

In Class 05 (100 points)

In this assignment you will develop “iTune Apps” app, which enables the user to display apps from different categories. You are provided with a skeleton application that includes the data and classes to use for this project. The DataServices class that includes a dictionary of apps lists for different categories. Setup requirements are listed below:

- Use the provided skeleton project, which includes all the activity, fragment and data classes.
- The provided DataServices provides the apps and app categories information to be used in this assignment.
- In this app you will have only one Activity and 3 fragments, all communication between fragments should be managed by the activity.

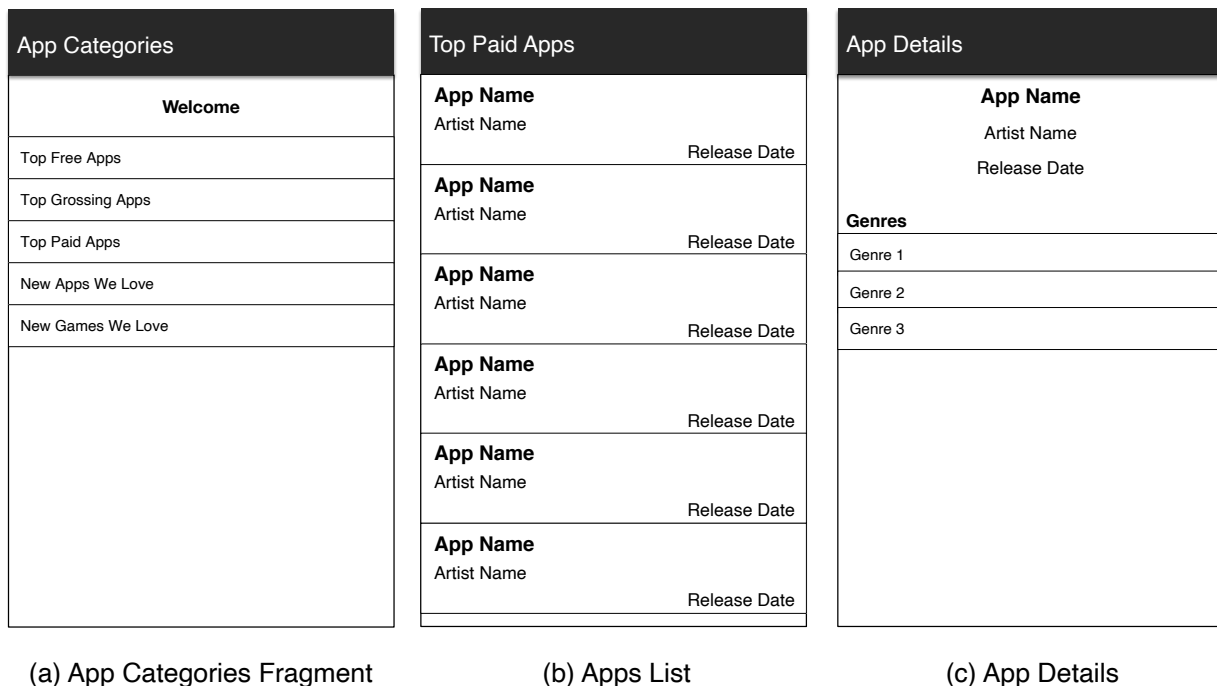


Figure 1, Application Wireframe

Part 1: App Categories (30 points):

This screen contains a ListView of app categories retrieved from the DataServices by calling `getAppCategories` method. This screen is shown in Figure 1(a). The requirements are as follows:

1. The app categories should be retrieved from DataServices by calling `getAppCategories` method in order to retrieve the list of categories.
2. The list of categories should be displayed using a ListView as shown in Figure 1(a).
3. If the user clicks on a category list item:
 - a. Replace the current fragment with the App List Fragment, pass the selected app category to the App List Fragment.
 - b. Push the current fragment on the back stack.

Part 2: Apps List (50 points):

This screen contains a ListView of apps based on the selected category in the App Categories Fragment as shown in Figure 1(b). The requirements are as follows:

1. The category is passed from the App Categories Fragment should be retrieved, the activity title should be set to the selected category name.
2. The list of apps corresponding to this category should be retrieved from the DataServices class by calling getAppByCategory method. The list should be used to populate the ListView as shown in Figure 1(b). Each list item shows the app name, artist name, and release date.
3. If the user clicks on an App list item:
 - a. Replace the current fragment with the App Details, pass the selected app to the App Details Fragment.
 - b. Push the current fragment on the back stack.

Part 3: App Details (20 points):

This screen displays the app details as shown in Figure 1(c). The requirements are as follows:

1. This screen should receive the app object from the previous Apps List fragment and should show the app details as shown in Figure 1(c).
2. The app details include the app name, artist name, and release date. In addition, the app genres should be displayed in a ListView as shown in Figure 1(c).