Contents

[1. Introduction 4](#_Toc152352953)

[1.1 Purpose 4](#_Toc152352954)

[1.2 Document Conventions 4](#_Toc152352955)

[1.3 Project Scope 5](#_Toc152352956)

[1.3.1 Scope Definition 5](#_Toc152352957)

[1.3.2 Core Features 5](#_Toc152352958)

[1.3.3 Subsequent Releases 6](#_Toc152352959)

[1.3.4 Alignment with User and Business Goals 6](#_Toc152352960)

[1.4 References 6](#_Toc152352961)

[2. Over all Description 7](#_Toc152352962)

[2.1 Product Perspective 7](#_Toc152352963)

[2.1.1 Product Context 7](#_Toc152352964)

[2.1.2 Product Origin 7](#_Toc152352965)

[2.1.3 Product Relationship to Existing Systems 7](#_Toc152352966)

[2.1.4 Product Ecosystem 7](#_Toc152352967)

[2.1.5 Visual Representation 8](#_Toc152352968)

[2.2 User Classes and Characteristics 8](#_Toc152352969)

[2.2.1 Tech Enthusiasts 8](#_Toc152352970)

[2.2.2 Casual Shoppers 8](#_Toc152352971)

[2.2.3. Favored User Class 9](#_Toc152352972)

[2.2.4. Alignment with User Needs 9](#_Toc152352973)

[2.3 Operating Environment 9](#_Toc152352974)

[2.3.1 Hardware Platform 9](#_Toc152352975)

[2.3.2 Operating Systems and Versions 10](#_Toc152352976)

[2.3.3 Geographical Locations 10](#_Toc152352977)

[2.3.4 Organizations Hosting Infrastructure 10](#_Toc152352978)

[2.3.5 Coexisting Software Components 10](#_Toc152352979)

[2.3.6 Infrastructure Requirements Specification 10](#_Toc152352980)

[2.4 Design and Implementation Constraints 11](#_Toc152352981)

[2.4.1 Programming Language and Framework 11](#_Toc152352982)

[2.4.2 Database Technology 11](#_Toc152352983)

[2.4.3 Cloud Infrastructure 11](#_Toc152352984)

[2.4.4 Third-Party Integrations 11](#_Toc152352985)

[2.4.5 User Interface Design 12](#_Toc152352986)

[2.4.6 Performance Optimization 12](#_Toc152352987)

[2.5 Assumptions and Dependencies 12](#_Toc152352988)

[2.5.1 Assumptions 12](#_Toc152352989)

[2.5.2 Dependencies 13](#_Toc152352990)

[3. System Features 13](#_Toc152352991)

[3.1 Product Catalog 13](#_Toc152352992)

[3.1.1 Description 13](#_Toc152352993)

[3.1.2 Functional Requirements 14](#_Toc152352994)

[4. Data Requirements 14](#_Toc152352995)

[4.1 Logical data model 14](#_Toc152352996)

[4.2 Data dictionary 14](#_Toc152352997)

[4.3 Report 14](#_Toc152352998)

[4.4 Data acquisition, integrity, retention, and disposal 14](#_Toc152352999)

[5. External interface requirements 14](#_Toc152353000)

[5.1 User Interfaces 14](#_Toc152353001)

[5.1.1 Design Standards and Guidelines 15](#_Toc152353002)

[5.1.2 Screen Layout and Resolution 15](#_Toc152353003)

[5.1.3 Standard Interface Elements 15](#_Toc152353004)

[5.1.4 Message Display and Phrasing 16](#_Toc152353005)

[5.1.6 Localization Considerations 16](#_Toc152353006)

[5.1.7 Accessibility Features 16](#_Toc152353007)

[5.2 Software Interfaces 17](#_Toc152353008)

[5.2.1 Payment Gateways 17](#_Toc152353009)

[5.2.2 Logistics Providers 17](#_Toc152353010)

[5.2.3 Database 18](#_Toc152353011)

[5.2.4 Non-Functional Requirements 18](#_Toc152353012)

[5.3 Hardware Interfaces 18](#_Toc152353013)

[5.3.1 Supported Device Types 18](#_Toc152353014)

[5.3.2 Data and Control Interactions 19](#_Toc152353015)

[5.3.3 Communication Protocols 19](#_Toc152353016)

[5.3.4 Inputs and Outputs 19](#_Toc152353017)

[5.3.5 Timing Issues 19](#_Toc152353018)

[5.4 Communications Interfaces 20](#_Toc152353019)

[5.4.1 Email Communication 20](#_Toc152353020)

[5.4.2 Web Browser Integration 20](#_Toc152353021)

[5.4.3 Network Protocols 20](#_Toc152353022)

[5.4.4 Electronic Forms 20](#_Toc152353023)

[5.4.5 Communication Constraints 21](#_Toc152353024)

[5.4.6 Synchronization Mechanisms 21](#_Toc152353025)

[5.4.7 Communication Security 21](#_Toc152353026)

[6. Quality Attributes 21](#_Toc152353027)

[6.1 Performance 22](#_Toc152353028)

[6.2 Reliability 22](#_Toc152353029)

[6.3 Usability 22](#_Toc152353030)

[6.4 Security 23](#_Toc152353031)

[6.5 Maintainability 23](#_Toc152353032)

[7. Internationalization and Localization Requirements 23](#_Toc152353033)

[7.1 Currency Support 23](#_Toc152353034)

[7.2 Date, Time, and Number Formatting 24](#_Toc152353035)

[7.3 Language Support 24](#_Toc152353036)

[7.4 Cultural and Political Sensitivity 25](#_Toc152353037)

[7.5 Global Adaptation 25](#_Toc152353038)

[7.6 Reusability 25](#_Toc152353039)

[8. Additional Requirements 26](#_Toc152353040)

[8.1 Legal and Regulatory Compliance 26](#_Toc152353041)

[8.2 Product Installation and Configuration 26](#_Toc152353042)

[8.3 Logging, Monitoring, and Audit Trails 26](#_Toc152353043)

[8.4 Data Backup and Recovery 27](#_Toc152353044)

[8.5 Accessibility Features 27](#_Toc152353045)

# Introduction

## 1.1 Purpose

This Software Requirements Specification (SRS) document outlines the functional and non-functional requirements of the iShopHub ecommerce app, version 1.0. It serves as a comprehensive guide for the development team, project managers, marketing staff, users, testers, and documentation writers to ensure the successful implementation and usage of the application.

