



Seminar Report
On
[GitHub]

Prepared by

[Rahil Mehta] [91900103002]

[Kandarp Kotadia] [91900103034]

[Hetaansh Gadhiya] [91900103043]

Guided by

[Prof. Shailendrasinh Chauhan]

Faculty of Technology
Marwadi University, Rajkot

KANDARP KOTADIA [91900103034]

2021-22
CERTIFICATE

This is to certify that the seminar report entitled [**GitHub**] has been carried out by [**Rahil Mehta – 91900103002, Kandarp Kotadia - 91900103034, Hetaansh Gadhiya - 91900103043**] under my guidance in partial fulfilment of the degree of Bachelor of Technology in Computer Engineering/information technology of Marwadi University, Rajkot during the academic year 2020-21.

Date: _____

Internal Guide

Prof. [Shailendrasinh Chauhan]
Assistant Professor

Head of the Department

KANDARP KOTADIA [91900103034]

ACKNOWLEDGEMENT

I would like to express my deep gratitude to professor Shailendrasinh Chauhan, whose contribution in stimulating suggestions and helping me to coordinate my project especially in writing this report and for his patient guidance, enthusiastic encouragement and useful critiques of this research work.

It also helped me in doing a lot of research and I came to know about so many new things. I am really thankful to the professor.

Secondly, I would also like to thank my project partners for their coordination and dedication towards this project and completing this project within the limited time frame.

ABSTRACT

→ What is GitHub?

GitHub is wikipedia for programmers. You can edit files, see who changed what, view old versions of files, and access it from anywhere in the world.

→ What is Git?

Git is an open-source code management system. The basic data idea of git is to keep track of different versions of code or text, so you can easily compare what has changed.

→ Elements of GitHub:

- Repository - The most fundamental element of github, it is essentially a project's folder. It stores every single project file, its documentation and its revision history of every document.
- Commit - Commits are easily one of the most frequented activities by a developer using github. Commit is like 'saving' an updated file to its original folder.
- Clone - Clones are literally copies of repositories that sit on the developer's computer instead of a server. Clones are great as you can download a code file to tinker around offline.
- Branch - A Branch is a parallel version of a repository. it is contained within the repository, but does not affect the primary branch allowing you to work freely without disrupting the live version.
- Fetch - Fetching refers to getting the latest changes from an online repository without merging them in. Once these changes are fetched you can compare them to your local branches.
- Fork - A 'fork' is a personal copy of another user's repository that lives on your github account. Forks allow you to freely make changes to a project without affecting the original.

→ Why GitHub?

GitHub provides a perfect medium for you and the main project's maintainer to communicate. If you've forked a repository, made a great revision to the project, and want it to be recognized by the original developers—maybe even included in the official project you can do so

by creating a pull request. The authors of the original repository can see your work, and then choose whether or not to accept it into the official project or not, when you issue a pull request. If you find a project on github that you'd like to contribute to, you can fork the repo, make the changes you'd like, and release the revised project as a new repo.

The social networking aspect of github is probably its most powerful feature, allowing projects to grow more than just about any of the other features offered. Each user on github has their own profile that acts like a resume of sorts, showing your past work and contributions to other projects via pull requests.

push - issues - merging – blame:s

- Push refers to sending your committed changes to a remote repository, if you change something locally, you'd want to then push those changes so that others may access them.
- Issues are suggested improvements, tasks or questions related to the repo. Issues can be created by anyone and are moderated by repo collaborators.
- Merging takes the changes from one branch and applies them to another. A merge can be done automatically via a pull request or can be done via the command line.
- Blame feature in git describes the last modification to each of a file, which generally displays the revision, author and time.

Index

Chapter 1: Introduction to GitHub

Chapter 2: Steps on how to use GitHub

Chapter 3: Applications of GitHub

Chapter 4: Advantages & Disadvantages of GitHub

Chapter 5: Conclusion

INTRODUCTION TO GITHUB

GitHub is one of the world's largest community of developers. It's an intricate platform that fosters collaboration and communication between developers. GitHub has a number of useful features that enable development teams to work together on the same project and easily create new versions of software without disrupting the current versions.

GitHub is a Git repository hosting service, but it adds many of its own features. While Git is a command line tool, GitHub provides a Web-based graphical interface. It also provides access control and several collaboration features, such as a wikis and basic task management tools for every project.

