## COMP3111: Introduction to Software Engineering

### Minutes of the 2nd Project Meeting

**Date**: 2023/10/13 (Friday)

**Time**: 17:00

**Venue**: Learning Commons/ LG1

**Attending**: Law Hui Nok (Tim) , Lam Hoi Yi (Kelly) , Lo Wing Yan (Kelly)

**Absent**: N/A

**Recorder**: Lo Wing Yan (Kelly), Law Hui Nok (Tim)

### 1. Report on process during the previous week

| Name | Tasks worked on in the previous week |
|---|---|
| Law Hui Nok | Completed the Class Diagram and Use Case specification for Function B (1st Draft) |
| Lam Hoi Yi | Completed the Class Diagram and Use Case specification for Function C (1st Draft) |
| Lo Wing Yan | Completed the Class Diagram and Use Case specification for Function A (1st Draft) |

### 2. Discussion of building Data Modeling

2.1. Tim clarified the confusion regarding building Data Modeling after the discussion with TA on 2023/10/12 (Thursday) (Details please refer to Appendix).

### 3. Goals for the coming weekend

| Name | Tasks that will be worked on in the coming weekend |
|---|---|
| Law Hui Nok | Modify the Class Diagram and Use Case specification for Function B |
| Lam Hoi Yi | Modify the Use Case specification for Function C |
| Lo Wing Yan | Modify the Use Case specification for Function A |

### 4. Meeting adjournment and next meeting

The meeting was adjourned at 19:00. The next meeting is not scheduled.

## Appendix [Note for discussion with TA]

**Note for Class Diagram:**

**The changes & explanations**

Q: Tom/Jerry/Crystal Location is an object or coordinate?

A: Location = object --> belong to moving object

Q: Entry/Exit Point is a kind of vertex location?

A: No, it can only be on top of clear vertex --> associate + XOR

Q: The definition of Possible Path?

A: One clear vertex can also be treated as possible path even if one clear vertex which is surrounded and trapped by barriers will also be treated as a path. --> Multiplicity changes to 1 .. *

**About the multiplicity between possible path and moving object:**

1. One moving object can have at least one possible path since it must stand on one clear vertex(can belong to one path or more that one path)

2. One possible path have one moving object. {(0,0), (0,1), (0,2)} is a path starts from (0,0), {(0,2), (0,1), (0,0)} is treated as different since the path is started at (0,2). So, one possible path can only have one moving object, otherwise the game will terminate.

Meetings Minutes #2

**Notes for use case specification:**

1. It is better for us to add the diagram of each part of the use case (copy the diagram from requirement note)

2. The different between assumption, precondition and postcondition:

Assumption: Some situations that need to be considered during the entire function call (conditions that should be existed before & during & after the function)

Precondition: The conditions require so that the function can be called. After the function called, the condition can be changes of does not exist anymore.

Postcondition: The conditions require when the function return

--> Some assumption & pre & post condition need to be changed belong to the precise definition.

3. For use case 1, Option 1 or Option 2 is decided by ourselves. The player is not able to choose.

4. Use case 1, CSV file generation is required in specification.

5. For se case 2, Actor = player, the basic flow structure is liked

```
void funB(){
    if (game start){
        Use BFS algo;

        explore possible path;

        Tom follow the shortest path to catch Jerry;
    }
}
```

-->The flow need to expand more based on the above

6. The use case is how the system behave --> in use case = need to be implemented. The features and behaviours can be decided freely