Possessed: Demonic Pact

Author: Nick Soare Stanciu
Tutor: Guillermo NWDD García Romero
Professor: Joan Arnedo Moreno
Degree In Computer Engineering
Computing Itinerary

15/01/2023

Credits/Copyright

Original Author: Ninu (Nick) Soare Stanciu – **2023**

FINAL DEGREE PROJECT WORKSHEET

Project Title:	Possessed*: Demonic Pact*
Author:	*Nick Soare Stanciu*
Teaching Collabolator:	Guillermo NWDD García Romero
PRA:	Joan Arnedo Moreno
Deadline (mm/aaaa):	*01/2023*
Degree:	*Grado de Ingeniería Informática* *(Degree In Computer Engineering)*
Final Project Area:	*TFG - Videojuegos (Videogames)*
Language:	*English*
Keywords:	*Aventure, Horror, Gothic*

Abstract (250 words or less):

The main objective of the project is to develop a first playable version of a videogame as part of the final degree project of the computer engineering career, applying all the knowledge acquired in the career and learning to use the new tools necessary to complete this development.

The videogame is called “Possession: Demonic Pact”, a hack-and-slash, side-scroller game with a 2D pixel art aesthetic. The game takes place in a gothic setting, set in a dark world plagued by monsters and demons. The main character, Adam, will have to fight his way through different types of enemies and avoid falling off cliffs as he fights for control of his own body against Azazel, the demon that possesses him, via an interesting gameplay modifier implementing the possession mechanic.

The project has been developed using the Godot Engine, a free open-source game engine with strong 2D capabilities, using an incremental methodology, creating small versions of the game that gradually expand the scope of features and mechanics implemented in the game.

Abstract (in Spanish, 250 words or less):

El objetivo principal del proyecto es desarrollar una primera versión jugable de un videojuego como parte del trabajo final de grado de la carrera de ingeniería informática, aplicando todos los conocimientos adquiridos en la carrera y aprendiendo a utilizar las nuevas herramientas necesarias para completar este desarrollo.

El videojuego se llama “Possession: Demonic Pact”, un juego del género hack-and-slash, side-scroller con una estética 2D pixel art. El juego tiene lugar en un entorno gótico, en un mundo oscuro plagado por monstruos y demonios. El personaje principal, Adam, tendrá que abrirse camino a través de diferentes tipos de enemigos y evitar caer por los acantilados mientras lucha por el control de su propio cuerpo contra Azazel, el demonio que le posee, mediante un interesante modificador de la jugabilidad.

El proyecto se ha desarrollado utilizando Godot Engine, un motor de juegos gratuito de código abierto con fuertes capacidades 2D, utilizando una metodología incremental, creando versiones pequeñas del juego que amplían gradualmente el alcance de las funcionalidades y mecánicas implementadas en el juego.

Dedication

I would like to dedicate this project to my better half, Cristina and to my mother, Ecaterina, as a humble gesture of gratitude for supporting me throughout my academic career.

Quotes

“Evil is Evil. Lesser, greater, middling… Makes no difference. The degree is arbitrary. The definition’s blurred. If I’m to choose between one evil and another… I’d rather not choose at all.”

- *Andrzej Sapkowski, The Last Wish (The Witcher Saga)*

“…fear is a luxury…”

- *Niklas Alparós-Lilah, Purgatory Diaries: My Soul To Take*

Index

Una o varias páginas con el índice de la memoria, que tiene que incluir los títulos de los capítulos (estilo de texto **Título 1**) así como sus secciones de primer nivel (estilo de texto **Título 2**) y sub-secciones de primer nivel (estilo de texto **Título 3**), sin profundizar más en la estructura. Una vez actualizado el índice, quitar las negritas del texto.

Figures & tables

List of images, tables, graphs, diagrams, etc., numbered, with titles and the pages on which they appear.

Figures Index

Tables Index

1. Introduction

1.1. Introduction/Preface

The videogame industry has changed and evolved drastically in recent history. It's relatively new concept of media, compared to other forms of media like movies, books and theatre, which have all been around for much longer than videogames, especially the former two. That being said, videogames have evolved at an exponential rate, going from a couple of pixels forming a simple ping-pong game, to various pixels forming rough depictions of real-life objects (like tanks, water, walls, etc.) and, finally, recent games having thousands of polygons on the screen, bringing to life a plethora of hyper-realistic pieces of art.

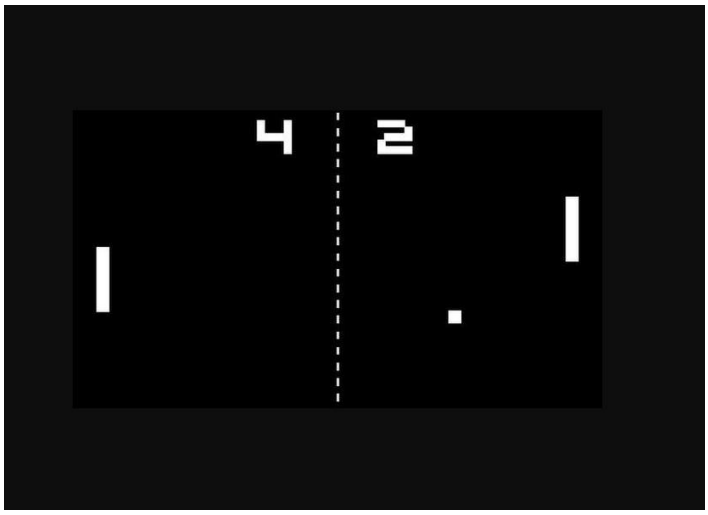

Figure 1: Pong (1972)

Figure 2: Battle City (1985)

Figure 3: The Witcher 3: Wild Hunt (2015)

Nowadays, videogames have massive budgets, being treated similar to big movie productions, many of which have started to cast established actors as their main and support characters. Today's multimedia platforms (Movies, TV Series, Videogames, etc.) now function as a "symbiotic entity", each borrowing and improving over each other living together in a certain harmony. Some successful videogames get a live-action movie and/or TV series adaptations, while some successful movies also get official licensed videogames to either expand or reimagine the plot of the movies.

Figure 4: Normal Reedus: Death Stranding (2019)

Figure 5: Mads Mikkelsen: Death Stranding (2019)

With the development of this project, the main objective is the creation of an entry into this vast world of videogames, in the shape of a modest retro-stylized videogame, in order to get a sense, as a single person, of what it feels like to be part of this industry. This videogame will certainly not change the history of the gaming industry. However, the goal is to offer a working product, free of errors and most important of all, enjoyable to the user.

1.2. Description/Definition

The main objective of this project is the development of a videogame, gaining knowledge on both how to start developing a videogame as a programmer but also learning how to use new tools, solve possible errors and fit the development duration into a limited predetermined timeframe using mainly free software, assets and other resources.

The starting point is quite literally “from square 1”, since there is no prior experience in this field, other than knowledge of the programming language Java and playing videogames from the perspective of a gamer. Therefore, the motivation of the author of this videogame is to acquire the ability to create a working videogame from scratch as a means for personal and professional growth for the author.

Possession: Demonic Pact, is a 2D side-scrolling hack-and-slash game with a pixel art aesthetic. The game is set in an alternative dark world, where demons and other unholy creatures and spirits roam the lands, terrifying the masses. The main plot of the game revolves around Adam, a monster hunter for hire that got possessed by a demonic entity, known as Azazel, during his last hunt. At that point, his main goal shifted from hunting night creatures for money to fiercely fighting for control over his own body against the demon, in order to survive.

