



# THE UNIVERSITY *of* ADELAIDE

## **Eng 3006 Software Engineering & Project**

### *Shortest Path Algorithm for Material Transportation*

## **Sprint Retrospective 1 of Group Path 12**

**SEP Group Path 12 Members:** Adam Rebes (a1793912), William Robinson (a1724943), Ayman Shaawi(a1799924), Gurvir Singh (a1796192), Stephanie Turner(a1833535), Nelson Peterson (a1801266), Murray Chahl (a1764549), Rukudzo Ndudzo (a1802422), Addy Dhingra (a1803893)

August 2023

# Snapshot

## Snapshot 1.1

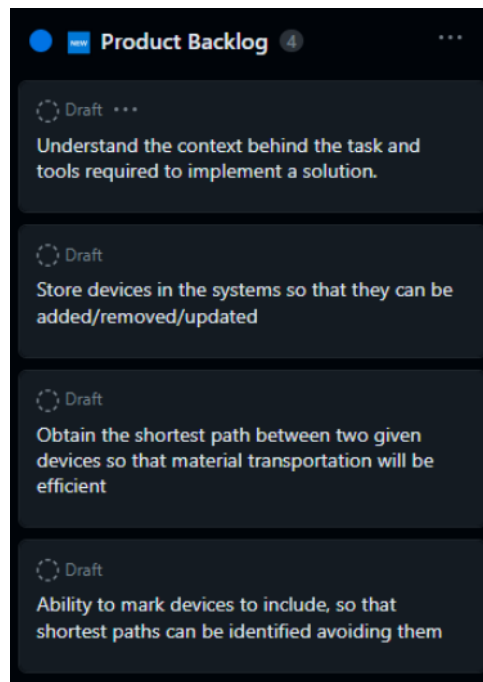


Figure 1: Product Backlog

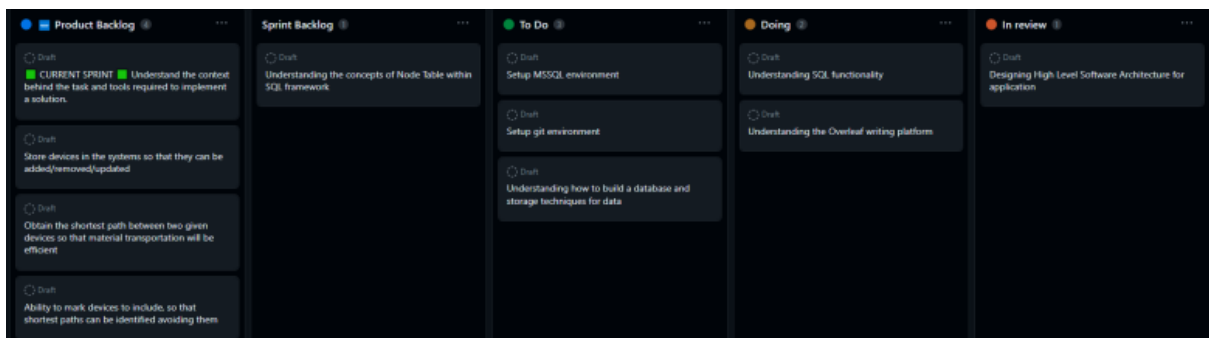


Figure 2: Task Board

## Sprint Backlog and User Stories

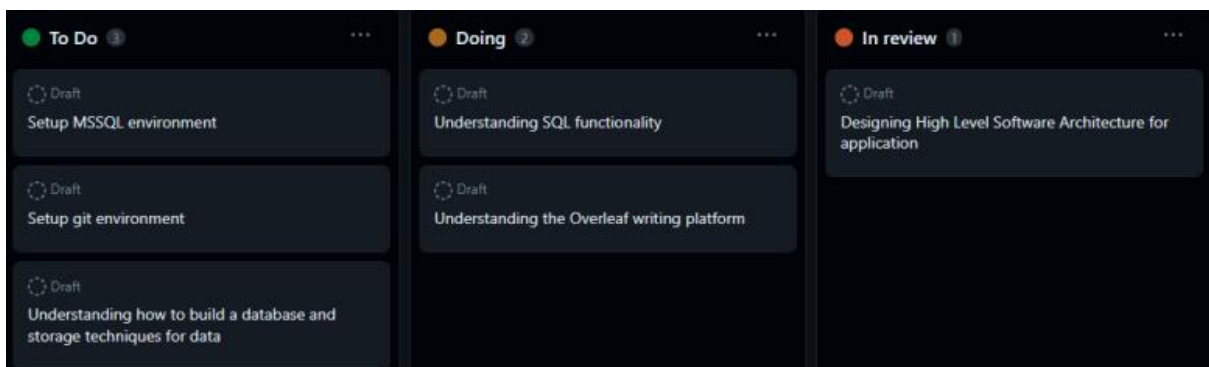


Figure 3: Sprint Backlog

For the first initial sprint, the first user story to solve is **“I want to store devices in the system so that I can add/remove/update them”**. This story was chosen first as it constructs the initial storage framework of the design which future code will need to reference and be built around. The design of adding and removing nodes will be important for future user stories, as these functions will underpin the rest of the software. The initial sub task involves the current product backlog of understanding the concepts behind the task and researching the tools required to implement the solution. This involves understanding the MSSQL programming environment and how to interact with the database to record and analyse data through the concepts of node tables. Once understood, it will give us the opportunity to store a digital map onto the database which can be used to add, remove, and update paths.

### **Definition of Done**

- If code has been written, it must follow the coding standards defined in the initial report.
- If code has been written, it must be reviewed by members to ensure proper functionality and appropriate software design, as well as its relation to the product backlog.
- If code has been written, it must pass localization testing defined by the programmer, automated testing written by the team, and automated regression testing for any future alterations to ensure correct behaviour. 12
- If code has been written, it must ensure functionality continues to work on build server, and local device.