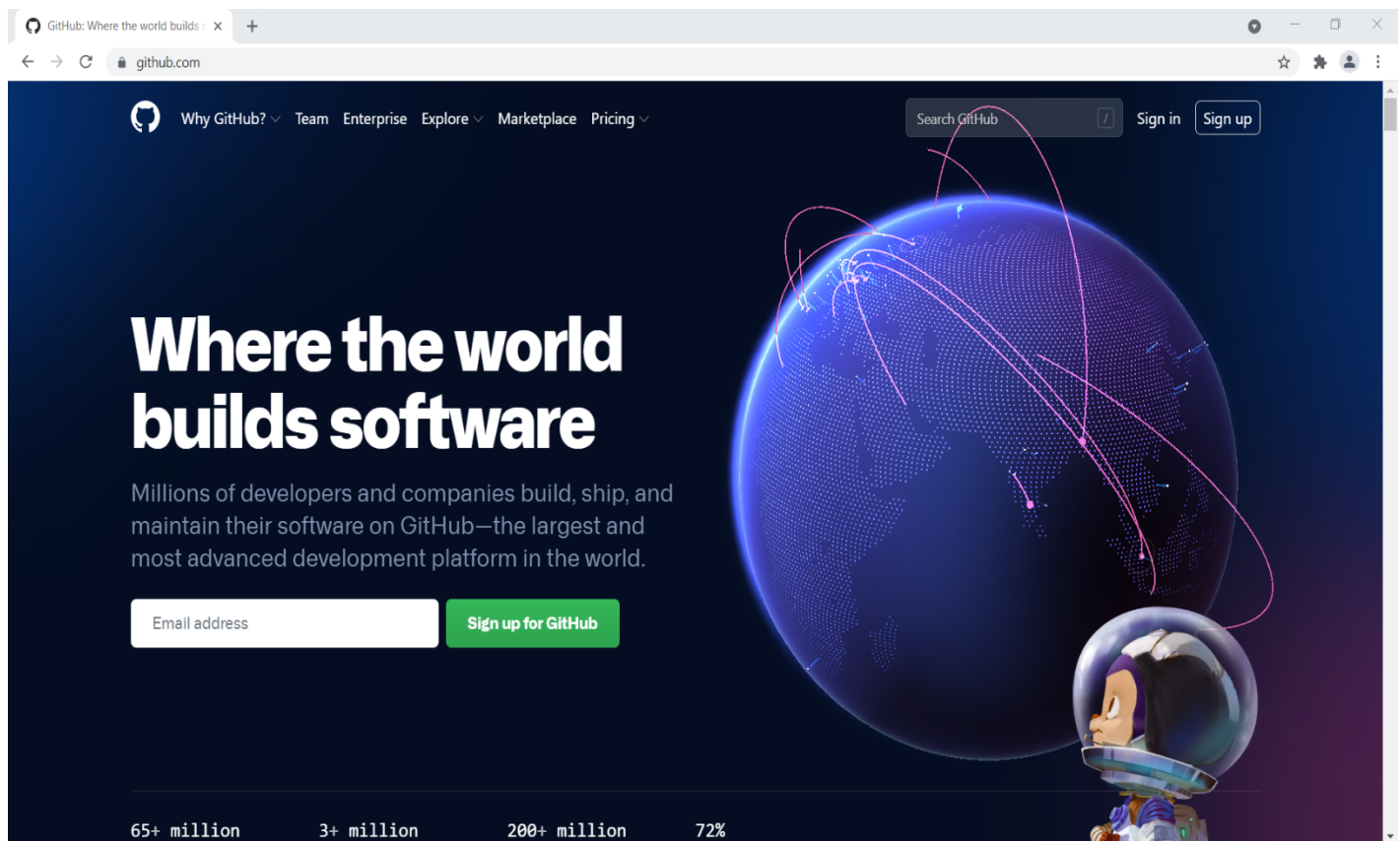
- **GitHub** is a **code hosting platform** for **version control** and collaboration.
- GitHub essentials are:
 - **Repositories:** A GitHub repository can be used to store a development project. It can contain folders and any type of files (HTML, CSS, JavaScript, Documents, Data, Images). A GitHub repository should also include a license file and a README file about the project. A GitHub repository can also be used to store ideas, or any resources that you want to share.
 - **Branches:** A GitHub branch is used to work with different versions of a repository at the same time. By default, a repository has a master branch (a production branch). Any other branch is a copy of the master branch (as it was at a point in time). New Branches are for bug fixes and feature work separate from the master branch. When changes are ready, they can be merged into the master branch. If you make changes to the master branch while working on a new branch, these updates can be pulled in.
 - **Commits:** At GitHub, changes are called commits. Each commit (change) has a description explaining why a change was made.
 - **Pull Requests:** Pull Requests are the heart of GitHub collaboration. With a pull request you are proposing that your changes should be merged (pulled in) with the master. Pull requests show content differences, changes, additions, and subtractions in colors (green and red). As soon as you have a commit, you can open a pull request and start a discussion, even before the code is finished. You merge any changes into the master by clicking a "Merge pull request" button. After merging you can delete the branch by clicking a "Delete branch button".
 - Git (the version control software GitHub is built on)

STEPS ON HOW TO USE GITHUB

GitHub is complex, but understanding a few basics will help us get started. Following are some steps we need to follow in order to use GitHub:

1. Sign up for GitHub

In order to use GitHub, we'll first need a GitHub account by going to <https://github.com>. We can create a free GitHub account and start using GitHub right away. With a free account, we'll get access to unlimited public and private repositories. We'll also get bug tracking and project management features. The only downside is that we'll only be allowed three collaborators for private repositories.

