Final Project – Movie Database: Movie Review App

Submitted by: Rawa Jalal & Steven Ye

PROG3180 Section 4 – Programming Mobile Applications

Submitted to: Sabbir Ahmed

Submission Due Date: April 23, 2018

**Table of Contents**

[**Project Charter** 3](#_Toc512177443)

[**Project Name** 3](#_Toc512177444)

[**Business Purpose** 3](#_Toc512177445)

[**Project Roles/Responsibilities** 3](#_Toc512177446)

[Rawa Jalal 3](#_Toc512177447)

[Steven Ye 3](#_Toc512177448)

[**Technical Architecture** 3](#_Toc512177449)

[**Events Table** 4](#_Toc512177450)

[**Use Case Diagram** 5](#_Toc512177451)

[**Use Case Descriptions** 6](#_Toc512177452)

[**User Interface Prototype** 7](#_Toc512177453)

[**Entity Relationship Diagram (ERD)** 15](#_Toc512177454)

[**User Documentation – Help System** 16](#_Toc512177455)

[**Description** 16](#_Toc512177456)

[**How To Use** 16](#_Toc512177457)

[**About The Programmers** 16](#_Toc512177458)

[**References** 17](#_Toc512177459)

# **Project Charter**

## **Project Name**

Movie Database

## **Business Purpose**

Our objective is to create a Cordova/PhoneGap app that allows multiple users to look up information about movies and read/write reviews for them. The user will be able enter data not only for multiple reviews, but for multiple reviewers as well. By keeping track of various reviews, the user can give themselves a frame of reference for what movies are well-received or enjoyed by certain individuals.

## **Project Roles/Responsibilities**

### Rawa Jalal

* Main programmer

### Steven Ye

* Documentation, Planning, Quality Assurance

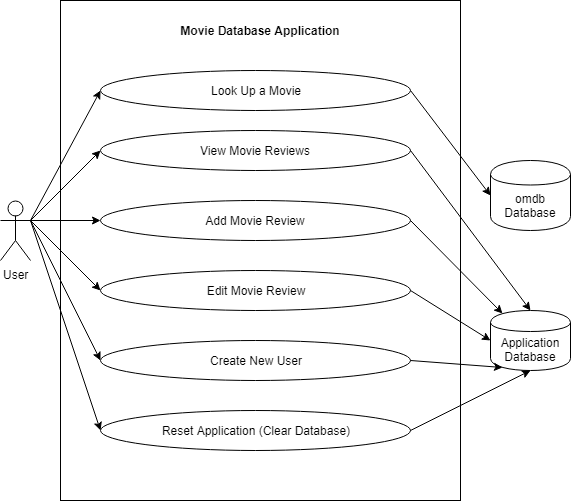
## **Technical Architecture**

The technologies under consideration for this project include HTML5, CSS3, JavaScript (and the JavaScript libraries jQuery and jQuery Mobile), and WebSQL, as well as PhoneGap/Cordova for packaging the app as a standalone Android APK. This project will also utilize The Open Movie Database, OMDb, for looking up information about movies.

# **Events Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Event** | **Source** | **Use Case #** | **Response** | **Destination** |
| User wants to look up a movie | User | UC01 | Movie details are displayed | User |
| User wants to see reviews of a movie | User | UC02 | Movie reviews are displayed | User |
| User wants to add a review for a movie | User | UC03 | Movie review added to database | System |
| User wants to edit a review for a movie | User | UC04 | Movie review in database modified | System |
| User wants to add a user to the app | User | UC05 | User added to database | System |
| User wants to reset the app to default | User | UC06 | Database cleared | System |

# **Use Case Diagram**



# **Use Case Descriptions**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case #** | UC01 | | | |
| **Use Case Name** | Look Up A Movie | | | |
| **Scenario** | User wants to look up the details for a movie. | | | |
| **Triggering Event** | User clicks the search button. | | | |
| **Brief Description** | The user clicks the search button, enters the name of the movie they would like to look up, and chooses the movie from the list of search results. | | | |
| **Actors** | User | | | |
| **Related Use Cases** | View Movie Reviews,  Add Movie Review,  Edit Movie Review | | | |
| **Pre-conditions** | User knows the name of the movie they would like to look up. | | | |
| **Post-conditions** | The system displays the requested movie’s details to the user. | | | |
| **Flow of Events** | **Actor** | | **System** | |
|  | 1. | User clicks Search | 2. | Display search form |
|  | 3. | Enter movie search query | 4. | Display results of search |
|  | 5. | User clicks desired movie | 6. | Display selected movie’s detailed information |
| **Exceptions** | Movie does not exist in database. | | | |

