Table Content

Part 1.0: Analysis and Design

1.1 Problem Statement:

- Using OOAD methodology and analyze, design and implement a 2-Dimensional map simulator that involves prey and predator behaviour.

1.2 Requirements of this simulator

- The map size should be a 20x20 array list in a total of 400 sizes of an array.
- The simulator program should be in GUI mode.
- The map should have bugs as the predator and ants as the prey.
- Both organisms can move randomly on the map.
- The simulator should implement predator consumes the prey.
- Both organisms can breed with steps. For the predator every 8 steps, each predator may breed a new predator on the map while for the prey every 3 steps, each prey may breed a new prey on the map.
- After a predator consumes a prey, the predator will have 3 steps before the predator has to consume another prey. Failing to consume after the given steps will lead to the elimination of the specific predator on the map.
- The user may click a button to start viewing the simulation on the map.

Part 2.0: Bug and Ants Documentation

2.1 Design Pattern

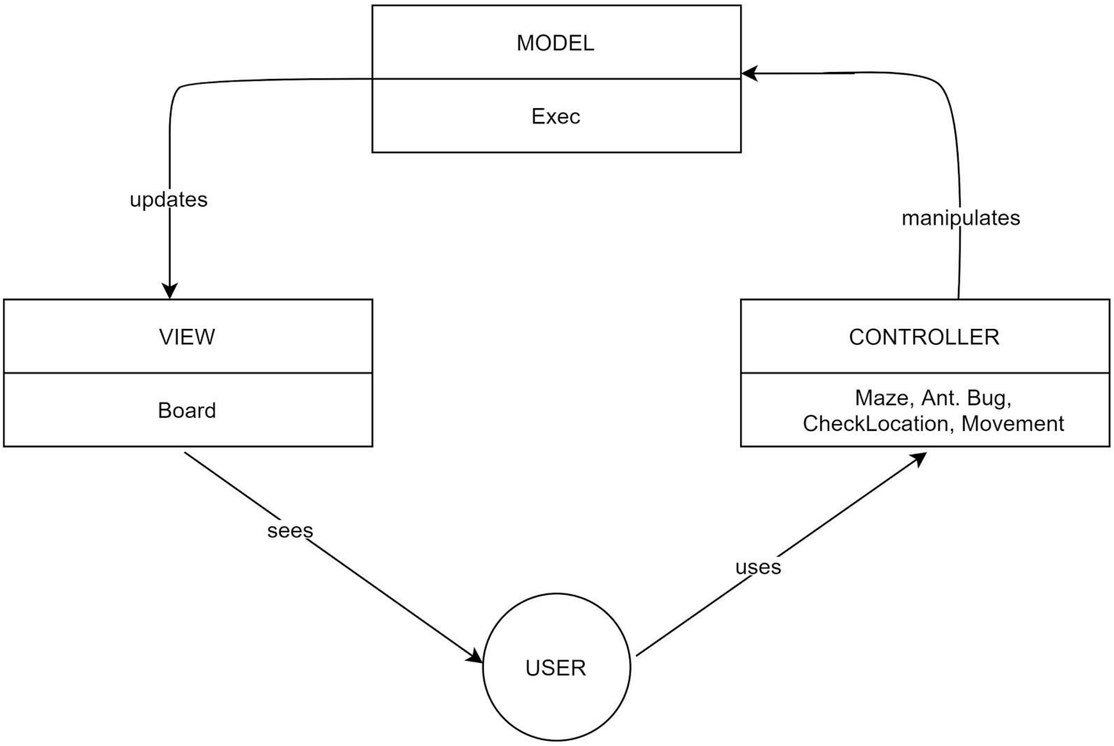
Design Pattern that we used for this project of bug and ants simulator is Model View Controller.

- Model: Contains data of the program.
- View: Present the model data to the viewer.
- Controller: It listens to the events triggered by the view to execute appropriate reactions to the events.