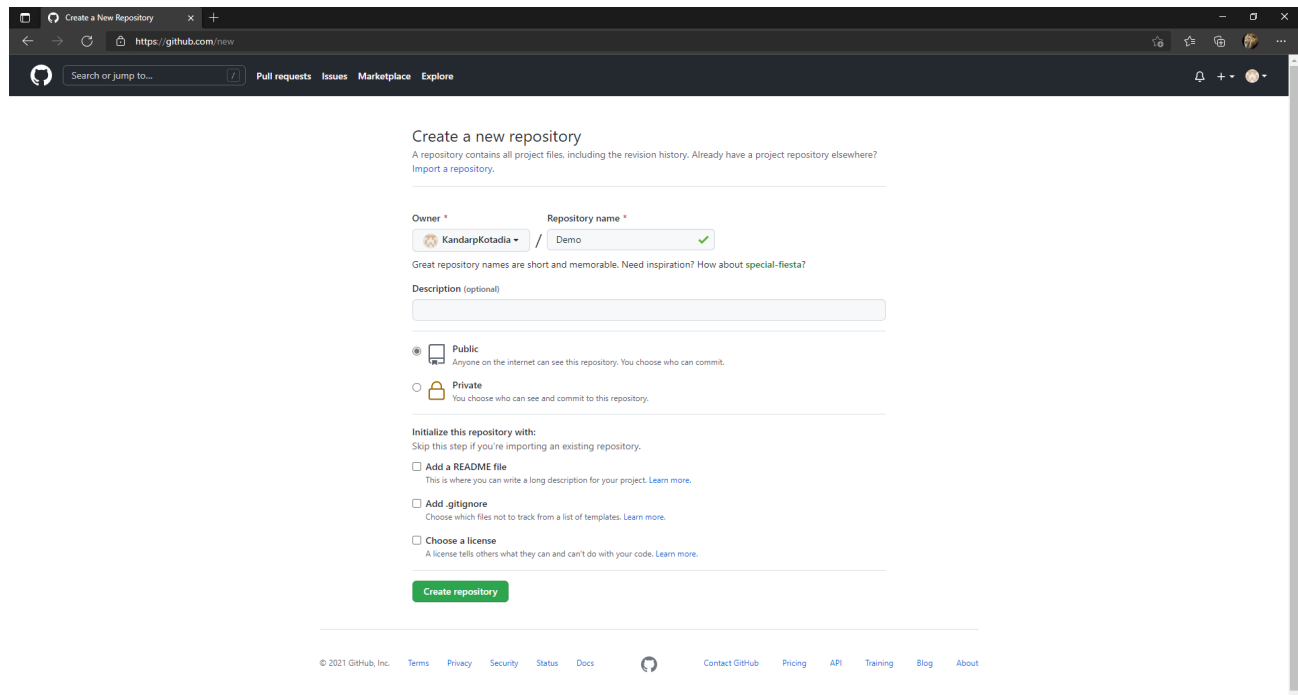


2. Install Git

GitHub runs on Git. Git is a version-control system (is a system that records all the modifications made to a file or set of data so that a specific version may be called up later if needed. The system makes sure that all the team members are working on the file's latest version, and everyone can work simultaneously on the same project) used for tracking changes in computer files, making it a top-rated utility for programmers world-wide. Git can handle projects of any size. Git can be downloaded from (<https://git-scm.com>).

3. Create Repository



To do anything in GitHub, we'll first need to know how to first start a repository. A repository (or repo) is essentially synonymous with the word "project." Quite simply, a repository stores everything pertinent to a specific project including files, images, spreadsheets, data sets, and videos, often sorted into files. It's best to include a README file within your repository that has specific information regarding the given project. On GitHub, you can add a README file right when you create a new repository. To create a new repository, you'll hit "new repository" in the upper-right-hand corner. You can then name your repository, include a brief description, and check the box that says "initialize this repository with a README." Finally, click "create repository."



Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere?
[Import a repository.](#)


Owner ^{*} Repository name ^{*}

 KandarpKotadia / Demo 

Great repository names are short and memorable. Need inspiration? How about [special-fiesta](#)?

Description (optional)

☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

Initialize this repository with:


Skip this step if you're importing an existing repository.

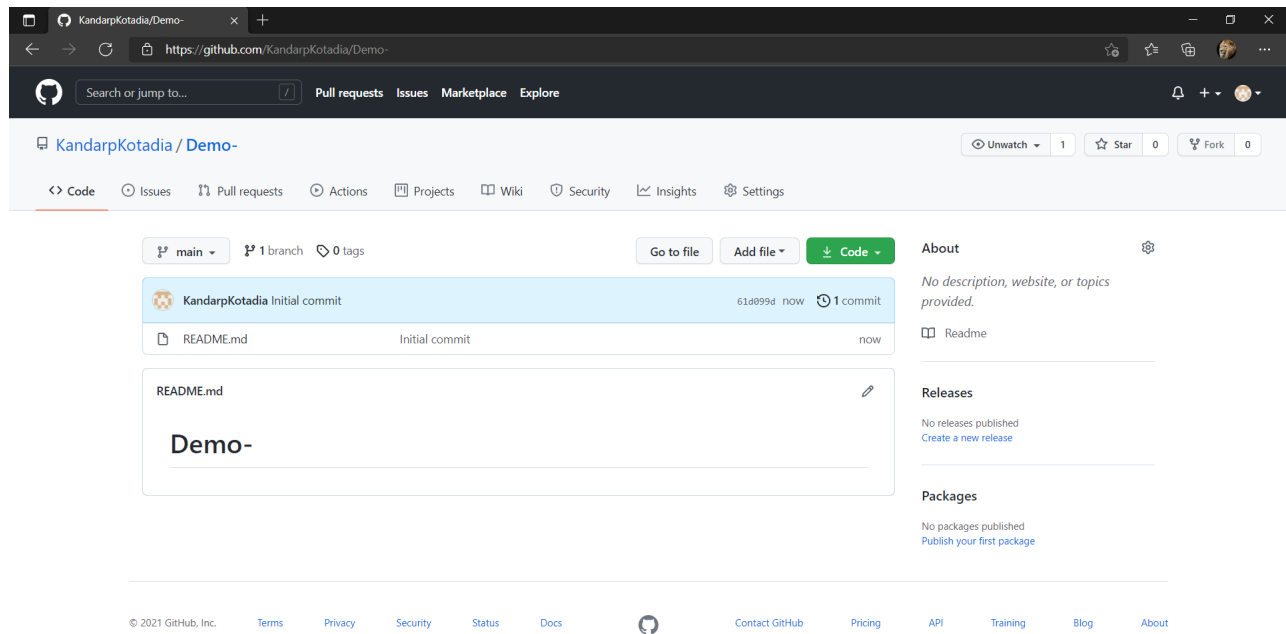
☐ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

