Taskforce

In-house Project Management Software

Project Inception

Version 2.0

Written by:
Brian Nge Jing Hong
Muhammad Ibrahim bin Mohd Yusni
Chua Xian Loong
Lucas Wee
Koe Rui En
Diana Wijaya

Taskforce

Contents

1. Introduction and Vision

1.1 Introduction

This project management software is a powerful, customizable, and efficient software that would empower teams to plan, track, and collaborate on different projects with ease. This streamlines the project lifecycle, encourages effective communication, and boosts productivity while ensuring to adhere to the highest standards of quality.

This comprehensive, cost-effective, and user-friendly software solution would meet the specific needs of large or small corporate teams while also providing a reliable platform for teams to manage all future projects seamlessly.

1.2 Project Vision

For corporate project teams that are struggling to manage multiple projects simultaneously to have a powerful, customizable, and efficient one stop centre to manage and access all projects with ease. Unlike Trello, Taskforce is able to be as customisable as possible for its clients.

2. Project Information

2.1 Meet the Team

Introducing the Taskforce team who are behind the project management software that provides solutions to corporate project teams.

Diagram 2.1: Ibrahim, Xian Loong, Brian, Lucas, Rui En, Diana

2.2 Team Contact

Should there be any urgent enquiries or requests, please contact the respective project team members using the contact details below.

Name	Email Address	Contact Number
Brian Nge Jing Hong	bnge0001@student.monash.edu	+60 11-1092 8932
Muhammad Ibrahim bin Mohd Yusni	mmoh0156@student.monash.edu	+60 19-662 1128
Chua Xian Loong	xchu0015@student.monash.edu	+60 11-563 20025
Lucas Wee	jwee0015@student.monash.edu	+60 13-613 4849
Koe Rui En	rkoe0003@student.monash.edu	+60 10-2978665
Diana Wijaya	dwij0020@student.monash.edu	+60 14-635 4650

Table 2.1: Team Contacts

2.3 Roles and Responsibilities

These are the roles and responsibilities of each project member. Should there be any enquiries regarding a specific section of the project, please feel free to contact the project member accordingly.

Name	Role	Responsibilities
Brian Nge Jing Hong	Scrum Master and Assistant Programmer	● Facilitate standups meetings ● Keep track of members' progress and ensure that deadlines are adhered to. ● Participate in sprint reviews and record feedback from sprints. ● Assist Head Programmer with relevant coding tasks.
Muhammad Ibrahim bin Mohd Yusni	Product Owner and Front End Developer	● Acquire the requirements for the project from the client by acting as the liaison between project team and client. ● Manage the product backlog. ● Evaluate feedback at each sprint iteration. ● Establish and code the front end for the website.
Chua Xian Loong	Head Programmer	● Coding out the project management software and debugging for the problems that arise. ● Designing and testing computer structures.
Lucas Wee	Database Programmer	● Create new database objects in languages such as SQL ● Initiate database setup and testing. ● Analyse data for correctness to ensure integrity, report errors to QA. ● Create and maintain tables/ Unix scripts as needed for new development. ● Perform data extraction, transformation and loading between systems using tools such as SQL tools.

Koe Rui En	Technical Writer and Tester	● Produce high-quality documentation for the code that adheres to applicable standards while making sure that context is appropriate for future audiences. ● Record meeting minutes for the standup meetings as well as the client meeting with the product owner.
Diana Wijaya	UI/UX Designer and Front-End Developer	● Illustrate design ideas using tools such as storyboards, process flows and sitemaps ● Design graphic user interface elements, like menus, tabs and widgets ● Build page navigation buttons and search fields ● Develop UI for the website ● Adhere to style standards on fonts, colours and images described by the client.

Table 2.2: Roles and Responsibilities

2.4 Process model

The process model that we will be using for this project is Scrum. The Scrum approach is an Agile framework that is often used for managing complex projects. This process model uses an iterative and incremental approach to divide work into "sprints".

There are three main roles in this process model, Product Owner, Scrum Master and the Scrum Team. The Product Owner is responsible for organising the work for a complex problem into the product backlog. The Scrum Master is responsible for ensuring that the Scrum Process is strictly followed and is in charge of turning high quality work into value that addresses the product backlog during each sprint. Lastly, the Scrum Team is a self organising team who are responsible for completing all the given tasks within each sprint and make sure that there is a shippable product each iteration.

The scrum artefacts in this project surrounds the product backlog. This includes all the requirements, user stories, testing, and bug fixes. The product owner will be managing and prioritising each product backlog based on its value. This allows the team to do sprint planning and sprints more efficiently, enhancing the project outcome.

In this project, we will have a total of 3 sprints, until product completion. There will be a period of 3 weeks for the client to list out all their requirements. Then, each sprint is roughly 2 weeks long. We will also rotate all the main roles and responsibilities each sprint, to make sure that all team members get to experience what each role demands for.

From our understanding of the challenges posed by each member's various commitments and time constraints, our team will be holding stand-up scrum meetings once a week, for approximately 1 hour each time, serving as a crucial portion of the project. We aim to maximise our efficiency and effectively discuss all the necessary contents during the meeting every week. These meetings provide us with an opportunity to share updates on individual progress, address any issue and discuss potential solutions for any problem arised. By conducting these meetings, we can ensure that everyone is informed about the project's progression, and prevent any misalignments.

All the scrum practices will be done in each sprint, including sprint reviews and sprint retrospectives by the end of each sprint.

