Management Information SENG3011

Team Jamva

Jay Patel, z5309776

Jesse Merhi - z5312498

Vishnu - z5280782

Jack Whaling - z5162212

Marko Wong - z5309371

1.0 Introduction

The purpose of this report is to provide a project plan showcasing team member responsibilities and working arrangements, and the overall project management including the project's charter and the software tools being used in development.

1.1 Table of Contents

2.0 Member Responsibilities

To effectively allocate work and allow the team to operate with an AGILE methodology, whereby roles have been defined and assigned to particular members. Specific roles have been defined to simplify member's working arrangements and allow the facilitation of AGILE practices between members. By employing particular working arrangements, the team members can easily collaborate and maintain communication ensuring AGILE practices are conducted.

2.1 Role Definitions

To define which responsibilities should be delegated to which member, role definitions have been described to elaborate on member's specific tasks. Role definitions will enable AGILE development practices as defining role responsibilities will bring transparency as to which members must work interdependently.

Role	Responsibilities
Product Owner	- Creates users stories that can be implemented - Analyses the project from a business perspective - Works with QA and stakeholders to ensure acceptance criteria are right
Scrum Master	- Updates the project manager on the progress of development - Provides technical leadership, design guidance, design review and code review - Ensures the quality of the product is maintain to an appropriate level at all stages of development - Updates Sprint Board Daily
Project Manager	- Ensures the development team no roadblocks - Keeps the development team on track
Developer	- Designs solutions to the problems in the project - Decides if major redesign is needed on existing systems - Codes and Unit Tests - Participates in design and code reviews
Tester	- Develops & maintains test scripts - Updates API test harnesses - Ensures unit testing happens - Finds bugs and reports them to the development team

2.2 Working Arrangements

Each team member has been allocated a particular role, where all members will be required to fulfil the role of Developer and Tester. As the software architecture being used may not be familiar with all members, allowing the team to work collaboratively as Developers and Testers will provide the opportunity to learn and enforce AGILE development practices.

Team Member	Role	Justification
Jay Patel	Product Owner Developer Tester	In addition to Developer and Tester, Jay has been given the role of Product Owner as he is a Commerce student. Thus, he is comfortable and familiar with general business practices such as stakeholder management.
Jesse Merhi	Project Manager Developer Tester	In addition to Developer and Tester, Jesse has been given the role of Project Manager as he is familiar with all members. Thus, he can effectively communicate with all members and ensure work is completed.
Vishnu	Developer Tester	As Vishnu's roles include Developer and Tester, he will be given a greater focus on the development and testing of the application. As such, he will be required to work interdependently with all other members.
Jack Whaling	Scrum Master Developer Tester	In addition to Developer and Tester, Jack has been given the role of Scrum Master as he is the most familiar with the current software architecture. As such, Jack is in a position where he can assist members with development and testing tasks.
Marko Wong	Developer Tester	Akin to Vishnu's role, Marko's roles of Developer and Tester will require him to focus on development and testing. Hence, he will also be required to collaborate with other team members.

3.0 AGILE Project Management

The project structure plan will follow the AGILE methodology to include multiple sprints for each main feature. Within each sprint, there will be 5 stages: requirements, design, implementation, verification and maintenance. By using sprints it will ensure the end product is fully functioning according to the requirement and is of high quality. The project's code will be shared on a Git repository for easy access for each member and version control. While the team's communication will be done through various platforms: Microsoft Teams, Discord and Jira.

