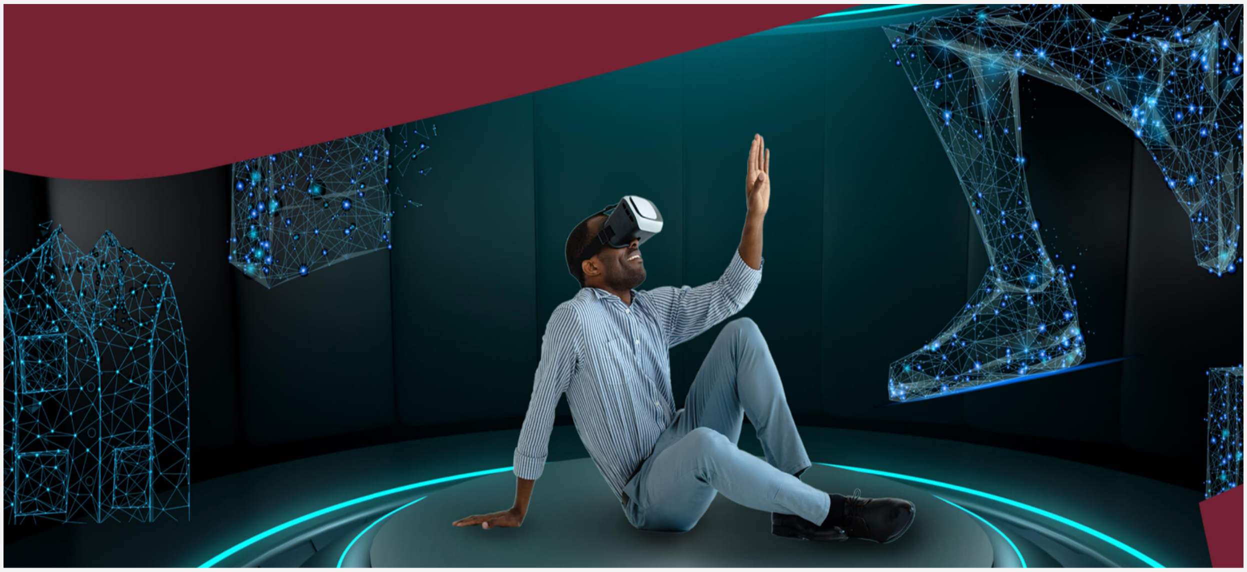
Software Engineering Project



ImmersiVR

GROUP - 3

Harshvardhan 2001084

Uma Kadam 2001095

Harshit Singh 2001083

Executive Summary:

ImmersiVR is a metaverse-based web application that enables users to explore a virtual mall, try on virtual outfits, and shop for items through an e-commerce website. This software requires an active Internet connection, and all user profile details and activities will be stored on a secure database web server. This Software Requirements Specification (SRS) document provides a detailed description of the functionality and requirements for the ImmersiVR e-commerce website, which incorporates futuristic features such as face recognition, AR/VR view of items, metaverse mall, and virtual try-on.

The product features of ImmersiVR include face recognition technology, AR/VR view of items, a metaverse mall, and virtual try-on. The website's design and implementation constraints include using JavaScript and Python programming languages, complying with the process for design, code, and maintenance documentation, and using MongoDB for storing information. The website's functional requirements include the ability for customers to create an account and log in using facial recognition technology, browse products in various categories, and view them in an AR/VR environment. Non-functional requirements include accessibility for people with disabilities, search engine optimization, and compliance with relevant laws and regulations.

The website will have several interfaces, including login/registration, product browsing and viewing, checkout, virtual mall, AI assistant, and virtual try-on interfaces. The website will be accessible on desktop computers, laptops,

tablets, and smartphones, have fast loading times, be easy to navigate and use, and protect customer information.

