

Name: Vedang Khandagale  
Division:D15A  
Roll No:29  
Batch: B

## **Experiment No 10**

### **Aim:**

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

### **Theory:**

#### **GitHub Pages**

Public web pages are freely hosted and easily published. Public webpages hosted directly from your GitHub repository. Just edit, push, and your changes are live.

GitHub Pages provides the following key features:

1. Blogging with Jekyll
2. Custom URL
3. Automatic Page Generator

Reasons for favoring this over Firebase:

1. Free to use
2. Right out of github
3. Quick to set up

GitHub Pages is used by Lyft, CircleCI, and HubSpot.

GitHub Pages is listed in 775 company stacks and 4401 developer stacks.

#### **Pros**

1. Very familiar interface if you are already using GitHub for your projects.
2. Easy to set up. Just push your static website to the gh-pages branch and your website is ready.
3. Supports Jekyll out of the box.
4. Supports custom domains. Just add a file called CNAME to the root of your site, add an A record in the site's DNS configuration, and you are done.

#### **Cons**

1. The code of your website will be public, unless you pay for a private repository.

2. Currently, there is no support for HTTPS for custom domains. It's probably coming soon though.
3. Although Jekyll is supported, plug-in support is rather spotty.

## Firebase

The Realtime App Platform. Firebase is a cloud service designed to power real-time, collaborative applications. Simply add the Firebase library to your application to gain access to a shared data structure; any changes you make to that data are automatically synchronized with the Firebase cloud and with other clients within milliseconds.

Some of the features offered by Firebase are:

1. Add the Firebase library to your app and get access to a shared data structure. Any changes made to that data are automatically synchronized with the Firebase cloud and with other clients within milliseconds.
2. Firebase apps can be written entirely with client-side code, update in real-time out-of-the-box, interoperate well with existing services, scale automatically, and provide strong data security.
3. Data Accessibility- Data is stored as JSON in Firebase. Every piece of data has its own URL which can be used in Firebase's client libraries and as a REST endpoint. These URLs can also be entered into a browser to view the data and watch it update in real-time.

Reasons for favoring over GitHub Pages:

1. Realtime backend made easy
2. Fast and responsive

Instacart, 9GAG, and Twitch are some of the popular companies that use Firebase. Firebase has a broader approval, being mentioned in 1215 company stacks & 4651 developer stacks

Pros

1. Hosted by Google. Enough said.
2. Authentication, Cloud Messaging, and a whole lot of other handy services will be available to you.
3. A real-time database will be available to you, which can store 1 GB of data.
4. You'll also have access to a blob store, which can store another 1 GB of data.
5. Support for HTTPS. A free certificate will be provisioned for your custom domain within 24 hours.

