



DELIVERY ASSISTANT

Assessment 3 - COSC2196

Streamlining deliveries for enterprise.

TechPro's

Arin Edwards, Dylan (omitted), Jacob Darling, Joshua Coppen, Ryan Bullock and Seth Undertown

Table of Contents

PROJECT PLAN – TOPIC	1
PROJECT PLAN – MOTIVATION	1
PROJECT PLAN – LANDSCAPE	1
PROJECT PLAN – AIMS	2
PLANS AND PROGRESS – OVERVIEW:	3
PLANS AND PROGRESS – ORIGINAL MOTIVATION AND PRIORITISING OF AIMS:	3
PLANS AND PROGRESS – WHY ANDROID?	4
PLANS AND PROGRESS – WHY GOOGLE MAPS?	4
PLANS AND PROGRESS – DETAILED PLANS	5
APPLICATION FLOWCHART	5
DESIGN / INTERFACE	6
TIMEFRAME	7
PLANS AND PROGRESS – ANDROID DEVELOPMENT INFORMATION:	7
PLANS AND PROGRESS – DEFINING ROLES	8
PROJECT PLAN – ROLES	9
PROJECT PLAN – SCOPE AND LIMITS	10
PROJECT PLAN – TOOLS AND TECHNOLOGIES	11
PROJECT PLAN – TESTING	12
PROJECT PLAN – TIMEFRAME	12
PROJECT PLAN – RISKS	14
PROJECT PLAN – GROUP PROCESSES AND COMMUNICATIONS	14
PROJECT PLAN – SKILLS AND JOBS	15
GRAPHICS DESIGNER	15
TECHNICAL SALES / MARKETING MANAGER	16
CUSTOMER SERVICE / HELP DESK	16
BUSINESS ANALYST	17
TOOLS (GITHUB / GOOGLEDOCS)	18
GROUP REFLECTION	19
GROUP FEEDBACK	19
ARIN’S FEEDBACK	20
DYLAN’S FEEDBACK	20
JACOB’S FEEDBACK	21
JOSH’S FEEDBACK	21
RYAN’S FEEDBACK	22
SETH’S FEEDBACK	22

Project Plan

Topic

The idea behind the Delivery Assistant is to help provide an easy way for delivery drivers to find where they need to go, without having the hassle of entering the location into an app themselves. This product will be useful for any delivery company that requires their deliveries to be done as quickly as possible and efficiently. This product will be designed to connect to the instore systems of the business and will then send the address of the delivery location to the Delivery Assistant. It will then automatically enter the destination and show a map that displays how to get to their destination.

We also believe that the Delivery Assistant will increase driver safety while on deliveries. This would happen from the fact that delivery drivers won't be needing to enter the location while driving to save time, and they will be able to keep their phone in a dashboard mount since they won't be required to take it out each time they want to enter a location.

Motivation

The motivation behind our Delivery Assistant project is the need for streamline navigation information in the delivery industry. This idea was thought up by a team member working in the food delivery industry. They noted there's currently extensive hassle involved in entering delivery addresses. The addresses are typically already available in existing systems, though are not synced directly to driver devices (smart phones). Opening navigation and entering an address could take as long as 2-3 minutes for each delivery. By eliminating this inefficiency, a delivery company making just one hundred deliveries a day could save as much as 5 hours in labour costs.

Due to time restraints, drivers currently often resort to setting up navigation while driving, resulting in distracted drivers. Distracted drivers are a real issue - in 2013, over 21% of automobile crashes in America were caused by distracted drivers using mobile devices (Woody 2015). By automating the navigation process, efficiency is increased while reducing or eliminating the risks involved in phone use while driving. Ultimately this would reduce delivery costs while increasing safety on our roads.

Landscape

The landscape of other software with similar features exists. For instance, a similar program exists for the delivery world. Deliverit is a POS (Point of Sale) which has an online ordering system with mobile apps for customers and driver apps for drivers. The driver app is always used for tracking the driver's location. In which a pizza restaurant and the customer can know where the pizza is in context to the delivery location.

A similar program exists called IntelliTrac which is made for tracking a fleet of company vehicles. This software can track hours of work, fuel purchases, employee performance data, Journey replay system, live tracking info and an electronic logbook. This software is more generalized in nature and focuses heavily on tracking, opposed to specialized software for delivery companies.

Woody, C 2015, 'Cell phones are causing more and more car crashes in the US', *Business Insider Australia*, 1 June, viewed 29 April 2019, <<https://www.businessinsider.com.au/cell-phones-causing-car-crashes-and-deaths-2015-5>>.

Project Plan – Aims

With our delivery assistant application, we desire to streamline delivery times by having delivery information synced directly to the driver's device. When the device receives this information, a course would be plotted, and navigation information automatically provided. The driver would not be required to interact with their device at all - this process would be completely automated.

Our aim is to create a prototype application demonstrating proof of concept for our idea. We plan to create a fully functional prototype which will receive addresses from a test server, automatically providing the user with navigation instructions on their mobile device.

Research Feasibility - Our first goal is to research the feasibility of our idea. Though on paper our idea sounds entirely feasible, there may be complications with maps licensing, Android development or other issues. We aim to further research the requirements, licensing and other factors involved in creating an application of this kind.

Creating an Application - Our second goal is to create a functional application. To begin, this application will not include the feature of addresses being synced from a server. Users would need to manually enter addresses. This application will show proof-of-concept for the navigational app side of our project, further showing feasibility.

Adding Features - Our next goal is to add the main features, particularly the address sync feature. Using a test server, we will modify the application to receive a delivery address from a server and automatically provide navigation instructions. We aim to investigate the possibility of further features, though first aim to create a working prototype with the address sync feature. This would allow us to present a working demonstration to potential customers or investors.

Alpha Testing - We aim to complete our application to a level that alpha testing can be done, ensuring our application is ready to be used for demonstrations to potential customers, investors or other interested parties. Though ideal, alpha testing isn't critical for completing our overall aim of creating a working prototype - it would simply ensure any hidden issues are found.

