**Experiment 1**

Create a GitHub account, then create a new repository with a README file. Upload a basic file named hello.txt, edit the file through GitHub’s web interface, commit the changes, and view the commit history. Share the link to your repository after completion.

**Step 1**: Create a GitHub Account

Go to <https://github.com>

Click “Sign Up”

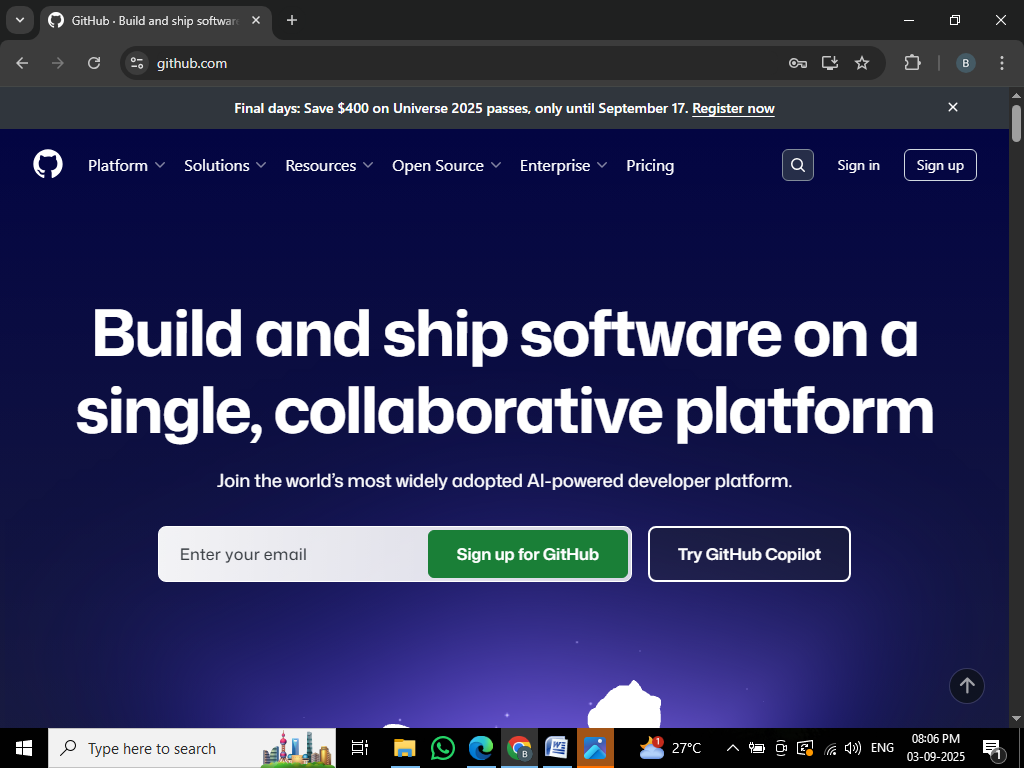
Enter:

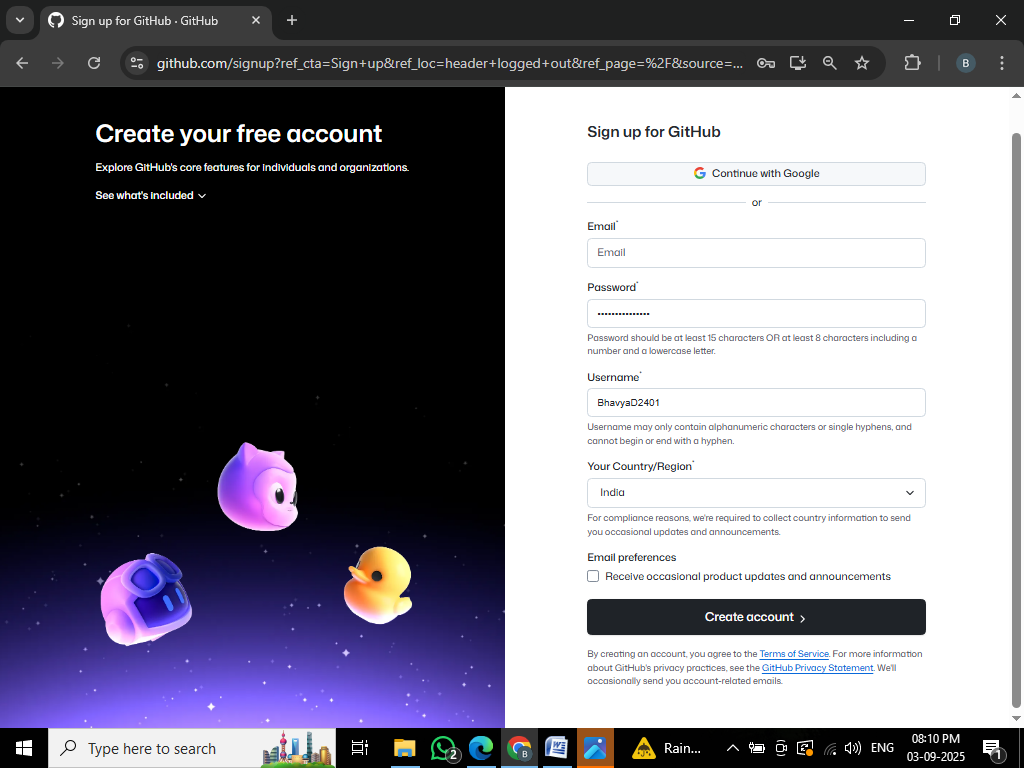
Email address

Username

Password

Verify your email and log in





Step 2: create new repository

After logging in, click the + icon → New Repository

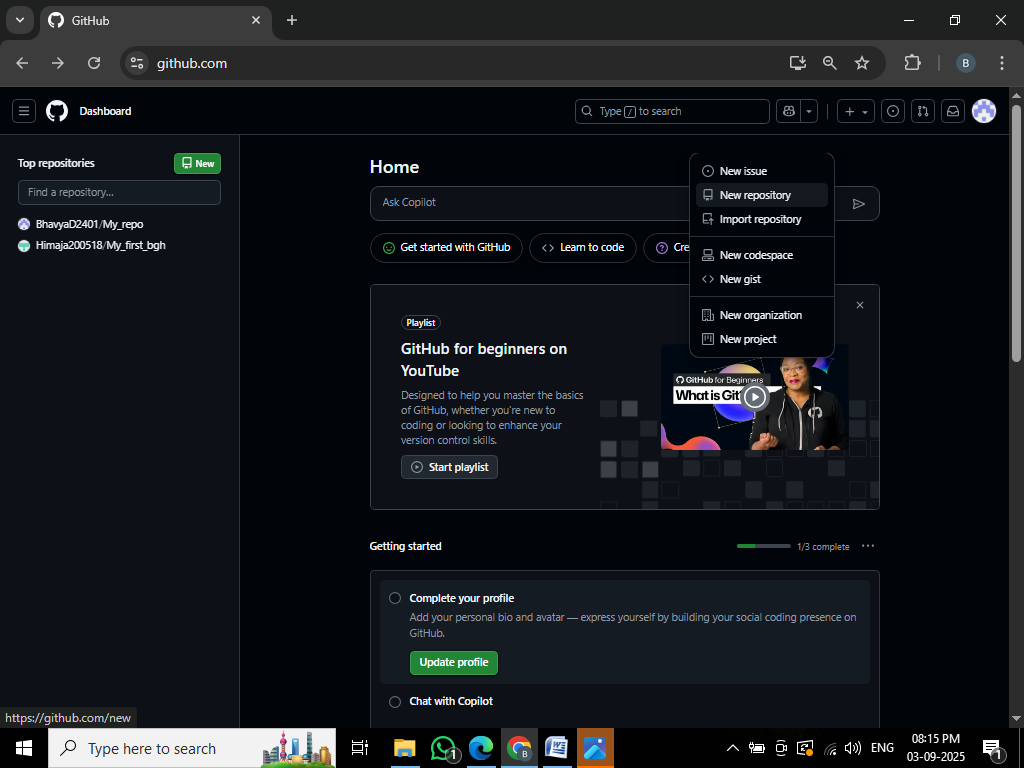
Fill the form:

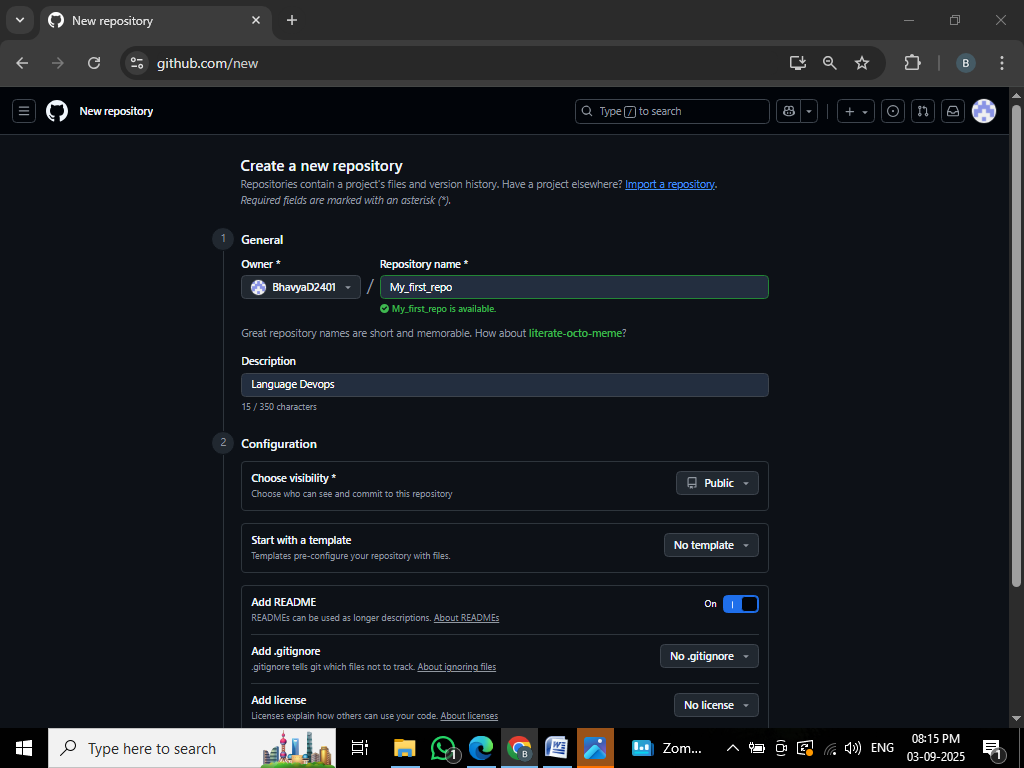
Repository name: my-first-repo

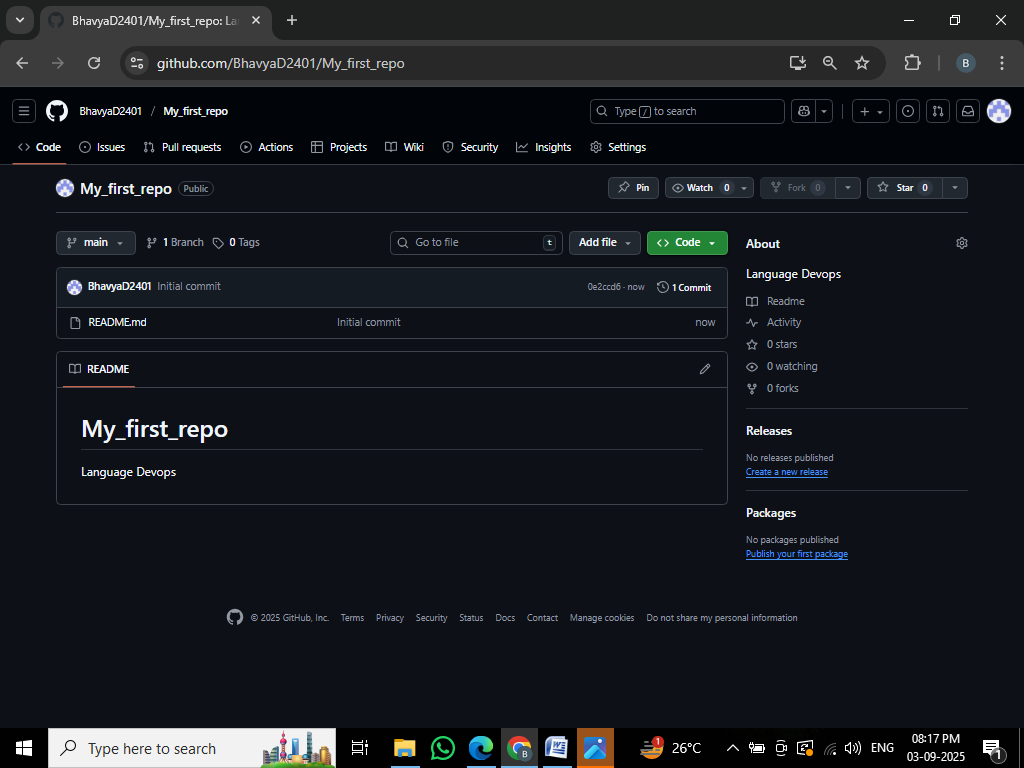
Description: Learning GitHub with DevOps

Check: “Initialize this repository with a README”

Click “Create Repository”







**Step 3**: Upload a File (hello.txt)

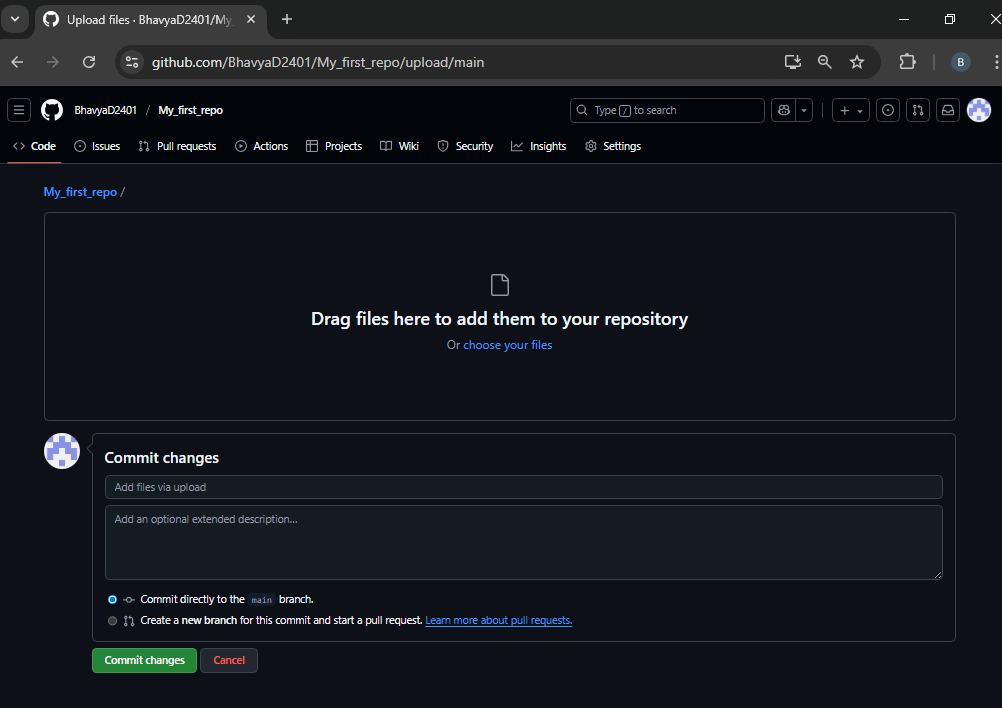
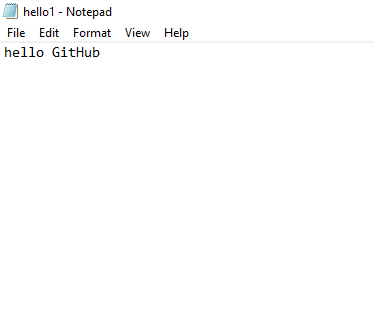
Inside your new repo, click “Add file” → “Upload files”

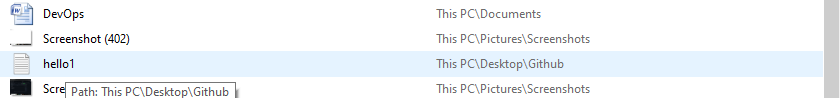
Drag & drop or select a file named hello.txt

