Overview

Topic

The project we plan to develop will be a Productivity Software application with the goal to present a niche solution to the growing productivity software market (currently worth AUD $66.551 billion) expected to grow 14% per year up to 2030 (Productivity Management Software Market Report, 2030, 2022). To achieve this goal, our project will be developed with the basis of being versatile, cheap and easy-to-use. Our approach to development will avoid bloated and unnecessary features to the application's main objective which is to assist a user with their productivity, time-management and organisation. Benefits of this approach include reduced resources required and greater efficiency to project development. For the end-user, the benefit they receive is a simple and intuitive product which can be quickly picked up and utilised to its full capabilities. Features that are implemented will feel familiar without the bells and whistles most applications tend to have. We avoid having these bell and whistle type features so as to not distract and cause detriment to the user's experience. The application will serve as an all-in-one package for user productivity on an individual and team level, avoiding the need for add-on extensions or usage of additional applications.

To begin the development of our project, we will first create a wireframe to map out the application's layout, user flow, information architecture and behaviours. From here, a prototype in the form of a full interactive mock-up will be developed, simulating the applications changes, behaviours, interactions, etc. It presents an opportunity for on-going testing, an understanding and feel for the UI/UX, plans for changes in the overall design/flow/behaviours, and raise issues requiring amendment. We will continuously refine the project so that the actual software development will have a clear goal to achieve. This approach does leave us limited with what we can test such as the storage of files, the implementation of APIs or the web app deployment process as these require code and specific tools to be tested.

Motivation

Our main motivation for this project comes from the potential to enter the rapidly growing productivity software market. It presents future opportunities to capitalise on the growing success of the market and thereby becoming successful ourselves. The popularity and success of such apps stems from how our society has such a focus on our work productivity. Where we are a student or person in the workforce, we are all working to achieve some form of output. With majority of our time spent working and focusing on said output, we often find ourselves neglecting to properly organise our own schedules. Therefore, the option of productivity apps is convenient and perhaps necessary, especially for individuals finding difficulty to organise by themselves.
The other motivation we have is based up upon our first-hand experiences with currently available productivity and time-management apps as students. Students have many aspects to manage such as time-management, note-taking, and project/assignment management. There are a variety of apps available dedicated to one of these aspects, but the main issue we identified are the lack of apps designed to fulfil all these requirements together. App choice variety is important and often enjoyed, but it does become tedious and time-consuming having to switch between apps just to use a specific function. Synchronising data across the apps is possible, but it is again tedious, time-consuming, and difficult for most users. Our motivations together aim to resolve these issues so the end-user can dedicate more time on their work, regardless if our end-user is a student or not.

Though it is not certain we can achieve our ambitious goals, the project will still prove to be a valuable learning experience for the team. It presents an attempt to mimic a real-world scenario where we can experiment with different aspects of project development and software development. This project will demonstrate to future employers our ability to identify a particular niche/requirement in a dominated market, how we plan to solve these requirements, and how well we utilise each team member's skill-set to cover each other's strengths and weaknesses. How we

react and adapt to obstacles impeding our progress, how we solve issues and how organised we are will be other aspects future employers will be looking out for.

Landscape

Based upon the “2022-2029 Global Productivity Apps Professional Market Research Report” synopsis, “Key players in the global Productivity Apps market…” included Evernote, Microsoft To Do, Headspace, TeamViewer, etc.

(2022-2029 GLOBAL PRODUCTIVITY APPS PROFESSIONAL MARKET RESEARCH REPORT, ANALYSIS FROM PERSPECTIVE OF SEGMENTATION (COMPETITOR LANDSCAPE, TYPE, APPLICATION, AND GEOGRAPHY), 2022)

Other similar products to our project also include Microsoft OneNote and Notion primarily for their note-taking features. From the few that have been identified, these products are not just competitors, but key players that dominate the market. It is extremely ambitious to believe that our project could compete directly with these products and companies. Therefore, our project will be capitalising on the one point of difference that these other applications lack. This niche refers to each apps lack of variety in their functions to assist with all aspects such as note-taking, productivity, project management and organisation. By presenting our project as an all-in-one solution, we fulfil a particular niche in the market. Our chances of success increase and we avoid directly competing with these existing products.