INTRODUCTION

ImmersiVR is an e-commerce website that aims to create a futuristic immersive e-commerce application that combines the usage of VR, metaverse technologies, and other advanced features. The website will allow customers to view products in AR/VR inside a virtual world assisted by an AI assistant. Customers can create an account and log in using facial recognition technology. The website's primary goal is to provide a user-friendly interface with various functionalities such as the ability to search and filter products and complete transactions securely using a payment gateway.

SDLC Model - EVOLUTIONARY MODEL

Feasibility Study

Technical Feasibility:

The project appears to be technically feasible as the necessary technologies and tools exist to develop the e-commerce website's features. JavaScript and Python programming languages will be used to develop the website, and the website's information will be stored in a MongoDB database. Three.js will be used to create the AR/VR environments, and Flask will handle facial recognition. The implementation will be divided into front-end and back-end development, and testing will be done to ensure the website's functionality. The project appears technically feasible.

Operational Feasibility:

The project's operational feasibility will depend on its usability and user acceptance. The website must be easy to navigate and use, with fast loading times and a responsive interface. User testing must be done to ensure the website meets the users' needs and preferences. The website's features must be marketed effectively to attract users and ensure their engagement.

Overall, based on the analysis above, the ImmersiVR project appears to be feasible, provided that the necessary resources are available, and the project is developed with legal and ethical considerations in mind. A thorough market analysis and testing are required to ensure user acceptance and profitability.

Requirement Analysis

REQUIREMENT ANALYSIS FOR IMMERSIVR

INTRODUCTION- ImmersiVR is an e-commerce website that offers a futuristic immersive shopping experience to its customers.

FUNCTIONAL REQUIREMENTS

- □ 2.1 User Management
- □ 2.2 Product Browsing and Recommendation
- □ 2.3 Virtual Mall
- □ 2.4 Virtual Try-on
- □ 2.5 Checkout and Payment

NON-FUNCTIONAL REQUIREMENTS

- □ 3.1 Performance
- □ 3.2 Security
- □ 3.3 Accessibility

DESIGN CONSTRAINTS

- □ 4.1 Programming Languages
- □ 4.2 Database
- □ 4.3 Documentation

USER INTERFACE DESIGN

- □ Facial recognition login
- □ AR/VR view of products
- □ Metaverse mall
- □ Virtual try-on

