**BREADTH OF THE WILD**

**Post Mortem**

1. **Datasheet**

* Publisher: Silly Goose Game Studio
* Developer: Silly Goose Game Studio
* Number of Full-time developers: 6 at ship date
* Number of Contractors: 0
* Length of Development: 7 weeks, 4 days
* Release Date: May 9th, 2023
* Platform: PC
* Development software used: Excel, Unity, GitHub Desktop, Krita, Audacity
* Development hardware used: Ranged over course of development from 2-3.60 GHz CPUs with 16-512GB RAM.
* Notable Technologies: N/A
* Project Size: Approximately 100,000 lines of source code for game, editor, and associated tools; 3,000 lines of scripts; 20 animations, 40 object and actor meshes.

1. **Overview**

* Final project of class CS4700 Game Development, Spring 2023 at California State Polytechnic University, Pomona
  + By the requirements of Prof. Tony Diaz
  + Work together in teams of 5-6 people to develop our own video game using the Unity Game Engine as shown in class.
  + Each person on the team will take on one of the indie game developer roles discussed in class to get a feel for separation of responsibilities in game development.
  + Every person on the team is still responsible for some coding responsibilities.
* Game must meet the following requirements:
  + Must fit one of the genres discussed in class (at least partially) and have the attributes of that genre.
  + Must be unique with innovative gameplay (not just copy and paste of someone else’s work)
  + Must have a single player mode.
  + Must be developed in Unity.
  + Must be feasible to complete before the due date.

1. **What Went Right**

* Prototype early and often. Plus lots of iteration – tweak until the code work
* Example: The goose ability to fight
  + Originally goose can only peck their opponent.
  + Is it fun for our main character to only have pecking as the only form of attack?
  + What if they can kick? Use their wing to slap enemy?
  + How about picking up the power/ learn from bosses?
  + Final – Goose finally will have the ability to learn new techniques from bosses.
    - Goose can begin to slap the enemy left and right.
    - Goose can pick up stick or butter knife to increase damage to use.
    - Etc.
* Everyone plays at least once/week.
* Constant internal feedback
  + Avoid attempting to please everybody since the time of development is very short.
  + Encourage to find bugs as early as possible.
* Keeps everyone up on design and project timeline.
* Gives everyone pride and ownership.
* Extreme Flexibility: small group.
  + Make decisions very fast.
  + Set up temporary test team over a weekend.
* Unity game engine was well designed and easy to use.
  + Prof. Tony provides Unity’s panel arrangement template, so everyone is on the same format.
  + Prof. Tony taught everyone how to use Unity the same way so everyone begins at the same starting point.

1. **What Went Wrong**

* Extreme Ambition
  + Ambitious project for such a short amount of time
  + Nobody worked on Unity before or has any prior knowledge of developing games.
    - We underestimate the game development process.
    - We overestimate our ability.
  + Design droves too much
    - Sometimes programmers can do things faster than tool + designer.
* Example features.
  + Lack of time due to the length of the class.
    - Significant ramp up time on tools and ideas.
  + Lack of motivation due to diversion of effort for other classes and projects.
  + Short on initial resources
  + Only 1-2 core developers on preproduction
* Basic nature of game environment not appreciated and understood.
* Sharing code
  + Unity offers very hard-to-use Version control.
  + The file is too large to upload to GitHub as is.
  + Other cloud storage services are highly compromised.
* Required lots of tweaking of engine.
* Many unfinished.
* Developers left waiting for features.