**Placement Empowerment Program**

***Cloud Computing and DevOps Centre***

Deploy your static website using Github Pages : Host your local Git repository’s static website directly using Github pages

Name: Harivasan S

Department: ADS



# Introduction:

GitHub Pages is a hosting service for static websites that enables you to publish projects directly from a GitHub repository. It provides developers with a simple, free, and effective way to showcase their work, build personal websites, or host documentation

# Overview:

This project showcases the process of deploying a static website with GitHub Pages. Beginning with the creation of a GitHub repository, we’ll walk through the essential steps to host a fully functional static site. This involves initializing a Git repository, uploading files to GitHub, and setting up GitHub Pages for deployment.

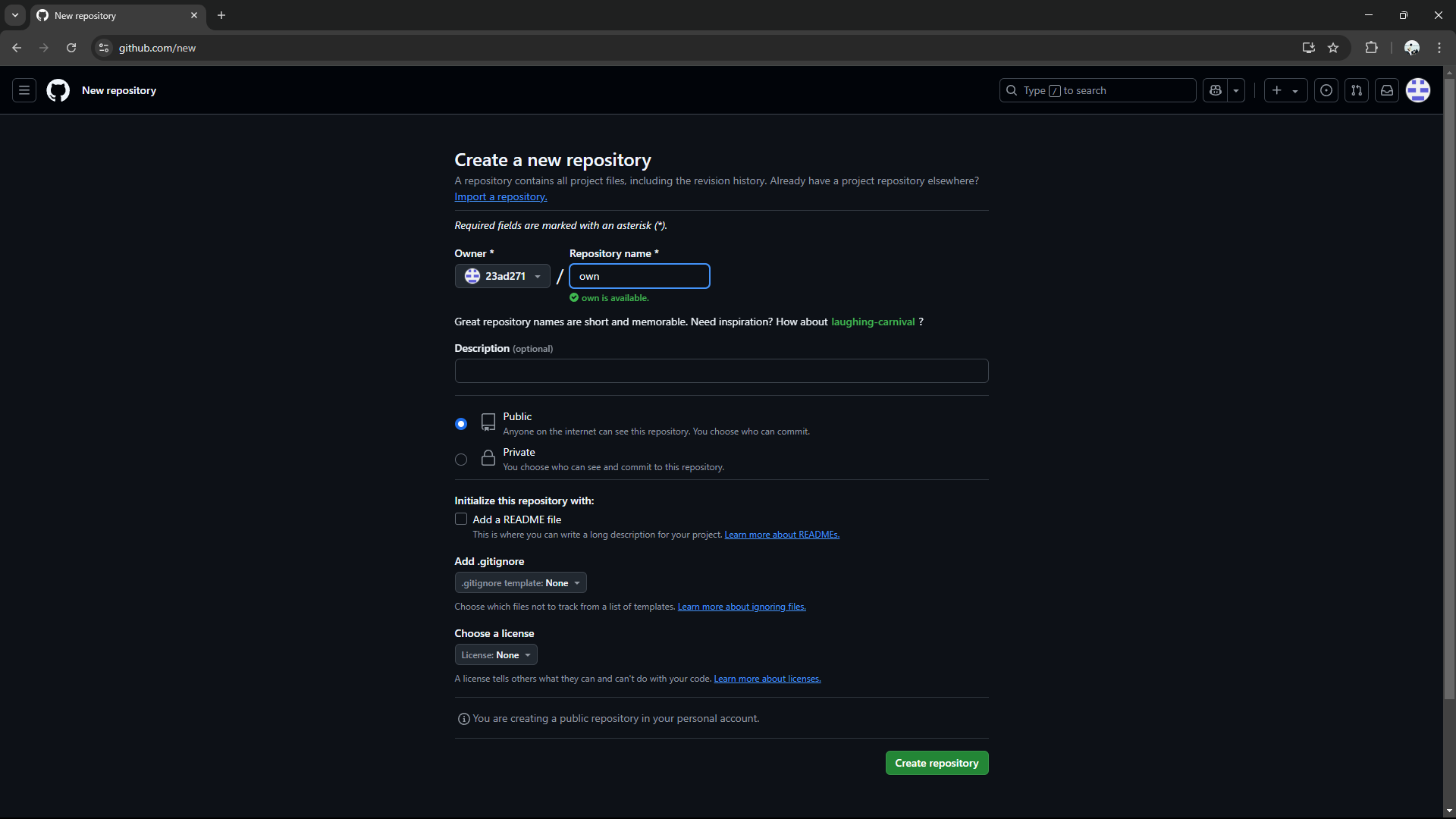
# Objectives:

