PROJECT IMPLEMENTATION PLAN

Get Ahead Mobile Application

Paige Clay – 2019012408
Bilee Billyard – 2023002556
Alia Haizat - 2022006035

Table Of Contents

Table of Figures

Project Overview

Project Description

The project that we will be producing is a time management app made with Android Studios called "Get Ahead". The development of this application is intended to ease the process of allocating time for different components of the project and addressing scheduling issues, particularly for students managing multiple assignments and coursework.

Designed as an adjustable Gantt chart, "Get Ahead" allows users to log the time spent on each stage of a project and make adjustments to the timeline as needed. This feature helps in accommodating unexpected delays and ensuring that the project stays on track. Additionally, users can share the updated Gantt chart with Project Managers and team members.

The project consists of creating:

- o Android App: Developed in Java for Android 10 using Android Studio as the IDE.
- o Local Database: Managed with SQL in Android Studio

The app includes features such as task creation and categorisation, notifications, progress tracking and calendar integration. The user interface is designed to be intuitive, ensuring ease of navigation and a smoother user experience.

Project Objective

Our main project objective is to ensure that the project members are better equipped to manage their time while developing a project. As mentioned above; to fulfil this objective we will develop a mobile Android application called Get Ahead. The choice of a mobile app is for convenience as most people have their mobile device on their person at all times, and Android has a market share of 64.5% in New Zealand (statcounter GlobalStats, 2024)

The application will function as an adjustable Gantt chart, where the original plan of the project and all of the stages can be adjusted as the project is executed, so that any complications during the project process can be logged and the schedule of the project adjusted as needed.

The user/project member is able to input the time it has taken them to do a stage of the project, or if they are needing to go over the scheduled time. When this occurs a new timeline will be created based off of what was input, and the other stages of the project will be pushed back to adjust the schedule. An option to share the Gantt chart will allow

the user to share this new schedule with the Project Manager and other project members.

The target users are students that have to do projects. This is because time management as a student can often be harder to coordinate due to all the assignments and coursework you need to do. This is just a starting point as we want to expand this to include users in the professional project workforce.

Project Importance

"Time is the scarcest resource and unless it is managed nothing else can be managed" (Drucker, 2001)

As was mentioned earlier, without proper time management the chances of project delays, team burnouts, increased cost and misuse of resources increase, which in majority of cases will have a negative effect on the overall output of the project. A well-structured time management system can mitigate these risks by ensuring that all tasks are laid out and carefully scheduled.

Gantt charts are visual depictions of timelines and schedules in a project and are an essential tool for the purpose of time management. They have proved to be compatible with most project development methodologies and are widely used in projects all over the world. A Gantt chart shows the project into broken into tasks, which each get

Figure 1 Example of Basic Gantt Chart (Product Plan, n.d.)

assigned start and end dates, as well as things like assignees, priority, and duration. By incorporating this into Get Ahead, we will provide the users with a flexible and adjustable scheduling tool which will allow them to manage their time more efficiently, as well as automatically adapt their timeline whenever they have a delay so they can stay on top of it, overall giving them better control over the outcome of the project.

Project Stakeholders

As "Get Ahead" is a time management application, the project stakeholders include a wide range of individuals and organisations who are interested in the app's development, implementation and use. These stakeholders represent a broad range of interests and influences, ensuring that the app is developed, tested and launched successfully while meeting the needs of its users.

Stakeholder Register

Project Manager	Alia Haizat	Project Phase	Planning Phase

ID	Role	Category	Interest	Influence	Expectations
1	Development Team	Internal	The technical success of the app (coding standards, user interface design and functionality)	Medium – High	Deliver a well-designed, fully functional and technically sound app.
2	Quality Assurance Team	Internal	Ensures the app is bug-free, user friendly and functions as intended	Medium	Identify and report any bugs, ensure a smooth and error-free user experience
3	End Users	External	Usability, effectiveness and features	Medium	Access an intuitive, effective app that helps end users manage their time efficiently
4	Educational Institutions	External	Tools that help students manage their time and succeed in their endeavours.	High	Receives a reliable tool that enhances student productivity and academic success.

Project Scope and Boundaries

The project scope is to develop a mobile application called Get Ahead, which is an Android app built for devices compatible with Android 10 operating system. The primary objective is to aid users, particularly students managing multiple assignments and

coursework, by providing an adjustable Gantt chart for effective project planning and time management. The application will feature task management capabilities, allowing users to create, categorise, and track tasks with associated start and end dates. It will generate a visual Gantt chart that dynamically updates based on user input, reflecting changed in project timelines and accommodating any delays. Additional features include notifications for upcoming deadlines, the device's calendar, progress tracking, and the ability to share the updated Gantt chart with Project Managers and team members.

The boundaries of the project are defined as follows:

- The application will be developed using Java and Android Studio, targeting Android 10 devices.
- Data will be managed locally using SQL database, without cloud-based synchronisation or external data storage.
- The user interface will be designed to be intuitive and user-friendly, but advanced features such as voice commands or AI-based scheduling recommendations will not be included. Sharing functionality will be limited to exporting the Gantt chart as a file or using built-in Android sharing mechanisms, without integration with external project management platforms.
- There is going to be no multi-user collaboration so team member can't work together in real time.
- Users that receive the Gantt Chart from another user are unable to load it into the app.

Functional Requirements

- Users can create a task for a project, with details such as title, description, start and end dates, priority, and status
- Users can do CRUD(create, read, update and delete) on the tasks
- Users can have multiple projects on the go at once
- Login functionality with login, registration and forgot password
- Users can change the end dates, and the following tasks will automatically adjust their start and end dates
- The tasks can be displayed as a Gantt chart, with the task details shown beside them
- If the status of a task is not changed to complete before the end date, a push notification will be sent
- 24 hours before the end date of a task, a push notification reminder will be sent
- The share function will allow the user to share the Gantt chart with Project Managers and team members through email or messages
- The database is stored locally on the device

Non-Functional Requirements

- Performance
    - Loading, requests and responses should occur under 3 seconds
    - App should be able to handle up to 100,000 concurrent users with little to no impact on app performance
    - Handle large storage and user traffic
- Security
    - Follows best practice for collecting, storing and protecting user data
    - User sign-in and authentication
    - Compliance with Privacy Act 2020 and other New Zealand or international standards
- Accessibility
    - Accommodates users with varying levels of technical experience
    - Minimalist design
    - Doesn't use harsh colours for users with visual problems
- Usability
    - User-friendly interface
    - Clear navigation, up navigation with obvious exits
    - Follows Material Design 3 standards
    - Match between the system and the real world for icons
    - Error prevention for user input

Constraints & Boundaries

Constraints in a project refer to limitations and risks that must be accounted for in a project. Generally these fall under the categories of cost, time, scope, resources, risk, and quality. Boundaries are certain things that define what is included or excluded from the project to ensure that scope does not expand to become unrealistic or divert from the objectives. Often constraints and boundaries come from lack of skills and resources. Below is a list of the main constraints and boundaries for this project:

- Scheduled time of 18 weeks, starting from the 8th of July 2024 and to the 7th of November 2024
- App platform is restricted to Android, and version is restricted to Android 10
- Budget is $0
- Software to plan and develop must be free, or is a pre-existing expense that won't be counted towards the project
- Database is restricted to be local
- App won't be hosted on server or have an API
- The data storage is limited to the user's device
- App's language is limited to English
- There is no security measures, other than login authentication

- The Gantt Chart will have no collaboration features
- Needs to follow Material Design 3, digital design and accessibility standards

Project Deliverables

The "Get Ahead" project will deliver several key components. The primary deliverable is the Android application, developed in Java for Android 10 using Android Studio. This app will feature task creation and management, an adjustable Gantt chart, notifications, progress tracking and calendar integration.

Accompanying this will be a local SQL database, ensuring that task data is stored securely on the user's device. The application will boast an intuitive user interface designed to adhere to Material Design 3 standards and accessibility guidelines, facilitating ease of navigation and task management.

Additionally, the project will include a dynamic Gantt chart that updates based on user input, reflecting any changes in project timelines and accommodating delays. Users will also have the capability to share updates with Project Managers and team members.

A detailed project plan will outline the timeline, milestones, and resource allocation, guiding the development process and ensuring the timely delivery of the application. Another document known as the project completion report will cover the experience of the execution of the project and summarize the final outcome.

