Project Plan V2

Team 16

27/08/2024

Table of Contents

Deliverable 1: The Project Plan

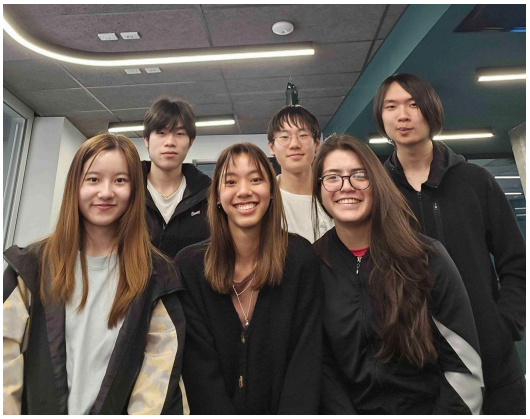
Project vision:

We aim to deliver a project management software for the middle management of companies, called the “Penny Project Partner,” we envision this to be an online project management tool that facilitates a product and sprint backlog, the creation and management of tasks and the ability to conduct multiple projects simultaneously. Our project is supportive of the ideas of cheap, timely and effective software, in stark contrast to that of our competitors.

Team name:

Scrum Force

Team photo:



Client Information:

Name: Deexita
Affiliation: Dex Studios
Contact details: deexita.goli@monash.edu

Team members:

1. Shaun Wong
2. Bernice Seto
3. Andrique Huang
4. Gwyneth Lee
5. Jaemin Park
6. Melissa Salazar Carrillo

Roles and responsibilities of members:

Scrum Master - Throughout the project we shall be rotating scrum masters each sprint. However, for the duration of the project, we shall have one fixed internal product owner who shall be in constant contact with the company liaison.

Risk Manager - The risk manager is responsible for identifying, assessing and mitigating potential risks that could impact the project’s success. They monitor risk factors throughout the project lifecycle, follow the contingency plans that we have outlined, and ensure that risks are effectively communicated to the team. While the Scrum Master is

responsible for facilitating Scrum practices within the development team, the risk manager will monitor risks that may impede the team's progress.

Internal Product Owner - When the Product Owner is not available, the internal or proxy product owner makes decisions on the requirements of the project on behalf of the product owner. They stand to settle any disputes related to the design requirements.

Our designated roles will be:
Sprint 1: Jaemin Park
Sprint 2: Shaun Wong
Sprint 3: Gwyneth Lee
Internal Product Owner: Andrique Huang
Risk Manager: Bernice Seto

Development Team
All members constitute the development team, and thereby are responsible for developing code towards each product increment. Development team members are held accountable for their own tasks, and are expected to deliver. The team can be divided into 2 main areas of specialisation:

Front end developer: Melissa Salazar Carrillo, Shaun Wong, Bernice Seto
Back end developer: Jaemin Park, Gwyneth Lee, Andrique Huang

Communications

As a team we will have determined 3 methods/tools for effective communication:

1. Use of discord chat functions to communicate with all team members
    - Discord chats are a instant communication tool that can be used to keep everyone up to date on a daily basis
2. Use of outlook emails to get in touch with product owners
    - Emails can be used to get in touch with product owners throughout the sprint, email frequency will be dependent on the amount of information necessary for the completion of the sprint.
3. Use of Google docs to complete collaborative deliverables of the assignment.
    - Google Docs participation is expected to occur 2 to 3 times a week, depending on the workload this may increase or decrease.

- Weekly face-to-face meetings will occur on Fridays during given time in classes, and before or after classes.

Team Process Model

Our team will follow a Scrum process model for our project, not all features of Scrum will be practical for our team. As such the following modified features suit our needs.

1. Sprints - The development of our web application will be divided into three sprint cycles, each lasting around 2 weeks. During each sprint, every team member will take responsibility for coding and developing a specific portion of the final product. Weekly meetings with our product owner will take place to address any issues.

2. Product backlog - We will collaborate with our Product Owner to create a comprehensive product backlog. This backlog will outline the features to be included in the final product, as well as future planned versions. This product

backlog will be reviewed regularly and updated to include newly identified requirements.

