### A

#### *PROJECT REPORT*

*on*

### Amazon Clone

*Submitted in partial fulfilment of the requirements for the degree of*

### BACHELOR OF TECHNOLOGY



Session: - Jan-June 2023

Under Guidance of Aaditya Maheshwari Head of Industry Project CSE & Techno India NJR Institute of Technology

Submitted by

Harsh Arora (19ETCCS023)

Manish Saini (19ETCCS037)

VIII semester, CSE

### DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY, UDAIPUR-313001 MAY – 2023

**A**

#### *PROJECT REPORT*

*on*

### Amazon Clone

*Submitted in partial fulfilment of the requirements for the degree of*

### BACHELOR OF TECHNOLOGY



**Session: - Jan-June 2023**

Under Guidance of Aaditya Maheshwari Head of Industry Project CSE & Techno India NJR

Submitted by

Harsh Arora (19ETCCS023)

Manish Saini (19ETCCS037)

VIII semester, CSE

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY, UDAIPUR-313001 MAY – 2023**

2

### OGY, UDAIPUR-313001 MAY – 2023

A red sign with white text

Description automatically generated with medium confidence

Department of Computer Science and Engineering

Techno India NJR Institute of Technology, Udaipur-313001

# Certificate

This is to certify that project work titled “Amazon Clone” by **Harsh Arora** was successfully carried out in the Department of Computer Science and Engineering, TINJRIT and the report is approved for submission in the partial fulfillment of the requirements for award of degree of Bachelor of Technology in Computer Science and Engineering.

Aaditya Maheshwari Dr. Rimpy Bishnoi

Head of Industry Project Head of Department

CSE, Techno India NJR Institute of Technology Dept. of CSE TINJRIT, Date......................... Date......................

A red sign with white text

Description automatically generated with medium confidence

### Y, U

Department of Computer Science and Engineering

Techno India NJR Institute of Technology, Udaipur-313001

# Certificate

This is to certify that project work titled “Amazon Clone” by **Manish Saini** was successfully carried out in the Department of Computer Science and Engineering, TINJRIT and the report is approved for submission in the partial fulfillment of the requirements for award of degree of Bachelor of Technology in Computer Science and Engineering.

Aaditya Maheshwari Dr. Rimpy Bishnoi

Head of Industry Project Head of Department

CSE, Techno India NJR Institute of Technology Dept. of CSE TINJRIT, Date......................... Date......................

A red sign with white text

Description automatically generated with medium confidence

Department of Computer Science and Engineering

Techno India NJR Institute of Technology, Udaipur-313001

# Examiner Certificate

This is to certify that the following student

### Harsh Arora

of final year B.Tech. (Computer Science and Engineering), was examined for the project work titled

#### “Amazon Clone”

during the academic year 2022 – 2023 at Techno India NJR Institute of Technology, Udaipur

### Remarks:

**Date:**

Signature Signature

(**Internal Examiner**) (**External Examiner**)

Name :- ……………………… Name :- ………………………

Designation:- ……………….. Designation:- ………………..

Department: - ………………. Department: - ……………….

Organization:- ……………… Organization:- ……………

A red sign with white text

Description automatically generated with medium confidence

Department of Computer Science and Engineering

Techno India NJR Institute of Technology, Udaipur-313001

# Examiner Certificate

This is to certify that the following student

### Manish Saini

of final year B.Tech. (Computer Science and Engineering), was examined for the project work titled

#### “Amazon Clone”

during the academic year 2022 – 2023 at Techno India NJR Institute of Technology, Udaipur

### Remarks:

**Date:**

Signature Signature

(**Internal Examiner**) (**External Examiner**)

Name :- ……………………… Name :- ………………………

Designation:- ……………….. Designation:- ………………..

Department: - ………………. Department: - ……………….

Organization:- ……………… Organization:- ……………



Department of Computer Science and Engineering

Techno India NJR Institute of Technology, Udaipur-313001

**CERTIFICATE**

This is to certify that this project report **“Amazon Clone”** is the confide work of **“Harsh Arora, Manish Saini”** who have carried out the project work under my supervision. I approve this project for submission of the Bachelor of Technology in the **Department of Computer Science and Engineering, Techno India NJR Institute of Technology**, affiliated to Rajasthan Technical University, Kota.

#### Aaditya Maheshwari

**Assistant Professor, Project In charge**

Department of Computer Science

# Preface

In this preface, we will provide an overview of the purpose and scope of the presentation, as well as introduce the key concepts and features that will be discussed. The Amazon clone is a popular e-commerce platform that aims to replicate the functionalities and user experience of the renowned Amazon website. By exploring its features, design, and significance in the e-commerce industry, we hope to shed light on the importance and potential of such platforms. Whether you are an aspiring entrepreneur, a developer, or simply interested in the world of online shopping, this presentation will provide valuable insights into the concept and significance of an Amazon clone ecommerce website. Let's dive in and discover the exciting world of e-commerce!

In Chapter 1, we gave an overview of the Amazon Clone Application. Topics include the functionalities, purpose, merits and demerits of the application. The chapter concludes with different genres of the amazon clone available.

Chapter 2 discusses the software requirement and specifications for the application. It is important to understand the correct and basic requirements to use the application. Chapter 2 presents a detailed overview of the tech stack used to create and run the application.

Chapter 3 presents the use case, activity flow, and data flow diagrams. Together, these diagrams provide a comprehensive view of the system, its functionality, and its interactions with external entities. They are essential tools for developers to ensure that the system is designed to meet the needs of the users and stakeholders, and to validate that it functions correctly.

Chapter 4 describes the data dictionary of the application. A data dictionary is a structured description of data elements and their relationships, providing metadata about the data used in an information system or database.

Chapter 5 showcases different real time screenshots of the application.

Chapter 6 describes various types of testing methodologies. Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not.

Chapter 7 summarizes the key findings and outcomes of the work, highlighting any significant contributions. The future scope outlines possible areas for further research or development, based on the findings of the current work.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY, UDAIPUR-313001

**ABSTRACT**

#### 1. Purpose

**1.1 Introduction**

A software development project called the "Amazon Clone Project" aims to build a working copy of the well-known e-commerce site Amazon. Users will be able to browse, search, and buy things online thanks to this initiative, which aims to offer a similar user experience and feature set as the original site.

#### 1.2 Scope

The scope of the project is to display products, so customers can select a variety of products and manage the number of products. Selected items will be collected in a cart. At checkout, the item on the cart will be presented as an order and customers will get the option to pay for that order via Card. For the security of the user, users must log in through login ID and password, so that no unauthorized user can access their account.

