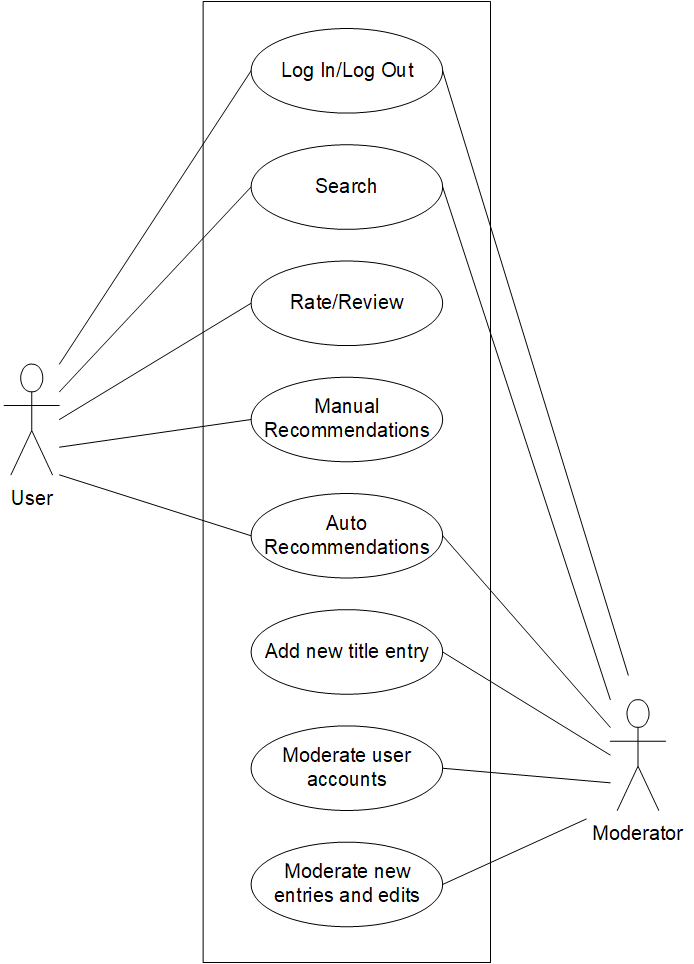
**Product name**: Mediaddict

**Problem Statement**: Currently, there are reliable movie and book review websites like Goodreads and Rotten Tomatoes. However, despite being trusted by users, these websites only collect titles[[1]](#footnote-0) from a specific medium; as a result, users cannot get title dependent recommendations of another media format.

**Objectives**: The objective of this web application is to provide a platform for entertainment (including but not limited to media contents like tv shows, movies, books) enthusiasts to keep a record of their favorite titles and get recommendations on new media contents across different mediums based on their interests. 

**6 Functional Requirements**:

1. Log In/Log Out
2. Search
3. Rate/Review
4. Manual Recommendations
5. Auto Recommendations
6. Add new title entry
7. Moderate user accounts
8. Moderate new entries and edits

**4 Non-functional requirements**:

1. Database: MongoDB
2. OOD
3. Programming language: Javascript
4. Framework: Node.js
5. Frontend: Bootstrap

Use case description 1

* Name: Review and rate movies/books
* Goal: User rates and reviews tv shows, movies, and books
* Summary: A user who has logged in can leave reviews and add ratings for movies and books
* Actors: Users
* Preconditions:
  + User has successfully logged on with a valid account
  + User is on the page of a specific movie/book title
* Trigger: User selects the “Leave a review” option
* Primary Sequence:
  + The system shows text box for reviewing the title
  + User chooses a rating based on a 5-star rating system
  + User types a review with recommendations in the textbox within the minimum and maximum character limit
  + User submits the review
  + New entry is pushed to the database
* Alternate sequences:

1 The user writes a review that exceeds the character limit

1.1 The system stops recording the data when the character limit is reached

2 The user writes a review that does not satisfy the minimum character requirement

2.1 The system displays “We would love to hear more, please elaborate”

3 The user leaves a rating but does not write a review

3.1 System displays “We would appreciate if you can leave a review”

* Postconditions:
  + The system displays the newly added review entry

Use case description 2

* Name: Manually recommend movies and books
* Goal: User recommends tv shows, movies, and books
* Summary: A user who has logged in can recommend another title for a particular show/movie/book
* Actors: Users
* Preconditions:
  + User has successfully logged on
  + User is on a page of a specific title
  + User has rated both titles
* Trigger: User selects the “Recommend” option
* Primary Sequence:

1. A a comment box together with a list of titles the user has reviewed is fetched and displayed
2. User search and pick a title to be recommended from the list as well as leave comment explaining the recommendation
3. User submit the entry

* Alternate sequences:

1 The user writes a recommendation that exceeds the character limit

1.1 The system stops recording the data when the character limit is reached

2 The user writes a recommendation that does not satisfy the minimum character requirement

2.1 The system displays “We would love to hear more, please elaborate”

3 The user recommendation but does not write an explanation

3.1 System displays “We would appreciate it if you would provide an explanation”

* Postconditions: New entry is stored in database, grouped with other recommendation of the same title and posted on the page

Use case description 3

* Name: Moderate new entries and edits
* Goal: Moderators accept or reject new entries and edits
* Summary: A moderator who has logged in will be able to accept or reject a new addition to the database. They are also able to edit existing media within the database.
* Actors: Moderators
* Preconditions:
  + Moderators are logged in
  + New entries and edits are made
* Trigger: moderator login
* Primary Sequence:

1. All changes including new entries and edits are fetched and listed in a feed for the moderator
2. Moderator have the options to approve or reject each changes
3. If a change is approved, it is dismissed from the feed.

* Alternate sequences:

1 If the change is rejected

1.1 The corresponding new entry and changes are reverted

1.2 The change entry is dismissed from the feed

* Postconditions:
  + Changes are either approved or rejected

Use case description 4

* Name: Auto Recommendations
* Goal: Automatically provide recommendations for any title without user input
* Summary: The system finds popular common interests among users of any two titles and list these as recommendations
* Actors: User and moderators
* Preconditions:
  + At least two of users rated the same two titles positively
* Trigger: When a page of a certain title is launched (but i dont think its efficient? Should this be done every hour? Or it can be triggered when there is a new rating entry?)
* Primary Sequence:

1. Compare users who rated the title positively
2. Look for other titles that are also being rated positively by the same group

* Alternate sequences:

1. There are no data to gather on the current media

1.1 System display “There are no current recommendation available by the system”

* Postconditions:
  1. Automatically generated recommendations are displayed in the page of a title

1. A book, a movie, a TV show or other media publication [↑](#footnote-ref-0)