**Project Description**

EcoSim is an application that aims to simulate basic ecosystems. The process starts by generating a randomized terrain. From there, animals will be placed in the simulation. Animal types can be sheep, wolves, bunnies, foxes, and more. The user can choose to let the simulation run on its own, resulting in a completely randomized simulation cycle. The user also has the choice to modify specific aspects of the simulation, such as offspring rate, food growth rate, average lifespan, and much more. The simulation aims to cover a large range of real-world aspects, such as population levels, prey vs. predator, mutations, and much more.

**Competitive Analysis**

**Structural Plan**

Animals: Every animal type will share the same core attributes (i.e. each animal will have hunger levels, thirst levels, health levels, offspring rate, mutation rate, etc.). Each animal will inherit these attributes from the parent class, Animal. Each animal type, however, will have their own specific starting attribute values. For example, a sheep’s starting strength levels will be far less than a wolf’s starting strength levels. Each animal type’s class code will also be stored in its own file for organization.

Assets: Each asset for every animal type will be stored in the ‘assets’ directory. Some animal types will have multiple assets, such as one asset for a blue sheep, one asset for a sheep with a specific mutation, one asset for a sheep who’s sick, etc. This is the goal, but I may not be able to reach it in time.

General Code: Most of the code will be stored in main.py. As of right now, the plan is to have all of the code pertaining to the simulation is main.py in what seems like the most intuitive, chronologically organized way possible.

**Version Control Plan**

All files are being backed up to GitHub. I already have a repo (currently private so that my mentor, Asad, can look at the ideas I've come up with) that is currently storing all versions of my code.

A screenshot of a computer

Description automatically generated with medium confidence

**Modules**

I will be using cmu\_112\_graphics. One feature I would like to implement requires graphing. Time allowing, I will use cmu\_112\_graphics and code my own graphing software. If I run out of time, I will use matplotlib. I won’t know whether I will use matplotlib until I decide whether I will be implementing this feature or not.