3. Sprint backlog/planning - With assistance/input from our product owner we will prioritise items from the product backlog to focus on during each sprint. This prioritisation will ensure we are addressing critical requirements in our final products. The team will also develop user stories to help define the features to be implemented in the sprint.

4. Stand up meetings - To accommodate the team members' schedules, standup meetings will be held on a weekly basis instead of daily. Each meeting will be longer in duration to compensate for the reduced frequency of meetings. Here, teammates will update each other on respective progresses as well as report on any anomalous circumstances.

5. Product backlog refinement - Alongside the review, we will be discussing what tasks/ requirements still need to be completed during the next sprint and consider any new requirements to the project.

6. Sprint review - After the conclusion of a sprint, the team and product owner will meet and review the product increment. The product owner may deliver feedback or outline further requirements, which the team can use to improve the product during the next sprint.

7. Sprint retrospective - Following the conclusion of each sprint, as a team we will discuss features implemented during the sprint, reviewing what worked and what still needs improvement during the next sprint.

Definition of Done:

Functional Requirements

- ☐ Functionality is complete as per the specifications agreed upon during sprint planning.

Testing

- ☐ User Acceptance Testing: Functionality is validated by the PO or stakeholder.

Documentation

- ☐ Code Documentation: Code is appropriately commented and documented.
- ☐ User Documentation: User documentation is provided if necessary.
- ☐ Backlog Update: Completed items are marked as "Done" in the product backlog and sprint backlog.

Code Review

- ☐ Code is reviewed by at least one other team member against the following matrix
    1. Code is well formatted and readable
        a. No commented out code
        b. Forms and buttons are functional
        c. Page load time is short (less than 1 second on average)
    2. Proper HTML structure is followed eg (H1, H2 ect)
    3. Javascript functions behave as expected

Deployment

- ☐ Able to be deployed on a chrome browser with expected functionality.

Product Owner Acceptance

- ☐ Functionality is reviewed and accepted by the PO during the Sprint Review.

Usability and Accessibility

- ☐ Can be made accessible by changing font sizes of webpages in accordance to preset levels (small, medium and large).

Team's weekly schedule:

Standup meetings will be held every Monday at 2:00PM - 2:30PM in a hybrid manner (members can attend online or in-person) on Discord. The Scrum Master will be responsible for scheduling the standup meeting on Google Calendar, as well as setting up the Discord meeting room and designated physical meeting point. They are also responsible for recording meeting minutes in the shared drive.

For the purpose of fostering collaboration and maintaining project alignment, additional team meetings will be scheduled on a weekly basis. The agenda of these meetings is to collaborate on the project together and outline tasks to be completed in members' own time. Meetings are held from 4:00PM - 5:00PM on Fridays in-person, and all team members are expected to attend in-person. Any absences must be outlined at least 24 hours before the next meeting.

The team will also make use of the weekly applied class on Fridays 5:00PM - 7:00PM to work on the project. The internal product owner can also liaison with the product owner. If the workload demands more time, project collaboration is permitted to extend past the applied session times at the discretion of the team members.

Finally, all team members are expected to commit a minimum of 3 hours per week of their personal time working on the project.

How team will allocate tasks to team members:

Task allocation will happen in accordance with each member's experience, technical skills or interest in the topic. Group discussions will allow each member to break down the tasks into smaller components, after which allocations can occur either through open discussion or individual volunteer efforts. This approach of breaking the tasks into smaller components facilitates overall understanding and ensures members have the knowledge and skills required to complete the task efficiently and to the highest standard.

For larger tasks, multiple members may collaborate on it to meet deadlines effectively. To assist with collaboration we will use project management tools like jira to assign tasks to team members and keep everyone updated on the progress of the project.

How team will keep track of progress on project:

To keep track of the progress of the project, team members can refer to the Jira scrum board, useful to monitor the state of tasks whether they are in progress, completed or not yet started.

How team will store and manage backlogs:

The chosen software to manage the team's backlogs, including the product backlog and sprint backlogs, is Jira. Jira is a free online project management tool developed by Atlassian, and is intended to be integrated with Agile and Scrum methodology.

