

All the King's Men for UTM CSCI 495 Fall 2018

Steven Alford & Ruchik Patel

Abstract

All the King's Men is a 2D, Action-RPG video game. It follows an ousted king, vying to regain control of his castle from the traitorous council members who've been overtaken by evil, black magic. However, the king must defeat the minions that stand between him and the council in order to harness their dark magic and turn it against them. He also knows that, within the castle walls, there are artifacts of great power that would give him an advantage, although, he would have to bypass the safeguards put in place to protect them in order to gain their help. He has only a short window of time to defeat the council before the entirety of his kingdom is thrust into darkness.

1. Introduction

This project is intended to make a 2D, Action-RPG (source), which will closely resemble features from games such as Dark Souls (source), the Legend of Zelda: a Link to the Past (source), and The Binding of Isaac (source). The main features that will be focused on will be fluid movement, use of abilities, and customized progression. The game will be gameplay driven with little to no story, and the player will have a choice in what way they want to progress through the game. A leveling system (source) will also be in place to give the player as much freedom as possible in choosing how they wish to approach different aspects of the game.

All the King's Men is for players looking for an action-based game that enjoy the dexterous movement of dodging multiple enemies, and the fast-paced action of an Action-RPG. The project intends to cater to people who enjoy the mechanical requirements of games like Dark Souls and those who are fans of the progression in games like The Binding of Isaac. The game will be designed to be difficult at first, but, once the player has grasped the mechanics, should be able to be completed with effort and determination.

1.1. Background

To give the reader some background information and increase readability, provided below are some definitions of words that will be prevalent throughout the document.

- **Action-RPG:**
emphasizes real-time combat (where the player has direct control over characters) over turn-based or menu-based combat. These games often use action game combat systems similar to hack and slash or shooter games.
- **Leveling System:**
a system that allows a player to choose what skills or abilities to level up in such a way that allows them control over how their character gets stronger over the course of the game.
- **Mechanics:**
anything in a game that the player uses to progress or anything the game uses to defeat the player. Generally, how the game works; the logic.
- **Random Number Generation (RNG):**
the generation of a sequence of numbers or symbols that cannot be reasonably predicted better than by a random chance, usually through a random-number generator.

1.2. Challenges

1) **Resource Management:**

Because we are a two man group, we will need to delegate tasks between all roles equally to ensure that the game is developed properly in a timely manner. Our focus will mainly be on the enemy AI, the boss fight, and the player's progression, so most of our time will be spent developing those aspects of the game. If time permits, we will attempt to implement audio and story into the game.

2) **Enemy AI:** Neither of us have experience with AI development, so this will be a slight challenge, although the basic nature of what our AI are doing should aid us in this regard. It should not be hard to implement them once we get a base case for our enemies.

3) **Boss Fight:** Since we will only have one boss fight, and the AI will be based around some of our enemy mechanics, this should not be hard. However, it does hinge on the execution of our enemy AI, which we may have trouble with in the beginning.

2. Scope

Outlined below are the basic, necessary goals that need to be met for the project to be complete and the stretch goals for the game that would be nice to have if the opportunity arises.

2.1. Basic Goals

- Player Resource Management (Health, Levels, Gold, etc)
- 4-6 iterations of enemy types
- 1 boss enemy
- Player Character Interface
- Player HUD
- Map Design

2.2. Stretch Goals

- 10-12 enemies
- 5-6 bosses
- Multiple castle levels
- Player abilities
- Playable classes

2.3. Requirements

- 1) Player movement
- 2) Player interaction with the game environment/map
- 3) Player interaction with the enemies
- 4) Player interaction with the boss
- 5) Player progression - item pickup/gold usage
- 6) Player stats - Display/Updating
- 7) Graphical User Interface - Display/Updating

2.4. Interface Mockups

- Figure 1 - This figure shows the tileset being used to generate the map, the enemies, and the players in the game interface.
- Figure 2 - This figure shows the user interface between the player and the game. The top left shows the player in which area of the castle they are currently in, the top right shows the amount of gold the player has, and the bottom left shows how much health the player has. In the layer beneath this, the game window shows the player within the game environment.



Figure 1. Tileset

3. Player Character

This section will be devoted to the player character. All abilities, resources, and the gameplay loop will be discussed in this section in full detail.