Implementation Plan

System Requirements and Scope Definition

- System Requirements:
    - The app should allow users to manage tasks by entering task details such as names, start dates, and end dates.
    - The app must generate a Gantt chart based on task data and update it when tasks are modified
    - A local SQL database is required to store tasks
    - The app should run on Android 10 and above.
    - Users must have a seamless experience, ensuring task management is intuitive.
- Variables for Consideration:
    - Handling task inputs, data storage in SQL, and a real-time chart generation.
- User Accounts:
    - Optional integration for user login for personal task data storage, if future features expand.
- Project Scope:
    - App development is focused on task management with Gantt chart visualisation. Other features such as reminders or notifications are beyond the current scope.

System Design and Architecture

- Application Structure:
    - Use of Android Studio with Java for app development
    - MVC (Model-View-Controller) design pattern to ensure separation between user interface, database, and logic.
- User Interface (UI):
    - Wireframe design showing task input forms, Gantt chart view, and main navigation.
    - Colour scheme, logo, and overall aesthetic should minimalistic and functional.
- Database Structure:
    - Local SQL database with a table to store tasks, with fields like task name, start date, end date and task ID.
- Implementation Plan:
    - Front-end:
        - Design task input forms and the Gantt chart interface using Android layouts (XML).
        - Implement task list and form functionality, handling user input.
        - User Bootstrap for any web or front-end dependencies, if needed for design consistency.
    - Back-end:
        - Develop local SQL database is SQLite to handle task storage.
        - Write queries to manage task insertion, updates, and deletions.

Resource Allocation and Timeline

- Resource Allocation:
    - Assign roles for development: Front-end and Back-end developers.
- Timeline:
    - Create a schedule for the app's development, outlining key milestones and deadlines.

Testing and Quality Assurance Procedures

- Testing Procedures:
    - Unit tests for individual functionalities (task creation, editing, and deletion)
    - Integration tests to ensure the Gantt chart accurately reflects task data.
    - Test the performance of the app across different Android devices running Android 10 and above.
- Quality Assurance:
    - Ensure the app performs smoothly without crashes.

- o Verify the accuracy of the Gannt chart generation based on task data

Deployment and User Training Plan

- Deployment:
    - o Build and test the APK file in Android Studio
    - o Ensure compatibility with Android 10
- User Training
    - o Develop a short user guide for first-time users on how to input tasks and interpret the Gannt chart.
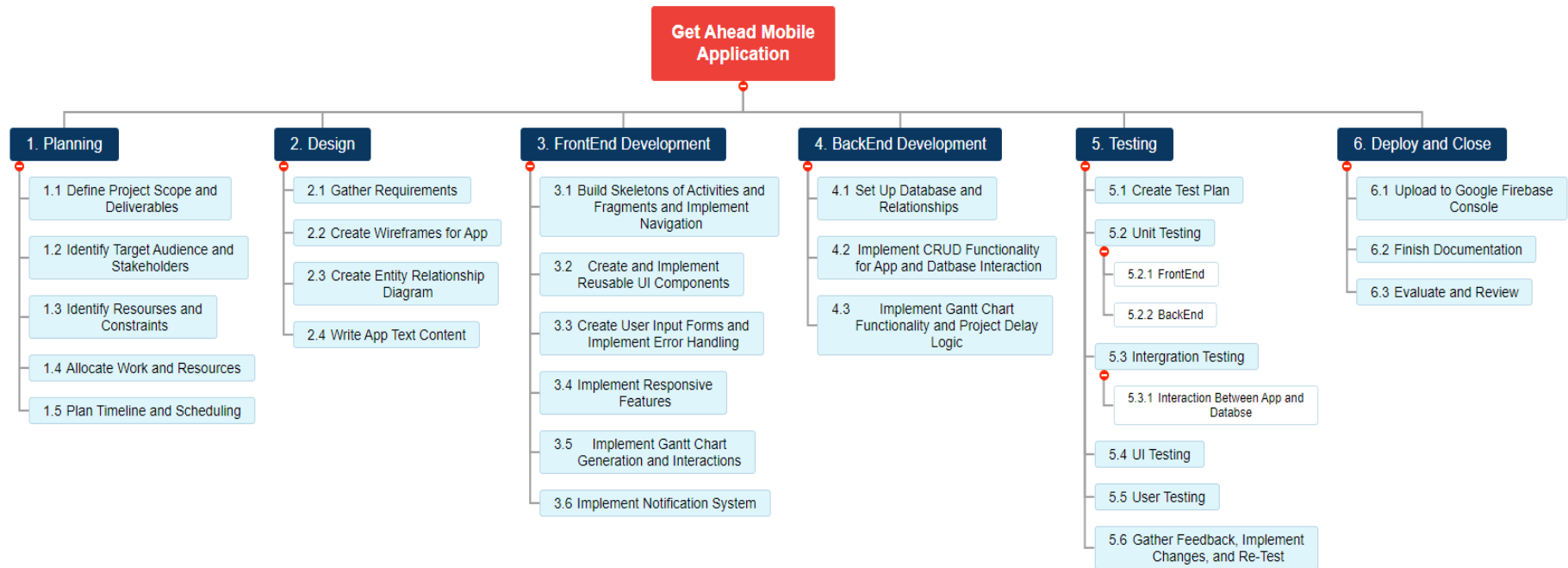
Work Breakdown Structure

Figure 2 Work Breakdown Structure

Project Schedule

Get Ahead Mobile Application

Task	Start Week	Duration (Weeks)	End Week
Planning Phase - Paige and Alia			
Define Project Scope and Deliverables	1	1	1
Identify Target Audience and Stakeholders	1	1	1
Identify Resources and Constraints	2	1	2
Allocate Work and Resources	2	1	2
Plan Timeline and Scheduling	2	1	2
Design Phase - Paige, Bilee, Alia			
Gather Requirements	3	1	3
Create Wireframes for App	4	1	4
Create Entity Relationship Diagram	5	1	5
Write App Text Content	5	1	5
FrontEnd Development Phase - Paige and Alia			
Build Skeletons of Activities and Fragments and Implement Navigation	6	2	7
Create and Implement Reusable UI Components	8	1	8
Create User Input Forms and Implement Error Handling	9	1	9
Implement Responsive Features	10	1	10
Implement Gantt Chart Generation and Interactions	11	1	11
Implement Notification System	12	1	12
BackEnd Development Phase - Alia and Bilee			
Set up Database and Relationships	8	2	9
Implement CRUD Functionality for App and Database Interaction	10	2	11
Implement Gantt Chart Functionality and Project Delay Logic	12	1	12
Testing Phase - Alia, Bilee, Paige			
Create Test Plan	13	1	13
Unit Testing (Frontend & Backend)	14	2	15
Integration Testing	16	1	16
UI Testing	17	1	17
User Testing	17	1	17
Gather Feedback, Implement Changes, and Re-Test	18	1	18
Deploy and Close Phase - Bilee			
Upload to Google Firebase Console	18	1	18
Finish Documentation	18	1	18
Evaluate and Review	18	1	18

Figure 3 Gantt Chart

Project Resource Management

Effective management of resources is crucial for the successful development and deployment of the "Get Ahead" application. The following resources will be utilised throughout the project:

1. Human Resources:
   a. Development Team: Composed of skilled developers proficient in Java and Android Studio, this team will handle coding, application design, and development tasks. Their focus will be on ensuring the application meets technical standards and performs efficiently.
   b. Quality Assurance (QA) Team: Responsible for testing the application to identify and address bugs, usability issues, and functionality concerns. The QA team will work to ensure a seamless and error-free user experience.
   c. Project Manager: Oversees the project's progress, coordinates between teams, manages timelines, and ensures that the project stays within scope and budget. The Project Manager will also handle stakeholder communication and resource allocation.

2. Technical Resources:
   a. Development Tools: The project will utilize Android Studio for app development and SQL database management tools for local data storage. These tools are essential for coding, debugging, and managing the application's data. Figma is going to be used for wireframe design.
   b. Hardware: Development and testing will be conducted on Android devices compatible with Android 10 to ensure the application operates as intended across target devices.

3. Budget and Time Allocation:
   a. Time: the project is scheduled to span 18 weeks, from July 8, 2024, to November 7, 2024. Adherence to this timeline is critical to ensure timely delivery of the application.
   b. Budget: The project operates with a zero-dollar budget. Therefore, all software and tools used must be free or pre-existing expenses not counted towards the project budget.

4. Project Constraints:
   a. Platform Limitations: Development is restricted to Android devices running version 10. No support will be provided for other platforms or versions.

b. Data Storage: The application will use a local SQL database for data management, with no external cloud-based storage or server hosting.
c. Language: the application will be developed and supported exclusively in English.
d. Security: The project will not include advanced security measures beyond standard practices. User data protection will adhere to basic privacy standards.

Effectively managing these resources and constraints, the "Get Ahead" project will deliver a robust time management tool that meets its objectives and helps users manage their projects efficiently.

