Software Requirements Specifications

E-Commerce Smart Store

Project Code:

Internal Advisor:

Dr. Faisal Shahzad

Project Manager:

Dr. Muhammad Illyas

Project Team:

Badar Munir (Group Leader)	BSCS51F21S095
Hafsa Hadi	BSCS51F22S001
Sana Jabbar	BSCS51F22S030

Submission Date:

21/10/2025

Project Manager's Signature

Document Information

Category	Information
Customer	CS, UOS
Project	E-commerce Smart Store
Document	Requirement Specifications
Document Version	1.0
Identifier	
Status	Draft
Author(s)	Badar Munir BSCS51F21S095 Hafsa Hadi BSCS51F22S001 Sana Jabbar BSCS51F22S030
Approver(s)	Dr. Muhammad Ilyas
Issue Date	Oct 17,2025
Document Location	
Distribution	1. Dr. Faisal Shahzad 2. Dr. Muhammad Ilyas

Definition of Terms, Acronyms and Abbreviations

This section should provide the definitions of all terms, acronyms, and abbreviations required to interpret the terms used in the document properly.

Term	Description
Term	**Description**
VR	Virtual Reality
AR	Augmented Reality
AI	Artificial Intelligence
UI	User Interface
DBMS	Database Management System
SRS	Software Requirements Specification
User	Customer using the system

1. Introduction

1.1 Purpose of Document

This document outlines the detailed software requirements for the **Virtual Try-On System with Voice Search**, an advanced eCommerce solution that leverages **AI and Virtual Reality** to enhance user experience. It is intended for developers, project managers, and stakeholders involved in the design, implementation, and evaluation of the system.

1.2 Project Overview

The **Virtual Try-On System** is designed to revolutionize the online shopping experience by enabling customers to **virtually try on** apparel, accessories, or cosmetics using **VR and AR technologies**. The system also features **voice-based search** and navigation for a more natural and interactive shopping process.

1.3 Scope

The system allows users to browse, visualize, and purchase products using VR. Through **voice commands**, customers can search products, filter items, and navigate the interface hands-free. Admins can manage product listings, monitor analytics, and ensure smooth operation of all system modules.

2. Overall System Description

2.1 The Viewpoint of the Product

Unlike conventional eCommerce platforms, this system focuses on **immersive, interactive shopping**. Users can try products in a **virtual environment** before purchasing, reducing return rates and improving confidence in buying decisions.

2.2 User Types and Personalities

- **Customers:** Individuals browsing and virtually trying products before purchasing.
- **Admins:** Users managing inventory, products, analytics, and orders.
- **Guests:** Visitors exploring the platform with limited access (no try-on or purchase).

2.3 Environment of Operation

- **Frontend:** React, Three.js, and WebXR API.
- **Backend:** Django or Node.js.
- **Database:** PostgreSQL or MongoDB.
- **Hosting:** Cloud-based deployment for scalability and uptime.
- **Hardware:** VR headset, webcam, or smartphone camera (for AR mode).

2.4 Premises and Requirements

- Requires internet connection and compatible camera device.

- Depends on AI libraries for facial/body detection.
- Relies on external APIs for payment and voice recognition (e.g., Google Speech-to-Text).

3. External Interface Requirements

3.1 User Authentication

- Users can register and log in with email/password.
- Password recovery and reset functionalities are available.
- Two-factor authentication enhances account security.

3.2 Voice-Based Search and Navigation

- The system supports voice commands for searching, sorting, and filtering products.
- Speech recognition must support multiple accents and languages.
- Voice feedback confirms recognized commands and assists visually impaired users.

3.3 Virtual Try-On Functionality

- Users can view how apparel or accessories look in real-time using AR/VR visualization.
- The system overlays products onto the user's live image using device camera or VR model.
- Users can rotate, zoom, and adjust fitting in the virtual environment.

3.4 Product Search and Browsing

- Supports filtering by category, price, and popularity.
- Search via keyword or voice input.
- AI recommendations suggest similar or trending products.

3.5 Shopping Cart and Checkout

- Users can add, remove, or edit items in their cart.
- System shows order summary before payment.

- Supports multiple payment methods (cards, wallets, bank transfer).

3.6 Order Management

- Users can view order history and status.
- Order cancellation and refund options available.
- Admins can manage and update order processing.

3.7 Admin Dashboard

- Product management (add/edit/delete).
- View user analytics and purchase patterns.
- Manage promotional banners and system configurations.

---

4. Non-Functional Requirements

4.1 Performance

- Pages should load within 3 seconds.
- System supports up to 10,000 concurrent users.
- VR rendering must maintain at least 30 FPS for a smooth experience.

4.2 Usability

- Intuitive UI with consistent layout.
- Voice control tutorials for first-time users.
- Accessibility for differently-abled users (voice feedback, screen reader support).

4.3 Security

- End-to-end encryption of all transactions and user data.
- Secure authentication and session management.
- Protection from SQL injection, XSS, and data breaches.

4.4 Scalability

- Modular architecture to add new AI or VR modules in the future.
- Cloud hosting allows horizontal scaling based on traffic.

---

5. External Interface Requirements

5.1 User Interfaces

- Responsive design compatible with desktop, mobile, and VR devices.
- Voice icon available for hands-free control.
- Intuitive product display and 3D visualization panels.

5.2 Hardware and Software Interfaces

- Integration with **Google Cloud Speech API** for voice recognition.
- Use of **OpenCV** or **TensorFlow** for virtual try-on tracking.
- Integration with payment gateways (PayPal, Stripe).

5.3 Communication Interfaces

- RESTful APIs between frontend and backend.
- WebSockets for real-time updates (chatbot, order tracking).

---

6. Assumptions and Dependencies

- Internet and camera access are mandatory for AR/VR features.
- Dependent on cloud AI models for speech and image processing.
- Integration delays with external APIs may affect performance.

---

7. System Architecture

The system architecture includes:

- **Frontend (React + WebXR)** for rendering and VR integration.
- **Backend (Django/Node.js)** for API handling and data management.
- **Database (PostgreSQL/MongoDB)** for product and user data.
- **Cloud Services** for AI/ML and voice APIs.
- **Payment Gateway Integration** for secure transactions.
- **Caching Layer (Redis)** for fast data access.

---

8. Legal and Regulatory Compliance

- Must comply with **GDPR** and **Pakistan's Data Protection Laws**.
- Payment gateways must be **PCI-DSS compliant**.
- Ensure secure data handling and storage according to organizational policy.

9. References

[1] Mozilla Developer Network, [Online]. Available: https://developer.mozilla.org
[2] Django Documentation, [Online]. Available: https://docs.djangoproject.com
[3] Google Speech API Docs, [Online]. Available: https://cloud.google.com/speech-to-text
[4] "Virtual Reality in Retail," ScienceDirect, [Online]. Available: https://www.sciencedirect.com
[5] WebXR Documentation, [Online]. Available: https://immersiveweb.dev

10. Appendices (Summary)

1. UI Mockups – Home, Product, Try-On, Checkout, Admin Dashboard
2. Workflow Diagrams – Voice Search Flow, Virtual Try-On Interaction, Order Cycle
3. Database Schema – Users, Products, Orders, TryOnLogs

4. API Documentation – Product, Voice, Order, Admin APIs
5. Glossary – AR, VR, WebXR, PCI-DSS, GDPR
6. Compliance – Data Privacy, Payment Security
7. System Requirements – Hardware & Software
8. Test Cases – Voice Search, Try-On, Checkout
9. Future Scope – AI Style Advisor, Multi-language Voice Support, Mobile App
10. References – As listed above