

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	3
Sprint Backlog and User Stories	5
Definition of Done	6
Summary of Changes	7

Product Backlog and Task Board

3

Product Backlog

+

...

3 results

🕒

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

7

Task Board

+

...

7 results

🕒

Write Simple to Complex Test Cases

...

#11 opened by a1824982

Test

🕒

Refactor the Code

...

#14 opened by a1824982

Refactor

User Story2

🕒

Implement Exclusion Functionality

...

#15 opened by a1824982

Coding

User Story3

🕒

Integrating new Functionality with Current Algorithm

...

#17 opened by a1824982

Algorithm

Coding

User Story3

🕒

Refactor the Code

...

#19 opened by a1824982

Refactor

User Story3

🕒

Write Simple to Complex Test Cases

...

#16 opened by a1824982

Test

User Story3

🕒

Test the Algorithm and Debug

...

#13 opened by a1824982

Test

User Story2

3

Product Backlog

+

...

1 result

🕒

User Story 1

...

#2 opened by a1824982

User Story1

7

Task Board

+

...

0 results

5

To do

+

...

1 result

🕒

Validation and Refinement

...

#10 opened by a1824982

Refactor

Test

User Story1

3

In progress

+

...

2 results

🕒

Optimize Operations using SQL Functions

...

#9 opened by a1824982

Refactor

User Story1

🕒

Implement and Test Basic Operations

...

#8 opened by a1824982

Coding

User Story1

15

Done

+

...

10 results

🕒

Research different method of setting up mysql database(Setup)

...

#28 opened by a1847649

User Story1

🕒

Define Device&Edge Table Schema(Data Plan)

...

#6 opened by a1824982

User Story1

🕒

Learn about MSSql(Setup)

...

#21 opened by a1847649

User Story1

🕒

Environment selection(Setup)

...

#20 opened by a1809681

User Story1

🕒

Environment Setup(Setup)

...

#5 opened by a1824982

User Story1

🕒

Selection of variables(initial designing)

...

#23 opened by a1809681

Algorithm

User Story1

🕒

Initial database schema(initial designing)

...

#25 opened by a1809681

User Story1

🕒

Create the Database Based on Data Plan(initial designing)

...

#7 opened by a1824982

Coding

User Story1

🕒

Simulate a scenario

...

#32 opened by a1847649

User Story1

🕒

Store graph information in the table

...

#22 opened by a1840890

User Story1

3Product Backlog+...

1 result

User_Story2

#3 opened by a1824982

User_Story2

7Task Board+...

2 results

Refactor the Code

#14 opened by a1824982

RefactorUser_Story2

Test the Algorithm and Debug

#13 opened by a1824982

TestUser_Story2

5To do+...

1 result

Develop and Implement the Algorithm

#12 opened by a1824982

AlgorithmCodingUser_Story2

3In progress+...

1 result

Implementation the high level design of single source shortest path algorithm

#29 opened by a1848890

AlgorithmUser_Story2

15Done+...

1 result

High level design single source shortest path algorithm

#25 opened by a1848890

User_Story2

3Product Backlog+...

1 result

User_Story3

#4 opened by a1824982

User_Story3

7Task Board+...

4 results

Implement Exclusion Functionality

#15 opened by a1824982

CodingUser_Story3

Integrating new Functionality with Current Algorithm

#17 opened by a1824982

AlgorithmCodingUser_Story3

Refactor the Code

#19 opened by a1824982

RefactorUser_Story3

Write Simple to Complex Test Cases

#16 opened by a1824982

TestUser_Story3

5To do+...

