

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
Національний університет біоресурсів і природокористування України

**Методичні вказівки для вивчення та підготовки до модульного контролю та іспиту з  
дисципліни**

**«Бізнес-аналіз в розробці програмного забезпечення (частина 1)»**

для студентів ОС «Магістр»,  
що навчаються за спеціальністю  
«Інженерія програмного забезпечення»

Київ 2026

УДК 004.94  
ББК 32.973-018.2

Укладач: к.е.н., Ніколаєнко Д.В.

Рецензенти: к.т.н., доцент Голуб Б.Л., д.е.н., доцент Кравченко В.М.

*Затверджено навчально-методичною комісією факультету інформаційних технологій  
Національним університетом біоресурсів і природокористування України (протокол № від р.)*

Методичні вказівки для вивчення та підготовки до модульного контролю та іспиту з дисципліни «Бізнес-аналіз в розробці програмного забезпечення (частина 1)»/ уклад.: к.е.н. Ніколаєнко Д.В. – К.:НУБіП, 2026. – 32 с.

Містить розгорнутий матеріал до кожної теми курсу, перелік контрольних питань, тем для самопідготовки та тем для підготовки до іспиту. Для студентів, що навчаються за ОПП «Програмне забезпечення інформаційних систем» ОС «Магістр»

## Зміст

Лекція 1. Загальна характеристика управління проектами .....	4
Контрольні питання .....	6
Теми для самостійної підготовки.....	6
Лекція 2. Життєвий цикл проекту .....	7
Контрольні питання .....	9
Теми для самостійної підготовки.....	9
Лекція 3. Управління ризиками.....	11
Контрольні питання .....	13
Теми для самостійної підготовки.....	13
Лекція 4. Програмні засоби .....	14
Контрольні питання .....	16
Теми для самостійної підготовки.....	16
Лекція 5. Waterfall .....	17
Контрольні питання .....	19
Теми для самостійної підготовки.....	19
Лекція 6. Agile та Scrum .....	20
Контрольні питання .....	24
Теми для самостійної підготовки.....	24
Лекція 7. Kanban.....	26
Контрольні питання .....	28
Теми для самостійної підготовки.....	29
Перелік тем для підготовки до іспиту .....	30

## **Лекція 1. Загальна характеристика управління проектами**

### *Вступ*

У даній лекції розглядаються базові поняття управління проектами, які формують фундамент для подальшого вивчення дисципліни. Основна увага приділяється трьом ключовим питанням: що таке проект, що таке управління проектами та яка роль менеджера проекту. Матеріал цієї лекції дозволяє сформулювати цілісне розуміння сутності проектної діяльності та її відмінностей від інших видів діяльності.

Проектне управління охоплює широкий спектр практичних задач. Воно включає планування та координацію робіт, залучення фінансування, ефективне використання ресурсів, формування команди, мотивацію персоналу та управління конфліктами. Таким чином, управління проектами поєднує як технічні, так і управлінські аспекти діяльності.

### *Поняття проекту*

Проект визначається як тимчасове підприємство, спрямоване на створення унікального продукту, послуги або результату. Це означає, що проект має чітко визначені межі існування і не є безперервною діяльністю.

Тимчасовий характер проекту означає, що він завжди має початок і завершення. Завершення проекту настає тоді, коли досягнуті поставлені цілі, або коли стає зрозуміло, що ці цілі не можуть бути досягнуті, або коли проект втрачає свою актуальність. Таким чином, кожен проект має чітку орієнтацію на результат.

Ще однією ключовою характеристикою проекту є унікальність. Результат проекту не є повністю повторюваним, він створюється під конкретні умови, обмеження та потреби. Навіть якщо використовуються однакові ресурси або підходи, кінцевий результат все одно має відмінності.

### *Підходи до визначення проекту*

Різні організації пропонують власні підходи до визначення проекту. Наприклад, проект може розглядатися як діяльність із чітко визначеними цілями, обмеженнями за часом, вартістю та якістю. Також проект може трактуватися як комплекс взаємопов'язаних заходів, що реалізуються в межах заданого бюджету та періоду часу.

Функціональний підхід визначає проект як цілеспрямовану діяльність, спрямовану на задоволення конкретних потреб. Така діяльність здійснюється в умовах обмежених ресурсів та впливу зовнішніх і внутрішніх факторів.

### *Проект як процес*

Проект можна розглядати як процес змін. У цьому контексті проект є механізмом переходу певної системи з одного стану в інший. Система може бути організацією, інфраструктурою або будь-яким іншим об'єктом.

Проект є своєрідним прискореним переходом, який ініціюється для досягнення бажаного стану системи. Це означає, що проектна діяльність завжди пов'язана зі змінами та розвитком.

### *Характеристики проекту*

Кожен проект приводить до створення унікального результату. Навіть якщо окремі елементи можуть повторюватися, це не порушує загальної унікальності проекту.

Наприклад, будівництво будівель може здійснюватися з використанням однакових матеріалів або однією командою, але кожен об'єкт має свої особливості: місце розташування, архітектуру, умови виконання та інші фактори.

Важливо розрізнити проектну діяльність і операційну діяльність. Операційна діяльність є постійною, повторюваною і рутинною, тоді як проектна діяльність є тимчасовою, унікальною та орієнтованою на досягнення конкретної мети.

### *Результати проекту*

Результатом проекту можуть бути різні типи результатів. Це може бути продукт, який є частиною іншого виробу або кінцевим виробом. Також результатом може бути здатність надавати послуги або певні функції.

Крім того, результатом можуть бути дані або документи, які використовуються для прийняття рішень або подальшого розвитку.

Прикладами проектів є розробка нових продуктів, організаційні зміни, створення інформаційних систем, будівництво та впровадження нових процесів.