Detailed Description

Aims

Aim - Develop a productivity web application

We aim to not only develop an application, but one that targets a niche requirement within a heavily dominated productivity software market. It will be developed with the premise and concept that it will versatile, simple and easy-to-use. To do this, there are functions, features and goals set out for the development of this project. The ideal milestone we hope to achieve is deploying a prototype application on AWS with limited core functionality. However, due to time constraints and a lack of necessary skills/experience, this milestone most likely will not be reached. Therefore, the minimum milestone we plan to achieve is a prototype in the form of a interactive mock-up which simulates all or majority of the core features. By doing this, we can still conduct testing and review the effectiveness of the UI,UX and functionality of the application.

Goal #1 – Establish a functionality flowchart and wireframe

The functionality flowchart will describe how the application processes will flow and represent the relationships functions have. The wireframe's purpose is to provide an overview of the UI design, the related functionalities and the intended behaviours of user interaction. This is our first and foremost priority to achieve. Without achieving this goal, development is impossible or at the very least, difficult to achieve and executed poorly.

Goal #2 – Create a Login/Landing page

For our prototype, this page will be the first page that greets users when they open up our web application. It will offer options for the user to either sign in with an existing account, or sign-up with a new account. Both options will take the user to a different page. In either option, it will not perform any actual email/password authentication as that is a function that needs to be coded. It is an important page to create given that this is the first page a user sees. However, when compared to other goals and core functions, it is not as important and time constraints may require diversion of resources from this page. This is because it is not a page or function necessary to achieving our main aim or application's functionality. This sentiment would change if we reached the stages of coding and required the sign-in authentication to be running.

Goal #3 – Create a Notepad page

This page allows a user to add and edit notes to a page with additional options for text formatting and fonts. In addition, an auto-save will be in-built into the page so changes will be saved to a database without pressing a button. This is a core feature that allows the application to exist as a productivity application and offer versatility as planned. However, if there were issues with time constraints, we may develop this page to have basic features such as adding notes, editing notes and creating or deleting a page and saving a page. Additional features for text formatting and fonts are still crucial but technically are not required for basic functionality of this page.

Goal #4 – Create a Whiteboard page

This page will simulate an actual whiteboard where you can draw on it, add text and shapes to it. It is meant to serve as a page to quickly visualise ideas and create diagrams whether on an individual setting or in a team setting. Though not an in-built feature, a user can technically user a screenshot/screen snip function to save their ideas and diagrams. Unfortunately this may be a feature we won't be able to simulate in Proto.io, therefore this goal might only be achieved to the extent of adding text to a blank white canvas, or simple setting the UI elements. The justification for this is because this page is not a core feature to our application and won't impact on the application ability to exist as a productivity application.

Goal #5 – Create a Sticky Notes page

This page is a board of sticky notes that a user can continuously add or delete sticky notes to their liking. The inspiration of this feature primarily came from Trello and the old Windows Sticky Notes feature. The sticky notes are organised in a grid format instead of free placement system as this leaves room for disorganisation on the board. Sticky notes would also have an additional feature to set a background colour to allow for further organisation. This page is considered a core feature as sticky notes have served to be a great way for some individuals to take down notes and organise their ideas. Therefore this page would still be a priority to complete, however some leeway would still be made. The most important feature for this page is to add/delete sticky notes and organise them in a grid format. The remaining features serve to enhance the usefulness of this page but are not necessary.

Goal #6 – Create a Roadmaps page
This page is meant to be a collation of a project timeline and task to-do list which can be used set up for individual or team setting. It would display a timeline graph for either a project or assignment depending on the user's need. Users are able to add tasks, delete tasks, complete tasks and assign tasks (exclusive in a team setting). This is a core feature which we believe to be one of the most appealing features of the application. For this reason, we would want to simulate almost all of the functions of this page. However time constraints may not allow us to fully develop the project timeline feature in Proto.io as it is unclear if this can be simulated to its full extent.

Goal #7 – Create an Overview Page
This page is the first page that a user sees once they login. It will be a simple hub of buttons that house internal links to the different pages of the application. Each button will have its own unique logo to represent the page that they link to. The button designs will be simplistic so that users can quickly find and intuitively understand what each button represents. Whilst technically not a core feature of the application, resources will still be put towards the full completion of this page. This is because from a UI/UX perspective, it is an essential page that not only makes page navigation easy, but also ties into our idea of a simplistic and intuitive design. Naming sense, this page might be changed to a 'Dashboard' instead of 'Overview'

Goal #8 – Create a page for document uploading
This page is primarily a file management feature of the application. Like other file management apps or features, the user can have the ability to add, delete, rename, open of move documents,files and folders. All document and folders are auto saved into the file manager. This feature is not a core feature so therefore, not much focus will be made to complete this page to its full extent. Furthermore the back-end processes of saving to a database can't be completely simulated. This can only see full effectiveness once it has been coded and a robust cloud server has been set up for cloud storage.