2.5 Definition of Done

Our team will ensure that every team member understands the definition of done which will be a clear and agreed upon set of criteria that must be met for a product backlog item. This will ensure that everyone is on the same page when we are working on the project. When we receive a new task or user story, we will break it down into smaller sub-tasks and consider all the various aspects involved such as coding, testing, documentation and design. We will also request peer and code review for any code changes that our team implements as this would help to catch errors and also promote code quality. It also ensures that team members are on the same page when working towards the definition of done. We will also go through extensive testing and demo of the code to make sure that the product works as intended and follows the definition of done that was agreed upon.

Here are the checklist of requirements:

Requirement Set 1

- Product backlog
- User have to be able to create new tasks and be able to enter the task
- Users have to be able to add names to the newly created tasks
- Users have to be able to add user stories to the tasks
- Users also have to be able to add description to the tasks
- Users have to be able to assign story points to the tasks (how much effort is needed in order to complete the task)
- Users have to be able to tag the task and be able to filter tasks by their tags and be able to assign multiple tags onto the task
- Task status (how done the tasks are) - [done, in-progress, closed]
- Tasks should be able to be filtered by priority as well - [low, medium, important, urgent] (urgents needs to be red, others to be decided)
- A drop down list of members to choose from for the tasks and be able to assign one member max to each task
- Users also have to be able to view the name, tag, priority, story point on the card view and be able to edit the status, description, story when tapped into the card.

Requirement Set 2

- A new sprint view
- A new button to create new sprint and when the button is clicked, a pop out to choose the name for the sprint view and start date and end date (the date has to be a calendar pop out)
- When the sprint is pressed, be able to see all the sprints listed out.
- Can change the sprint view into a list or cart
- Be able to save the sprint and go back to the sprint view or click manage view

- Be able to start the sprint from the initial sprint view
- Must be able to see the product backlog and sprint together on one singular screen (split screen) and be able to drag and drop from the product backlog window to the sprint window
- Must be able to save it after dragging and dropping.
- When go into start sprint must be able to see all the tasks
- If start sprint is clicked the sprint, override start date
- Inside the sprint board must have: not started, in progress, completed
- The task can be started by dragging into the sprint or manually set it to the above state
- Timelog, allow manage the start time and end time to see how much time is logged
- When sprint is completed can end sprint
- Be able to see the velocity chart (expected velocity vs actual velocity)

Requirement set 3

- Admin view for project owner.
- User login and sign up page, using username and password.
- Name, email address, and password will be asked during sign up.
- Setting a profile page allows users to access their name and email address.
- Users will be able to change and save the account password.
- Users must be able to add and remove team members.
- There will be no maximum number of members.
- View for team member contributions.
- Each member's name and email address will be shown.
- A contribution bar chart of individual members will be shown when the name in the list is pressed.
- A dashboard icon will be given.
- When the dashboard icon is clicked, a new view will be displayed, showing the overall contribution in the team and being able to remove team members that's working on the tasks.
- The user will be able to choose the time period for contribution view, by using the start and end date.
- The visual representations shown as part of the burndown chart will be a bar chart for the overall hours worked from all team members, or a bar chart for total hours of work in the team

2.6 Task Allocation

Our team will allocate tasks accordingly by conducting regular sprint planning meetings to determine tasks that are to be completed in each sprint where the project requirements would be broken down into smaller user stories and tasks. We will allocate tasks based on voluntary assignment of tasks according to Scrum practice. Team members are allowed to express their preference for specific tasks or user stories during sprint planning. If multiple team members express the same interest in a particular task, skill matching will be used to determine who would be assigned with that task. Team members are encouraged to volunteer for tasks that align with their expertise and interests. A Kanban board will be used

to organise the actionable tasks and their progress through various stages of the workflow. Tasks will be placed in the "To Do" column of the Kanban board and members can take tasks from that column.

In the spirit of Agile values and principles, our team will trust each other in allocating tasks to ourselves that best suit our abilities and interests. As stated in the "Individuals and Interactions" core value in Agile Manifesto, our team will communicate frequently to understand each other's tasks to foster collaboration and transparency. When required, team members will help one other when difficulty arises during the tasks or when reallocation of tasks is necessary.

This table displays the prior technical experience that each team member has as well as their technical interest. This table would assist us in the decision making process if there is conflict in task allocation.

Name	Course	Name of units taken so far	Computer languages known	Technical interests
Brian Nge Jing Hong	Bachelor of Software Engineering (Honours)	ENG1011,ENG1012,ENG1013,ENG1014, FIT1051, FIT1047, MAT1830, FIT2099, FIT2093, FIT2085	Python, Java	Back-end
Muhammad Ibrahim bin Mohd Yusni	Bachelor of Software Engineering (Honours)	ENG1005, ENG1011, ENG1012, ENG1013, ENG1014, MEC2404, PHS1002, FIT1051, FIT2099, MAT1830, ENG2801, FIT2085	Python, Java	Front-end
Chua Xian Loong	Bachelor of Software Engineering (Honours)	ENG1011, ENG1012, ENG1013, ENG1014,ENG 1005, ENG2801, FIT2085, FIT 1047, FIT 1051,FIT 2099, FIT 2081, MAT1830	Python, Java	Back-end
Lucas Wee	Bachelor of Software Engineering (Honours)	ENG1011, ENG1012, ENG1013, ENG1014 FIT1051, FIT1047, MAT1830, FIT2099, ENG1090, FIT2085	Python, Java	Back-end, database
Koe Rui En	Bachelor of Software Engineering (Honours)	ENG1011, ENG1012, ENG1013, ENG1014, ENG1005, ENG1090, FIT1051, ENG 2801	Python, Java	Back-end
Diana Wijaya	Bachelor of Computer Science (Advanced Computer Science)	FIT1045, FIT1047, FIT1043, FIT1008, FIT1055, FIT1051, MAT1830, MAT1841	Python, Java, Javascript, Typescript,	UI/UX

Table 2.3: Technical Experience and Interests

This table displays the confidence level that each team member has in the specific programming language. The scoring is given out of 5. This table would also assist us in the decision making process if there is conflict in task allocation.

