. Який з наведених аспектів не входить до ролі фінансової аналітики?
- a) Аналіз ринкових тенденцій;
 - b) Прогнозування майбутніх ризиків;
 - c) Прийняття стратегічних рішень;
 - d) Розробка рекламного слогану. ✓
4. Інтеграція фінансових аналітичних систем дозволяє:
- a) Автоматизувати збір і обробку даних; ✓
 - b) Зменшити продуктивність персоналу;
 - c) Виключити аудит;
 - d) Ігнорувати KPI.
5. Взаємодія фінансової аналітики з HR допомагає:
- a) Визначати зарплати без бюджету;
 - b) Планувати бюджет на персонал та оцінювати продуктивність; ✓
 - c) Розробляти дизайн офісу;
 - d) Наймати нових керівників без аналізу.
6. Модель ARIMA застосовується для:
- a) Управління персоналом;
 - b) Прогнозування часових рядів; ✓
 - c) Розрахунку WACC;
 - d) Диверсифікації активів.
7. Multiples (Comparable Company Analysis) дозволяє:
- a) Порівнювати компанію з аналогами на ринку; ✓
 - b) Прогнозувати сезонні продажі;
 - c) Визначати кредитний ризик;
 - d) Контролювати операційні витрати.
8. Операційний ризик включає:
- a) Кредитні втрати;
 - b) Технічні збої, помилки персоналу та шахрайство; ✓
 - c) Курс валют;
 - d) Оцінку мультиплікаторів.
9. Оптимальна структура капіталу забезпечує:
- a) Мінімізацію витрат на маркетинг;

- b) Мінімізацію загальної вартості фінансування та фінансову стійкість; ✓
- c) Ігнорування ризиків;
- d) Виключно використання позикових коштів.

10. Стрес-тестування в управлінні капіталом дозволяє:

- a) Підвищувати зарплати;
- b) Моделювати вплив кризових ситуацій на фінансову стійкість; ✓
- c) Визначати бренд компанії;
- d) Прогнозувати маркетингові тренди.

Бібліографічний список:

1. Дзуліт З.П. (2024). Вплив бізнес-аналітики на корпоративне управління: можливості та виклики. *Менеджмент та підприємництво в Україні: етапи становлення та проблеми розвитку*. № 2. С. 39-47. DOI: <https://doi.org/10.23939/smeu2024.02.0039>.
2. Захаркін, О., Пігуль, Н., & Захаркіна, Л. (2025). РОЛЬ ФІНАНСОВОГО АНАЛІЗУ У СТРАТЕГІЧНОМУ ФІНАНСОВОМУ МЕНЕДЖМЕНТІ СУЧАСНИХ ПІДПРИЄМСТВ. *Економічні горизонти*, (4(33), 34–44. [https://doi.org/10.31499/2616-5236.4\(33\).2025.341370](https://doi.org/10.31499/2616-5236.4(33).2025.341370)
3. Іванова М., Варяниченко О., Архипенко Т. (2025). Стратегічне планування експортної діяльності: інструменти аналізу, контролю та оцінки ефективності для забезпечення сталості бізнес-процесів підприємства. *Економіка та суспільство*. №. 78. DOI: <https://doi.org/10.32782/2524-0072/2025-78-87>
4. Ковернінська, Ю. (2024). РОЛЬ АНАЛІЗУ ФІНАНСОВОГО ПОТЕНЦІАЛУ ПІДПРИЄМСТВА У ФОРМУВАННІ ФІНАНСОВОЇ СТРАТЕГІЇ ЙОГО ДІЯЛЬНОСТІ. *Економічні горизонти*, (2-3(28), 266–274. [https://doi.org/10.31499/2616-5236.3\(28\).2024.312229](https://doi.org/10.31499/2616-5236.3(28).2024.312229)
5. Крищенко, К.Є., Гончарук, Я.М., Рошка, Д.І., & Русаль, А.І. (2025). Роль бізнес-аналітики у стратегічному менеджменті фінансових деривативів для хеджування ризиків. *Актуальні питання економічних наук*, (17). <https://doi.org/10.5281/zenodo.17712949>

6. Чорновол А.О., Штерма Т.В., Коваль О.А. (2025). Використання бізнесаналітики для розроблення стратегій управління ризиками на фондовому ринку. *Наукові записки Львівського університету бізнесу та права*. № 46. С. 69-78. DOI: <https://doi.org/10.5281/zenodo.17332686>
7. Fu Y. (2025). The application of big data analytics to stock and derivatives trading strategies. *Economics and Management Innovation*. Vol. 2, № 3. P. 21–27. DOI: <https://doi.org/10.71222/k5gw0z43>
8. Marthanda Suherlan, A.L. (2024). Strategic Financial Management for Performance Enhancement and Sustainable Growth. *Arthatama: Journal of Business Management and Accounting*, 8(1), 18-29. Retrieved from <https://journal.lifescifi.com/index.php/art/article/view/518>
9. Olayinka, A.A. (2022). Financial statement analysis as a tool for investment decisions and assessment of companies' performance. *International Journal of Financial, Accounting, and Management*, 4(1), 49-66. <https://doi.org/10.35912/ijfam.v4i1.852>
10. Purnomo E., Alfiansyah R. (2025). A dynamic nexus: integrating big data analytics and distributed computing for real-time risk management of derivatives portfolios. *International Journal of Intelligent Data and Machine Learning*. Vol. 2, № 10. P. 37-44. DOI: <https://doi.org/10.55640/>

Тема 2. Маркетингова аналітика: поведінка споживачів, сегментація та цифрові канали

Мета: Сформувати у студентів системне розуміння методів та інструментів маркетингової аналітики, зокрема аналізу поведінки споживачів, сегментації ринку та ефективності цифрових каналів, що дозволяє приймати обґрунтовані маркетингові рішення на основі даних.

Завдання:

- Ознайомитися з джерелами та типами маркетингових даних, навчитися інтегрувати їх для аналітики.

- Вивчити методи аналізу поведінки споживачів, включаючи виявлення патернів, уподобань та тенденцій покупок.
- Засвоїти підходи до сегментації ринку та таргетування клієнтських груп для підвищення ефективності маркетингових стратегій.
- Опанувати інструменти аналітики цифрових маркетингових каналів, оцінку ROI, ефективності кампаній та поведінки онлайн-аудиторії.

7.2.1. Дані в маркетингу: джерела, типи, інтеграція.

Маркетингова аналітика сьогодні виступає ключовим інструментом для глибокого розуміння ринку та побудови ефективних стратегій, поєднуючи комплексний аналіз поведінки споживачів, сучасні підходи до сегментації та оптимізацію цифрових каналів комунікації. Насамперед поведінкова аналітика дозволяє виявити, як, чому та за яких обставин споживач ухвалює рішення, аналізуючи весь шлях клієнта, від першого контакту з брендом до повторної покупки та рекомендацій.

Для цього використовуються дані вебаналітики, теплові карти сайтів, соціальні сигнали, відгуки та транзакційні дані, що дає змогу не лише передбачати мотивацію користувачів, а й активно формувати її, пропонуючи персоналізований досвід. Водночас сегментація ринку набуває багатовимірного характеру. Окрім класичних демографічних та географічних параметрів дедалі більшого значення набувають психографічні, поведінкові та ціннісні групи, створені на основі алгоритмів машинного навчання та кластеризації великих масивів даних. Це дозволяє компаніям переходити від масового маркетингу до мікросегментації, де комунікація налаштовується з точністю до окремого користувача.

Завершальним елементом є оптимізація цифрових каналів, таких як: SEO, SMM, контент-маркетинг, таргетинг, e-mail та омніканальні платформи, що завдяки аналітиці перетворюються з інструментів просування на системи точного впливу. Аналітика дозволяє оцінювати ефективність, прогнозувати конверсії, визначати найцінніші сегменти та розподіляти бюджет так, щоб забезпечити максимальну віддачу від інвестицій у маркетинг. Таким чином,

інтелектуальна маркетингова аналітика формує основу для стратегічного управління брендом, створюючи конкурентні переваги, які базуються на точному знанні потреб і поведінки споживачів та гнучкому використанні цифрових каналів у швидкозмінному середовищі.

У сучасному маркетинговому середовищі дані перетворилися на ключовий стратегічний актив, що формує конкурентоспроможність компаній та визначає ефективність їхніх рішень. На відміну від традиційного інтуїтивного маркетингу, аналітика дозволяє підприємствам розуміти реальну поведінку та мотиви споживачів, оцінювати ефективність кампаній і прогнозувати зміни попиту. Величезні масиви даних, від цифрових слідів у соціальних мережах до точних записів транзакцій, створюють нову реальність, де маркетинг стає наукою, а не набором припущень.

Компанії формують власні дата-екосистеми, інвестують у хмарні сховища, CDP (Customer Data Platform) та інструменти автоматизації, адже саме завдяки даним вони здатні пропонувати релевантний контент, персоналізувати взаємодію та будувати довгострокові відносини зі споживачами. Таким чином, дані в маркетингу стають не лише основою аналітики, а й фундаментом для розвитку бізнес-моделей, оптимізації бюджетів і формування цінності бренду.

Джерела маркетингових даних охоплюють широкий спектр внутрішньої та зовнішньої інформації, що дає змогу компаніям отримувати комплексне бачення ринку та своїх клієнтів. Внутрішні джерела – це CRM-системи, історія покупок, дані кол-центрів, результати рекламних кампаній, вебаналітика, мобільні застосунки, програми лояльності та дані операційних процесів. Вони відображають реальні взаємодії клієнтів з брендом і дозволяють аналізувати поведінку, частоту покупок, середній чек, задоволеність та відтік.

Зовнішні джерела охоплюють ринкові дослідження, соціальні мережі, відкриті дані, звіти дослідницьких агентств, конкурентну аналітику, дані з маркетплейсів і сторонніх платформ. Особливо важливими стають дані соціальних трендів, поведінкові сигнали та контекстні фактори (наприклад,

економічна ситуація або сезонність), що дозволяють коригувати стратегії у режимі реального часу. Комбінація внутрішніх і зовнішніх джерел створює повну картину клієнтської поведінки та ринкових можливостей.

Різноманіття джерел формує широкий спектр типів даних, кожен із яких потребує специфічних методів аналізу та обробки. Структуровані дані – це організована інформація у вигляді таблиць, числових показників та чітких категорій, наприклад: продажі, клієнтські профілі, демографія. Вони ідеально підходять для класичного статистичного аналізу та моделей прогнозування. Неструктуровані дані складаються з текстів, зображень, відео, коментарів у соцмережах, відгуків, пошукових запитів, тобто всього, що не має чіткої структури. Робота з ними вимагає методів NLP, класифікації емоцій, розпізнавання образів та семантичного аналізу.

Третій тип – потокові дані (streaming data), які надходять у режимі реального часу, наприклад поведінка користувачів на сайті, кліки у мобільному додатку чи реакції на онлайн-рекламу. Такі дані важливі для динамічної оптимізації кампаній, таргетингу, A/B тестів і рекомендаційних систем. У сукупності всі типи даних створюють багатовимірний інформаційний простір, який забезпечує можливість швидко реагувати на зміни й приймати точні рішення.

Інтеграція даних – це критичний етап, що перетворює розрізнені джерела інформації на цілісну аналітичну систему. У більшості компаній дані зберігаються у різних відділах, таких як: маркетингу, продажах, підтримці клієнтів, фінансах, e-commerce, що створює фрагментацію та інформаційні “сайли”. Щоб подолати ці обмеження, організації використовують CDP, DMP, ERP, API-інтеграції та хмарні дата-платформи. Інтеграція дозволяє створити універсальний клієнтський профіль (Single Customer View), у якому об’єднуються всі дані про поведінку споживача: покупки, вподобання, історія комунікацій, реакція на рекламу, контентні інтереси. Такі системи забезпечують синхронізацію офлайн- і онлайн-даних, автоматизацію маркетингових кампаній, сегментацію та персоналізацію. Завдяки

інтегрованим даним компанії отримують повну прозорість маркетингової діяльності та здатні будувати предиктивні моделі, оптимізувати бюджет, визначати найцінніші канали й підвищувати рентабельність інвестицій.

Попри значні переваги, інтеграція та використання маркетингових даних супроводжуються низкою викликів, а саме: забезпеченням якості даних, кібербезпекою, відповідністю нормам GDPR та локального законодавства, а також необхідністю високої цифрової компетентності персоналу. Компанії стикаються з проблемами дублювання, неповноти або застарілості даних, що впливає на точність аналітики й ефективність рішень.

Водночас розвиток технологій, а саме: штучного інтелекту, машинного навчання, розподілених баз даних, блокчейну відкриває нові можливості для безпечного, швидкого та масштабованого управління інформацією. Персоналізація в реальному часі, прогностична аналітика, автоматизовані маркетингові платформи та автономні цифрові кампанії стають реальністю. Перспективи розвитку маркетингових даних лежать у створенні гнучких дата-екосистем, де інформація працює на формування унікального клієнтського досвіду, інноваційних продуктів і стабільної конкурентної переваги бренду.

7.2.2. Аналіз поведінки споживачів.

Аналіз поведінки споживачів є фундаментальним елементом сучасного маркетингу, оскільки він дозволяє зрозуміти, як і чому люди ухвалюють рішення про покупку, які емоційні та раціональні чинники на це впливають, та які бар'єри можуть зупинити їх від придбання продукту. У цифровому середовищі поведінка перестає бути лінійною й перетворюється на складну мережу взаємодій із брендом, медіаконтентом та інформаційними сигналами. Споживач одночасно виступає дослідником, критиком і творцем контенту, тому маркетинг більше не може покладатися на інтуїцію чи окремі метрики — потрібен системний аналіз усіх етапів життєвого циклу клієнта. Поведінкові моделі допомагають не лише передбачити майбутні дії покупця, а й формувати їх, створюючи релевантний досвід і точковий вплив.

