

Gold and Glory

**Fantastic Four © 2014**

Created in Experimental Game Design, Fall 2014

at Rensselaer Polytechnic Institute

Developers

**Alexander Foo** (fooa@rpi.edu) – Character and Enemy Art

**Dustin Tsui** (tsuid@rpi.edu) – Enemy AI

**Peregrin Garet** (garetp@rpi.edu) – Character and UI Programming

**Reginald Franklin** (frankr4@rpi.edu) – HUD and Environment Art



*Gold and Glory* is a bullet hell following in the footsteps of classics in the genre such as Touhou and Ikaruga, but with a cooperative twist. The two players control members of a ragtag mercenary squad seeking out fame and fortune wherever they please. At any given time, one player is the Attacker and the other the Defender, while the enemies shoot two different types of bullets. The Attacker returns fire while the Defender utilizes his shields, whose affinity can be swapped between one type of bullet or the other, to provide the Attacker a path of least resistance. At the same time, for every bullet the Defender’s shield absorbs, he gains ammo, which he can use when the players decide to swap roles, creating a powerful dynamic between the two players which they can use in combination with their unique mechanics and specials to destroy everything in their path. *Gold and Glory* is built in Unity with pixel art.

**Game Philosophy**

**Philosophical point #1: The WHY FACTOR:Why create this game? Why would someone want to play it?**

We wanted to make a game with relatively simple, straightforward gameplay that still encouraged highly skilled play. Shoot-’em ups (commonly referred to as “Shmups”) have a natural sense of skill progression- the relatively simpler task of increasing difficulty by adjusting stats like bullet speed and firing rate and number of enemies allows us to focus attention on characters and bosses, which enhances the game’s replayability and the feeling of satisfaction on beating a level.

All of the ideas we came up with when brainstorming featured multiplayer, but we wanted to separate ourselves from the competitive fad and try for a more pure cooperative experience, where players are rewarded for teamwork, rather than individual skill. A large part of our design process has been considering how to make multiplayer rewarding and making sure that we don’t stray from our roots of encouraging players to work together and think as a team.

*Gold and Glory* drifts closer to the days of arcade games from a gameplay perspective. People can first engage in the struggle that is successfully defeating all the bosses, and then start working on optimizing their score.

**Philosophical point #2: Immediate and Long Term Projected Socio-Cultural Project Impact?**

In both short term and long term, we are addressing absences in contemporary gamer culture. In the short term, we want to raise awareness of the potential of cooperative gameplay. While games such as *Call of Duty* and *League of Legends* leave no question that competitive multiplayer is in no danger of extinction, even the games with stronger cooperative elements like *Smash Bros* and *Dark Souls 2* tend to use it as a secondary feature at most, with the main focus of the gameplay remaining individual skill. *Gold and Glory* aspires to utilize the potential camaraderie and catharsis of team play to remind players why working as a team is its own reward.

In the long term, we want to encourage players to remember that in a time before AAA titles and decade-long series, there were one-shot arcade games that had simple and clear mechanics while maintaining depth of gameplay. While it is entirely possible that arcades as they once were can never come back thanks to changes in technology and culture, I do think it plausible that a new generation can use games not just to demonstrate dominance or overcome overwhelming odds, but to bond and work together.

**Philosophical point #3: Predecessor or previous games/ distinctive factors in this genre**

In the realm of games with cooperative elements, I mentioned *Smash Bros* and *Dark Souls 2*. Of these examples, *Smash Bros* is probably the stronger example since the game can be played PvE. Our game shares with *Smash* in that there are a variety of pre-designed characters for the player to choose from and the dynamics of a team are based on which 2 of those characters are brought into play. However, our game does not engage in any of the customization that one finds in that game, or *Dark Souls 2* for that matter.

In terms of shoot ‘em ups, specifically bullet hells, the most well-known example is *Touhou*, although our game is perhaps closer to *Ikaruga* in terms of the polarity system, where the defending ship can absorb one kind of bullet but not the other. Like both of the above, players have an individual score, though at the end of the day in this game it’s the team score (a combination of the individual scores with some added variables based on team performance) that remains as the high score.

For arcade games, probably the two closest relatives are *Galaga* and *Space Invaders*, which are very simplified versions. Neither of the above are multiplayer, but *Galaga* in particular shares in the difficulty of dodging enemies and taking down targets.