Table 1: UC01 – Look Up A Movie

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case #** | UC02 | | | |
| **Use Case Name** | View Movie Reviews | | | |
| **Scenario** | User wants to read reviews for a movie. | | | |
| **Triggering Event** | User clicks the View Reviews button on a movie details page. | | | |
| **Brief Description** | After looking up the details of a movie, the user can then choose to see reviews for the movie. | | | |
| **Actors** | User | | | |
| **Related Use Cases** | Look Up A Movie | | | |
| **Pre-conditions** | User has just looked up the movie they wish to read reviews for within the app. (UC01) | | | |
| **Post-conditions** | The system displays any reviews for the movie that have been previously stored. | | | |
| **Flow of Events** | **Actor** | | **System** | |
|  | 1. | User clicks View Reviews | 2. | Display reviews of the movie. |
| **Exceptions** | Database does not contain any reviews. | | | |

Table 2: UC02 – View Movie Reviews

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case #** | UC03 | | | |
| **Use Case Name** | Add Movie Review | | | |
| **Scenario** | User wants to write a review for a movie. | | | |
| **Triggering Event** | User clicks the Write Review button on a movie details page. | | | |
| **Brief Description** | After looking up the details of a movie, the user can then choose to write a review for the movie. | | | |
| **Actors** | User | | | |
| **Related Use Cases** | Look Up A Movie | | | |
| **Pre-conditions** | User has just looked up the movie they wish to write a review for within the app. (UC01) | | | |
| **Post-conditions** | The system stores the user’s newly submitted movie review. | | | |
| **Flow of Events** | **Actor** | | **System** | |
|  | 1. | User clicks Write Review | 2. | Display movie review form. |
|  | 3. | Enter movie review details and submit | 4. | Save information, display confirmation. |
| **Exceptions** | None. | | | |

Table 3: UC03 – View Movie Reviews

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case #** | UC04 | | | |
| **Use Case Name** | Edit Movie Review | | | |
| **Scenario** | User wants to edit a review for a movie. | | | |
| **Triggering Event** | User clicks a review on the View Reviews page of a movie. | | | |
| **Brief Description** | After looking up the reviews of a movie, the user can then choose to edit a review for the movie. | | | |
| **Actors** | User | | | |
| **Related Use Cases** | Look Up A Movie,  View Movie Reviews | | | |
| **Pre-conditions** | User has just looked up reviews for the movie they wish to edit a review for within the app. (UC02) | | | |
| **Post-conditions** | The system stores the user’s newly edited movie review. | | | |
| **Flow of Events** | **Actor** | | **System** | |
|  | 1. | User clicks review they wish to edit | 2. | Display movie review edit form. |
|  | 3. | Enter edited movie review details and submit | 4. | Save information, display confirmation. |
| **Exceptions** | User wishes to delete review; user clicks Delete button at step 3, system deletes entry from database and displays confirmation. | | | |

Table 4: UC04 – Edit Movie Review

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case #** | UC05 | | | |
| **Use Case Name** | Create New User | | | |
| **Scenario** | User wants to add a user to the app | | | |
| **Triggering Event** | User clicks the User Registration button. | | | |
| **Brief Description** | The user clicks the User Registration button, enters the details of the user they would like to add, and clicks the Create user button. | | | |
| **Actors** | User | | | |
| **Related Use Cases** | None | | | |
| **Pre-conditions** | None | | | |
| **Post-conditions** | The system stores the newly created user for writing reviews. | | | |
| **Flow of Events** | **Actor** | | **System** | |
|  | 1. | User clicks User Registration. | 2. | Display user registration form. |
|  | 3. | Enter new user details and submit. | 4. | Save information, display confirmation. |
| **Exceptions** | Database does not contain any reviews. | | | |

Table 5: UC05 – Create New User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case #** | UC06 | | | |
| **Use Case Name** | Reset Application (Clear Database) | | | |
| **Scenario** | User wants to erase all reviews and users. | | | |
| **Triggering Event** | User clicks the User Registration button. | | | |
| **Brief Description** | After looking up the details of a movie, the user can then choose to see reviews for the movie. | | | |
| **Actors** | User | | | |
| **Related Use Cases** | None | | | |
| **Pre-conditions** | None | | | |
| **Post-conditions** | The system will be cleared of all data previously input by the user. | | | |
| **Flow of Events** | **Actor** | | **System** | |
|  | 1. | User clicks User Registration. | 2. | Display Clear Database button |
|  | 3. | Click Clear Database button | 4. | Drop database, display confirmation. |
| **Exceptions** | None | | | |