Programming Language	Brian Nge Jing Hong	Muhammad Ibrahim bin Mohd Yusni	Chua Xian Loong	Lucas Wee	Koe Rui En	Diana Wijaya
Python	3	3	3	3	2	3
Java	2	2	2	2	2	2
JavaScript	1	-	-	-	-	2
HTML	1	-	-	-	-	1
CSS	1	-	-	-	-	1

Table 2.4: Programming Language Confidence Level

2.7 Progress Tracker

Our team will develop several rules to keep track of our project progress and tasks. This is to ensure our project is always on the right track, meets our client's requirements and can be completed by the assigned deadline.

The progress tracking rules for our team are stated as follows:
1. Weekly standup meetings during tutorial classes and if necessary outside class from 2pm to 3pm Thursday.
2. Establish a group communication platform, such as WhatsApp and Discord, for all members to share their weekly progress updates.
3. Each team member should complete their assigned tasks before every next workshop class.
4. Utilise Google Calendar to set up reminders.
5. Create task checklists or Kanban boards using Trello.
6. Have the product owner host a meeting with the client every Thursday 11am to 2pm to review whether their requirements are being met and to collect feedback as soon as possible.

2.8 Backlogs

We will be managing the backlogs using Trello. By leveraging Trello's customizable boards, we will be able to prioritise backlog items for upcoming sprints. This will include all the user stories and tasks, with its details. Using this method, we can easily transition tasks from "To do" to "In progress" to "Done", providing real-time visibility to each task's importance. This method allows us to adapt to shifting requirements, keeping our progress aligned with the main goal of the project. This will also allow us to collaborate more efficiently and manage tasks better, in response to evolving project needs.

In this process, we are able to add cards onto each compartment. Each card can be named, and a brief description of the task can be added. Other than that, each member can assign themselves as the person in charge for each respective task. We can also label each task based on its main characteristics such as UI/UX, backend programming or frontend programming. After each task is proceeded to the next stage, it can be dragged to the next stage.

The following links are the GitLab repository, google drive, google calendar, Trello and time tracking links that the team would be utilising to make project management easier for the team and allow collaboration between team members with ease.

GitLab Repository Link:
https://git.infotech.monash.edu/FIT2101-S2-2023/MA_Thursday11am_Team1/project

Google Drive Link:
https://drive.google.com/drive/folders/1lZmrqR8rKx2S2bhtvkiSRo1fzd6bMTAc?usp=drive_link

Google Calendar Link:
https://calendar.google.com/calendar/u/1?cid=Y18wZThiMjFmNTcwMzA5YzdjYTQ0OTkxNThiMzExYjhlOGIxMDlkNzVhYWYzMzhiOGUzZDg2ZWJiMGNiMzhjMGY3QGdyb3VwLmNhbGVuZGFyLmdvb2dsZS5jb20

Trello Link:
https://trello.com/invite/b/XmURix1g/ATTIe0d0e2825b610f0d63a9d2b296856c6cCE19854A/project-1

Time Tracker Link:
https://app.timecamp.com/invited/by/mmoh0156/1aqzn/gjv1/s0njjf/7iykp1lv

2.9 Time Tracker

The key to a successful project is efficient time tracking. Our project incorporates diligent time tracking through the usage of a power-up feature (extension) in Trello itself, named "Track Time". This tool can help us in capturing the allocation of each team member's time spent on various parts of the project. By logging in our time records, we will all be able to gain insights on resource distribution and project progression. These capabilities also grant us insights on each member's effort and time put into this project, allowing us to further analyse workflows. Through this method, we will be measuring our productivity and dedication, allowing us to gauge each team member's contribution to the project's success.

Therefore, In our Trello board, we must all install the "Track Time" power-up, in order to track the time of each task's completion. When a member wants to start a task, they can manually start the timer, and end it upon completion. The total time would be added up showing the total time spent each time.

2.10 Git Policy

Our team will be using GitLab as our central repository platform. A 'main' branch will be used to store the base deployable software, there will then be a 'develop' branch that branches off 'main' for further development. If necessary, there will also be a 'features' branch that branches off 'develop' to implement new features, together with a 'bug-fixes' branch to fix any bugs. Commit messages should begin with a brief summary, followed by a detailed description if necessary. Before merging any new features or bug fixes, a merge request with a descriptive title should be opened for review. Preferably two team members should review and approve merge requests, by ensuring all tests are passed and merge conflicts are resolved. Every team member should participate in code review and provide constructive feedback. Files pushed to the repository should be well-documented.

