PLACEMENT EMPOWERMENT PROGRAM

CLOUD COMPUTING AND DEVOPS CENTRE

TASK 8 : AUTOMATIC STATIC WEBSITE

DEPLOYMENT LOCALLY: CREATE A SCRIPT THAT UPDATES THE SERVER WHENEVER CHANGES ARE PUSHED

NAME: AVINASH S   **DEPT: CSE**

Introduction

In this task, you will create a script that listens for changes in your Git repository and automatically deploys your static website to a local server or environment whenever changes are pushed. This can be achieved by using tools such as **Git hooks** (specifically the postcommit or post-push hooks) combined with simple automation scripts.

By automating this process, you ensure that the server always reflects the latest changes from the repository, saving time and reducing human error in manual deployment.

Key Steps:

- Set up a **Git hook** to listen for changes (such as a push) in your Git repository.
- Create a **deployment script** that automates the process of deploying the static website to your local server.

Configure the **local server** to serve your static website, ensuring it reflects the latest updates.

STEP BY STEP PROCESS:

STEP 1 - SET UP PROJECT DIRECTORY

Create a new folder and name it web. I am using the terminal and git commands to version control my github.

STEP 2: INITIALIZE A GIT REPOSITORY

Initialise a git repo and paste the address of the MyStaticWebsite to show the path. This will link with the git repository.

STEP 3: ADD A REMOTE GIT REPOSITORY

By using this command, the linking is done as such PS C:\Users\Desktop\web> git remote add origin

STEP 4: CREATE AN INDEX.HTML FILE TO WRITE THE HTML CODE FOR THE STATIC WEBSITE

Inside the website folder, create an index.html file.

Open the file and add basic HTML code for the basic static website.

```
nothing to commit, working tree clean
PS C:\Users\UVARAJ.A\web> git branch -M main
PS C:\Users\UVARAJ.A\web> git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 436 bytes | 436.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Mahaamimiii/MyStaticWebsite.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
PS C:\Users\UVARAJ.A\web>
```

STEP 5 - ADD THE FILE AND COMMIT FILE

```
PS C:\Users\UVARAJ.A\web> git add .
PS C:\Users\UVARAJ.A\web> git commit -m "Initial commit"
On branch main
nothing to commit, working tree clean
```

STEP 6 - PUSH THE FILE TO GITHUB REPOSITORY

PS C:\Users\UVARAJ.A\web> git push -u origin main

The above command will push the index.html file into the github repository.

```
nothing to commit, working tree clean
PS C:\Users\UVARAJ.A\web> git branch -M main
PS C:\Users\UVARAJ.A\web> git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 436 bytes | 436.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Mahaamimiii/MyStaticWebsite.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
PS C:\Users\UVARAJ.A\web>
```
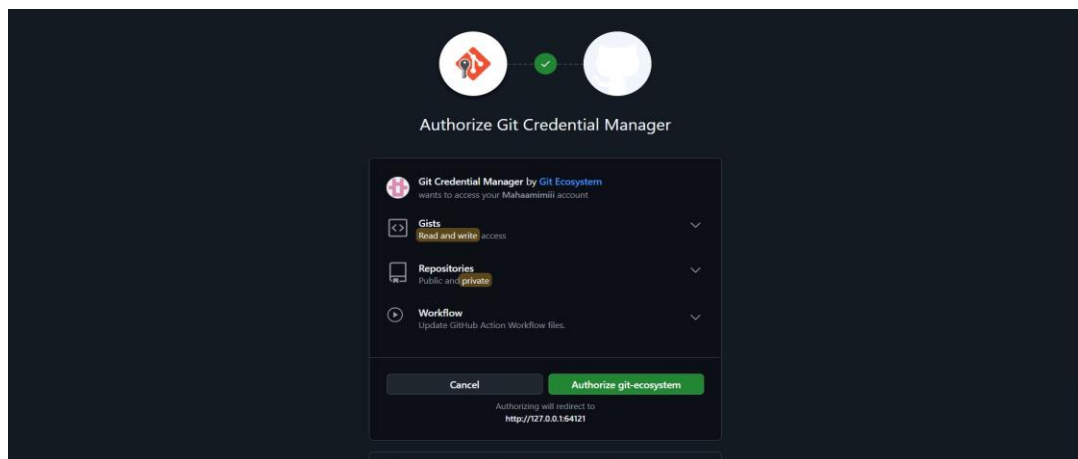
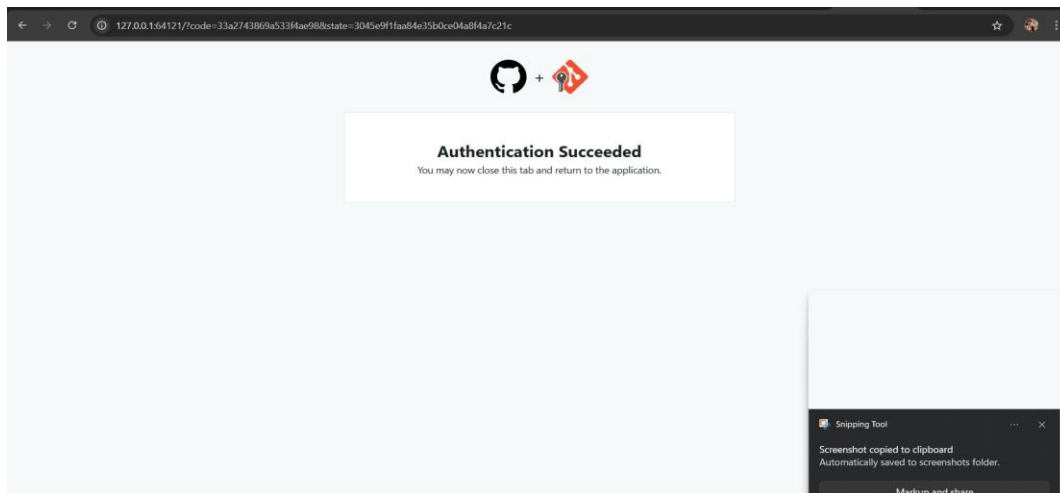## STEP 7 - THE GITHUB LOGIN PAGE OCCURS WITH THE SIGNUP OPTIONS.

