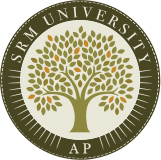
**Software Requirements Specification (SRS)**

**For**

**Fashion Community Platform**

****

**Table of Contents**

* **Introduction**
  + Purpose
  + Scope
  + Definitions, Acronyms, and Abbreviations
  + References
  + Overview
* **Overall Description**
  + Product Perspective
  + Product Functions
  + User Characteristics
  + Constraints
  + Assumptions and Dependencies
* **External Interface Requirements**
  + User Interfaces
  + Hardware Interfaces
  + Software Interfaces
  + Communications Interfaces
* **System Features**
* **Other Non-Functional Requirements**
  + Performance Requirements
  + Software System Attributes
    - Reliability
    - Availability
    - Security
    - Maintainability
  + Business Rules
* **Other Requirements**
* **Appendices**
  + Glossary
  + Analysis Models

**Introduction**

**Purpose**

The Fashion Community Platform is a web-based social networking platform for fashion enthusiasts. It allows users to post fashion-related content, interact with others, explore trending styles, and receive AI-driven recommendations.

This document outlines the functional, external, and non-functional requirements of the system.

**Scope**

The system is a MERN-stack web application that enables users to:

* Create and manage profiles.
* Post fashion-related content (images, videos, captions, hashtags).
* View a personalized feed based on followed users and trending content.
* Like, comment, and share posts.
* Follow and unfollow other users.
* Search for profiles and hashtags.
* Receive AI-driven recommendations and trend predictions.

The platform will be built using the MERN stack (MongoDB, Express.js, React.js, Node.js) and will be optimized for both desktop and mobile devices.

**AI/ML**

AI Feature:

* Personalized Recommendations: Users will receive fashion post suggestions and recommended users based on their selected preferences during profile creation and engagement history.

Future Enhancements:

* Automatic Image Tagging: AI auto-tags images with fashion styles (e.g., "Streetwear," "Casual").
* Trend Analysis: AI predicts emerging fashion trends from user engagement.
* Post Quality Scoring: AI ranks high-quality content based on interactions.

**Definitions, Acronyms, and Abbreviations**

* MERN Stack: MongoDB, Express.js, React.js, Node.js
* AI/ML: Artificial Intelligence / Machine Learning
* JWT: JSON Web Token (used for authentication)
* UI/UX: User Interface / User Experience

**References**

* MERN Stack Documentation
* Tailwind CSS Documentation
* TensorFlow.js for AI integration

**Overview**

This document outlines the platform's functionality, external interfaces, performance requirements, design constraints, and non-functional attributes.

**The Overall Description**

**Product Perspective**

The Fashion Community Platform is a social media-inspired web application built using the MERN (MongoDB, Express.js, React.js, Node.js) stack. It serves as an interactive platform where users can share and explore fashion-related content, engage with other fashion enthusiasts, and receive AI-driven recommendations.

System Architecture

The platform follows a client-server architecture with the following components:

* Frontend (Client-Side): Built with React.js, offering an interactive and dynamic user interface.
* Backend (Server-Side): Developed using Node.js and Express.js, responsible for handling API requests, user authentication, and business logic.
* Database: MongoDB is used to store user profiles, posts, comments, and follow relationships.
* AI/ML Module: Uses TensorFlow.js or Python-based ML models to provide intelligent recommendations and trend predictions.

Key System Interactions

* Users sign up and create profiles with fashion interests.
* Users can upload images or videos, add captions, and tag fashion trends.
* AI algorithms automatically tag uploaded images with relevant fashion terms.
* Posts appear in feeds based on follows and trending algorithms.
* Users can like, comment, and share posts to increase engagement.
* The search functionality allows users to discover content by keyword or hashtag.
* The admin panel lets moderators manage flagged posts.

**Product Functions**

The system provides the following core functionalities:

User Authentication & Profile Management

* Signup/Login: Users can register and authenticate via email/password.
* Secure Authentication: Uses JWT tokens for session management.
* Profile Customization: Users can upload a profile picture, write a bio, and set fashion interests.
* Follow System: Users can follow/unfollow other users to personalize their feed.

Posting & Interaction Features

* Post Creation: Users can upload images/videos with captions, hashtags, and tags.
* Post Engagement: Users can like, comment, and share posts.
* Editing & Deleting Posts: Users can update or delete their posts.
* Notifications System: Users receive real-time notifications for likes, comments, and follows.