Project Plan – Plans and Progress

Overview:

We have begun the planning and research process for our application. We are researching various technologies and application development methods for use with our project. This process includes researching potential navigation software, such as Google Maps and Open Street Maps. We plan to complete this process by week 6, beginning application development in week 7. Front end and user interface development will begin in Week 9, with completion scheduled for week 10. A test server will be created in week 10 or 11, allowing for realistic in-house testing in week 12. After fixing any identified bugs or issues, we plan to conduct alpha testing in week 15, smoothing out any final issues. A completed prototype application is scheduled to be ready for demonstrations at the end of week 15. It was agreed these dates and timeframes may change based on the outcome of any research in weeks 4-6.

Original Motivation and Prioritising of Aims:

The original motivation behind this product was to help increase efficiency with delivery drivers as they are leaving for their delivery. With the small-town store one of our group members works at, they have an average of 80 deliveries a day and the average time for the delivery drivers to start driving to the delivery location after receiving the order is three minutes. Most of the time spent in those three minutes is dedicated to entering the destination into mapping software on their phones. With this product we believe that time could be drastically reduced and save over an hour of time dedicated to entering their destination. That hour of labour could be better spent on in store duties or by increasing the rate that deliveries are taken.

There are many smaller goals we would like to solve with this software. These goals include: Improving driver safety, single and multiple delivery options (single such as a pizza driver or multiple such as a courier driver) and having an interface for payroll software amongst others. It was agreed these goals would be left for later versions of the program, instead focusing on single delivery address pushing at this time.

Driver safety was originally a big motivation for our application, though during meetings we agreed this may not be a major concern for delivery companies. Instead, we believe delivery companies may have a larger concern around cost(s) and efficiency. With this in mind, we decided to focus more heavily on pushing the costs savings and efficiency increases, with reduced phone use as a secondary selling point. We believe this aligns more closely with what the industry would want. We did however still like the idea of extra features such as speed tracking and device use alerts. It was ultimately decided these would be left for a later version of the application to avoid unnecessary development times, allowing a prototype to be developed in a timely manner.

As a team we would like to explore the possibility of customisations specifically for drivers completely multiple deliveries. A courier driver (as an example) typically completes many deliveries before returning to the warehouse/storage facility. The goal would be to have our software map out deliveries in an organised, logical manner, updating the driver as they complete each delivery with no human interaction necessary. We agreed this feature is beyond the scope of our current project and would also be left for a later version.

Project Plan – Plans and Progress

We made considerations to adding payment services to our application, though on the surface this feature looks as if it would require extensive research. We would need to make considerations to what payment software to use, how it would work (i.e. payWave addon?), security risk(s) and various other factors. On top of this, the addition of a payment gateway would add an extensive workload to the development stage of our project. With all this in mind, this feature was also left until a later version.

Why Android?

Our team decided to focus on an Android application in this project. This decision was in part due to the higher mobile market share held by Android (citation needed), along with the Play store at a glance looking less restrictive than Apple iTunes. We have not yet decided if we will make use of the Play Store, though the less restrictive nature would make this a possibility if decided upon.

The Android Development Kit is well documented, and applications are coded in Java, a language all our team members have experience with. On the contrary, iOS makes use of Swift - a language none of our team is familiar with.

Considerations were also made to the ease of installing third party applications. It was noted having users download a customised version of our application and installing it as a third-party app may be easier. This would allow us to provide customised versions of our application to each customer, providing only the features or compatibility they require. Installation of third-party applications on Android is as simple as ticking 'allow third-party apps' in settings. Apple iOS does allow for the installation of third-party applications too, though it's less documented and a more tedious process.

In summary, our team felt Android would be the most widely beneficial choice due to market share, ease-of-development and ease-of-use for end users.

Why Google Maps?

We have decided that Google Maps would be perfect for testing this application, but we may consider other mapping software for the final release of the product. We believe Google Maps will be useful because Google allows easy implementation of Google Maps into Android applications.

Using the Google maps SDK, we have access to functions that allow for tracking distance travelled, route travelled, speed limits along a route and providing navigation instructions. We only plan to implement the directions for the prototype, though intend to implement further features in later stages of the development. While using the Google Maps SDK we would be billed at least \$0.012 for every navigation query. During testing this won't be an issue as Google provides \$200 in free credit, allowing for testing and development without costs. Based on Google Maps Platform Calculator (Google), we predict our use will be far below the \$200 limit.

Google Maps Platform - Calculator, Google, viewed 14 May 2019, <<https://mapsplatformtransition.withgoogle.com/calculator>>.

Project Plan – Plans and Progress

Detailed Plans - Flowchart

Our application will interface with existing delivery management systems to receive delivery addresses. An order or delivery request will be created in the stores delivery management system, then a delivery driver assigned. Once assigned, our application will come into play. The delivery address will be sent directly to the driver's device, displaying navigation instructions on screen instantly - no input required.

Our application will now work with the existing delivery management system, receiving address information and displaying navigation instructions on the assigned driver's device. This process is completely automated with no driver input required. Once a delivery is completed, the application will return a response confirming the delivery in the delivery management system.

