

Software Requirements Specifications (SRS)

Project Name - **Letterboxd**
Prepared by - Naveen Srinivas
Date - 7 January 2025

1. Introduction	2
1.1 Purpose	2
1.2 Scope	2
1.3 Definitions	2
2. Overall Description	2
2.1 Product Perspective	2
2.2 Product Features	2
2.3 User Classes, Characteristics	2
2.4 Operating Environment	3
2.5 Design and Implementation Constraints	3
2.6 Assumptions and Dependencies	3
3. Functional Requirements	3
3.1 User Registration	3
3.2 Movie Browsing	3
3.3 Rating System	3
3.4 User Profiles	3
3.5 Community Features	3
4. Non-Functional Requirements	4
5. Other Requirements	4

1. Introduction

1.1 Purpose

The purpose of this document is to define **Letterboxd**, a platform to allow users to discover, rate and review movies.

1.2 Scope

Letterboxd is a web and mobile application designed for movie watchers to discover, rate and review movies and interact with like minded people

1.3 Definitions

- UI - User Interface
- API - Application Program Interface
- CRUD - Create, Read, Update, Delete

2. Overall Description

2.1 Product Perspective

It works on web and mobile platforms. It integrates with external APIs for movie database and supports user interaction

2.2 Product Features

- Discovery: Browse trending films
- User Profiles: Create accounts and manage watchlists
- Rating: Review and rate on a 5-star scale
- Watchlists: Create watchlists for personal use and sharing
- Social Features: Follow users and interact with them

2.3 User Classes, Characteristics

- General: Can browse movies, view public reviews, and interact with the community.
- Registered: Can write reviews, rate movies, create watchlists, and interact socially.
- Administrators: Manage the platform, moderate content, and handle technical issues.

2.4 Operating Environment

- Web: Compatible with browsers such as Chrome, Brave etc
- Mobile: Available on mobiles such as Android, iOS

2.5 Design and Implementation Constraints

- Compliance with web standards and mobile app guidelines
- Integration with external movie databases like IMDb
- Support for high traffic and concurrent users

2.6 Assumptions and Dependencies

- Users have access to stable internet connections
- External movie database APIs are functional and accessible

3. Functional Requirements

3.1 User Registration

- Users can login through email or third-parties like Google
- Login and Logout functions

3.2 Movie Browsing

- Search for movies by title
- View details of the movie

3.3 Rating System

- Rate movies on a 5-star scale
- Review movies

3.4 User Profiles

- Manage personal watchlist
- View personal history

3.5 Community Features

- Follow and interact with other users
- Comment, review and participate in discussions

4. Non-Functional Requirements

- Performance: The system must handle up to 100,000 concurrent users
- Security: Use SSL/TLS encryption for data transfer
- Usability: Provide an intuitive UI for both web and mobile platforms
- Scalability: Support future growth with scalable architecture
- Accessibility: Comply with WCAG 2.1 guidelines for accessibility.

5. Other Requirements

- APIs: Integrate with movie bases like IMDb