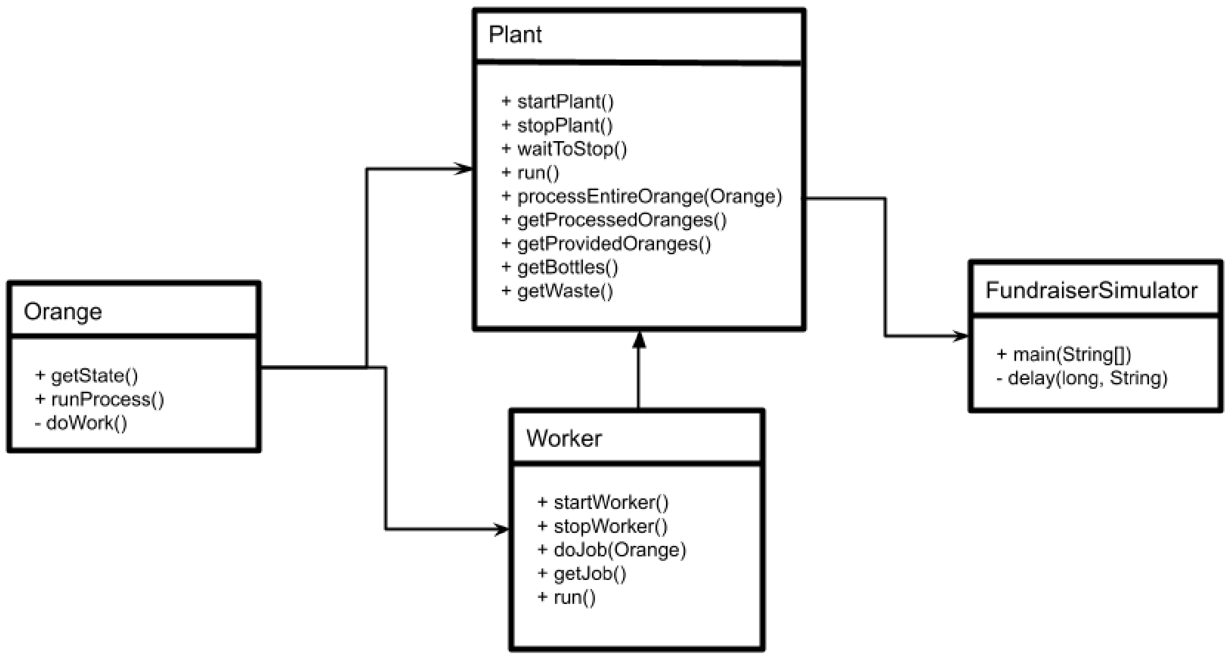
Robert Hereth
CS-410-A
1 October 2021
Lab 1 Documentation

UML Diagram and Class Descriptions



The classes shown in the UML diagram above were used to complete this lab. The Orange class was left untouched. The Worker class was made by me in order to divide the work within the plant. The worker class uses the Orange class' states in order to give each worker an assigned "job". This "job" is all that each worker is allowed to do, and they work in their respective plants as a team to complete the processing task. The two plants run simultaneously, and the Plant class is based off of the MultiPlant class given. The FundraiserSimulator class is the main class that runs the project, and it calls the Plant class to create the two parallel plants.

Challenges Faced

Throughout this lab, I faced the issues of getting the worker threads to cooperate with the plant class, as well as how I wanted to implement the workers and divide the tasks. Once I found a way to reference the states from the Orange class, the workers were able to split the task. This method is very inefficient, as it only processes a total of 98 oranges every 5 seconds. I am unsure if I completed this project correctly, as I feel like something with the multithreading of the worker class is off. I intend on revisiting this code once we discuss the solution, and finding the best way to optimize this method. I mostly had an issue with modifying the MultiPlant to accept my worker threads, and the other main challenge was solving how to divide the tasks up with the worker threads. This lab was very challenging and was slightly confusing at parts. Though I still do not understand the full extent of why my code works how it does, I would like to continue looking into the workings of threads in Java.