#### 2. Document Overview

The remainder of this document is 8 chapters, the first providing introduction of the project. It lists all the functions performed by the system. The second chapter consists of software requirements specification. The third chapter provides details about system analysis and design. The fourth chapter gives data dictionary information. The fifth chapter consists of snapshots of the complete project. The sixth chapter gives testing for the project. The seventh chapter talks about the conclusion and future enhancements of the project. The final chapter concerns with the bibliography.

## ACKNOWLEDGEMENT

We take this opportunity to record our sincere thanks to all who helped us to successfully complete this work. Firstly, We are grateful to our **supervisor Aaditya Maheshwari** for his invaluable guidance and constant encouragement, support and most importantly for giving us the opportunity to carry out this work.

We would like to express our deepest sense of gratitude and humble regards to our

**Head of Department Dr. Rimpy Bishnoi** for giving invariable encouragement in our endeavors and providing necessary facility for the same. Also a sincere thanks to all faculty members of CSE, TINJRIT for their help in the project directly or indirectly.

Finally, We would like to thank my friends for their support and discussions that have proved very valuable for us. We are indebted to our parents for providing constant support, love and encouragement. We thank them for the sacrifices they made so that we could grow up in a learning environment. They have always stood by us in everything we have done, providing constant support, encouragement and love

**Harsh Arora (19ETCCS023)**

**Manish Saini (19ETCCS037)**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY, UDAIPUR-313001

## CONTENTS

[**Abstract**. iii](#_bookmark0)

[**Acknowledgement** iv](#_bookmark1)

[**Table Of Content** v](#_bookmark2)

[**List Of Tables** vii](#_bookmark3)

[**List Of Figures** viii](#_bookmark4)

[**List Of Symbols** ix](#_bookmark5)

CHAPTER : 1 Introduction 1-7

* 1. [Introduction of Amazon Clone 2](#_bookmark6)
  2. Functionalities 2
  3. [Why we need to make this website……………………………………………………………..3](#_bookmark7)
  4. Purpose……………………………………………………………………………………………5
  5. Merit and Demerits …5
  6. Report of Amazon Clone………………………………………………………………………..6
  7. Module of Amazon Clone………………………………………………………………………..6

CHAPTER : 2 Software Requirement Specification 8-17

* 1. JaveScript………………………………………………………………………………………..9
  2. React.js……………………………………………………………………………………………9
  3. Next.js……………………………………………………………………………………………12
  4. Redux…………………………………………………………………………………………….13
  5. Firestore…………………………………………………………………………………………14
  6. System Requirements………………………………………………………………………....15
  7. Full Stack Developer Responsibilities………………………………………………………..16
  8. Full Stack Developer Requirements………………………………………………………….16

CHAPTER : 3 System Analysis And Design 18-24

* 1. Use Case Diagram……………………………………………………………………………19
  2. Context Diagram………………………………………………………………………………21
  3. Data Flow Diagram (DFD)……………………………………………………………………22

CHAPTER : 4 Data Dictionary………………………………………………………….25-27

* 1. Tabel Detail……………………………………………………………………………………26
  2. ER Diagram……………………………………………………………………………………27

CHAPTER : 5 Project Profile 28-35

* 1. Login Page……………………………………………………………………………………..29
  2. Index Page……………………………………………………………………………………..30
  3. Shopping Basket Page………………………………………………………………………,,31
  4. Checkout Page…………………………………………………………………………………32
  5. Order Confirmed Page………………………………………………………………………...33
  6. Order History Page…………………………………………………………………………….34
  7. Header…………………………………………………………………………………………..35
  8. Footer……………………………………………………………………………………………35

CHAPTER : 6 Testing 36-40

6.1. Testing Methodology…………………………………………………………………………..37

6.2. Functional Testing……………………………………………………………………………..37

6.3. Usaibility Testing……………………………………………………………………………….37

6.4. Compatibility Testing…………………………………………………………………………..38

6.5. Performance Testing…………………………………………………………………………..38

6.6. Security Testing…………………………………………………………………………………38

6.7. Cross selling and Recommendation Testing………………………………………………..38

6.8. Test report with test data………………………………………………………………………39

CHAPTER : 7 Conclusion And Future Enhancement……………………………41-42

Conclusion………………………………………………………………………………………….42

Future Scope………………………………………………………………………………………..42

**CHAPTER: 8** **Bibliography**………………………………………………………………….43-44

## 

## LIST OF TABLES

|  |  |  |
| --- | --- | --- |
| **Table No.** | **Table Name** | **Page No.** |
| Table 4.4.1 | Table Details | 18 |
| Table 6.1 | Test Report with test data | 35 |

## 

## LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Figure Name** | **Page No.** |
| Figure 2.1 | Concepts of React.js | 7 |
| Figure 3.1 | Use Case for Portal | 12 |
| Figure 3.2 | Use Case for Admin Panel | 13 |
| Figure 3.3 | Use Case for User Panel | 13 |
| Figure 3.4 | Context Diagram | 14 |
| Figure 3.5 | DFD Level 0 | 16 |
| Figure 3.6 | DFD Level 1 | 16 |
| Figure 4.1 | ER Diagram | 19 |
| Figure 5.1 | Login Page | 21 |
| Figure 5.2 | Index Page | 22 |
| Figure 5.3 | Shopping Basket Page | 23 |
| Figure 5.4 | Checkout Page | 24 |
| Figure 5.5 | Order Confirmed Page | 25 |
| Figure 5.6 | Order History Page | 26 |
| Figure 5.7 | Header | 29 |
| Figure 5.8 | Footer | 30 |

**LIST OF SYMBOLS**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Admin | The only user who has the permission to insert or update category etc. in the database. |
| QA | Quality assurance |
| SCMP | Software Configuration Management Plan |
| SDD | Software Design Document |
| SQAP | Software Quality Assurance Plan |
| SRS | Software Requirements Specification |
| Web Site | A place on the world wide web |
| HTTP | Hyper Text Transfer Protocol |
| PDF | Personal document Format |
| OS | Operating System |
| ER | Entity Relationship |
| CSS | Cascading Style Sheet |

# CHAPTER – I INTRODUCTION

## INTRODUCTION

#### 1.1 Introduction of Amazon Clone

* Amazon Clone Project: An effort in software development to make a copy of the well-known e-commerce site Amazon.
* A working online marketplace modelled after Amazon will be built in order to learn and practice web development skills.
* The following core features are included: user registration and authentication, product catalogue, product search and filters, shopping cart and checkout, payment integration, order management, reviews and ratings, wishlists and favourites, recommendations and personalization, and customer care.
* Learning Opportunities: Develop a large-scale e-commerce platform firsthand and learn about cloud infrastructure, databases, and web frameworks.
* Understanding the complexities of scalability, security, and performance optimisation in an e-commerce environment is one of the challenges.
* Skill Development: Implement best practises in software development and improve web development abilities.
* Key Benefits: Replicating Amazon's essential features and functionality.