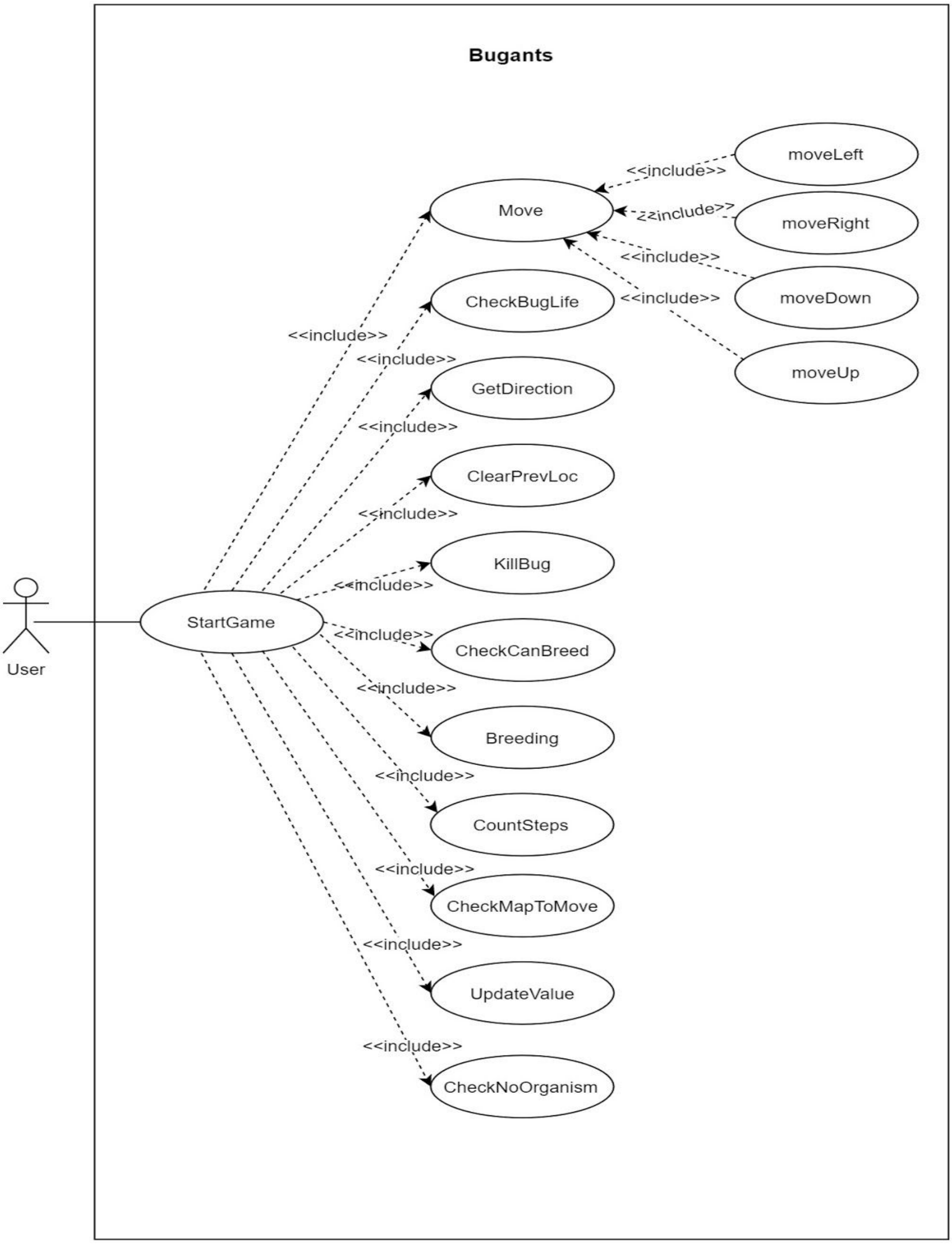
Model View Controller design pattern did solve problems like:

- Allow multiple developers to work together simultaneously.
- Enables logical grouping for the related actions on a controller.
- Allows more control to the program which also decreases the redundancy of codes.

