МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Західноукраїнський національний університет

Факультет комп’ютерний інформаційних технологій

Кафедра інформаційно-обчислювальних систем та управління.

Лабораторна робота № 6

з дисципліни «Основи комп’ютерних наук»

Виконав:

Студент групи

КН-12

Туз В.А.

Тернопіль

2023 рік

**Методи розробки програмного забезпечення**

**Мета:** Ознайомитися з основними методами розробки програмного забезпечення.

**Хід роботи:**

Варіант 24

Проект Openbravo — це відкрите програмне забезпечення для управління бізнесом (ERP), яке дозволяє управляти фінансами, запасами, виробництвом та іншими аспектами бізнесу. Основні задачі включають автоматизацію бізнес-процесів, оптимізацію управління ресурсами та забезпечення ефективної роботи компаній.

Даний проект використовує мову програмування Java для розробки. Процес розробки включає у себе спільну роботу розробників з усього світу, які співпрацюють через системи контролю версій та спільні комунікаційні канали. Існує можливість вносити зміни в проект через систему підтримки кодування (commit).

Файл змін проекту (change log) використовується для реєстрації всіх внесених змін, включаючи оновлення, виправлення помилок, нові можливості та інші модифікації в коді. Це дозволяє команді розробників відстежувати еволюцію проекту та зберігати його історію змін для подальшого аналізу та розвитку.

Процес розробки Openbravo включає співпрацю глобальної команди розробників, які використовують різні інструменти для спільної роботи над проектом.

1. **Комунікація та співпраця:** Розробники спілкуються через різноманітні канали, такі як форуми, чати, електронна пошта та спеціалізовані платформи для спільної роботи. Це дозволяє їм обмінюватися ідеями, вирішувати проблеми та обговорювати нові функції.
2. **Контроль версій:** Використання систем контролю версій, таких як Git, дозволяє розробникам працювати над кодом, вносити зміни та відстежувати їх історію. Кожна зміна зберігається, що полегшує співпрацю та розвиток проекту.
3. **Запис змін (change log):** Всі внесені зміни, оновлення та виправлення помилок реєструються у файлі змін (change log). Це дозволяє відстежувати розвиток проекту, а також аналізувати історію змін для подальшого вдосконалення.
4. **Тестування та відгуки спільноти:** Проект відкритий для співпраці зі спільнотою користувачів та розробників. Вони можуть тестувати нові функції, виявляти помилки та надавати відгуки, що допомагає вдосконалювати програмне забезпечення.
5. **Регулярні випуски версій:** Openbravo випускає нові версії з певною регулярністю, включаючи оновлення з новими функціями, виправленнями помилок та покращеннями. Це дозволяє користувачам отримувати нові можливості та стабільні версії програмного забезпечення.

Цей процес роботи спрямований на активну співпрацю команди розробників та спільноти користувачів для постійного вдосконалення та розвитку програмного забезпечення Openbravo.

**Контрольні запитання:**

1. Метод розробки програмного забезпечення (SDLC - Software Development Life Cycle) - це систематичний процес планування, створення, тестування, впровадження та обслуговування програмного продукту. Включає етапи аналізу, проектування, реалізації, тестування, впровадження та підтримки. Основна мета - забезпечити ефективну та якісну розробку програми в рамках визначених термінів та бюджету.
2. Існує кілька основних методів розробки програмного забезпечення (ПЗ):

* **Каскадна модель (Waterfall Model):** Лінійний підхід, де кожен етап розробки виконується послідовно, починаючи з аналізу і закінчуючи тестуванням.
* **Методологія гнучкої розробки (Agile):** Ітеративний підхід, що розбиває весь процес на короткі цикли розробки (ітерації) з акцентом на співпрацю, комунікацію та зміну вимог.
* **Прототипування (Prototyping):** Включає створення прототипів або моделей продукту для розуміння вимог та отримання зворотного зв'язку від користувачів.
* **RAD (Rapid Application Development):** Акцент на швидкості і гнучкості, з використанням компонентів та готових модулів для швидкої розробки.
* **DevOps:** Інтеграція розробки (Dev) та операцій (Ops) з метою покращення швидкості та якості постачання програмного забезпечення.
* **Spiral Model:** Комбінація каскадного та ітеративного підходів, де кожен оборот спіралі представляє один ітеративний цикл розробки.

