Hamdard University
Department of Computing
Final Year Project

SNEAKER VISION
FYP-034/FL24
Software Design Specifications

Submitted by

Muhammad Rizwan
BSCS/F21/2122

Abdul Basit Abbasi
BSCS/F21/2069

Minahil Khan
BSCS/F21/2587

SUPERVISOR
Mr. Waqas Pasha

CO-SUPERVISOR
Mr. Maaz Ahmed

Fall 2024

Document Sign off Sheet
Document Information

Project Title	Sneaker Vision
Project Code	FYP-034/FL24
Document Name	Software Design Specifications
Document Version	<1.0>
Document Identifier	FYP-034/FL24 SRS
Document Status	Final
Author(s)	M.Rizwan, Abdul Basit Abbasi, Minahil Khan
Approver(s)	Mr. Waqas Pasha
Issue Date	10/01/2025

Name	Role	Signature	Date
Abdul Basit Abbasi	Team Lead		
Muhammad Rizwan	Team Member 2		
Minahil Khan	Team Member 3		
Mr. Waqas Pasha	Supervisor		
Mr. Maaz Ahmed	Co-Supervisor		
Mr. Mohsin Raza	Project Coordinator		

Revision History

Date	Version	Description	Author
	1.0		

Definition of Terms, Acronyms, and Abbreviations

Term	Description
App	Application
ARKit	Augmented Reality Kit
AR Core	Augmented Reality Core
PCI DSS	Payment Card Industry Data Security Standard.
GDPR	General Data Protection Regulation
CCPA	California Consumer Privacy Act

Table of Contents

1 Introduction

Sneakers Vision, the cutting-edge online marketplace that is revolutionizing the way sneaker heads buy, sell and interact. Imagine a platform where you can easily find your dream sneakers, communicate with a thriving community of like-minded people, and participate in frictionless bidding procedures. This concept has become a reality because to Sneakers concept. Our objective is to build a vibrant community built on uniqueness, self-expression, and a common love of shoes. Sneakers Vision is more than just an online store; it's a community where you can comfortably bid, buy, sell, and network with other sneaker heads. Sneakers Vision prioritizes our community's requirements and goals, providing a user-friendly experience for both seasoned collectors and newbies. Our platform includes comprehensive search capabilities, real-time bidding, and safe transactions, making it easier than ever to locate and purchase the sneakers you've always desired. What distinguishes us is our dedication to building authentic interactions among sneaker heads. Our interactive features allow you to participate in discussions, share your collections, and learn from industry professionals and other collectors. Sneakers Vision is more than simply a marketplace; it's a flourishing environment that fosters sneaker culture. Join the Sneakers Vision family now and experience the ultimate sneaker destination. Together, we shall confidently and stylishly enter the future of sneaker culture. Accept the transformation of the sneaker market and join a community that values originality and a shared passion for the ideal pair of sneakers.

1.1 Purpose of Document

The purpose of this document is to provide a comprehensive Software Design Specification (SDS) for the Sneaker Vision project. It outlines the design and architecture of the software, including its functionalities, system requirements, and design methodology.

1.2 Intended Audience

The intended audience for this document includes the stakeholders involved in the development, design, and implementation of the SneakerVision application. This includes the project team, product managers, developers, UI/UX designers, quality assurance teams, and business analysts. Additionally, it is relevant for the marketing team responsible for promoting the app and customer support representatives who will assist users. Finally, the document is also for potential investors and partners interested in understanding the functionality and vision of SneakerVision.

1.3 Document Convention

This document will be using 10 font size and Arial font for paragraphs and 14 font size and Arial font for headers

1.4 Project Overview

SneakerVision is an innovative online marketplace designed to transform the way sneaker enthusiasts buy, sell, and interact. It addresses common issues faced by online shoppers, such as sizing concerns and

discrepancies between product images and actual products, by integrating cutting-edge augmented reality technology. The platform offers features like virtual try-ons, a comprehensive catalog of sneakers, frictionless bidding procedures, and an engaging community space for users. By combining technology with user-centric design, SneakerVision ensures a seamless, accurate, and enjoyable shopping experience while saving users time and effort.

1.5 Scope

1.5.1 IN SCOPE

- User registration and login.
- Browse and search for sneakers.
- Catalog of available shoes with product details and images
- Integration of augmented reality technology to allow users to try on shoes virtually.
- Add sneakers to a cart and purchase them through the app.
- Payment Gateway Integration.
- User feedback and ratings system.
- The admin should be able to manage users' accounts, including creating new accounts, updating user information, and deleting user accounts.

1.5.2 OUT SCOPE

- Developing an e-commerce platform that supports selling other products besides sneakers.
- Shoe cleaning, repair, or maintenance services.
- User cannot customize or design their shoes.

2 Design Considerations

The design of SneakerVision prioritizes user experience, accessibility, and functionality. The application will feature an intuitive interface for easy navigation and usability across all devices. The augmented reality feature will be optimized to provide accurate and realistic visualizations of how sneakers look and fit. Secure payment gateway integration will ensure safe transactions, and user feedback mechanisms will allow continuous improvement of the app. Scalability will be a key consideration to handle the growing user base, and data privacy will be prioritized to build user trust.

1. **User Experience (UX):**

- **Simple Navigation:** Users will be able to easily browse, search, and filter sneakers through an organized interface that minimizes complexity.
- **Clean Layout:** A minimalist design will reduce clutter and make it easy for users to focus on product details.
- **Consistency:** Uniform design elements, such as buttons, icons, and fonts, will provide a cohesive visual experience across all devices.
- **Responsive Design:** The platform will be optimized for mobile devices, tablets, and desktops, ensuring a consistent experience regardless of the device being used.

2. **Accessibility:**

- **Screen Reader Support:** The app will include alt text for images and AR visualizations to aid visually impaired users.
- **Keyboard Navigation:** Full functionality will be accessible through keyboard input for users who may not use a mouse or touch controls.
- **Contrast and Readability:** High contrast and legible fonts will improve readability for users with visual impairments.

