



Assignment 2 - Android Application

Important Information

This assignment is due on **SUNDAY 24TH April 23:55** (Week 8).

This is an **individual** assignment.

This assignment is worth 20% of your final grade.

This assignment addresses ILOs 1, 2 and 3.



Marking Rubric

[Assignment 2 2022](#)

NOTE: Turns out the rubric doesn't show weighting of each component to students (it shows for me, the Unit Coordinator). So, I have added a table of weightings to the end of this report. Sorry about that!

Background

In assignment 1, your task was to design, prototype, and evaluate a mobile application for stroke rehabilitation. Your task for this assignment 2 is to now implement an **Android application** *loosely* based upon that prototype which provides the required functionality from the [Assignment Theme](#).

The goal of this assignment is to assess your ability to implement Android Applications, and to demonstrate your understanding of the fundamentals of mobile device programming when applied to a practical context.

Specification

You must create a *native* Android application (i.e. you cannot use Flutter or other Cross-Platform tools for this assignment).

Aside from checking for functionality, your program will be assessed on its robustness (i.e. does the application crash), usability, and aesthetics. You will not be assessed on your coding style, commenting, or indentation, however given the scale of this application, you are advised to use good programming practice to assist in development.

UPDATE: What is "Robustness"?

When marking this assignment, we will be checking for bugs by trying to break your application. Common things we will do are:



- Try and enter words/letters into a number text box
- Try various unexpected routes throughout your app

You will only be marked down for very obvious bugs. We *won't* be trying very crazy edge-cases like seeing how the app works without internet connectivity for example.



Clarification from Assignment 1

The following points clarify and *add on* to the [Assignment Theme](#).

You do not need to implement *any* of the following functionality:

- Sign-in/register/multiple users
- Videos or *complex* instructions
- Payments/microtransactions

Even if your prototype had these features you do not need to implement these things. Similarly, if your Assignment 1 prototype is **missing** some of the features specified here and in the Assignment Theme, you should still include the features in this assignment.

There are *no marks* for how well your assignment matches your Assignment 1 prototype.

Data Persistence

Data entered by the user should *persist* between runs of the application. You **must** use Firebase Firestore as your data persistence method (i.e. you *cannot* use SQLite or SharedPreferences as your primary persistence method).

Because your app will be connecting to the database on the Firebase servers, it means that the data you see in your app will be *the same data* that your marker can see when they mark your application. As a result:

- It is okay for your application to already have data in it when it is marked
- The data in the database when you submit it should be sensible (not things like "asdf" "blah" etc), no swears, but funny things are allowed to make the marking process more fun 😊

- You should check that your application still works when there is nothing in the database (i.e. it should not crash, and should function properly if there are no attempts or button presses).

For a top-level assignment, pictures associated with attempts will persist in the database. However the mark for this is in the CRA is associated with the “Camera/Photo” item, not the “Data Persistence” item (i.e. you can still get an HD on “Data Persistence” if your pictures don’t save).

Devices / Screen Size

In mobile development, a big issue is that of making sure our applications look good and function well on the wide range of devices that are available (with varying screen size, aspect ratio, DPI, etc). We use tools like ConstraintLayouts and detection of screen size (not covered in this unit) to provide alternate layouts for different devices.

For the purposes of ensuring this assignment isn't too difficult (within the time frame provided), you are only required to ensure your application looks good on **one** given device (/emulator) of your choice.

Similarly, you may choose design your application to only work with **one** given screen orientation (i.e. portrait or landscape).

To assist with marking, please ensure you indicate either in your documentation or on MyLO submission notes what device and orientation you have tested your layout on, so that your marker may use the same one.

Designed Exercise




The exercise you designed in Assignment 1 was designed by you, without thought for how difficult it would be to code. As such, the difficulty of replicating this exercise in real code will vary greatly from student to student.

To help alleviate the effects of this, the following rules apply to this part of the assignment:

- The weight of this part of the assignment is relatively small (10%)
- You may choose to change your designed exercise for this assignment
 - This may be a small change, however completely changing the exercise should be first **discussed with the Unit Coordinator**

(Drag and) Drop it Like It's Hot

Many of you will have designed an exercise with dragging and dropping as part of the interactions. The code and documentation for this aspect of Android can be a bit confusing, but here are some starting references:

- <https://developer.android.com/guide/topics/ui/drag-drop>  (particularly the part "Respond to drag events: An example")
- <https://www.raywenderlich.com/24508555-android-drag-and-drop-tutorial-moving-views-and-data> 
- (if you find any better resources, please post them in [#references-resources](#)  on Discord!)