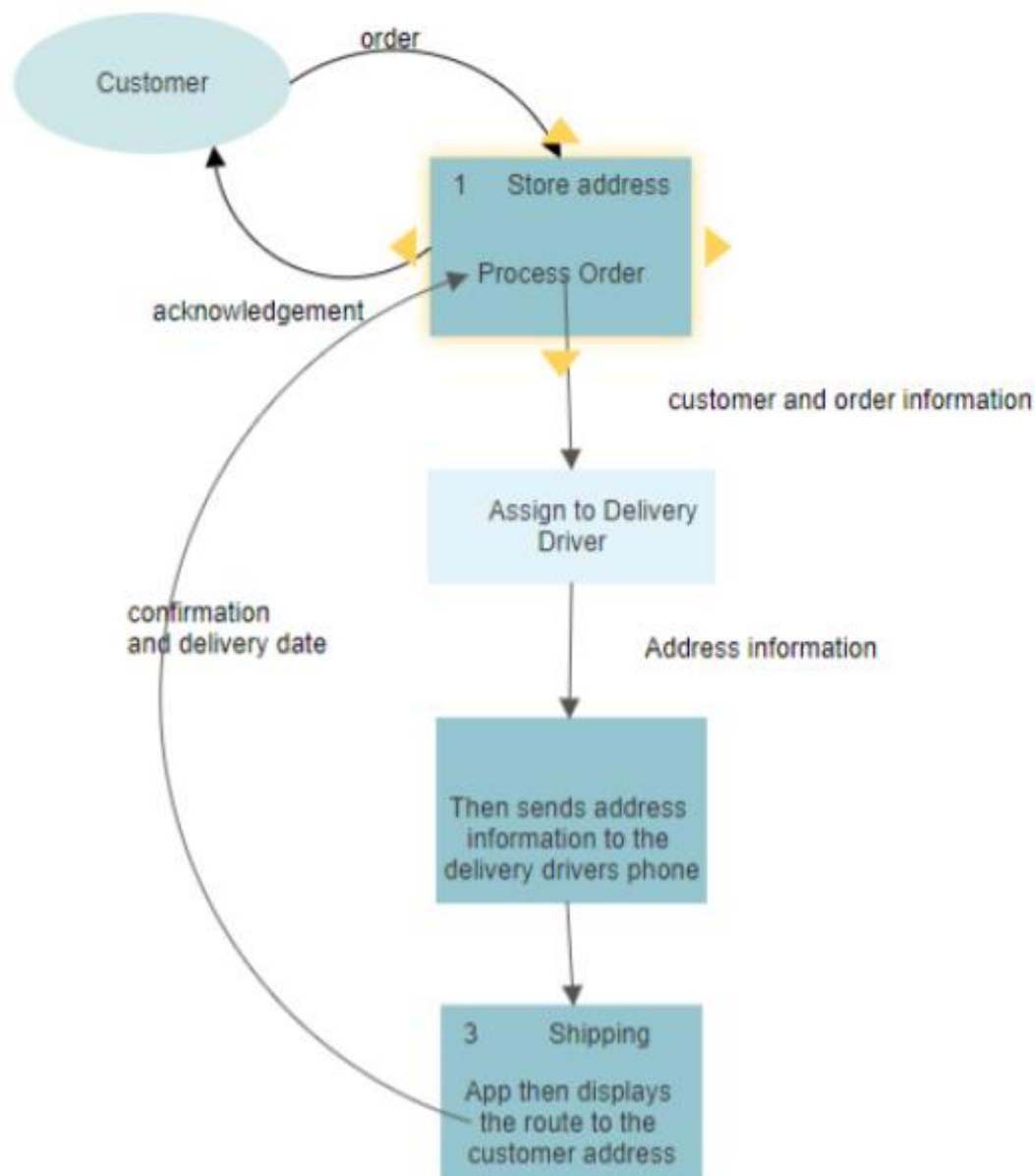


Diagram 1 - Application flowchart

Project Plan – Plans and Progress

Detailed Plans - Design / Interface:

We decided a minimalist design would work best for our application. This restricts the level of distraction and driver interaction, increasing safety. We aim to ensure no driver interaction is required throughout the whole delivery process. The only interaction will be entering an employee and store ID upon start up. Pictured below is a diagram of how we plan the interface to look with no address assigned:



Diagram 2 - No current deliveries assigned / awaiting delivery

Once a delivery is assigned and an address received the driver would see a screen like below. The screen will show only navigation information, along with the driver's name and store/base information. Audible navigation instructions would be played, verbally instructing the driver. Google maps used for demonstration purposes (Google Maps 2019).

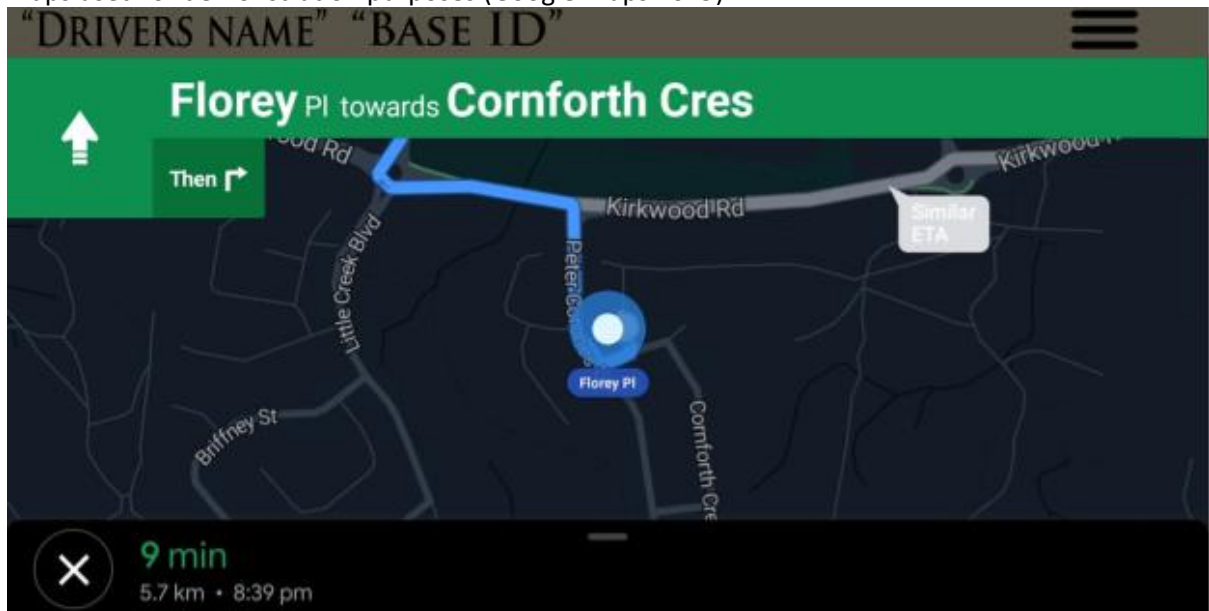


Diagram 3 - Delivery assigned / navigation instructions displayed

Google Maps 2019, Example Map - Florey Pl towards Cornforth Cres, map,
<<https://www.google.com/maps>>.