☐ **Add .gitignore**
Choose which files not to track from a list of templates. [Learn more.](#)

☐ **Choose a license**
A license tells others what they can and can't do with your code. [Learn more.](#)

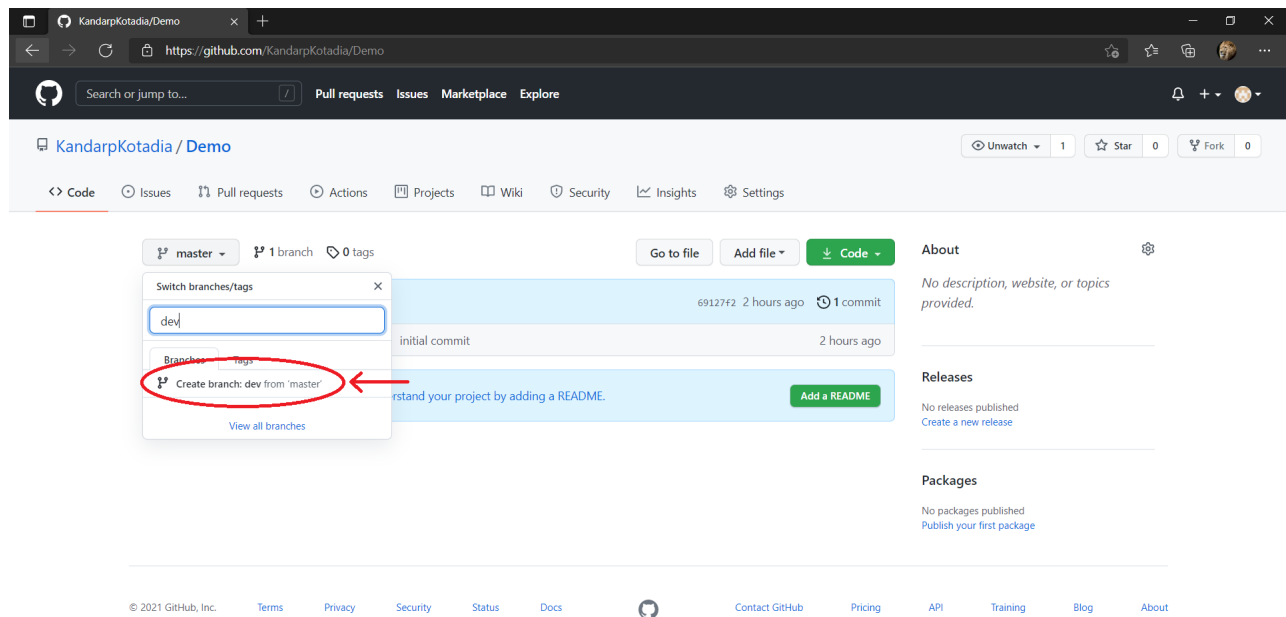
[Create repository](#)

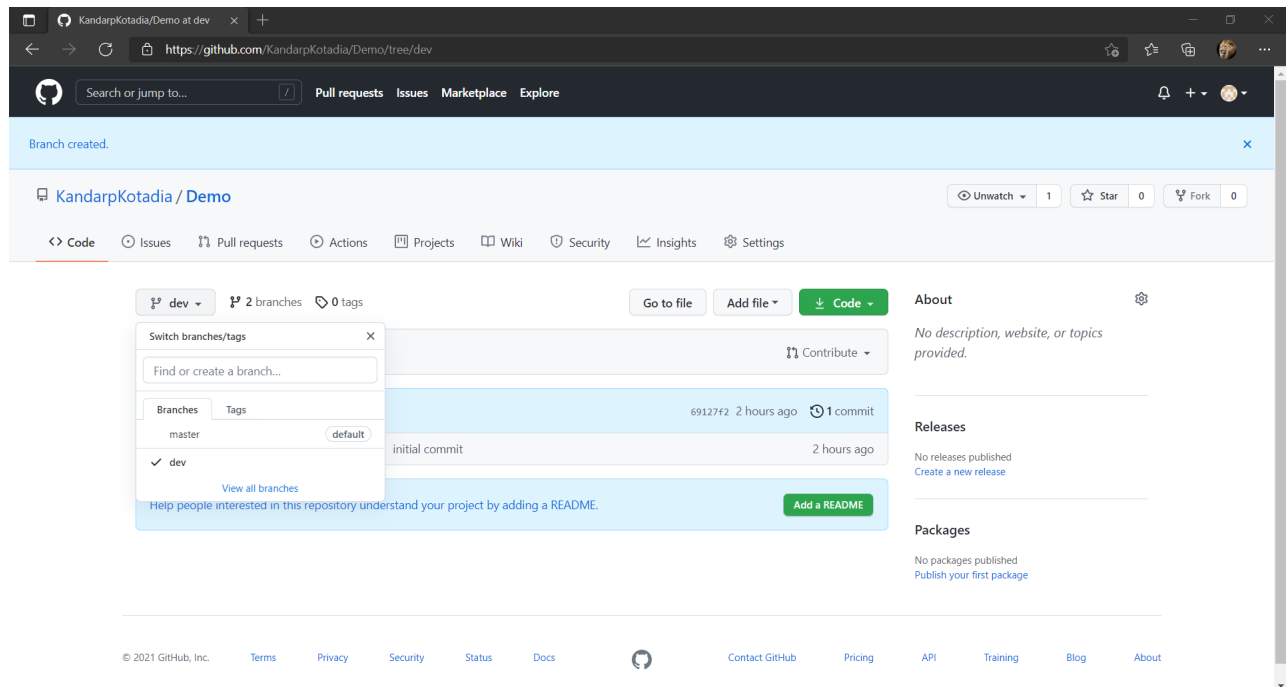
© 2021 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#)  [Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [About](#)