3. **Augmented Reality Feature:**

- **Provide Accurate Visualizations:** The AR technology will be optimized to display sneakers in real-world proportions, ensuring users can see how the shoes will look and fit.
- **Compatibility:** The feature will be built using **ARKit (iOS)** and **ARCore (Android)** for smooth performance across major platforms.
- **Real-Time Feedback:** Users will see instant results while trying on sneakers virtually, creating an immersive experience.

4. **Secure Payment Gateway:**

- **Multiple Payment Methods:** Users can choose from credit/debit cards, digital wallets, and bank transfers.
- **Encryption:** End-to-end encryption will secure sensitive payment details.
- **Fraud Prevention:** Measures such as two-factor authentication and real-time transaction monitoring will protect users from fraud.

5. **User Feedback Mechanism:**

- **Product Ratings:** Users can rate products on a scale and write detailed reviews about their purchase experience.

- **Platform Feedback:** Suggestions for platform improvement will be collected through surveys or dedicated feedback sections.
- **Community Insights:** Engaging with user reviews will help build trust and credibility within the community.

6. Scalability:

- **Cloud-Based Infrastructure:** Leveraging cloud services will ensure flexibility to handle large volumes of users and data.
- **Microservices Architecture:** Independent components will allow for easy scaling of specific features, such as AR or payment processing, without affecting the entire platform.
- **Load Balancing:** Traffic management tools will distribute user requests efficiently to prevent downtime or slow performance.

7. Data Privacy and Security:

- **Compliance with Regulations:** The platform will comply with GDPR, CCPA, and PCI DSS standards to ensure user data is handled responsibly.
- **Data Encryption:** All user data, including personal and payment information, will be encrypted both at rest and in transit.
- **Access Control:** Role-based access will limit sensitive data access to authorized personnel only.

2.1 Assumptions and Dependencies

It is assumed that users have access to a smartphone or device compatible with augmented reality technology to fully utilize the virtual try-on feature. The app depends on reliable internet connectivity for seamless browsing, AR functionality, and transactions. It also requires partnerships with sneaker retailers and manufacturers to maintain a comprehensive and updated catalog. Payment gateway integration depends on third-party services, and compliance with legal standards for data protection and e-commerce regulations is necessary for smooth operations.

2.1.1 User Accessibility and Device Compatibility:

- **Assumption:** Users will have access to a smartphone or device compatible with augmented reality (AR) technology.
    - o The platform assumes that a significant portion of its target audience uses modern smartphones that support AR features via ARKit (iOS) or ARCore (Android).
    - o Users must also have devices with adequate hardware specifications, such as high-quality cameras and sufficient processing power, to ensure a smooth AR experience.
- **Impact:** The lack of a compatible device may limit the ability of some users to fully utilize the virtual try-on feature, although other functionalities like browsing and purchasing will still be accessible.

2.1.2 Internet Connectivity:

- **Assumption:** Reliable internet connectivity is necessary for the platform to function effectively.
    - o High-speed internet is required for seamless browsing of sneaker catalogs, AR functionality, and secure transactions.
    - o Features like virtual try-on, real-time pricing updates, and user reviews depend heavily on stable network connections.
- **Impact:** Users in areas with poor or inconsistent internet connectivity may face delays or disruptions, impacting their experience with the app.

2.1.3 Partnerships with Retailers and Manufacturers:

- **Assumption:** The app depends on partnerships with sneaker retailers and manufacturers to maintain a comprehensive and updated product catalog.
    - o Retailers and brands must provide timely updates on product availability, pricing, and inventory levels to ensure accuracy.
    - o Collaboration with trusted suppliers ensures access to exclusive and limited-edition sneakers, which are critical to attracting the target audience.
- **Impact:** The absence of strong partnerships could result in a limited product catalog, reducing the platform's appeal to sneaker enthusiasts.

2.1.4 Payment Gateway Integration:

- **Assumption:** Secure and efficient payment processing is dependent on third-party payment gateway services.
    - o Integration with reputable payment gateways will enable various payment methods, including credit/debit cards and digital wallets.
    - o The platform assumes these services will operate reliably, ensuring smooth transactions without errors or delays.
- **Impact:** Issues with payment gateway providers, such as downtime or security breaches, could disrupt transactions and affect user trust.

2.2 Risks and Volatile Areas

Several risks and volatile areas may impact the project. The integration of augmented reality technology could face challenges related to device compatibility and user adoption. Ensuring accurate virtual try-on representations may be difficult due to variations in user device configurations and lighting conditions. Payment gateway security breaches could harm user trust, while potential delays in catalog updates from retail partners could impact user satisfaction. Lastly, competition in the online sneaker marketplace could influence user retention and market share, requiring continuous innovation and marketing efforts.

2.2.1 Integration of Augmented Reality Technology:

- **Risk:** Challenges may arise during the integration of AR technology, particularly in ensuring compatibility with a wide range of devices and operating systems.
    - o **Device Compatibility:** Not all users may have smartphones or devices that support ARKit (iOS) or ARCore (Android). This could limit the reach and utility of the virtual try-on feature.
    - o **User Adoption:** Some users may be unfamiliar or uncomfortable with AR technology, leading to lower adoption rates for this innovative feature.
    - o **Impact:** Limited adoption or technical difficulties could reduce the effectiveness of the platform's primary differentiator and diminish user satisfaction.

2.2.2 Accurate Virtual Try-On Representations:

- **Risk:** Ensuring that the AR-based virtual try-on feature provides accurate and realistic visualizations may prove challenging.
    - o **Device Configuration Variability:** Differences in camera quality, screen resolution, and processing power across user devices could affect the accuracy of AR projections.
    - o **Lighting Conditions:** Poor or inconsistent lighting environments might interfere with AR functionality, leading to less accurate representations of sneakers.
    - o **Impact:** Inaccurate representations could lead to user dissatisfaction, increased returns, and negative reviews, ultimately harming the platform's reputation.