Goal #9 – Create a Communication Channel page/tab
One of our core features yet potentially most difficult to prototype/simulate. We would not develop our own communication servers/channels but rather simulate the API integration for Microsoft Teams. This would prove to be a more cheaper and effective method to establish some form of communication means for the application. However, for the purpose of the interactive mockup, we have opted to create a button that links directly to Microsoft Teams. Though this feature is a core feature, the most we would be able to simulate is opening up Microsoft Teams within the app. Any further behaviours/interactions with this feature is technically done within Microsoft Teams, therefore there is no point simulating this behaviour.
Goal #10 – Allow for Direct Messaging capabilities (Extension of Goal #9)
This goal would still utilise the API integration for Microsoft Teams for direct messaging. It is not a priority to be developed in the prototype since the functionality depends on Microsoft Teams.

Goal #11 – Side Bar Menu
This feature is a side bar on all pages after you log in to the web application. It features a Home button for users to go back to the Overview page. A drop down can be expanded or collapsed which contains buttons to navigate to all the pages in the application. It will be designed to be simple and minimalist. This is a priority feature that we need to simulate its entire functionality. Without a side bar, a user does not have an effective means of navigating through the application.

Plans and Progress
The first week of the Assignment, spanning from those dates 02/05/22 – 08/05/22, the team has unanimously agreed upon taking this period slow, treating it as a time where members that pushed themselves could rest and be recharged for the follow 3 weeks. That is not to say that no work will be completed during the first week, but rather, we will analyze and delegate the sections evenly regarding the written components; disregarding the prototyping and wireframing which we intend to complete during the latter half of the Assignment 3 duration. A lot of members pushed themselves in contributing more than their fair share during Assignment 2 and in conjunction with the hasty nature that was completing the Assignment on the night of the submission date, we have split the work for the first week of Assignment 3 accordingly:

- Bo: Skills and Jobs (Marketer)
- Hayden: Skills and Jobs (Software Developer)
- Van: Skills and Jobs (UX/UI Designer)
- Daniel: Skills and Jobs (Project Manager)
- Russell: Team Profile Paragraphs

In addition to these tasks, with expectations to be completed during the end of the sprint week, the bigger sections of the written component must also be addressed and delegated during this first week, especially for parts of the Assignment that need to be done frequently such as Plans and Progress. The following work will be considered an ongoing task, and assigned to the members respectively:

- Bo: Timeframe, Tools
- Hayden:
- Van: Plans and Progress, Meetings and Agenda
- Daniel: Report Design
- Russell: Website Skeleton

Additionally, every member was reminded that the Spark Feedback pertaining to Assignment 2 needed to be done before the deadline. Unfortunately, some members, because of inaction, which will be a constant theme throughout this documentation, did not complete their feedback in time.

Throughout the week, members such as Daniel and Hayden, that did not contribute evenly during the last Assignment, showed a resurgence in their enthusiasm with the act of completing their assigned tasks of their respective Skills and Jobs earlier than the deadline. Some members were eager to carry on additional tasks, however, per the timeline that we had decided on, the additional delegation of tasks was to be discussed further during on usual weekly meetings, where we discuss progress as well as tasks to be assigned during the following week. This was a good indication that everyone was willing to try and contribute as best as they can.

The progress completed during this week was not without roadblocks and troubles. Trying to have an upkeep of 2 meetings per week, a meeting was scheduled on Friday, but due to the lack of members present, it was rescheduled to Saturday. Despite this, only 3 members showed up to the meeting: Bo, Russell, and Van where the section of Project Overview was assigned to Bo and Group processes and communication to Van. This selection of tasks was because the team believed the sections of Aims, scopes and limits, and other sections relating to the actual Project itself, would be in better handled by Hayden. No new task was assigned to Russell as creating the website skeleton

by himself was already quite a demanding task and asking him to take on any more responsibilities would be unjust to him. At the end of the week, the reason for unattendance from Hayden and Daniel was ascertained with Hayden having no internet throughout the second half of the week and Daniel notifying that Saturday was a bad day to hold meetings for him, resulting from work.

