Final Game Submission

Finalized Game Document, Github Code Submission, Demo Video

Due: Wednesday May 3 23:59:59 EST
Submission: Submit this document as a pdf on Canvas and include the link to your github**.** Also include this document as your main README on your project's github homepage.
Congratulations! You have made it to the final submission for your 2D Game Engine project! This final submission will include a finalized game document, the source code and resources for the game itself, and a 30+ second video demoing the game. The requirements are as follows:

How to Run:
Download all files
Execute command py main.py

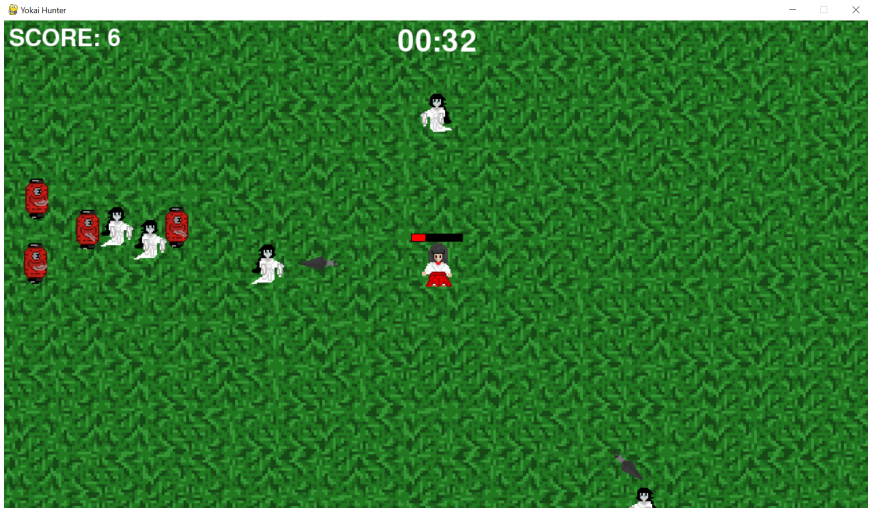
Team Members:
Evan Sok
Thomas Personett

GitHub Repo: https://github.com/TomPy19/Yokai-Hunter-WIP

The Game Document

Part 1: Introduction

- What is the name of your game?
  Yokai Hunter
- Create an iconic or descriptive screenshot or image of your game



- Include instructions on how to download, run, and play the game.
    - What software versions are needed to run your game?
        - Python, pygame
- List the game controls.
  W - Forward
  A - Left
  S - Down
  D - Right
  ESC - Close Game

Part 2: Game Design

Describe your game and its design! Here are some questions you may answer (highly recommended questions are marked with an asterisk)

- Mechanics/Technology
    - What is your game's gimmick and how does it contribute to your game?*
      Random spawning monsters. We had a much more interesting gimmick but were not able to implement it in time. As for random spawning monsters, The player has to be more conscious of their movement since they will spawn out of nowhere and, they have to navigate their way out of the horde of monsters trying to take the least amount of damage.

    - How does your game differ from other games in the same genre?*
      Unfortunately, the current version of the game has the same formula as most in the genre: attacking mobs and defeating the boss.

    - How does your game utilize the strengths of the game engine you're working with?

Certain functions within pygame proved to be useful in simplifying some parts of our algorithms/functions for the game.

- Story
    - Does your game have a story or theme? What are they?
      Kinda sorta. Our story was in the line of: MC getting sent back in time to defeat other monsters for some goal(probably for the sake of humanity) and getting a rare item from each boss that will be useful to save humanity. However, our current product does not portray that story as well as we would have liked it.

    - Who is the protagonist of your game, and what is their motivation?
      Mari Chitose. Her goal is to defeat the monsters to save humanity.

    - Who or what is the antagonist of your game, and what are their goals?
      The antagonists are all the monsters in the world. Their goal is the eradication of mankind.

- Player Experience
    - What kind of challenges will the player face, and how will they be overcome?*
      The hordes of monsters that try to attack the player, they have to be smart with their movement while attacking them.