The iShopHub ecommerce app is a user-centric platform designed to cater to the needs of tech enthusiasts and casual shoppers. It provides a convenient and enjoyable shopping experience for exploring, purchasing, and managing a diverse array of technology products.

## 1.2 Document Conventions

This document adheres to the following conventions:

* **Boldface:** Denotes key terms, important concepts, and specific elements of the system.
* **Italics:** Emphasizes supplemental information, explanatory notes, and points of particular interest.
* **Bullets:** Lists related items, sequential steps, and distinct options or choices.
* **Numbers:** Denotes ordered steps, prioritized items, and enumerations.

In addition to these general conventions, the following specific notations are used:

* **Requirement IDs:** Each requirement is assigned a unique identifier, such as "REQ-001," to facilitate traceability and organization.
* **Requirement Types:** Requirements are classified into functional requirements (describing what the system should do) and non-functional requirements (specifying how the system should perform).
* **Requirement Specifications:** Each requirement is clearly and concisely stated, using plain language and avoiding ambiguity.
* **Acceptance Criteria:** For each functional requirement, specific acceptance criteria are defined to verify that the requirement has been met.
* **Verifications:** Where applicable, specific methods or techniques for verifying the implementation of requirements are indicated.

## 1.3 Project Scope

### 1.3.1 Scope Definition

This Software Requirements Specification (SRS) document outlines the functional and non-functional requirements for the initial release of the iShopHub ecommerce app, version 1.0. The scope encompasses the core features and functionalities necessary to provide a seamless and enjoyable shopping experience for tech enthusiasts and casual shoppers.

### 1.3.2 Core Features

The initial release of the iShopHub ecommerce app will include the following core features:

* User Authentication and Profiles: Users can create and manage their profiles, including setting personal preferences and updating their information. The app provides secure login and registration mechanisms, and users can save their shopping cart items for future reference.
* Product Catalog and Search: A comprehensive product catalog showcases detailed descriptions, high-quality images, and specifications. Users can search and filter products based on various criteria, such as price, brand, category, and features. The app implements a product comparison feature to allow users to compare the specifications of different products.
* Secure Payment Gateway Integration: The app integrates with secure payment gateways to ensure safe and reliable transactions. Users can choose from various payment methods, including credit cards, debit cards. The app provides secure storage of payment information and complies with data privacy regulations.
* Order Tracking and Notification System: Users can track their orders in real-time, from order placement to delivery. The app sends notifications to users about order updates, including shipment confirmations, delivery status, and any potential delays. Users have access to order history and invoices within the app.
* Customer Support Chat Functionality: A live chat feature provides immediate assistance to customers for inquiries, order assistance, and technical issues. Customer support representatives can access customer information and order details to provide efficient assistance. The app maintains a chat history for future reference.

### 1.3.3 Subsequent Releases

Subsequent releases of the iShopHub ecommerce app will introduce additional features based on user feedback and market trends. These features may include:

* Personalized product recommendations
* User ratings and reviews
* An expanded product catalog
* Mobile app versions for iOS and Android

### 1.3.4 Alignment with User and Business Goals

The iShopHub ecommerce app is designed to address the needs of tech enthusiasts and casual shoppers by providing a user-friendly platform for exploring, purchasing, and managing technology products. The app aligns with the business objectives of establishing iShopHub as a go-to platform for technology purchases, increasing customer base, and achieving a customer satisfaction rate of 90% or above.

This SRS document serves as a comprehensive guide for the development team to ensure the successful implementation of the iShopHub ecommerce app, version 1.0. Subsequent SRS documents will be developed for future releases, incorporating additional features and enhancements based on user feedback and market trends.

## **1.4 References**

This Software Requirements Specification (SRS) document references the following documents:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Document Title** | **Author** | **Version** | **Date** | **Source** | **URL** |
| Scope and Vision Document | Basit and Fatima | 1.0 | 20-10-2023 | iShopHub internal Document Repository | <https://github.com/BasitIqbalatgit/SRS-Document.git> |

# 2. over all Description

## **2.1 Product Perspective**

### 2.1.1 Product Context

The iShopHub ecommerce app is an entirely new product designed to address the evolving needs of tech enthusiasts and casual shoppers in the digital marketplace. It aims to provide a seamless and enjoyable shopping experience for exploring, purchasing, and managing an extensive range of technology products.

### 2.1.2 Product Origin

The iShopHub ecommerce app emerged from the vision of establishing iShopHub as a leading technology provider and creating a centralized platform for tech enthusiasts to access and purchase the latest gadgets and innovations.

### 2.1.3 Product Relationship to Existing Systems

The iShopHub ecommerce app will initially function as an independent application accessible through mobile devices. In the future, it may integrate with the iShopHub website and other iShopHub services to provide a unified customer experience across different platforms.

### 2.1.4 Product Ecosystem

The iShopHub ecommerce app exists within a broader ecosystem that includes:

* **Tech product suppliers:** Collaborations with reputable tech suppliers will ensure a comprehensive product catalog and timely product updates.
* **Payment gateways:** Integration with secure payment gateways will facilitate safe and reliable transactions for customers.
* **Logistics providers:** Partnerships with efficient logistics providers will ensure timely product delivery and order fulfillment.
* **Customer support services:** Integration with iShopHub's customer support infrastructure will provide seamless assistance to customers.

### 2.1.5 Visual Representation

A context diagram illustrating the iShopHub ecommerce app's relationship to other systems is presented below:

[Image depicting the iShopHub ecommerce app at the center, connected to tech product suppliers, payment gateways, logistics providers, and customer support services]

This context diagram highlights the interconnected nature of the iShopHub ecommerce app within the broader technology ecosystem.

## 2.2 User Classes and Characteristics

The iShopHub ecommerce app targets two primary user classes:

### 2.2.1 Tech Enthusiasts

Tech enthusiasts are individuals with a keen interest in technology and a desire to stay updated on the latest gadgets and innovations. They possess a strong understanding of technology products and seek a platform that provides access to a comprehensive product catalog, detailed specifications, and personalized recommendations.

