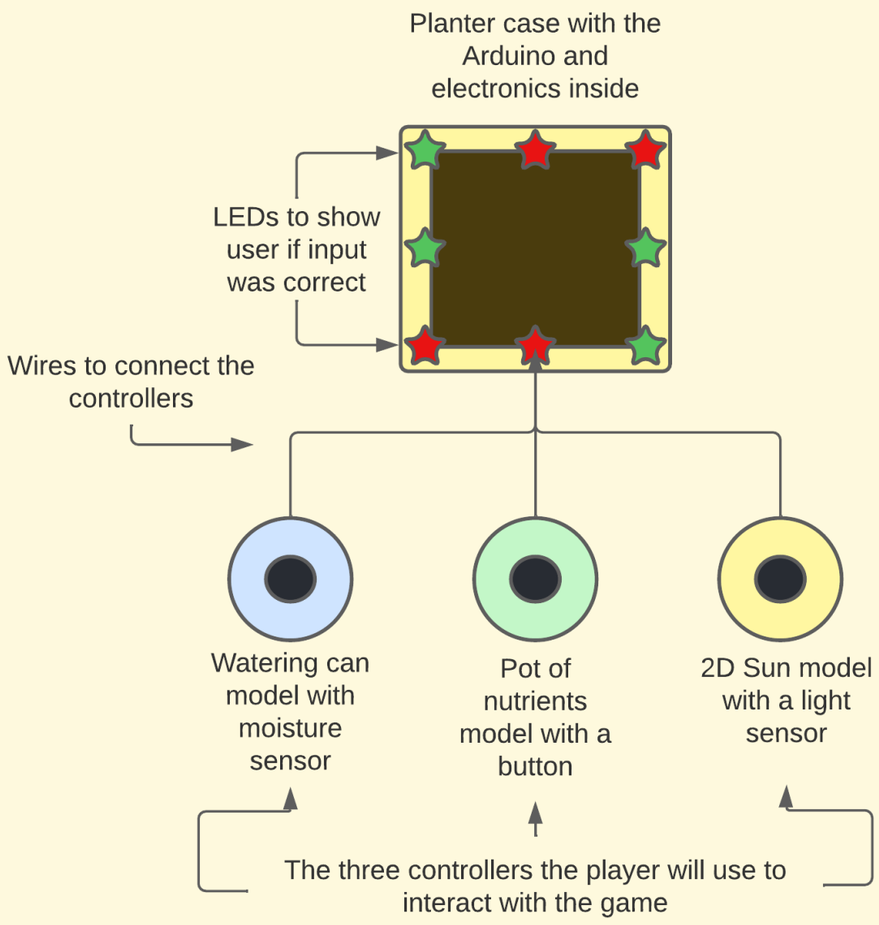
Gardening sim controller
Comp 140- LW260328