In retrospect, comparing the beginning of the week's planned tasks with what has been done at the end of the week, the team has completed most, if not all our weekly tasks, excluding those that are continuous. Resultingly, the smooth progression provided the team with a moral boost and confidence that during this Assignment, we would be able to better execute on our individual parts, and additionally, collectively work better as a team.

The second week of the Assignment spanned a timeline between 09/05/22 – 15/05/22. Given that the team had rested and taken a slow start to the Assignment, the start of this week was a sign that we were going to ramp up the responsibilities given to each member. However, considering that Hayden contracted COVID, we were down to 4 expendable members during this weekly sprint. In consideration of this roadblock, it was agreed during the tutorial that the team would try and finish the rest of the written components for the assignment during this week, for the purpose of focusing on the wireframing and prototyping during the latter half of the Assignment 3 period. Therefore, the spread of new tasks can be examined by the following:

- Bo: Career Plan, Overview
- Hayden: Tools and Tech, Aims, Career Plans
- Van: Career Plan, Group processes and communication, Scopes and Limits
- Daniel: Career Plan, Testing, Risks
- Russell: Career Plans, Roles

During this period, the following Tasks were completed:

- Bo: Career Plan, Overview
- Van: Career Plan, Group processes and communication, Scopes and Limits
- Daniel: Career Plan, Testing

Considering that most tasks were completed by the members available, there are many positive things to bring up during the reflection of tasks completed during week 2. Completing accordingly to our conceived plans, we were able to complete most of the written components, leaving the more specialized parts for Hayden to write up when he was made available again. Russell was also able to push onto Github the components of his website, although being in a different in structure, with no index.html for an actual Github page to work, it was relieving to find out that the expendable members were able to keep up and deliver on their tasks.

The main areas of concern pertaining to roadblocks during this week, was not only having to operate at a limited capacity, but also the fact that meetings were not being attended, with Russell joining Hayden in absence. At this stage of the Assignment, it would mark the beginning when these two members went missing from the Assignment, whereby responses to Bo's text messages were not coming through. This theme of non-attendance and not communicating with the team took root during the week and became extremely prevalent in the following week 3.

In retrospect, the tasks assigned during the sprint were able to be completed by the respective members overseeing them. As mentioned before, the roadblocks of non-communication and non-attendance would cause the team some trouble, with outstanding tasks being kept in the backlog, incomplete, as well as cause trouble for assigning new tasks and consequently, some members

would have to take on more than their fair share of the workload and complete the outstanding tasks that were assigned to the inactive members.

Week 3 spanned from the 16/05/22 – 22/05/22. During this week, only Bo, Daniel and Van attended the tutorials, where the discussion of the next steps and tasks relating to this week were held. In accordance with the Sunday meeting, the team agreed to focus on the fundamentals of operating new tools such as Figma and Proto.io, and familiarize themselves with them until the next meeting, which would be held on the Friday of that week. Considering the absence of our other members, we decided to defer the discussion of the actual prototyping and artifact discussion until the next meet. Therefore, the assignment of new tasks would also be delayed until the meeting.
During this week, a meeting was scheduled for Friday, in which was discussed and agreed on a consensus from the 3 members who attended. Unfortunately, only 2 members showed up; Bo and Van. Resultingly and reluctantly, the meeting would have to be rescheduled again until the Sunday, as it was the usual time in which we have conducted ourselves, and that time worked out well for every member of the group.
Although the meeting was to be held on Sunday, in lieu of lack of attendance, Bo and Van discussed the procedure relating to the prototyping. More specifically, Bo had drawn up a flow chart diagram to exemplify how the interaction between features and elements were to be implemented. At this stage, the bulk of the flow chart had been completed, with a few tweaks to be added during the following days.
Using this as a reference, Van implemented the basic functionalities of the Sticky Notes, Login-in screens, Notebook and Calendar during this week in which the complexity of prototyping these features was made simple with the help of Bo's flow chart.
During the rescheduled meeting on Sunday, only Van and Bo were present yet again. Regardless, the discussion of the work done, as well as the focus for next week, pertaining to the tasks that each member would be assigned to was also discussed during this meeting. Bo had discuss wanting to focus on preparing for the presentation, with the task of prototyping to be continued by Van. As the inactive nature of the other members, it became difficult to access the progress of work done on their behalf, and even more difficult in assigned tasks to them as they already have a handful of outstanding tasks that had yet to be completed.
Another setback that was discovered during this week was the fact that proto.io does not support that use of condition statements and logics. This further revised the scopes and limits of what we would be able to implement using this tool. Taking this into account, we would aim to prototype the functions as close to their expected behaviors as possible and if it becomes too difficult to implement with proto.io, the basic flow and transitions between screens would be opted for these sections.