Team members are able to add product backlog items using the backlog feature, and drag any chosen items into the sprint backlog so that they may be completed during the following sprint. Once the sprint has begun, sprint backlog items will appear on the Scrum board as required. The functionality of Jira makes it suitable for storing and managing the project backlogs.

How team will keep track of time spent of project tasks:

Time management will happen through the Jira story points, where tasks are filtered by sections To Do, In Progress and Done. Through this board, the team can check on the progress of remaining tasks to decide when it needs to be completed by and shift resources as required. A generic track of time spent on task projects will be kept as there will be weekly meetings and classes which equates to 3 hours of team collaboration per week.

Git policies:

Git will be used in order to keep track of and maintain the developing software by segmenting development into key stages. In essence, Git will be used to store the main chunk of the code and when to be developed on, team members will branch off from the main piece of code in order to work on their respective sections whereby they will merge the code back into the main section.

This not only allows for an accurate tracking of team member contribution, but is also the grounds by which effective damage control can be taken. Truly, should a team member unfortunately introduce bugs of various levels to the software we shall be able to effectively track and eliminate such bugs through the use of version control.

Deliverable 2: Analysis of Alternatives

Chosen Programming Language
In order to develop our software, our team has selected a programming language stack based off the following criteria: ● Familiarity ● Applicability to the project ● Learning curve The options we have identified for **frontend development** is 1. HTML, CSS, Javascript 2. Wix (no-code website builder) **Frontend Development (HTML, CSS, and JavaScript)** HTML (HyperText Markup Language), CSS (Cascading Style Sheets), and JavaScript are fundamental technologies used to construct and style web pages and applications.These languages will be used to create the structure, design, and interactive elements of the user interface. Many team members have an exceptional background in HTML and CSS, giving them a strong grasp of these technologies. Additionally, half of the team is already skilled in JavaScript, allowing us to develop a highly functional web application. Since its introduction in the early 1990s, HTML has become a standard in web development, alongside CSS and JavaScript (W3C, n.d.). These technologies have a wide variety of documents, with extensive resources and community research available online, allowing all team members to easily find solutions and enhance their knowledge. Moreover, there are numerous frameworks and libraries, such as Bootstrap for CSS and React for JavaScript, which can further boost efficiency and streamline the overall development process. Essentially, HTML, CSS, and JavaScript provide a solid and flexible foundation for frontend development, ideally suited for the project's needs. **Frontend Development (Wix)** Wix is a no-code website builder that has previously been utilised by several team members to quickly create websites. While it offers an extremely straightforward approach to web development, allowing users without any prior coding knowledge to build their web application, it does come with its own setbacks. For example,some of their more advanced features require payments to be fully utilised, ultimately restricting the project's capabilities when more complex functionalities aren't able to be obtained by team members. However, the overall learning curve for Wix is minimal, as being able to create the web application without needing any prior knowledge of coding languages makes it the most accessible option to all team members regardless of skill level. Nevertheless, this simplicity comes at a cost as—Wix lacks the normal flexibility and control provided by traditional coding methods like HTML, CSS, and JavaScript. Down the line. This limitation can be a major drawback should our project require a higher degree of customization and scalability (Wix, n.d.). Therefore, while Wix has its uses, particularly for extremely rapid development and ease

of use, its flaws shine through making it less than ideal for a project that demands robust, flexible, and highly customizable frontend development. Based on these reasons, Front -end Development through HTML, CSS, and JavaScript seem to align with our project requirements more.

The options we have identified for **backend development** is

1. Python
2. JavaScript

Backend Development (Python)
All members of the team have extensively studied and utilised Python for coding and stands to be the language we all have the highest experience with, developing a backend system due with our combined understanding of the language can allow the project to be completed in a timely manner. Python has a lot of versatility, it is a readable language which is helpful during development however it has some drawbacks regarding its applicability to the project, whilst members understands the language by itself, integrating it with other web development languages such as HTML and CSS poses a challenge. The learning curve of Python would be minimal due to all members being familiar with the language, however the learning curve to integrate python with other frontend development languages is steeper as members are unfamiliar with this process.