3. Analysis of Alternatives

Alternative analysis is the evaluation of different paths that we can take to achieve a project goal or specific project management objective. In this section, we evaluate the different programming language options that we can use to program the software and the different platforms we can use to deploy the software such as on a web application or a mobile application as well as the database platform to host the website.

In this section, we evaluate the following programming languages and platforms:

Programming languages
- Python
- Java
- JavaScript
- HTML and CSS

Platforms
- Web Application
- Mobile Application

Database Management
- Firebase
- Access

Effective Testing
-

3.1 Terms of reference

The solution for a project management tool should be cost effective as there is limited budget in financing for the project. This means that the solution should be less labour intensive and time consuming as they are equivalent to loss of finance. The solution should

also be user friendly and easily accessible for teams. This is to allow corporate teams to be able to utilise the software with ease.

The solution for choosing the right programming language is that it should have large amounts of resources and comprehensive documentation so that the project can be easily built upon. The programming languages should also be accessible to developers of various skill levels, with straightforward syntax and strong community support to ensure that help is readily available. Furthermore, the language must enable development across different devices and browsers without significant adjustments so the team can work collaboratively effectively.

The solution for the platform should allow cross platform accessibility as teams would be working from different devices. The solution should not require any prior installation or configuration for the application. The solution should also maintain its functionality and design across platforms. Besides, the application should allow automatic updates from the applications server.

The solution for the database management should be able to accommodate real time synchronisation so that multiple users are able to see changes almost instantaneously. It should also be scalable to be deployed on an industrial scale. The system should also have reasonable security measures to ensure the database management platform meets the data protection requirements.

3.2 Python Programming Language

Python is one of the most popular programming languages that is greatly used by the world. Python is known for it being user friendly and its readability. It is considered a general-purpose language, so it can be used in many aspects, including data science or software and web development. In this case, python would be a great programming language to be used in this project.

Python is a programming language that is simple to read and write. This way, it will be more friendly to both beginners and advanced programmers in the team. Python also comes with an extensive amount of existing libraries that users can import and use. This will be highly beneficial to our project because it can save us a lot of time and increase our productivity when dealing with complex tasks. Other than that, python also works on all platforms and systems. This way, when written on one computer system, it can be easily edited and used on other systems, allowing everyone in the team to access it.

While Python offers numerous advantages, it also has its limitations. Python may not be the best choice if the project is considerably complex and big. Since Python is an interpreted language, it is slower compared to other languages when compiled. This does not suit applications or web browsers that require heavy parallel processing. The overall performance of the application will automatically decrease if Python is used in a high-speed and intensive demanding application.

Taskforce

3.3 Java Programming Language

Java is commonly used for desktop applications development including graphical user interfaces and it can be used to create web development as well as dynamic and interactive web applications. Furthermore java is also used for creating android applications using the Android SDK.

Java has a lot of advantages like performance, java can compute more computationally intensive tasks and applications as java statically typed nature can contribute to its performance. Furthermore, java's static typing enforces strict type checking at compile time, which can catch errors early and improve code robustness and reduce errors and provide consistency making it a good choice for building large scale enterprise applications.

While java has a lot of advantages, it has really strict syntax requirements which means it has a big learning curve to it and errors might occur more often and java requires more time to code the boiler plates and it is relatively hard to maintain the codebase as everyone in the team might have different understanding of language meaning not everyone will be able to understand each other's code or what is happening with the code. This can lead to longer and more complex code compared to languages like python. Furthermore, java has a slower startup time compared to languages like python due to the need for compilation and initialization of the java virtual machine.

3.4 JavaScript Programming Language

JavaScript is among the most versatile and popular programming languages available to date. Initially used as a programming language designed to add interactivity to websites, it has now evolved to become a multi-purpose language as it can run both frontend and backend systems of advanced applications through platforms like Node.js.

Using JavaScript as a programming language, would offer advantages like its universality across platforms, as the same code can often be used on mobile devices, desktops and Internet of Things (IoT) devices with minimal modifications. In addition, JavaScript also offers a wide range of frameworks and libraries like React, Vue.js, Angular and Node.js which can speed up the backend and frontend development process, together with a vast community that provides reliable support, continuous updates and an abundance of resources. Through this, a singular programming language can be used to control our tech stack helping with the debugging process.

Despite the multiple advantages, JavaScript does come with its disadvantages. One of which is that its single-threaded nature can become a limitation for CPU-intensive tasks. Though there are potential workarounds, like child processes in Node.js, JavaScript's asynchronous behaviour can frequently lead to complex and nested callbacks, complicating code maintenance and readability. In addition, the absence of strict typing syntax can sometimes lead to runtime errors, which are issues that statically typed languages would detect during the compilation process. Its interpreted nature also means that it will frequently lag behind the performance of compiled languages, especially for CPU-bound tasks. Lastly,

JavaScript's asynchronous programming structure can be quite confusing to newcomers due to its learning curve.

3.5 HTML and CSS Programming Language

HTML and CSS have been the standard for building websites for decades as they are supported by all modern web browsers and therefore can be supported on almost all devices that are connected to the internet. For example, major desktop platforms like Windows, macOS and mobile platforms like Android support HTML and CSS.