Characteristics:

* Avid users of technology products
* Actively seek information about new releases and trends
* Prefer in-depth product descriptions and specifications
* Appreciate personalized product recommendations based on their preferences

### 2.2.2 Casual Shoppers

Casual shoppers are individuals who occasionally purchase technology products but may not possess extensive knowledge about the latest technologies. They seek a user-friendly platform that simplifies the shopping process and provides guidance in making informed decisions.

Characteristics:

* Periodically purchase technology products
* May not be fully aware of the latest technologies
* Value clear product descriptions and simplified comparisons
* Appreciate helpful suggestions and recommendations

### 2.2.3. Favored User Class

While the iShopHub ecommerce app caters to both user classes, tech enthusiasts are considered the favored user class due to their deeper engagement with technology and their potential for repeat purchases. The app's design and features will prioritize the needs and preferences of tech enthusiasts while still providing a seamless experience for casual shoppers.

### 2.2.4. Alignment with User Needs

The iShopHub ecommerce app aims to address the specific needs of each user class:

* Tech enthusiasts: The app provides a comprehensive product catalog, detailed specifications, personalized recommendations, and a user-friendly interface to support their informed decision-making and exploration of new technologies.
* Casual shoppers: The app offers clear product descriptions, comparison tools, helpful suggestions, and simplified navigation to guide them through the purchasing process and make confident choices.

By catering to the distinct needs of both user classes, the iShopHub ecommerce app strives to provide a satisfying and enjoyable shopping experience for all its users.

## **2.3 Operating Environment**

### **2.3.1 Hardware Platform**

The iShopHub ecommerce app will initially be released for Android smartphones, with iOS versions to follow in subsequent releases. It will require mobile devices compatible with the respective operating systems and sufficient hardware capabilities to ensure smooth app performance.

### **2.3.2 Operating Systems and Versions**

* **Android:** The initial release will target Android smartphones running Android version 5.0 (Lollipop) or later.
* **IOS:** Future releases will target iOS devices running iOS version 11.0 or later.

### **2.3.3 Geographical Locations**

* **Users:** The iShopHub ecommerce app will be targeted towards users worldwide, with a primary focus on regions with a high adoption of technology products.
* **Servers:** The backend infrastructure supporting the app will be hosted in secure data centers located in strategic regions to ensure optimal performance and accessibility for users worldwide.
* **Databases:** The app's data will be stored in secure and scalable databases hosted in data centers that meet industry standards for data protection and compliance.

### **2.3.4 Organizations Hosting Infrastructure**

The organizations responsible for hosting the iShopHub ecommerce app's infrastructure include:

* **Cloud service providers:** The app's backend infrastructure will be hosted on cloud platforms provided by reputable cloud service providers, ensuring scalability, reliability, and security.
* **Data center operators:** The app's databases will be stored in secure data centers operated by reliable data center providers, ensuring data integrity, availability, and redundancy.

### **2.3.5 Coexisting Software Components**

The iShopHub ecommerce app is designed to coexist seamlessly with other software components and applications used by users. It will integrate with existing payment gateways, logistics providers, and customer support systems to provide a unified user experience.

### **2.3.6 Infrastructure Requirements Specification**

Due to the complexity of the technical infrastructure involved, a separate infrastructure requirements specification will be developed to detail the specific technical requirements for hosting and operating the iShopHub ecommerce app. This document will provide detailed specifications for cloud services, data center infrastructure, network configurations, and security measures to ensure the successful deployment and operation of the app.

## 2.4 Design and Implementation Constraints

The development of the iShopHub ecommerce app is subject to several design and implementation constraints that influence the technological choices and approaches:

### 2.4.1 Programming Language and Framework

* Programming Language: The app will be primarily developed using the Kotlin programming language, adhering to modern Android development practices and utilizing the Android Jetpack suite of libraries.

Rationale: Kotlin is the preferred language for Android app development due to its conciseness, safety features, and interoperability with Java, ensuring compatibility with existing Android libraries and components.

### 2.4.2 Database Technology

* Database: The app's data will be stored in a cloud-based NoSQL database, such as MongoDB or Amazon Dynamo DB.

Rationale: NoSQL databases offer scalability, flexibility, and performance, enabling efficient handling of the app's dynamic data requirements and high user traffic.

### 2.4.3 Cloud Infrastructure

* Cloud Platform: The app's backend infrastructure will be hosted on a reputable cloud platform, such as Amazon Web Services (AWS) or Google Cloud Platform (GCP).

Rationale: Cloud platforms provide scalability, reliability, and security, ensuring that the app can handle increasing user demands and maintain data integrity.

### 2.4.4 Third-Party Integrations

* Payment Gateways: The app will integrate with secure payment gateways, such as Stripe or PayPal, to facilitate secure online transactions.

Rationale: Integrating with established payment gateways ensures compliance with industry standards and provides a seamless payment experience for users.

### 2.4.5 User Interface Design

* User Interface Guidelines: The app's user interface will adhere to Material Design guidelines, ensuring a consistent and intuitive user experience across Android devices.

Rationale: Material Design provides a standardized design language that is optimized for touch-based interactions, ensuring user familiarity and ease of use.

### 2.4.6 Performance Optimization

* Performance Considerations: The app will be optimized for performance, considering factors such as network latency, data caching, and efficient memory management.

Rationale: Performance optimization is crucial for ensuring a smooth and responsive user experience, especially when dealing with real-time data updates and product images.

These constraints shape the development approach and ensure that the iShopHub ecommerce app meets the specified requirements while adhering to industry best practices and maintaining a high standard of quality and performance.

## **2.5 Assumptions and Dependencies**

The development and successful operation of the iShopHub ecommerce app rely on several assumptions and dependencies:

### **2.5.1 Assumptions**

* **Continued Growth in Technology Product Demand:** The iShopHub ecommerce app is founded on the assumption that the demand for technology products will continue to grow, providing a sustainable market for the app's services.
* **Adequate Supply Chain Management:** The app's ability to fulfill customer orders and ensure timely product delivery depends on the efficiency and reliability of the supply chain, including product availability from suppliers and logistics providers.
* **Reliable Internet Infrastructure:** The app's functionality and user experience are contingent upon the availability of a reliable internet connection for both users and the backend infrastructure.
* **Collaboration with Tech Product Suppliers:** The app's comprehensive product catalog and up-to-date information rely on ongoing collaboration with reputable tech product suppliers to provide accurate and detailed product listings.