The game’s mechanics will be pretty basic, with the exception of a couple of caveats. The main character, Adam, will be able to move left and right in order to move through the areas, as well as crouch and jump in order to climb on platforms, jump over cliffs, avoid enemies, traps and projectiles. Adam’s health is measured by a “Life” meter that decreases when damage is inflicted by enemies or traps. An interesting aspect of this character, is that he is actively possessed by the demon, therefore, in certain scenarios, the demon can gain control over the character’s actions putting him in precarious situations and even cause his death. This control will be controlled by a “Demonic Corruption” meter that increases or decreases by completing certain actions in the game. The player can fight to keep the demon at bay or try to find a way to work together against to assure Adam’s survival. Adam’s main attack is a mele attack with his main weapon, the Hunter’s Sword

The level is passed when Adam reaches the Gothic Church, a place warded with inscriptions and spells that tame any kind of demonic activity, while also offering shelter from the terrible night creatures. If Adam’s health reaches zero, or he falls of a cliff (instant death) the game will be over. The demonic bar slowly fills with the pass of time and can be lowered by finding pickup items called “demonic seals”, that temporarily lowers the control of the demon over its host. When the demonic corruption bar is full, Azazel, the demon, gains control of Adam, changing his aspect to a corrupted vision of Adam surrounded by black smoke. While the demon has control, he realizes a series of random movements that the player cannot counteract, possibly leaving Adam in a dangerous situation, like falling off a cliff or running into an enemy without attacking.

As far as pick-ups go, they are pickable objects that appear randomly through the level, and can be picked up by Adam to control his demonic passenger. When clearing a level, the player is given an option to pick one of various side-grades with trade-offs, that offer advantages and/or disadvantages.

Adam will fight multiple types of enemies, both walking and flying, and also a final Boss in the form of Azazel, the demon possessing him. In order to beat the game, the player must keep an eye on Adam's health and demonic corruption levels. The definition of the Health & Corruption system:

- The Health and Corruption bars reset once a level is finished and Adam reaches a Gothic Church
- The Corruption bar can be reset/lowered by picking up demonic seals
- The Health bar cannot be refilled, unless a level ends.

At the end of the game, there will be a result page, indicating if the player won or lost the game, the time spent, enemies killed, seals collected. There is also the intention of adding some kind of collectible items in the game if the timeframe allows it. This would be some kind of gothic religious or demonic artefacts found in hard to reach or hidden areas of the levels. Each of these items have a "point" value, for example winning would grant you 1000 points, losing would grant you 0 and each seal collected would give you 10. So, there will be a column indicating the total points gained in the current playthrough.

1.3. General Objectives

1.3.1. Main Objectives

Objectives of the application/product/service:

- Design a 2D videogame, playable on Windows (PC) platform.
- Develop it from the initial planning stage until a fully playable version is published.
- The playable version must be free of game-breaking bugs or errors.

Objectives for the client/user:

- The user must be able to run the game and play it.
- Complete a level while gathering the items and pick-ups available without being blocked in any area, with no option to advance.
- Have user-friendly gameplay mechanics and controls so as to make the game playable and enjoyable, despite its challenging gameplay modifiers.

Objetivos personales del autor del TF:

- Review and compare different game development engines, in order to choose the best candidate that adapts to this project's necessity and the developer's experience or lack thereof.
- Apply programming concepts learnt in the degree and learn new concepts, programming languages and methodologies needed to finalize the scripting of this game's functions.
- Understand what Game Designing and Developing means and learn about its implications, fix issues that appear while developing a game and learn to fix and move past them within a given time frame.
- Identify the important core elements of the game, and prioritize them over other optional elements, features and goals, in order to respect the agreed final publishing date, in order to deliver a fully functional quality product, free of errors.
- Apply Agile Methodology to the planning and implementation of this project.
- Learn and/or improve the authors planning and organizational skills in a complete project with a strict time frame.
- Gain experience in videogame development projects.

1.3.2. Secondary Objectives

- Design or modify free-to-use character and enemy models with graphical editing software to give a personal touch to the game's sprite assets.

- Implement a soundtrack to give the player that feeling of tension and horror that a videogame in this genre should offer.
- Create a script and story for the game, with lore elements described to the player via collectible items, in order to paint a more complete picture of the world this game takes place in and learn about its history.
- Learn different programming languages and game engines.

1.4. Methodology & Work Process

Videogame development does not have a defined set of rules, but usually the designing and planning of a videogame includes a team of developers working on it, each having a certain roles and tasks assigned to them. In the case, the whole project will be designed and developed by one person, that has to assume all the possible roles in this project, in order to deliver a stable functional product.

This project will consist of developing a small-scale 2D videogame with basic mechanics and art style, in order to fulfil the scheduled publishing date, with the possibility of further expanding the game's scope at later date. To accomplish this, an incremental Agile Methodology will be used, focusing on implementing small versions of the game with limited functionality. It involves continuous improvement at every stage, implementing objectives one by one, up until the last version which will include all the desired elements, levels and mechanics of the final game. With the delivery of each PEC in mind, each PEC will be treated as an agile Sprint with the goal of delivering a playable version at the end of each one, excepting PEC 1, which mainly consists of research and the project planning.

1.5. Planning

A Gantt diagram has been used for the project planning, specifying the start and end date of phase of the project, and detailed the tasks that fall under each of them:

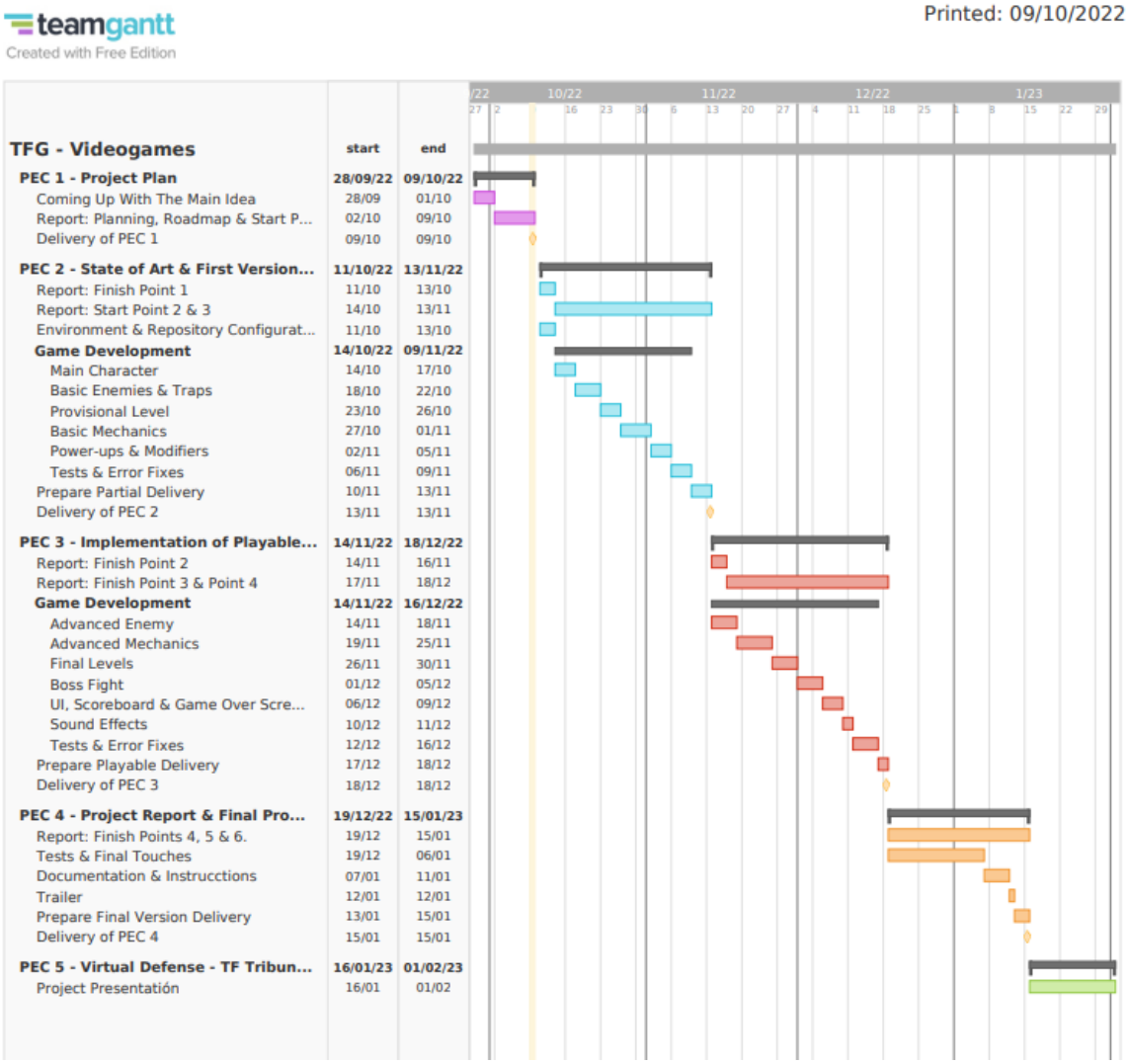

Figure 6: Planning Gantt Chart

The tests & error fixes phases of each stage (PEC) will include user testing with feedback that will be compiled and used for error fixes and quality of life improvements.

Task				**Start Date**	**End Date**
TFG - Videogames				28/09/2022	01/02/2023
PEC 1 - Project Plan				**28/09/2022**	**09/10/2022**
Coming Up With The Main Idea				28/09/2022	01/10/2022

Report: Planning, Roadmap & Start Point 1				02/10/2022	09/10/2022
Delivery of PEC 1				09/10/2022	09/10/2022
PEC 2 - State of Art & First Version Of Project				**11/10/2022**	**13/11/2022**
	Report: Finish Point 1			11/10/2022	13/10/2022
	Report: Start Point 2 & 3			14/10/2022	13/11/2022
	Environment & Repository Configuration			11/10/2022	13/10/2022
	Game Development			***14/10/2022***	***09/11/2022***
		Main Character		14/10/2022	17/10/2022
		Basic Enemies & Traps		18/10/2022	22/10/2022
		Provisional Level		23/10/2022	26/10/2022
		Basic Mechanics		27/10/2022	01/11/2022
		Pick-ups & Modifiers		02/11/2022	05/11/2022
		Tests & Error Fixes		06/11/2022	09/11/2022
	Prepare Partial Delivery			10/11/2022	13/11/2022
	Delivery of PEC 2			13/11/2022	13/11/2022
PEC 3 - Implementation of Playable VersiÃ³n				**14/11/2022**	**18/12/2022**
	Report: Finish Point 2			14/11/2022	16/11/2022
	Report: Finish Point 3 & Point 4			17/11/2022	18/12/2022
	Game Development			***14/11/2022***	***16/12/2022***
		Advanced		14/11/2022	18/11/2022

		Enemy			
		Advanced Mechanics		19/11/2022	25/11/2022
		Final Levels		26/11/2022	30/11/2022
		Boss Fight		01/12/2022	05/12/2022
		UI, Scoreboard & Game Over Screen		06/12/2022	09/12/2022
		Sound Effects		10/12/2022	11/12/2022
		Tests & Error Fixes		12/12/2022	16/12/2022
	Prepare Playable Delivery			17/12/2022	18/12/2022
	Delivery of PEC 3			18/12/2022	18/12/2022
PEC 4 - Project Report & Final Product				**19/12/2022**	**15/01/2023**
	Report: Finish Points 4, 5 & 6.			19/12/2022	15/01/2023
	Tests & Final Touches			19/12/2022	06/01/2023
	Documentation & Instructions			07/01/2023	11/01/2023
	Trailer			12/01/2023	12/01/2023
	Prepare Final Version Delivery			13/01/2023	15/01/2023
	Delivery of PEC 4			15/01/2023	15/01/2023
PEC 5 - Virtual Defense - TF Tribunal				**16/01/2023**	**01/02/2023**
	Project Presentation			16/01/2023	01/02/2023

Table 1: Planning

1.6. Budget

BUDGET		
Software		
	Godot (Game Engine)	0€
	Github (Repository)	0€
	SourceTree (Git Tool)	0€
	GIMP (Image Editor)	0€
	Shotcut (Video Editor)	0€
	Audacity (Audio Editor)	0€
Hardware		
	Personal PC	1.500€
Manpower		
	Project Author, the only developer working on the project (estimated 300h at an average of 30 €/h)	9.000€
Other		
	Freeware assets, sprites and sounds	0€
TOTAL		**10.500€**

Table 2: Budget

1.7. Structure Of The Rest Of The Document

Brief description of the other chapters of the Memory.

Explanation of the contents of each chapter and its relationship with the work as a whole.

- **Point 1**: Introduction
- **Point 2**: State of Art & Market Analysis – This chapter presents an analysis of the videogame genre, it's past, present and future state and will analyse the different game developing engines and design tools.
- **Point 3**: Proposed Project – This chapter presents the main objective of the videogame, definition of the characters, enemies, game mechanics etc.
- **Point 4**: Design – This chapter presents the technical documentation of the project, the development environment, tools and an inventory of all the assets that will be used during the whole duration of the project.
- **Point 5**: Implementation – This chapter presents the user instructions needed to install and launch the videogame.
- **Point 6**: Demonstration – This chapter presents the videogame's controls scheme and the different tests that have been realized during the development, showing off the minimum requirements of the videogame. This chapter's content can be used as a user guide.
- **Point 7**: Conclusions & Future Plans – Finally, this chapter presents a summary of the whole development process, pointing out what goals have been achieved or not, what has been learned from said process and also plans for future versions of the videogame.

2. Market Analysis

2.1. Target Audience & User Profiles

The main target audience for this videogame are

Delimitación del público objetivo al que se dirige el producto/servicio según sus características (demográficas, culturales, etc.). Puede ser interesante explicar el escenario/contexto de mercado, tecnológico, social, etc. en el cual se enmarca el TF.

Listado y descripción detallada de los perfiles tipo de los usuarios que se espera que hagan uso del producto/servicio.

2.2. Competition/Background (Or Theoretical Framework/State Of The Art)

Possession: Demonic Pact is a videogame based in a vast genre of 2D side scrolling & platformer games with 2D pixel art aesthetic.

- Background: One of the most popular games and pioneers of this genre would be the 1985 Super Mario Bros. videogame, where the main character advances from the left side of the screen to a scrolling 2D level avoiding pitfalls, jumping over obstacles and fight against enemies.

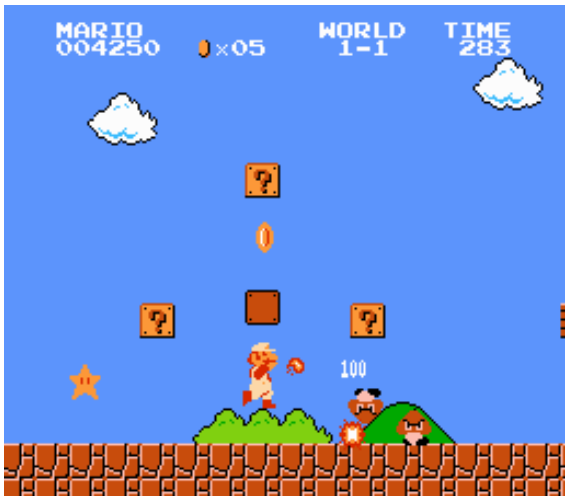

Figure 7: Super Mario Bros. (1985)

Another important game to the author, both from the point of view of a gamer and also as an inspiration for this game, would be the 1997 videogame Castlevania: Symphony Of The Night for PlayStation 1. Once again, we have a 2D side-scrolling gameplay, not un like the Super

Mario game. However, in this particular game, and also all of the similar game of the Castlevania franchise, we observe a beautiful gothic aesthetic present in both the characters' design and the backgrounds that showcase this dark world inhabited by Dracula, and his legions of monsters and vampires. Unlike the previous example, this game's world is dark and gritty, with a more believable story, albeit a fictional supernatural one.

