Personal information

Strategy game

Lucas Foley 910527

Kone- ja rakennustekniikka

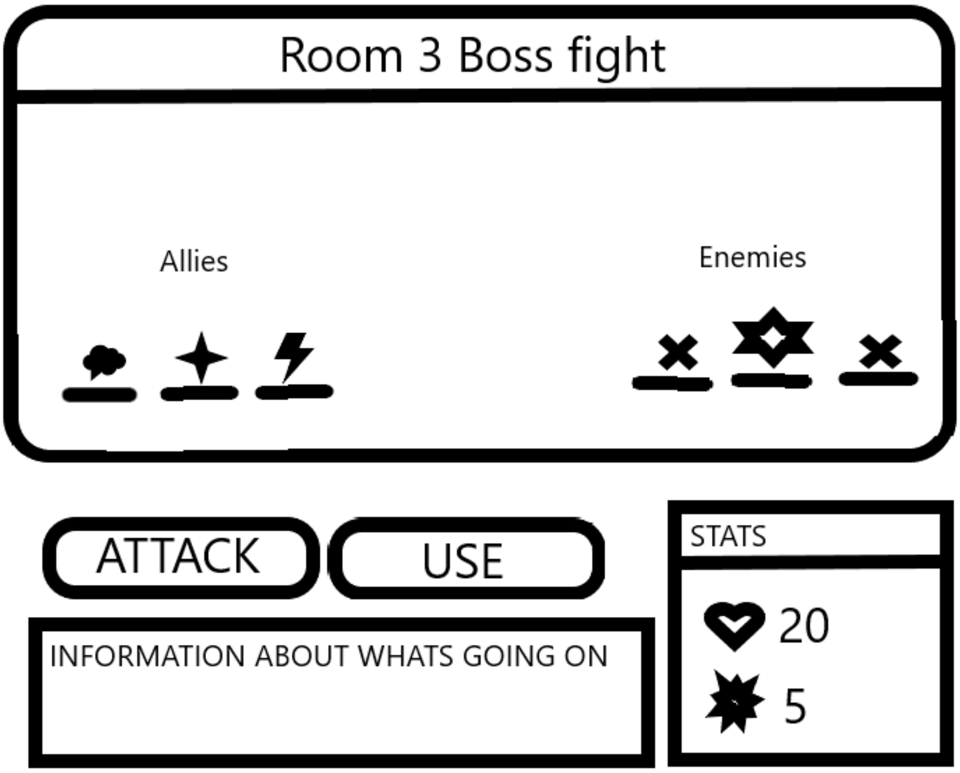
18.02.2022

Overview and difficulty level

Medium level

- Graphical interface
- Different self-made characters
- Intelligent (not just random) enemy AI
- The game should be configurable with outer files (as much as possible)
- Unit tests for at least part of the program