Overall, the beginning of the week started off slow, being down 2 members and having the actual discussion of the implementation of our project deferred to the latter half of the week, proved to be quite mistake to some extent. The majority of week 3's plan was intended to relate to prototyping and creating a mockup of our project. During this week, certain members had to step up and do more than their fair share of the work.

Week 4 spanned the period from 23/05/22 – 29/05/22. The remaining written components that were originally assigned to Hayden were absorbed and completed by Bo. The preparation of the Video Assignment after the completion of Assignment 3 will also be discussed in this period, in which the discussion how to conduct, as well as discussing compromises in terms of members able to participate. This week is also dedicated to try and complete the remaining artifacts for the prototyping. Additionally, a functioning website with all the content uploaded will also need to be complete during this period. The report collation will also be done along with the rest of these tasks.

Since Hayden has still yet to contact the group regarding his wellbeing and progress, Bo has taken additional responsibilities in taking over his outstanding tasks, which were the Aims and Tools and Tech section of the report.
The prototyping has been travelling smoothly, yet some functions cannot be completely envisioned in this sprint. This is a result of a combination of time restraint, limited knowledge regarding the use of proto.io and its conventions, and the biproduct of having a single member perform most of the prototyping despite it being a two-man designated task. From the last weeks progress up until a couple of days before the deadline, Daniel has yet to show significant progress on his chosen feature to implement being “Roadmaps”. Consequently, the rest of the functional requirements for the prototype (which can be viewed from a diagram perspective here: https://www.figma.com/file/JDsQbrtl37ZdcLE57IzGgI/Block-Market-Flow-Chart?node-id=0%3A1) has been completed by Van.

Week 4 Overall:
Ultimately, the team fell short of what we had intended to carry out during this week because of non-attendance, non-communication, and unfinished outstanding work. Certain members had to step up, resulting in a reasonable reflection of our intended plans for the week, albeit the bulk of this work was done during the end of the week. Therefore, the end of the Assignment will yet again be ensued by another crunch period, where we aim to complete the remaining requirements per the Assignment specifications including the remaining function of roadmaps for the prototyping, the uploading of our content on the website, and final report collation.

Roles

Scopes and Limits

Firstly, it is important that we recognise the restraints that we have, coming into this Assignment. The development of a productivity App would require significant resources in developing the front-end and back-end of the software. The creation of a fully functioning front-end would require resources in coding HTML/CSS, which the team commands a basic understanding of the fundamentals. The use of JavaScript would also be important, considering that our project idea is a web-based application, requiring the use of JavaScript Libraries such as React. Considering that only 1 member has experience using this Library, in conjunction with the time restraint of less than 1 month to work on this Assignment, the team has unanimously agreed to opted out of producing a web-based application, favouring the option of produce a functioning prototype in lieu of this decision.

Additionally, only 1 member is well acquainted in PHP for the back end, proceeding with creating a functional web-application would require a significant amount of work from the two members, rendering in an unfair amount of work distributed.

Designating members as front-end and back-end developers would require them to learn specific languages, libraries, and how to use them to a degree where they can contribute evenly to the development of the project would be unfeasible, as the learning curve far outweighs the time we have.

From out Assignment 2 Project Idea Description, we intend to prototype and develop a mock-up regarding the main features which include:

- Notepad
- Whiteboard
- Calendar
- Sticky Notes
- Roadmaps
- Overview Pages
- Document Uploading
- Channels
- Direct Messaging

We will be ignoring the extra features in this iteration of the development. The omitted items include:

- In-built Media Player
- In-built contact book

Furthermore, the use of Proto.io as our application to implement our features also needs to be addressed. Proto.io is very useful for simple prototyping projects. Where it comes short, is when complex arguments and conditional statements are needed to be implemented. As this prototyping tool does not support the use of if statements and simple logic, it is extremely difficult for use to be able to fully envision our project with Proto.io. However, due to the time constraints, amalgamated with our limited abilities in software development, we have further revised our expectations in the scopes and limits. Functionally, we are still aiming to provide the same experience as if one were using a fully developed application. For instance, for the Sticky Notes function, we intend to provide the user with the experience to type on a note, create a new note and delete an existing note. However, because of not being able to utilize conditional logic, we will cut some of our functions short, in terms of what we can present as well as make compromises regarding the ways we implement some features.

