SWEN325 – Software Development for Mobile Platforms

SWEN325 – Assignment 1

# Contents

[Contents 2](#_Toc48301328)

[Introduction and Task Description 3](#_Toc48301329)

[Core: Pre-processing of COVID-19 cases in a given area [40 marks] 4](#_Toc48301330)

[Select a dataset and analyse (10 marks) 4](#_Toc48301331)

[Describe the results of each technique used on the one dataset (10 marks). 4](#_Toc48301332)

[Pipeline used for Classification with Naïve Bayes 4](#_Toc48301333)

[Pipeline used for Clustering with Simple K Means 6](#_Toc48301334)

[Pipeline used for Linear Regression 7](#_Toc48301335)

[Identify how these aspects of the techniques are different (10 Marks) 10](#_Toc48301336)

[Revisited Business understanding (10 marks) 10](#_Toc48301337)

[Part 2 Completion: Feature Importance to COVID19 cases [40 marks] 10](#_Toc48301338)

[Business Understanding, question to be answered and data sources (10 Marks) 10](#_Toc48301339)

[Merge the Data sources using WEKA (10 marks) 10](#_Toc48301340)

[Use Dimensionality Reduction (10 Marks) 12](#_Toc48301341)

[Analyse the output of the data (10 Marks) 15](#_Toc48301342)

[Challenge: Visualisation of results [20 marks] 16](#_Toc48301343)

[Visualise the most interesting results (10 marks) 16](#_Toc48301344)

[Consider the consequences and ethics of reporting your findings (10 marks) 16](#_Toc48301345)

[Possible consequences 16](#_Toc48301346)

[Background - supplemental 17](#_Toc48301347)

[Features in the original files 17](#_Toc48301348)

[Using suitable techniques Merge the files together (10 Marks) 17](#_Toc48301349)

[Features in the combined file - output-data.csv 18](#_Toc48301350)

[Features of secondary combined file output-hospital-data.csv 19](#_Toc48301351)

[Dimensionality Reduction 19](#_Toc48301352)

[CRISP-DM Steps 21](#_Toc48301353)

[CODE 25](#_Toc48301354)

[Main-countyHealthSummary.ps1 25](#_Toc48301355)

[Main-hospital.ps1 26](#_Toc48301356)

[Get-canary 28](#_Toc48301357)

[Get-FutureStats 28](#_Toc48301358)

[Get-MaximumIncreaseInCasesOverDays 28](#_Toc48301359)

# Introduction and Task Description

## Written Report (To Be Done Individually) (Worth 60%)

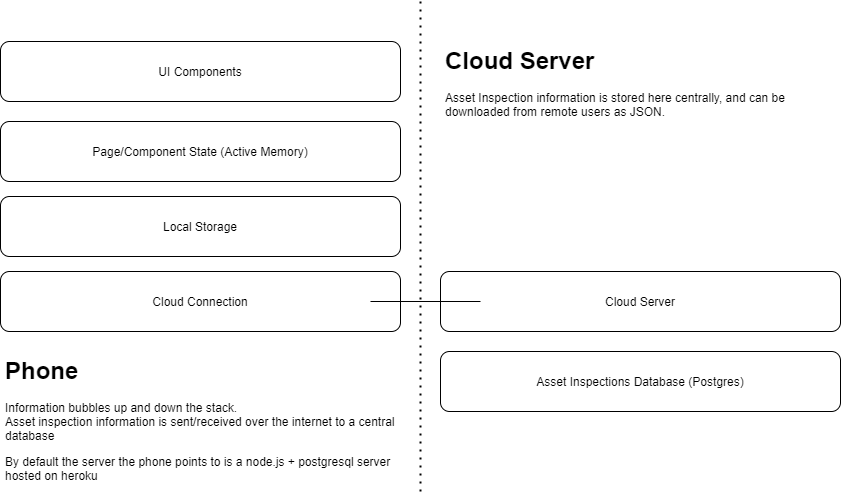
The reports are assessed individually and therefore if you are in a group of two or three people, you will need to write it yourself and present a different usability test plan from your other group members.

### What to submit.

Submit a file called 325-a1-report-username.pdf that contains:

* Description of the overall architecture of your application. 2 Pages of A4, Worth 10% out of 60%
* Include a description of how you organised your source code to match your architecture.
* Description of how you utilised at least 1 major existing external component (e.g. Firebase but hopefully much more interesting one than that) to provide either persistent data storage or other major functionality with a description of how it was integrated into your App architecture. 1+ Page of A4, Worth 10% out of 60%
* Reflective report on Ionic as a framework for App design including its advantages and disadvantages. 2 Pages of A4, Worth 10% out of 60%
* Usability test plan 3+ Pages of A4, Worth 30% out of 60% that includes:
  + Goals for the usability test
  + Format of the usability test
  + Target users (and how many)
  + Tasks that users have to perform with the App (including screen shots)
* Finally, include an appendix with as many pages as there are separate screens in your app, with each page containing a screenshot of the screen and a one paragraph description of its design.

# Application Architecture – TO DO (2 pages)



For Architecture, you **need to** describe how your code is structured into layers and refer to the book we discussed in the lectures as well as provide relevant diagrams and descriptions of how and why your code design fulfils the architecture you described. You **have to** show significant insight to get A grade for this part.

The general application architecture splits the application into 6 major sections, four on the device and two located in the cloud.

Include a description of how you organised your source code to match your architecture.

