
Software Requirements Specification

for

FilmHub

Version 1.2 approved

Prepared by

Agile Industries Inc.

04/29/2020

Table of Contents

Table Of Contents	1
Introduction	2
1.1 Purpose.....	2
1.2 Intended Audience and Reading Suggestions.....	2
1.3 Project Scope.....	2
Overall Description.....	
System Features	
3.1 Sorting Movies by title, name, genre, actor, director.....	
3.2 Dynamically adding movies and genres to the home page.....	
3.3 Logging into a user account.....	
3.4 Creating an account	
3.5 Sending a “Forgot Password” email	
3.6 Thumb-up/Thumb-down a movie	
3.7 View a movie/ genre page.....	
3.8 “More Movie’s in this Genre” at bottom of movie page.....	
3.9 User profile page with user information.....	
3.10 Adding “favorite movies” to user profiles.....	
3.11 Error page for movie/genre page.....	
3.12 Admin’s adding a new movie or new genre to the website.....	
3.13 Different account permission levels (admin/user).....	

3.14 User requesting to add a new movie to the website.....	2
3.15 Dark mode.....	

1 Introduction

1.1 Purpose

The product we have is FilmHub, which is a collective catalog of movies, actors, and directors. Revision or release number is 1.2.

1.1.1 Vision

The aim of this product is to list a catalog of movies with their own actors and directors, and allow users to browse the list with the intent of leaving reviews and keeping track of their favorite movies.

1.1.2 Scope

The scope of this product includes a graphical user interface available through a website. Data from the top 25 movies on IMDB were gathered and stored in the program's dedicated database (Firestore). The algorithms in Firestore work with the data to allow sorting and searching.

1.2 Intended Audience and Reading Suggestions

This document was created for use by the project developers, clients, and project overseer. The information in this document is subject to change by authority of all stakeholders of the project -- including the project developers, clients, or project overseer -- following the introduction of new suggestions or product requirements.

1.3 Project Scope

FilmHub includes features that allow a user to browse movies from those included in the database. Users can access the product from a browser, create an account, search for a movie, and add it to their list of favorites.

2 Overall Description

2.1 Product Perspective

FilmHub is an online catalogue designed to allow users to browse, rate, and interact with their favorite movies, actors, directors, and genres. The product implements a Firebase database and through a javascript backend and HTML frontend renders a page of what the user is searching for. Users will be able to search for movies based on name, genre, actor, and director. Once on the movie page the user is able to rate (with a thumbs up or down), comment, and add the movie to their favorites.

2.2 Product Features

FilmHub implements the following features:

1. Search for movies by title, actor, director, and genre.
2. Rate movies through a thumbs up/down system.
3. Comment on a specific movie's page.
4. Login with an Email and password.
5. Basic users will be able to make requests to add movies to the database which can be approved by admins.
6. Different account levels with different features.
7. View upcoming film festivals.

2.3 User Classes and Characteristics

2.3.1 Customer:

Customers will more frequently use this website as a database for themselves to keep a log of the movies that they like and to find more movies that may be either by the same director or have the same actors.

2.3.2 DBA:

2.3.3 Data Entry Level Personnel:

2.4 Operating Environment:

The website will operate through the following software components:

Firebase will be where the database of all movies, actors, and directors are stored which will be managed by javascript and HTML to display it in the browser. All background operations will be done through javascript code which has been implemented in the HTML.

2.5 Design and Implementation Constraints

2.6 User Documentation

2.7 Assumptions and Dependencies

3. System Features

3.1 Sorting objects on the homepage

3.1.1 Objective - Users should be able to sort the database by what they're looking for, i.e. Sort by actors, genres, title, and directors.

3.1.2 Priority - This requirement is high priority.

3.1.3 Source - The source of this use case is the user profile.

3.1.4 Actors - Any user of the website would be the relevant actor for this use case.

3.1.5 Flow of Events -

3.1.5.1 Basic Flow -

3.1.5.1.2 - User navigates to the FilmHub home page

3.1.5.1.3 - System displays a list of everything in the database

3.1.5.1.4 - User clicks to display actors, genres, directors, movies, festivals

3.1.5.1.5 - System displays all actors, genres, directors, movies, or festivals respectively based on click.