Figure 8: Castlevania: Symphony Of The Night (1997)

- Global vision of the current situation of the scope of the proposed project: Currently there are numerous videogames inspired by Castlevania. One the one hand, we have pixel art styled games like Hollow Knight (2017), that explore the simplicity and beauty of a modern look on 2D pixel art videogames and fluid responsive gameplay.

Figure 9: Hollow Knight (2017)

On the other hand, we can also see the essence of this genre moved to a 3D environment, in games such as Bloodborne (2015), thar combines the gothic aesthetic and feel of the Castelvania games, applying it to a beautiful dark 3D world and "Dark Souls-like" unforgiving gameplay mechanics.

Figure 10: Bloodborne (2015)

In the case of Possession: Demonic Pact, the author aspires to achieve the look and feel of a retro gothic Castlevania-like aesthetic and atmosphere, with simplistic gameplay mechanics, due to the lack of experience as a videogame developer. However, the main hook will be the demonic possession that influences the player's movements, that offers an interesting gameplay modifier, that hopefully proves to be intriguing enough for users to want to try it out, despite having other, much more technically advanced, videogames to choose from.

3. Proposal

3.1. Definition Of Objectives

The main objective of this game's development is to implement the core elements necessary for its playability:

- Generate 2D side scrolling levels with a pixel art aesthetic, featuring:
    - Obstacles, cliffs and traps that can damage or kill the player.
    - Enemies that can move around the level, walking or flying, with the capacity of attacking the player with mele attacks or projectiles.
    - Two different levels, featuring different types of enemies, challenging the player to employ different tactics for each level.
    - A Boss battle against the game's demonic antagonist, Azazel (the demon possessing the main character).
- Main character:
    - Basic movements and attacks.
    - Health bar: if it reaches 0, the player loses the game.
    - Corruption bar: if it reaches MAX. level, the demon takes control over the player's movements for a brief period of time, via limited random moves, jumps and/or attacks, that can put the player in danger or even cause the character to die.
- Level checkpoints: once the player reaches the gothic church, the level is complete, and the health and corruption bars get reset for the next level.
- Endgame: the main mechanic of the game is to pass the 2 levels without dying, then proceed to a Boss fight where the player fights the demon:
    - If the player's health drops to 0, the player loses and it's "GAME OVER!".
    - If the boss' health drops to 0, the player wins and beats the game.
    - If neither health bars drop to 0, the player restarts level 1, with increased difficulty, and has to reach the boss level again. The boss's health remains the same, but his strength will be increased, in order to give the game an "infinite loop" feeling comparable to a never-ending nightmare one cannot wake up from.
    - Endgame strategy: Defeat the boss as soon as possible, before the levels and/or the boss become too difficult to fight.
- Main Menu, Help Instructions & End Screen: Implement a main screen before starting the game, a Help screen showing the game's main controls and basic instructions, and an end screen, where the player can see the results of his playthrough.

Secondary objectives:

- Pick-ups: During the level, pick-ups can spawn when defeating an enemy, in the form of a demonic seal, which is an item that resets the Corruption bar when picked up.
- Side-grades: at the end of each level, the player has a choice between different side-grades that come at the cost of either health or corruption bar's cost. These can be used to tame the demon's hold over Adam at the expense of health, or grant new abilities (such as fire-ball projectiles) at the expense of the corruption bar filling faster.
- Implement collectibles: These items will grant points, useful for the results at the end of the game, while also providing details such as brief texts of the world's lore.
- Challenges: Implement challenges for the player that will provide more points if completed. They would be informed on the results page and/or on the pause menu. An optional goal could be to have it "pop-up" during the game at the exact moment of completing the challenges, similar to PlayStation's trophies system.
- Unlockable: Implement unlockable such as alternate skins/costumes, weapons, etc., via collectables and/or challenges.

3.2. Business Model

The game itself will rely on using sprites, sounds and other assets from 2D game developing communities, available under CC licenses, in order to generate the games graphical elements.

3.3. Marketing Strategy

This project is mainly an adventure of self-learning for the author, in order to gain experience in the field of videogame development. Therefore, the objective is not monetary gain or commercial distribution of the videogame, since it will be uploaded, in its entirety, publicly on GitHub's repositories and its code will be available to anybody willing to check it out, play it and even try to implement and improve features of it. It has the potential to become a community driven project where other developers and designers could help to build upon the base game, to either provide improvements and new features, or generate an entire new game as a spin-off using the base game's source code and game mechanics.

As for the marketing strategy, it consists of advertising the release of the game on social media platforms, via an attractive videogame trailer, highlighting the game's unique selling points and showcase some core elements of the gameplay, joined by a narrator explaining the main plot of the game and the interesting relationship between the protagonist, Adam, and the demon possessing him, Azazel. Once the trailer is published, it will be joined by a social media post across all available platforms, confirming the expected release date, in order to gain the interests of potential buyers,

along with the proposed price for the game. There is also the intent of implementing pre-order bonuses such as a Making-Of Video, Concept Art, T-Shirts (showcasing the game's pixel art), a personal "Thank You!" message from the publisher, among others.

For the future of the game's development, beyond the initial delivery, the goal is to adapt the game to mobile platforms (such as Android and/or iOS) and consider publishing them on the App Stores, either available for a fixed price, or having the option to enjoy it as a F2P title (free to play) with in-game advertisements, with the added option of paying a fee to unlock the "Ad-Free" version of the game.

4. Design

Figure 11: Teaser Image

4.1. General Architecture Of The Application/System/Service

The general functionality of the Godot game engine basically consists of Scenes and Nodes. Each Scene has a main or Primary Node and all other Sub-Nodes extends from it. A scene can also contain as many instances of child sub-scenes as needed. To each of the Nodes we can attach a Script file containing its code, in order to define the node's desired functionality.

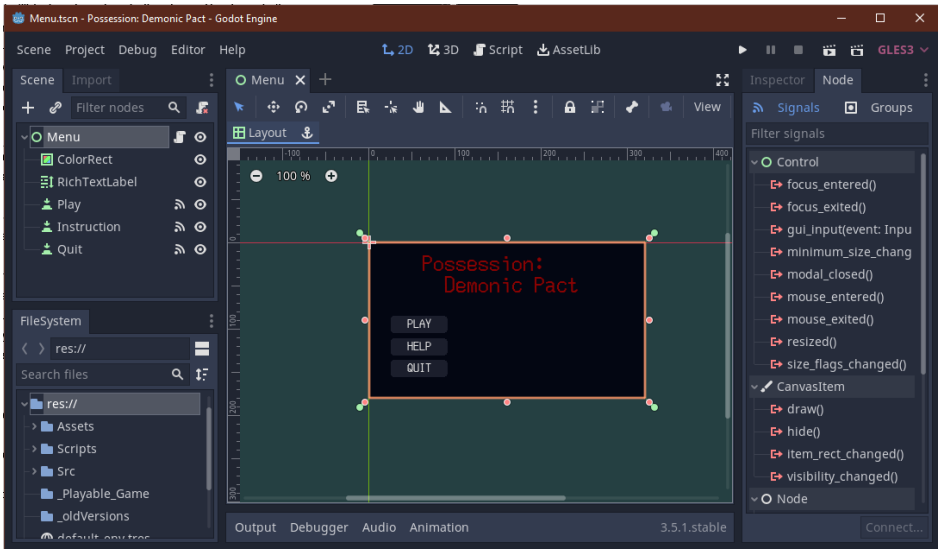

Figure 12: Godot Workspace

Finally, these nodes communicate between each other via signal calls. Therefore, when a certain event/action occurs (such as a collision, pick up an item, etc.), the node in question emits a signal and it's caught by the corresponding function from the script (related to this signal call) and it determines the node's response to said event. For example, the event "collision", when the player collides with an enemy, trap or projectile, causes the reaction of the player's sprite flashing white and lowering the health bar.

