Meet App: a case study

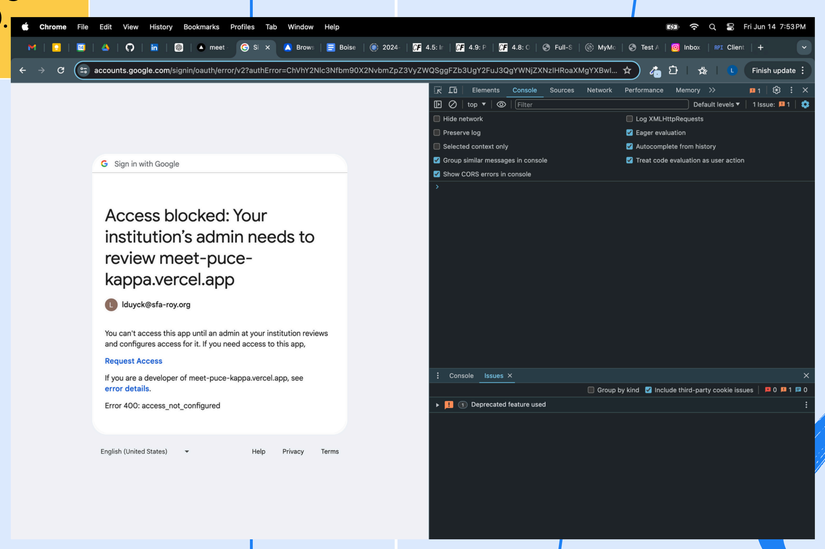
Planning

Objective:
Objective: As part of my CareerFoundry curriculum, this app is designed as an event management tool. It was built using TDD (Test-Driven Development). As per the project requirements, it is built using React, deployed as a PWA (Progressive Web App), includes real-time event updates through the Google Calendar API, and authenticates users through Google OAuth 2.0.

Conceptualization:
The development for this app started with clear user stories and Gherkin stories, providing a structured vision for the app and its key features. The goal was to make a user-friendly app that allows users to search for customized event data, visualize event data through dynamic charts, and use the app while offline.

Development

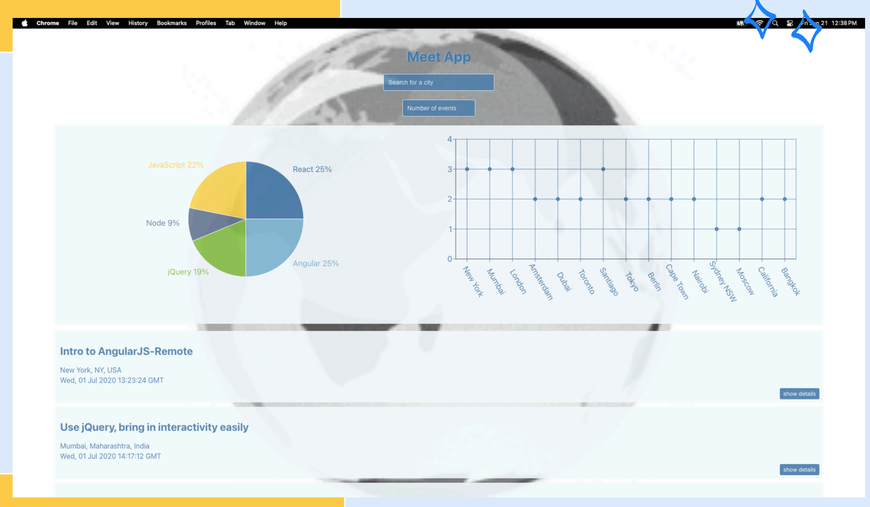
Structure:
I initiated the setup via CRA (Create React App), again demonstrating my progress in demystifying the terminal. Throughout this project, I used the terminal as much as GitHub Desktop or my PC interface. CRA provided a solid foundation for the app, and I began integrating the necessary technologies.



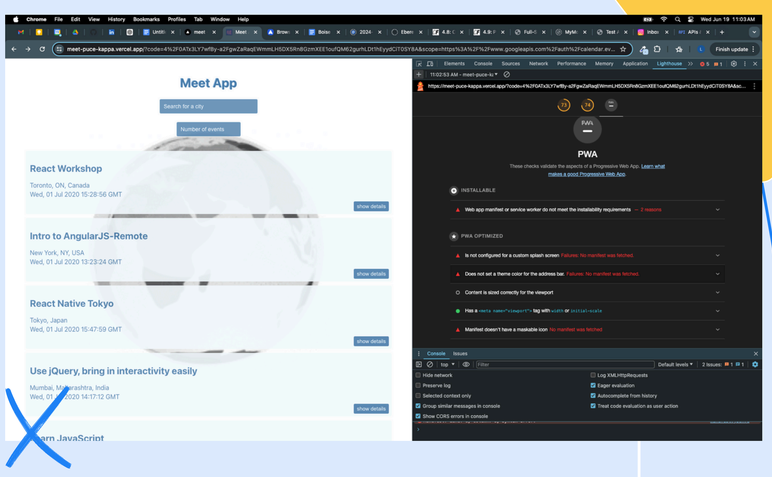
Testing:
Testing was an integral part of the development process, especially since the app was built using TDD. I relied on ChatGPT for assistance during feature testing, which ensured that all functionalities met the required specifications.

Design:
For state management, I opted to use React instead of Redux for simplicity and effectiveness. Building with React was a smooth experience due to its organization-based logic, and my comfort level with the framework continued to grow throughout the project. By using React in a SPA design, I maximized the benefits of React's virtual DOM and asynchronous data fetching, providing users with a dynamically responsive app without page refreshes. React Hooks (useState, useEffect) facilitated real-time updates in the UI. I further customized the basic UI using React-Bootstrap to provide a consistent and visually appealing design, and Recharts for dynamic data visualization. Integrating Recharts was particularly enjoyable, and I see a lot of potential for using this resource in future projects.

Authorization:
For authentication, I used Google OAuth 2.0. Setting this up required some troubleshooting. With the help of a few friends acting as test users, I was able to understand the difference between 'testing' and 'production' modes. Successfully implementing Google OAuth 2.0 felt like a significant achievement, as it's a prevalent feature in modern apps.



Deployment:
Deployment was straightforward, thanks to Vercel's excellent platform. Vercel provides clear build logs, very fast build times, and auto-deployment, making the process smooth and efficient.



Serverless Functions: Next, I integrated AWS Lambda serverless functions to handle real-time data processing, with API requests managed by API Gateway. Understanding and implementing serverless functions was initially challenging. To enable offline capabilities, I added Service Workers for data and asset caching. Additionally, I included cookies to manage user sessions for enhanced offline accessibility. This stage involved considerable troubleshooting, particularly with getting Service Workers to function correctly. A bit of brainstorming with my tutor and independent troubleshooting with ChatGPT resolved these issues and ensured everything worked as expected, passing all PWA tests.

Reflection

Conclusion:
Building the Meet App was a significant milestone in my development journey. It allowed me to explore and integrate several advanced technologies, including AWS Lambda, Google OAuth 2.0, and Recharts, while applying TDD principles to ensure a high-quality product. Throughout this project, I honed my problem-solving skills, continuously learned new technologies, and demonstrated the ability to troubleshoot complex issues independently. If I were to rebuild the app, I would consider using a different Google Calendar API with broader relevance so a greater number of users can find this app useful.

See this repo on GitHub: https://github.com/leanneduyck/Meet.git or see the live site: https://meet-puce-kappa.vercel.app/