

TEAM MEMBERS

1. William Susanto

- a. Contact : +6011-23283342
- b. Email : wsus0001@student.monash.edu
- c. Role : Testing, Back-End Developer
- d. Responsibility : Perform tests on the program and contribute to the back-end development.

2. Bryan Ho Yung Kynn

- a. Contact : +6012-3662683
- b. Email : bhoo0005@student.monash.edu
- c. Role : Scrum Master, Back-End Developer, Documentation
- d. Responsibility : Manages Scrum Meetings, develops back-end queries as well as providing improvements on documentation quality.

3. Adrienne Rio Wongso Atmojo

- a. Contact : +6016-7445121
- b. Email : aatm0001@student.monash.edu
- c. Role : Front-End and Back-End Developer
- d. Responsibility : Mainly in charge of developing the UI interface and back-end queries on GitHub.

4. Mohamed Nazmi

- a. Contact : +6016-6125096
- b. Email : mnmmoh48@student.monash.edu
- c. Role : Documentation, Testing, Front-End Developer
- d. Responsibility : Mainly in charge of documentation and making sure it is up to date. Also contributes to testing and UI interface development.

We would contact each other using group chats on both WhatsApp and Discord. Furthermore, we also created a team drive in Google Drive for this project's purposes. While every member are given their respective responsibilities, they still have to be cross-functional given the circumstances.

PROJECT VISION

For software developers

who need to track which task/work is being done by every team member

the GitHub Inspector

is a repository tracking and inspection system

that tracks and analyses changes done by each developer in the team.

Unlike the regular information provided by the GitHub repository,

our product provides more detailed information on the contribution of each team member.

PROCESS MODEL

The process model that our group will be following is scrum. However, there are some parts of the scrum process model that would be infeasible to follow as university students and not full time workers. Here are the parts of the model that we will be cutting corners on, and the alternatives that we will be implementing instead.

i) Daily standup meetings

- Having daily meetings is difficult as we have different schedules and finding a time each day that everyone is free would be difficult. As an alternative, we will be having bi-weekly meetings, one in person every Wednesday before our tutorial and another virtual meeting via Skype.
- We will also be constantly communicating via Discord, to keep each other up to date on the task we are currently working on or if we need any help.

With the above in mind, our process model is as follows. Our sprint will be 2 weeks long, at the start of each sprint we will be allocating tasks to each team member. We will identify the milestones that we intend to reach by the end of the sprint. Bi-weekly meetings will be conducted to discuss our progress, one physical meeting and one virtual meeting via skype. At the end of each sprint we will have a product review within the team and also with our product owner. A retrospective session will be conducted at the end of the sprint, to see if the milestones have been reached. If the milestones have not been reached we will try and identify the reasons why and also ways we can improve our effectiveness and productivity in order to reach these milestones.

DEFINITION OF DONE AND VISION STATEMENT

Our team's vision statement

"Our team's vision is to provide a platform where users can analyse their repositories with accurate and detailed information using the most innovative browser technology."

Team AGILEAF

Definition of Done

EARLY PHASE

- Requirement specifications is approved by the product owner
- User stories are completed with respect to the product theme

MIDDLE PHASE

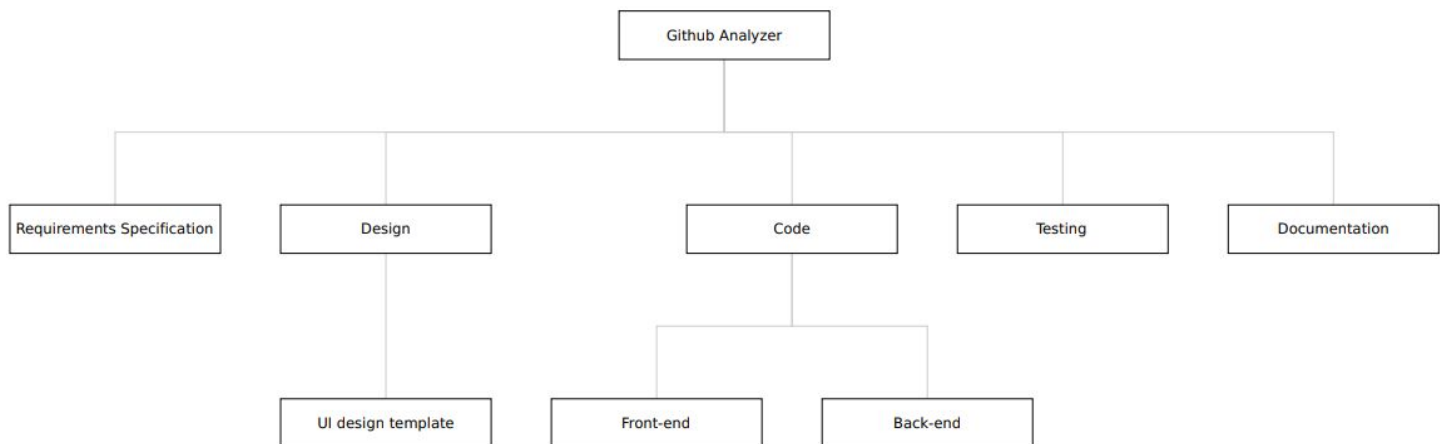
- All unit test cases has been written
- Design of the software is completed (UI mockups)
- User documentation started and updated
- Code review by peers is performed

