

## MPL Practical 10

**Aim:** To study and implement deployment of Ecommerce PWA to GitHub Pages.

**GitHub Link:** <https://harshpadyal.github.io/PWA-CAMPUS-CRAZE/>

### Theory

GitHub Pages is a free hosting service provided by GitHub to publish static websites directly from a GitHub repository. It supports HTML, CSS, JavaScript, and other front-end files, making it ideal for hosting PWAs.

### What we implemented in our code

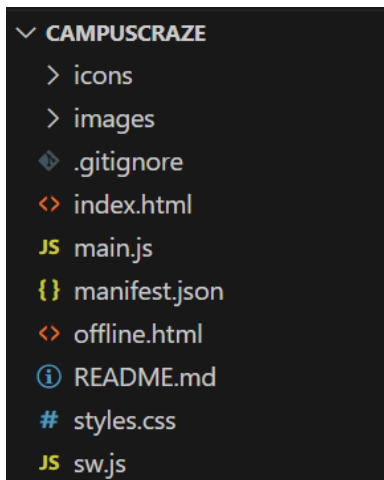
In this experiment, we deployed our E-commerce PWA project, "CampusCraze - Budget-friendly stationery, hostel decor, and fashion for college students

", to GitHub Pages using these steps:

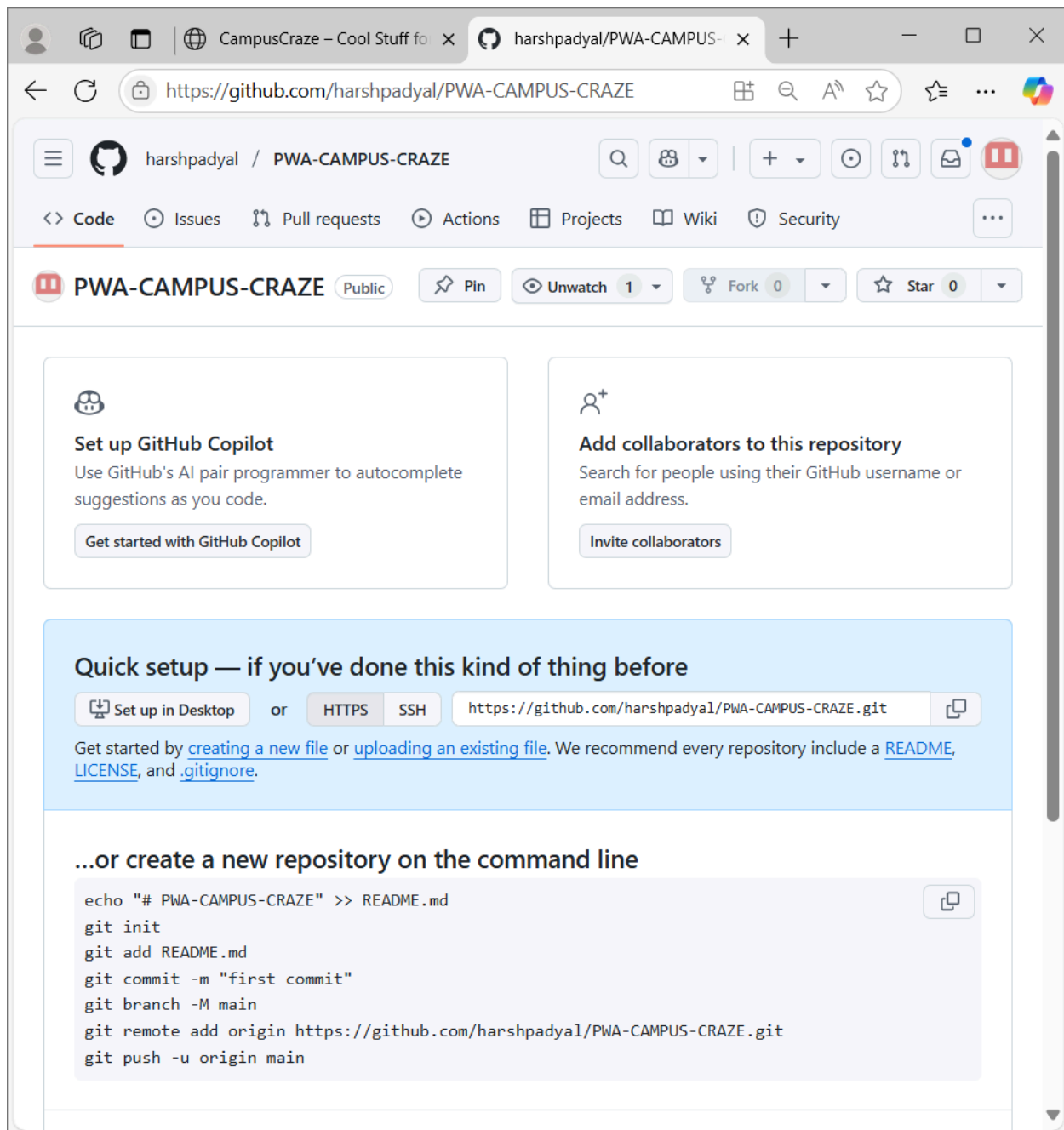
- We ensured all necessary PWA files were in place, such as index.html, manifest.json, sw.js, and offline assets.
- The index.html file used relative paths correctly so resources load properly after deployment.
- We registered the service worker to enable offline access and caching of essential files.
- The manifest.json provided metadata to support "Add to Home Screen" functionality.
- We created a GitHub repository and pushed all project files to it.
- Then, from GitHub Settings → Pages, we selected the main branch and the / (root) folder as the source.
- After saving, GitHub generated a live link through which we accessed the deployed PWA.

