МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ  
ФАКУЛЬТЕТ КОМП’ЮТЕРНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра економічної кібернетики

Комплексне практичне індивідуальне завдання

з дисципліни «Конструювання ПЗ»

на тему:

«Контроль версій в комп’ютерних іграх»

Виконав :

студент групи ПЗС-32

Ензельт А. О.

Перевірив:

Сирник О. Й.

Тернопіль

2015

Зміст

[Вступ 3](#_Toc436869516)

[Загальні відомості 4](#_Toc436869517)

[Git 5](#_Toc436869518)

[Система контролю версій в комп’ютерних іграх 6](#_Toc436869519)

[Висновки 7](#_Toc436869520)

[Список використаних джерел 8](#_Toc436869521)

# Вступ

Система керування версіями – програмний інструмент для керування версіями одиниці інформації: вихідного коду програми, скрипту, веб-сторінки, веб-сайту, 3D моделі, текстового документу тощо.

Система керування версіями – це потужний інструмент, який дозволяє одночасно, без завад один одному, проводити роботу над груповими проектами.

Системи керування версіями зазвичай використовуються при розробці програмного забезпечення для відстеження, документування та контролю над поступовими змінами в електронних документах: у сирцевого коду застосунків, кресленнях, електронних моделях та інших документах, над змінами яких одночасно працюють декілька людей.

Кожна версія позначається унікальною цифрою чи літерою, зміни документу занотовуються. Зазвичай також зберігається автор зробленої зміни та її час.

Інструменти для контролю версій входять до складу багатьох інтегрованих середовищ розробки.

# Загальні відомості

Система контролю дозволяє зберігати попередні версії файлів та завантажувати їх за потребою. Вона зберігає повну інформацію про версію кожного з файлів, а також повну структуру проекту на всіх стадіях розробки. Місце зберігання даних файлів називають репозиторієм. В середині кожного з репозиторіїв можуть бути створені паралельні лінії розробки — гілки.

Гілки зазвичай використовують для зберігання експериментальних, незавершених(alpha, beta) та повністю робочих версій проекту(final). Більшість систем контролю версії дозволяють кожному з об'єктів присвоювати теги, за допомогою яких можна формувати нові гілки та репозиторії.

Використання системи контролю версії є необхідним для роботи над великими проектами, над якими одночасно працює велика кількість розробників. Системи контролю версії надають ряд додаткових можливостей:

* Можливість створення різних варіантів одного документу;
* Документування всіх змін (коли ким було змінено/додано, хто який рядок змінив);
* Реалізує функцію контролю доступу користувачів до файлів. Є можливість його обмеження;
* Дозволяє створювати документацію проекту з поетапним записом змін в залежності від версії;
* Дозволяє давати пояснення до змін та документувати їх;

# Git

Git – розподілена система керування версіями файлів та спільної роботи. Проект створив Лінус Торвальдс для управління розробкою ядра Linux, а сьогодні підтримується Джуніо Хамано. Git є однією з найефективніших, надійних і високопродуктивних систем керування версіями, що надає гнучкі засоби нелінійної розробки, що базуються на відгалуженні і злитті гілок. Для забезпечення цілісності історії та стійкості до змін заднім числом використовуються криптографічні методи, також можлива прив'язка цифрових підписів розробників до тегів і комітів.

Прикладами проектів, що використовують Git, є ядро Linux, Android, LibreOffice, Cairo, GNU Core Utilities, Mesa 3D, Wine, багато проектів з X.org, XMMS2, GStreamer, Debian DragonFly BSD, Perl, Eclipse, GNOME, KDE, Qt, Ruby on Rails, PostgreSQL, VideoLAN, PHP, Beryl, One Laptop Per Child (OLPC), АБІС Koha, GNU LilyPond та ELinks і деякі дистрибутиви GNU/Linux.

Програма є вільною і випущена під ліцензією GNU GPL версії 2.

Система спроектована як набір програм, спеціально розроблених з врахуванням їхнього використання у скриптах. Це дозволяє зручно створювати спеціалізовані системи управління версіями на базі Git або користувацькі інтерфейси. Наприклад, Cogito є саме таким прикладом фронтенда до репозиторіїв Git. А StGit використовує Git для управління колекцією латок.

Система має ряд користувацьких інтерфейсів: наприклад, gitk та git-gui розповсюджуються з самим Git.

Віддалений доступ до репозиторіїв Git забезпечується git-демоном, SSH або HTTP сервером. TCP-сервіс git-daemon входить у дистрибутив Git і є разом з SSH найпоширенішим і надійним методом доступу. Метод доступу HTTP, незважаючи на ряд обмежень, дуже популярний в контрольованих мережах, тому що дозволяє використання існуючих конфігурацій мережевих фільтрів.

# Система контролю версій в комп’ютерних іграх

Звичайно, без сумніву можна стверджувати, що системи контролю версій постійно використовуються розробниками комп’ютерних ігор. На сучасному етапі розвитку ігрової індустрії без таких систем просто не можна обійтися і зрозуміло чому. Ігри є не просто об’ємними, а надзвичайно об’ємними, над їх створенням працюють сотні людей. І кожна людина в команді працює над своєю частиною роботи, чи не кожну годину вносячи певні зміни в загальний проект. Система контролю версій створює «безпечну зону» для розробників. В разі чого, одна помилка не призведе до збоїв у цілій програмі.

Відомо, що, наприклад така система, як Git надає можливість інтегрувати себе в так звані «рушії гри», що дозволяє розробникам без зайвих зусиль керувати версіями їхнього продукту.

# Висновки

Ознайомившись з інформацією про систему контрою версій можна сказати – що це потужний інструмент в руках розробника програмного забезпечення. За допомогою контролю версій зміни в програмі відслідковувати легше, це допомагає також у командній роботі.

Можна припустити, що більш за все саме командам розробників доцільно використовувати системи контролю версій. А як відомо – комп’ютерні ігри створюються великою кількістю людей, часом це понад 200 розробників, які працюють над проектом одночасно. Коли один член команди працює над своєю ділянкою роботи, а інший над своєю, то в кінці-кінців можуть виникнути різні складні ситуації. Для того, щоб у процесі розробки незначні помилки не призводили до збоїв у всій системі і використовуються системи контролю версіями.

# Список використаних джерел

1. Git – [Електронний ресурс] – <https://ru.wikipedia.org/wiki/Git>
2. Система керування версіями – [Електронний ресурс] – <https://uk.wikipedia.org/wiki/Система_керування_версіями>
3. Система контролю версій – [Електронний ресурс] – <http://gamersoul.ru/sistema-kontrolya-versij-v-igrax.html>
4. Unity(Рушій гри) – [Електронний ресурс] – https://uk.wikipedia.org/wiki/Unity\_(рушій\_гри)