#### 1.2 Functionalities provided by Amazon Clone are as follows:

* User Registration and Authentication:
  + Users can create new accounts and provide necessary information for registration.
  + Authentication mechanisms ensure secure login and protect user data.
* Product Catalog:
  + A comprehensive catalog of products spanning various categories, including electronics, fashion, books, etc.
  + Each product listing displays relevant information such as title, description, price and images.
* Shopping Cart and Checkout:
  + Users can add products to their cart for later purchase.
  + The cart functionality allows users to modify quantities, remove items, and calculate the total cost.
  + Users can proceed to the checkout process to enter shipping details, select payment methods, and place orders.
* Payment Integration:
* Integration with popular payment gateways enables secure and convenient online transactions.
* Users can choose from multiple payment options, such as credit/debit cards, net banking, or digital wallets.
* Wishlist and Favorites:
* Users can create wish lists to save products for future reference or potential purchase.
* They can also mark products as favorites for easy access.
* Admin Dashboard:
* An intuitive interface for administrators to manage products, inventory, user accounts, and system settings.
* Admins can add, update, or remove products, monitor sales, manage user accounts, and generate reports.

#### 1.3 Why do we need to make this website?

#### Learning and Skill Development: Building an Amazon clone project provides an opportunity for developers to enhance their web development skills and gain hands-on experience in building a complex and feature-rich application. It allows developers to practice various technologies, frameworks, and design patterns commonly used in e-commerce development.

#### Understanding E-commerce Architecture: Developing an e-commerce website like Amazon requires understanding the underlying architecture, components, and functionalities. By building an Amazon clone, developers can gain insights into the complexities and challenges of designing and implementing a large-scale e-commerce platform.

#### Practical Application: The Amazon clone project serves as a practical application to showcase and apply web development skills in a real-world scenario. It allows developers to demonstrate their abilities in building a functional and user-friendly online marketplace.

#### Customization and Innovation: Creating an Amazon clone provides the opportunity to tailor the platform to specific business requirements and explore innovative ideas. Developers can add unique features, experiment with new technologies, or incorporate additional functionalities to differentiate their platform.

#### Entrepreneurship and Business Ventures: Building an Amazon clone can be the foundation for starting an e-commerce business. It allows entrepreneurs to establish their own online marketplace, cater to a specific niche, or target a particular set of customers.

#### Educational or Research Purposes: Academic institutions or researchers may undertake an Amazon clone project to study consumer behavior, analyze market trends, or explore strategies employed by e-commerce platforms. It can serve as a valuable resource for educational purposes or market analysis.

#### Competitive Analysis: Developing an Amazon clone can help businesses or individuals gain insights into the strategies, features, and user experience of successful e-commerce platforms. It can be used as a tool for competitive analysis, market research, or benchmarking against industry leaders.

#### 

#### 1.4 Purpose

* Proof of Concept: Building an Amazon clone can demonstrate the technical capabilities and feasibility of developing a large-scale e-commerce platform. It serves as a proof of concept to showcase the potential of an idea or validate a business model.
* Research and Analysis: An Amazon clone can be utilized for research purposes, such as studying consumer behavior, analyzing market trends, or conducting experiments related to e-commerce. It can also be used for competitive analysis and benchmarking against established platforms to gain insights into their strategies and success factors.
* Education and Training: Academic institutions or training programs may use an Amazon clone as an educational tool to teach web development, e-commerce principles, or business management. It can serve as a practical case study for students to understand the complexities of building and operating an e-commerce platform.
* Prototyping and MVP Development: An Amazon clone can be developed as a minimum viable product (MVP) to test market demand, gather user feedback, and iterate on the product concept. It allows businesses to quickly validate their ideas and make data-driven decisions before investing in a fully-fledged e-commerce solution.

**1.5 Merit and Demerits of the project**

#### Merits:

* Skill development and practical experience in web development and e-commerce.
* Understanding the complexities of building an e-commerce platform.
* Customization and innovation opportunities.
* Potential for entrepreneurship and business ventures.
* Insights for competitive analysis and benchmarking.

#### Demerits:

* Legal and intellectual property concerns.
* Limited differentiation from the original platform.
* Scalability and performance challenges.
* Time and resource-intensive development.
* Marketing and user acquisition efforts required.
* Ongoing maintenance and updates needed.

#### 

#### 1.6 Reports of Amazon Clone

* Market analysis, including trends, growth potential, and competition in the Amazon clone market.
* User feedback and satisfaction levels with Amazon clone platforms.
* Technical assessments, evaluating scalability, performance, security, and stability of Amazon clone projects.
* Comparative studies, comparing features, functionalities, and user interfaces of different Amazon clone implementations.
* Case studies showcasing successful Amazon clone projects, highlighting their strategies and business models.

#### 1.7 Modules of Amazon Clone:

* User Management:
* User registration and authentication.
* User profile management.
* Address book for shipping and billing details.
* Account settings and preferences.
* Product Catalog:
* Product listing and categorization.
* Product details, including title, description, price, images, and specifications.
* Inventory management.
* Product search and filtering.
* Shopping Cart and Checkout:
* Add/remove products to/from the shopping cart.
* Quantity and pricing calculations.
* Checkout process with shipping, payment, and order summary.
* Payment Integration:
* Integration with payment gateways to process online transactions.
* Support for various payment methods, such as credit cards, debit cards, net banking, or digital wallets.
* Recommendations and Personalization:
* Product recommendations based on user preferences, browsing history, and purchase behavior.
* Personalized product suggestions and offers.
* Admin Dashboard:
* Admin login and role-based access control.
* Product management (add, edit, delete products).
* User management and moderation.
* Order management and reporting.
* Analytics and insights.

# CHAPTER – II

**SOFTWARE REQUIREMENT SPECIFICATION**

## SOFTWARE REQUIREMENT SPECIFICATION

**2.1 JavaScript**

JavaScript is a high-level programming language that is used to add interactivity to web pages. It is a dynamically typed language, which means that the type of a variable is not defined at compile time. Instead, the type of a variable is determined at runtime based on the value that is assigned to it. JavaScript is also a single-threaded language, which means that only one piece of code can be executed at a time.