3.1. Attributes

The player will have six attributes. Each attribute will give the player some sort of advantage given the level of the attribute. All of the attributes leveled up using gold, which will increase the player's level and value in that attribute.

- **Health:**
Description: The player's total hitpoints. This determines how much damage the player is able to take before being killed.
- **Speed:**
Description: Speed is how fast the player is able to move through the environment.
- **Attack Speed:**
Description: Attack speed is how quickly the player is able to shoot subsequent projectiles.
- **Damage:**
Description: Damage is how much health the player is able to remove from the enemy per attack projectile.
- **Lifesteal:**
Description: Lifesteal is how much health the player is able to steal back from the enemy, adding it to their own current health. This does not increase the player's total health, or overheal the player.
- **Range:**
Description: Range is how far the player is able to shoot before their projectile dissipates.

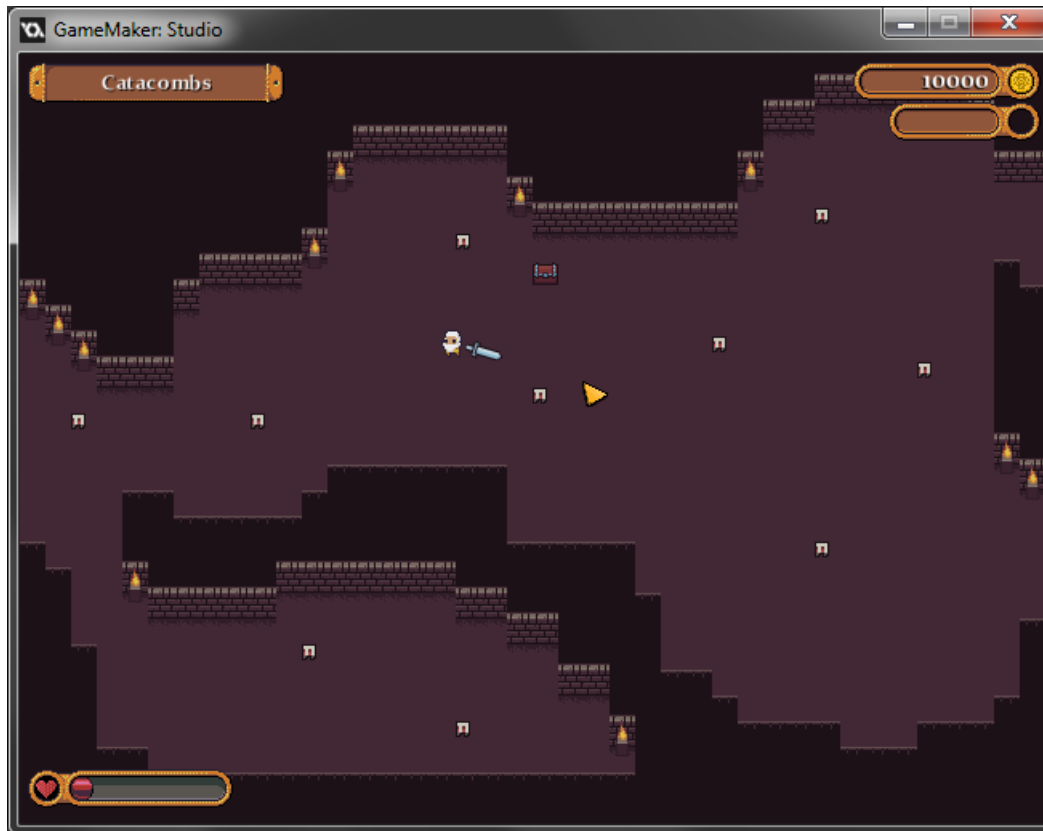


Figure 2. Interface Mockup

4. Enemies

- Parasite:**
 Description: The parasite is a melee-type enemy that is generally found in large numbers, which they use to swarm the player, overwhelming them and soaking up damage that would otherwise be dealt to other enemy types.
- Rogue:**
 Description: The rogue is a basic ranged enemy, throwing daggers at the player at a decently fast rate. This enemy type has no overarching mechanic, just a basic ranged enemy.
- Shieldbreaker:**
 Description: The shieldbreaker is a tanky enemy type that does damage by throwing its shield at the player. If the shield hits the player, it will stun the player for 3 seconds, allowing other enemies to either catch up to the player, or deal more damage while the player is unable to act.
- Plague Puddle:**
 Description: The plague puddle does not do damage, but it does slow the player using a projectile that, once fired, will stick on the ground. If the player walks through this puddle, they will be slowed by roughly 40% of their current movespeed, allowing other enemies to catch up to the player.
- Headhunter:**
 Description: The headhunter is a more complex enemy than the basic Rogue enemy mentioned earlier. This enemy throws a skull, which, upon impact, explodes into 8 directions, with each projectile able to damage the player.
- Necromancer:**
 Description: The necromancer does not do damage, but, instead, revives nearby enemies. This enemy is a priority target, since it can revive defeated enemies multiple times.

5. Boss (The Keeper of the Archives)

5.1. Phases

- **Phase 1:**
Description: The Keeper will start off shooting 4 projectiles in a cone towards the player. Each of these projectiles will damage the player if they connect. This phase continues from 100% health to 70% health.
- **Phase 2:**
Description: At 70% health, the Keeper will shoot off a persistent projectile that will continue to bounce around the room until the fight has been completed. This projectile is much larger than the projectiles that the Keeper shoots off in the cone, and will do significantly more damage to the player. This phase will also introduce the shielding mechanic, as well as the positive runes on the ground that will allow the player to reduce the shield on the Keeper. This phase lasts from 70% health to 40% health.