Backend Development (Javascript)
Regarding familiarity only half of the team members have used Javascript for previous projects. With varying degrees of familiarity this may result in a possible delay in the project plan, as inexperienced members require extra time and assistance to learn Javascript. Utilising Javascript for both front and back end tasks reduces the confusion and streamlines the development process by reducing the amount of languages used. Moreover, Javascript boasts a vast quantity of libraries and frameworks such as Express.js, React and Node.js which allows for simple integration between frontend and backend development. As for the learning curve, Javascript does have a harder syntax than Python (Radixweb, 2023) but with heavy collaboration amongst team members and by utilising the abundance of extensive research and documentation team members should also be able to easily learn Javascript to be applied to both front and backend tasks.

The options we have identified for **database management** is

1. SQL
2. MongoDB

Database Management (SQL)
SQL (Structured Query Language) is a language used to manipulate relational databases. Within the scope of the by project, SQL would be implemented for the purpose of storing data provided by the user, such as task details.

With several team members having experience with SQL in their prior studies, there is familiarity with the language. Furthermore, in comparison to other languages, SQL has a much easier learning curve, allowing inexperienced team members to adjust with ease (*How Long Does It Take to Learn SQL? - Multiverse*, n.d.).
SQL has been established since the 1970s (CodeCademy, n.d.) and has extensive documentation and support available online which team members can fully utilise, allowing for ease of use. There are several open-source relational database management systems available which support the SQL language such as MySQL, which can be used to

implement the backend.

Database Management (MongoDB)
MongoDB is a NoSQL database program that stores data in collections of documents as opposed to relational tables. It was first introduced in 2007, making it a relatively newer tool (*About Us - Our Story*, n.d.). Due to the way it stores data, MongoDB is an optimal choice for projects when flexibility and scalability are crucial.

However, the team's familiarity with MongoDB is minimal, which would result in a steeper learning curve as the project grows in complexity. While a flexible database is desirable, it is not essential when considering the size of the project. As it is a newer technology, its querying capabilities also lack robustness in comparison to SQL.

Chosen Option (HTML,CSS, Javascript, SQL)
Overall, we have selected to utilise SQL for the database managment, HTML,CSS focused on the frontend and Javascript for both front and backend. These languages were selected based on factors such as familiarity between group members, their applicability towards our project and the relative steepness of the language's learning curve to ensure that the project deadline will be complied with.

Chosen Targeted Platform

Our team wishes to pick a platform by which to launch our product management product. We have identified the options of launching on either a mobile app or web application based on the following criteria.

- Cost
- Ease of access
- Scalability
- Ease of development

Mobile app
First we shall analyse launching our platform on a mobile app. Regarding cost, there is only a one time upload fee of $25 for the Google play store, whilst for the App Store, a yearly fee of $99 applies (Patel, 2022). With ease of access, considering the absolute presence of smartphones in modern society, almost everyone should be able to interact with our app. Moving onto the issue of scalability, though for small scale operations, our team believes that there will be no issues given sound design decisions, due to the less powerful nature of smartphones, we believe that there may be performance issues should truly hefty updates be implemented. Finally, the team believes development on an app will be relatively challenging due to a lack of app development experience on the team.

Website
Second, we shall systematically deconstruct the viability of a web application. Indeed, the cost of a website can be broken down into two main sectors, that of the domain name and the case of web hosting. Whilst domain names can be obtained at no cost, web hosting can set one back about $3 per month (Leonard 2023). Moving on, the issue of ease is essentially non-existent due to the prevalence of the internet throughout the modern and developing worlds. Scalability should be excellent with a website, considering the ability to easily link different sites and the considerably more powerful systems websites can be run on. To close, our team is deftly experienced in web development and hence believe it shall

be no issue.

Ultimately, considering the:

- Cost
- Ease of access
- Scalability
- Ease of development

of apps and web applications, we can see a tilt towards the side of web applications, mainly due to the ease of development. Our team is much more comfortable bringing together a competent web application as opposed to that of an app. In addition to this, the projected costs of development and launch, seem to be considerably lower on websites. To close, considering how all other factors are tied up, we as a team have decided to go forward with developing this project management software on a website.

Target Browser/OS

All team members are using Microsoft Windows on our laptops, so we will be using Windows as our target operating system.

