ITE1008 – Open Source Programming

Digital Assignment – I
Fall Semester 2020 - 2021

Topic: Case Study on GitHub Version Control

Name-*Shourya Chambial*

Reg No-*19BIT0246*

The GitHub repository link is as follows: -

https://github.com/Shourya-Chambial/Shourya-Chambial.github.io

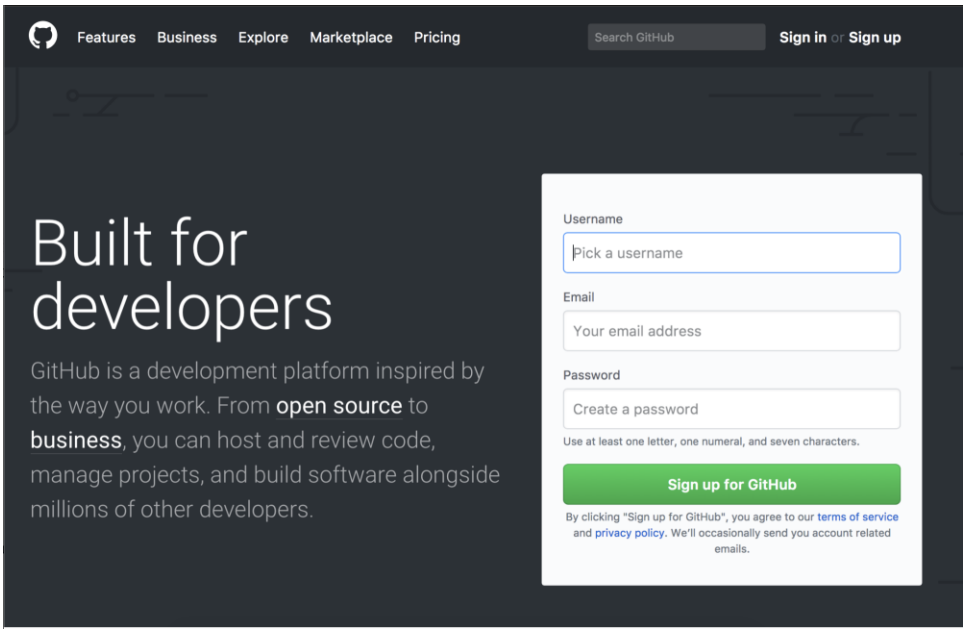
The link for the hosted personal portfolio is as follows: -

https://shourya-chambial.github.io/19BIT0246/

Step by step process of GitHub working methodology

GitHub is a web-based platform which is used for version control. Git simplifies the process of working with other people and makes it easy to collaborate on projects. Team members can work on files and easily merge their changes in with the master branch of the project.

Step 1: *Create a GitHub account on GitHub.com*

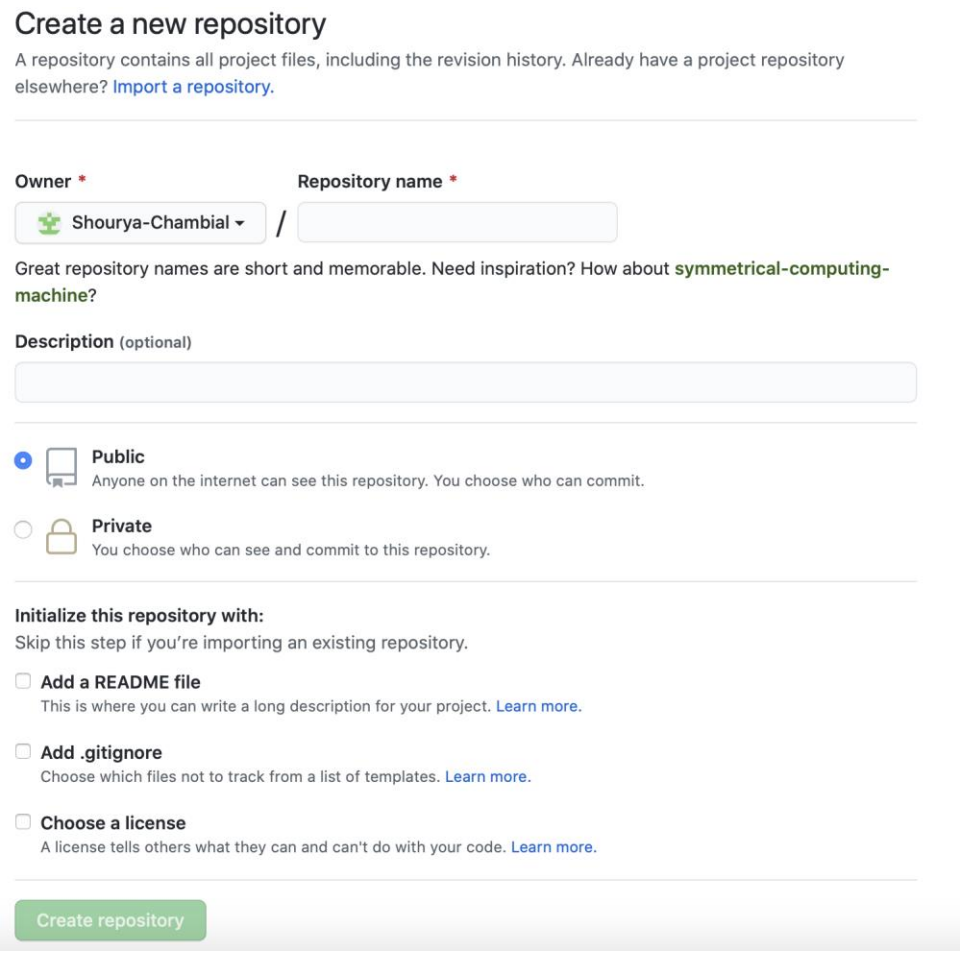


Step 2: *Create a repository*

A ***repository*** is usually used to organize a single project. Repositories can contain folders and files, images, videos, spreadsheets, and data sets – anything your project needs.

To create a new repository: -

- ★ In the upper right corner, next to your avatar or ident icon, click and then select New repository.
- ★ Name your repository.
- ★ Write a short description.
- ★ Select Initialize this repository with a README.
- ★ Click on create repository.

