Ben Morledge-Hampton

First Deliverable for Final Project: **Sandy Sovereign**

For my final project, I will be designing a strategy management game where the goal of the player is to guide a colony of miniature “sand people” to greatness as their king. These sand people reside on a balmy beach fraught with the perils of incoming tides, ravenous seagulls, and careless tourists, as well as other dangers yet to be described. The goal of the sand people is to create a glorious sand castle on the beach that will offer them protection and recognition from the outside world. Your goal as king is to guide your sand people to protect them from these increasing dangers while all the time working towards the completion of your castle!

The gameplay will be centered around a myriad of choices given to the player which will all have varying impacts on the resources available to them. The player will have to manage stockpiles of useful building materials such as seashells, seaweed, sand, and sticks. These resources will need to be diverted to building different establishments such as walls, housing, or the grand sand castle itself. Furthermore, the state of the people must always be considered in terms of their happiness, health, and population. Besides all of that, other resources may come into play such as highly lucrative sand dollars or the elusive wonder-clam. All of the choices given to the player will allow them to alter these resources in some way shape or form. For example, in the event of a storm, the player may choose to risk the safety of their people to gather supplies blown in by the wind, or they may choose to keep their people safe but only gather a minimal amount of supplies after the storm has passed.

On the coding side of things, one class called the “Manager” will extend JFrame to handle all of the input from the user as well as display output. The Manager will contain instances of several extensions of JComponents to display various menus all contained in one superclass, “Menu.” It will also use an array list keep track of all of the possible Events and Disasters (Also their own class, “Event”) that could occur and trigger those events on a regular basis. The results of those events will affect a Stockpile class which will handle all of the resources and other statistics that are important to the city. Obviously, the Manager will have access to this stockpile as well. Throughout the menus handled by the manager, input will be handled by a whole lot of buttons! Buttons to make choices for events, buttons to view current resources, buttons to build new stuff, etc. etc.

As for some more specifics on gameplay, the game will be paced around a series of disasters which happen after a fixed number of normal events. These disasters get progressively more punishing until the player has either completed their castle, or all of their sand people are dead.☹ In between these disasters, the events that occur are random. The player may take their time in between or during events to view the statistics of their sand colony, but new buildings can only be made in between events. Furthermore, not all buildings will be unlocked at the start. Building the sand castle up will unlock new buildings. Building these new buildings will help to provide a steady income of resources which will be invaluable as the disasters ramp up in difficulty.

One thing that I foresee difficulties in is creating a dynamic range of events from a single class. I want to have enough events that it doesn’t make sense to create a new extension of the class for each event, but they will still be quite varied. I will deal with this in a couple ways. First of all, the event class will have a set of “Result” objects which will store information about the results of a variable number of choices available to the user. Secondly, these results will contain “Conditional” objects which will exist to double check that if an event requires certain supplies to reach a desired outcome, the result acts according to whether or not the player has those supplies. Hopefully this is a suitable solution, but I get the feeling I may run into further issues later… Only time will tell!

Yeah! That’s the gist of it. I’m sure things will change more as I develop the code, but I think this is a good starting point!