CHITKARA UNIVERSITY

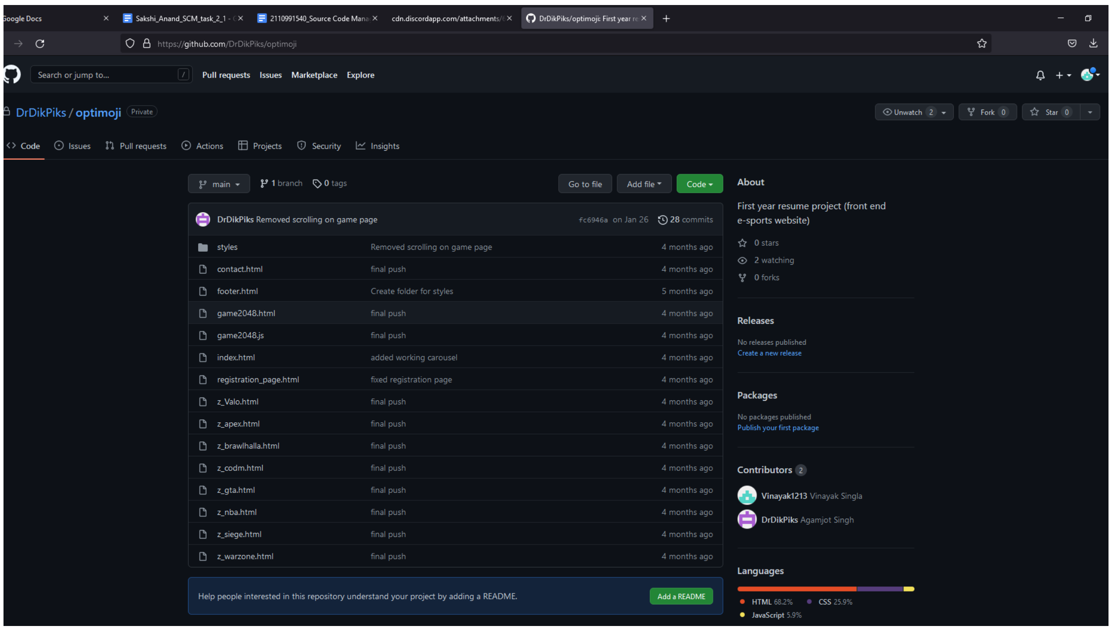
git GitHub

RESOURCES

VERSION CONTROL SYSTEM WITH GIT

Create a distributed Repository and add members in project team

- Login to your GitHub account and you will land on the homepage as shown below. Click on Repositories option in the menu bar.



DrDikPiks Removed scrolling on game page fc6946a on Jan 26 28 commits

styles	Removed scrolling on game page	4 months ago
contact.html	final push	4 months ago
footer.html	Create folder for styles	5 months ago
game2048.html	final push	4 months ago
game2048.js	final push	4 months ago
index.html	added working carousel	4 months ago
registration_page.html	fixed registration page	4 months ago
z_Valo.html	final push	4 months ago
z_apex.html	final push	4 months ago
z_brawlhalla.html	final push	4 months ago
z_codm.html	final push	4 months ago
z_gta.html	final push	4 months ago
z_nba.html	final push	4 months ago
z_siege.html	final push	4 months ago
z_warzone.html	final push	4 months ago

Help people interested in this repository understand your project by adding a README. Add a README

About

First year resume project (front end e-sports website)

0 stars
2 watching
0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

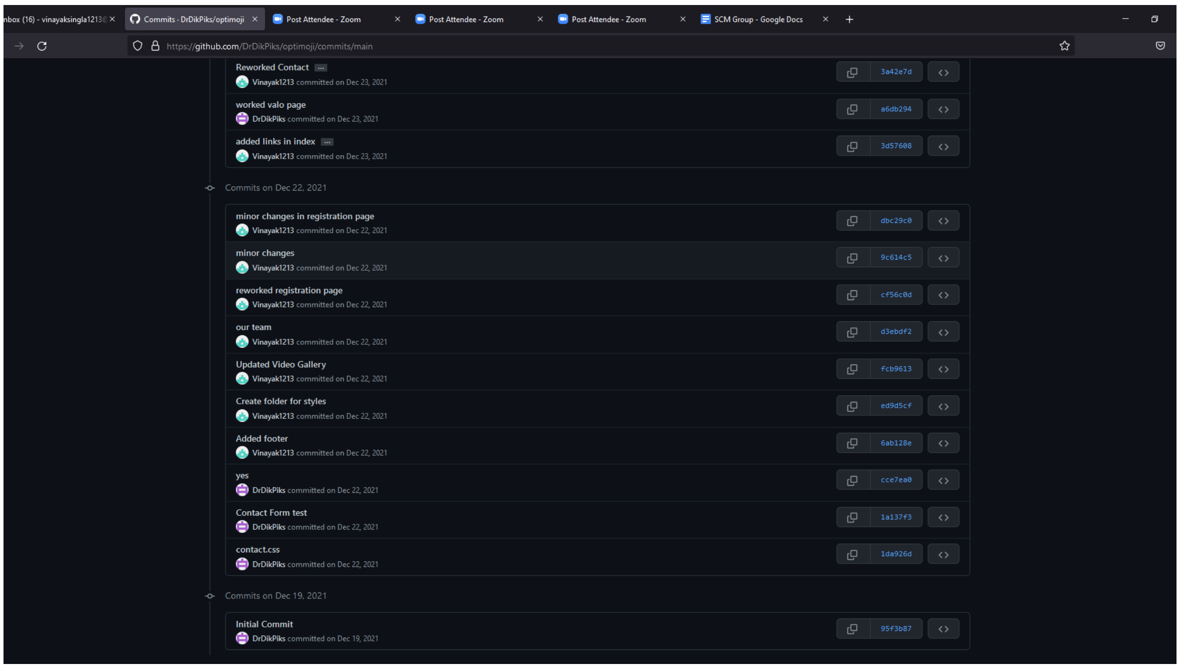
Contributors 2

Vinayak1213 Vinayak Singla
DrDikPiks Agamjot Singh

Languages

● HTML 68.2% ● CSS 25.9%
● JavaScript 5.9%

- Click on the ‘New’ button in the top right corner.


nbox (16) - vinayaksingla1213@
Commits · DrDikPiks/optimoji
Post Attendee - Zoom
Post Attendee - Zoom
Post Attendee - Zoom
SCM Group - Google Docs
https://github.com/DrDikPiks/optimoji/commits/main

Reworked Contact
Vinayak1213 committed on Dec 23, 2021 — 3a42e7d

worked valo page
DrDikPiks committed on Dec 23, 2021 — a6db294

added links in index
Vinayak1213 committed on Dec 23, 2021 — 3d57608

Commits on Dec 22, 2021

minor changes in registration page
Vinayak1213 committed on Dec 22, 2021 — dbc29c0

minor changes
Vinayak1213 committed on Dec 22, 2021 — 9c614c5

reworked registration page
Vinayak1213 committed on Dec 22, 2021 — cf56c0d

our team
Vinayak1213 committed on Dec 22, 2021 — d3ebdf2

Updated Video Gallery
Vinayak1213 committed on Dec 22, 2021 — fcb9613

Create folder for styles
Vinayak1213 committed on Dec 22, 2021 — ed9d5cf

Added footer
Vinayak1213 committed on Dec 22, 2021 — 6ab128e

yes
DrDikPiks committed on Dec 22, 2021 — cce7ea0

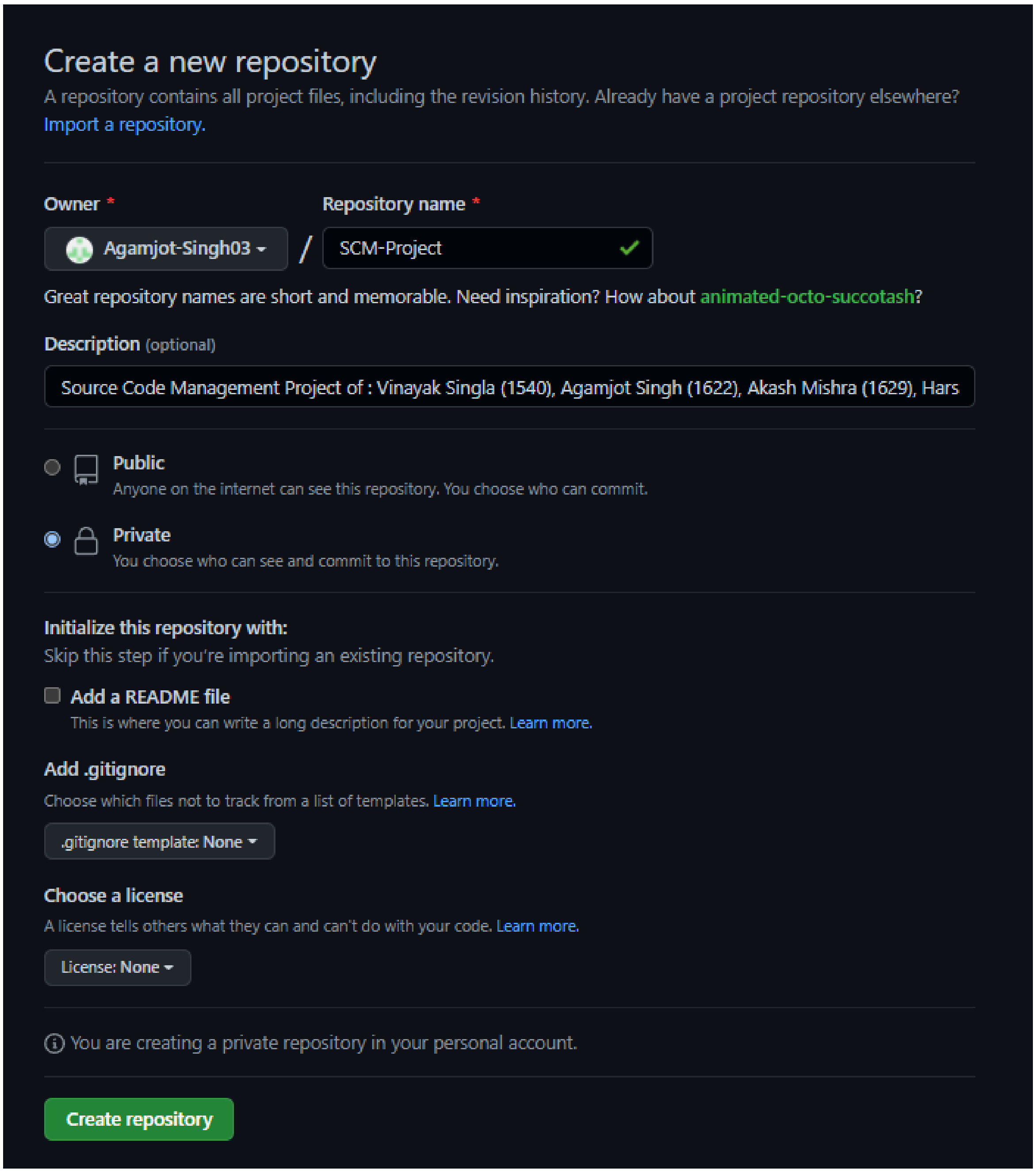
Contact Form test
DrDikPiks committed on Dec 22, 2021 — 1a137f3

contact.css
DrDikPiks committed on Dec 22, 2021 — 1da926d

Commits on Dec 19, 2021

Initial Commit
DrDikPiks committed on Dec 19, 2021 — 95f3b87

- Enter the Repository name and add the description of the repository.
- Select if you want the repository to be public or private.
- Click to the 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 * Agamjot-Singh03 / Repository name * SCM-Project

Great repository names are short and memorable. Need inspiration? How about animated-octo-succotash?

Description (optional)

Source Code Management Project of : Vinayak Singla (1540), Agamjot Singh (1622), Akash Mishra (1629), Hars

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.

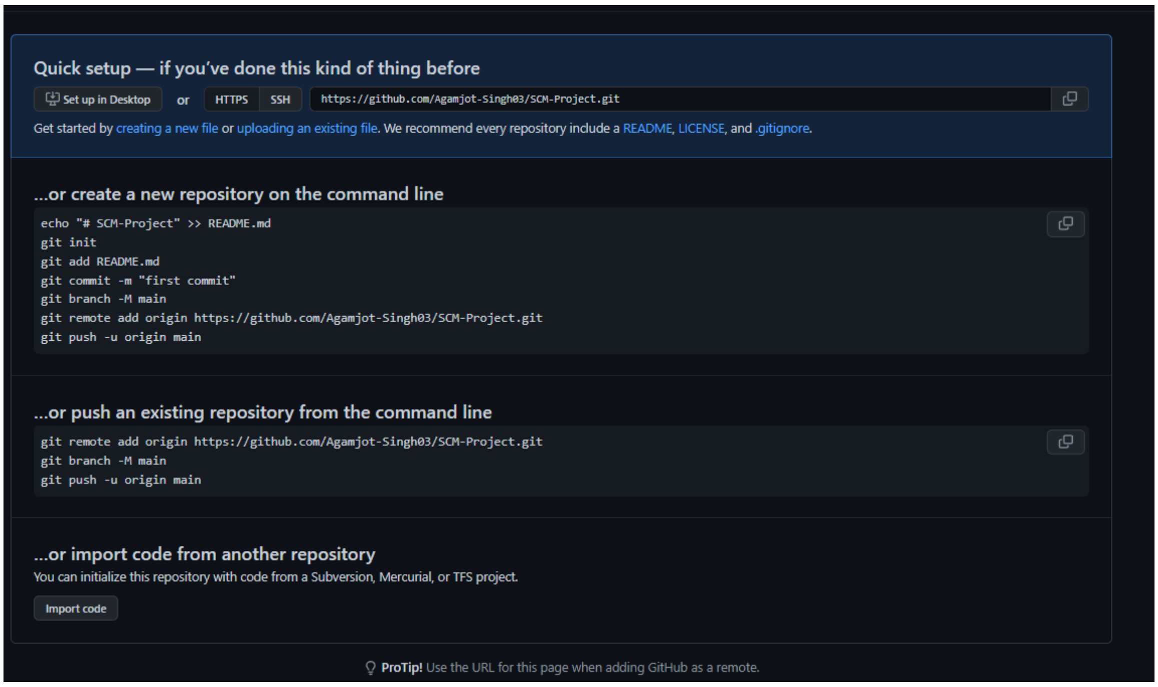
.gitignore template: None

Choose a license
A license tells others what they can and can't do with your code. Learn more.