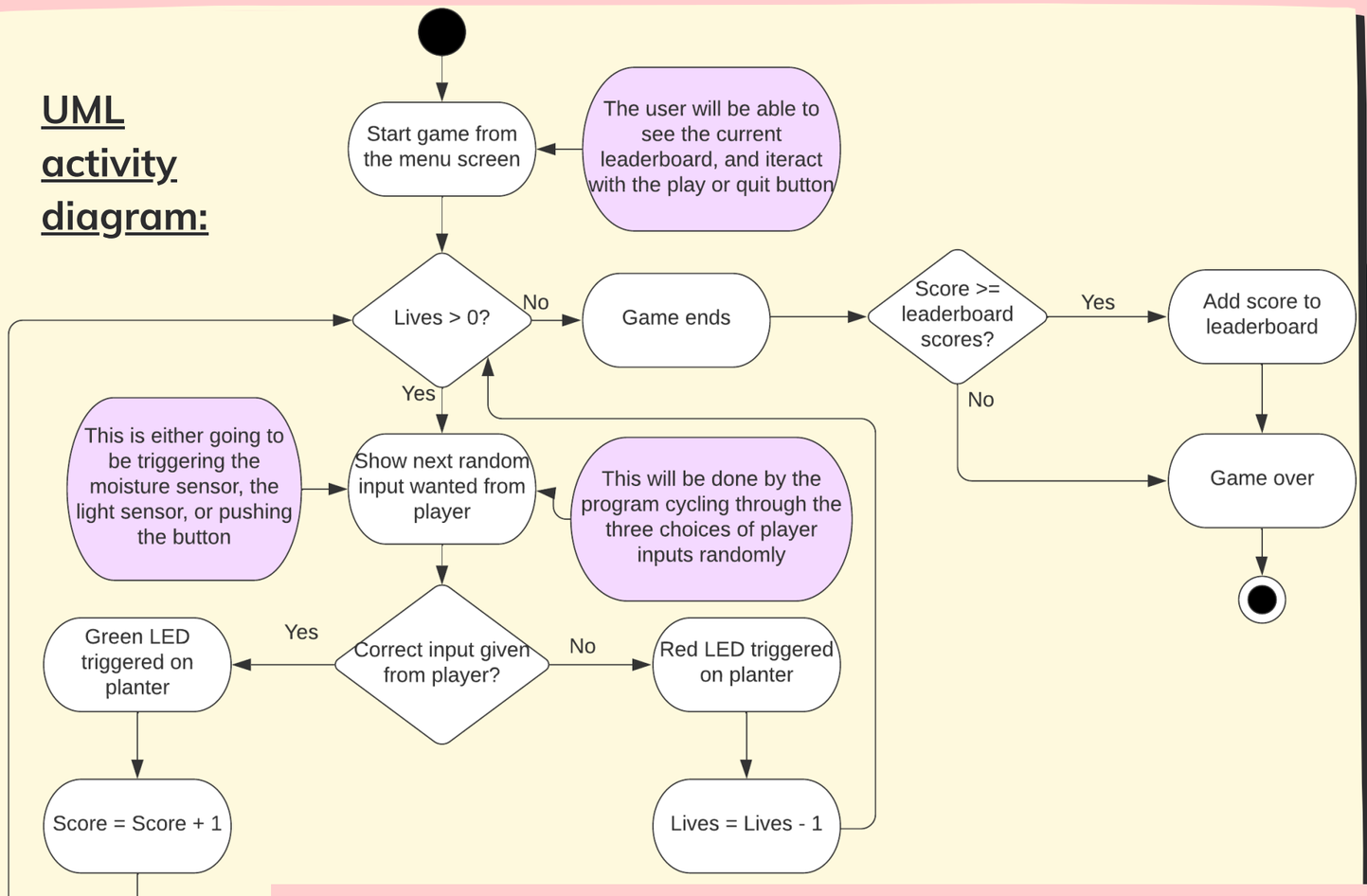
Hardware design diagram:



Hardware design outline:

As I have three types of sensors, each of them are going to be encased in or attached to the corresponding models which will be 3D printed. For these, I'm going to have a moisture sensor for the watering can, a button on the pot of nutrients, and a light sensor on the sun.
There are going to be several LEDs going around the planter which will be green or red. These will be triggered to turn on or off depending on if the player input was correct.
In order to make my controller more portable and appealing to look at, I'm going to make the planter box hollow with access to the bottom section of it. This will allow me to hide the Arduino and breadboard inside

UML activity diagram:



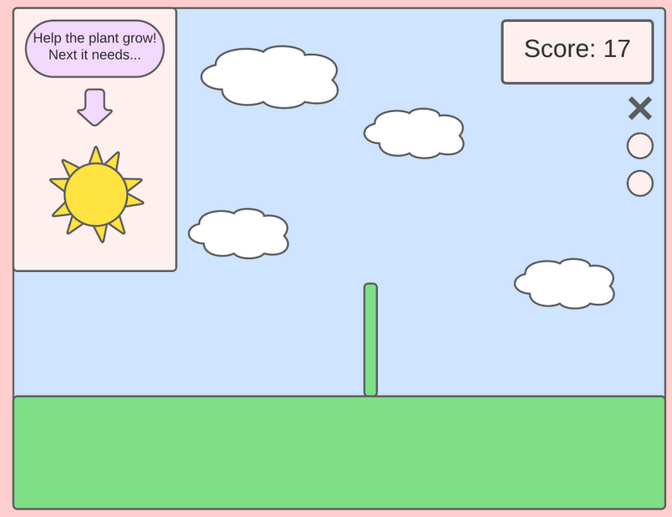
Controller overview:

My gardening sim controller is about helping a plant grow as big as possible by providing it with its basic need in order for it to grow. The player will have to pay attention to the game screen to see what randomly generated input is requested next to help the plant grow, those of which will be in the form of a 3D printed watering can, a pot of nutrients, or a sun. With all three using different sensors, it will create a fast paced game similar to 'bop-it'. To aid the controllers, the game is going to have a 2D interface in which the player can see the growth progress of the plant and such to make the game more immersive and complete.

