**Part 3:**

**What is GitHub? When was it created? Why? By who? What similar platforms exist? Why would you use such a platform?**

GitHub is a cloud platform that uses Git as its core technology. GitHub simplifies the process of collaborating on projects and provides a website, more command-line tools, and overall flow that developers and users can use to work together. The site was launched in April 2008 by Tom Preston-Werner, Chris Wanstrath, P. J. Hyett and Scott Chacon. It’s main purpose is to facilitate the version control and issue tracking aspects of software development. There are various platforms that are similar to GitHub like, TaraVault, Bitbucket, Sourceforge, Gogs, Gitbucket, Phabricator. Speaking of GitHub, I would use it as it is a very good platform for developers. Whenever we make any changes to the code, the version control systems keep these revisions straight, storing the modifications in a central repository. This allows the developers to easily collaborate, as they can download a new version of the software, make changes, and upload the newest revision. Every developer can see these new changes, download them, and contribute. Also, people working together on the same project can collaborate easily because of GitHub.

**Part 4:**

**Define the following terms in the context of Git.**

**Repository:**

Repositories in Git contain a collection of files of various different versions of a Project. These files are imported from the repository into the local server of the user for further updations and modifications in the content of the file.

**Commit:**

Commit is basically an individual change which is made to a file or a set of files. When we save a file it creates a unique ID which allows you us to keep record of what changes were made when and by who.

**Push:**

The git push command uploads the content of the local repository to the remote repository. After a local repository has been modified, a push is executed to share the modifications with remote team members.

**Branch:**

InGit, a branch is a lightweight movable pointer to one of these commits. There is a master branch that points to the last commit which is made. Whenever we commit, the master branch pointer moves forward automatically. Master is basically just the default name given to the branch in Git.

**Fork:**

Forking is a git clone operation executed on a server copy of a project’s repository. Fork is a copy of a repository. Forking a repository allows us to freely experiment with changes without affecting the original project.

**Merge:**

Merge is Git's way of putting a forked history back together again. The git merge command lets us to take the independent lines of development created by git branch and integrate them into a single branch.

**Clone:**

In this, a copy of a specific repository or branch within a repository is created. By cloning with Git, we get the entire repository - all files, all branches, and all commits.

**Pull:**

The git pull command is used to fetch and download content from a remote repository and immediately update the local repository to match that content.

**Pull request:**

Pull request is requesting another developer in the project to pull a branch from our repository into their repository. It is a method of submitting contributions to an open development project.

**Part 6:**

**List the commands and strategy you used to do this part of the exercise in the *LastnameFirstnameGitTutorial-mm-dd-yyyy.docx* file and push it to YOUR repository.**

1. Go to repository in <https://github.com/paceuniversity/courses>
2. Forked the repository.
3. Created the local clone of the repository: “git clone <repo path>”
4. Added my name, date in the README.md file and saved
5. Committed the change: “git add .” and “git commit -m ‘Commit message’ ”
6. Pushed the edited file to the forked repository: “git push origin <branch\_name>”
7. Created a pull request to the repository in: <https://github.com/paceuniversity/courses>.