License: None

You are creating a private repository in your personal account.

- If you want to import code from an existing repository select the



Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH https://github.com/Agamjot-Singh03/SCM-Project.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# SCM-Project" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Agamjot-Singh03/SCM-Project.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/Agamjot-Singh03/SCM-Project.git
git branch -M main
git push -u origin main
```

...or import code from another repository

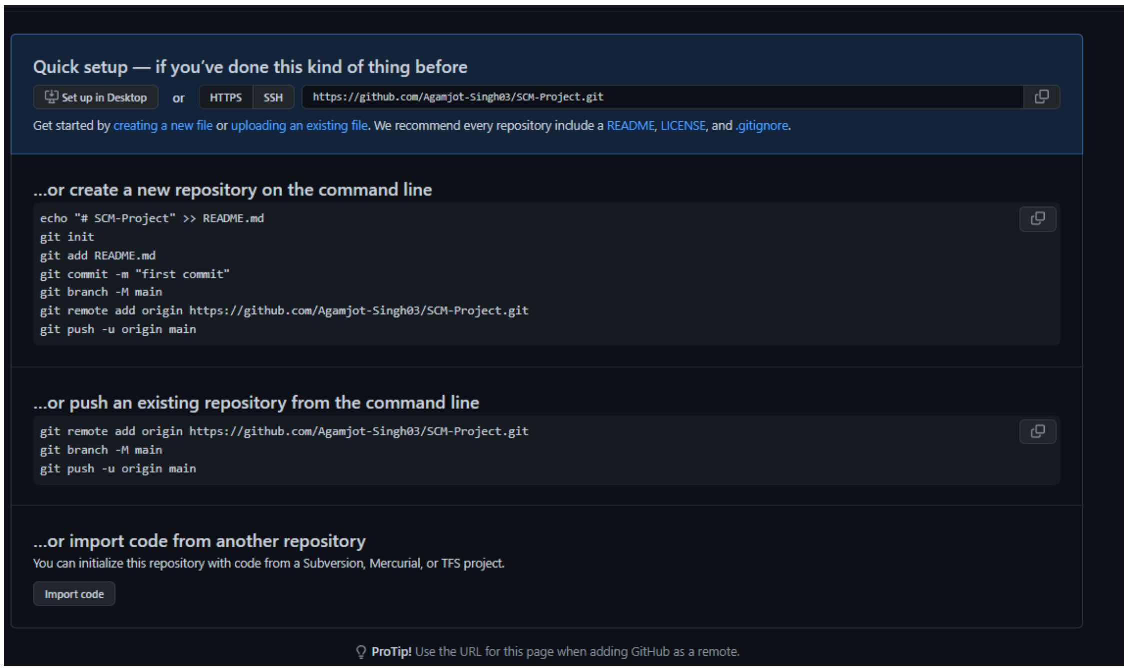
You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

ProTip! Use the URL for this page when adding GitHub as a remote.

import code option.

- Now, you have created your repository successfully.
- To add members to your repository, open your repository and select the settings option in the navigation bar.



Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH https://github.com/Agamjot-Singh03/SCM-Project.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# SCM-Project" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Agamjot-Singh03/SCM-Project.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/Agamjot-Singh03/SCM-Project.git
git branch -M main
git push -u origin main
```

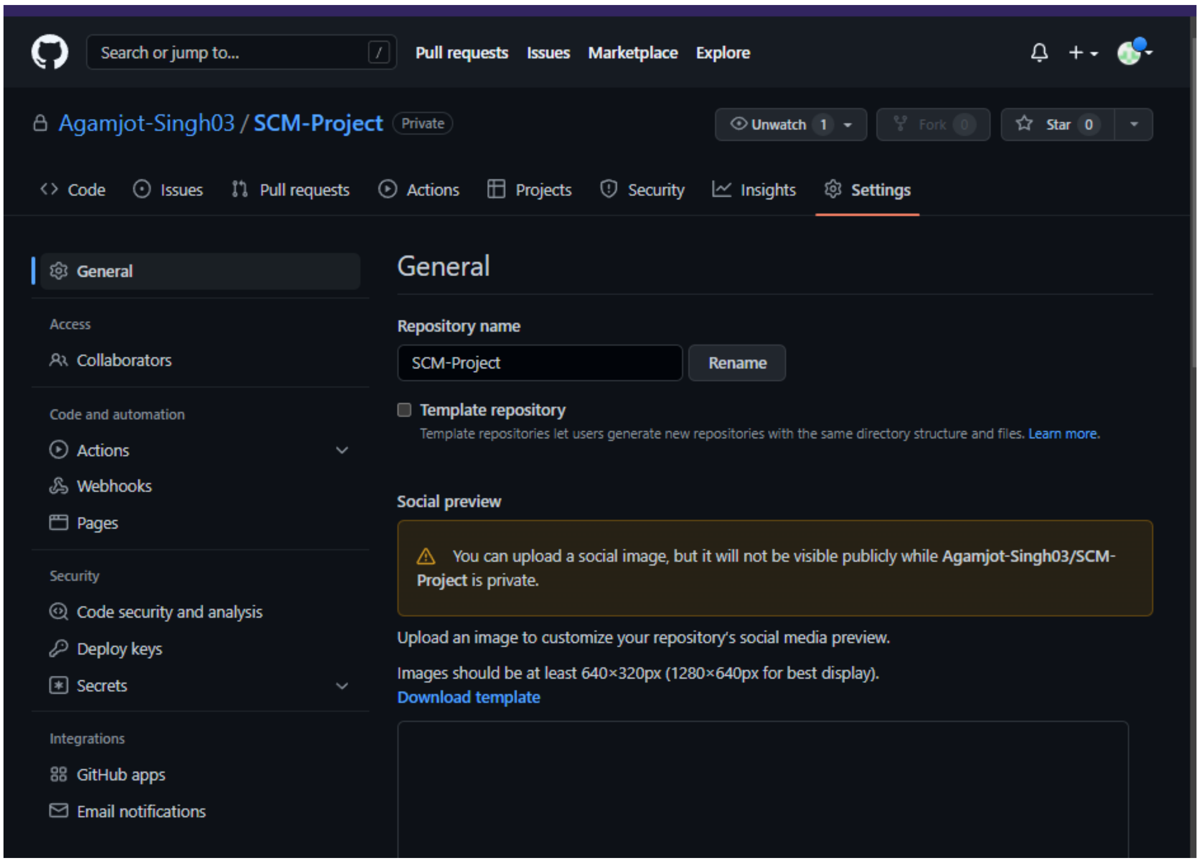
...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

ProTip! Use the URL for this page when adding GitHub as a remote.

- Click on Collaborators option under the access tab.
-



General

Repository name

SCM-Project Rename

Template repository

Template repositories let users generate new repositories with the same directory structure and files. Learn more.

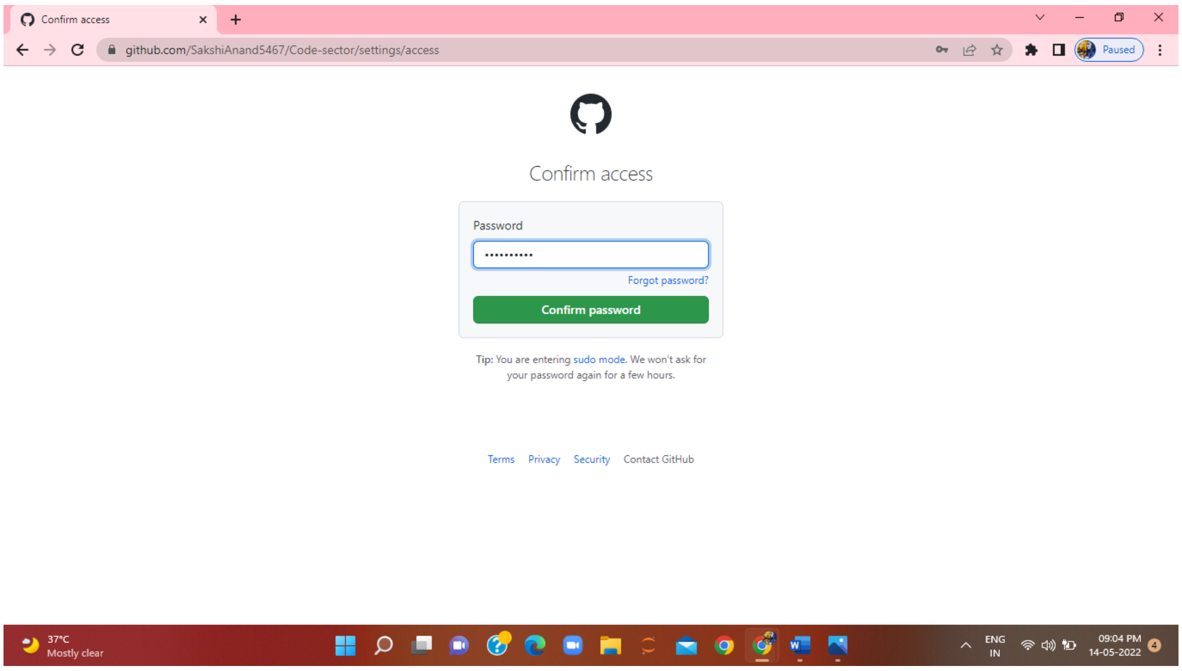
Social preview

You can upload a social image, but it will not be visible publicly while Agamjot-Singh03/SCM-Project is private.

Upload an image to customize your repository's social media preview.

Images should be at least 640×320px (1280×640px for best display).
Download template

- After clicking on collaborators GitHub asks you to enter your password to confirm the access to the repository.

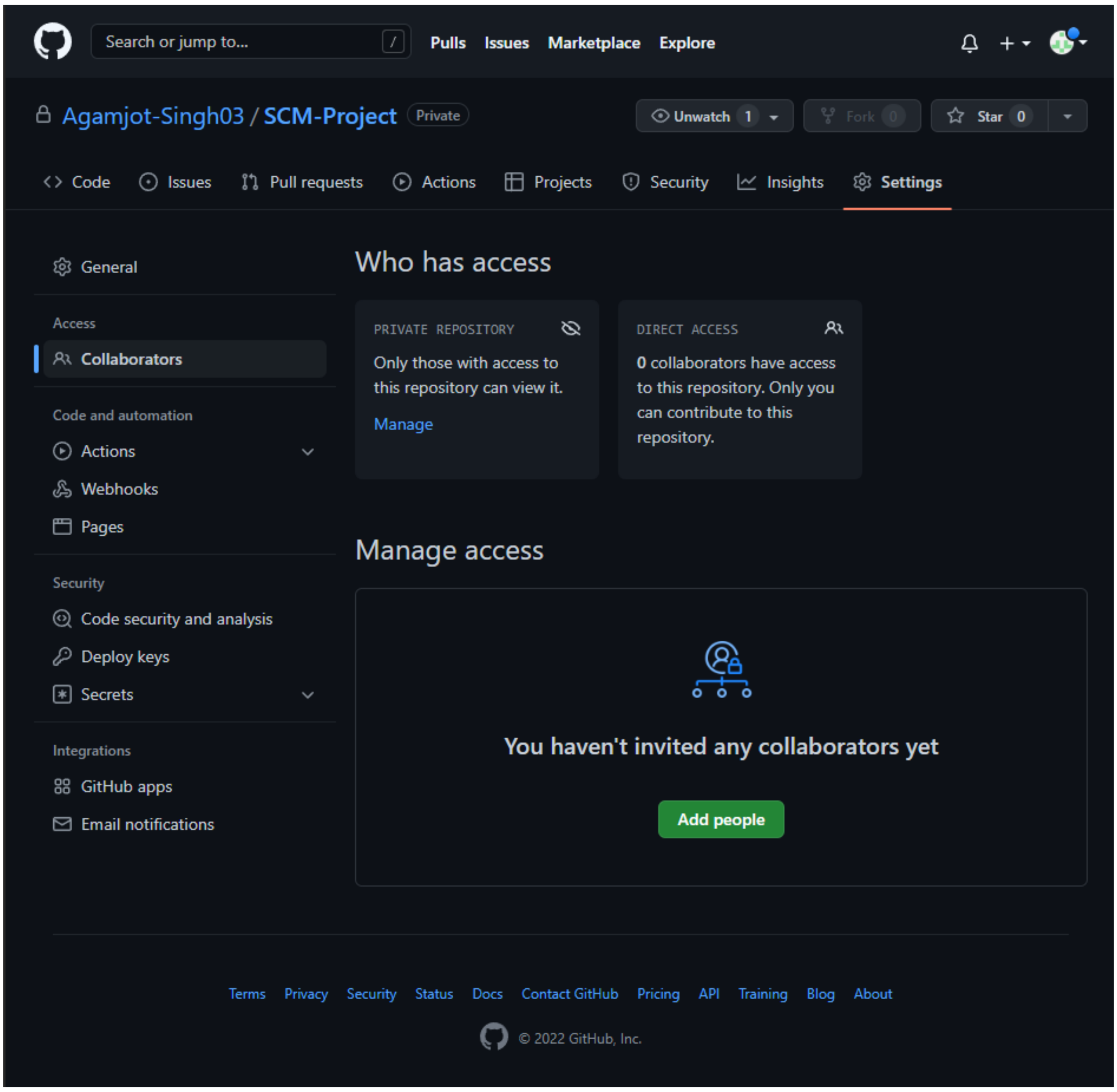


Tip: You are entering sudo mode. We won't ask for your password again for a few hours.