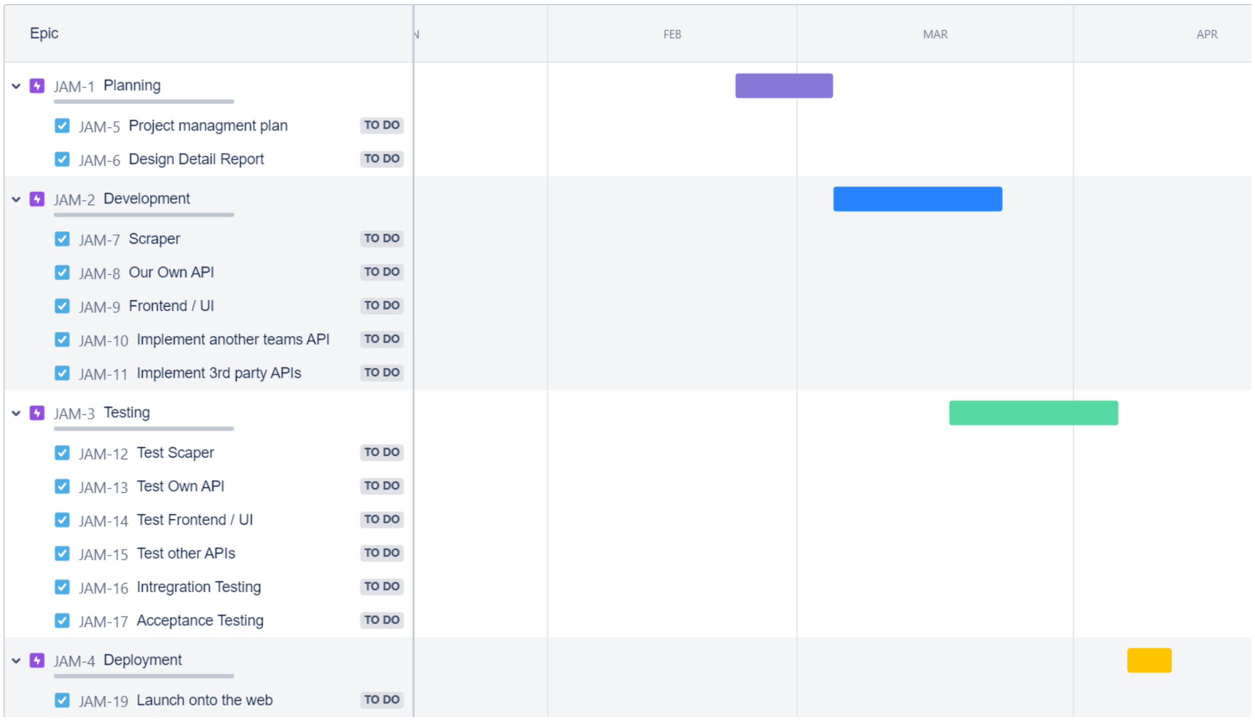
3.1 Sprint Plan

The following sprint plan outlines the initial objectives and tasks set for the team to complete per deliverable or sprint.

Sprint no.	Backlog Item	Tasks	Time Limit (Days)
1	Scapper	Write up initial scapper code	2
		Test scraper code	1
	API	Design API	2
		Code up API	2
		Test API	1
	Frontend	Design UI	2
		Code up Frontend	2
		Test Frontend	1
2	Integrate another team API	Modify code to accept another teams API	3
		Test newly integrated API	2
	Integrate 3rd party API	Modify code to accept another teams API	3
		Test newly integrated API	2

3.2 Timeline

The following timeline is an initial outline of which tasks need to be completed and the date in which each task should be completed. Jira was used to develop the initial timeline and will be used in the future to amend this existing timeline.



Epic	
JAM-1 Planning	
☑ JAM-5 Project managment plan	TO DO
☑ JAM-6 Design Detail Report	TO DO
JAM-2 Development	
☑ JAM-7 Scraper	TO DO
☑ JAM-8 Our Own API	TO DO
☑ JAM-9 Frontend / UI	TO DO
☑ JAM-10 Implement another teams API	TO DO
☑ JAM-11 Implement 3rd party APIs	TO DO
JAM-3 Testing	
☑ JAM-12 Test Scaper	TO DO
☑ JAM-13 Test Own API	TO DO
☑ JAM-14 Test Frontend / UI	TO DO
☑ JAM-15 Test other APIs	TO DO
☑ JAM-16 Intregration Testing	TO DO
☑ JAM-17 Acceptance Testing	TO DO
JAM-4 Deployment	
☑ JAM-19 Launch onto the web	TO DO

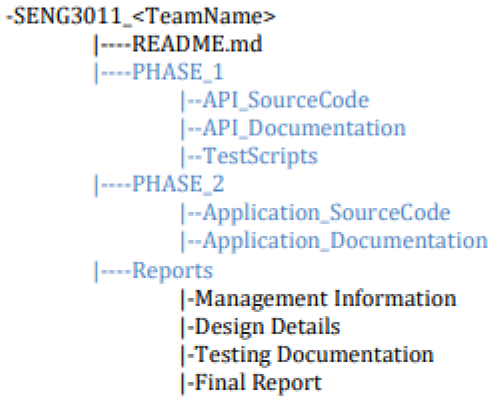
3.3.0 Software Tools

To assist project management, various software tools have been employed to facilitate collaboration between team members. By using software tools such as Git which allow individual members to work simultaneously and Jira for general project management, the team can imbed AGILE development practices into our development process.

3.3.1 Code Repository

The team has selected GitHub as the code repository. Git has been selected as each member is extremely familiar with Git and GitHub. Moreover, as Git offers branching capabilities, members can simultaneously work allowing for greater collaboration and efficiency.

