Experiment No: - 10

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

Theory:-

GitHub is a web-based platform for version control using Git, primarily used for source code management and collaboration in software development projects. It allows developers to host their code repositories, manage changes, and coordinate work with collaborators.

GitHub Pages is a feature of GitHub that allows users to host static websites directly from their GitHub repositories. It's often used for personal or project websites, documentation, and blogs.
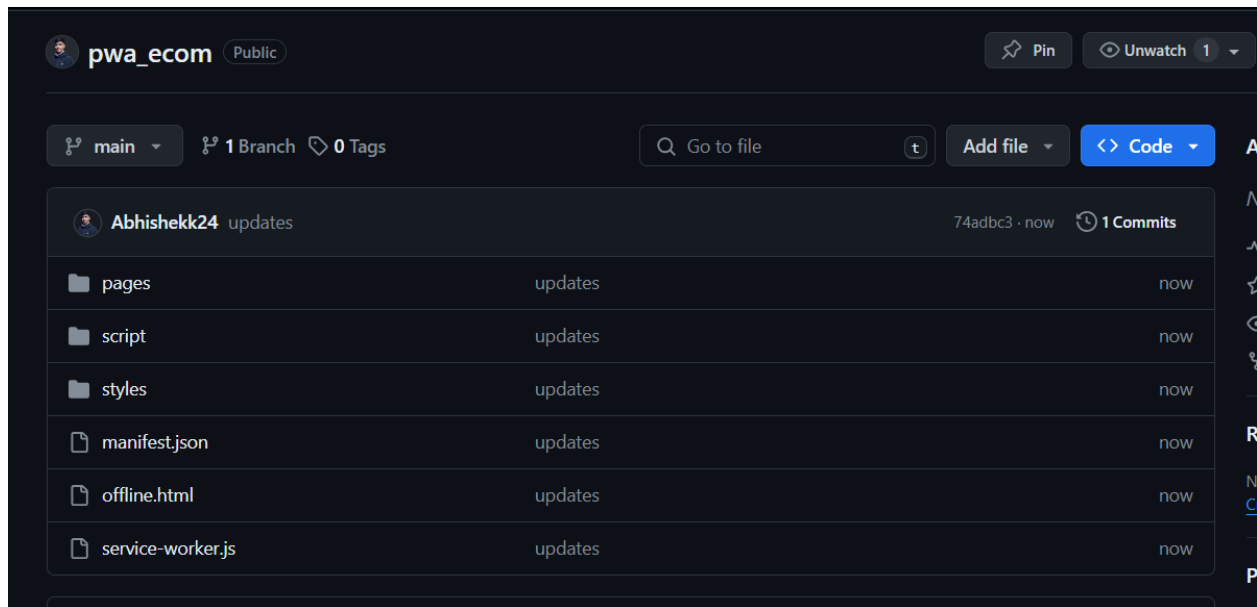
To host a website using GitHub Pages:

1. Create a Repository: Start by creating a new repository on GitHub. Name it in the format `<username>.github.io` if it's for a personal site (replace `<username>` with your GitHub username) or `<repositoryname>` if it's for a project site.

2. Add Content: Add your website's files (HTML, CSS, JavaScript, etc.) to the repository. Ensure that your main HTML file is named `index.html` for the site to be served correctly.

3. Commit Changes: Commit your changes to the repository using Git commands or GitHub's web interface.

4. Enable GitHub Pages: Go to your repository's settings tab on GitHub, scroll down to the GitHub Pages section, and choose the branch you want to serve your site from (usually the `main` or `master` branch). GitHub Pages will then build and deploy your site automatically.