### *Управління проектами*

Управління проектами — це застосування знань, навичок, інструментів і методів для виконання робіт проекту та досягнення його цілей.

Цей процес реалізується через п'ять груп процесів: ініціація, планування, виконання, моніторинг і управління, завершення. Разом вони формують життєвий цикл проекту.

### *Складові управління проектами*

Управління проектом включає визначення вимог, роботу з зацікавленими сторонами та вирішення проблем, що виникають під час реалізації проекту.

Також важливою складовою є балансування обмежень проекту. До таких обмежень належать зміст, якість, розклад, бюджет, ресурси та ризики.

### *Менеджер проекту*

Менеджер проекту — це особа, відповідальна за досягнення цілей проекту. Він координує роботу команди та забезпечує виконання проекту в межах встановлених обмежень.

Роль менеджера проекту відрізняється від ролі функціонального або операційного менеджера. Функціональний менеджер відповідає за окрему область діяльності, а операційний — за постійну діяльність організації.

### *Ефективний менеджер проекту*

Ефективність менеджера проекту визначається трьома складовими: знаннями, результативністю та особистими якостями.

Знання включають розуміння принципів управління проектами. Результативність означає здатність застосовувати ці знання на практиці. Особисті якості охоплюють поведінку, лідерські здібності та здатність працювати з командою.

Особиста ефективність менеджера проявляється у здатності керувати командою, досягати поставлених цілей та балансувати обмеження проекту.

### **Контрольні питання**

- Що таке проект і які його ключові характеристики?
- У чому полягає тимчасовість проекту?
- Що означає унікальність результату проекту?
- Які існують підходи до визначення поняття «проект»?
- У чому різниця між проектною та операційною діяльністю?
- Які типи результатів може мати проект?
- Що таке управління проектами?
- Які групи процесів управління проектами існують?
- Які обмеження проекту необхідно балансувати?
- Яка роль менеджера проекту?

### **Теми для самостійної підготовки**

- Порівняння різних визначень поняття «проект»
- Проект як система змін
- Відмінність проектної та операційної діяльності
- Обмеження проекту та їх взаємозв'язок
- Роль менеджера проекту в сучасних організаціях

## Лекція 2. Життєвий цикл проекту

### *Вступ*

У даній лекції детально розглядаються ключові аспекти організації проекту: його основні ознаки, формування цілей і задач, процес управління, а також життєвий цикл проекту. Ця тема є логічним продовженням попередньої лекції та дозволяє перейти від загального розуміння проектів до більш структурованого бачення їх реалізації.

Розуміння життєвого циклу проекту та принципів його організації є критично важливим, оскільки саме це дозволяє ефективно планувати роботи, розподіляти ресурси та контролювати виконання проекту на всіх його етапах.

### *Ознаки проекту*

Будь-який проект має ряд характерних ознак, які дозволяють відрізнити його від інших видів діяльності. Перш за все, це унікальність. Унікальність означає, що кожен проект має неповторні умови виконання або новизну результату. Навіть якщо організація вже реалізовувала подібні проекти, кожен новий проект відрізняється контекстом, ресурсами або цілями.

Другою важливою ознакою є наявність чіткої цілі. Проект не існує сам по собі - він завжди спрямований на досягнення конкретного результату. Ця ціль визначає структуру робіт, логіку виконання та критерії успіху.

Проект також обмежений у часі. Це означає, що всі роботи мають бути виконані у визначені терміни, а їх послідовність повинна бути логічно обґрунтованою.

Не менш важливою є обмеженість ресурсів. У реальних умовах завжди існує дефіцит ресурсів - фінансових, людських або матеріальних. Це змушує менеджера проекту приймати рішення щодо їх оптимального використання.

Проект має чіткі межі - початок і завершення. Це дозволяє розглядати його як окрему систему з власною структурою.

Також проект часто вимагає специфічної організації, яка може відрізнитися від стандартної структури підприємства. І, нарешті, обов'язковою ознакою є наявність менеджера проекту та команди, які відповідають за його реалізацію.

### *Цілі та задачі проекту*

Проекти можуть бути спрямовані на досягнення різних результатів. Наприклад, це може бути збільшення обсягів продажу, підвищення якості продукції, розширення асортименту або вирішення соціально значущих проблем.

Цілі проекту - це ті результати, яких необхідно досягти. Вони можуть бути короткостроковими або довгостроковими. Зазвичай довгострокова ціль досягається через виконання ряду короткострокових цілей.

Важливо розуміти, що між цілями має існувати логічний зв'язок. Якщо цілей декілька, необхідно чітко визначити, як вони взаємодіють між собою.

Задачі проекту - це конкретні дії або події, які можна виміряти. Вони є практичними кроками, що ведуть до досягнення цілей.

Процес цілепокладання передбачає формалізацію цілей. Це означає, що цілі повинні бути чітко сформульовані, вимірні, обмежені у часі та досяжні.

### *Специфікація проекту*

Для ефективного управління проектом необхідно розбити його на складові частини. Саме для цього використовується структура робіт проекту (специфікація).

Специфікація представляє проект у вигляді ієрархії, де верхній рівень - це весь проект, а нижчі рівні - це підпроекти та окремі пакети робіт.

Такий підхід дозволяє краще зрозуміти обсяг робіт, визначити відповідальних та забезпечити контроль виконання.

Кожен елемент структури спрямований на досягнення кінцевого результату, що забезпечує цілісність проекту.

### *Процес управління проектом*

Управління проектом є системним процесом, який базується на взаємодії суб'єктів і об'єктів управління.

Суб'єкти управління - це активні учасники проекту, які приймають рішення. До них належать інвестори, замовники, підрядники та команда управління.

Об'єкти управління - це те, чим управляють: сам проект, його фази та процеси.