3.1.6 Includes - None.

3.1.7 Requirements -

3.1.7.1 - The system displays all items in the database

3.1.7.2 - The system displays buttons for different sorting methods

3.1.7.3 - The system displays whatever items are selected from the buttons

3.1.8 Preconditions - None.

3.1.9 Post conditions - All elements link to the desired page.

3.1.10 Nonfunctional Requirements -

3.1.10.1 Performance Requirements - Sorting of items should be instant.

3.1.10.2 Safety Requirements - None.

3.1.10.3 Security Requirements - None.

3.1.10.4 Software Quality Requirements - None.

3.1.11 Notes/Issues - There is an issue with overlapping items when displaying them all on the home page.

3.1.12 Special Requirements - In order to fix the issue you need to turn off browser caching and refresh the page.

3.2 Dynamically adding movies to the homepage

3.2.1 Objective - Users should be able to always have a complete list of movies, genres, and actors available to them

3.2.2 Priority - This requirement is high priority.

3.2.3 Source - The source of this use case is the user profile.

3.2.4 Actors - Any user of the website would be the relevant actor for this use case.

3.2.5 Flow of Events -

3.2.5.1 Basic Flow -

3.2.5.1.2 - User navigates to the FilmHub home page

3.2.5.1.3 - System runs scripts that populate the homepage with movies, genres, actors, and festivals.

3.2.5.1.4 - System displays all actors, genres, directors, movies, or festivals respectively based on click.

3.2.6 Includes - None.

3.2.7 Requirements -

3.2.7.1 - The system displays all items in the database

3.2.7.2 - The system displays all items in proper category

3.2.7.3 - The system automatically updates the page based off of changes to the database