-

- After entering the password, you can manage access and add/remove team members to your project.

- To add members, click on the add people option and search the id of your respective team member.



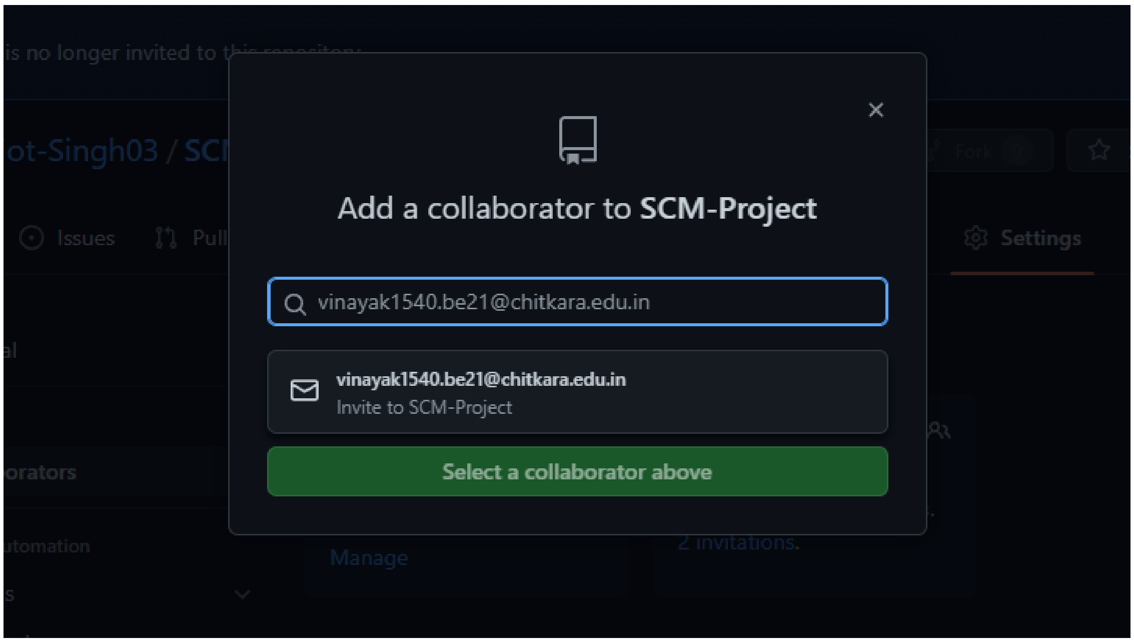
Who has access

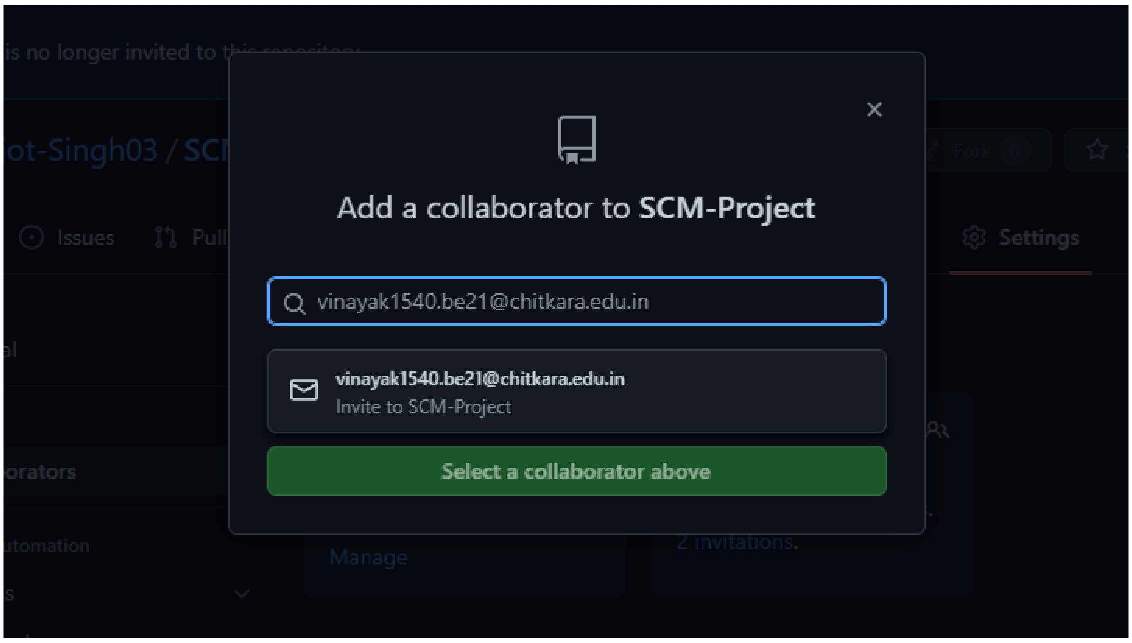
PRIVATE REPOSITORY
Only those with access to this repository can view it.
Manage

DIRECT ACCESS
0 collaborators have access to this repository. Only you can contribute to this repository.

Manage access

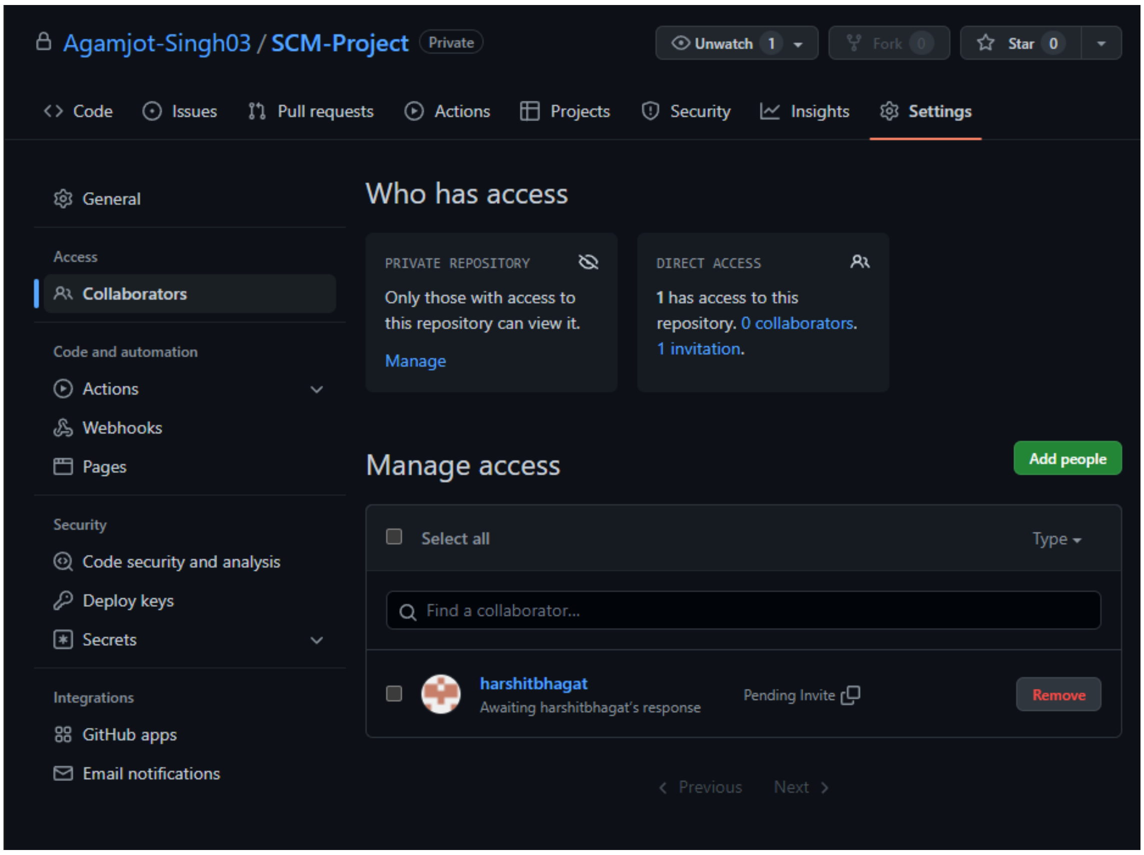
You haven't invited any collaborators yet

- 
Add a collaborator to SCM-Project
vinayak1540.be21@chitkara.edu.in
vinayak1540.be21@chitkara.edu.in
Invite to SCM-Project
Select a collaborator above

- 
Add a collaborator to SCM-Project
vinayak1540.be21@chitkara.edu.in
vinayak1540.be21@chitkara.edu.in
Invite to SCM-Project
Select a collaborator above

- Now, click on green box to add member in your repository.

- To remove any member, click on remove option available in the last column of the member's respective row.



Who has access

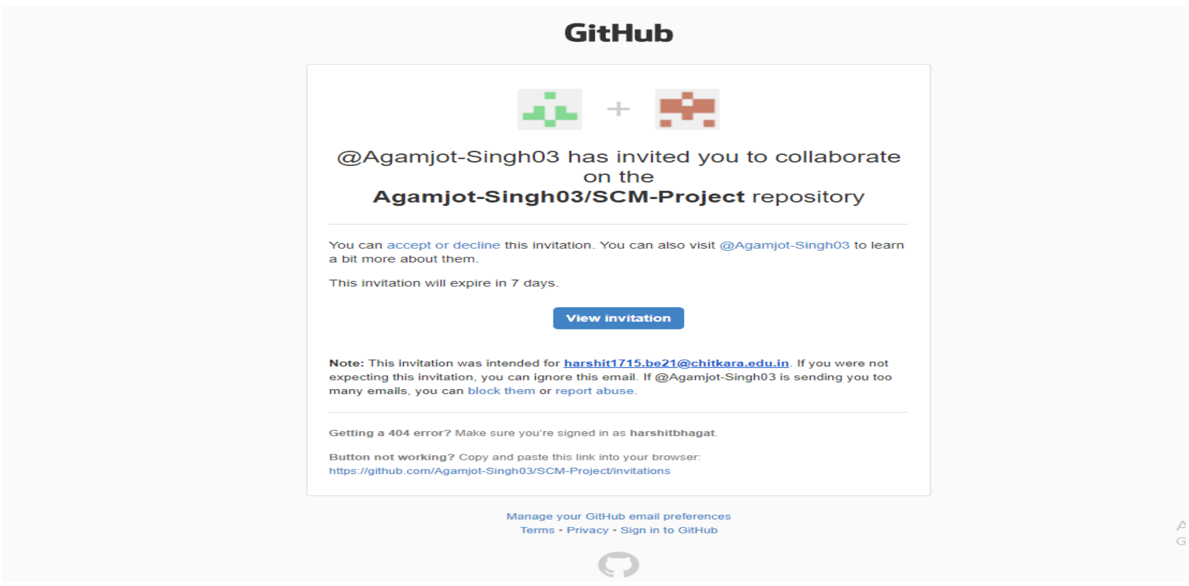
PRIVATE REPOSITORY
Only those with access to this repository can view it.
Manage

DIRECT ACCESS
1 has access to this repository. 0 collaborators. 1 invitation.

Manage access

harshitbhagat
Awaiting harshitbhagat's response
Pending Invite

- To accept the invitation from your team member, open your email registered with GitHub.
- You will receive an invitation mail from the repository owner. Open the email and click on accept invitation.
- You will be redirected to GitHub where you can either select to accept or decline the invitation.



GitHub

@Agamjot-Singh03 has invited you to collaborate on the **Agamjot-Singh03/SCM-Project** repository

You can accept or decline this invitation. You can also visit @Agamjot-Singh03 to learn a bit more about them.

This invitation will expire in 7 days.

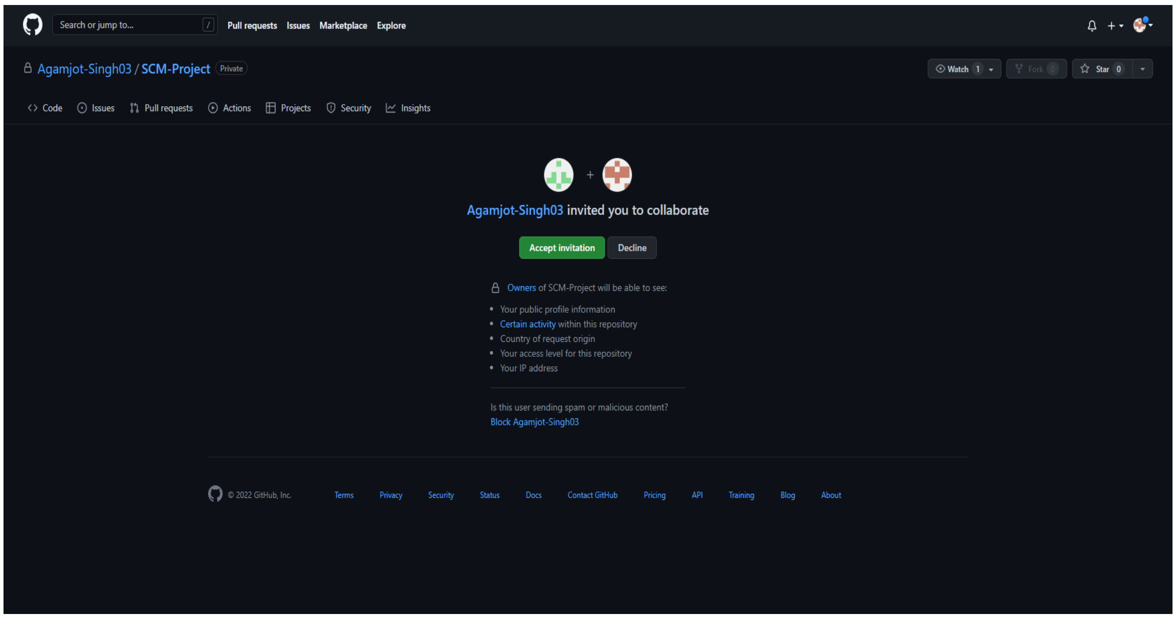
View invitation

Note: This invitation was intended for harshit1715.be21@chitkara.edu.in. If you were not expecting this invitation, you can ignore this email. If @Agamjot-Singh03 is sending you too many emails, you can block them or report abuse.