### **2.5.2 Dependencies**

* **External Payment Gateways:** The app's secure payment processing functionality depends on the availability and reliable operation of external payment gateways, such as Stripe or PayPal.
* **Logistics Providers:** The app's ability to fulfill customer orders and deliver products to users is dependent on the services of logistics providers, ensuring timely and efficient transportation of goods.
* **Cloud Infrastructure Providers:** The app's backend infrastructure relies on cloud platforms, such as AWS or GCP, to provide scalable, reliable, and secure hosting for the app's data and services.
* **Operating System Compatibility:** The app's functionality is dependent on compatibility with the specified operating systems, including future versions of Android and iOS.
* **Third-Party Libraries and APIs:** The app's development may utilize third-party libraries and APIs, which may require ongoing maintenance and compatibility considerations.

These assumptions and dependencies are crucial considerations for the project team. Monitoring and addressing any changes or potential disruptions in these factors is essential to ensure the continued success of the iShopHub ecommerce app.

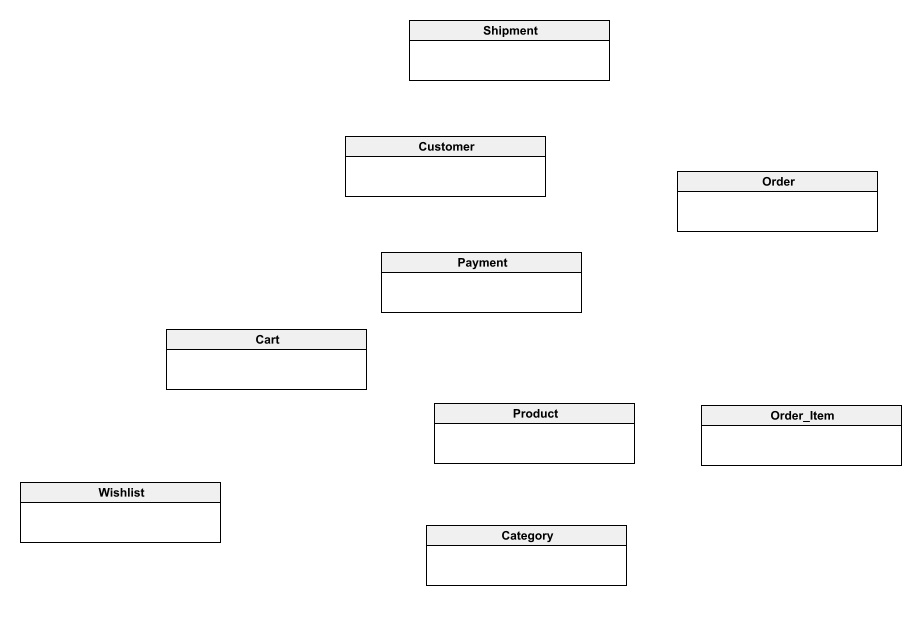
# **3. System Features**

|  |  |  |
| --- | --- | --- |
| Feature | Description | Requirements |
| User Profile Management | Manage and maintain user accounts | 1. Allow users to create, update, and delete profiles.  2. Ensure secure storage of user information. |
| Product Catalog | Manage and display products | 1. Admin can add, update, and remove products.  2. Upload images and descriptions.  3. Set pricing and availability. |
| Inventory Management | Track and manage product inventory | 1. Track stock levels for products.  2. Receive updates from suppliers.  3. Create purchase orders. |
| Order Management | Process and manage customer orders | 1. Accept and manage orders for products.  2. Update order status.  3. Process payments.  4. Handle order cancellations and refunds. |
| Payment Processing | Securely process payments | 1. Integrate with secure payment gateways.  2. Process payments securely.  3. Manage refunds. |
| Delivery Management | Manage the delivery of orders | 1. Assign orders to delivery personnel.  2. Track delivery progress for products.  3. Provide delivery updates to customers. |
| Customer Relationship Management | Manage customer relationships | 1. Track customer orders, preferences, and interactions.  2. Provide personalized recommendations.  3. Offer efficient customer support. |
| Reporting and Analytics | Generate reports and analyze data | 1. Generate reports on sales, inventory, and customer behavior for technology gadgets.  2. Identify trends and opportunities.  3. Optimize business operations. |
| Administrative Dashboard | Manage app settings and features | 1. Manage user permissions.  2. Create and manage promotions for laptops, PCs, and mobiles.  3. Monitor app performance.  4. Ensure app security. |
| User Authentication | Manage user authentication process | 1. User must register in to the system.  2. User can login to the system.  3. Use can logout to the system.  4. User can delete his/her account. |
| Search Functionality | Enable product and order search | 1. User can search product by name and keywords. |
| Product Recommendation | Provide personalized product recommendations | 1. Recommend technology gadgets based on customer purchase history.  2. Use predictive analytics.  3. Consider seasonal availability and trending products. |