Tools and Technologies

The main tools utilised for the initial development of the project will be Figma and Proto.io. Figma will be primarily used for creating the diagram process flowchart, whilst Proto.io will be used to create the wireframe and interactive mock-up of the project. Due to the small time constraint that is set to create these plans, a premium plan/license for Figma and Proto.io will not be required. We can use the free trial period to utilise the premium features as much as possible. We plan to use Russell's prior experience with these tools to guiding and teach the team on how to use them. Once we reach the coding stage after testing and reviewing the UI/UX of the interactive prototype, we plan to use HMTL, Javascript and PHP. Hayden and Russell have extensive experience with these languages and therefore would be most suitable with the responsibility of the actual software development. Visual Studio Code and Atom will be our primary choices of IDE, with GitHub being version control tool. In terms of hardware, it would only extend as far as having a standard laptop or a PC fast enough to handle coding very efficiently.

Testing

Throughout the stages of development of our application, we will test our program in many ways. The most common way we will test is after each revision is the functionality test, where someone working on it will go through to make sure there are no unexpected errors when performing tasks, and that all the buttons and links directly to where they're meant to go, if everything works as it's designed to, then it's succeeded. After that there is usability testing, this will take place through user testing, users will be found through connections staff members have with family and/or friends who would use a productivity app, only about 5 people would be needed each time a usability test is done. This type of testing involves making sure that everything more or less feels right to people who aren't exposed to the software already and ensuring that all of Nielsen's ten usability heuristics are being followed appropriately, such as user control, consistency, recognition, and aesthetics. This stage of testing will be completed successfully once each user tester can successfully use the prototypes of the program within the tasks instructed without outside help.

Interface testing will also be done to ensure that the program connects to the servers correctly and that any error messages displayed are the correct ones, also that the database can be read correctly and work as designed. Also important is compatibility testing, where the app will be tested on all device types, such as Windows, Mac, Linux, Chrome, Firefox, Android, iOS, etc., to ensure that all tasks can be achieved on each system without error. Lastly is performance and security testing, the former involves putting the software to its limits by overloading it with information, this is to make sure that the limits of the software are well within the potential use cases of what users would do; and if it does crash, to make sure no data is lost. Security testing on the other hand is making sure all data on the servers, such as documents, email addresses, phone numbers, passwords, and payment details, are all securely kept, encrypted, and inaccessible by the end-user.

Time Frame

	Van	Bo	Hayden	Daniel	Russell
Week 1	Design, collate, and audit Assignment 2 report. Fill in missing details of the report where required. Assist to add report information to Block Market's Assignment 2 website.	Collate Assignment 2 MS Info document. Complete overview, motivation, skills and outcome of Project Ideas. Assist to add report information to Block Market's Assignment 2 website.	Complete description and tools/tech of Project Ideas. Amend website presentation issues.	Assist with report collation and add report information to Block Market's Assignment 2 website	Develop and design Block Market's Assignment 2 website. Assist to add report information to Block Market's Assignment 2 website. Amend website presentation issues.
Week 2	Research job ads for UI/UX Designers and create our own job ad for this position. Start writing upon Plans & Progress.	Research job ads for Advertising Specialists and create our own job ad for this position.	Research job ads for Software Engineers and create our own job ad for this position.	Research job ads for Project Managers and create our own job ad for this position.	Create Block Market's Assignment 3 website skeleton. Finish Team Profile (Group Processes and Career Plans).
Week 3	Write up paragraph on their own career plans. Write up Group Processes & Communication, Plans & Progress, and Scopes & Limits. Continue writing upon Plans & Progress.	Write up paragraph on their own career plans. Write up Overview. Continue updating Tool and Timeframe. Plan out A3 Presentation.	Write up paragraph on their own career plans. Write up Aims and Tools & Tech.	Write up paragraph on their own career plans. Write up Risks and Testing. Continue report collation.	Write up paragraph on their own career plans. Write up Roles. Continue on website skeleton.
Week 4	Start on wireframing/prototyping. Continue on Plans & Progress and Scopes & Limits.	Start on wireframing/prototyping. Finish up on Risks and Timeframe.	Finish writing up on Aims and Tools & Tech. Start on wireframing/prototyping.	Start on wireframing/prototyping. Continue report collation.	Finish up on Roles. Continue with website development
Week 5	Finalise/test project prototype. Write up group reflection for self. Collate final documents for assignment submission.	Finalise/test project prototype. Write up group reflection for self. Finish writing up on Tools. Write up Presentation plan/content.	Finalise/test project prototype. Write up group reflection for self.	Finalise/test project prototype. Write up group reflection for self. Finish up with report collation.	Finalise/test project prototype. Write up group reflection for self. Deploy assignment website. Start creating Presentation website.
Week 6	Do project presentation. Complete A3 contribution form/feedback. Review project prototype and user feedback.	Do project presentation. Complete A3 contribution form/feedback. Review project prototype and user feedback.	Do project presentation. Complete A3 contribution form/feedback. Review project prototype and user feedback.	Do project presentation. Complete A3 contribution form/feedback. Review project prototype and user feedback.	Do project presentation. Complete A3 contribution form/feedback. Add video to Presentation website.

