ASSESSMENT

UCOL
Te Pae Mātauranga ki te Ao
Universal College of Learning

D303 – MOBILE APP DEVELOPMENT

ASSESSMENT 03 – ASSESSMENT

DUE DATE

November 7th, 2021

(Week 13)

FACULTY

Humanities and Business

DUE TIME

11:59PM (MID-NIGHT)

SCHOOL

Business and ICT

WEIGHTING

30%

PROGRAMME

Bachelor of Information and Communications Technology

SUBMISSION METHOD

Complete All Tasks and Submit your work via Moodle.

LEARNING OUTCOMES ASSESSED

1. Design and develop mobile applications in a major mobile platform
2. Apply current software technologies, framework architecture and standards used in mobile application development
3. Securely transfer local data to a remote real-time database

CONDITIONS

Individual Assessment

LECTURER

M G Abbas Malik

MODERATOR

Sandra Cleland

1. Aim of the Assessment

In this assessment, you need to design and develop an Android App with Firebase using Android Studio and Java programming language.

2. Assignment Overview

This assessment will consist of a series of tasks to be completed in students own time. It is to be completed using the Android Studio, Java language and various techniques learnt in the course using the description given below.

- This is an open book individual assessment.
- Due date is Sunday, 7th November 2021 at 23:59 (Mid-night).
- Total Marks: 302

3. Deliverables

You will create an Android App prototype in Android Studio. You will ZIP the Android project folder and submit it via Moodle.

4. Submission

You will find a Moodle submission link on Moodle to submit your ZIPED Solution. You have to submit your work by the due date and time mentioned above.

5. Tasks

App Scenario

The Android app is for a company called "Tuck Box". This company is looking to provide a lunch delivery service for customers in Palmerston North, Feilding, Ashhurst, and Longburn. **Tuck Box** will initially offer 4 different meals options for customers to select from. There will be some customisation allowed for the options:

- Green Salad Lunch (Dressing Choices: none / ranch / vinaigrette)
- Lamb Korma (Spices Choice: mild / med / hot)
- Chicken Sandwich (Bread Choices: white / rye / wholemeal)
- Beef Noodle Salad (Chilly Choices: no chilli flakes / regular chilli flakes / extra chilli flakes

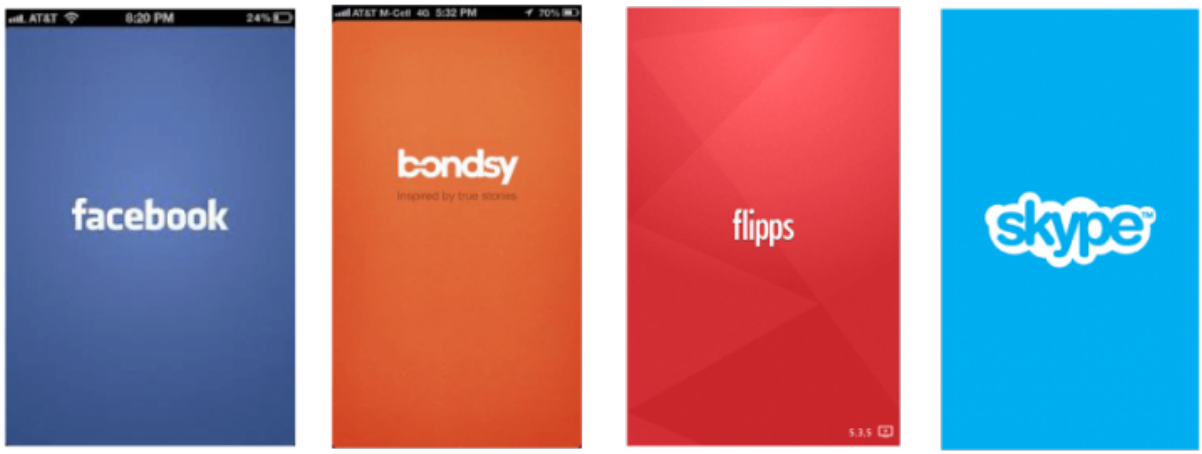
Customers need to place their order by 10.00am for delivery between 11.45-1.45. They are able to choose a delivery window ("11.45-12.15, 12.15-12.45, 12.45-1.15, 1.15-1.45") during which they will receive their ordered meal. Meals are paid on delivery. All users of the app will need to first register by providing a **valid mobile number** before they can order lunches.