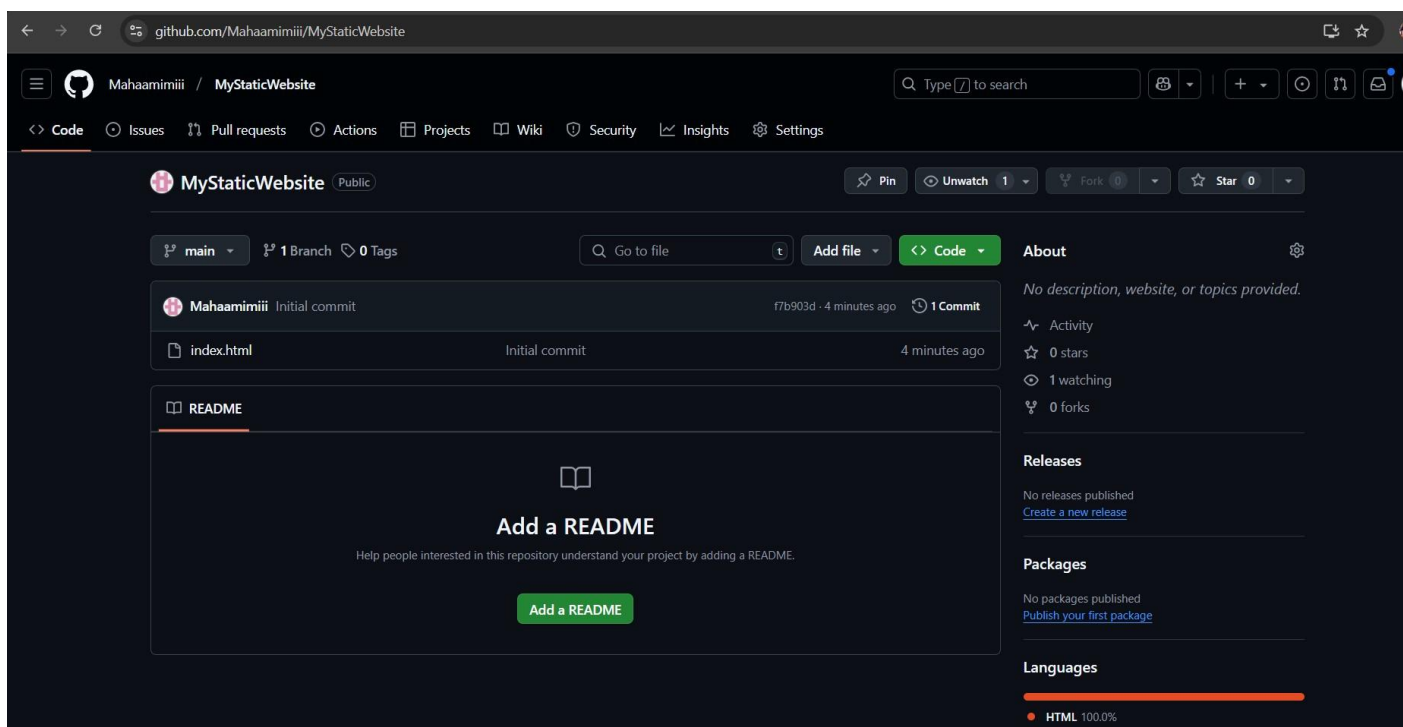Enter your login credentials to access the github repository.
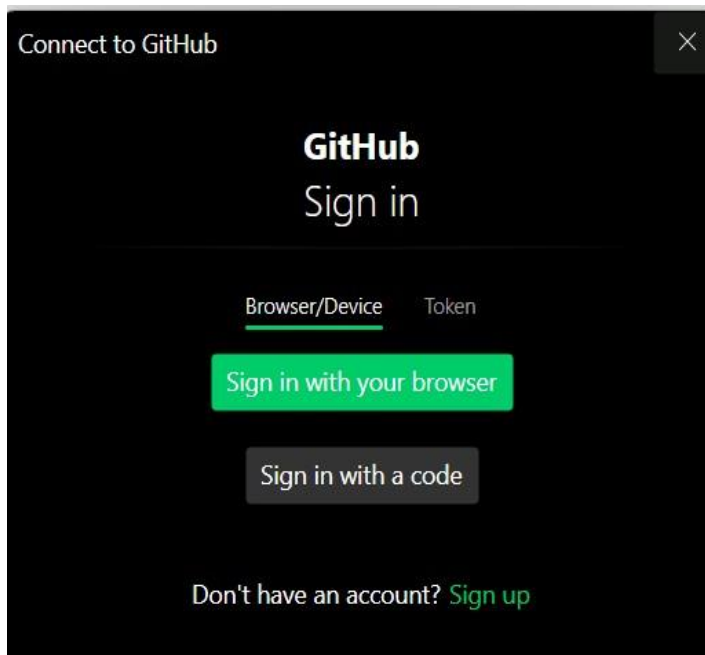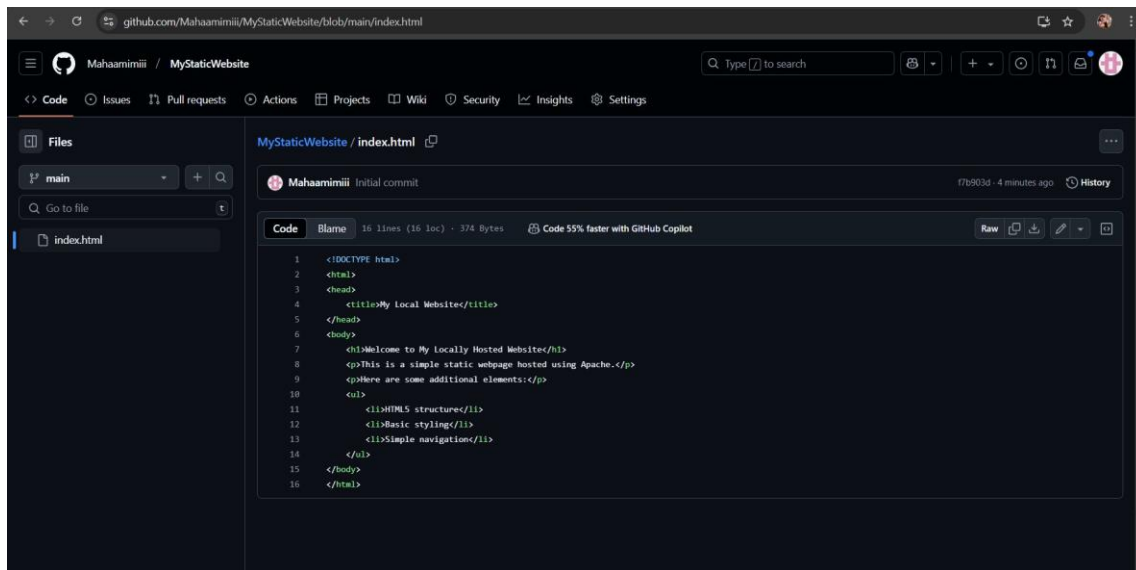

STEP 8 LOGIN

STEP 9 AUTHENTICATION

STEP 10 - VIEW THE PUSHED INDEX.HTML FILE IN THE REPOSITORY .

4. Outcomes

After completing the task, you should be able to:

- **Create and Use Git Hooks**: You will be able to set up Git hooks such as post-commit or post-push to trigger the deployment script automatically.

- **Write a Deployment Script**: You will be able to write a script that deploys your static website to a local server.

- **Automate Updates**: Whenever changes are pushed to the Git repository, the deployment script will automatically update the server, ensuring the website remains up-to-date. **Optimize Development Workflow**: By automating the deployment process, you will save time, avoid errors, and improve the overall development workflow.

Conclusion

By automating the deployment of your static website, you ensure that your local server is always up-to-date with the latest changes from your Git repository. This task helps streamline the development and deployment process, reducing the risk of human error and enhancing productivity. Once you have automated the deployment locally, you can extend this approach to more advanced environments, such as cloud-based servers or continuous deployment .