Table 2: UC02 – View Movie Reviews

# **User Interface Prototype**

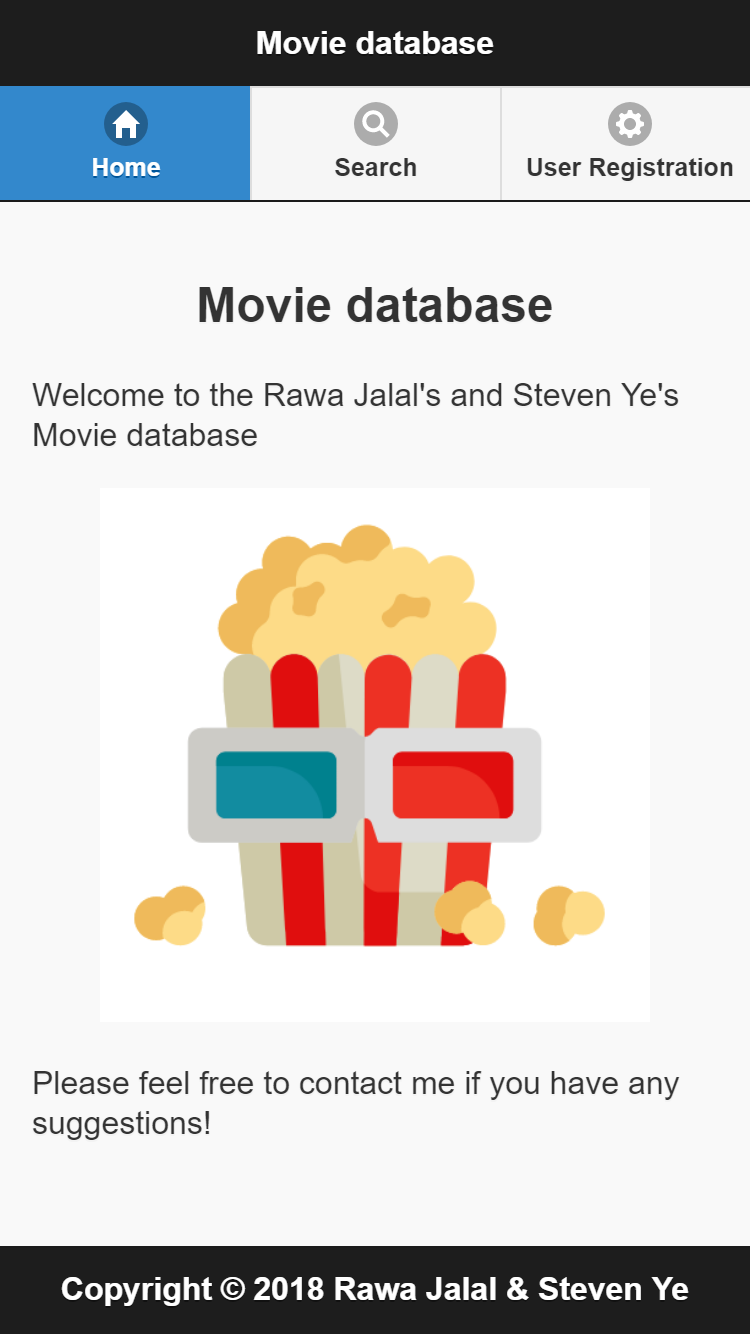
