

## **ITIS/ITCS 5180 Mobile Application Development In Class Assignment 10**

### **Basic Instructions:**

1. Each group should submit only one assignment on behalf of all the other group members.
2. Your assignment will be graded for functional requirements and efficiency of your submitted solution. You will lose points if your code is not efficient, does unnecessary processing or blocks the UI thread.
3. Export your Android project and create a zip file which includes all the project folder and any required libraries.
4. Submission details:
  - a. Compress the contents of your project folder. The file name is very important and should follow the following format: **IC10\_Group#.zip**
  - b. Only one group member is required to submit on behalf of the whole group.
  - c. You should submit the assignment through Canvas: Submit the zip file.
- 5. Failure to follow the above instructions will result in point deductions**

### **In Class Assignment 10 (100 Points)**

In this assignment we will revisit Homework 04 utilizing Firebase Cloud Firestore. This is a simple movie database app, where you can save your favorite movie details. You can add new movies, edit them, display them by their release year or rating and delete them. The main screen should look like the Figure 1.



**Figure 1: Main Activity**

#### **Main Activity (20 Points):**

The main activity contains five buttons arranged as displayed in Figure 1. The activity requirements are as follows:

1. The Movie list should be stored in Firebase Cloud Firestore.
2. The buttons labeled as Add Movie, Edit, Delete Movie, Show List by Year and Show List by Rating should start the Add Movie Activity, Edit Activity, Delete Movie Activity, Display List Year Activity and Display List Rating Activity respectively.

#### **Add Movie Activity (20 Points):**

The Add Movie activity is displayed in Figure 2. Implement the following requirements:

1. When you click the “Add Movie” button in the Main Activity, the Add Movie should be started as shown in Figure 2.
2. The Movie name must not exceed 50 characters.
3. There should be a multiline EditText for the small description. It should not exceed 1000 characters.
4. There should be a drop down box for selecting the Genre of the Movie. The Genres you should put here are: Action, Animation, Comedy, Documentary, Family, Horror, Crime and Others.

5. You should implement a SeekBar for getting the Rating. The minimum value for SeekBar is 0, and the maximum should be set to 5. The step value should be set to 1.

