E-Commerce Application Clone

Low Level Design Documentation (LLD).

Project for:- iNeuron (E-Commerce Application Clone Internship) By:- Aman Negi

Context.

1. Abstract:

This project entails the design and development of a frontend-only E-Commerce Application Clone aimed at providing users with a seamless online shopping experience. The application allows users to browse through a collection of products, explore individual items, and conveniently add desired products to their virtual shopping cart. Key functionalities include a dynamic homepage showcasing available products, category pages for organized browsing, and a user-friendly cart page for managing selected items. Leveraging modern frontend technologies such as React.js and React Router, the application ensures smooth navigation and responsive design across various devices. Data mocking techniques simulate backend API calls, enabling the presentation of sample product data, while local storage facilitates temporary cart item storage. Emphasizing user experience, the design integrates accessibility features, performance optimizations. Development tools such as ESLint, Prettier, and version control systems ensure code quality and collaboration efficiency. Testing strategies encompass unit testing, end-to-end testing, and accessibility testing to guarantee functionality and compliance with standards. Finally, deployment involves hosting the frontend application on static site hosting platforms with continuous integration and deployment pipelines for automated testing and deployment processes. This abstract outlines the comprehensive approach to building a robust frontend solution for an E-Commerce Application Clone, catering to modern development standards and user expectations.

2. Introduction.

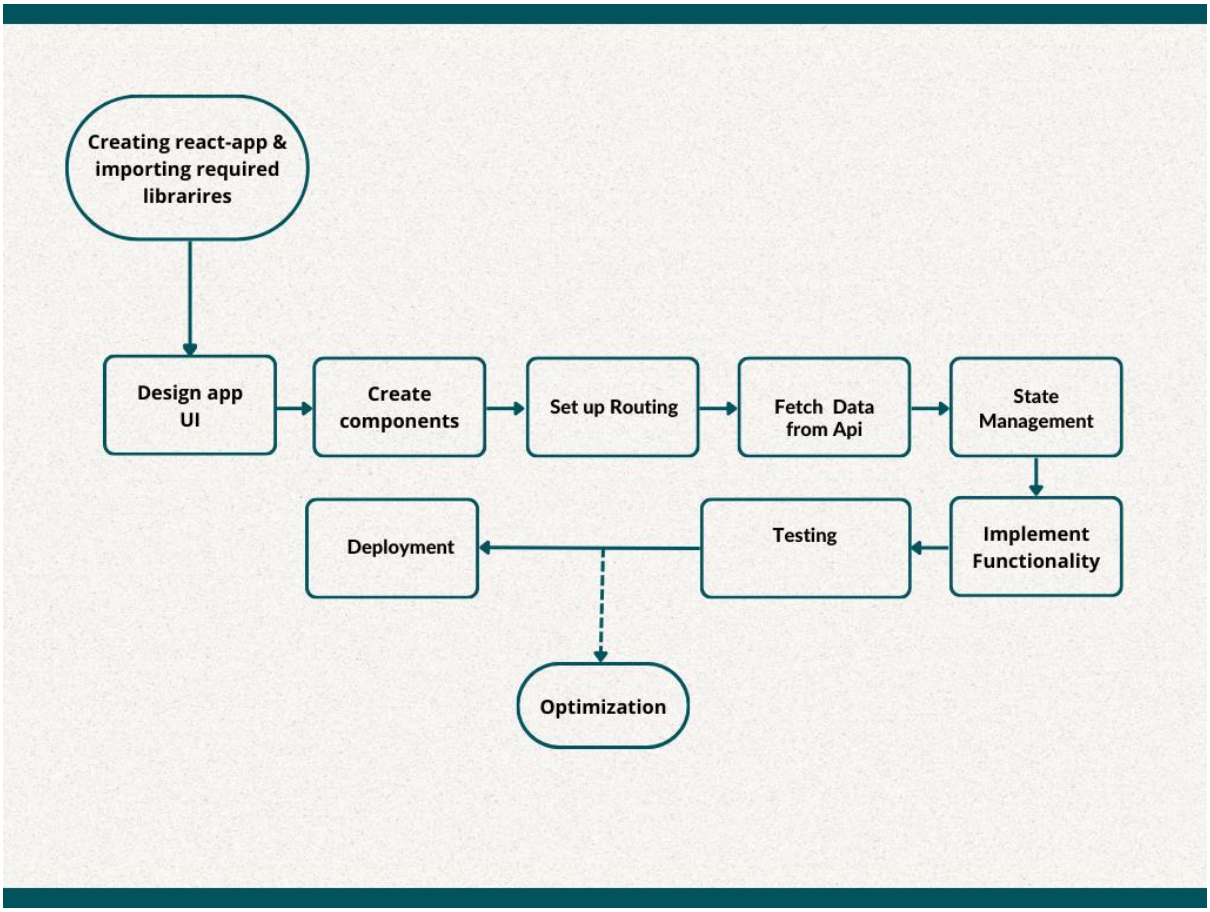
2.1. What is a Low-Level design document?