Getting a 404 error? Make sure you're signed in as **harshitbhagat**.

Button not working? Copy and paste this link into your browser:
https://github.com/Agamjot-Singh03/SCM-Project/invitations

Manage your GitHub email preferences
Terms • Privacy • Sign in to GitHub



Agamjot-Singh03 invited you to collaborate

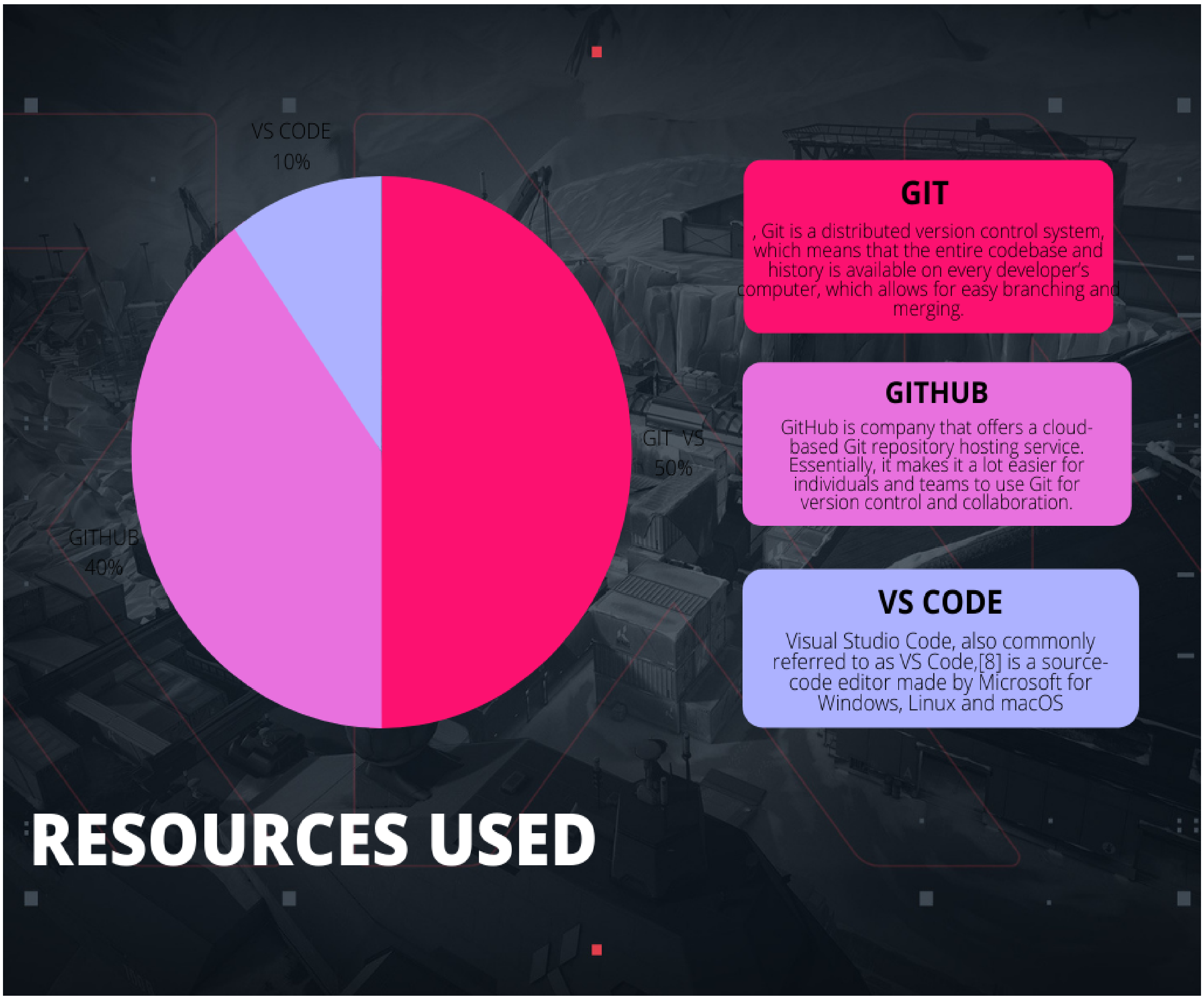
Owners of SCM-Project will be able to see:

- Your public profile information
- Certain activity within this repository
- Country of request origin
- Your access level for this repository
- Your IP address

Is this user sending spam or malicious content?
Block Agamjot-Singh03

- Now all members are ready to contribute to the project.

RESOURCES USED



GIT

, Git is a distributed version control system, which means that the entire codebase and history is available on every developer's computer, which allows for easy branching and merging.

GITHUB

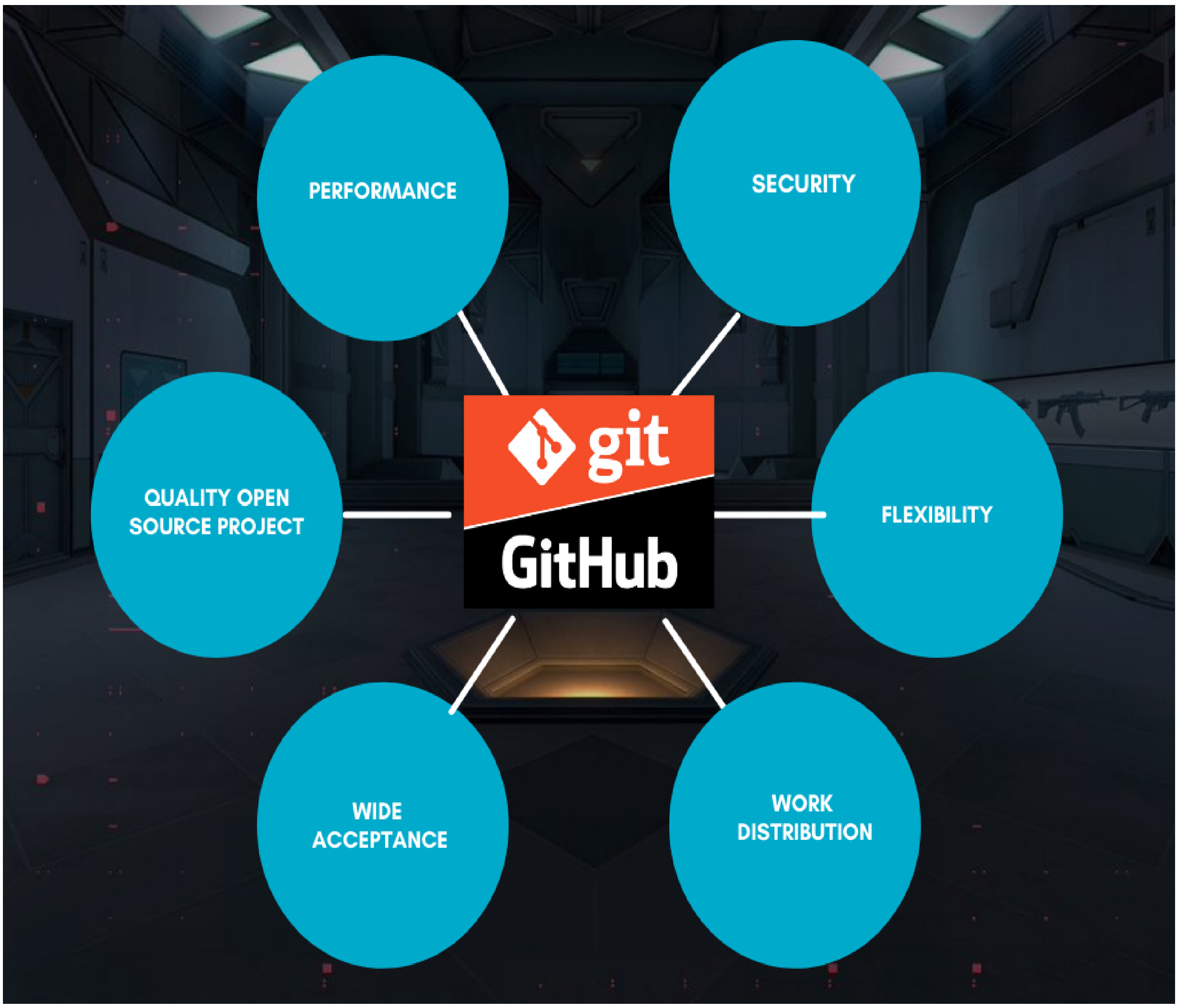
GitHub is company that offers a cloud-based Git repository hosting service. Essentially, it makes it a lot easier for individuals and teams to use Git for version control and collaboration.

VS CODE

Visual Studio Code, also commonly referred to as VS Code,[8] is a source-code editor made by Microsoft for Windows, Linux and macOS

RESOURCES USED

PROBLEM STATEMENT



1. Performance

Git performs very strongly and reliably when compared to other version control systems. New code changes can be easily committed, version branches can be effortlessly compared and merged, and code can also be optimized to perform better. Algorithms used in developing Git take the full advantage of the deep knowledge stored within, with regards to the attributes used to create real source code file trees, how files are modified over time and what kind of file access patterns are used to recall code files as and when needed by developers. Git primarily focuses upon the file content itself rather than file names while determining the storage and file version history. Object formats of Git repository files use several combinations of delta encoding and compression techniques to store metadata objects and directory contents.

2. Security

Git is designed specially to maintain the integrity of source code. File contents as well as the relationship between file and directories, tags, commits, versions etc. are secured cryptographically using an algorithm called SHA1 which protects the code and change history against accidental as well as malicious damage. You can be sure to have an authentic content history for your source code with Git.

3. Flexibility

A key design objective of Git is the kind of flexibility it offers to support several kinds of nonlinear development workflows and its efficiency in handling both small scale and large scale projects as well as protocols. It is uniquely designed to support tagging and branching operations and store each and every activity carried out by the user as an integral part of "change" history. Not all VCSs support this feature.

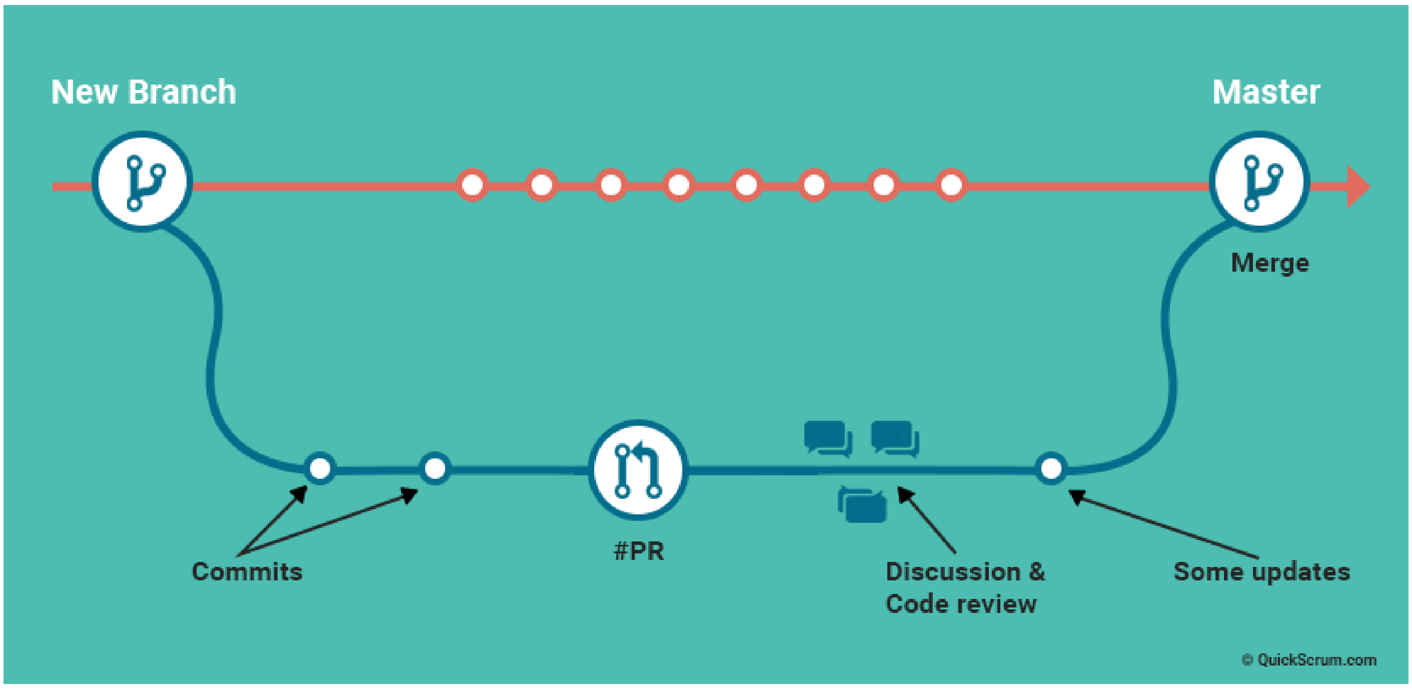
4. Wide acceptance

Git offers the type of performance, functionality, security, and flexibility that most developers and teams need to develop their projects. When compared to other VCS Git is the most widely accepted system owing to its universally accepted usability and performance standards.