****

Figure 1: Screen 1 - Home

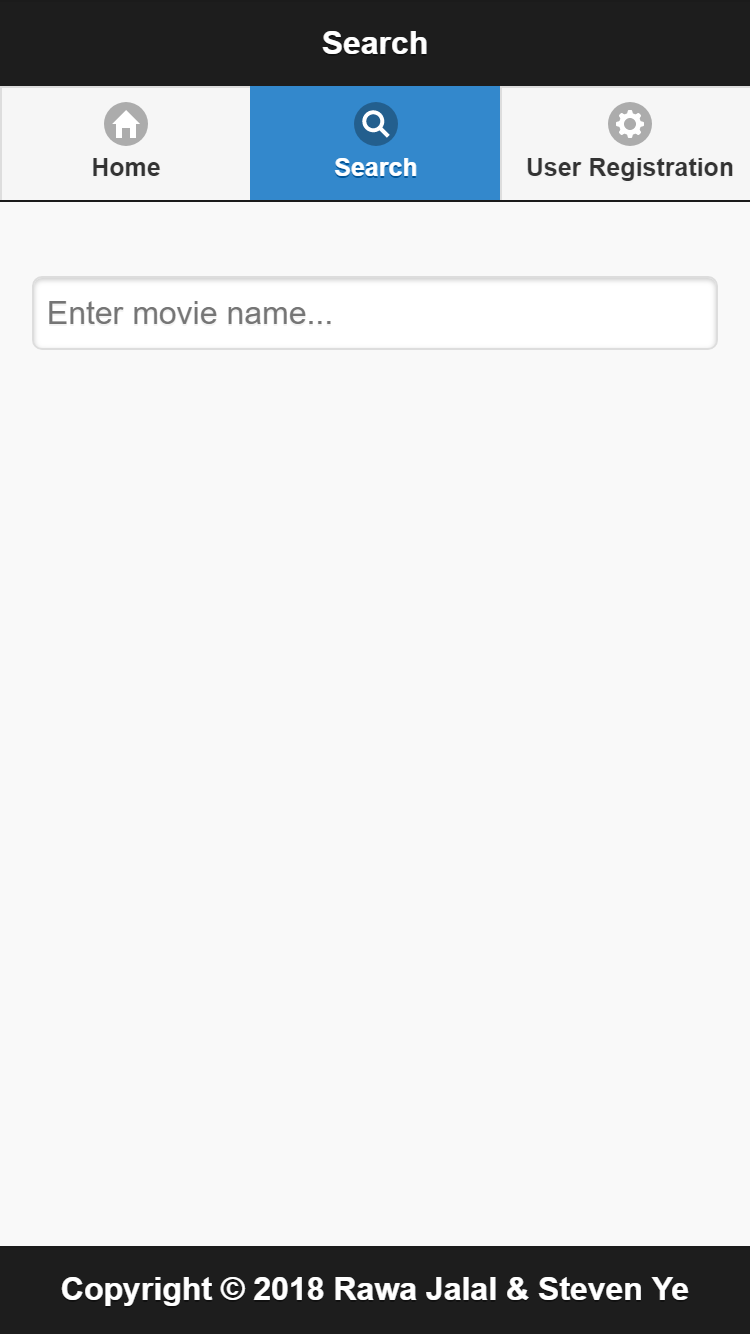