Ключовим елементом поведінкового аналізу є виявлення мотивацій, що стоять за рішеннями споживачів. Вони можуть бути функціональними (наприклад, потреба у вирішенні конкретної проблеми), емоційними (наприклад, бажання статусу, комфорту, безпеки) або соціальними (наприклад, вплив друзів, трендів, зміна норм споживання). Сучасні компанії активно використовують методи нейромаркетингу, аналіз біометричних показників та текстову аналітику, щоб зрозуміти, яка комбінація емоцій рухає покупцем у момент вибору. Мотивація часто прихована, неусвідомлена, тому традиційні опитування вже не дають повної картини, тобто необхідний аналіз поведінкових патернів, реакцій на контент, швидкості перегляду сторінок та мікровзаємодій.

Вивчення поведінки неможливе без аналізу клієнтського шляху, який охоплює всі точки контакту, від першого знайомства з брендом до післяпродажної взаємодії. У сучасних умовах *journey* поділяється на десятки мікрофаз, які включають пошук інформації, порівняння альтернатив, взаємодію з рекомендаціями, перегляд відгуків, споживання контенту та повторні покупки. Поведінкова аналітика фіксує кожен з цих кроків, дозволяючи виявити фрустраційні точки, моменти ухвалення рішень та імпульсні тригери. Завдяки цьому компанії отримують можливість адаптувати UX, оптимізувати контентні ланцюжки та створювати персоналізовані сценарії, що збільшують конверсію в усіх каналах.

Сучасний аналіз поведінки значною мірою ґрунтується на цифрових слідах, таких як: кліках, скролах, часі на сторінці, переходах між блоками сайту, реакціях у соціальних мережах і навіть шаблонах навігації. Ці мікровзаємодії дозволяють зрозуміти реальні наміри споживача: чи він лише цікавиться продуктом, чи вже готовий до покупки. Теплові карти, трекінг курсору, алгоритми рекомендацій та аналіз патернів роблять можливим прогнозування поведінки ще до того, як споживач сам усвідомить свою потребу. Компанії використовують ці дані для запуску тригерних кампаній,

попередження відтоку та формування релевантного досвіду в режимі реального часу.

Щодо психографіки та поведінкової сегментації як інструментів точного таргетингу, компанії переходять до використання психографічних моделей, що включають цінності, стилі життя, особистісні риси, мотивації та поведінкові звички. Машинне навчання та кластеризація великих масивів даних дозволяють виявити нові сегменти, які не піддаються традиційному опису. Наприклад, «емоційні покупці миттєвих рішень» або «раціональні дослідники ринку». Такі групи мають різні реакції на комунікацію, різну чутливість до ціни та різні стимули до повторної взаємодії. Поведінкова сегментація дає змогу переходити до мікротаргетингу, де меседж адаптується під конкретну поведінкову модель.

Завдяки даним та алгоритмам штучного інтелекту, компанії можуть прогнозувати майбутні дії споживачів, а саме: імовірність покупки, час наступної транзакції, ризик відтоку, реакцію на знижку або специфічний контент. Predictive-моделі використовують історичні та потокові дані, що дозволяє оптимізувати бюджет і зосередитись на користувачах із найбільшою конверсійною цінністю. Це змінює підхід до маркетингу, а саме: замість реактивних дій компанії діють проактивно, не чекають, поки клієнт втратить інтерес, а діють у момент, коли найімовірніше вплинути на його поведінку.

Соціальні мережі стали одним із головних каналів впливу на поведінку, оскільки вони формують соціальні норми, цінності й тренди. Аналіз настроїв, емоційних реакцій, контенту, що користувачі створюють або поширюють, дозволяє виявити не лише раціональні, а й культурні чинники поведінки. Бренди активно використовують емоційну аналітику для прогнозування вірусності контенту, визначення рівня довіри та оцінки ризиків репутації. Розуміння того, як емоції циркулюють у соціальних мережах, допомагає компаніям будувати автентичний діалог зі споживачами та впливати на поведінку через спільноти, мікроінфлюенсерів та соціальні патерни.

Останнім етапом поведінкової аналітики є інтеграція отриманих інсайтів у стратегію компанії. Це включає розробку персоналізованих комунікацій, динамічних цінових моделей, удосконалення UX-дизайну, оптимізацію рекламних кампаній та креативу, а також адаптацію продуктів під реальні потреби споживачів. Поведінкові інсайти дозволяють створювати ціннісний клієнтський досвід, що стимулює залученість, лояльність і довгострокову взаємодію з брендом. Вони трансформують маркетинг у точну, науково обґрунтовану дисципліну, у якій реальні дані, а не припущення, визначають успіх продукту.

Важливим виміром аналізу поведінки споживачів стає дослідження їхніх емоційних тригерів, які визначають реакції на маркетингові стимули. Сучасні нейромаркетингові методи, від відстеження мікроміміки до аналізу частоти серцевих скорочень під час взаємодії з рекламою, дозволяють зрозуміти, які емоційні стани викликає конкретний бренд, продукт або повідомлення. Оскільки більшість рішень ухвалюється підсвідомо, компанії прагнуть ідентифікувати патерни, що формують емоційний відгук: довіру, натхнення, бажання, страх втратити можливість або потребу у приналежності. Емоційні реакції безпосередньо впливають на лояльність, тому бренди працюють над створенням багат шарових досвідів, а саме: аудіо-візуального контенту, адаптивних інтерфейсів, персоналізованих наративів, які стабільно викликають позитивну емоційну взаємодію зі споживачем.

У цифровій екосистемі споживачі демонструють нову форму поведінки, а саме: постійне перемикавання уваги та миттєву зміну рішень залежно від контексту взаємодії. Це вимагає аналізу мікромоментів, тобто конкретних точок часу, коли людина відчуває потребу «знайти», «купити», «порівняти» або «дізнатися». Маркетологи досліджують швидкість і послідовність цих реакцій, щоб зрозуміти, які фактори впливають на споживача в конкретний момент. Поведінковий аналіз дозволяє створювати тригери, що активуються у реальному часі: push-сповіщення, персоналізовані рекомендації, адаптивні банери, які реагують на поведінку користувача в секунду її прояву. Компанії,

які коректно працюють з мікромоментами, здатні перехопити увагу споживача та підвищити конверсію без додаткового тиску чи агресивної реклами.

Аналіз поведінки споживачів також включає вивчення соціальних динамік, що формуються у спільнотах, онлайн-групах, нішевих форумах та мережесистемах. Сучасні користувачі часто орієнтуються на думку лідерів думок, мікроінфлюенсерів або членів локальних ком'юніті, що значно зміщує акцент з традиційної реклами на рекомендаційні механізми. Поведінкові моделі в спільнотах будуються на довірі, репутації та відчутті участі, тому бренди аналізують інтонації, частоту обговорення, соціальні ролі та структури впливу в онлайн-середовищах. Це дозволяє впроваджувати програмне “соціальне слухання”, через яке компанії відстежують настрої, болі й очікування аудиторії, завчасно реагуючи на негатив або формуючи позитивний наратив у природний для спільнот спосіб.

Ще одним важливим аспектом є аналіз мультиканальної поведінки, де споживач одночасно взаємодіє з брендом через фізичні та цифрові точки. Потреба зрозуміти такі крос-канальні маршрути змушує бізнес використовувати моделювання customer journey з деталізацією кожної точки контакту. Поведінкові дані допомагають зрозуміти, що реально впливає на ухвалення рішення: чи то фізичний досвід у магазині, чи push-повідомлення під час пересування, чи соціальний доказ у вигляді відгуків. Аналізуючи перемикання між пристроями, а саме: телефон, ноутбук, офлайн-магазин, мобільний додаток, компанії виявляють закономірності та розробляють гібридні стратегії, що підсилюють кожен канал, створюючи безперервний та комфортний шлях для користувача.

Глибокий аналіз поведінки споживачів також вимагає дослідження їхніх довгострокових змін, трансформації цінностей, очікувань і моделей сприйняття брендів. Зміна поколінь, економічні кризи, війни, цифровізація та екологічні виклики формують нові поведінкові архетипи. Наприклад, молоде покоління Z демонструє прагнення до автентичності, швидкої комунікації та технологічної інтерактивності, тоді як покоління міленіалів цінує стабільність, репутацію

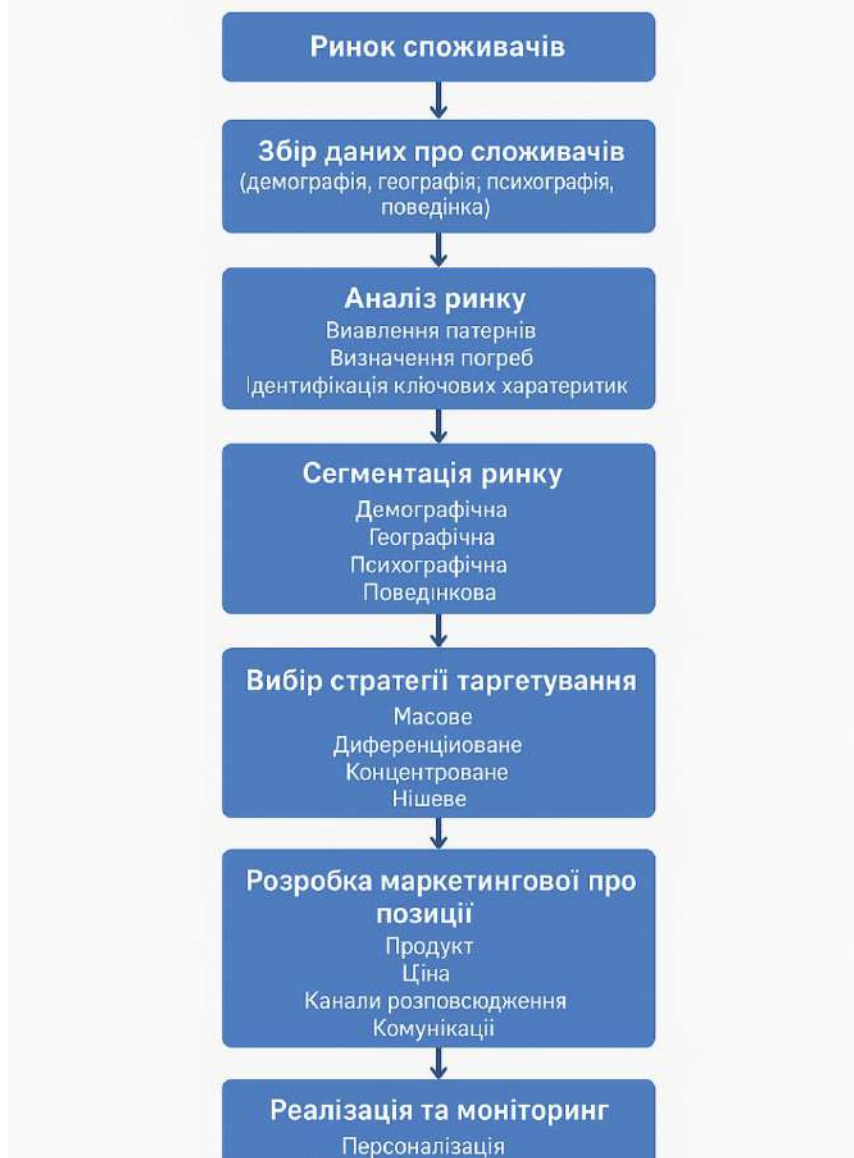
бренду й корисність продуктів. Тому поведінковий аналіз включає когортні дослідження, оцінку довгострокової лояльності, аналіз зміни моделей споживання та прогнозування поведінкових трендів. Це дозволяє компаніям не лише реагувати на існуючі потреби, а й будувати продукти, які відповідатимуть майбутнім очікуванням ринку.

7.2.3. Сегментація та таргетування.

Сегментація ринку – це стратегічний процес поділу великого ринку на окремі групи споживачів, які мають схожі характеристики, потреби або поведінкові патерни. Основною метою сегментації є ефективне задоволення потреб клієнтів за допомогою більш персоналізованих продуктів і маркетингових стратегій. Наприклад, компанія Procter & Gamble застосовує сегментацію для окремих лінійок засобів догляду за шкірою, розділяючи продукти за типом шкіри та віковими групами споживачів. Такий підхід дозволяє підвищити точність рекламних кампаній та покращити взаємодію з цільовою аудиторією.

Сегментація ринку може здійснюватися за різними критеріями, такими як: демографічними, географічними, психографічними та поведінковими. Демографічна сегментація включає такі параметри, як: вік, стать, дохід і освіта, а географічна: країну, регіон або кліматичну зону. Психографічна сегментація розглядає стиль життя, цінності та особистісні риси, тоді як поведінкова орієнтується на звички споживачів, рівень лояльності та частоту покупок.

СЕГМЕНТАЦІЯ ТА ТАРГЕТУВАННЯ



Географічна сегментація передбачає поділ ринку на основі місця проживання споживачів або специфіки регіону. Компанії враховують кліматичні умови, культурні особливості та регіональні вподобання. Наприклад, McDonald's адаптує меню залежно від країни. Наприклад, в Індії він пропонує більше вегетаріанських страв, тоді як у США – класичні бургери та картоплю фрі. Такий підхід дозволяє збільшити привабливість бренду на локальних ринках. Демографічна сегментація базується на об'єктивних характеристиках споживачів: вік, стать, сімейний стан, освіта, дохід тощо. Цей тип сегментації є одним із найпоширеніших завдяки своїй простоті та

зрозумілості. Наприклад, L'Oréal розробляє косметичні лінійки для різних вікових груп та типів шкіри, щоб кожна категорія клієнтів отримувала продукти, оптимально адаптовані до їхніх потреб.

