

Use Cases

User Related

- Sign Up (Completed)
- Login (Completed)
- Log Out (Completed)
- View Profile (Completed)
- Edit Profile (Completed)
- Delete Account (Completed)
- Reset Password (Not Started)

Follower Related

- Identify Friends (Completed)
- Prod Follower (Completed)
- View Recommendations: Prods (Completed)

Movie Related

- Search Movies (Completed)
- Read Movie Information (Completed)
- Rate and Review Movies (Completed)
- View Playlists (Completed)

Admin Related

- View Logs (Completed)

Use Case:	Sign Up
Primary Actor:	End-User
Goal in Context:	We want to allow users to sign up for our platform.
Preconditions:	<ul style="list-style-type: none"> • User has an email address • The email used for sign up must not be in the database already. • A user must not be logged in
Trigger:	Select the Sign-up for account button on the home page
Scenario:	A new user navigates to the home page of the site. After seeing an interesting homepage describing what the site is like they decide to register for a new account.
Exceptions:	<ul style="list-style-type: none"> • If they enter an email for a new account that has already been used they receive an error message. • If the requested password or email is empty they receive an error message. • If they do not specify their favorite movie Genre they will receive an error message.

Status: **Completed**

Use Case:	Login
Primary Actor:	End-User / Admin
Goal in Context:	As an end user or an admin they must be able log in to the site.
Preconditions:	<ul style="list-style-type: none"> User already registered for an account
Trigger:	Select the login button
Scenario:	A registered user navigates to the home page of the site. They select the login button.
Exceptions:	If you are already logged in you can't log in again If you have not registered your account you can't login

Status: **Completed**

Use Case:	Log out
Primary Actor:	End User/Admin
Goal in Context:	When a user has decided to end their session they can log out.
Preconditions:	Must be logged into the site.
Trigger:	Select the logout button
Scenario:	On any page of the site a user can use a menu to select the Log out button
Exceptions:	If you have already logged out you can't log out again

Status: **Completed**

Use Case:	View any Profile
Primary Actor:	End User/Admin
Goal in Context:	A User must be able to view their profile so they can set the information publicly available to others. They must also be able to view another users profile.
Preconditions:	Be logged in
Trigger:	Select the view profile button
Scenario:	On any page of the site a user can use a menu to select the view profile button. On movie detail pages a user can click on a reviewers name to go to their profile
Exceptions:	If you are already on the profile page you won't see the link in the menu.

Status: **Completed**

Use Case:	Edit Profile
Primary Actor:	End User/Admin
Goal in Context:	A user should be able to edit the information in their profile. An Admin user should be able to edit anyone's profile.
Preconditions:	Must be logged in Must be on the profile page
Trigger:	Select the edit profile button which allows you to update profile fields
Scenario:	Once a user navigates to the view profile page they will be able to select an edit profile button, this will make the profile fields editable.
Exceptions:	You cannot edit the email address of a profile.

Status: **Completed**

Use Case:	Delete account
Primary Actor:	End User/Admin user
Goal in Context:	Users are able to delete their accounts when they no longer want to use the service. Admins should be able to delete unwanted user accounts.
Preconditions:	<ul style="list-style-type: none"> • End user account must exist.
Trigger:	Admin selects 'Delete Account' on an end user's profile.
Scenario:	<ul style="list-style-type: none"> • Admin user logs in. • Admin user navigates to an end user's profile. • Admin user triggers use case. • Admin user confirms the deletion. • End user account is deleted.
Exceptions:	If the end user account has been deleted by another admin after navigating but before confirming deletion

Status: **Completed**

Use Case:	Reset password
Primary Actor:	Admin user or end user
Secondary Actor:	End user
Goal in Context:	Create a new password for the end user.
Preconditions:	<ul style="list-style-type: none"> • End user account must exist.
Trigger:	Primary selects 'Reset Password' button on end user profile. End user selects 'Forgot Password' at login.
Scenario:	<ul style="list-style-type: none"> • Admin user logs in. • Admin user navigates to end user profile. • Admin user triggers use case. • End user receives notification that their password has been reset, and a link to enter a new one. • End user logs in. • End user navigates to 'Edit Profile' page. • End user triggers use case. • End user enters new password. • End user navigates to login page. • End user triggers use case. • End user submits username or email. • End user receives notification that their password has been reset, and a link to enter a new one.
Exceptions:	If user is trying to reset password without logging in, but account cannot be found, notify them of this and do not send notifications.

Status: Not Started

Use Case:	Identify friends
Primary Actor:	All users
Goal in Context:	To find the profiles of other users in the system that the user knows, or profiles that the user is interested in keeping up with.
Preconditions:	The user must have an account, be logged in, and must have a desire to find other users in the system. The user must have an interest in the activities of other users.
Trigger:	Selecting a function to search for users in the system. Selecting an Add Friend function on another user's profile.
Scenario:	<ul style="list-style-type: none"> • User enters a potential name of another user. • User views entries of other user profiles in the system. • User selects to confirm the addition of a friend.
Exceptions:	<ul style="list-style-type: none"> • Overwhelmingly large search results. The user will need to search more specifically or potentially navigate a lot of information. • The system will not accept contentless queries.