Figure 2: Screen 2 - Search

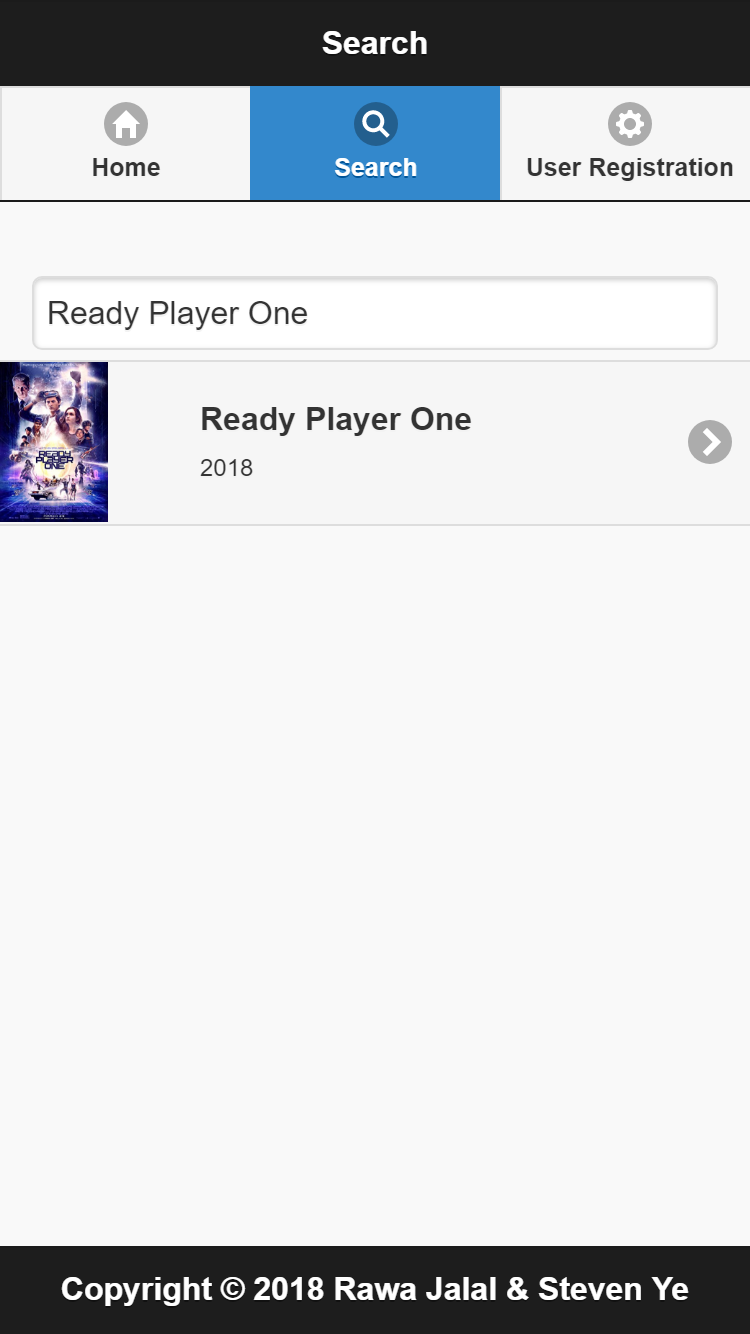
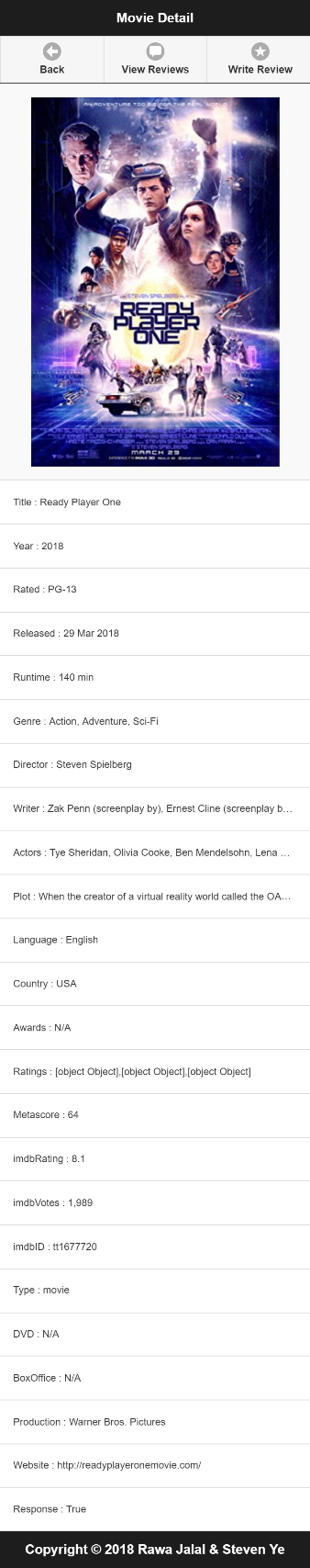


