

FLIXMIX

Requirements Specifications



TeamBYR

Brandon Christensen, Roshani Shiwakoti, Yuta Mogi

Course: CptS 322 - Software Engineering Principles I

Instructor: Sakire Arslan Ay

TABLE OF CONTENTS

I.	INTRODUCTION.....	3
II.	REQUIREMENTS SPECIFICATION	3
II.1.	CUSTOMER, USERS, AND STAKEHOLDERS.....	3
II.2.	USE CASES	3
II.3.	NON-FUNCTIONAL REQUIREMENTS	9
III.	USER INTERFACE REQUIREMENTS	10
IV.	REFERENCES.....	10
APPENDIX: GRADING RUBRIC		ERROR! BOOKMARK NOT DEFINED.

I. Introduction

Our group has decided to create our own web application called “FlixMix”. From the name, you can possibly guess that it will be a movie discussion board of sorts. We want an app where users can login and keep track of movies they have seen as well as movies they plan to watch. Users will be able to make their accounts public or private and will be remembered in a database. The app will implement a movie database that users will be able to sort and search in. We want the movies to be sortable by and to display relevant information, i.e. title, description, tags, genres, etc... If we cannot find a movie database, then instead we will allow the users to manually enter in this information. The app will have a rating system as well as a discussion/comment section to discuss specific films. We also would like to implement a weekly movie recommendation, where a movie is recommended automatically site wide.

The following section (section II) will include the requirement specifications for our application. This will contain information about our target customers, users, and stakeholders. We will also discuss the different use cases, their fields, as well as any non-functional requirements we plan on addressing. Next, in section III, we will include different design elements, and sketches to give an idea of what our final product will look like. Specifically, this section will focus on user interface design. Finally, section IV will cite any references that we plan to use in our creation of our application. It is important to give credit to any external resources that we find and integrate.

Document Revision History

Rev 1.0 2020-10-14 Initial Version

II. Requirements Specification

II.1. Customer, Users, and Stakeholders

For our application, the customer, user, and stakeholders are all part of the same group. We (our group) can be seen as also being a part of the stakeholder group, since doing well on this project will benefit our grades and improve our skills for the future. However, other than us, the customer, user and stakeholder all belong to the same group and we will design the application with that in mind. The customers that “buy” our application will also be the users using our service. The user will also be the one to benefit from the service, as the application offers a service to them, therefore making them a stakeholder too. Since the purpose of the application is to be a discussion board and movie watchlist, then just using the application makes you a member of all three groups.

II.2. Use Cases

Use case # 1

Name	Movie Register
------	----------------

Users	All Users (who have an account)
Rationale	When registering a movie, users must enter a “title”, “description of movie”, “tag”, and “genres”. Users are given the choice to add a movie poster, a movie rating, and an already watched flag.
Triggers	The user selects the “Movie Register” button
Preconditions	User is logged in
Actions	<ol style="list-style-type: none"> 1. The user chooses the option to create a new movie. 2. The application responds by requesting the user to enter movie information. 3. The user inputs the movie information 4. The application registers the data into the database.
Postconditions	Movie information is correctly stored in the database.
Acceptance Tests	Make sure all movie information is correct.
Iteration	Back to Index page.

Use case # 2

Name	Movie Sort
Users	All Users (who have an account)
Rationale	When sorting a movie, users have the option to sort by either a “title”, “description of movie”, “tag”, or “genres”.
Triggers	The user selects the “sorting” option in a sort by drop down

Preconditions	User is already logged in and has at least one movie in the database.
Actions	<ol style="list-style-type: none"> 1. The user is logged in and chooses the option to sort movies. 2. The application responds by requesting for the movie sorting information. 3. Application displays movies sorted by given sorting information.
Postconditions	Movies are displayed correctly by given sort field
Acceptance Tests	Test if the movies are sorted in the correct order.
Iteration	Back to Index page.

Use case # 3

Name	Hide/Show Account Information
Users	All Users (who have an account)
Rationale	The user chooses an option to “Hide” or “Show” their account information, then the account type changes. If the user selects hide, then this user's movies and account information will not show up on any other user's application. If the user selects show, then information and movies will show up.
Triggers	The user selects the “/information/hide-show” option
Preconditions	User is already logged in
Actions	<ol style="list-style-type: none"> 1. The user is logged in and goes to account settings 2. The user chooses the option to “Hide” or “Show” their account publically.