HTML and CSS will enable us to create websites easily and quickly as they provide a simple and structured syntax that allows websites to organise and present content on the web. HTML can be used to define headings, paragraphs, lists, links, images, and other elements to be presented on a webpage. While CSS can be used to style the website and its elements in a consistent manner. This increases the readability and usability of the website. Since we are collaborating together, HTML provides modularity; enabling easier maintenance and updates of our website. Changes to the codebase can be made without affecting the overall design. It will also be easy for us to create prototypes and mock-ups of websites which fits the Agile methodology of focusing on collaboration, adaptability, and delivering incremental improvements.

Furthermore, writing HTML and CSS code can be done on any text editor like the built-in NotePad in Microsoft Windows or TextEdit in macOS. It can be also written in more advanced integrated development environments (IDEs) like JetBrains WebStorm. However, this option is paid. HTML and CSS can also be edited straight from the Internet Browser as the browser interprets the HTML code and renders the content according to the specified structure and formatting instructions. This means that any changes made to the HTML code or any new content added will be displayed on the refreshed page. The browser will also reapply any associated styles (CSS), leading to a complete rendering of the updated content.

The disadvantages of HTML and CSS is that they are not a programming language and cannot be used to create dynamic websites and interactive web experiences that require user input and real-time updates like the software our team is trying to build. Our team also lacks the experience and knowledge in HTML and CSS. Older browsers might also not support the latest version of HTML and CSS and will render the website in an unintended way which will degrade the user experience.

3.6 Web Application Deployment

Using a web application that would be built with HTML, CSS, and JavaScript for a project management tool would offer multiple benefits over using a mobile application.

It allows cross-platform accessibility because web applications can be accessed from any device with a web browser, such as desktops, laptops, tablets, and mobile phones. This

would ensure that team members are able to collaborate with ease regardless of their device, addressing the concern for compatibility issues.

Besides, web applications do not require installation or configuration from an app store. Therefore, making them easier to access. Users can easily navigate to the website and log in to start using the software. However, if it is a mobile application, the application often needs to be downloaded and installed on the device first before using..

The web application offers a consistent user experience to users across various devices and platforms. The application design and functionality would remain the same, ensuring that users do not need to re-adapt to using the app when switching devices.

Moreover, web applications can be automatically updated from the server side. This means that users are able to access the latest features and bug fixes without needing to download or install updates manually. In contrast, mobile applications require users to update through app stores first, which could lead to version inconsistencies if not updated constantly.

As developers, it is more cost effective to develop a single web application than creating separate applications for different mobile platforms such as iOS and Android. Therefore, this can save time and resources in development, maintenance, and updates for the software. It is easier for maintenance as updates and changes can be implemented on the server side, making maintenance and bug fixes more straightforward. However, mobile applications require the updates through app stores, which causes delays.

Although mobile applications have their advantages, such as offline access and availability of device-specific features, the benefits of using a web application for a project management tool often outweigh the drawbacks when it comes to accessibility, collaboration, and ease of use across different devices.

3.7 Mobile Applications Deployment

Mobile applications are highly utilised to manage personal or work-related tasks, as mobile gadgets are becoming more common in our daily life. Thus, they often come with advantages and disadvantages when used as tools to manage and track projects.

Compared to web applications, mobile applications can offer users the ability to work offline, although some applications may also require internet connectivity to fully function. Their performance is usually higher than web applications since all their data is stored locally on mobile devices, unlike web applications that rely on web servers for data storage. Additionally, users can receive relevant notifications based on their preferences, such as in-app notifications or push notifications from mobile apps, informing them about project progress or task statuses. Furthermore, mobile applications provide personalization options for users, enhancing their experience by allowing customization of app content according to their priorities and usage patterns. One of the most significant features of mobile apps is their heightened security; they are often considered safer than web apps as they must undergo approval by the app store management before release.

As mentioned above, mobile applications also come with disadvantages. One such drawback is the high cost associated with building them compared to web apps. Various factors influence the cost of mobile app development, including development platforms, device compatibility, and app functionalities. Mobile apps need to be developed using specific languages and Integrated Development Environments (IDEs) compatible with the targeted mobile platforms, as different platforms support different languages and IDEs. Apart from the high initial building costs, maintenance and updates are also expensive, particularly for apps that support multiple platforms. The associated maintenance costs include server management, push notifications, and more.

3.8 Firebase Database Management

Firebase is able to accommodate real-time synchronisation which allows users to view changes almost immediately. This would allow the users to work on software together at the same time. This is beneficial for our software since project management tools require teams to collaborate simultaneously.

Firebase is also scalable and can handle large amounts of data and user connections. This means that users are able to add multiple sprints into the project management system and the software wouldn't crash so easily. This would greatly benefit the system as it would mean that this software is ready for deployment in future and the team wouldn't need to migrate the system over to a new database management system in future to scale the system.

Besides, firebase also provides authentication and security as it has built in authentication and security measures to control the data access. This would prevent unauthorised access from hackers and external parties to the projects that are logged into the software that the current project team is working on.

However, firebase requires paid subscription if the team decides to improve and upgrade the existing software to include other advanced features such as cloud storage, cloud functions, hosting, test lab and phone authentication. Although firebase has a paid subscription service, it can also be accessible through the free tier with limited features.

3.9 Access Database Management

Access is highly customizable and developers are able to customise the database designs to tailor it to specific needs. This means that the developers can customise the database so that the team is able to store only the relevant and valuable information. Hence, we are able to display concise and accurate information, eliminating the redundant data to efficiently execute queries and improve the database performance. Moreover, access is a product by microsoft and therefore would be easily integrated with other Microsoft office products and applications such as Microsoft Excel which is commonly used to analyse data.