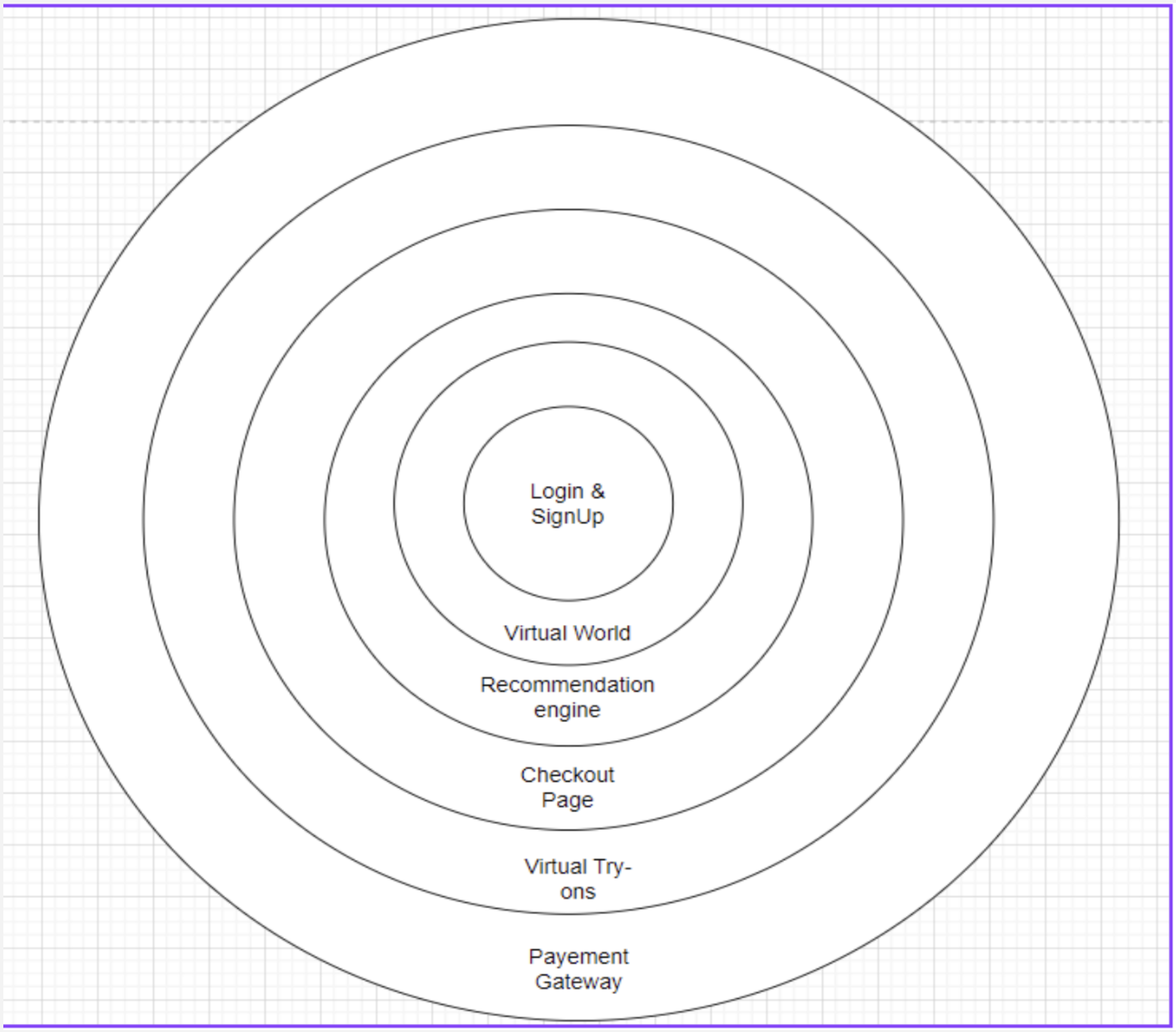
PLATFORM SUPPORT

The website should be accessible on desktop computers, laptops, tablets, and smartphones.

TESTING AND MAINTENANCE

The website should undergo rigorous testing to ensure that it meets the functional and non-functional requirements. The website should be tested using both manual and automated testing techniques. The website should be maintained regularly

Product Features



The ImmersiVR e-commerce website's product features include the following:

1. **Face recognition**: Customers will be able to log in to their accounts using facial recognition technology.
2. **AR/VR view of items**: Customers will be able to view products in an augmented or virtual reality environment.

3. **Metaverse Mall**: Customers will be able to shop in a virtual mall that allows them to interact with other customers and products in a 3D environment.
4. **Virtual try-on**: Customers will be able to virtually try on clothing and other items to see how they look before purchasing.
5. **Recommendation System**: Customers will be able to virtually try on clothing and other items to see how they look before purchasing.
6. **Design -**

5.1 Architecture Design

The ImmersiVR application will be a web-based application developed using Node.js, Flask, and React.js as the primary technologies. Three.js will be used to create the AR/VR environments for products. The application will use a MongoDB database to store user profiles and transaction details.

The application will have a client-server architecture. The client will be the user's web browser, and the server will be the Node.js application. The Flask application will handle the facial recognition feature. The React.js library will be used for the front end of the application, while the back end will be developed using Node.js.