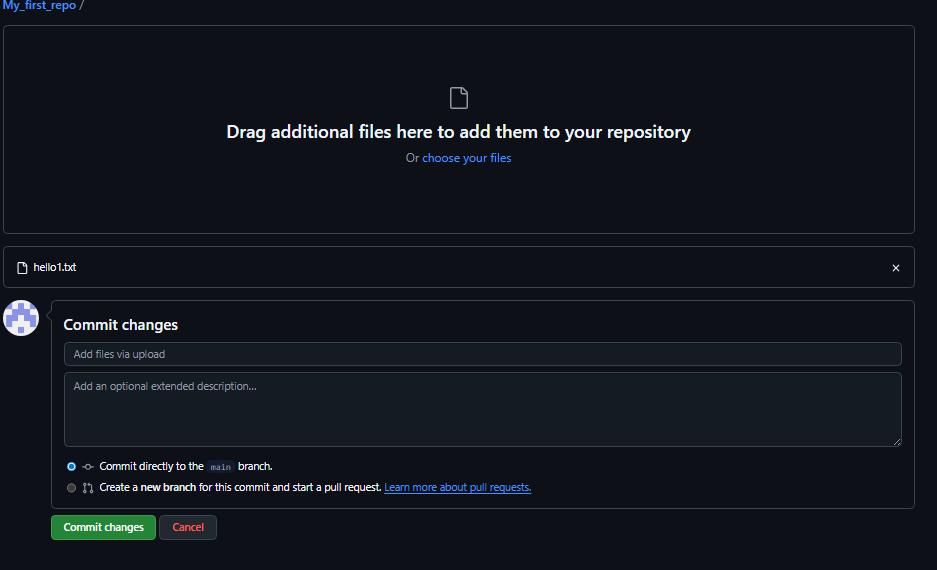
You can create it using Notepad or just type “Hello GitHub” inside

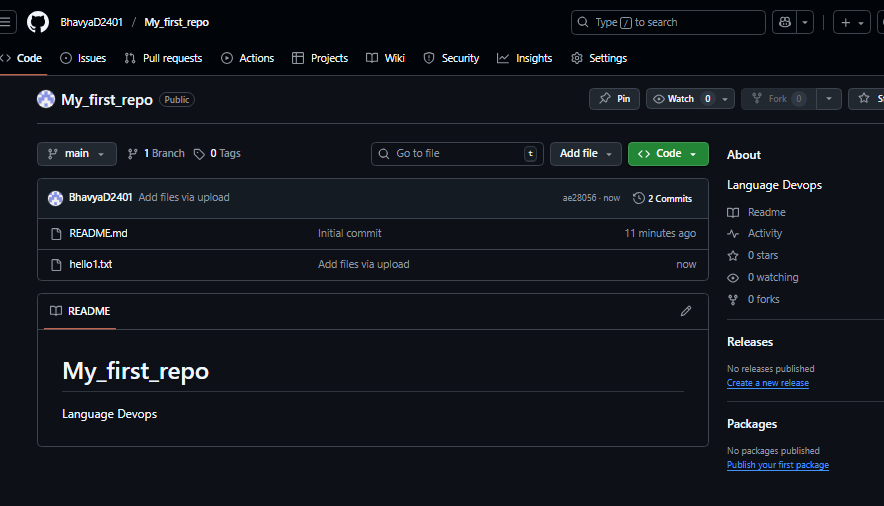
Scroll down → Add a commit message: added hello.txt

Click “Commit changes”









**Step 4**: Edit the File Online

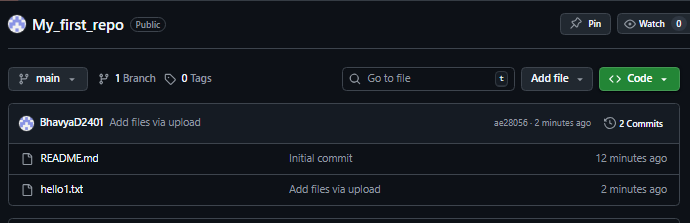
Click on hello.txt from the repo file list

Click the pencil icon (Edit this file)

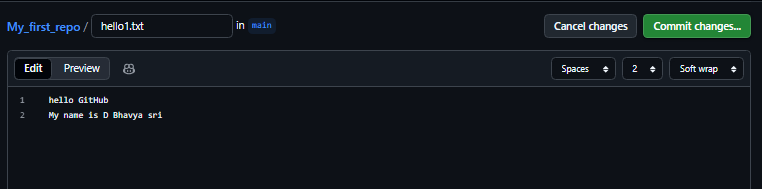
Change the content, e.g., add your name or a second line

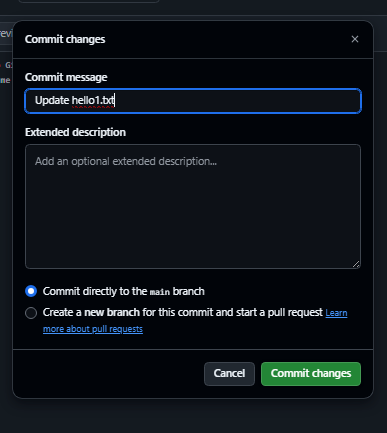
Scroll down, add a commit message: updated hello.txt

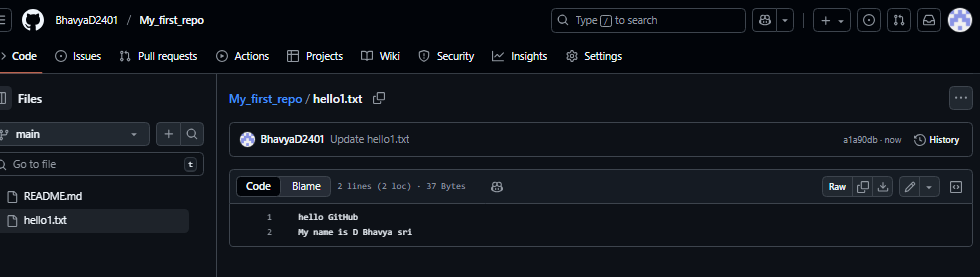
Click Commit changes

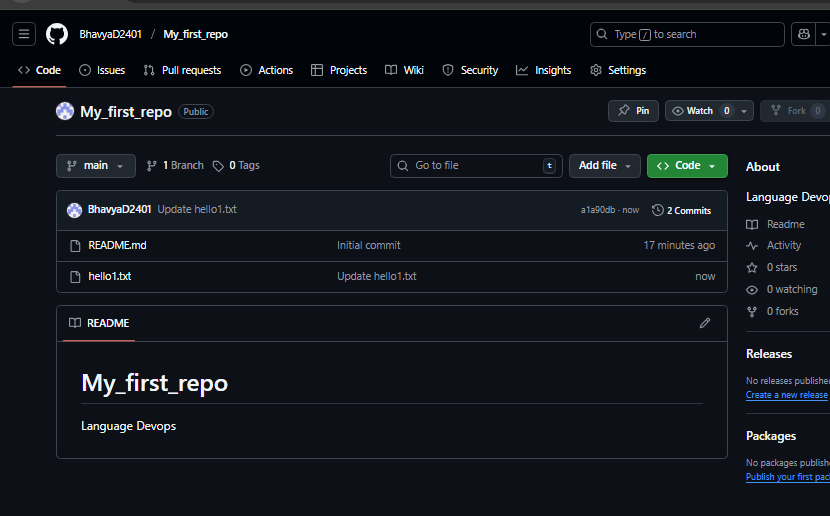










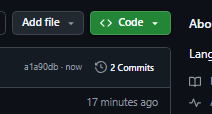


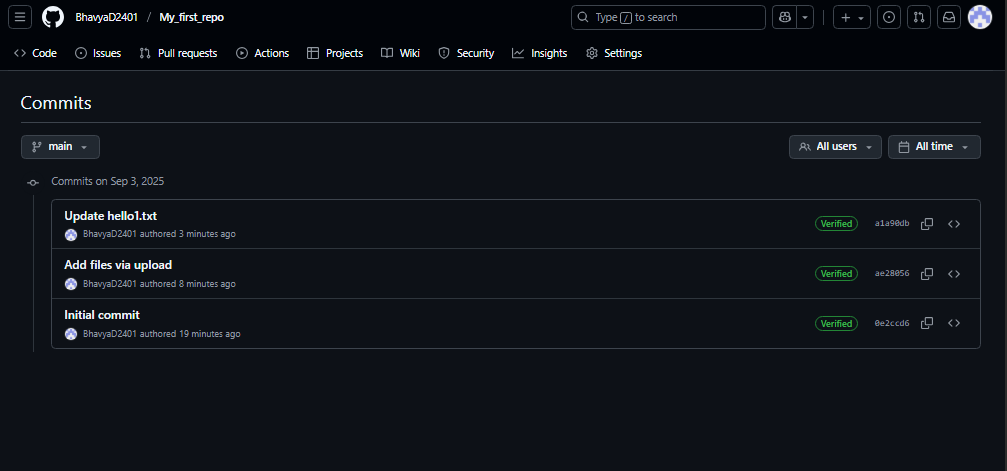
Step 5: View Version History

At the top of your repo, click the “Commits” tab

You will see all your commit messages and time of changes

Click on a commit to view the file version at that point





**REPO LINK:**

<https://github.com/BhavyaD2401/My_first_repo/commits?author=BhavyaD2401>

EXPERIMENT:2

Create a new file named about.txt in your GitHub repository. Inside the file, write your name, course, and one thing you learned about GitHub. Commit the file to your repository and share the link after completing the task.

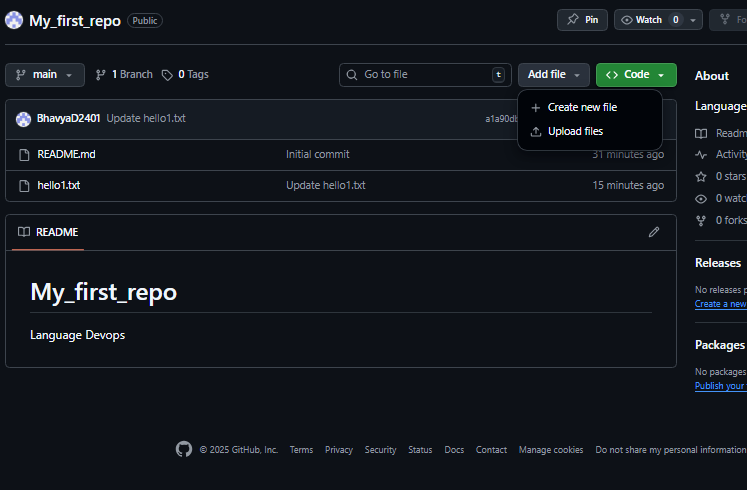
**PROCESS**: create a new file

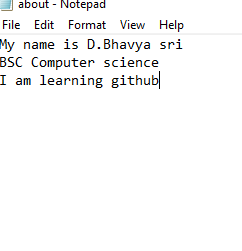
Name it as adout.txt

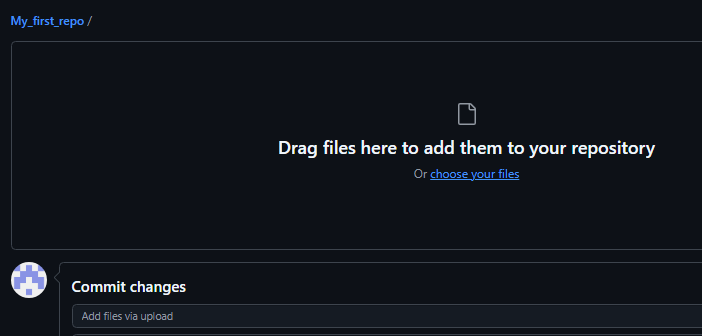
Wrie name,course,about github in that file

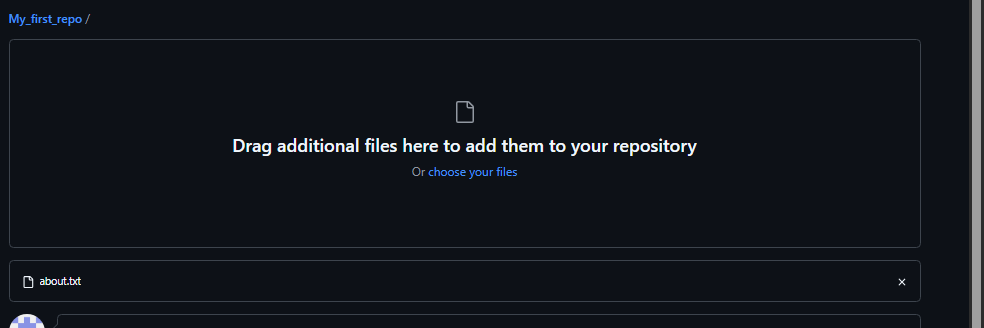
Commit the file

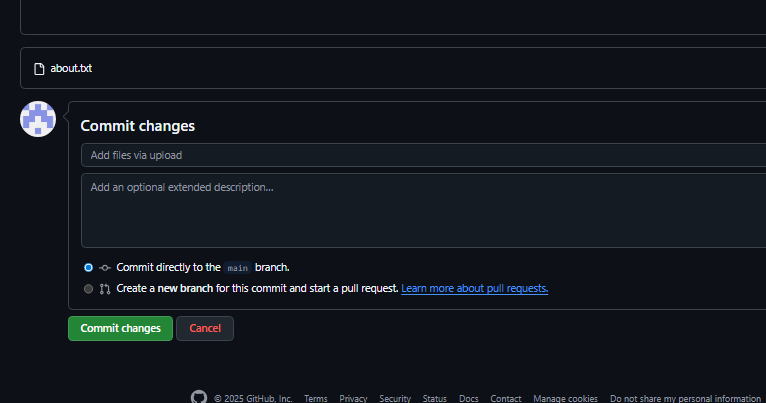
Copy the link of the path

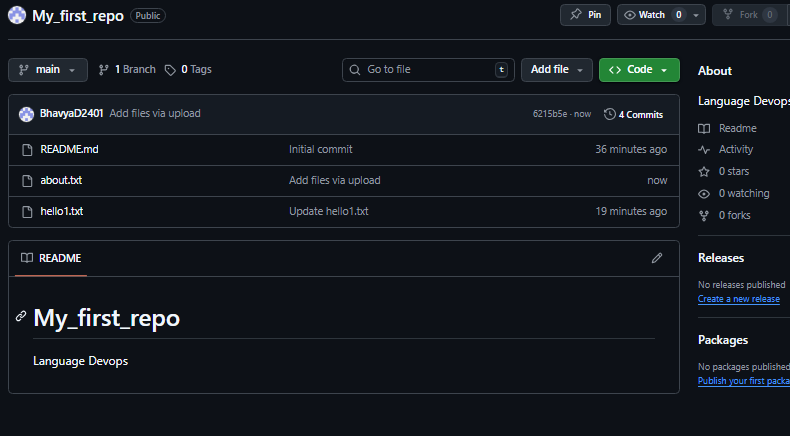


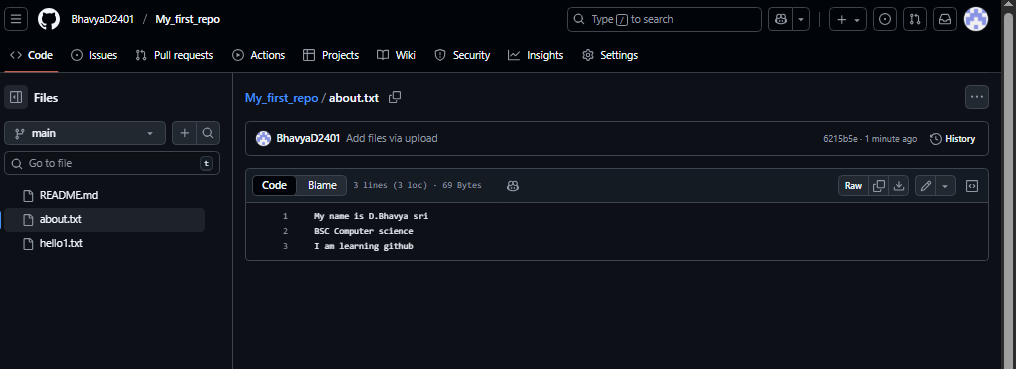












**File path:**

<https://github.com/BhavyaD2401/My_first_repo/blob/main/about.txt>

EXPERIMENT:3

Create a new GitHub repository named student-portal-demo. Initialize it with a README file describing a basic idea for a student portal application. Share the link to your repository after completing the task.