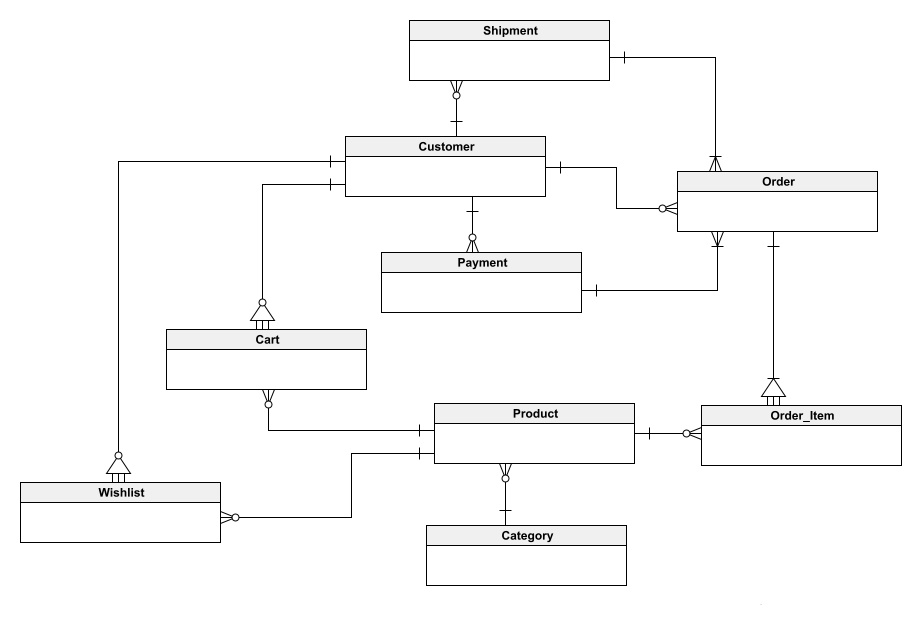
# 4. Data Requirements

## 4.1 Logical data model

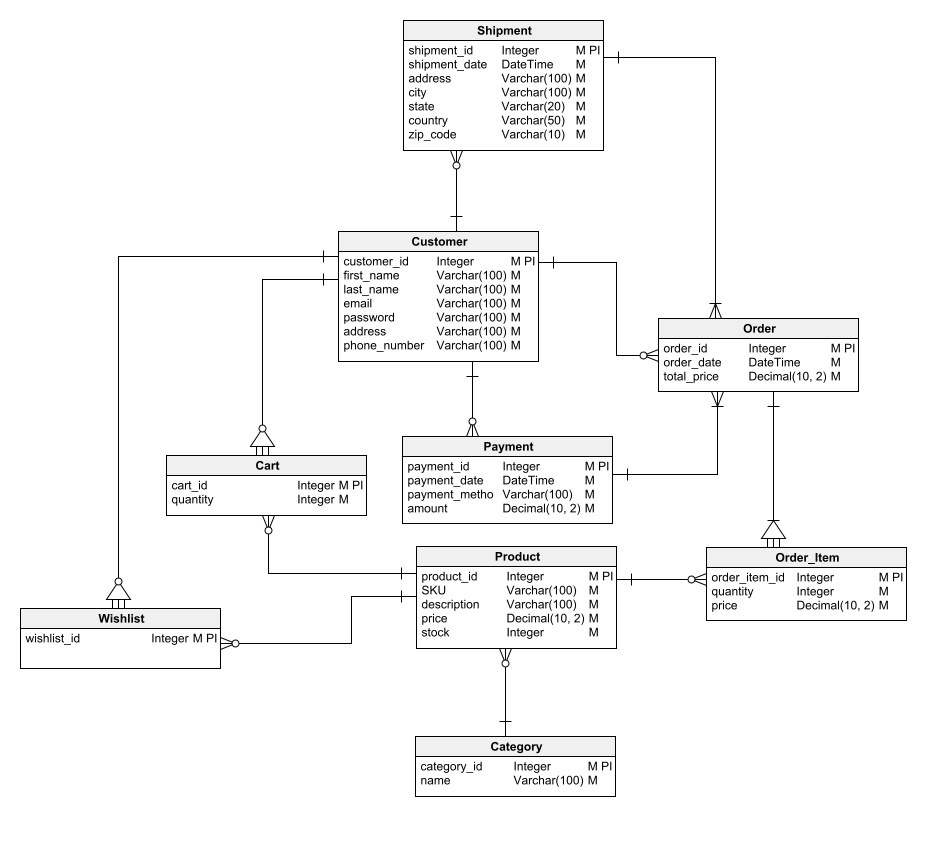
### 4.1.1. Level 0



### 4.1.2. Level 1



### 4.1.3. Level 3



## 4.2 Data dictionary

|  |  |
| --- | --- |
| **Customer** | |
| customer\_id | integer |
| first\_name | string |
| last\_name | string |
| email | string |
| password | string |
| address | string |
| phone\_number | string |

|  |  |
| --- | --- |
| **Cart** | |
| cart\_id | integer |
| quantity | integer |

|  |  |
| --- | --- |
| **Wishlist** | |
| wishlist\_id | integer |

|  |  |
| --- | --- |
| **Product** | |
| product\_id | integer |
| description | string |
| price | decimal |
| stock | integer |

|  |  |
| --- | --- |
| **Category** | |
| category\_id | integer |
| name | string |

|  |  |
| --- | --- |
| **Order** | |
| order\_id | integer |
| order\_date | date/time |
| total\_price | decimal |

|  |  |
| --- | --- |
| **Order\_Item** | |
| order\_item\_id | integer |
| quantity | integer |
| price | decimal |

|  |  |
| --- | --- |
| **Payment** | |
| payment\_id | integer |
| payment\_date | date |
| payment\_method | string |
| amount | decimal |

|  |  |
| --- | --- |
| **Shipment** | |
| shipment\_id | integer |
| shipment\_date | date |
| address | string |
| city | string |
| state | string |
| country | string |
| zip\_code | string |

## 4.3 Report

Various reports will be generated to provide insights into different aspects of the business. These reports are crucial for monitoring and optimizing operations. Below are descriptions of the key reports:

### 4.3.1. Sales Report:

**Characteristics:** Provides a comprehensive overview of sales performance.

**Content:** Includes information on sales volume, revenue, and trends.

**Totaling Levels:** Summarizes sales data at daily, weekly, and monthly intervals.

### 4.3.2. Inventory Report:

**Characteristics**: Offers insights into product stock levels.

**Content:** Displays current inventory status, product quantities, and stock movements.

**Totaling Levels:** Totals inventory levels and highlights products requiring restocking.

### 4.3.3. Order Fulfillment Report:

**Characteristics:** Monitors the fulfillment process of customer orders.

**Content:** Tracks order status, from processing to delivery, including any pending or canceled orders.

**Totaling Levels:** Totals the number of orders fulfilled, pending, and canceled.

## 4.4 Data acquisition, integrity, retention, and disposal

# 5. External interface requirements

## **5.1 User Interfaces**

The iShopHub ecommerce app's user interface will be designed to provide a seamless, intuitive, and user-friendly experience for both tech enthusiasts and casual shoppers. It will adhere to modern design principles and accessibility guidelines to ensure inclusivity and ease of use for all users.