Tuck box has left the app design entirely up to you. You can decide on the GUI design, architecture of the app and data model. You must ensure the use of Material Design Guidelines in your application design. You also need to improve the user-experience of your application by implementing smooth animations and transitions.

Task 1. Splash Screen

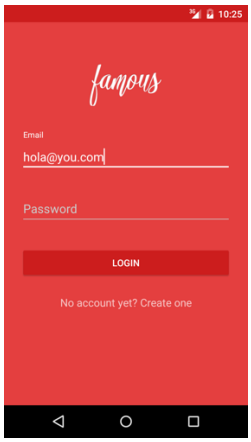
This is a full window Screen that appears for a few second, before your App's main screen show up for User login (Task 2). You can design this app to show the logo of Tuck Box and its business idea to App's users.

Here are some examples of splash screen for different apps:



Task 2. User Login

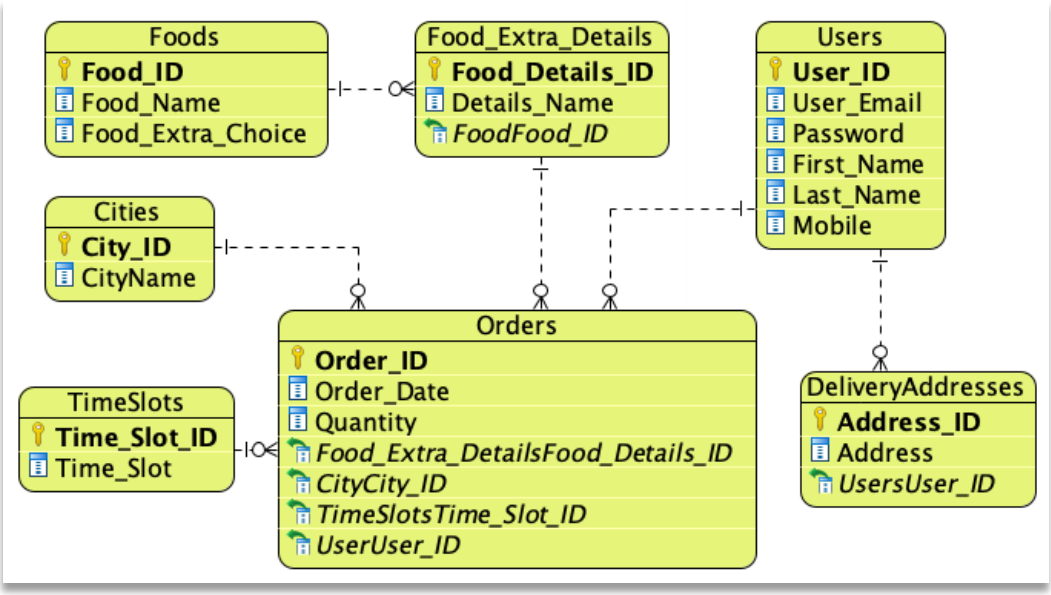
This is the first interactive full window screen that will allow the already registered users to login and use the App's services. This screen will also allow the new users of the App to register themselves and should start the User Registration process (User Registration – Task 6, given below). No App Menu will be shown here on this screen. Here is an example login screen design:



Note: *These examples of GUI design are indicative and only to guide you for you GUI design. You do not need to follow exactly these GUI designs. You can use your imagination and your designing skills to develop this app.*

Task 3. Create you Data Model using Room Library

Create the data model as suggested in the following ERD.