Distinctive factors in the genre that our game touches on are primarily high score, large numbers of enemies with distinctive and flashy bullet patterns, and bosses with multiple phases.

**Philosophical point #4 Target Audience**

Our target audience is hardcore gamers or arcade gamers- those who don’t necessarily play for the achievements or the flashy graphics, but for the thrill of defeating a difficult foe or clearing a particularly intense map. Furthermore, since the game is based on same-computer multiplayer, this is not something a person just picks up and plays when bored. I would see this as a game a couple of friends can pick up and play for a while to pass some time, moderately analogous to party games like Wario Ware or Mario Party except specifically for 2 players.

# 

# **Game Overview**

### **What is the game?**

We are creating a two player bullet hell that takes a unique cast of four off-beat characters and hurls them against a multitude of opponents, over the course of three levels and their respective bosses.

### **Why create this game?**

We wanted to make a cooperative game that had gameplay depth and was accessible for anyone willing to take the time to get better at the game. In addition, we wanted to make something that hailed back to the days of the older arcades, whose focus was on replay and learning the game rather than item accumulation or skill trees, while at the same time taking advantage of more recent developments in technology and gameplay.

### **Where does the game take place?**

The game takes place in an unnamed fantasy universe, where mages, monsters, and shopkeepers struggle to coexist. Specifically, our game features a large estate called Fog Manse owned by an elemental mage who has called together a counsel of monsters with the plan of betraying the shield wavelength to a large mage city.

### **What do the players do?**

The players battle a multitude of monsters firing a large volume of projectiles. One player servers as the Attacker, the other player the Defender. The Attacker can shoot until he runs out of ammo, the Defender has a shield with one of two polarities which can transmute enemy bullets of that polarity into ammo that can be used when the Defender becomes the Attacker, and can swap the shield polarity at any given time. The two players can also swap roles at any time

### **How many characters are involved?**

There are 4 protagonists and 3 bosses, with a multitude of nameless minions with varying attributes.

**What is the main focus?**

The initial objective is to clear each level and win the game; the secondary objective is to get the highest score possible. We don’t expect that players who don’t have significant experience with this game mode will be able to beat any of the levels on the first try; a large part of the game for the players initially will be learning the levels and the patterns as well as developing strategies with their favorite character combinations, whereas later they will figuring out how to optimize their play to get the highest score.

# **Design History**

This document began in early October of 2014 as the team brainstormed and argued its way to getting the design principles nailed down, addressed a large number of gameplay issues, and came up with a schedule to make sure that the game is ready on time.

Using this Design History section, you as the reader can get a picture into how our design and development processes went and where we made decisions. For us as developers, the Design History is a useful referendum about the decisions we’ve made and a convenient tool to make sure that any finalized thoughts are kept safe and visible in case there are questions or concerns.

## Version 1.10 added in the philosophy, game over view, and schedule sections.

Version 1.11 modified frame of production and timeline to make them more viable

Version 1.12 modified frame of production and timeline a third time to make them more representative of the skills involved

Version 2.0 added in basic character and enemy information

Version 2.1 renamed Wizard Mansion to Fog Manse

Version 3.0 added in rendering information

Version 3.1 added in further rendering information based on new understanding of the relationship between 2D and 3D Unity

Version 4.0 added in some more specific numbers in terms of number of types of enemies and their properties

Version 4.1 added in specific mechanics for Boss 1

Version 4.2 added in specific mechanics for Boss 2

Version 4.3 added in specific mechanics for Boss 3

Version 4.4 added in changes to Boss 3 based on revisiting the Lightning premise

Version 5.0 added information about the game engine, as it became more finalized

Version 5.1 added specific information about Enemy and Spawn as they became more powerful as a result of increased need

Version 6.0 added in musical sound. After contact with Johnathan Reed and our first sample, we had a better understanding for potential sound-related scope.

Version 6.1 added in representatives from our BFXR sound database that we’re using for this game.

Version 7.0 added in further musical sounds, based on what John sent us.

Version 7.1 added in objects based on the now-present functionality of coins.

Version 7.2 added in final numbers confirming that there are in fact 4 playable characters and 26 enemy types and 3 fully operational bosses as well as 3 levels that move comfortably in a game loop.

