# Mobile Assignment 2 Personal Statement

## **I**ntro

The purpose of this document is to highlight some of the learnings I have made from my experience developing my mobile for the Mobile Development module.

## Background

The application I chose to develop is a simple fitness app for tracking workouts. The idea came from an app that I use regularly. I was interested in seeing what goes into making an app such as this and how far I could take it in a short period of development time.

## Highlights

### Android Studio

After a painful beginning I grew to like developing in android studio, my usual IDE of choice is vscode with a number of plugins and other customisations. Android studio out of the box suited me quite well. The painfulness of it is debugging with the emulator, but its very easy to hook your PC and Mobile up on wireless debugging. I found it much faster to develop this way.

### Kotlin

The last couple of assignments previous to this one had been Web based, so I had grown used to Javascript. Initially with Kotlin, I found it very restrictive due to its typing – but once I had completed a basic implementation of the app I realised how much easier it is to develop with null safety and static types, making changes after the initial implementation was much easier than it was in Javascript. The initial setup took longer but once the typings are in place, strict returns will usually leave you free to change up any functions implementation. This made me realise how weakly typed the backend of the app actually was and I’ll most likely be dropping Javascript for Typescript going forward.

### Jetpack Compose

I really enjoyed developing with Compose, especially compared to the XML way from the first assignment.

## Lowlights

### Java Date Util / Kotlin Date Util?

The app would crash whenever it received a String date in the format “dd-MMM-YYYY” from the backend, no matter what format I tried to parse it with, Java / Kotlin could not parse the date, ultimately I changed the backend to just give me formatted dates from moment.js

### Buttons

Buttons in Jetpack compose should take a composable function as an onClick argument – I don’t see why they wouldn’t, it seems so much cleaner.

### Navigation in Jetpack Compose

The NavHost element is great, very easy to understand and set up. Passing arguments between elements is different though than the normal API. In the standard API you can pass parcelables, but they removed this for the Jetpack Compose implementation. It’s described as an anti-pattern and the preferred data retrieval pattern would seem to be re-querying a data source with a string ID. I can’t say I agree.