Психографічна сегментація враховує стиль життя, цінності, інтереси та соціальні статуси споживачів. Цей підхід дозволяє компаніям більш глибоко зрозуміти мотивацію клієнтів. Наприклад, Apple орієнтується на інноваційних користувачів та творчу аудиторію, використовуючи психографічний підхід для просування своїх технологічних продуктів. Поведінкова сегментація передбачає поділ ринку за звичками, частотою покупок, рівнем лояльності та ставленням до продукту. Вона дозволяє компаніям створювати програми лояльності та спеціальні пропозиції. Наприклад, Starbucks аналізує поведінку своїх клієнтів через мобільний додаток, пропонуючи персоналізовані знижки та бонуси для постійних відвідувачів.

Цей підхід передбачає поділ споживачів залежно від того, які вигоди вони шукають у продукті. Наприклад, у сфері косметики одні клієнти цінують натуральність, інші – ефективність, а треті – преміум-бренд. Компанія Dove сегментує ринок на тих, хто шукає зволоження шкіри, тих, хто бажає догляду за волоссям, і тих, хто приділяє увагу етичності продукту, що дозволяє максимально точно задовольнити запити кожного сегменту. Після визначення сегментів важливо оцінити їх потенціал: розмір, зростання, прибутковість, доступність і відповідність ресурсам компанії. Аналітичні методи включають використання статистичних даних, опитувань, фокус-груп та цифрової аналітики. Наприклад, Amazon активно застосовує цифрову аналітику та машинне навчання для оцінки потенційних сегментів та створення персоналізованих рекомендацій.

Таргетування полягає у виборі конкретних сегментів, на які компанія зосередить свої маркетингові зусилля. Основні стратегії таргетування включають диференційоване, концентроване, нішеве та масове таргетування. Наприклад, Tesla використовує концентроване таргетування, орієнтуючись на

сегмент споживачів із високим доходом, що цікавляться електромобілями та інноваційними технологіями. Масове таргетування орієнтується на широкий ринок без сегментації, пропонуючи стандартний продукт для всіх. Цей підхід ефективний, коли потреби клієнтів досить однорідні. Наприклад, Coca-Cola Classic використовується як універсальний продукт для глобальної аудиторії, що дозволяє знизити витрати на виробництво та маркетинг.

Диференційоване таргетування передбачає розробку окремих маркетингових стратегій для кожного сегменту. Наприклад, Nestlé пропонує різні види кави, шоколаду та дитячого харчування, створюючи продукти для різних вікових і соціальних груп, що дозволяє максимально відповідати потребам споживачів і підвищувати лояльність. Концентрований маркетинг передбачає зосередження на одному великому сегменті, а нішевий – на вузькій специфічній групі. Наприклад, Rolex використовує нішевий підхід, орієнтуючись на сегмент заможних клієнтів, для яких престиж і статус важливіші за функціональність. Такий таргетинг дозволяє створити унікальний бренд і високий рівень лояльності.

Сучасні технології дозволяють значно підвищити точність сегментації та таргетування. Великі дані, штучний інтелект та аналітика поведінки користувачів дають змогу компаніям адаптувати продукти та комунікації до індивідуальних потреб. Наприклад, Netflix аналізує перегляди та оцінки користувачів, щоб створювати персоналізовані рекомендації та контент, який максимально відповідає вподобанням кожного сегмента. Сегментація та таргетування є ключовими елементами маркетингової стратегії сучасних компаній. Вони дозволяють більш точно задовольняти потреби клієнтів, оптимізувати ресурси та підвищувати прибутковість. Використання комбінованих методів сегментації, цифрових інструментів та різних стратегій таргетування допомагає компаніям залишатися конкурентоспроможними на динамічних ринках. Приклади таких компаній, як: Nike, Coca-Cola, Apple,

Tesla, Nestlé та Netflix, демонструють ефективність поєднання аналітичного підходу та інноваційних технологій для досягнення стратегічних цілей.

7.2.4. Аналітика цифрових маркетингових каналів.

Аналітика цифрових маркетингових каналів є фундаментальним компонентом сучасного маркетингу, оскільки дозволяє компаніям оцінювати ефективність своїх онлайн-кампаній, розуміти поведінку користувачів та приймати обґрунтовані стратегічні рішення. В умовах зростання кількості цифрових платформ, від соціальних мереж до пошукових систем та мобільних додатків, організації стикаються з необхідністю інтеграції великих обсягів даних, щоб зрозуміти, які канали забезпечують максимальну віддачу від маркетингових інвестицій (ROI). Наприклад, компанія Nike активно використовує аналітику для вимірювання ефективності своїх рекламних кампаній у соціальних мережах, що дозволяє оптимізувати бюджет і підвищувати взаємодію з клієнтами.

Цифрові канали можна поділити на органічні та платні. Органічні включають: SEO, контент-маркетинг і соціальні мережі без прямої оплати за просування, тоді як платні – це контекстна реклама, банерна реклама, платні кампанії у соціальних мережах та email-маркетинг. Кожен канал має свої метрики ефективності. Наприклад, SEO-кампанії оцінюються через органічний трафік та позиції у пошукових системах, а платні кампанії у Google Ads – через CTR (Click-Through Rate), CPC (Cost Per Click) та конверсії. Компанія Coca-Cola демонструє ефективне поєднання органічних та платних каналів, використовуючи дані аналітики для підвищення впізнаваності бренду та залучення молодшої аудиторії.

Аналітика цифрових маркетингових каналів базується на ключових показниках ефективності (KPI), таких як: CTR, конверсії, середній час на сайті, коефіцієнт відмов і LTV (Life Time Value). Наприклад, електронна комерція Amazon використовує KPI для оцінки того, наскільки рекламні кампанії впливають на продажі та повторні покупки, корелюючи поведінку

користувачів із джерелами трафіку. Важливою частиною аналітики є також атрибуція, тобто визначення внеску кожного каналу у досягнення бізнес-цілей, що дозволяє маркетологам правильно розподіляти бюджет між різними цифровими каналами.

Сучасні маркетологи використовують широкий спектр інструментів аналітики, включаючи такі як: Google Analytics, Adobe Analytics, Hotjar, SEMrush та HubSpot. Ці платформи дозволяють відстежувати трафік, конверсії, поведінку користувачів та ефективність кампаній у реальному часі. Наприклад, Airbnb інтегрує дані з кількох джерел у HubSpot для комплексного аналізу користувацького шляху, що дозволяє виявляти слабкі точки у маркетинговій стратегії та оптимізувати рекламні витрати.

Соціальні мережі є ключовим каналом комунікації брендів із аудиторією. Аналітика соцмереж включає вимірювання охоплення, залученості (engagement), кількості підписників, відгуків та конверсій. Наприклад, Netflix активно використовує аналітичні інструменти для оцінки реакції глядачів на нові серіали через Instagram та TikTok, коригуючи контентні стратегії на основі отриманих даних. Також соціальні платформи пропонують власну аналітику, наприклад Facebook Insights або LinkedIn Analytics, які дають змогу оцінити ефективність конкретних публікацій та рекламних кампаній.

SEO-аналітика допомагає оцінювати видимість сайту у пошукових системах, обсяг органічного трафіку, позиції за ключовими словами та технічний стан ресурсу. Компанії, як Booking.com, активно використовують SEO-аналітику для оптимізації контенту на своїх сайтах, покращення ранжування та збільшення органічного потоку користувачів. Крім того, SEO-аналітика включає аналіз конкурентів і пошукових трендів, що дозволяє приймати стратегічні рішення щодо створення нових сторінок та контенту.

Email-маркетинг залишається ефективним каналом комунікації, особливо для сегментованої аудиторії. Аналітика цього каналу включає відкриття листів (Open Rate), кліки (Click Rate), конверсії та показник відмов. Наприклад,

компанія Spotify аналізує, які листи та плейлисти отримують найбільше уваги користувачів, щоб підвищувати персоналізацію пропозицій та збільшувати утримання клієнтів. Розширені аналітичні платформи дозволяють тестувати A/B-версії повідомлень і визначати оптимальні час та частоту розсилок.

Платні рекламні канали, такі як: Google Ads та медійні банери, вимагають детальної аналітики для оцінки ефективності. Основні метрики включають CPC (вартість кліку), CPM (вартість за тисячу показів) та ROI. Наприклад, Zara застосовує Google Ads для просування нових колекцій, аналізуючи дані про кліки та конверсії для оптимізації ставок та розподілу бюджету. Аналітика дозволяє швидко реагувати на зміну ринку і підвищувати ефективність рекламних кампаній.

З ростом використання смартфонів мобільний маркетинг стає критично важливим. Аналітика мобільних каналів включає показники встановлення додатків, активність користувачів, час у додатку та конверсії. Компанія Uber активно відстежує поведінку користувачів у своєму мобільному додатку, аналізуючи, які функції сприяють повторним замовленням, що дозволяє підвищувати утримання клієнтів і знижувати відтік.

Для онлайн-ритейлу аналітика каналів включає моніторинг конверсій, середнього чеку, кошика та відсотка завершення покупки. eBay використовує інструменти аналітики для виявлення проблемних етапів у користувацькому шляху, оптимізації сторінок товарів і рекламних кампаній, що сприяє збільшенню продажів та покращенню досвіду покупців. Інтеграція аналітики з CRM-системами дозволяє компаніям будувати комплексне бачення клієнтського шляху.

Важливим аспектом аналітики є моделювання атрибуції, тобто визначення внеску кожного каналу у досягнення конверсії. Існують моделі Last Click, First Click, Linear, Time Decay та Data-Driven, що дозволяють розподіляти ефективність між різними точками контакту користувача. Наприклад, Adobe застосовує Data-Driven атрибуцію для комплексного розуміння, які канали

найбільше впливають на продажі, що допомагає більш раціонально планувати бюджет маркетингових кампаній.

Сучасна аналітика цифрових каналів неможлива без Big Data та штучного інтелекту. AI дозволяє прогнозувати поведінку користувачів, автоматизувати персоналізацію та оптимізацію рекламних кампаній. Наприклад, Alibaba застосовує алгоритми машинного навчання для передбачення потреб користувачів і рекомендації товарів, що значно підвищує конверсії та ефективність маркетингових зусиль.

Візуалізація даних є ключовим елементом аналітики, оскільки дозволяє маркетологам швидко оцінювати результати кампаній. Інструменти, такі як: Tableau, Power BI або Google Data Studio, допомагають створювати інтерактивні дашборди з усіх каналів. Наприклад, Coca-Cola використовує дашборди для відстеження глобальних кампаній у реальному часі, що дозволяє миттєво реагувати на зміни ринку і поведінку споживачів.

Незважаючи на численні переваги, аналітика цифрових каналів має виклики, серед яких фрагментація даних, обмеження конфіденційності (GDPR, CCPA), недосконалість атрибуційних моделей та складність інтеграції даних з різних джерел. Наприклад, компанії як Facebook і Google постійно адаптують аналітичні платформи до нових регуляторних вимог, що вимагає від маркетологів гнучкості та постійного навчання.

Майбутнє аналітики цифрових каналів пов'язане з подальшою автоматизацією, інтеграцією AI, прогновною аналітикою та персоналізацією на основі поведінкових даних. З розвитком метавсесвітів та нових цифрових платформ компанії, як: Meta та TikTok, матимуть можливість збирати ще більше даних про поведінку користувачів, що відкриває нові можливості для таргетованого маркетингу. Ефективна аналітика стане не лише конкурентною перевагою, а й необхідною умовою для успіху бренду в цифровому середовищі.

Отже. маркетингова аналітика є ключовим інструментом сучасного бізнесу, що дозволяє глибоко розуміти поведінку споживачів, ефективно сегментувати аудиторію та оптимізувати використання цифрових каналів. Аналіз даних з різних джерел, таких як сайти, соціальні мережі, email-розсилки та мобільні додатки, дозволяє компаніям відстежувати ключові метрики, як: CTR, конверсії, LTV, ROI та інші. Використання аналітичних платформ і дашбордів забезпечує своєчасну обробку та візуалізацію інформації, що сприяє прийняттю стратегічних рішень, таких як персоналізація комунікацій, оптимізація маркетингового бюджету та покращення користувацького досвіду. Інтегрована маркетингова аналітика дозволяє не лише підвищити ефективність кампаній, а й формувати довгострокові взаємини з клієнтами, ґрунтуючись на точних даних і реальному розумінні потреб аудиторії. У сучасному цифровому середовищі її роль стає особливо важливою для конкурентоспроможності та стійкого розвитку бізнесу.

Питання для самоперевірки:

1. Які основні внутрішні джерела маркетингових даних використовують компанії для аналізу поведінки споживачів?
2. Назвіть зовнішні джерела даних, що дозволяють оцінити ринкові тренди та конкурентне середовище.
3. Чим відрізняються структуровані, неструктуровані та потокові дані у маркетинговій аналітиці?
4. Що таке клієнтський шлях (customer journey) і як його аналіз допомагає підвищити конверсію?
5. Як цифрові сліди (кліки, скролли, реакції у соцмережах) використовуються для прогнозування поведінки споживачів?
6. У чому полягає роль психографічної та поведінкової сегментації у точному таргетингу?
7. Чим відрізняються диференційоване, концентроване, нішеве та масове таргетування?

8. Як компанії використовують великі дані та штучний інтелект для персоналізації продуктів і комунікацій у сегментації та таргетуванні?
9. Які основні метрики ефективності (KPI) цифрових каналів використовуються для оцінки ROI маркетингових кампаній?
10. Назвіть приклади інструментів аналітики цифрових каналів та поясніть, як вони допомагають оптимізувати маркетингові рішення.