JavaScript is a popular language for web development, and it is used by millions of developers around the world. JavaScript can be used to create a variety of interactive web pages, including forms, games, and animations. JavaScript can also be used to access and manipulate the DOM (Document Object Model), which is a tree-like representation of a web page.

### 2.2 React.js

ReactJS is a JavaScript library for building user interfaces. It uses a component-based architecture, provides a virtual representation of the DOM for efficient updates, and follows a declarative syntax. With JSX, developers can write HTML-like syntax within JavaScript. React promotes reusability, supports unidirectional data flow, and introduced hooks for managing state in functional components. It has a large ecosystem and community support, making it popular for building interactive web applications.

#### 2.2.1 Features of React.js:

ReactJS offers several notable features that contribute to its popularity among developers. Here are some key features of ReactJS:

* Virtual DOM: React uses a virtual representation of the Document Object Model (DOM) called the Virtual DOM. It efficiently updates and renders only the necessary parts of the UI, resulting in better performance and faster rendering.
* Component-Based Architecture: React follows a component-based approach, where the UI is divided into reusable and self-contained components. This promotes code reusability, modular development, and easier maintenance.
* JSX: React uses JSX, a syntax extension that allows developers to write HTML-like code within JavaScript. JSX simplifies the creation of UI components by providing a familiar syntax and enabling the combination of JavaScript logic and UI structure.
* Declarative Syntax: React uses a declarative syntax, allowing developers to describe how the UI should look based on the application state. Developers specify the desired UI output, and React takes care of updating and rendering the components efficiently.
* Unidirectional Data Flow: React follows a unidirectional data flow, where data flows from parent components to child components via props. This makes it easier to understand and track data changes, leading to predictable and manageable state management.
* React Hooks: React introduced hooks, which are functions that enable the use of state and other React features in functional components. Hooks simplify the management of component state, side effects, and lifecycle methods, eliminating the need for class components in many cases.
* Component Lifecycle Methods: React provides a set of lifecycle methods that allow developers to hook into specific phases of a component's lifecycle, such as initialization, rendering, and unmounting. This enables developers to control and manage component behavior effectively.
* Virtual Testing: React's component-based architecture and virtual rendering make it easier to write unit tests for components. Developers can simulate different states and interactions to ensure the correctness of the UI.
* Rich Ecosystem and Community: React has a thriving ecosystem with a wide range of libraries, tools, and community support. This makes it easier for developers to find solutions, leverage existing components, and integrate React with other technologies.

These features contribute to ReactJS's efficiency, flexibility, and developer-friendly nature, making it a popular choice for building interactive and dynamic user interfaces in web applications.

#### 2.2.2 Who uses React.js?

ReactJS is used by a wide range of companies and organizations across different industries. Here are some examples of entities that use ReactJS:

* Facebook: ReactJS was developed by Facebook, and it continues to be extensively used within the company for various products and features, including the Facebook website itself.
* Instagram: Instagram, which is owned by Facebook, also utilizes ReactJS for its web interface. React's component-based architecture allows for easier management of the complex UI elements found in Instagram.
* Airbnb: The popular online marketplace for lodging and vacation rentals, Airbnb, uses ReactJS to build its user interface. React's performance and reusability make it well-suited for handling the dynamic search and filtering functionality on the platform.
* Netflix: Netflix leverages ReactJS for its customer-facing web interface. React's efficiency in rendering and updating components aligns with the need for seamless browsing and playback experiences on the platform.

#### 2.2.3 Concepts:

The following diagram depicts some important parts of React.js which we will discuss in detail in the subsequent.

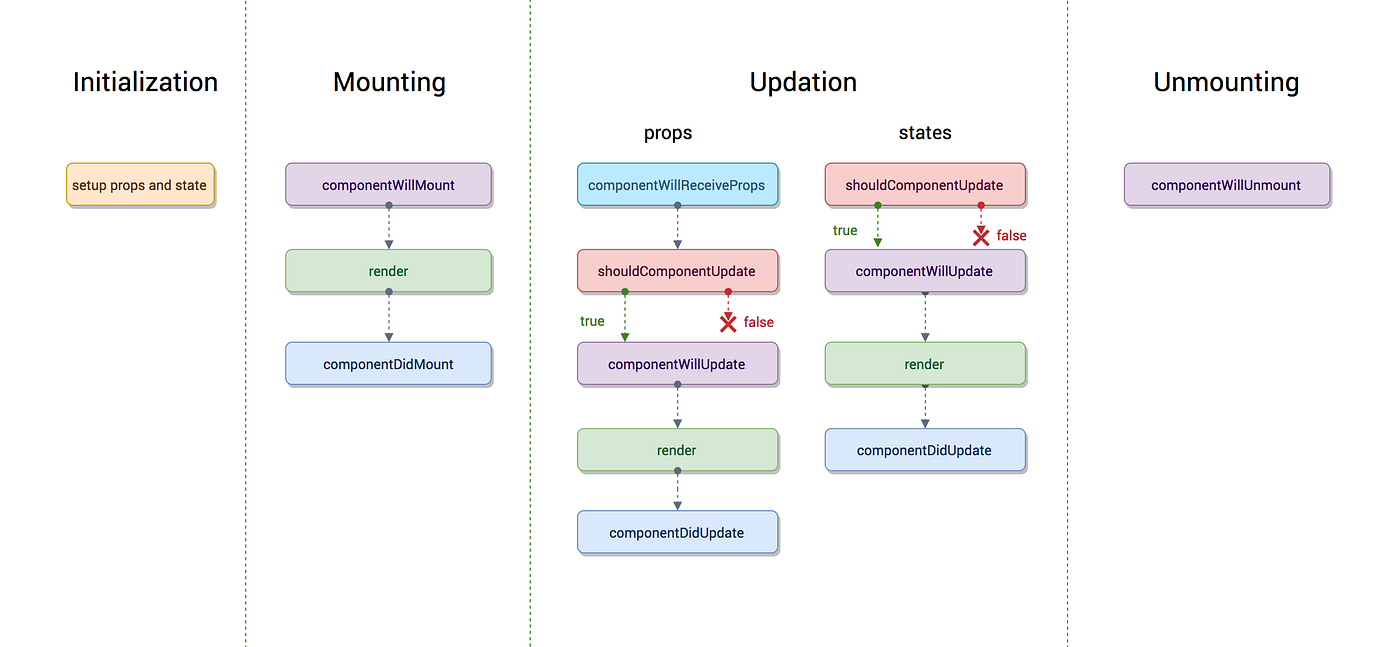


Fig 2.1 – Concepts of React.js

**2.2.4 Where do we use React.js?**

* Web Applications: ReactJS is widely used for building single-page applications (SPAs) and complex web interfaces. It helps create responsive, interactive, and efficient UIs for a wide range of web applications.
* Mobile Applications: React Native, a framework built on top of ReactJS, enables developers to build cross-platform mobile applications. By utilizing ReactJS concepts and components, developers can create native-like mobile apps for iOS and Android platforms.
* E-commerce Platforms: ReactJS is well-suited for building e-commerce platforms and online shopping websites. Its component-based architecture allows for modular development, easy integration with back-end systems, and efficient management of product catalogs and user interactions.
* Social Media Applications: Social media platforms often require real-time updates, dynamic content rendering, and interactive user interfaces. ReactJS's virtual DOM and component reusability make it a suitable choice for developing engaging social media applications.
* Content Management Systems (CMS): ReactJS can be used to build user-friendly interfaces for content management systems. It allows for easy content editing, drag-and-drop functionality, and seamless updates to the UI without page reloads.

