

Retrospective Sprint <2> of Group <Path5>

ATSYS_Shortest Path Algorithm for Material Transportation

Shize Liu a1844323

Yuze Li a1848890

Ruoyu Xiong a1847649

Yuejun Zhao a1829813

Shijie Zhang a1809881

Yuchen Peng a1824982

1. What went well in the sprint?

During this sprint, one standout quality has been our team's quick responsiveness and strong cohesion. These qualities enable us to adjust rapidly to changes and stay calm in a tense situation. For example, because we didn't check Slack messages frequently, we initially missed the requirements change. We were falling behind as it took us a while to see the message, but once we did, everyone responded quickly and positively. No one was in panic. Instead, we supported each other and communicated calmly. We quickly arranged an urgent meeting on the following day, discussed the changes, and made necessary adjustments. This rapid response within our group prevents us from falling further behind in our progress and ensures we can catch up quickly. The strong cohesion trait of our group also allows us to communicate peacefully with each other even when the team encounters problems. These qualities help reduce the negative impacts on the software development process when unexpected things occur, we quickly adjust our goals and catch up the progress through communication.

2. What could be improved?

One thing that could be improved is the way we manage GitHub issues. Currently, our task board seems disorganised, and the tasks are not well structured and divided. From my point of view, the issue titles are often unclear and lacking in detail, for example, there is an issue title '(Req Change) Update Algorithm' and my thoughts would be what is the req change? Which user story does it relate to? What algorithm needs to be updated and how? This is a clear evidence how our issue titles are awfully written. This problem will create challenges for team members to understand the tasks. The lack of clarity in titles can lead to confusion, as team members may not have a clear idea of current progress and what needs to be addressed. This can result in misunderstandings on the software and slow down the development process. More importantly, because of how vague and uncertain our tasks are, team members might come with different understandings about the tasks and prevent them from efficiently collaborate with each other.

3. What will the group commit to improve in the next sprint?

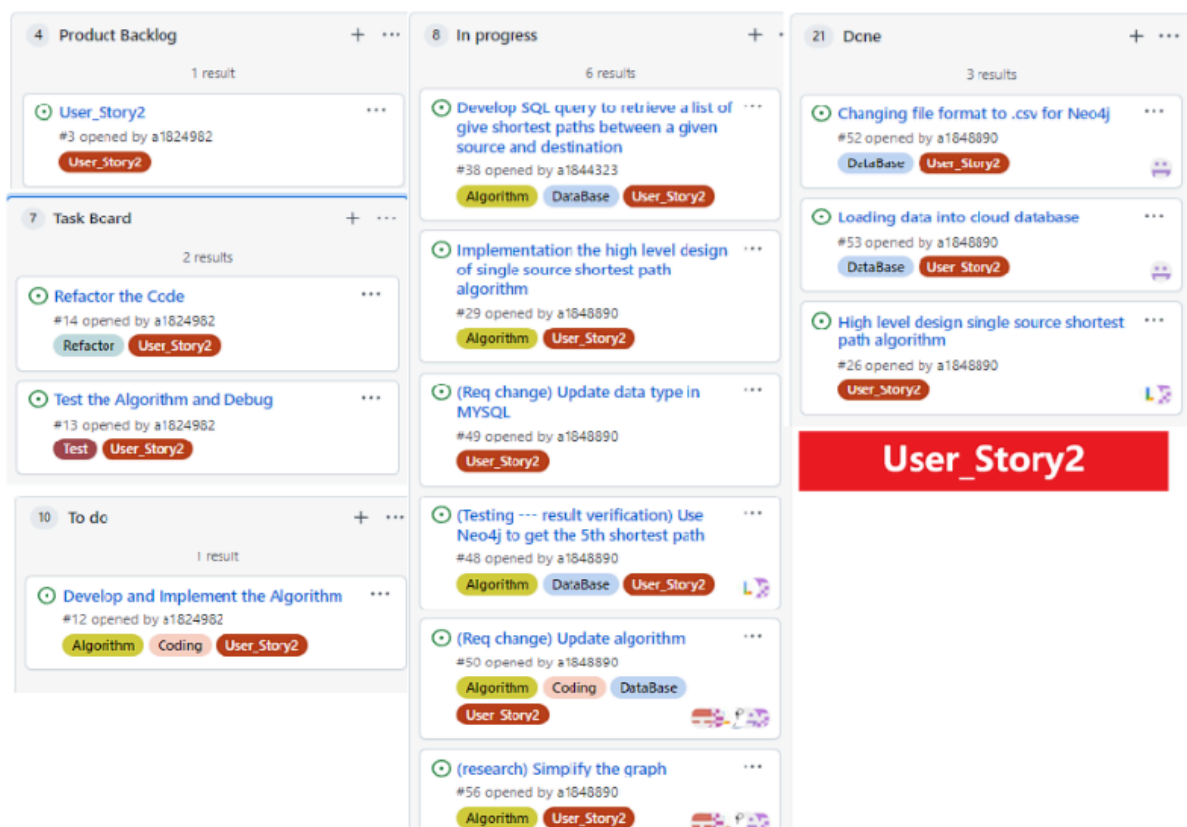
- More frequently check on Slack messages