Тести для самоконтролю:

1. Яка основна мета маркетингової аналітики?
 - a) Підвищити продажі без аналізу ринку;
 - b) Зрозуміти поведінку споживачів і покращити маркетингову стратегію; ✓
 - c) Тільки збирати дані про конкуренцію;
 - d) Створювати рекламу без досліджень.
2. До внутрішніх джерел даних у маркетингу належить:
 - a) Соціальні мережі;
 - b) Дані CRM та історія покупок клієнтів; ✓
 - c) Дослідження ринку сторонніх компаній;
 - d) Статистика галузі.
3. Який тип даних у маркетингу є неструктурованим?
 - a) Продажі за місяць;
 - b) Відгуки користувачів на сайті; ✓
 - c) Середній чек покупки;
 - d) Кількість відправлених email.
4. Який вид сегментації ринку базується на віці, статі, доході та освіті?
 - a) Географічна;
 - b) Демографічна; ✓
 - c) Психографічна;
 - d) Поведінкова.
5. Приклад поведінкової сегментації:
 - a) Розподіл клієнтів за регіонами;
 - b) Визначення лояльних покупців за історією покупок; ✓
 - c) Вік та стать клієнтів;
 - d) Освіта та професія.
6. Які з наведених KPI відносяться до оцінки цифрових каналів?
 - a) CTR (Click Through Rate); ✓
 - b) Валовий прибуток;

- c) ROAS (Return on Ad Spend); ✓
d) Чистий дохід компанії.
7. Модель атрибуції в цифровому маркетингу дозволяє:
a) Визначати ефективність окремих каналів; ✓
b) Підвищувати SEO;
c) Збирати демографічні дані;
d) Збільшувати кількість підписників.
8. Дані, зібрані з соціальних мереж, завжди структуровані.
a) Правда;
b) Неправда. ✓
9. Мікротаргетинг допомагає показувати персоналізовані пропозиції конкретним групам споживачів.
a) Правда; ✓
b) Неправда.
10. Інтеграція даних у маркетингу допомагає об'єднати різні джерела інформації про клієнта.
a) Правда; ✓
b) Неправда.

Бібліографічний список:

1. Бедій Н. (2024). Сегментація ринку продукції в контексті сучасних тенденцій споживчої поведінки. *Інвестиції: практика та досвід*. № 9. С. 84–90. URL: http://www.investplan.com.ua/pdf/9_2021/14.pdf
2. Галстян М., Квіта, Г. (2025). ПОВЕДІНКОВА СЕГМЕНТАЦІЯ ЯК ІНСТРУМЕНТ ПЕРСОНАЛІЗАЦІЇ МАРКЕТИНГОВИХ КОМУНІКАЦІЙ. *Економіка та суспільство*, (78). <https://doi.org/10.32782/2524-0072/2025-78-155>
3. Кабінет Міністрів України (2024). Стратегія цифрового розвитку інноваційної діяльності України на період до 2030 року. URL: <https://zakon.rada.gov.ua/laws/show/1351-2024-%D1%80#n14>
4. Коваленко Н., Малахова Ю., Гордєєв Ю. (2024). Сегментація цільової аудиторії як елемент стратегічного планування маркетингової активності в

- соціальних медіа. *Вісник Хмельницького національного університету*. № 6. С. 370-375. DOI: <https://doi.org/10.31891/2307-5740-2024-336-58>
5. Котельникова Ю.М., Петровська Є.М. (2025). Цифрова трансформація маркетингової стратегії просування підприємства. *Український журнал прикладної економіки та техніки*. Том 10. № 2. С. 313-316.
 6. Маркетинг в умовах діджиталізації економіки країни : матеріали Всеукраїнської науково-практичної інтернет-конференції (м. Полтава, 23–24 листопада 2023 року). – Полтава: ПУЕТ, 2023. – 230 с.
 7. Струк Н., Михайлик Н. (2024). Маркетингові стратегії сегментації та вибору цільового ринку. *Економіка та суспільство*. № 59. DOI: <https://doi.org/10.32782/2524-0072/2024-59-21>
 8. Naim A. (2023). Consumer Behavior in Marketing Patterns, Types, Segmentation. *European Journal of Economics, Finance and Business Development*. Vol. 1. URL: <https://cognitioncommerce.ca/wp-content/uploads/2023/11/consumer-behavior-in-marketing-patterns->
 9. Digital 2025: Global Overview Report. *Datareportal.com*. URL: <https://datareportal.com/reports/digital-2025-global-overview-report>.
 10. Unlocking the next frontier of personalized marketing. *McKinsey & Company*. 2021. URL: <https://www.mckinsey.com/capabilities/growth-marketing-and-sales/our-insights/unlocking-the-next-frontier-of-personalizedmarketing>.

Тема 3. Аналітика в HR: управління талантами, ефективність персоналу та HR-передбачення

Мета: Сформувати системне розуміння сучасних підходів до HR-аналітики та навчитися застосовувати аналітичні інструменти для управління талантами, оцінки ефективності персоналу та прогнозування HR-показників, що забезпечує підвищення результативності людського капіталу організації.

Завдання:

- Дослідити сучасні підходи до HR-аналітики та визначити її роль у прийнятті управлінських рішень.
- Проаналізувати використання HR-даних у процесах управління талантами та розвитку персоналу.
- Вивчити методи прогнозування ключових HR-показників, таких як продуктивність, плинність кадрів та залученість співробітників.
- Оцінити ефективність HR-аналітики у наймі, адаптації та формуванні стратегічної цінності людського капіталу.

7.3.1. Сучасні підходи до HR-аналітики.

Аналітика в HR на сьогодні є стратегічним інструментом для організацій, які прагнуть максимально ефективно управляти персоналом та розвивати людський капітал. Вона поєднує збір, обробку та аналіз даних про співробітників для прийняття обґрунтованих управлінських рішень, прогнозування тенденцій і підвищення продуктивності команди. Управління талантами з використанням HR-аналітики дозволяє ідентифікувати ключових співробітників, оцінювати їхні компетенції та визначати індивідуальні плани розвитку, що відповідають стратегічним цілям компанії. Наприклад, компанія Google застосовує People Analytics для визначення факторів, що впливають на ефективність команд, а Microsoft використовує HR-дані для оцінки продуктивності та задоволеності співробітників.

Аналітика персоналу також оптимізує процеси найму та адаптації, знижує плинність кадрів і підвищує залученість команди; наприклад, Unilever застосовує цифрові платформи для прогнозування успішності кандидатів ще на етапі рекрутингу. Прогнозування HR-показників, таких як продуктивність, рівень задоволеності або ймовірність звільнення, дозволяє організаціям проактивно планувати кадрові ресурси і своєчасно реагувати на ризики. Впровадження HR-аналітики підвищує не лише ефективність поточних процесів, а й формує стратегічну цінність людського капіталу, забезпечуючи компаніям конкурентні переваги у сучасному динамічному бізнес-середовищі.

Сучасні підходи до HR-аналітики базуються на інтеграції цифрових технологій, великих даних та інтелектуальних систем підтримки управлінських рішень, що дозволяє перетворити управління персоналом із традиційного адміністративного процесу на стратегічно орієнтовану діяльність. Вони спрямовані на оцінку продуктивності, прогнозування плинності кадрів, оптимізацію процесів найму та розвитку талантів, а також на підвищення залученості співробітників.

Наприклад, компанія IBM активно використовує HR-аналіз даних для прогнозування плинності та планування кар'єрного росту співробітників, що дозволяє ефективно розподіляти ресурси та знижувати ризики втрати ключових талантів. Google застосовує People Analytics для вивчення факторів ефективності команд та побудови оптимальної структури роботи, а Unilever інтегрує цифрові платформи у процеси рекрутингу, оцінюючи потенціал кандидатів ще на етапі відбору. Такі сучасні підходи дозволяють організаціям не лише підвищувати ефективність управління людським капіталом, а й створювати стратегічну цінність персоналу, адаптуючись до швидких змін на ринку праці та забезпечуючи конкурентні переваги.

Сучасні організації все активніше впроваджують цифрові рішення для управління людським капіталом, серед яких ключове місце займають, такі як: HRIS (Human Resource Information Systems), HCM-платформи (Human Capital Management) та системи обліку персоналу. HRIS є комплексними інформаційними системами, що дозволяють автоматизувати базові HR-процеси, включаючи ведення персональних даних співробітників, облік робочого часу, відпусток, лікарняних та інших адміністративних функцій. Вони забезпечують централізоване зберігання даних, знижують ризик помилок при ручному введенні інформації та спрощують звітність для менеджменту та контролюючих органів.

HCM-платформи розширюють функціонал HRIS, поєднуючи управління персоналом із стратегічними задачами організації, а саме: плануванням

ресурсів, оцінкою компетенцій, розвитком талантів, навчанням і плануванням кар'єрного росту. Вони дозволяють інтегрувати аналітичні інструменти для прогнозування потреб у персоналі, оцінки продуктивності та визначення потенціалу співробітників, забезпечуючи підхід до управління персоналом як до стратегічного активу компанії. Системи обліку персоналу у свою чергу виконують базову адміністративну роль, контролюючи документообіг, трудові контракти, нарахування заробітної плати та інші кадрові операції.

Поєднання HRIS, HCM та систем обліку персоналу дозволяє організаціям автоматизувати рутинні процеси, підвищити точність даних та забезпечити аналітичну підтримку для прийняття рішень. Наприклад, SAP SuccessFactors та Workday є прикладами HCM-платформ, які інтегрують модулі навчання, оцінки продуктивності та управління талантами в єдину систему, забезпечуючи повний цикл управління людським капіталом. Компанії, що впроваджують такі рішення, отримують можливість більш ефективно планувати ресурси, адаптувати політики мотивації та розвитку персоналу, а також оперативно реагувати на зміни в бізнес-середовищі. Крім того, сучасні платформи забезпечують мобільний доступ, гнучкі інтеграції з іншими корпоративними системами та можливість використання штучного інтелекту для автоматизації рутинних завдань, таких як сортування резюме або прогнозування плинності кадрів.

Завдяки цьому HR-фахівці отримують можливість зосередитися на стратегічних задачах, зокрема на формуванні корпоративної культури, розвитку талантів та підвищенні залученості співробітників. Впровадження таких систем підвищує прозорість процесів управління персоналом, скорочує адміністративні витрати та забезпечує оперативне отримання точних даних для аналітики. HRIS і HCM-платформи стають основою для побудови People Analytics, дозволяючи компаніям не лише реагувати на поточні кадрові потреби, а й прогнозувати розвиток організації, планувати навчання та оцінювати ефективність програм мотивації. В умовах глобалізації,

дистанційної роботи та швидких змін на ринку праці цифрові HR-системи забезпечують необхідну гнучкість та адаптивність, підтримуючи стратегічні цілі бізнесу.

Сучасні рішення також пропонують інтеграцію з аналітичними інструментами та BI-платформами, що дозволяє створювати комплексні дашборди для керівників і HR-команд, відстежувати ключові показники ефективності та приймати дані на основі реальних, актуальних даних. Важливо, що впровадження таких систем потребує ретельного планування, навчання персоналу та адаптації процесів під специфіку компанії, адже ефективність HRIS та HCM значною мірою залежить від правильного введення даних та дисципліни користувачів.

Переваги цифрових HR-рішень проявляються не лише у скороченні витрат і автоматизації рутинних завдань, а й у можливості побудови стратегічного підходу до управління людським капіталом, створенні аналітичної бази для прийняття управлінських рішень та підвищенні конкурентоспроможності організації на ринку. В результаті, HRIS, HCM-платформи та системи обліку персоналу стають невід'ємною частиною сучасної HR-екосистеми, забезпечуючи синергію між адміністративною ефективністю, аналітичними можливостями та стратегічним розвитком персоналу.

У сучасному HR-менеджменті ефективне прийняття рішень значною мірою залежить від правильного збору та аналізу різних типів даних про персонал, які умовно поділяють на кадрові, операційні та поведінкові. *Кадрові дані* охоплюють базову інформацію про співробітників, включаючи особисті дані, контактну інформацію, освіту, досвід роботи, професійні сертифікати та історію займаних посад. Вони формують основу для ведення кадрового обліку, формування штатного розкладу та управління документацією, а також забезпечують юридичну та адміністративну відповідність організації.

Операційні дані відображають щоденну активність співробітників і процеси, пов'язані з їхньою роботою, такі як облік робочого часу, присутність, відпустки, лікарняні, виконання поставлених завдань та участь у проектах. Аналіз цих даних дозволяє оцінювати продуктивність команди, визначати вузькі місця у процесах та оптимізувати розподіл ресурсів. *Поведінкові дані* включають інформацію про взаємодію співробітників у робочому середовищі, їхню залученість, мотивацію, комунікаційні патерни, участь у навчальних та мотиваційних програмах, а також реакції на зміни в організації. Збір і аналіз поведінкових даних дає змогу HR-фахівцям прогнозувати рівень задоволеності, ймовірність плинності кадрів та ефективність програм розвитку персоналу.

Компанії, що використовують комплексний підхід до HR-даних, отримують стратегічну перевагу. Наприклад, Google і Microsoft аналізують поведінкові патерни співробітників для підвищення ефективності командної роботи та планування навчальних програм, а Unilever інтегрує кадрові та операційні дані для оптимізації процесів рекрутингу та адаптації нових співробітників. Поєднання кадрових, операційних та поведінкових даних дозволяє створювати повноцінні People Analytics-проекти, які забезпечують комплексне бачення стану персоналу та допомагають прогнозувати потреби компанії у майбутньому.

Крім того, сучасні цифрові платформи та HCM-системи дозволяють інтегрувати всі три типи даних в єдину систему, забезпечуючи можливість глибокої аналітики, візуалізації показників та автоматизованого формування звітів для управлінців. Це не лише підвищує ефективність HR-процесів, а й дозволяє організаціям формувати стратегію управління людським капіталом на основі реальних, об'єктивних даних, зменшуючи суб'єктивність рішень та ризики людського фактора. Таким чином, систематизація та аналіз кадрових, операційних та поведінкових даних стають невід'ємною частиною сучасної HR-аналітики, забезпечуючи гнучкість, прозорість та стратегічну цінність управління персоналом у динамічному бізнес-середовищі.