    - What kind of rewards will the player receive for progressing through the game?*
      Satisfaction of living long enough to defeat the boss.

    - What kind of audio and visual elements will enhance the player's experience?*
      No audio is implemented, however there are plenty of visual elements. Such as three mobs with unique animations, a boss, and a kunai that is thrown.

Part 3: Game Design Changes

Tell the story of your game's design and how it has changed over time. You may copy and paste from your project proposal and milestone, but be sure to include updated details. Answer these questions:

What was the original design and concept when you proposed the game?
A 2D roguelike game similar to Vampire Survivors with additional attributes similar to Risk of Rain.

How and why did the design change over time?
Massive time and skill difference.

What was your original plan for the game mechanics? How and why have they changed?
Our plan for the game was going to be far longer than it currently was, having three different maps, random chests, different items with unique characteristics, actual boss battles, etc. However, all these would require more time than we would have thought.

What was your original plan for the game gimmick? How and why has it changed?
The original game gimmick was the interaction with chests that contain random, unique items/weapons. These would make it so that each run would be unique from the rest and would require the player to work with what they have. This gimmick was changed because of the troubles implementing different weapon animations and what effect they would have on monsters.

In answering these questions, explain the reasoning and justification behind any changes from the original proposal. What challenges or limitations did you face? What factors led to those changes?
We did not coordinate our time accordingly, did not expect the difficulty of such a task, and did not have the skills/knowledge required to complete it within the deadline given. We were able to reach our Minimum Viable Product, however we were not satisfied with what it was.

Part 4: Game Development & Documentation

Outline your game's codebase. You may do this with a visual diagram or a descriptive list.

- Include all classes and major functionality methods.
    - Character.py - Base parent class for entities
    - Controller.py - Obtains input
    - Enemy.py - Child of Character, used for enemies
    - Kunai.py - Class for weapon sprites
    - Mari.py - Child of Character, used for main player
    - Model.py - Used to hold many functions for the game
    - Talisman.py - Item class for a healing item
    - Ui.py - Handles ui elements
    - Main.py - Handles main game loop
- Include details on how your game:
    - receives player input (Controller)
        - Reads keyboard presses and handles movement of player
    - stores the state of the game (Model)
        - The state of the game is held in many different class files
    - updates the screen/renderer (View)

        - The class files also all have custom draw methods

Answer these questions:

- Are there any major bugs or flaws in your game we should be aware of? (Undocumented issues/bugs will result in a score deduction)
- What tools did you use to facilitate collaboration or code versioning (e.g. git/github, VSCode Liveshare, etc.)
    - GitHub, Discord, Google Drive

Part 5: Group Member Roles, Tasks, and Performance

Provide a finalized timeline of the development of your game.

Milestone 1: March 29

Completed by Thomas:

- Basic player movement and monster that follows player: March 15
- Figure out monster ai: March 27

Completed by Evan:

- Full concept art for all entities and items: 27 //map still in progress

Milestone 2: April 12

Completed By Thomas:

- Mechanic done where a weapon comes into contact with entity: March 15

- Full sprites and level design complete (unpolished): April 12 //STILL IN PROGRESS
- Figure out all item management: March 27 //POSTPONED
- Figure out damage scaling math: March 22 //POSTPONED
- Figure out core gameplay loop mechanics: April 12 //POSTPONED

Final Game Submission: April 26

Completed by Evan:

- Full sprites and level design complete: April 12

Completed by Thomas:

-health bar and damage mechanics
-implemented different monsters with varying speeds, health, and damage
-boss battle triggered by timer

Completed and polished game
Completed Game Document game

Source Code

Provide your final source code on github.
Include your Game Document in your github's README. Be sure to include instructions on your main github page on how to download, run, and play your game. Provide software versions (python, pygame, any additional libraries).

Demo Video

https://youtu.be/XzxytgQ7Xlg

Final Class Session

You will demo your game on the final day of classes to myself and your classmates. We'll have an open demo session for the duration of the final period (Thursday, 5/4 3:00-5:30 pm). If you have multiple teammates, have the game ready to run on each laptop. Don't forget to bring your laptop chargers.