	<p>2. The application responds by asking which option they would like selected.</p> <p>3. If users select the “hide” option, the account will be set to private.</p> <p>4. If users select the “show” option, the account will be set to public.</p>
Postconditions	User’s information changes to public or private based on the user’s selection.
Acceptance Tests	Make sure displayed account information is changed to what user requested
Iteration	Back to Index page.

Use case # 4

Name	Movie discussion
Users	All Users (who have an account)
Rationale	When a user goes to the movie details page, there is an existing discussion page. If a user searches for a specific movie, then transfer to that movie’s page. Here users can comment and discuss the movies. Create a thread of comments, and display a comment thread.
Triggers	The user selects the “/movie/discussion” option, or the user selects a movie from movies sort.
Preconditions	User is already logged in
Actions	<p>1. The user logs in to their account and chooses a movie.</p> <p>2. The application responds by sending the user to that movies page.</p> <p>3. Users are able to comment in the discussion.</p> <p>4. If a user comments on the discussion page, the application registers the information into the database.</p>

Postconditions	Movie information exists in the database.
Acceptance Tests	Make sure movie information and discussion for that movie is correct.
Iteration	After commenting in the discussion, refresh the page (to register the comment).

Use case # 5

Name	Movie Recommendation
Users	Administrator
Rationale	When a user goes to the home page, show a recommended movie from the database (this changes every week).
Triggers	The user views the home screen
Preconditions	Movie information is registered to the database already.
Actions	<ol style="list-style-type: none"> 1. When a user goes to this page, the application will check the database to choose a recommended movie. 2. The application will then display the recommended movie
Postconditions	Movie recommendation displayed on the home page.
Acceptance Tests	Make sure the recommended movie's information is shown on the page.
Iteration	Back to Index page.

Use case # 6

Name	Movie Watchlist
Users	Users (that have an account)
Rationale	When a user goes to their watchlist page, they will have a list of movies that they added to watch later.
Triggers	The user clicks a “watchlist” button that redirects them to a watchlist page
Preconditions	Users must have an account.
Actions	<ol style="list-style-type: none">1. The user logs in2. The user clicks the watchlist button3. The user is redirected to a watchlist page4. If the user has movies already added to this list, then the movies are displayed5. If the user wants to add a movie to this list, there will be a search bar that the user can use to find a movie and select it6. Selected movie will be added to a database that is connected to the user account
Postconditions	Movies that were added are listed to this page
Acceptance Tests	All movies that are in the watchlist list are displayed
Iteration	Refresh to watchlist

