



Electrical and Computer Engineering Department
ENCS5150 – Advanced Computer Systems Engineering Laboratory

Summer Semester – 2023/2024

Dr. Ayman Hroub

Submission Deadline: August 15, 2024 at 23:59

Team Work: Teams of up to two students.

Project Title: A Very Simple Flight Reservation System

Grade: the project will be graded out of 135. The project grade weight is 45%.

Project Description:

In this project, you will develop an android application to help an airline company to manage their flight reservations. There are two different types (roles) of users of this application:

1. Airline admin
2. The user (passenger).

The application should support any number of users of both types. It allows both types of users to

1. Signup
2. Sign in
3. Logout
4. Edit profile

When the application starts:

- It should load flight data from an API that you create (which contains data created by you, there should be at least 10 flights).
- If loading the data fails for any reason (e.g., no internet connection, faulty API URL), the user should be redirected to a "Failed to Login" activity, which has "retry" button, when clicked it should retry to connect to the API.
- If the data loads successfully, redirect the user to a login page where they can either log in or sign up.
- Upon successful login, the application should display the five flights with the closest departure dates.

1. Admin user can do the following:

- a. Create a new flight with the following information (for simplicity, assume all flights are direct):
 - Flight number
 - Departure place (city)
 - Destination (city)
 - Departure Date
 - Departure Time
 - Arrival Date
 - Arrival Time
 - Duration
 - Aircraft Model (e.g., AIRBUS A320 JET)
 - Number of current reservations on the plane (will be updated later per new reservation or flight cancelation)
 - Maximum number of seats (plane capacity)
 - Number of people of who missed the flight (this field is filled after the departure time)
 - Booking open date
 - Price for economy class
 - Price for business class
 - Price per extra baggage
 - Is recurrent (the flight might be no recurrent, recurrent daily or weekly)
- b. Edit or remove an existing flight
- c. View the reservations of a certain flight
- d. List all flights
- e. Filter the list of flights by departure and arrival places and dates.

2. User (Passenger) can do the following

- a. Search for flights
- b. Make reservations
- c. Reschedule or cancel reservations

The application should have the following functionalities:

3. Signup, Sign in, and Logout [25 points]

This layout should have a “Sign in” and a “Sign Up” buttons.

- a. **[10 points]** In the main page (first page), the user can enter his or her e-mail address and password that are registered in the database to sign in. This layout must have a check box called “remember me” which will save the email in shared preferences, so that next time the user does not need to re-type the email address.
- b. **[15 points]** The “Sign Up” button redirects the user to another form where he or she chooses the user role. In other words, Admin or Passenger. Each user type has a different signup page.

The following required fields are common between the **admin** and the **passenger**

- **Email address.** Must be in a correct email format. It is the primary key of the user, and the username.
- **Valid Phone number**
- **First name.** Minimum 3 characters and maximum 20 characters.
- **Last name.** Minimum 3 characters and maximum 20 characters.
- **Password.** Minimum 8 characters and maximum 15 characters. It must contain at least one number, one lowercase letter, one uppercase letter, and one special character.
- **Confirm password** field.

c. The following are additional fields of the passenger

- Passport number
- Passport issue date
- Passport issue place
- Passport expiration date
- Food preference (seafood, vegetarian, etc.)
- Date of Birth
- Nationality

If all fields in the sign-up form are filled with valid data, then and only then, the user can be registered. Otherwise, an error message is displayed (toast message), and the missing or invalid fields are colored in red.

4. Home Layout [80 points]

This layout should be a **Navigation Drawer Activity**. The navigation bar should have the following functionality. Each user type will have a different layout view

I. Admin View [40 points]

- [10 points]** Schedule (create) a new flight
- [5 points]** Edit or delete an existing flight
- [5 points]** View the existing flights open for reservation
- [5 points]** View the flights that are not available for reservation yet
- [5 points]** View the flights archive (old flights)
- [5 points]** View all reservations of a certain flight
- [5 points]** Filtering flights based on their departure and arrival dates and places

II. Passenger View [40 points]

- [10 points]:** Search flights by specifying the:
 - The departure city
 - The arrival city
 - Departure Date
 - One way or round-trip
 - Return date in the case of a round trip (this field should be disabled/hidden if the user selects one-way).
 - Sorting the search results by lowest cost or by the shortest trip duration
- [10 points]:** Make flight reservations
 - The passenger selects one flight from the flight search results

- The user information (first name, last name, email, mobile number, passport information, food preference, etc.) are pulled from the user profile. The user can modify the food preference only. In addition to that, the user needs to specify:
 - i. Flight class (economy or business class)
 - ii. Number of extra bags
 - After the user confirms the reservation, he or she will receive the reservation summary and the total trip cost
- c. **[5 points]** Receive notification if there are any changes on any of his or her flight reservations
 - d. **[5 points]** Receive notification as a reminder one day before the flight departure date
 - e. **[5 points]:** View the current reservations
 - f. **[5 points]:** View the previous reservations

Design [30 points]:

- We left the application design for you, we expect a creative design

Remarks:

- Use tween animations (translation, rotation, ...) at least once.
- Use appropriate methods to insert special fields (e.g., time, date) rather than entering them as regular strings. Additionally, allow the user to select the city and aircraft model from a drop-down menu using a list of predefined values.
- Utilize either a SQLite database or Firebase for storing and querying the data.
- It is recommended to create the design of all activities using **Adobe XD** before starting the project. This will help you better manage your project and implement all the required tasks.
- To manage the collaboration between you and your partner and to allow us to track what you have contributed to the project, you should create a **private** GitHub repository and use it to manage your project. Before the discussion of the project, you will be asked to share the repository with the instructor.