#### 2.3 Next.js

Next.js is a powerful framework built on top of React.js that facilitates server-side rendering (SSR) and static site generation (SSG). It simplifies the development of React applications by providing an intuitive file-based routing system and a Link component for seamless client-side navigation. Next.js enables faster page loads, better SEO performance, and improved user experience through SSR and SSG capabilities. Its versatility makes it suitable for building a wide range of applications, from dynamic web apps to static websites, and its integration with React's component model and ecosystem ensures a smooth development experience.

**Installation command** - npx create-next-app@latest your-app-name

#### 2.3.1 Applications of Next.js:

* Single-Page Applications (SPAs): Next.js can be used to build interactive SPAs with server-side rendering (SSR) capabilities. It provides efficient routing, data fetching, and client-side navigation, resulting in fast and responsive web applications.
* E-commerce Websites: Next.js is suitable for building e-commerce platforms and online stores. Its server-side rendering and static site generation capabilities help in optimizing SEO, improving performance, and providing a better user experience for online shopping.
* Content Management Systems (CMS): Next.js can power content management systems, allowing for easy content creation, editing, and publishing. Its server-side rendering and API routes enable seamless integration with back-end systems and efficient content delivery.
* Blogs and Publishing Platforms: Next.js is commonly used for creating blogs and publishing platforms. Its server-side rendering and static site generation features enable fast loading times and improved SEO for blog posts and articles.
* Dashboards and Analytics Platforms: Next.js is suitable for developing dashboards and analytics platforms that require real-time data updates and interactive visualizations. Its server-side rendering capabilities facilitate data fetching and rendering on the server, ensuring optimal performance and usability.

#### 2.4 Redux

Redux is a popular JavaScript state management library commonly used with frameworks like React. It follows a unidirectional data flow pattern, where the application state is stored in a single store, and changes are made through dispatched actions. Reducers, pure functions, handle these actions and update the state immutably. Redux promotes predictability, consistency, and helps manage complex application states efficiently by providing a centralized and structured approach.

**Installation command** - npm install redux

The above command saves the installation locally in the node modules directory and creates a directory redux inside node modules. You should install the following important modules along with redux –

`Provider` component, which allows you to wrap your React application and provide access to the Redux store. It also provides the `connect` function, which connects your React components to the Redux store, enabling them to access state and dispatch actions.

\*redux-thunk\*: This middleware is frequently used to handle asynchronous actions in Redux. It allows you to dispatch functions as actions, giving you the ability to perform asynchronous operations, such as API calls, before dispatching the actual actions. Redux Thunk is widely used for managing side effects and asynchronous logic in Redux applications.

\*redux-logger\*: This middleware provides logging capabilities for Redux actions and state changes. It logs detailed information about dispatched actions and state updates to the console, making it easier to track and debug changes in your application's state over time. The logging output helps in understanding how actions flow through reducers and affect the state.

\*reselect\*: This library helps create memoized selectors in Redux. Selectors allow you to compute derived data from the Redux state, and Reselect optimizes their performance by caching the results. By using memoization, Reselect avoids unnecessary recomputations of selectors, enhancing the efficiency of your application.

#### 2.5 Firestore – Database

#### Firestore is a flexible and scalable NoSQL cloud-based database provided by Google Cloud Platform. It is designed to store and synchronize data for web, mobile, and server applications. Here are some key features and concepts related to Firestore:

#### \*Document-based model\*: Firestore organizes data into documents, which are structured as collections of key-value pairs. Each document is identified by a unique document ID and resides within a collection.

#### \*Collections\*: Collections are containers that hold related documents. They act as top-level entities for organizing data. Collections can contain multiple documents, and they provide an additional level of hierarchy within Firestore.

#### \*Fields\*: Fields are the individual key-value pairs within a document. Each field has a name (key) and a value associated with it. Firestore supports various data types, including strings, numbers, booleans, timestamps, arrays, and nested objects.

#### \*Queries\*: Firestore offers powerful querying capabilities to retrieve data from the database. You can perform queries to filter documents based on specific conditions, sort and order results, and limit the number of retrieved documents. Queries are performed using the Firestore Query API.

#### \*Real-time updates\*: One of Firestore's notable features is its real-time synchronization. It provides real-time updates to clients whenever changes occur in the database. This allows multiple clients to stay in sync with the latest data and receive immediate notifications about modifications.

#### \*Security rules\*: Firestore allows you to define security rules to control access and enforce data validation. These rules determine who can read or write to specific documents or collections and help ensure data integrity and security.

#### \*Integration\*: Firestore can be used with various platforms and programming languages, including JavaScript (for web applications), iOS (for mobile apps), and Android (for mobile apps). It provides SDKs and client libraries to interact with the database from different environments.

#### 2.6 System Requirements

#### Supported Operating Systems

* Windows 7 and above
* Windows Server 2008 R2+

#### Development Environments

By using the following IDEs, you can develop the Bold Reports React:

* [Microsoft Visual Studio Code](https://code.visualstudio.com/)
* Internet Information Services (IIS) 7.0+
* [Next JS](https://nodejs.org/en/) (version 12.x.x)
* [NPM](https://docs.npmjs.com/getting-started/installing-node#install-npm--manage-npm-versions) (v3.x.x or higher)
* [React JS](https://nodejs.org/en/) (version 18.x.x)

#### Browser Compatibility

* IE 9+
* Microsoft Edge
* Mozilla Firefox 22+
* Chrome 17+
* Opera 12+
* Safari 5+

**2.7 Full Stack Developer Responsibilities:**

* Developing front end website architecture.
* Designing user interactions on web pages.
* Developing back-end website applications.
* Creating servers and databases for functionality.
* Ensuring cross-platform optimization for mobile phones.
* Ensuring responsiveness of applications.
* Working alongside graphic designers for web design features.
* Seeing through a project from conception to finished product.
* Designing and developing APIs.
* Meeting both technical and consumer needs.
* Staying abreast of developments in web applications and programming languages.