Figure 3: Screen 2 – Search Results

  
Figure 4: Screen 3 – Movie Details

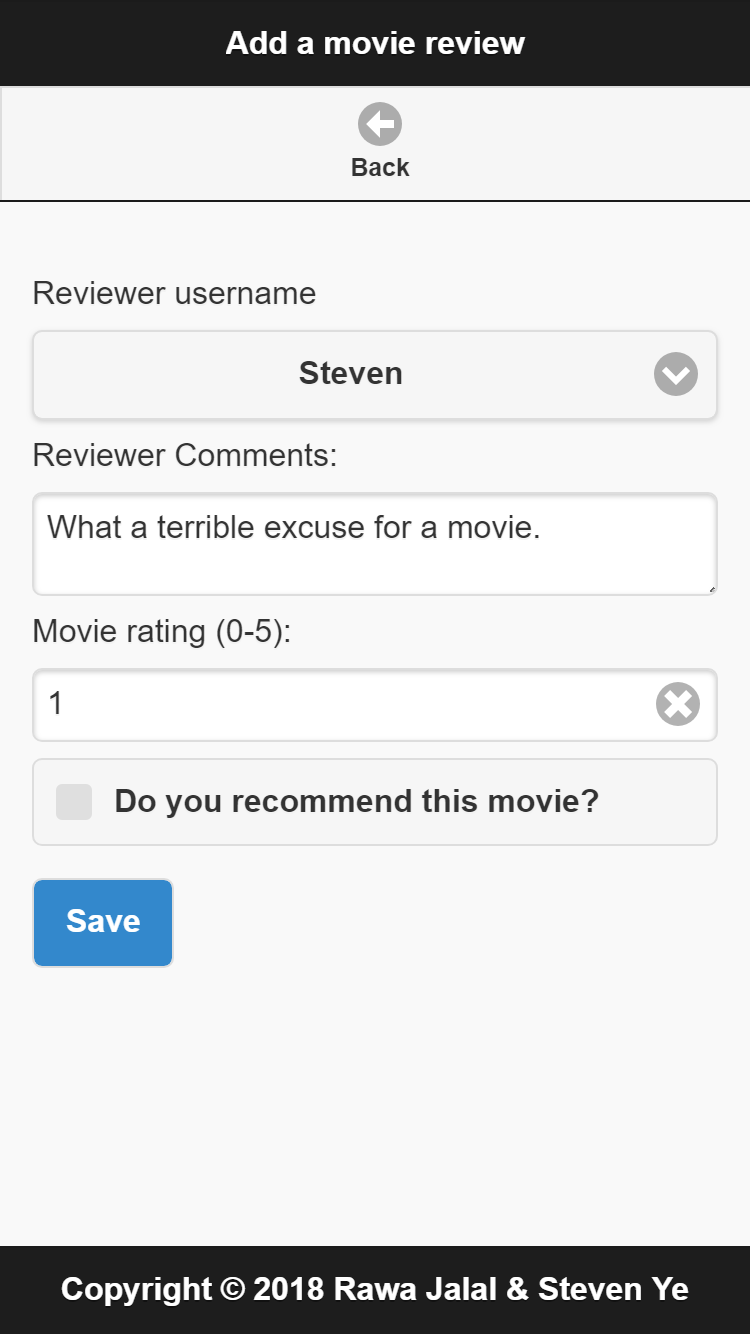


Figure 5: Screen 4 – Add Review

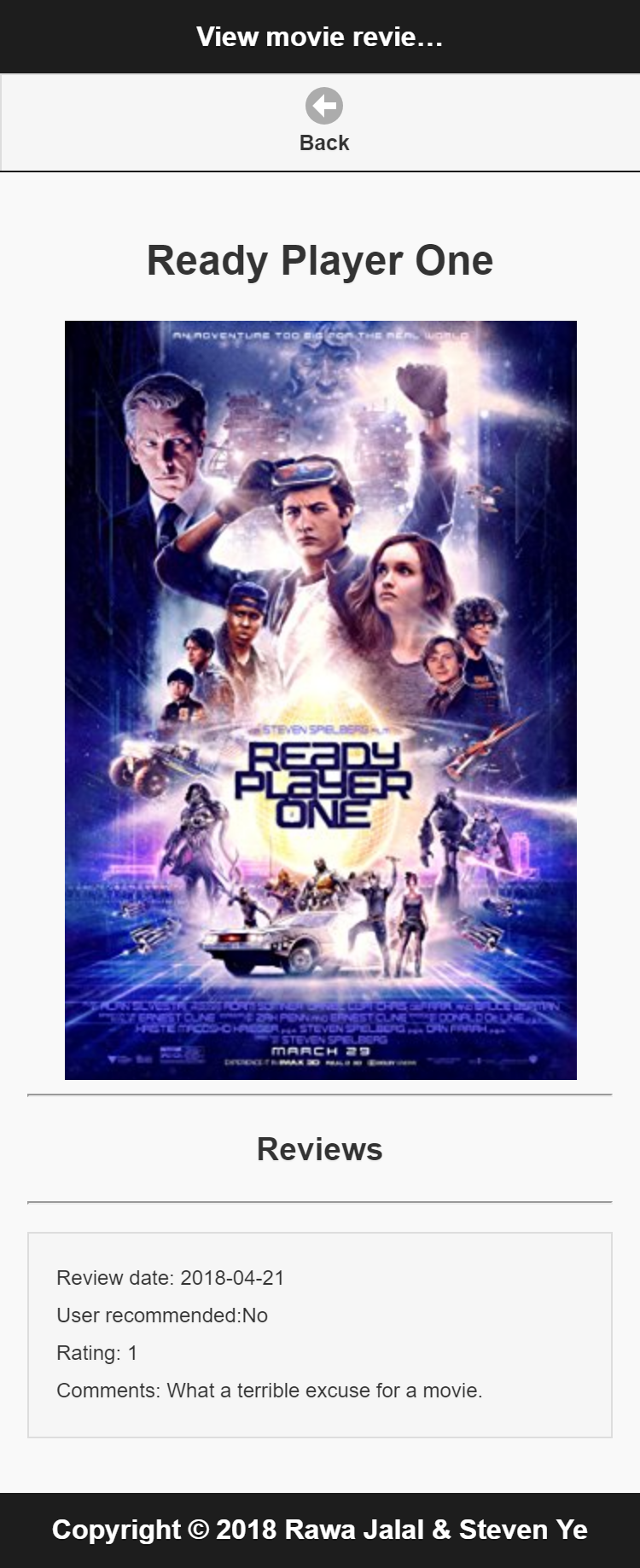


Figure 6: Screen 5 – View Reviews

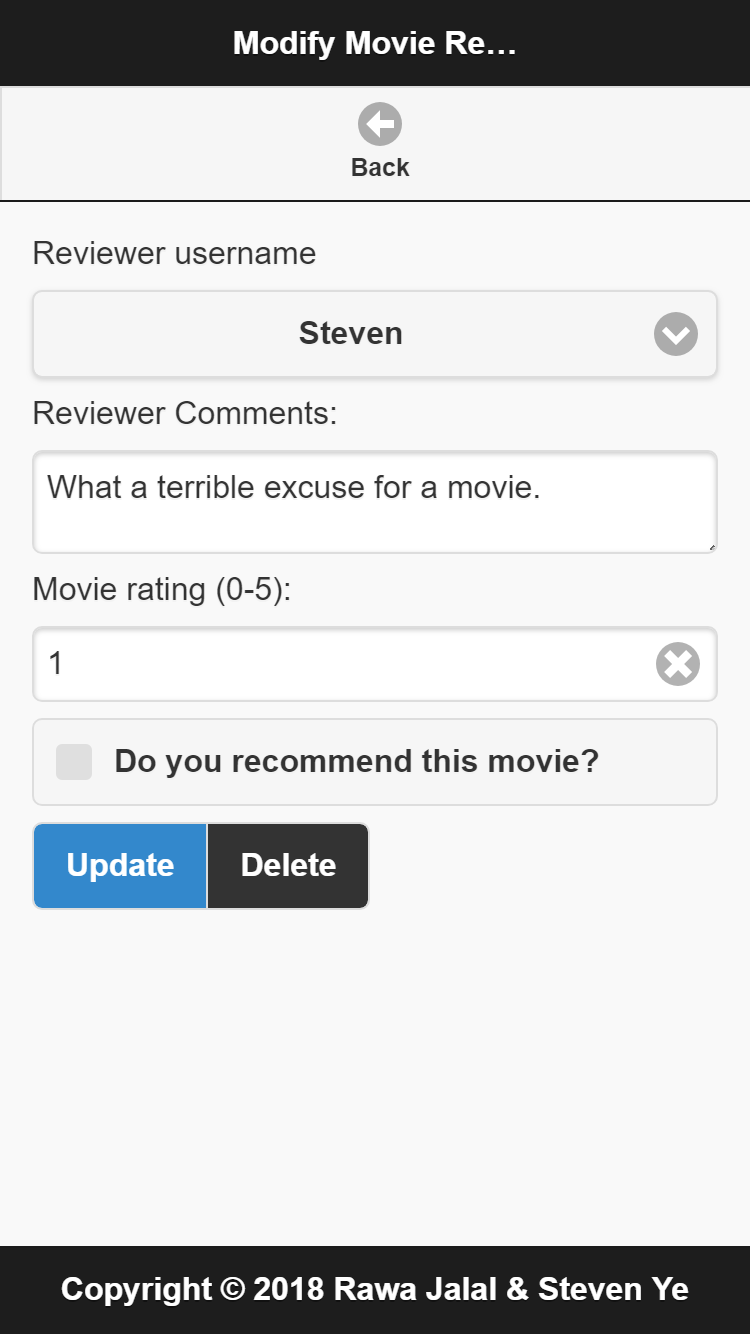


Figure 7: Screen 6 – Edit or Delete Review

