

# **Project Name: CineVerse**

**Date:** 24/11/2024

**Version:** 1.0

**By:** Aakash Choudhary

## **Table of Contents**

1. Introduction
  - 1.1 Product Scope
  - 1.2 Product Value
  - 1.3 Target Audience
  - 1.4 Intended Use
2. Functional Requirements
  - 2.1 User Authentication
  - 2.2 Search Functionality
  - 2.3 Task/Item Addition
  - 2.4 Review and Rating System
  - 2.5 Watchlist/Task List
  - 2.6 Social Features
  - 2.7 Recommendations
  - 2.8 Curated Lists
3. External Interface Requirements
  - 3.1 User Interface Requirements
  - 3.2 Hardware Interface Requirements
  - 3.3 Software Interface Requirements
  - 3.4 Communication Interface Requirements
4. Non-Functional Requirements
  - 4.1 Security
  - 4.2 Capacity
  - 4.3 Compatibility
  - 4.4 Reliability
  - 4.5 Scalability
  - 4.6 Maintainability

## 5. Definitions and Acronyms

# Software Requirement Specification

## 1. Introduction

**CineVerse** is a social networking platform for film enthusiasts where users can discover, rate, and review films, as well as interact with a community of like-minded people. Users can create watchlists, log films they've watched, and write reviews. Additionally, the platform offers curated lists, personalized recommendations, and the ability to follow other users to see their activity.

### 1.1 Product Scope

CineVerse is a movie-centric social networking site where users can discover, rate, and review films, as well as engage with a community of movie enthusiasts.

### 1.2 Product Value

The platform fosters community engagement and provides tools for organizing and sharing movie preferences. It enhances user experiences with personalized recommendations and curated lists.

### 1.3 Target Audience

- Movie enthusiasts who are seeking to track, review, and discuss films.
- Users looking for personalized recommendations and social features related to movies.

### 1.4 Intended Use

Users can log watched movies, create watchlists, write reviews, and interact with other like-minded movie enthusiasts.

## **2. Functional Requirements**

### **2.1 User Authentication**

Users can log in, register, and reset forgotten passwords securely.

### **2.2 Search Functionality**

Users can search for movies by title, genre, release date, or other filters.

### **2.3 Task/Item Addition**

Users can add new movies with details such as title, description, images, and categories.

### **2.4 Review and Rating System**

- Users can write, edit, or delete reviews for movies.
- Users can rate movies.
- Other users can interact with reviews, just as users can interact with their reviews.

### **2.5 Watchlist/Task List**

Users can create and manage personalized watchlists for tracking movies to watch later.

### **2.6 Social Features**

Users can follow other users, view their activity, and comment on their reviews.

### **2.7 Recommendations**

Personalized movie suggestions based on user preferences or genre.

### **2.8 Curated Lists**

Featured lists curated by the website or users, highlighting specific themes or categories.

## **3. External Interface Requirements**

### **3.1 User Interface Requirements**

- A responsive and user-friendly UI/UX designed in Figma.
- Clear navigation for movie search, reviews, and social features.

### **3.2 Hardware Interface Requirements**

Compatibility with mobile devices and standard desktop browsers.

### **3.3 Software Interface Requirements**

- **Frontend:** Developed using the Next.js framework for fast, scalable, and SEO-friendly applications.
- **Backend:** Flask REST APIs for handling data and requests to the server.
- **Database:** To store user data, movie details, and reviews.

### **3.4 Communication Interface Requirements**

Notifications sent via email for account-related updates.

## **4. Non-Functional Requirements**

### **4.1 Security**

Secure user authentication and encrypted storage for sensitive data.

### **4.2 Capacity**

Support for large datasets of movies and user activity logs.

### **4.3 Compatibility**

Cross-browser and cross-device compatibility.

#### **4.4 Reliability**

Ensure 99% uptime for a consistent and convenient user experience.

#### **4.5 Scalability**

Handle a sudden surge in users and maintain stability during concurrent sessions.

#### **4.6 Maintainability**

Modular codebase to facilitate updates, feature additions, bug fixes, and future scalability.

### **5. Definitions and Acronyms**

- **API:** Application Programming Interface
- **SEO:** Search Engine Optimization
- **UI:** User Interface
- **UX:** User Experience