In the upcoming sprint, we'll be increasing our frequency in checking Slack messages, ensuring that we receive updates in a timely manner. This approach is meant to improve the communication awareness with the product owner. By doing this, we will be well-prepared next time, respond quickly and on time. This allows us more time to carefully consider, discuss, and develop the right software solutions.

- Better organisation on GitHub issues

In the next sprint, we will fix the Github task boards. By providing clearer titles and more detailed issue descriptions, our software development process will be improved in the aspects of collaboration, communication and accountability. A well organised task board promotes a shared and common understanding of project goals which can reduce the likelihood of duplicated efforts, missed tasks and foster collaborations. Furthermore, detailed descriptions of tasks will make it easier for scrum master to assign and track responsibilities among the group. A better management of task board would enable each team members to have a clear road map of how is our software developed, making it easier to meet deadlines and maintain accountability.

4. Comment on your progress this sprint (Individually Written)



Written Snapshots

- Captured project milestones
- The complexity of this task is not high

Managed Github task board

- Managed progress and tasks updates

- Created new issues when changes/new requirements arise
- The complexity of this task is harder than I thought because it requires a deep understanding of the whole project in order to divide tasks well. I need to have a clear roadmap of how our project or software is built, considering from different aspects such as technical issues, team member capabilities, etc. This is the part I did not do well, and I will improve in the next sprint

Organised meetings

- Contacted team members for attending meetings
- Our team responds quickly, so it is not difficult to organise a meeting

Researched how to generate the path diagram(graph)

- Researched methods to generate the graph rather than manually drawing it since the number of devices (nodes) and edges becomes very large
- Python and Neo4j are being taken into consideration
- It is a relatively challenging task because it takes a long time to learn how to automatically generate graphs using Python and Neo4j and also evaluate which method is more suitable for our situation.

I attended the sprint review/planning meeting on the 22nd of August, 5th of September with the tutor.

Snapshot Week <5> of Group <Path5>

Project: ATSYS_Shortest Path Algorithm for Material Transportation

Members:

Shize Liu_a1844323

Yuze Li_a1848890

Ruoyu Xiong_a1847649

Yuchen Peng_a1824982

Yuejun Zhao_a1829813

Shijie Zhang_a1809881

Product Backlog and Task Board	2
Sprint Backlog and User Stories	3
Definition of Done	4
Summary of Changes	4

Product Backlog and Task Board

Product Backlog

4

Product Backlog

+

...

User Story 1

#2 opened by a1824982

User_Story1

User_Story2

#3 opened by a1824982

User_Story2

User_Story3

#4 opened by a1824982

User_Story3

As a user, I want to store devices in the system so that I can add/remove/update them.

As a user, I want to get the shortest path between 2 given devices so that material transportation will be efficient.

As a user I want to mark devices to exclude, so that shortest paths can be identified avoiding them.

Task Board for Sprint 2 Snapshot 2.1 (User Story2)

To Do

In Progress

Done

5

To do

+

...

2.6) Set up Cloud Neo4j Database and Load Files into it

#53 opened by a1848890

DataBase

User_Story2

2.7) Implement Functionality to Draw Graphs in Neo4j Based on Input Files

#61 opened by a1824982

DataBase

User_Story2

2.8) Update the Shortest Path Algorithm in Neo4j Based on Requirement Change

#50 opened by a1848890

Algorithm

Coding

DataBase

User_Story2

2.9) Test/Debug the Algorithm/Current Script

#48 opened by a1848890

Algorithm

DataBase

User_Story2

2.10) Snapshot 2.1 and 2.2

#62 opened by a1824982

Document

User_Story2

2

In progress

+

...

2.4) Update Data Type in the Database Schema Based on Requirement Change

#60 opened by a1824982

DataBase

User_Story2

2.5) Research on Visualising the Graph with Large Inputs

#52 opened by a1848890

DataBase

User_Story2

14

Done

+

...