5.2 Database Design

The ImmersiVR application will use a MongoDB database to store user profiles, product information, and transaction details. The database will be hosted on a secured web server. The database will have the following collections:

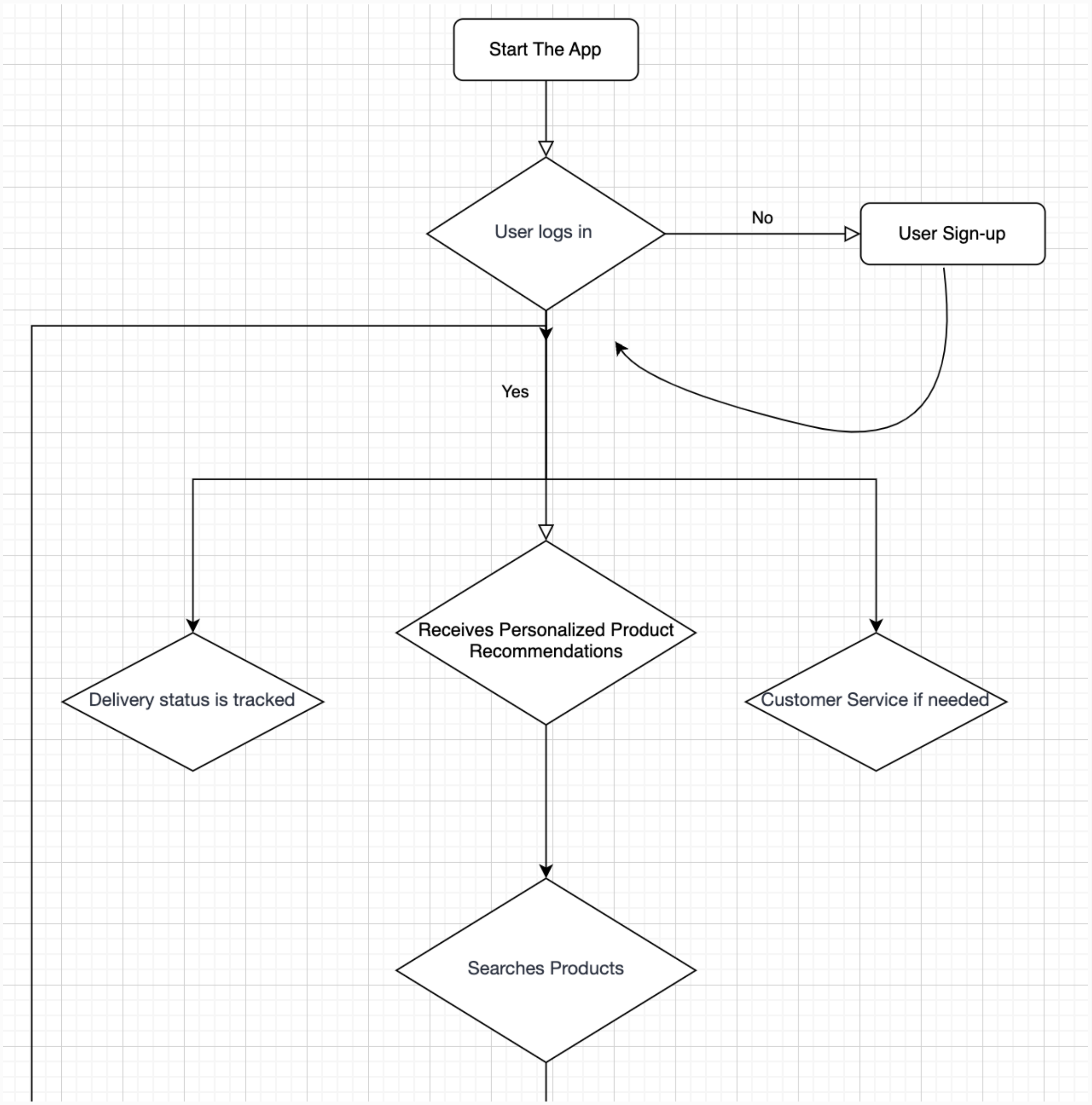
- User collection: This collection will store user information such as name, email, password, and profile picture.
- Product collection: This collection will store product information such as name, description, price, and image.
- Transaction collection: This collection will store transaction details such as order number, product name, quantity, and total price.

