

React Native Frontend Assessment

E-Commerce Product Listing App

Objective: Build a simple e-commerce product listing mobile app that displays products and their details using React Native Framework for Android or iOS.

Requirements:

Fetch product data from the FakeStore API (<https://fakestoreapi.com/>).

Display a grid/list of products with images, titles, and prices.

Implement a product details page with a description and "Add to Cart" button.

Use React Navigation for navigation between screens.

Store cart items in AsyncStorage and display the cart total.

Brownie Point:

- ✓ Implement a wish list feature.
- ✓ Add a search and filter option.

```
=====
=====
|||||
```

Python Backend LLM Assessment

AI Agent - Stock Market Agent

Description : Create an AI Agent Python, LangChain, and LLM API that can:

- Fetch the latest stock market prices for the asked ticker/stock.
- Create an API which should be deployed on a public server and accessible using curl or postman or similar tool.
- Provide code documentation and a README file explaining the setup & execution process.
- You may use a websearch engine (hosted locally or via API integration) to search for latest stock price like duckduckgo, brave etc.

Requirements :

- Use LangChain for agent development.
- Use OpenAI/Claude or similar LLM API for generating responses and analysis.
- Create a requirements.txt or Pipfile for dependencies.
- Document the code and structure the project properly.

Brownie Point:

- Analyze the prices and provide appropriate buy/sell/hold recommendation.
- This can be clubbed as same response or a follow up request response.

```
=====
=====
|||||
```

Java Backend LLM Assessment

PDF Reader API Using LLM

=====

Description : Create an API using Java and integration with LLM API that can:

- Parse a CASA statement PDF, this is generally the PDF which you get monthly from your bank account
- Create an API which should be deployed on a public server and accessible using curl or postman or similar tool.
- Provide code documentation and a README file explaining the setup & execution process.
- The parsing can help with an external tool, preferably using an LLM.
- Upon executing the API, response should pass name, email, opening balance and closing balance.

Requirements :

- Use OpenAI/Claude or similar LLM API for parsing PDF and gathering apt data points.
- The API can take PDF file as input or pick it from a certain location on disk or from project structure.
- Document the code and structure the project properly.
- You may use an unlocked file (password already provided).

Brownie Point:

- Take Firstname, date of birth (or other required data points) as API input and create password on the fly for the chosen type of bank account

=====