5. Quality open source project

Git is a widely supported open source project with over ten years of operational history. People maintaining the project are very well matured and possess a long-term vision to meet the long-term needs of users by releasing staged upgrades at regular intervals of time to improve functionality as well as usability. Quality of open source software made available on Git is heavily scrutinized a countless number of times and businesses today depend heavily on Git code quality.

Pull Requests

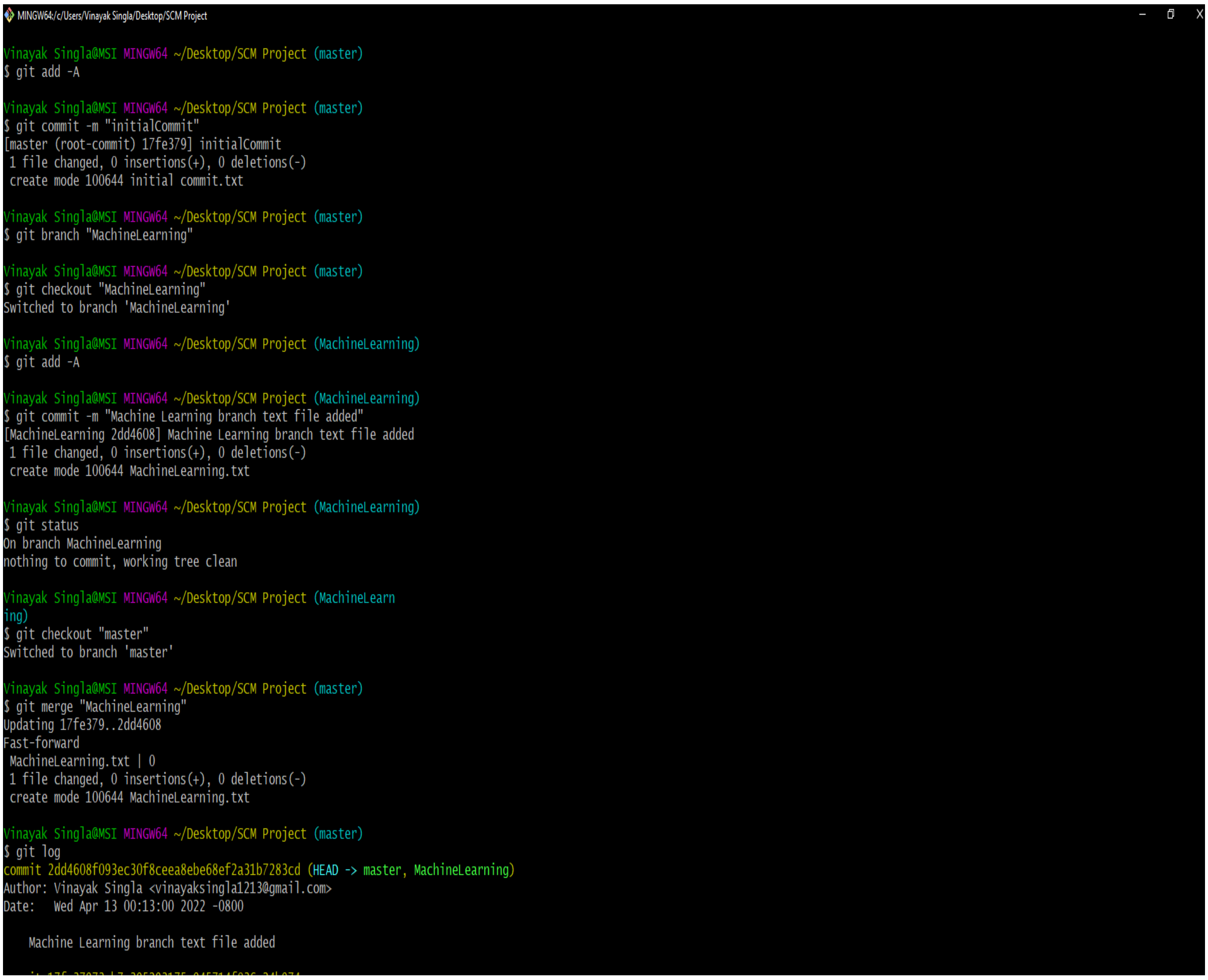


A developer calls a pull request to ask another developer to merge one of his/her branches into the other's repository. Besides making it a lot easier for project leaders to monitor and track code changes, "pulling" also facilitates other developers to discuss their work before integrating the code with the codebase. Moreover, if a developer can't continue work owing to some hard technical problem, he/she can initiate a pull request to seek help from the rest of the team.

We are going to represent the Problem Statement through one of the examples which will help us to understand better how it can reduce the work time and in this particular example we are going to cover how we can pull a repository directly from github instead of sending the file to a number of the users individually. How one can pull directly from Github by following some basic steps

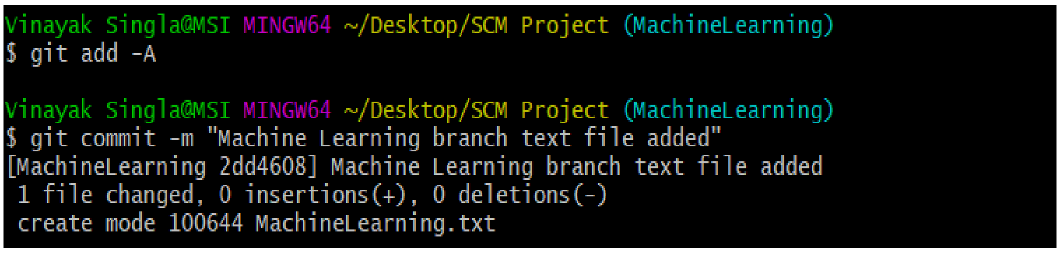
Open and Close a Pull Request

- To open a pull request we first have to make a new branch, by using git branch.
- After making a new branch we add a file to the branch or make change in the existing file.



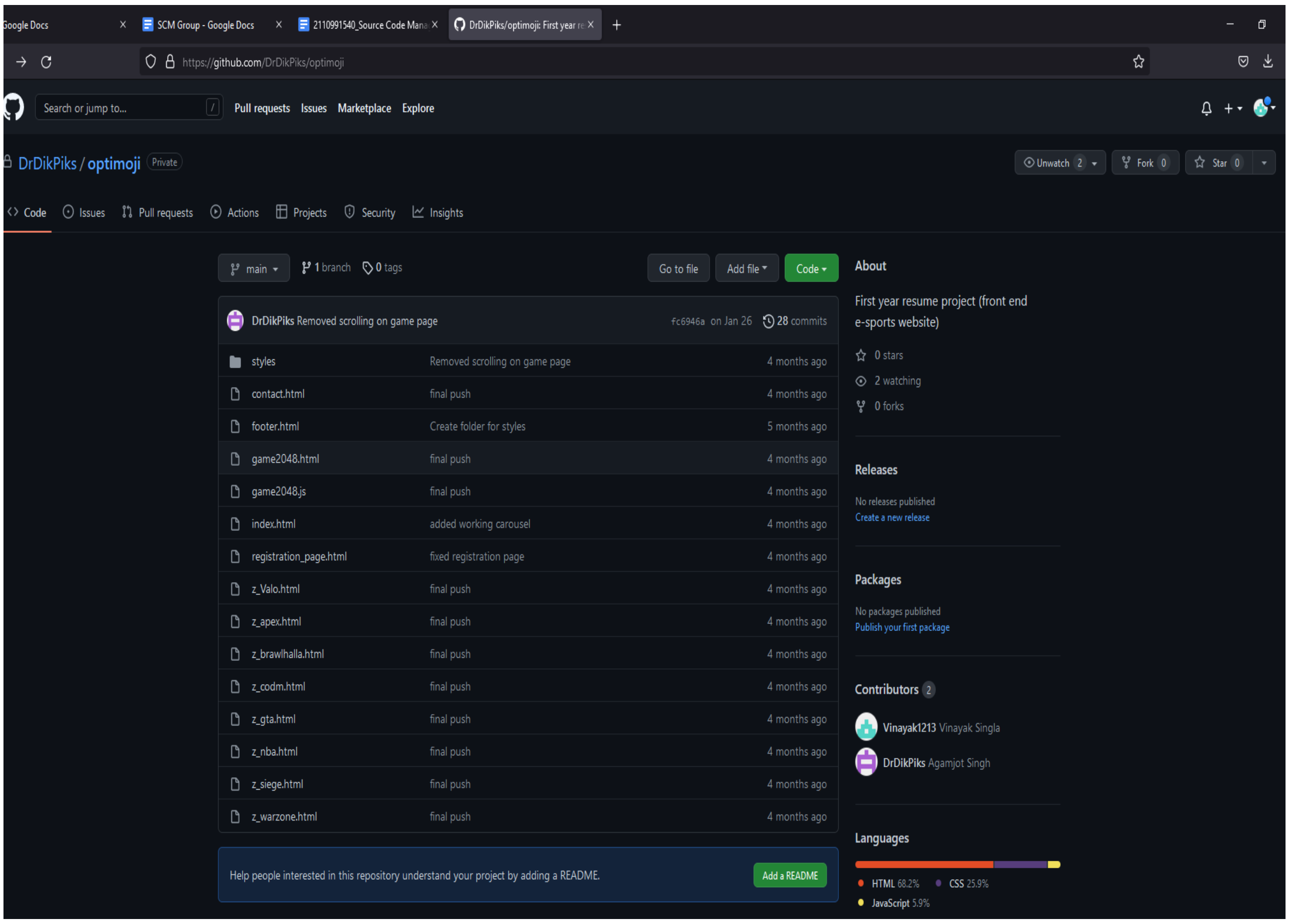
```
MINGW64:/c/Users/Vinayak Singla/Desktop/SCM Project

Vinayak Singla@MSI MINGW64 ~/Desktop/SCM Project (master)
$ git add -A

Vinayak Singla@MSI MINGW64 ~/Desktop/SCM Project (master)
$ git commit -m "initialCommit"
[master (root-commit) 17fe379] initialCommit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 initial commit.txt

Vinayak Singla@MSI MINGW64 ~/Desktop/SCM Project (master)
$ git branch "MachineLearning"

Vinayak Singla@MSI MINGW64 ~/Desktop/SCM Project (master)
$ git checkout "MachineLearning"
Switched to branch 'MachineLearning'

Vinayak Singla@MSI MINGW64 ~/Desktop/SCM Project (MachineLearning)
$ git add -A

Vinayak Singla@MSI MINGW64 ~/Desktop/SCM Project (MachineLearning)
$ git commit -m "Machine Learning branch text file added"
[MachineLearning 2dd4608] Machine Learning branch text file added
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 MachineLearning.txt

Vinayak Singla@MSI MINGW64 ~/Desktop/SCM Project (MachineLearning)
$ git status
On branch MachineLearning
nothing to commit, working tree clean

Vinayak Singla@MSI MINGW64 ~/Desktop/SCM Project (MachineLearn
ing)
$ git checkout "master"
Switched to branch 'master'

Vinayak Singla@MSI MINGW64 ~/Desktop/SCM Project (master)
$ git merge "MachineLearning"
Updating 17fe379..2dd4608
Fast-forward
 MachineLearning.txt | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 MachineLearning.txt

Vinayak Singla@MSI MINGW64 ~/Desktop/SCM Project (master)
$ git log
commit 2dd4608f093ec30f8ceea8ebe68ef2a31b7283cd (HEAD -> master, MachineLearning)
Author: Vinayak Singla <vinayaksingla1213@gmail.com>
Date:   Wed Apr 13 00:13:00 2022 -0800

    Machine Learning branch text file added
```

- Use git push origin to push the new branch to the main repository.



```
Vinayak Singla@MSI MINGW64 ~/Desktop/SCM Project (MachineLearning)
$ git add -A

Vinayak Singla@MSI MINGW64 ~/Desktop/SCM Project (MachineLearning)
$ git commit -m "Machine Learning branch text file added"
[MachineLearning 2dd4608] Machine Learning branch text file added
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 MachineLearning.txt
```

- After pushing new branch GitHub will either automatically ask you to create a pull request or you can create your own pull request by selecting the option compare & pull request.



https://github.com/DrDikPiks/optimoji

DrDikPiks / optimoji Private

Code | Issues | Pull requests | Actions | Projects | Security | Insights

main · 1 branch · 0 tags · Go to file · Add file · Code

DrDikPiks Removed scrolling on game page · fc6946a on Jan 26 · 28 commits

styles	Removed scrolling on game page	4 months ago
contact.html	final push	4 months ago
footer.html	Create folder for styles	5 months ago
game2048.html	final push	4 months ago
game2048.js	final push	4 months ago
index.html	added working carousel	4 months ago
registration_page.html	fixed registration page	4 months ago
z_Valo.html	final push	4 months ago
z_apex.html	final push	4 months ago
z_brawlhalla.html	final push	4 months ago
z_codm.html	final push	4 months ago
z_gta.html	final push	4 months ago
z_nba.html	final push	4 months ago
z_siege.html	final push	4 months ago
z_warzone.html	final push	4 months ago

Help people interested in this repository understand your project by adding a README. Add a README