4.2. Information Architecture & Navigation Diagrams

4.2.1. Story & Background

Possession: Demonic Pact is a game that takes place in a supernatural gothic setting, set in a dark world plagued by monsters and demons. The main character, Adam, is a hardened monster hunter that inadvertently gets possessed by a powerful demon as an aftermath of his last hunt. The demon calls itself Azazel and vows to make Adam's life a living nightmare. Every day Adam must fight Azazel over control of his own body, whilst at night Azazel haunts Adam's dreams, causing vivid nightmares that drives Adam closer to the edge of madness. Adam's only hope is to seek refuge within the walls of the Gothic Church, a place warded with inscriptions and spells that tame any kind of demonic activity, while also offering shelter from the terrible night creatures. Ultimately, he must face his Demon both in the physical ward and in the dreamworld, if he hopes to gain back control of his life. This is the fiercest battle Adam has ever faced yet.

4.2.2. Basic Game Mechanics

The game's mechanics are quite basic, with the exception of a couple of caveats. The main character, Adam, will be able to move left and right in order to move through the areas, as well as crouch and jump in order to climb on platforms, jump over cliffs, avoid enemies, traps and projectiles.

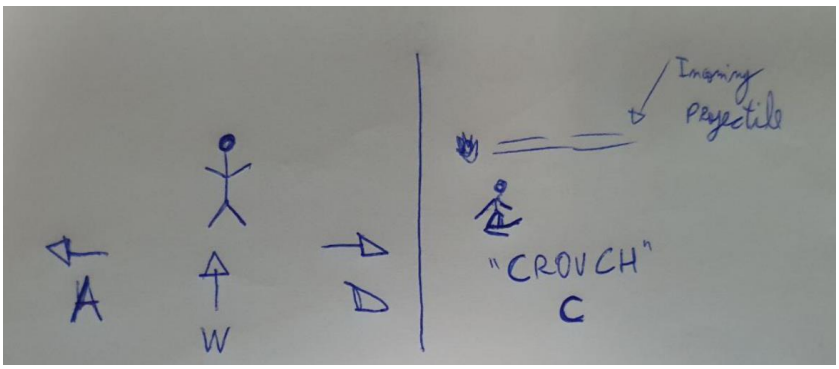

Figure 13: Basic Movement Sketch

Adam's health is measured by a "Life" meter that decreases when damage is inflicted by enemies or traps. An interesting aspect of this character, is that he is actively possessed by the demon, therefore, in certain scenarios, the demon can gain control over the character's actions putting him in precarious situations and even cause his death. This control will be controlled by a "Demonic Corruption" meter that increases or decreases by completing certain actions in the game. The player can fight to keep the demon at bay or try to find a way to work together against to assure Adam's survival. Adam's main attack is a mele attack with his main weapon, the Hunter's Sword by clicking the Left-Mouse Button (**LMB**).

Figure 14: Ranged Attack Sketch

Adam can also acquire the ability to shoot a ranged attack (Fire Ball), which can be done by Right-Clicking (**RMB**) the mouse anywhere on the screen, to send a fire ball in that direction. The ranged attack can damage enemies and is the only type of attack that can destroy traps.

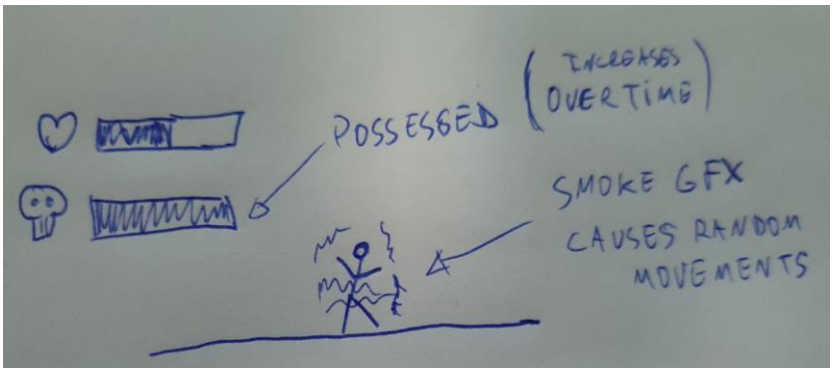

Figure 15: Health & Possession Bar Sketch

Each game level is passed when Adam reaches the Gothic Church, a place warded with inscriptions and spells that tame any kind of demonic activity, while also offering shelter from the terrible night creatures. If Adam's health reaches zero, or he falls of a cliff (instant death) the game will be over. The demonic bar slowly fills with the pass of time and can be lowered by finding pickup items called "demonic seals", that temporarily lowers the control of the demon over its host. When the demonic corruption bar is full, Azazel, the demon, gains control of Adam, changing his aspect to a corrupted

vision of Adam surrounded by black smoke. While the demon has control, he realizes a series of random movements that the player cannot counteract, possibly leaving Adam in a dangerous situation, like falling off a cliff or running into an enemy without attacking.

4.2.3. Levels

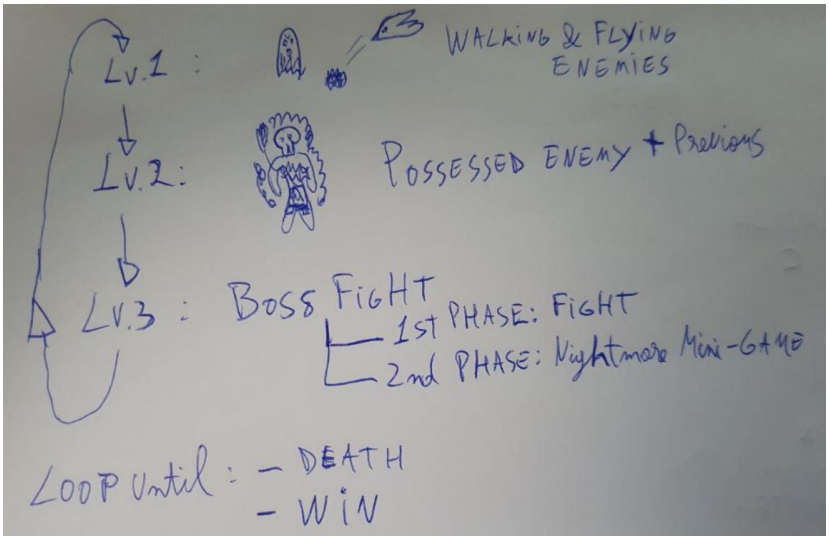

Figure 16: Levels Sketch

The game consists of 3 levels:

- Level 1:
    - The first level that includes basic enemies (the walking & flying variety) and traps. The objective is to get to the Gothic Church before the possession affects Adam too much, possibly causing his demise, while also fighting the enemies and avoiding traps to keep his health bar from becoming empty.
- Level 2:
    - This second level, introduces a stronger enemy type (demonic possessed variety) that slowly hurts and corrupts Adam if he gets too close. The objective remains the same, get to the safety of the Gothic Church.
- Level 3: Final level that consists in a 2-Phase Boss Fight against Adam's nemesis, the demon Azazel during one of Adam's vivid nightmares:
    - First Phase: Physical fight. The player must attack Azazel to lower his health bar. If his health reaches 0, the player wins. If the player dies, the Game is Over. If the player manages to survive the fight, but doesn't defeat Azazel, the game moves into the Boss Fight's second phase.