2.2.3 Payment Gateway Security Breaches:

- **Risk:** Any compromise of the payment gateway could lead to unauthorized access to sensitive user payment data.
  - ○ **Security Vulnerabilities:** Cyberattacks or lapses in encryption protocols could expose user information and financial details.
  - ○ **Impact:** A security breach would severely damage user trust, potentially resulting in user attrition, reputational damage, and legal liabilities.

2.2.4 Delays in Catalog Updates from Retail Partners:

- **Risk:** The app's reliance on retail partners to provide timely updates on product availability, pricing, and inventory levels introduces a dependency.
  - ○ **Data Sync Issues:** Delays or errors in catalog updates could lead to discrepancies between what users see on the platform and actual stock availability.
  - ○ **Impact:** Such inconsistencies could frustrate users, leading to abandoned purchases, reduced satisfaction, and damage to the platform's credibility.

2.2.5 Competition in the Online Sneaker Marketplace:

- **Risk:** The highly competitive nature of the sneaker resale and e-commerce market poses a significant challenge to user retention and market share.
  - ○ **Established Competitors:** Larger, well-established platforms may attract users with their existing reputation, extensive catalogs, and promotional deals.
  - ○ **User Retention:** Without continuous innovation and effective marketing efforts, users may shift to competitors.
  - ○ **Impact:** Failure to differentiate the platform through unique features, user-centric design, or exclusive partnerships could limit growth and profitability.

3   System Architecture

The system architecture for SneakerVision is designed to provide a seamless, efficient, and scalable online marketplace experience. It is divided into several layers, each with specific responsibilities to ensure smooth functionality and interaction between components. Below is an outline of the architecture:

1. **Client Layer (Frontend)**
- Components: Mobile application (iOS and Android), Web interface.
- Technology: React Native for mobile apps, React.js or similar framework for web.
- Responsibilities:
    - Provides an intuitive user interface for registration, browsing sneakers, and virtual try-on functionality.
    - Enables users to add sneakers to the cart, manage profiles, and interact with the community.
    - Communicates with backend services via APIs for data retrieval and submission.
    - Implements the augmented reality (AR) feature using SDKs like ARKit (iOS) and ARCore (Android).

2. **Application Layer (Backend)**
- Components: Server-side application hosted on a cloud platform.
- Technology: Python, Dart.
- Responsibilities:
    - Manages business logic, including user authentication, profile management, and AR processing.
    - Handles requests from the frontend and retrieves or updates data in the database.
    - Ensures smooth functionality for virtual try-on, bidding procedures, and user feedback.
    - Provides APIs for frontend communication.

3. **Database Layer**
- Components: Relational database Firebase.
- Responsibilities:
    - Stores user data, including login credentials, profiles, and transaction history.
    - Maintains sneaker catalog details, including product images, descriptions, and AR models.
    - Logs user interactions and feedback for analysis and future improvements.

4. **Augmented Reality (AR) Layer**
- Components: AR SDKs (e.g., ARKit, ARCore) and 3D model storage.
- Responsibilities:
    - Processes and renders AR visualizations for virtual try-ons.
    - Ensures accurate scaling and alignment of 3D sneaker models with user foot dimensions.
    - Updates AR models based on catalog changes or new product additions.