4. Create a Branch

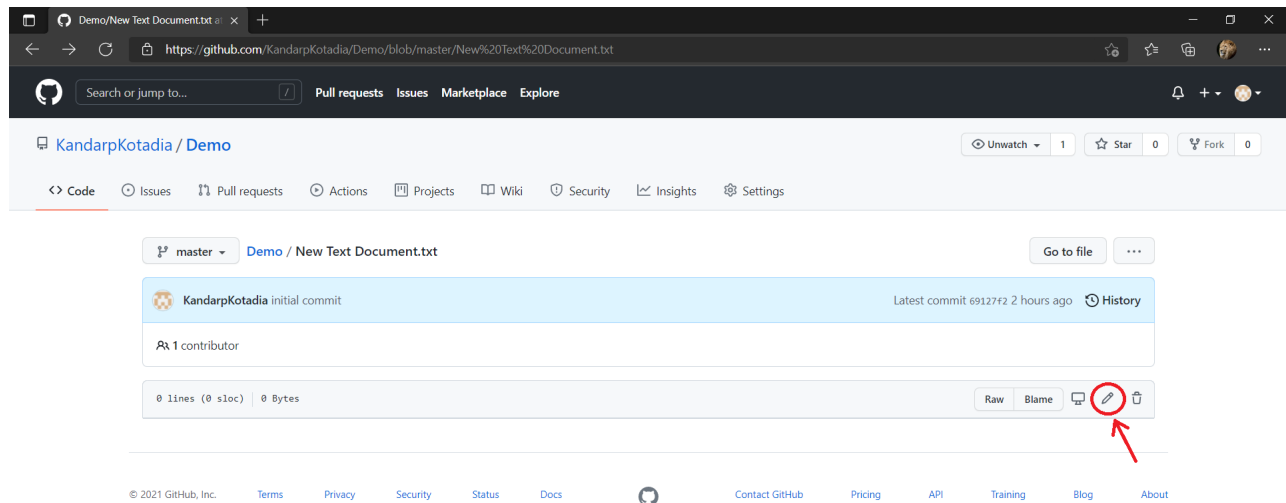
Projects are multi-faceted and many program versions are required when you're building. Branching enables you to edit multiple unique versions of a repository at once. Your repository automatically has a definitive branch called **master**. You can work on several different branches in order to make edits before eventually committing them to the master branch.





5. Create and Commit changes to the Branch

To make changes to a branch in GitHub, go to the code view for your newly created branch. Click the file you want to change, then hit the pencil icon in the upper right, make any necessary edits, describe your changes by writing a commit message, and then click “commit changes.” Each saved change is called a commit. Every individual commit has its own commit message which gives more details into why a specific change was done. The commit messages give a history of changes and help project contributors understand how the project has changed over time.



Editing Demo/New Text Document.txt x +

https://github.com/KandarpKotadia/Demo/edit/master/New%20Text%20Document.txt

Search or jump to... Pull requests Issues Marketplace Explore


KandarpKotadia / Demo Unwatch 1 Star 0 Fork 0

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Demo / New Text Document.txt in master Cancel changes

<> Edit file Preview changes Spaces 2 No wrap

1 Hello, This id for demo.

 Commit changes


Create New Text Document.txt

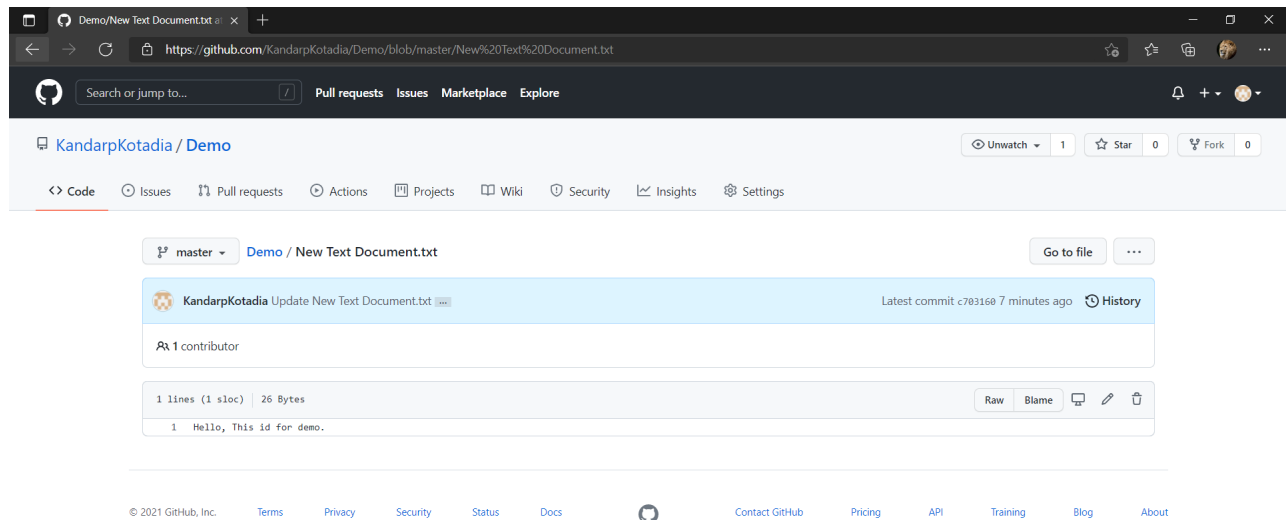
Added a description.