Some concepts to be sure you understand before diving into this:

- `ClipData` (drag data, this can be really simple in your app)
- `startDragAndDrop` function
- `setOnDragListener` and the associated events such as `ACTION_DRAG_STARTED` and `ACTION_DRAG_LOCATION`, `ACTION_DRAG_ENDED`

Bonus note, somewhat unintuitively, if you want to modify the X/Y position of the dragged item, you should call `setOnDragListener` on the PARENT `ConstraintLayout`, NOT the thing being dragged. In that case, only use `ACTION_DRAG_LOCATION`

Use of Tutorial Code and Outside Code

You are more than welcome to use the Week 5 tutorial base code or completed Week 5 tutorial work as a base of your assignment—in fact this is encouraged. You do not need to reference this in your assignment.

You are expressly **prohibited** from using any other code online as a *template* for this assignment. Small snippets of code (such as stackoverflow answers and definitely code from the Android documentation) may be used with code comments showing the URL of where the code came from.


Third-party libraries (for example, for things like date-pickers, data persistence, user interface elements, drag/drop, and camera/sharing, etc.) **must not be used**.

While the tutorials and lectures in this unit will have shown you how to build a simple application with navigation, listing data, updating data from a Firestore database, sharing and camera functionality, it is expected that you may come across problems you will need to solve yourself by researching documentation or guides online. If you do so, you need to include this information in your documentation (see below).

It is expected that before asking your tutor or the lecturer for help, that you have already done some basic research on your problem or error message. After that, we are more than happy to provide as much help as possible (it's what we're here for!)

Exception: Use of Glide and Firebase Cloud Storage for HD-level Camera/photo data persistence



Storage of images in a Firebase database is a slightly complex topic given the 1MB limit of Firebase documents. You could just resize your image to be quite small, and store the base64 encode of the image in a standard Firebase field, or you might opt to use Firebase Cloud Storage, documented here: <https://firebase.google.com/docs/storage/android/download-files> .

One approach in that documentation talks about using a third-party plugin called Glide (although the bytes[] alternative doesn't need this). You may use this plugin **ONLY** for the data persistence of images component of the assignment.

Assignment Submission

The following files must be submitted via MyLO before 23:55 on Sunday 24th April (Week 8):

- One zip file, containing the project files.
 - You should create this ZIP file using the File -> Export -> Export to Zip option. Submit the ZIP file which is created.



Assignment Submission Box

[Assignment 2 - Android Application](#)

Plagiarism and Cheating

Practical assignments are used by the Discipline of ICT for students to both reinforce and demonstrate their understanding of material which has been presented in class. They have a role both for assessment and for learning. It is a requirement that work you hand in for assessment is your own.

Working with others

One effective way to grasp principles and concepts is to discuss the issues with your peers and/or friends. You are encouraged to do this especially on the class Discord. We also encourage you to discuss aspects of practical assignments with others. However, once you have clarified the principles of the problem, you must develop a solution entirely by yourself. In other words; you must develop the application *yourself*. You can discuss problems, but not share entire solutions (one or two line code-snippets on Discord are fine). Assistance with solutions should be provided by staff.

Cheating

- Cheating occurs if you claim work as your own when it is substantially the work of someone else.
 - This includes the use of third-party code from online resources.
- Cheating is an offence under the Ordinance of Student Discipline within the University. Furthermore, the ICT profession has ethical standards in which cheating has no place.
- Cheating involves two or more parties.
 - If you allow written work, computer listings, or electronic versions of your code to be viewed, borrowed or copied by another student you are an equal partner in the act of cheating.
 - You should be careful to ensure that your work is not left in a situation where it may be used/stolen by others.

Where there is a reasonable cause to believe that a case of cheating has occurred, this will be brought to the attention of the unit lecturer. If the lecturer considers that there is evidence of cheating, then no marks will be given to any of the students involved and the case will be referred to the Head of Discipline for consideration of further action.

But Weight, There's More!

The [Assignment 2 2022](#) Rubric doesn't show weights to students (thanks Obama Mylo), so here is a table of weights for each of the CRA items in the assignment (all numbers = marks):

	HD+	HD	DN	CR	PP	NN
History - List	15	12	10.5	9	7.5	0
History - Deleting	10	8	7	6	5	0
Prescribed Game - Basic Functionality	10	8	7	6	5	0
Prescribed Game - Goal Mode	10	8	7	6	5	0
Prescribed Game - Customization	10	8	7	6	5	0
Designed Exercise	10	8	7	6	5	0
Data Persistence	20	16	14	12	10	0
Camera	10	8	7	6	5	0

Sharing	5	4	3.5	3	2.5	0
TOTAL	100	80	70	60	50	0