1. Explore the basics of GitHub Pages and how to deploy a website.
2. Understand the significance of static website hosting and its applications.
3. Develop practical skills in using Git and GitHub for version control and hosting.
4. Successfully launch a static website and make it available to the public.

# Step-by-Step Overview

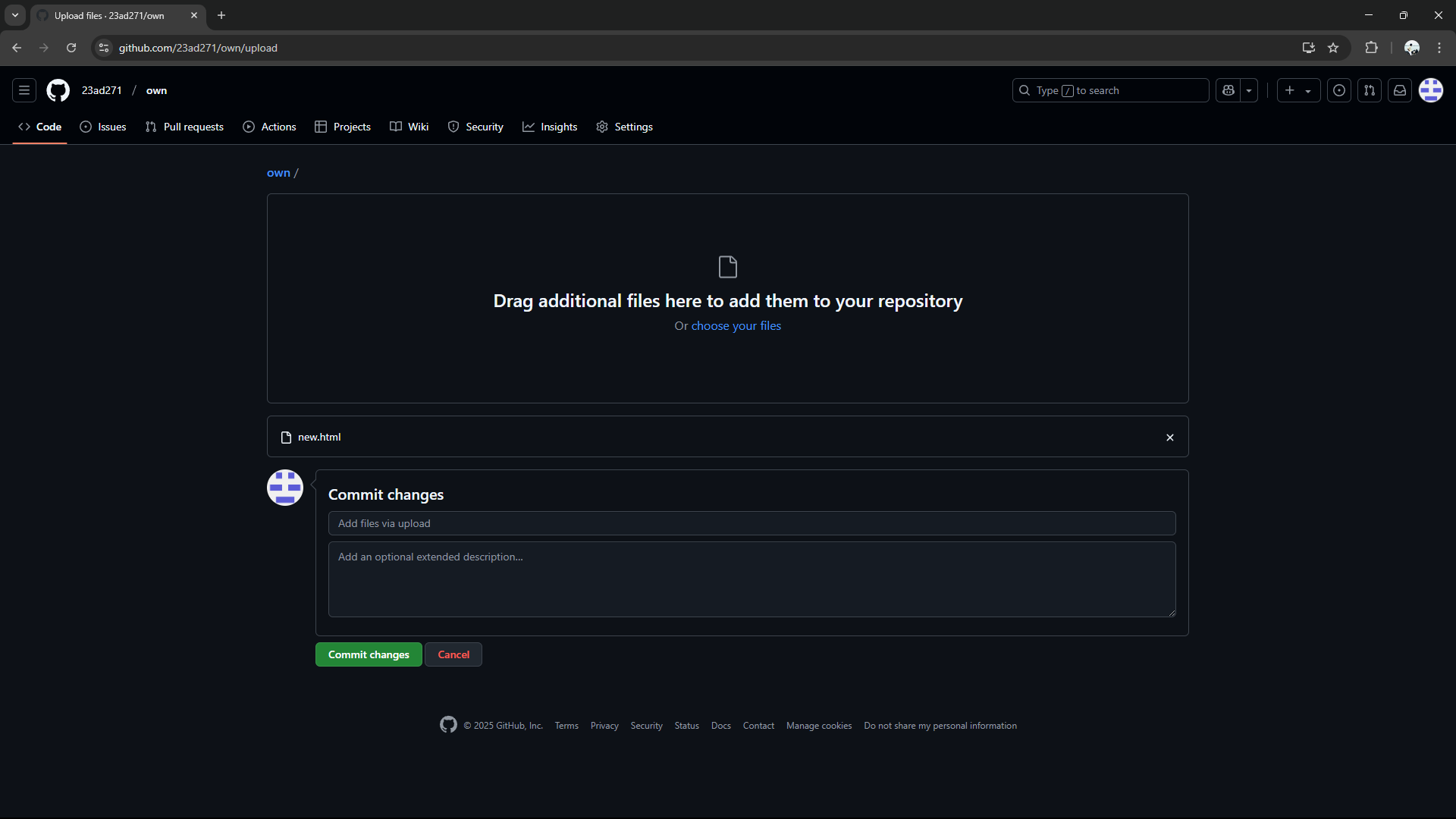
Step 1:

**Create a New Repository**:



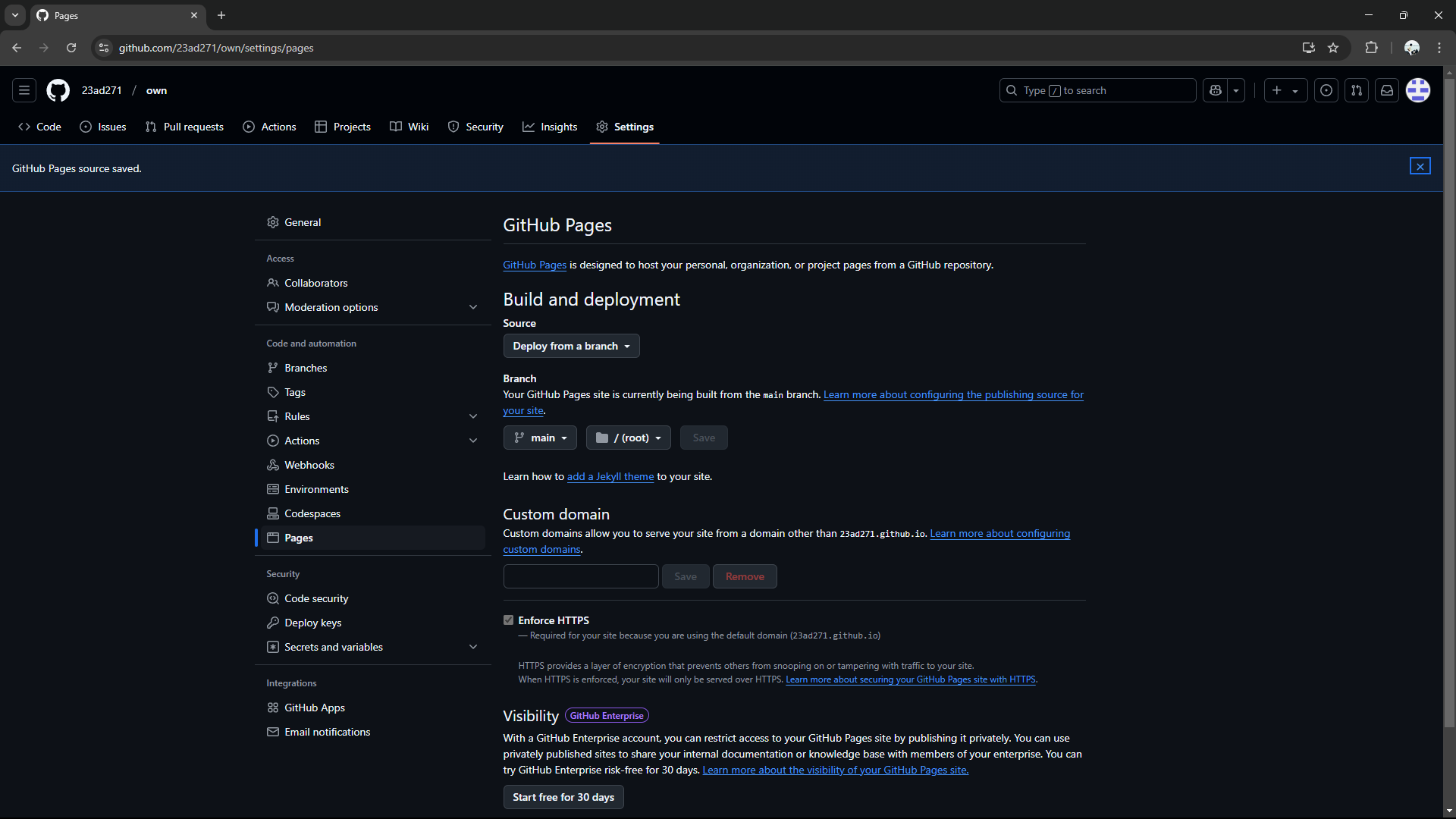