Use case # 7

Name	Create new user/ Login
------	------------------------

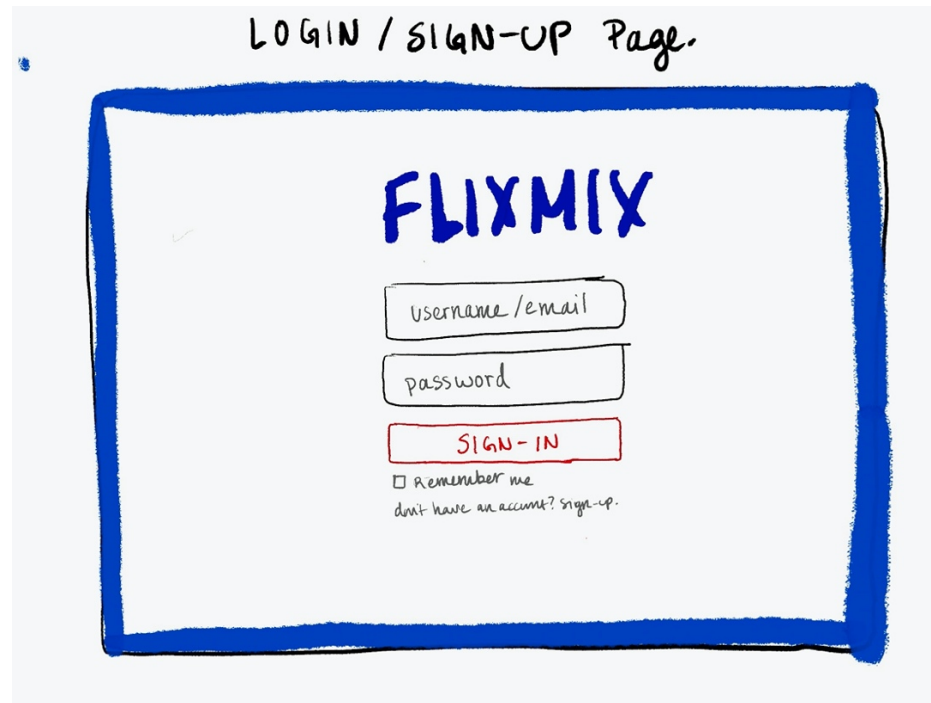
Users	New users that do not have an account yet / Users that haven't logged in yet
Rationale	This is how a user will create an account to use our application. There will be a login page where users before users can view anything, and they must either login or create an account from there
Triggers	Create new button goes to a create new page. Login button goes to the home page
Preconditions	None / have an account already
Actions	1. If the user has an account already, enter username and password in text boxes, if correct then go to the home page. 2. If the user does not have an account, click create new button. 3. Redirect to a create new page, asking for a username and password 4. Add new user (username and password) to database
Postconditions	User is correctly added to the database. User cannot login
Acceptance Tests	See if the user can now login
Iteration	Refresh back to login in screen

II.3. Non-Functional Requirements

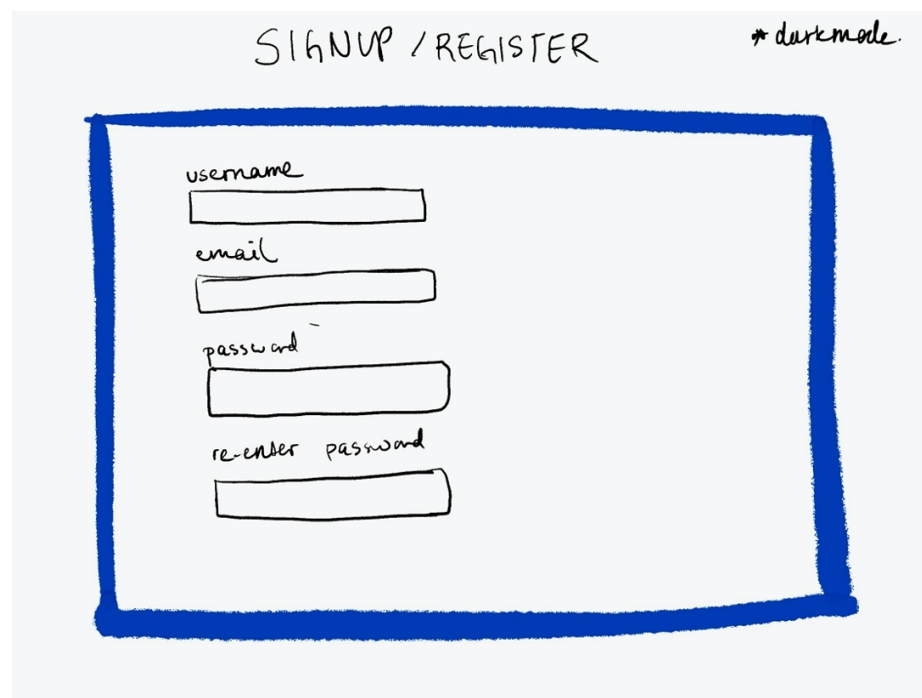
1. Application should scale to any size screen
2. Will use multiple databases, one for users and one for movies
3. Should handle at least a classroom size amount of users (100)
4. Each user should be allowed to at least add 100 movies to watchlist

III. User Interface

Here we display and describe a tentative user interface for our website, this UI may be changed, reviewed, and redesigned as seen fit.

