Шнурок Владислав Сергійович, студент

ІФНТУНГ, м. Івано-Франківськ

Михайлюк Ірина Романівна, канд. пед.наук, доцент

ІФНТУНГ, м. Івано-Франківськ

<https://orcid.org/0000-0002-6489-3982>

**РОЛЬ КОНТРОЛЮ ВЕРСІЙ У РОЗРОБЦІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

Розробка програмного забезпечення (ПЗ) – це складний та тривалий процес, під час якого дуже важливо забезпечити ефективну організацію роботи над проєктом. Створення програмних продуктів вимагає від команди розробників злагодженої роботи, збереження історії змін та уникнення помилок і дублювання рішень. Для ефективного управління цим процесом необхідне впровадження систем, які дозволяють контролювати кожен етап розробки. Одним із ключових інструментів, що дозволяють досягти цієї мети, є система контролю версій [1].

Система контролю версій дозволяє розробникам зберігати різні версії коду, відстежувати зміни, відновлювати попередні стани та ефективно працювати в команді, мінімізуючи конфлікти та помилки. Завдяки таким системам кожен учасник може незалежно вносити зміни, маючи доступ до актуального та перевіреного коду на будь-якому етапі розробки, що є критично важливим при командній роботі.

Розглянемо основні переваги використання систем контролю версій:

1. *Співпраця та командна робота.* Сучасні програмні проєкти часто створюються великими командами, де кожен розробник відповідає за конкретний компонент або функціональність. Системи контролю версій, як-от Git, дозволяють працювати над окремими частинами коду незалежно, що знижує ризик конфліктів та втрати даних. Використання гілок забезпечує можливість створювати окремі версії проєкту, які після завершення роботи легко об'єднати з основною версією, роблячи процес командної роботи більш організованим та ефективним.
2. *Відстеження змін та збереження історії.* Системи контролю версій надають можливість переглядати повну історію змін, включно з тим, що було змінено, коли і ким. Це спрощує аналіз і виправлення помилок, дозволяючи швидко знайти джерело проблеми та повернутися до попередньої стабільної версії. Крім того, історія змін допомагає відстежувати прийняті раніше рішення та уникати повторення помилок.
3. *Підвищення якості коду.* Завдяки можливості паралельного тестування різних версій коду, інтеграції з інструментами автоматичного тестування та контролю якості (наприклад, CI/CD), зменшується ймовірність виникнення критичних помилок. Також системи контролю версій дозволяють проводити огляд коду (code review), що сприяє покращенню загальної якості програмного продукту.

Найпопулярнішими системами контролю версій є Git, SVN, Mercurial та Perforce [2]. Серед них найбільш широко використовується Git, завдяки своїй гнучкості, простоті та потужним можливостям для командної роботи. Git — це розподілена система контролю версій, яка дозволяє кожному розробнику мати локальну копію всієї історії проєкту, що забезпечує швидку роботу та підвищує автономність, зменшуючи залежність від інших учасників команди.

Git став стандартом у розробці програмного забезпечення завдяки широким можливостям, таким як робота з гілками, підтримка розподілених репозиторіїв і інтеграція з популярними платформами на зразок GitHub, GitLab і Bitbucket. Це дозволяє розробникам ефективно співпрацювати над проєктами з будь-якої точки світу.

У сучасному світі вимоги до якості, надійності та швидкості розробки програмного забезпечення постійно зростають. Системи контролю версій стали невід'ємними інструментами, які допомагають командам розробників підтримувати високі стандарти якості, мінімізувати ризики втрати даних і забезпечувати ефективну співпрацю. Це дозволяє не лише покращити організацію робочих процесів, а й скоротити витрати часу та ресурсів, що є особливо важливим в умовах стиснених термінів розробки та високої конкуренції на ринку.

**Література**

1. Про систему керування версіями https://git-scm.com/book/uk/v2
2. Системи керування версіями https://cherto4ka.xyz/2021/11/03 /