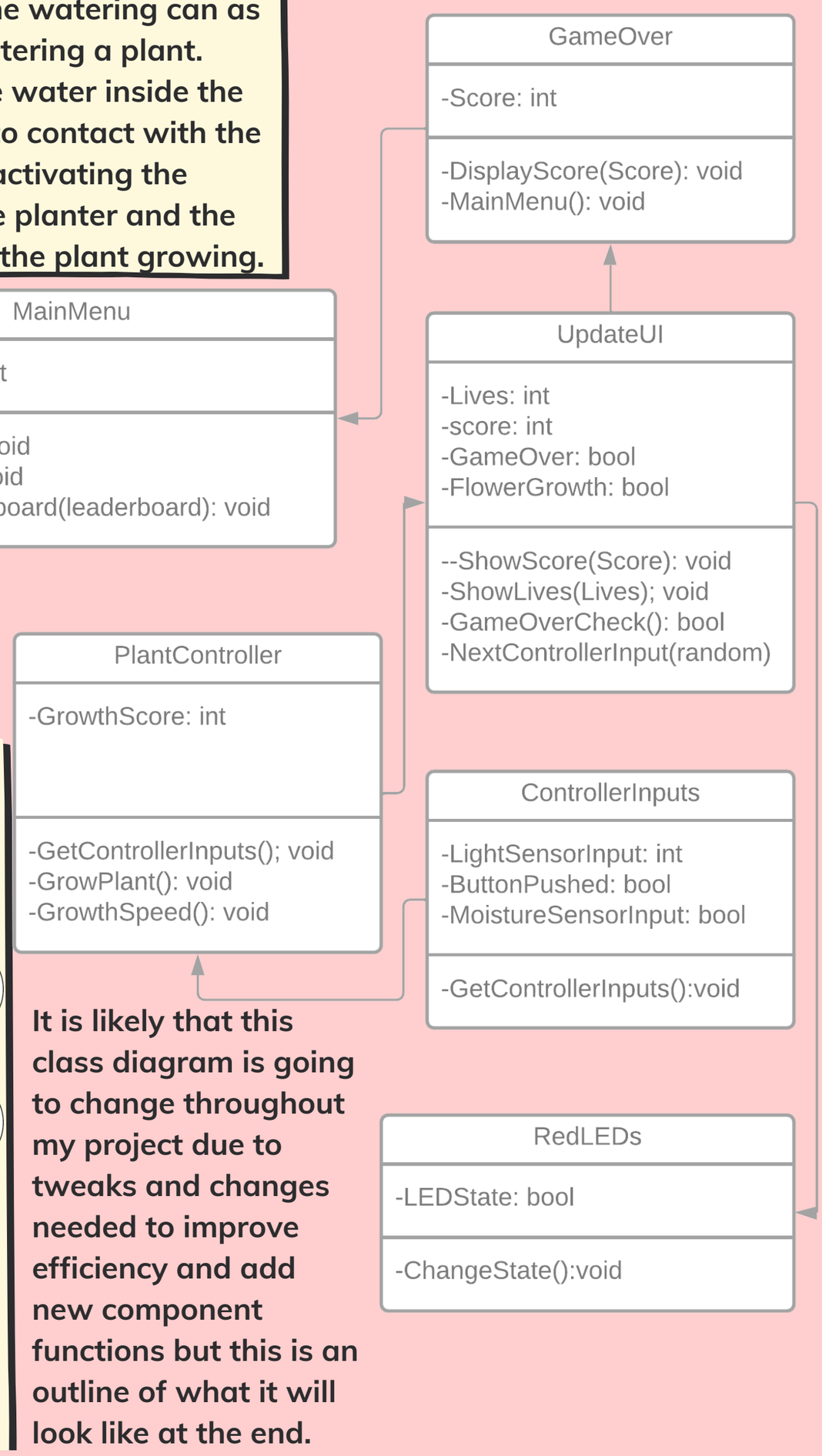
Software design outline:

The interface for the game will be made with unity, using the Uduino extension to allow for interaction between the Arduino and game interface. For the UI, I'm going to have a very simple screen, only needing to show the user their current score, the plant progress, their remaining lives, and finally, the next input wanted from the user. This will then correspond to the 3D printed controllers that the player will have to interact with. For example, tilting the watering can as if the player is watering a plant. This will move the water inside the model to come into contact with the moisture sensor, activating the green LEDs on the planter and the UI to change, like the plant growing.

Example of what the UI will look like:



UML class diagram:



It is likely that this class diagram is going to change throughout my project due to tweaks and changes needed to improve efficiency and add new component functions but this is an outline of what it will look like at the end.

The types of sensors i want for the game inputs have changed several times due to considering the best player experience with the controller. For example, changing the thermal sensor to a light sensor for the sun. This is because the thermal sensor would have issues changing temperature fast enough to keep up with the game. Having a light sensor instead makes the game more interactive and gives the player more to do, whilst also working more efficiently with my controller as a whole.