Project Plan – Plans and Progress

Detailed Plans - Timeframe:

We began the process of creating our application by creating a project plan and timeframe. It was decided we would begin by researching available technologies which meet our needs. This included researching Android Development Kit and Google Maps / Open Street Maps. Information provided previously outlines the outcome(s) of this research.

We plan to begin development in week 7, first working on a barebones application without the functionality or polished user interface. In week 9 the user interface will be added with the application serving as a simplified, standalone navigation app. For testing purposes, this application would prompt for an address input before providing navigation instructions as per the final prototype.

In weeks 10 to 11 we will begin adding the address pushing functionality. A mock delivery management system will be created for the purposes of testing. The functionality to receive address information from this mock server would then be added, allowing us to test our application in a realistic manner.

Weeks 12 - 15 will involve various stages of testing and bug fixes, with real-world alpha testing planned for in the final week. We believe this will allow us to get non-bias feedback on our interface(s) and make any final changes as required. We plan to have a final prototype ready for demonstrations to potential customers and investors by the end of week 15.

Android Development Information:

Android Software Development can be done in multiple languages. These languages are Java, Kotlin and C++. Other languages can be used such as JavaScript and Go but these will need the help of JVM (Java Virtual Machine) language code. The Android SDK (Software Development Kit) includes a large set of development tools such as a debugger, libraries, handset emulator, examples of code, documentation and tutorials. The official integrated development environment (IDE) for android development is Android Studio, which is made by Google and powered by IntelliJ, however developers may use other IDE's such as NetBeans and Eclipse with the help of a plugin. Text Editors can be used with the help of command line tools such as JDK for compiling and debugging code.

Android SDK Platform Tools is a separate part of the full SDK which includes command line tools such as adb and fastboot. Adb or Android Debug Bridge is a tool that allows a user to run commands on a connected Android device. Fastboot is a protocol within the SDK that is used mainly for flashing the android filesystem via a USB connection. The android device is required to be started in bootloader mode. Android NDK (Native Development Kit) is used for the compiling of C/C++ code to ARM or x86 native code. Android uses Bionic as its C Library and LLVM libc++ for its C++ Library. The NDK also includes other APIs such as OpenGL ES, Vulkan graphics APIs and OpenSL ES Audio.

Java and Kotlin are the main languages for Android App Development each have pros and cons. Java is easy to learn and understand but has the problem of requiring a lot of code. For instance, 50 lines of code in Java may only take 1-2 lines of code in Kotlin. With this decrease in code readability is improved and indirectly debugging. Kotlin can also use Java libraries and frameworks, making the transition from java easy. Kotlin does have a steep learning curve thanks to its highly concise syntax. Kotlin is now an official language (Along with Java) of Android Development and will eventually become the main language used by Android Developers. As Kotlin is still quite young, being only officially supported by Android in 2017, it lacks the level of documentation seen with Java.

Wikipedia contributors 2019, *Android software development*, viewed 18 May, 2019, <https://en.wikipedia.org/wiki/Android_software_development>.

Project Plan – Plans and Progress

Defining Roles:

As a group we found our ideal jobs listed in a previous assignment matched very closely to the role requirements for this project. These roles addressed both the needs of this project and that of future projects adding further features or completing real-world implementations of our application. We added additional duties to two members to address further staffing requirements. These members demonstrated experience or knowledge in the respected areas and were considered the best candidates for those positions.

We agreed a project manager is essential for the purpose of running meetings and serving as a leader in our team. This role was assigned to Joshua as he often stood up to chair meetings.

A compliance manager would implement policies and procedures to ensure compliance with relevant legislation and contractual agreements. This may include privacy legislation, financial legislation, consumer law, and licensing agreements. In future projects legal advice would be sought to address increased risk(s) relating to compliance failure, though we agreed this isn't required during the development of a prototype application. Dylan was assigned to take on the duties of our compliance manager due to having previous business-related education and experience, along with an understanding of current privacy legislation.

We decided on three software developers due to the expected workloads involved in developing an Android application. The developers would be responsible for creating the application, including implementing navigation software and constructing the user interface.

The software engineer and server manager would be responsible for constructing a mock delivery management system for use with our application. The software engineer would work with our software developers to assist in making changes to add the address push functionality to our application.

The security analyst would be responsible for working with software developers to ensure our Android application is secure and aligns with industry standards. This will include addressing any identified data protection or privacy concerns, avoiding unnecessary levels of access, etc. Though this is only a prototype, we agreed that privacy concerns arising in the early stages of development could be detrimental to our team's reputation. With this in mind, we agreed a dedicated security analyst was essential to avoid such a scenario.

Project Plan – Roles

Our roles were selected based on project requirements and previous experience/knowledge of our team members. Considerations were made to our 'ideal jobs' as selected in previous assignments.

Some consideration was made to the staffing requirements of potential future projects. The roles selected will all remain similar / the same in potential future projects. All the roles selected will still be required in any future projects relating to the application, especially once it's released to customers.

Arin - Security Analyst

The security analyst is responsible for identifying and addressing any privacy or security concerns in our application(s), server(s) or other software/hardware. A privacy breach at even the early stages of development could be detrimental for our team's reputation in the industry.

Dylan - Software Developer and Compliance Manager

A compliance manager is required to ensure compliance when any applicable legislation (ie privacy), licensing terms (i.e. navigation software licensing) and any other applicable contracts. Dylan will also assist with the software development workloads.

Jacob - Software Developer

The software developer(s) are responsible for the development of the application. Multiple software developers were assigned due to the workloads expected in developing an application. Software developers will work with all other teams as/when required to work towards team goal(s).

Joshua - Software Developer and Project Manager

A project manager is required to ensure a leadership structure. Joshua was chosen as he often takes a lead in meetings, assigning tasks and keeping things on track. The project manager is responsible for guiding the course of the project and assisting members as required.