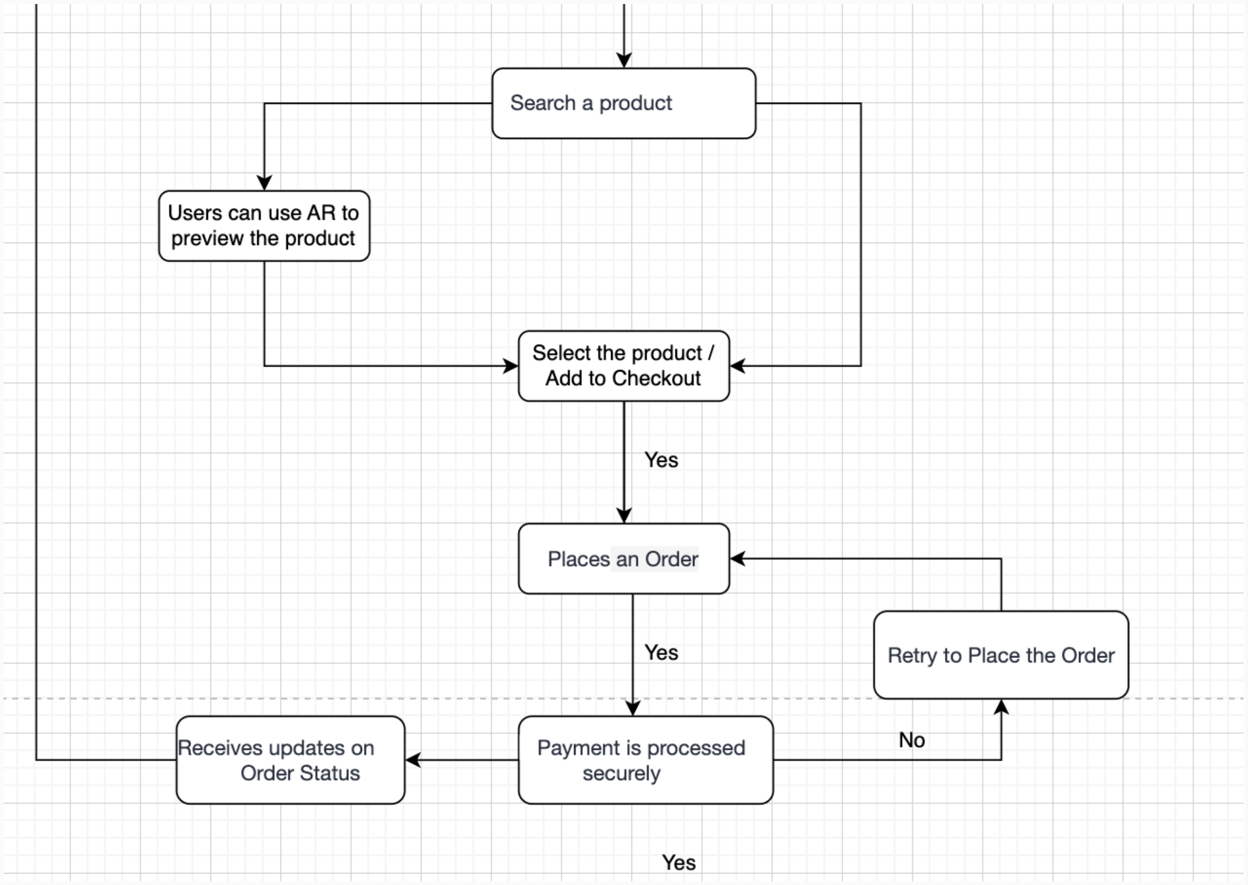
5.3 UI/UX Design

The UI/UX design of the ImmersiVR application will be focused on providing a seamless and immersive experience to the users. The front end will be developed using React.js, which will provide a responsive and user-friendly interface. The UI/UX design will include the following features:

1. Facial recognition login: The application will use facial recognition technology to log in users into their accounts.
2. AR/VR view of products: The application will allow users to view products in an augmented or virtual reality environment.
3. Metaverse Mall: The application will provide users with a virtual mall where they can interact with other users and products.
4. Virtual try-on: The application will allow users to virtually try on clothing and other items.