Use case description and user interface sketch

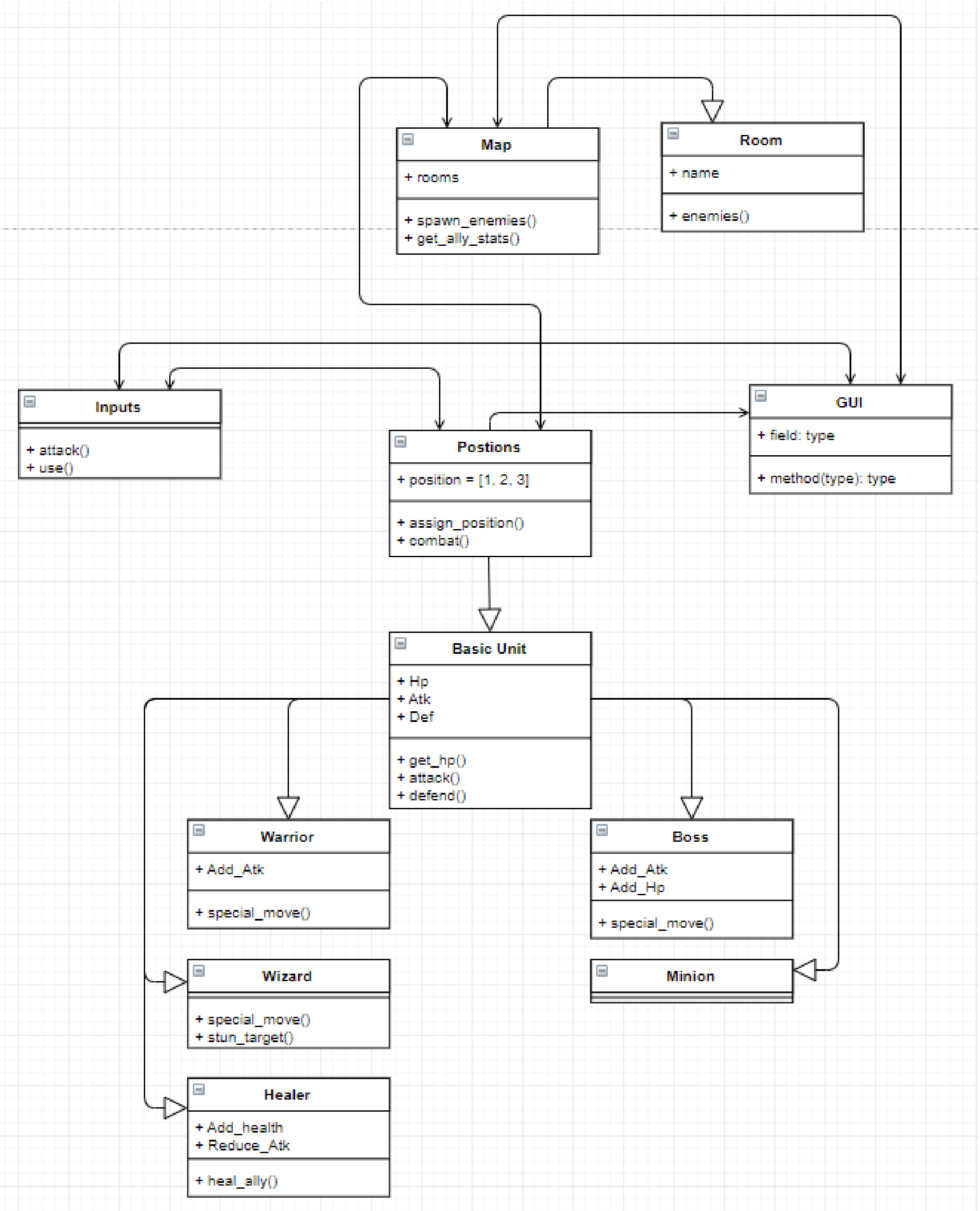


The inputs for attack and use could use assigned keys or possibly be accessible by mouse if I can implement it.

Information regarding whose turn it is and what is happening would appear in the lower wide box.

Stats would be displayed in the lower right box (could be moved to above allies and enemies if more convenient)

Structure plan of the program



Here is the initial structure for the program. It still needs improvement and more specific methods but the overall picture of how it would work together can be seen.

I am not yet quite sure where to implement the information for each turn, so I have left it out for now until I can do some consulting with the assistant. If I had to place it somewhere I think it could maybe fit in the inputs class or maybe a subclass.

Data structures

I will need to use list quite a bit to update the variables and keep track of the rooms and stats for the different mobs. Other than this it will be mostly adding and subtracting from the health points in combat which can be simply handled.

Dictionaries could also be as useful part of the program for giving initial stats and things

Files and file formats

If the game is to be saved then there needs to be a way of storing the position, room number, and stats for the individual characters. Since it isn't too much information to be stored, I think it could be accomplished by writing it to a text file and then reading it again when needed.

Algorithms

The algorithms needed in this project would mainly concern the AI. It would need to be able to determine which enemy target is the most beneficial to attack. I believe it could be accomplished simply by programming the AI to focus the lowest health target and/or the target that deals the most damage. For it to do so it would only need to compare the health and damage stats of the enemy.

If there are more specific special moves or status effects implemented, then it may also be a good idea to program specific tactics for these moves.

Test plan

Testing that the combat works correctly would be the most important thing for the project. Reducing the right amount from the right target is crucial for the game to work. The mechanics of a unit dying and being out of play could also be good thing to test for since if it's not working then the whole game would be flawed.

So put shortly everything involving combat testing is crucial for the project since the whole game realise on it working.

Libraries and other tools

PyQt is the only allowed library so I assume it would be of some use in the project. I am not familiar with it but I am sure that I can find some use for it.

Schedule

Since combat is the most important part of the project, I think most of my time will be spent on programming the combat interactions and unit stats/moves.

The map and rooms will also take some time but are simpler and can be predetermined so I think that it will take less time to program but maybe more planning time will be needed to make a good map and room combination.

For the input there is going to be a lot of work but it should be simpler code but since the input ties everything together there could be more work than I expect. I do think there will be a large part of my time spent coding the inputs.

Lastly for the GUI I can't say how much work it will need since I am unfamiliar with working on GUI's.

Schedule:

Basics of combat and inputs established and some code for it should be ready for the first checkpoint

Fully written code for combat and testing for it as well as the code for the inputs and map should be complete for checkpoint 2.

Lastly the GUI and polishing will be implemented before the final deadline.

By following this schedule, the core of the program should be finished and ready by the second check point leaving plenty of time for the final touches and any unexpected problems.

Literature references and links

I am planning on using sites like the ones below for basic and implementation of the code:

https://www.w3schools.com/python/python_intro.asp

https://pythonbasics.org/class/

I also plan on using YouTube for tutorials and maybe some inspiration on how to program the game.

For a game reference I have chosen darkest dungeon since it is at it's core something like the game I would like to program even though it has a lot more dept to it.

https://en.wikipedia.org/wiki/Darkest_Dungeon