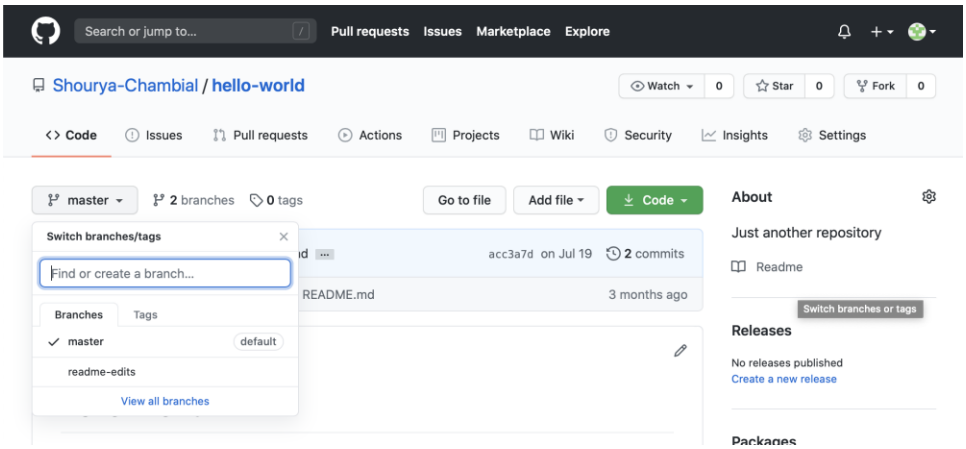


Step 3: *Create a branch*

Branches help you to work on different versions of a repository at one time. Let's say you want to add a new feature (which is in the development phase), and you are afraid at the same time whether to make changes to your main project or not. This is where git branching comes to rescue. Branches allow you to move back and forth between the different states/versions of a project.

To create a new branch: -

- ❖ Go to your new repository.
- ❖ Click the drop down at the top of the file list that says branch: main.
- ❖ Type a branch name of your wish into the new branch text box.
- ❖ Select the blue Create branch box or hit "Enter" on your keyboard.

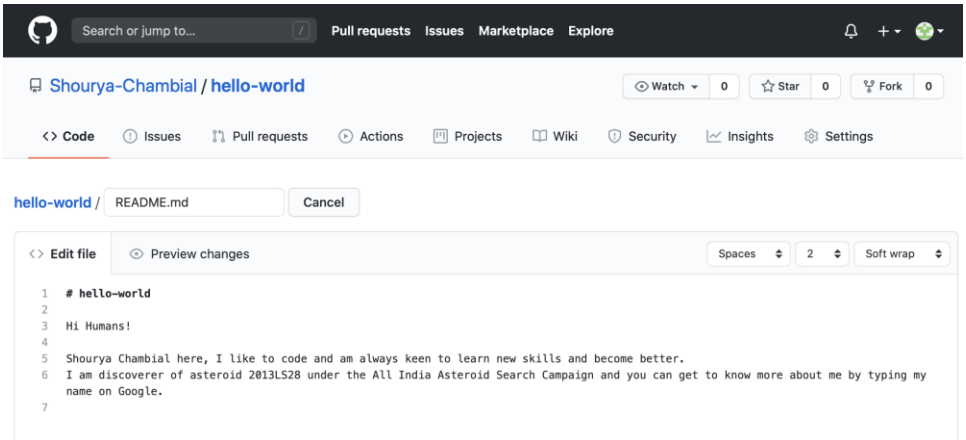


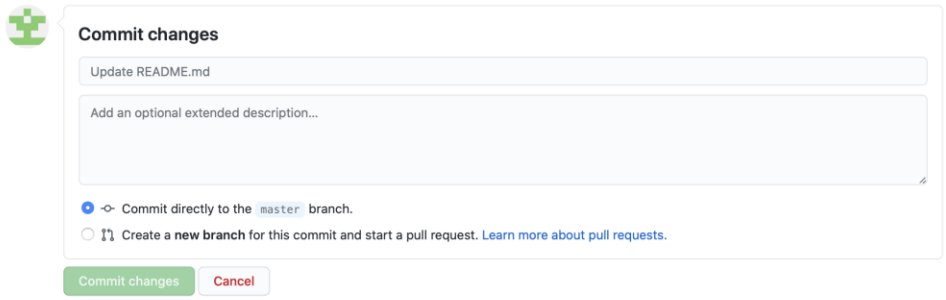
Step 4: *Make and commit changes*

This operation helps you to save the changes in your file. When you commit a file, you should always provide the message, just to keep in the mind the changes done by you. Though this message is not compulsory but it is always recommended so that it can differentiate the various versions or commits you have done so far to your repository.

To make and commit changes: -

- Click the README.md file.
- Click the pencil icon in the upper right corner of the file view to edit.
- In the editor, write a bit about yourself.
- Write a commit message that describes your changes.
- Click Commit changes button.



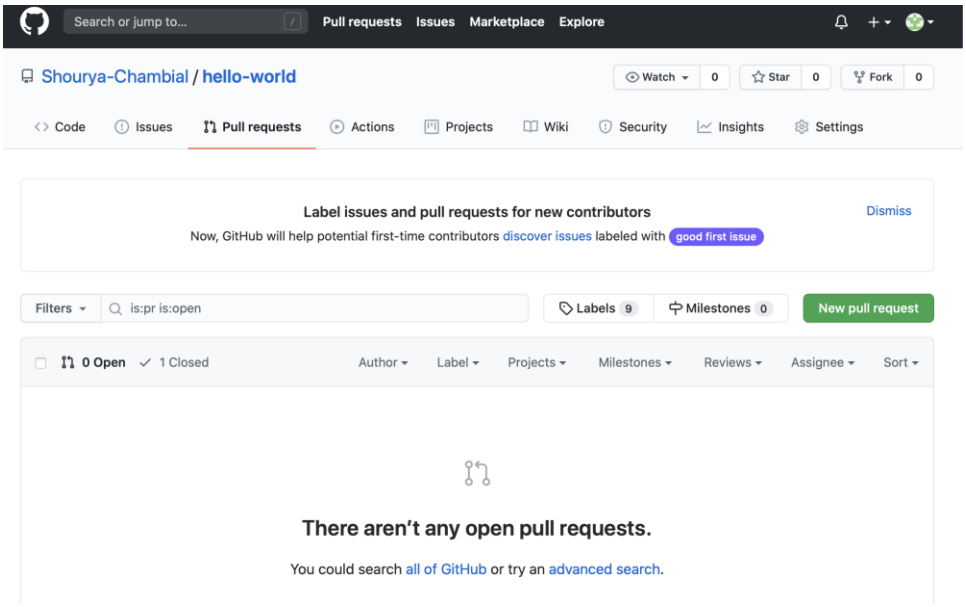


Step 5: *Open a pull request*

Pull command is the most important command in GitHub. It tells the changes done in the file and request other contributors to view it as well as merge it with the master branch. Once the commit is done, anyone can pull the file and can start a discussion over it. Once it's all done, you can merge the file. Pull command compares the changes which are done in the file and if there are any conflicts, you can manually resolve it.

For pull request: -

- Click the 'Pull requests' tab.
- Click 'New pull request'.
- Once you click on pull request, select the branch and click 'readme- changes' file to view changes between the two files present in our repository.
- Click "Create pull request".
- Enter any title, description to your changes and click on "Create pull request"

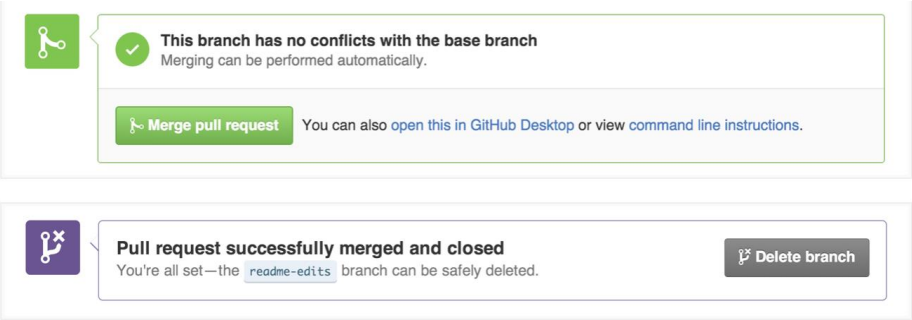


Step 6: *Merge your pull request*

Here comes the last command which merge the changes into the main master branch.