3 results

Test the Algorithm and Debug

#18 opened by a1824982

TestUser_Story3

Add checking condition if a path is available

#31 opened by a1848890

AlgorithmCodingDatabaseUser_Story3

Trigger the state of a device

#30 opened by a1848890

DatabaseUser_Story3

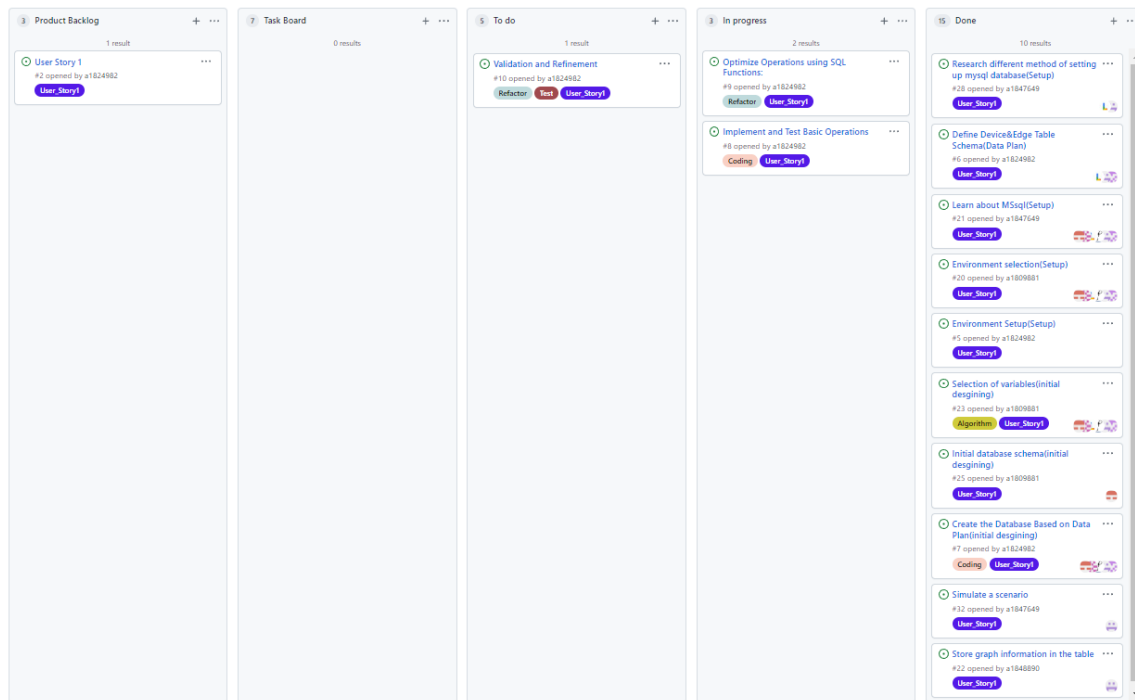
3In progress+...

0 results

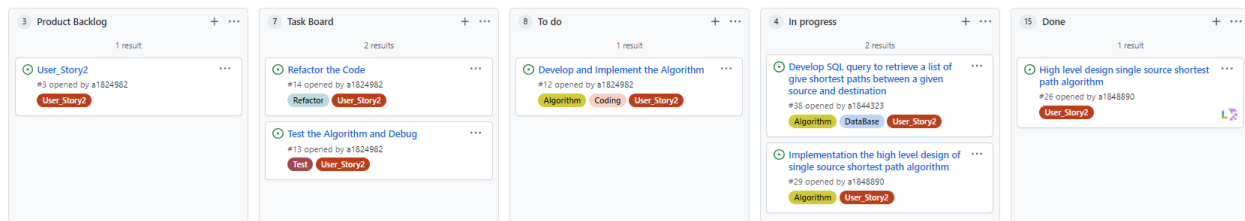
15Done+...

0 results

Sprint Backlog and User Stories



User Story(2): *“As a user, I want to get the shortest path between 2 given devices so that material transportation will be efficient.”*



https://github.cs.adelaide.edu.au/SEP23S2PATH/PATH_5/projects/1?card_filter_query=label%3A+label%3Auser_story2

In this user story, users are requesting a method to determine the top 5 efficient routes between two specified devices. This feature will assist them in effectively and promptly managing their plants resulting in increased productivity and cost efficiency.

To fulfill this requirement we have implemented an algorithm that utilizes a MySQL database. The algorithm calculates the cost of the path between the selected devices. It uses recursion to ensure that every device is visited and

explores all routes before sorting them in ascending order. The algorithm looks at where the devices are, if devices are in use(not available) or not, and other stuff to find the most cost efficient way.

The user specifies the starting and ending node and our implemented algorithm in MySQL returns the 5 paths that're cost efficient. Eventually this application could provide users with the ability to manage devices and plants in a cost manner.

Definition of Done

- A coding task is considered complete when the code has been written in accordance with the coding standards outlined in the report reviewed, 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 assignment is considered complete when it has been reviewed, discussed, documented and agreed upon by the team in a meeting to ensure everyone is on the page. Additionally any specific problems should be reported in detail using our project page, on Github.

Summary of Changes

Github task board wise:

We reviewed the existing user stories. Included additional detailed tasks, in the “To do” section. We also went through and added more specific tasks. Additionally we completed some tasks in the “In progress” and moved them to the "Done" section.

Code wise:

- We plan to implement the shortest path algorithm in SQL, the algorithm is expected to return the shortest path between the start Node and the destination Node. In the coming sprints we plan to combine the SQL recursive CTE method and this algorithm to implement the shortest path algorithm that returns the top-5 shortest path.
- We plan to implement the test for user stories 2. This involves creating graphs, designing tables and conducting algorithm testing.