Project Risk Management

As with all projects there are going to be risks in regards to completion dates and if the deliverable of the project addresses the issues to which it was made to ease or remove. The project members are responsible for making sure that the chances of these risks becoming detrimental to the project and its deliverables are reduces to a size that is controllable to deal with.

Time Risk

Time risk is named after the risk of the project not becoming completed on time or being completed far before the expected completion date. The project being completed early is of course in most cases a positive thing as it reduces the money spent on the project. On the other hand the project being completed after the estimated completion date is not ideal in any situation as it increases the project scope which in turn increases the money and resources spent on the project.

With the increase of the project scope and money and resources spent. The data and documentation you have for the project can and should be used to help with the planning of future projects as it can show where the weak spots of time management lay within the timeline of the project and where the project members can improve on getting to milestones faster. The use of the project that has crumbled to Time risk and seeing where members can improve can potentially help decrease the time, money and resources spent on future projects.

Currently the project members are having weekly team meetings to make sure that by each week they are where they need to be and what needs to be completed next. This is helping with the management of Time Risk as it allows the team to carefully plan out each week and the weeks to follow all while holding constant communication with the other members. Making sure that all members are on the same page with the timings and milestone dates within the project itself. It also helps the project members rearrange and redelegate work if it is deemed necessary.

Scope Creep

Scoop Creep occurs when the project itself goes outside the lines of the scope of the project. Meaning, when too many changes are made to the project during the later phases such as Execution and have not been properly accounted for after such changes were made or pushed. With the "Get Ahead" project there are many differing ideas being tossed around within the group to decide what needs to go where and if an extra aspect or additional feature could/needs to be added to the app to improve the potential User Experience (UX).

To keep the risk of Scope Creep down, during the first meeting within the team, the team members agreed on what the base idea of the app would do and what problem they wish to tackle with creating this application. The next week during the meeting they were asked to share what additional features they wanted the app to have and combined their ideas into a list/ the requirements of what they wanted to app to have. That list of requirements is non-changing and is going to be stuck too all though out the development/execution phase of the "Get Ahead" project.

Communication

As with any project concerning a group, communication is very important and it keeps all members within the loop of what has happened, is happening and what needs to happen. If the communication between the team members is poor then the outcome of the project will also be poor and vice versa. As the members within the "Get Ahead" project are all familiar with each other it is rather easy to keep communication up and constant. The importance of communication between team members throughout the entire project was brought up even before the project planning had commenced.

 The project members keep constant communication with one another in a form of a group chat in the application Discord, where updates questions and just general comments about the project are shared regularly.

Lack of Documentation

A project is only as good as its planning phase and the key to having a good planning phase is to have good documentation. Having poor and/or a lack of documentation can lead to the projects getting Scope Creep as there was no way of knowing what the scope properly is. The chance of falling into Time Risk increases as well due to there potentially being either no or a very bad Gantt chart to help project members keep track of time and the tasks that need to be completed before hitting certain milestones or even knowing when certain milestones need to be completed by.

The project members of the "Get Ahead" project are making sure that the documentation for this project is always kept up to scratch and is completed before moving onto the next step. Once again this is achieved by having the weekly meeting and being in constant communication with each other over the group chat within Discord. There is also the fact that all the documentation is done within shared Word Document that each member can edit, this makes it so that there is no information that is getting left behind when sharing information between individual documents and it also keeps all members up to date with the most current information and documentation we have.

Proper Planning

As stated above a project is only as good as its planning. As most if not all about the project is decided within the Planning phase it makes sense that the most amount of time and focus must be spent on the planning. The way that the "Get Ahead" project members are making sure that the planning phase is running smoothly and is being completed to the highest degree is once again by the meetings that are held weekly by its members. These meetings are essentially checkpoints of information to be shared within the team to help use move through the planning process smoothly.

Project Control

What is Project Control?

When projects are left unchecked, changes are made without any thought, budgets get overrun, the scheduling turns messy, and the operations of the project as a whole can be at risk. As the Project Manager oversees the project, they are responsible for monitoring and controlling all aspects of the project to ensure it stays on track and within the scope.

Project control refers to the processes used to regulate the progress of a project as well as assess and identify any issues, which in turn assist the Project Manager with making informed decisions. According to the Asana website, the five main project controls are schedule, resources, risk, change and performance. The next sections explain how our group has implemented these project controls in our project.

Schedule

As we are under a very tight time frame for this project, any major delays in the project will cause us to be unable to complete the project. We have implemented the following measures to ensure that our project stays on track:

1. WBS(Work Breakdown Structure) – When were first planning the project, we made a list of all the phases of the project and what tasks were needed for each phase, then if needed broke down each task further. Then they were organised into a WBS so it was clear what was needed to be done for the project to be deemed completed

2. Gantt Chart – Then we took everything from the WBS and analysed how long it would take how us to do each task. Then we picked who did which task, and played to our strengths so that would be the best possible outcome for that task. The Project Manager created a Gantt Chart(has shown above) which depicts the project phases, all task within them, who is responsible for that task, the duration of the task, and then a calendar-like view of the weeks in the project with coloured areas showing when the task is going to be done

3. Weekly Meetings – During the duration of the project, the project team come together least once a week for a 30 minute meeting where we discuss how the progress is coming along and where we bring up any issues or delays we have faced. From here the Project Manager will take any necessary actions after we have reported to ensure that we are on track for the coming weeks. Any adjustment to the WBS and Gantt Chart also take place during this time so we are all on the same page

Resources

During the planning and initiation stages of a project is when the Project Manager should determine and gather the resources needed to complete the project, as well as allocate as necessary. And as the project goes on, there must processes in place to monitor the usage and depletion of these resources to ensure that issues don't arise that are detrimental to the project. This is how we implemented resource controls:

1. Resource Allocation – At the start of the project the Project Manager allocated who would do what based on the skills of the team member.
2. Scheduling and Time – We used a Gantt Chart to schedule how long we would allocate for each task
3. Monitoring – The Project Manager would regularly check-in to see how our tasks were progressing and ensure everyone is on the right track and without issues

Risk Register

ID	Risk Description	Risk Category	Odds 1-3, Impact 1-3	Risk Score (Odds x Impact)	Risk Mitigation
1	Application components chosen for interface aren't compatible or accessible on chosen development platform	Technical	1, 1	1	Explore design features of platforms before creating wireframes
2	Team member/s away for extended periods of time, causes delays	HR	2, 2	4	Redistribute tasks based ahead of time, if sudden establish times when remote meetings can occur
3	Developer/ team member training and skills are inadequate	Operational	2, 3	6	When creating requirements for project deliverables ensure that team members have the necessary skills to complete these requirements or that the knowledge to do so is in reach
4	Scope creep, sudden changes in requirements	Operational	1, 3	3	Ensure that stakeholders are clear in their expectations and that the list of requirements drawn from those are also approved from the stakeholders. Regulation communication with stakeholders. Have a change management process prepared

5	Communication issues, lack of check-ins and updates	HR, Operational	2, 2	4	Plan regular team meetings, have a common/causal method of communication or check-ins and updates
6	Data loss	Technical	1, 3	3	Back up all work, when major changes are being made copy and version
7	Insufficient testing, lack of user approval	Technical	2, 3	6	Ensure that thorough testing plans are created to cover all areas of testing
8	Mobile app incompatible with many devices	Technical	2, 2	4	Research best Android version to build the app on, as well as ensure that features of app are scalable

Change

Requirements and other variables are prone to change at any time in a project, even if during the planning stage everything was perfectly written, and the decisions were firm. Stakeholders may change their mind, the budget might get tight early in the project, or resources may get depleted a lot quicker than what was originally thought. Normally a change will come under one of three categories: scope, time, or budget.

When a change isn't handled correctly, it can leave a huge negative impact on the project as a whole. To correctly handled changes, the Project Manager should implement what is commonly known as a change approval process, which will manage any unforeseen changes.

The change approval process has five main stages, although there are various versions out there, these are the most common ones. Below are these five stages and how we apply this to our project:

1. Submission of Change Request – when one of the team members wants to propose a change to the project, they present their case to the Project Manager so that the reason for the change, the adjustments to the scope and timeline, and any other impacts of the change to the project will be known
2. Initial Assessment – the Project Manager will then ensure that all the necessary information regarding this change has been collected so that when it comes time to evaluate the proposal they can be sure that there is no piece missing. They will also determine if this change request has some validity, or if it is an unreasonable request
3. In-depth Evaluation – normally an evaluation would be carried out by a special board with representees from the various departments in an organisation which would review the change request and its impact on the departments. As it is just

three members for our project, we get together to evaluate this change and the impact it will have on our project, but the Project Manager has the final decision

4. Implementation of the Change – if the change request has been approved then the project team members can begin to plan and execute the change, but they also have to be cautious to ensure that the impact on the current project is minimal

5. Verification and Validation of the Change – once the change has been officially implemented, then the Project Manager will check to ensure that the change was implemented successfully and that the project still meets the requirements of the stakeholders and that no risks have been identified

Performance

Performance is normally measured using KPIs(Key Performance Indicators), but as that is for calculating the performance of the result of the project, we will not use this method. Instead performance is measured in a less tangible way through assessment of team members work by other team members and the Project Manager.