**PROCESS:**

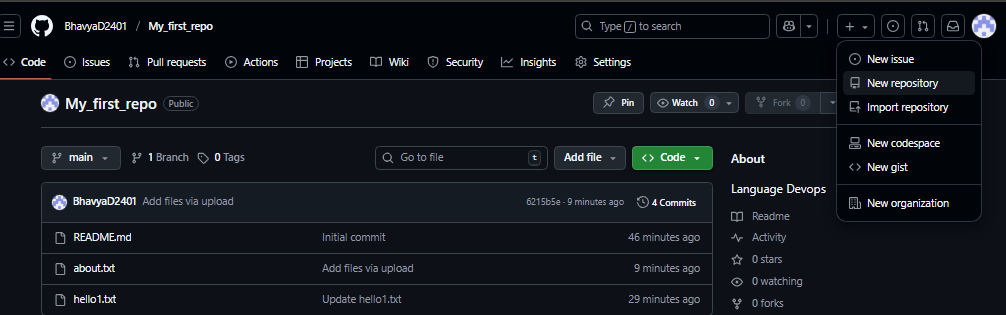
create a new repository

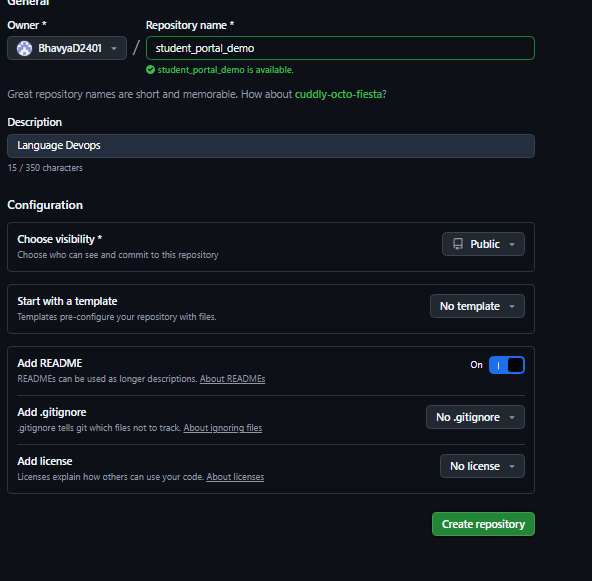
initialize it with README

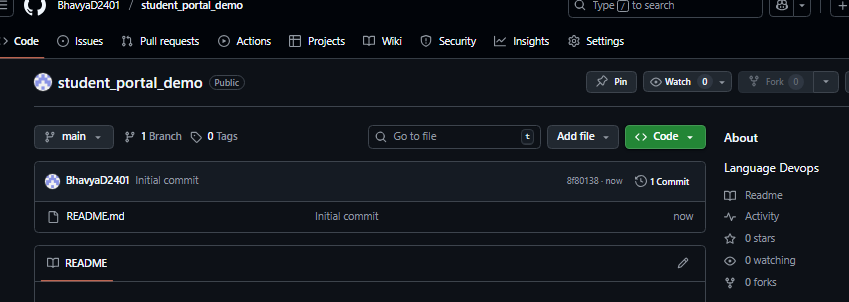
create a new file in it

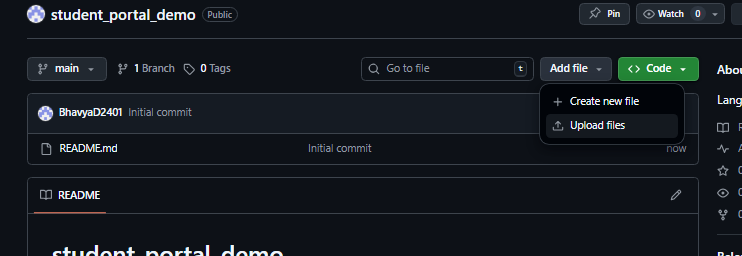
add about some details of student application

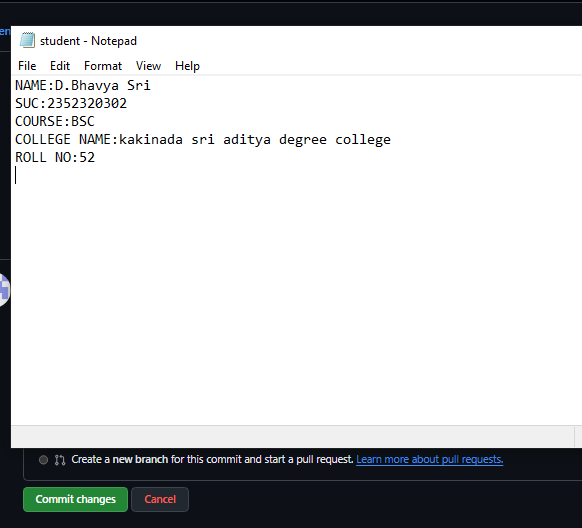
share the link path

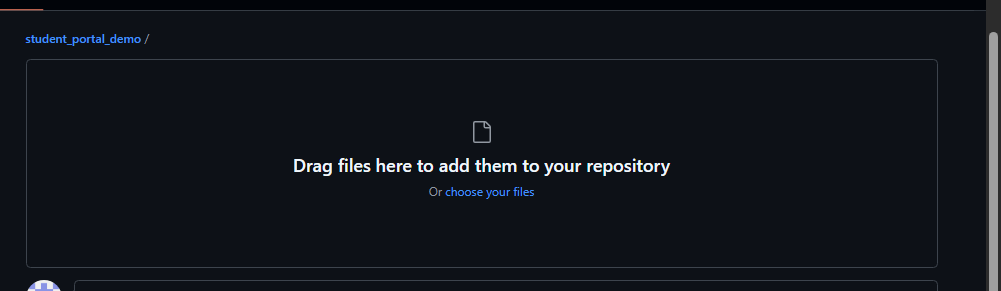


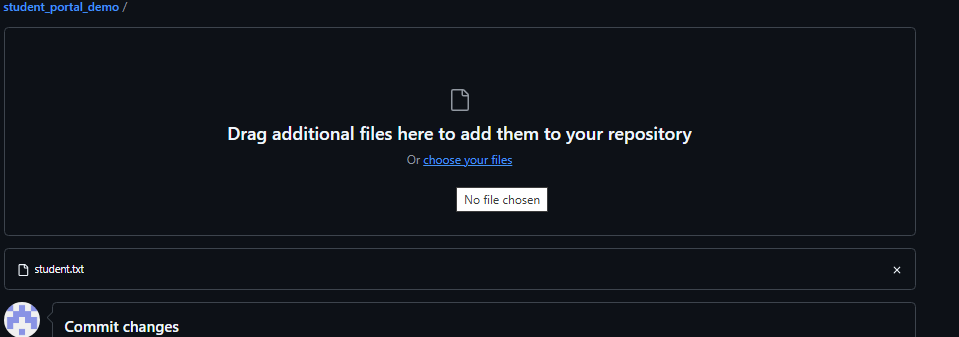


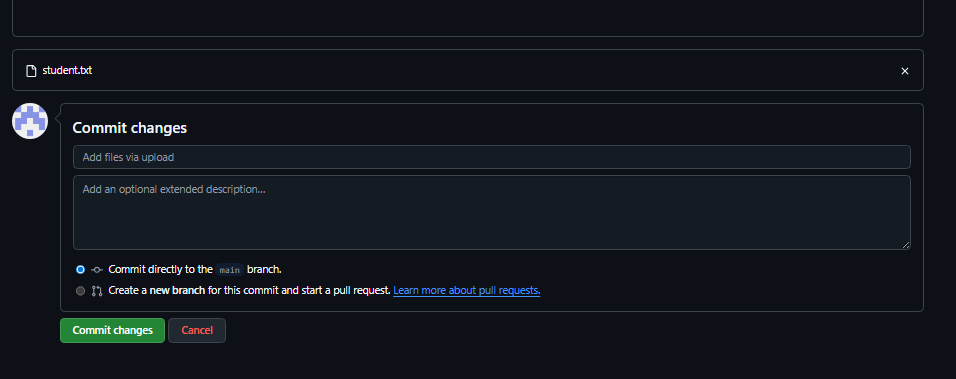


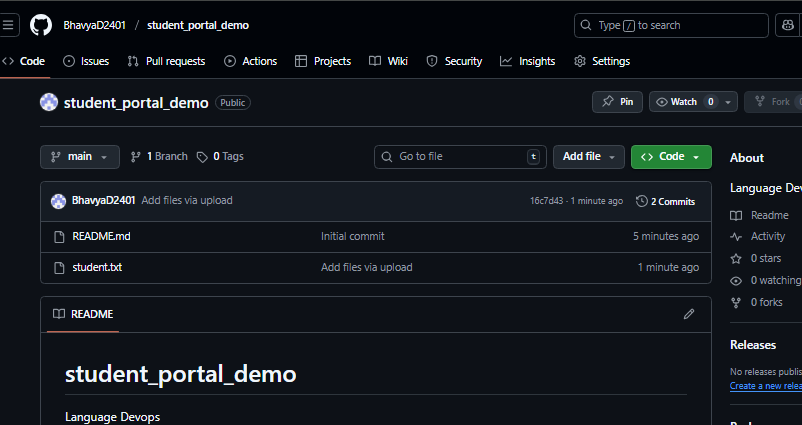


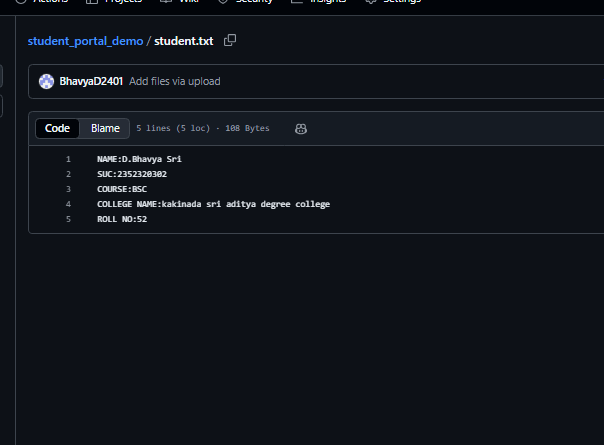












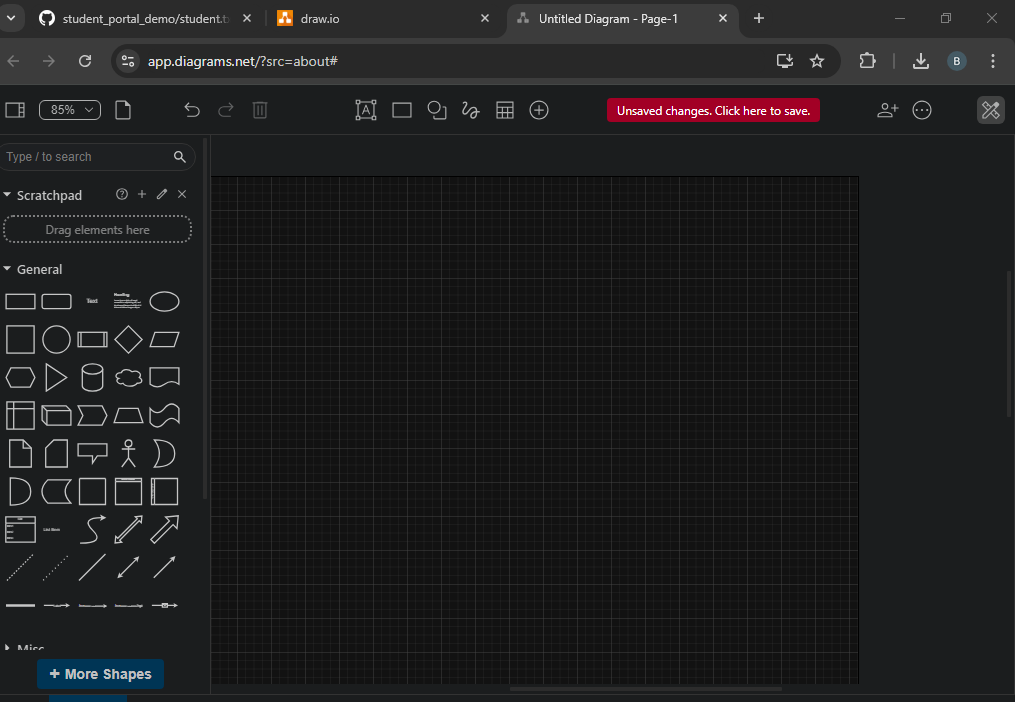
**File Path:**

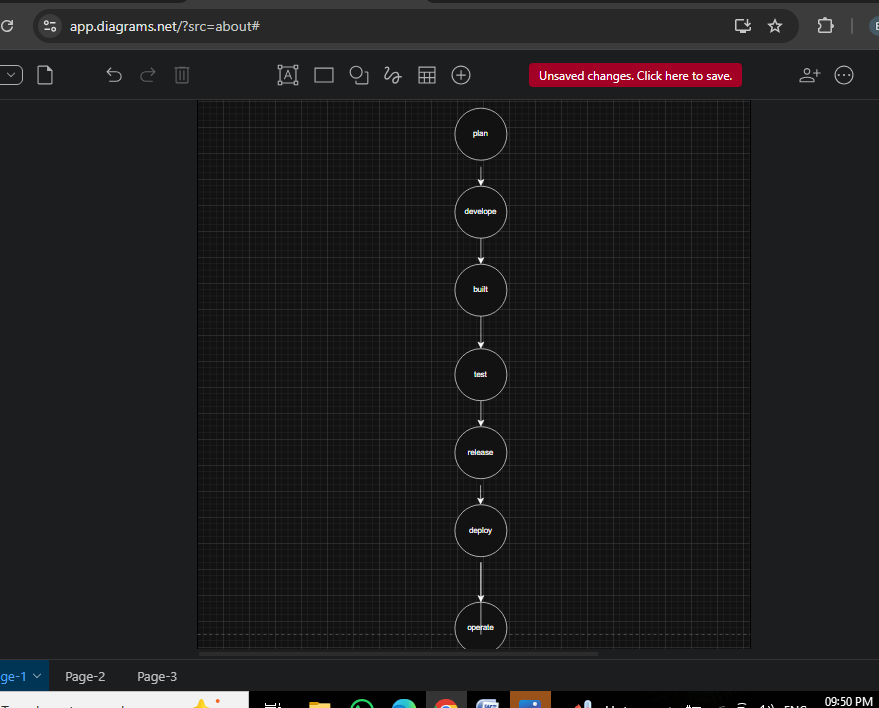
<https://github.com/BhavyaD2401/student_portal_demo/blob/main/student.txt>

**Experiment:4**

**Design a DevOps Lifecycle Using diagrams.net Tool:**

Go to <https://www.drawio.com/>





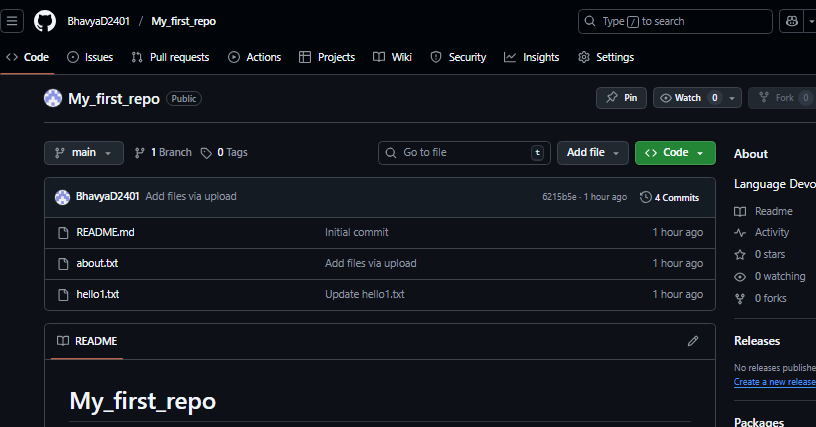
Upload to GitHub:

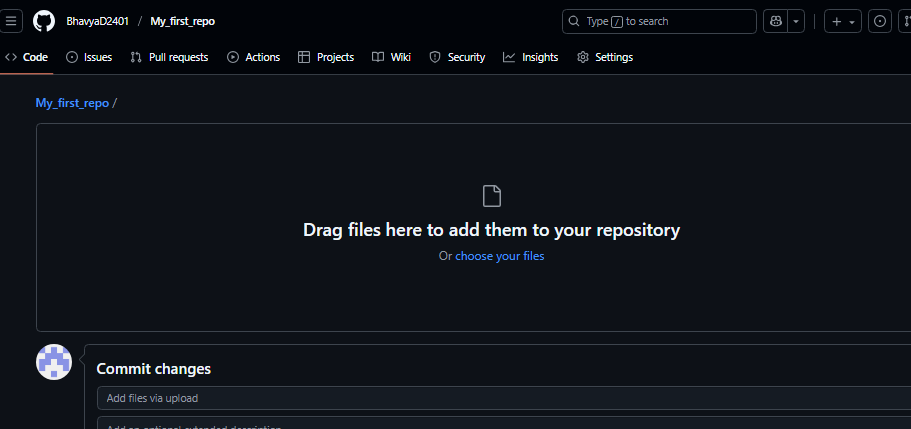
Go to your repository

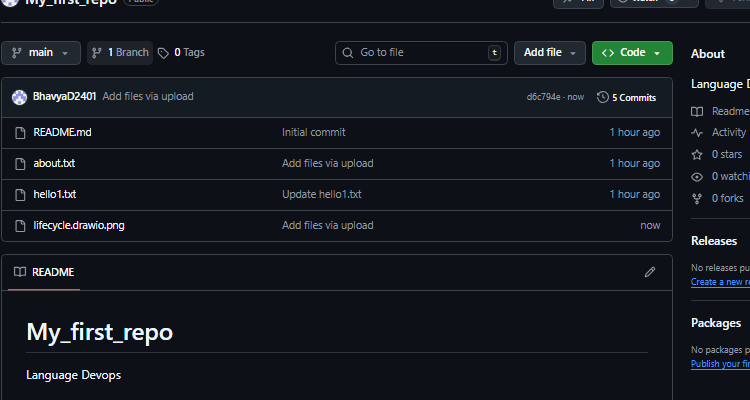
Click “Add file → Upload files”