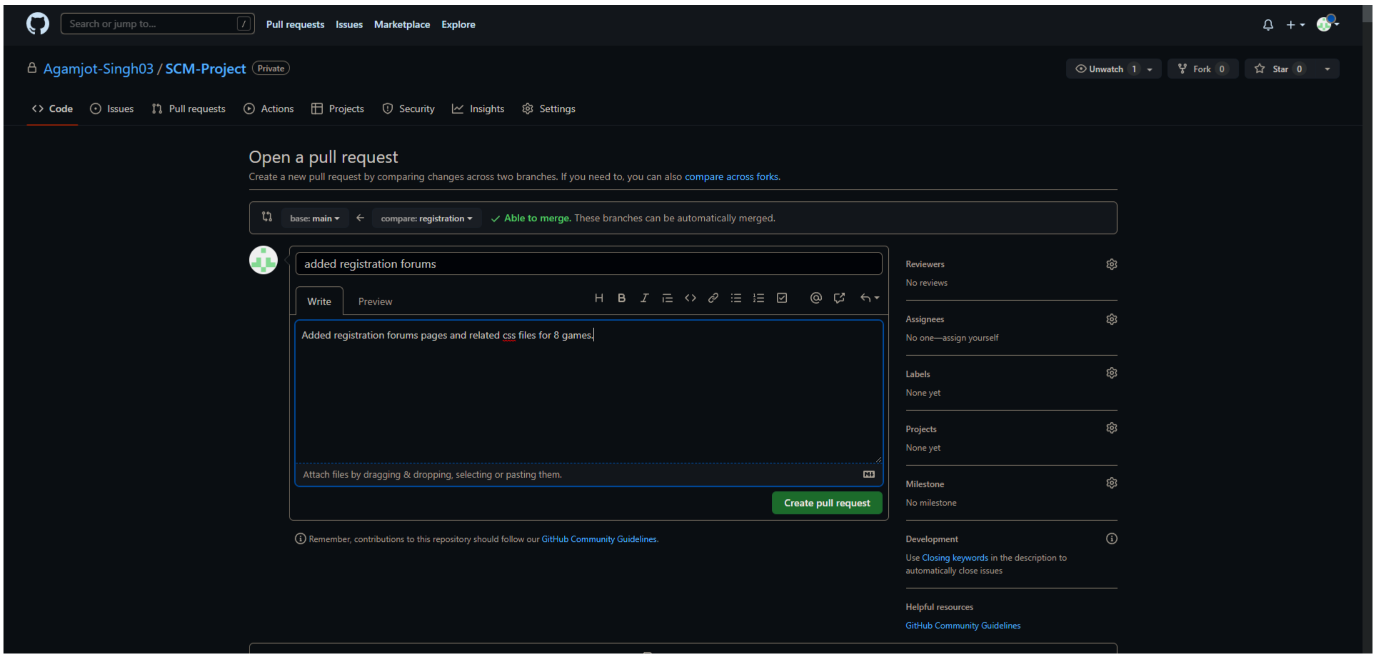
About: First year resume project (front end e-sports website) · 0 stars · 2 watching · 0 forks

Releases: No releases published · Create a new release

Packages: No packages published · Publish your first package

Contributors 2: Vinayak1213 Vinayak Singla · DrDikPiks Agamjot Singh

Languages: HTML 68.2% · CSS 25.9% · JavaScript 5.9%



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: main ← compare: registration ✓ Able to merge. These branches can be automatically merged.

added registration forums

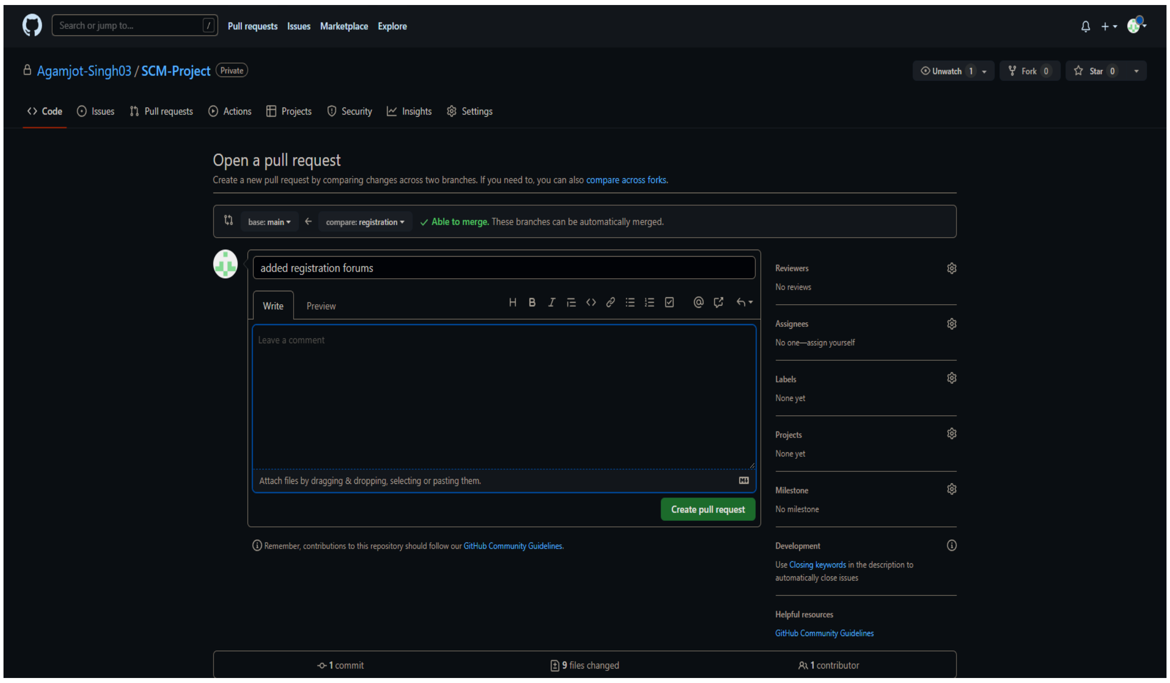
Added registration forums pages and related css files for 8 games.

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

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

Use Closing keywords in the description to automatically close issues

- To create your own pull request, click on create pull request option.
- GitHub will detect any conflicts and ask you to enter a description of your pull request.
- After opening a pull request all the team members will be sent the request if they want to merge or close the request.



Agamjot-Singh03 / SCM-Project Private

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: main ← compare: registration ✓ Able to merge. These branches can be automatically merged.

added registration forums

Leave a comment

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

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

Reviewers
No reviews

Assignees
No one—assign yourself

Labels
None yet

Projects
None yet

Milestone
No milestone

Development
Use Closing keywords in the description to automatically close issues

Helpful resources
GitHub Community Guidelines

1 commit | 9 files changed | 1 contributor

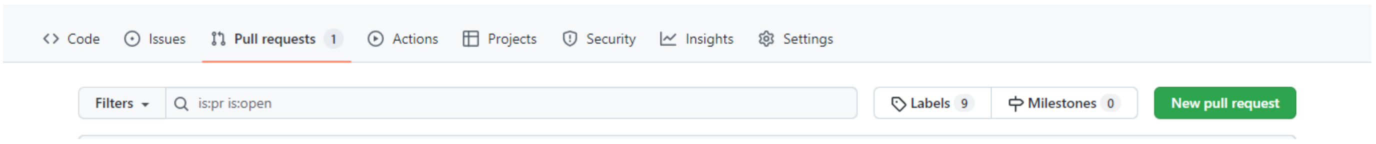
- If the team member chooses not to merge your pull request they will close the pull request.
- To close the pull request simply click on close pull request and add comment/ reason why you closed the pull request.
- You can see all the pull requests generated and how they were dealt with by clicking on pull request option.

CONCEPTS, COMMANDS AND OBJECTIVE

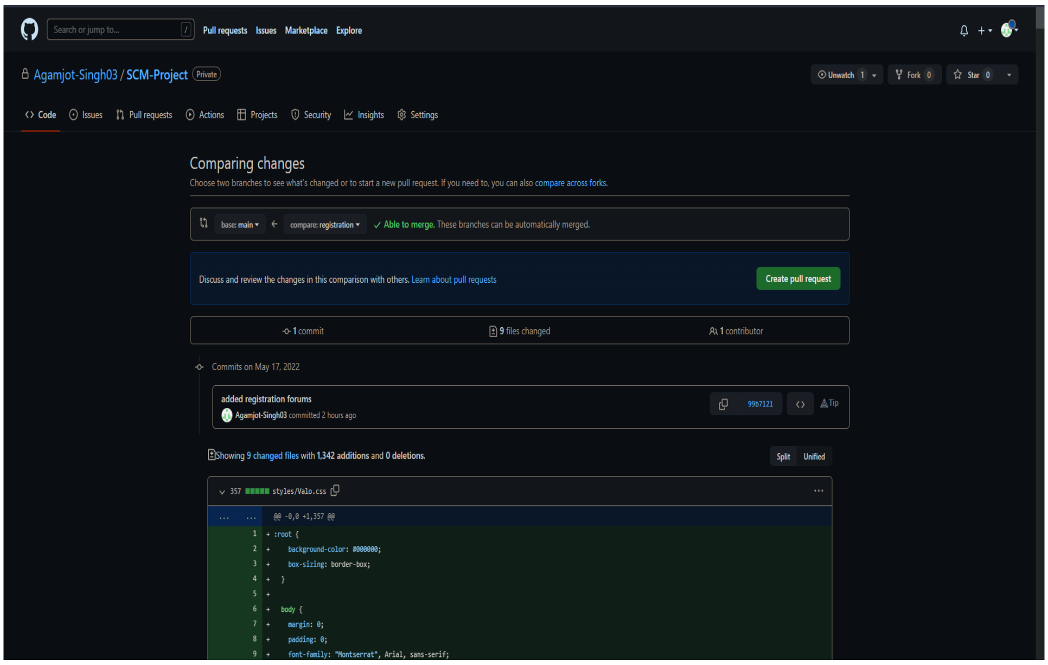
Aim: Create a pull request on a team member's repo and close pull requests generated by team members on own Repo as a maintainer

To create a pull request on a team member's repository and close requests by any other team members as a maintainer follow the procedure given below: -

- Do the required changes in the repository, add and commit these changes in the local repository in a new branch.
- Push the modified branch using git push origin branchname.
- Open a pull request by following the procedure from the above experiment.
- The pull request will be created and will be visible to all the team members.
- Ask your team member to login to his/her Github account.
- They will notice a new notification in the pull request menu.



- Click on it. The pull request generated by you will be visible to them.


Search or jump to...
Pull requests Issues Marketplace Explore
Agamjot-Singh03 / SCM-Project Private
Unwatch 1 Fork 0 Star 0
Code Issues Pull requests Actions Projects Security Insights Settings
Create pull request
Split Unified

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks.

base: main ← compare: registration ✓ Able to merge. These branches can be automatically merged.

Discuss and review the changes in this comparison with others. Learn about pull requests

1 commit · 9 files changed · 1 contributor

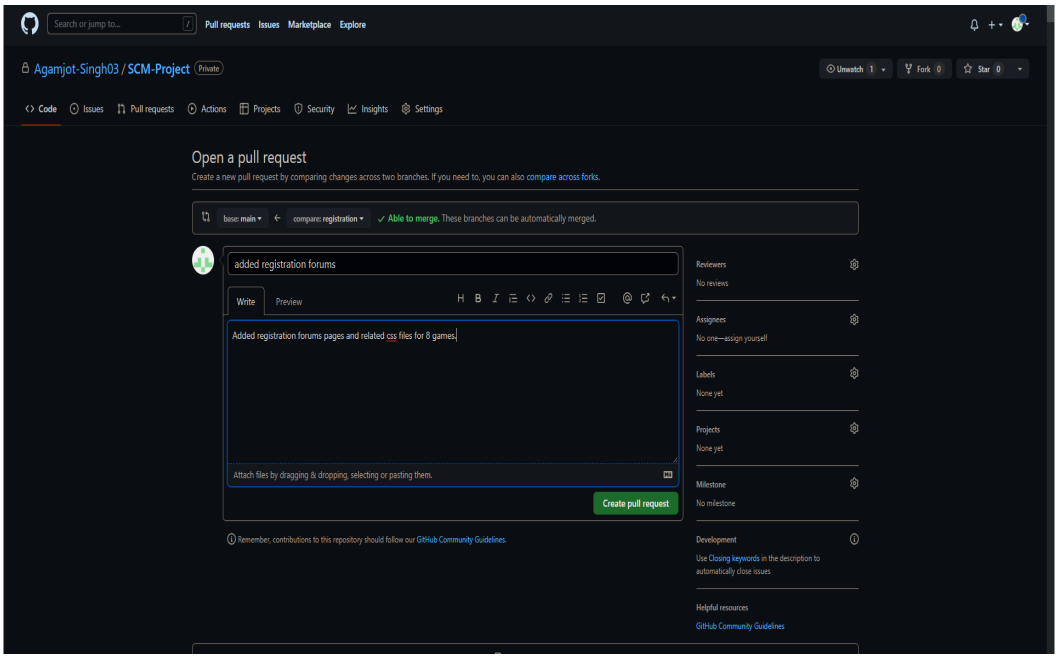
Commits on May 17, 2022

added registration forums
Agamjot-Singh03 committed 2 hours ago · 99b7121

Showing 9 changed files with 1,342 additions and 0 deletions.

```
357 styles/Valo.css
@@ -0,0 +1,357 @@
+ :root {
+     background-color: #000000;
+     box-sizing: border-box;
+   }
+
+   body {
+     margin: 0;
+     padding: 0;
+     font-family: "Montserrat", Arial, sans-serif;
```

- Click on the pull request. Two options will be available, either to close the pull request or merge the request with the main branch.
- By selecting the merge pull request, the main branch will get updated for all the team members.
- By selecting close the pull request the pull request is not accepted and not merged with the main branch.