5. **Payment Gateway Integration**
- Components: Secure third-party payment services (e.g., Stripe, PayPal).
- Responsibilities:
    - Facilitates secure transactions for sneaker purchases.
    - Ensures compliance with PCI DSS standards for payment data protection.
    - Supports multiple payment methods, including credit/debit cards and digital wallets.
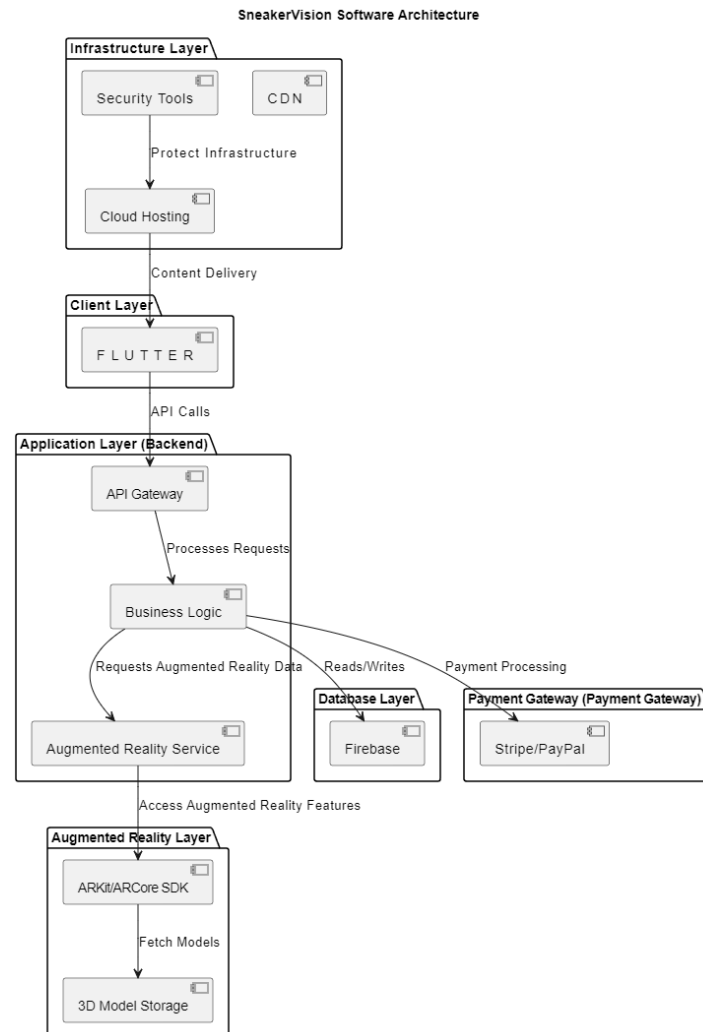
3.1 System Level Architecture
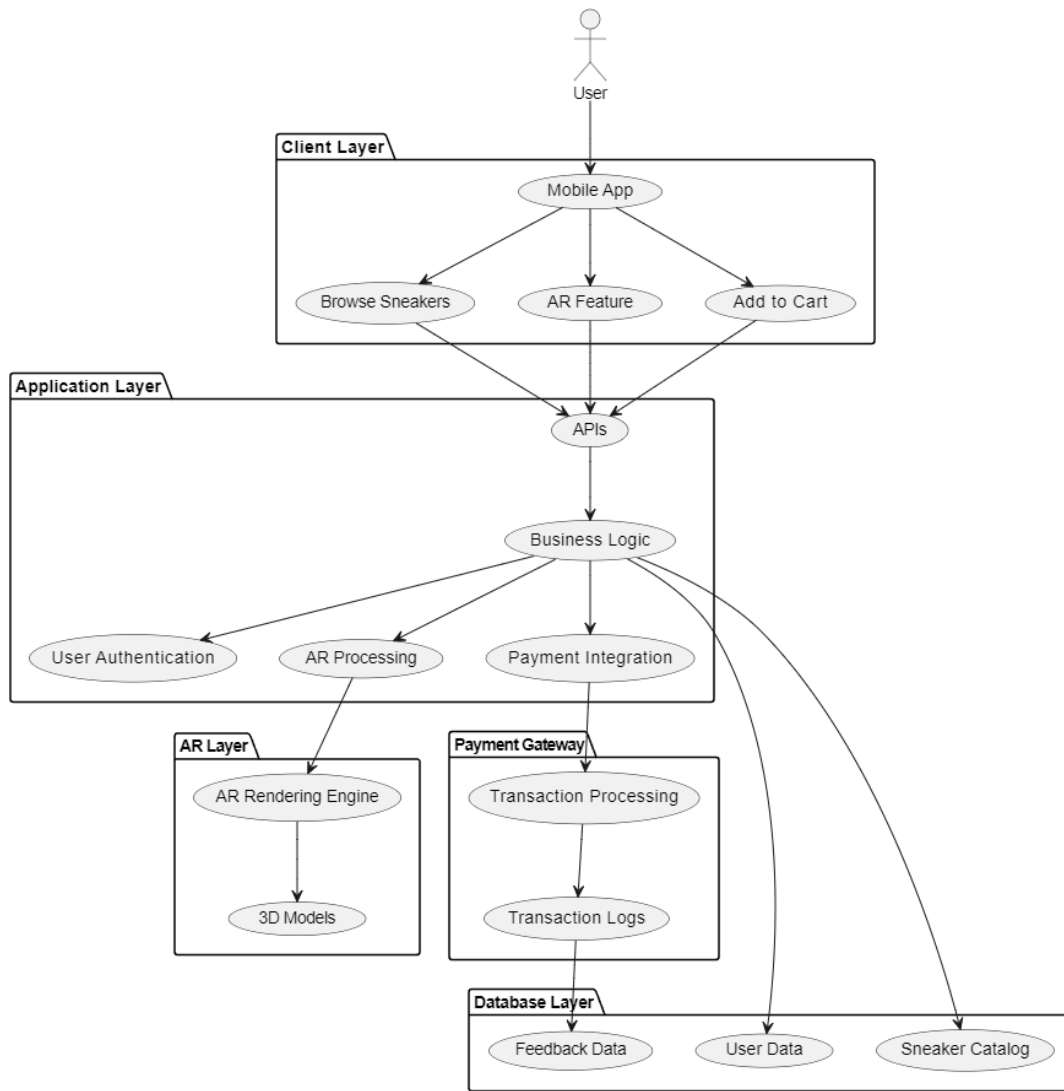
Figure 1: System Level Architecture

1. **Client Layer**: Users access the system via the web interface or mobile app, which allows browsing sneakers, utilizing AR functionality, and adding items to the cart.
2. **Application Layer**: This layer includes APIs for communication between the client and backend, and handles core functionalities like user authentication, AR processing, and payment integration.
3. **AR Layer**: Features an AR rendering engine for displaying 3D sneaker models and managing augmented reality experiences.
4. **Payment Gateway**: Ensures secure transactions with features like transaction processing and maintaining transaction logs.
5. *.* **Database Layer**: Stores user profiles, sneaker catalog details, and feedback data, while supporting data flow and processing through the business logic.

3.2 Software Architecture

Figure 2: System Level Architecture

1. **Infrastructure Layer**: Secures and optimizes content delivery with security tools, CDN, and cloud hosting.
2. **Client Layer**: Uses Flutter for cross-platform app development and API calls to interact with the backend.
3. **Application Layer**: Manages requests via the API Gateway, processes logic, and handles AR tasks.
4. **Database Layer**: Powered by Firebase to store and manage user data, product catalogs, and feedback.
5. **Payment Gateway**: Ensures secure transactions through Stripe/PayPal, integrated with business logic.
6. **Augmented Reality Layer**: Leverages ARKit/ARCore SDK for AR rendering and 3D model storage for AR experiences.

4 Design Strategy

1. Future System Extension or Enhancement
The SneakerVision system is designed with scalability and extensibility in mind, allowing for future enhancements to meet evolving user needs and market trends. Potential extensions include integrating a machine learning recommendation engine to suggest sneakers based on user preferences and behavior, expanding the AR feature to include more advanced customization options like color changes, and enabling community-driven features such as live auctions or collaborative collections. Additionally, partnerships with third-party retailers could bring an even broader range of sneakers to the platform.

2. System Reuse
SneakerVision's modular architecture allows for high reusability of system components. The API Gateway, AR Service, and Payment Gateway modules are designed to be repurposed for similar applications, such as online marketplaces for other apparel or accessories. Similarly, the user authentication system and feedback mechanisms can be reused in future projects requiring robust and secure user management. The backend and database configurations can also serve as a foundation for developing other e-commerce platforms.

3. User Interface Paradigms
The user interface follows a modern, responsive design paradigm, ensuring seamless usability across devices, including smartphones, tablets, and desktops. The interface employs a minimalist, user-centric approach, with intuitive navigation and interactive elements to enhance engagement. Augmented reality is integrated as a core feature, providing a highly immersive experience for users. The platform also supports adaptive interfaces for accessibility, catering to users with different needs and ensuring inclusivity.