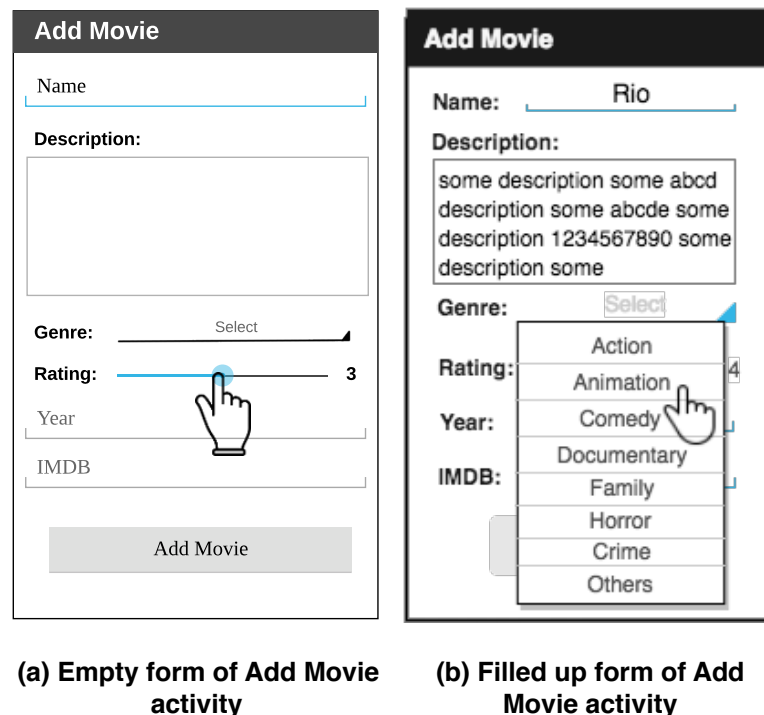


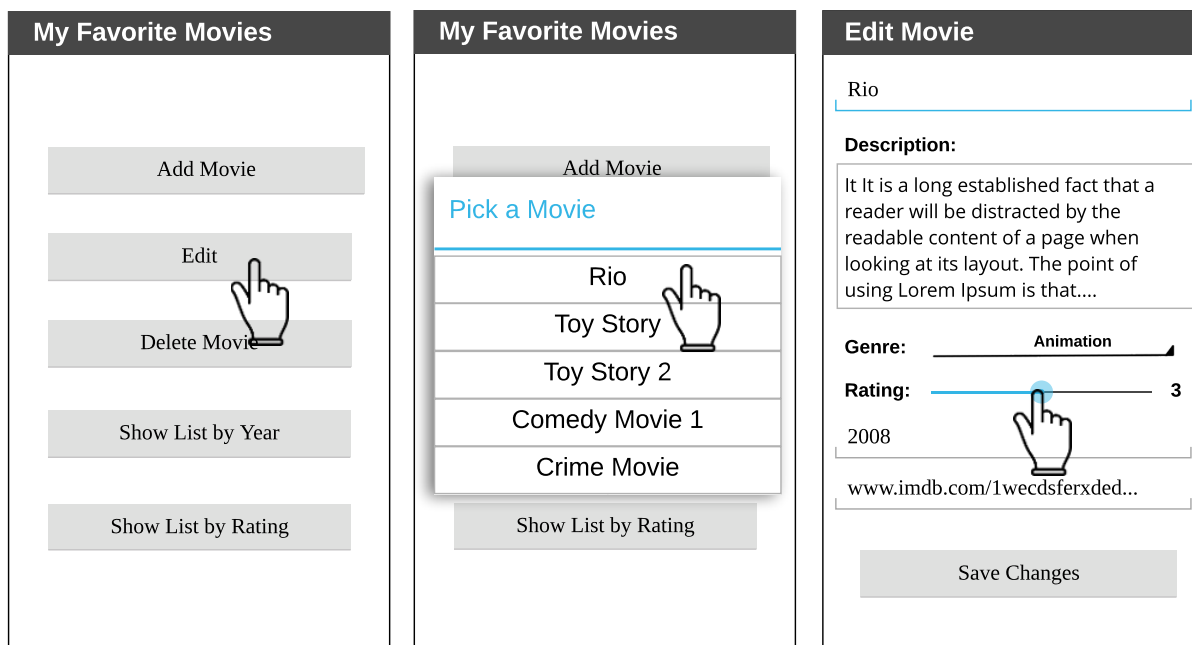
Figure 2: Add Movie activity

6. There should be an EditText to take input for the Year of Release.
7. There should be another EditText to take input for the IMDB link.
8. There should be a button labeled as, “Add Movie”.
9. Clicking on the “Add Movie” button should save the fields as a Movie object, and add it to Firebase Document into the movies collection.
10. You should detect any anomaly and make Toasts accordingly. You should ensure every input here is put accordingly.

### **Edit Movie Activity (25 Points)**

The Edit Movie activity is displayed in Figure 3. Implement the following:

1. When you click the “Edit” button on the Main Activity, you should start an Alert Dialogue having a list of all the movies you saved in Firebase, as it is in Figure 3(b).
2. Clicking on any of the movies should start the Edit Movie activity to edit that entry.
3. The UI for Edit Movie activity is similar to the Add Movie Activity and should follow the same text size and implementation and restrictions.
4. It should load the saved fields of the selected movie from the saved movies collection.
5. Clicking on the Save Changes button should save the fields as a Movie object and update the corresponding movie in Firebase.