2.3) Refactor the Code

#14 opened by a1824982

Refactor

User_Story2

2.2) Test/Debug the Algorithm

#13 opened by a1824982

Test

User_Story2

2.1) Implement Shortest Path Algorithm

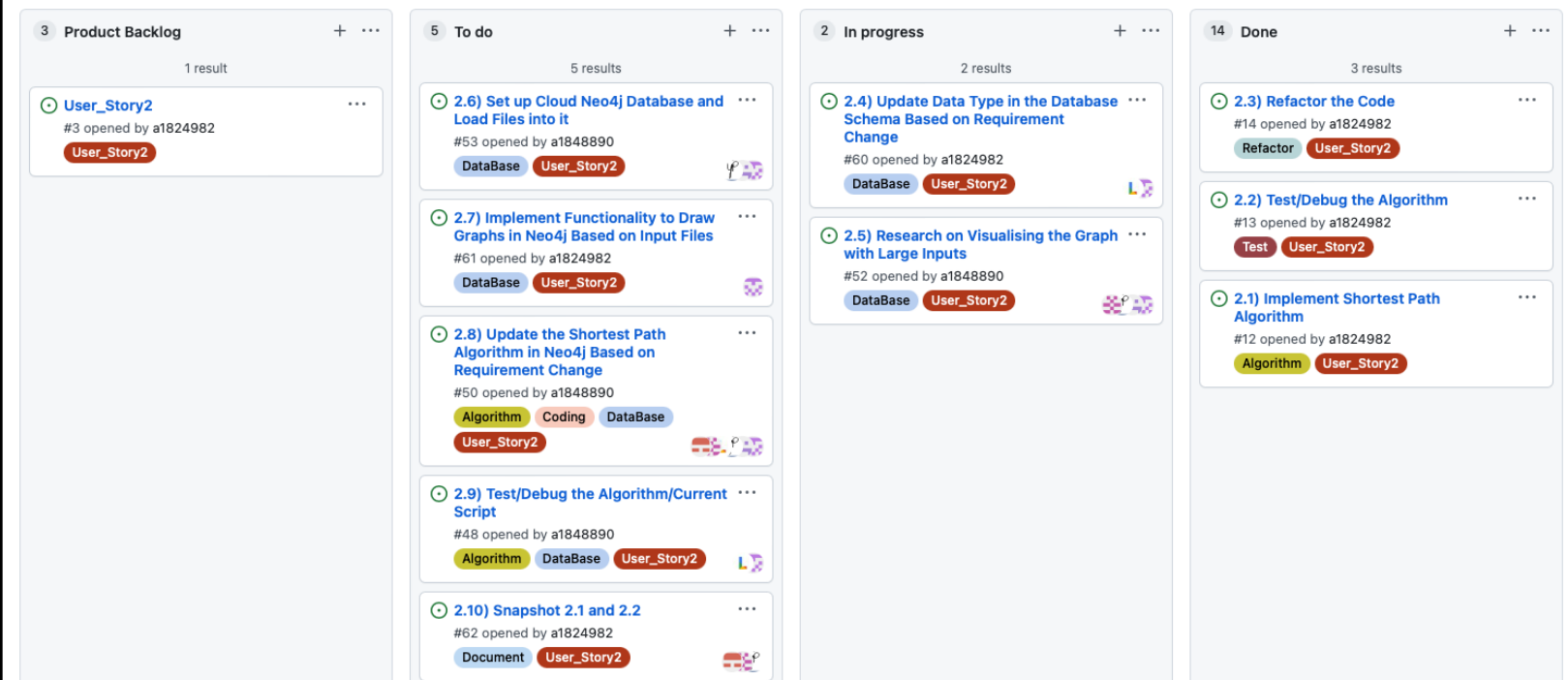
#12 opened by a1824982

Algorithm

User_Story2

Sprint Backlog and User Stories

Sprint Backlog



In the second sprint, the user story our group is working on is: ***‘As a user, I want to get the shortest path between 2 given devices so that material transportation will be efficient’.***

In this user story, users are requesting a method to determine the top 5 efficient routes between two given devices. This functionality will enable them to effectively and promptly manage the transportation between their plants. The purpose is to increase the overall productivity and cost efficiency.

An algorithm needs to be designed to accomplish this requirement. This algorithm should calculate the cost of paths between the selected devices. It will make a recursion approach to ensure that every device is visited, all routes are explored before calculating the cost and sorting them in order.