The quality of the output and how long it takes to do a task(if the time taken to complete the task is over the expected duration is unreasonable) are what the Project Manager assesses to determine whether or not the team member is functioning at a good level. If the Project Manager finds that the team member isn't producing quality work in a timely manner then they will take corrective action to ensure that there is minimal disruption to the project quality and schedule.

This assessment occurs during the weekly meetings and after a task has been completed.

Techniques Used in IT Projects

Change Management

As mentioned above, requirements and other variables are prone to change at any time in a project, and IT projects are no exception to that. "IT change management is the process of tracking and managing a change throughout its entire life cycle, from start to closure, with the aim to minimize risk."(Manage Engine, n.d.)

Having a solid change management system ensures that if these sudden changes get requested, they can be carefully integrated with the current system with minimal disruption to the project and it's development. The people who are responsible for overseeing the change management process normally include the Project Manager and a board of representatives.

The five main stages of this process are:

1. Submission of Change Request – the stakeholder submits a request to make a change to the Project Manager
2. Initial Assessment – the Project Manager will ensure all necessary information is gathered and it is a reasonable request
3. In-depth Evaluation – the Project Manager will present the request to the board who will evaluate the risks and impact of the proposed change
4. Implementation of the Change – the Project Manager will oversee the change during the implementation of the request and make sure to mitigate risks
5. Verification and Validation of the Change – the Project Manager will check that the implementation result fits the stakeholders request and that the impact was minimised

Configuration Management

An IT project like this one involves software, hardware and documentation, which are essential to completing a quality project. Configuration management is the management process of ensuring that the configuration of an IT system is consistent through the project's lifecycle, or that any changes made to the configuration will have minimal impact on the project.

When a configuration change is left unchecked like a new software update or hardware upgrade, issues like version conflicts, incompatible software and hardware, system crashes, and the documentation will not contain the necessary information to trace back to what went wrong. This can lead to severe delays in a project.

There are a few people that are responsible for overseeing configuration management, the Project Manager, System Manager/Administrator, and the Developers. Below is a summary of the configuration management process:

1. Planning – create a configuration management document that contains details on how the IT system configuration will be tracked/audited
2. Identification – all the configuration requirements for a project will get recorded in the document, so that the original configuration is known
3. Control – if the project scope changes, then the impact on the configuration must be looked at, this is done during the change management process
4. Status Accounting – if the configuration changes at all, then the new configuration details must get recorded in the document
5. Audit – this is when the deliverable gets tested on the configurations to ensure that any issues get identified, normally occurs after major milestones or changes in system configuration

Quality Control

When you first think of quality control, you picture a production line with workers testing a product to see if it works correctly, and if the product is faulty it will get thrown away so only the working product makes it way to the stores. Quality control in a project works a little bit differently to quality control in a production line, but they both have the same goal of delivering only the best result to the stakeholders.

No matter which methodology one follows during a project there is always going to be the following five phases slotted somewhere:

- Initiation
- Planning
- Execution
- Closure
- Monitoring and Controlling

The first four phases are done one by one, but the last one, monitoring and controlling, should be done throughout the project lifecycle.

"In a project life cycle, monitoring and controlling is the process of regularly observing and tracking the progress of your project and making any necessary proactive corrections." (Johansson, 2023). Monitoring means that there is regular tracking and observation during the project and controlling means taking action if needed and then putting controls in place to prevent issues(works together with risk management).

Monitoring and controlling is essential in quality control as it ensures that the project will meet the requirements while making sure the process goes as smoothly as possible, providing a high-quality deliverable.

References

ActivTrak. (2024, February 1). *What Is Project Time Management and Why Is It Important?* Retrieved July 25, 2024, from ActivTrak: https://www.activtrak.com/blog/project-time-management/#:~:text=Without%20someone%20managing%20project%20timelines,team%20members%20fall%20further%20behind.

Asana, T. (2024, February 1). *What is a risk register: a project manager's guide (and example)*. Retrieved July 22, 2024, from asana: https://asana.com/resources/risk-register

Drucker, P. F. (2001). *The Essential Drucker: Selections from the management works of Peter F. Drucker*. Harper Collions.

Gurnov, A. (n.d.). *What is configuration management in project management?* Wrike. Retrieved September 9, 2024, from https://www.wrike.com/project-management-guide/faq/what-is-configuration-management-in-project-management/

Hathibelagal, A. (2015, March 16). *Add charts to your Android app using MPAndroidChart*. Code Envato Tuts+. Retrieved October 24, 2024, from https://code.tutsplus.com/add-charts-to-your-android-app-using-mpandroidchart--cms-23335t

 Johansson, J. (2023, April 13). *Project monitoring and control: Complete guide (+ checklist)*. Resource Guru. Retrieved September 9, 2024, from https://resourceguruapp.com/blog/project-management/project-monitoring-and-control#:~:text=it%20really%20is.-,What%20is%20monitoring%20and%20controlling%20in%20project%20management%3F,on%20track%20and%20within%20budget

ManageEngine. (n.d.). *IT change management: A comprehensive guide - ManageEngine*. Retrieved September 9, 2024, from https://www.manageengine.com/products/service-desk/it-change-management/what-is-change-management.html#:~:text=IT%20change%20management%20is%20the,with%20a%20high%20success%20rate

Sarthi Technology. (2018, September 9). *001 Introduction : MP Android Chart Tutorial* [Video]. YouTube. https://www.youtube.com/watch?v=N-_X6G1KgAY

statcounter GlobalStats. (2024, July). *Mobile Operating System Market Share in New Zealand - July 2024*. Retrieved August 2024, 15, from https://gs.statcounter.com/os-market-share/mobile/new-zealand

Team Asana. (2024, January 18). *5 project controls and where to implement them*. Asana. Retrieved September 12, 2024, from https://asana.com/resources/project-controls

Wrike. (n.d.). *What is change management in project management?* Retrieved August 27, 2024, from https://www.wrike.com/project-management-guide/faq/what-is-change-management-in-project-management/

PROJECT COMPLETION REPORT

Get Ahead Mobile Application

ITC601 Project Management
Assignment 2
2024

Paige Clay – 2019012408
Bilee Billyard – 2023002556
Alia Haizat - 2022006035

Table of Contents

Table of Figures

Project Overview

The outcome of this project has delivered a mobile application called Get Ahead, which is a time management app built for Android users. Our inspiration for building this app is because all of our team members are students, and it is a constant battle to effectively manage our time.

When the teams were first formed, there were four members in our group, each with a good knowledge of Mobile Development so we were confident that we could produce a quality deliverable. There was also discussion of creating a Windows Forms application with a server database that could also connect to the mobile app, but after a meeting with Alex Ewing (IT601 Tutor) it was decided that it would be too much work for the short amount of time.

The project teams members consisted of the following:

- Alia Haizat (Current Project Manager)
- Bilee Billyard
- Paige Clay
- Jacob Warren (Original Project Manager)

A couple of weeks into the project Jacob stopped attending classes and it was difficult to contact him. We later found out that he was leaving the course and therefore was no longer going to be a part of our group and participate. Due to his sudden leaving and the fact that the idea and vision mostly came from him, it did leave us scrambling to adjust without our project manager.

Alia stepped up as Project Manager and played a big role in reassessing and redefining the project concept and redistributing the workload. Unfortunately, as it was a sudden occurrence, it did cause a few weeks of delay.

Project Description

Get Ahead was developed using Android Studios as a mobile application built for Android devices with the Android 10 operating system. As mentioned above, it is a time management app which allows users to create a project and then schedule tasks for that project. We wanted the app to then generate a Gantt Chart, but due to time restraints and skill issues, we were only able to get as far as displaying the Gantt Chart with pre-set data, same for the "next three weeks" Gantt Chart view. There is a working To-Do List view that displays the tasks by start date.

Users can adjust the end date for a project (maybe due to a deadline changing), which then will cause the app to automatically adjust the dates of the tasks in that project if they are unfinished. The app was created to encourage effective time management and help users to deal with unexpected delays and adjust their timelines accordingly.

The deliverables of this project were:

- Android App: Developed in Java for Android 10 using Android Studio as the IDE.