To merge a pull request: -

- Click the green Merge pull request button to merge the changes into main.
- Click Confirm merge.
- Go ahead and delete the branch, since its changes have been incorporated, with the Delete branch button in the purple box.



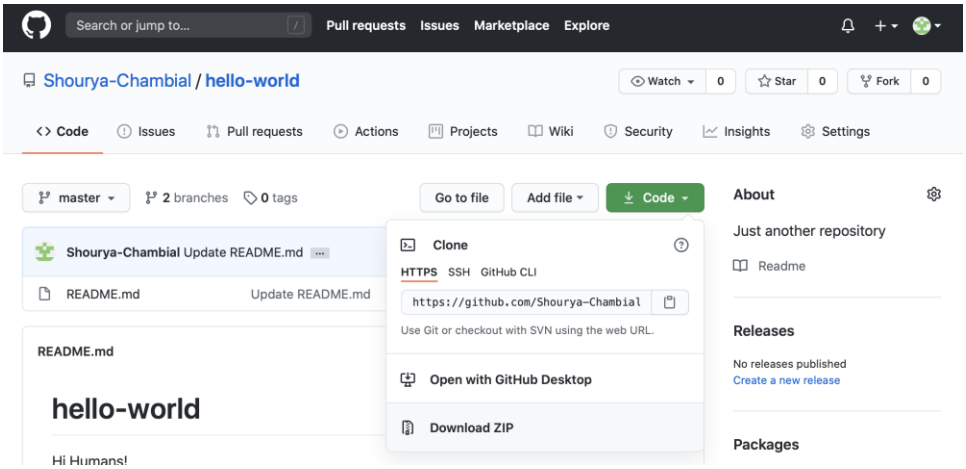
Step 7: *Cloning and forking GitHub repository*

Suppose you want to use some code which is present in a public repository, you can directly copy the contents by cloning or downloading.

Suppose, you need some code which is present in a public repository, under your repository and GitHub account. For this, we need to fork a repository.

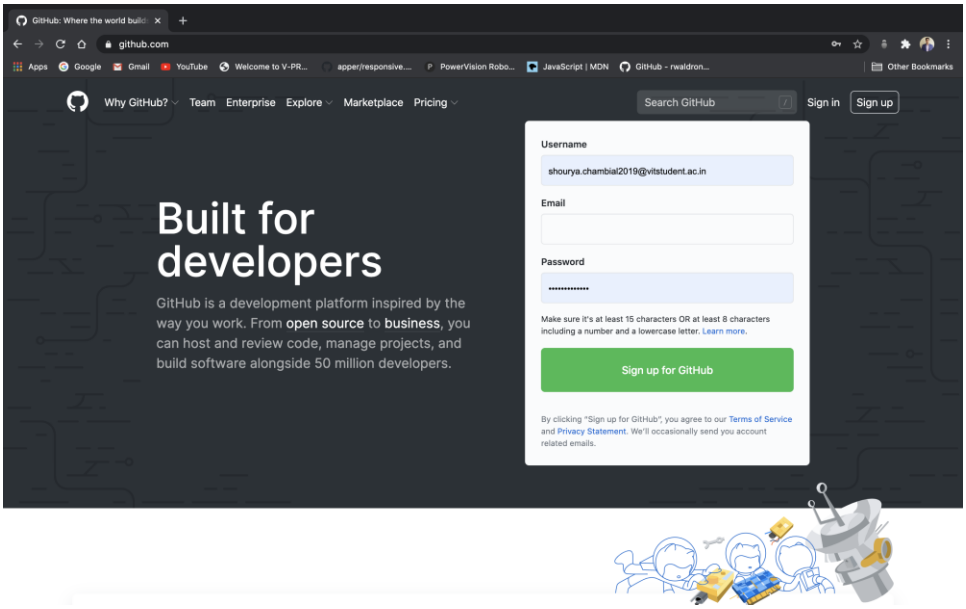
Changes done to the original repository will be reflected back to the forked repository.

If you make a change in forked repository, it will not be reflected to the original repository until and unless you have made a pull request.

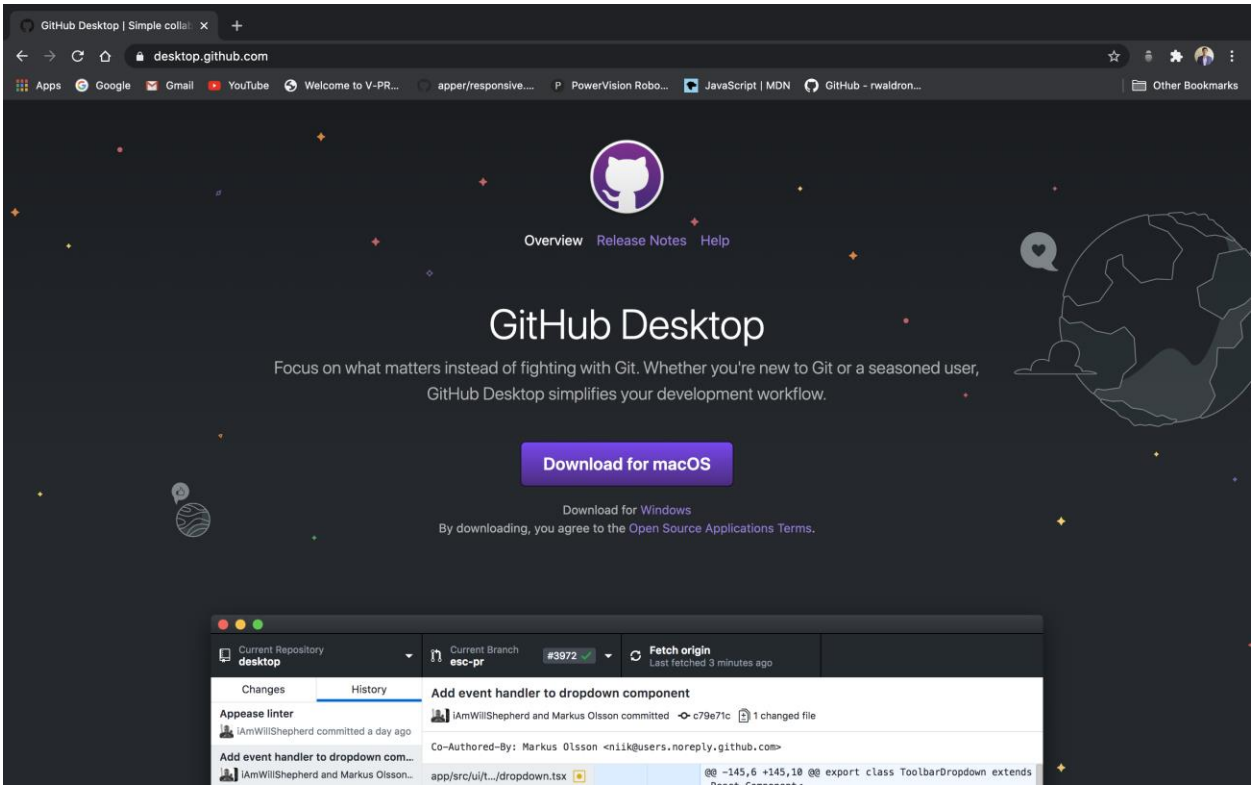


Different ways to access GitHub

GitHub can be accessed online using latest web browsers like the current versions of Chrome, Firefox, Safari, and Microsoft Edge.