Model View Controller Design Pattern

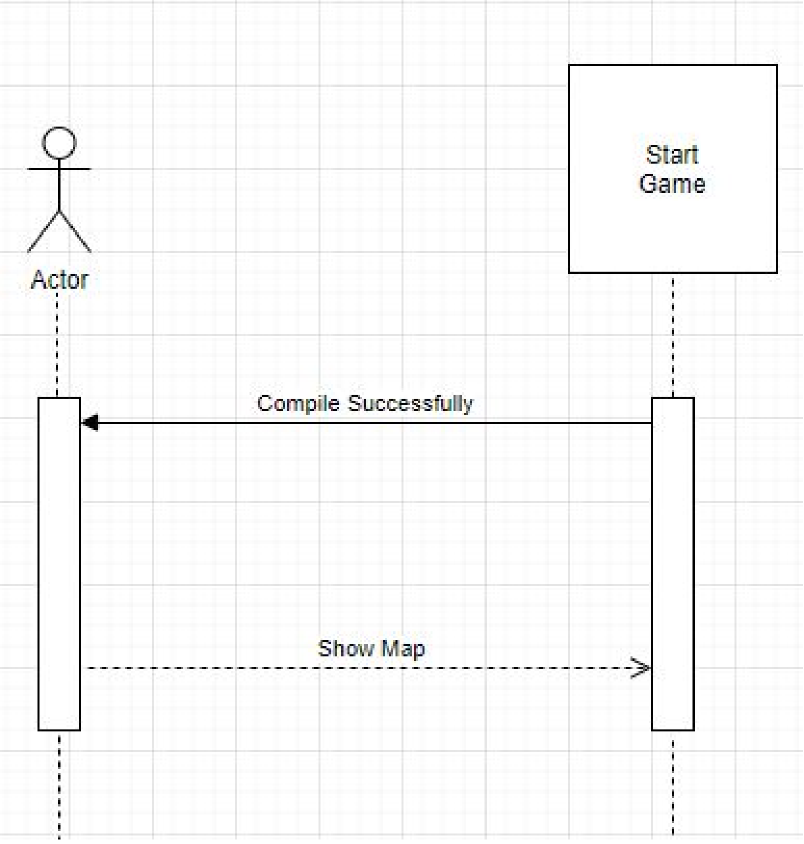


2.2 Use Case Diagram

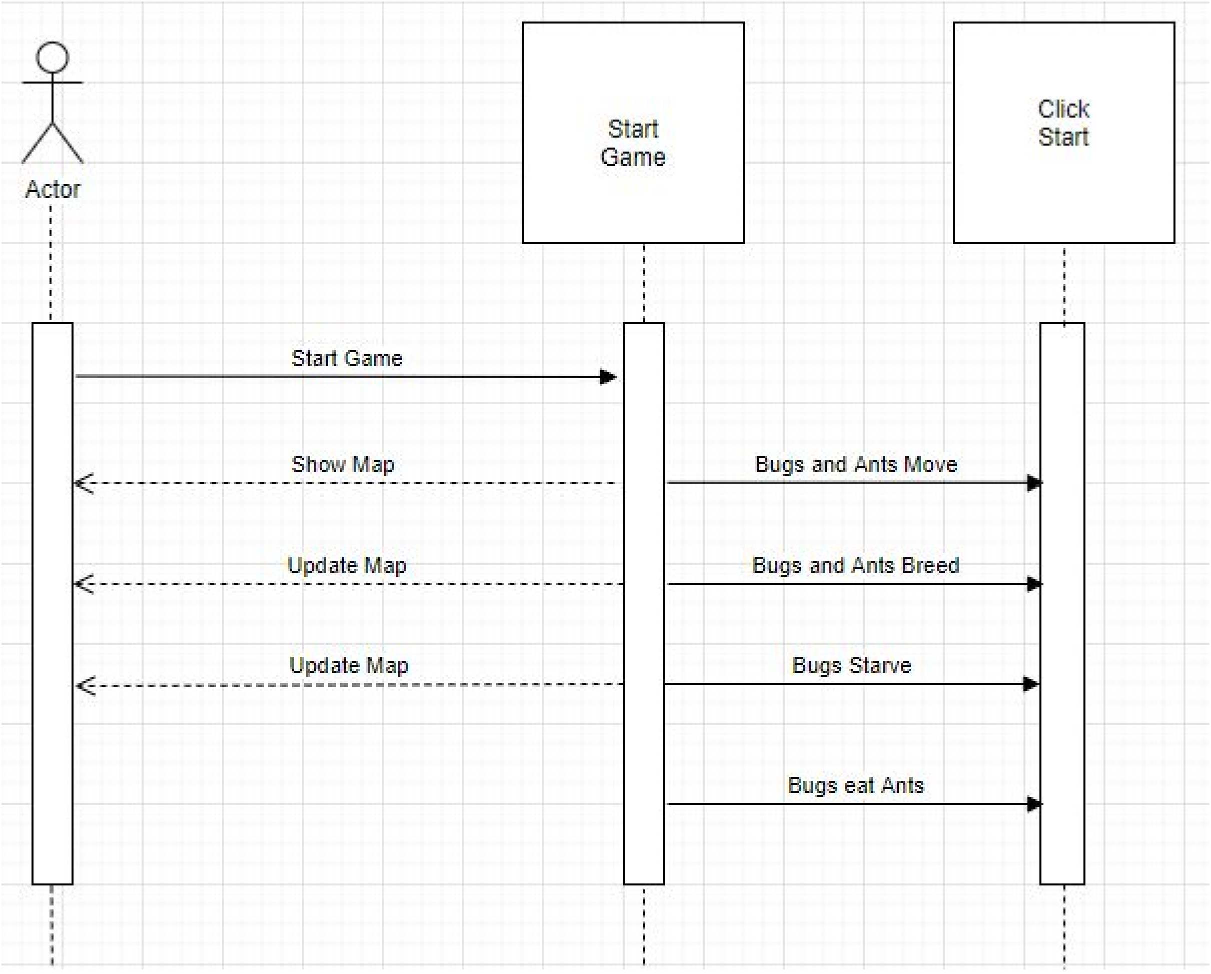


2.3 Sequence Diagram

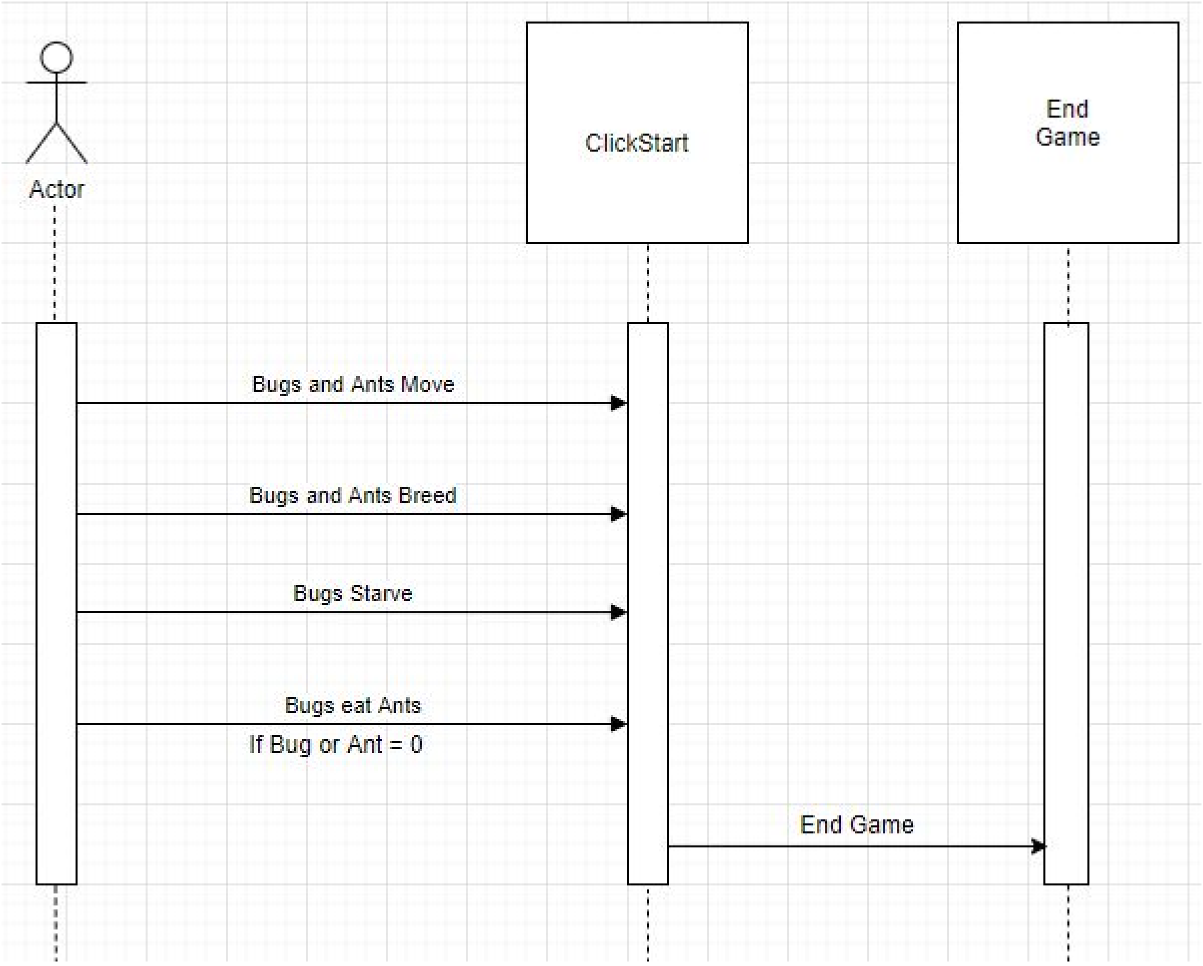
2.3.1 Start Game



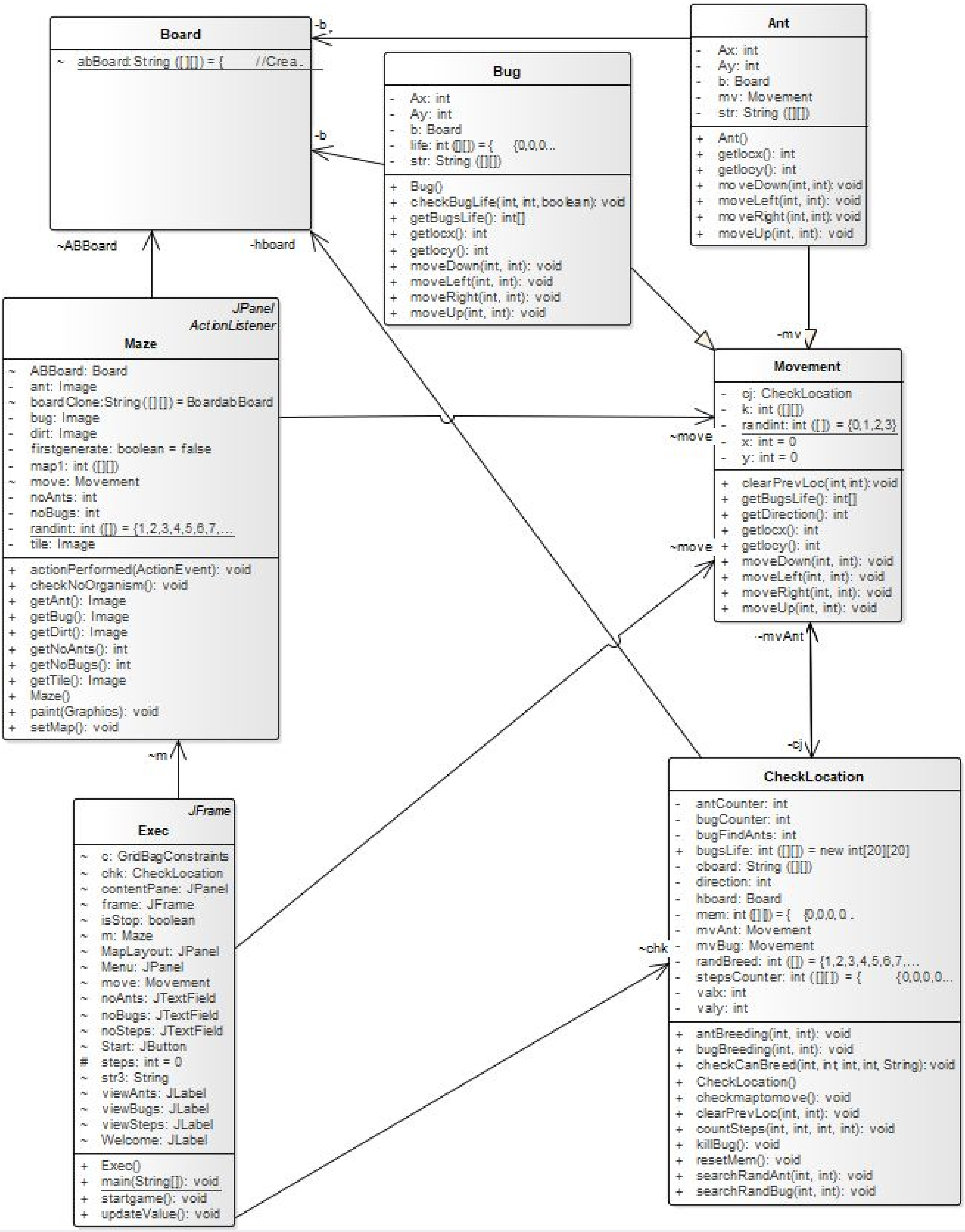
2.3.2 Click Start