Ryan - Systems Administrator / Server Manager

As our project will require a test delivery management system, a server manager/system admin will be required. This role will involve the initialisation and management of a demo delivery management server. The server manager will work with our software engineer to push addresses to our application.

Seth - Software Engineer

The software engineer will work with the systems administrator to modify our application to receive addresses from the demo delivery management system. The software engineer will use APIs to communicate with the delivery management system created by the systems administrator. Though this role could technically be done by developers, software engineers typically have more specialised experience in the area of APIs and modifying existing technology.

Project Plan – Scope and Limits

The objective of this project is to produce a prototype application called delivery assistant. The software will be provided as a Software as a Service (SaaS) product. The Delivery assistant application will fix an issue with delivery drivers wasting time manually inputting their delivery address into their mobile devices.

Our secondary objective is to make our roads safer by limiting the need for the driver to interact with their phone while driving.

We plan on delivering an application that can synchronize with our clients already existing customer database. The application will pull information from the client's database and sync the information to the delivery driver's mobile device. The navigation system will be provided by google maps. We will be altering Google maps to have no alerts or notifications while driving. The navigation will be a simple point A to point B. An Alert will be sent back to the store when the delivery has been completed.

Techpro's has a very tight timeframe to work on this project. This timeframe will put a limit on how many features we can develop into the application. At this stage we plan to have a prototype version developed and tested with limited features.

As a group we have a team of six highly skilled technicians, as discussed in the above roles section. We believe with six members working hard on this project we can meet our deadline for this project. Each member has a good understanding of the other members work thus we can share the workload among the team.

Our project scope will include brainstorming any technologies that we may need to use to complete our application. Making a project plan, listing any ideas or technologies from our brainstorm. Do research into the chosen navigation system, programming language and API tools. Once all Research has been completed, start developing and testing the prototype.

We will be excluding non-core relevant features from our prototype based on our timeframe limits. These features will exclude, a live application for real world companies, disabling touch screen functionality above 40 kph, live alerts to store regarding driver activity and an IOS version of our product.

Project Plan - Tools and Technologies

Team TechPro's will require various tools and technologies to aid in the development and testing of our application. Our team has briefly researched the requirements and chosen various pieces of software and hardware which would aid in our project. Our team has had experience with some of the technologies required, such as virtual private servers. All of our team has experience with Java, something which may prove beneficial for Android development.

Google Maps API - Google Maps API will be used as the navigation software for our application. Google Maps Platform can be accessed via an API key, with competitive pricing. We would need to setup an account with billing information to gain access, though predict our costs would be below the \$200 of free credit provided for development and testing purposes. None of our team has experience working with Google Maps API, though there's a great level of documentation available which should make things easy.

Android Development Kit – Androids Software Development kit is free and can run on windows, Mac and Linux devices. Our team will utilise this kit in the development process of our application. Android applications are typically coded in Java, something all our team members have at least some experience with. Licensing of the Android SDK is governed by the T&Cs (Android Developers 2019).

Various Android Phones, Software & Emulating– Android mobile devices differ heavily across brands, devices and even age. It's important we ensure our application is supported across most devices, making a strong effort to avoid potential compatibility issues. To make this possible, we will require various emulators and devices, allowing us to test across different versions of Androids and differing hardware. Without these technologies we may miss out on potential customers when our application is available, as we haven't made the necessary customisations for older hardware or Android versions. We will need to test screen compatibility across different pixel densities, sensor differences (i.e. different compass or GPS chips/etc) and differences across Android versions. Most of our team already uses Android phones, as such this process should be easy for us.

Server(s) – Cloud VPS would be beneficial for starting out as it is a lot cheaper than having your own dedicated servers. Being able to utilise cloud features would also mean hardware that fails on one of the servers while using a Cloud VPS would be able to continue working as data would be stored on multiple other servers. We plan to rent a VPS server opposed to purchasing or hosting our own hardware. Various of our members have experience working with virtual private servers and could assist in the setup process.

Android Developers 2019, Android Developers Terms and Conditions, Android Developers, viewed 19 May 2019. <<https://developer.android.com/studio/terms>>

Project Plan - Timeframe & Testing

Testing

The testing of the application will start off with the modularity by using GitHub flow, allowing for the ability to test code additions in different branches. This will allow for code testing without impacting the main master branch of the application. Testing of the interface will ensure correct functionality of the software. A prototype of the application will be developed where an address can be manually entered for early testing of the GPS/application software. Various Android devices of varying hardware and Android versions will be tested to make sure the application can work on a wide variety of Android devices. Later stages of testing will involve a test server used to send mock data such as delivery addresses to the prototype mobile application. This will allow us to ensure the automatic delivery address update feature works, without the need of a live delivery management system. Testing will conclude with real-world tests by third parties. Testers will be provided a generic address (i.e. local business/etc) and instructed to follow navigation directions. Any final changes required will be made at the conclusion of this testing.

The software testing will require users to test the mobile application in real world scenarios. As using prototype software could be considered high risk for a business. Getting a business to allow us to test the software in the real world may be a problem. This could mean a very small amount of businesses may allow us to test the software. Advertising the software to delivery-based businesses will be the way to get people to test the software. It would be ideal that the business allows testing on multiple vehicles for stress testing purposes.

Timeframe

Techpro's will have a timeframe of fifteen weeks to develop a delivery assistant prototype. We will be working on the planning stages in week one and two which will be made up of identifying problems and technologies needed. In week three our aim is to have a project plan draft drawn up and then use weeks four to six to do research on the technologies and problems from weeks one to three. Weeks seven to nine we will be striving to develop our backend code and have a first draft of our delivery assistant application frontend interface ready.

Weeks ten and eleven will be used to develop a test environment for the already developed application. The test environment will include a local server with junk client data to test the transferring of information from the server to the mobile devices. Once the test server is available Techpro's will spend week twelve doing in house bug testing and weeks thirteen and fourteen fixing any bugs that have been found. When we are satisfied that all known bugs have been fixed, week fifteen will be used for alpha testing with an external contractor.