Content Discovery & AI-Based Recommendations

The platform ensures users easily discover content and receive personalized suggestions to enhance engagement.

Core Features:

* Personalized Feed: Users see posts from followed accounts and recommendations based on preferences.
* Explore Page: A grid-based layout showcasing trending and suggested fashion content.
* AI-Powered Recommendations: AI suggests fashion posts and users to follow based on preferences and engagement.

Future Enhancements:

* AI-Generated Image Tags: AI classifies images into fashion styles.
* Trending Analysis: AI predicts fashion trends based on engagement.
* Advanced Post/User Recommendations: AI suggests content beyond user-selected preferences.

Search & Filtering System

* Keyword & Hashtag Search: Users can find content using search queries.
* Filtered Results: Sort by most liked, recent, or trending.
* Fashion Category Tags: Users can search for outfits, accessories, footwear, etc.

Admin & Moderation System *(Optional but recommended)*

* Dashboard for Admins: Displays total users, posts, flagged content.
* Content Moderation: Admins can delete flagged posts violating guidelines.
* User Banning: Admins can suspend accounts engaged in spam or offensive behaviour.

**User Characteristics**

* Fashion Enthusiasts: Individuals who want to share and explore fashion ideas.
* Content Creators: Users who actively post fashion-related content.
* Casual Browsers: Users who browse for trends without posting.
* Admin Users: Manage and moderate content.

**Constraints**

* Cross-Platform Compatibility: The website must be responsive on desktop and mobile.
* Real-Time Performance: AI-based recommendations should run with minimal latency.
* Storage Limitations: Free-tier hosting services may impose limits on media uploads.
* Content Moderation: Automated NSFW filtering might be needed.
* Scalability: Must handle growth in user base and media uploads.

**Assumptions and Dependencies**

* Internet Connectivity: Users require a stable internet connection.
* Third-Party APIs: External services like Cloudinary (image hosting) or TensorFlow.js (AI models) must be operational.
* User-Generated Content: The platform’s engagement relies on active user participation.
* Device Support: Assumes users access the site via modern web browsers (Chrome, Firefox, Edge, Safari).

**External Interface Requirements**

**User Interfaces**

The platform will have a modern, responsive UI optimized for both desktop and mobile devices. The user interface will be clean, minimalistic, and fashion-focused.

Navigation & Layout

* Navigation Bar (Sticky Header)
  + Logo (clicking redirects to Home)
  + Search bar (allows searching for users, hashtags, and posts)
  + Icons for:
    - Home (Shows personalized feed)
    - Explore (Shows trending posts and suggestions)
    - Notifications (Shows likes, comments, new followers)
    - Profile (Takes user to their own profile)

Page Layouts

* Home Page (Feed)
  + Displays posts in a card format with:
    - User profile picture and username (clicking redirects to profile)
    - Image/video with captions and hashtags
    - Like, Comment , and Share buttons
    - Post timestamp
* Signup/Login Page
  + Left side: Fashion-themed image or animation
  + Right side: Login/signup form with:
    - Input fields for email, username, password, confirm password
    - Social media login buttons (Google, Facebook, etc.)
    - "Forgot Password?" link
* User Profile Page
  + Profile section with:
    - Profile picture, username, bio
    - Follower/Following count
  + Tabs:
    - Posts: Grid layout of user’s posts
    - Liked Posts: Posts liked by the user
  + Buttons:
    - Follow/Unfollow button (if viewing another user's profile)
    - Edit Profile button (if viewing own profile)
* Post Creation Page
  + Upload image/video
  + Add caption (text box with character limit)
  + Add hashtags (suggestions appear as the user types)
  + Preview post appearance before submitting
  + Submit button to upload post
* Explore Page
  + Grid-based layout of posts (similar to Instagram Explore)
  + Infinite scroll for continuous discovery
  + Suggested users to follow
* Notifications Page
  + List format displaying:
    - Profile picture of the user who interacted
    - Text notification (e.g., “@user liked your post”)
    - Timestamp for when the event occurred

**Hardware Interfaces**

The platform will work across different hardware devices:

* Client-side:
  + Devices: Desktop, Laptops, Tablets, Smartphones
  + Browsers: Chrome, Firefox, Edge, Safari
  + Minimum requirements:
    - RAM: 4GB
    - CPU: 2-core processor
    - Storage: 100MB (for caching, offline storage)
* Server-side:
  + Backend Server: Deployed on AWS, DigitalOcean, or Firebase
  + Database Server: MongoDB (Hosted on MongoDB Atlas or self-hosted)
  + AI Processing: TensorFlow.js for real-time AI processing (alternative: Google Cloud AI APIs)

**Software Interfaces**

* Frontend:
  + Developed using React.js and Tailwind CSS
  + Fetches data from the backend using REST API calls
  + Uses local storage/session storage for JWT-based authentication
* Backend:
  + Node.js + Express.js framework
  + Handles user authentication, post management, and AI-based recommendations
* Database:
  + MongoDB (NoSQL)
  + Stores user data, posts, likes, comments, and follow relationships
  + Indexed searches for faster querying
* Third-party Integrations:
  + Cloudinary (for storing and optimizing images/videos)
  + Google OAuth (for social media login)
  + TensorFlow.js (for AI-powered recommendations)
  + Stripe/PayPal (if later adding e-commerce features)

**Communications Interfaces**

The system will communicate using:

* RESTful API Endpoints
  + Authentication APIs: Handle login, signup, JWT token validation
  + User APIs: Manage profiles, followers, and settings
  + Post APIs: Handle post creation, editing, and deletion
  + Interaction APIs: Handle likes, comments, and shares
* WebSockets (for Real-time Updates)
  + Live notifications for likes, comments, new followers
  + Real-time chat feature (if added in future updates)
* Security Protocols
  + HTTPS encryption for secure data transmission
  + JWT authentication for securing API endpoints
  + Rate limiting to prevent spamming/DDoS attacks

**System Features**

The system features are categorized based on core functionalities, user interactions, and AI-powered enhancements.

**User Authentication & Profile Management**

Description: Users can register, log in, and manage their profiles securely.  
Functionalities:  
User Registration:

* Users sign up using email and password.
* Profile creation includes username, bio, profile picture.
* Email verification via OTP (Optional).

User Login & Authentication:

* Secure login via email & password.
* JWT-based authentication for session management.
* OAuth-based login with Google/Facebook (Optional).

Profile Management:

* Users can update profile information (bio, profile picture, password).
* View follower and following count.
* Users can deactivate or delete accounts.

User Roles & Permissions:

* Standard Users: Can post, like, comment, and follow/unfollow others.
* Admins: Can remove flagged posts, ban users, and moderate content**.**

**Posting & Interaction**

Description: Users can create posts, interact with content, and engage with the community.  
Functionalities:

Create Post:

* Users can upload images/videos with captions and hashtags.
* Preview section before submitting a post.
* Image optimization and compression for faster loading.

Like & Comment System:

* Users can like/unlike posts.
* Users can comment on posts (with a reply option).
* Option to edit or delete own comments.

Share Posts:

* Share posts via copy link, social media, or direct messaging.

Save Posts:

* Users can bookmark/save posts for later viewing.

Post Privacy Options:

* Public posts (visible to all users).
* Private posts (visible only to followers).

Report & Flagging System:

* Users can report inappropriate posts/comments.
* Admins receive flagged content for review.

**Feed & Content Discovery**

Description: Users can browse and engage with personalized and trending fashion content.  
Functionalities:

Personalized Feed:

* Shows posts from followed users.
* Prioritizes posts with high engagement (likes, comments).

Trending Section:

* Displays top trending fashion posts based on engagement.
* Highlights trending hashtags and topics.

Infinite Scroll:

* Users can continuously scroll to load more posts dynamically.

Category-Based Filtering:

* Users can filter posts based on fashion categories (e.g., Streetwear, Vintage, High Fashion, etc.).

Search & Hashtag System:

* Search for users by username.
* Search for posts using hashtags or keywords.

**Follow System**

Description: Users can follow and unfollow other users to build their fashion network.  
Functionalities:

Follow & Unfollow Users:

* Users can follow others to see their posts in the feed.
* Unfollowing removes posts from the feed.

Follower & Following Count:

* Users can view a list of their followers and accounts they follow.

Suggested Users to Follow:

* AI-powered recommendations suggest users based on mutual followers, interests, and interactions.

**AI-Powered Features**

Description

AI/ML is used to improve content discovery and user engagement. The current focus is on Personalized Recommendations, with other features planned for future iterations.