**Frame of Production (per Syllabus)**

**10/1:** GDD with Frame of Production, Schedule, & Outline of Individual Responsibilities and Deadlines

**10/8:** Updated GDD with Game Overview Sheet, Refined Storyboard, 5 citations of games/websites/reading/literature/film that have influenced the game

**10/15:** Evidence of Progress, Definitive Schedule, Refined Game Overview Sheet & Image, Philosophical Question section

**10/22:** Evidence of Progress in line with blocked out schedule

**10/29:** Playtesting plans due, Three Week Trajectory

**11/5:** Playtesting Summary results and recommendations, Printed Project Posters

**11/12**: High quality edited HD 1920x1080 H264 mp4 video promotional trailer

**11/19:** Informal reviews

THANKSGIVING

**12/3:** End of the road

**Schedule/Responsibilities**

**Peregrin:** Protagonist/HUD/UI Code

**Dustin:** AI Code

**Alex:** Protagonist/Boss Art/Stage 1 Minion art

**Reginald:** Background/Stage 2 & 3 Minion art/Sound

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date** | **Peregrin** | **Alex** | **Dustin** | **Reginald** |
| **Oct 8** | Punch Knight start | Punch Knight | Movement patterns for minions | Level 1  Background |
| **Oct 15** | Punch Knight finish | Dragon Sitter | Rain and Sprinkler Bullet Patterns | Level 1 Boss Background  HUD v1.0 |
| **Oct 22** | Dragon Sitter start | Stage 1 Enemies | Boss 1 | Level 2  Background |
| **Oct 29** | Dragon Sitter finish | Stage 1 Boss | Boss 2 start | Level 2 Boss  Background |
| **Nov 5** | Coins,  Animations,  Tutorial | Tinker | Boss 2 finish | Level 3  Background  Character select |
| **Nov 12** | Tinker start | “Mirror Mage” | Boss 3 start | Level 3 Boss  Background  Title screen |
| **Nov 19** | Tinker Finish | Boss 2 | Boss 3 finish | Level 2 Baddies |
| **Nov 26** | Mirror Mage start | Boss 3 | Debugging/  Stretch room | Level 3 Baddies |
| **Dec 3** | Mirror Mage Finish | Last minute stuff | Debugging/  Stretch room | Last minute stuff |

# 

# **Feature Set**

## General Features

Linear story/game play

2 players

Offbeat fantasy world

2D graphics

Bullet hell

## Multiplayer Features

2 players

Local (same computer)

2 sides of the screen correspond to 2 players

## Gameplay

Two players, one in attack mode and one in defense mode

The player in attack mode can shoot bullets and use his special

The player in defense mode can absorb enemy bullets that have the same affinity as his shield, and swap the shield affinity to make sure that his shield matches the affinity of the incoming bullets.

Destroyed enemies drop coins, which either player can pick up to get experience points. Players can level up 3 times, each time gives them a special bonus stat.

There are 3 levels to the game, each with a boss. These levels are designed to go from easy to medium to hard, with bosses that go from medium to hard to expert.

There is also a tutorial that players are encouraged to utilize in their first run through, to learn how the controls work.

# **The Game World**

## Overview

*Gold and Glory* takes place in an unnamed fantasy world where magic and technology, mages and monsters, struggle to coexist. Our players, a mercenary group in desperate need of funds, have committed themselves to a hail mary attack on Fog Manse, the estate belonging to a particularly violent and wealthy consortium of mages. Both players engage the enemy as one of four characters, shooting up enemies, shielding their comrade, and collecting coins for power ups.

## Combat

*Gold and Glory* utilizes a bullet polarity system where enemies fire bullets of type A or B and the Defender has shields which can be swapped between types A and B. If a bullet of type A strikes a shield of type A, the Defender absorbs that bullet. When the attacking player is running out of bullets, it may be time for the Attacker and Defender to swap roles, such that the previous Defender is now using the bullets he’s been gathering while the previous Attacker utilizes his shields to gather some more bullets for the next swap.

Waves

The game features a wave system that spawns on a timer, so that there is a constant yet survivable stream of enemies coming in. This system comes together as a result of play testing combined with prior bullet hell precedent to try and optimize the volume of bullets on the screen. Additionally, by color coding enemies and creating bullet patterns that encourage players to work together and think as a strategic team both to survive and pick up as many bullets as possible, we seek to make our game both cooperative and immersive.

## The Physical World

### Overview

Overall, the physical world that the player experiences is the estate of Fog Manse, where the Elemental Mage has his stronghold. This includes the external area, the courtyard, and the inside.