5. Custom Domain (Optional): If you have a custom domain, you can configure it to point to your GitHub Pages site by adding a CNAME file to your repository and setting up DNS records with your domain registrar.
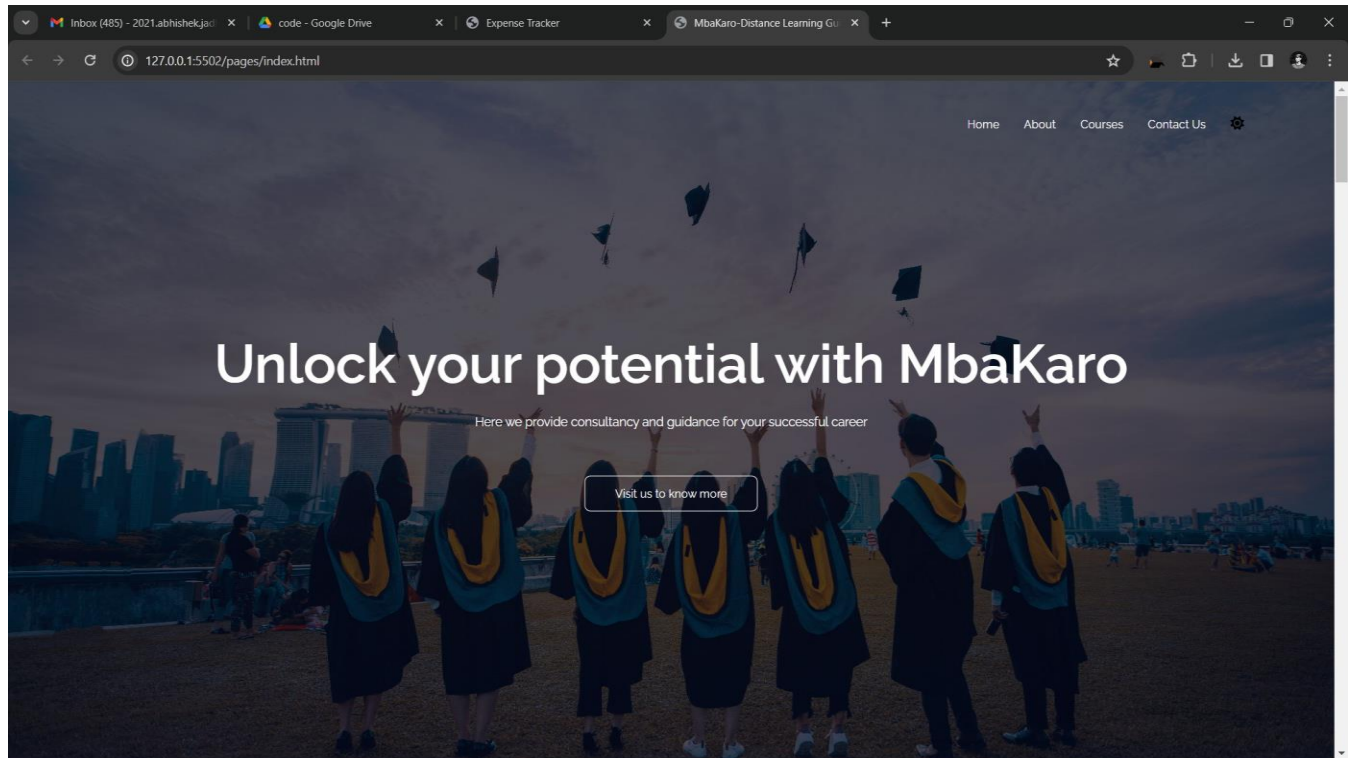
6. Access Your Site: Once GitHub Pages has deployed your site (usually just a few minutes), you can access it via `https://<username>.github.io` or `https://<repositoryname>.github.io`, depending on whether it's a personal or project site.

With these steps, you can easily host your website using GitHub Pages, leveraging GitHub's version control features and seamless deployment process.

Output:

Github Repo Link: https://github.com/Abhishekk24/pwa_ecom

Conclusion:

In conclusion, studying and implementing the deployment of an Ecommerce Progressive Web App (PWA) to GitHub Pages has provided valuable insights into the process of hosting web applications on a static hosting platform. Leveraging GitHub Pages offers a cost-effective and efficient solution for showcasing and distributing PWAs, ensuring wider accessibility and reach for users. This deployment method enhances the visibility and usability of the Ecommerce PWA, facilitating seamless access for potential customers.