4. Data Management (Storage, Distribution, Persistence)
Distributed storage ensures high availability and performance, leveraging cloud services to handle large-scale data efficiently. Data persistence is managed through regular backups and replication, ensuring reliability and disaster recovery. The system complies with data privacy laws, including GDPR and CCPA, to secure user information.

5. Concurrency and Synchronization
SneakerVision handles concurrent operations efficiently by employing asynchronous programming and multi-threading techniques. This ensures that multiple users can browse, interact, and transact simultaneously without performance degradation. Synchronization mechanisms, such as database locks and message queues, are implemented to maintain data consistency during high-volume operations, such as simultaneous bidding or purchase activities. Load balancers and distributed systems further optimize concurrency, ensuring seamless scalability and user satisfaction even during peak traffic.

• **Detailed System Design**

4.1 Design Class Diagram
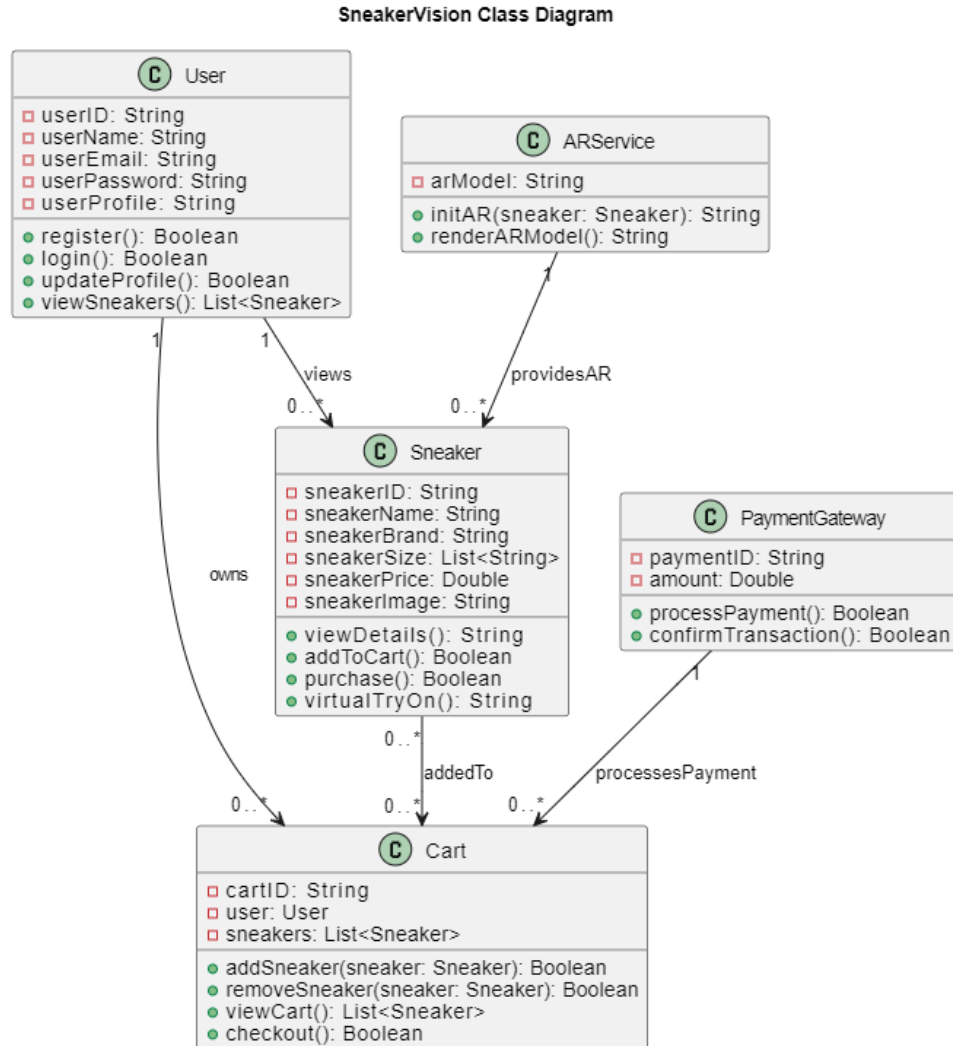
Figure 3: Design Class Diagram

1. **User Class**: Represents users with attributes like userID, userName, and provides methods for register(), login(), and viewing sneakers.
2. **Sneaker Class**: Defines sneaker details (e.g., sneakerID, sneakerName, sneakerBrand) and supports operations like viewDetails(), addToCart(), and virtualTryOn().