### Key Locations

The key locations include the forest path leading to Fog Manse, the courtyard of the Fog Manse, and the interior of Fog Manse.

### Scale

The scale is pretty small- due to the volume of items we need to have on the screen and the size of the screen, we need to compress things in order to fit everything and still have the game be playable. With that in mind, we tried to make sure everything was large enough to be recognizable.

### Objects

We have coins that the player can pick up to earn power ups. We have enemies that the player shoots. We have bullets flying from both players and the enemies. Boss 3 sends out unlockable projectiles which the player has to avoid.

### Time

Waves of enemies in each level will spawn based on the time elapsed during a level.

## Rendering System

We will use Unity’s 2D rendering system, which places textures on the face of a 3D object facing towards the camera.

### Overview

Our game is rendered in a top down format using a camera which watches the faces of 3D objects in its viewport. The objects in the game (ie bullets, players, enemies) are rendered as textures on the 3D objects.

### 2D/3D Rendering

We used Unity’s 2D engine, which is effectively top-down 3D.

## Camera

The camera is top-down, so it is at the center of the canvas and moved back in the Z direction around 10 units.

### Overview

The camera never moves- all movement in the game is other objects. For example, the background scrolling might make it look like the camera is moving, but that is only a visual effect.

## Game Engine

### Overview

Our game engine works with passing time to generate enemies. Starting with a character select screen and going into game and level transitions, a manager in the background monitors enemies, player stats, and the presence of bosses to determine background and HUD display. Each player has an identical script that monitors their current stats and a specific script based on which character they showed. Enemies are largely homogenous, with all sharing the same scripts but different prefabs having different specific stats like speed or health.

### Game Engine Detail #1

There is a script attached to our camera that generates all the HUD materials. It also is responsible for notifying other scripts, specifically that responsible for backgrounds and that responsible for enemy spawning, when it is appropriate to take an action.

**Game Engine Detail #2**

Each player has two scripts, a script that is identical for both and a script that is individual to the character. You might think of these scripts as the brain and the hands- the stats script is responsible for monitoring inputs, health, speed, and a number of other relevant variables. Meanwhile, the character script is notified when its time to take actions like attacking or swapping shield affinity and handles the character-specific relevant tasks.

### Collision Detection

Depending on the nature of the object, our game generally speaking uses box colliders or rendering bounds. Enemies and their bullets and player characters use box colliders specifically sized and located to make the player feel more comfortable. Meanwhile, bosses use render bounds to make them a little easier to hit and player bullets use render bounds because they come in all different shapes and sizes.

# **The World Layout**

## Overview

The player’s view of the world is composed of three linear regions. The first region is the estate grounds, where the player begins learning the mechanics, takes on low level enemies, and faces the Slime Mage. The second region is the courtyard, where the player takes his skills to the next level and fights higher difficulty opponents, as well as taking on the Gargoyle. The third region is the inside of the manse, where the player will face the highest intensity of bullets and the most durable enemies, with the game culminating in an epic three phase battle against the Elemental Mage.

## Level 1

Level 1 is the estate grounds, where we can find all manner of low level enemies. These enemies generally have simple movement and firing patterns and there aren’t many of them at a time. Fight your way between rocks, shrubs, and dirt until you come across the infamous cheap mercenary Slime Mage, who will summon a barrage of slimes and other enemies to take you on in addition to its own set of fearsome attacks including your first exposure to the rain pattern.

## Level 2

Level 2 is the courtyard, where fog drifts across cobblestone and enemies are more frequent and dangerous. Begin your venture into the dimly lit layer and fight your way through larger and more powerful waves of opponents, including some more complicated movement patterns and shooting patterns as well as new color variations. Level 2 concludes with the Gargoyle, loyal and fierce protector of the estate, who will try his best to ram you down when he’s not trying to shoot you in the face.

Level 3

Level 3 is inside Fog Manse, where the strongest, fastest, toughest enemies are preparing for your approach. Face wave after wave of intensive resistance from the widest array of opponents yet, including some very tricky movement and shooting patterns. And Level 3 features the culmination of the entire game, the 3 phase Elemental Mage, who will over time introduce you to the dangers of Fire, Ice, and Lightning.

# **Game Characters**

## Overview

We have four playable characters: the Punch Knight, the Dragon Sitter, the Tinker, and the Mirror Mage. Each character has an entirely unique play style, with highly varied skill sets and specials. One of our biggest goals in creating the game was to have personable and engaging characters so that the players could pick a character they feel attuned to.