Status: **Completed**

Use Case:	Prod follower
Primary Actor:	End user
Secondary Actor:	End user 2
Goal in Context:	Users should be able to recommend movies to their people who follow them.
Preconditions:	<ul style="list-style-type: none"> • User must be logged in. • User must have at least one follower. • User must be on a movie information page.
Trigger:	A user selects the 'Prod Friend' button on a movie page.
Scenario:	<ul style="list-style-type: none"> • User logs in. • User searches for a movie. • User triggers the use case. • User selects a friend (end user 2) to prod. • User confirms the prod. • End user 2 received the recommendation.
Exceptions:	<p>If user is not logged in, they should be prompted to log in.</p> <p>If user has no friends, they should be notified and prompted to add friends.</p>

Status: **Completed**

Use Case:	View recommendations (Prods)
Primary Actor:	End user
Goal in Context:	Users should be able to receive recommendations for movies they may like.
Preconditions:	<ul style="list-style-type: none"> • User must be logged in.
Trigger:	User selects the 'Notifications' button.
Scenario:	<ul style="list-style-type: none"> • User logs in. • User triggers the use case. • User is shown movie recommendations. • If user has been prodded, show those recommendations.
Exceptions:	If user is not logged in, they should be prompted to log in.

Status: **Completed**

Use Case:	Search movies
Primary Actor:	End user
Goal in Context:	Find a particular movie's information page.
Preconditions:	<ul style="list-style-type: none"> • User must be logged in. • User must navigate to search page.
Trigger:	User enters a value into a search field.
Scenario:	<ul style="list-style-type: none"> • User logs in. • User navigates to search page. • User triggers use case. • User receives a filtered list of movies based on the value entered into the search field.
Exceptions:	If the user enters a value that returns no results, they should be notified that there are no results and prompted to enter a different search term.

Status: **Completed**

Use Case:	Read movie information
Primary Actor:	All users
Goal in Context:	To gain insight about movies that the user may or may not be interested in watching, or have an opinion about as already having watched it.
Preconditions:	User must have an account and be logged in. User must be able to find movie entries in the system.
Trigger:	Selecting a movie entry from a list, or by recommendation from another user in the system.
Scenario:	<ul style="list-style-type: none"> • User finds a movie title interesting in a list, or by recommendation. • User selects the movie and is shown detailed information about the movie. • User views, in addition to general movie information, ratings and reviews from other users.
Exceptions:	<ul style="list-style-type: none"> • User may fail to locate a movie they're interested in. • The system will need to strive to keep information accurate and not duplicated. This may become a task for administrative users.

Priority:	This use case has high importance as one of the main attracting functions of the system.
Secondary Actor:	Administrative users An external movie information database
Channels to Secondary Actors:	<p>Administrative users may need to maintain movie entry accuracy, and be sure duplicate movies or false movies are not in the system. They'll do so with the application's web interface.</p> <p>An external database will be needed to gather movie titles and other information to display to users. This interaction will happen from the system's backend via web APIs.</p>
Open Issues:	The need for an administrative function to viewing movie information is a secondary thought as a way for the system to be maintained or corrected, but it isn't yet fully clear if this role is necessary.

Status: **Completed**

Use Case:	Rate and review movies
Primary Actor:	All users
Goal in Context:	To express their opinion about a movie to the platform community and their friends.
Preconditions:	User must have an account, be logged in, and be able to navigate to the movie they'd like to review. User must also have an opinion about the movie and a desire to share this with their friends and/or the community.
Trigger:	Selecting the write a review button.
Scenario:	<ul style="list-style-type: none"> • User views a form in the context of a movie with two fields: rating and review • User selects a rating • User writes a long-form text review
Exceptions:	<ul style="list-style-type: none"> • The system will not allow ratings outside of a certain range (i.e. 1-5). It should be impossible for the user to get into this state. • The system may have limits on length of reviews. There will need to be error states explaining to users when they've reached this limit.

Priority:	This has necessarily lower priority than crucial functions such as login, but we place it with moderate priority.
Channel to actor:	The system's web interface
Secondary Actor:	Secondary actors are other users or friends of the user that are interested in reading the review or viewing an average user rating. This falls in line with the "Read movie information" use case.
Channels to Secondary Actors:	<the means through which any secondary actors may interact with the system or other actors>

Status: **Completed**

Use Case:	View playlists
Primary Actor:	All users
Goal in Context:	To explore movies that the user may or may not know about.
Preconditions:	User must have an account and be logged in. The system must have average ratings of movies internally and externally. The user may or may not know that they want this use case, but it is a primary function of the system and will be derived for the user without their action.
Trigger:	Playlists will be displayed deliberately to the user. The user will have to do little or nothing to find this information.
Scenario:	<ul style="list-style-type: none"> • User navigates to a playlist and is shown the curation much like a gallery or set of search results. • User selects the movie and is shown detailed information about the movie. This may transition to the rate/review movie use case. • User views, in addition to general movie information, ratings and reviews from other users.
Exceptions:	<ul style="list-style-type: none"> • Overwhelmingly large results. The user will need to search more specifically or potentially navigate a lot of information. The system should strive to avoid this.

Priority:	While this is a use case the relies on most other use cases working before it can work, it maintains relatively high priority as it is a crucial part of the value of the system.
Secondary Actor:	An external system for getting critics' ratings of the movies will be necessary for at least part of the algorithmic work needed.
Channels to Secondary Actors:	The system's backend will communicate with external services via web APIs.
Open Issues:	Defining specifics of algorithms to generate playlists

Status: **Completed**

Use Case:	View Logs
Primary Actor:	Admin
Goal in Context:	We want to allow users Admin users to easily view system logs in case a user reports an error or unexpected behavior
Preconditions:	<ul style="list-style-type: none"> • User is logged in • User has admin access
Trigger:	Select the 'View Logs' button on the Nav bar.
Scenario:	An admin sees that an account cannot follow a user, to confirm there is an issue and check the payload of the failed request they go to logs tab of the site and they are able to view a complete list of requests.
Exceptions:	<ul style="list-style-type: none"> • If you are not an Admin you cannot see the logs

Status: **Completed**