

# Requirements

Latest revision: 2012-09-28

**ID**

1

**Title**

Add Movie

**PreCon**

There's a view with interface controls in which you can add movies.

**PostCon**

The movie is added to the list.

**Description**

To add a movie you use the button "Add Movie". This brings you to a view where you insert information about the movie, i.e. title, rating, etc. After this you push a new "Add movie"-button.

**Depend**

-

**Rank**

1

**ID**

2

**Title**

Remove Movie

**PreCon**

The user has already added a movie

**PostCon**

An alert box is shown. If the user chooses "Yes", the movie is removed from the application

**Description**

If the user makes a long tap on a movie in the list of movies and alert box is shown that asks the user to confirm if he wants to remove the movie.

**Depend**

1

**Rank**

2

**ID**

3

**Title**

Edit Rating

**PreCon**

The user has already added a movie

**PostCon**

The information about the rating of the movie is updated in the database. The movie list item is updated

**Description**

The user is in the activity where you can see information about the movie. The rating is shown in "filled in"-stars. If the user taps on another star the rating is changed to that.

**Depend**

1

**Rank**

1

**ID**

4

**Title**

Get Notification

**PreCon**

The user has added a movie.

**PostCon**

The user will get a notification on the date set on the movie (defaults to the same day).

**Description**

When adding or editing a movie a user should be able to specify a date when he wants to get a notification. A notification can for example tell the user if the Movie has entered a theatre or has come out on DVD. The notification will pop up in the status bar.

**Depend**

6

**Rank**

1

**ID**

5

**Title**

User can e-mail list of movies

**PreCon**

There are at least one movie in the list. There is a UI control for e-mailing the movie list.

User has specified an e-mail address to send to in a view.

**PostCon**

A list of the movies in plain text is e-mailed to an e-mail address

**Description**

The user can e-mail the whole list of movies in order to share to friends or just as a backup. When the user taps the "E-mail list" button, he/she gets the chance to fill in an e-mail address. An e-mail will be sent, and a status box will notify the user about the success/failure of the operation.

**Depend**

1

**Rank**

3

**ID**

6

**Title**

Edit date

**PreCon**

The user has already added a movie

**PostCon**

The information about the date of the movie is updated in the database. The movie list item is updated

**Description**

The user is in the activity where you can see information about the movie. The date is pressable. If the user taps on the date he can edit it.

**Depend**

1

**Rank**

1

**ID**

7

**Title**

Edit Note

**PreCon**

The user has already added a movie.

**PostCon**

A note has been set on the movie.

**Description**

A user can go into a view where he can see information about a movie. Here he can press a textfield and write down a note. When focus is moved from the textfield the note will be saved.

**Depend**

1

**Rank**

2

**ID**

8

**Title**

Search Movie List

**PreCon**

There are at least one movie in the list

**PostCon**

The movies with a title which matches the search input are shown (filtered) in the list

**Description**

In order to find a specific movie in the list, a search field is shown when the user taps the search button in the Action bar. While the user is writing text in the search field, the movie list is filtered by movie titles in real time to match the search input.

**Depend**

1

**Rank**

3

**ID**

9

**Title**

Sort Movie List

**PreCon**

The user has already added a movie

**PostCon**

The movie-list is sorted after the specifications.

**Description**

There is a sort-button in the main-view. When it is pressed the user gets to choose how to sort the list: by date, title or rating.

**Depend**

1

**Rank**

3

**ID**

10

**Title**

Fetch info from IMDb.

**PreCon**

The user has added a movie.

**PostCon**

The user will see information about a movie fetched from an IMDb API.

**Description**

The user will be able to go into a view in which he can see information such as genre, actors, etc., about a movie. The information will be fetched from an IMDb API.

**Depend**

1

**Rank**

1