Task 4. Connect you App with Firebase (Google Cloud Services)

Now, You need to connect your app with Firebase (Google Cloud

Task 5. Create your Data Model on Firebase and Fill it with Data using JSON file

Now, you need to replicate your Data Model in Firebase's Realtime Database. You need to make sure that your app can access, add, delete and update data in the Firebase's Realtime Database. Your App's architecture should follow MVVM design pattern as shown below:

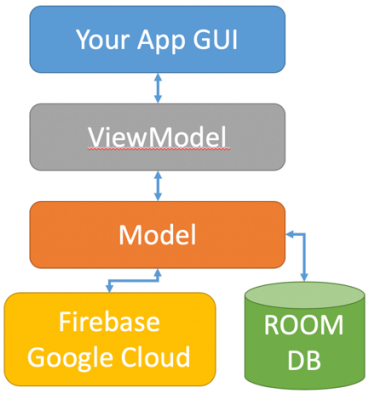


Figure 1. App Architecture

So to build this architecture, you need to implement ViewModel and Model. ViewModel is an access point for your App's GUI to access data. This separates the data access from the User GUI and User Interaction. Whenever the GUI needs data, it will ask the ViewModel and it will provide the data to the GUI after collecting it from the Model. You also need to use **LIVEDATA** concept so that whenever data in the model changes, it is automatically updated in the GUI.

Secondly, you need to implement Model. Model is responsible for the data access whether from Firebase's online Realtime Database or from the ROOM local database. Model is based on the following principles:

- Data from the Firebase's Realtime Data will be saved in the ROOM's local database, whenever App has the internet access or data in the Firebase has changed.
- GUI always gets the data from the ROOM local database via a LiveData object that will keep GUI data updated at all times.

Task 6. User Registration: ROOM Library

In this task, you need to create registration screen that will collect all the necessary information about the user and will register a new user with your Mobile App. For all the necessary information about the user, should be stored in Firebase Realtime Data. The changed data in Firebase should automatically updated in the local database using the ROOM library in Android for storing these information locally in your device. Once, a user has registered, then the user can login using his/her login credentials (email) on the User Login screen (Task 2) to user App's services. This screen should also not have any Application Menu. You will validate the data before you can store it in the Firebase and give appropriate message to the user in case of failure as well as success. In case of success, the app will display the Login Screen. Otherwise, the app will display the error(s) message(s) to user that will help to fix the registration problems.

Task 7. App Services

Once, a user has logged into the App, it will show its **4 services**: Place an Order, Update User Registration Information (personal info, delivery address – one user can have multiple delivery addresses saved in the database), Current Order (Todays orders), and Order History (older orders).

In this screen, an App Menu will be shown that will also provide the two services: Place an Order and Update User Registration Information as ICONs in the Application BAR on top of the screen. In addition to these two services, it will also show the HOME icon in the Menu that will always bring the App back to App Services Screen. Menu bar will also so the current order and order history in the menu bar or drop-down list.

This main user screen, in addition to update and place order buttons, should also show the current order(s) (order for today) for the user, if any.

Task 8. Unser Information Update

This screen will allow the user to updates its Registration Information that divided into two parts:

1. Personal information like name, telephone, email, etc.
2. Delivery Information: the address where User want the food to be deliverer. Again one use can have multiple addresses registered in his account.

User will have the options to update any of these information, described above. You will show appropriate messages to user in case of failure and success.

User can also have the ability to delete his account from the App. Before you can do this, you will show a dialog box to user to confirm his deletion request. Once the account has been deleted, the App will show the Login Screen (Task 2)

In this screen, the same APP menu will displayed as described in TASK 7. You need to handle this intelligently using the concepts of Object Oriented Programming (inheritance).