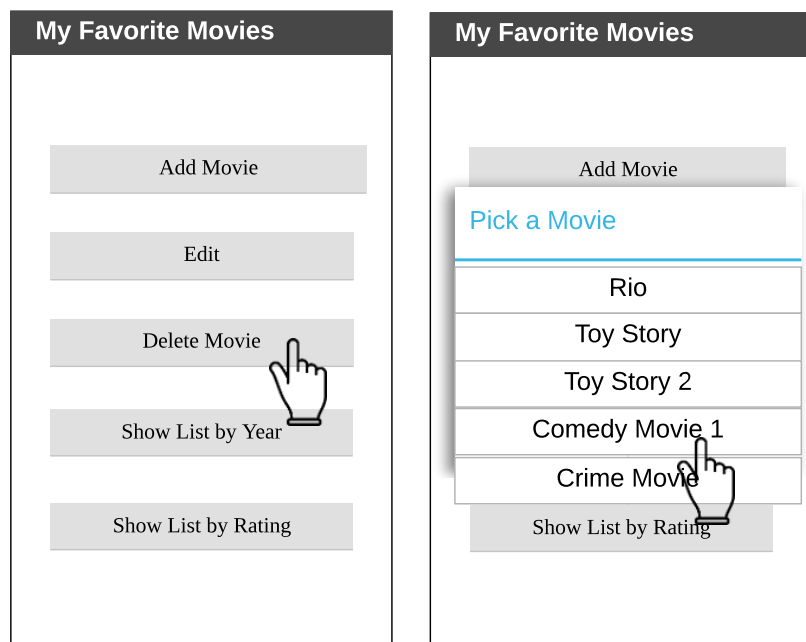


(a) Click on Edit button

(b) Open the Alert Dialogue

(c) Edit the movie details

Figure 3: Edit Movie activity.



(a) Click on Delete Movie button

(b) Alert Dialogue with the list, click on a movie

Figure 4: Delete Movie

### Delete Movie (5 Points)

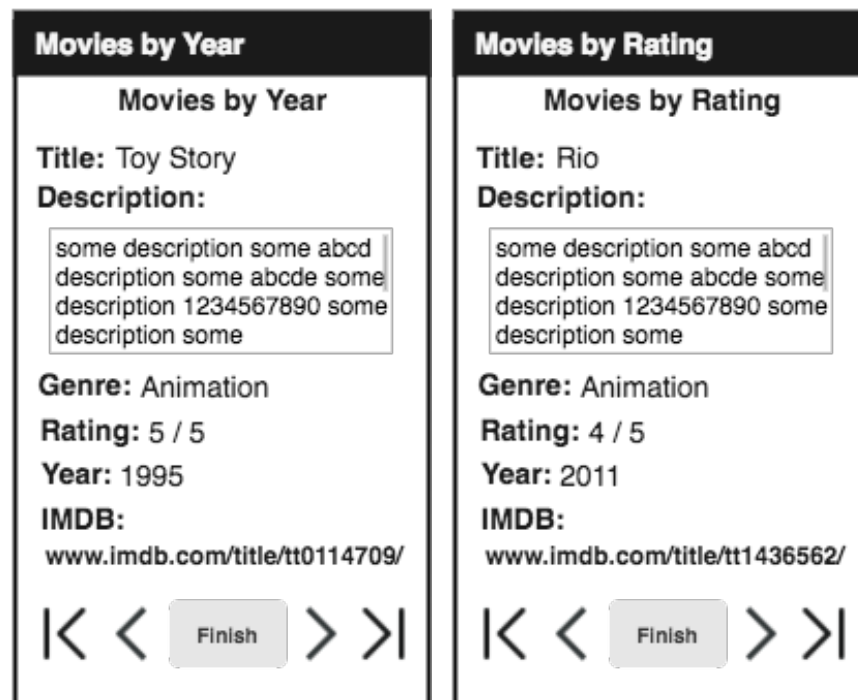
The Delete Movie process is displayed in Figure 4. Implement the following requirements:

1. When the user selects “Delete Movie” button in the Main Activity, just like the Edit Movie part, an Alert Dialogue should come up with the movie list.
2. When you select one of the movies in the list, it should delete it from Firebase.
3. Toast a message showing that the movie is deleted that includes the movie name.

### **Show List by Year/Rating (15+15 Points)**

The final activities you will be creating are Movie by Year activity and Movie by Rating activity. The requirements are as follows:

4. If you click on Show List by Year button you should start the Movies by Year activity.
5. In the activity you should display the string “Movies by Year” on the Top.
6. You will be sorting the movie list by the Ascending order of Year. The actions required from each of the GUI images are listed in Table 1. Upon clicking on Finish button the activity should be finished.



(a) Movies by Year Activity

(b) Movies by Rating Activity

Figure 5: Show List by Year/Rating

Button	Action
<	Show First Movie
<	Show Previous Movie
>	Show Next Movie
>	Show Last Movie

Table 1, GUI Actions

7. The Movie by Rating will actually have the same functionalities as the Movie by Year activity. The only difference is that, you need to sort the List by the Descending order of Rating.

Good Luck!