Assignment Report

Assignment 1
Git and Version control session

Prepared by
Pukar Giri
crazzy.lx75@gmail.com
Software Engineering Intern (SE15)
31st Internship Batch

Questions

Q.1) What is a .git directory?

Ans:

Whenever a git repository is initialized or cloned the git version control system creates a .git folder in the current directory. This folder has all the information which git needs in order to track the version history of the repository. It also has information like the remote address, commits, branches, etc.

The leading . in this folder is to hide it in the linux operating system while this folder is created with hidden properties in windows OS.

This folder has:

4 sub-directories:
1. hooks/ : example scripts
2. info/ : exclude file for ignored patterns
3. objects/ : all "objects"
4. refs/ : pointers to commit objects

And 4 files:
1. HEAD : current branch
2. config : configuration options
3. description
4. index : staging area

Here "*object* " includes:
- blobs(files)
- trees(directories)
- commits(reference to a tree, parent commit, etc)

Q.2) What is the difference between Git and Github?

Ans:

Git is a distributed Version Control System (VCS) developed by Linus Torvalds in 2005, this is a software system used to track and control the version history of files in a repository.

Github is a cloud service launched in 2008 which hosts the git repository and acts as a public git server.

In simpler terms Github is the hub where the git repositories can be hosted.

Q.3) Can we have a global .gitignore file?

Ans:

Yes, we can have a global .gitignore file. Whenever git wants to know which files to ignore it first looks for a file which is pointed by core.excludesFile then looks for the .gitignore file in the current folder/repository. So in order to have a global .gitignore file we can:

1. Create a file with all the global ignoring rules.
2. Place the file in some folder with access permission (preferably home directory of the user)
3. Point this file by core.excludesFile by running the command

```
git config --global core.excludesFile '<path to global gitignore file>'
```

Q.4) How to find the difference between two branches?

Ans:

We can find the differences between two branches by issuing the command:

```
git diff <branch name 1> <branch name 2>
```

The better way to view this difference is to use some 3rd party GUI tools which formats the output in better and comprehensible way.

Q.5) How to find the difference between two commits?

Ans:

We can find the differences between two commits by issuing the command:

```
git diff <Commit Hash 1> <Commit Hash 2>
```

The commit hash does not require to be complete but just long enough to be unique.

Q.6) How to find change of a file between two commits?

Ans:

We can find the differences between two branches by issuing the command:

```
git diff <Commit Hash 1> <Commit Hash 2>  <filename>
```

The commit hash can just be long enough to be unique. And the file name is the name of the file whose change is to be found out.

Q.7) How to find change of a file between two branches?

Ans:

We can find the differences between two branches by issuing the command:

```
git diff <Branch Name 1> <Branch Name 2>  <filename>
```

The file name is the name of the file whose change is to be found out.