Requirements Definition

*Movshare*

1. **Introduction and Context**

Oftentimes, someone will purchase a movie or book simply so they can read/watch it once, only to find out that a friend already has it. Alternatively, someone may own a popular movie and be constantly inundated with requests from friends to borrow it, making it difficult to keep track of who is in possession at any given time.

This project aims to build an app that helps people manage and keep track of different kinds of media owned by themselves and others. In order to reduce duplicate purchases of movies and other media, this app will allow a group of users to store titles and details relating to their media collection. Each user will them be able to search for titles in each other’s libraries in order to share this media amongst themselves. This would also allow a user to purchase a gift for a friend without worrying about whether that friend already has the gift. The system will also allow for better record-keeping and accountability when loaning out titles.

1. **Users and their Goals**

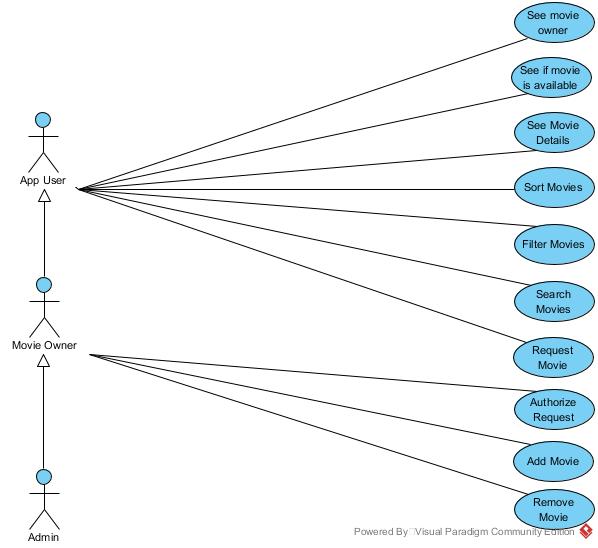
This project has at most three users: a normal user, a movie owner, and an admin. A movie owner has access to all that a user has access to. An admin has access to everything both a user and a movie owner has access to. Therefore, the projects users have the following hierarchy: An admin is a movie owner, who is a normal user.

Normal User. A normal user needs to be able to request a movie from the database. They also need to see if the movie is available for borrowing, as well as have a tentative date when the movie will become available. They need to be able to select a movie and see details on that specific movie. They also need to manipulate the display of the database with the following functionality namely sort and filter by various categories (i.e. rating, title, genre, etc.).

Movie owner. A movie owner needs to be able to do all a normal user can do in addition to adding and removing movies they own from the database. They also need to be able to approve movie requests made by normal users for a movie they own.

Admin. An admin is there to be able to edit all aspects of the database regardless of who owns the movie. This account would be separate from the other accounts and would need to have a normal user account to use the program as intended.

Use Case Diagram.



1. **Functional Requirements**
2. User Authentication
   1. The system must require all users to authenticate themselves before giving them access to view or take actions on Movshare.
      1. They must be able to create an account with a username and password the first time they access Movshare
      2. They must be able to log in on subsequent visits to Movshare using the same username and password. If they log in successfully, they should gain access to Movshare. If they did it incorrectly, they should be prompted to try again.
3. User Libraries
   1. The system should allow users to add media entries to their User Library.
      1. When an entry is created, users will be required to list a title, media format, and whether the item is available. They should also have the option of list additional information under an ‘additional notes’ category. Entries will also have a ‘scheduled loans’ category.
         1. If the item is unavailable, the user should be prompted (but not required) to enter an estimated availability date.
   2. After entry creation, the system should allow users to edit any element of a media entry.
      1. The system should allow users to edit anything in the entry by using an ‘edit entry’ button.
      2. The system should allow users to quickly change the availability of the entry via a ‘loan entry’ button. This will allow them to change the estimated availability date, and optionally provide the name of the person they are loaning the media to.
4. Search and Request Function
   1. The system should allow users to browse other user’s libraries.
      1. Users should be able to browse libraries by user.
      2. Users should be able to search all libraries by title and media format.
      3. Users should be able to filter search results by availability.
      4. Searches should show all information stored in the media entry in a search.
   2. The system should allow users to ‘request’ a title from another user.
      1. When requesting a title, the requester will be prompted to enter a date range and a message.
      2. The media owner should receive an alert containing the date range and the message entered by the requester.
         * 1. The media owner should be able to accept or deny the request.

If the request is accepted, the system should add the date range to the ‘scheduled loans’ category of the requested media.

* + - * 1. The system should send an alert to the requester informing them whether the owner refused or denied their request.

1. **Non-functional Requirements**

* Development will follow an Agile/Scrum-like process, a Scrum board and Sprints lasting roughly a month. Scrum meetings will occur on Mondays, Wednesdays, and Fridays after class.
* Team members will use Slack to communicate in-between scrum meetings.

*This section contains requirements that describe and/or constrain the development process. For example, a non-functional requirement may state that development will follow an Agile method with weekly iteration meetings.*

1. **Future Features**
2. Integration with a mobile app that includes support for scanning barcodes. Scanning the barcode of a piece of media might allow a user to search for that media in other user libraries, or to add the media to their own library.
3. A wishlist function.
4. **Glossary**

User – A generic person who uses Movshare.

Media – A movie, book, magazine etc. that is owned by a user

Media Collection – The physical collection of all pieces of media owned by a user

User Library – A database that users may add media titles to, which may then be searched by other users.