We have gathered the options for a potential browser to use and test our web application on, including Microsoft Edge, Google Chrome and Firefox. We will recommend the target browser from the following criteria:

- Compatibility with Windows
- Fast performance
- Familiarity with the browser

Microsoft Edge
Microsoft Edge is a web browser that is built-in to the Windows computers that all members use, making it highly compatible. It has a simple user interface which loads web pages quickly and efficiently. However, none of the development team members use Microsoft Edge as their main browser.

Google Chrome
With its user-friendly interface, which works seamlessly with Windows and macOS, Google Chrome is a highly compatible web browser for all members to use. Whilst Google Chrome has the potential to be one of the fastest web browsers, it requires good internet connection. The fast performance also trades for very high RAM consumption of the computer (Kingpin, 2024). Google Chrome is the primary browser all members in the development team use, therefore everyone is familiar with the browser.

Firefox
Firefox is an open source browser that supports operating systems like Windows, MacOS and Linux. It is known as the most secure and reliable browser that values user's data privacy. However, It lacks performance. Although Firefox has been constantly optimising its speed, there are still issues with slow browsing experience when extensions are added or many tabs are opened (Schmidt, 2023). Additionally, not all members are familiar with the browser.

Considering that the target browser must meet the criteria of compatibility with Windows, fast performance, and familiarity with the browser, we have decided to use Google Chrome. While all suggested browsers are compatible with Windows computers, the main reason is that our team is familiar with the browser. Additionally, using a high performance browser will prevent any frustration caused by long loading time or latency lag during development.

Deliverable 3: Risk register

ID	Risk	Estimated Likelihood	Estimated Severity	Estimated Impact	Monitoring Strategy	Mitigation Plan	Contingency Plan	Risk Owner
R1	Not enough time to complete designated tasks in a sprint	Medium	Medium	Medium	Utilise the project management software to track the completion of sprint backlog items in reference to the sprint deadline.	Complete a sprint planning session with all members prioritising essential tasks ensuring crucial items are completed first. Include possible buffer time at the end of each sprint to manage unexpected delays.	Reevaluate and prioritise the most critical tasks in collaboration with the PO. This may involve extending the sprint if necessary or breaking down tasks into smaller, more manageable pieces.	Scrum master
R2	Team members are inexperienced with chosen programming language	High	Medium	Medium	Conduct regular code reviews of code, improving code quality while allowing the opportunity for more inexperienced members of the development team to learn.	Dedicate a portion of the sprint to learning the programming language and reviewing/ work from other experienced members.	Organise training sessions or workshops to quickly bring everyone up to speed also, pairing less experienced developers with those who are more knowledgeable will provide mentoring opportunities and improve overall competency	Scrum master
R3	Programmin	High	Very Low	Very Low	Conduct regular pull	Test and review code	Implement strict code	Scrum

	g errors are merged into the main branch of the repository				request reviews before any branch or changes can be merged into the main branch. Ideally, another member must approve on git before merging.	before and after merging into the main branch to catch any potential errors early ensuring a working solution at the end.	review and pull request procedures. Automated testing and continuous integration tools will be employed to catch issues early in the development process.	master
R4	Project schedule is not definitive enough	Medium	Medium	Medium	Adhere to weekly meetings, where project timeline will be reviewed to identify any indefinite or unclear deadlines.	Conduct a detailed planning session during each sprint, making use of project management tools to manage schedules.	Conduct regular check-ins and reviews to adjust the timeline as needed. Tasks will be broken down into more detailed milestones with clearer deadlines to ensure progress is measurable. The team will work closely with stakeholders to clarify expectations and solidify deadlines.	Scrum Master
R5	Project documents are lost or destroyed	Very Low	Very High	Low	Continuously monitor the shared Google drive to ensure documents are still there.	Continual backups made to version control tools like git can ensure only some sections of a document are lost compared to the whole document.	Regular backups will be made to a secure cloud storage service, ensuring that the latest versions of all documents are preserved (use git or github to ensure this).	Scrum Master
R6	Technical skill difficulty in translating	Medium	High	Medium	Team members will raise any technical difficulties in implementing design requirements to the	Manage client expectation and understand the limits of team members.	The design will be broken down into smaller, more achievable	Product Owner