☒ Commit directly to the master branch.

☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes Cancel

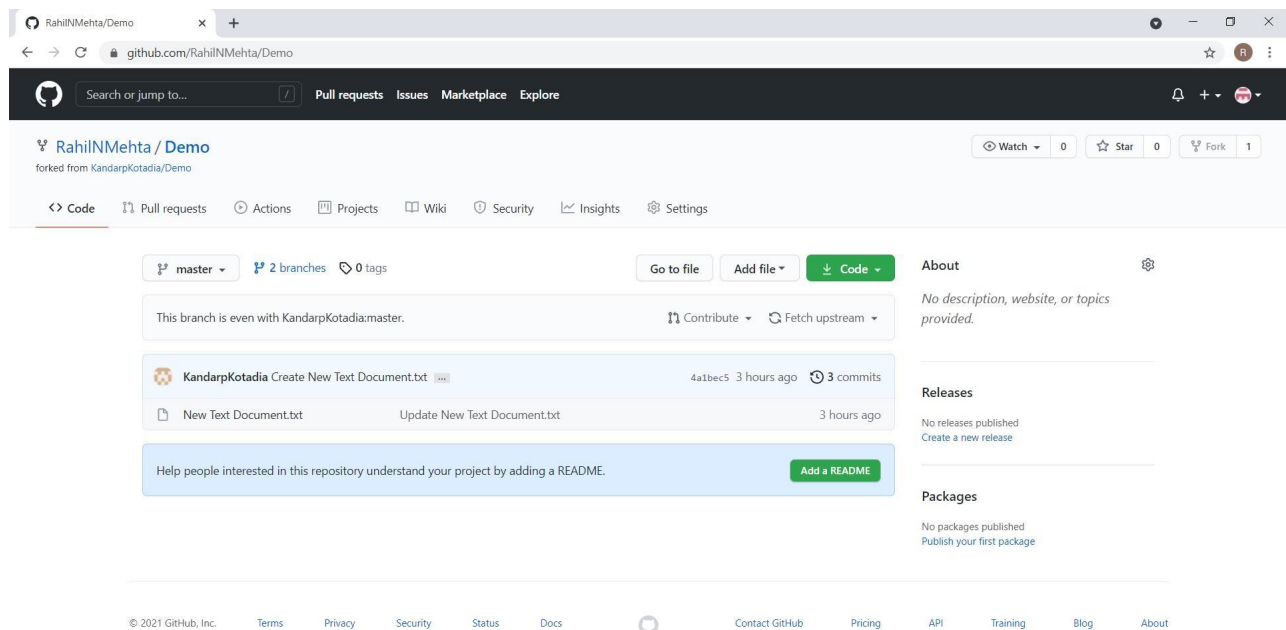
© 2021 GitHub, Inc. Terms Privacy Security Status Docs  Contact GitHub Pricing API Training Blog About

