

Rationale

Requirement 3

The new class I added in are EnemiesBehaviour, Koopa and SuperMushroom class.

New class Koopa class

- Create Koopa and it will handle all functionality relevant to Koopa, such as turn into shell, create SuperMushroom instance, behaviour of Koopa and as well as the damage and hit rate of Koopa. The new class will create Koopa which is the enemy required by the task
- dependency AttackBehaviour so I can let Koopa to attack the player automatically
- Inherit Actor class to have the method to implement Koopa and Goomba hp, Koopa and Goomba display character whether they are conscious, being hurt by player and etc
- Dependency on SuperMushroom to create SuperMushroom instance once the shell is broken
- Dependency on GameMap is to remove Koopa once the condition is met such as if the Koopa shell is broken

Modified Goomba class

- Changes: Add hurt() method to attack the player
- Inherit Actor class to have the method to implement Goomb and Goomba hp, Goomb and Goomba display character whether they are conscious, being hurt by player and as well as the damage and hit rate of Goomba
- dependency AttackBehaviour so I can let Goomba to attack the player automatically
- Dependency on GameMap is to remove Goomba once the condition is met such as if the Goomba is not conscious.

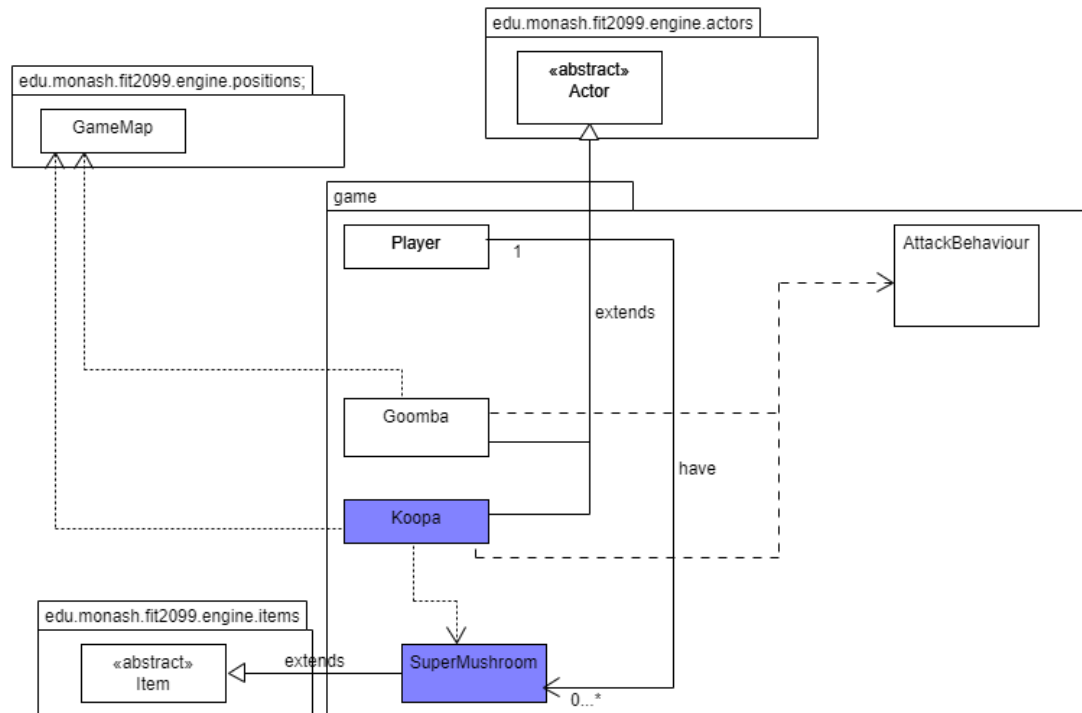
New class SuperMushroom class

- Represent the SuperMushroom instance of mushroom
- Player associate SuperMushroom class so that I can have a list of SuperMushroom instance in Player class, since a player can have 0 to many supermushroom in the inventory, so the list of supermushroom will be representing the player's inventory.
- inherit abstract class item to have the method to add into player inventory.