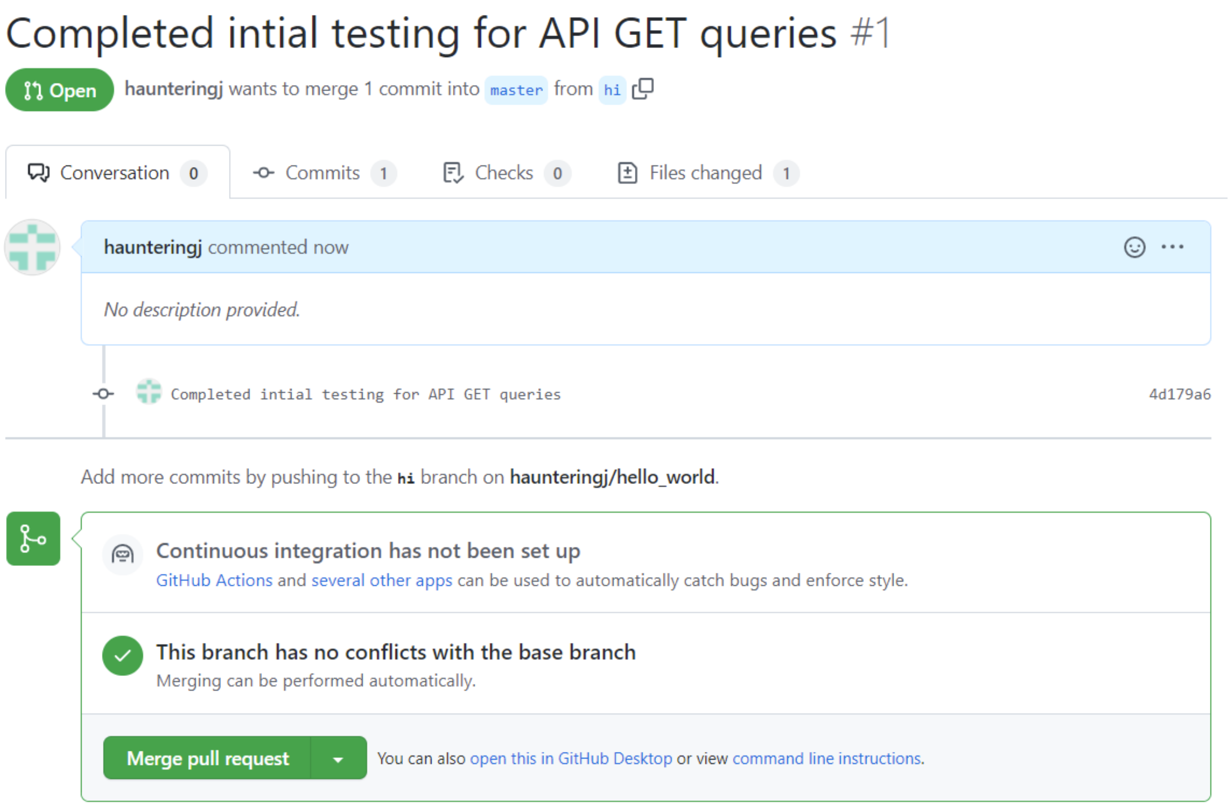
The GitHub repository will include sections for all major development phases and reports. The initial phase, PHASE_1, will include API source code, API documentation, and any test scripts. The second and final phase, PHASE_2, will include the application's source code, and the application's documentation. Finally, the reports section will include various reports including Management Information, Design Details, Testing Documentation, and the Final Report.



```
-SENG3011_<TeamName>
        |----README.md
        |----PHASE_1
                |--API_SourceCode
                |--API_Documentation
                |--TestScripts
        |----PHASE_2
                |--Application_SourceCode
                |--Application_Documentation
        |----Reports
                |-Management Information
                |-Design Details
                |-Testing Documentation
                |-Final Report
```

The team will use GitHub's branches to ensure working code is segregated from developing code. As such, the team will merge working branches with the Master branch to ensure the Master branch is always up-to-date with the latest working update. Additionally, by branching off Master, each member can append updates to the latest working version of the application. This will significantly reduce merge conflicts between branches and ensure the version of the application stored in the Master branch is always working.

To ensure members are effectively collaborating during development, the team has implemented a code review system. In doing so, members will have the opportunity to have their code peer reviewed when merging branches. The review system will require members to get a different Developer to review and accept their code. Using GitHub's pull request system, once a member is ready to merge their branch with another, they will be required to submit a pull request. Once a pull request has been submitted, another member will peer review the submitter's work. If the code is acceptable with no merge conflicts, the reviewer will comment that the code is ready to be merged. The submitter will then be allowed to merge the code.