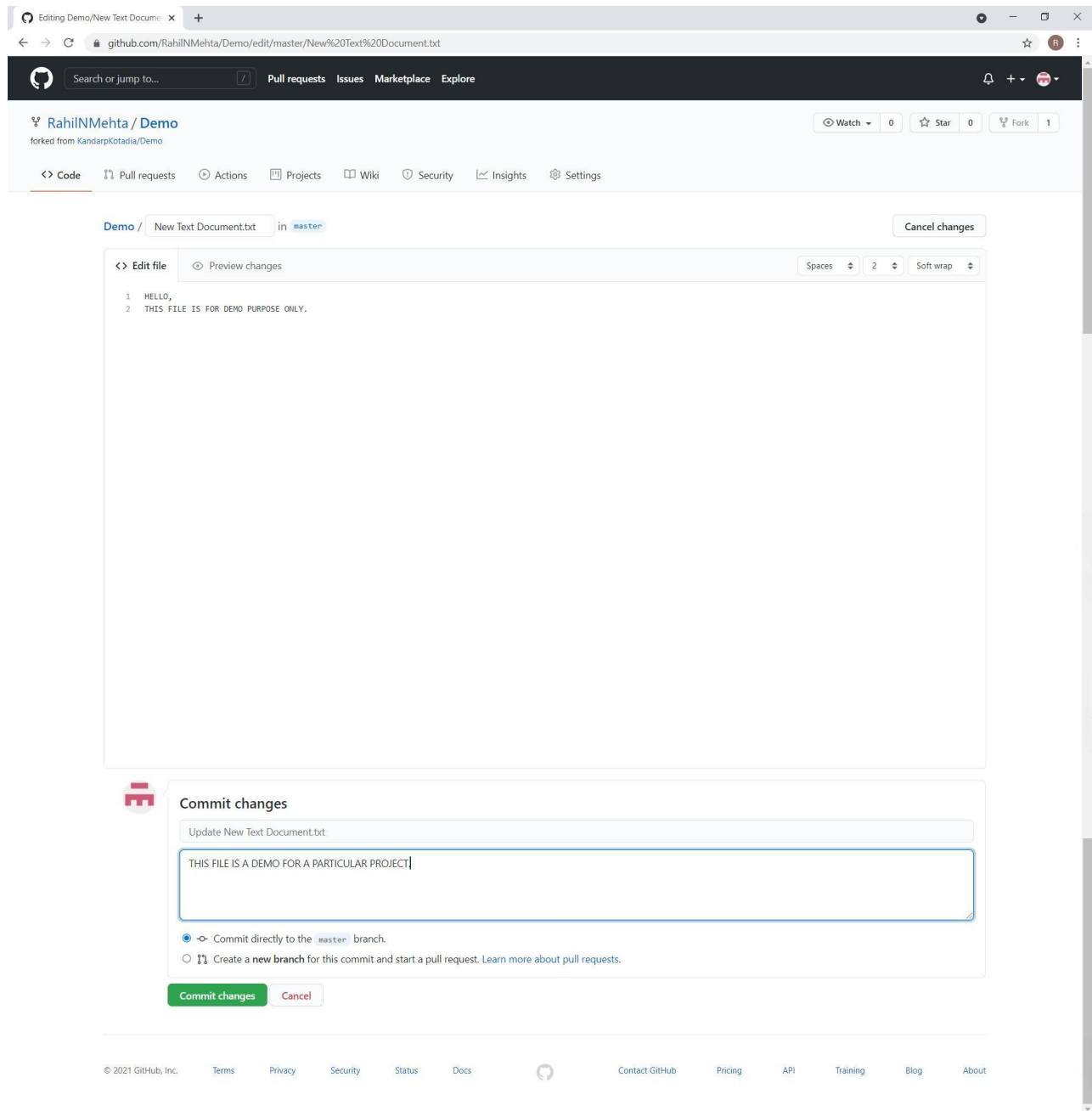


6. Open a pull request

In order for any branch to be merged into another person's branch, you must open a pull request. A pull request is GitHub's way of notifying relevant parties about your request to incorporate changes into their branch. A pull request will show in red and green the differences of the content between branches. You can make a pull request any time you complete a commit. For best results, when sending a pull request, you can use the "@" feature to mention specific people from whom you need feedback.

To open a pull request, you'll go to the "pull request" tab and hit the button that says "new pull request." Next, in the "example comparisons" box, find the branch you made and compare it with the master. Ensure you like the changes and then click the "create pull request" button. Title your pull request and briefly describe the changes. To finish, click "create pull request."





7. Merge your pull request

Merging your pull request with the master branch is something you may need to pass on to your superiors to handle. For the sake of learning, though, you can practice doing it yourself. Simply hit the button that says “merge pull request,” select “confirm merge,” and then delete the branch you merged once it has been incorporated into the master.

Browser window showing a GitHub repository page for **RahilNMehta / Demo**. The page displays the file **New Text Document.txt** with the following content:

```
1 HELLO,  
2 THIS FILE IS FOR DEMO PURPOSE ONLY.
```

The page also shows the commit history, contributors, and navigation links.

Browser window showing a GitHub pull request page for **KandarpKotadia / Demo**. The page displays the comparison between the **base repository: KandarpKotadia/Demo** and the **head repository: RahilNMehta/Demo**. The pull request is titled **Update New Text Document.txt** and is marked as **Verified**.

The pull request shows 1 commit, 1 file changed, 0 comments, and 1 contributor. The changes are summarized as **Showing 1 changed file with 2 additions and 1 deletion.**

The diff view shows the following changes in **New Text Document.txt**:

```
@@ -1,2 @@  
1 - Hello, This id for demo.  
1 + HELLO,  
2 + THIS FILE IS FOR DEMO PURPOSE ONLY.
```

The page also displays the commit history and navigation links.

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).



base repository: KandarpKotadia/Demo


base: master

←

head repository: RahilNMehta/Demo

compare: master










✓ **Able to merge.** These branches can be automatically merged.



Update New Text Document.txt


Write

Preview

H B I         


THIS FILE IS A DEMO FOR A PARTICULAR PROJECT.


Attach files by dragging & dropping, selecting or pasting them.


☒ Allow edits by maintainers 


Create pull request


Helpful resources
[GitHub Community Guidelines](#)


 Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).




 1 commit

 1 file changed


 0 comments

 1 contributor



 Commits on Jul 01, 2021

  Update New Text Document.txt 

Verified b4fd176

 Showing 1 changed file with 2 additions and 1 deletion.

Unified Split

3  New Text Document.txt 

... .. @@ -1 +1,2 @@

1 - Hello, This id for demo.

1 + HELLO,

2 + THIS FILE IS FOR DEMO PURPOSE ONLY.