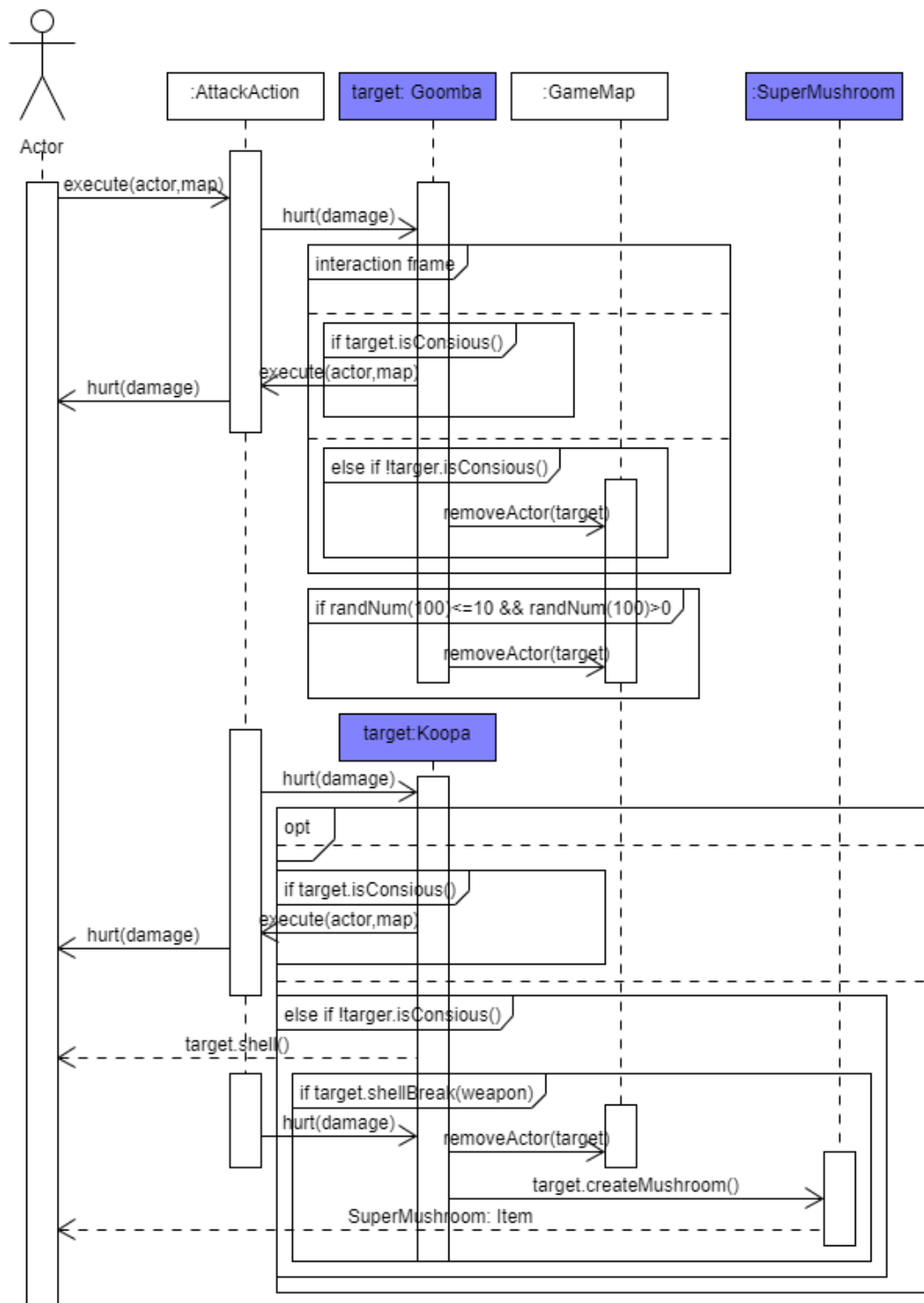
Requirement 3

(ps: also pushed onto git for these pictures, can look for the pictures in the folder if the pictures here are not that clear to read)

Class diagram



Sequence diagram



Requirement 4

new class SuperMushroom class

- create the SuperMushroom instance
- Player class have an associate on SuperMushroom so that the player can have 1 to many superMushroom in inventory
- inherit abstract class item to have the method to add into player inventory.