У сучасному HR-менеджменті ключовим інструментом оцінки ефективності роботи персоналу та ефективності HR-процесів є система HR KPIs (Key Performance Indicators) та метрики продуктивності. HR KPIs дозволяють кількісно вимірювати досягнення стратегічних цілей організації, пов'язаних з управлінням людським капіталом, і забезпечують прозорість у прийнятті управлінських рішень.

Основні HR KPIs охоплюють різні аспекти діяльності персоналу: продуктивність співробітників, рівень залученості та мотивації, плинність кадрів, середній термін адаптації нових працівників, час закриття вакансій, ефективність програм навчання та розвитку, а також витрати на персонал. Наприклад, рівень залученості співробітників вимірюється через опитування, оцінку участі у корпоративних програмах та показники продуктивності, а плинність кадрів дозволяє виявляти проблемні підрозділи та своєчасно реагувати на ризики втрати ключових талантів. Метрики ефективності, такі як: кількість виконаних завдань на одного співробітника, середній час закриття вакансії або ROI від програм навчання, дають змогу порівнювати результати між командами та оцінювати ефективність HR-стратегій у цілому.

Компанії, що активно використовують HR KPIs, отримують можливість не лише оцінювати поточний стан, а й прогнозувати майбутні потреби у персоналі та оптимізувати кадрові ресурси. Наприклад, IBM застосовує аналітичні моделі для моніторингу KPIs, щоб прогнозувати продуктивність та визначати ризики плинності кадрів, а Google інтегрує KPIs у People Analytics для оцінки ефективності команд та планування розвитку талантів. Використання HR KPIs у поєднанні з сучасними HCM-платформами та аналітичними інструментами дозволяє автоматизувати збір даних, створювати дашборди для керівників і HR-команд та приймати рішення на основі об'єктивної інформації. Це підвищує точність оцінки результатів, зменшує суб'єктивність управлінських рішень і забезпечує стратегічне управління людським капіталом.

HR KPIs та метрики ефективності

HR KPI / Метрика	Опис	Приклад використання компаніїю
1	2	3
Рівень залученості співробітників	Вимірює мотивацію, зацікавленість та активність співробітників	Google – People Analytics для оцінки ефективності команд
Плинність кадрів	Відсоток співробітників, які залишають компанію за певний період	IBM – прогнозування плинності для планування ресурсів
Середній час закриття вакансії	Середній термін від відкриття вакансії до найму нового співробітника	Unilever – оптимізація процесу рекрутингу та адаптації
Продуктивність співробітників	Кількість виконаних завдань або досягнення ключових показників роботи	Microsoft – аналітика продуктивності команд
ROI від навчальних програм	Показник ефективності інвестицій у розвиток персоналу	SAP – оцінка ефективності тренінгів
Час адаптації нових співробітників	Період, за який новий працівник досягає повної продуктивності	Accenture – моніторинг onboarding
Рівень задоволеності персоналу	Оцінка задоволеності роботою та корпоративною культурою	Salesforce – регулярні опитування та аналітика
Витрати на персонал	Загальні HR-витрати на одного співробітника або на відділ	PepsiCo – контроль бюджету на HR
Відсутність на роботі (Absenteeism)	Відсоток днів, пропущених співробітниками через хворобу або інші причини	Deloitte – планування ресурсів
Співвідношення керівників і підлеглих	Кількість співробітників на одного менеджера	Amazon – оптимізація управлінських навантажень
Успішність рекрутингу	Відсоток кандидатів, які пройшли випробувальний термін успішно	Unilever – підвищення якості рекрутингу
Рівень розвитку талантів	Частка співробітників, які пройшли навчання або підвищили кваліфікацію	IBM – оцінка ефективності навчальних програм
Відсоток внутрішніх переведень	Частка співробітників, які просунулися або перевелися на іншу позицію внутрішньо	GE – розвиток кар'єрного потенціалу
Кількість конфліктів і інцидентів	Показник робочих конфліктів або дисциплінарних проблем	Siemens – управління корпоративною культурою
Баланс роботи та життя (Work-Life Balance)	Оцінка задоволеності співробітників балансом між роботою та особистим життям	Microsoft – підвищення мотивації через опитування

1	2	3
Співвідношення план/факт навантаження	Вимірює відповідність робочих завдань запланованим нормам	IBM – оцінка ефективності планування проєктів
Ефективність програм мотивації	Вимірює, наскільки бонуси та винагороди впливають на продуктивність	Salesforce – аналіз впливу бонусів на залученість
Час відповіді на запити персоналу	Середній час реакції HR на запити співробітників	Accenture – оптимізація HR-процесів
Рівень використання HR-платформ	Частка співробітників, які активно користуються HR-системами	Workday – моніторинг цифрової взаємодії персоналу
Відсоток завершених навчальних програм	Частка співробітників, які успішно закінчили тренінги та курси	SAP – оцінка ефективності навчання
Рівень залученості у корпоративні програми	Вимірює участь співробітників у тренінгах, командуванні та проєктах	Google – People Analytics для командного розвитку
Середній час обробки кадрових запитів	Вимірює оперативність HR у вирішенні адміністративних завдань	PepsiCo – автоматизація HR-процесів
Вартість залучення одного співробітника (Cost per Hire)	Витрати на рекрутинг, включно з рекламними кампаніями та співбесідою	Unilever – оптимізація процесу найму
Рівень підготовки до заміщення ключових позицій	Частка співробітників, готових замінити критичні ролі в організації	IBM – планування наступництва та управління талантами
Відсоток співробітників з високим потенціалом	Частка персоналу, ідентифікованого як стратегічний талант	GE – People Analytics для розвитку лідерів

Впровадження HR KPIs дозволяє організаціям формувати прозору систему оцінки персоналу, оптимізувати HR-процеси, підвищувати продуктивність і залученість співробітників, а також створювати довгострокову стратегічну цінність людського капіталу в умовах динамічного ринку праці. Таким чином, HR KPIs та метрики ефективності стають не просто інструментом контролю, а важливим елементом стратегічного управління персоналом, що забезпечує конкурентні переваги організації та підтримує її адаптивність у сучасному бізнес-середовищі.

7.3.2. Аналітика управління талантами.

Аналітика управління талантами на сьогодні перетворилася на один із ключових стратегічних інструментів сучасних організацій, оскільки вона дозволяє глибоко, багатовимірно й системно оцінювати всі етапи взаємодії компанії зі своїм людським капіталом, від найму та адаптації до розвитку, продуктивності й утримання співробітників. Її сутність полягає у використанні великих масивів HR-даних, інтегрованих із бізнес-показниками, щоб з високою точністю прогнозувати кадрові ризики, виявляти приховані закономірності у поведінці працівників, оптимізувати витрати на персонал та підсилювати управлінські рішення на основі доказів, а не інтуїції.

Сучасні компанії застосовують інструменти продвинутої аналітики, такі як: машинне навчання, предиктивне моделювання, аналіз соціальних зв'язків, багатфакторну оцінку потенціалу, щоб визначити, які компетенції є критичними для ключових ролей, хто з працівників має найбільший лідерський потенціал, які фактори найбільшою мірою впливають на плинність кадрів, продуктивність або залученість. Наприклад, глобальні корпорації на кшталт Google, IBM чи Unilever будують комплексні системи Talent Intelligence, які дозволяють автоматично ранжувати кандидатів за культурною відповідністю, прогнозувати ризик вигорання або визначати індивідуальні траєкторії розвитку з урахуванням поведінкових даних та історії результативності.

Аналітика талантів також підтримує стратегічне планування робочої сили: вона допомагає моделювати майбутні потреби у навичках, оцінювати вплив автоматизації, планувати заміну критичних ролей і формувати резерв лідерів на основі об'єктивних даних. У ширшому контексті аналітика талантів створює культуру прозорості й підзвітності в HR, перетворюючи його з операційної функції на центр стратегічної експертизи, що безпосередньо впливає на конкурентоспроможність бізнесу. Результатом стає більш точне управління людським капіталом, мінімізація ризиків, підвищення ефективності

команд і створення довгострокової цінності для організації, де дані слугують основою для прийняття кожного важливого рішення щодо людей.

Моделі оцінки компетенцій становлять систематизовану структуру, яка дозволяє визначати, вимірювати та порівнювати професійні й поведінкові характеристики працівників із вимогами конкретної ролі та стратегічними потребами організації. У сучасному HR-менеджменті компетенції розглядаються не лише як сукупність навичок, а як інтегральне поєднання знань, поведінкових патернів, когнітивних здібностей, мотивацій та ціннісних орієнтирів, які разом формують здатність людини ефективно виконувати робочі завдання. Моделі оцінки компетенцій забезпечують прозорий та об'єктивний підхід до управління персоналом: вони дозволяють мінімізувати суб'єктивізм керівників, стандартизувати вимоги до різних функцій, побудувати єдину мовну систему опису ролей та очікувань, а також закласти фундамент для подальших HR-практик, від рекрутингу до succession planning. Для компаній це стає інструментом вирівнювання людського потенціалу зі стратегією організації, забезпечення якості прийняття рішень та створення передумов для ефективного розвитку талантів, що особливо важливо в умовах високої конкуренції за кваліфіковану робочу силу.

Однією з найпоширеніших моделей оцінки є поведінкові компетенції, які фокусуються на тому, як працівник поводить себе у професійних ситуаціях, приймає рішення, взаємодіє з колегами, вирішує конфлікти чи ініціює зміни. На відміну від технічних компетенцій, що часто мають чітку, вимірювану природу, поведінкові залежать від культурного контексту компанії, особистісних особливостей і динаміки командної роботи. Сучасні організації використовують такі компетенції як “критичне мислення”, “співпраця”, “орієнтація на результат”, “емоційний інтелект”, “адаптивність”, “інноваційність”, формуючи для кожної чіткі індикатори поведінки, які дозволяють об'єктивно фіксувати прояви на різних рівнях, від базового до експертного.

Наприклад, компетенція “комунікації” може включати здатність структурувати повідомлення, переконливо доносити складну інформацію, адаптувати стиль до різних аудиторій, демонструвати активне слухання та будувати психологічно комфортний діалог. Поведінкові моделі дозволяють проводити оцінку через інтерв’ю за методом STAR, структуровані тести, ассесмент-центри та 360-градусний feedback, забезпечуючи глибокий погляд на реальні патерни поведінки співробітника у робочих умовах.

Інший важливий напрям – це моделі технічних або функціональних компетенцій, які описують професійні знання та навички, необхідні для успішного виконання специфічних завдань у певній галузі: фінансах, ІТ, маркетингу, виробництві, логістиці тощо. Ці моделі, як правило, мають більш формалізований характер, оскільки спираються на стандарти професії, сертифікації або внутрішні регламенти компанії. Наприклад, технічні компетенції аналітика даних можуть включати володіння мовами програмування (Python, R), уміння працювати з SQL, знання статистики, досвід побудови моделей машинного навчання та навички роботи з BI-платформами.

Для інженера – це може бути знання CAD-систем, принципів безпеки, технології обробки матеріалів; для фінансиста – це знання IFRS, моделювання DCF, аналізу ризиків. Оцінювання технічних компетенцій проводиться через тестування, практичні завдання, іспити, портфоліо або моделювання робочих ситуацій. Для організацій важливо, що саме технічні компетенції часто є прямим драйвером продуктивності та якості роботи, тому їх точна оцінка критично впливає на кадрові рішення, формування системи навчання та визначення рівня кваліфікації співробітника.

Сучасні моделі оцінки компетенцій дедалі частіше використовують аналітичні та цифрові підходи, інтегруючи великі дані, машинне навчання та поведінкову аналітику. Це дозволяє створювати динамічні, адаптивні моделі, які змінюються разом із розвитком ринку, технологій та корпоративної стратегії. Компанії впроваджують платформні рішення, що автоматично

аналізують дані з HRIS, LMS, систем продуктивності, оцінок 360°, внутрішніх комунікацій та навіть корпоративних соціальних мереж, щоб формувати індивідуальні профілі компетенцій.

Такі системи можуть прогнозувати, які навички будуть критичними для працівника через 2-3 роки, виявляти прогалини у компетенціях, оцінювати ризик невідповідності ролі або потенціал до швидкого розвитку. Деякі платформи застосовують Natural Language Processing для аналізу текстів job descriptions, CV та робочої документації з метою автоматичного картування компетенцій і визначення рівня складності завдань. Це значно прискорює HR-процеси, підвищує точність оцінки й зменшує людський фактор, створюючи більш прозору та надійну систему управління талантами.

У стратегічному вимірі моделі оцінки компетенцій відіграють ключову роль у побудові взаємопов'язаної системи розвитку персоналу, оскільки вони лежать в основі навчання, планування кар'єри, формування кадрового резерву та побудови індивідуальних траєкторій розвитку. Вони виступають "єдиною мовою" між HR-функцією, керівниками та співробітниками, забезпечуючи узгодженість очікувань та зрозумілість критеріїв оцінки. На рівні організації це дає можливість створювати матриці компетенцій, будувати моделі потенціалу, прогнозувати готовність співробітників до переходу на вищі позиції та формувати керівні резерви на основі об'єктивних даних.