Upload the exported PNG file

Commit with message: Added DevOps Lifecycle

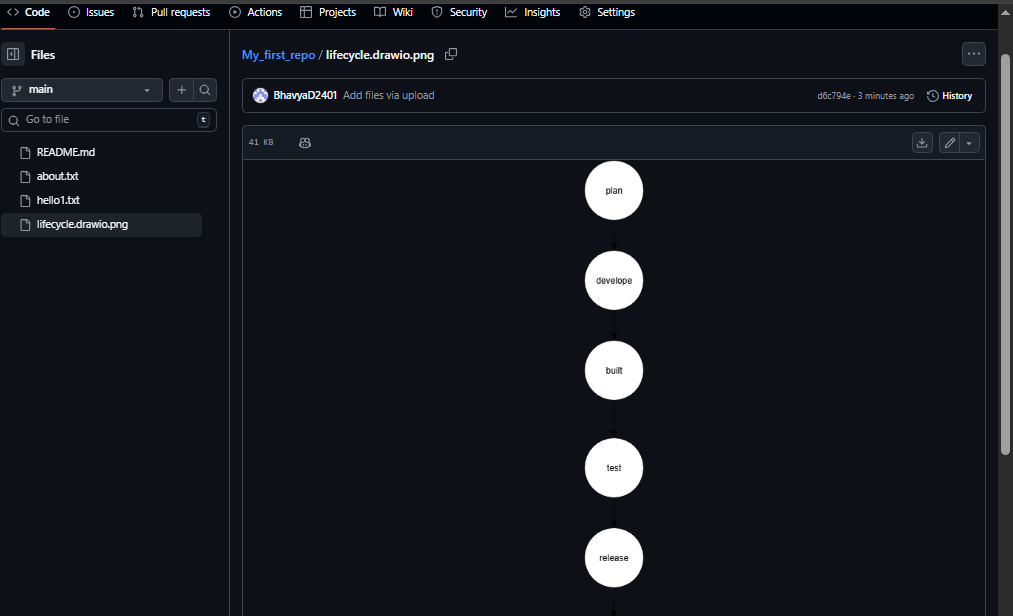
====





Expected File Name in GitHub Repo:

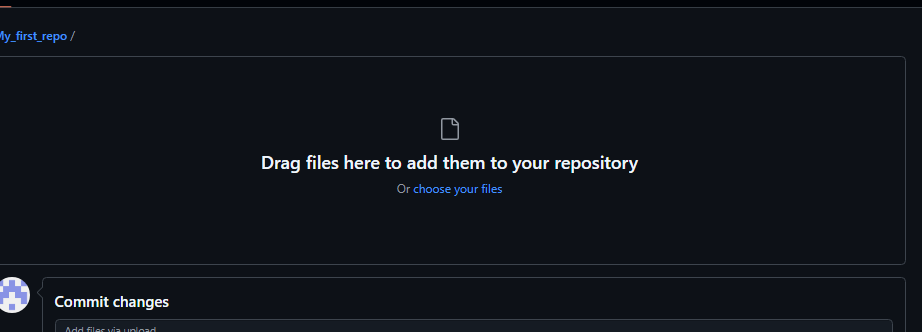
devops-lifecycle.png

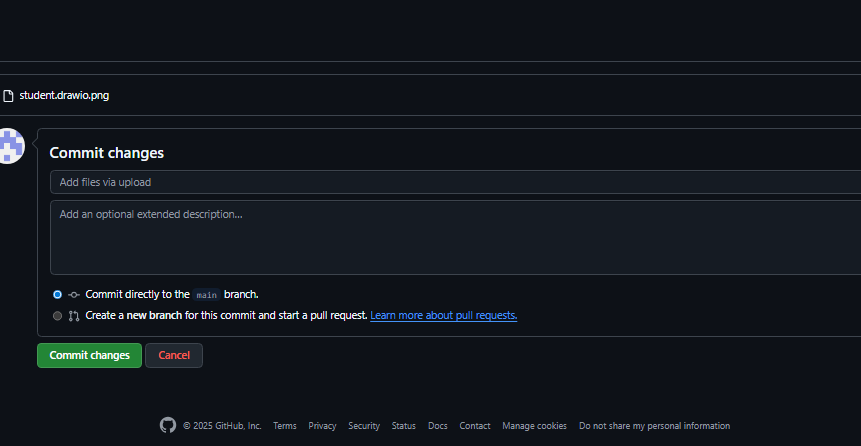


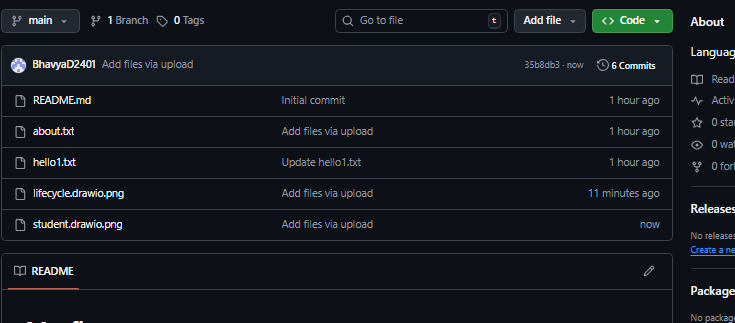
**Experiment: 05**

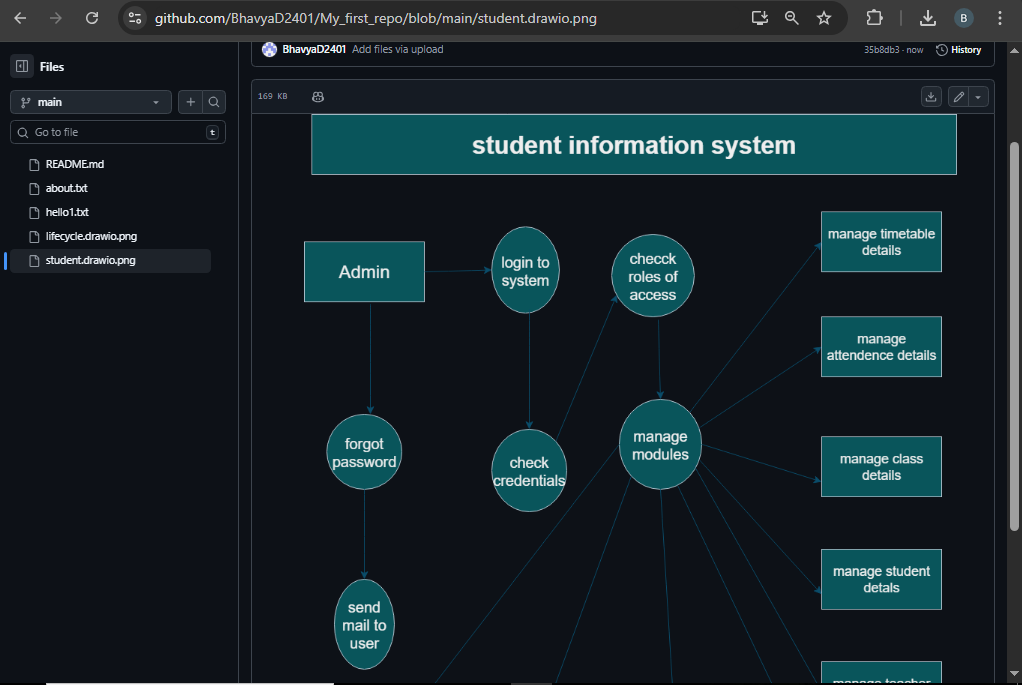
Design a DevOps Lifecycle diagram for a Student Information System.









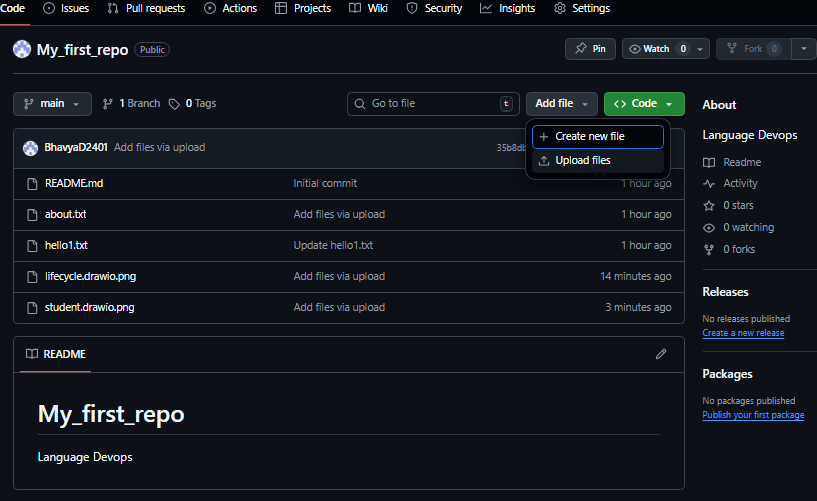


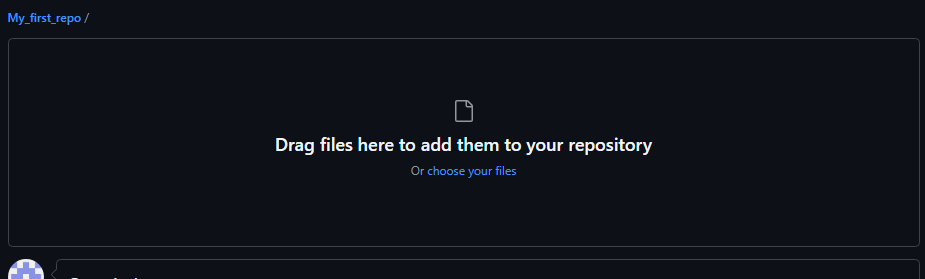
**File path:**

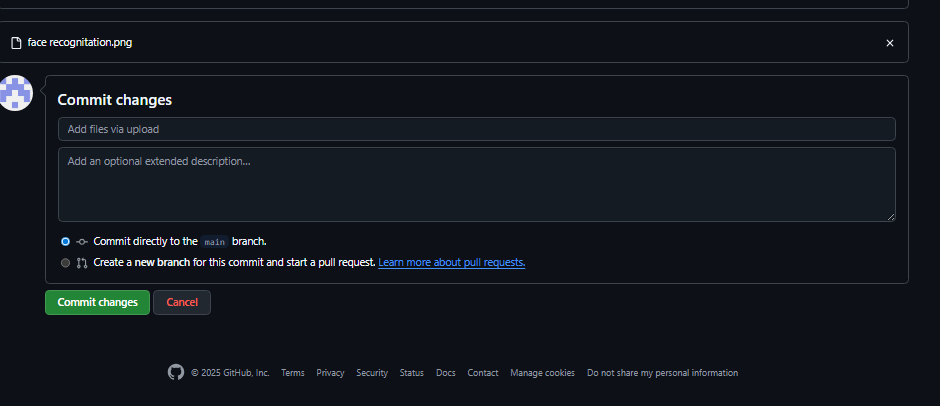
https://github.com/BhavyaD2401/My\_first\_repo/blob/main/student.drawio.png

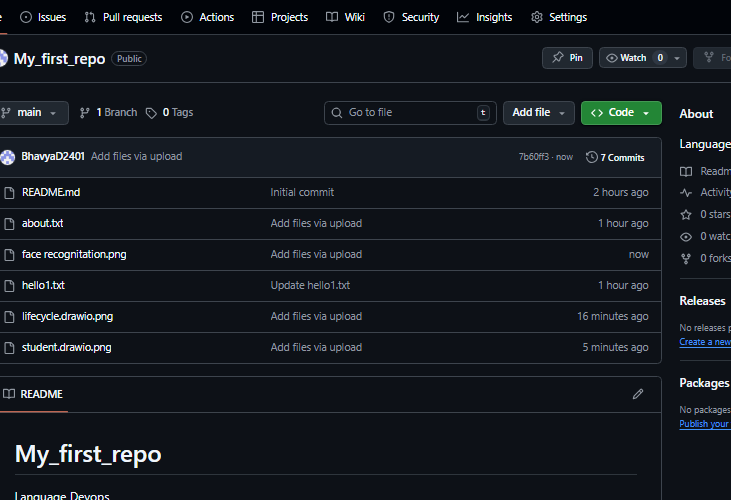
**Experiment: 06**

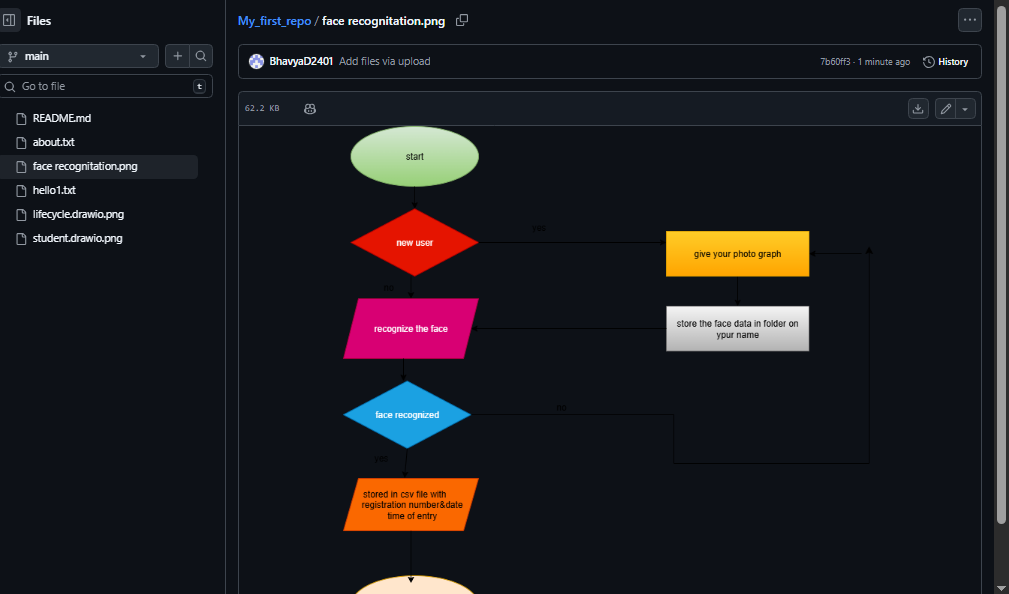
Design a DevOps Visualization Diagram – Face Recognition Attendance System











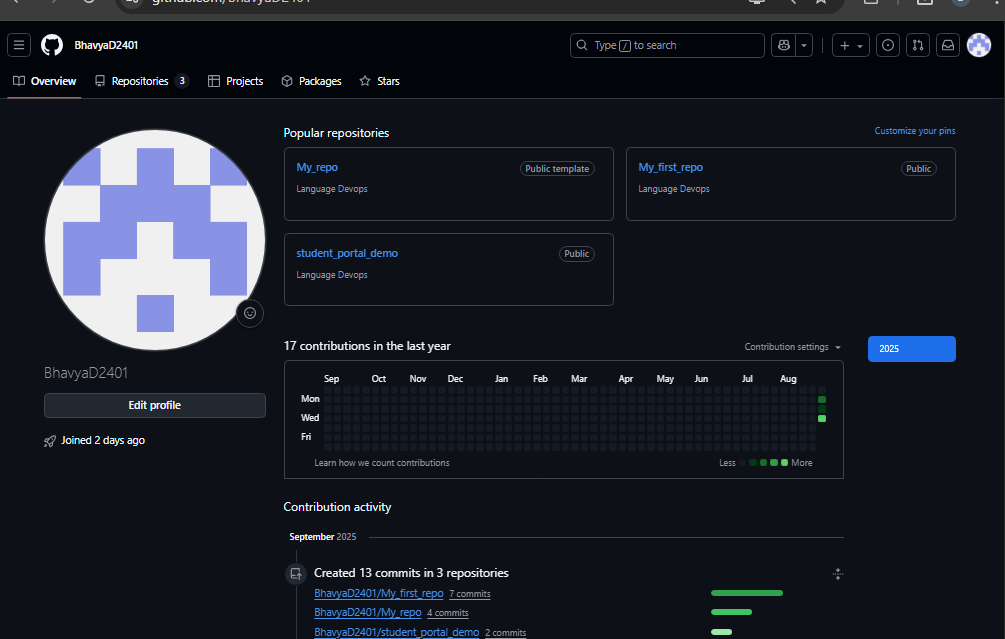
File path:

https://github.com/BhavyaD2401/My\_first\_repo/blob/main/face%20recognitation.png