2.3.3 End Game


Actor
ClickStart
End Game
Bugs and Ants Move
Bugs and Ants Breed
Bugs Starve
Bugs eat Ants
If Bug or Ant = 0
End Game

2.4 Class Diagram



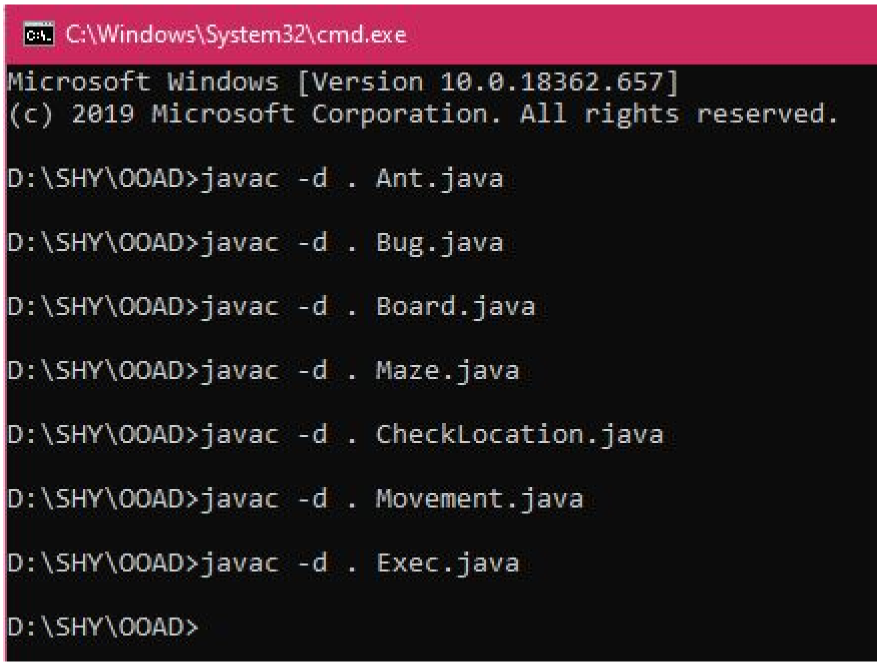
2.5 User Documentation

How to Launch the Simulation.

1. Open OOAD folder. There will be 7 java files and 4 image files.
2. Open the command prompt.
3. Change directory to the OOAD folder.
4. To compile the files: (1) Run **javac -d . Ant.java**
   (2) Run **javac -d . Bug.java**
   (3) Run **javac -d . Board.java**
   (4) Run **javac -d . Maze.java**
   (5) Run **javac -d . Movement.java**
   (6) Run **javac -d . CheckLocation.java**
   (7) Run **javac -d . Exec.java**
5. Once compiled, you can run the file.
6. Run **java bugants.Exec**
7. You can play the simulation.

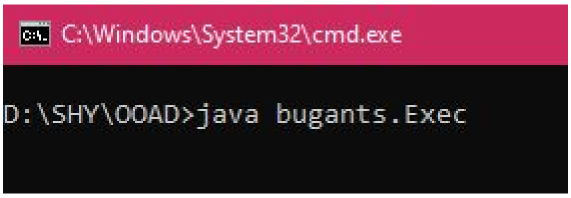
How to Play the Simulation.