Управління здійснюється через прямий і зворотний зв'язок. Це означає, що результати виконання постійно аналізуються, і на основі цього приймаються нові управлінські рішення.

З точки зору часу виділяють три рівні управління: стратегічний, тактичний та оперативний. Стратегічний рівень охоплює весь проект, тактичний - середньострокове планування, а оперативний - щоденне управління роботами.

### *Функції управління проектом*

Управління проектом включає ряд функцій, які охоплюють різні аспекти діяльності. До них належать управління інтеграцією, часом, вартістю, якістю, ризиками, персоналом та комунікаціями.

Також важливими є управління контрактами, змінами, конфліктами та безпекою. Ці функції дозволяють комплексно контролювати проект і реагувати на можливі проблеми.

### *Фактори успіху проекту*

Успішність проекту залежить від багатьох факторів. Серед них особливо важливими є чіткість цілей, підтримка керівництва та наявність ресурсів.

Не менш важливим є детальне планування та контроль виконання робіт.

Контакт із замовником також відіграє ключову роль, оскільки дозволяє своєчасно уточнювати вимоги.

Здатність реагувати на непередбачені ситуації є критичною, оскільки в реальних проектах завжди присутня невизначеність.

### *Життєвий цикл проекту*

Життєвий цикл проекту - це період часу від моменту виникнення ідеї до повного завершення проекту.

Він включає послідовність фаз, які можуть частково перекриватися, але в цілому формують логічну структуру проекту.

Основні фази життєвого циклу: початок, організація і підготовка, виконання робіт та завершення проекту.

На початкових етапах рівень витрат і залучення персоналу є відносно низьким. У процесі виконання вони зростають, а на етапі завершення - зменшуються.

Також змінюється характер впливу різних факторів. На початку проекту вплив зацікавлених сторін є високим, а вартість змін - низькою. З часом ситуація змінюється: внесення змін стає дорожчим, а вплив зменшується.

### **Контрольні питання**

- Які основні ознаки проекту?
- Що таке цілі проекту і як вони формуються?
- У чому різниця між цілями і задачами?
- Що таке специфікація проекту (структура робіт)?
- Що входить до процесу управління проектом?
- Хто є суб'єктами та об'єктами управління?
- Які рівні управління проектом існують?
- Які функції управління проектом виділяють?
- Які фактори впливають на успішність проекту?
- Що таке життєвий цикл проекту?

### **Теми для самостійної підготовки**

- Цілепокладання в управлінні проектами
- Структура декомпозиції робіт (WBS/PBS)

- Рівні управління: стратегічний, тактичний, оперативний
- Фактори успіху проектів
- Динаміка параметрів проекту на різних фазах

## **Лекція 3. Управління ризиками**

### *Вступ*

У даній лекції розглядається одна з ключових складових управління проектами - управління ризиками. Будь-який проект реалізується в умовах невизначеності, тому розуміння природи ризиків, їх джерел, методів аналізу та способів управління є критично важливим.

Матеріал лекції охоплює визначення ризику, джерела його виникнення, підходи до аналізу ризиків, а також основні методи управління ними.

### *Поняття ризику*

Ризик визначається як подія, настання якої ставить під загрозу досягнення цілей проекту. Це означає, що ризик безпосередньо пов'язаний із невизначеністю та можливими відхиленнями від запланованого результату.

Важливо розуміти, що ризик - це не обов'язково подія, яка вже відбулася. Це потенційна подія, яка може відбутися в майбутньому і вплинути на проект.

Наприклад, затримка постачання матеріалів або зміна вимог замовника - це ризики, оскільки вони можуть завадити досягненню цілей проекту.

### *Управління ризиками*

Управління ризиками - це процес реагування на події та зміни ризиків у процесі виконання проекту. Це не одноразова дія, а постійний процес, який триває протягом усього життєвого циклу проекту.

Ключовим елементом управління ризиками є моніторинг. Це означає постійне відстеження стану ризиків та змін у середовищі проекту.

Ефективне управління ризиками дозволяє не лише реагувати на проблеми, але й запобігати їм ще до їх виникнення.

### *Моніторинг ризиків*

Моніторинг ризиків включає контроль ризиків протягом усього життєвого циклу проекту. Це означає регулярне оновлення інформації про ризики, оцінку їх актуальності та коригування планів реагування.

Якісний моніторинг забезпечує наявність актуальної інформації, яка допомагає приймати управлінські рішення до настання ризикових подій.

Таким чином, моніторинг є інструментом попередження проблем, а не лише їх фіксації.

### *Джерела виникнення ризиків*

Ризики можуть виникати з різних джерел. У загальному вигляді виділяють політичні, господарські, форс-мажорні та проектні ризики.

Кожна з цих груп має свої особливості та потребує окремого підходу до аналізу і управління.

### *Політичні ризики*

Політичні ризики пов'язані з нестабільністю зовнішнього середовища. До них належать зміни державного устрою, часті зміни уряду, нестабільність політичної влади та неадекватність політичних рішень.

Такі ризики є складними для прогнозування та можуть мати значний вплив на проект, особливо якщо він залежить від державного регулювання.

### *Господарські ризики*

Господарські ризики пов'язані з економічною діяльністю. Вони включають зміни податкового законодавства, ринкові ризики, інфляцію, зміну цін постачальників.

Також до них належать затримки платежів та ризики, пов'язані з неефективним управлінням.

Наприклад, відсутність попиту на продукцію може призвести до фінансових втрат.

### *Форс-мажорні ризики*

Форс-мажорні ризики - це події, які не залежать від учасників проекту. До них належать стихійні лиха, конфлікти та інші непередбачувані обставини.

Такі ризики складно контролювати, тому основний акцент робиться на підготовці до можливих наслідків.