- The created sprint task must be linked towards the product backlog which in turn must be linked towards the user story.
- Any feedback provided by the client, product owner, and scrum team should be analysed for practicality/feasibility and implemented once deemed beneficial to the user.
- For any implementation of code or system architecture, sufficient documentation is required to justify its relation with the user story, and configuration changes must be well-documented.
- The documentation and code must be deemed acceptable by the criteria given by the product owner, as well as basic software/engineering standards, and be signed off by the product owner.
- Verification and validation of key scenarios are considered to ensure basic functionality before in depth processes are considered for implementation.
- The final product must address the provided user stories to a reasonable standard, being able to at the very least produce a shortest path between two specified devices.

### **Summary of changes**

As this is the first snapshot, there are no changes to report since the previous one.

***“I attended the sprint review/planning meeting on Thursday the 24<sup>th</sup> of August with the tutor Dileepa Pitawela”***

# Sprint

## What went well?

We set up several communication channels including discord, messenger and slack, and everyone in the group was fairly easy to reach and active throughout the sprint. We were able to decide on a weekly meeting time where everyone could be present. We also setup a GitHub project where we were able to put the current user stories into the product backlog and choose the **“I want to store devices in the system so that I can add/remove/update them”** user story for the sprint and break that into tasks for the task board. We decided we would do documentation using the overleaf LaTeX editor and got ourselves accustomed with it and were able to use it to create some documents such as the snapshot and initial report. We decided on the use of MSSQL for our algorithm and watched some videos to help us learn more about how working in MSSQL and with node and edge tables might look. We also planned on how our application architecture might look and created a diagram of it.