Functionalities

Personalized Recommendations *(Implemented)*:

* AI suggests posts and users based on user-selected fashion preferences and interactions.
* Uses a basic recommendation algorithm (Collaborative Filtering or Rule-Based).

Future Enhancements:

* Automatic Image Tagging: AI auto-tags fashion images for better categorization.
* Fashion Trend Analysis: ML predicts fashion trends from user engagement.
* Post Quality Scoring *(Optional)*: AI ranks high-quality fashion content.

**Notifications System**

Description: Keeps users informed about important interactions on the platform.  
Functionalities:

Real-Time Notifications:

* Users receive instant notifications for likes, comments, shares, and follows.
* WebSockets enable real-time updates.

Notification Panel:

* Users can view all notifications in one place.
* Option to mark notifications as read.

Email & Push Notifications (Optional):

* Users can enable email notifications for major updates.
* Push notifications for mobile users.

**Explore Page (Discover New Content & Users)**

Description: Users can explore new fashion trends, posts, and people.  
Functionalities:

Grid-Based Layout:

* Displays popular and new posts in a Pinterest-style grid.

Trending Fashion Tags:

* Showcases top hashtags trending in fashion.

Top Fashion Creators Section:

* Highlights popular fashion influencers and creators.

Infinite Scroll for Continuous Discovery:

* Seamless exploration with endless scrolling**.**

**Security & Data Privacy Features**

Description: Ensures safe interactions and user data protection.  
Functionalities:

Secure User Authentication:

* JWT-based authentication for session management.

Data Encryption:

* AES encryption for sensitive data (passwords, user details).

Rate Limiting & DDoS Protection:

* Prevents spam and abuse using rate limiting.

GDPR Compliance:

* Users can download or delete their data anytime.

**Other Non-Functional Requirements**

**Performance Requirements**

Capacity

* The system should support at least 10,000 concurrent users without significant degradation in performance.
* Each user profile should be able to store up to 500MB of data, including images and videos.
* The database should efficiently handle millions of records of posts, comments, and user interactions.

Dynamic Requirements

* The platform should provide real-time updates for likes, comments, and new followers using WebSockets.
* The AI-powered recommendation engine should respond within 1-2 seconds when suggesting posts or users.

Quality

* Code Quality:
  + The project should follow clean coding principles for maintainability.
  + The backend should follow RESTful API design.
* UI/UX:
  + The UI should be responsive and optimized for mobile, tablet, and desktop screens.
  + Transitions and animations should be smooth, ensuring a great user experience.

**Software System Attributes**

Reliability

* The system should have an error handling mechanism that logs and reports errors without exposing sensitive data.
* If a server failure occurs, the system should automatically restart within 30 seconds.

Availability

* The platform should have 99.9% uptime with minimal downtime for maintenance.
* A backup server should be available in case of a primary server failure.

Security

* User Authentication:
  + Secure JWT authentication with token expiration policies.
  + Implement OAuth 2.0 for social media login options.
* Data Encryption:
  + Use AES-256 encryption for sensitive data such as passwords and user details.
* Access Control:
  + Users can only modify their own posts and profile details.
  + Admins have the ability to ban users and moderate flagged content.
* DDoS Protection:
  + Implement rate-limiting and IP-based blocking for preventing automated attacks.

Maintainability

* Modular Development:
  + The system should follow a microservices architecture to allow independent development and updates.
* Code Documentation:
  + The codebase should be fully documented using tools like JSDoc for JavaScript files.
* Logging & Monitoring:
  + Use Winston for logging errors and Prometheus for system monitoring.

**Business Rules**

* Community Guidelines:
  + No hate speech, nudity, or abusive content is allowed.
  + Users can report inappropriate content, which will be reviewed by admins.
* Advertisement Policy:
  + If advertisements are introduced, they should be fashion-related and must not affect user experience.
* User Engagement Metrics:
  + The system will track user engagement metrics such as likes, comments, shares, and watch time to improve AI recommendations.

**Appendices**

**Appendix A: Glossary**

* Trending Posts: Most liked and commented posts on the platform.
* AI Recommendations: Personalized content suggestions based on user behavior and preferences.

**Appendix B: Analysis Models**

* ER Diagram (User, Post, Comment relationships).
* Sequence Diagrams (User login, posting, commenting).