

## ПРАКТИЧНА РОБОТА №1

**ТЕМА:** Інсталяція ПЗ. Визначення предметної області. Створення UML-діаграм.

**МЕТА:** Виконати перші два етапи життєвого циклу розробки ПЗ.

### ТЕОРЕТИЧНІ ВІДОМОСТІ

#### 1. Життєвий цикл розробки ПЗ

Загальна схема життєвого циклу розробки будь-якого програмного продукту включає в себе п'ять етапів (див. рис. 1.1).



Рисунок 1.1 – Етапи життєвого циклу ПЗ

Першим етапом є «Початок». На цьому етапі визначається ідея, мета, обґрунтування та очікувані результати проекту. Також відбувається формування команди проекту та встановлення контакту з зацікавленими сторонами.

Далі відбувається «Планування». На цьому етапі розробляється детальний план проекту, який включає в себе визначення вимог, завдань, ресурсів, ризиків, бюджету, графіка та інших аспектів проекту.

«Реалізація» – третій етап, що виконує план проекту, тобто виконуються конкретні роботи, що призводять до створення продукту або послуги. Також на цьому етапі здійснюється управління якістю, комунікацією, змінами та ризиками.

«Тестування» – четвертий етап, на якому відстежується прогрес проекту, порівнюються фактичні та плановані показники, виявляються та вирішуються проблеми, збираються відгуки від користувачів та зацікавлених сторін.

На останньому етапі – «Завершення» – проект офіційно закривається, здійснюється передача продукту або послуги замовнику, оцінюються досягнуті результати, аналізуються уроки та рекомендації, проводиться підсумкова зустріч команди проекту.

## **2. UML-діаграми та Use Case (варіанти використання)**

UML – це мова моделювання, призначена для візуального представлення структури та поведінки програмних систем. UML дозволяє розробникам створювати графічні моделі, які відображають різні аспекти програмного продукту. Основні типи UML-діаграм включають:

1. **Діаграма класів (Class Diagram):** Відображає структуру системи шляхом показу класів програм та їх зв'язків.

2. **Діаграма варіантів використання (Use Case Diagram):** Відображає функціональність системи та її використання від імені користувача.

3. **Діаграма послідовності (Sequence Diagram):** Показує послідовність обміну повідомленнями між об'єктами або компонентами системи.

4. **Діаграма активності (Activity Diagram):** Відображає потік управління та послідовність дій у процесі.

5. **Діаграма компонентів (Component Diagram):** Показує фізичну організацію системи та залежності між компонентами.

6. **Діаграма станів (State Diagram):** Відображає стани об'єктів та переходи між ними в залежності від подій.

Use Case – це спосіб опису функціональних вимог до системи, який орієнтований на взаємодію користувачів з системою. Варіанти використання дозволяють описувати, які дії можуть виконувати користувачі та як система реагує на їх дії. Кожен варіант використання описується у вигляді сценарію, що включає в себе акторів (користувачів), дії та взаємодії між користувачами та системою.

Наприклад, для онлайн магазину можуть бути такі варіанти використання:

- Реєстрація нового користувача.
- Пошук товарів.
- Додавання товарів до кошика.
- Оформлення замовлення.
- Оплата замовлення.
- Відстеження статусу замовлення.

Ці варіанти використання допомагають розробникам та стейкхолдерам краще зрозуміти потреби користувачів та функціональність системи.

Узагальнюючи, UML-діаграми надають інструмент для моделювання різних аспектів програмних систем, в той час як варіанти використання допомагають описати функціональні вимоги та взаємодію з користувачами. Обидва ці підходи є важливими для розробки програмного забезпечення та розуміння вимог до системи.

## **ХІД РОБОТИ**

### **На оцінку «три»:**

1. Визначити предметну область застосування бази даних з урахуванням реалізації майбутнього веб-інтерфейсу або додатку.
2. Описати таблиці бази даних: їх призначення, вміст, ієрархію, зв'язки. Необхідно реалізувати як мінімум 5 таблиць та 5 стовпців у кожній. Унікальне поле ідентифкатора id є обов'язковим для кожної таблиці.
3. Побудувати UseCase діаграму варіантів використання бази даних з відображенням ролей користувачів.
4. Побудувати фізичну модель даних з відображенням зв'язків таблиць.
5. Виконати інсталяцію програмного середовища розробки веб-інтерфейсу/додатку, СКБД та БД, необхідних для реалізації проекту.
6. Оформити звіт з описом та візуальною демонстрацією виконання пунктів роботи.
7. Звіт завантажити у відповідну скриньку практичної роботи на курсі.

### **На оцінку «чотири»:**

1. Визначити предметну область застосування бази даних з урахуванням реалізації майбутнього веб-інтерфейсу або додатку. Обґрунтувати свій вибір.
2. Описати таблиці бази даних: їх призначення, вміст, ієрархію, зв'язки. Необхідно реалізувати як мінімум 5 таблиць та 5-10 стовпців у кожній. Унікальне поле ідентифкатора id є обов'язковим для кожної таблиці.
3. Побудувати UseCase діаграму варіантів використання бази даних з відображенням ролей користувачів та доступності даних на основі призначеної ролі.
4. Побудувати логічну та фізичну моделі даних з відображенням зв'язків таблиць.

5. Виконати інсталяцію програмного середовища розробки веб-інтерфейсу/додатку, СКБД та БД, необхідних для реалізації проекту. Обґрунтувати свій вибір.
6. Оформити звіт з описом та візуальною демонстрацією виконання пунктів роботи.
7. Звіт завантажити у відповідну скриньку практичної роботи на курсі.

**На оцінку «п'ять»:**

1. Визначити предметну область застосування бази даних з урахуванням реалізації майбутнього веб-інтерфейсу або додатку. Обґрунтувати свій вибір.
2. Описати таблиці бази даних: їх призначення, вміст, ієрархію, зв'язки. Необхідно реалізувати як мінімум 5 таблиць та 5-10 стовпців у кожній. Унікальне поле ідентифкатора id є обов'язковим для кожної таблиці.
3. Побудувати UseCase діаграму варіантів використання бази даних з відображенням ролей користувачів та доступності даних на основі призначеної ролі.
4. Побудувати логічну та фізичну моделі даних з відображенням зв'язків таблиць та типу зв'язків.
5. Виконати інсталяцію програмного середовища розробки веб-інтерфейсу/додатку, СКБД та БД, необхідних для реалізації проекту. Обґрунтувати свій вибір.
6. Оформити звіт з описом та візуальною демонстрацією виконання пунктів роботи.
7. Звіт завантажити у відповідну скриньку практичної роботи на курсі.