### *Проектні ризики*

Проектні ризики виникають безпосередньо в процесі реалізації проекту. Вони можуть бути пов'язані зі зміною складу команди, недостатньою компетентністю, проблемами з вимогами або взаємодією між учасниками.

Причинами таких ризиків можуть бути неправильне визначення цілей, обмеженість ресурсів, конфлікти інтересів або недостатня кваліфікація персоналу.

Також до причин належать зовнішні фактори, проблеми з контрактами та якістю виконання робіт.

### *Аналіз ризиків*

Аналіз ризиків поділяється на якісний і кількісний. Якісний аналіз дозволяє визначити природу ризиків, їх види та джерела.

Кількісний аналіз спрямований на оцінку ймовірності ризиків та можливих втрат.

Для аналізу використовують різні методи: метод аналогій, експертні оцінки, розрахунково-аналітичний та статистичний метод.

Наприклад, метод аналогій базується на досвіді інших проектів, а експертний метод - на думці спеціалістів.

### *Методи управління ризиками*

Існує декілька основних методів управління ризиками: скасування, запобігання та контролювання, страхування та поглинання ризику.

Скасування ризику означає відмову від діяльності або її зміну таким чином, щоб ризик зник.

Запобігання та контролювання передбачає організацію роботи таким чином, щоб зменшити ймовірність настання ризику або його наслідки.

Страховання дозволяє компенсувати можливі збитки за рахунок спеціальних фондів.

Поглинання ризику означає прийняття можливих втрат, якщо вони є незначними.

### **Контрольні питання**

- Що таке ризик у проекті?
- Які основні джерела ризиків?
- Що таке управління ризиками?
- Яку роль відіграє моніторинг ризиків?
- Які існують види ризиків?
- Що таке якісний аналіз ризиків?
- Що таке кількісний аналіз ризиків?
- Які методи аналізу ризиків застосовуються?
- Які існують методи управління ризиками?
- У чому різниця між запобіганням і прийняттям ризику?

### **Теми для самостійної підготовки**

- Класифікація ризиків у проектах
- Методи оцінки ризиків
- Побудова системи управління ризиками
- Вплив ризиків на параметри проекту
- Практичні підходи до мінімізації ризиків

## Лекція 4. Програмні засоби

### *Вступ*

У даній лекції розглядаються програмні інструменти, які використовуються для управління проектами. Сучасні проекти практично неможливо ефективно реалізовувати без використання спеціалізованого програмного забезпечення. Такі інструменти допомагають планувати роботи, керувати ресурсами, контролювати виконання задач та організувати комунікацію.

У презентації розглядаються декілька популярних інструментів: MS Excel, MS Project, Atlassian Jira, MS OneNote, Todoist.

### *MS Excel як інструмент управління проектами*

MS Excel є універсальним інструментом, який широко використовується в управлінні проектами, незважаючи на те, що він не є спеціалізованою системою управління проектами.

За допомогою Excel можна реалізувати різні аспекти управління проектом. Наприклад, створюється матриця трасування вимог (RTM), яка дозволяє аналізувати юзкейси системи.

Також використовується структура декомпозиції проекту (PBS), яка дозволяє розбити проект на складові частини та оцінити обсяг робіт.

Excel активно застосовується для розрахунку бюджету проекту, що дозволяє контролювати витрати.

Окремо важливим є створення плану навантаження команди (Capacity Plan), що дозволяє оцінити, наскільки ефективно використовуються ресурси.

Крім того, в Excel можна будувати діаграми Ганта для візуалізації плану проекту.

Наприклад, менеджер може створити таблицю задач і відобразити строки їх виконання у вигляді діаграми.

### *MS Project*

MS Project є спеціалізованим інструментом для управління проектами, який дозволяє комплексно керувати всіма аспектами проектної діяльності.

У цьому інструменті можна створювати детальний план проекту, визначати залежності між задачами та встановлювати строки виконання.

MS Project дозволяє вести облік ресурсів, включаючи людські ресурси, та контролювати їх завантаження.

Інструмент підтримує різні варіанти представлення даних, що дозволяє аналізувати проект з різних точок зору.

Також можна здійснювати аналіз та оцінку ходу виконання проекту, що дає змогу своєчасно виявляти відхилення.

Окремою функцією є управління бюджетом проекту.

Наприклад, менеджер може відслідковувати перевищення витрат у режимі реального часу.

### *Atlassian Jira*

Atlassian Jira є популярним інструментом для управління проектами, особливо в ІТ-сфері.

Вона дозволяє управляти вимогами, що є важливим для розробки програмного забезпечення.

Також Jira використовується для управління проектами та командами, що дозволяє організувати роботу учасників.

Важливою функцією є управління багами, тобто відстеження та виправлення помилок.

Jira також забезпечує ефективну комунікацію із замовником.

Наприклад, всі задачі можуть бути представлені у вигляді беклогу, де кожна задача має статус і відповідального.

### *MS OneNote*

MS OneNote використовується для організації інформації в проекті.

Цей інструмент дозволяє зберігати нотатки, ідеї, документи та іншу інформацію в структурованому вигляді.

Він є зручним для ведення документації проекту та фіксації домовленостей.

### *Todoist*

Todoist є інструментом для управління задачами.

Він дозволяє планувати задачі, встановлювати нагадування та організувати роботу.

Такий інструмент особливо корисний для персонального планування або роботи невеликих команд.

Наприклад, менеджер може створювати списки задач на день або тиждень.

### *Slack, Teams, Signal, ...*

Використовується як засіб комунікації в проекті.

Він дозволяє організувати зустрічі, обговорення та обмін інформацією між учасниками проекту.

Комунікація є критично важливою для успішної реалізації проекту, тому такі інструменти відіграють важливу роль.

### *Інші інструменти*

Окрім розглянутих інструментів, існує велика кількість інших програмних продуктів для управління проектами.