This deployment made our PWA fully functional and accessible online with service worker support, offline capabilities, and installability features.

### Folder Structure



## Create a GitHub Repository



Link Your Local Project to GitHub and Push Changes.

```
PS D:\Users\harsh\Documents\CampusCraze> git init
Initialized empty Git repository in D:/Users/harsh/Documents/CampusCraze/.git/
PS D:\Users\harsh\Documents\CampusCraze> git add .
PS D:\Users\harsh\Documents\CampusCraze> git commit -m "PWA"
[main (root-commit) 432a17c] PWA
27 files changed, 1774 insertions(+)
```

```
PS D:\Users\harsh\Documents\CampusCraze> git remote add origin https://github.com/harshpadyal/
PS D:\Users\harsh\Documents\CampusCraze> git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 31, done.
Counting objects: 100% (31/31), done.
Delta compression using up to 4 threads
Compressing objects: 100% (31/31), done.
Writing objects: 100% (31/31), 879.62 KiB | 10.35 MiB/s, done.
Total 31 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/harshpadyal/PWA-CAMPUS-CRAZE.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
PS D:\Users\harsh\Documents\CampusCraze>
```

The screenshot shows a web browser window with the URL <https://github.com/harshpadyal/PWA-CAMPUS-CRAZE>. The repository is public and has 0 stars, 0 forks, and 1 watcher. The repository contains a file tree with the following files and folders:

File/Folder	Type	Time
icons	PWA	2 minutes ago
images	PWA	2 minutes ago
.gitignore	PWA	2 minutes ago
README.md	PWA	2 minutes ago
index.html	PWA	2 minutes ago
main.js	PWA	2 minutes ago
manifest.json	PWA	2 minutes ago
offline.html	PWA	2 minutes ago
styles.css	PWA	2 minutes ago
sw.js	PWA	2 minutes ago

The repository also has a README file. The right sidebar shows the repository's statistics and links to the README, Activity, and Releases. The Languages section shows the following distribution:

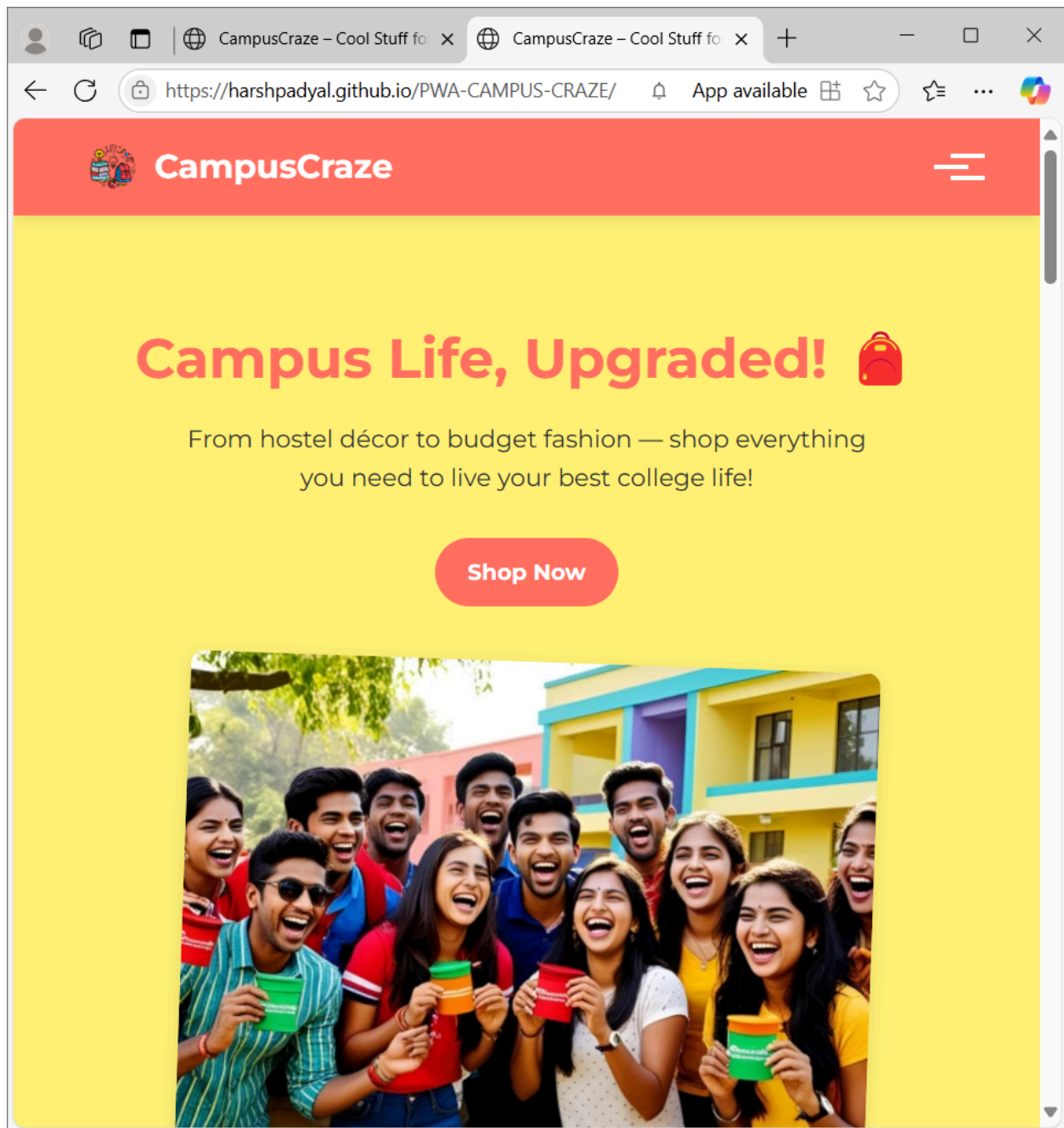
Language	Percentage
HTML	37.1%
CSS	36.3%
JavaScript	26.6%