WORKFLOW DIAGRAM:


Start The App
User logs in
No
User Sign-up
Yes
Delivery status is tracked
Receives Personalized Product Recommendations
Customer Service if needed
Searches Products


Search a product
Users can use AR to preview the product
Select the product / Add to Checkout
Yes
Places an Order
Yes
Retry to Place the Order
Receives updates on Order Status
Payment is processed securely
No
Yes

IMPLEMENTATION

The implementation of the ImmersiVR application will be divided into two parts: front-end and back-end development. The front end will be developed using React.js, and the back end will be developed using Node.js.

The front-end development will involve creating UI components such as the login page, product browsing and viewing interface, checkout interface, virtual mall interface, AI assistant interface, and virtual try-on interface. The front end will also include the Three.js library to create AR/VR environments for products.

The back-end development will involve creating RESTful APIs using Node.js to handle user authentication, product browsing, purchasing, and virtual try-on. The Flask application will handle the facial recognition feature.

EVOLUTIONARY MODEL

Module 1: User Authentication and Profile Management

This module will handle user authentication and profile management. It will allow users to create an account, log in using facial recognition technology, and manage their profile details such as name, email, and password. It will also ensure that user information is securely stored in a MongoDB database.

Module 2: Product Catalog and Search

This module will handle product browsing and searching. It will provide users with a catalog of products in various categories and allow them to search for products using keywords or filters. It will also ensure that product information is securely stored in a MongoDB database.

Module 3: AR/VR View of Products

This module will allow users to view products in an augmented or virtual reality environment. It will use Three.js library to create AR/VR environments for products and ensure that the viewing experience is seamless and immersive.

Module 4: Metaverse Mall

This module will provide users with a virtual mall where they can interact with other users and products in a 3D environment. It will ensure that users can navigate the virtual mall easily and that the products are accurately displayed.

Module 5: Virtual Try-On

This module will allow users to virtually try on clothing and other items to see how they look before purchasing. It will use advanced algorithms and image recognition technology to provide an accurate virtual try-on experience.

Module 6: AI Assistant(Recommendation System)

This module will provide users with an AI assistant that can assist them in their shopping experience. It will use natural language processing and machine learning algorithms to provide personalized recommendations and assistance to users.

Module 7: Payment Gateway Integration

This module will handle payment processing and integration with a third-party payment gateway. It will ensure that all transactions are securely processed and that users can complete their purchases without any issues.

Module 8: Maintenance and Support

Handle website maintenance and ensure regular updates. Resolve any issues or bugs promptly. Provide technical support and assistance to users when needed.

How we have implemented Evolutionary Model for our Software:

In the evolutionary process model software requirements are broken into several sub-modules like M1, M2, M3……Mn. These sub-modules are designed and developed in form of release. In the first iteration module, M1 is developed and delivered.

In the second iteration module, M2 is developed and integrated with module M1.

Now, this increment M1+M2 is delivered to the customer after the second iteration. In the same way, the whole software was developed and in the end, the final software product is delivered to the customer.

The problems we faced while integrating every new feature have been mentioned ahead and the bugs were fixed before moving on to the next iteration of the SDLC.

The testing of the ImmersiVR application will be divided into two parts: unit testing and system testing. Unit testing will involve testing individual components of the application such as UI components, RESTful APIs, and facial recognition features. System testing will involve testing the entire application as a system. We have currently manually done testing.

CONCLUSION

The ImmersiVR application is a futuristic e-commerce application that combines AR/VR technologies with a metaverse mall, virtual try-on, and facial recognition technology. The application will provide a seamless and immersive experience to the users. The application will be developed using Node.js, Flask, and React.js technologies and hosted on a MongoDB database. The application will be tested using Jest, Enzyme, Mocha, and Chai libraries.