Experiment no: 7

Renaming the Repository using Github Web Interface

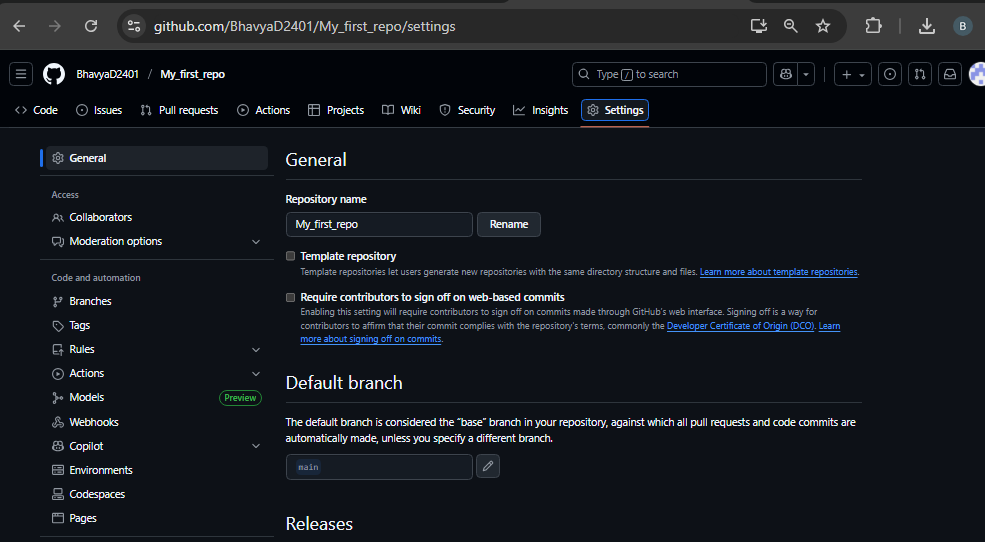
1. Select your Repository that you wanted to rename



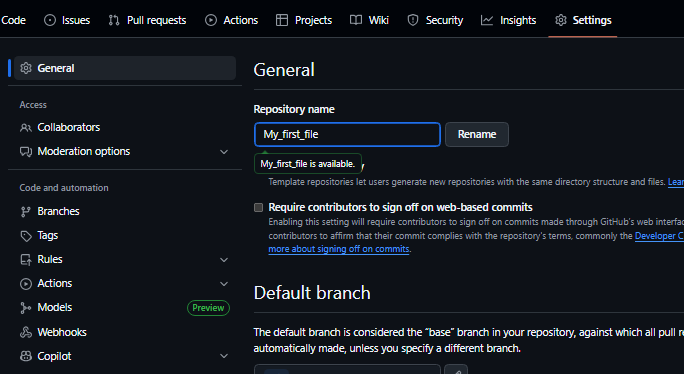
2.goto settings

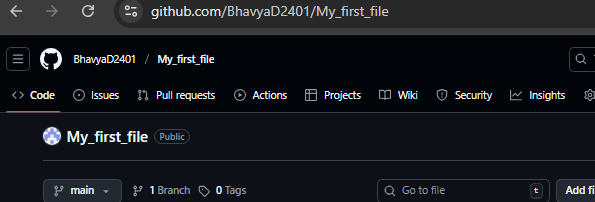


3. In General tab, at the top you will find Repository name label



4. Rename it as per your requirement and click on Rename button to apply changes





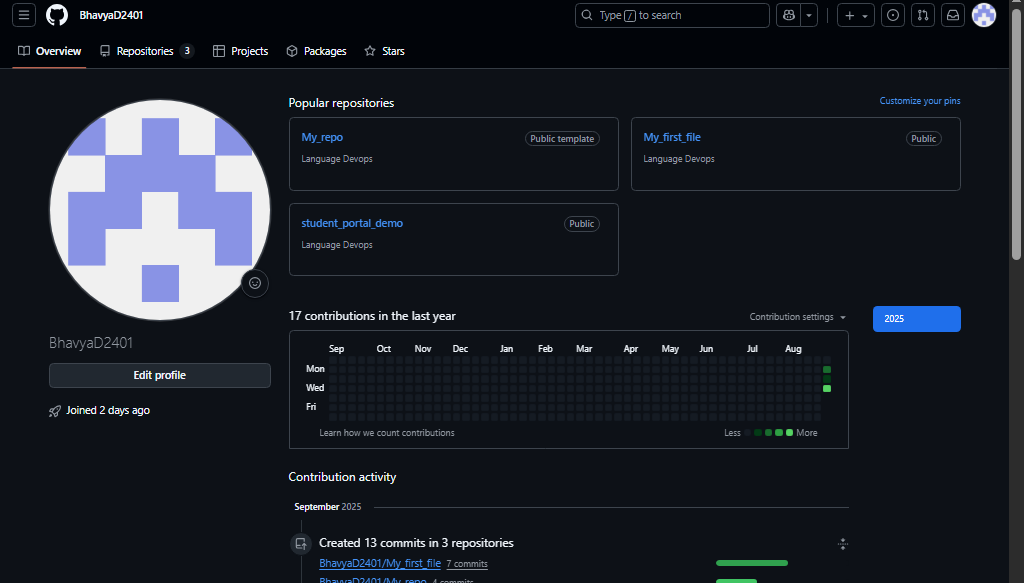
File path:

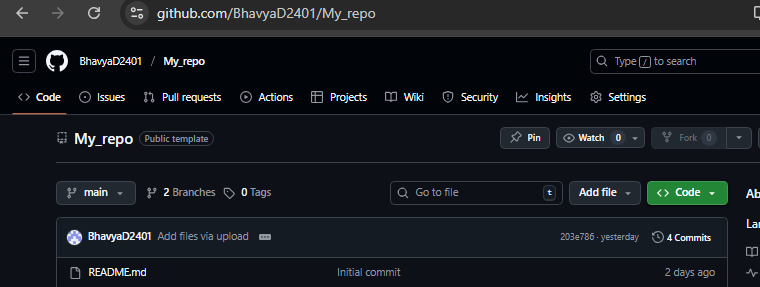
http://github.com/BhavyaD2401/My\_first\_file

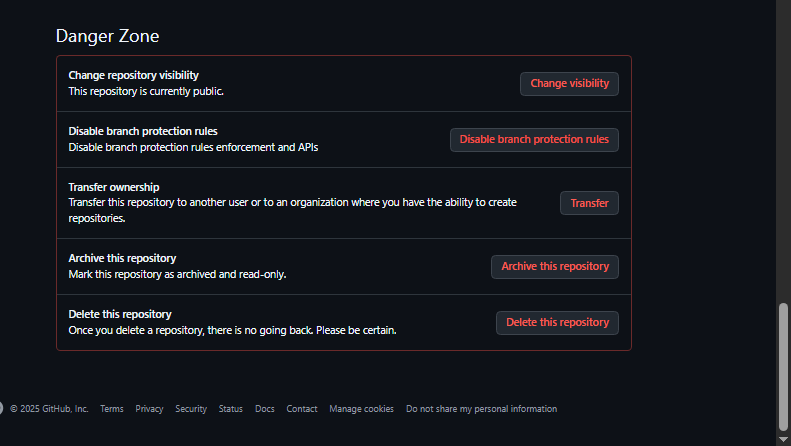
**Experiment no: 8**

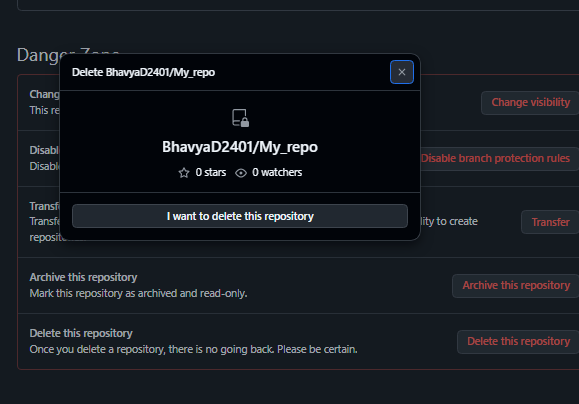
Deletion of Repository using Github Web Interface

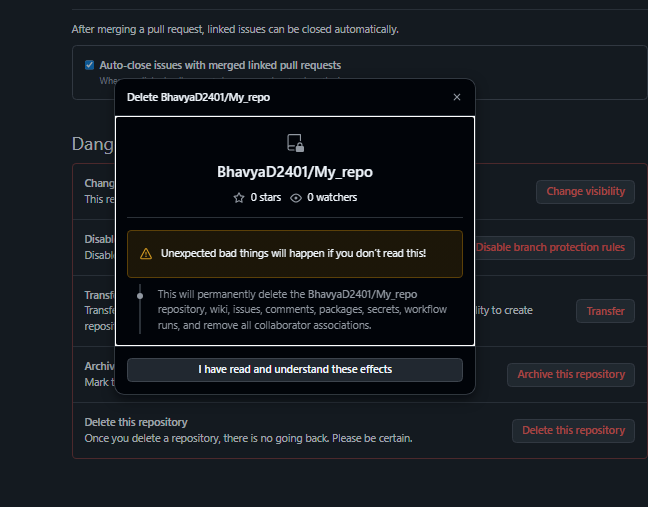
1. Select your Repository that you wanted to delete



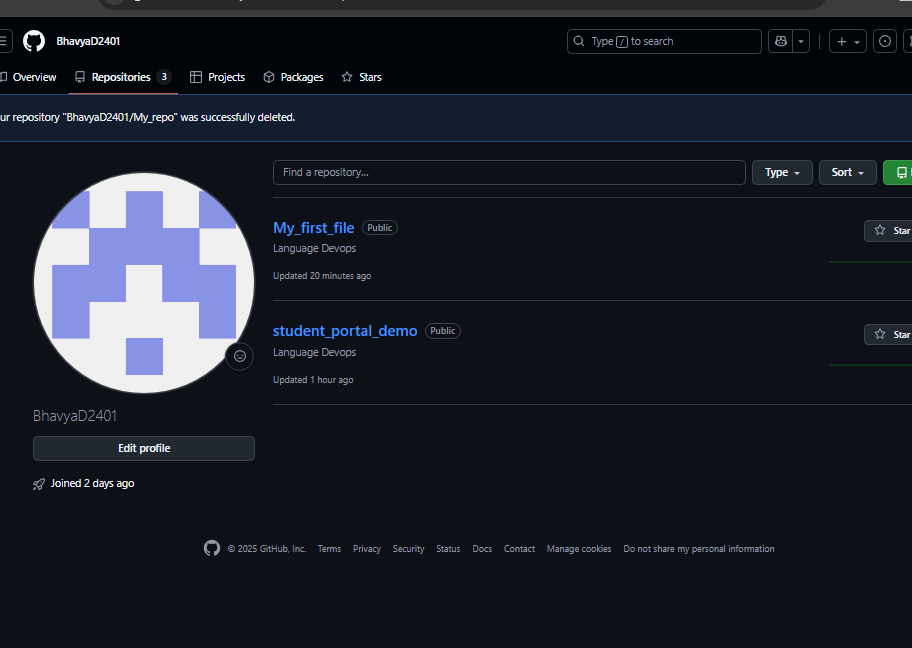






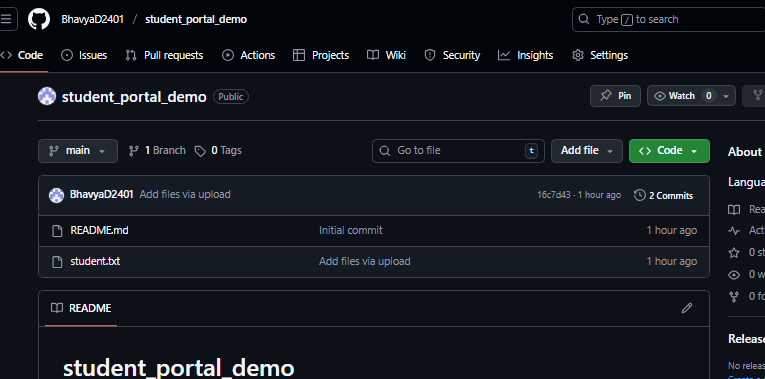


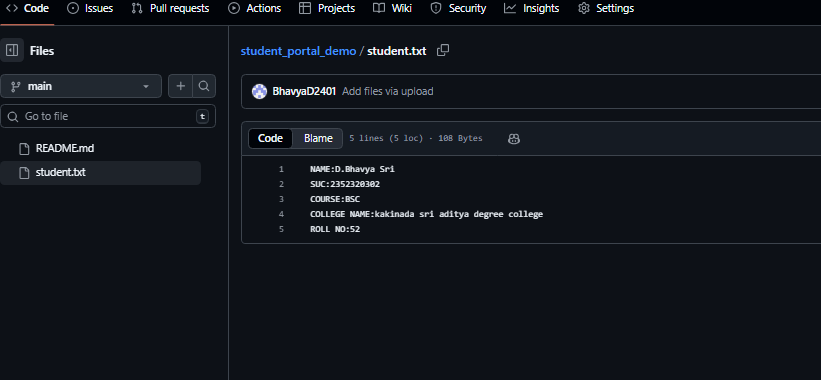


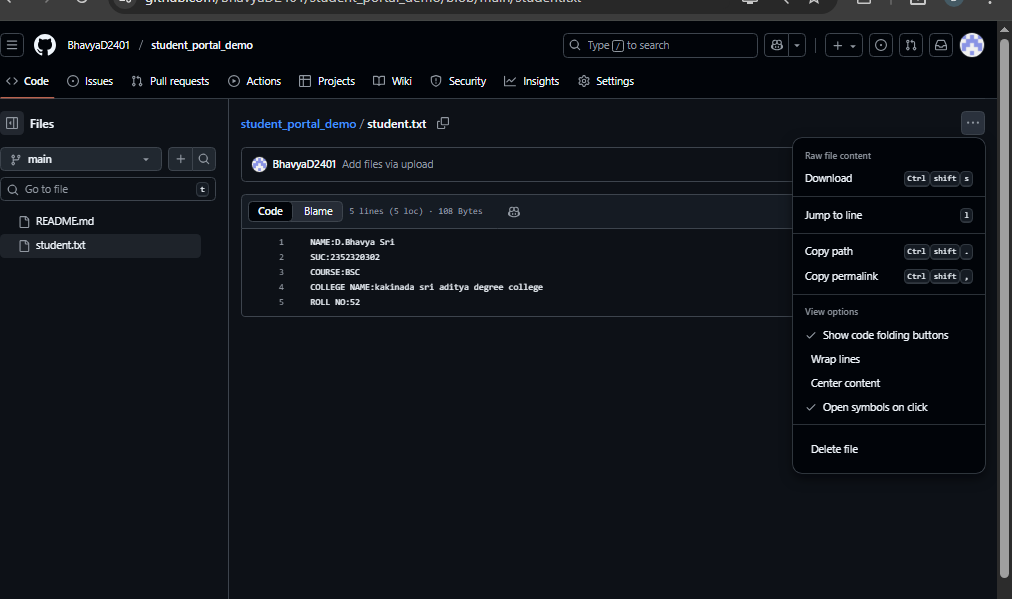


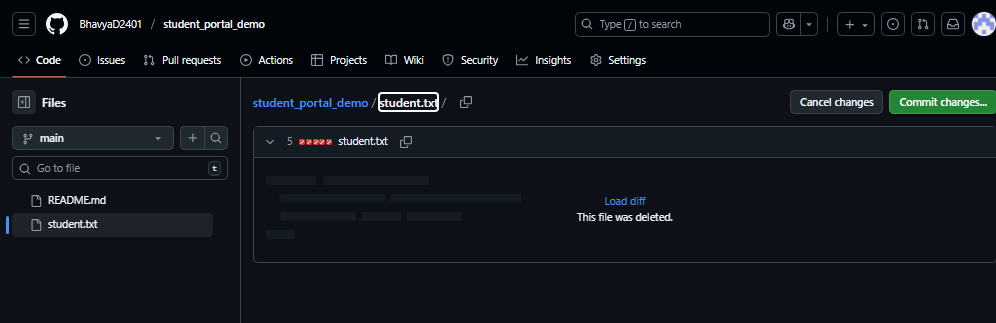
**Experiment no: 9**

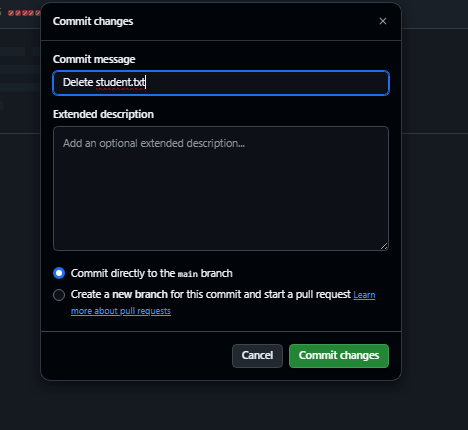
Deletion of file from Repository using Github Web Interface

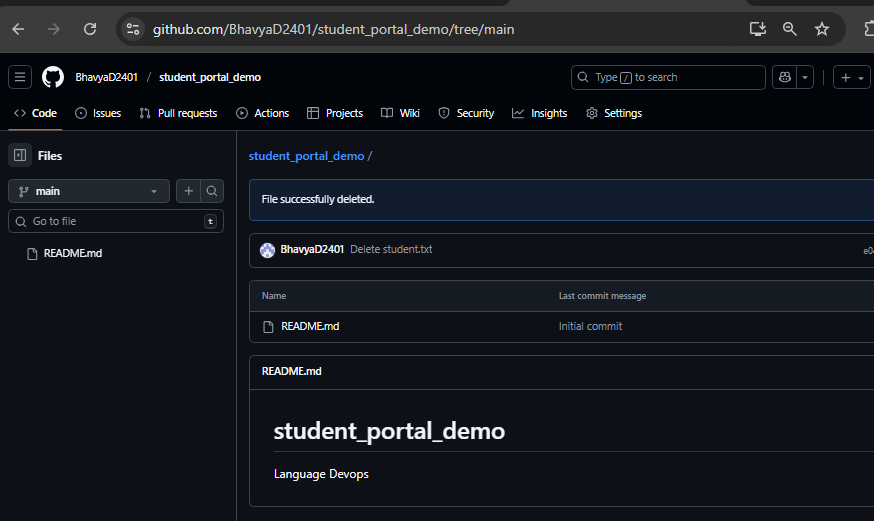












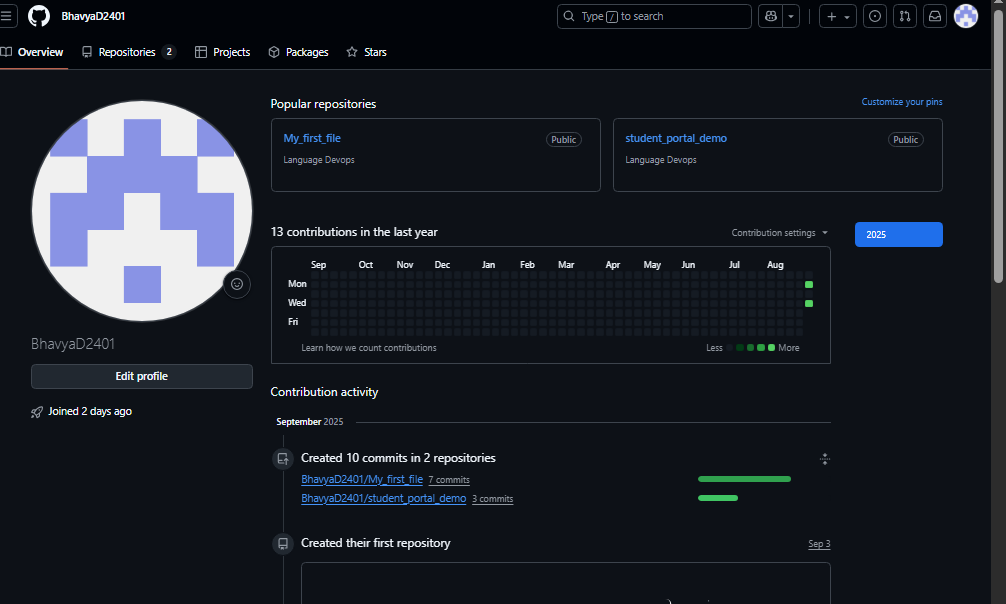
**Experiment 10:**

Design a DevOps lifecycle that could be applied to a Food Delivery App like Zomato, Swiggy, or Uber Eats.