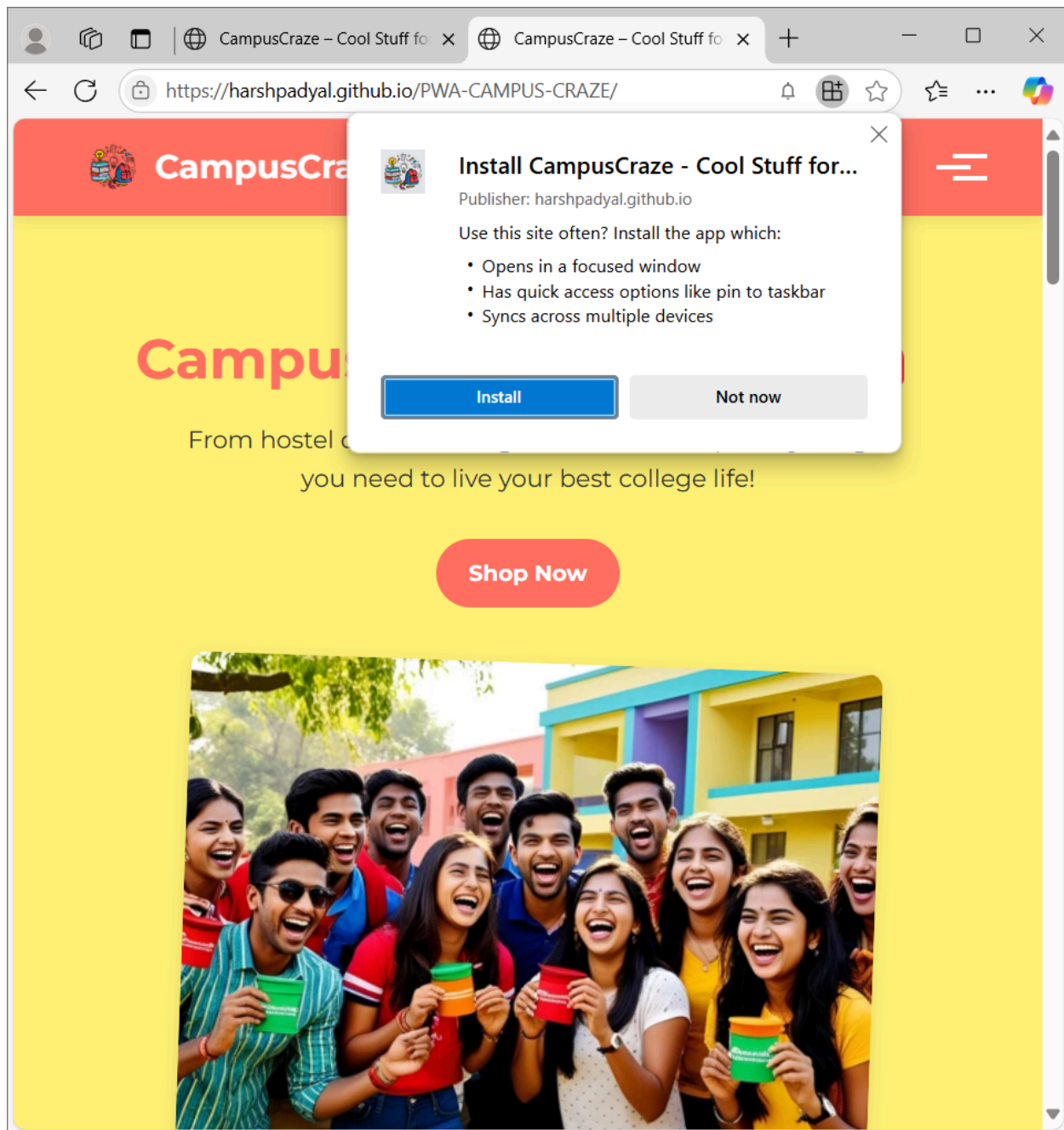
The screenshot shows the GitHub Pages settings page for the repository 'PWA-CAMPUS-CRAZE' by user 'harshpadyal'. The browser address bar shows the URL 'https://github.com/harshpadyal/PWA-CAMPUS-CRAZE/setting...'. A blue notification banner at the top states 'GitHub Pages source saved.'.

The left sidebar contains a list of settings categories: General, Access, Collaborators, Moderation options, Code and automation, Branches, Tags, Rules, Actions, Webhooks, Environments, Codespaces, **Pages** (selected), Security, Code Security, Deploy keys, and Secrets and variables.

The main content area is titled 'GitHub Pages' and includes the following sections:

- General:** A description stating 'GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.'
- Build and deployment:**
  - Source:** A dropdown menu set to 'Deploy from a branch'.
  - Branch:** A message stating 'Your GitHub Pages site is currently being built from the main branch.' with a link to 'Learn more about configuring the publishing source for your site.' Below this, there are dropdowns for 'main' and '/ (root)', and a 'Save' button.
  - A link to 'Learn how to add a Jekyll theme to your site.'
- Custom domain:**
  - A message: 'Custom domains allow you to serve your site from a domain other than harshpadyal.github.io. Learn more about configuring custom domains.'
  - An empty text input field for a custom domain, with 'Save' and 'Remove' buttons.
  - Enforce HTTPS:** A checked checkbox with the text 'Required for your site because you are using the default domain (harshpadyal.github.io)'.





## Conclusion

In this experiment, we successfully deployed the CampusCraze PWA to GitHub Pages by ensuring proper setup of service workers and manifest files. Initially, we faced 404 errors for a CSS-based image and offline caching issues, which we resolved by correcting the image path and ensuring only valid resources were cached.