- Local Database: Managed with SQL in Android Studio
- Project Implementation Plan
- Project Completion Report

Problem and Solution

As students, it can be very challenging to balance classes, assignments, work, and home life. Assignments in particular can be difficult to manage, as there can be multiple on the go at once each with different tasks. Also, as delays can be quite common when handling many things at once, it can be hard to manage.

Our team's solution to this problem is creating an efficient method of tracking all the stages of an assignment, similar to how the stages of a project get tracked, using a Gantt Chart. We decided that the best way to implement this solution was by developing a mobile app as most people have mobile devices and it would always be accessible.

Project Objectives

Our main objective is to ensure that users are better equipped to manage their time when doing anything that involves multiple tasks or stages e.g. projects and assignments by delivering Get Ahead as a time management app.

We successfully met this objective by developing the Get Ahead prototype, which mostly functions as intended. It allows the user to switch between projects, as well as view the project schedule in a To-Do List view. If time allowed us, then the Gantt Chart view and "next three weeks" week would have been able to pull and display the stored data. If the user has a delay or advancement for a project they can change the end date of the project which will adjust dates of all its tasks.

Deliverables

Although the main deliverable of this project is a fully functional prototype of the application, professional documentation is also required. Below is a summary of the deliverables as well as a breakdown of what was functionality was needed for the prototype to be considered completed.

- Executive Summary
- Plan Report Document
  - Scope
  - WBS
  - Gantt Chart
  - Stakeholder Register
  - Risk Register
- Completion Report Document
  - Wireframes
  - Prototype Screenshots
  - Challenges and Solutions

33

- o Lessons Learnt
- Presentation
  - o Project Overview
  - o Persons Involved
  - o Demonstration of Prototype
- Mobile Application(Get Ahead)
  - o Login and Registration System
  - o Multiple Views for Project Schedule
  - o User Can Add, Edit, and Delete Tasks and Projects
  - o User Can Have Multiple Projects
  - o Download Gantt for Sharing

Stakeholders

Stakeholder Roles	Members	Project Phase/s Involved
Project Manager	Alia Haizat	All
Development Team	Alia Haizat, Paige Clay, Bilee Billyard	Planning, Design, Development, Launch
Quality Assurance Team	Alia Haizat, Paige Clay, Bilee Billyard	Testing
Educational Institution	Alex Ewing	All
End Users	Students	Launch

ID	Role	Category	Interest	Influence	Expectations
1	Development Team	Internal	The technical success of the app (coding standards, user interface design and functionality)	Medium – High	Deliver a well-designed, fully functional and technically sound app.
2	Quality Assurance Team	Internal	Ensures the app is bug-free, user friendly and functions as intended	Medium	Identify and report any bugs, ensure a smooth and error-free user experience
3	End Users	External	Usability, effectiveness and features	Medium	Access an intuitive, effective app that helps end users

					manage their time efficiently
4	Educational Institutions	External	Tools that help students manage their time and succeed in their endeavours.	High	Receives a reliable tool that enhances student productivity and academic success.

Target Users

We believe that students of all ages would benefit from this app and so they are our target users. As students ourselves, we understand that it is difficult to manage everything at once with different workloads and different due dates.

As this app is limited to the Android 10 operating system, we do understand that this does affect the accessibility of the app, but we do hope that in the future to expand to iOS and increase our target user base to include professionals as well.

Scope and Boundaries

Scope

The project scope was to develop a mobile application called Get Ahead, which is an Android app built for devices compatible with Android 10 operating system. The primary objective was to aid users, particularly students managing multiple assignments and coursework, by providing an adjustable Gantt chart for effective project planning and time management.

The application was to have task management capabilities, which would allow users to create and track tasks in with associated start and end dates in a project. It would generate a visual Gantt Chart that dynamically updated based on user input, reflecting changes in project timelines and accommodating any delays. There would also be options to view the project as a To-Do list, and a "next three weeks" view.

Boundaries

The boundaries of the project were defined as follows:

- The application is developed using Java and Android Studio, targets Android 10 devices only
- Data is managed locally using SQL database and does not have cloud-based synchronisation or external data storage.
- The user interface is designed to be intuitive and user-friendly but does not have advanced features such as voice commands or AI-based scheduling recommendations due to lack of development skill. Sharing functionality is be limited to exporting the Gantt chart as a file or using built-in Android sharing mechanisms, without integration with external project management platforms.
- There is no multi-user collaboration so team member can't work together in real time.
- Users that receive the Gantt Chart from another user are unable to load it into the app.

Project Schedule
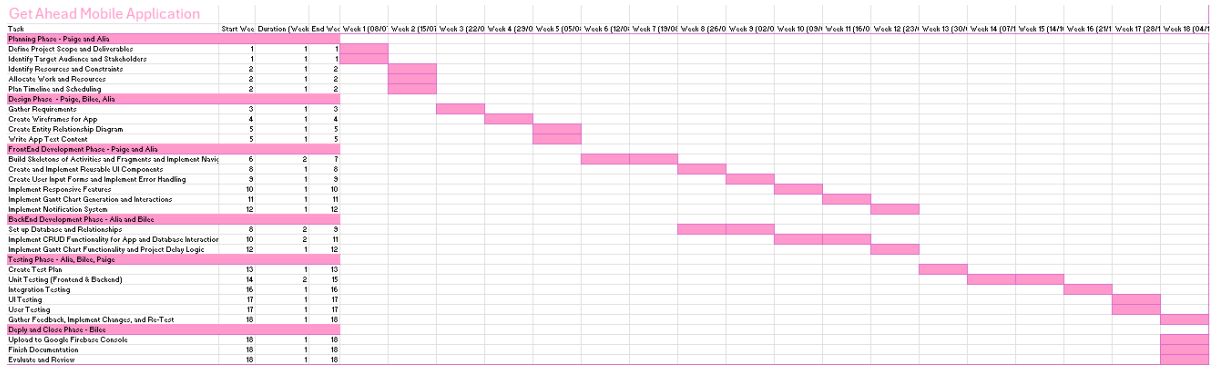
Since Jacob left the group we had a bit of a delay as the project was for the most part his vision and so we had to figure out the details as we went along which caused a delay. The above Gantt Chart represents how the project stages were scheduled before Jacob left. Due to negative impact his sudden departure had on our team, the project planning was off by a few weeks and didn't start until the 8th of August when we got confirmation that he really was leaving.

For the next month until the mid-semester break(7th September to the 22nd September) we focused on creating the project plan. We divvied out the document sections between the three of us and once a section was completed we would get the others to look over and approve our work.

During the mid-semester break, Paige created the wireframes using Figma and then started making the skeleton for the mobile app in Android Studio. Bilee created an ERD so that Paige was able to start add the basic navigation and display functionality for the main activities, especially for the To-Do list view activity so that basic testing could be done.

At the start of October Bilee designed SQL queries and then implemented then in Android studios. Alia then tested all the queries to ensure that the CRUD functions worked before implementing the actual Gantt Chart and 3-week views.

By the end of October we were ready to create the presentation and demonstration video with our fully functional mobile app. We also worked on the completion report during this time by dividing up the sections between us.

Project Implementation

Documentation

 As mentioned before, all team members contributed to the documentation. The Project Manager divvied up the sections in the Planning document and then distributed them out. Below is a rough outline of how the sections were assigned. All of the planning was done before the execution was started as we were following the Waterfall model.

