**Project Details Architecture**

This project is built using React and follows a component-based architecture. Key aspects:

* Components are organized into pages, layouts, and reusable UI components
* Global state management via React Context API
* React Router for routing and navigation
* Integration with Appwrite for user management and database
* Integration with Stripe for payments

Main folders:

* Pages - Top level page components
* Components - Reusable UI components
* Context - React context and providers
* Config - Environment variables and configs
* Services - Integrations with external services like Appwrite and Stripe

**Running the Project**

To run the project locally:

1. Cd into the main folder
2. Install Node.js from <https://nodejs.org/en/download/current>
3. Install yarn by running command: npm install --global yarn
4. Run `yarn install` to install dependencies
5. Create .env file with Appwrite and Stripe API keys
6. Run yarn run dev to start the dev server
7. Access the app at [http://localhost:3000](http://localhost:3000/)

To run tests:

`yarn run test`

This will run all tests with Jest and provide a coverage report.

**Leveraging Generative AI**

This project leveraged generative AI models like ChatGPT during development for accelerating parts of the process.

See chat history with model prompts and responses in the same folder:

* ChatGPT\_history.pdf
* ChatGPT\_history\_2.pdf

The chat history provides examples of how the AI model was used to generate code snippets, documentation, and tests based on descriptions and context from the code base. This helped accelerate development by automating repetitive tasks.