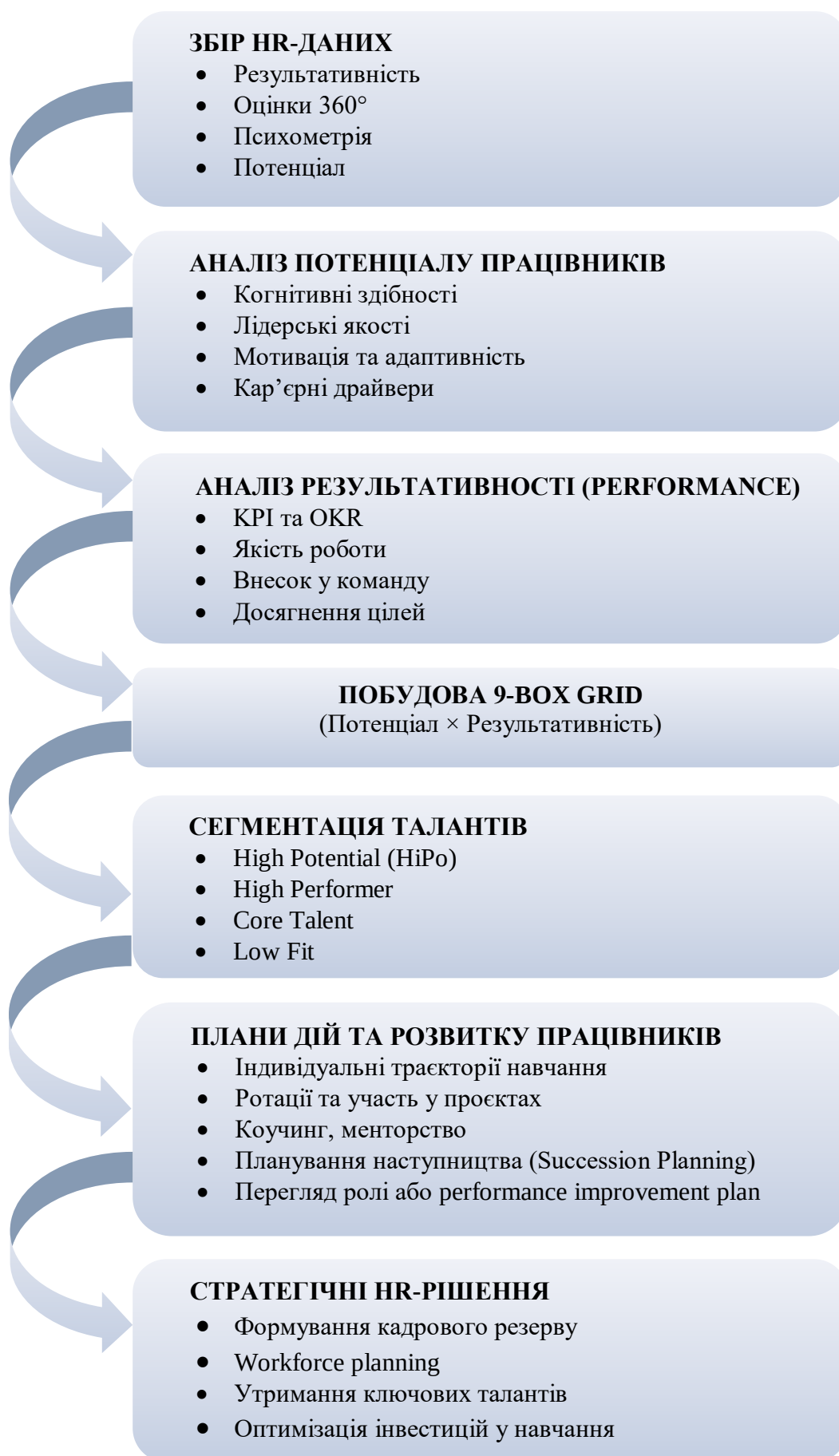
У довгостроковій перспективі правильно побудована модель компетенцій стає одним із найпотужніших інструментів управління організаційною ефективністю, оскільки дозволяє розвивати саме ті навички та поведінкові якості, які забезпечують конкурентні переваги бізнесу. Таким чином, моделі оцінки компетенцій виступають не просто інструментом вимірювання, а фундаментом стратегічного HR-менеджменту та невід'ємним елементом культури високих стандартів і постійного професійного розвитку.

Концепція оцінювання потенціалу працівників за допомогою 9-box grid стала одним із найбільш поширених інструментів стратегічного HR-

менеджменту, оскільки вона дозволяє об'єднати в єдиній системі два ключові виміри, а саме: поточну результативність співробітника та його майбутній потенціал до зростання. Матриця 9-box – це тривимірна сітка з дев'яти комірок, де по горизонталі зазвичай розташовується рівень продуктивності (низький, середній, високий), а по вертикалі – це потенціал (низький, середній, високий). Така структура дає організації змогу швидко сегментувати таланти, визначати співробітників, які готові до кар'єрного ривка, хто потребує розвитку, а хто виконує свої обов'язки стабільно, але не демонструє перспектив до подальшого росту. Для компаній це дає стратегічну перевагу: рішення щодо навчання, ротації, підвищень, формування кадрового резерву та утримання ключових фахівців приймаються на основі систематизованих, узгоджених між керівниками оцінок, що значно зменшує суб'єктивність та підсилює точність управлінських рішень.

Однією з ключових переваг 9-box grid є можливість виявляти high-potentials (HiPo) співробітників, які не лише демонструють високу ефективність сьогодні, а й мають значний лідерський або експертний потенціал на майбутнє. У сучасних організаціях HiPo-працівники становлять критичну групу, оскільки саме вони здатні забезпечити розвиток бізнесу, впроваджувати інновації та брати на себе складні стратегічні ролі. Матриця дозволяє чітко ідентифікувати цих співробітників та сформувати індивідуальні програми їхнього розвитку, що включають менторство, участь у складних проєктах, лідерські тренінги, ротації у різних підрозділах або включення до “succession planning”. Компанії на кшталт Microsoft, Amazon чи Deloitte активно використовують 9-box grid у процесі Talent Review, де керівники узгоджують оцінки потенціалу та продуктивності кожного ключового працівника, формуючи глибоке розуміння кадрової ситуації в організації та забезпечуючи прозорий стратегічний погляд на розвиток талантів.

Модель 9-box grid



У рамках методики 9-box grid велика увага приділяється критеріям оцінювання потенціалу, оскільки саме вони визначають якість і точність розподілу співробітників у матриці. Потенціал зазвичай оцінюється через комплекс факторів: когнітивні здібності (аналітичне мислення, швидкість навчання, інтелектуальна гнучкість), лідерські якості (вміння впливати, стратегічне бачення, управління командою), особистісні характеристики (стійкість до стресу, ініціативність, адаптивність), мотиваційні драйвери та культурна відповідність. Для підвищення об'єктивності компанії застосовують структуровані інструменти: оцінки 360°, інтерв'ю про потенціал (Potential Interview), психометричні тести, аналіз поведінкових патернів та рекомендації керівників. Усе це дозволяє оцінити не просто те, як співробітник працює зараз, а його здатність брати на себе більш складні завдання, приймати рішення в умовах невизначеності, розвивати інновації та працювати на стратегічному рівні. Таким чином, матриця не лише класифікує, а й розкриває глибинні характеристики, які визначають майбутній успіх у компанії.

Одним із найцінніших результатів використання 9-box grid є можливість формування персоналізованих карт розвитку для кожної категорії співробітників. Працівники, що потрапляють у верхній правий квадрант (висока ефективність + високий потенціал), зазвичай отримують пріоритетний доступ до розвитку та майбутніх лідерських ролей. Ті, хто має високу ефективність, але низький потенціал, розглядаються як надійні експерти та “стабільні зірки”. Вони отримують поглиблене навчання у своїй сфері та стабільні кар’єрні траєкторії. Співробітники з низькою ефективністю, але високим потенціалом часто стають об’єктом інтенсивних програм навчання, менторства й коучингу, адже вони можуть швидко вийти на новий рівень за умови правильного управлінського супроводу. Нарешті, група з низькими показниками за обома критеріями потребує чіткого плану покращення або перегляду ролі, тоді матриця дає змогу аргументовано обґрунтувати такі

рішення. Таким чином, 9-box grid перетворюється на інструмент, який не лише оцінює таланти, а й спрямовує розвиток кожного працівника.

У стратегічному вимірі методика 9-box grid допомагає створити прозору, передбачувану та системну модель управління талантами, що охоплює весь цикл, від виявлення перспективних працівників до планування їхньої ролі у майбутньому розвитку компанії. Завдяки матриці керівники отримують повну картину кадрового портфеля, де є ризики втратити ключових людей, які компетенції потрібно розвивати, які ролі у майбутньому залишаться без наступників, як змінюється баланс талантів у різних підрозділах. Це суттєво посилює можливості стратегічного “workforce planning”, впровадження кадрових резервів, управління плинністю та забезпечення організаційної стійкості. Компанії можуть точніше прогнозувати кадрові потреби, оптимізувати інвестиції в навчання, будувати сильні лідерські вертикалі та підвищувати довгострокову ефективність бізнесу. У результаті 9-box grid перетворюється на не просто інструмент HR-аналітики, а на фундаментальний елемент управління людським капіталом, що дозволяє компанії мислити стратегічно, діяти системно та формувати культуру рішень, побудованих на даних і потенціалі.

Планування навчання на основі даних формується як цілісна система, що поєднує стратегічні цілі організації, дані про поточні компетенції персоналу та прогноз майбутніх навичок, необхідних компанії для конкурентоспроможності. На відміну від традиційного підходу, де рішення про навчання приймалися на основі суб'єктивних оцінок керівників або загальних корпоративних трендів, data-driven learning development спирається на кількісні метрики, які точно показують, які прогалини в компетенціях існують і якими методами їх найефективніше усунути. Сюди входять результати оцінки компетенцій, 9-box grid, продуктивність, проєктні KPI, результати опитувань, дані з LMS, показники залученості та дані про ефективність попередніх навчальних

програм. Такий підхід дозволяє побудувати навчальну екосистему, яка підтримує стійкий розвиток талантів, а не лише реагує на окремі проблеми.

Ключовим етапом планування навчання є точна діагностика skill gaps, і саме аналітичні інструменти HRIS, LMS та Talent Management System дають змогу робити це у структурованій формі. Організації аналізують продуктивність працівників у динаміці, результати оцінювання компетенцій, профілі майбутніх ролей і вимоги до роботи, щоб визначити, які навички стають критичними. Багатовимірні моделі, такі як skill matrices та heatmaps компетенцій, дозволяють візуалізувати розриви по підрозділах та посадах. Наприклад, якщо аналітика показує, що команда продажу має високі результати, але недостатньо розвинені навички SPIN-продажів або роботи з CRM, навчання можна таргетувати саме на ці сфери. Використання прогностичної аналітики дає змогу не лише виявляти прогалини, а й передбачати майбутні потреби. Наприклад, зростання важливості data literacy, cybersecurity awareness або AI-компетенцій.

Однією з найважливіших переваг підходу data-driven є можливість формувати індивідуальні навчальні траєкторії для кожного співробітника. На відміну від масових програм, сучасні системи здатні аналізувати дані про поведінку в LMS, рівень завершення курсів, типи контенту, які співробітник обирає найчастіше, швидкість проходження модулів, а також результати тестів і практичних завдань. Ці дані дозволяють алгоритмам рекомендувати оптимальні формати навчання, а саме: відеокурси, мікролернінг, практичні воркшопи, коучинг або програму наставництва. Наприклад, якщо співробітник швидко засвоює технічні навички, але повільніше розвиває управлінські, система може запропонувати серію фасилітаційних сесій або симуляції прийняття рішень. Таким чином, персоналізація ґрунтується не на припущеннях, а на реальній поведінці та вимірюваних результатах.

Планування навчання на основі даних неможливе без оцінки його впливу на продуктивність бізнесу. У цьому контексті використовуються різні моделі:

Kirkpatrick, ROI Learning Model, CIPP, Phillips ROI тощо. Дані з LMS інтегруються з HR KPI, а саме: плинність кадрів, час адаптації новачків, підвищення результативності, ефективність продажів, показники помилок, швидкість виконання операцій. Аналітика допомагає визначити причинно-наслідкові зв'язки: чи призвело навчання до реального покращення роботи, чи покращення було спричинене зовнішніми факторами. Наприклад, якщо після програми з лідерства у 30% нових менеджерів зменшився рівень конфліктності в командах і зріс рівень виконання OKR, аналітична модель може підтвердити прямий вплив програми на організаційні результати. Таке системне вимірювання дозволяє приймати рішення щодо масштабування або перегляду навчальних ініціатив.

Data-driven Learning Strategy інтегрується з управлінням талантами, плануванням наступництва та workforce planning. Компанія використовує прогностичні моделі для розуміння, які навички стануть ключовими в найближчі 2-3 роки, і формує навчальний портфель під ці потреби. Наприклад, якщо аналіз ринку та внутрішніх даних вказує на зростання ролі автоматизації, організація може заздалегідь інвестувати в reskilling співробітників, замість найму нових спеціалістів на конкурентному ринку. Таке стратегічне планування забезпечує не лише ефективність навчальних програм, а й стійкість компанії до змін, включно з трансформаціями бізнес-моделі, впровадженням цифрових технологій та виходом на нові ринки. У результаті навчання перестає бути «витратною статтею» і стає драйвером стратегічного розвитку та конкурентної переваги.

7.3.3. Прогнозування HR-показників.

Прогнозування плинності персоналу є однією з центральних задач сучасного HR-аналітичного підходу, адже дозволяє не лише розуміти причини звільнень постфактум, а й заздалегідь передбачати ризики втрати цінних працівників. У глобальній практиці аналітика плинності розглядається як інструмент стратегічного управління талантами, який дає змогу зменшити

витрати на рекрутинг і навчання, уникнути провалів у роботі команд та підвищити організаційну стійкість. Компанії використовують великий масив HR-даних: кадрову інформацію, результати оцінок, анкетування залученості, історію кар'єри, зміну ролей, дані перформансу, участь у навчальних програмах, рівень компенсацій та навіть поведінкові сигнали в системах співпраці. Таке поєднання створює комплексну картину ризиків та дозволяє встановити закономірності, які неможливо помітити без аналітики. Таким чином, прогнозування плинності виходить за межі простого раху-нку звільнень, перетворюючись на інструмент стратегічного foresight управління персоналом.