#### 2.8 Full Stack Developer Requirements:

* Degree in Computer Science.
* Strong organizational and project management skills.
* Proficiency with fundamental front end languages such as HTML, CSS and JavaScript.
* Familiarity with JavaScript frameworks such as Angular JS, React and Amber.
* Proficiency with server-side languages such as Python, Ruby, Java, PHP and .Net.
* Familiarity with database technology such as MySQL, Oracle and MongoDB.
* Excellent verbal communication skills.
* Good problem-solving skills.

# 

# CHAPTER – III

**SYSTEM ANALYSIS AND DESIGN**

## SYSTEM ANALYSIS AND DESIGN

#### 

#### 3.1 Use Case Diagram:

Use-case diagrams illustrate and define the context and requirements of either an entire system or the important parts of the system. You can model a complex system with a single use-case diagram, or create many use-case diagrams to model the components of the system. You would typically develop use-case diagrams in the early phases of a project and refer to them throughout the development process.

Use-case diagrams are helpful in the following situations:

* + - Before starting a project, you can create use-case diagrams to model a business so that all participants in the project share an understanding of the workers, customers, and activities of the business.
    - While gathering requirements, you can create use-case diagrams to capture the system requirements and to present to others what the system should do.
    - During the analysis and design phases, you can use the use cases and actors from your use- case diagrams to identify the classes that the system requires.
    - During the testing phase, you can use use-case diagrams to identify tests for the system.

A picture containing text, diagram, line, parallel

Description automatically generated

Fig 3.1. – Use case Diagram of Portal

#### 3.3.1 Use Case Diagram for Admin:

##### A picture containing diagram, line, text, circle Description automatically generated

##### Fig 3.2 – For Admin Panel

##### (Login)

##### (Manage Products)

##### (Add Product)

##### (Edit Product)

##### (Delete Product)

##### (View Order Details)

#### 3.3.2 Use Case Diagram for User:



**Contact**

**Open**

**Website**

Add to Basket

Fig 3.3 – For User Panel

* Login as user and see the products.
* Search for products.
* View product detail.
* Add product to card.
* Checkout

#### 3.2 Context Diagram:

A context diagram, sometimes called a level 0 data-flow diagram, is drawn in order to define and clarify the boundaries of the software system. It identifies the flows of information between the system and external entities. The entire software system is shown as a single process.

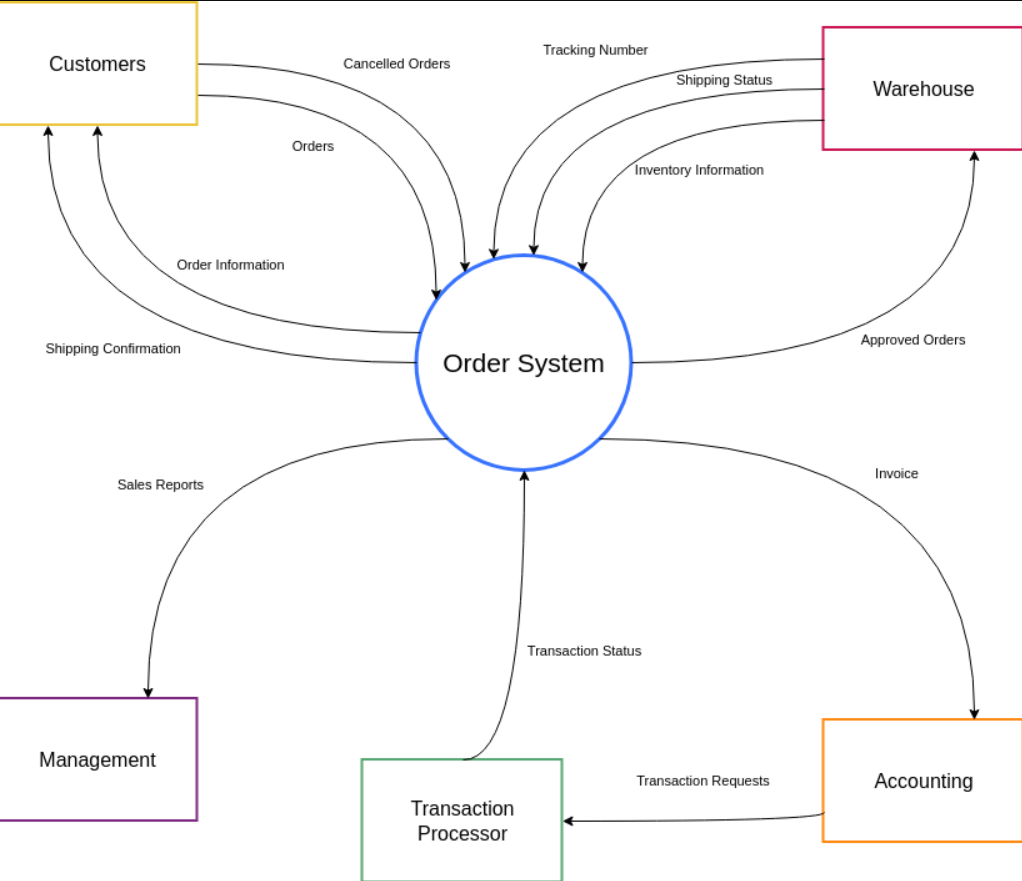


Fig 3.4 – Context Diagram

#### 3.3 Data Flow Diagram (DFD):

A DFD is a graphical representation of the “flow” of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated DFDs can also be used for the visualization of data processing (structured design).