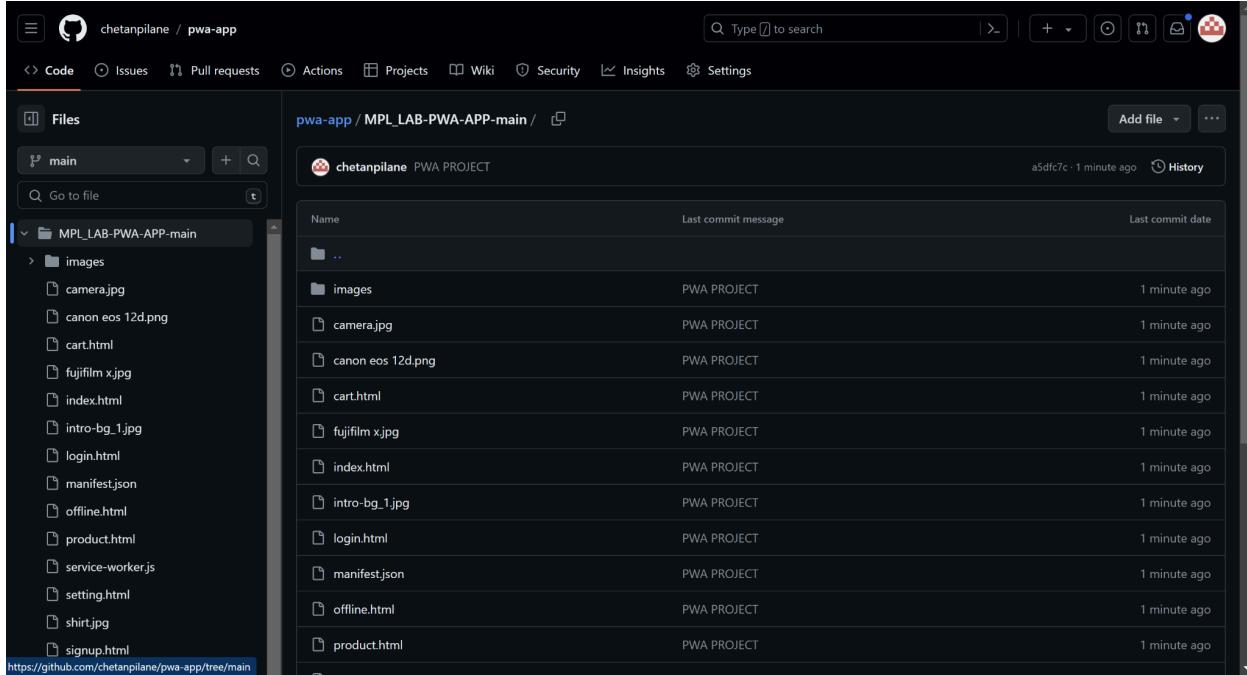
Cons

1. Only 10 GB of data transfer is allowed per month. But this is not really a big problem, if you use a CDN or AMP.
2. Command-line interface only.
3. No in-built support for any static site generator.

Link for GitHub: [https://github.com/VedangKhandagale/MPL\\_LAB-PWA-APP](https://github.com/VedangKhandagale/MPL_LAB-PWA-APP)

GitHub ScreenShots

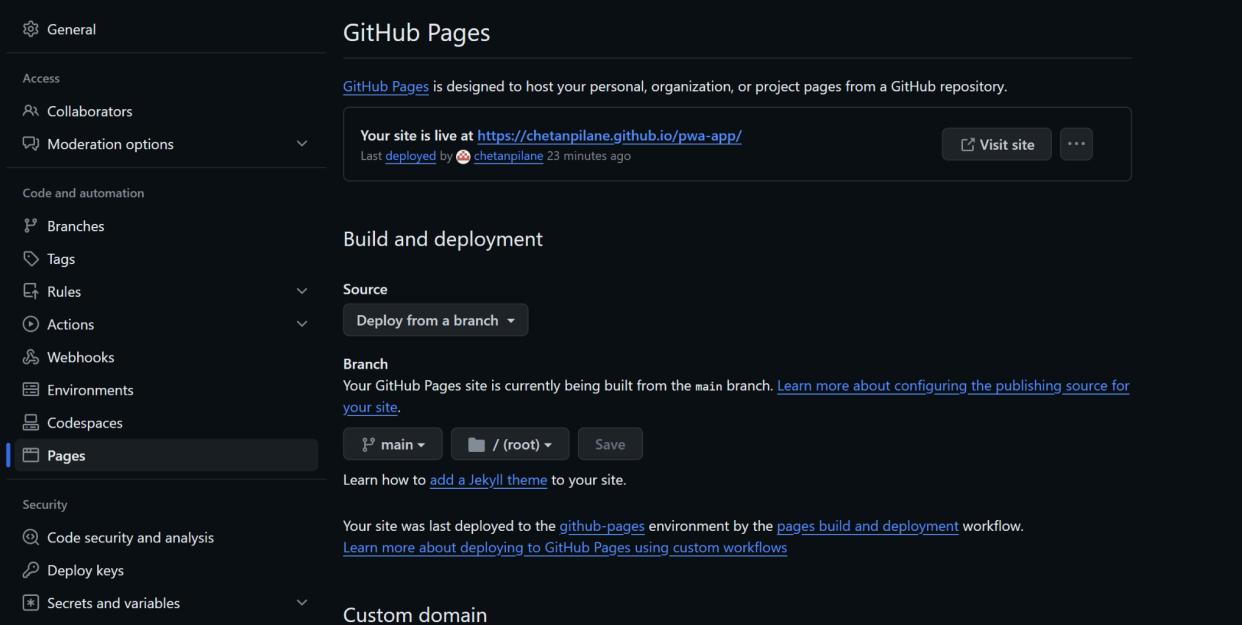
Step1: Make your GitHub Repository public and push your PWA into the repository