Вибір інструменту залежить від специфіки проекту, розміру команди та вимог до функціональності.

### **Контрольні питання**

- Яку роль відіграють програмні засоби в управлінні проектами?
- Які задачі можна вирішувати за допомогою MS Excel?
- Які можливості має MS Project?
- Для чого використовується Jira?
- Яку роль відіграють інструменти для нотаток (OneNote)?
- Що дають task-менеджери (Todoist)?
- Яке значення мають інструменти комунікації?
- Як обрати інструмент для проекту?
- Які обмеження універсальних інструментів?
- Чим відрізняються спеціалізовані системи?

### **Теми для самостійної підготовки**

- Порівняння інструментів управління проектами
- Використання Excel у проектному менеджменті
- Системи управління задачами
- Інструменти командної комунікації
- Критерії вибору ПЗ для проектів

## Лекція 5. Waterfall

### *Вступ*

У даній лекції розглядаються основні підходи до організації процесу розробки проектів, які об'єднуються поняттям методологій управління проектами. Методологія визначає, як саме буде організовано процес роботи: у якій послідовності виконуються етапи, як приймаються рішення та як контролюється результат. У презентації основна увага приділяється каскадній (Waterfall) моделі.

### *Категорії методологій*

Методології управління проектами можна умовно поділити на декілька категорій. Серед них виділяють структуровані підходи, гнучкі (Agile) методології, а також підходи типу RAD.

До гнучких методологій належать різні варіанти, такі як Scrum, Kanban та інші. Вони відрізняються підходом до організації роботи та рівнем гнучкості.

Також існують підходи, які передбачають прототипування або поетапний розвиток проекту.

### *Каскадна модель (Waterfall)*

Каскадна модель є одним із класичних підходів до управління проектами. Вона передбачає послідовне виконання етапів проекту.

Процес відбувається строго поетапно: спочатку виконується аналіз, потім дизайн, далі розробка, тестування і завершується все впровадженням та підтримкою.

Кожен етап починається тільки після завершення попереднього, що забезпечує чітку структуру процесу.

#### *Етап аналізу*

На етапі аналізу формується розуміння того, що саме потрібно створити. Результатом цього етапу є моделі, схеми та бізнес-правила, які описують вимоги до системи.

Цей етап є критично важливим, оскільки саме тут закладається основа майбутнього проекту.

#### *Етап дизайну*

На етапі дизайну розробляється внутрішня архітектура програмного забезпечення. Визначаються способи реалізації вимог, структура системи та взаємодія її компонентів.

Важливо розуміти, що дизайн - це не лише зовнішній вигляд, а й логіка роботи системи.

#### *Етап розробки*

На етапі розробки відбувається безпосереднє написання коду програми. Також виконується інтеграція різних компонентів системи.

Це найбільш ресурсомісткий етап, на якому створюється основний функціонал продукту.

### *Етап тестування*

На етапі тестування перевіряється якість створеного продукту. Тестувальники виявляють дефекти та фіксують їх у спеціальних системах.

У разі виявлення помилок вони можуть бути виправлені, якщо дозволяють ресурси проекту.

### *Етап впровадження та підтримки*

На завершальному етапі продукт впроваджується та адаптується до реальних умов використання.

Також здійснюється технічна підтримка та оновлення продукту для виправлення помилок і додавання нового функціоналу.

### *Переваги Waterfall*

Каскадна модель має ряд переваг.

- Вона є зрозумілою та простою у використанні, що знижує поріг входження для команди.
- Чітка структура дозволяє легко відстежувати прогрес, ресурси та витрати.
- Вимоги визначаються на початку проекту і залишаються стабільними, що спрощує планування.
- Також є можливість точно оцінити вартість і строки виконання проекту.

### *Недоліки Waterfall*

Разом із перевагами існують і недоліки.

- Основним є відсутність гнучкості.
- Зміни у вимогах складно або неможливо внести під час виконання проекту.
- Процес є інерційним, і якщо виникають помилки на ранніх етапах, їх складно виправити.
- Також тестування відбувається на пізніх етапах, що підвищує ризики.

### *Модифікації Waterfall*

Існують модифікації каскадної моделі, які частково усувають її недоліки.

Наприклад, модель із перекриттям фаз дозволяє виконувати деякі етапи паралельно.

Інша модифікація передбачає поділ проекту на підпроекти.

Також існує підхід зі зниженням ризиків, який додає додаткові етапи контролю.

### *Застосування Waterfall*

Каскадна модель підходить для середніх і великих проектів, особливо коли працює декілька команд.

Вона ефективна у випадках, коли вимоги чітко визначені на початку і не змінюються.

Також вона підходить для проектів із фіксованим бюджетом та для впровадження типових рішень.

### **Контрольні питання**

- Що таке методологія управління проектами?
- Які існують типи методологій?
- У чому суть каскадної моделі (Waterfall)?
- Які етапи включає Waterfall?
- Яка роль етапу аналізу?
- Що відбувається на етапі дизайну?
- Які особливості етапу тестування?
- Які переваги Waterfall?
- Які недоліки Waterfall?
- Коли доцільно використовувати Waterfall?

### **Теми для самостійної підготовки**

- Порівняння Waterfall і Agile
- Детальний аналіз етапів життєвого циклу
- Ризики каскадної моделі
- Модифікації Waterfall
- Сфери застосування Waterfall

## Лекція 6. Agile та Scrum

### *Вступ*

У даній лекції розглядаються гнучкі підходи до управління проектами, зокрема Agile та Scrum. Ці підходи використовуються у проектах із високим рівнем невизначеності, де вимоги можуть змінюватися протягом усього життєвого циклу. На відміну від класичних моделей, гнучкі методології дозволяють швидко адаптуватися до змін і поступово створювати цінність для замовника.

