

README for Hallow Way
By Hanniee Tran, Kaylie White, and Joshua Bonilla
Professor Howard
DIG4715C
Spring 2020

Basic Gameplay Requirements:

- Game is played in the first person view in Unity 3D.
- The game runs properly in 1920x1080 resolution.
- The game has a Win State (triggered upon interacting with the cat in the final room) and a Lose state (triggered upon being killed by the ghost guard).
- The guard (the ghost) detects the player while the player is in the hallway.
- The guard reacts to the player's action by turning to face their direction and launching projectiles at them.
- The guard can trigger the Lose State by removing the player's health until they are dead.
- The guard cannot be permanently neutralized or removed from the game by the player.
- There is a Win Object in the final room. It is the cat and it is interacted with to achieve the Win State. The player cannot get into the final room before defeating the guard.
- The game's UI clearly indicates the player's health and the boss's health when engaged in combat.
- Music starts with the start of the game. There is also different music for when the game is over.
- Sound FX is used when the sword is swung and projectiles are shot by the guard.
- Gameplay is between 60 and 90 seconds.

Art Requirements:

- All art is made by the team members.
- There are at least 7 unique and visible 3D models (the cat, the carrot sword, the ghost, the pillars, the box obstacles, the torches, the sign, and the player character).
- The game has 2 unique animations. One is on the guard (floating animation) and the other is the weapon swing.
- There are 2 unique particle effects in the game. One is the torch fire and the other is on the ghost's projectile.
- Menu contains art that is stylistically consistent with the game (same characters and objects).
- Art does not have significant issues.

Code Requirements:

- All code is implemented by team members.
- The game uses at least one NavMesh. It is in Empty object called Navmesh inside the hierarchy.

- The game is plausible and unpausable by pressing the 'P' key. Pausing displays pause menu.
- Game does not crash or break.
- The game supports keyboard and mouse controls (keyboard WASD and mouse click for attack).
- The game has three significant mechanics: 1) projectiles from the ghost guard, 2) picking up the sword to use for melee combat, and 3) completing objectives to unlock doors.

Tech Requirements:

- The game has 3 distinct areas: 1) starting area with pickup, 2) hallway area with guard, 3) ending room where Win State is triggered.
- Main Menu has "Start Game", "About", and "Exit Game" options.
- About Screen has a How to Play section, a Credits section, and a "Main Menu" option.
- Pause Menu is triggered by 'P' key and displays the same info as About Screen. Also tells the player how to unpause.
- Game Over Screen - triggered by the Lose State and is clear. Has "Main Menu" and "Exit Game" options.
- There are no dead ends in the levels or menus.

Other Game Elements:

- All team members collaborated on the following:
 - World Construction
 - Progression Management
 - Enemy Behavior
 - Player Controller
 - Sound design
- Requirements for all game were included as follows:
 - Windows game only.
 - No copyright material was used.
 - Game's controls are clearly indicated.
 - Game can be completed within the required time limit.
 - Game can be easily completed.
 - Game is not obnoxious, offensive, or problematic.
 - Game can be closed at any time with the ESC key.