### **5.1.1 Design Standards and Guidelines**

* **User Interface Style Guide:** The app will follow the Material Design guidelines, ensuring a consistent and familiar user experience across Android devices.
* **Fonts and Typography:** The app will use clear and readable fonts appropriate for mobile devices, ensuring optimal text visibility and legibility.
* **Icons and Button Labels:** Icons and button labels will be consistent, easily recognizable, and convey their function clearly, minimizing ambiguity and confusion.
* **Images and Color Schemes:** Images will be high-quality and relevant to the products, while color schemes will be aesthetically pleasing and adhere to accessibility standards.
* **Field Tabbing Sequences:** Tabbing sequences will be intuitive and logical, allowing users to navigate through input fields seamlessly using keyboard shortcuts.
* **Commonly Used Controls:** The app will utilize standard and familiar controls, such as buttons, menus, and dropdowns, to enhance user familiarity and ease of interaction.
* **Branding Graphics:** The app will incorporate consistent branding elements, including the iShopHub logo, color palette, and design motifs, to maintain brand identity.
* **Copyright and Privacy Notices:** Copyright and privacy notices will be clearly displayed and easily accessible, ensuring transparency and compliance with legal requirements.

### **5.1.2 Screen Layout and Resolution**

* **Screen Size:** The app will adapt its layout to various screen sizes and resolutions, ensuring optimal viewing and interaction across different Android devices.
* **Layout Constraints:** The app will adhere to layout guidelines to ensure consistent and visually appealing arrangements of elements across different screen sizes.

### **5.1.3 Standard Interface Elements**

* **Help Button:** A consistent help button will be prominently displayed throughout the app, providing users with easy access to context-sensitive assistance and support information.
* **Shortcut Keys:** Keyboard shortcuts will be implemented for frequently used actions, allowing users to navigate the app efficiently using keyboard input.

### **5.1.4 Message Display and Phrasing**

* **Message Conventions:** Message display formats, including error messages, confirmation prompts, and informational notifications, will follow consistent conventions, ensuring clear and understandable communication with users.
* **Phrasing and Terminology:** The app will use clear and concise language, avoiding technical jargon and ensuring that all users can easily understand the information presented.

### **5.1.5 Data Validation Guidelines**

* **Input Value Restrictions:** Input fields will incorporate appropriate validation rules and restrictions to ensure the accuracy and integrity of user-entered data.
* **Field Content Validation:** Field contents will be validated promptly, providing users with immediate feedback and preventing errors from being submitted.

### **5.1.6 Localization Considerations**

* **Layout Standards:** The app's layout will be designed to facilitate software localization, enabling effortless adaptation to different languages and cultural contexts.
* **Text Translation:** Textual elements, including product descriptions, error messages, and user interface labels, will be translated accurately and culturally appropriately.
* **Regional Settings:** The app will support regional settings, such as currency formats, date and time displays, and language preferences, to cater to users worldwide.

### **5.1.7 Accessibility Features**

* **Visual Impairment:** The app will incorporate accessibility features for visually impaired users, such as alternative text for images, compatible screen readers, and high-contrast color themes.
* **Color Blindness:** The app's color palette will be designed with color blindness in mind, ensuring that color-based information remains accessible to all users.
* **Other Limitations:** The app will strive to accommodate users with other limitations, such as physical or motor impairments, by providing alternative input methods and navigation options.

## **5.2 Software Interfaces**

The iShopHub ecommerce app will interact with various software components to provide a seamless and integrated user experience. These interfaces will ensure data synchronization, enable secure transactions, and facilitate communication with external services.

### **5.2.1 Payment Gateways**

* **Purpose:** The app will integrate with secure payment gateways, such as Stripe or PayPal, to process online payments for customer purchases.
* **Formats and Contents:** The app will exchange data with payment gateways in JSON format, including customer information, order details, and payment credentials.
* **Data Mapping:** Customer information from the app will be mapped to corresponding fields in the payment gateway's data structure, ensuring accurate processing of payments.
* **Services:** The app will utilize payment gateway services to validate payment information, authorize transactions, and provide payment confirmation notifications.

### **5.2.2 Logistics Providers**

* **Purpose:** The app will integrate with logistics providers, such as FedEx or UPS, to handle order fulfillment and product delivery.
* **Formats and Contents:** The app will exchange data with logistics providers in XML format, including order details, shipping addresses, and tracking information.
* **Data Mapping:** Order information from the app will be mapped to corresponding fields in the logistics provider's data format, ensuring accurate order processing and shipment tracking.
* **Services:** The app will utilize logistics provider services to generate shipping labels, track shipment status, and provide delivery notifications to users.

### **5.2.3 Database**

* **Purpose:** The app will connect to a cloud-based NoSQL database, such as MongoDB or Amazon Dynamo DB, to store and retrieve product catalog data, user information, and order details.
* **Formats and Contents:** The app will communicate with the database using JSON format, exchanging product information, user data, and order data.
* **Data Mapping:** Product, user, and order data will be mapped to corresponding collections and documents in the database, ensuring efficient data storage and retrieval.
* **Services:** The app will utilize database services to store and retrieve product information, maintain user profiles, and manage order records.

### **5.2.4 Non-Functional Requirements**

* **Response Times:** The app will maintain consistent response times for data exchange with external systems, ensuring a smooth and responsive user experience.
* **Security Controls:** The app will employ robust security measures, such as encryption and access control, to protect sensitive data transmitted between the app and external systems.
* **Service Availability:** The app will rely on the availability of external services, such as payment gateways, logistics providers, and the database, to ensure uninterrupted operation and fulfill customer orders.

## **5.3 Hardware Interfaces**

The iShopHub ecommerce app will primarily operate on Android smartphones and tablets, interacting with various hardware components to provide a comprehensive user experience.

### **5.3.1 Supported Device Types**

* **Android Smartphones:** The app will support Android smartphones running Android version 5.0 (Lollipop) or later.
* **Android Tablets:** The app will support Android tablets running Android version 5.0 (Lollipop) or later.

### **5.3.2 Data and Control Interactions**

* **Touchscreen:** The app will utilize the touchscreen capabilities of Android devices to provide intuitive gesture-based interactions, including product selection, navigation, and input.
* **Accelerometer and Gyroscope:** The app will utilize the accelerometer and gyroscope sensors to detect device orientation and adjust the app's interface accordingly, ensuring a consistent user experience in both portrait and landscape modes.
* **Camera:** The app will integrate with the device's camera to enable barcode scanning for product identification and augmented reality features for product visualization.
* **Microphone:** The app may utilize the microphone for voice search functionality and potential future integrations.