3.2.8 Preconditions - None.

3.2.9 Post conditions - All elements link to the desired page.

3.2.10 Nonfunctional Requirements -

3.2.10.1 Performance Requirements - Addition of items should be instant.

3.2.10.2 Safety Requirements - None.

3.2.10.3 Security Requirements - None.

3.2.10.4 Software Quality Requirements - None.

3.2.11 Notes/Issues - There is an issue with overlapping items when displaying them all on the home page.

3.2.12 Special Requirements - In order to fix the issue you need to turn off browser caching and refresh the page.

3.3 Logging into a user account

3.3.1 Objective - Users should be able to log into their personal FilmHub account

3.3.2 Priority - This requirement is high priority.

3.3.3 Source - The source of this use case is the user profile.

3.3.4 Actors - Any user of the website would be the relevant actor for this use case.

3.3.5 Flow of Events -

3.3.5.1 Basic Flow -

3.3.5.1.2 - User navigates to the FilmHub home page

3.3.5.1.3 - User clicks the drop down hamburger menu

3.3.5.1.4 - User clicks the login page

3.3.5.1.5 - System displays the login page for the user

3.3.5.1.6 - User types in their login information

3.3.5.1.7 - System logs them in to their account

3.3.6 Includes - None.

3.3.7 Requirements -

3.3.7.1 - The system properly logs in the user to their account

3.3.7.2 - Secure login and transfer of data is used

3.3.8 Preconditions - User has a valid account.

3.3.9 Post conditions - User is logged into the proper account.

3.3.10 Nonfunctional Requirements -

3.3.10.1 Performance Requirements - Logging in should be instant.

3.3.10.2 Safety Requirements - None.

3.3.10.3 Security Requirements - Secure transfer of data to the database.

3.3.10.4 Software Quality Requirements - None.

3.3.11 Notes/Issues - None.

3.3.12 Special Requirements - None.

3.4 Creating a user account

3.4.1 Objective - Users should be able to create a new personal FilmHub account

3.4.2 Priority - This requirement is high priority.

3.4.3 Source - The source of this use case is the user profile.

3.4.4 Actors - Any user of the website would be the relevant actor for this use case.

3.4.5 Flow of Events -

3.4.5.1 Basic Flow -

3.4.5.1.2 - User navigates to the FilmHub home page

3.4.5.1.3 - User clicks the drop down hamburger menu

3.4.5.1.4 - User clicks the login page

3.4.5.1.5 - System displays the login page for the user

3.4.5.1.6 - User clicks the create account button

3.4.5.1.7 - User types in new account information

3.4.5.1.7.1 - Name

3.4.5.1.7.2 - Email address

3.4.5.1.7.3 - Password

3.4.5.1.8 - System creates new account and logs them into the new account

3.4.6 Includes - None.

3.4.7 Requirements -

3.4.7.1 - The system properly processes creating accounts

3.4.7.2 - Secure transfer of data is used

3.4.8 Preconditions - None.

3.4.9 Post conditions - User is logged into the proper account.

3.4.10 Nonfunctional Requirements -

3.4.10.1 Performance Requirements - Creating an account should be instant.

3.4.10.2 Safety Requirements - None.

3.4.10.3 Security Requirements - Secure transfer of data to the database.

3.4.10.4 Software Quality Requirements - None.

3.4.11 Notes/Issues - None.

3.4.12 Special Requirements - None.

3.5 Sending forgot password email

3.5.1 Objective - Users should be able to have an email set to them so they can reset their password if it is forgotten.

3.5.2 Priority - This requirement is medium priority.

3.5.3 Source - The source of this use case is the user profile.

3.5.4 Actors - Any user of the website with an account would be the relevant actor for this use case.

3.5.5 Flow of Events -

3.5.5.1 Basic Flow -

3.5.5.1.2 - User navigates to the FilmHub home page

3.5.5.1.3 - User clicks the dropdown hamburger menu

3.5.5.1.4 - User clicks the login button

3.5.5.1.5 - User clicks the forgot password button and inputs their email

3.5.5.1.6 - The system send the user a reset password link to their email

3.5.6 Includes - None.

3.5.7 Requirements -

3.5.7.1 - The system sends a reset password link to the user

3.5.8 Preconditions - Already existing user account.

3.5.9 Post conditions - The user types in a new password that is valid.

3.5.10 Nonfunctional Requirements -

3.5.10.1 Performance Requirements - Sending of reset email should be instant.

3.5.10.2 Safety Requirements - None.

3.5.10.3 Security Requirements - Making sure the email gets sent to the proper person.

3.5.10.4 Software Quality Requirements - None.

3.5.11 Notes/Issues - None.

3.5.12 Special Requirements - None.

3.6 Thumbs up and thumbs down a movie

3.6.1 Objective - Users should be able to thumbs up and thumbs down a movie.

3.6.2 Priority - This requirement is medium priority.

3.6.3 Source - The source of this use case is the user.

3.6.4 Actors - Any user of the website would be the relevant actor for this use case.

3.6.5 Flow of Events -

3.6.5.1 Basic Flow -

3.6.5.1.2 - User navigates to any movie page on the website.

3.6.5.1.3 - User can thumbs up or thumbs down the movie

3.6.5.1.4 - System stores either thumbs up or thumbs down to user profile

3.6.6 Includes - None.**3.6.7 Requirements -**

3.6.7.1 - The system stores the thumbs up and thumbs down of the
user

3.6.8 Preconditions - User has a profile on the site.**3.6.9 Post conditions - The user thumbs up or thumbs down is saved.****3.6.10 Nonfunctional Requirements -**

3.6.10.1 Performance Requirements - None.

3.6.10.2 Safety Requirements - None.

3.6.10.3 Security Requirements - None.

3.6.10.4 Software Quality Requirements - None.

3.6.11 Notes/Issues - None.**3.6.12 Special Requirements - None.****3.7 View a movie or genre page**

3.7.1 Objective - Users should be able to view a movie or genre page with all data properly displayed.

3.7.2 Priority - This requirement is high priority.

3.7.3 Source - The source of this use case is the movie and genre classes.

3.7.4 Actors - Any user of the website would be the relevant actor for this use case.

3.7.5 Flow of Events -

3.7.5.1 Basic Flow -

3.7.5.1.2 - User navigates to any movie or genre page.

3.7.5.1.3 - System displays a list of everything in the related search.

3.7.6 Includes - None.

3.7.7 Requirements -

3.7.7.1 - The system displays all items in the related movie or genre page

3.7.8 Preconditions - None.

3.7.9 Post conditions - None.

3.7.10 Nonfunctional Requirements -

3.7.10.1 **Performance Requirements** - Loading of movie or genre pages should be instant.

3.7.10.2 Safety Requirements - None.

3.7.10.3 Security Requirements - None.

3.7.10.4 Software Quality Requirements - None.

3.7.11 Notes/Issues - None.

3.7.12 Special Requirements - None.

3.8 More movies in this genre

3.8.1 Objective - Users should be able to sort the database by what they're looking for, i.e. Sort by actors, genres, title, and directors.