## What could be improved?

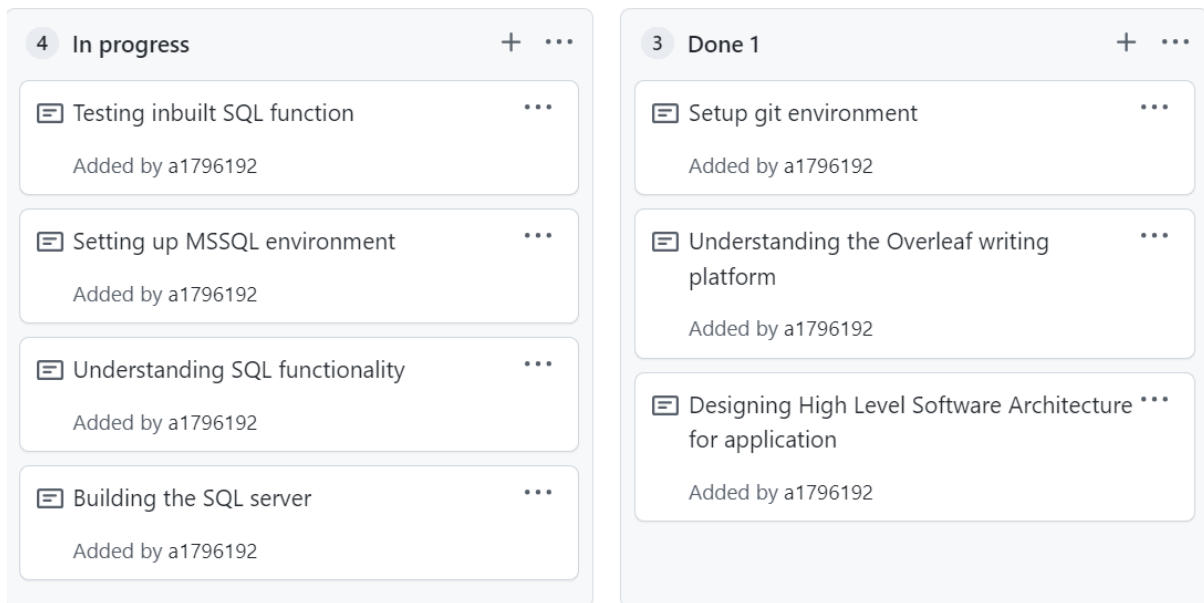
There were many things we didn't do as we wanted more information that wasn't given initially and weren't able to get clear answers to until the meeting with the product owner at the end of the sprint. We didn't know how we were going to be given user input and whether or not we were allowed to use another language such as Python to handle that and provide a user interface. We struggled to get started on writing any SQL because we weren't sure if we needed to create our own tables using input which we could get via user input using an interface creating in another programming language or if we were given input in the form of database tables. It may have been better for us to create some tables in a form efficient for the algorithm and then adapt the data in those tables regardless of how we receive input, however we weren't confident enough with MSSQL and perhaps should have spent more time learning about it.

## What will the group commit to improve in the next sprint?

For the next sprint we've changed the user story for the sprint from **“I want to store devices in the system so that I can add/remove/update them”** to the **“As a user, I want to get the shortest path between 2 given devices so that material transportation will be efficient.”** user story after our meeting with the tutor as we realised that it's too specific of a user story and that although there's significant overlap, all the tasks that we are currently doing are tasks of a broader scope for the purpose of ultimately determining the shortest paths between 2 devices; also it makes more sense to focus on our shortest paths algorithm before focusing on allowing the user to make changes to devices.

In the next sprint, the focus will be heavily on SQL, including: having the MSSQL environment and server set up, determining how tables will be set out and getting some basic node and edge tables in place, and testing out SQL functions and queries in the environment. We will also spend some more time thinking about the input interface and start on building that. We could also try to be more organized in how we split up tasks to improve efficiency, something which wasn't as relevant in the first sprint because it was mainly about brainstorming, learning, and setting up our own respective environments.

## Sprint Progress



We haven't been using the issue feature, or started on any code as of yet, instead focusing on the GitHub task board.

I've contributed to discussions during meetings regarding the layout of the database and tables and what sort of questions we'll need to ask the tutor in order to decide how we handle user input and build our tables. I worked with two other students in the group to design and justify the diagram for our application architecture which was also part of the initial report created using LaTeX, also contributing to some definitions of done for the snapshot. I watched a few videos about MSSQL, downloaded the MSSQL developer edition, and tried some basic queries.