	Van	Bo	Hayden	Daniel	Russell
Week 7	Conduct further market research, surveys and identify potential avenues to develop towards. Start networking with potential clients.	Start setting up the project's cloud server with AWS. Review cloud server plans and ensure within budget/resources.	Start coding up the project. Primary focus on back-end.	Plan and develop the entire app database design. Also submit plans on the deployment/storage of the database for budgeting planning.	Start coding up project. Primary focus on front-end.
Week 8	Write up report on findings based on market research, surveys etc. Goal is to identify and relay any additional wants/need from our target market and see if our application can deliver up on it. Continue networking with potential clients. Assist with on-going software testing.	Continue configuring AWS cloud server for app deployment. Review database deployment plans and decide on whether to use SQL or NoSQL. Monitor software development progress. Update stakeholders on progress.	Continue coding the project. Primary focus on back-end.	Research and review DBMS software options most useful for the project database. Finalise the database design. Assist with on-going software testing.	Continue coding the project. Primary focus on front-end.
Week 9	Continue networking with potential clients. Start creating promotion campaign. Assist with on-going software testing.	Finalise AWS cloud server configuration for app deployment. Update stakeholders on progress. Communicate additional requirements from stakeholders to team. Monitor software development and database progress.	Continue coding the project. Primary focus on back-end. Completion/near completion of front-end components.	Start database development in the chosen DBMS software. Assist with on-going software and database testing.	Continue coding the project. Primary focus on front-end. Completion/near completion of front-end components.
Week 10	Finalise promotion campaign. Continue networking with potential clients. Assist with on-going software and database testing.	Write up a project status report for stakeholders. Update stakeholders on progress. Communicate additional requirements from stakeholders to team. Monitor software development and database progress.	Finalise project development. Ensure completion of almost all components. Work with Daniel on implementing database with the application.	Finalise database development in chosen DBMS software. Work with Hayden/Rusell on implementing database with the application. Assist with on-going software and database testing.	Finalise project development. Ensure completion of almost all components. Work with Daniel on implementing database with the application.
Week 11	Deploy promotion campaign on social medias. Continue networking with potential clients and answering queries.	Present project status and progress report to stakeholders. Thorough testing of application. Review current budget usage and projected user	Thorough testing and debugging of application.	Thorough testing and debugging of application. Primary focus on the database.	Thorough testing and debugging of application.

	Van	Bo	Hayden	Daniel	Russell
		count.			
Week 12	Continue promotion campaign on social medias. Continue networking with potential clients and answering queries. Draft up user survey page.	Update stakeholders on progress and application deployment. Analyse projected data user count and budget usage.	Deploy application on AWS.	Deploy application on AWS.	Deploy application on AWS.
Week 13	Deploy user survey page. Continue promotion campaign on social medias. Continue networking with potential clients and answering queries. Review and analyse project performance and user feedback.	Update stakeholders on progress, user feedback and application success. Review and analyse project performance and user feedback. Monitor user usage, ensure cloud server usage still within budget.	Ongoing application support and debugging. Start coding up on additional features to add to application.	Ongoing application support and debugging. Start coding up on additional features to add to application. Expand upon database design/requirements where necessary.	Ongoing application support and debugging. Start coding up on additional features to add to application.
Week 14	Continue promotion campaign on social medias. Continue networking with potential clients and answering queries. Monitor user usage, ensure cloud server usage still within budget.	Update stakeholders on progress, user feedback and application success. Write up report/summary on project performance and user feedback. Monitor user usage, ensure cloud server usage still within budget.	Ongoing application support and debugging. Deploy additional features to application.	Ongoing application support and debugging. Deploy additional features/database extensions to application.	Ongoing application support and debugging. Deploy additional features to application.
Week 15	Continue networking with potential clients and answering queries. Monitor user usage, ensure cloud server usage still within budget.	Present report summary review on project performance, user feedback and application success to clients. Present KPI/performance review to each team members.	Ongoing application support and debugging.	Ongoing application support and debugging.	Ongoing application support and debugging.

