



**1.**

In this case, I will suggest the AGILE METHODOLOGY for doing the project. I will introduce the reason of this choose below:

First is about requirements characteristic:

- Reliability: The requirements were defined pretty clear for a basic remote working software helper and the need of products is very high.

- Functional requirements and non-functional requirement: At lease 5 functional requirement as well as at least 3 non-functional requirements.

- Requirements will be able to change in the future? The requirements can change in the future because this is the first time that human has faced to the pandemic that very huge negative impact like this. We just know the need is the software products which can help people can remote work but the feature of this software can be change depends on the user experiment when it releasing. As well as, it will be considered for user to add, justify, remove some feature for remote working become easier.

- Requirements can be defined clearly in the early stage? The requirements were defined pretty clear in the early.

- The desirable time for introducing the products or some part of the product to the release? Very high because the impact of COVID-19 pandemic is very large. Everyone can’t go out of the home, so the need of the software product for working from home efficiently and effectively is very large and immediately.

- Tools and technology that was used in the project is consistence or can be change in the future? Tools and technology that was used in this project can be change because we need to update the new technique for increasing the user experiment avoiding to the lag, disconnect situation or the application can be drop down when the number of user access in the same time is very large. The technology is changing very fast so if we know to use the newest technology that will help increasing the user experiments.

Second is about development team:

+ The number of team member: 6 developers and 2 QA

+ The understand of development team about the project: The development team know a little bit about the situation that need be solve because this is the first time that human facing the large negative effect pandemic like this. So even development team have triad or doing some software like this.

Third is about user involvement in the project:

+ The level of user involvement in the project: Large because this project focus for using indoor the company, focus on employee in the company. So user involvement in the project is large.

+ The project size: Medium

Following the characteristics that I listed above the methodology that most propriate for this project is AGILE METHODOLOGY because the project size is small, speed to the release is high, the level of user involvement is large, the understand of development team is just a little bit, tools and technology can change in the future, requirements can be change, ….

The project has three core features, which are

+ Task management

+ Deadline tracking

+ Progress tracking

**2.**

Functional Requirements:

- Task List: Allow team member to prioritize tasks and organize their work in a way that makes sense to them

- Assigning Tasks: Allow the leader to give tasks to themselves or others. This helps ensure that tasks are distributed evenly across the team and that everyone knows what they need to do

- Task Prioritization: Allow the leader to prioritize tasks based on importance and urgency and re-prioritize functions as needed

- Progress Tracking: Track the progress of tasks and communicate with other team members about tasks status. This helps ensure that everyone is on the same page and that there are no surprises regarding task completion.

Non-functional Requirements:

- The module should be able to load and display large numbers of tasks without significant delay.

- User actions should result in immediate system response with minimal lag time.

**3.**

- As a team member, I want to be able to prioritize my task so that I can consider the task I need to do first and immediately.

- As a team leader, I want to be able to re-prioritize functions as needed so that I can handle if the requirements of users was change immediately.

**4.**

**Story Map for the leader’s screen in the web app.**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Login | | Task Lists | | Task Prioritization | | Deadline Tracking | | Log out |
| Login Normal | Login by Google | Show task list | Priority Task | Prioritize Task of Team | Re-prioritize task of Team | Set the deadline for task | Notification to member | Log out by button |
| - User input username, password  - Check the username, password in the database  - Login successfully if username, password is math. Otherwise is not  - Notification that login successfully and redirect to the home page  - Notification and back to the login page if login unsuccessfully. | - User choose the icon Google to login with Google  - Check the return of Google Service  - Login if Google return true. Otherwise is not  - Notification that login successfully and redirect to the home page  - Notification and back to the login page if login unsuccessfully. | - Show task that not been done  - Filter task list that not done by priority  - Show task list that done | - Prioritize task for the top show  - Notify if change priority successfully.  - Notify if change priority unsuccessfully. | - Edit the priority for all member of team  - Redirect to the team  task list to continuate change  - Notify to all member that priority change. | - Re-prioritize functions of project if needed  - Notify to all member that priority change. | - Set the deadline of project for all member can do.  - Notify if set deadline successfully  - Notify to all member that deadline is set or change. | - Notify to member if deadline is nearly  - Notify to member if has any needed situation | - User click in the log out button  - Redirect to the login page if log out successfully  - Notify if log out is unsuccessfully. |

**5.**

Low impact if wrong, Low probability of it being wrong:

**-** The member will need to notify if the leader changes the deadline of task.

Reason: If members don’t need to notification that not impact too much in the task. But all the member need a notification if deadline change to the immediate response.

High impact if wrong, High probability of it being wrong:

- If the deadline of task change, the priority of all team will be change.

Reason: If priority of it change, that impact very much to the progress of doing that task and affect to the progress of doing others task. The probability of this is high because priority of all team is the things that hard to change.

High impact if wrong, High probability of it being wrong:

- If the deadline of task change, the progress of it will be change.

Reason: If progress of it will be change, all the previous progress will be drop down. So that impact very large to doing it. And the probability of it being wrong is high because no one want to change the previous progress of team.

**6.**

In this case, I suggest using a combination of manual testing and automated testing. Because there are some things that cannot be tested with automated testing such as performance, as well as whether the application interface is user-friendly, ...

With manual testing, there will be two types: White box, Blackbox.

- White box: Used to test all errors in the code and the flow of the code so that the next software maintenance time the development team can easily find and fix the errors encountered.

- Blackbox: Used to test whether the application meets all the functional and non-functional requirements of the customer or not as well as to check if the user experience related issues such as user-friendly interface, performance, etc. good function, ...

With Automated Test: Automated testing will help reduce costs, reduce time in the testing process as well as increase the efficiency of finding errors because it is automatic and is deployed regularly.