### *Поняття Agile*

Agile - це підхід до управління проектами, який базується на принципах гнучкої розробки. Його основна ідея полягає у тому, щоб не намагатися детально спланувати весь проект наперед, а рухатися невеликими кроками, постійно отримуючи зворотний зв'язок і коригуючи напрям роботи.

Такий підхід особливо ефективний у ситуаціях, коли неможливо точно визначити всі вимоги на початку. Наприклад, при розробці програмного забезпечення замовник може змінювати свої очікування у процесі роботи.

### *Цінності Agile*

Agile базується на чотирьох ключових цінностях, які визначають пріоритети в роботі команди:

- Люди та взаємодія важливіші за процеси та інструменти.
- Працюючий продукт важливіший за вичерпну документацію.
- Співпраця із замовником важливіша за формальні домовленості.
- Готовність до змін важливіша за дотримання початкового плану.

Ці цінності визначають, як саме команда повинна приймати рішення в умовах невизначеності.

### *Принципи Agile*

Принципи Agile деталізують ці цінності. Наприклад, одним із ключових принципів є регулярна поставка працюючого продукту, що дозволяє замовнику бачити результат і вчасно вносити корективи.

Іншим важливим принципом є постійна взаємодія між бізнесом і розробниками. Це означає, що команда не працює ізольовано, а постійно узгоджує свої дії із зацікавленими сторонами.

Також важливими є принципи самоорганізації команди, простоти та постійного вдосконалення процесу роботи.

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

Найвищим пріоритетом для нас є задоволення потреб замовника завдяки регулярній та ранній поставці цінного програмного забезпечення.

2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

Зміна вимог вітається навіть у пізніх стадіях розробки. Agile-процеси дозволяють використовувати зміни для забезпечення конкурентної переваги замовника

3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

Працюючий продукт слід випускати якнайчастіше, з періодичністю від двох тижнів до двох місяців

4. Business people and developers must work together daily throughout the project.

Протягом усього проекту розробники та представники бізнесу мають щодня працювати разом

5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

Над проектом мають працювати вмотивовані професіонали. Щоб роботу було зроблено, створіть їм умови, забезпечте підтримку і повністю їм довіртеся

6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Безпосереднє спілкування - найбільш практичний та ефективний спосіб обміну інформацією як із самою командою, так і всередині команди

7. Working software is the primary measure of progress.

Працюючий продукт – основний показник прогресу

8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

Agile допомагає налагодити сталий процес розробки. Інвестори, розробники та користувачі повинні мати можливість нескінченно підтримувати постійний ритм

9. Continuous attention to technical excellence and good design enhances agility.

Постійна увага до технічної досконалості та якості проектування підвищує гнучкість проекту

10. Simplicity - the art of maximizing the amount of work not done – is essential.

Простота як мистецтво скоротити до мінімуму зайву роботу вкрай необхідна

11. The best architectures, requirements, and designs emerge from self-organizing teams.

Найкращі вимоги, архітектурні та технічні рішення народжуються у команд, що самоорганізуються.

12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Команда повинна систематично аналізувати можливі способи покращення ефективності та відповідно коригувати стиль своєї роботи

### *Scrum як реалізація Agile*

Scrum є однією з найпоширеніших реалізацій Agile. Він базується на ітеративному підході, коли робота виконується невеликими циклами - спринтами.

Кожен спринт має фіксовану тривалість і завершується створенням частини продукту, яка має цінність для замовника.

Такий підхід дозволяє зменшити ризики, оскільки проблеми виявляються на ранніх етапах, а не в кінці проекту.

### *Принципи Scrum*

Scrum базується на трьох ключових принципах:

- Прозорість - всі процеси та результати повинні бути зрозумілими і доступними.
- Перевірка - необхідно регулярно перевіряти результати роботи.
- Адаптація - у разі виявлення проблем необхідно швидко вносити зміни.

Ці принципи забезпечують контроль і гнучкість у процесі виконання проекту.

### *Ролі в Scrum*

Scrum-команда складається з трьох ролей: розробники, власник продукту та Scrum-майстер. Кожна роль має чітко визначені обов'язки і відповідальність.

#### *Розробники*

Розробники безпосередньо виконують роботу і створюють інкремент продукту.

Команда є крос-функціональною, тобто включає всі необхідні компетенції для виконання роботи.

Важливою особливістю є самокерованість: команда сама визначає, хто і як буде виконувати задачі.

Наприклад, якщо виникає нова задача, команда самостійно розподіляє її між учасниками без втручання зовнішнього менеджера.

#### *Власник продукту*

Власник продукту відповідає за максимізацію цінності продукту.

Його основне завдання - управління беклогом продукту: формування задач, визначення пріоритетів та постановка цілей.

Він виступає як зв'язуюча ланка між бізнесом і командою розробки.

Наприклад, якщо замовник змінює вимоги, саме власник продукту визначає, як ці зміни вплинуть на пріоритети задач.

#### *Scrum-майстер*

Scrum-майстер відповідає за правильне застосування Scrum-підходу.

Він допомагає команді працювати ефективно, усуває перешкоди та забезпечує дотримання процесу.

Його роль часто порівнюють із фасилітатором або сервісним лідером.

Наприклад, якщо команда стикається з організаційними проблемами, Scrum-майстер допомагає їх вирішити.

### *Артефакти Scrum*

Артефакти Scrum представляють собою результати роботи та інформацію про прогрес.

До них належать:

- Беклог продукту - список усіх вимог до продукту.
- Беклог спринту - задачі, які команда планує виконати у поточному спринті.
- Інкремент - готова частина продукту.

Кожен артефакт має відповідну ціль, яка дозволяє оцінювати прогрес.

### *Події Scrum*

Scrum включає набір регулярних подій, які забезпечують контроль і координацію роботи команди.