## Creating a Character

When we were first creating the characters, we took several factors into account. For one, we wanted to make the game in the fantasy genre, which generally means warriors, monsters, and mages. But we also wanted to venture outside that cliche, and two of the best known games for their offbeat characters are Borderlands and Team Fortress 2, so we thought about the most interesting aspects of their

## Enemies and Monsters

As referred to above, there are 26 types of minions and 3 bosses. These minions are divided into three categories: bat, whisp, and slime. Within each of these categories there are 9 colors except one, each of which enemy has at least slightly different statistics. Enemies start basic, red and blue reflecting the red and blue bullets they are shooting. As the game progresses in difficulty, the visual connection between characters fades and the player must learn to recognize different characters and be prepared for their attacts.

# **User Interface**

## Overview

The primary user interface is the character selection screen, in which the players pick their characters in a format similar to that of popular games like *League of Legends* or *Super Smash Bros*. The game canvas interface involves using the two relevant Xbox controllers, one for each player, to navigate the game. When the game ends, the player is shown a screen with their score and then returned to character select, and the loop begins once more.

## User Interface Detail #1

Character select is, as I said, formatted like *League* or *Super Smash Bros.* By this I mean that icons are available for each character, and the players click on a character to see an expanded information set which they can use to make a decision. Once both players have selected a character, the game can begin.

## User Interface Detail #2

After the game, the players are faced with a score screen for a few seconds before returning to the title screen. The focus shouldn’t be on the score exactly, but on the achievement, so high score isn’t shown and the focus is on the return to the title screen, and the continuation of the activity.

# **Musical Scores and Sound Effects**

### 

## Overview

We have four pieces of music (one a remix of another) that we play throughout the game. In addition, we have a set of sound effects that include attacking, enemies being destroyed, and swapping.

## 3D Sound

We’re not using 3D sound because it wasn’t necessary and sounded on in a 2D environment like we had here.

## 2D Sound

We have 4 songs (Character Select, Main Theme, Fantasy, and Evil). The first is self-evident, playing during title and character select. Main Theme plays in Level 1, Fantasy in Level 2, and Main Theme again in Level 3, with Evil during boss fights.

In addition, we have sounds including coin pick up, swap, power-up, player attack, and enemy explosion.

## Sound Design

Ideally, sound design is going to be working together with visual responsiveness to create an effective system of feedback that enables the player to feel in control and understand what is going on at any given time.

# **Multiplayer Game**

## Overview

Two players share a PC monitor, each on an Xbox controller, and play together to defeat the enemy. The goal is to optimize their cooperative experience.

## Max Players

2 Players at a time

## Customization

Each player can pick one of four characters, creating 8 potential set ups, which allows for diversity of gameplay and experience.

## Persistence

No, that would have caused a ton of chaos given the number of enemies shooting and dying and the same thing occurring to the player. We also made the group decision to avoid the sense of progression through repetition. We reward our player through score, skill, and fun, rather than handing them achievements or items to saturate their experience.

## Saving and Loading

There is no loading or saving because we felt it was out of scope to cover effectively. Especially as we neared the end of the project, it became quite transparent that that simply wasn’t high enough on the list of priorities to be covered at that time.

# **Character Rendering**

## Overview

Characters are pixel art, animated as a texture on a single face of a 3D object. They are rendered onto a background to make them clearly visible and distinct.

## Character Rendering Detail #1

Animations are generally divided between different states and different affinities: for example, the shields for the characters change color when they change affinity, and Tinker’s gadgets change color based on which attack mode it is in.

# **World Editing**

## Overview

The world editor is in several parts depending on what you’re trying to change. GUI Manager handles almost all of the HUD and interface, including title screen and character select. The spawn script is responsible for all minion waves and their timings, with Enemy script responsible for their bullet patterns and movement patterns. Lastly, PlayerStats determines the stats for each character.

## World Editing Detail #1

GUI Manager monitors modes- for example, if mode is zero, we’re at title screen. If mode is two, we’re in gameplay. We can use it to determine what icons show up, what the HUD looks like in game, and what numbers and variables are displayed.

## World Editing Detail #2

Spawn and Enemy work together to make the waves work. There are 4 bullet patterns and 22 movement patterns. Each enemy can pick between the two affinities for what it shoots, and there are 26 different enemies with varying damages, speeds, and healths.