Patrick Laffey
44373821

The program runs off the class Instance which calls upon the other classes as they are needed. This allows for the potential to have multiple instances of the program running at once. When a player or pet is added to the game, the classes are called from the instance class. They are initialized and saved in relevant ArrayLists in the instance class where they can be called upon later when needed.

Although it would have been preferable to have the toy and food classes to be subclasses of a class item, which would encapsulate common or similar elements of both classes, the decision was made not to do this. This is because the development of the program was already quite far when the discovery that this parent class would improve the program was made. Due to the time constraint it was decided to not implement it, for it was better to get the program running with what there was rather than making improvements but not being able to complete the program on time.

Each individual food, toy and pet class has been kept very small due to the majority of methods and parameters being encapsulated inside the parent Food, Toy and Pet classes. The changes that needed to be made for the individual classes simply override their relevant parent class.

The program was made in such a way that every element of the program could be run from one class that just gathered other classes and called upon the when they were needed.

The UML diagrams provided are those of the original planned program. Some changes were made to the program structure however there was not enough times to make up new UML diagrams.

Though out the entire development of the program I thoroughly tested each method as it was completed, as well as full run throughs of everything that had been completed at major milestones. I have done a brief JUnit test on the Player class, however did not have time to do this for all classes.

All java docs are located in the doc folder inside the program folder.

All images used were taken from https://pixabay.com/ and listed as free for commercial use with no attribution required. From there I edited the images myself using Adobe Photoshop CS6.

I enjoyed this assignment, it had lots of potential and you could do anything with it. It involved many different aspects of Java coding which has helped me become more proficient in the language. I do wish I had more time to work on it and I feel we should have been free to use whatever GUI development we wanted, for I had some major issues with WindowBuilder to begin with.

This is not my first game application, but was my first in java. I am happy with how the classes communicate and the package hierarchy works but there were lots of improvements I wish I could have implemented if I had more time. The major changes I would have liked to make involved changing the base of several classes, which by the time I thought of them was not possible due to not having a partner. I kept a brief note of some of these changes but continued in the development for I thought it would be better to get it going with what I had rather than making these improvements and not finishing due to the time constraint. Some major changes I would have liked to make include the following.

With the Pet class I would have liked to remove all the individual getters and setters for the stats of the pet and replaced them with one general getter and setter which took an additional parameter which determined which stat the method was to get or set.

In respect to the Food and Toys classes I really wish I had made a class, Item, which both Food and Toys would be a subclass of. This would have made it possible to have one player inventory and one shop inventory much easier, as well as removing the need for doubled up or very similar methods in Food and Toys. However, I would have still kept the Food and Toys classes so that they are still defined differently. I also did not have enough time to do as much balancing with the different food and toys as I would have liked.

I also would have liked to combine the add and remove methods in class Player so that they work for pets, food and toys.

My partner dropped out of SENG201 on the 17th of May after having done no work towards the assignment. The assignment is entirely my own work.