Ці методи варіюються за ступенем формальності, гнучкістю та підходами до управління процесом розробки. Обирають метод відповідно до конкретних вимог проекту та умов роботи.

1. Модель водоспаду включає наступні етапи розробки програмного забезпечення:

* **Аналіз вимог (Requirements Analysis):** Збір та формалізація вимог до системи.
* **Проектування (Design):** Розробка архітектури системи та деталізація компонентів.
* **Реалізація (Implementation):** Написання програмного коду та створення системи на основі проектування.
* **Тестування (Testing):** Верифікація та валідація системи, виявлення та виправлення помилок.
* **Впровадження (Deployment):** Реліз та впровадження системи в реальне середовище користувачів.
* **Обслуговування та підтримка (Maintenance):** Підтримка, виправлення помилок, вдосконалення та розвиток системи.

Кожен етап виконується послідовно, і результати одного етапу використовуються наступним. Модель водоспаду передбачає лінійний, послідовний характер робіт.

1. Особливість моделі прототипів полягає в тому, що вона передбачає ітеративний підхід до розробки. Замість того, щоб визначити всі вимоги перед початком розробки, у цій моделі спочатку створюється прототип або проміжний варіант системи. Прототип служить засобом вивчення та уточнення вимог замовника.
2. Ітераційна розробка - це методологія розробки програмного забезпечення, яка передбачає поділ процесу на короткі ітерації або цикли. Кожна ітерація включає в себе аналіз, проектування, реалізацію та тестування певного функціоналу. Процес ітераційний, і після кожної ітерації може бути внесено зміни або додано новий функціонал. Цей підхід сприяє гнучкості, можливості швидко адаптуватися до змін вимог та надає замовнику можливість бачити проміжні результати швидше.
3. Основна відмінність полягає в тому, що інкрементна модель розглядає розробку через послідовні інкременти, тоді як спіральна модель акцентується на контролі ризиків та використанні ітерацій в процесі.
4. Модель швидкої розробки, або Rapid Application Development (RAD), має такі особливості:

* **Швидкість розробки:** Спрямована на швидше створення функціональних прототипів та продукту за коротший період часу.
* **Активна участь користувача:** Заохочує активну участь та зворотний зв'язок від користувачів під час всього процесу розробки.
* **Ітеративність та інкрементальність:** Використовує ітераційні цикли та поступове додавання функціоналу, що полегшує зміни та вдосконалення.
* **Використання компонентів:** Заохочує використання готових компонентів або бібліотек для прискорення процесу розробки.
* **Фокус на прототипах:** Створення прототипів для конкретизації вимог та отримання зворотного зв'язку від користувачів.
* **Гнучкість та адаптивність:** Забезпечує гнучкість у відповіді на зміни вимог та можливість швидкої корекції.
* **Можливість швидкої поставки:** Дозволяє швидко виводити прототипи або функціонально готові продукти на ринок.

Ці особливості роблять модель швидкої розробки ефективною для проектів, де важлива швидкість, гнучкість та активна взаємодія з користувачами.

**Висновок:** Мета ознайомлення з основними методами розробки програмного забезпечення полягає в тому, щоб отримати розуміння процесу створення програмних продуктів. Це включає у себе знання процесів співпраці в команді розробників, використання інструментів контролю версій, методів тестування та впровадження змін у програмний код. Ознайомлення з цими методами допомагає розуміти етапи розробки програмного забезпечення та сприяє в подальшому вдосконаленні навичок у програмуванні та управлінні розробкою програм.