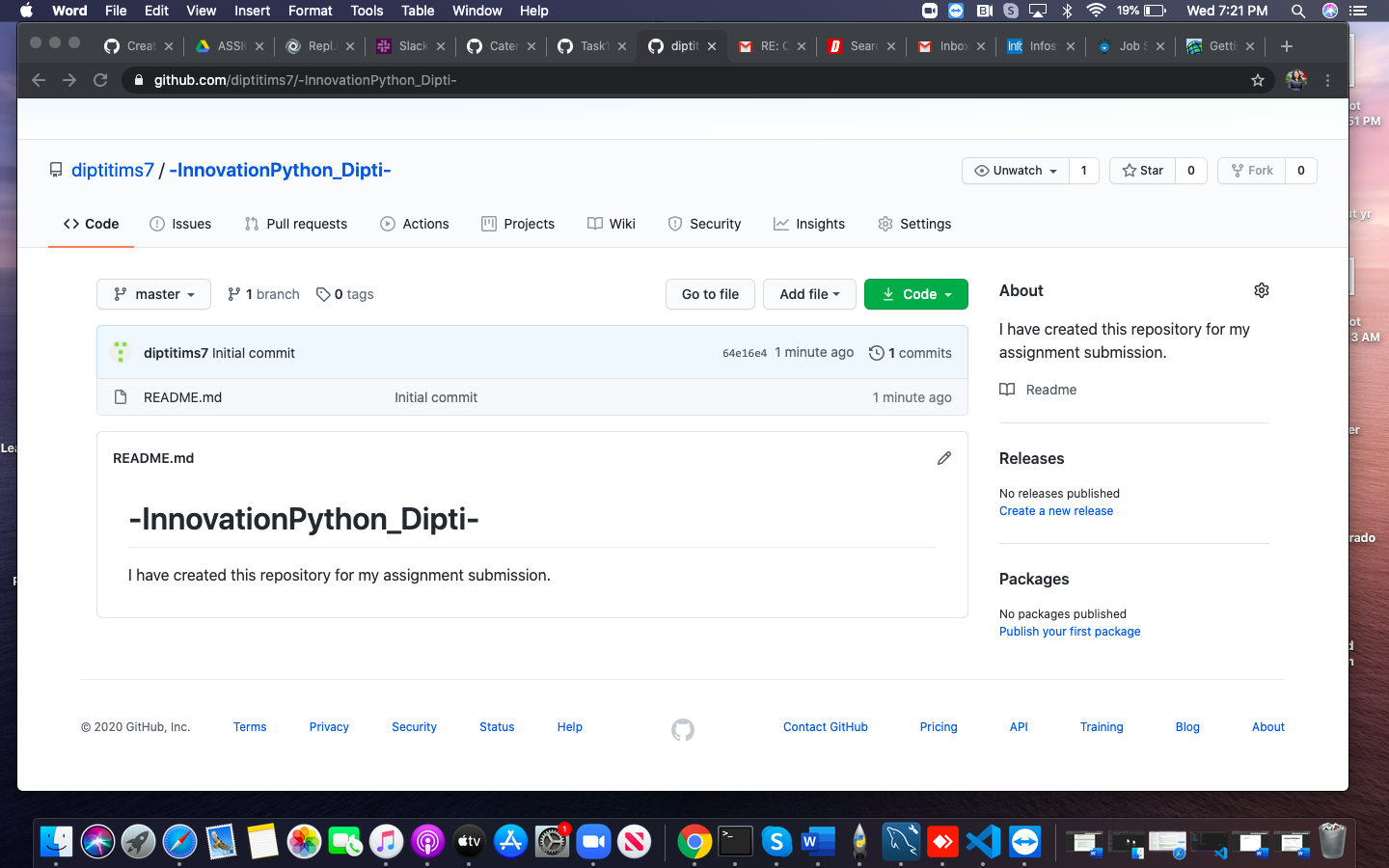
1. **Making a repository in Github Console:**

**“InnovationPython Dipti”**



1. **Difference between Git and Git Hub**

Git is a version control system that lets us manage the source code history and tracks all the changes that are conducted whereas Github is a cloud-based hosting service that lets the developers to manage the Git repositories.

1. **Git Workflow**

It is a method or recommendation for how we can use Git to accomplish the task in a better and more consistent manner. It is also very important for evaluating the workflow that needs to be done and also it encourages the users to leverage Git effectively and consistently where the decision of the user usually depends on the requirements of the team’s culture.Git workflow is centralized system which is great for teams transitioning to SVN where it initializes the central repository by hosting, cloning, making necessary changes and committing the repositories and do new commitments and also manage the conflicts.

**Types of Version control:**

There are three types of version control systems:

1. **Local version control system:** It is one of the simplest type of version control system which has a database that keeps all the changes to file under revision control. Eg: RCS (Revision Control System).
2. **Centralized Version Control System:** It is a kind of method that has only one repository in which each user gets their own working copy.
3. **Distributed Version Control Systems:** It consists of multiple repositories where each user has their own repository and working copy e.g.: Git, where it helps in avoiding the problem of a single point of failure.
4. **Explain branching concept in Git**

Branching in Git is very crucial factor of the everyday business process where the Git branches are effective pointer to view the changes. And also, if we want to add a new feature or fix a bug we can make a new branch to encapsulate the changes which makes it tougher for unstable code to get merged into the main code and contributes to clean up the future’s history before merging it into the main branch. It is also an independent line of development where command lets a user create, list, name and delete branches.

**Q5. Explain forking workflow in Git.**

It is fundamental workflow which is different than other popular Git workflows where a single server-side repository to act as the “central” codebase that gives every developer their own server-side repository. This means that each contributor has not one, but two Git repositories: a private local one and a public server-side one. The Forking Workflow is most common workflow among public and open source projects. The major benefit of the Forking Workflow is that contributions can be integrated without the need for everybody to push to a single central repository. The Forking Workflow typically follows a branching model i.e. based upon the [Gitflow Workflow](https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow) which means that complete feature of branches will be purposed for join into the original project maintainer's repository and the end result is a distributed workflow which provides a flexible way for large, organic teams.