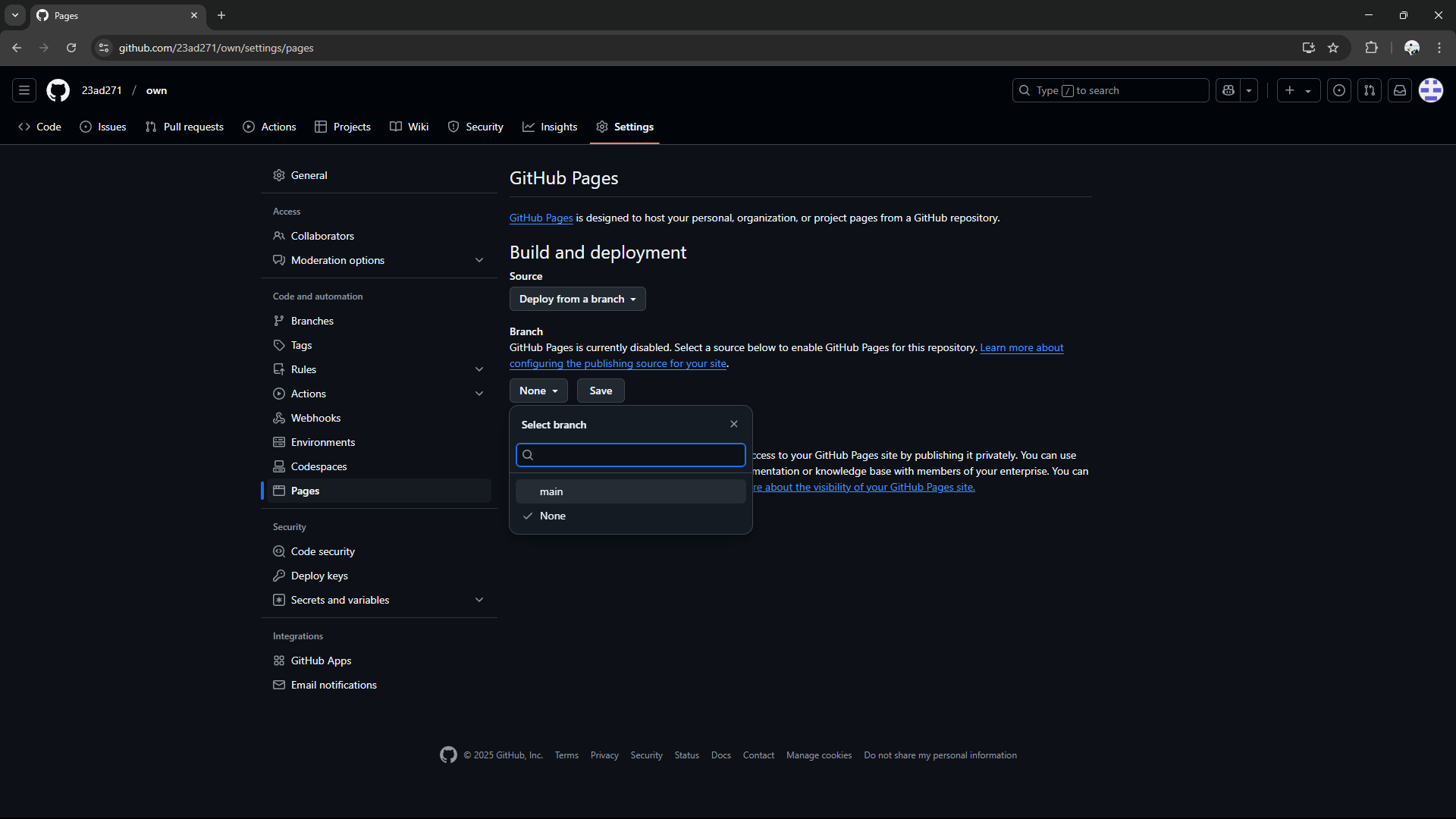
Step 2:

Create a folder where you’ll keep all your file inside it and save as index.html



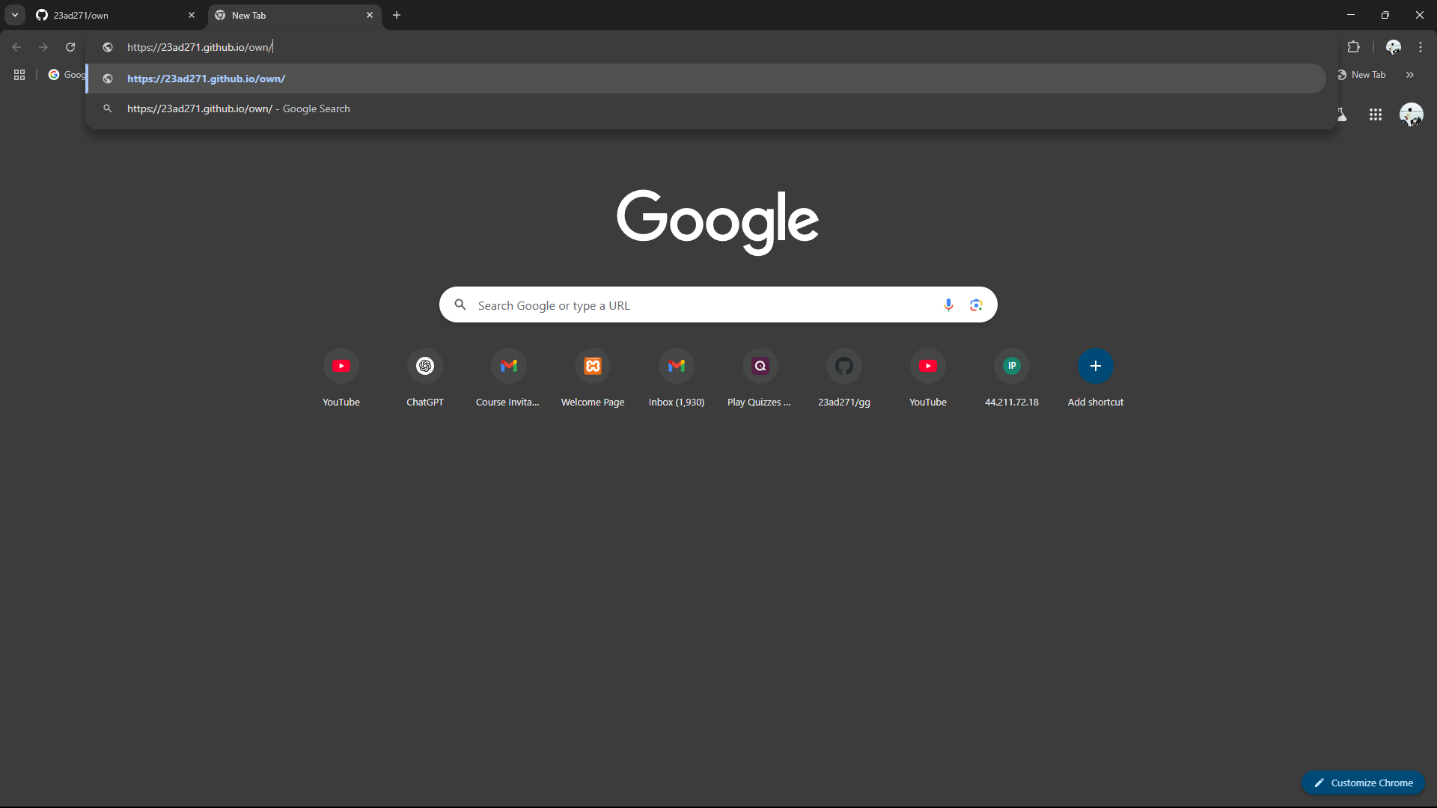
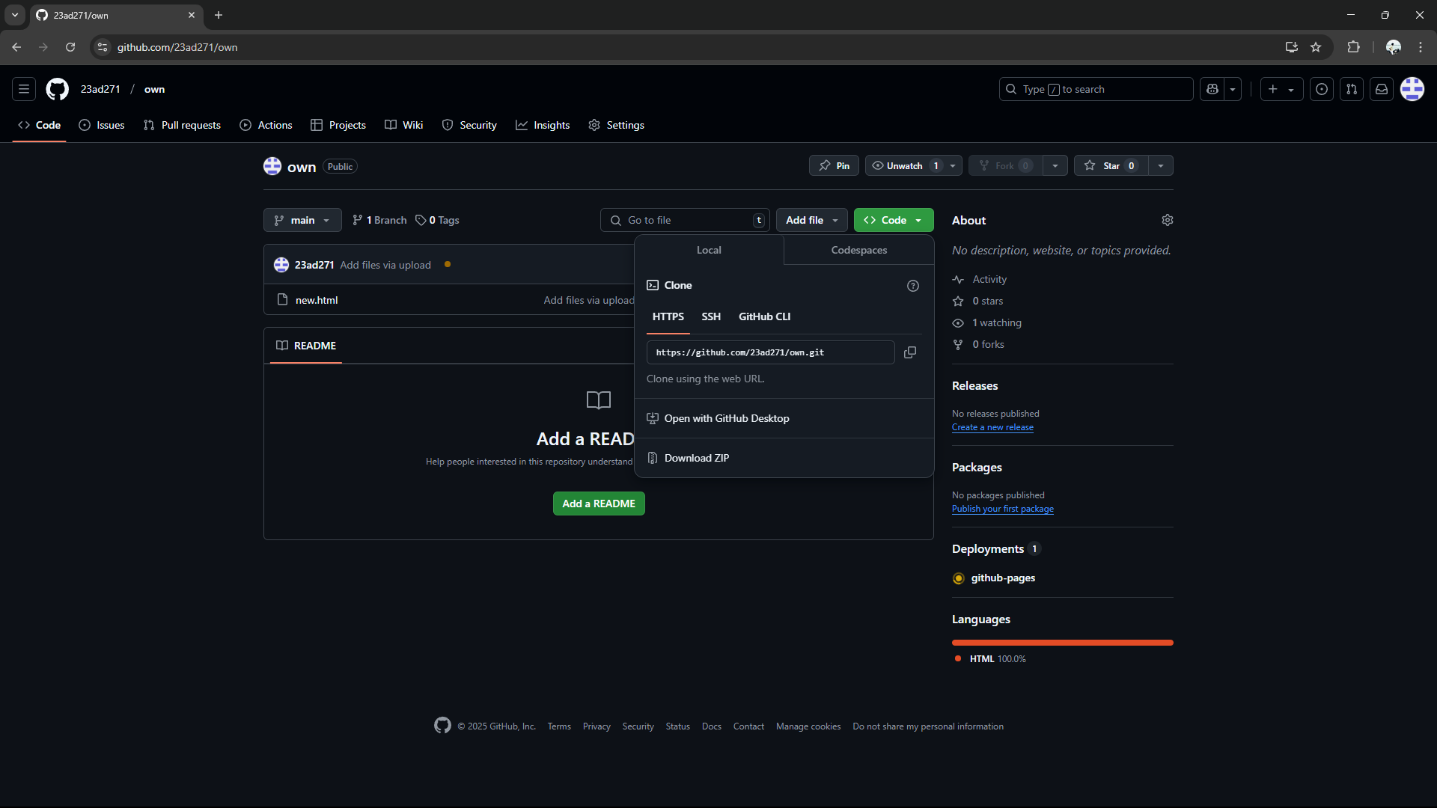
Step 3:

Open Settings in github and enter into an pages option after entering save that file in an main



Step 4:

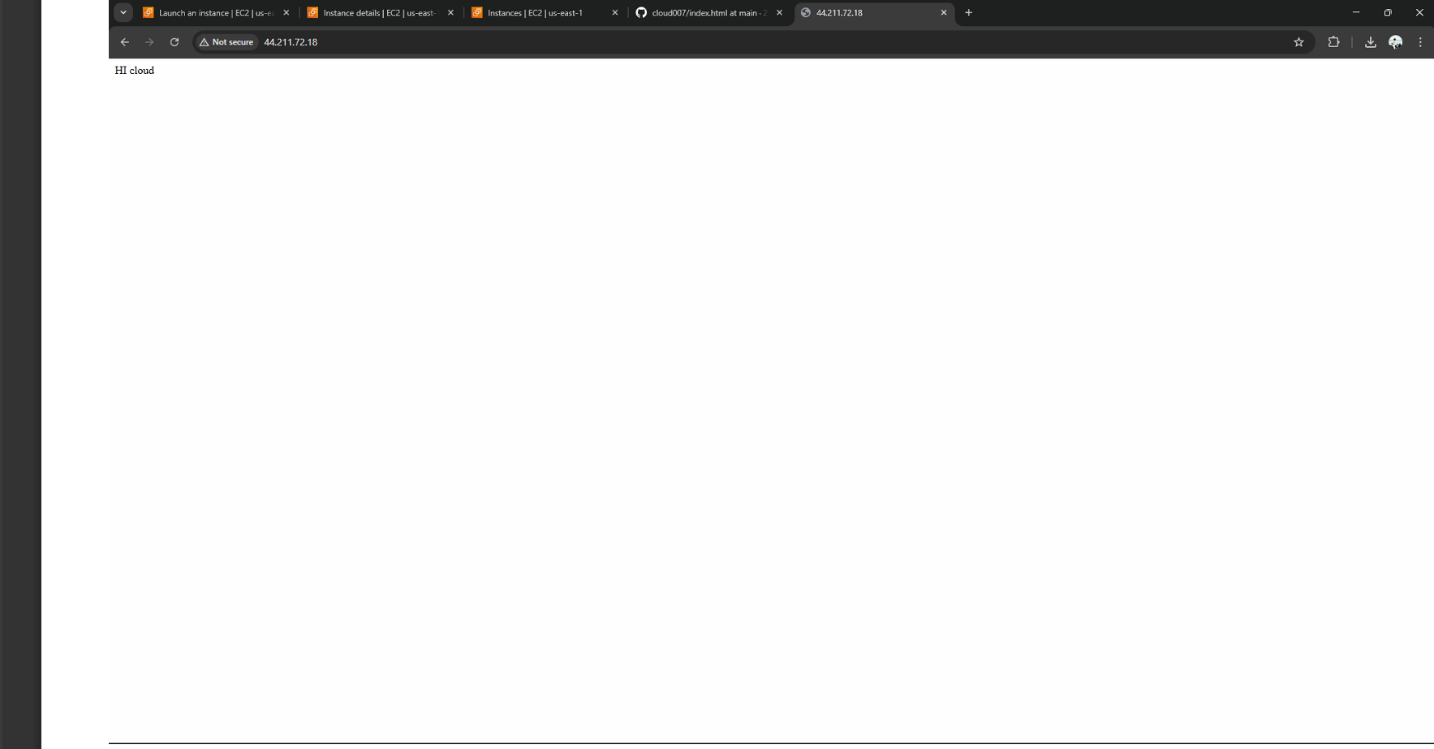
After saving it deploy one code copy that code and paste in an browser it will show tht image what we saved in that file



Step 5:

Finally you visit your website

:



Expected Outcome :

1. Set up and configure a GitHub repository for your project.
2. Initialize a local Git repository and connect it to GitHub.
3. Push your static website files (HTML, CSS, JavaScript) to the GitHub repository.
4. Activate GitHub Pages in the repository settings to publish your site.
5. Access your live website via the GitHub Pages URL.