

Week_2_pf

Portfolio Activity

Based on your experience using GitHub Pages to create a website:

- How do you see yourself using Git and GitHub in the software engineering field?

Let's start with a question: what do we need for a source control system. It is a version control. When we are writing a code, we constantly change it. It can be adding new features or fixing a bug or even putting in some documentation. What we want is to at least keep track of history/changes. For example, we added some features, deployed our application into production but somehow, entered a critical bug. Having track of changes, we easily can return our code to the previous state, when the world was pink and fluffy. So far, so good. Now, we decided to divide our work into pieces, each peace per defined time slot. Version control system, git helps here also quite well. Developers can take someone else's changes, and continue working on their own functionality.

Version control is not only used by developers, architects, product managers and who knows how to use it.

So, git itself is a command line set of tools. It is not a platform. Each developer should install this set of tools itself. But if we stop here and think a minute, we immediately discover that it is not enough. We need a platform with central fashion, with additional capabilities, like UI, CI/CD and a web site for the project. It also should be fault tolerant and replicated storage.

What if I would like to take on a project, and to make changes without anybody's approval. Github gives me such an ability - a fork. I can easily create a branch on my own by forking the mainstream. Another feature which is most useful and required is a pull request. We are working on branches, to keep the main branch clean and fresh. Having additional eyes on my work may make new code safer. Or say, a code can be aligned with project code standards.

I personally use github on a daily basis. Looking at PRs, improving documentation, adding new features etc. Organization on the other hand, can cop on me, my counting how commits I did for some period of time. Or having code in one place, we can do some security checks, code coverage checks and code complexity.

- Which software development method would work best with GitHub. Why?

As we understood earlier, github is a platform which provides some useful functionality. Let's have a look at what we need to have in the process of development. We are trying to work following agile methodology, scrum or kanban. In any case, we manage iterations, say two weeks. Github supports iterations. Next, we would like to search, navigate and run commands. A platform provides us with a feature called a command palette. As each team member works on specific tasks each spring, github provides a creation of a branch which when sprint finishes,

can be merged to the master/main branch. Who guarantees us that our code is secured? Github makes life easier and from time to time does security scanning.

Another very useful feature is pull requests. When a code is written, it is better that someone else will have a look at it. It is good practice, in my opinion. And my favorite feature is CI/CD - github actions. Github can run some jobs related to the kind of commit, for example, if it was a commit on a specific branch, it can run Jenkins for example to compile the code, to run integration tests and do some checks.

And say as a startup company, or any other developer, with github no need to have your own servers, maintain git and other tools. Keep eye on storage etc. Github solves a lot of maintenance problems, at a reasonable price.

Self reflection. My difficulties were around case studies. Somehow I could not get it, what does it mean and what to do. But stopping and thinking helped me to open new horizons to tackle a problem from a different angle.

Another challenge was to create a github page. I started uploading files one by one, and as a result missed something. Then I was thinking, what if I create a docs folder in my repo, copy everything from the zip file and then commit it and rebuild the pages. So, I tried this and it worked.

References:

1. Korbin Brown (Nov 13, 2019) What is github - <https://www.howtogeek.com/180167/htg-explains-what-is-github-and-what-do-geeks-use-it-for/>
2. Amy Reichert (Aug, 2022) 7 essential github features - <https://www.techtarget.com/searchsoftwarequality/tip/Evaluate-GitHub-features-for-dev-and-project-management>