#### Спринт

Спринт є центральним елементом Scrum.

Це період, протягом якого команда створює інкремент продукту.

Усі інші події відбуваються всередині спринту, що робить його основною одиницею роботи.

#### Планування спринту

Під час планування команда визначає, що саме буде зроблено у спринті.

Це включає вибір задач із беклогу та оцінку їх виконання.

Наприклад, команда може вирішити реалізувати конкретний функціонал, який є пріоритетним для замовника.

#### Daily Scrum

Daily Scrum - це коротка щоденна зустріч (до 15 хвилин), яка допомагає синхронізувати роботу команди.

На цій зустрічі обговорюються поточні задачі, проблеми та плани на день.

Це дозволяє швидко виявляти перешкоди і оперативно реагувати.

#### Огляд спринту

Огляд спринту - це демонстрація результату роботи.

Команда показує створений продукт і отримує зворотний зв'язок від стейкхолдерів.

Це дозволяє переконатися, що проект рухається у правильному напрямку.

### Ретроспектива

Ретроспектива спрямована на покращення процесу роботи.

Команда аналізує, що було зроблено добре, а що можна покращити.

Наприклад, команда може вирішити змінити підхід до планування або комунікації.

### Правила Scrum

Scrum має ряд правил, які забезпечують стабільність процесу:

- Спринти мають однакову тривалість.
- Після завершення одного спринту одразу починається наступний.
- Усі події є обов'язковими.
- Зміни в беклозі під час спринту не рекомендуються.

### Обмеження Scrum

Scrum має певні обмеження.

- Його складно застосовувати у великих командах без додаткових підходів.
- Ефективність залежить від мотивації команди.
- Відсутність загального дедлайну може створювати ризики.
- Також значною проблемою може бути втрата ключових учасників команди.

### Контрольні питання

- Що таке Agile і в чому його суть?
- Які основні цінності Agile?
- Які принципи Agile?
- Що таке Scrum?
- Які ролі існують у Scrum?
- Які обов'язки Product Owner?
- Яка роль Scrum-майстра?
- Що таке спринт?
- Які події Scrum існують?
- Що таке ретроспектива і навіщо вона потрібна?

### Теми для самостійної підготовки

- Agile vs Waterfall

- Ролі Scrum-команди
- Життєвий цикл спринту
- Принципи Agile та їх застосування
- Обмеження Scrum

## Лекція 7. Kanban

### *Вступ*

У даній лекції розглядається методологія Kanban як один із підходів до організації роботи в проектах. Kanban є частиною гнучких підходів (Agile) і орієнтований на безперервний потік задач та оптимізацію процесу виконання робіт.

Основна ідея Kanban полягає не у жорсткому плануванні, а в балансуванні навантаження, візуалізації процесу та постійному вдосконаленні роботи команди.

### *Місце Kanban серед методологій*

Kanban належить до категорії Agile-підходів і розглядається поряд з іншими методологіями, такими як Scrum, Lean або Extreme Programming.

На відміну від структурованих підходів, Kanban не задає жорсткої послідовності етапів, а фокусується на управлінні потоком роботи.

### *Поняття Kanban*

Kanban - це система організації виробництва і постачання, яка дозволяє реалізувати принцип «точно в термін» (Just-in-Time).

Ця система була вперше застосована в компанії Toyota і з часом адаптована для управління проектами, зокрема в IT.

Суть підходу полягає в тому, щоб виконувати тільки ту роботу, яка потрібна в даний момент, і не створювати надлишкового навантаження.

### *Ключові властивості Kanban*

Однією з ключових властивостей Kanban є використання карток. Кожне завдання представляється у вигляді окремої картки, яка містить всю необхідну інформацію.

Другою важливою характеристикою є обмеження кількості задач на кожному етапі. Це дозволяє уникнути перевантаження і швидко виявляти проблеми.

Kanban також базується на принципі безперервного потоку, коли задачі виконуються послідовно відповідно до пріоритету.

Окремо слід виділити принцип постійного вдосконалення (kaizen), який передбачає регулярний аналіз процесу і пошук шляхів його оптимізації.

### *Основна ідея Kanban*

Основна ідея Kanban полягає в обмеженні кількості роботи, яка виконується одночасно. Нова задача не повинна починатися, поки не завершена попередня.

Це дозволяє уникнути перевантаження системи і забезпечити стабільний темп роботи.

Наприклад, якщо команда одночасно бере занадто багато задач, жодна з них не буде завершена швидко. Kanban змушує фокусуватися на завершенні.

### *Обмеження як механізм виявлення проблем*

Обмеження роботи дозволяють виявляти проблеми в процесі.

Якщо задача не може перейти на наступний етап через обмеження, це сигналізує про наявність вузького місця.

Таким чином, Kanban не приховує проблеми, а навпаки - робить їх очевидними.

### *Правила Kanban*

Kanban базується на трьох основних правилах:

1. Візуалізація процесу - всі задачі відображаються на дошці.
2. Обмеження кількості задач у роботі.
3. Вимірювання часу виконання задач і постійна оптимізація процесу.

Ці правила формують основу практичного застосування Kanban.

### *Kanban-дошка*

Kanban-дошка є основним інструментом візуалізації процесу.

У класичному вигляді вона містить три стовпці: To Do, In Work, Done.

Кожна задача рухається через ці стовпці, що дозволяє легко відстежувати її статус.

Однак у реальних проектах структура дошки може бути значно складнішою.

### *Kanban в «ідеальному світі»*

У більш деталізованому варіанті Kanban-дошка може включати такі етапи: беклог, аналіз, розробка, тестування, документування, прийняття та впровадження.

Кожен етап може мати підстатуси, наприклад «в процесі» та «виконано».

Такий підхід дозволяє детально відслідковувати процес розробки.