FINAL PHASE

- All code has been well documented
- All code passes all unit tests and has been tested throughout
- Final software satisfies all software requirements specifications
- All code has gone through user testing
- The final software has been well documented in a documentation

HOW TASKS ARE ALLOCATED

Within the project lifecycle, the tasks are broken down into smaller activities. The following WBS chart (Work Breakdown Structure) illustrates the main activities our group will perform within the project:



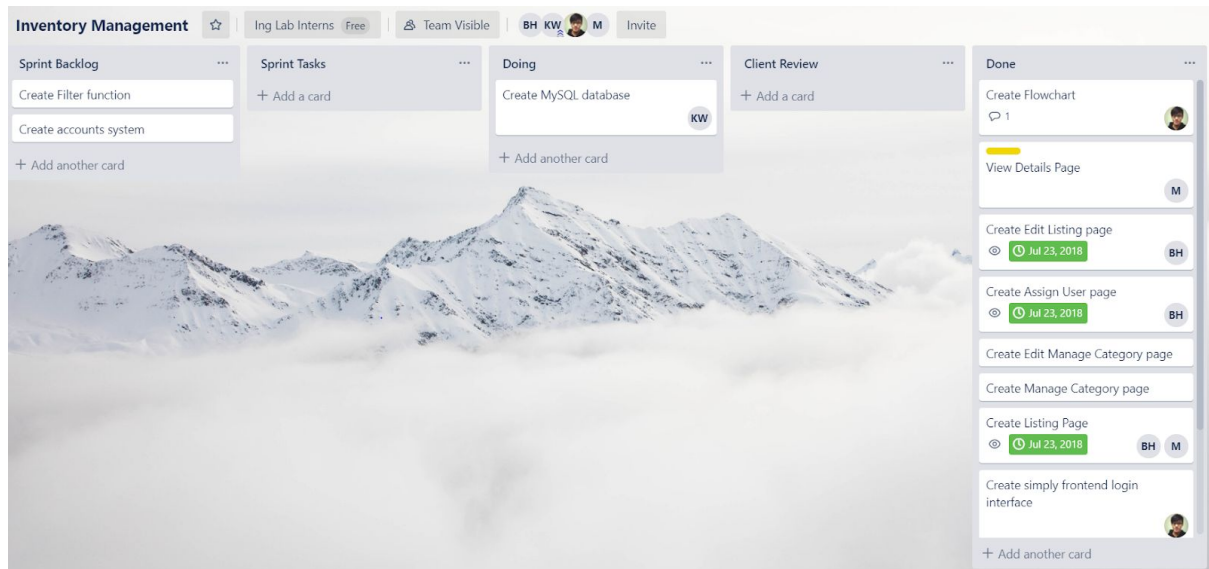
Each activity is decided and allocated to different team members, and each team member is responsible for their particular activity. However, each activity can be further subdivided if needed to enhance and speed up the development on that activity.

- Requirements specification
 - Requirements specification ensures the software meets the required specifications and allows team members to keep track of those specifications. The team member responsible for this activity will list down the main requirements for the website.
- Design
 - The overall user interface design of the website.
 - UI design template allows front-end specifications to be presented and laid out early to the team.
 - Those responsible for this activity will create an overall mock layout of the website and present it to the team for verification.
- Code
 - This is the most crucial part of the project, where team members are allocated to either front-end development or back-end development.
 - Team members allocated to front-end development will deal with front-end issues such as UI implementation, the overall look of the website and user experience.

- Team members allocated to back-end development will deal with back-end issues such as querying data from Github API.
- Testing
 - The team inspects the website for any potential errors and reports back to the team members responsible for the code activity of any errors.
- Documentation
 - Documentation involves documenting the overall website of its functionality, purpose, and usage.

HOW TIME AND PROGRESS ARE TRACKED

In addition, our team will be using Trello to keep track of what needs to be done, what is being done, and what is already done. Essentially, it will be the location where we record and manage our backlogs as well as keeping track of the team's progress and Sprints. Trello is an excellent tool since it helps to communicate and remind all team members of what activity is potentially needed to be done. Here is an illustration of Trello's interface:



In Trello, we can:

- Create lists for our Product Backlog and Sprint Backlogs respectively.
- Create a list for adding cards of Sprint Tasks to be done.
- Assign each task added under the list with a due date and the team member responsible for it.
- Watch a list or a card and to be notified with any changes made to the tasks or the upcoming due dates to keep track of time and progress through our emails.

Link to our Trello:

<https://trello.com/agileaf>