Definition of Done

- A coding task is considered to be completed when the code has been written in accordance with the coding standards outlined in the initial report, tested (both unit and integration) refactored as needed, successfully passed peer review and obtained approval from all members of the team.
- A non-coding task task is considered to be completed when it has been brainstormed, discussed, documented, reviewed and agreed upon by the team in a meeting to ensure everyone is aligned and informed about the task. Additionally, any specific problems that arose during the Sprint should be reported to the team in detail and converted to an issue on the GitHub task board.

Summary of Changes

During this week's snapshot, we have implemented several changes and updates to the project. These changes mainly focus on moving from user story 1 to user story 2, implementing the functionality of finding the shortest path as required in user story 2. Highlights include:

1. Finished the testing for codes/scripts written in user story1.
2. Implemented the algorithm to find the shortest path by using SQL commands.
3. Tested the scripts in the scenario stimulated in user story1.
4. Refactored the commands.
5. Updated everything on Github Taskboard.

Overall, these are the main changes our group made between the previous snapshot.

Snapshot Week <6> of Group <Path5>

Project: ATSYS_Shortest Path Algorithm for Material Transportation

Members:

Shize Liu_a1844323

Yuze Li_a1848890

Ruoyu Xiong_a1847649

Yuchen Peng_a1824982

Yuejun Zhao_a1829813

Shijie Zhang_a1809881

Product Backlog and Task Board	2, 3
Sprint Backlog and User Stories	4
Definition of Done	5
Summary of Changes	5

Product Backlog and Task Board

Product Backlog		
User Story 1	User Story 2	User Story 3
As a user, I want to store devices in the system so that I can add/remove/update them.	As a user, I want to get the shortest path between 2 given devices so that material transportation will be efficient.	As a user I want to mark devices to exclude, so that shortest paths can be identified avoiding them.

Task Board for Sprint 2 Snapshot 2.2 (User Story2)	
To Do	In Progress
<div> <div>1 To do</div> <div>+</div> <div>...</div> <div>1 result</div> <div> <div>2.10) Snapshot 2.1 and 2.2</div> <div>...</div> <div>#62 opened by a1824982</div> <div>Document User_Story2</div> <div></div> </div> </div>	<div> <div>1 In progress</div> <div>+</div> <div>...</div> <div>1 result</div> <div> <div>2.9) Test/Debug the Algorithm/Current Script</div> <div>...</div> <div>#48 opened by a1848890</div> <div>Algorithm DataBase User_Story2</div> <div></div> </div> </div>

Task Board for Sprint 2 Snapshot 2.2 (User Story2)

Done

2.4) Update Data Type in the Database Schema Based on Requirement Change ...

#60 opened by a1824982

DataBase

User_Story2



2.3) Refactor the Code ...

#14 opened by a1824982

Refactor

User_Story2

2.2) Test/Debug the Algorithm ...

#13 opened by a1824982

Test

User_Story2

2.1) Implement Shortest Path Algorithm ...

#12 opened by a1824982

Algorithm

User_Story2

2.8) Update the Shortest Path Algorithm in Neo4j Based on Requirement Change ...

#50 opened by a1848890

Algorithm

Coding

DataBase

User_Story2



2.7) Implement Functionality to Draw Graphs in Neo4j Based on Input Files ...

#61 opened by a1824982

DataBase

User_Story2



2.6) Set up Cloud Neo4j Database and Load Files into it ...

#53 opened by a1848890

DataBase

User_Story2



2.5) Research on Visualising the Graph with Large Inputs ...