Search or jump to...
Pull requests Issues Marketplace Explore
Agamjot-Singh03 / SCM-Project Private
Unwatch 1 Fork 0 Star 0
Code Issues Pull requests Actions Projects Security Insights Settings
Write Preview
Create pull request

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: main ← compare: registration ✓ Able to merge. These branches can be automatically merged.

added registration forums

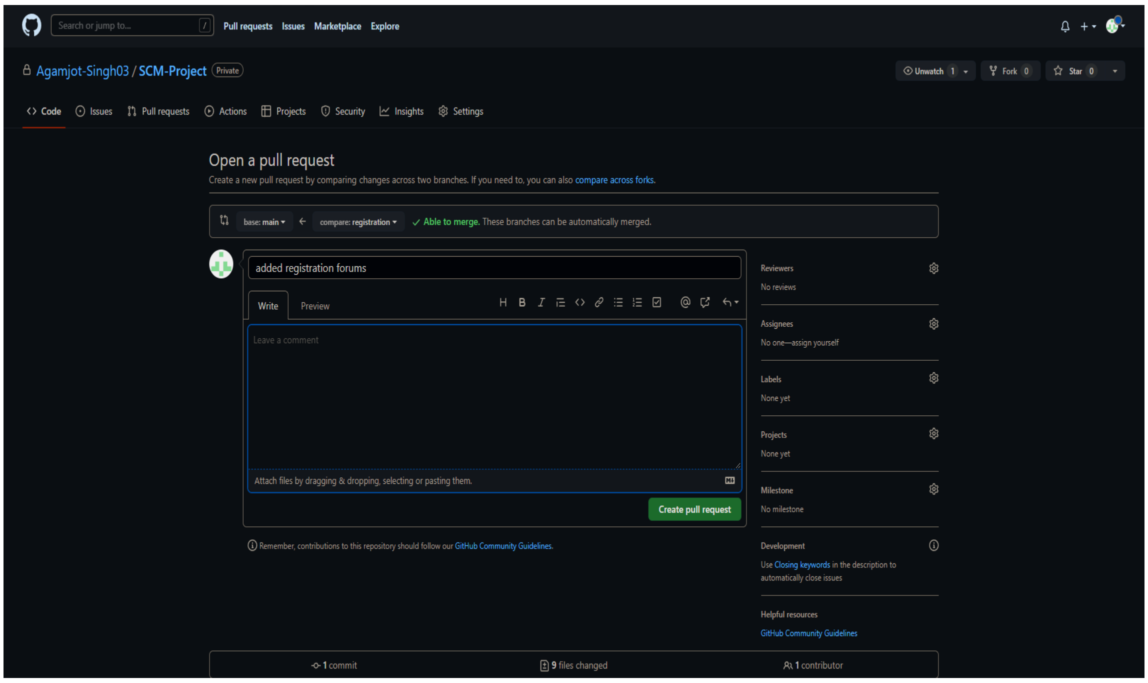
Added registration forums pages and related css files for 8 games.

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

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

Reviewers: No reviews
Assignees: No one—assign yourself
Labels: None yet
Projects: None yet
Milestone: No milestone
Development: Use Closing keywords in the description to automatically close issues
Helpful resources: GitHub Community Guidelines

- The process is similar to closing and merging the pull request by you. It simply includes an external party to execute.
- The result of merging the pull request is shown below.



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: main ← compare: registration ✓ Able to merge. These branches can be automatically merged.

added registration forums

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

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

No reviews

No one—assign yourself

None yet

None yet

No milestone

Use Closing keywords in the description to automatically close issues

1 commit · 9 files changed · 1 contributor

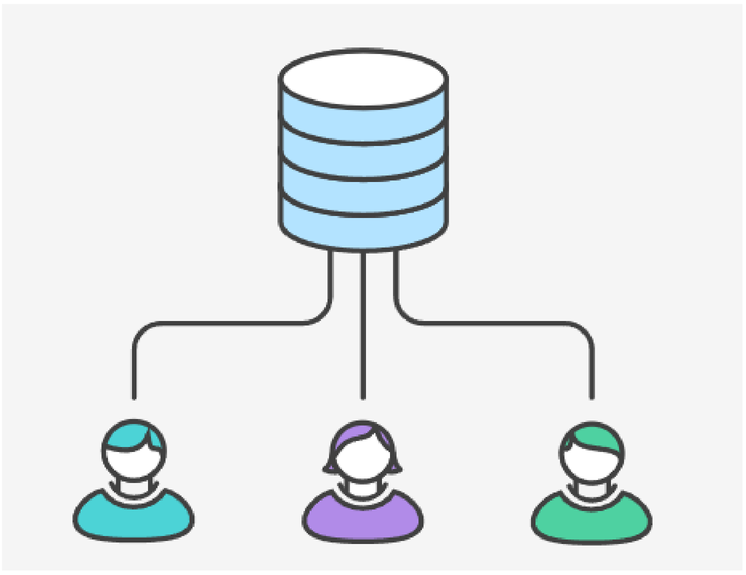
FLOWCHART, WORKFLOW AND DISCUSSIONS IN GIT

FLOWCHART AND WORKFLOW

What is a successful Git workflow?

When evaluating a workflow for your team, it's most important that you consider your team's culture. You want the workflow to enhance the effectiveness of your team and not be a burden that limits productivity. Some things to consider when evaluating a Git workflow are:

- Does this workflow scale with team size?
- Is it easy to undo mistakes and errors with this workflow?
- Does this workflow impose any new unnecessary cognitive overhead to the team?



Centralized Workflow

The Centralized Workflow is a great Git workflow for teams transitioning from SVN. Like Subversion, the Centralized Workflow uses a central repository to serve as the single point-of-entry for all changes to the project. Instead of trunk, the default development branch is called main and all changes are committed into this branch. This workflow doesn't require any other branches besides main.

It gives every developer their own local copy of the entire project. This isolated environment lets each developer work independently of all other changes to a project - they can add commits to their local repository and completely forget about upstream developments until it's convenient for them.

How it works

Developers start by cloning the central repository. In their own local copies of the project, they edit files and commit changes these new commits are stored locally - they're completely isolated from the central repository. This lets developers defer synchronizing upstream until they're at a convenient break point.

To publish changes to the official project, developers "push" their local main branch to the central repository.

- **Initialize the central repository**
  First, someone needs to create the central repository on a server.
  Central repositories should always be bare repositories
  Be sure to use a valid SSH username for user, the domain or IP address of your server for host, and the location where you'd like to store your repo
- **Hosted central repositories**
  Central repositories are often created through 3rd party Git hosting services
  The hosting service will then provide an address for the central repository to access from your local repository.
- **Clone the central repository**

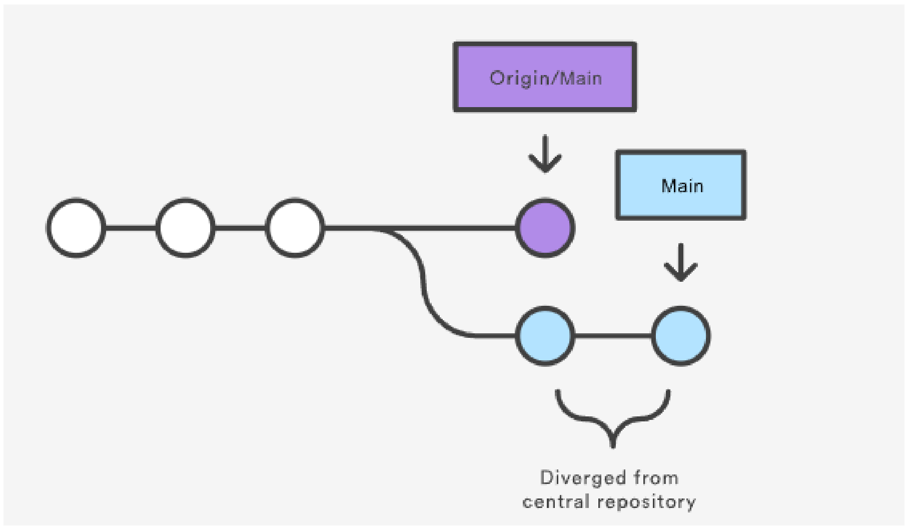
  Next, each developer creates a local copy of the entire project. This is accomplished via the git clone command:

- **Make changes and commit**
  Once the repository is cloned locally, a developer can make changes using the standard Git commit process: edit, stage, and commit.
- **Push new commits to central repository**
  Once the local repository has new changes committed. These change will need to be pushed to share with other developers on the project.
  This is accomplished via the git push origin command