However, access databases are not ideal for supporting large scale data or multiple users. Therefore, this will hinder the performance of the project management software because the

software must accommodate multiple users at the same time to ensure close collaboration between scrum members. In addition, there might be security vulnerabilities in the access database if the configurations are not properly performed and proper measures are not taken. This means that the database would be exposed to the risk of unauthorised access from external parties which might cause problems when teams are working on a confidential project which should not be accessible externally.

3.10 Unit Testing

Unit testing is a software testing technique, where small testable parts of a software application, called "units" are individually tested. A "unit" usually refers to a single function, or a small section of code that performs a specific task in the software. The main purpose of unit testing is to make sure that each small portion of the code functions correctly, and produces the expected results.

Unit testing will provide a more precise testing to the software, since small sections of the code are all individually tested. This allows us to detect defects easily in each section, which enables us to make changes to fix the bug in early development. Other than that, it would be easier for us to detect the section with errors in the entire code when software is used, allowing us to make changes easily. Moreover, unit testing supports regression testing, ensuring that the modifications of the code does not create new issues.

However, unit testing may not be the most suitable technique in this case, since it can be time consuming to make test cases for each and every functionality in the software, considering that this is a large-scale project. This will slow down the testing process and the overall process. Additionally, unit testing may not determine the overall functionality of the project. Since all sections of the code are tested, there is a chance that when all functions are used together, it may not work, even if the functionality of each small section works.

3.11 Integration Testing

Integration testing in software testing is a technique that focuses more on verifying if the interface and functionality between different components are working as expected. Integration testing aims to identify the issue when all components of the code are combined.

Integration testing is a suitable technique when identifying problems on the interaction of all components combined, and data exchanges. This also tests the overall usability and interface. Integration testing provides a more realistic testing on the software's behaviour, since it evaluates how the software works when all components are combined. This enhances the quality of the software and can help in uncovering integration issues early in the development.

On the other hand, it can be quite complex to do integration testing in this project. Since this project includes many different functionalities that are connected with each other, it can be quite difficult to test out the overall functionality through integration testing. This method of

testing hugely depends on the readiness of each component. Any delay in any component will affect the readiness to use the software testing method.

3.12 User Acceptance Testing

User acceptance testing mainly focuses on assessing whether the software meets the user's requirements and expectations. This method of testing ensures that the software is ready for deployment.

User acceptance testing allows a more user-centric evaluation of the software, as it ensures that the software meets the needs and requirements of the client or product owner. This validates that the software meets the business level objectives, achieving organisational goals. This testing technique also allows more flexible testing techniques in regards to the feedback provided. This leads to iterative refinements, resulting in a more client-friendly software product.

However, the user acceptance testing method requires a lot of involvement from the client and product owner side. This can be time consuming for both sides. This method of testing can also be very biased and subjective to the perspective of end-users. Since different users have different expectations, it can be challenging to reach a consensus.

3.13 Recommendations

In conclusion, HTML, CSS, and JavaScript are the most appropriate languages to develop web software. This is because they provide a versatile and interactive user interface, allowing for efficient data presentation and manipulation. HTML structures the content, CSS styles it for a polished look, and JavaScript adds dynamic functionality for real-time updates and user-friendly features. A proof of concept/tech spike has also been conducted to prove out a theory of functionality or determine if these languages are appropriate without writing a whole application. Brian, the scrum master has previously coded a personal resume website using HTML, CSS and Javascript that functions correctly with scrolling and page navigation options. CSS has performed the styling and design of the website. Therefore, these 3 languages are the most appropriate for the software.

Other than that, a web application is the most appropriate platform for the software. This is because it offers cross-device compatibility and accessibility. Users can access the project management software from various devices without the need for installing or updating dedicated applications unlike mobile applications. Additionally, web applications facilitate simultaneous collaboration among team members, promoting efficient project management and communication. To test the tech spike, we have observed Trello, a project management software being used regularly as a web application instead of mobile/computer application. This is likely due to the versatility it provides to users.

Moreover, the database management platform that we will be using in our project is firebase. The reason behind this is because firebase provides real-time synchronisation for all users, which will be beneficial for a team-based project. The amount of data handling will be an

asset to this specific project, as users often save extensive amounts of data such as the product backlog and sprint details. Furthermore, firebase offers authentication and security features to manage the data access, preventing unwanted access from unauthorised users. On the contrary, an access database management system is not recommended due to the limited collaborative features and scalability. Additionally, there is a huge risk in security if the access database management system is not set up properly with a strong security measure. This would be a huge issue, especially with user's confidential information. To test the tech spike to analyse if firebase is applicable and suitable as our database management platform, an observation and research into hosting the software on firebase were performed. It is observed that firebase could be used to host and support a variety of content such as websites from CSS and HTML files to Express.js microservices and APIs. Therefore, since this project management website will most likely be coded with HTML, CSS and javascript according to prior research and experience, the team is confident that firebase is able to support this project management software and will allow the feature such as credentials cross checking with database to ensure that only authorised access is allowed.

Finally, the effective testing method that we will be using is user acceptance testing. Unlike integration testing and unit testing, user acceptance testing places direct feedback from the end-users and client. User acceptance testing provides insights into the usability and functionality of the software, and allows the integration of clients too. Additionally, this method of testing allows more quality assurance. Early testing phases are essential for identifying technical issues, so this method ensures that the software is working properly on both technical aspects and user aspects. Moreover, in our scrum team, the integration of client and product owner is very important, for reaching their expectations.