### *Kanban у реальних умовах*

У реальних проектах структура Kanban-дошки може бути довільною.

Команда самостійно визначає етапи залежно від специфіки роботи.

Наприклад, це можуть бути етапи: беклог, аналіз, розробка, тестування, прийняття та завершення.

### *Kanban як підхід балансу*

Канбан часто називають підходом балансу.

Його мета - забезпечити рівномірне навантаження на всіх учасників команди.

Це дозволяє уникнути ситуацій, коли одна частина команди перевантажена, а інша - простоює.

Важливою особливістю є відсутність жорстко визначених ролей, таких як Scrum-майстер або власник продукту.

### *Ефективність Kanban*

Основним показником ефективності є середній час проходження задачі через всі етапи.

Якщо задача проходить швидко - це свідчить про ефективну роботу команди.

Якщо ж виникають затримки, необхідно аналізувати, на якому етапі виникла проблема.

Таким чином, Канбан орієнтований на вимірювання і постійне покращення процесу.

### *Переваги Kanban*

Серед основних переваг Канбан можна виділити:

- Гнучкість планування - команда фокусується на поточній роботі.
- Висока залученість команди завдяки прозорості процесів.
- Скорочення тривалості циклу виконання задач.
- Наочність процесу та можливість швидко знаходити вузькі місця.

### *Недоліки Kanban*

Разом із перевагами існують і обмеження:

- Методологія погано підходить для великих команд.
- Ефективність залежить від універсальності навичок учасників.
- Не підходить для проектів із жорсткими дедлайнами.
- Слабка підтримка довгострокового планування.

### **Контрольні питання**

- Що таке Канбан?
- У чому його основна ідея?
- Що таке WIP-обмеження?
- Як працює Канбан-дошка?
- Які принципи Канбан?
- Що таке безперервний потік?

- Як виявляються «вузькі місця»?
- Як оцінюється ефективність?
- Які переваги Kanban?
- Які недоліки Kanban?

**Теми для самостійної підготовки**

- Kanban vs Scrum
- Візуалізація процесів
- Оптимізація потоку задач
- Управління навантаженням
- Метрики ефективності Kanban

## Перелік тем для підготовки до іспиту

1. Сутність проекту та його основні характеристики
2. Цілі проекту та процес цілепокладання
3. Результати проекту: види та характеристики
4. Життєвий цикл проекту та його фази
5. Суб'єкти управління проектом та їх ролі
6. Роль менеджера проекту та відповідальність за досягнення цілей
7. Рівні управління проектом (стратегічний, тактичний, оперативний)
8. Організаційні особливості виконання проектів
9. Структура робіт проекту (WBS) та декомпозиція
10. Мережеве планування проекту
11. Метод критичного шляху та його властивості
12. Поняття ризику в проекті та його вплив на параметри проекту
13. Класифікація ризиків (господарські, політичні, форс-мажорні тощо)
14. Причини виникнення проектних ризиків
15. Якісний аналіз ризиків та визначення їх характеристик
16. Методи аналізу ризиків (зокрема метод аналогій)
17. Управління ризиками як процес (сутність та етапи)
18. Методи реагування на ризики (поглинання, страхування тощо)
19. Моніторинг і контроль ризиків
20. Часові аспекти управління ризиками (управління на всіх фазах проекту)
21. Фактори успіху проекту
22. Принципи та цінності Agile-маніфесту
23. Працюючий продукт як основний показник прогресу в Agile
24. Гнучкість та управління змінами вимог у Agile
25. Роль комунікації в Agile-підходах
26. Scrum як фреймворк управління розробкою
27. Основні характеристики Scrum-команди (крос-функціональність, самокерованість)
28. Ролі в Scrum (Product Owner, Scrum Master, Developers)
29. Відповідальність Product Owner за управління цінністю продукту
30. Роль Scrum Master у підтримці команди та усуненні перешкод
31. Події Scrum та їх призначення
32. Послідовність подій у межах спринту
33. Daily Scrum та його особливості
34. Артефакти Scrum та відповідні їм commitments
35. Definition of Done та поняття інкременту
36. Три стовпи Scrum (прозорість, перевірка, адаптація)
37. Основи Kanban та візуалізація роботи
38. Елементи Kanban (дошка, картки, стовпці)
39. Управління потоком робіт у Kanban
40. Обмеження WIP та їх роль у процесі
41. Метрики ефективності в Kanban (cycle time)

42. Відсутність ролей у Kanban та командна організація
43. Waterfall як каскадна модель розробки
44. Фази життєвого циклу в Waterfall
45. Реалізація вимог на різних етапах Waterfall
46. Переваги Waterfall (структурованість, контроль, звітність)
47. Недоліки Waterfall (негнучкість, інерційність)
48. Обмеження застосування Waterfall (нестабільні вимоги)
49. Модифікації Waterfall та підходи до зниження ризиків
50. Порівняння Agile, Scrum, Kanban та Waterfall

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Waterfall Methodology: A Comprehensive Guide <https://www.atlassian.com/agile/project-management/waterfall-methodology>
2. What is scrum and how to get started <https://www.atlassian.com/agile/scrum>
3. What is kanban <https://www.atlassian.com/agile/kanban>
4. The New Methodology <https://martinfowler.com/articles/newMethodology.html>
5. Principles behind the Agile Manifesto <https://agilemanifesto.org/iso/en/principles.html>
6. Development managers vs. scrum masters <https://www.atlassian.com/agile/software-development/dev-managers-vs-scrum-masters>
7. Моделі мережевого планування і управління [https://stud.com.ua/80815/ekonomika/modeli\\_merezhevogo\\_planuvannya\\_upravlinnya#927](https://stud.com.ua/80815/ekonomika/modeli_merezhevogo_planuvannya_upravlinnya#927)