Task 9. Place an Order

This is the 2nd service provided by the APP. In this service, a valid registered user who has added a valid mobile number in his account can place a food order. Placing a food order is further subdivided into the following steps:

a. Selection of Delivery Region

First of all, a user need to select the delivery region in which he want the food to be delivered. These regions are:

1. Palmerston North
2. Feilding
3. Ashhurst
4. Longburn

User will select can only chose one region. Once the user has selected the region, this information will be saved and we will move to the next step of the order placement.

b. Select Delivery Address

In this step, user will select the delivery address where he want food to be delivered.

c. Display Meal Options (including customization)

In this step, Tuck Box app will display the meals options with customization choices:

- Green Salad Lunch (Dressing Choices: none / ranch / vinaigrette)
- Lamb Korma (Choices: mild / med / hot)
- Open Chicken Sandwich (Bread Choices: white / rye / wholemeal)
- Beef Noodle Salad (Choices: no chilli flakes / regular chilli flakes / extra chilli flakes

Here, user can select the multiple meals, select the choices and add the quantity of each meal. The User can also add a note to the order, if he want to.

d. Selection of Delivery Time

In this step, the user will select the delivery time from the available time slots:

- 11:45-12:15
- 12:15-12:45
- 12:45-13:15
- 13:15-13:45

User can only select one time slot for the delivery timing.

e. Order Confirmation

In this step, app will show the summary of information about the order and ask for user's confirmation. If the user confirm the order, then it will show the appropriate message to user and save the order information in the database. Finally, app will show the main screen (Task 7).

If the user do not confirm the order, app will show an appropriate message to the user to confirm the order cancelation. On cancellation confirmation by the user, it will show the main screen (Task 7).

Task 10. See Order History

In this task, you will show the user's order history to him.

MARKS DISTRIBUTION:

Activity/Task	Details & Marking Criteria	Individual Marks	Total Marks
Task 01 **Splash Screen**	Screen Design Java Code	3 2	5 marks
Task 02 **Loging Screen**	Screen Design Java Code: Login Java Code: Registration	3 5 2	10 marks
Task 03 **Create Data Model**	Food Entity FoodExtraDetails Users Cities TimeSlots Orders DeliveryAddress	3 3 5 2 2 11 4	30 marks
Task 04 **Connect to Firebase**		5	5 marks
Task 05 **Create Firebase and add data**	ViewModel Implementation LiveData Implementation Model Firebase Realtime DB	10 10 20	40 marks
Task 06 **User Registration**	Database Design Data Model in Java Screen Design Validation of Data Saving in Firebase Auto Update local DB Success Message Failure Message Transition to Login	10 5 5 7 5 5 2 3 3	45 Marks
Task 07 **Main Screen – App Services**	Screen Design Transition to user Update Transition to Place Order Menu Handling (Inheritance) Current Order Display Transition to Order History	5 2 2 6 7 3	25 Marks
Task 08 **User Info Update**	Screen Design Personal Info Update Financial Info Update/Add Delivery Info Update/Add Delete User Appropriate Messages Transition to other Screens	8 3 3 3 3 3 4	29 Marks

Task 09 **Order Placement**	Region Selection UI Design Storing Region Info Pass info to next Step Meal Options UI Design Storing Meal Info Passing to next step Payment Selection UI Design Showing registered info Adding new info Remembering current info Passing to next step Delivery Address UI Design Showing registered info Adding new info Remembering current info Passing to next step Order Confirmation Screen UI Design Showing Summary Info Confirm Order Button Cancel Order Button Transition to Main Screen	 4 4 2 5 4 1 5 5 4 5 2 5 5 4 5 2 5 5 3 5 2 	 10 Marks 10 Marks 21 Marks 21 Marks 20 Marks
Task 10 **Order History**	UI Design Transition to Order Details Order Details UI Design Transition Back to Order History Transition back to Main Screen	5 2 5 2 2	16 Marks
Overall Design			10 Marks
Transition Animations between Activities			5 Marks
		Total	**302 Marks**