Alia	Paige	Bilee
Project Stakeholders	Project Objectives	Project Objectives
Stakeholder Register	WBS	Project Importance
Implementation Plan	Project Control	Project Scope and Boundaries
Project Schedule	Techniques Used in IT Project	Project Risk Management
Project Resource Management		

Wireframes
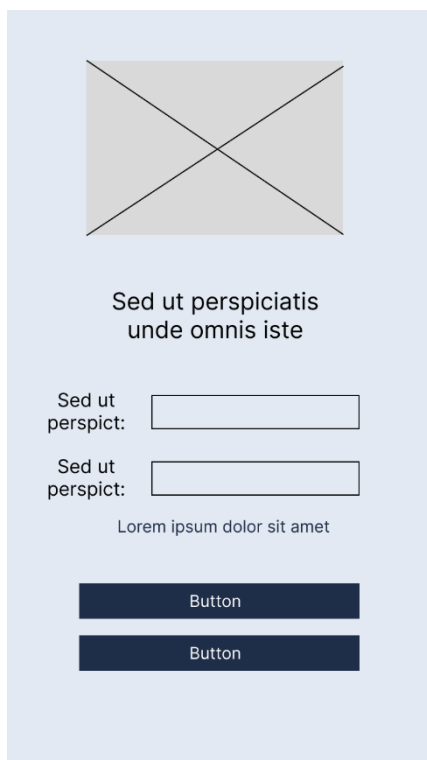
Figure 4 Login Activity
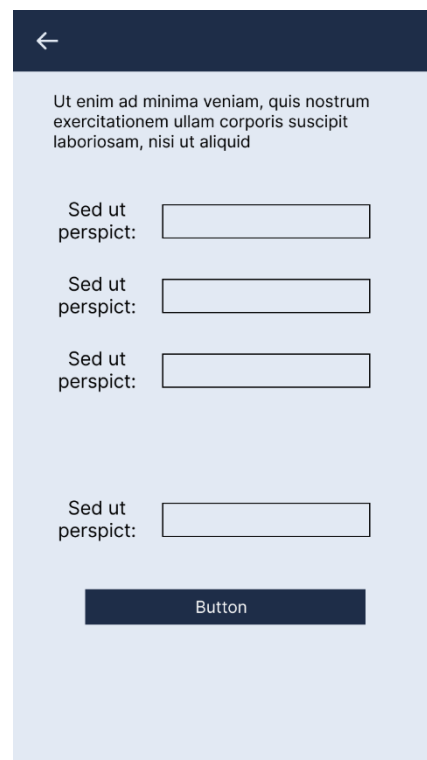
Figure 5 Create Account Activity
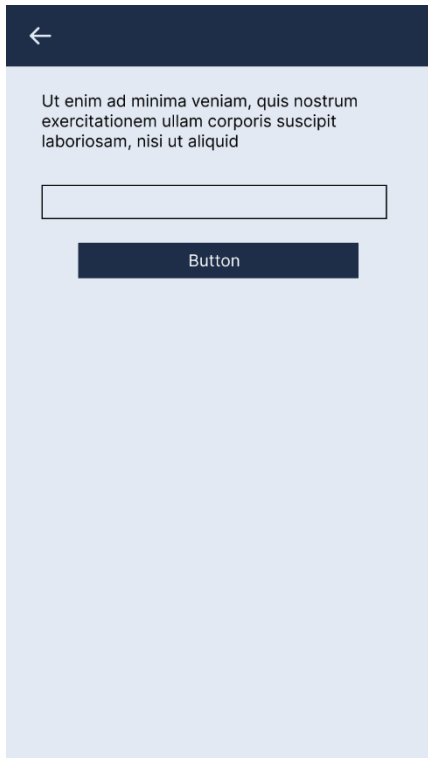
Figure 6 Forgot Password Activity
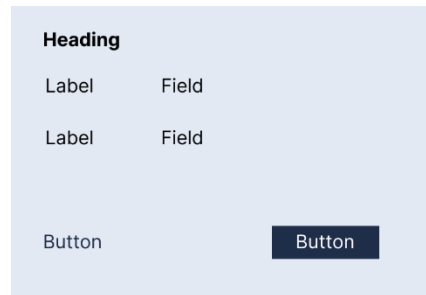
Figure 7 Add/Edit Project Fragment
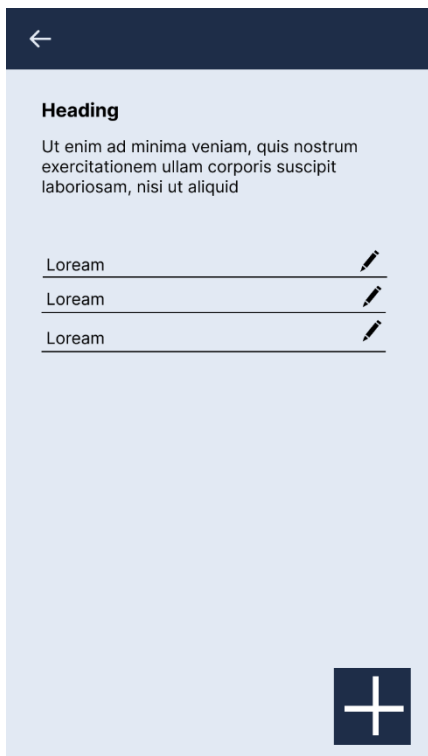
Figure 8 Home Activity
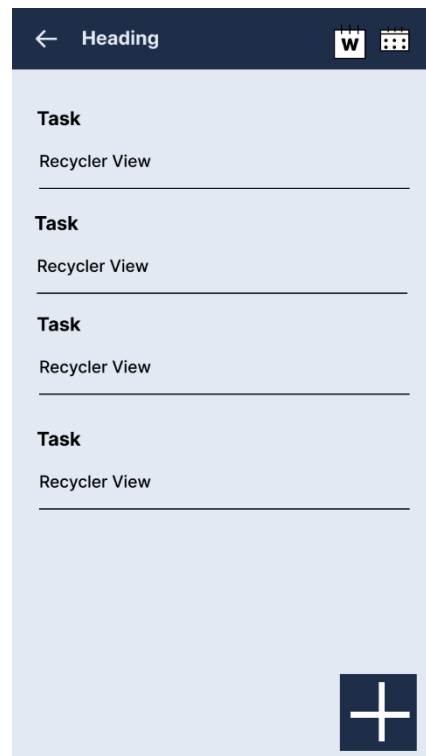
Figure 9 List View Activity

39
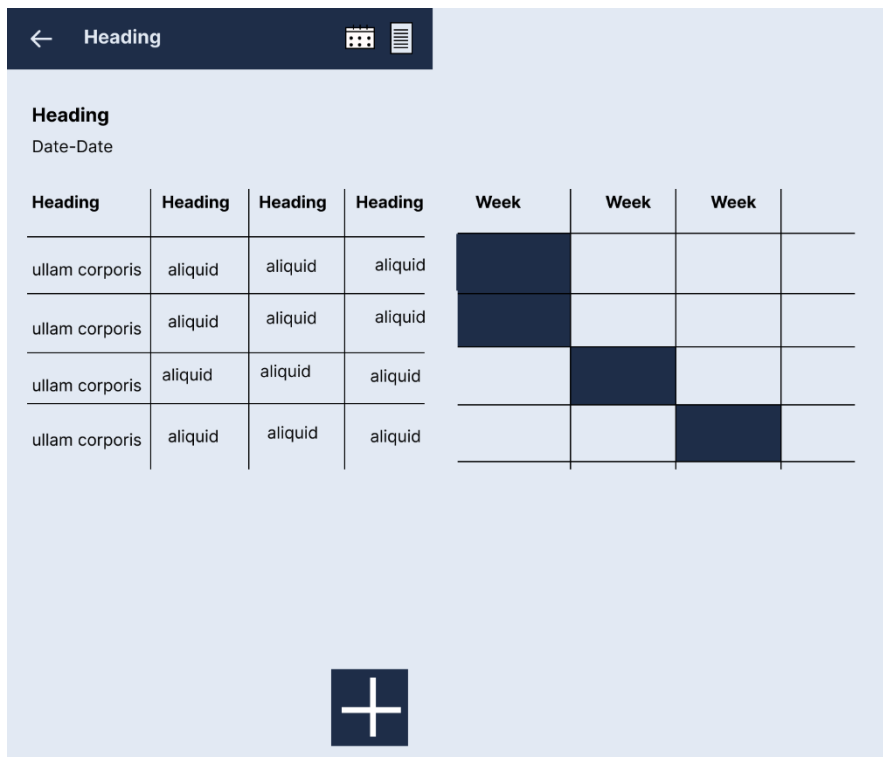
Figure 10 Next Three Weeks View Fragments

