Design Rationale

Grass

Methods are added in pre-existing dirt class, stated below:

1. Chances for grass to grow on dirt (2%)
2. Chances for grass to grow on dirt if there is at least one adjacent grass (10%)
3. Chances for grass to grow on dirt if there is at least one adjacent tree (2%)

Methods are added in pre-existing tree class:

1. Chances for fruit to drop (5%) **drops same tile as tree**

Methods are added in pre-existing player class:

1. Option Playturn to add hay to inventory if standing on grass, grass changes back to dirt
2. Option Playturn to add fruit to inventory if standing on tree square and fruit is dropped
3. Option Playturn to add fruit to inventory by chance if standing on tree square (40%)

Hungry Dinosaurs

To include hunger, we may be **extending Actor** to create a class named “Dinosaur” and then extending again for both Stegosaur and Allosaur, but this is not confirmed.

Also, to implement a Hunger Behaviour, we must introduce a new class, “Hunger Behaviour”, making use of Behaviour Interface to allow movement of Dinosaurs when they are hungry.

Methods are then added to Dinosaur Class (new class):

1. Add Hunger into constructor, starts at 50 (integer)
   Every Playturn, checks if Dinosaur hunger level is below a certain level (e.g. 30), if yes Hunger Behaviour is called and output to inform Player that Dinosaur is hungry.
   Every Playturn, checks if Dinosaur hunger level is 0, if it is Dinosaur become unconscious. Not allowed to move
   After 20 turns of unconscious, Dinosaur dies.
2. Every Playturn, checks if Dinosaur standing on a food source, or a Player is near it, if yes Hunger can be increased based on actions.

Methods are also added to Hunger Behaviour (new class):

1. When called, it moved Dinosaur one tile closer to a food source (e.g. Hay, Fruit, Grass). It can move horizontally, vertically and diagonally.

Breeding

To implement breeding, Dinosaur Class must have a boolean added to their constructor to determine their gender. (True for Male, False for Female)

Also, to implement a Breeding Behaviour, we must introduce a new class, “Breeding Behaviour”, making use of Behaviour interface to allow movement of Dinosaurs when they want to breed.

Methods added to Dinosaur class:

1. Every Playturn, checks hunger level, if above an amount (e.g. 60), calls breeding behaviour.
2. When breeding behaviour returns true (mating successful), after 10 turns, Dinosaur lays an egg on the tile its at. That egg itself is a new object of the Dinosaur class.
3. If Dinosaur is an egg, after 10 turns it hatches into a baby dinosaur. It is not allowed to breed and has a low hunger.

Eco Points for Purchasing

To implement Eco Points, Player Class must have an integer added to their constructor.

In certain methods in Player, Dinosaur and Dirt, there will call methods to increase the Player’s Eco Points.

To Implement the vending machine, we extend ground with a new class “Vending Machine”, and it can only added at the start of the game.

Vending Machine has methods added below:

1. Checks Player’s Eco Points
   Option Playturn to add *items* into inventory depending on Eco Points, if Option Selected, items will be added to inventory and Eco Points will be deducted.

Allosaur

To implement Allosaur, it will have an extra Behaviour Interface, called “Carnivore Behaviour”, which replays “Hunger Behaviour” for Stegosaur

Carnivore Behaviour will have methods below:

1. When called, it moved Allosaur one tile closer to a food source (e.g. Stegosaur). It can move horizontally, vertically and diagonally.
   When beside an alive Stegosaur, it attacks Stegosaur.
   When beside a dead Stegosaur, it eats Stegosaur.