## Feb 10, 2024 | [Night Owlers Status Update](https://www.google.com/calendar/event?eid=NmM3dDM1OG1tZ2VpNzQzbnRkbzl0Y2YxZ24gc2hhbWlpdmFuQG0)

Attendees: Theebika, Angru, Ivan, Jutipong

Absent : Ryan

Meeting time: 6:30 pm - 8:30 pm

Location: Zoom

# Agenda

## Status update

* (thee) Discuss user story (**CRUD operations on vehicles**)
  + Improvement(add narrative, DOD, Acceptance criteria)
  + Fix US : As a system administrator, I want to update the system, so that the vehicle details are up to date.
    - System is vague
* (Ivan, Jutipong,angru) agree User story that thee write
* (Ivan) point-out how to make the user story clear
* Ryan’s
  + user story 1: break down acceptance criteria for User Story 1
    - EXAMPLE :
  + User story 2 conflicts with Thee’s(both did the same work )
  + US 5 is a sub
  + It is not necessary that we don't have customer representative
* Angru
  + Need to improve section 1.1 and 1.2
  + No background need get to the project
* Jutipong
  + For project timeline, you have to be more specific
  + Correct Section 2.3(weekly sprint planning)
  + Google → google drive
  + For project management : Git
* Break down search
  + Search a in type
  + Search can also mean browse, select cate
* Roles and Background for teams(Not done, thee will send)

Read Me

* An exceptional README
* project description(not done) - Jutipong
* installation instructions and usage guidelines, with references to the wiki for additional team(Jutipong)
* Wiki rules and management, including git rules for managing the repository, labels, etc.
  + - Not done, Jutipong will take care of it
* Thoroughly explains the project approach and technology stack, including rationale. Provide a qualitative assessment of 3 backend, and 3 frontend frameworks (done)
* and comprehensive sprint plan with timelines,
  + Needs refinement see notes
* objectives, and dependencies
  + Needs refinements
* Comprehensive task breakdown with clear assignments, dependencies, and estimated effort

well-organized, named correctly, and include comprehensive details of each meeting.(Ivan with Ryan)

## Questions

Should we do DOD for every user story?

Do we need to have car rental selection features

## Other talking points

* ticeLog file: we need a log document and put it on the GitHub(tree)

Jutipong present how to push and pull content for the github(JUtipong)

## Action items

* Thee add user stories to github

## How to do user stories

A user story is a concise and informal way of expressing the functional requirements of a software system from an end-user's perspective. User stories are typically used in Agile development methodologies like Scrum. The elements of a user story include:

1.**Title or Card:** A brief, descriptive title or identifier for the user story, often written on a physical card or in a digital tool. For example, "As a customer, I want to log in to my account."

2.**Narrative:** A short, natural-language description that provides context and tells the story of the user's interaction with the system. It follows the format: "As a [type of user], I want [an action] so that [benefit or reason]." For example, "As a customer, I want to log in to my account so that I can access my order history."

3.**Acceptance Criteria:** Detailed conditions or criteria that must be met for the user story to be considered complete and done. Acceptance criteria are typically written in a "Given-When-Then" format, specifying the initial conditions (Given), the actions (When), and the expected outcomes (Then). For example:

1.Given that I am on the login page,

2.When I enter my valid username and password,

3.Then I should be logged in and redirected to my account dashboard.

4.**Priority:** A designation of the importance or priority of the user story. This can be expressed using labels like "high," "medium," or "low," or through a numerical scale.

5.**Estimate:** An estimate of the effort or complexity required to implement the user story. This can be done using story points, hours, or other relevant units of measurement. Estimation helps the development team plan their work during sprint planning.

6.**Definition of Done (DoD):** A set of predefined criteria that must be met before the user story is considered completed and ready for release. The DoD ensures that the work is of sufficient quality and meets the team's standards.

7.**Dependencies:** Any external factors or prerequisites that the team needs to consider while working on the user story. Dependencies may include other user stories, external systems, or data sources.

8.**Attachments:** Any supplementary materials or documents that provide additional context or details related to the user story. This could include wireframes, mockups, diagrams, or reference documents.

9.**Conversations and Notes:** Ongoing discussions and notes related to the user story. These can capture conversations between team members, clarifications, or changes in requirements that evolve over time.

10.**Owner:** The individual or team responsible for implementing and delivering the user story. This helps to assign accountability within the development team.