Heading	Heading	Heading	Heading	Week	Week	Week	
ullam corporis	aliquid	aliquid	aliquid	■			
ullam corporis	aliquid	aliquid	aliquid	■			
ullam corporis	aliquid	aliquid	aliquid		■		
ullam corporis	aliquid	aliquid	aliquid			■	
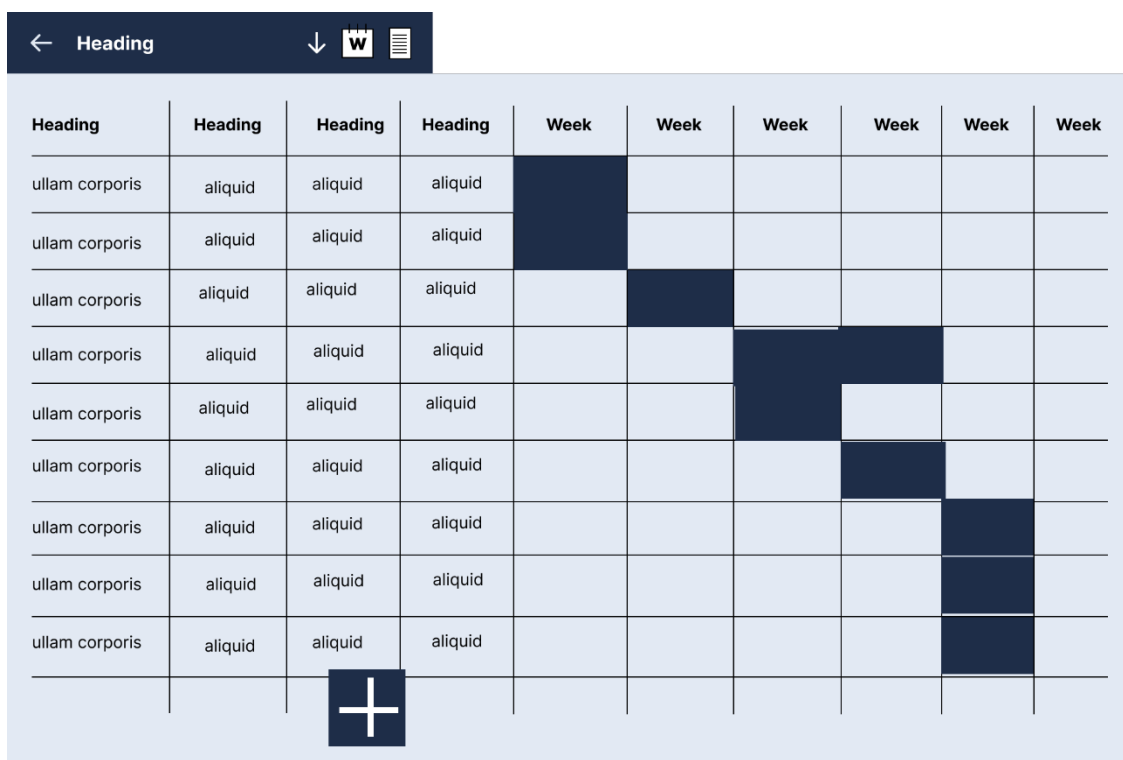
Figure 11 Gantt Chart View Activity

Heading	Heading	Heading	Heading	Week	Week	Week	Week	Week	Week
ullam corporis	aliquid	aliquid	aliquid	■					
ullam corporis	aliquid	aliquid	aliquid	■					
ullam corporis	aliquid	aliquid	aliquid		■				
ullam corporis	aliquid	aliquid	aliquid			■	■		
ullam corporis	aliquid	aliquid	aliquid			■			
ullam corporis	aliquid	aliquid	aliquid				■		
ullam corporis	aliquid	aliquid	aliquid					■	
ullam corporis	aliquid	aliquid	aliquid					■	
ullam corporis	aliquid	aliquid	aliquid					■	

Figure 12 Add Task Fragment

Figure 13 Edit Task Fragment

Early Designs
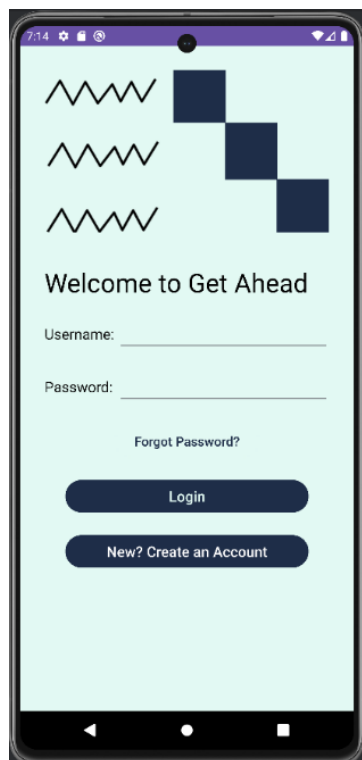
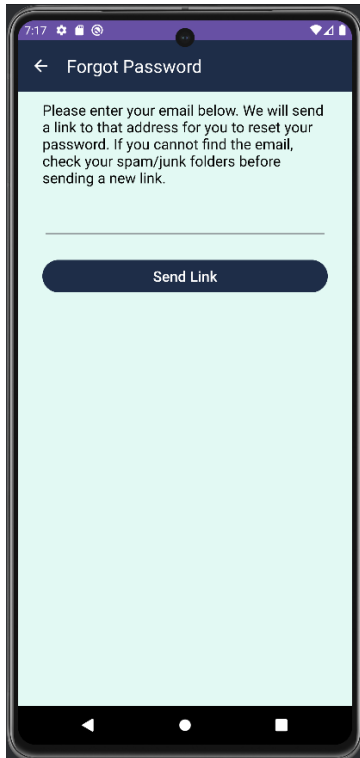Figure 14 Login Activity
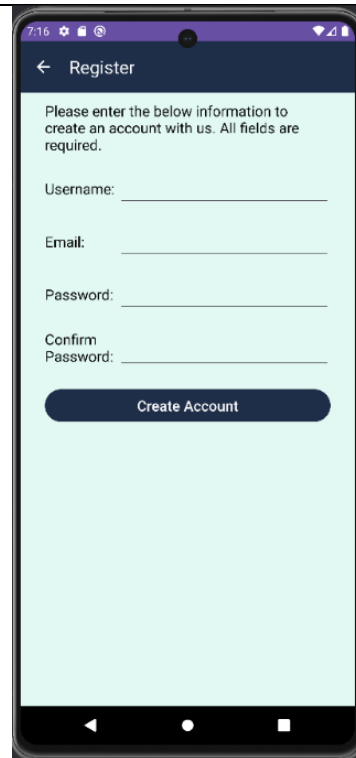
Figure 15 Create Account Screen
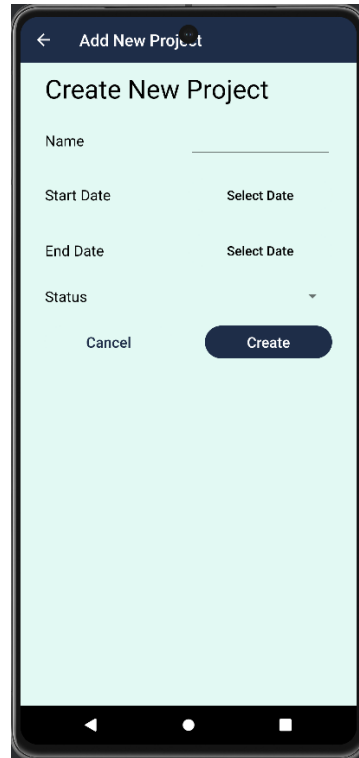
Figure 16 Forget Password Screen
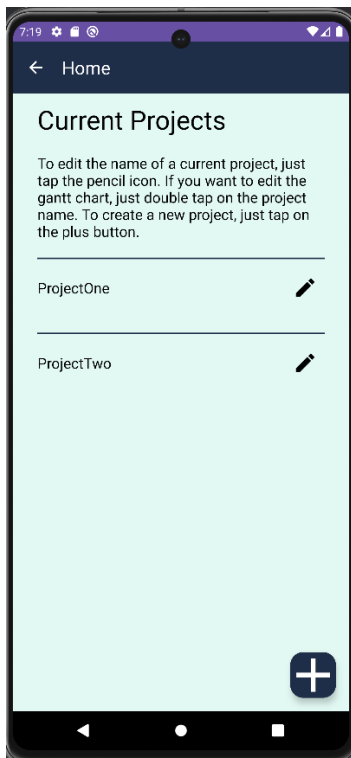