Project Plan - Timeframe & Testing

Iteration #	Description	Timeframe
1	Planning stage <i>Identify problems and needs</i> <i>Define goals and objectives</i> <i>Develop strategies and project plan</i> <i>Identify required technologies</i>	Week 1 - 2
2	Project Plan Draft <i>A final draft of the project plan will be available, and the researching process will begin.</i>	Week 3
3	Researching - Google Maps <i>Research licensing requirements and ToS</i> <i>Open Street maps an alternative backup</i>	Week 4
4	Researching - Android development <i>Research useful studios, documentation, processes, optimized languages, platforms, limitations</i>	Week 5
6	Development - Back End <i>Development of the application back-end (prototype application without full user interface development)</i>	Week 7 - 8
7	Development - Front End <i>Development of user interface and other front-end features / etc</i>	Week 9
8	Development - Test Company + Address Sync <i>Adding address sync feature</i> <i>Setting up test server to push example addresses</i>	Week 10 - 11
9	Testing - In House <i>Internal testing of the application.</i>	Week 12
10	Bug Fixes <i>Fixing of any bugs identified during in house testing</i>	Week 13 - 14
11	Alpha testing + Final Fixes <i>Final round of testing involving contracted third-party testers. Any final bugs/issues addressed.</i>	Week 15

Diagram 4 - Timeframe Table

Project plan – Risks & Group Processes and Communication

Risks

Our group has identified various potential contingencies with our project. These risks range from potential licensing or contractual issues to lack of technical knowledge/experience with certain programs, applications and development techniques.

Android SDK is fundamental to our project's success. It's required for the development of Android applications. We have established that the development of Android applications requires a thorough understanding of Java. Our team all have a basic understanding of Java, though there may be difficulties involved in more advanced concepts. Java and Android SDK are both highly documented and such issues could be overcome through research, though this could prove a timely setback to our project.

Android is a very versatile operating system, installed on millions of phones, tablets and computing devices all with varying hardware. Our team may find difficulty ensuring our application works across a wide variety of hardware and Android versions. In such an event, we may narrow our project scope and focus on later Android versions with specific hardware requirements. This, for example, could involve creating a prototype which may not work on Intel powered mobiles.

We may have compatibility issues which create difficulties interfacing with our test delivery management server. Though our team has some experience with APIs, this experience is limited, and the process could prove difficult. As stands, we have addressed this by allocating an entire week to this portion of the project, ensuring we have enough time for any contingencies.

Our team had concerns around backing up code and revision control, these concerns are all addressed using GitHub. GitHub branches will allow for automatic revision control without the possibility of losing or ruining working code.

There are many unforecastable risks we may encounter through our group prototype, but if we try to minimise them, there will be less surprises along the way.

Group Processes and Communications

The group will communicate via discord, using both voice and text communication. As a group we will have structured voice meetings once a week, discussing what needs to be done that week and assigning tasks. Members may communicate directly anytime throughout the week, though all members are expected at scheduled meetings.

We will make use of Google Docs to allow for easier collaboration on documentation preparation. This also allows all members to see what others are currently working on, ensuring everyone stays informed on what's completed and what's yet to be done.

At later stages of the project we will make use of GitHub for hosting and sharing code developments. GitHub will allow all members to simultaneously work on code using revision control. GitHub branches will allow for code additions to be tested without potentially impacting the latest main version of the application.

Project Plan - Skills and Jobs

When deciding on staffing requirements to further develop our project, considerations were made to the knowledge and skill(s) of our existing team members. We decided on four roles.

Currently no team members have relevant design or graphics experience, as such we believe a graphics designer is essential to further our project. A graphics designer would assist in both development of advertisements, diagrams and presentations, and UI/UX development.

A technical sales / marketing manager would be beneficial in getting our application into the industry. Their role would involve liaising with industry contacts to establish the wants and needs of the industry, reporting to the project manager. This information would be vital in customising our application to meet wants and needs of potential customers and the industry. The technical sales manager would also be responsible for negotiating sales with industry contacts, including conducting presentations on our application(s) to relevant individuals.

A customer service / help desk staff member would be required to deal with increased level of help requests that would come with a higher number of sales. As our application grows, our software developers would be focusing more heavily on fixing bugs and adding requested features. Responding to support requests would take away from this, resulting in unnecessary delays. The customer service representative would assist customers with basic enquiries, directing requests as applicable.

A business analyst would assist in converting our project idea into a successful business endeavour. The business analyst will have project management experience and take lead of projects in a professional manner, with Joshua stepping down to the role of tech lead. The business analyst would provide specialist advice on the design of systems from a business perspective.

Graphics Designer

Techpro's requires a Graphics Designer to work on company logos, application UI/UX design and general media needs. The graphics designer will assist the marketing manager in creating slideshows used in presentations to potential customers or investors.

Experience / Knowledge:

- *2+ years' Experience in graphics design*
- *Tertiary qualifications desirable (not essential)*
- *Superior design and conceptual skills*
- *Knowledge of Adobe Creative Suite*
- *Strong attention to detail*
- *Strong communication skills*
- *Ability to work well as a team*

Position Duties:

- *Responsible for the design of the application*
- *Ensuring the design created is user friendly*
- *Collaborate with developers and other team members to ensure requirements met*
- *Develop advertisements for Delivery Assistant*
- *Develop presentations for potential clients/investors*

Project Plan - Skills and Jobs

Technical Sales / Marketing Manager

TechPro's will be looking for a person to be able to meet with potential and existing clients with a view of generating sales and maintaining relationships. This person will also have experience in marketing, with an ability to develop and execute marketing strategies.

