

MADS4006 PROJECT DESCRIPTION

(35% of overall course grade)

SUBMISSION REQUIREMENTS

1. Compress your project files and any related assets into a zip file
2. Name your zip file: `advancedandroid-project-petersmith-10123456.zip`
Substitute `petersmith` and `10123456` with your student name and id.
3. Upload the zip file to Blackboard by the date specified in the dropbox.

Please note that a submission will not be graded if:

1. The project does not compile
2. The project is missing required files or assets
3. The submission is late
4. The submission contains academic dishonesty.

ACADEMIC INTEGRITY

This assessment must be completed individually.

Providing **any help** to other students is **not permitted**.

Help includes, but is not limited to:

- Exchange of ideas, advice about how to approach the problem
- Sharing website links or reference materials
- Sharing of devices, such as laptops, physical IOS devices, or MacInCloud accounts
- Sharing of credentials to college provided or personal email accounts, Google Drives, Dropbox, Blackboard, or similar

Communication with individuals either in the class, or outside the class, is strictly prohibited.

Any learner suspected of breaking the above rules will not have their submission marked; and, a report will be filed with the College's Academic Integrity department.

To prevent issues of academic integrity, it is imperative that you are vigilant that your code is not duplicated, in part or whole, by another learner. Duplication can mean: identical code; or code that is semantically identical in naming and overall construction. Should the instructors discover academic dishonesty as defined above and per the College Academic Integrity Policies, **a grade of 0** will be assigned to all parties involved.

PROBLEM DESCRIPTION

In this project, you will be making an Android version of your IOS course's Meal Delivery app; with the following modifications.

1. All data must be stored in the cloud
2. You must provide two versions of the app:
 - a. Administrator app: The administration app that enables the company to modify meal kits and pricing. Any changes made by the administrator app are automatically reflected in the customer app.
 - b. Customer app: Similar to what was produced for the IOS course ([see below for exact requirements](#)). Allows user to order meals through the app.

USER INTERFACE REQUIREMENTS

Where reasonable, use RecyclerViews and master-detail views to present information.

Master-detail is the typical user interaction style we do in class, eg:

- Present your data in a list
- Clicking on an item in the list takes you to a screen with more details about the item.

DATABASE SCHEMA

All data for this app must be persisted to Firestore. You are responsible for developing a reasonable no-sql schema for your app.

FEATURES

1. USER SIGNUP AND LOGIN

User login and registration must be handled by Firebase Authentication. You will provide the user with 2 ways to login:

- Email and password
- One (1) other method of your choosing from the Firebase Authentication list of supported providers: Phone, Yahoo, Facebook, Twitter

2. ORDERING MEALS

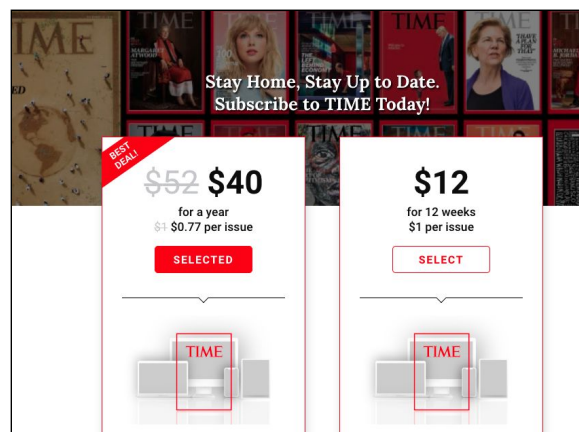
Given the popularity of the original IOS app, the company would like to make the following modification to the Android app.

The company has changed their business model to be a *subscription service*. Instead of ordering 7 days of meals at a time, customers can subscribe to either 1 month or 6 months worth of meals.

Your app should price the subscription appropriately based on the length of time of the subscription and the number of meals you are providing during that time.

Your app should also incentivize the customer to choose a longer subscription period by giving a discount.

Example: Time Magazine offers a discount if you subscribe for a year vs. on a monthly basis.



3. SUBSCRIPTION SERVICE DATA

Data for each subscription is provided by the administrator in the administrator app. Each subscription should have the following information:

1. Length of subscription
2. Price of the subscription
3. What's included
4. Photos of sample meals
5. Allergy warnings (if applicable)

The admin must be able to add photos from either the camera or photo gallery.

Any changes made by the administrator will automatically be reflected in the customer app. Admins can also add new subscriptions.

Changes in pricing or verbiage do not affect the pre-existing orders, only new orders.

4. RECEIPT OF ORDER

After ordering, the customer app should show a confirmation receipt of the order. The receipt should display:

- Summary of the order
- The amount that will be billed to their credit card (after tax)
- A randomly generated order number consisting of uppercase letters and numbers:

Example: M 2 X 5 5

Example: A C 7 2 B

5. PICKING UP THE ORDER

The user picks up their prepared meals directly from the store on a monthly basis.

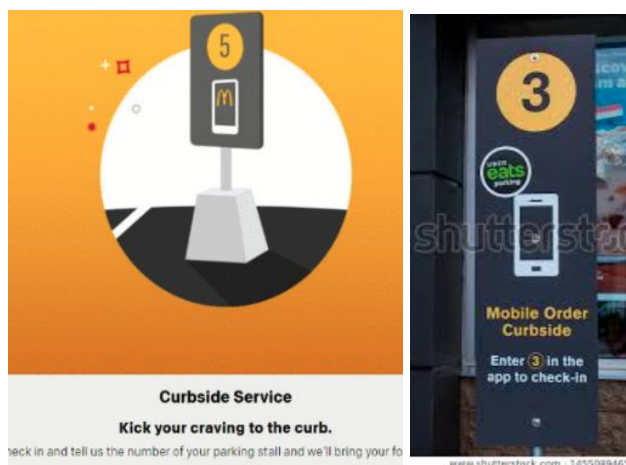
Because of the large quantity of meals in each order, the user needs to wait 1 week after their initial order before they can pick up their first order. Subsequent orders can be picked up exactly 30 days after the initial order date.

Please display the date of order pickup in the app.

6. ARRIVING AT THE STORE

Due to social distancing restrictions, the store has issued *curbside pickup* for all orders. The store has 3 parking spots that are specifically designated for pickup.

Example of curbside pickup at McDonalds



Your app must provide a way for the user to “CHECK IN” when they reach the store. When they “check in”, the app uses the device’s GPS or Wifi to check if the user’s location is actually within 100 meters of the store.

If yes, then give the user a way to enter which parking spot they are in (spot 1, 2, 3).
If no, then the app should display a message that says: "You aren't at the store yet! Please retry when you are at the store".

The user, their order number, and the parking spot will automatically be available in the administrator app.