

Mobile App Development
Bonus Midterm Make-Up Exam

Basic Instructions:

1. This is the Bonus Midterm Make-Up Exam, which will add up to 38 points towards your current overall midterm grade.
2. This assignment is only for students who have an overall average that is lower than 70%. This assignment is intended to give these students a chance to demonstrate their individual ability to deliver applications.
3. This bonus assignment is an individual effort. Each student is responsible for her/his own bonus and its submission.
4. Once you have picked up the exam, you may not discuss it in any way with anyone until the exam period is over.
5. During the exam, you are allowed to use the course videos, slides, and your code from previous home works and in class assignments. You can use the internet to search for answers. You are NOT allowed to use code provided by other students or solicit help from other online persons.
6. Answer all the exam parts, all the parts are required.
7. Please download the support files provided with the Midterm and use them when implementing your project.
8. 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.
9. Create a zip file which includes all the project folder, any required libraries, and your presentation material. Submit the exported file using the provided canvas submission link.
- 10. Do not try to use any Social Messenger apps, Emails, Or Cloud File Storage services in this exam.**
- 11. Failure to follow the above instructions will result in point deductions.**
- 12. Any violation of the rules regarding consultation with others will not be tolerated and will result disciplinary action and failing the course.**

Bonus Midterm Make-Up Exam (38 Points)

In this midterm makeup exam assignment you will develop a news app. The app will enable the user to select the news source and will display the news for the selected news source. The api used for this assignment is the News API <https://newsapi.org/>.

API Setup and App Requirements:

1. Go to <https://newsapi.org/register> and to setup a new account and get an API key.
2. In this app you will have only one Activity and 3 fragments, all communication between fragments should be managed by the activity.
3. You are required to use OkHttp library in this application in order to make all the http connections and API calls. All the data returned by the APIs is in JSON format.
4. All the network calls and parsing should be done in a background thread.
5. All UI changes, updates and edits should be performed on the main thread only.

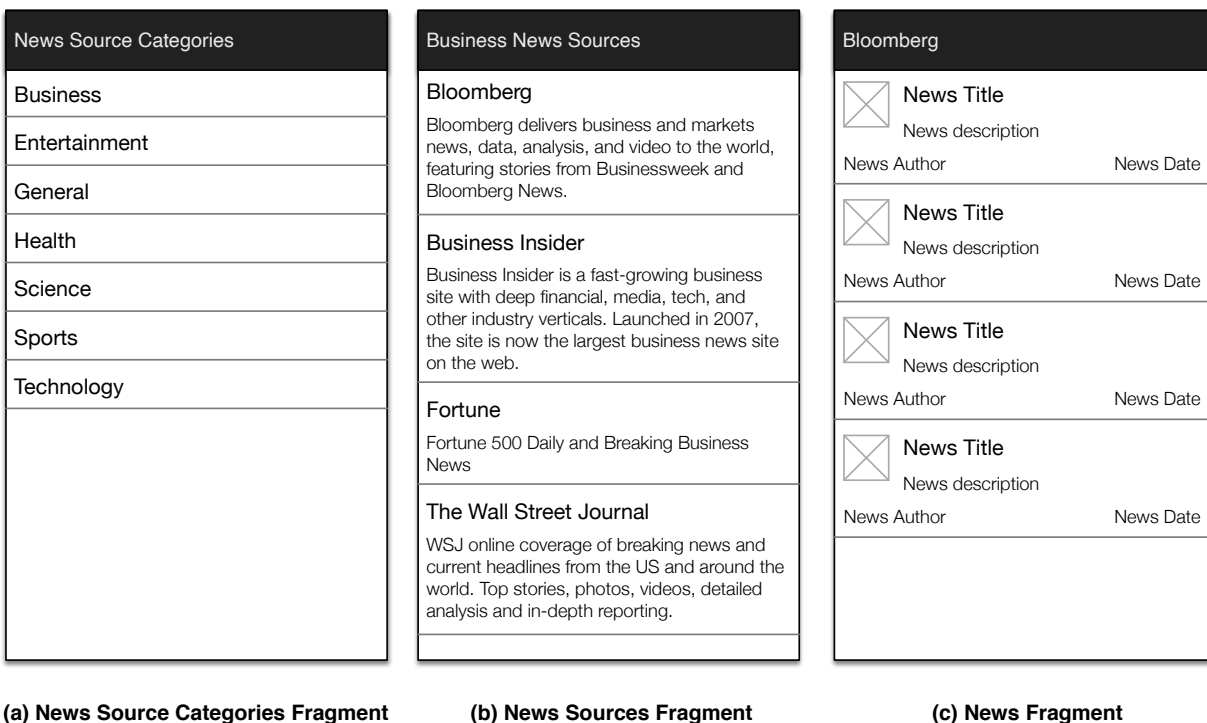


Figure 1, Application Wireframe

Part 1: News Source Categories Fragment (4 Points)

This fragment shows the news source categories available as shown in Figure 1(a). The requirements are as follows:

1. You are provided with NewsData which contains a list of categories, which should be used to display the different category names as show in Figure 1(a).
2. You are not allowed to edit or update the NewsData or the Category classes and should use them as provided.
3. The categories should be presented using ListView or RecyclerView as shown in Figure 1(a). You should create a custom adapter.
4. Clicking on a row item should replace the current fragment with the News Sources Fragment, pass it the required data and Push the current fragment on the back stack.

Part 2: News Sources Fragment (14 Points)

The fragment shows the list of news sources for the selected category as shown in Figure 1(b). The requirements are as follows:

1. The screen title should be updated to show the name of the selected category.
2. This fragment should retrieve the news sources using the “Sources” api, use the documentation provided at <https://newsapi.org/docs/endpoints/sources>
 - a. The correct parameters should be passed to the api to retrieve the list of sources that are in the US, and for the selected category.
 - b. All api communication should be performed using OkHttp library.
 - c. Create a Source class, and parse the returned list of sources into an ArrayList containing the parsed Source objects. Use the parsed list of Source objects to display the source list in a ListView or RecyclerView as shown in Figure 1(b). You should create a custom adapter.
3. Each row item should display the source name and description as shown in Figure 1(c).
4. Clicking on a row item should replace the current fragment with the News Fragment, pass it the required data and Push the current fragment on the back stack.

Part 3: News Fragment (20 Points)

The fragment shows the news items for the selected news source as shown in Figure 1(c). The requirements are as follows:

1. The screen title should be updated to show the name of the selected news source.
2. This fragment should retrieve the news articles using the “Everything” api, use the documentation provided at <https://newsapi.org/docs/endpoints/everything>
 - a. The correct parameters should be passed to the api to retrieve the list of news articles that belong to the selected news source.
 - b. All api communication should be performed using OkHttp library.
 - c. Create a News class, and parse the returned list of news articles into an ArrayList containing the parsed News objects. Use the parsed list of News objects to display the news list in a ListView or RecyclerView as shown in Figure 1(c). You should create a custom adapter.
3. Each row item should display:
 - a. The news “title”, “author”, “description”, “publishedAt”. The publishedAt should be formatted as MM/DD/YYYY HH:MM AM/PM.
 - b. The news image using the “urlToImage” value. Use the Picasso library to display the news image.
4. Upon clicking a news item it should open the url for the selected news article in the browser. You should use the “url” that is returned in the news item JSON.