1. Click the “**START**” button.
2. The simulation will move with every click of the **START** button.



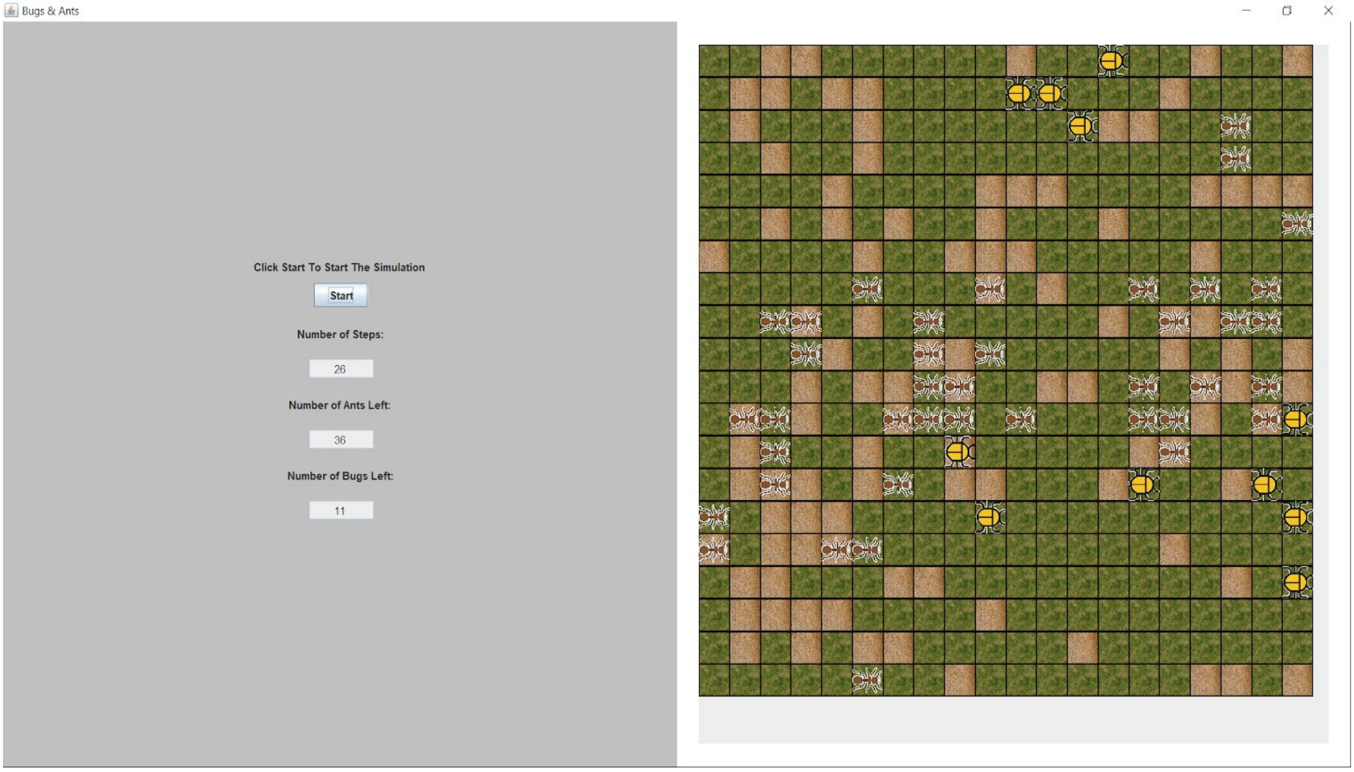
```
Microsoft Windows [Version 10.0.18362.657]
(c) 2019 Microsoft Corporation. All rights reserved.

D:\SHY\OOAD>javac -d . Ant.java

D:\SHY\OOAD>javac -d . Bug.java

D:\SHY\OOAD>javac -d . Board.java

D:\SHY\OOAD>javac -d . Maze.java

D:\SHY\OOAD>javac -d . CheckLocation.java

D:\SHY\OOAD>javac -d . Movement.java

D:\SHY\OOAD>javac -d . Exec.java

D:\SHY\OOAD>
```

Figure 2.5.1 shows the methods to compile each and every file of the Bugs & Ants Simulation.



```
D:\SHY\OOAD>java bugants.Exec
```

Figure 2.5.2 shows the method to run the Bugs & Ants Simulation.



Bugs & Ants

Click Start To Start The Simulation

Number of Steps:

26

Number of Ants Left:

36

Number of Bugs Left:

11

Figure 2.5.3 shows the display of Bugs & Ants Simulation.