**Request**

**View Website**

**Website**

**User**

**Server**

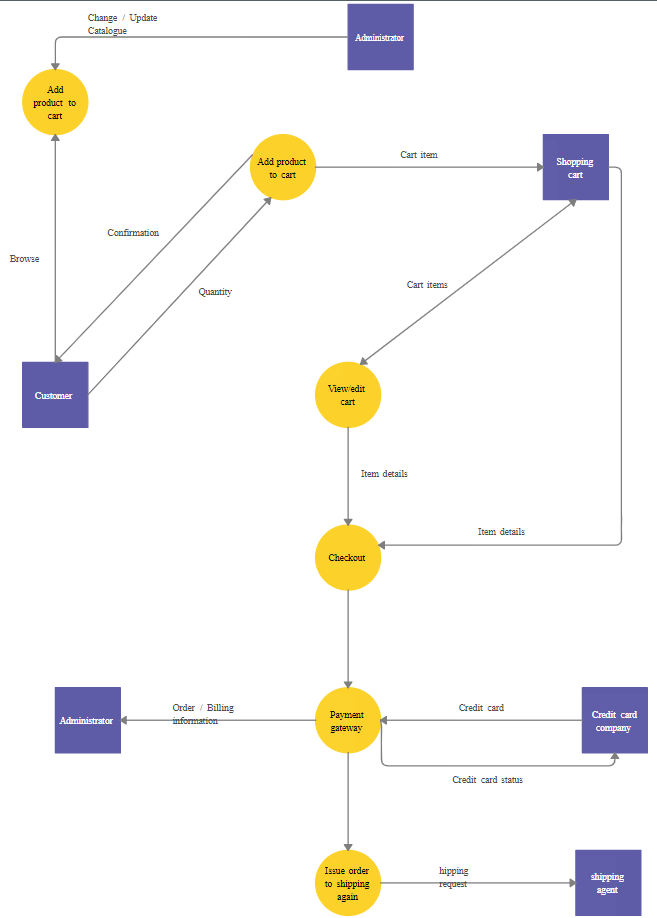
**Response**

##### 

##### Fig 3.5 – DFD Level 0

This is a 0 Level Data Flow Diagram of the amazon clone. In this DFD, user can see the products and website will send request to server and server gives respond of that request. In this diagram the portal accommodates three entities that are as follows:

* User
* Products Website
* Server response



##### Fig 3.6 – DFD Level 1

As described previously, context diagrams (level 0 DFDs) are diagrams where the whole system is represented as a single process. A level 1 DFD notates each of the main sub-processes that together form the complete system. We can think of a level 1 DFD as an “exploded view” of the context diagram.

# 

# CHAPTER – IV DATA DICTIONARY

## DATA DICTIONARY

A data dictionary is a catalogue-a-repository of the elements in a system. As the name suggests, their elements centre on data and the way they are structured to meet user requirements and organization needs. In a data dictionary you will find a list of all the elements composing the data flowing through a system. The major elements are data flows, data stores and processes. The data dictionary stores details and descriptions of these elements.

If analysis want to know characters are in a data item by what other names it is referenced in the system, or where it is referenced in the system, or where it is issued in the system, they should be able to find the answers in issued in the system, they should be able to find the answer in properly developed data dictionary.

The Dictionary contains two types of description for the data following through the system.

##### Data Elements:

The most fundamental data is the elements. They are building blocks for all other data in the system. Data elements are also alternatively known as fields, data item or elementary item.

##### Data Structure:

A data structure is a set if items that are related to one another and described a components in the system.

#### 

#### 4.1 Table Details

* Product Details: {Electronics + Clothes + Jewelry…}
* User Details: {User Id + Email Id + Password}
* Content Details: {Email Id + Message}
* Account Details: {User+ Old Password + New Password}
* View Detail: {Headlines + Description} Search Detail: {Product + Category}

**4.4.1 Admin Login:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Description** | **Size** | **Data Type** |
| Username | Unique username of the user | 50 | varchar |
| Password | User password | 50 | varchar |

#### 

#### 4.2 ER Diagram

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system.

A picture containing text, screenshot, diagram, font

Description automatically generatedFig. 4.1 – ER Diagram

# CHAPTER – V PROJECT PROFILE

## PROJECT PROFILE

#### 5.1 Login Page

# 

##### Fig 5.1 – Login Page

The login page allows a admin to gain access to an application by entering their username and password.

#### A screenshot of a website Description automatically generated with medium confidence 5.2 Index Page

##### Fig 5.2 – Index Page

* The home page is the name of the main page of a website where visitors can find hyperlinks and functionality to find other pages.
* Visitor navigating to a website from a search engine will see.

**5.3 Shopping Basket Page**

##### 

##### 

##### Fig 5.3 – Shopping Basket Page

The shopping basket allows customers to review and modify their selections before proceeding to the checkout process. They can remove or adjust quantities of items, apply any applicable discounts or promotional codes, and sometimes even save items for later purchase. This feature enables customers to have a convenient and personalized shopping experience, as they can easily keep track of the items they intend to buy and adjust as needed.

#### 5.4 Checkout Page

#### A screenshot of a computer Description automatically generated with medium confidence

##### Fig 5.4 – Checkout Page

A checkout page is the final step in the online shopping process where customers provide their billing and shipping information to complete a purchase. It is a secure web page where customers enter details such as their name, address, payment method, and any applicable discount codes. The checkout page typically includes a summary of the items in the shopping basket, the total cost, and shipping options. Once the necessary information is provided, customers can proceed to place the order and finalize the transaction.

#### 5.5 Order Confirmed Page

**A screenshot of a computer

Description automatically generated with medium confidence**

##### Fig 5.5 – Order Confirmed Page

The confirmation email or page may also provide information about shipping or delivery options, estimated delivery dates, and any tracking numbers available. It serves as a receipt for the customer, providing them with a record of their purchase. Additionally, order confirmations often include customer support contact information and instructions on how to make changes or contact the seller if necessary. Receiving an order confirmation provides customers with peace of mind and reassurance that their purchase was successful.

#### 5.6 Order History Page

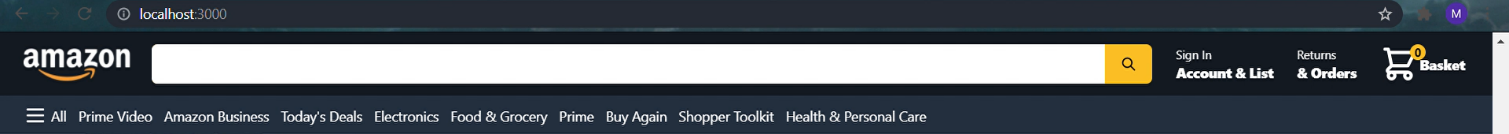
**A screenshot of a website

Description automatically generated with medium confidence**

##### Fig 5.6 – Order History Page

Order history is a record of a customer's past purchases on an e-commerce platform, containing details such as order dates, numbers, items bought, quantities, prices, and discounts. It allows customers to conveniently review their previous orders, reorder items, and check order statuses, while also providing valuable data for inventory management and customer service improvements on the platform's end.Overall, the order history feature enhances the user experience, providing customers with a convenient way to review their past purchases and facilitating smoother interactions between customers and e-commerce platforms.

#### 5.7 Header

****

##### Fig 5.7 - Header

Header is the top section of the web page. Headers are often the first thing site visitors see so having the logo is important.