GitHub can also be accessed through GitHub Desktop, an app interface for GitHub for various operating systems like macOS and windows.



My Personal Portfolio

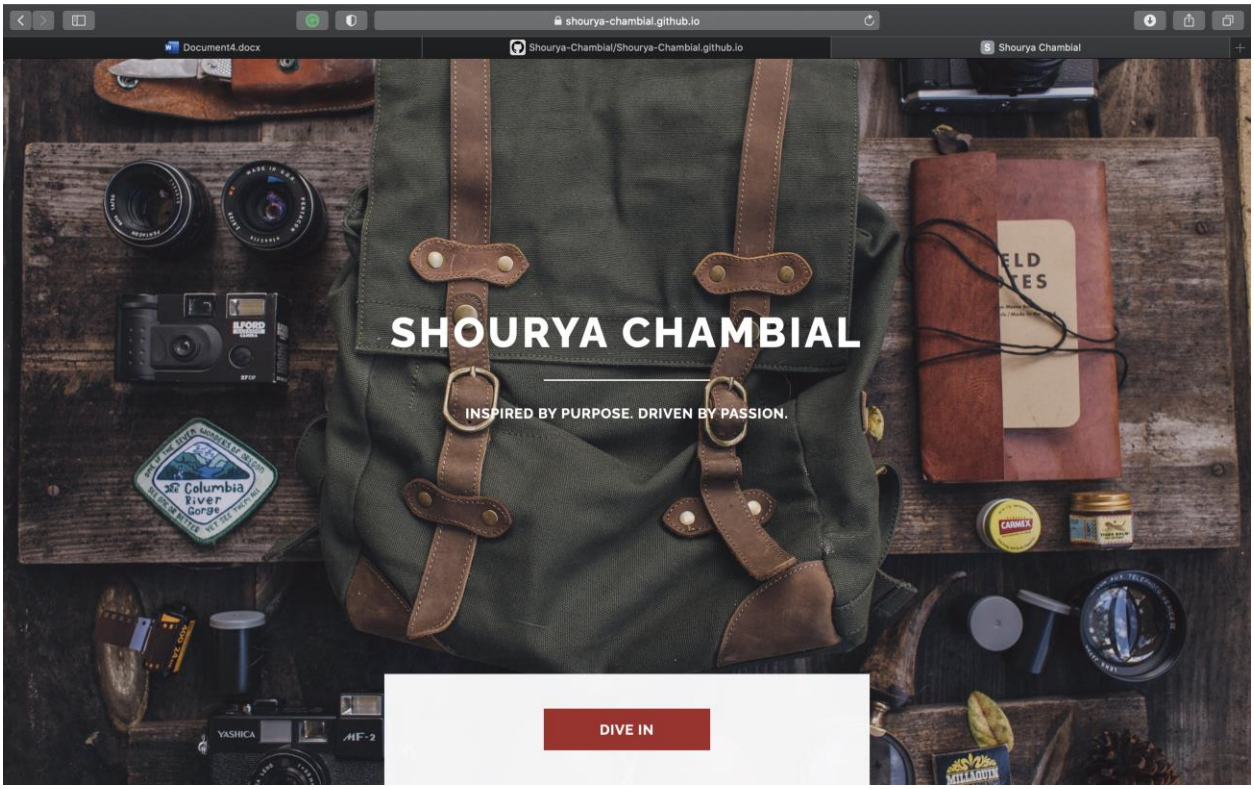
The GitHub repository link is as follows: -

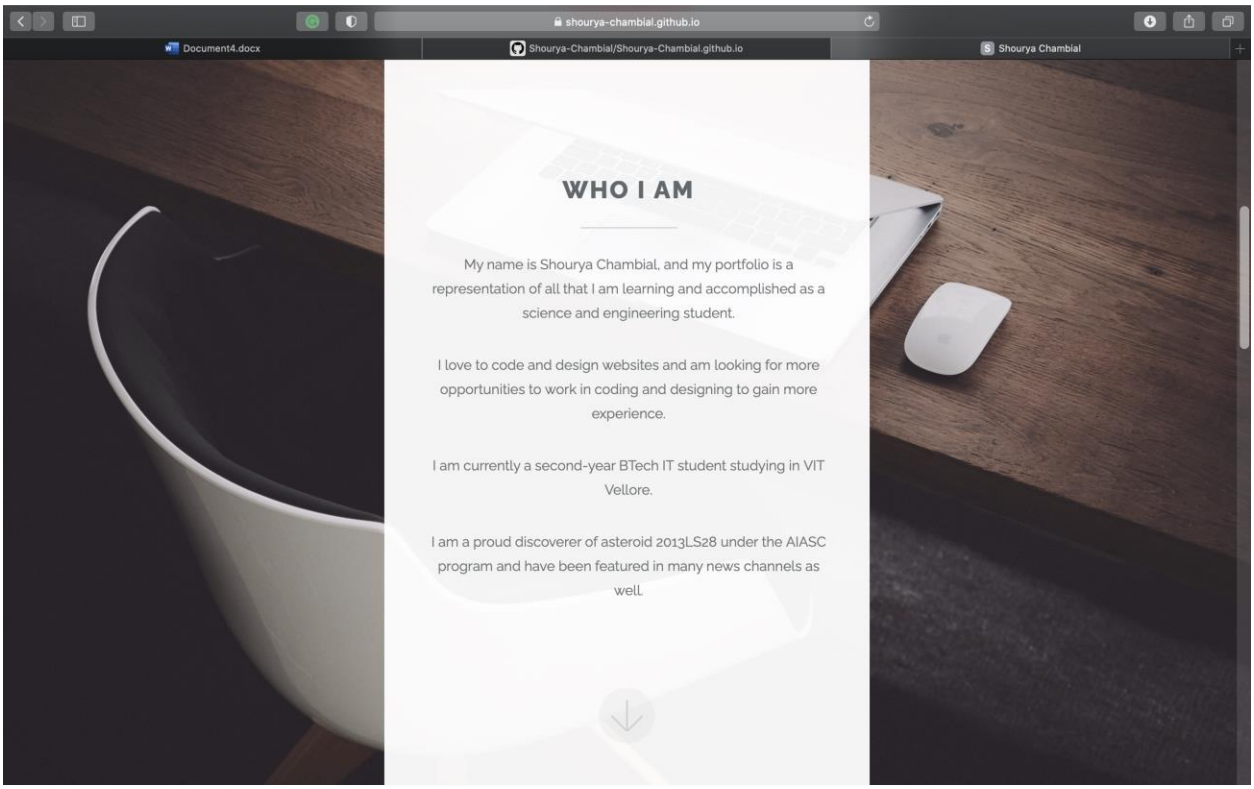
https://github.com/Shourya-Chambial/Shourya-Chambial.github.io