—

No commit comments for this range

KANDARP KOTADIA [91900103034]

Update New Text Document.txt x

github.com/KandarpKotadia/Demo/pull/1

Search or jump to... Pull requests Issues Marketplace Explore

KandarpKotadia / Demo

Watch 1 Star 0 Fork 1

< Code Issues Pull requests Actions Projects Wiki Security Insights

Update New Text Document.txt #1

Open

RahilNMehta wants to merge 1 commit into [KandarpKotadia:master](#) from [RahilNMehta:master](#)

Conversation Commits Checks Files changed +2 -1

RahilNMehta commented 2 minutes ago

THIS FILE IS A DEMO FOR A PARTICULAR PROJECT.

update New Text Document.txt

Verified b4fd176

Add more commits by pushing to the [master](#) branch on [RahilNMehta/Demo](#).

This branch has no conflicts with the base branch

Only those with [write access](#) to this repository can merge pull requests.

Write Preview

H B I

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Close pull request

Comment

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

ProTip! Add comments to specific lines under [Files changed](#).

Reviews

No reviews

Still in progress? Convert to draft

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Linked issues

Successfully merging this pull request may close these issues.

None yet

Notifications

Customize

Unsubscribe

You're receiving notifications because you authored the thread.

KANDARP KOTADIA [91900103034]

Update New Text Document.txt | x +

https://github.com/KandarpKotadia/Demo/pull/1

Search or jump to... Pull requests Issues Marketplace Explore

KandarpKotadia / Demo

Unwatch 1 Star 0 Fork 1

<> Code Issues Pull requests 1 Actions Projects Wiki Security Insights Settings

Update New Text Document.txt #1

Edit Open with

Merged KandarpKotadia merged 1 commit into kandarpKotadia:master from RahilNMehta:master now

Conversation 0 Commits 1 Checks 0 Files changed 1 +2 -1

RahilNMehta commented 20 minutes ago Contributor

THIS FILE IS A DEMO FOR A PARTICULAR PROJECT.

Update New Text Document..txt

Verified b4fd176

Merged KandarpKotadia merged commit 02a2ab6 into kandarpKotadia:master now

Merged Update New Text Document.txt #1 KandarpKotadia merged 1 commit into kandarpKotadia:master from RahilNMehta:master 1 minute ago

Revert

Write Preview

H B I

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Comment

Remember, contributions to this repository should follow our GitHub Community Guidelines.

ProTip! Add comments to specific lines under Files changed.

Reviewers

No reviews

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

No milestone

Linked issues

Successfully merging this pull request may close these issues.

None yet

Notifications

Customize

Unsubscribe

You're receiving notifications because you're watching this repository.

2 participants

Lock conversation

© 2021 GitHub, Inc. Terms Privacy Security Status Docs

Contact GitHub Pricing API Training Blog About

APPLICATIONS OF GITHUB

GitHub, can be divided into the Git, and the Hub. The service includes access controls as well as a number of collaboration features like tools for basic task management and for all projects you handle. GitHub hosts your source code projects in a variety of different programming languages and keeps track of the various changes made to every iteration. So, the “Git” implies the version control system; a tool which allows developers to keep track of the constant revisions to their code. The “Hub” is the community of like-minded individuals who participate. It is all about the collaborative effort of the community, in reviewing, improving, and deriving new ideas from the uploaded code.

- **It makes it easy to contribute to your open source projects**

GitHub is free if your project is open source and includes a wiki and issue tracker that makes it easy to include more in-depth documentation and get feedback about your project.

- **Showcase your work**

GitHub is the best tool you can rely on for attracting recruiters if you are a developer. When searching for new recruits for their project, most companies look into the GitHub profiles.

- **GitHub is a repository**

This means that it allows your work to get out there in front of the public. Moreover, GitHub is one of the largest coding communities around right now, so it's wide exposure for your project.

- **Track changes in your code across versions**

GitHub makes it easy to take care and keep a track of all the changes that have been made to the repository.

ADVANTAGES & DISADVANTAGES OF GITHUB

Advantages:

- **It's Fast & Reliable:** Most of the Operations are preferred locally on GitHub but it doesn't change it's speed; although it allows huge benefits in terms of speed.
- **It's Free & Open-Source:** GitHub is a completely free platform and we can use its services without paying or entering credit/debit card details and it's Open Source as well, so we can download the Source Codes, Programs without any hesitation and can also make changes as per requirement.
- **Back-Up:** GitHub has extremely poor backup services; although there is so less chances of losing data because we can make multiple copies of it.
- **Multi-User Support:** GitHub allows multiple users or developers to work on a project at the same time. It helps all the developers to share ideas and work together from different countries and locations.

Disadvantages:

- **Challenges:** New users often face difficulty finding our way around GitHub due lack of time and experience as we are new to the platform. Sometimes it becomes normal due to the new platform but we're on the focused audience and i.e. developer's. So, we may take time to get a grip with the models, practice and invested time.
- **Security:** GitHub Provides or Offers Private Repositories, but this may not be an option for everyone. For the People or Developers with Intellectual (High Value) Property, they are putting all of their hands into GitHub as well as anyone who has login credentials can do anything, these types of security breaches are constantly targeted over GitHub Security. Except there's no privacy or security concerns over GitHub. It's a Secure Platform.

CONCLUSION

The best programmers are always learning. The world of computer programming is constantly evolving and new problems are being solved with code every day. In order to stay sharp, we need to be consistently discovering new ways to tackle problems. We need to be trying new things and collaborating with other programmers.

GitHub is a great place to not only expand our programming knowledge, but also showcase our best work. github profile is a great place for recruiters to find us and reach out about potential jobs and projects. github has a wealth of features that enable us to collaborate with and learn from the best in the industry.