To test the tech spike to analyse whether user acceptance testing is effective for the current web application, we have observed and analysed based on past experience. The team has previously dealt with user acceptance testing of a supermarket application that logs and calculates the final price of products purchased. From this, it is observed that user acceptance testing would be able to test the software in the "real world" by the intended audience or business representative. Therefore, we are able to verify that the application will meet the needs of the end-user, with scenarios and data representative of actual usage in the field.

4. Risk Register

Risk involves uncertainty of a problem or issue arising. Therefore, risk management is required as it is about contingency planning which would streamline the process of problem solving when the risk turns into a problem.

Acknowledging the overall severity of each risk will give us a brief idea on how much time we will need to spend on risk mitigation. The impact of a risk shows the extent to how much it will negatively impact the project. The likelihood of a risk is the probability or chances of a risk from happening.

As shown in table 4.1, the impact and the likelihood of a risk is determined by three scales, low, medium, and high.

		Impact		
		Low	Medium	High
Probability	High	Low	Medium	High
	Medium	Low	Medium	Medium
	Low	Low	Low	Low

Table 4.1: Risk measurements

The possible ethical risk that would be encountered during this project is conflict of interest. This could happen during a situation where different teams are working towards different goals that incentivised them to jeopardise the progress of each other. Besides, there could be corruption occurring during the project which would be an ethical risk. This could happen because the client might be pushing for a certain agenda for their individual company's benefit and would sometimes offer bribes to the project management team in return for certain actions.

Table 4.2 shows the associated risk as well as their likelihood of occurrence, impact as well as severity. The table also suggests possible mitigation strategies to prevent the specific risk.

ID	Date raised	Risk Description	Likelihood of the risk occurring	Impact if the risk occurs	Severity	Owner	Monitoring Strategy	Mitigation Plan
1	17/8/2023	Software bugs in the system risk	High	High	High	Head of programmer	Ensure that code always runs as expected before working and developing upon the existing code.	1. Develop test cases using unittest. 2. Make sure code works without error before pushing to gitlab.

							This is to monitor the current code and ensure that it works.	
2	25/8/2023	Conflict of ideas between team members risk	High	High	High	Scrum Master	Ensure that conversations are moderated by the third person participating in the conversation. The person is responsible to monitor and detect when the conversation becomes heated.	1. Both sides must ensure that they listen to the opposite views. 2. Both sides must agree that everyone is working towards the same project goal. 3. Gather consensus from all team members to decide which option to choose to move forward.
3	25/8/2023	Team member burnout risk	High	High	High	Scrum team	Observe team spirits during meetings and each person should be vigilant in detecting when negative emotions are brought up during conversations as that could be a sign of burnout.	1. Organise wellness programs for the team to support mental health 2. Give support and motivation to team members when members are feeling down or discouraged due to overwhelming load from projects or other units. 3. Encourage and create a positive working environment for the team.
4	27/08/2023	Data Leak Risk	High	High	High	Scrum team	Ensure that detailed logs of who accessed what data and when is being tracked to ensure that the data access is monitored and data leak	1. Limit who has access to more sensitive data 2. Encrypt sensitive data 3. Regularly update and patch the software.

			Medium	High	Medium		is prevented.	
5	17/8/2023	Lack of communication risk	Medium	High	Medium	Scrum Master	If there is a lack of communication, it should be visible because team members will not be aware of each others' progress.	1. Weekly Standup Meetings 2. Use Google Docs for Centralised Documentation
6	21/8/2023	Unrealistic or unreasonable requirements	Medium	High	Medium	Product Owner	Ensure that the group has frequent communications with the client to clarify all the expectations and address any unreasonable expectation when needed. This can be monitored through discussion internally with the scrum team to determine if the requirements are unrealistic or unreasonable.	1. Negotiate with the client and provide reasonable alternatives. 2. Give examples of successful alternatives to clients.
7	25/8/2023	Unrealistic time frame given risk	Medium	High	Medium	Product owner	This can be monitored by observing the current sprint or product backlog. Through internal discussion, the team can decide if the time frame for	1. Plan the project ahead, make sure the tasks are divided reasonably for each iteration. 2. Estimation of the time needed to finish each task must be done. 3. Avoid taking on extra tasks that clients did not ask for.

			Medium	High	Medium		such tasks are unrealistic or reasonable.	
8	25/08/2023	Unethical approach or conduct risk	Medium	High	Medium	Scrum team	Ensure a strict ethical guideline is implemented. This can be monitored by researching the methodology or code used to ensure that it is ethical.	1. All team members should be aware of the ethics and ethical behaviours that should be maintained. 2. Immediate action must be taken and investigation must be done by the entire team when an unethical behaviour has been conducted.
9	25/08/2023	Dependency on third-party libraries or APIs risk	Medium	High	Medium	Scrum team	This can be monitored by observing the libraries that are imported and used in the code.	1. Ensure that licences are compliant, and regularly check for updates or changes to third-party components.
10	27/08/2023	Project Quality Risk	Medium	High	Medium	Scrum team	This can be monitored and maintained by having regular code reviews to ensure the quality is being maintained.	1. Ensure that all team members consistently check the requirements when building the system for each user story.
11	17/8/2023	Conflicts not resolved in a timely manner	Medium	Medium	Medium	Scrum master	Observe if there is any negativity lingering during discussions. If there is, conflicts should be resolved as soon as possible.	1. Weekly standups for conflict resolution 2. Vote by general consensus after hearing from each side
12	17/8/2023	Lack of skills resource	Medium	Medium	Medium	Scrum team	Ensure that all team members	1. Online resources 2. Each team member studies content