Risks

Majority of the potential risks come from our usage of software and online services. With regards to our prototyping/mock up, we plan to use Proto.io for this purpose. From our experience with learning and understanding Proto.io, there is the issue in which it is limited in what type of interactions and actions can be simulated. Therefore, some components that have been planned out will not be simulated accurately or at all. This impacts our capability to quickly identify and resolve bugs/issues that originate from these unimplemented components. Such bugs/issues may only arise once we reach the software development stage and will hinder the project timeline.

Additionally we have to think about the usage of cloud services since we plan to deploy a web application. Regardless of which one we use, there still comes the risk that the application may not be deployed properly and further configuration will be needed, hindering the project timeline. Additionally, there comes the potential risk where we have more users than projected which is generally a positive consequence. However, it also opens the possibility that the cloud service plan may automatically upgrade to accommodate the additional users and become a paid service. There would of course be constraints set to avoid such a situation but the possibility is still there. Hackers attempting to obtain the cloud service API key is a major security risk we need to prevent and be vigilant against, lest we suffer major consequences that may lead to the immediate shutdown of the application or additional billing.

Later on in the project where we need to decide on a Database Management System (DBMS), we need to consider in particular if we use a SQL/NoSQL DBMS. SQL has benefits of an accurate and robust database, but it does come general risks such as being too expensive to maintain, constraints causing conflict with its interaction to the software. Such risks naturally will cause delays to project progress and in some cases, halt it entirely depending on the scenario. On the other hand with NoSQL, it will be more high performing and quicker for querying. However this also means we need to place extra emphasis on our software development and implement data constraints through our code rather than the DBMS. Additionally, due to the lack of emphasis on data integrity that NoSQL has, the chances of bugs to the DBMS and application greatly increase.

With our choice of programming languages, we may not identify limitations and shortcomings until well into the software development stage. Delays to project timeline may occur due to bugs, lack of ability to program certain functions, and attempting to create a workaround that may have otherwise been easier in a another programming language. We also have to be mindful in the capabilities to interact and deploy with our choice of cloud service and even DBMS.

Group processes and communications

We are continuing the format for meetings from Assignment 2, aiming to hold 2 meetings per week, on Thursday and Sunday. The purpose of the meetings is to check on progress set for the week from every member, discuss if they need any help on their end, as well as discuss the assignment of new tasks, especially for members that have completed all their assigned tasks. This is conducted as a formal meeting on Microsoft Teams, where we are utilizing new tools such as a task board, as a rectification of the problems pertaining to task tracking from Assignment 2.
Bo has also set up a Discord Server, intended to be used as an informal, alternative channel of communication as it is easier to reach everyone there, having the application on mobile phones. Additionally, the use of Microsoft Teams chat has also been established as a means of communication, as if a member wants to reach out to a specific individual, they do not have to resort to the Group Channel and can do so on their accord.
Additionally, the tutorials like the way we conduct a Teams Meeting, but in more length and detail. Task allocation, progress tracking and other issues relating to the assignment are discussed both as a team, as well as confiding in Anthony for problems needing clarification and guidance on tools and processes to employ. Therefore, to summarize, we intend to hold a face-to-face meeting, in the form of the tutorial, as well as 2 additional meetings.
In the absence of a member in the meeting, they will be caught up by either Bo or Van, and discuss the tasks allocated to them, as well as understanding why they were not able to attend. For the writing component of this assignment, the tasks were split, to be completed individually. However, the advent of wireframing and creating the mockup, the team expects the number of times we communicate to increase.

References

Grandviewresearch.com. 2022. Productivity Management Software Market Report, 2030. [online] Available at: <https://www.grandviewresearch.com/industry-analysis/productivity-management-software-market>

Market Reports World. 2022. *2022-2029 GLOBAL PRODUCTIVITY APPS PROFESSIONAL MARKET RESEARCH REPORT, ANALYSIS FROM PERSPECTIVE OF SEGMENTATION (COMPETITOR LANDSCAPE, TYPE, APPLICATION, AND GEOGRAPHY)*. [online] Available at: <https://www.marketreportsworld.com/2022-2029-global-productivity-apps-professional-market-20226339>