#### 5.8 Footer

**A picture containing text, screenshot

Description automatically generated**

##### Fig 5.8 - Footer

The footer appears at the bottom of site page. It usually contains small-print items like copyright Information and footer design is about choosing what to include, with the intention to helping Visitors and meeting business goals.

# 

# 

# CHAPTER – VI

# TESTING

## TESTING

#### Testing Methodology

Companies rely on software more than ever to provide and manage information with strategic and operational importance and to provide key decision support. Rising customer expectations for fault-free, requirements-exact software have increased awareness of the importance of software testing as a critical activity.

We begin the testing process by developing a comprehensive plan to test the general functionality and special features on a variety of platform combinations. Strict quality control procedures are used. The process verify that the application meets the requirements specified in the system requirements document and is bug free. At the end of each testing day, we prepare a summary of completed and failed tests. Applications are not allowed to launch until all identified problems are fixed. A report is prepared at the end of testing to show exactly what was tested and to list the final outcomes.

**1. Functional Testing:**

* Test user registration: Verify that users can create new accounts successfully and that all required fields are validated correctly.
* Test product search: Conduct searches using different keywords, categories, and filters to ensure accurate and relevant search results are displayed.
* Test adding items to the cart: Verify that users can add items to the cart, update quantities, and remove items without any issues.
* Test placing orders: Go through the entire order process, including selecting items, entering shipping and payment information, and verifying that orders are successfully placed.
* Test order tracking: Validate that users can track their orders, view order status, and receive timely updates on the progress of their deliveries.

**2. Usability Testing:**

* Evaluate the website's user interface: Assess the overall layout, design, and navigation to ensure it is intuitive and easy to use.
* Test mobile responsiveness: Verify that the website is responsive and adapts well to different screen sizes and orientations on mobile devices.
* Conduct user surveys or interviews: Gather feedback from testers to identify any usability issues, confusing elements, or suggestions for improvement.

**3. Compatibility Testing:**

* Test on different web browsers: Verify that the Amazon clone functions correctly on popular web browsers such as Chrome, Firefox, Safari, and Internet Explorer.
* Test on various operating systems: Validate that the website performs well on different operating systems, including Windows, macOS, and Linux.
* Test on different devices: Ensure that the Amazon clone is compatible with different mobile devices, including smartphones and tablets, across various platforms like iOS and Android.

**4. Performance Testing:**

* Conduct load testing: Simulate high user traffic and transactions to assess the system's performance under peak load conditions.
* Measure response time: Monitor the website's response time for different operations, such as searching, adding items to the cart, and checking out, to ensure optimal performance.
* Test scalability: Validate that the system can handle increasing loads without significant degradation in performance.

**5. Security Testing:**

* Test data encryption: Verify that sensitive user information, such as passwords and payment details, are encrypted during transmission and storage.
* Test authentication mechanisms: Validate the effectiveness of user authentication processes, ensuring that only authorized users can access their accounts.
* Test secure payment processing: Verify that payment transactions are securely processed, using secure payment gateways and protocols.

**6. Cross-selling and Recommendation Testing:**

* Validate product recommendations: Assess the accuracy and relevance of product recommendations based on user preferences, purchase history, and browsing behavior.

**Table 6.1 Test Report with test data**

|  |  |  |
| --- | --- | --- |
| **TEST REPORT WITH TEST DATA**  **(To be filled by System Analyst/Programmer)** | | |
| **Project Name: Amazon Clone** | | |
| **S No.** | **Testing Parameter** | **Observations** |
| A. | INTERFACE TESTING   1. User-friendliness 2. Consistent menus | OK NA |
| B. | CONTROL FLOW TESTING   1. IF-THEN-ELSE 2. DO WHILE 3. CASE-SWITCH | OK OK OK |
| C. | VALIDATION TESTING   1. Check for improper or inconsistent typing 2. Check for erroneous initialization or default values 3. Check for incorrect variable names 4. Check for inconsistent Data Types 5. Check for relational/arithmetic operators | OK OK OK OK OK |
| D. | DATA INTEGRITY/SECURITY TESTING   1. Data Insertion/ Deletion/ Updating 2. Boundary condition (Underflow, Overflow Exception) 3. Check for unauthorized access of data 4. Check for data availability | OK OK  OK OK |
| E. | EFFICIENCY TESTING   1. Throughput of the system 2. Response time of the system 3. Online disk storage required by the system 4. Primary memory required by the system | OK OK OK OK |
| F. | ERROR HANDLING ROUTINES   1. Error description are intelligent/ understandable 2. Error recovery is smooth 3. All error handling routines are tested and executed atleast once | OK OK OK |

# CHAPTER – VII

**CONCLUSION AND FUTURE ENHANCEMENTS**

## CONCLUSION AND FUTURE ENHANCEMENTS

#### 7.1 Conclusion

The Amazon clone project achieved its goal of creating a fully functional replica of the original Amazon platform. The project successfully implemented essential features such as a user-friendly interface, comprehensive product catalog, seamless shopping cart and checkout process, robust user management, order fulfillment and tracking, and an efficient admin dashboard. These elements combined to provide users with a familiar and satisfying shopping experience. The project showcased the ability to develop a complex e-commerce platform and highlighted the skills in web development, user experience design, and system administration. Overall, the Amazon clone project demonstrated successful replication of key functionalities while maintaining a high level of usability and performance.

#### 7.2 Future Scope

The Amazon clone project holds several potential avenues for expansion and improvement. One possibility is the integration of augmented reality (AR) technology, enabling users to visualize products in their real-world environment before making a purchase. Additionally, incorporating voice search and a virtual assistant, such as Amazon's Alexa, can enhance user convenience and accessibility. The introduction of subscription and membership programs, akin to Amazon Prime, could offer exclusive benefits and foster customer loyalty. Social commerce integration would allow users to make purchases directly on social media platforms, leveraging their networks for product discovery and increased sales. Advanced personalization techniques, including machine learning algorithms, could deliver highly targeted product recommendations. Exploring the use of blockchain technology for supply chain transparency and expanding into international markets are further possibilities, by embracing these future scopes.

# CHAPTER – VIII BIBLIOGRAPHY

## BIBLIOGRAPHY

We are using various type of books and websites from which we get the Knowledge about Stack which are following: -

1. Full-Stack JavaScript Development with React.js and Next.js
2. https://reactjs.org
3. www.google.com
4. www.w3schools.com
5. Photoshop
6. https://nextjs.org
7. [www.tutorialspoint.com](http://www.tutorialspoint.com)
8. [www.wikipedia.org](http://www.wikipedia.org)

### A