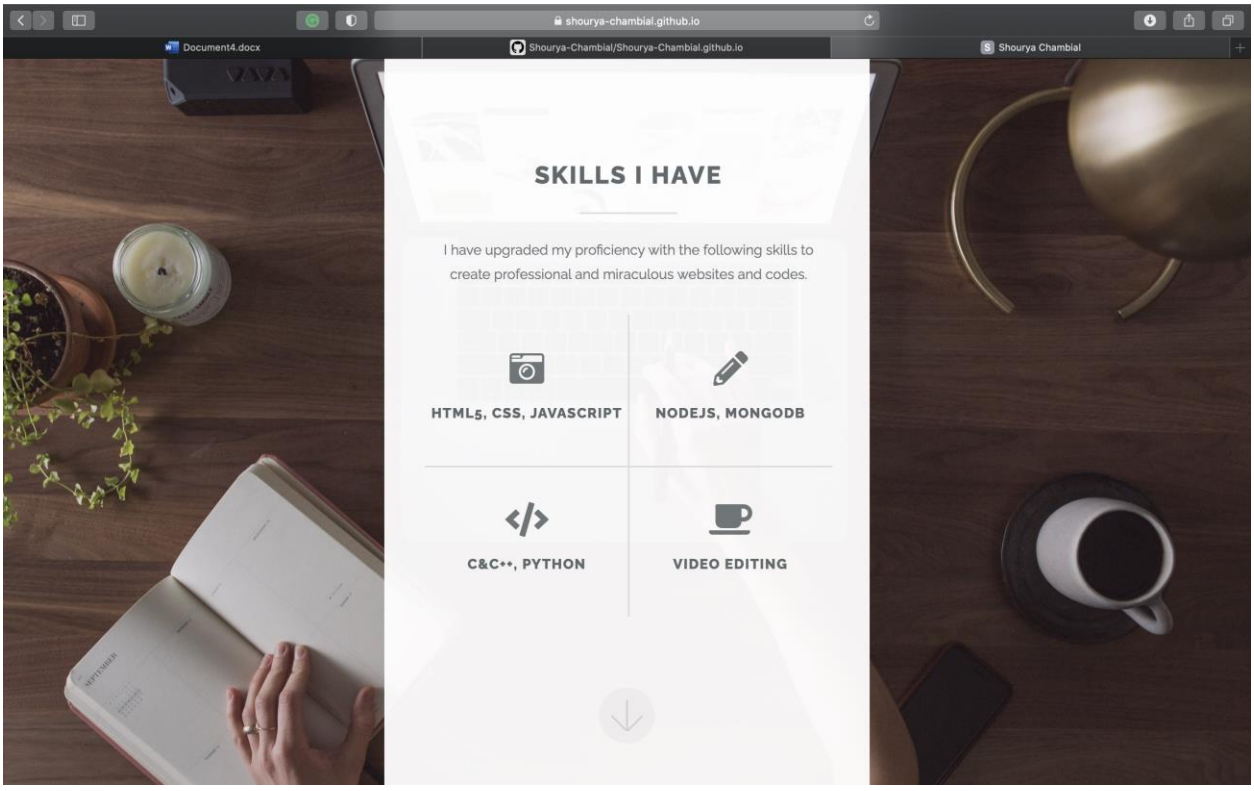
The link for the hosted personal portfolio is as follows: -

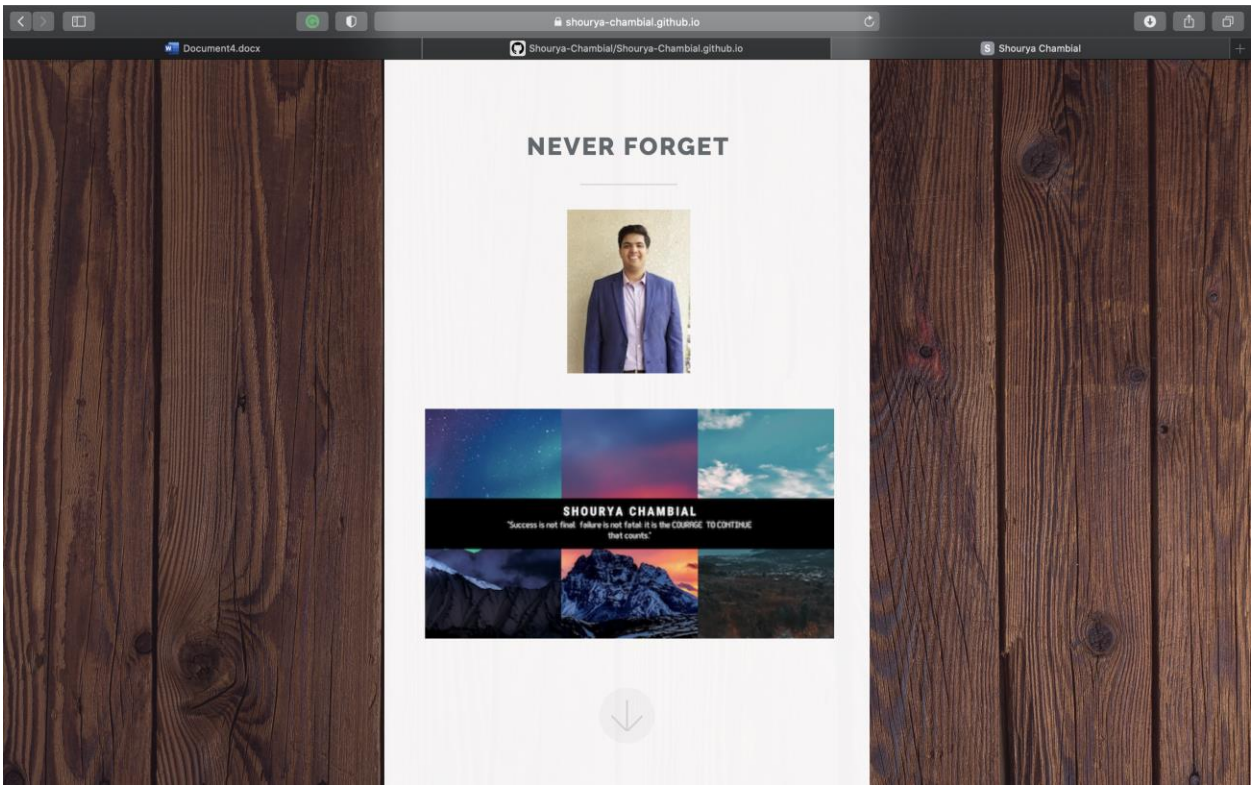
https://shourya-chambial.github.io/19BIT0246/