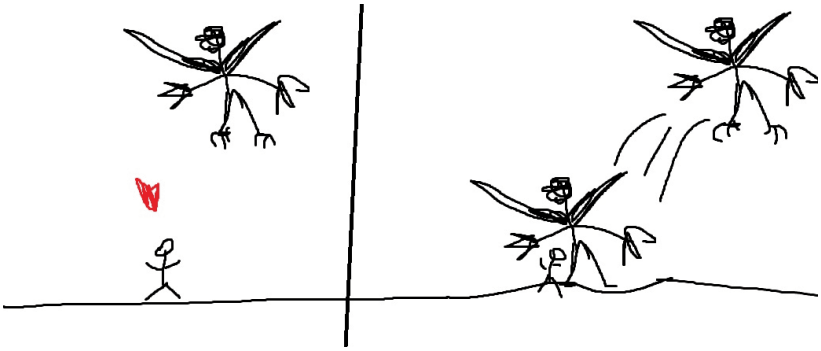

Figure 17: Boss Fight – First Phase

- Second Phase: Nightmare Mini-Game. Azazel rises to the top of the room and starts shaking uncontrollably, filling the room with dark demonic smoke. Adam is on the ground and is paralyzed by Azazel's hold on his body and soul, so he must struggle to escape Azazel's grasp and wake up from this nightmare. As the mini-game progresses, Skulls appear floating in random locations on the screen, and the player must click on them to avoid taking damage. If the player fails to click on too many of the skulls, his health will drop to 0 and will lose the game. However, if the player manages to click on enough skulls, he will escape from Azazel's grasp and wake up from the nightmare. At this point the game continues to level 1 with the difficulty slightly increased.

Figure 18: Boss Fight – Second Phase Mini-Game

The game's levels are basically on a never-ending loop, symbolizing Adam being trapped by the Demon possessing him, and the only way out is to face his demon and get rid of his control. Therefore, once the player reaches the Boss fight, he either defeats him, dies or continue the loops after escaping the mini-game. When the player escapes from the Boss Fight, he starts Level 1 again, maintaining his current side-grade and the difficulty of the game is slightly increased, so the player takes more damage than before. Each time the player fails to defeat the boss and escapes the mini-

game, the difficulty increases further, so the player must try to defeat Azazel as quick as possible, otherwise he will become too powerful to defeat and the player will eventually die.

4.3. Graphic Design & Interfaces

Esbozos, croquis, modelos, etc., creados durante el proceso de trabajo, incluyendo especialmente:

4.3.1. Main Character

Adam is the playable character. He can move left, right, jump, attack and fall down (upon death).
These are the different states with their respective sprites that build each animation:

- **Movement**: Using "**A**" or "**D**" key the player faces & moves left or right, and by using the "**W**" key the player jumps, and it can also be directed left or right with the previously mentioned movement keys.
- **Idle**: Player stands still, while breathing slowly.

Figure 19: Idle Assets

- **Crouch**: Player crouches down. He cannot move or jump while crouched, but he can attack.

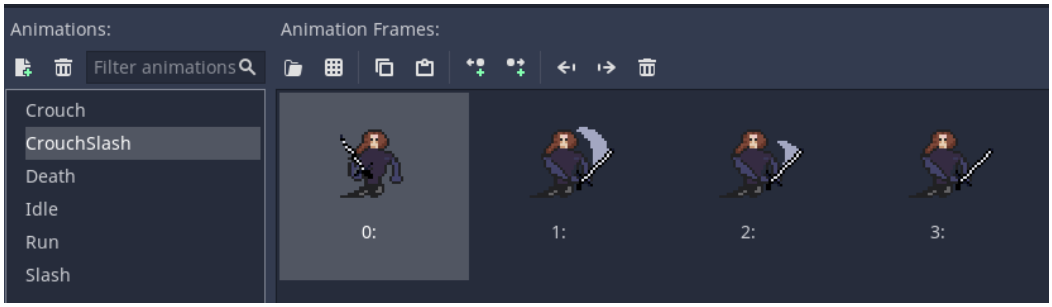

Figure 20: Crouch Assets

- **Slash**: Player slashes with his sword towards the direction he is facing. If it collides with enemies, it will damage them.

Figure 21: Slash Assets

- **Run**: Player runs towards the direction he is facing.

Figure 22: Death Assets

- **Possessed:** Once the player's corruption bar fills up, Adam is surrounded by demonic black smoke and remains possessed for a short amount of time. During this possessed state, the demon Azazel forces Adam to move randomly, seriously endangering him against potential enemies and even causing him to step into traps or jump off cliffs.

Figure 23: Possessed State

4.3.2. Enemies

The game has three types of enemies (two basic, one corrupted), three types of traps (two basic, one corrupted) and a Boss that has two different phases.

Walking Enemy (Basic)

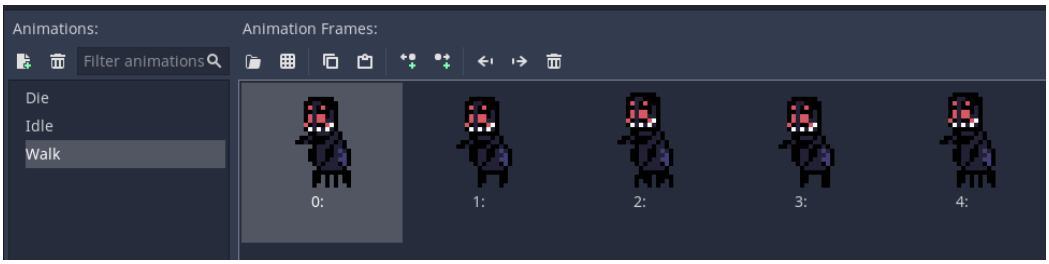

Figure 24: Walker Assets

Walking humanoid night creature that uses claw attacks. When defeated, there's a chance of dropping a pick-up. This enemy spawns in a fixed position and remains idle, until the player gets close. Once it detects the player, it follows him and attacks with a Claw mele attack. The walker moves fast and also can jump very high, avoiding any obstacle in its path to destroy the player.

Figure 25: Walker Death Animation Assets

Upon being defeated, the walker explodes and has a chance drop a pick-up for the player to obtain.

Flying Enemy (Basic)

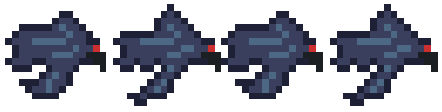

Figure 26: Flying Enemy Assets

A bat-like night creature that has the ability to fly and ranged attacks. This enemy has a pre-defined patrolling route that it always follows. If it collides with the player, it hurts him while not taking any damage itself.