New class Coin[covered in another requirement]

New class WalkBehaviour

- To update the player walk behaviour after using PowerStar where the player can walk through any type of ground
- Implements interface class Behaviour because the player can have different behaviour so later the player can have a list or hashmap of behaviour

New Class JumpBehaviour

- To update the player jump behaviour after using SuperMushroom the player have 100% success jump rate
- Implements interface class Behaviour because the player can have different behaviour so later the player can have a list or hashmap of behaviour

New Class PowerStar Class

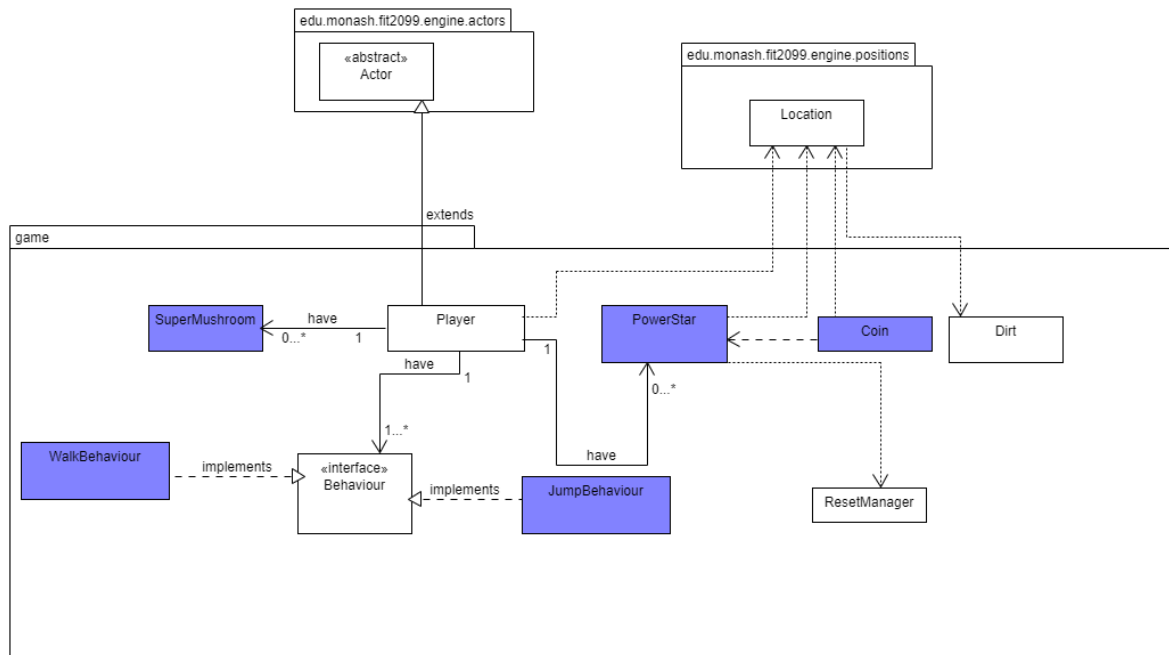
- create the PowerStar instance
- Player class have an associate on PowerStar so that the player can have 1 to many powerStar in inventory
- inherit abstract class item to have the method to add into player inventory.

Modified player class

- To change the player behaviour, hp, set player display character
- Use the setDisplayChar(char displayChar) inherited from the super class to change the player name for example from m to M
- Add new method Immunity() which will set the player to immune from the attack of enemies.
- Add new method setMaxDmg() which will set the player Damage to max and kill enemy with only one hit

Requirement 4

Class diagram



Sequence diagram