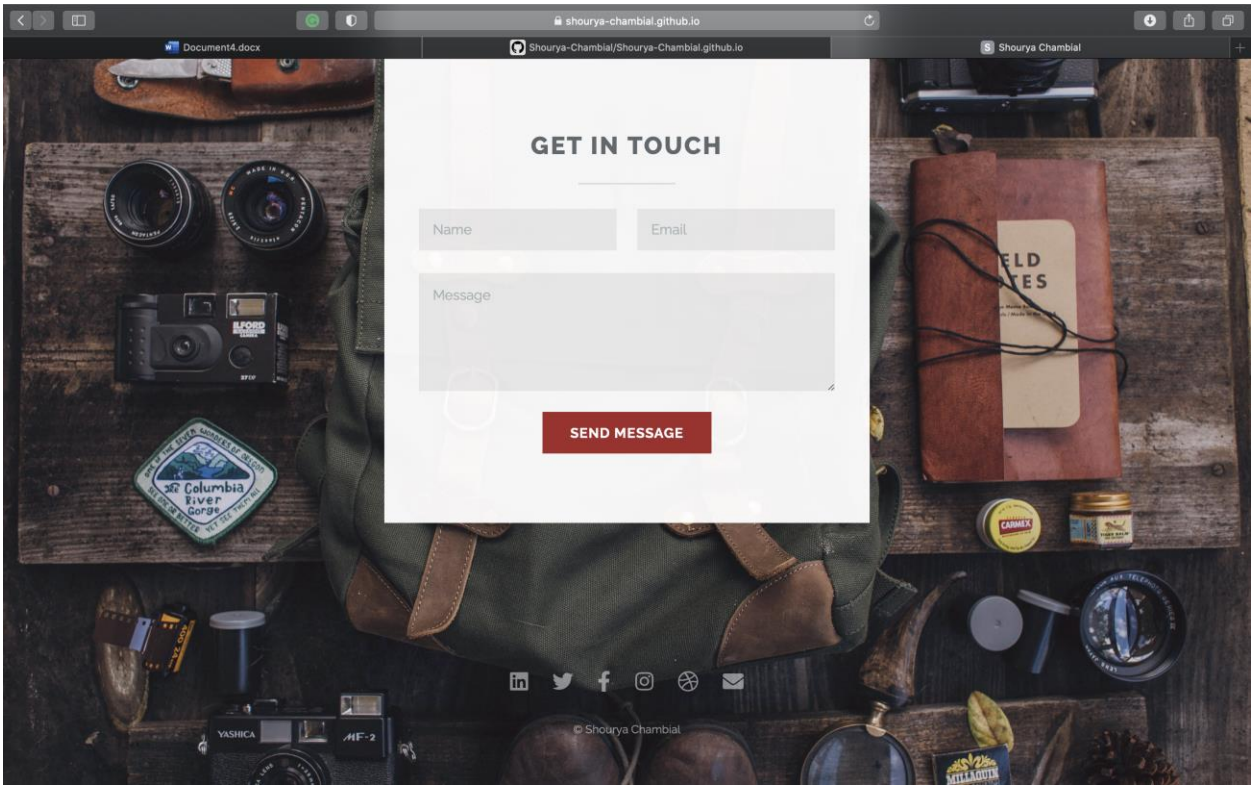
Project Screenshots: -



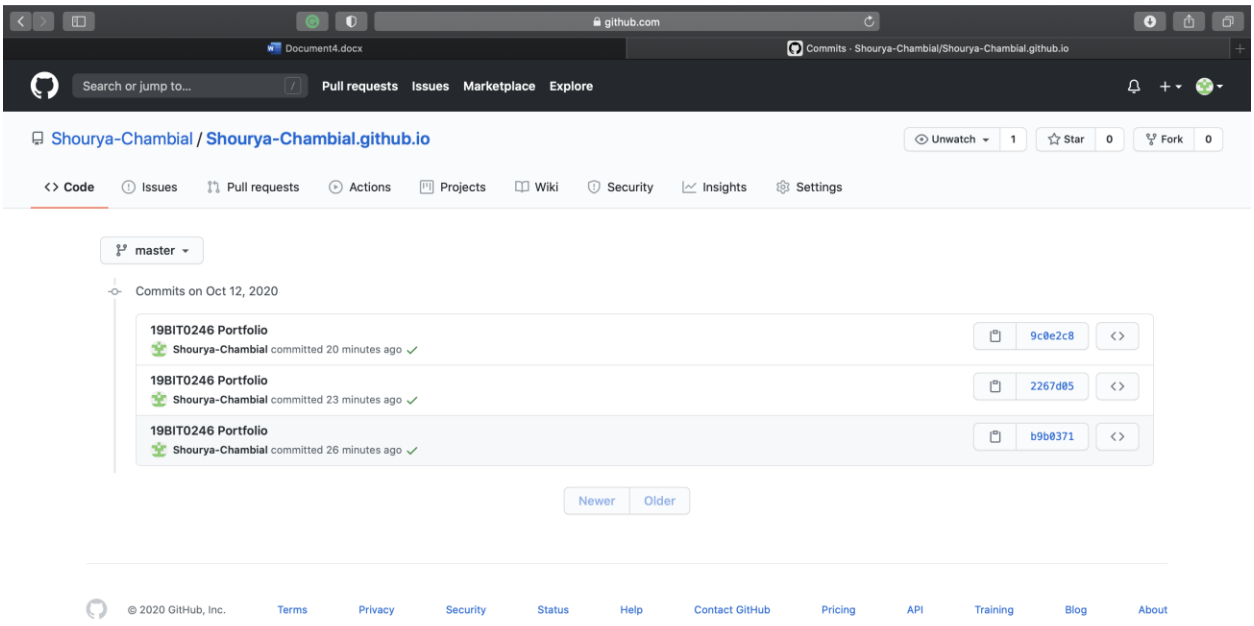

shourya-chambial.github.io
Document4.docx
Shourya-Chambial/Shourya-Chambial.github.io
Shourya Chambial
WHO I AM
My name is Shourya Chambial, and my portfolio is a representation of all that I am learning and accomplished as a science and engineering student.
I love to code and design websites and am looking for more opportunities to work in coding and designing to gain more experience.
I am currently a second-year BTech IT student studying in VIT Vellore.
I am a proud discoverer of asteroid 2013LS28 under the AIASC program and have been featured in many news channels as well.