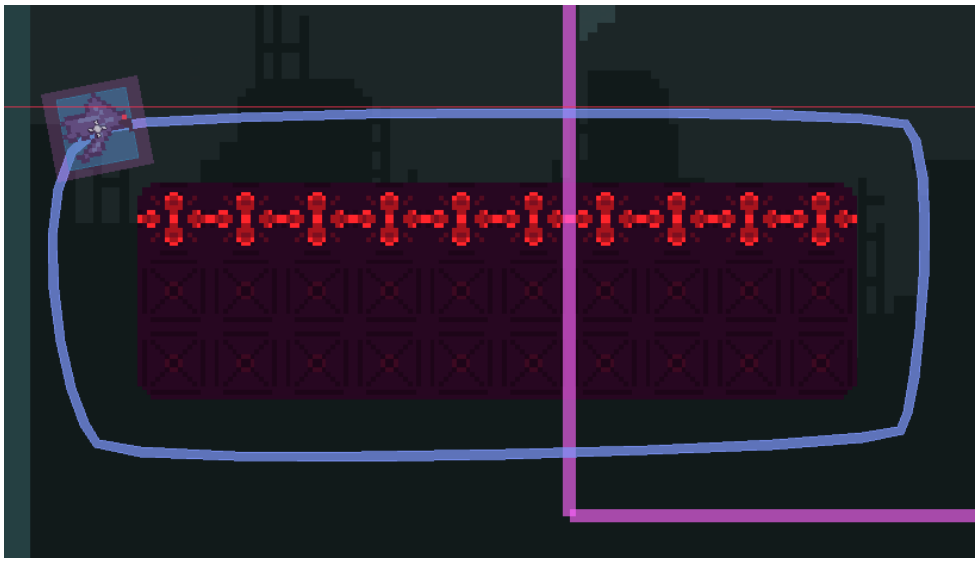

Figure 27: Flying Enemy Route

When the player is in the flying enemy's line-of sight, this enemy can launch a ranged purple fire ball towards his direction, hurting him on impact.

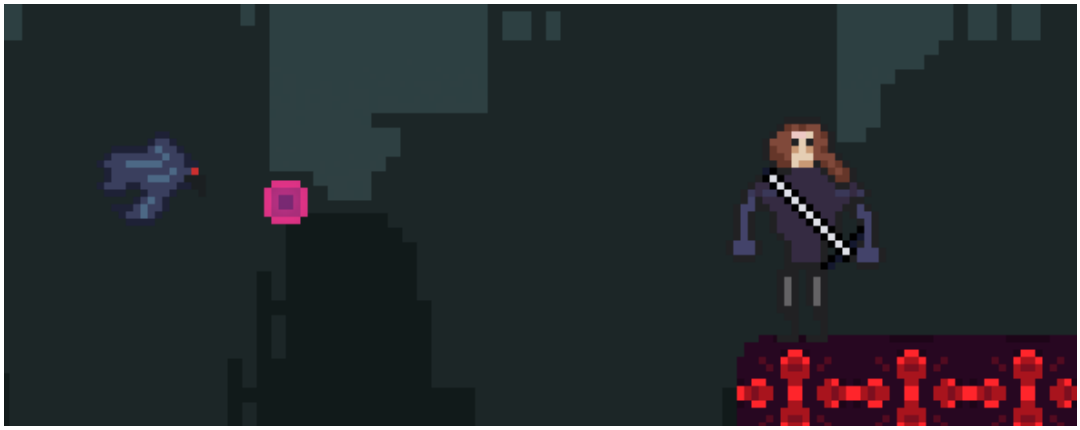

Figure 28: Flying Enemy Ranged Attack

Upon being defeated, the flying enemy has a chance to drop a pick-up for the player to obtain.

Possessed Enemy (Corrupted)

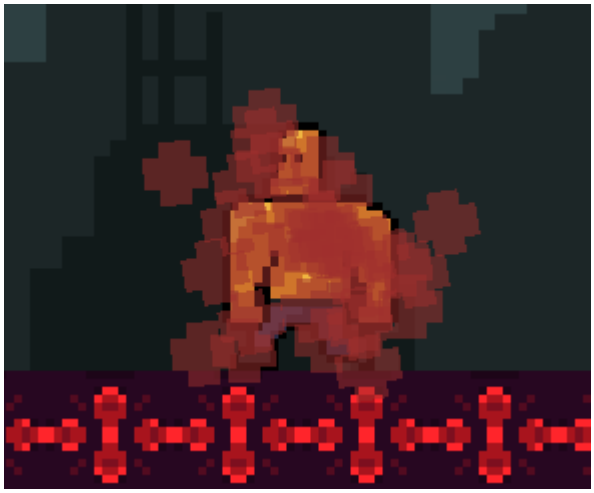

Figure 29: Possessed Enemy

Demonic Entities, with full control over their hosts. They have a larger health and causes damage and corruption to fill faster when close to the player. This enemy is a possessed by a demon and he is surrounded by red demonic smoke. If the player gets too close to him, he'll take damage and the corruption bar will fill-up. Some times this type of enemy shoots a red fire ball in a straight line from his eyes, so the player must jump over or crouch under it to avoid taking damage. Upon being defeated, the corrupted enemy always drops a pick-up for the player to obtain.

Spike Trap (Basic)

Figure 30: Spike Trap

This is a basic trap formed by spikes on the ground. The player gets damaged if he steps over them. The only way this trap can be destroyed is with the Fire Ball ranged attack. Upon destruction, this trap does not drop any pick-ups.

Disappearing Platform Trap (Basic)

Figure 31: Disappearing Platform Trap

This is a platform that flashes white and then disappears, shortly after the player steps on it. Therefore, the player must jump away to avoid falling to his death.

Demonic Skull Trap (Corrupted)

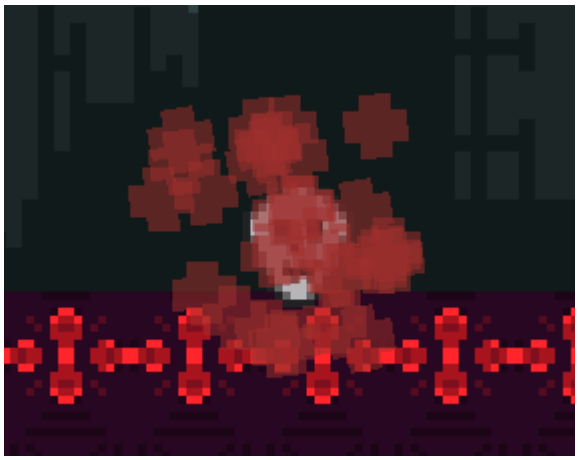

Figure 32: Demonic Skull Trap

Similar to the possessed enemy, this trap is a demonic skull of a dead corrupted enemy, and it maintains it's aura of demonic red smoke that, if in close proximity, damages the player and fills up his corruption bar. The only way this trap can be destroyed is with the Fire Ball ranged attack. Upon destruction, this corrupted trap always drops a pick-up.

Boss: The Demon Azazel

The main antagonist of the game is none other than Azazel, the demon that possessed Adam. Once Adam reaches the safety of the Gothic Church, he falls asleep and starts a whole new kind of battle. He must face his Demon in his nightmares and try to either defeat him once and for all or, at least, survive the night in order to fight another day.

The boss fight consists of two phases:

- First Phase: Physical Fight
- Second Phase: Nightmare Mini-Game

First Phase

Figure 33: Boss – Intangible Form

The boss patrols the upper part of the screen from side to side while staying in an “intangible form”, appearing “greyed-out”, during which Adam can’t deal damage to him.

Figure 34: Boss – Diving Talon Attack

Every now and then, Azazel will turn back to physical form in order to attack Adam. In this form, Adam can also deal damage to Azazel.

Azazel employs two different types of attacks. The first one being a diving claw attack, where he dives towards Adam's position and tries to strike him with his deadly talons. After performing this attack, he will remain with his talons stuck into the ground for a short period of time, during which Adam can take advantage to deal damage.

Figure 35: Boss – Ranged Attack

The second type of attack is the ranged Demonic Fireball attack, where Azazel turns back to physical form and rains a burst of red fire balls aimed at Adam's current position. The player must keep moving to avoid getting damaged. During this attack the player can deal damage by jumping and slashing Azazel, or by using his own ranged fire ball attack, if equipped.

This first phase has a timer, that slowly drains. If the player defeats the boss before it runs out, he wins the game and Adam rids himself of his demon. Alternatively, the player can also die during the boss fight, which will end the game in a loss. If neither of the two alternatives occur, once the time runs out, the boss fight moves to its second phase.