Completed intial testing for API GET queries #1

haunteringj wants to merge 1 commit into `master` from `hi`

haunteringj commented now

No description provided.

`Completed intial testing for API GET queries` 4d179a6

Add more commits by pushing to the **hi** branch on **haunteringj/hello_world**.

Continuous integration has not been set up

GitHub Actions and several other apps can be used to automatically catch bugs and enforce style.

This branch has no conflicts with the base branch

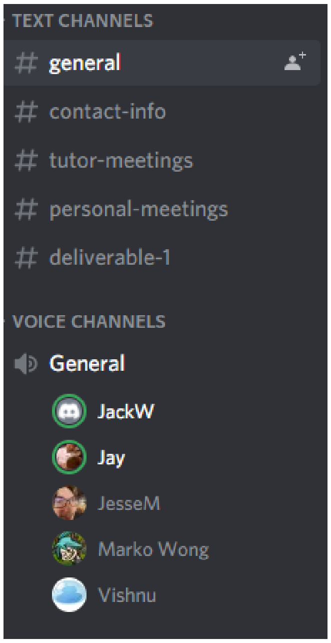
Merging can be performed automatically.

You can also open this in GitHub Desktop or view command line instructions.

As a result of using GitHub's features such as the pull request system, the team will ensure effective AGILE practices are maintained and members will continue to work collaboratively.

3.3.2 Communication

The communication throughout the project will be carried out on digital platforms mainly on Microsoft Teams, Discord and Jira with each platform having its own role. Microsoft Teams will be used for formal communication between the development team and the supervisors of this project. While the informal communication within the development team about anything relating to the project will be done on Discord. The team chose Discord because all members of the development team were already familiar with the platform from everyday uses. It is also easily managed and can categorise communication with channels. Jira will be for maintaining a healthy progression of the project by showing the past, present and future tasks to be done. As well as showing overall progress the project is on. Jira was chosen for its wide range of project management tools like a roadmap and it is also highly compatible with other software like Git. With all these platforms of communication, it will make sure no information is lost within all members of the project while keeping all the communication categorised to enable effective communication.



TEXT CHANNELS

general

contact-info

tutor-meetings

personal-meetings

deliverable-1

VOICE CHANNELS

General

JackW

Jay

JesseM

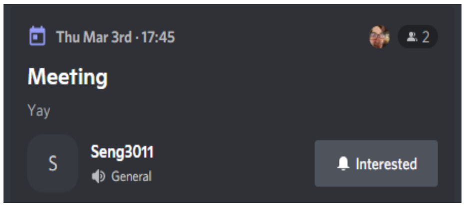
Marko Wong

Vishnu

3.3.3 Organisation within platforms

Discord is organised and sorted mainly by deliverables and other miscellaneous channels. These channels include general, contact-info, tutor meetings, personal-meetings, deliverable-1,etc . With future deliverable channels to be created when the time is appropriate. The general channel will be used for overarching discussion about the project as a whole. The contact-info channel is for each team member to quickly look up how to directly contact other team members. While the tutor-meetings channel will be for all the notes taken during the tutor meeting to ensure all members have access to a backlog on what was mentioned in the tutor meetings. Personal-meeting channel will be for the teams own meetings notes. The deliverable channels are for anything related to the deliverable.

By using these channels it will keep a widely accessible backlog for team members that needs it. Useful for when a team member misses a meeting or forgets something. Development team meetings will be held with the discord inbuilt voice channels as it provides a quick and convenient online meeting place whenever it is needed. The meeting event function will also serve as a helpful reminder to make sure no team member forgets to join the scheduled team meeting.