Experience / Knowledge:

- *Previous technical sales or marketing experience (or relevant qualification) is essential*
- *Previous experience working in teams*
- *Communication skills*
- *Highly motivated*
- *Graphics design experience is desired though not required*

Position Duties:

- *Create and execute marketing plans to promote TechPro's application(s)*
- *Work with external agencies (graphics designer, UI/UX designer, software developers)*
- *Increase social media presence and build engagement*
- *Track and analyse website traffic and growth*
- *Liaise with clients to receive and document specific needs / feedback / requests*

Customer Service / Help Desk

The customer service / help desk representative would be the first line respondent to help requests, assisting customers in resolving common issues and providing technical assistance. The employee would document and submit information to the tech lead, technical sales manager or project manager where applicable. This would include in-depth documentation of bug reports and feedback, feature requests or any other relevant feedback.

Experience / Knowledge:

- *Previous help desk / customer service experience*
- *Technical / computing skills desired though not essential*
- *Problem solving abilities*
- *Strong communication skills*
- *Documentation skills*
- *Passionate about helping customers*
- *Positive attitude*
- *Excellent organisational skills*

Position Duties:

- *Delivery of exceptional customer service*
- *Providing support over telephone and/or via email*
- *Leave customers with a positive experience*
- *Answer customer queries in a timely manner*
- *Escalate and direct complaints, concerns and/or feedback as applicable*

Project Plan - Skills and Jobs

Business Analyst

The business analyst is responsible in forwarding our project idea into a successful business endeavour. The employee will act as a project manager, create procedures and policies and create strategic business goals and aims. The analyst works with all employees and stakeholders, assisting in the implementation of new systems, tracking project progression and providing support and guidance as required.

Experience / Knowledge:

- *5+ years' experience as an IT business analyst*
- *Ability to identify opportunities for improvements and efficiencies*
- *Understanding of software development and the IT industry*
- *Passionate about problem solving*
- *Exceptional attention to detail*
- *Strong documentation skills*
- *Great communication, leadership and teamwork skills*

Position Duties:

- *Provide support and guidance to appropriate project team members as applicable*
- *Ensure technologies are used efficiently, profitably and securely*
- *Evaluating, developing, planning and implementing new systems and infrastructure*
- *Develop and implement a detailed project management plan to track progress*
- *Build quality assurance and data protection processes*
- *Use stakeholder feedback to perform necessary improvements and adjustments to technology*
- *Assist in creation of application user documentation, manuals and procedures*
- *Maintain strong relationship with developers and stakeholders*

There are many other roles we would like to have part of our team to help us develop the best possible product. We determined most of these roles wouldn't be required full time and contracting may be more beneficial. For example, a legal advisor may be contracted to construct contractual agreements, sign off on policies ensuring legislative compliance, etc.

***Tool smith** - Someone to develop any custom software tools that might be needed when further adding features to the application*

***Database Administrator** - During further development of features we may need to customize or add new tables to our clients database*

***Tester** - During the final stages of developing our application, contracting a tester or testers would be beneficial to iron out any remaining bugs.*

***Release Manager** - Someone with experience releasing applications would help us have the smoothest release possible.*

***Network Engineer** - Having a network engineer would be important to maintain the network systems that Ryan our Systems Engineer has put in place.*

***Legal Advisor** - Will need a legal advisor to negotiate contracts, ensure compliance, provide counsel and more.*

Tools (GitHub / Google Docs)

How well does the audit trail on the git repository reflect the groups work?

Our team has made limited use of GitHub in the early stages of this project. GitHub will be utilised later in the assignment for revision control when developing the application. At current times, the GitHub audit trail only shows moving a copy of the old website over, along with minor changes where required. This site forms only a small part of our project and as such the audit trail isn't reflective of our groups work. We believe later in the project when Git is used for code development purposes the audit trail will more directly reflect our groups work and contributions.

How well does the audit trail on the git repository reflect the groups work?

Google Docs was used heavily for collaboration on documents, presentations and images. Our team made a total of over 200 comments, with Josh and Dylan making the majority of these. Comments allowed our team to highlight sections of text and make direct comments on it, instructing others on what needs to be completed or changed. The amount of comments isn't directly reflective of contributions, though we felt was a good statistic to analyse. We believe this demonstrates the leadership position Josh and Dylan took in this assignment, assisting everyone with suggestions and assigning tasks. There was over 25 revisions and 300 individual edits of our Assignment 3 document, each with changes varying in size. It's unclear which team member(s) performed edits as many were performed while logged out and appear as *all anonymous users* in version control.

Our Git page is viewable at:

<https://joshcoppen.github.io/Assignment-3-TechPros-IT-Project/>

Our Git Repository is viewable at:

<https://github.com/JoshCoppen/Assignment-3-TechPros-IT-Project>

Group Reflection

Group Feedback

What went well?

As a group we went well in the execution of tasks thanks to weekly voice meetings on Discord. During these meetings we were able to assign tasks, check/report progress on certain parts of the Group Project. Thanks to the use of Google Docs we were able to quickly upload word documents, edit the content, then add the content to the main document quickly. With live editing of the document members were able to communicate issues with parts of the assignment then ask another member to quickly check/edit the document.

What could be improved?

The weekly Discord voice meetings had the problem of not having all members present at the same time, which meant certain members could miss important information about parts of the assignment. This also affected checking of progress on assigned tasks which led to delays in uploading and editing of certain sections of the assignment. However, most of this was fixed by the fact of someone keeping notes and asking about progress through text chat. Although the missed meetings were not a major problem for the group, the team could have benefited by having all members in attendance.

What was surprising?

The amount of work that was in the assignment was very surprising. Google Docs ability to edit documents live was extremely helpful for members of the team quickly adding/editing/proofreading paragraphs for the assignment. It was also surprising how fast the time went after we started getting into the assignment, it felt like we flew through the time. It was surprising how fast the group put together a presentation slideshow for the video once a couple of members got together. The slideshow only took a couple of hours once the team started it.

What has the group learnt about groups?

The group has learnt that communication is key for proper completion of projects. Making sure everybody knows what they are doing by assigning tasks and making sure they understand the goals of the group, helps everyone know what they are doing. Without proper communication members of a team may get confused and frustrated about who is doing what task, which leads to people not properly contributing to the project/goal. Collaborative Technologies such as Google Docs and GitHub are extremely helpful for teams, especially if the teams work online. Collaborative tools allow teams to understand easily what other people are doing over long distances. Especially with Google Docs which has real time online editing of files.