	the design into a functional product.				team through weekly meetings, as well as setting the status of a task to “blocked” on the scrum board.		components, allowing the team to tackle the development in more manageable steps. Additional time will also be allocated for prototyping and iterative development to refine the design and ensure it can be implemented effectively.	
R7	Multiple people unintentionally working on the same sprint task	Medium	Low	Low	During weekly meetings, team members will offer a brief overview of any progress made and future plans, allowing the team to identify any overlaps.	Attend weekly meetings, which will reduce chances of multiple people accidentally working on the same task.	Tasks and responsibilities will be clearly assigned at the start of each sprint, with regular updates provided to the team. Jira will be used to allow team members to claim tasks and update their status, ensuring visibility for all. If overlaps do occur, tasks will be reassigned as necessary to ensure efficient progress.	Scrum Master
R8	Team members unable to deliver	Medium	Medium	Medium	During team meetings, members should raise any issues with meeting agreed deadlines.	Ensuring adequate buffer time at the end of each sprint can allow other team members to	Redistribute the workload among other team members who have the capacity to	Scrum master

	individually allocated tasks within the agreed timeframe.				Members will also continuously monitor the sprint board for any members that are struggling to complete items or making slow progress.	assist in completing the task within the given timeframe.	assist. The team will also evaluate the remaining tasks and adjust priorities or deadlines if necessary. Regular check-ins will be conducted to monitor progress and provide timely support to ensure tasks are completed on schedule.	
R9	Team unable to compromise on a final design feature	Low	High	Low	Team members are required to raise any major design decisions to the team during meetings before beginning implementation.	To reduce the impact of this risk, important decisions should be discussed as a group and with the product owner, gathering feedback can help determine which design feature resonates better with the intended target user.	Arrange a meeting with the PO to help guide the team towards a consensus. The team will explore alternative solutions, weighing the pros and cons of each option, and consider the impact on the overall project goals.	Product Owner
R10	Project requirements are unclear/ lacking, resulting in a solution that does not address	Low	Medium	Low	Weekly meetings will also serve as requirement reviews. Team members will outline what requirements are being met and how when providing a briefing on progress made.	Adopting an iterative requirement gathering approach from the agile model ensures that requirements can be refined and elaborated on throughout the duration of the project.	Engage consultations with the client and stakeholders to clarify expectations and gather more detailed requirements. A thorough review of the project scope and objectives will be	Product Owner

	the clients problems.						conducted to ensure all critical elements are captured. Continuous feedback loops will be established with the client to validate that the solution aligns with their needs as the project progresses.	
R11	Incompatible installed packages or libraries between team members' systems	Low	Low	Low	Regularly document any packages or external resources used in the project in the shared drive.	To avoid confusion from the beginning, discussions regarding development will occur as a team. Outlining any packages, and external resources to be used.	Standardise the development environment by documenting and sharing a list of required packages and their specific versions. Regular updates and checks will be conducted to ensure compatibility.	Scrum master
R12	Power outage causing loss of data	Very Low	Medium	Low	Development team should monitor backup or save status of all project artefacts in the storage system, including google drive, Gitlab and Jira	Making use of version control tools such as git to reduce the impact of lost data.	Implement automatic save and backup systems across all workstations, ensuring that recent work is preserved even in the case of an unexpected outage. The team will also establish a protocol for regularly committing changes to the version control	Risk manager

							system, minimising the amount of work lost during an outage.	
R13	Group members sustain physical injuries	Medium	Medium	Medium	Any physical injuries that would impede the progress of the project should be raised to the team through the primary communication channel as soon as possible.	Have a flexible schedule to ensure other members can cover for whoever sustained injury.	Tasks will be reassigned or redistributed to other team members to ensure that project timelines are not significantly affected. The team will remain flexible and compassionate, adjusting workloads and deadlines as necessary while the affected member recovers. Communication will be maintained to keep the injured member informed and engaged in the project as much as possible.	Risk manager
R14	Stolen/lost hardware	Low	High	Low	Any stolen or lost devices that would impede the progress of the project should be raised to the team through the primary communication channel as soon as possible.	Have security measures in place which protects devices. Consider attaching a tracking device to hardware such as air tags which can be used to locate lost devices.	Team members will ensure that all critical data is backed up to a secure cloud storage solution, minimising the impact of hardware loss.	Risk manager