3. **Cart Class**: Manages the cart with cartID and sneakers list, and methods to addSneaker(), removeSneaker(), viewCart(), and checkout().
4. **ARService Class**: Handles augmented reality functions through methods initAR() and renderARModel().
5. **PaymentGateway Class**: Manages payments with attributes like paymentID and methods to processPayment() and confirmTransaction().
6. *Relationships*: Users own carts, carts contain sneakers, sneakers interact with AR services, and payment gateways handle cart transactions.
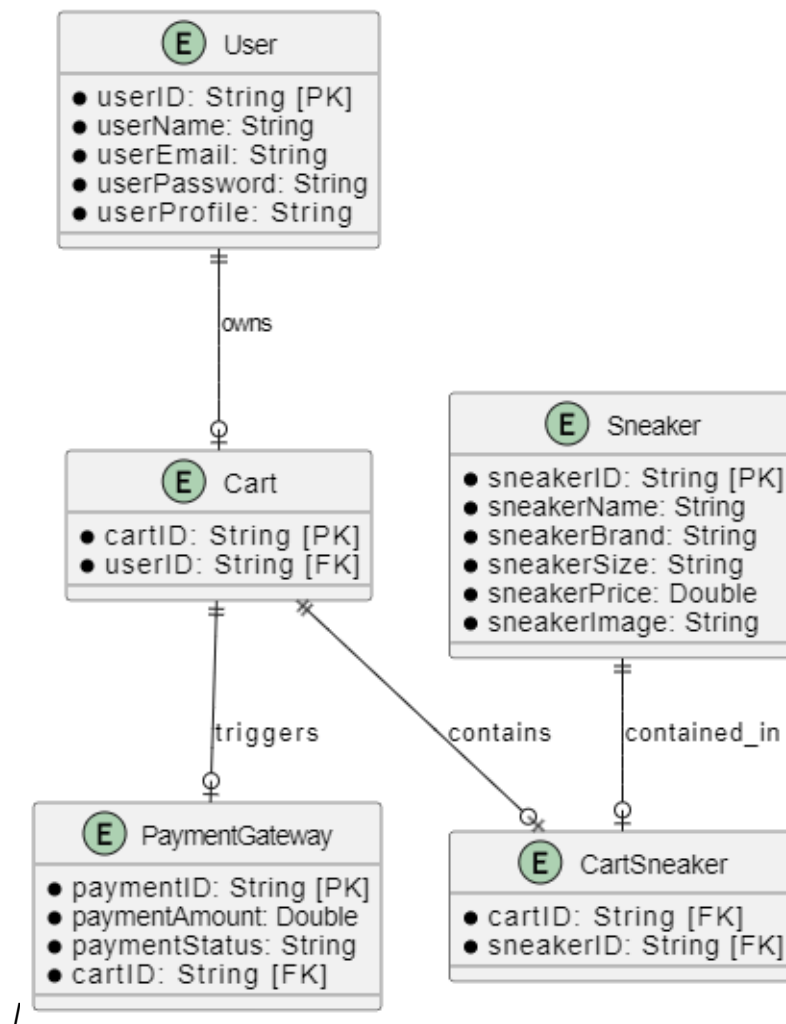
4.2 Database Design

Figure 4: Database Design

1. **User Entity**: Contains user details (userID, userName, userEmail) with userID as the primary key (PK).
2. **Cart Entity**: Represents shopping carts with cartID as PK and userID as a foreign key (FK) linking it to the User entity.
3. **Sneaker Entity**: Stores sneaker details (sneakerID, sneakerName, sneakerBrand) with sneakerID as PK.
4. **CartSneaker Entity**: A junction table mapping sneakers to carts, with cartID and sneakerID as composite FK.
5. **PaymentGateway Entity**: Manages payments (paymentID, paymentAmount, paymentStatus), with cartID as FK linking it to the Cart entity.
6. **Relationships**: Users own carts, carts contain sneakers, CartSneaker connects sneakers to carts, and payments are triggered by carts.
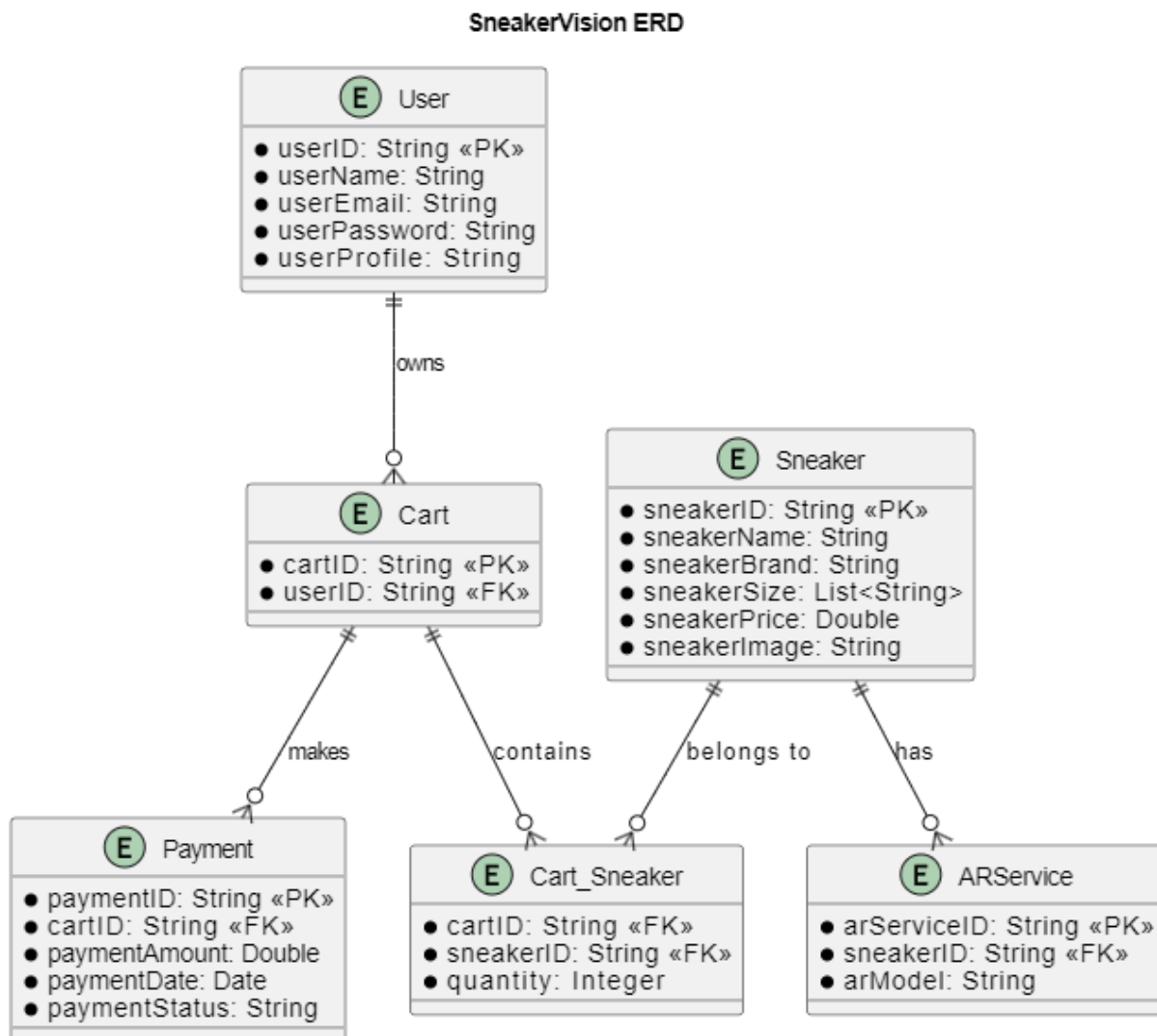
1. ER Diagram