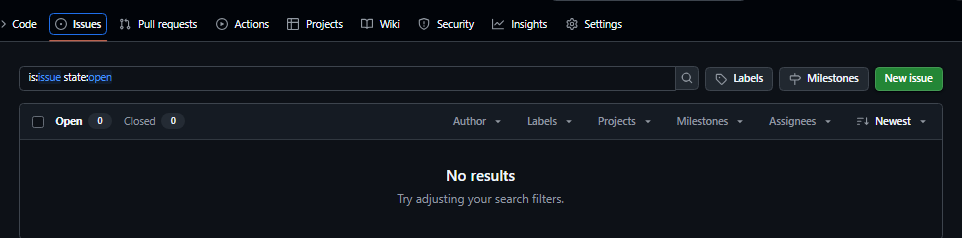
**Experiment 11:**

Design and build a basic Web Application based on your own idea using a shared GitHub repository with 2 team members collaborating.

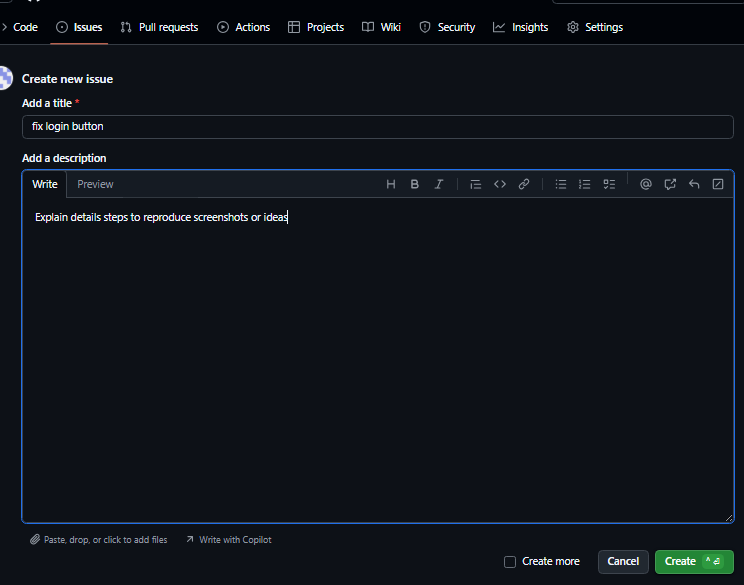
Go to the GitHub repository.



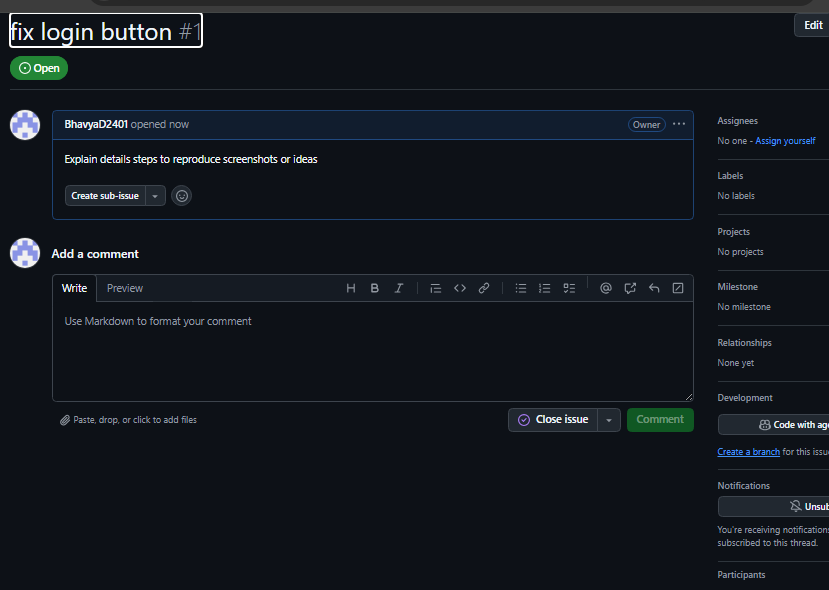
At the top, click the “Issues” tab.



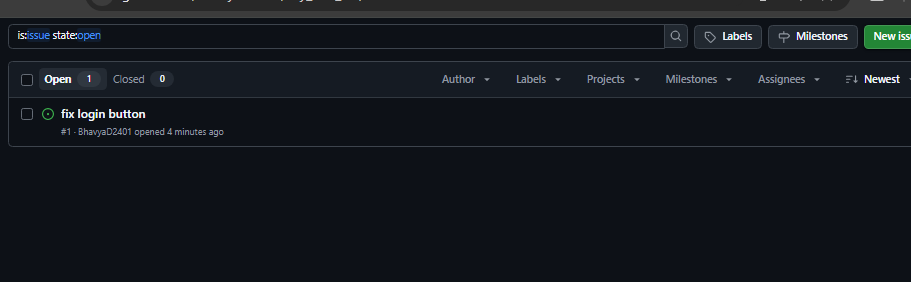
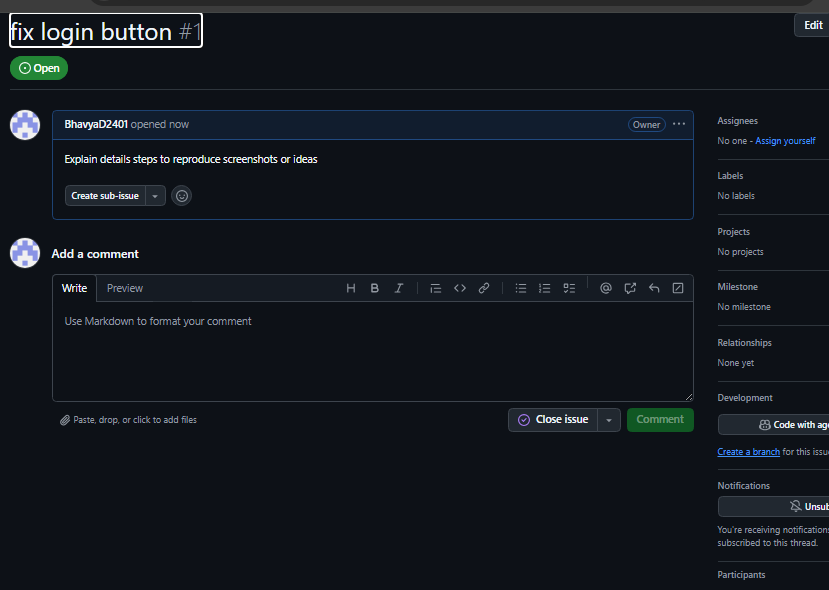
Click “New Issue”

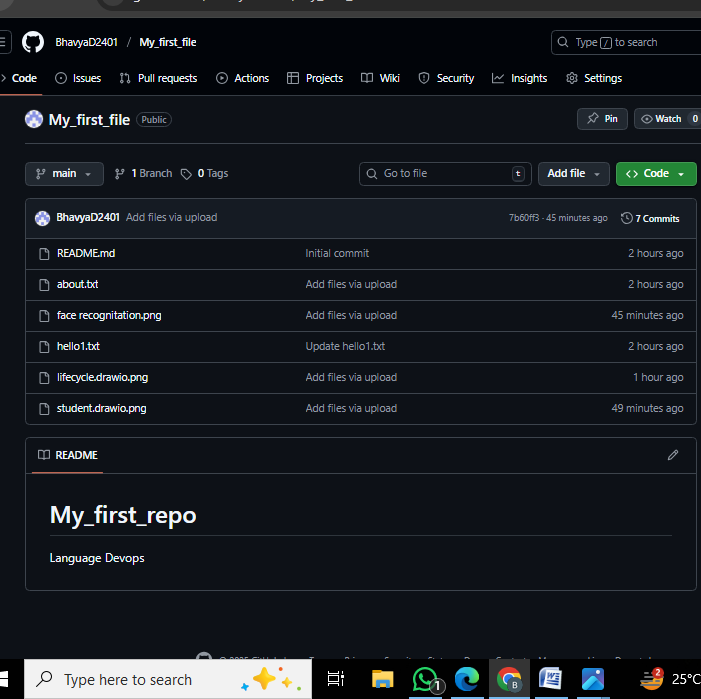


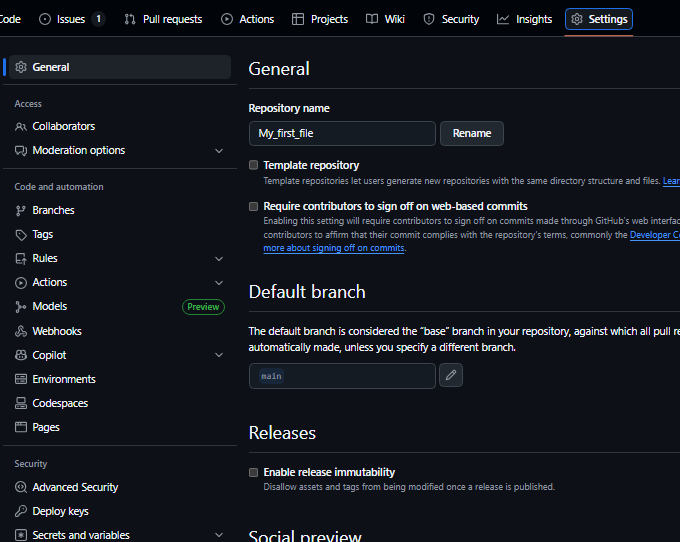
Click on create button to create a new issue.

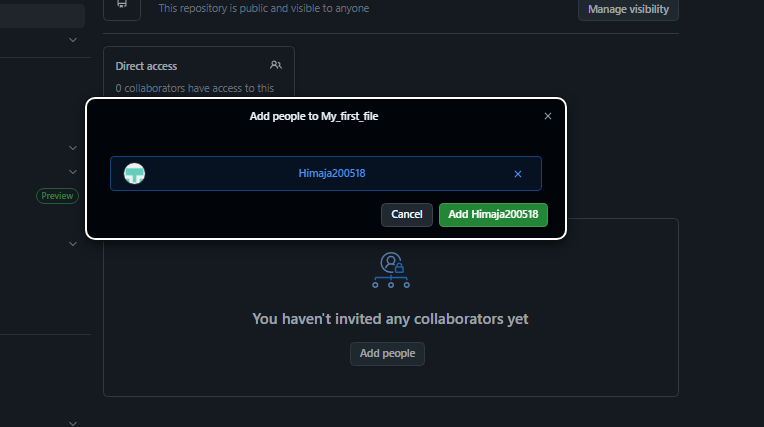


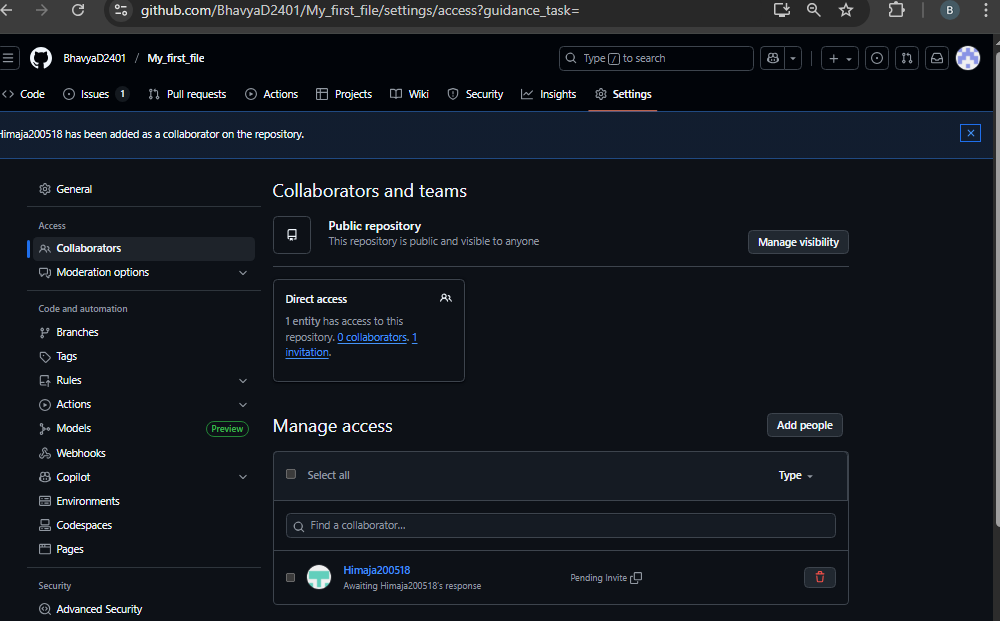
Now, your issue will be created and added to the Issues tab of the repository.