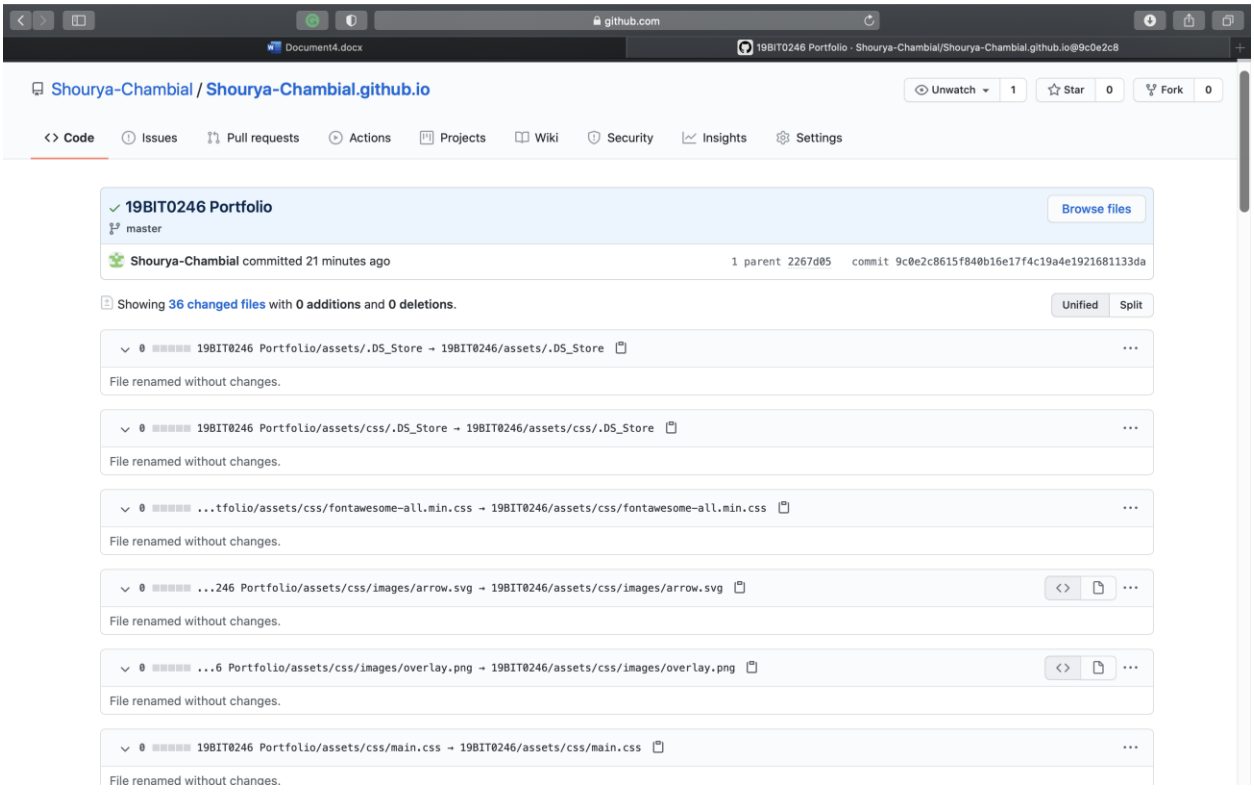

shourya-chambial.github.io
Document4.docx
Shourya-Chambial/Shourya-Chambial.github.io
Shourya Chambial
SKILLS I HAVE
I have upgraded my proficiency with the following skills to create professional and miraculous websites and codes.
HTML5, CSS, JAVASCRIPT
NODEJS, MONGODB
C&C++, PYTHON
VIDEO EDITING

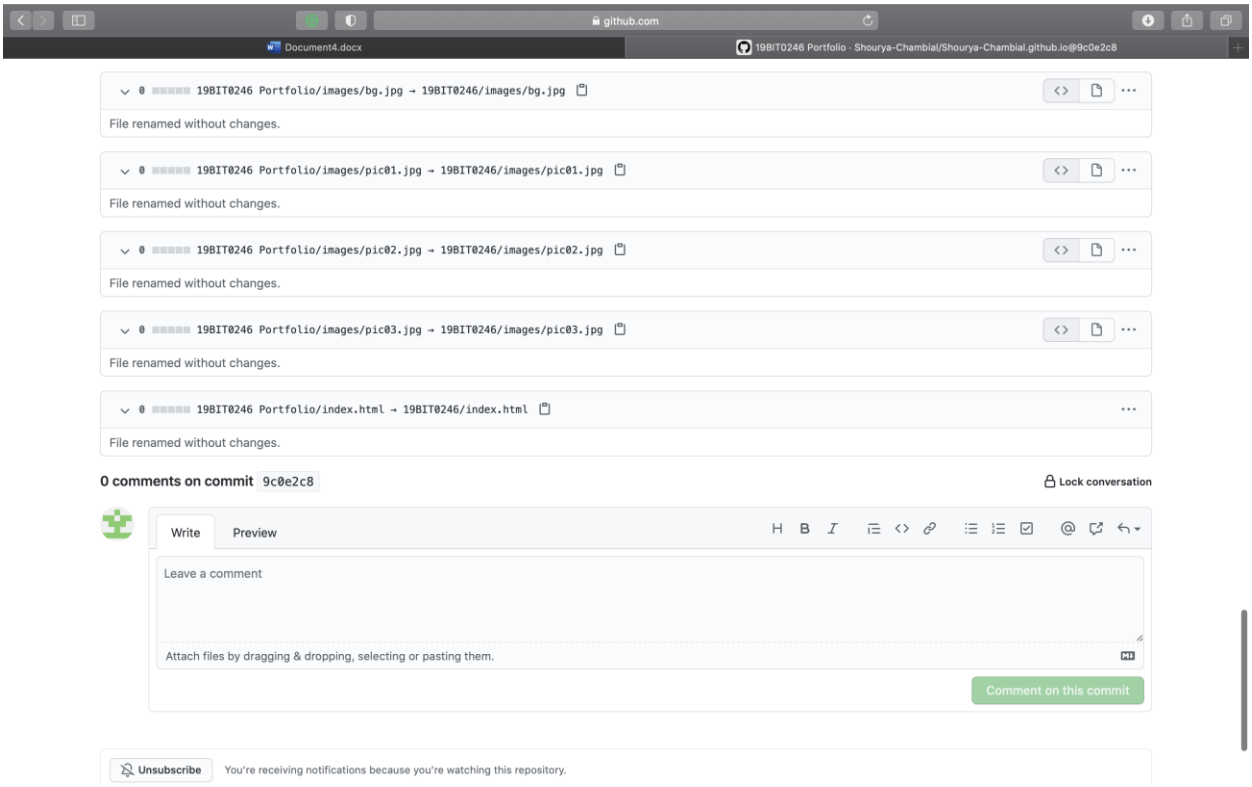




Version History Screenshots: -


github.com
Document4.docx
Commits · Shourya-Chambial/Shourya-Chambial.github.io
Search or jump to...
Pull requests
Issues
Marketplace
Explore
Shourya-Chambial / Shourya-Chambial.github.io
Unwatch 1
Star 0
Fork 0
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
master
Commits on Oct 12, 2020
19BIT0246 Portfolio
Shourya-Chambial committed 20 minutes ago
9c0e2c8
19BIT0246 Portfolio
Shourya-Chambial committed 23 minutes ago
2267d05
19BIT0246 Portfolio
Shourya-Chambial committed 26 minutes ago
b9b0371
Newer
Older
© 2020 GitHub, Inc.
Terms
Privacy
Security
Status
Help
Contact GitHub
Pricing
API
Training
Blog
About