Group Reflection

Arin's Feedback

What went well?

The use of google documents has been great for this assignment. You can add comments to any section of text within document. I feel this is better than using GitHub and uploading constantly as it can be difficult to add comments within their UI at times and therefore have to user other programs to let others know.

What could be improved?

Compared to the last assignment it did feel like there was a lot more going on in this one. Sometimes some of us would have busy days and can't make it to some of the chat session. Keeping group notes for people that couldn't make meetings seems extremely important so we can keep everyone on track.

What was surprising?

The amount of work we had really surprised me. At first glance it doesn't seem like a lot and looked very familiar to what we were doing in our last assignment. However, at lot of these thoughts changed when we had to do a presentation. submit an outline/ script for it as well and it all did start to feel a bit overwhelming.

What has Arin learnt about groups?

I've become more at ease with knowing that everyone can help each other out no matter where they live. Communication however is crucial as we sometimes only get a certain amount of time to talk to each other and writing notes is a must for any group. I have also enjoyed my time with this group and wouldn't mind working with them for future projects.

Dylan's Feedback

What went well?

In this project we made more heavy use of Google Docs for collaboration on documents and presentations. This change was very beneficial over GitHub as it allowed multiple individuals to work on the same documents at once.

What could be improved?

I believe in the future we could have more formal and structured minute notes to ensure members unable to attend meeting(s) are kept informed. Sometimes things come up last minute and people are unable to attend, keeping those people informed is vital. We had a few occasions for various people for varying reasons were unable to attend, such meeting minutes would have been very beneficial in those situations.

What was surprising?

I was surprised how quickly our team pulled together a presentation once a few of us got together in a voice chat and worked on it. In only a few hours we went from having a generic template to a completed draft slideshow.

What has Dylan learnt about groups?

I've learnt that groups typically work best where all members have shared goal(s) and understand what their individual role(s) are. I learnt that collaborative technologies (i.e. Google Docs/GitHub/etc) are extremely beneficial to teams, especially when working online.

Group Reflection

Jacob's Feedback

What went well?

This project went way differently than the last group project. This is because we used a lot of Google Docs instead of GitHub. Google Docs was really good in the way that it allowed real time group editing of a document which improved our ability to quickly fix errors and add extra information without weird version control of the document. As for group communication we used Discord Again which helped us assign tasks and complete them.

What could be improved?

I think one of the main issues we had was the Discord meetings. Getting everybody on at the same time proved difficult at some points. This could have been improved by making one night every week our 'meeting' night. In which everybody had to make sure this date was free.

What was surprising?

To me what was surprising was how easy Google Docs makes group work. With real time editing with other people online is really cool. Thinking about it was also surprising how fast the time went with the group assignment, now being at the end of the assignment and course.

What has Jacob learnt about groups?

I have learnt how hard it is for a group to communicate and complete a project online. Without proper communication the entire project could fail. Meaning that teamwork is key when working in groups. Assignment of tasks is important to make sure that all members are contributing to the entire goal of the group.

Josh's Feedback

What went well?

To me it felt like we tackled this project in a completely different manner to the last one. The biggest change we made was instead of all uploading our different sections as word documents to our GitHub repository, we used Google Docs. Using google docs made it simpler and easier to work on the same task together at the same time. Discord once again has been a great application for communication.

What could be improved?

Although we had weekly group chats, where all group members were obliged to attend, there were multiple occasions where one or more members was unable to make the chat. This isn't having a dig at any of my fellow members as one of those weeks it was me, and it was a critical chat that I missed. If we could have made a certain night every week that we all had to keep free, I believe would have made communication easier.

What was surprising?

Many of the previous issues I thought was surprising last time around had now already been negated. How quickly time goes by remains surprising. Before you know it, you are a week away from deadline, and plenty of work has been done, but plenty remains.

What has Josh learnt about groups?

After working with a group for around eight weeks makes one realise how quickly you can become comfortable with people who not long ago, were complete strangers. I have enjoyed working with this team and hope to work with them in the future.

Group Reflection

Ryan's Feedback

What went well?

Working with the same group as assignment two meant that we have a better understanding of each team members strengths and weaknesses. The lessons learnt from the last assignment about which tools to use made the work load a lot easier. Last time we used GitHub to edit documents with a late change to google docs. This time around we started out, with google docs saving us time.

What could be improved?

I believe we could have assigned tasks to each group member more efficiently. This would have helped us with our time management and to keep on top of the workload. We found that a lot of work still needed to be done close to the submission date.

What was surprising?

I found it surprising how easy it is to edit documents and create a presentation using google docs. Google docs allowed us to edit the same presentation while at the same time we were using voice chat over discord. We used a feature of Google Docs called comments so we could leave feedback on members completed sections.

I found time management got away with us, as there was still a lot of work that hadn't been done. It felt like we had lots of time at the beginning of the assignment, but after a short period of time it was almost time to submit.

What has Ryan learnt about groups?

I have learnt that when working in groups, every team member needs to know their task. Each team member needs to be clearly told what task they are doing and when it is expected to be completed. If tasks are not assigned member get confused, who is doing which task and become frustrated.

Seth's Feedback

What went well?

This project felt easier with the use of Google Docs and everyone being able to edit the same project simultaneously without accidentally removing someone else's work. We were also very well coordinated and the use of the comment button on Google Docs helped us see what needed to be done.

What could be improved?

I feel like having more Discord meetings each week could have helped us have a better understanding of what needed to be done. Some weeks only a few would be able to attend so having a second one could have been beneficial to those that couldn't attend the first and we could cover more work in that second meeting.

What was surprising?

It surprised me how well our group worked together in getting this assignment done. It also surprised me how useful the comment function was in keeping us organised while having no direct communication with each other.

What has Seth learnt about groups?

I have learned that groups can work really well together and really get a lot of work done in a short amount of time. I have also learned that sticking with the same group also helps as you gain an understanding of how the others work and how they are as people which can make you feel more comfortable.