3.8.2 Priority - This requirement is low priority.

3.8.3 Source - The source of this use case is the movie class.

3.8.4 Actors - Any user of the website would be the relevant actor for this use case.

3.8.5 Flow of Events -

3.8.5.1 Basic Flow -

3.8.5.1.2 - User navigates to any movie page

3.8.5.1.3 - System displays a list of recommended movies in a similar

genre

3.8.6 Includes - None.

3.8.7 Requirements -

3.8.7.1 - The system displays related movies in a related genre

3.8.8 Preconditions - None.

3.8.9 Post conditions - None.

3.8.10 Nonfunctional Requirements -

3.8.10.1 Performance Requirements - 3 related movies should be displayed at the bottom of the page.

3.8.10.2 Safety Requirements - None.

3.8.10.3 Security Requirements - None.

3.8.10.4 Software Quality Requirements - None.

3.8.11 Notes/Issues - There is an issue with duplicate movies showing up and no movies showing up if there are less than 3 movies in that related genre.

3.8.12 Special Requirements - None.

3.9 User profile page

3.9.1 Objective - Users should be able to view all of their profile information.

3.9.2 Priority - This requirement is high priority.

3.9.3 Source - The source of this use case is the user profile.

3.9.4 Actors - Any user of the website would be the relevant actor for this use case.

3.9.5 Flow of Events -

3.9.5.1 Basic Flow -

3.9.5.1.2 - User navigates to the user page after login in.

3.9.5.1.3 - System displays a list of all relevant information for the specific

user

3.9.6 Includes - None.

3.9.7 Requirements -

3.9.7.1 - The system displays all user information.

3.9.8 Preconditions - User has an account.

3.9.9 Post conditions - None.

3.9.10 Nonfunctional Requirements -

3.9.10.1 Performance Requirements - System should display the correct user info.

3.9.10.2 Safety Requirements - None.

3.9.10.3 Security Requirements - None.

3.9.10.4 Software Quality Requirements - None.

3.9.11 Notes/Issues - None.

3.9.12 Special Requirements - None.

3.10 Adding favorite movies

3.10.1 Objective - Users should be able to add any movie to their list of favorite movies.

3.10.2 Priority - This requirement is medium priority.

3.10.3 Source - The source of this use case is the user profile.

3.10.4 Actors - Any user of the website would be the relevant actor for this use case.

3.10.5 Flow of Events -

3.10.5.1 Basic Flow -

3.10.5.1.2 - User navigates to any movies page.

3.10.5.1.3 - User clicks add to favorites button.

3.10.5.1.4 - System adds the users selection to their list of favorite movies.

3.10.6 Includes - None.

3.10.7 Requirements -

3.10.7.1 - The system records the added movie in the database

3.10.8 Preconditions - User already has an account.

3.10.9 Post conditions - None.

3.10.10 Nonfunctional Requirements -

3.10.10.1 Performance Requirements - Movies should be added to the correct users list of favorite movies..

3.10.10.2 Safety Requirements - None.

3.10.10.3 Security Requirements - None.

3.10.10.4 Software Quality Requirements - None.

3.10.11 Notes/Issues - None.

3.10.12 Special Requirements - None.

3.11 Error page

3.11.1 Objective - System should display an error page if the requested page does not exist.

3.11.2 Priority - This requirement is low priority.

3.11.3 Source - The source of this use case is the system.

3.11.4 Actors - Any user of the website would be the relevant actor for this use case.

3.11.5 Flow of Events -

3.11.5.1 Basic Flow -

3.11.5.1.2 - User navigates to a non-existent movie or genre page

3.11.5.1.3 - System displays an error 404 page

3.11.6 Includes - None.

3.11.7 Requirements -

3.11.7.1 - The system displays the error page at the correct time

3.11.8 Preconditions - User navigates to a movie or genre page that doesn't exist.

3.11.9 Post conditions - None.

3.11.10 Nonfunctional Requirements -

3.11.10.1 Performance Requirements - Error page should display only when a page is not found.

3.11.10.2 Safety Requirements - None.

3.11.10.3 Security Requirements - None.

3.11.10.4 Software Quality Requirements - None.

3.11.11 Notes/Issues - None.

3.11.12 Special Requirements - None.

3.12 Admins adding new movies or genres to the website

3.12.1 Objective - Admins can add movies or genres to the website from the list of user requested movies and genre.

3.12.2 Priority - This requirement is high priority.

3.12.3 Source - The source of this use case is the admin profile.

3.12.4 Actors - Any admin of the website would be the relevant actor for this use case.

3.12.5 Flow of Events -

3.12.5.1 Basic Flow -

3.12.5.1.2 - Admin navigates to the movie and genre requests page

3.12.5.1.3 - Admin can confirm or deny user requests

3.12.5.1.4 - On approval, the system added the new movie or genre to the database

