Kelly Herstine

Jeff Timanus

AI Programming

September 22, 2018

Assignment 1

Overall Architecture of your Assignment:

All steering classes are in their own files. Some of these classes include: Wander, Face, Arrival, Wander and Chase, and Arrive and Face steering. The steering class, Wander and Chase, uses variables that include references to the Wander, Face, and Seek classes. The steering class, Arrive and Face, uses variables that include references to the Arrive class and the Face class. For the Arrive and Face steering class, I get the steering from each of the Face and Arrive classes and assigned them to their own variable. I then made sure that they were not NULL and used the acceleration and the rotation acceleration from the class. If they returned NULL , I made sure to set the objects rotation acceleration, acceleration, velocity, and rotation velocity equal to 0. I then return the data that is used. In Wander and Chase Steering, I get the player location and then I get the distance from the owner and the player using a Vector2D. Using this data, I determine whether the unit should seek or wander around. I then return the acceleration of the unit. I used the Input system from my architecture class from last semester. This system is based on the Event, Event Listener, Listener, and Event System classes that we had used. The Event class holds different enums for each event that the keyboard and mouse will produce. It then makes sure that they are valid event types so that the program can run. The Event Listener is used as a virtual class that handles listening to events. The Event System is used to add, remove, fire, and

dispatch events from the keyboard and mouse. It uses a map in order to handle each event that is sent to it. It goes through the map using an Iterator and handles each event inside of it. The Listener class is used to listen for events that are passed in the program. It then adds the listener to the Event System. The Input system is where I fire all of my events. The Input system is updated every frame to check whether the enter key, the d key, the mouse left button, and the escape key are pressed. If they are pressed, they then fire the corresponding event which the Event system then picks up and hands off for the system to listen to. In the Game.cpp file, I then listen for the events. If there is an event that is sent, the handle event in the Game file handles the event and does a corresponding action. This continues until the program ends.

Challenges Faced in Development:

Wander steering was giving me some major issues when I tried to implement it. I followed the pseudo code that was in the book and it seemed that it just didn't want to work. I spent most of my time trying to figure out Wander and try to make it work.

Areas where further improvements could be made:

I think that something that could still be worked on is my Wander steering and Face steering. I used the book to implement these two steering methods but I don't think that they are 100% perfect. I spent a very long time trying to figure out both of these steering methods and I got them working to an extent.