

## **Design Rationale**

### **REQ1: Lava Zone**

Warp pipe is created to allow Players to move between 2 different zones. WarpPipe class is an extension of Ground abstract class. It has two attributes, the Location which is the WarpPipe's location and a target WarpPipe, which is where the WarpPipe teleports the Player to. It has a dependency with PiranhaPlant as it spawns a PiranhaPlant once the game starts. WarpPipe also implements Resettable interface as when player resets the game, WarpPipe will spawn a PiranhaPlant on it again. Lava class is an extension of Ground abstract class which has a property of burning the Player if Player steps on it while actors belonging to Enemy abstract class are not allowed to step on it.

### **REQ2: More Allies and Enemies**

PrincessPeach is an extension of Actor abstract class that offers the Player an InteractAction. The InteractAction checks if Player has a Key object that extends Item abstract class dropped by Bowser. If it does, InteractAction will display a victory message and the game ends. Bowser is an extension of enemy class. Bowser has an item in his inventory which is the Key object that extends item abstract class. Bowser also has a capability FIRE\_ATTACK which AttackAction checks to spawn FireAttack objects which are also extensions of Item abstract class which burns any actors that are in that location. PiranhaPlant class extends Enemy abstract class and is generally the same as the other classes that extends Enemy abstract class but by default already has a AttackBehaviour so it will attack the Player if it is nearby and will not have a FollowBehaviour or WanderBehaviour as the PiranhaPlant cannot move. Lastly there is the FlyingKoopas class that extends Enemy abstract class as well which is identical to a regular Koopa class but has an additional capability, FLY. This allows the FlyingKoopas to move onto HighGround objects.

### **REQ3: Magical Fountain**

Bottle class extends Item abstract class and associates with Water abstract class as it stores an ArrayList of Water objects. There are two child classes of Water abstract class which are HealthWater class and PowerWater class. There are also PowerFountain class and HealthFountain class that extends Fountain abstract class which extends Ground abstract class. Water abstract class and Fountain abstract class implements Consumable as it could be targeted by ConsumeAction. Fountain abstract class and Bottle offers the player ConsumeAction to consume it. Fountain abstract class additionally offers the player RefillAction which adds a Water instance into the player's Bottle in the inventory. ConsumeAction targets the Water instance while RefillAction targets the Fountain instance to identify what water to be consumed by the player and apply the appropriate effects.

## **REQ4: Flowers**

FireFlower class extends Item abstract class which implements Consumable as it offers ConsumeAction to the player to consume it. Sprout and Sapling class have a dependency with FireFlower class as they spawn a FireFlower instance after 10 rounds. Consuming the FireFlower would allow the player to have FIRE\_ATTACK capability. FireAttack class also extends Item abstract class and is spawned by AttackAction. AttackAction has a dependency with FireAttack as it is spawned when the actor executing AttackAction has FIRE\_ATTACK capability.

## **Updates to Assignment 2:**

### **New HighGround abstract class:**

The HighGround abstract class extends the Ground abstract class. It is then extended by Tree abstract class, Wall class and WarpPipe class. It has methods inside that allows the Actor to execute JumpAction to enter that ground's location.

### **New Consumable Interface:**

The Consumable interface is implemented by classes that could be consumed by the Player. ConsumeAction only targets objects that implement the Consumable interface. Classes that implement the Consumable interface are PowerStar, SuperMushroom, Fountain and Bottle.

### **New Tradable Interface:**

The Tradable interface is implemented by classes that could be traded by the Player with Toad. TradeAction only targets objects that implement the Tradable interface. Classes that implement the Tradable interface are PowerStar, SuperMushroom and Wrench.