Figure 5: ER Diagram

1. **User Entity**: Includes attributes (userID, userName, userEmail, userPassword, userProfile), with userID as the primary key (PK).
2. **Sneaker Entity**: Includes attributes (sneakerID, sneakerName, sneakerBrand, sneakerSize, sneakerPrice, sneakerImage), with sneakerID as the primary key (PK).
3. **Cart Entity**: Includes attributes (cartID, userID), where userID is a foreign key (FK) referencing the User entity; cartID is the primary key (PK).
4. **Cart_Sneaker Entity**: A junction table with attributes (cartID, sneakerID, quantity), handling the many-to-many relationship between Cart and Sneaker.
5. **ARService Entity**: Includes attributes (arServiceID, sneakerID, arModel), where sneakerID is a foreign key (FK) referencing the Sneaker entity.
6. **Payment Entity**: Includes attributes (paymentID, cartID, paymentAmount, paymentDate, paymentStatus), where cartID is a foreign key (FK) referencing the Cart entity.

- **Application Design**

1. Sequence Diagram

Figure 6: Sequence Diagram

1. **Browsing Sneakers**: The app requests the list of available sneakers from the API Gateway, which retrieves the data from the Database and sends it back to the app.
2. **Viewing Sneaker Details**: When a sneaker is selected, the app requests detailed information from the API Gateway, which fetches the data from the Database and returns it to the app.
3. **Virtual Try-On**: Upon selecting "Virtual Try-On," the app requests the AR model from the AR Service, which retrieves the model data from the Database and renders it in the app.
4. **Adding Sneaker to Cart**: The app sends a request to the API Gateway to update the user's cart. The API updates the Database and confirms the addition.
5. **Making Payment**: At checkout, the app interacts with the Payment Gateway to validate payment via the API Gateway, which confirms the transaction with the Database and completes the payment process.
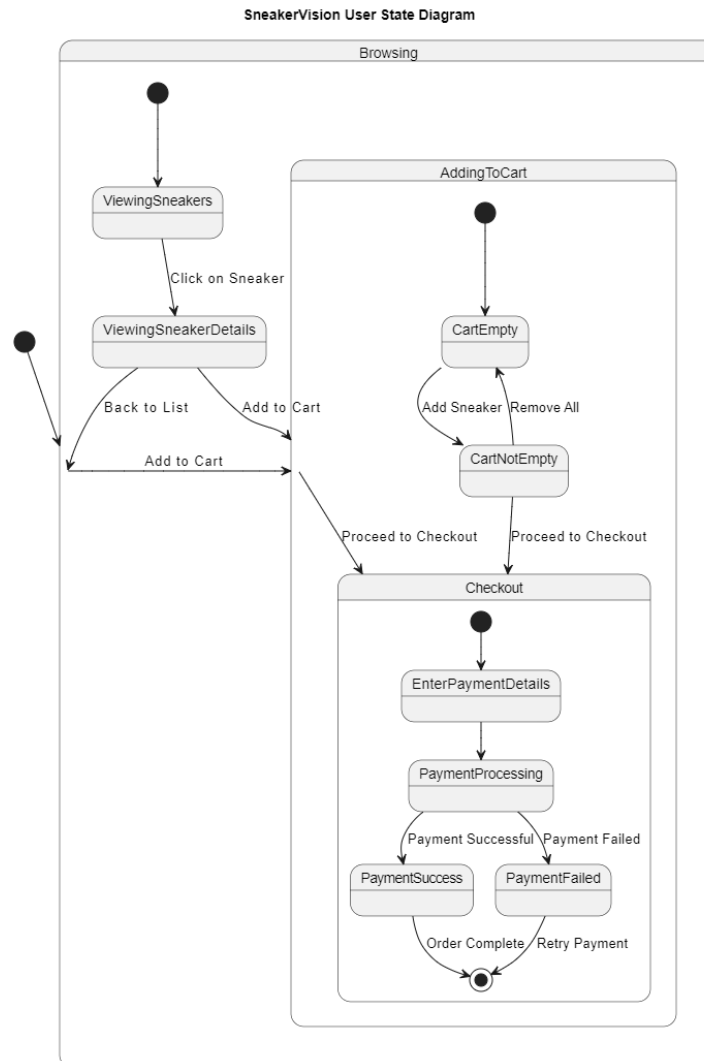
2. State Diagram

Figure 7: State Diagram

1. **Browsing Sneakers**: The user starts by exploring the platform's available sneakers, with options to view details or add items to the cart.
2. **Viewing Sneakers**: The user views a list of sneakers and can either see more details of a specific sneaker or add one to the cart.
3. **Viewing Sneaker Details**: The user accesses detailed information about a selected sneaker and can return to browsing or add it to the cart.
4. **Adding to Cart**: The user adds sneakers to their cart, transitioning the state to CartNotEmpty. Removing all items moves the state back to CartEmpty, or the user can proceed to checkout.
5. **Checkout Process**:
   o Enter Payment Details: The user provides payment information.
   o Payment Processing: The payment is processed.
   o Payment Success: The order is completed.
   o Payment Failed: The user can retry the payment.