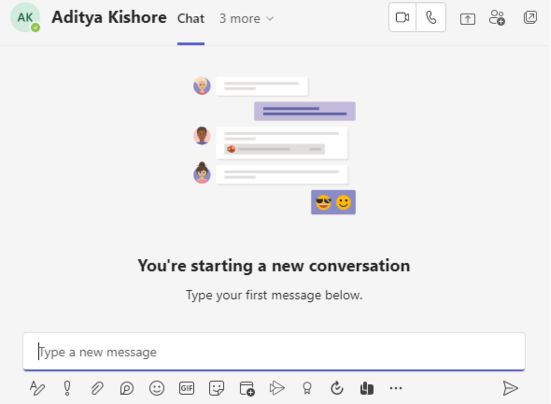

Thu Mar 3rd · 17:45

Meeting

Yay

Seng3011
General

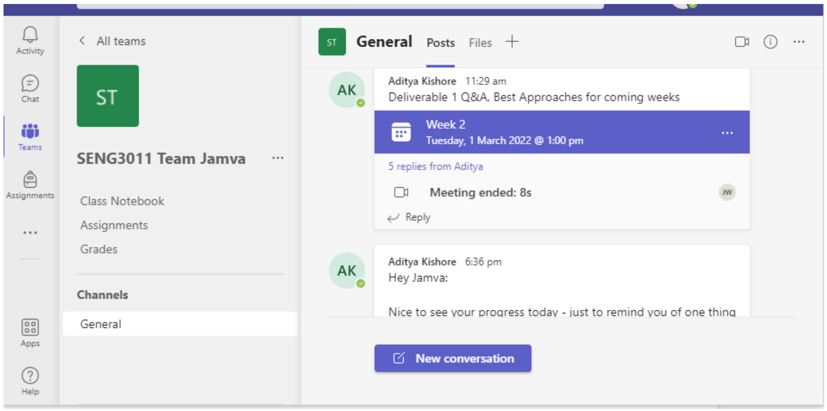
Microsoft Teams is layed out with one general channel with a post and files tab. Due to the small amount of information only one general channel is needed. Causing no information will be lost as it is all in one place. Also there is a files tab in case the team and the tutor need the share file. Additionally in Teams it allows individual team members to directly message the tutor in the chat section.


Aditya Kishore

You're starting a new conversation

Type your first message below.

Type a new message


SENG3011 Team Jamva

General

Aditya Kishore 11:29 am
Deliverable 1 Q&A, Best Approaches for coming weeks

Week 2
Tuesday, 1 March 2022 @ 1:00 pm

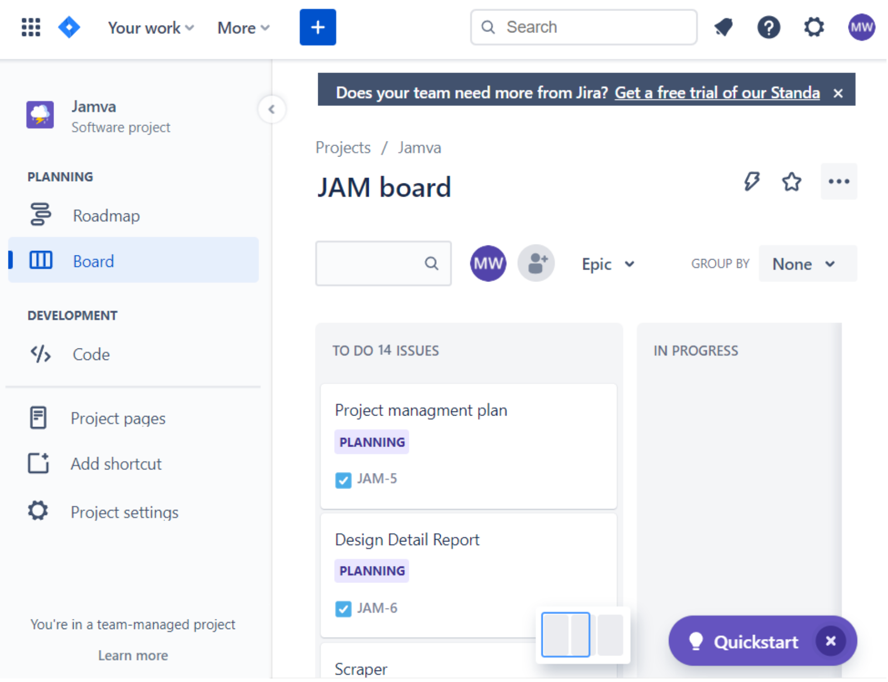
5 replies from Aditya

Meeting ended: 8s

Aditya Kishore 6:36 pm
Hey Jamva:

Nice to see your progress today - just to remind you of one thing

Jira is formatted with a Roadmap, Board and code section. With the Roadmap for a visualisation of the tasks active time period as well as an overview of the whole project process. The board is a list view of all the tasks relating to the project, where team members can assign each other to tasks, clearly indicating the state of each task. The code section linked to the Git repository showing the progress of the actual code.



Does your team need more from Jira? Get a free trial of our Standa ×

5.0 Conclusion

The team has formed the project’s management with a large focus on healthy AGILE management practices. In defining member roles and responsibilities, areas which require interdependence between members have become clearer. In developing a preliminary project charter through the use of a sprint plan and timeline, the team has set goals which will be required to complete the project. Finally, the software tools used for code repository management, and general communication are critical to the team’s AGILE management practices as effective communication will provide the team with the opportunity to collaborate.