The screenshot shows a GitHub repository named 'pwa-app' under the user 'chetanpilane'. The repository has a single branch 'main'. On the left, the file structure is shown with a folder 'MPL\_LAB-PWA-APP-main' containing several files like 'index.html', 'product.html', and 'manifest.json'. On the right, a table lists the commit history for the 'main' branch. All commits were made one minute ago by the user 'chetanpilane' and are labeled 'PWA PROJECT'. The commits correspond to the files in the repository.

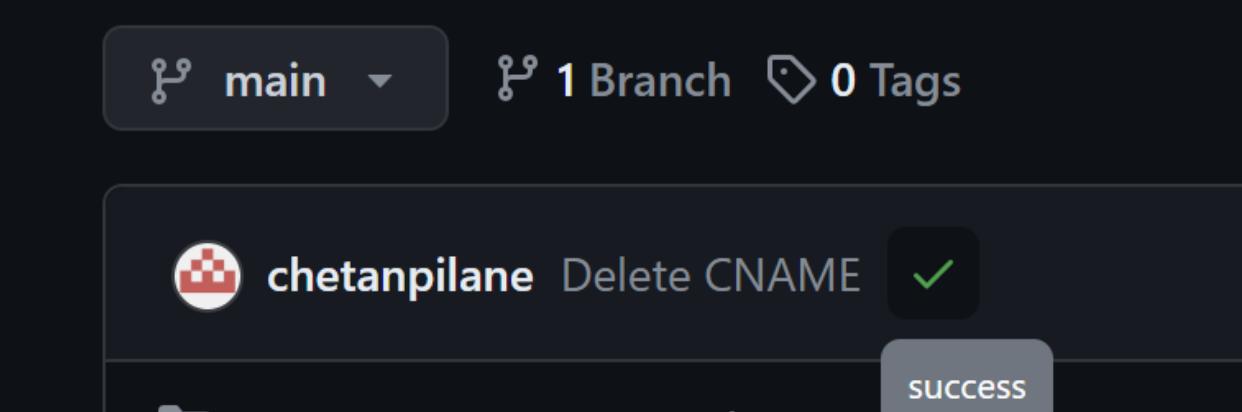
Name	Last commit message	Last commit date
...	PWA PROJECT	1 minute ago
images	PWA PROJECT	1 minute ago
camera.jpg	PWA PROJECT	1 minute ago
canon eos 12d.png	PWA PROJECT	1 minute ago
cart.html	PWA PROJECT	1 minute ago
fujifilm x.jpg	PWA PROJECT	1 minute ago
index.html	PWA PROJECT	1 minute ago
intro-bg_1.jpg	PWA PROJECT	1 minute ago
login.html	PWA PROJECT	1 minute ago
manifest.json	PWA PROJECT	1 minute ago
offline.html	PWA PROJECT	1 minute ago
product.html	PWA PROJECT	1 minute ago
service-worker.js	PWA PROJECT	1 minute ago
setting.html	PWA PROJECT	1 minute ago
shirt.jpg	PWA PROJECT	1 minute ago
signup.html	PWA PROJECT	1 minute ago