4.3 GUI Design
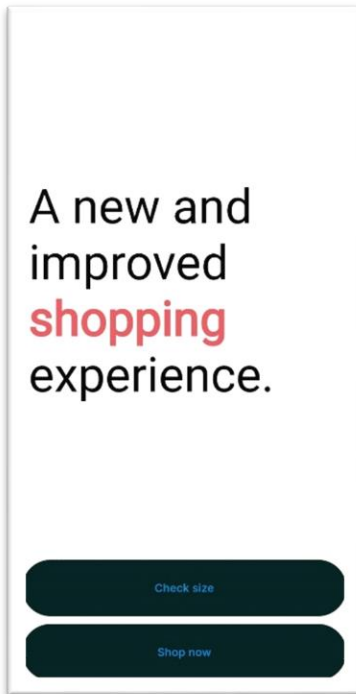

Purpose: The Home Page serves as the initial interface for users and offers two primary options:

- Check Size: Allows the user to check or confirm their shoe size before shopping. This can be integrated with augmented reality or a size guide feature to help users find the perfect fit.
- Shop Now: Directs the user to the catalog of available sneakers where they can browse and select items to add to their cart.

User Experience:
- Upon entering the app, users will be presented with these two options on the homepage. By selecting either option, the user moves to the next step, whether they are looking to confirm their size or begin browsing products.
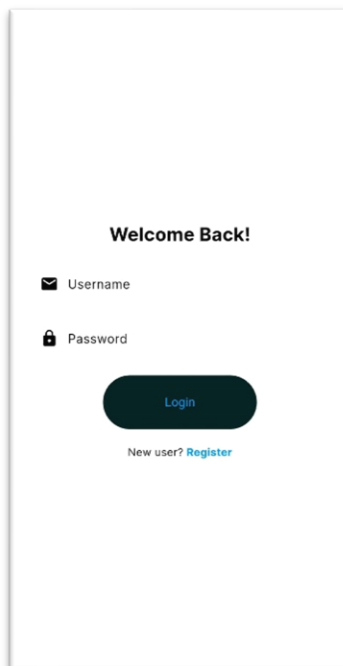
Figure 8: Main Screen

Purpose: This page enables users to either sign in or create a new account to access the platform and personalize their experience.
- Login: Existing users enter their credentials (username/email and password) to log in and continue shopping.
- Signup: New users can create an account by providing necessary details like name, email, and password. Upon signing up, users may have the option to input their preferences for a more customized shopping experience.

• **User Experience:** Users will be prompted to log in or sign up if they are not already authenticated. For those already logged in, the app can bypass this step and take them straight to the catalog or home page.

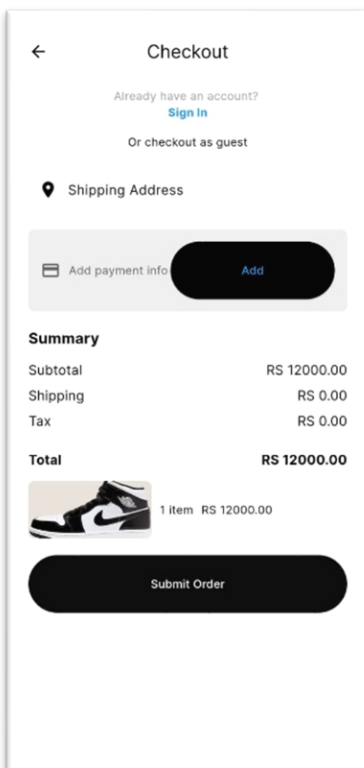**Figure 9: Login/Signup Screen**

Figure 10: Catalog Screen
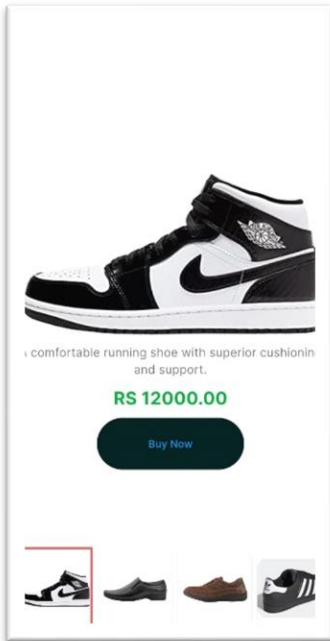
Entering the Catalog:
Users land on the catalog page directly or after selecting "Shop Now" from the home page.

Browsing Sneaker Listings:
Users browse through the list of sneakers, clicking on items that catch their interest for more details. They can either add items directly to their cart or navigate to the product details page for more information.

Figure 11: Checkout Screen

Review Order:
The user reviews the sneakers they've selected, including quantities, sizes, and prices.

Enter Shipping Details:
The user fills in their shipping information or selects a previously saved address.

Select Delivery Method:
The user chooses between standard or express delivery.

Proceed to Payment:
Once all details are confirmed, the user clicks "Proceed to Payment" to finalize the checkout process.

Enter Payment Details:
The user selects a payment method and inputs their billing information and payment details.

Verify Payment:
If 3D Secure or two-factor authentication is enabled, the user completes the verification process.

Payment Confirmation:
A confirmation screen shows the order ID, estimated delivery date, and other relevant details.

Figure 12: Payment Gateway Screen

Survey Questions:
The survey will include a combination of rating scales, multiple-choice questions, and open-ended text fields.

Submission Confirmation:
After completing the survey, users will see a thank-you message, such as: Thank you for your feedback! Your input helps us improve and provide a better experience for you.

Figure 12: Survey Screen

References

- Caboni, F., & Lucia Pizzichini . (2022, June 28). *Emerald Insight* . Retrieved from Emerald: https://www.emerald.com/insight/content/doi/10.1108/IJRDM-10-2021-0509/full/html
- Chen, S.-C., Tung-Hsiang Chou, Tanaporn Hongsuchon, Athapol Ruangkanjanases, Santhaya Kittikowit , & Tse-Ching Lee. (7, Feb 2022). *Emerald Insight* . Retrieved from Emerald : https://www.emerald.com/insight/search?q=Tung-Hsiang%20Chou
- Wang, C.-H., Yi-Chen Chiang, & Mao-Jiun Wang. (2015, October 23). *Evaluation of an Augmented Reality Embedded On-line Shopping System*. Retrieved from Science Direct: https://www.sciencedirect.com/science/article/pii/S2351978915007672
- Yoo, J. (2023, Feb 15). *The effects of augmented reality on consumer responses in mobile shopping: The moderating role of task complexity*. Retrieved from Natutal Libray of Medicine: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9981924/