Моделі прогнозування плинності використовують десятки факторів, які можуть впливати на рішення працівника залишити компанію. До найбільш значущих предикторів належать: зміни в продуктивності, низька участь у корпоративних активностях, падіння залученості, часті зміни керівників, відсутність кар'єрних переходів, затримки в підвищеннях, дисбаланс оплати праці, надмірне навантаження або недостатня кількість навчальних можливостей. Поведінкові дані можуть включати тривалий час відповіді у внутрішніх системах, різке зростання лікарняних або зниження співвідношення завершених до запланованих задач. Деякі компанії використовують аналіз соціальних мереж співробітників (organizational network analysis), що дозволяє оцінити ступінь інтегрованості людини в команду. Такі дані створюють багатофакторний профіль ризику, що стає основою для машинних моделей прогнозування. Чим багатшими є дані, тим точніший прогноз, оскільки модель здатна виявляти складні нелінійні взаємозв'язки.

Для прогнозування ризику звільнень застосовуються різні математичні та машинні підходи, серед яких логістична регресія, дерева рішень, random forest, gradient boosting, XGBoost, нейронні мережі та моделі виживаності (survival analysis, Cox regression). Логістична регресія дозволяє отримати інтерпретовані результати й чітко зрозуміти, які фактори мають найбільший вплив. Моделі ensemble-класу забезпечують високу точність завдяки комбінуванню різних

наборів дерев рішень та виявленню складних патернів у даних. Survival analysis є особливо корисною, коли важливим є не просто факт можливого звільнення, а й час до звільнення, що дає можливість HR-функції планувати точні інтервенції. У великих компаніях моделі можуть працювати в реальному часі — щоденно оновлюючи ризики для кожного працівника. Точність таких підходів часто сягає 75-90%, якщо база даних якісна та достатньо велика, що дозволяє впроваджувати превентивні заходи замість реактивних.

Побудова прогнозової моделі — це лише початок процесу; ключовим є розуміння її результатів та перетворення прогнозів на практичні HR-рішення. Аналітики створюють risk-scoring моделі, які класифікують працівників за рівнем ризику: низький, середній, високий. Для кожної групи формується набір інтервенцій: розвиткові програми, коучинг, коригування навантаження, індивідуальні плани кар'єрного зростання, участь у стратегічних проектах або компенсаційний перегляд. Якщо модель показує, що ризик зумовлений недостатніми можливостями розвитку, компанія може запропонувати більше навчальних ресурсів чи внутрішню ротацію. Якщо фактором є перевантаження, керівники можуть оптимізувати workload або перерозподілити задачі. У результаті аналітика переноситься в реальні управлінські дії, які допомагають зберігати ключових співробітників та зменшувати витрати на заміну кадрів. Правильна інтерпретація даних стає каталізатором стратегічного управління талантами.

У довгостроковій перспективі прогнозування плинності не лише дозволяє зменшити відтік кадрів, а й стає інструментом стратегічного планування людського капіталу. За допомогою моделей прогнозу організація може визначати підрозділи з високими структурними ризиками, оновлювати політику оплати праці, покращувати якість лідерства та будувати більш привабливий внутрішній бренд роботодавця. Прогнозні моделі дозволяють оцінювати ефективність змін: чи зменшився ризик після впровадження навчальної програми, зміни керівника або оновлення компенсаційної системи.

Також вони допомагають будувати сценарні моделі, які показують, як зміна умов. Наприклад, перехід на гібридну роботу або скорочення навантаження вплине на рівень утримання персоналу. У результаті прогнозування плинності стає ключовим елементом HR-стратегії, що поєднує аналітику, управління талантами та організаційний розвиток, формуючи передбачувану, стабільну та гнучку робочу силу.

Аналіз факторів ризику вигорання є критично важливим елементом сучасного управління людським капіталом, оскільки вигорання негативно впливає на продуктивність, залученість і загальну ефективність команди. Систематичне виявлення ризиків допомагає організаціям попереджати критичні ситуації, які можуть призвести до емоційного та фізичного виснаження працівників. Аналіз починається з оцінки робочих умов, навантаження, взаємодії з керівництвом та колегами, а також із вивчення особистісних характеристик співробітників, що можуть підвищувати ймовірність вигорання. Це дозволяє створювати профілі ризику для різних груп персоналу та планувати превентивні заходи.

Одним із ключових підходів є використання HR-аналітики та психометричних інструментів для збору даних про рівень стресу, задоволеність роботою, баланс між роботою та особистим життям та інші психологічні показники. Анкетування, опитування «pulse survey» і регулярні бесіди з керівниками допомагають виявити ранні сигнали емоційного виснаження. Наприклад, зниження продуктивності, підвищена кількість лікарняних або часті конфлікти в команді можуть сигналізувати про зростання ризику вигорання. Використання таких даних у моделі прогнозування дозволяє організаціям своєчасно реагувати на проблеми.

Структурний підхід до аналізу ризиків включає ідентифікацію як організаційних, так і індивідуальних факторів. До організаційних відносяться перевантаження завданнями, неясність ролей, недостатня підтримка керівництва, низька автономія та відсутність можливостей для розвитку.

Індивідуальні фактори включають особистісні риси, рівень стресостійкості, перфекціонізм та здатність до саморегуляції. Аналіз взаємодії цих факторів дозволяє формувати комплексну карту ризиків і виявляти «гарячі точки» у структурі компанії, де необхідні превентивні заходи. Сучасні моделі аналізу вигорання часто використовують алгоритми машинного навчання та регресійні моделі для прогнозування ризику на основі історичних і поточних даних. Наприклад, алгоритми можуть аналізувати співвідношення робочих годин, участі в проєктах, частоти відпусток і результатів опитувань, щоб визначити ймовірність вигорання для конкретного працівника або команди. Інтеграція таких моделей у HR-платформи та дашборди дозволяє керівникам оперативно відслідковувати ризики та приймати цільові заходи для підтримки персоналу.

Аналіз факторів ризику вигорання

Збір даних

- HRIS: робочі години, участь у проєктах
- Опитування «pulse survey»: задоволеність роботою, баланс життя
- Лікарняні, конфлікти, результати оцінок продуктивності

Аналіз даних

- Статистичні моделі: регресія, факторний аналіз
- Машинне навчання: класифікація ризику вигорання
- Сегментація: команди, ролі, департаменти

Сигнали ризику

- Підвищена кількість лікарняних
- Зниження продуктивності
- Низька залученість та мотивація
- Часті конфлікти

Превентивні дії

- Індивідуальні плани розвитку
- Програми менторства та навчання
- Збалансоване навантаження, гнучкий графік
- Підтримка корпоративної культури

KPI та моніторинг

- Індекс благополуччя персоналу
- Текучість кадрів
- Рівень залученості та продуктивності
- Відстеження ефективності превентивних заходів

На стратегічному рівні аналіз факторів вигорання стає основою для формування політик управління благополуччям працівників, програм менторства, навчання та розвитку, а також стратегій роботи з корпоративною культурою. Вчасне виявлення ризиків дозволяє зменшити текучість кадрів, підвищити залученість і продуктивність, а також підтримувати позитивний психологічний клімат у компанії. Наприклад, компанії, такі як: Microsoft, SAP та Google впроваджують комплексні системи моніторингу стресу та благополуччя, що дозволяє поєднувати дані аналітики з практичними заходами для запобігання вигорання.

7.3.4. HR-аналітика в процесах найму та адаптації.

HR-аналітика відіграє ключову роль у процесах найму, дозволяючи компаніям приймати обґрунтовані рішення на основі даних замість інтуїції. Використання аналітичних інструментів дозволяє оцінювати ефективність джерел кандидатів, прогнозувати їх відповідність посаді та визначати оптимальні стратегії залучення талантів. Систематичний збір та аналіз даних, таких як резюме, онлайн-профілі, результати тестів і попередніх оцінок продуктивності, допомагає відсіювати менш відповідних кандидатів і фокусуватися на тих, хто має найвищий потенціал для розвитку в компанії. Це значно скорочує час найму та підвищує його якість.

Аналітика в адаптаційних процесах дозволяє оцінювати ефективність онбордингу та швидкість інтеграції нових співробітників. Використання даних із систем HRIS, опитувань «pulse survey» і зворотного зв'язку від наставників дозволяє відстежувати прогрес новачків, визначати складнощі на ранніх етапах та вчасно реагувати на потенційні проблеми. Наприклад, якщо новий співробітник не досягає ключових показників ефективності у перший місяць, HR-команда може застосувати індивідуальний план підтримки, додаткові навчальні модулі або наставництво.

Аналітика допомагає також прогнозувати ймовірність плинності серед новачків. На основі історичних даних про адаптацію попередніх працівників

можна визначити, які фактори впливають на успішну інтеграцію, а які підвищують ризик звільнення. Показники, такі як тривалість проходження навчальних програм, участь у командних заходах, рівень задоволеності роботою та частота звернень до HR, дозволяють створювати прогнози та превентивні стратегії утримання. Це сприяє стабільності команди та економії ресурсів компанії.

Важливою складовою є аналіз ефективності процесів найму та адаптації з точки зору ROI (окупності інвестицій у персонал). HR-аналітика дозволяє оцінити, наскільки інвестиції в рекрутинг, навчання та онбординг окупаються продуктивністю та залученістю співробітників. Наприклад, аналіз даних може показати, що певні канали залучення кандидатів приносять більше працівників, які залишаються у компанії довше та демонструють високі результати. Це дозволяє оптимізувати бюджет HR та покращувати якість підбору персоналу.

На стратегічному рівні HR-аналітика стає основою для побудови системи управління талантами. Дані з процесів найму та адаптації дозволяють створювати довгострокові плани розвитку, визначати критичні компетенції та формувати програми навчання і наставництва. Це забезпечує ефективне управління талантами, підвищує залученість і мотивацію співробітників та створює стійку кадрову платформу для досягнення стратегічних цілей компанії. Використання аналітики перетворює HR-процеси з рутинних операцій на стратегічний інструмент розвитку організації.

Скринінг кандидатів на основі даних стає невід'ємною частиною сучасного HR-процесу, оскільки дозволяє організаціям підвищувати ефективність найму та зменшувати ризики помилкових рішень. Традиційний підхід до відбору, заснований на резюме та інтерв'ю, часто суб'єктивний і не дає повного розуміння потенціалу кандидатів. Використання аналітичних інструментів дозволяє автоматично оцінювати навички, досвід і поведінкові характеристики кандидатів, надаючи HR-командам об'єктивну основу для

прийняття рішень. Завдяки цьому компанії можуть швидше і точніше ідентифікувати найбільш підходящих кандидатів для конкретних ролей.

Скринінг кандидатів на основі даних включає:

1. Збір даних

- Резюме та CV
- Онлайн-профілі (LinkedIn, GitHub тощо)
- Результати тестувань та опитувань
- Історія досягнень і рекомендації

2. Аналіз та оцінка

- Алгоритми машинного навчання для ранжування кандидатів
- Оцінка ключових компетенцій та потенціалу
- Профілі за демографічними та освітніми показниками

3. Поведенкові та когнітивні оцінки

- Психологічні тести
- Симуляційні завдання / кейси
- Оцінка стресостійкості, комунікативних та лідерських якостей

4. Сигнали ризику та попередній відбір

- Низький рівень відповідності ролі
- Відсутність необхідних компетенцій
- Потенційні проблеми адаптації

5. Прийняття рішень і KPI

- Кандидати, рекомендовані для інтерв'ю
- Час закриття вакансії
- Джерела найкращих кандидатів
- Ефективність процесу підбору.

Скринінг на основі даних включає збір і аналіз різноманітних джерел інформації про кандидатів. Сюди входять резюме, онлайн-профілі, результати тестувань та опитувань, а також історія попередніх досягнень. Сучасні HR-платформи інтегрують ці дані та застосовують алгоритми машинного навчання

для ранжування кандидатів за ключовими компетенціями та ймовірністю успіху на конкретній посаді. Наприклад, платформи як LinkedIn Talent Insights або HireVue використовують аналітику, щоб оцінити рівень професійної відповідності, комунікативні здібності та потенціал до розвитку.

Важливою складовою є оцінка поведінкових та когнітивних характеристик кандидата. Використання психологічних тестів, симуляційних завдань та кейсів дозволяє отримати дані про мислення, стресостійкість, здатність до командної роботи та лідерські якості. Аналітичні моделі поєднують ці дані з демографічними та освітніми показниками, щоб створити комплексний профіль кандидата. Такий підхід зменшує ризик суб'єктивної оцінки і дозволяє приймати рішення на основі об'єктивних метрик, а не тільки інтуїції рекрутера.

Інтеграція скринінгу на основі даних з системами управління талантами (HCM/HRIS) забезпечує збереження інформації про кандидатів і полегшує подальший моніторинг. Організації можуть створювати бази даних потенційних кандидатів, аналізувати тенденції найму та визначати ефективність джерел підбору персоналу. Наприклад, аналітика може показати, які канали рекрутингу приносять найбільш кваліфікованих кандидатів, а які менш успішні, що дозволяє оптимізувати витрати на пошук та підбір персоналу.

На стратегічному рівні скринінг кандидатів на основі даних стає основою для побудови довгострокової кадрової стратегії та управління талантами. Організації можуть прогнозувати потребу у нових компетенціях, ідентифікувати ключові ролі, де найм критичний, і планувати програми навчання для підвищення кваліфікації. Крім того, використання даних дозволяє підтримувати принципи об'єктивності та недискримінації, зменшуючи ризик упередженого підбору та підвищуючи рівень прозорості процесу. В результаті компанія отримує більш стабільну, мотивовану та ефективну команду, готову до досягнення стратегічних цілей.

