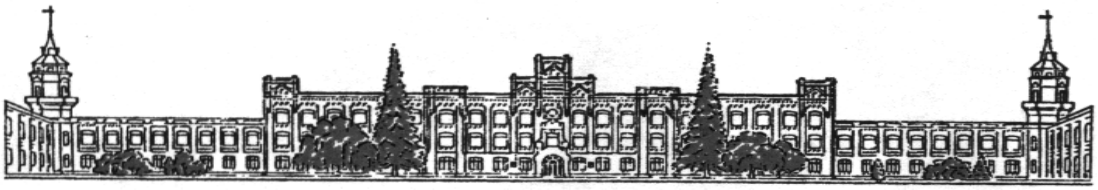

Міністерство освіти і науки, молоді та спорту України
Національний Технічний Університет України
«Київський Політехнічний Інститут»
Навчально-науковий комплекс
«Інститут прикладного системного аналізу»
Кафедра системного проектування

“«ПРОЕКТУВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ»”

Лабораторна робота № 1

“Системи контролю версій SVN, GIT."

Виконала:
студентка 4 курсу,

Група ДА-61

Фецун Анна

Варіант 24

м. Київ

2019

Мета роботи: за допомогою системи контролю версій завантажити коди програми у репозіторій. Відтворити типовий цикл розробки програмного забезпечення з використанням системи контролю версій.

Задача:

1. Вивчити основні команди роботи з репозіторіями.
2. Завантажити код програми у репозіторій.
3. Показати основний цикл роботи з програмним кодом за допомогою системи контролю версій.

Завдання

1. Обрати безкоштовну систему репозиторія для системи контролю версіями, наприклад projectlocker, або інш.
2. Встановити клієнтське безкоштовне програмне забезпечення для роботи с системою контролю версій (GIT, SVN clients).
3. Протягом роботи над лабораторними роботами 2-6 використовувати систему контролю версіями.
4. Описати цикл розробки програмного забезпечення з використанням системи контрою версій.

Зміст звіту

1. Мета роботи.
2. Завдання роботи.
3. Оформлення результатів роботи:
    3.1. Опис команд, які використовувалися протягом виконання роботи з системою контролю версіями.
    3.2. Лістинг каталогів у репозиторію для програмних кодів для л.р. 2-6.
4. Висновки.

Завдання

1. Обрати безкоштовну систему репозиторія для системи контролю версіями, наприклад projectlocker, або інш.
2. Встановити клієнтське безкоштовне програмне забезпечення для роботи с системою контролю версій (GIT, SVN clients).
3. Протягом роботи над лабораторними роботами 2-6 використовувати систему контролю версіями.

4. Описати цикл розробки програмного забезпечення з використанням системи

контрою версій.

Короткі теоретичні відомості

Система керування версіями (від англ Version Control System, VCS або Revision Control System.) – це спеціальне програмне забезпечення для полегшення роботи з інформацією, яка часто змінюється. Система керування версіями дозволяє зберігати декілька версій одного і того ж документа, при необхідності повертатися до більш ранніх версій, визначати, хто і коли зробив ту чи іншу зміну, керувати гілками різноманітних версій програми.

CREATE

Clone an existing repository -> $ git clone ssh://user@domain.com/repo.git

Create a new local repository -> $ git init

LOCAL CHANGES

Changed files in your working directory	$ git status
Changes to tracked files	$ git diff
Add all current changes to the next commit	$ git add .
Add some changes in <file> to the next commit	$ git add -p <file>
Commit all local changes in tracked files	$ git commit -a

COMMIT HISTORY

Show all commits, starting with newest -> $ git log

Show changes over time for a specific file -> $ git log -p <file>

Who changed what and when in <file> ->$ git blame <file>

BRANCHES & TAGS

List all existing branches -> $ git branch -av

Switch HEAD branch -> $ git checkout <branch>

Create a new branch based on your current HEAD -> $ git branch <new-branch>

Create a new tracking branch based on a remote branch -> $ git checkout --track <remote/branch>

Delete a local branch -> $ git branch -d <branch>

Mark the current commit with a tag -> $ git tag <tag-name>

MERGE & REBASE

Merge <branch> into your current HEAD -> $ git merge <branch>

Rebase your current HEAD onto <branch> -> $ git rebase <branch>

Abort a rebase -> $ git rebase --abort

Continue a rebase after resolving conflicts ->$ git rebase --continue

Use your configured merge tool to solve conflicts -> $ git mergetool

Use your editor to manually solve conflicts and (after resolving) mark file as resolved

$ git add <resolved-file>

$ git rm <resolved-file>