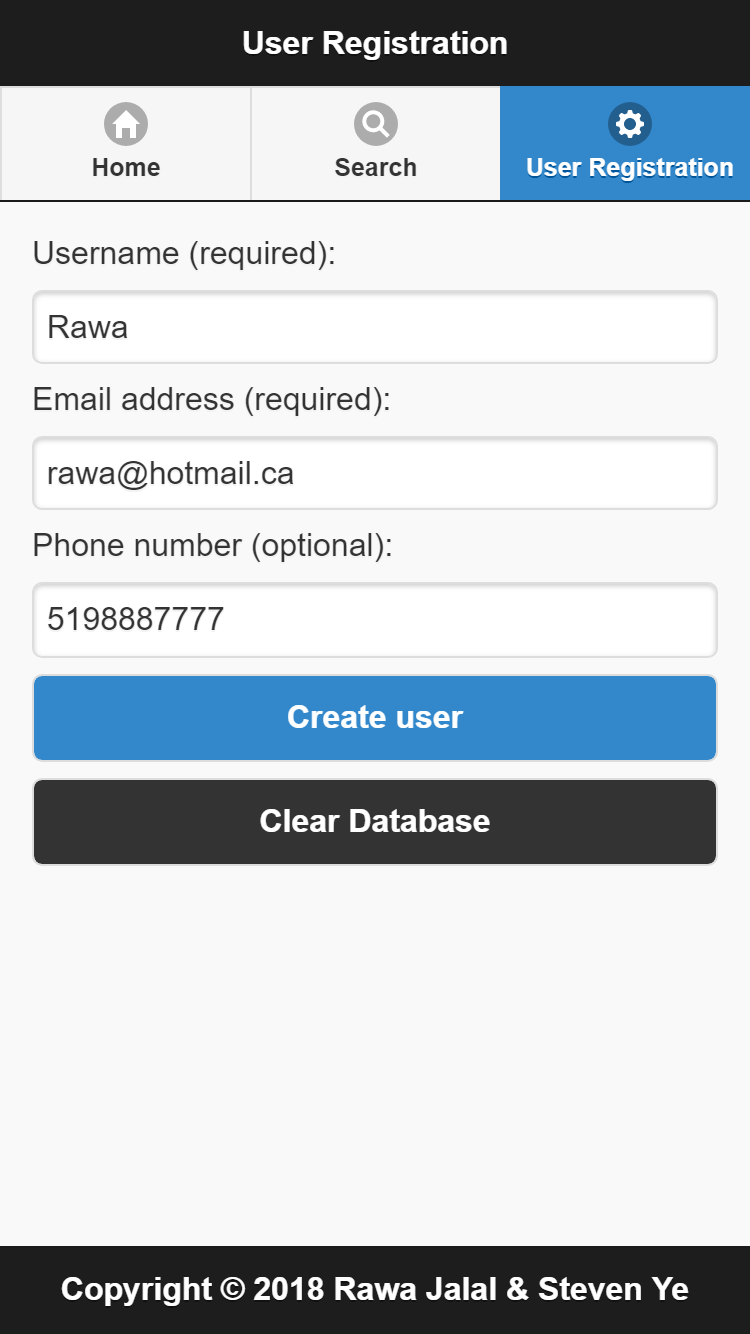
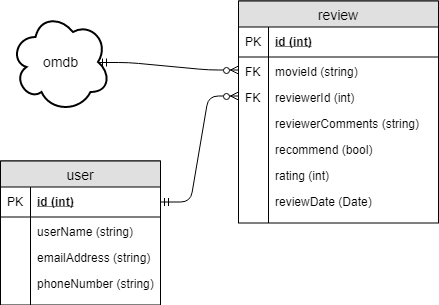


Figure 8: Screen 7 – User Registration & Settings

# **Entity Relationship Diagram (ERD)**



# **User Documentation – Help System**

## **Description**

Here’s a brief description.

## **How To Use**

Screenshots for how to use

## **About The Programmers**

Rawa Jalal and Steven Ye are classmates in Conestoga College’s Computer Programmer/Analyst program. Their collective interest in computer science and film led them to create this app.

# **References**

1. jQuery Mobile: <https://jquerymobile.com/>
2. PhoneGap API: <http://docs.phonegap.com/plugin-apis/>
3. OMDb API: <http://www.omdbapi.com/>