NLP-аналіз резюме (Natural Language Processing) є одним із найбільш інноваційних підходів у сучасному рекрутингу, що дозволяє автоматизувати оцінку та сортування великої кількості кандидатів. Традиційний відбір резюме часто повільний і суб'єктивний, тоді як NLP використовує алгоритми обробки природної мови для виділення ключових навичок, досвіду, компетенцій та інших релевантних характеристик кандидата. Це дозволяє створити структуровані дані з неструктурованих документів, таких як PDF або DOCX, і швидко порівнювати кандидатів за необхідними критеріями, що значно підвищує ефективність підбору.

Основою NLP-аналізу є токенізація та семантичне розпізнавання тексту, що дозволяє алгоритму виділити ключові слова, фрази та контекстні зв'язки. Наприклад, система може розпізнати різні способи написання однієї і тієї ж компетенції («Project Management», «управління проектами») і об'єднати їх в один показник. Крім того, алгоритми можуть класифікувати досвід роботи за галузями, рівнем відповідальності або технічними навичками, що дозволяє HR-командам створювати рейтинг кандидатів та визначати тих, хто найбільше відповідає відкритій позиції.

Важливим аспектом NLP-аналізу є оцінка якості тексту та релевантності інформації. Алгоритми можуть виявляти ключові досягнення, визначати рівень професійних компетенцій і навіть оцінювати комунікативні навички на основі стилю та структури тексту. Наприклад, резюме, де чітко вказані результати проектів, ключові метрики і конкретні досягнення, буде оцінено вище, ніж резюме з абстрактними описами. Це дозволяє рекрутерам швидше фокусуватися на найбільш перспективних кандидатах і зменшувати ймовірність упередженого відбору.

Сучасні платформи рекрутингу інтегрують NLP-аналіз із системами ATS (Applicant Tracking System) та HRIS, що дозволяє автоматично оновлювати базу кандидатів, зберігати аналітичні дані та будувати звіти. Наприклад, алгоритми можуть створювати статистику за джерелами резюме, галузями

досвіду, ключовими навичками та іншими параметрами. Це дозволяє компанії не лише оперативно відбирати кандидатів, а й аналізувати ефективність каналів рекрутингу та оптимізувати процес підбору.

На стратегічному рівні NLP-аналіз резюме дозволяє підвищити якість управління талантами та скоротити час закриття вакансій. Дані, отримані з NLP, можна використовувати для прогнозування успішності кандидатів, формування пулу перспективних фахівців і створення довгострокових програм розвитку персоналу. Крім того, цей підхід забезпечує об'єктивність та прозорість процесу підбору, що сприяє недопущенню упередженості та дотриманню корпоративних стандартів найму. У результаті компанія отримує ефективний, структурований та прогнозований процес підбору талантів.

Аналітика адаптації нових працівників є критично важливим інструментом для забезпечення швидкої інтеграції співробітників у компанію та підвищення їх продуктивності. Збір даних на етапі онбордингу дозволяє оцінити, як швидко новачки опановують внутрішні процеси, знайомляться з командою та засвоюють необхідні навички. HR-аналітика допомагає ідентифікувати проблемні зони, наприклад, низьку активність у тренінгах або недостатню взаємодію з наставником, що дозволяє своєчасно вживати заходів для покращення адаптації.

Аналітика адаптації нових працівників включає:

1. Збір даних

- HRIS, база співробітників
- Результати опитувань і тестів
- Зворотний зв'язок від наставників

2. Аналіз прогресу

- Відстеження освоєння навичок та знань
- Активність у тренінгах та проектах
- Взаємодія з командою та наставником

3. Оцінка ефективності

- Досягнення ключових показників продуктивності
- Рівень залученості та задоволеності
- Частота звернень до HR та наставників

4. Сигнали ризику

- Низька активність у програмах навчання
- Відсутність інтеграції з командою
- Потенційні ризики плинності

5. Рекомендації та дії

- Індивідуальні плани підтримки та навчання
- Наставництво та коучинг
- Оптимізація адаптаційних програм.

Аналітичні системи використовують різні джерела даних, включаючи HRIS, результати опитувань, тестування знань та відгуки наставників. Ці дані дозволяють створювати індивідуальні профілі адаптації для кожного співробітника, відстежувати прогрес у реальному часі та визначати співробітників, які потребують додаткової підтримки. Наприклад, низька участь у навчальних програмах або запізнення з виконанням завдань може сигналізувати про потребу у додаткових ресурсах або консультаціях з HR.

Важливим елементом є оцінка ефективності адаптаційних програм на основі ключових метрик. HR-аналітика дозволяє вимірювати швидкість досягнення новим співробітником запланованих показників продуктивності, рівень залученості та задоволеності роботою, а також частоту звернень до наставників і HR-підтримки. Ці дані дозволяють виявляти закономірності та оптимізувати програми онбордингу для різних департаментів або типів посад, підвищуючи ефективність усієї системи адаптації. Аналітика також дозволяє прогнозувати ризики плинності серед нових працівників. На основі історичних даних про адаптацію попередніх співробітників можна визначити фактори, які впливають на ймовірність звільнення на ранніх етапах. Наприклад, недостатня взаємодія з командою або низький рівень задоволеності навчальними

матеріалами можуть бути сигналами високого ризику. Це дає можливість HR-командам заздалегідь планувати втручання та підтримку, що підвищує ймовірність успішної інтеграції.

На стратегічному рівні аналітика адаптації нових працівників стає основою для побудови системи управління талантами та розвитку корпоративної культури. Дані, отримані під час онбордингу, допомагають формувати довгострокові програми навчання та розвитку, створювати ефективні наставницькі програми та адаптувати HR-стратегії під потреби різних команд. Використання аналітики дозволяє перетворити процес адаптації з рутинної операції на стратегічний інструмент підвищення залученості, продуктивності та довготривалої лояльності співробітників до компанії.

Отже, HR-аналітика є стратегічним інструментом сучасного управління людським капіталом, що об'єднує управління талантами, оцінку ефективності персоналу та передбачення HR-процесів на основі даних. Використання аналітичних методів дозволяє компаніям не лише виявляти і розвивати ключові компетенції співробітників, а й прогнозувати потреби у нових кадрах, оптимізувати програми навчання та підвищення кваліфікації, а також передбачати ризики плинності та вигорання. Завдяки збору та обробці кадрових, операційних та поведінкових даних, HR-аналітика забезпечує об'єктивну оцінку продуктивності та залученості, дозволяє будувати динамічні системи мотивації і винагороди, а також створює основу для стратегічного планування кадрових ресурсів, забезпечуючи довгострокову стійкість та конкурентоспроможність організації.

Питання для самоперевірки:

1. Як управління талантами з використанням HR-аналітики підвищує ефективність команд? Наведіть приклади компаній.
2. Які цифрові HR-системи є основою для впровадження People Analytics, і які їхні ключові функції?

3. Які основні типи HR-даних виділяють у сучасному HR-менеджменті?
Наведіть приклади кожного типу.
4. Як компанії використовують HR KPIs для прогнозування плинності кадрів та розвитку талантів?
5. Як сучасні цифрові та аналітичні підходи підвищують точність оцінки компетенцій?
6. Що таке 9-box grid і як вона допомагає у стратегічному управлінні талантами?
7. Яким чином 9-box grid визначає high-potentials (HiPo) і їхні програми розвитку?
8. Як data-driven learning development використовується для планування навчання та розвитку персоналу?
9. Які методи та показники застосовуються для оцінки ефективності навчальних програм у HR-аналітиці?
10. Як прогнозування HR-показників, зокрема плинності персоналу, допомагає організаціям знижувати ризики та підвищувати ефективність управління людським капіталом?

Тести для самоконтролю:

1. Який основний ефект управління талантами через HR-аналітику?
 - a) Підвищення ефективності команд та зниження плинності; ✓
 - b) Збільшення кількості співробітників;
 - c) Лише автоматизація зарплат;
 - d) Зменшення навчальних програм.
2. Які системи використовуються для People Analytics?
 - a) Excel та Google Docs;
 - b) HRIS, HCM-платформи та системи обліку персоналу; ✓
 - c) Лише соціальні мережі;
 - d) CRM-системи.
3. Які типи HR-даних виділяють?
 - a) Кадрові, операційні, поведінкові; ✓
 - b) Фінансові, бухгалтерські, логістичні;
 - c) Соціальні, медійні, маркетингові;
 - d) Лише особисті дані.

4. Як HR KPI допомагає прогнозувати плинність персоналу?

- a) Через аналіз історичних даних та тенденцій; ✓
- b) Через автоматичне продовження контрактів;
- c) Через маркетингові кампанії;
- d) Через зменшення зарплат.

5. Які є види компетенцій?

- a) Поведінкові та технічні; ✓
- b) Фінансові та маркетингові;
- c) Соціальні та медійні;
- d) Лише поведінкові.

6. Як цифрові підходи підвищують точність оцінки компетенцій?

- a) Через аналітику великих даних та автоматизовані оцінки; ✓
- b) Через вручну заповнені таблиці;
- c) Через усні співбесіди лише;
- d) Через маркетингові інструменти.

7. Як 9-box grid визначає high-potentials (HiPo)?

- a) Відображає співробітників із високими показниками результативності та потенціалу; ✓
- b) Визначає лише стаж співробітника;
- c) Визначає рівень зарплати;
- d) Відзначає тих, хто бере участь у корпоративних заходах.

8. Які методи застосовуються для оцінки ефективності навчальних програм?

- a) KPI, опитування, аналіз продуктивності; ✓
- b) Лише кількість відвіданих занять;
- c) Тільки час навчання;
- d) Продажі компанії.

9. Як прогнозування HR-показників допомагає організації?

- a) Знижує ризики, підвищує ефективність управління персоналом; ✓
- b) Лише збільшує витрати на HR;
- c) Не впливає на бізнес;
- d) Тільки для бухгалтерії.

10. Що таке data-driven learning development?

- a) Планування навчання та розвитку на основі аналітики HR-даних; ✓
- b) Список книг для читання;
- c) Таблиця обліку робочого часу;
- d) Лише оцінка зарплат.

Бібліографічний список:

1. ВОЛЯНСЬКА-САВЧУК, Л., & КЛІМАС, В. (2025). HR ТЕХНОЛОГІЇ: ВПЛИВ НА ЕФЕКТИВНІСТЬ УПРАВЛІННЯ ПЕРСОНАЛОМ У СУЧАСНОМУ БІЗНЕС-СЕРЕДОВИЩІ. *Herald of Khmelnytskyi National University. Economic Sciences*, 338(1), 133-139. <https://doi.org/10.31891/2307-5740-2025-338-19>
2. Драган, О., & Пилипенко, М. (2021). РОЗВИТОК УПРАВЛІННЯ ТАЛАНТАМИ В СИСТЕМІ МЕНЕДЖМЕНТУ ПЕРСОНАЛУ ПІДПРИЄМСТВА. *Економіка та суспільство*, (33). <https://doi.org/10.32782/2524-0072/2021-33-52>
3. Касьмін, Д., & Котельникова, Ю. (2024). ІННОВАЦІЙНІ HR-ТЕХНОЛОГІЇ: АДАПТАЦІЯ ПЕРСОНАЛУ ДО ЦИФРОВОГО СЕРЕДОВИЩА. *Цифрова економіка та економічна безпека*, (4 (13), 141-147. <https://doi.org/10.32782/dees.13-21>
4. Короленко, О., & Кутова, Н. (2023). HR-МЕНЕДЖМЕНТ ПІДПРИЄМСТВА: ВИКЛИКИ ТА РЕАЛІЇ СЬОГОДЕННЯ. *Економіка та суспільство*, (53). <https://doi.org/10.32782/2524-0072/2023-53-60>
5. Надточій І.І. (2024). Економічні передумови, тенденції розвитку та особливості впровадження HR технології крізь призму цифрових трансформацій. *Український журнал прикладної економіки та техніки*. Том 9. № 1. С. 261–265. DOI: <https://doi.org/10.36887/2415-8453-2024-1-4415>.
6. Підвальна О., Антипенко Я. (2024). СУЧАСНІ ТЕХНОЛОГІЧНІ ПІДХОДИ У СФЕРІ УПРАВЛІННЯ ПЕРСОНАЛОМ. *АГРОСВІТ* № 20. С. 102-108. DOI: 10.32702/2306-6792.2024.20.102
7. Пучкова С.І. (2022). HR-АНАЛІТИКА ЯК СУЧАСНИЙ ПІДХІД ДО ПРИЙНЯТТЯ КАДРОВИХ РІШЕНЬ В БІЗНЕСІ. *Ефективна економіка*. № 10. С. 1-17. DOI: <http://doi.org/10.32702/2307-2105.2022.10.32>

8. Чернікова, Н., Вороніна, В., & Чеботарьов, К. (2023). ІННОВАЦІЙНІ ПІДХОДИ В УПРАВЛІННІ HR-ПРОЦЕСАМИ НА ВІТЧИЗНЯНИХ ПІДПРИЄМСТВАХ. *Трансформаційна економіка*, (3 (03), 70-75. <https://doi.org/10.32782/2786-8141/2023-3-12>
9. Diyani Balthazaar. (2025). Digital technology and Human Resource Practices: A Systematic Literature Review, 12 August 2025, *PREPRINT* (Version 1). <https://doi.org/10.21203/rs.3.rs-7345845/v1>
10. Mahmoud, M.H., Ali, A.A., Alrifae, A.A. *et al.* (2025). The impact of digital HRM system and digital transformation on HR efficiency with organizational agility as a moderator. *Discov Sustain* 6, 1038 <https://doi.org/10.1007/s43621-025-01713-9>

Vydavatel:

Publishing house Education and Science s.r.o. IČO : 271 56 877.
Frýdlanská 15/1314 , Praha 8. MS v Praze , oddíl C,vložka 100614

Вступ до Бізнес-аналітики

Signed for printing on December 14, 2025
Format 60x90/8. Headset Times New Roman.
Mental printing. arc. 10,68. Edition online.