- **Phase 3:**
Description: At 40% health, the Keeper will send out 2 additional persistent projectiles, and it's shield will activate again. This phase also introduces the negative runes, which, when stood on will damage the player and increase the damage of the Keeper's projectiles.

6. Project Timeline

- (Summer 2018) Project planning and prototype. Research for free/CC tileset to use, get ideas for music. Implementation of tileset and player animations.
- (07-10-18) Begin the project planning phase.
- (08-23-18) Begin project proposal.
- (08-26-18) Project proposal complete.
- (08-27-18) One of the level layouts have been completed (Great Hall). Responsibilities have been delegated.
- (09-02-18) The player can now shoot projectiles and the dynamic lighting has been completed.
- (09-17-18) Implemented the health bar: will accurately reduce the player's health and visually represent the value on the screen. Completed and implemented the exploration song. Loaded in all art for enemies, along with their weapons/projectiles. Created some extra projectiles that were missing (puddle for the plague puddle, shield for the shieldbreaker). Created all of the runes for the Keeper fight.
- (09-24-18) General enemy idle and attacking states implemented. Enemy health bars implemented with damage from player projectiles correctly updating the enemy's current health. Some work done on specific enemy attack scripts (parasite, rogue)
- (10-01-18) Remaining enemy AI completed (Headhunter, Shieldbreaker, Plague Puddle). Player now has 1 additional state: stunned. Plague Puddle enemy drops puddles on the ground that slows the player, shieldbreaker enemy throws shield that stuns the player. Necromancer still a work in progress.
- (10-15-18) Player level up interface implemented. Player can now level up health, speed, attack speed, damage, lifesteal, and range.
- (10-31-18) Necromancer complete. Keeper of the Archives AI started.
- (11-06-18) Keeper AI phases completed. Runes AI in progress.
- (11-10-18) Keeper fight completed (runes, phases, etc).
- (11-16-18) Presentation.

6.1. Toolkits

- 1) GameMaker: Studio - This was the primary tool in our set of tools. We used this as the basis of our game, as it provided many tools to aid in our quick development time. Gamemaker Studio is a full 2D game engine that is enabled by its proprietary scripting language GML. This engine allowed us the ability to develop our game quickly and efficiently without worrying about building our own game engine from scratch and then attempting to build a game on top of that.

7. Results

Overall, we accomplished all of the goals that we set out to accomplish. While there were a few things that could have been done better, or differently, we believe that we did the best we could do with the amount of time and resources available to us. With time, we believe that this project could go a lot farther, given the time and effort put into the planning phase and the possibility of expansion upon the ideas and implementations in the game thus far.

7.1. Future Work

For future work, we would like to add 10-12 new enemy types, 5-6 more bosses (Keepers), multiple castle levels, player abilities, 3 or 4 new playable classes, and the ability for New Game+. As we progress through these goals, we may add additional work to this section.