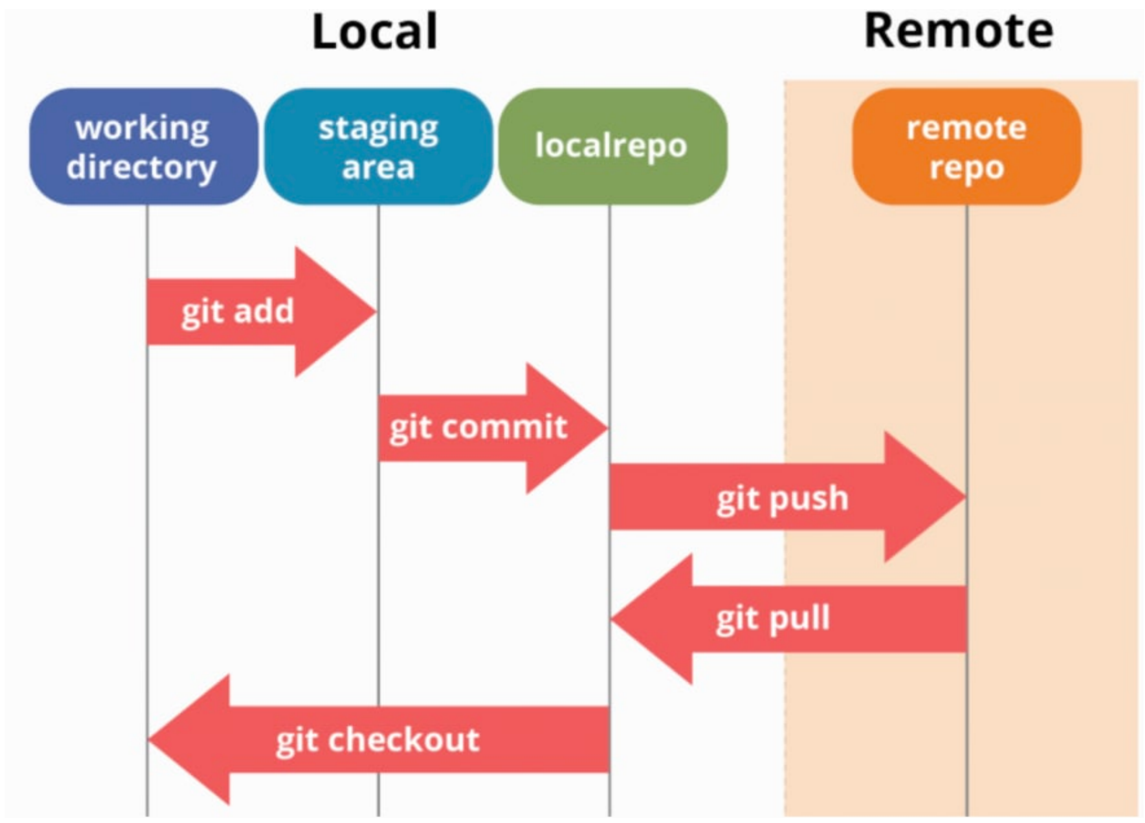
```
git push origin main
```

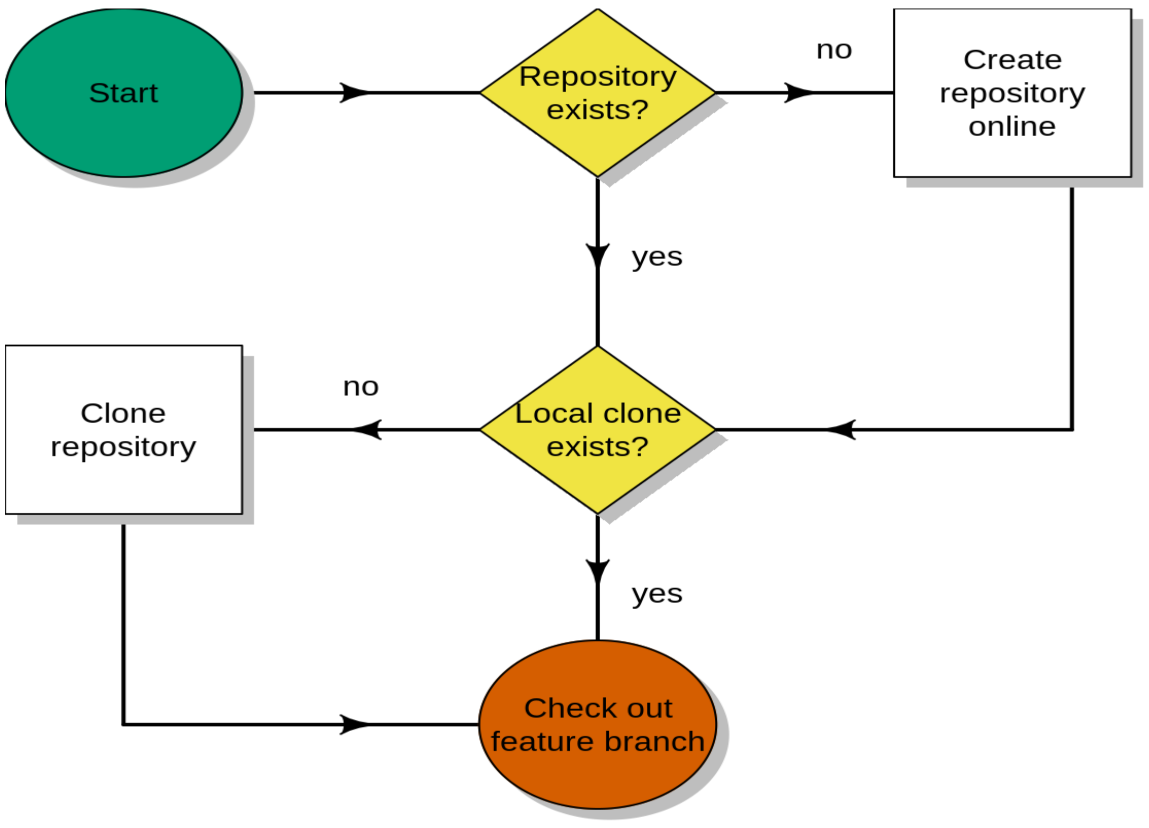
- **Managing conflicts**

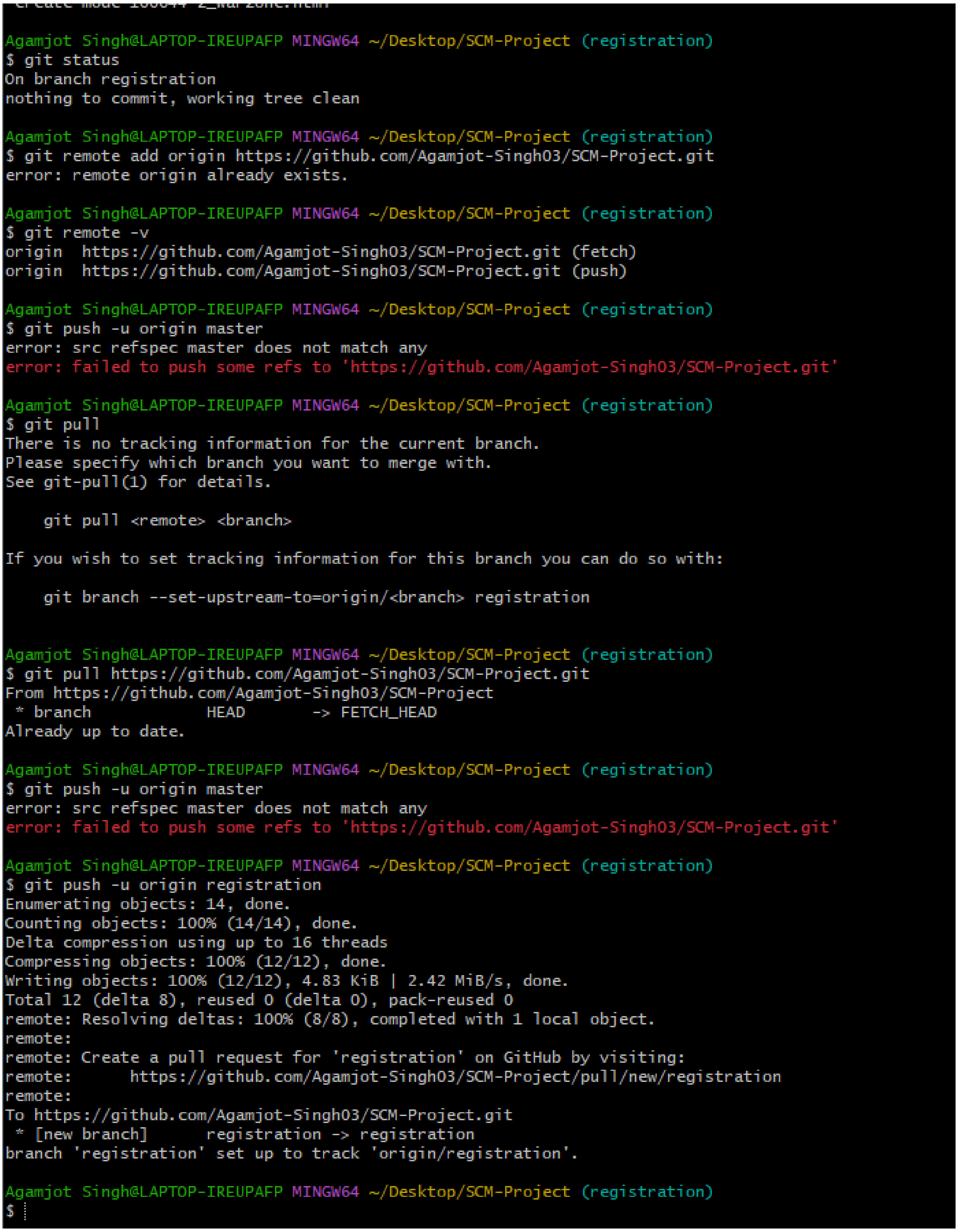
  The central repository represents the official project, so its commit history should be treated as sacred and immutable. If a developer's local commits diverge from the central repository, Git will refuse to push their changes because this would overwrite official commits.



  Before the developer can publish their feature, they need to fetch the updated central commits and rebase their changes on top of them. This is like saying, "I want to add my changes to what everyone else has already done."
  If local changes directly conflict with upstream commits, Git will pause the rebasing process and give you a chance to manually resolve the conflicts. The nice thing about Git is that it uses the same git status and git add commands for both generating commits and resolving merge conflicts. This makes it easy for new developers to manage their own merges. Plus, if they get themselves into trouble, Git makes it very easy to abort the entire rebase and try again


Local
Remote
working directory
staging area
localrepo
remote repo
git add
git commit
git push
git pull
git checkout


Start
Repository exists?
no
Create repository online
yes
Local clone exists?
no
Clone repository
yes
Check out feature branch



```
 create mode 100644 2_war zone.html

Agamjot Singh@LAPTOP-IREUPAFP MINGW64 ~/Desktop/SCM-Project (registration)
$ git status
On branch registration
nothing to commit, working tree clean

Agamjot Singh@LAPTOP-IREUPAFP MINGW64 ~/Desktop/SCM-Project (registration)
$ git remote add origin https://github.com/Agamjot-Singh03/SCM-Project.git
error: remote origin already exists.

Agamjot Singh@LAPTOP-IREUPAFP MINGW64 ~/Desktop/SCM-Project (registration)
$ git remote -v
origin  https://github.com/Agamjot-Singh03/SCM-Project.git (fetch)
origin  https://github.com/Agamjot-Singh03/SCM-Project.git (push)

Agamjot Singh@LAPTOP-IREUPAFP MINGW64 ~/Desktop/SCM-Project (registration)
$ git push -u origin master
error: src refspec master does not match any
error: failed to push some refs to 'https://github.com/Agamjot-Singh03/SCM-Project.git'

Agamjot Singh@LAPTOP-IREUPAFP MINGW64 ~/Desktop/SCM-Project (registration)
$ git pull
There is no tracking information for the current branch.
Please specify which branch you want to merge with.
See git-pull(1) for details.

    git pull <remote> <branch>

If you wish to set tracking information for this branch you can do so with:

    git branch --set-upstream-to=origin/<branch> registration


Agamjot Singh@LAPTOP-IREUPAFP MINGW64 ~/Desktop/SCM-Project (registration)
$ git pull https://github.com/Agamjot-Singh03/SCM-Project.git
From https://github.com/Agamjot-Singh03/SCM-Project
 * branch            HEAD       -> FETCH_HEAD
Already up to date.

Agamjot Singh@LAPTOP-IREUPAFP MINGW64 ~/Desktop/SCM-Project (registration)
$ git push -u origin master
error: src refspec master does not match any
error: failed to push some refs to 'https://github.com/Agamjot-Singh03/SCM-Project.git'

Agamjot Singh@LAPTOP-IREUPAFP MINGW64 ~/Desktop/SCM-Project (registration)
$ git push -u origin registration
Enumerating objects: 14, done.
Counting objects: 100% (14/14), done.
Delta compression using up to 16 threads
Compressing objects: 100% (12/12), done.
Writing objects: 100% (12/12), 4.83 KiB | 2.42 MiB/s, done.
Total 12 (delta 8), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (8/8), completed with 1 local object.
remote:
remote: Create a pull request for 'registration' on GitHub by visiting:
remote:      https://github.com/Agamjot-Singh03/SCM-Project/pull/new/registration
remote:
To https://github.com/Agamjot-Singh03/SCM-Project.git
 * [new branch]      registration -> registration
branch 'registration' set up to track 'origin/registration'.

Agamjot Singh@LAPTOP-IREUPAFP MINGW64 ~/Desktop/SCM-Project (registration)
$
```

Events

An event is a specific activity in a repository that triggers a workflow run. For example, activity can originate from GitHub when someone creates a pull request, opens an issue, or pushes a commit to a repository. You can also trigger a workflow run on a schedule, by posting to a REST API, or manually.

For a complete list of events that can be used to trigger workflows, see Events that trigger workflows.

NETWORK GRAPHS

The network graph is one of the useful features for developers on GitHub. It is used to display the branch history of the entire repository network, including branches of the root repository and branches of forks that contain commits unique to the network.

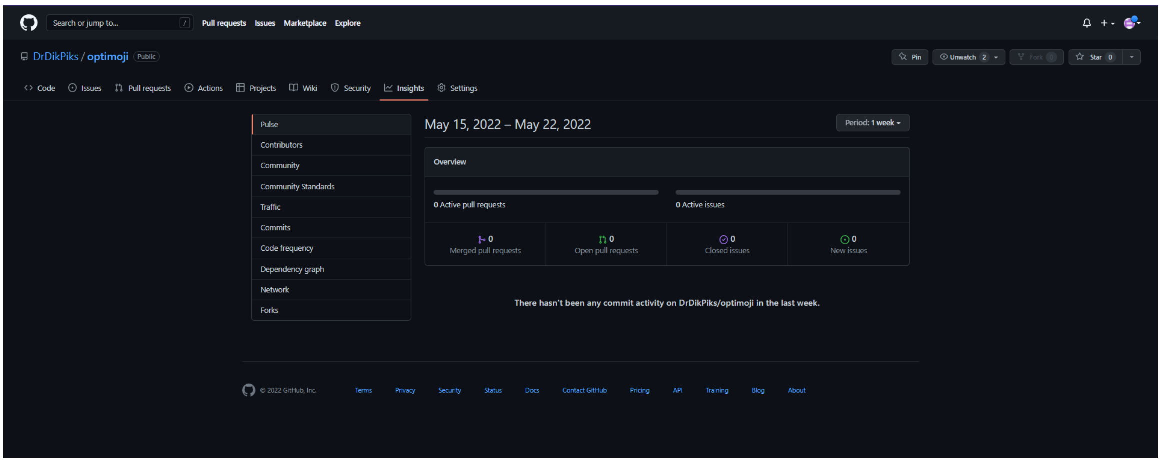
A repository's graphs give you information on traffic, projects that depend on the repository, contributors and commits to the repository, and a repository's forks and network. If you maintain a repository, you can use this data to get a better understanding of who's using your repository and why they're using it.

Some repository graphs are available only in public repositories with GitHub Free:

- Pulse
- Contributors
- Traffic
- Commits
- Code frequency
- Network

<u>Steps to access network graphs of respective repository</u>

1. On GitHub.com, navigate to the main page of the repository.



May 15, 2022 – May 22, 2022

0 Active pull requests

0 Active issues

0 Merged pull requests

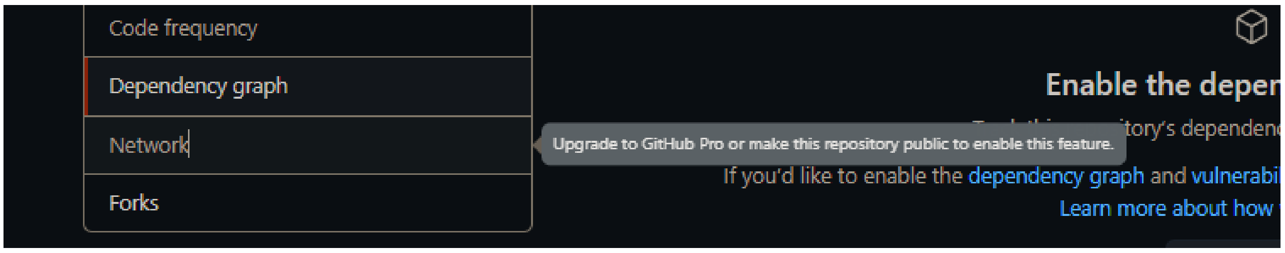
0 Open pull requests

0 Closed issues

0 New issues

There hasn't been any commit activity on DrDikPiks/optimoji in the last week.

2.Under your repository name, click Insights.



Enable the deper

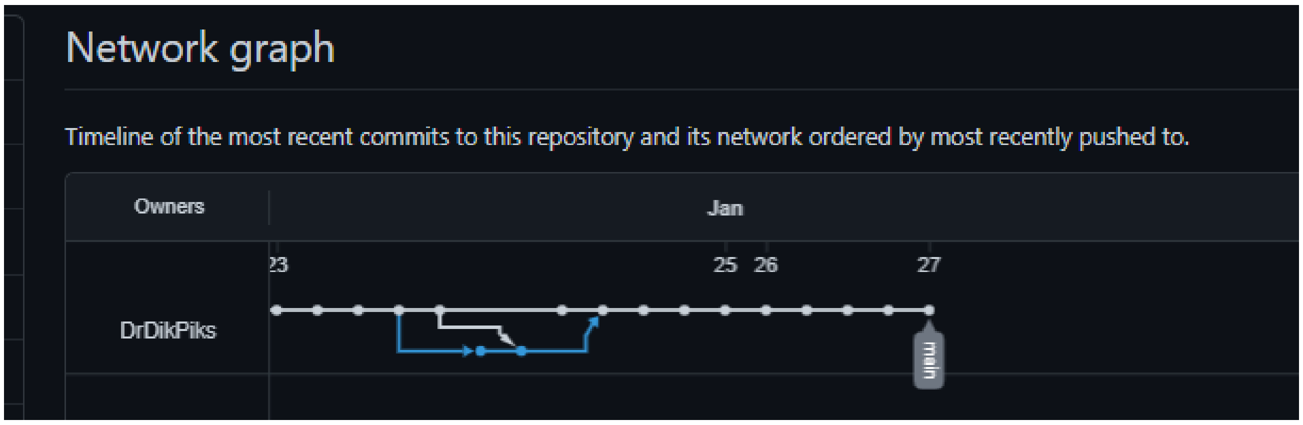
Upgrade to GitHub Pro or make this repository public to enable this feature.

tory's dependenc

If you'd like to enable the dependency graph and vulnerabi

Learn more about how

You will get the network graph of your repository which displays the branch history of the entire repository network, including branches of the root repository and branches of forks that contain commits unique to the network.



Network graph

Timeline of the most recent commits to this repository and its network ordered by most recently pushed to.