github.com
Document4.docx
19BIT0246 Portfolio · Shourya-Chambial/Shourya-Chambial.github.io@9c0e2c8
Shourya-Chambial / Shourya-Chambial.github.io
Unwatch 1
Star 0
Fork 0
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
19BIT0246 Portfolio
master
Browse files
Shourya-Chambial committed 21 minutes ago
1 parent 2267d05
commit 9c0e2c8615f840b16e17f4c19a4e1921681133da
Showing 36 changed files with 0 additions and 0 deletions.
Unified
Split
0 19BIT0246 Portfolio/assets/.DS_Store → 19BIT0246/assets/.DS_Store
File renamed without changes.
0 19BIT0246 Portfolio/assets/css/.DS_Store → 19BIT0246/assets/css/.DS_Store
File renamed without changes.
0 ...tfolio/assets/css/fontawesome-all.min.css → 19BIT0246/assets/css/fontawesome-all.min.css
File renamed without changes.
0 ...246 Portfolio/assets/css/images/arrow.svg → 19BIT0246/assets/css/images/arrow.svg
File renamed without changes.
0 ...6 Portfolio/assets/css/images/overlay.png → 19BIT0246/assets/css/images/overlay.png
File renamed without changes.
0 19BIT0246 Portfolio/assets/css/main.css → 19BIT0246/assets/css/main.css
File renamed without changes.



Pros of GitHub: -

- It is fast: Since most of the operations are preferred locally, it allows huge benefit in terms of speed.
- It provides good backup: Here chance of losing data is very low as it provides the multiple copies of it.
- It's free and it is open source: GitHub is completely free and we can use it without paying and since it is an open source you can download the source code and can make changes as per the requirements.
- Multiple developers can work: GitHub allows multiple developers to work on a single project at a time. It helps all the team members to work together on a single project at a time from different locations.
- It makes it easy to contribute to your open source projects
- Documentation: By using GitHub, you make it easier to get excellent documentation. Their help section and guides have articles for nearly any topic related to git that you can think of.
- Integration options: GitHub can integrate with common platforms such as Amazon and Google Cloud, services such as Code Climate to track your feedback, and can highlight syntax in over 200 different programming languages.

- ❖ Markdown: Markdown allows you to use a simple text editor to write formatted documents. GitHub has revolutionized writing by channeling everything through Markdown: from the issue tracker, user comments, everything. With so many other programming languages to learn for setting up projects, it's really a big benefit to have your content inputted in a format without having to learn yet another system.
- ❖ Track changes in your code across versions: When multiple people collaborate on a project, it's hard to keep track revisions—who changed what, when, and where those files are stored. GitHub takes care of this problem by keeping track of all the changes that have been pushed to the repository. Much like using Microsoft Word or Google Drive, you can have a version history of your code so that previous versions are not lost with every iteration.

Cons of GitHub: -

- ☹ Reviewing large pull requests can be tedious and it can be tough to identify recent changes (e.g. a one line change) in new files or files with lots of changes.
- ☹ It should be a bit harder to push unresolved merge conflicts, we've had these slip through once in a while.
- ☹ You have to be careful with merge operations; a bad merge can be painful to reverse.
- ☹ Potential Drawback: Security GitHub does offer private repositories, but this isn't necessarily perfect for many. For high value intellectual property, you're putting all of this in the hands of GitHub as well as anyone who has a login, which like many sites has had security breaches before and is targeted constantly. It is often better than nothing, but it's not perfect. In addition, some clients/employers will only allow code on their own secure internal Git as a matter of policy.
- ☹ GitHub packs a lot of functionality into its website; sometimes it can be difficult to navigate to the correct sub-page; the various drop-down selection boxes and sidebar menus can be confusing to users, especially ones with lightweight experience with the interface.

Comparison between three version control applications: -

GitHub: -

- It is a social repository for open source code projects that use Git for source code versions control.
- This host file is distributed file version management systems.
- The mission of GitHub is to make development process quick and captivating, especially when few individuals simultaneously are working on the same task.
- Apart from being a good tool, GitHub is a large educational source.
- This service includes more than 21 million of code hosts.

- Benefits include: -

★ Bug tracking
★ Find your projects easily
★ Meet new developers
★ Share your experience
★ Coordinate your projects together
★ Integrated code search
★ High compatibility

- Drawbacks include: -

☹ Only premium users can use all GitHub repositories
☹ Restrictions on file size

GitLab: -

- Developed on the basis of Git version control system, GitLab is one of the best web platforms for hosting project source codes
- Although GitLab Is similar to GitHub in terms of functionality, it seems to be a better choice for teamwork than its famous counterpart and GitLab features may be different somewhat

- Benefits include: -

★ Free service
★ Open source code repository
★ Free hosting
★ Files editing in web interface

★ LDAP integration

- Drawbacks include: -

☹ The interface is rather slow
☹ Technical problems with repositories are quite frequent

Bitbucket: -

- Bitbucket can be called as a real worthy competitor to GitHub
- Different operating systems support this platform
- It is based on a source code management mechanism
- You can simultaneously use the code with integrated comments, and also make requests, manage and share your Git repositories
- It is possible to create a flexible deployment model for any team
- Moreover, one can get access to private and public repositories without any difficulties

- Advantages are: -

★ Free for small teams
★ Query management system
★ Authentication via GitHub
★ Integrated JIRA tool
★ Repositories importing
★ Discount for teachers and students

- Disadvantages are: -

☹ Private repositories are free for a team with five members or fewer
☹ The only way to search is to launch repository and find it locally or use external apps
☹ Not so stable as GitHub

Common features between them include: -

✓ Pull request

- ✓ Code review
- ✓ Integrated editing
- ✓ Big tracking
- ✓ Markdown support
- ✓ Two-factor authentication
- ✓ API with extended features
- ✓ Forks/clones of repositories
- ✓ Snippets
- ✓ Integration of third-party developers

Features that are needed to be added to GitHub: -

- ❖ The integrated CI/CD system like GitLab can be added
- ❖ Another feature is file locking. It allows you to prevent some file or directory from being modified.
- ❖ It is easier to filter commits by the commit message present in GitLab which can be added here
- ❖ If you add ?w=1 to the commit's URL, whitespace changes will be ignored. GitLab has a button in UI to do it, but there is no such button in GitHub which can be added

- ❖ GitLab has an amazing button to automatically merge PR when all required checks pass. It is often the case that your PR is approved, you make some final fixes and then you have to wait for CI to pass before you can merge it. This button allows you to set automatic merging and close the tab without having to wait for CI. This can be added.

THANK YOU