### **5.3.3 Communication Protocols**

* **Wi-Fi:** The app will utilize Wi-Fi connectivity to access the internet for data retrieval, communication with backend servers, and online transactions.
* **Cellular Data:** The app will support cellular data connectivity to ensure seamless access to the app's features and services even when Wi-Fi is unavailable.

### **5.3.4 Inputs and Outputs**

* **Input:** The app will receive user inputs through touchscreen interactions, microphone input (if applicable), and sensor data (accelerometer, gyroscope).
* **Output:** The app will provide output through the device's display, speaker, and haptic feedback mechanisms.

### **5.3.5 Timing Issues**

* **Response Times:** The app will strive to maintain consistent and responsive interaction times, ensuring a smooth and user-friendly experience.
* **Sensor Data Processing:** The app will efficiently process sensor data, such as accelerometer and gyroscope readings, to provide real-time orientation adjustments and augmented reality features.
* **Network Latency:** The app will account for network latency and provide appropriate feedback mechanisms to inform users of potential delays in data retrieval and interactions.

## **5.4 Communications Interfaces**

The iShopHub ecommerce app will utilize various communication channels to interact with users, external systems, and support services.

### **5.4.1 Email Communication**

* **Purpose:** Email will be used for sending order confirmations, shipping notifications, promotional offers, and password reset instructions to users.
* **Message Formatting:** Email messages will adhere to standard email formats, including clear subject lines, concise content, and appropriate attachments.
* **Security:** Email communication will utilize secure protocols, such as TLS/SSL, to protect user information and prevent data breaches.

### **5.4.2 Web Browser Integration**

* **Purpose:** The app may integrate with web browsers to display product reviews, external product links, and additional information resources.
* **Data Transfer:** Data exchange between the app and web browsers will adhere to standard web protocols, such as HTTP and HTTPS, ensuring secure and reliable data transfer.

### **5.4.3 Network Protocols**

* **Purpose:** The app will utilize network protocols, such as TCP/IP and UDP, to establish connections with backend servers, payment gateways, and logistics providers.
* **Data Exchange:** The app will exchange data with external systems using JSON or XML formats, ensuring compatibility and efficient data transfer.
* **Security:** Network communication will employ secure protocols and encryption techniques to protect sensitive data, such as user information and payment details.

### **5.4.4 Electronic Forms**

* **Purpose:** Electronic forms will be used for user registration, contact forms, and feedback submission.
* **Data Validation:** Form fields will incorporate appropriate data validation rules to ensure the accuracy and completeness of user-entered information.
* **Security:** Data submitted through electronic forms will be protected using secure transmission methods and stored in encrypted form.

### **5.4.5 Communication Constraints**

* **Email Attachments:** The app will not accept email attachments to prevent potential security risks and malware transmission.
* **Unsolicited Communication:** The app will respect user preferences regarding communication channels and avoid sending unsolicited promotional emails or messages.

### **5.4.6 Synchronization Mechanisms**

* **Data Synchronization:** The app will implement data synchronization mechanisms to ensure that product catalog data, order status information, and user profiles remain consistent across devices.
* **Real-Time Updates:** The app may utilize real-time update mechanisms to provide users with immediate notifications of order status changes, product availability updates, and promotional offers.

### **5.4.7 Communication Security**

* **Data Encryption:** All sensitive data transmitted between the app and external systems will be encrypted using industry-standard encryption algorithms.
* **Access Control:** Access to sensitive data and communication channels will be restricted using appropriate authentication and authorization mechanisms.
* **Secure Storage:** Sensitive data stored on user devices will be encrypted and protected using secure storage protocols.

# **6. Quality Attributes**

In addition to the functional requirements outlined in previous sections, the iShopHub ecommerce app must adhere to a set of nonfunctional quality attributes to ensure a seamless and satisfactory user experience.

## **6.1 Performance**

* **Response Times:** The app should maintain consistent and responsive interaction times for user actions, such as product browsing, search queries, and order placements. Aim for an average response time of less than 2 seconds for all user interactions.
* **Data Retrieval:** The app should efficiently retrieve product information, user data, and order details from the database and external systems. Strive for data retrieval times within 5 seconds for product listings and order status updates.
* **Network Optimization:** The app should optimize network data usage to minimize data consumption and ensure efficient operation on various internet connections. Implement mechanisms to prioritize essential data transfers and reduce unnecessary data requests.

## **6.2 Reliability**

* **System Uptime:** The app should maintain a high level of availability, minimizing downtime and ensuring uninterrupted access for users. Aim for a system uptime of 99.9% or higher.
* **Error Handling:** The app should gracefully handle unexpected errors and provide clear and informative error messages to users. Implement mechanisms to log errors and facilitate debugging for maintenance purposes.
* **Data Integrity:** The app should maintain the integrity of product data, user information, and order details, preventing data loss or corruption. Employ data validation techniques and implement secure data storage mechanisms.

## **6.3 Usability**

* **Ease of Use:** The app should be easy to navigate and use, even for users with limited technical expertise. Design a user interface that is intuitive, consistent, and accessible to all user demographics.
* **Ease of Learning:** The app should be easy to learn and understand, allowing users to quickly grasp its functionalities and features. Provide clear instructions, tooltips, and context-sensitive help throughout the app.
* **User Error Prevention:** The app should incorporate design elements and input validation techniques to minimize user errors and prevent unintentional actions. Provide clear feedback and error messages to help users correct mistakes.

## **6.4 Security**

* **Data Security:** The app should safeguard sensitive user information, including payment details, personal data, and order histories. Implement robust encryption techniques, access control mechanisms, and secure data storage practices.
* **Communication Security:** The app should protect all data transmitted between the app and external systems, such as payment gateways and logistics providers. Utilize secure protocols, such as TLS/SSL, for all network communication.
* **Vulnerability Mitigation:** The app should be regularly tested for security vulnerabilities and patched promptly to address any identified weaknesses. Implement a secure development lifecycle and employ industry-standard security practices.

## **6.5 Maintainability**

* **Modular Design:** The app should be designed with modularity in mind, allowing for easy maintenance, updates, and bug fixes. Implement a well-structured codebase with clear separation of concerns.
* **Documentation:** The app should be accompanied by comprehensive documentation, including code comments, design specifications, and user manuals. Provide clear instructions and guidance for developers to understand and maintain the codebase.
* **Testability:** The app should be designed to be testable, allowing for easy identification and correction of bugs. Implement unit tests, integration tests, and system tests to validate the app's functionality and performance.