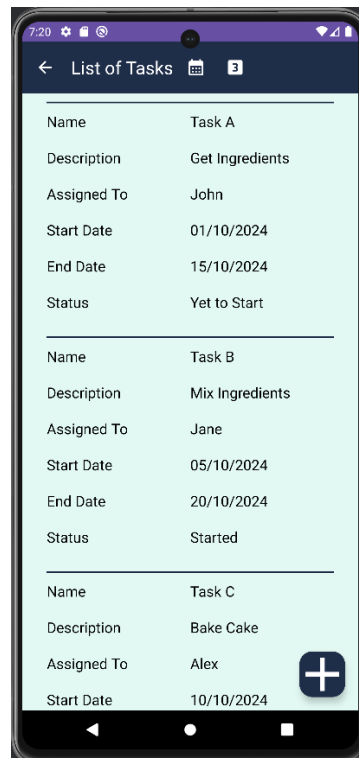*Figure 17 Add Task Screen*

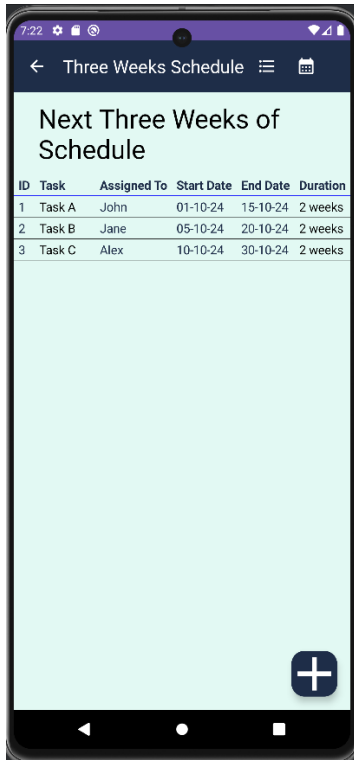*Figure 18 Home Screen*

Figure 19 To-Do List View Screen

Figure 21 Next Three Week View Description Screen
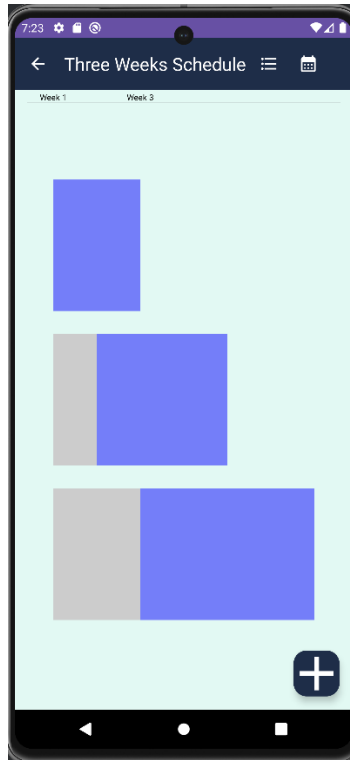
Figure 20 Next Three Week View Gantt Screen

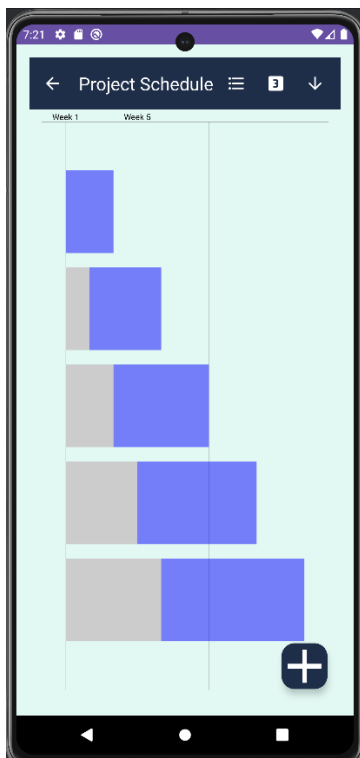*Figure 22 Gantt Chart View Screen*

Figure 23 Add Task Screen

Figure 24 Edit Task Screen

Resource Management

For this project, there were three categories of resources that were used, human, technical, and time. These resources were allocated and managed by the Project Manager using project control and the management system we had formed in the planning stage.

The human resources consisted of our team members. As we were the sole personnel developing this project, all of us played a part every stage of the project. As mentioned before, the Project Manager divvyed up the work between, playing to our strengths so that individually we would not waste time working on something we were unfamiliar with.

The technical resources we used to complete this project are:

- Android Studios – Development Software & SQL Database Helper
- Figma – Wireframe Design Software
- Discord – Main Communication Channel
- Own Laptops
- 2x USB sticks – Transportation Devices

As the software technical resources were free to download and use, not much managing was requiring when using them, except ensuring that the software was keep up to date and items were correctly saved and backed up. For the USB sticks, we had to make sure that we had correctly saved and versioned the project, as well as keep track of who had which USB stick.

In terms of time as a resource, we had a Gantt chart at the beginning which had our breakdown of tasks and the duration of them, but after Jacob left the group we struggled to stick to that timeline as we had to completely re-plan the project. This did cause a delay and so we began the execution stage about three or four weeks later than what we would have liked. With us all having other assignments and commitments this did mean that as we were nearing the end we were struggling to get the project completed, and ultimately some of the apps functionality is not at 100%.

Risk Management

During the planning stage of the project, the team identified various potential risks in the risk register, one of which was the possibility of a team member being away for extended periods, which could cause delays in the project's timeline. This risk was rated as significant, with a mitigation plan that included redistributing tasks ahead of time or in the event of a sudden absence, establishing remote meeting times to ensure continued communication and progress.

This risk materialised with the departure of Jacob a key team member. His departure led to delays in both planning and execution phases of the project. As Jacob was the main architect of the projects vision and objectives, he left the team in a difficult position. However, the team was able to mitigate the risk promptly by reorganizing tasks and leadership role. Alia became the new Project Manager and as a team we redistributed responsibilities. We were able to recover from the disruption and continue as of nothing happened, allowing the project to continue and stay on track and complete the project successfully.

Challenges and Solutions

Project Manager Leaving

One significant challenge was the departure of the original Project Manager during the planning stage. His unexpected absence disrupted the team's workflow as well as a gap in leadership. As mentioned earlier much of the project vision and concept came from Jacob, there was a big display. In response to this challenge the team swiftly reassigned the Project Manager to Alia. For the next few weeks after that we spent a lot of time meeting together to break down the project, figure out the scheduling and task allocation to ensure we were well prepared for the project execution.

Bugs

One of the most common forms of delays we had was due to our code not working. We often ran into all sorts of bugs. Whenever one of us encountered an unfixable bug, we would pass it to one of the other group members who had a better understanding of the code and normally they would be able to fix it. To minimise the impact on our scheduling we would spend a bit more time on the task during the week to ensure that this delay wouldn't affect any dependent tasks.

Working on the Project Together

Working on the documentation together was easy as Word document allows for multiple users to edit a document together. Whereas Android Studio has no multiuser functionality or a "share with others" function, and so it was impossible for all three of us to work on the application at one time. Also, as the files even in a .zip format were too big to share through discord or email attachments there was no quick method of sending the app back and forth. Our workaround to this was using a USB stick for transferring the app, which was fine when we were in class. But it did mean that if we needed to pass it over during the weekend then we had to travel to drop it off.

Lessons Learned

Thorough Planning is the Key to a Project

Having a strong project plan made the execution of the project so much easier as it helped us to define clear objectives, identify risks, break down tasks, and establish a schedule with milestones. Making a detailed WBS, Gantt Chart, and Risk Register ensured that everything was set out and all the team members knew what is required of them during the timeline and how they were able identify and mitigate possible risks. Establishing good management systems for the different aspects of project control; scheduling, resources, risks, change, and performance also played a big role in effectively managing our project.

Project Boundaries and Constraints Make a Project More Achievable

As the original idea of our project involved developing three intertwined deliverables, we decided pretty early on that this was going to be too much work for the little timeframe we had so we narrowed it down to one. From there we carefully went through and decided what functions the app will have and what it would not have, as well as any other limitations we might have to face. This ensured that we would have a solid scope and to avoid the project becoming something that we didn't have the skills or resources for.

Review Meetings are Essential

Review meetings were done to discuss what we did and what the next steps were. They were done on a rough weekly basis and helped to update all the team members on the progress and any delays or issues with a task. During these meeting we would also raise any issues with the Project Manager and if we needed to make a change to the scope, we would be able to use the change management system the project manager put in place to create a smooth integration.

Project Team Reflections

Team Member	Reflections
Bilee	I mostly did work on the Database backend work. One of the challenges I faced was an error within my SQL triggers as they would set each other off meaning that one of them had to be deleted it was an issue as it was one of the planned deliverables that we have within the goal for the app. Another issue was I had was dealing with time management I had 3 group assignments going at the same time and I found it quite difficult to find time to keep up with each one. This project also gave me a chance to practices and gain more skills with SQL which is something I very much enjoy doing.
Paige	I worked mostly on the documentation and the app wireframe design and the skeleton of the app. One of the challenges I faced the most was bugs in the code, which often caused long delays. I also struggled with using the Waterfall method in a group project, as it meant I had wait for certain tasks to be done before I could start my work. As it had worked for me in individual projects, I thought it would work but was difficult. It did end up being quite rewarding as my knowledge of app development grew and I enjoyed learning more about SQL and databases from my team member.
Alia	I worked mainly on the Gantt chart side of the app, making it work and connect with the database and the look of it as well as organising the project schedule. Leading the team was difficult for me as I am not a naturally assertive person. I prefer to lead from the back and not be solely responsible or looked at for tasks. The team was very capable, and the communication was smooth and easy between all of us making it easier. Though I was not able to get the Gantt chart of the app work properly, it was a good learning experience as I learnt about how to incorporate charts and sync them to the database even though it is not available in the app, if we had endless amount of time it would definitely be something I would like to add.

References