Design Rationale

Bushes, Trees, and Fruit

The implementation of bushes would be done through the Dirt object, where Bush would be initialized as a class attribute within Dirt. The associated methods in Dirt would account for the necessary parameters which indicates if the Bush object would be instantiated or not. Each Dirt block would only either have 1 bush or no bush. The Bush and Tree class would generate instances of the fruit object placed in an arraylist as part of its attribute. For now, we have implemented a maximum limit of 10 fruits for each tree, and a maximum of 5 fruits for each bush.

Dinosaurs

The dinosaur classes (Stegosaur, Brachiosaur, Allosaur) will have a character attribute to represent its gender, where the character 'F' indicates a female and 'M' represents a male. In our implementation, a baby dinosaur will be an instance of their dinosaur classes (Stegosaur, Brachiosaur, Allosaur). However, these classes will have a boolean attribute to indicate if it is a baby dinosaur.

Eating and Feeding

We have implemented an Eating Action for each dinosaur species as each of them have different food conditions. Each dinosaur class will implement a similar eatItem(item) method that accepts the food item and increases the food level accordingly. This makes it more flexible and allows one method to control eating fruit, meal kits, etc. To highlight, the Allosaur class has a slightly different implementation for its Eating Action. First, if the Allosaur is at the same location as another dinosaur, it will check if the said dinosaur is a Stegosaur. If the check passes, the Allosaur will attack the Stegosaur. The damage done to the target is the amount of food level the Allosaur obtains. If the Stegosaur is killed as a result of the attack, a corpse item would be generated on the location. Corpses are another form of consumption for Allosaurs. At the location of a Corpse item, the Allosaur will consume the instance of the corpse object, removing it from the game map.

Eggs and Breeding

In our implementation, after a Breeding Action has occurred between two dinosaurs, an Egg object will be created and added to the female dinosaur's inventory. When it is time to lay the egg, it will then be removed and placed at the location of the female dinosaur. In the Egg class, one of it's attributes will be a String that represents its species.

Death

To tackle the Death portion for relating Dinosaur objects, a subclass of item named Corpse would be instantiated. It first checks if the dinosaur is alive with an innate method, if the

method returns false, a Corpse item would be generated and placed at the current location of said dinosaur, before the dinosaur actor object is ultimately removed from the game map.

Eco Points and Vending Machine

An EcoPointStorage integer attribute will be instantiated within the Player class, as each Player will have a different amount of eco points obtained, depending on the world event instances. The Player class would have the appropriate setters and getters for retrieving and altering the Eco Points. With the initialization of Eco Points, the Vending Machine class would be instantiated as a child of the item object and will be spawned within the Application class. With the creation of the Vending Machine object, a new action subclass would be added to aid its functionalities, named as PurchasingAction. PurchasingAction is the main form of interaction action between the Player and the Vending Machine object.