The goal of LLD or a low-level design document (LLD) is to give the internal logical design of the actual program code for Rental Bikeshare Demand Prediction. LLD describes the class diagrams with the methods and relations between classes and program specs. It describes the modules so that the programmer can directly code the program from the document.

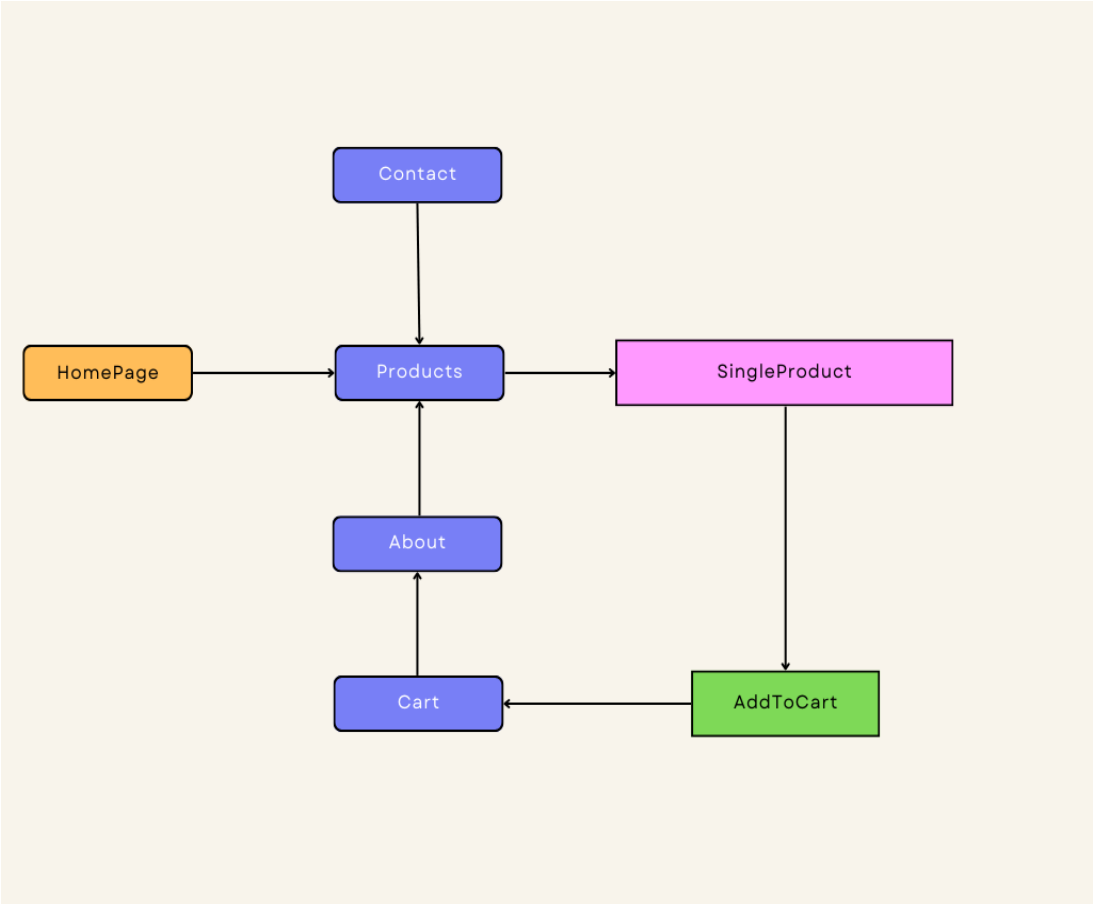
2.2. Scope of (LLD)

Low-level design (LLD) is a component-level design process that follows a step-by-step refinement process. This process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work

2. Architecture



3.1 User Interface(UI) flow



3.3. Tools / Software Used.

- Integrated Development Environment (IDE)
- Node.js and npm (Node Package Manager)
- React.js and React Router
- Git: Version control system for tracking changes in codebase.
- GitHub: Platforms for hosting and managing Git repositories.
- Create React App: Development environments for setting up a local server and building React applications.

3.4. Data Handling

- Utilizing React Context API along with Axios for data handling.
- Context API provides a centralized state management solution for sharing data across components.
- Axios is used for making HTTP requests to external APIs and fetching data.

3.5. Error Handling

- Utilize try-catch blocks to handle exceptions and errors.
- Surround critical sections of code, such as API calls or asynchronous operations, with try-catch blocks to catch and handle any potential errors.
- Describe how try-catch blocks are used to prevent unhandled exceptions from crashing the application and provide a fallback mechanism for error recovery.

4.Conclusion.

We have successfully built an e-commerce web application clone using ReactJs. Its user-friendly interface, advanced filtering options, and seamless checkout process, the app aims to meet the diverse needs of our customers and enhance their online shopping journey.