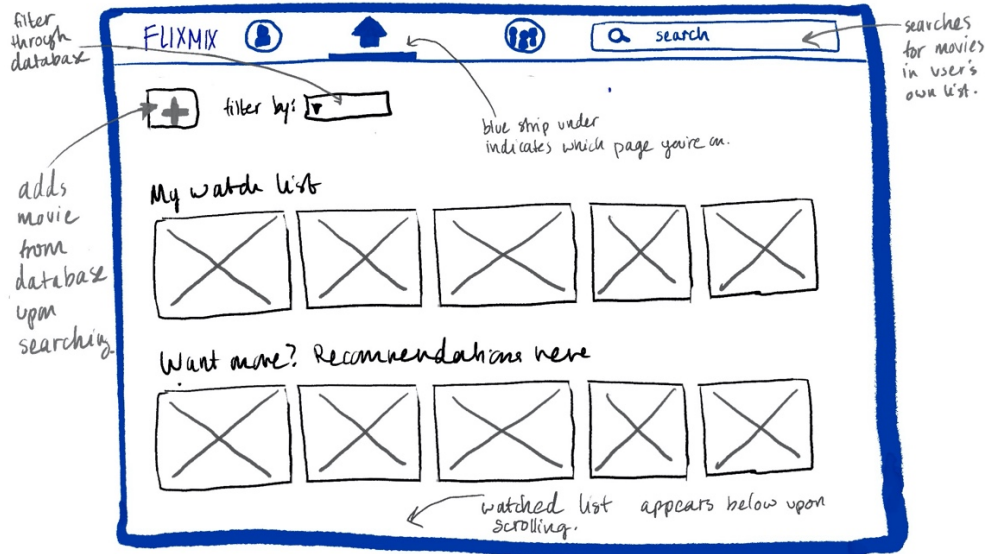


Log-in page, user is able to log-in to their account and is given the option to sign up if they do not already have an account



Sign-up page, user can sign up and create an account to use the service. Note, the website will be dark mode by default

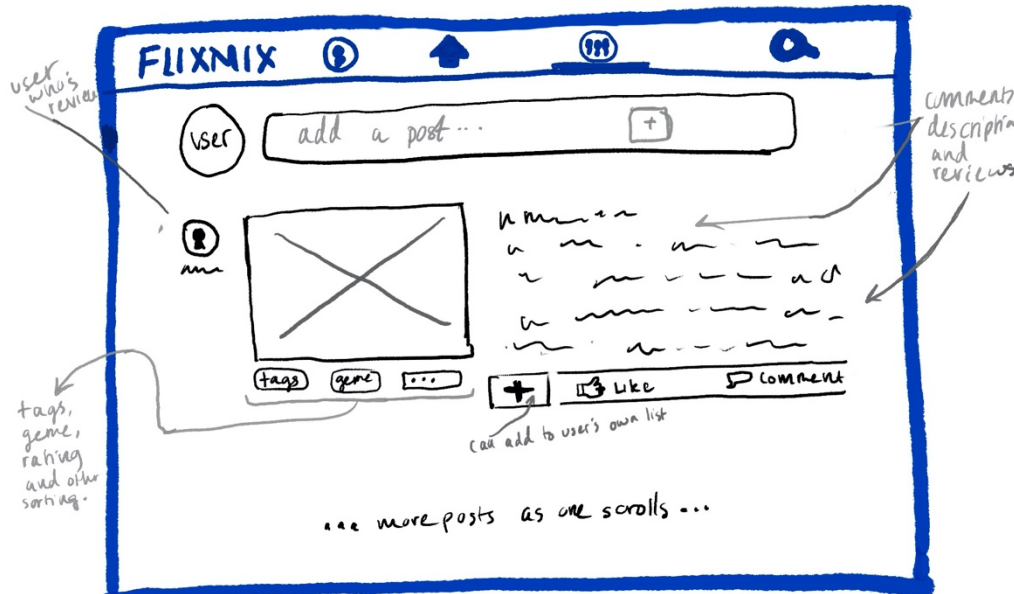
LANDING PAGE user's profile.



darkmode :)

Loading page, where the user first lands upon logging on to their account. This gives a list of their watch lists, recommendations, archives, and allows the user to add movies from the database and filter depending on genre/tag/etc.

DISCUSSION FORUM



Discussion forum, user can see posts from other users, and friends, view reviews, recommendations, and communicate with other movie goers about the movies. User can also add movies that others have watched to their list.

IV. References

N/A