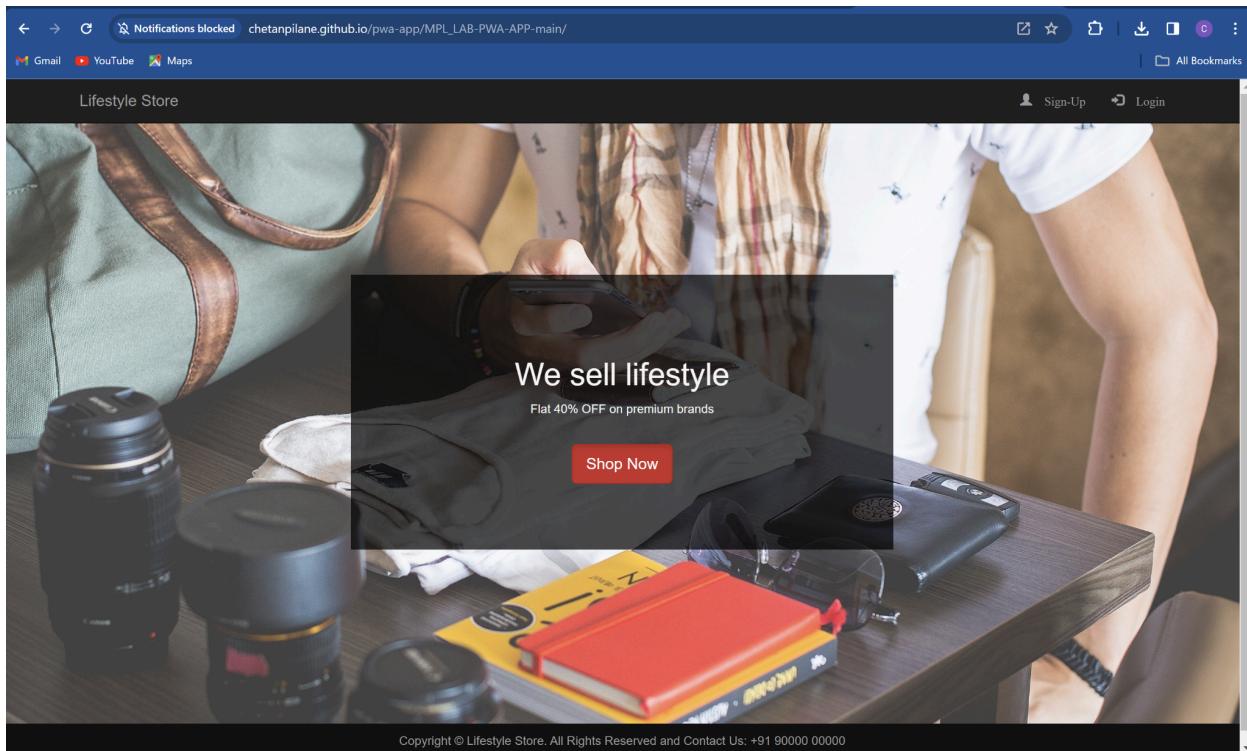
Step 2: Go to settings -> pages and choose your root directory and save it

A screenshot of the GitHub Pages settings page for a repository. The left sidebar shows sections like General, Access, Collaborators, Moderation options, Code and automation, Security, and a selected 'Pages' tab. The main area displays the title 'GitHub Pages' and a message stating 'Your site is live at https://chetanpilane.github.io/pwa-app/'. It includes a 'Visit site' button and a '...' button. Below this, the 'Build and deployment' section shows 'Source' set to 'Deploy from a branch' with 'main' selected. A 'Branch' section indicates the site is built from the main branch. There's also a note about adding a Jekyll theme and information about the deployment environment.

Step 3: Now go to your Code and you will see a small circle near your recent commit( Mine is finished deploying so i am getting a tick-mark sign)

On clicking Logs of all the deployment is shown for convenience

A screenshot of the GitHub deployment logs for the 'main' branch. It shows a summary: 1 Branch and 0 Tags. Below this, a specific log entry for 'chetanpilane' shows a successful deployment of a CNAME record, indicated by a green checkmark icon and the word 'success'.



Conclusion: Hence we deployed our E-Commerce Progressive Web App Successfully on GitHub Pages