R15	Unresponsive team members or not showing up to meetings without prior notice	Low	High	Medium	Attendance for team meetings will be continuously monitored and documented. Any absences without prior notice will be raised to the absentee immediately.	Set clear expectations ensuring all members understand the importance of attending team meetings.	The scrum Master will reach out directly to understand the situation and offer support if needed. The team will reassess task allocations and responsibilities to ensure that project progress is not hindered by the absence of the unresponsive member. Regular attendance and communication expectations will be reiterated to all team members to prevent similar issues in the future.	Scrum Master
R16	Group members timetables not lining up	High	Medium	High	When2meets will be utilised whenever meetings need to be scheduled, aside from weekly meetings. This allows the team to identify any scheduling issues.	Consider alternative meetup methods such as over zoom or consider recording meetings for members who cannot attend in real time.	Establish a shared calendar to identify overlapping availability and schedule meetings or collaborative sessions during those windows. Asynchronous communication tools (Discord) will be utilised to ensure that all members can contribute and stay	Scrum master

							informed even when they are not available simultaneously. Tasks that require collaboration will be planned in advance, and clear deadlines will be set to accommodate varying schedules.	
R17	Arguments between group members leading to fall out	Very Low	Medium	Low	All team meetings will also serve as personal check-ins. Members can address any disagreements or concerns.	Establish clear guidelines for respectful and honest communication to resolve any issues.	Facilitate a calm and open discussion to address the issues at hand. An impartial team member will guide the conversation, ensuring that all parties have a chance to voice their concerns. The team will focus on finding a mutually agreeable resolution, emphasising the importance of maintaining professionalism and collaboration.	Risk manager
R18	Group members ignore and neglect their duties	Low	Medium	Low	Attendance will be taken for team meetings. Furthermore, all members are required to share any progress made, potentially	Making use of project management tools such as Jira to provide reminders for upcoming tasks.	The Scrum Master will have a private conversation with the individual(s) involved to understand the reasons behind the neglect and	Scrum master

					identifying any signs of neglectful behaviour.		offer support or solutions. Clear expectations and deadlines will be reinforced, and the team will consider reassigning tasks or providing additional resources to help the member get back on track. Regular progress checks will be instituted to ensure that all members are fulfilling their responsibilities.	
R19	Product Owner being unable to express what they actually want due to a lack of technical skill/knowledge	Medium	High	High	Negotiation with the product owner is performed as a team, so that any difficulties in communication have a higher likelihood of being noticed.	Conduct brainstorming sessions and offer possible solutions for the product owner to review and gain some insights on the technical aspect of the solution.	The team will ask clarifying questions, providing examples or prototypes, and breaking down technical concepts into more understandable terms. Regular meetings will be held to ensure alignment, and the team may involve a technical liaison who can translate the Product Owner's vision into actionable technical requirements.	Product owner
R2	Team	Low	Medium	Low	Periodically review	Consider having an	The Scrum Master will	Scrum

0	members too controlling and dictatorial				decision dynamics within the team to identify any over or under assertive teammates, and whose ideas get pushed more often.	intervention/ open discussion about the issue.	speak privately with the individual to discuss the impact of their behaviour on the team's dynamics and performance. The importance of collaboration and shared decision-making will be emphasised, and the team may establish more formal guidelines for communication and task delegation to ensure a balanced and inclusive environment.	master
R21	Sickness within the team	Medium	Medium	Low	Any sickness contracted by team members or their close contacts resulting in quarantine restrictions should be raised to the team through the primary communication channel as soon as possible.	Have a flexible schedule to ensure other members can cover for whoever is sick.	The affected members will be encouraged to prioritise their health and take the necessary time off. The team will redistribute the sick member's tasks among the remaining members to ensure that project deadlines are not compromised. If the sickness is expected to be prolonged, the team may seek temporary external assistance or adjust the project	Risk Manager