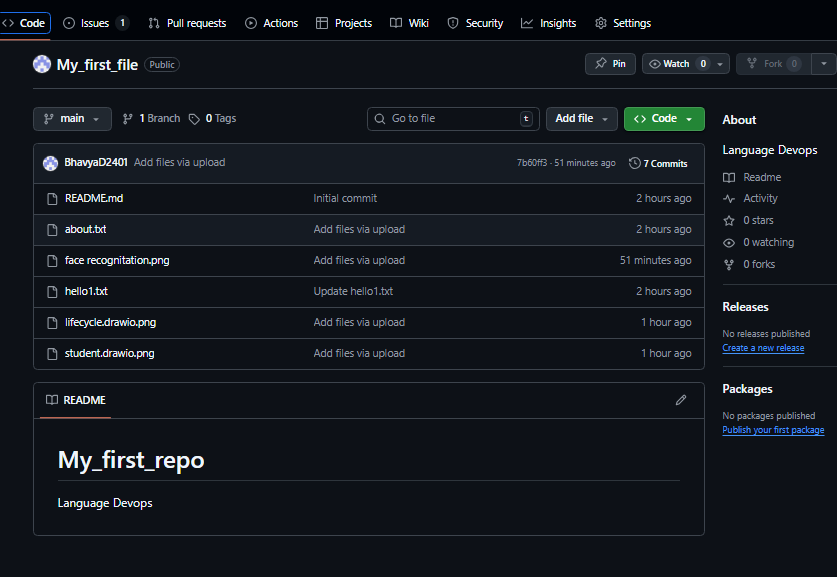


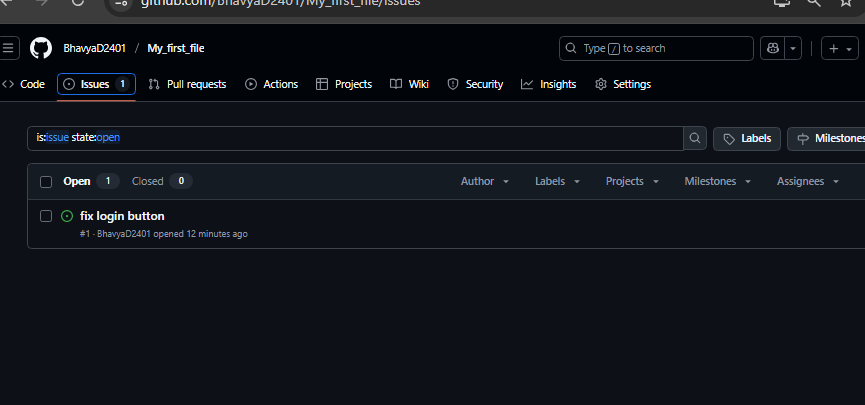


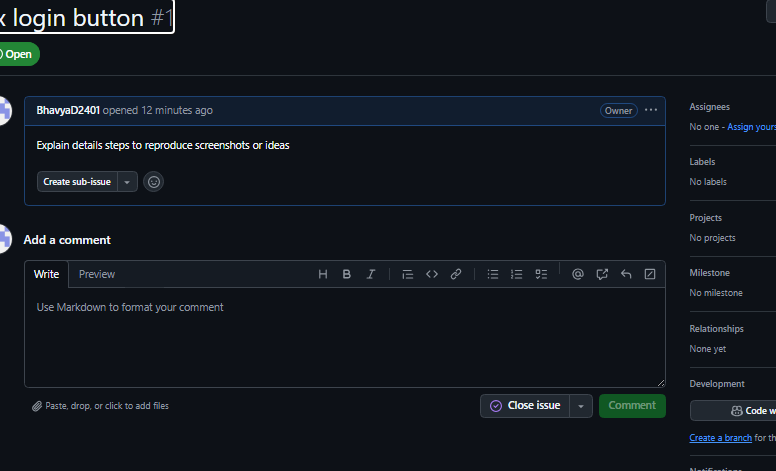


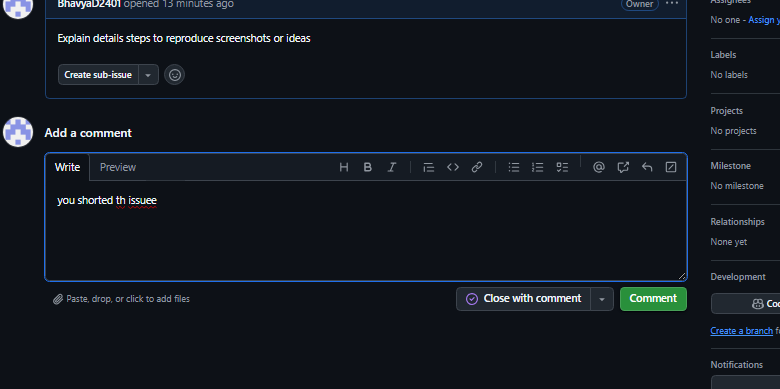


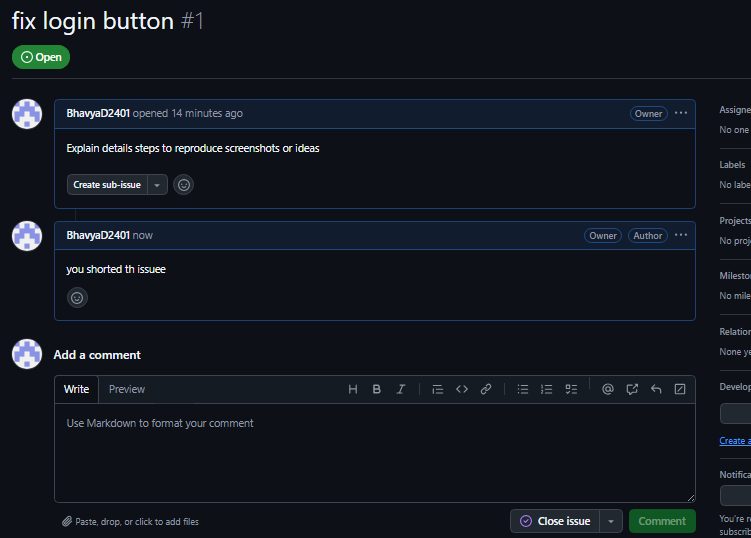


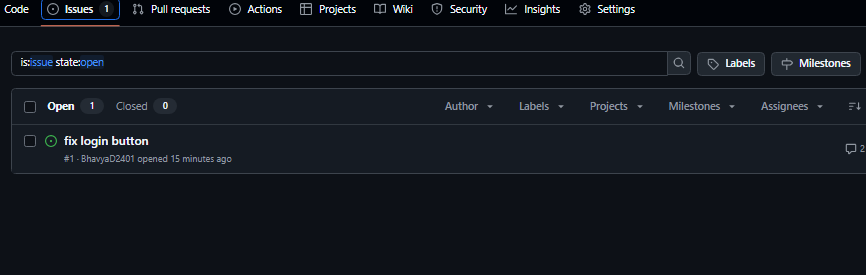












**Experiment-12**

GitHub Pull Requests (PRs):

Creating and Using Pull Requests:

**Experiment 13:**

Demonstrate how to upload a project to GitHub and deploy it using Vercel, implementing a basic CI/CD workflow.

Go to https://github.com

Click + New Repository

Name: simple-cicd-demo

Keep it Public

Once created, click “Add file” → “Upload files”

Drag and drop all the files & folders from the unzipped folder.

Click Commit changes.

Step 2: Deploy on Vercel Go to <https://vercel.com>

Sign up or log in with your GitHub account

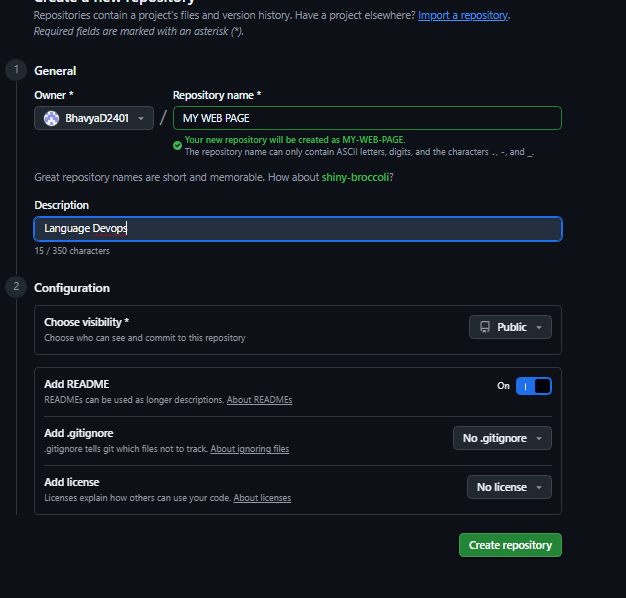
Click “Add New” → “Project”

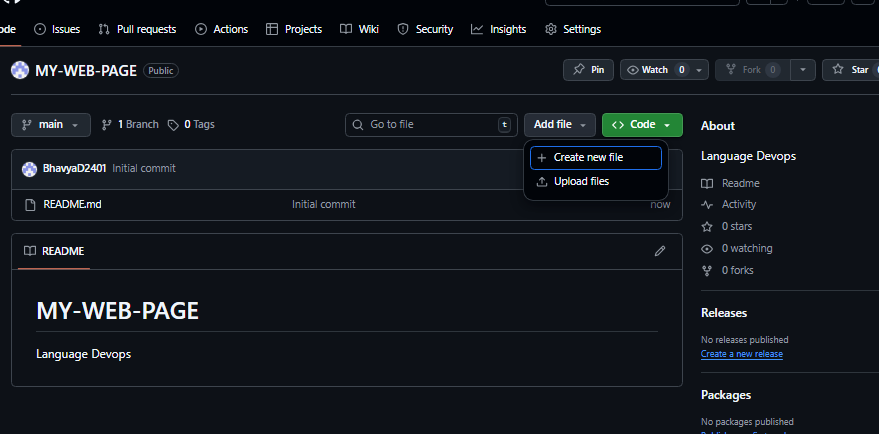
Choose the simple-cicd-demo repo from your GitHub

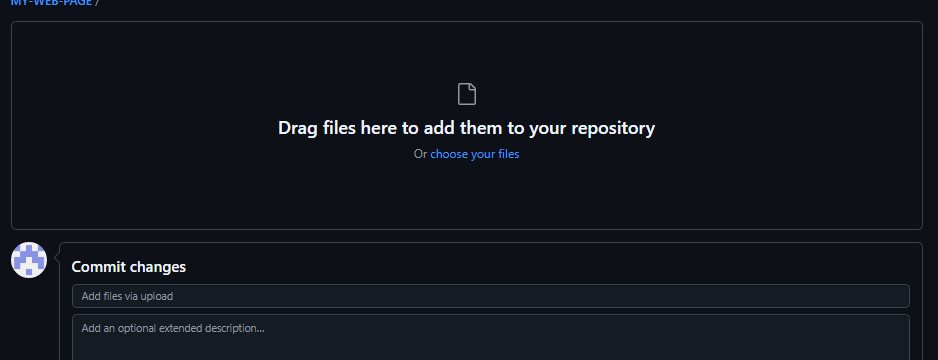
Click Import

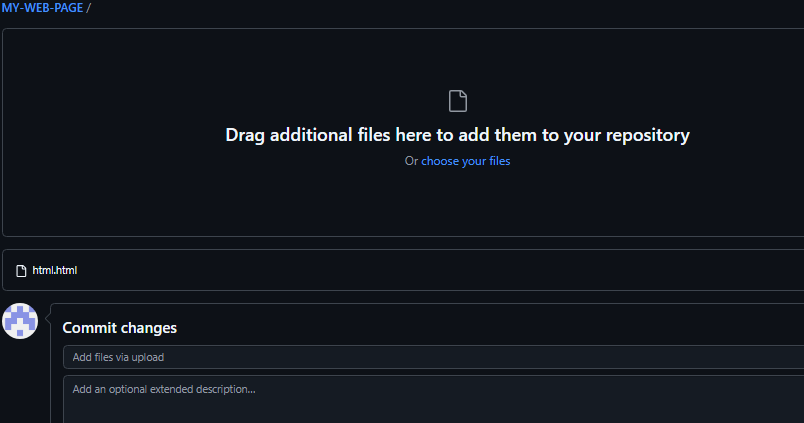
No need to change any settings

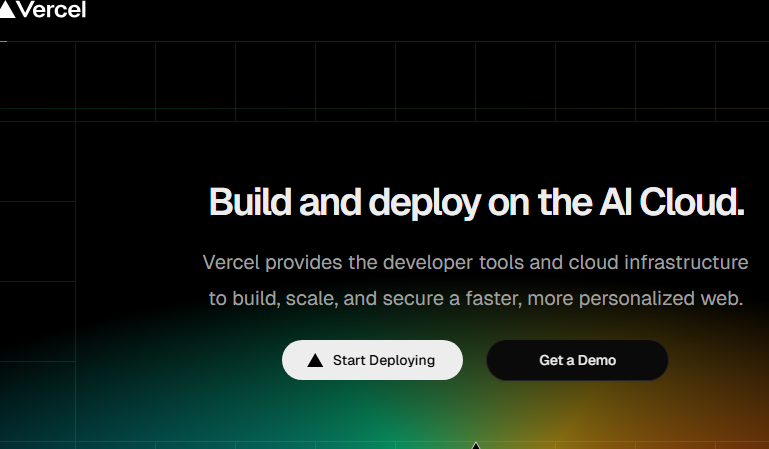
Click Deploy

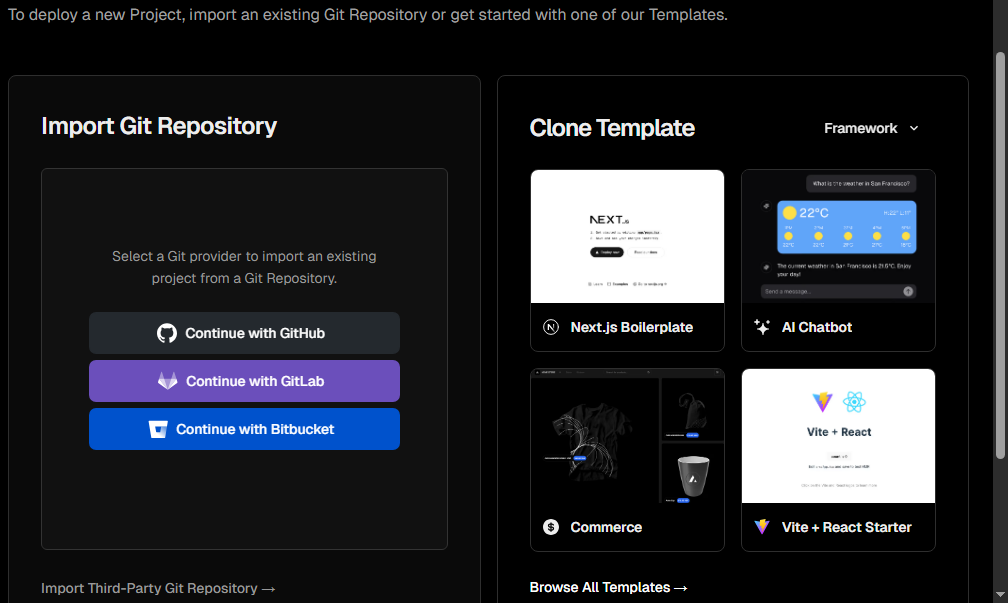


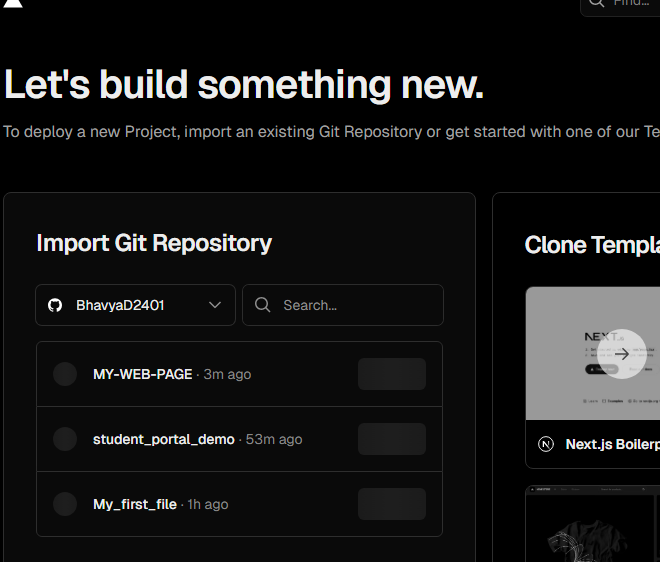


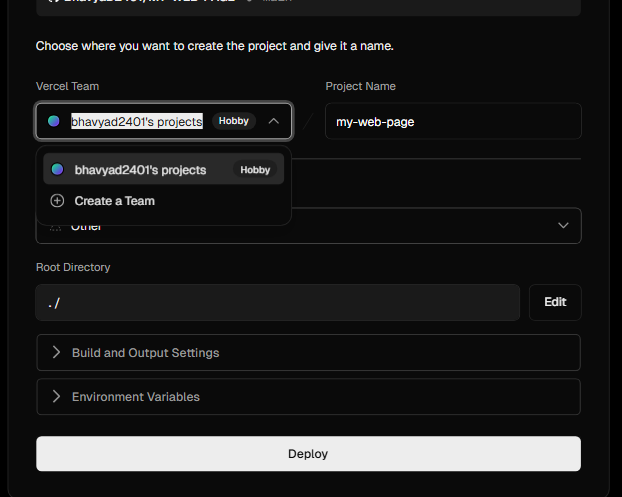


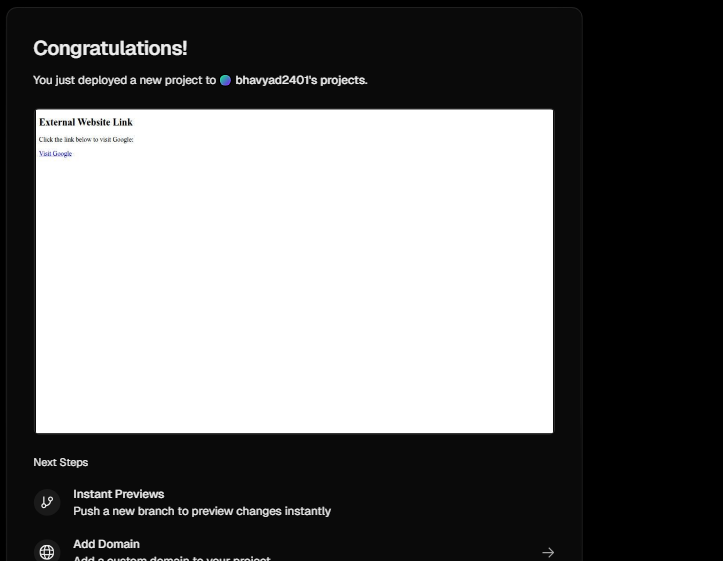






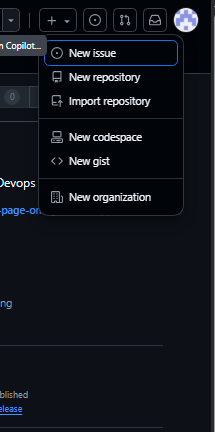


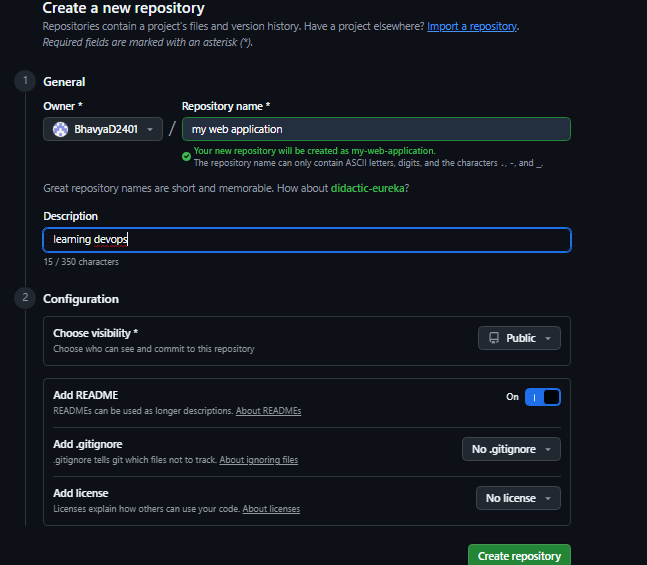




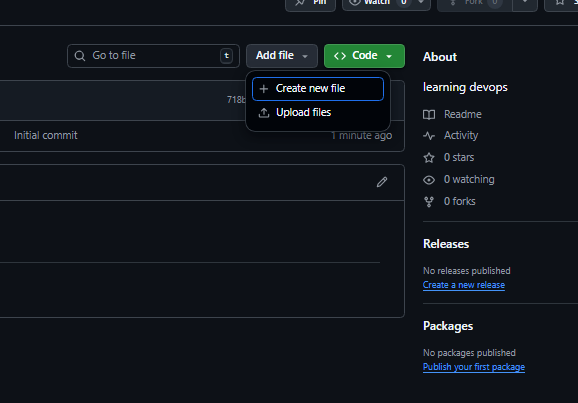
**Experiment 14**:

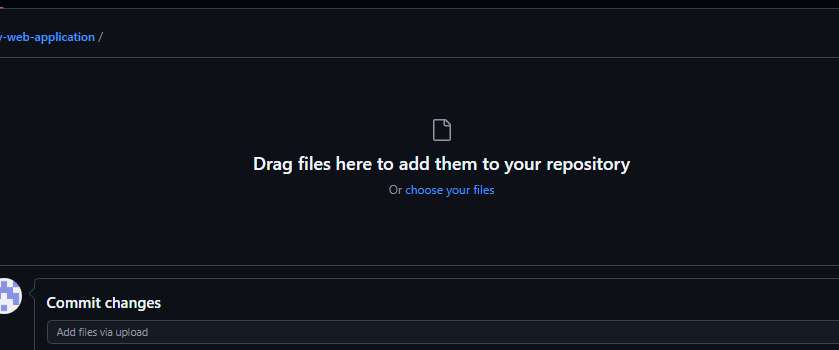
Design and develop a unique single-page web application based on your own creative idea. Once completed, upload the project to GitHub and deploy it using Vercel. The application must include at least one HTML file.

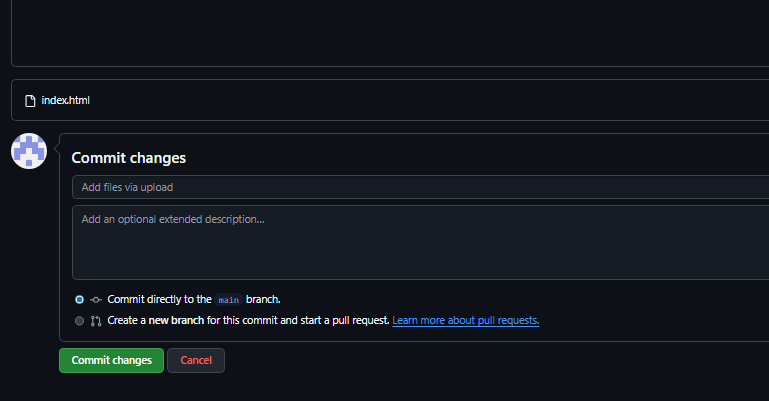


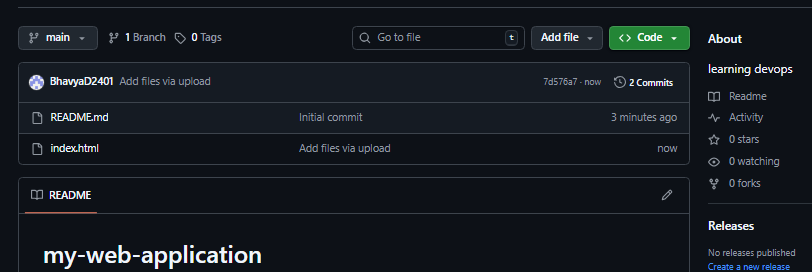


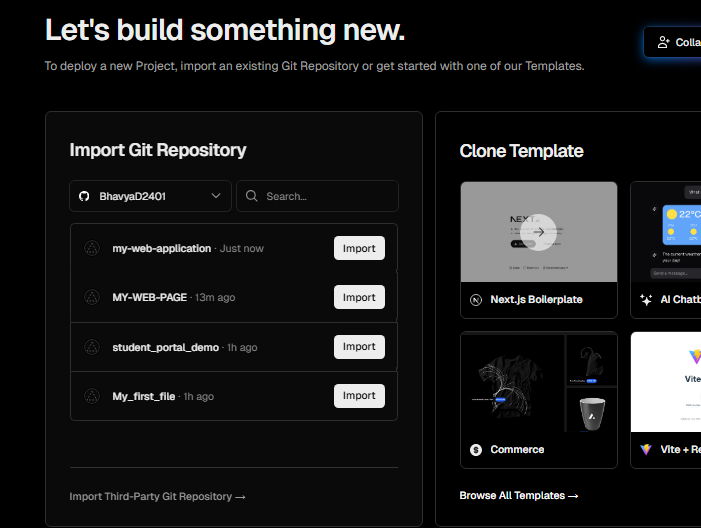


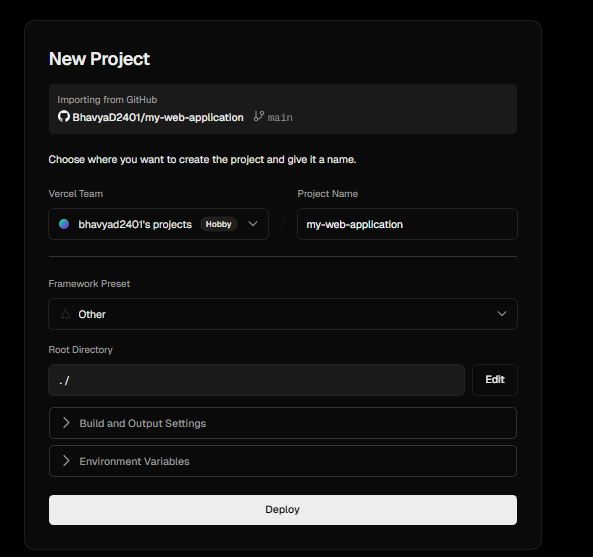


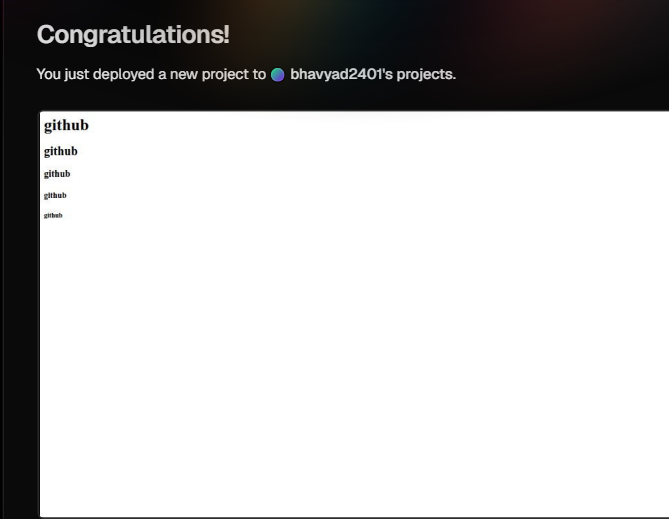






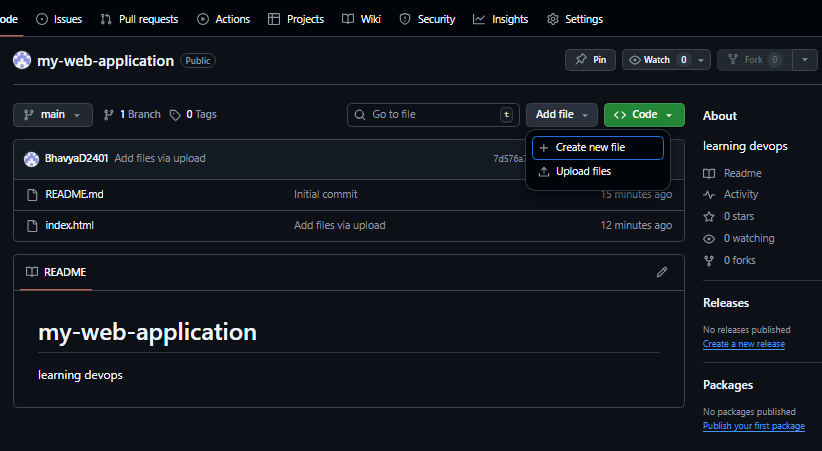


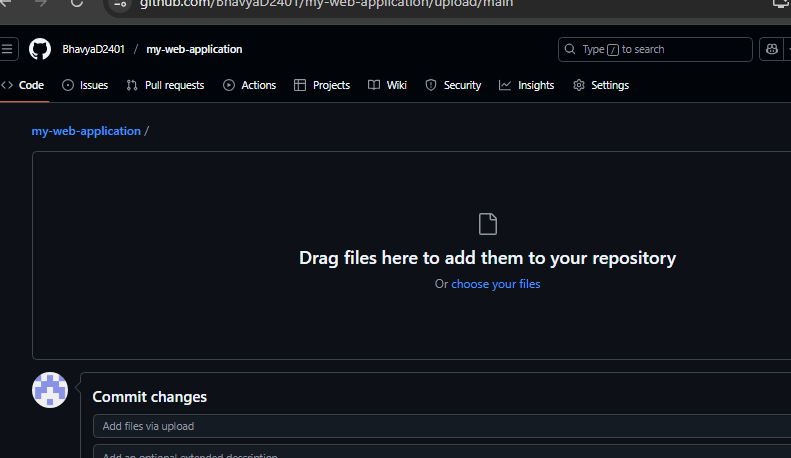


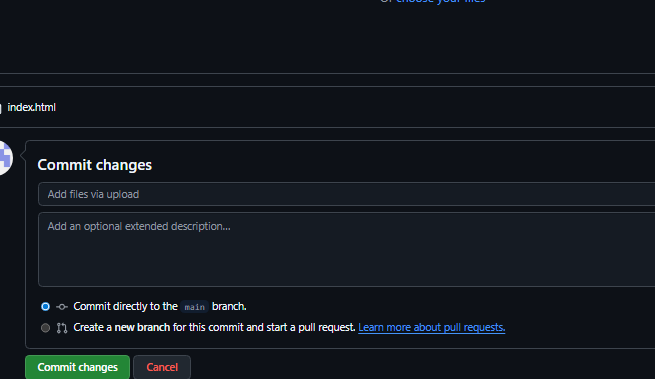


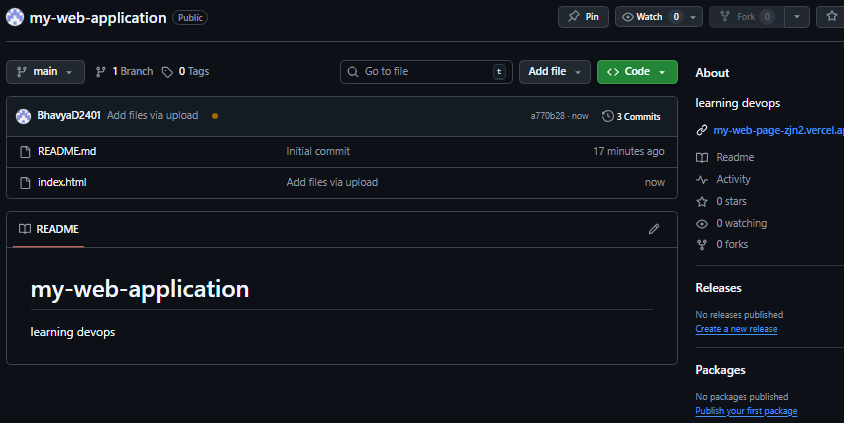
**Experiment 15:**

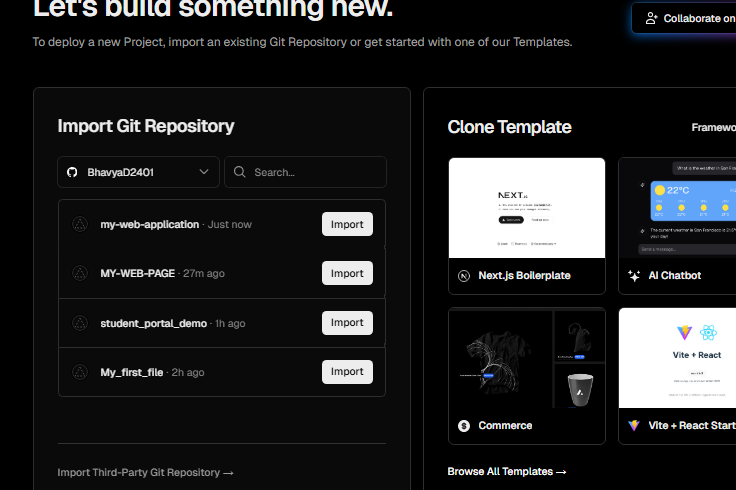
Create an HTML file that visually explains your previous experiment DevOps lifecycle using diagrams, shapes, and labels. Ensure the file clearly displays each stage of the DevOps lifecycle (Plan, Develop, Build, Test, Release, Deploy, Operate, Monitor). Upload the HTML file to a GitHub repository and deploy it using Vercel.

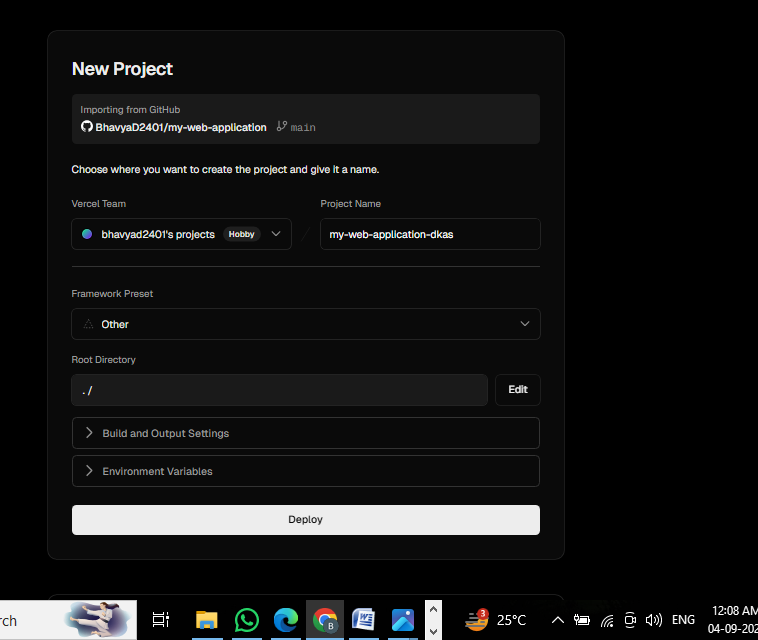


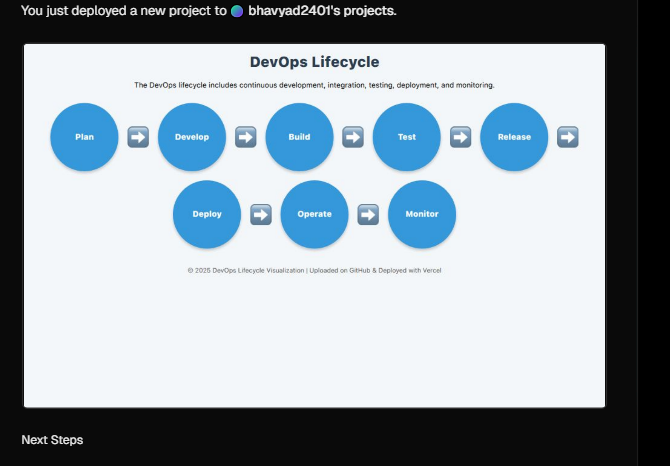












Githud

https://github.com/BhavyaD2401/my-web-application

Versal link:

https://my-web-application-dkas.vercel.app/