3.12.6 Includes - None.**3.12.7 Requirements -**

3.12.7.1 - The system adds approved movies or genres to the database

3.12.7.2 - The system deletes unapproved requests

3.12.8 Preconditions - User is an admin.**3.12.9 Post conditions - Movie or genre get properly added to the database.****3.12.10 Nonfunctional Requirements -**

3.12.10.1 Performance Requirements - Approve or deny requests.

3.12.10.2 Safety Requirements - None.

3.12.10.3 Security Requirements - None.

3.12.10.4 Software Quality Requirements - None.

3.12.11 Notes/Issues - None.**3.12.12 Special Requirements - None.**

3.13 Different account levels and permissions

3.13.1 Objective - There is a distinction between actions that can be taken by users and admins.

3.13.2 Priority - This requirement is high priority.

3.13.3 Source - The source of this use case is the user and admin profile.

3.13.4 Actors - Any user of the website would be the relevant actor for this use case.

3.13.5 Flow of Events -

3.13.5.1 Basic Flow -

3.13.5.1.2 - Depending on user permission level, different actions can be taken on the site

3.13.6 Includes - None.

3.13.7 Requirements -

3.13.7.1 - The system has the proper knowledge on how to differentiate between users and admins

3.13.8 Preconditions - Account type.

3.13.9 Post conditions - Access given to the account.

3.13.10 Nonfunctional Requirements -

3.13.10.1 **Performance Requirements** - Permissions should be granted accordingly.

3.13.10.2 Safety Requirements - None.

3.13.10.3 Security Requirements - None.

3.13.10.4 Software Quality Requirements - None.

3.13.11 Notes/Issues - None.

3.13.12 Special Requirements - None.

3.14 User requesting to add a new movie

3.14.1 Objective - Users should be able to send a request to the admins to add a new movie.

3.14.2 Priority - This requirement is medium priority.

3.14.3 Source - The source of this use case is the user profile.

3.14.4 Actors - Any user of the website would be the relevant actor for this use case.

3.14.5 Flow of Events -

3.14.5.1 Basic Flow -

3.14.5.1.2 - User navigates to the request movie page

3.14.5.1.3 - User inputs relevant movie information and submits

3.14.5.1.4 - System sends request to the admin accept or deny page

3.14.6 Includes - None.

3.14.7 Requirements -

3.14.7.1 - The system stores and displays each request to all admin accounts

3.14.8 Preconditions - User has an account.

3.14.9 Post conditions - None.

3.14.10 Nonfunctional Requirements -

3.14.10.1 Performance Requirements - Request should be sent to the admins.

3.14.10.2 Safety Requirements - None.

3.14.10.3 Security Requirements - None.

3.14.10.4 Software Quality Requirements - None.

3.14.11 Notes/Issues - Users can add multiple actors to the request.

3.14.12 Special Requirements - None.

3.15 Dark theme

3.15.1 Objective - Users should be able to toggle between a light mode and dark mode on the website.

3.15.2 Priority - This requirement is lowpriority.

3.15.3 Source - The source of this use case is the user profile.

3.15.4 Actors - Any user of the website would be the relevant actor for this use case.

3.15.5 Flow of Events -

3.15.5.1 Basic Flow -

3.15.5.1.2 - User navigates to hamburger menu on any page of the website

3.15.5.1.3 - User clicks the “Change Theme” button in the menu

3.15.5.1.4 - System toggles between display mode

3.15.6 Includes - None.**3.15.7 Requirements -**

3.15.7.1 - The system toggles between the light mode and dark mode

3.15.8 Preconditions - None.**3.15.9 Post conditions - The page changes theme.****3.15.10 Nonfunctional Requirements -**

3.15.10.1 Performance Requirements - The theme should toggle back and forth on each click.

3.15.10.2 Safety Requirements - None.

3.15.10.3 Security Requirements - None.

3.15.10.4 Software Quality Requirements - None.

3.15.11 Notes/Issues - None.**3.15.12 Special Requirements - None.**