#52 opened by a1848890

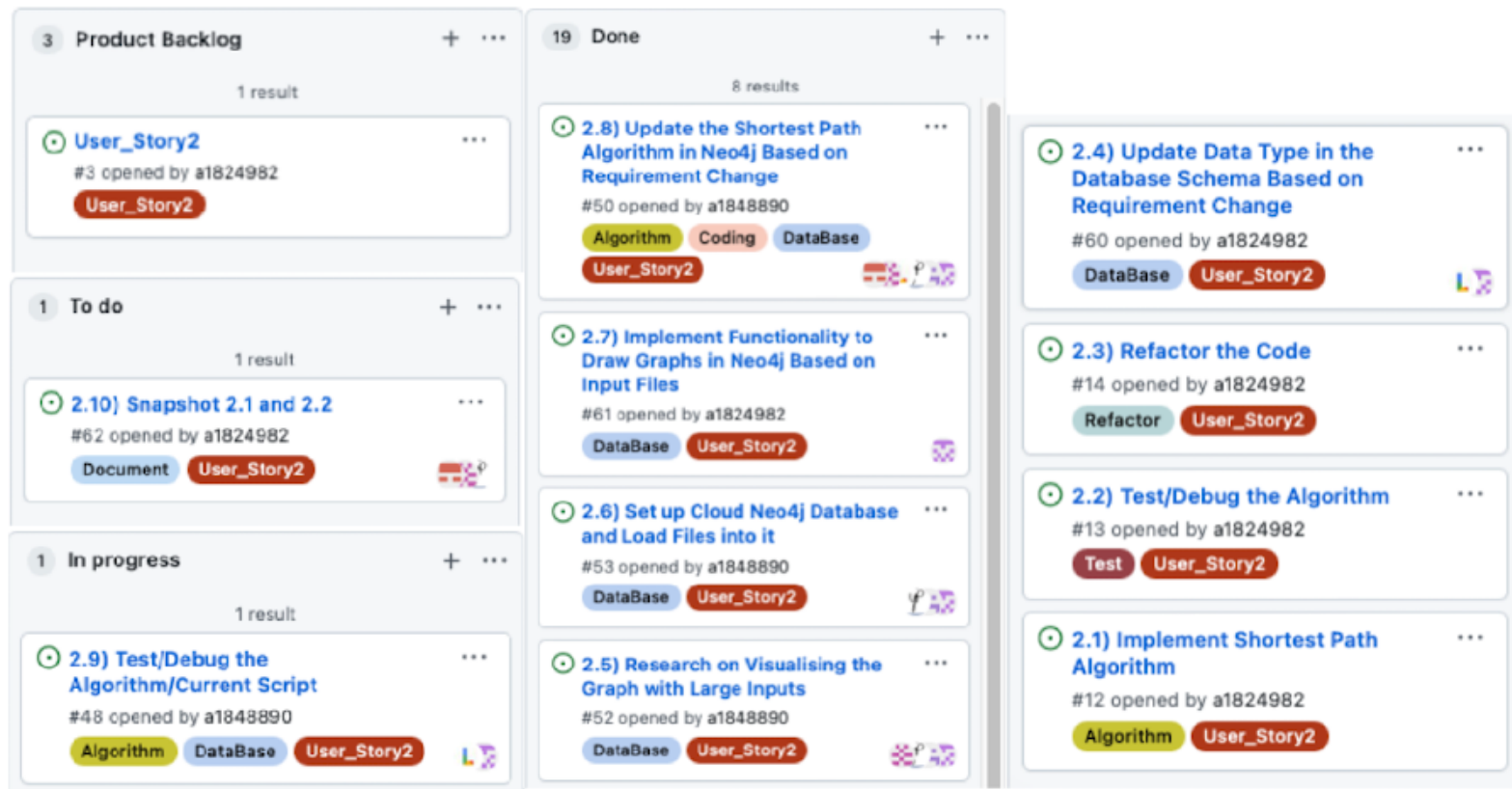
DataBase

User_Story2



Sprint Backlog and User Stories

Sprint Backlog



In the second sprint, the user story our group is working on is: ***‘As a user, I want to get the shortest path between 2 given devices so that material transportation will be efficient’.***

In this user story, users are requesting a method to determine the top 5 efficient routes between two given devices. This functionality will enable them to effectively and promptly manage the transportation between their plants. The purpose is to increase the overall productivity and cost efficiency.

An algorithm needs to be designed to accomplish this requirement. This algorithm should calculate the cost of paths between the selected devices. It will make a

recursion approach to ensure that every device is visited, all routes are explored before calculating the cost and sorting them in order.

Definition of Done

- A coding task is considered to be completed when the code has been written in accordance with the coding standards outlined in the initial report, tested (both unit and integration) refactored as needed, successfully passed peer review and obtained approval from all members of the team.
- A non-coding task is considered to be completed when it has been brainstormed, discussed, documented, reviewed and agreed upon by the team in a meeting to ensure everyone is aligned and informed about the task. Additionally, any specific problems that arose during the Sprint should be reported to the team in detail and converted to an issue on the GitHub task board.

Summary of Changes

During this week's snapshot, we have implemented many changes and updates to the project. Our primary focus was addressing the requirements change from the product owner. Highlights include:

1. Updated screen shots in Product Backlog and Sprint Backlog to make it more clearer to see and easier to understand.
2. Updated everything on Github Taskboard.
3. Moved from Mysql Database to Neo4j Cloud Database.
 - a. Implemented commands to read in csv files.
 - b. Able to generate path diagrams based on inputs provided by PO(tutor).
 - c. More convenient when inputs are large.
4. Updated the Mixed attribute in our device table.
5. Updated the Cost attribute in our edge table.

Overall, these are the main changes our group made between the previous snapshot.