Project Description

IT Project, COMP30022, is a subject that basically dwells around a **software engineering project** where you will practice and demonstrate your skills you learned through out your degree. Students will be working on a project through out the semester and deliver the **outcome by Week 12** of the semester. All of the assessment components of this subject are derived from this engineering effort. During Week 1 lectures, we will discuss how the assessment components are divided into different components. Essentially, **every aspect of your engineering efforts will be assessed**, i.e., project code is not the only component we will look at while deciding your marks.

The project is a **team project** and students are expected to **pick a team during week 1**. The teams are supposed to have **five students**. All members of the team should contribute to all aspects of the project. Every week, there will be tasks announced during the lectures that the students are expected to complete as a team in the following weeks. It is important to note that most of the work will start during the tutes but will end with your work that you have done with other team members on your own time. And you should expect to meet team members and work regularly outside subject lecture, tute, lab hours. Basically, the tutors and the lectures are there to guide you but as this is your finishing project, it is ultimately you and your team who should be driving the project to a successful completion.

The project is **modelled after a start-up company**. Basically, each team will assume that they are starting a high-tech company to **develop a smartphone application**. We will not focus on financial aspects of the company in the project but rather on the software engineering aspects. Aligned with the direction of mimicing a small start-up, you will be choosing your own finishing project to develop (and if you wish put it on an online app store). We give the following guidelines for the projects so there is some limitations on what you can choose as a finishing project.

The limitations regarding your project are commonly not on the side of the logistics of what programming language or platform you can use. You are encouraged and will be directed to choose and setup your own development environment and platform, with minimal restrictions from the subject coordinators. Having said this, **we would strongly encourage choosing Android** based devices as it is a common platform that we have inhouse know-how, and historically, most of the student cohort for this subject have chosen this previously and have given positive feedback about the experience to us in comparison to many other platforms. Other platforms, if you agree as a team, could also be chosen but the less popular a platform is, the less likely you will find help about it (from us, online, etc.)

Historically, we have seen and accepted the fact that most want to use their own laptops, phones etc. Most of the students have done so with success so we do not limit you on this front either. Regardless of the platform of choice and how you work, students are expected to **take their own backups** of the project work though. Also, technical help on development issues could be given by the tutors if the tutor feels that the team is really stuck on a non-issue problem -- but loosing the project due to lack of backups or some installation of some package going wrong on your devices will not be seen as an acceptable reason for lack of delivery. It is also important to note that in this subject, as it is common in the IT industry, the project is not about learning a particular environment, but rather developing something that people will use and hence picking and learning the right environment for it is the aim. Most of you will be new to your platform of choice but will soon realize the IDEs etc are trivial to pick up and the language/development platforms are similar to what you have used earlier. And as an IT professional, you should now be picking up and switching platforms with little effort depending on the project you are working on.

Back to the essence of the project and the guidelines of picking a project: Teams will be implementing a **location-based service app for smartphones**. The app is **about helping an elderly person to reach a destination**, e.g., go to a library, a bus station, etc. Our target audience are novice users of smartphones and the key theme of this app is **giving them real-time help in the form of online media rich**

data/support. You are expected to use the fact that your development is happening on a mobile platform with GPS, motion sensors, camera etc are available to you. Thus, you should design an app with abundant use of these sensors etc. Each team will define their own Location-based Service that basically serves the above mentioned purpose. For example, you can chose to give support to our elderly user in terms of grabbing his screen and connecting online to his son/daugther and communicating what the elderly user sees to get help to him. Other key aspects of your app are given below that you have to do to be deemed to have finished a complete project. After the **approval of their tutor** regarding their app/project, teams will commence the development effort. We will go through a semester of engineering practice to achieve the best outcome for each team. (It is important to note that the app cannot be a piece of software that any one of the team members have implemented or helped implement before.) The main aspects that we want you to have in this app are:

- The app has to be a **location-based service with abundant use of rich location and mobile sensor data** and should have media-rich data processing capabilities, such as enabling others to grab the screen of the elderly user, exchange images, videos, descriptions and annotations on maps etc.
- The app has to involve **at least two devices communicating** with each other (two emulations communicating on two different machines are acceptable if you do not have two devices but after finishing your development we encourage students to try their best to use their own real devices to demonstrate the app). The two devices should be communicating and exchanging rich data to help the elderly user go to a destination.
- The app has to **have continuous updates and frequent communication** (such as location updates and data exchange).
- The app has to **access a designated server** to store user data, as well as the state of the app if users decide to interrupt the activity for a while (the server to store this data could simply be running on a PC).
- The app should have a **voice and text chat functionality** as well.

Students are **allowed to use packages** and services from various vendors for their implementations. (We are guessing, you will soon be using online map services for example which is also fine.) If you choose to use such packages, you will need to install it on your own laptops etc and maintain it yourself. Limited advise is available for some packages if you are stuck as we have in-house experience with a variety of packages. Having said this, ultimately responsibility for choices, integration issues and any other maintenance related to packages sits with the teams. Thus, there is a trade off between doing all on your own and remaining flexible, versus rapid development etc. We will not give extra credit for using a package but we do accept that it is more realistic to use some of the packages for development and thus for sense of achievement in some cases. You need to communicate your tutor and clearly identify in the code which package does what.

The **code developed by each team should meet the requirements stated in the handbook for COMP30022** and any other vendor code is not going to be judged/marked. Thus, if you choose to use a package or a combination of packages and online services, you still need to develop about 5000 or so lines of your code for a team of 5. This means if you find a combination of packages out there that help you with all of the above main requirements/aspects of the project, then you need to make sure that you step up your specs to implement a more challenging version of your project. Thus a simple app done through patching some packages should not be the aim of your finishing project. The app can be implemented without using any external packages and students will not loose any marks due to this. Having said this, please be realistic in setting expectations for yourself and delivering many tens of thousands of line code should also not be your aim.

As it is the case for all subjects, **plagirism, collusion among students/teams, etc will not be tolerated**. If in doubt, please ask the lecturer, e.g., whether a certain type of discussion among students will be seen as a collusion. It is also important to note that the projects are monitored through out the semester by the tutors as they will become apart of your teams while accessing repositories etc. Thus, it is very easy in this subject to notice odd development efforts. In addition, trading duties among team members to avoid coding, design etc, is not acceptable. Basically, all members should contribute to all aspects of development unless it is explicitly authorized by the lecturer for specific types of development