		s risk					have prior knowledge to the resources needed in this project. This is monitored by making sure that all team members present their findings during the stand-up meetings.	accordingly. 3. Seek help to other team members when needed.
13	17/8/2023	Uneven work distributio n risk	Medium	Medium	Medium	Scrum team	If there is uneven work distribution, it should be detected as soon as possible. It should be visible by observing the task list for each team member. Therefore, if there's an obvious difference between each other, there is an uneven work distribution.	1. Weekly discussion on the topics or project that has to be done and make sure everyone is satisfied with their workload. 2. If workload is too burdening, the team member should voice out to other members in the team. 3. Make sure everyone is clear with their roles and responsibility and make sure that everyone in the team is satisfied with their roles.
14	25/8/2023	Limited access to resource s risk	Medium	Medium	Medium	Scrum team	This can be detected when there are limited references when doing a report or when there is difficulty in developing the code due to limited online resources.	1. Identify all the necessary resources needed early on during planning. 2. Find alternatives to access materials or software tools if some are not accessible to us. 3. Leverage library resources or seek support from faculty.

15	25/08/2023	Scheduling conflicts and clashes risk	Medium	Medium	Medium	Scrum team	Ensure the team does frequent updates and transparency on scheduling, to ensure a more efficient scheduling process. This can be monitored by observing the scrum team members' class timetable.	1. Organise google calendar for everyone to input their scheduling day by day. 2. Communicate within teams often and find the best time when everyone is available. 3. Use scheduling tools such as LettuceMeet.
16	25/08/2023	Inconsistent user experience across platforms risk	Medium	Medium	Medium	Scrum team	This can be monitored by observing the graphics and interface across platforms when it has been deployed.	1. Ensure that regular testing is conducted on different devices and browsers, following responsive design principles.
17	17/8/2023	Team members' health risk	Low	High	Low	Product Owner	Regularly communicate with team members about their health status. This is monitored by making sure that members update their current health status.	1. Discuss with client to negotiate possible extension of deadline. 2. Discuss with other team members who are willing to hand over the tasks assigned to that member. 3. Ensure to exercise daily. 4. Ensure to stay hydrated everyday.
18	21/8/2023	Academic misconduct risk	Low	High	Low	Scrum team	Ensure that all the work are sent into a plagiarism checker or checked by all team members, before	1. Each team member is responsible for their own work. 2. If a member is suspected of breaching academic misconduct, the team will try to solve the problem personally. If

			Low	High	Low		submission. This can be monitored with a plagiarism checker.	the issue is not solved over time, approach the teaching team.
19	25/8/2023	Laptop hardware breakdown risk	Low	High	Low	Scrum team	Ensure and monitor that hardware works as expected before continuing on to code and develop upon the current repository.	1. Ensure that backup copy is made and saved after every change. 2. Utilise gitlab for version control to save code to a repository. 3. Ensure that all team members have their own backup of the new code.
20	25/8/2023	Natural disasters risk	Low	High	Low	Scrum team	This can be monitored by looking at the natural disaster forecast online.	1. Plan ahead for extreme weather that may cause any possible natural disasters. 2. No exact way to solve this issue, workload will have to be carried over to other team members if it happens to specific people.
21	25/8/2023	Loss of key personnel risk	Low	High	Low	Scrum Master	Ensure open communication within the team. The loss of key personnel can be monitored by keeping track of the group chat. If a particular scrum member does not respond after several days, actions have to be taken.	1. All team members should frequently update about their progress. 2. If risk does occur, workload will have to be fairly divided amongst the rest of the team members. 3. Scrum master should monitor the scrum team at all times for any irregularities.
22	25/08/2023	Data	Low	High	Low	Scrum	This can be	1. Ensure that the code

		Loss Risk				team	monitored by observing any data loss before continuing to expand on the existing code base.	base is backed up in several locations: a. Online (GitLab, Google Drive) b. Offline (Scrum team's hard drive)
23	21/8/2023	Poor time management risk	Low	Medium	Low	Scrum team	Ensure that all the tasks are completed on schedule and identify any potential delays early. Everyone should monitor their time management and ensure that they complete tasks before the dateline.	3. All team members should share their management plans and all team members can review it. 4. If a team member is struggling with time management and falling behind on the project, a meeting will be held to address and resolve this matter.
24	25/8/2023	Poor process management risk	Medium	Low	Low	Scrum master	Ensure that the team understands their roles respectively and any inquiries must be addressed immediately.	1. Scrum master should understand their role properly and is able to provide guidance to the team, and address concerns with the scrum team. 2. Seek for consultation from the teaching team to check if the team is on track.
25	21/8/2023	Overdue project risk	Medium	Low	Low	Scrum team	Ensure that all team members are transparent with each other about their current workload over time. Make sure each members' voices are heard. Everyone	1. Weekly checkup on each member's workload. 2. Discussion on tasks handover to other team members if necessary. 3. All team members work together to make sure the work gets done, despite the amount of other workloads.

							should be responsible to monitor the project deadline regularly.	26

Table 4.2: Risk register