# **7. Internationalization and Localization Requirements**

The iShopHub ecommerce app will cater to a global audience, encompassing users from diverse cultures, languages, and regions. To ensure a seamless and user-friendly experience for all, the app must adhere to comprehensive internationalization and localization requirements.

## **7.1 Currency Support**

* **Support for Multiple Currencies:** The app should support a variety of currencies, including major global currencies and regional currencies relevant to target markets.
* **Dynamic Currency Conversion:** The app should provide real-time currency conversion, allowing users to view product prices, order summaries, and payment information in their preferred currency.
* **Currency Formatting:** The app should display currency amounts in the correct format for each supported currency, adhering to regional conventions and decimal separators.

## **7.2 Date, Time, and Number Formatting**

* **Date and Time Formats:** The app should adapt date and time displays to suit the user's locale, including formats for dates, times, and time zones.
* **Number Formats:** The app should display numbers according to regional conventions, including decimal separators, thousand separators, and currency symbols.
* **Address Formatting:** The app should adapt address input fields to match regional address formats, including address line order, postal code formats, and country-specific requirements.
* **Telephone Number Formatting:** The app should recognize and format telephone numbers according to international standards and regional conventions.

## **7.3 Language Support**

* **Multilingual User Interface:** The app should provide a multilingual user interface, translating product descriptions, error messages, and user interface labels into multiple languages.
* **National Spelling Conventions:** The app should adhere to national spelling conventions for languages like English, ensuring that users encounter familiar spellings and terminology.
* **Symbol Usage:** The app should use symbols and icons that are culturally appropriate and understandable to users from different regions.
* **Character Sets:** The app should support the character sets of all supported languages, allowing for proper display of text and input of characters specific to each language.

## **7.4 Cultural and Political Sensitivity**

* **Avoid Cultural Offensiveness:** The app should be designed with cultural sensitivity in mind, avoiding imagery, language, or references that could be considered offensive or insensitive to users from different cultures.
* **Compliance with Local Laws and Regulations:** The app should comply with all applicable laws and regulations in the regions where it operates, including data privacy laws, consumer protection laws, and e-commerce regulations.
* **Respect for Local Customs and Traditions:** The app should demonstrate respect for local customs and traditions, avoiding content or features that could be considered disrespectful or inappropriate in certain regions.

## **7.5 Global Adaptation**

* **Paper Sizes:** The app should support printing of documents, such as invoices and receipts, in standard paper sizes used in different regions.
* **Weights and Measurements:** The app should display product measurements, such as weight, dimensions, and capacity, in units that are commonly used and understood by users in different regions.
* **Electrical Compatibility:** The app should provide information on product compatibility with electrical voltages and plug shapes used in different regions, ensuring safe and proper usage worldwide.

## **7.6 Reusability**

* **Modular Design:** The app's internationalization and localization features should be designed in a modular fashion, allowing for easy adaptation to new languages, currencies, and regions without disrupting existing functionality.
* **Configuration Management:** The app should utilize configuration management techniques to store and manage internationalization and localization settings, enabling dynamic adaptation to user preferences and regional settings.
* **Translation Management Tools:** The app should integrate with translation management tools to streamline the translation process, ensuring consistency and accuracy across different languages.

# **8. Additional Requirements**

In addition to the functional and non-functional requirements outlined in previous sections, the iShopHub ecommerce app must adhere to specific legal, regulatory, and compliance requirements.

## **8.1 Legal and Regulatory Compliance**

* **Data Protection and Privacy:** The app must comply with all applicable data protection and privacy laws, including the General Data Protection Regulation (GDPR) and other regional regulations.
* **Consumer Protection Laws:** The app must adhere to consumer protection laws, ensuring fair and transparent business practices, clear refund policies, and protection against fraudulent activities.
* **E-commerce Regulations:** The app must comply with e-commerce regulations, including tax laws, payment processing standards, and consumer transaction safeguards.

## **8.2 Product Installation and Configuration**

* **Minimal Installation Requirements:** The app should have minimal installation requirements, allowing for easy installation on a wide range of Android devices.
* **Configuration Options:** The app should provide users with customizable configuration options, such as language preferences, currency selection, and notification settings.
* **Startup and Shutdown Procedures:** The app should ensure a smooth startup process, displaying clear loading screens and minimizing startup times. Shutdown procedures should be straightforward and prevent data loss.

## **8.3 Logging, Monitoring, and Audit Trails**

* **User Activity Logging:** The app should maintain logs of user activity, including login attempts, product searches, order placements, and system interactions.
* **System Performance Monitoring:** The app should implement mechanisms to monitor system performance, including resource usage, response times, and error occurrences.
* **Audit Trail Generation:** The app should generate audit trails for critical actions, such as user account creation, order modifications, and administrative changes.

## **8.4 Data Backup and Recovery**

* **Regular Data Backups:** The app should implement a regular data backup schedule to ensure the safety and integrity of user data, product information, and order records.
* **Disaster Recovery Plan:** The app should have a comprehensive disaster recovery plan in place to restore data and functionality in the event of system failures or data breaches.

## **8.5 Accessibility Features**

* **Support for Assistive Technologies:** The app should support assistive technologies, such as screen readers and voice control tools, to enable usage by individuals with visual impairments or other limitations.
* **Accessibility Settings:** The app should provide accessibility settings that allow users to customize the interface and interactions to suit their individual needs and preferences.
* **Compatibility with Accessibility Standards:** The app should strive to adhere to recognized accessibility standards and guidelines to ensure inclusive usage for all users.

Appendix A: Glossary

|  |  |
| --- | --- |
| TERMS | DEFINITION |
| API | Application Programing Interface |
| GDPR | General Data Protection Regulation |
| HTTP | Hyper Text Transfer Protocols |
| HTTPs | Hyper Text Transfer Protocol Secure |
| JSON | javaScript Object Notation |
| TLS | Transport Layer Security |
| SSL | Secure Socket Layer |
| DB | Database |
| SQL | Structured Query Language |
| XML | Extensible Markup Language |
| GCP | Google Cloud Platform (GCP). |
| AWS | Amazon Web Services |
| iOS | Iphone Operating System |