## Source code organisation

src

│ App.tsx

│ index.tsx

│

├───components

│ │ Default.css

│ │ ExploreContainer.css

│ │ ExploreContainer.tsx

│ │

│ ├───assetListItem

│ │ AssetListItem.tsx

│ │

│ ├───currentSession

│ │ CurrentSessionContainer.tsx

│ │

│ ├───home

│ │ HomeContainer.tsx

│ │ IntroductionContainer.tsx

│ │

│ ├───map

│ │ map.css

│ │ Map.jsx

│ │

│ ├───menu

│ │ Menu.css

│ │ Menu.tsx

│ │

│ ├───newAsset

│ │ NewAssetComponent.tsx

│ │ NewAssetContainer.tsx

│ │

│ ├───previousSessions

│ │ PreviousSessionsContainer.tsx

│ │

│ ├───settings

│ │ SettingsContainer.tsx

│ │

│ └───upload

│ UploadContainer.tsx

│ UploadInformationContainer.tsx

│

├───models

| AssetsContext.tsx

│ ServerConverter.js

│ SettingsContext.tsx

│

├───pages

│ Page.css

│ Page.tsx

│

└───secrets

googleMaps.js

### Components – NOT COMPLETE

Figure : Source code organisation for components and an individual component

Figure : Default CSS stored in the root of the components folder

### Data – NOT COMPLETE

#### Mobile Application – State, Local and connections to Cloud

#### Mobile Application – Secrets

Figure 3:Secrets and API Keys stored in a folder that can be ignored when replicating (no api keys in git)

# Major External Components – TO DO (1 page)

For External Component, you **need to** describe how your way of implementing it fits into your proposed architecture and provide technical details of how it works and **justify** your choices and why it is the best way to fit the component into your app architecture and design.

Description of how you utilised at least 1 major existing external component (e.g. Firebase but hopefully much more interesting one than that) to provide either persistent data storage or other major functionality with a description of how it was integrated into your App architecture. 1+ Page of A4, Worth 10% out of 60%

## Map

## Camera

## Remote Server (using postgres)

# Framework Reflection (2 pages)

For Framework Reflection, you are expected to provide a lot of technical details, examples, and justifications for how the Ionic or React Native helped or hindered your development. It is essential to not just describe how it works and whether you liked it but rather justify and show technical examples of how it affected your application design and what further improvements or insight you could have hypothetically provided to framework developers.

Reflective report on Ionic as a framework for App design including its advantages and disadvantages. 2 Pages of A4, Worth 10% out of 60%

# Usability Test Plan (3 pages)

## Application flow diagram

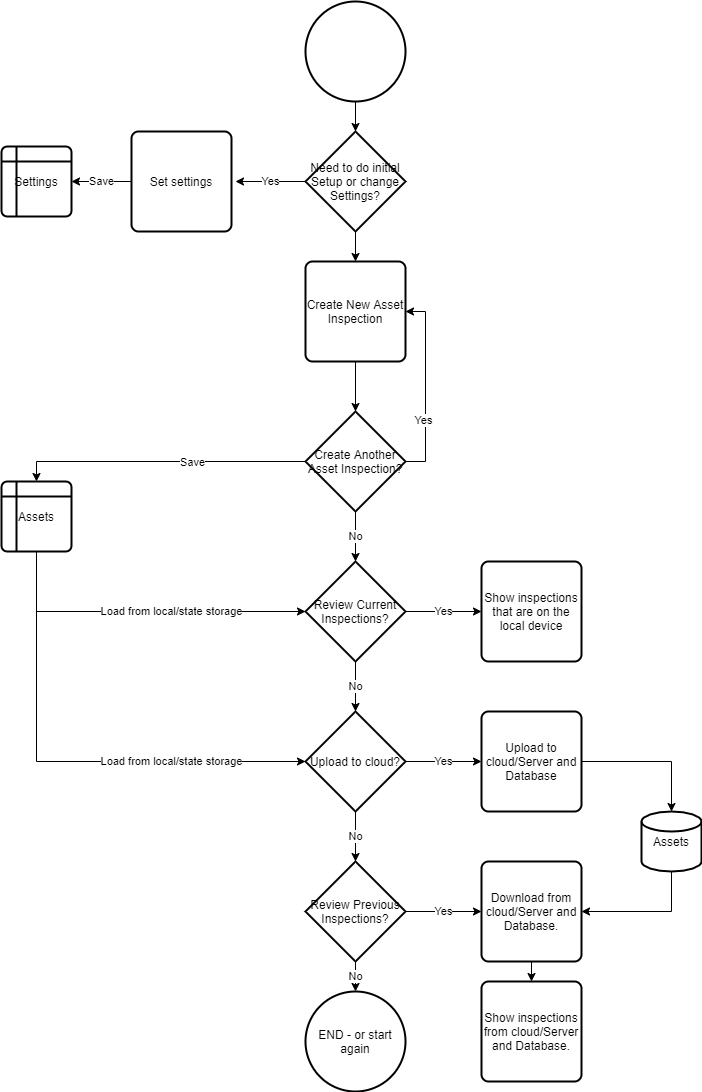


Figure 4: Application flow diagram, showing choices and flow through the application

## Goals for the usability test – NOT COMPLETE

The goal of the usability test is to identify design flaws, to allow for them either to be corrected in their entirety or to be mitigated with other elements.

## Format of the usability test –TO DO

## Target users (and how many) –TO DO

## Tasks that users have to perform with the App (including screen shots) –TO DO

# Appendix with screens

Finally, include an appendix with as many pages as there are separate screens in your app, with each page containing a screenshot of the screen and a one paragraph description of its design.