							timeline to accommodate the absence. Regular updates will be provided to the affected members to keep them informed and engaged with the project during their recovery.	
R22	Team members steal each others credit within project	Low	Medium	Medium	Team members should all be continuously monitoring the project management tool to ensure that any tasks are properly assigned to them.	Use of project management tools to track team members' contribution log this method prevents credit theft.	A private discussion will be held with the individuals involved to clarify contributions and ensure proper acknowledgment is given. The team will establish clear guidelines for credit and recognition, emphasising the importance of honesty and fairness. Acknowledging contributions publicly in team meetings or project documentation will help prevent similar issues in the future.	Risk manager
R23	Team members experience hardware	Very low	High	Low	Team members should regularly check their hardware for any potential issues. Any	Consistently using version control tools such as git can reduce the amount of work lost	Ensure that backup systems and devices are available to minimise downtime.	Risk manager

	malfunctions				confirmed malfunctions should be raised to the team through the primary communication channel as soon as possible.	in the event of hardware malfunctions. In the event of a short term hardware malfunction for instance 1 week whilst a member's computer is being fixed, Monash has temporary computers that can be rented for a short duration.	Critical data and work will be regularly backed up to cloud storage to prevent data loss. The team will also maintain a list of recommended hardware and software setups to reduce the likelihood of compatibility issues.	
R24	Team members unable to work due to personal family/social issues	Low	High	Low	Any family or social issues should be raised to the team through the primary communication channel when members are comfortable, but ideally as soon as possible. The team will also regularly check in on other members during meetings as an opportunity to raise such issues.	Have a flexible schedule to ensure other members can cover for whoever is unable to work due to personal issues. Planning in buffer time can ensure the other team members have enough time to pick up the slack and complete any outstanding tasks.	Approach the situation with empathy and flexibility. The members will be encouraged to communicate their needs, and the team will adjust deadlines or reassign tasks to accommodate their circumstances. The team may also offer the member the option to work asynchronously or take a temporary leave if necessary.	Risk manager
R25	Mental health issues amongst team members	High	Medium	Medium	Any mental health issues should be raised to the team through the primary communication channel when members are comfortable, but	Encouraging team members to seek support from Monash wellbeing support can accommodate members experiencing mental	The team will encourage open communication about mental health and may provide resources or referrals to professional	Risk manager

					ideally as soon as possible. The team will also regularly check in on other members' mental wellbeing during meetings.	health issues. Planning in buffer time can ensure the other team members have enough time to pick up the slack and complete any outstanding tasks.	support if needed. Adjustments to workload, deadlines, and expectations will be made to reduce stress and accommodate the individual's needs. The team will also promote a culture of self-care and mutual support, ensuring that members feel comfortable seeking help when needed.	

References

About Us - Our Story. (n.d.). MongoDB. https://www.mongodb.com/company

CodeCademy. (n.d.). *What Is a Relational Database Management System?* Codecademy. https://www.codecademy.com/article/what-is-rdbms-sql

Kingpin (n.d.). *Advantages and Disadvantages of Google Chrome.* Kingpin. https://kingpinbrowser.com/blog/advantages-disadvantages-chrome/

Leonard, K. (2023, March 6). *How Much Does A Website Cost? (2022 Guide)* – Forbes Advisor. Forbes. https://www.forbes.com/advisor/business/software/how-much-does-a-website-cost/

Multiverse. (n.d.). *How Long Does It Take To Learn SQL?* Multiverse. https://www.multiverse.io/en-US/blog/learn-SQL

Patel, B. (2022). *How Much Does It Cost to Put an App on the App Store?*. Space Technologies. https://www.spaceotechnologies.com/blog/cost-to-put-app-on-app-store/

Radixweb. (2023). *Python vs JavaScript: Which language is right for your next project?* Radixweb. https://radixweb.com/blog/python-vs-javascript

Schmidt R. (2023, December), *Firefox exceptionally slow compared with Chrome.* Mozilla. https://discourse.mozilla.org/t/firefox-exceptionally-slow-compared-with-chrome/125319

World Wide Web Consortium (W3C). (n.d.). *HTML & CSS standards*. https://www.w3.org/standards/webdesign/htmlcss