Second Phase:

Figure 36: Boss – Second Phase Mini-Game

Azazel rises to the top of the room and starts shaking uncontrollably, while Adam is on the ground and is paralyzed by Azazel's hold on his body and soul. As the mini-game progresses, Skulls appear floating in random locations on the screen, and the player must click on them to avoid taking damage. If the player fails to click on too many of the skulls, his health will drop to zero and will lose the game. However, if the player manages to click on enough skulls, he will escape from Azazel's grasp and wake up from the nightmare. At this point the game continues to level one with the difficulty slightly increased.

4.3.3. Items

Pick-ups

In this game defeated basic enemies have a chance to drop a pick-up. Corrupted enemies and destroyed corrupted traps always drop a pick-up.

List of Pick-ups

Item	Image	Description
Heart		Restores the Health bar to its full capacity.
Demonic Seal Script		Cleanses the Corruption Bar, leaving it empty.

Power-ups

After clearing a level, the player is given the option to choose one of various upgrades. These power-ups are more a kind of “side-grades” rather than upgrades since they consist of a trade-off mechanic, where the player gets a certain advantage at a cost.

List of Side-grades:

Item	Image	Content	Description
Holy Water		-	Adam drinks the holy water, and it tames his demon, but it burns because of the being possessed. **Effect**: Adam’s corruption bar fills 50% slower, but Adam takes increased damage.
Deal With The Devil (Fire Ball)		-	Adam makes a pact with Azazel to cooperate in order to survive this dark World’s Night Creatures, granting him demonic powers. **Effect**: Adam gains the ranged ability of throwing Demonic Fire Balls. However, the corruption bar fills a little bit with every use. This is the only ability able to destroy traps.
Treasure Chest		-	An antique chest with unknown contents and strange markings on its lock. **Effect**: Randomized chance of getting one of four different contents that give the player an advantage or disadvantage for the next level.

-		Prayer Scroll	An old scroll containing a prayer meant to soothe those at odds with their demons. **Effect**: The player starts the next level with a temporary shield that can absorb up to two hits.
-		Jump Scare	A night creature jumps out of the dark chest, startling Adam. Azazel finds it amusing and mocks his host. **Effect**: The player starts the level with the corruption bar filled at 25% capacity.
-		Trap Chest	A trap chest equipped with spikes, similar to those of a beartrap, that shuts itself almost instantly, crushing Adam's arm in the process. **Effect**: The player starts the next level with 25% less health.
-		Holy Shield	A holy artifact that glows whenever the bearer kneels and raises it to the sky. **Effect**: When crouching, the player becomes immune to any kind of damage to his health. However, it does not affect the corruption bar.

4.3.4. Styles

The main style of the game is a dark gothic pixel art aesthetic. Dark Red, black and shades of grey are a predominant part of the color pallet of the game.

Figure 37: Color Palette

The game uses a simple white font called “Merchant Copy”, that has a slightly pixelated aesthetic

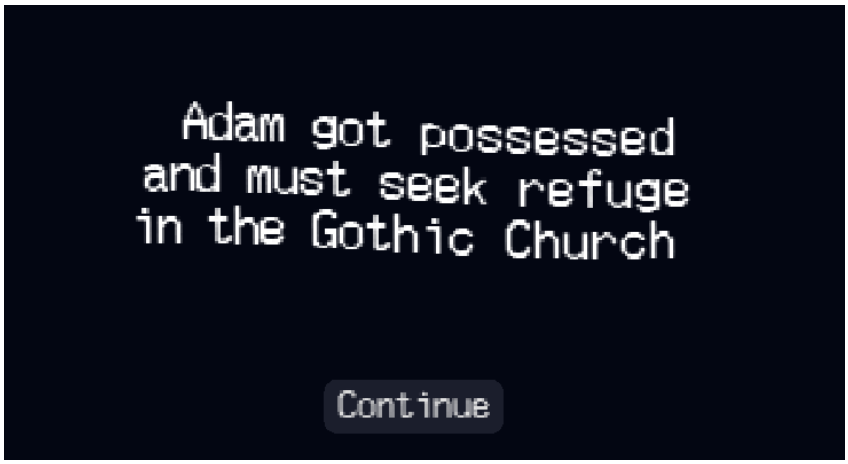

Figure 38: Text Font

The game’s main menu consists of the red-coloured title, in the same font mentioned above, and three buttons that allows the player to “Play”, “Quit” or read the game’s user instructions by clicking the “Help” button.

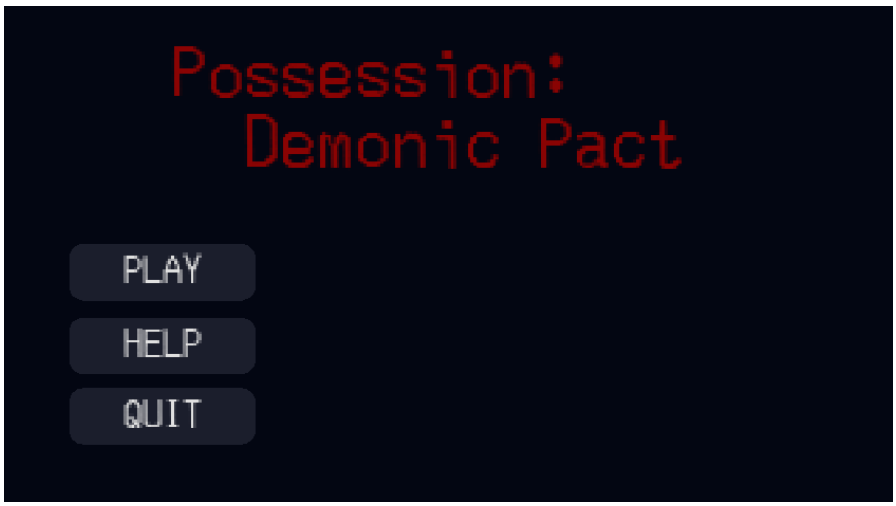

Figure 39: Main Menu

4.3.5. Usability/UX

The user interface of the game is quite simple. It consists of a main menu with buttons that allow the player to play the game, exit the game and view the User instructions and game's controls.

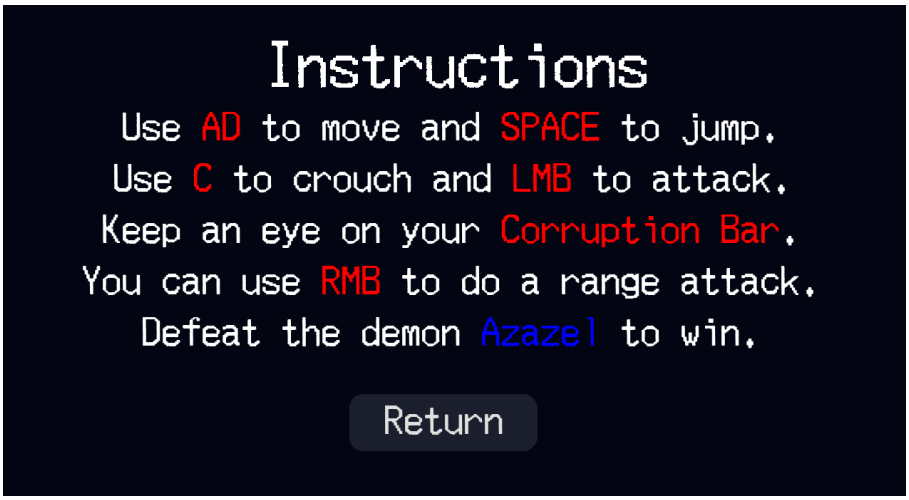

Figure 40: User Instructions & Controls

This is the navigation map of the game's menu and levels:

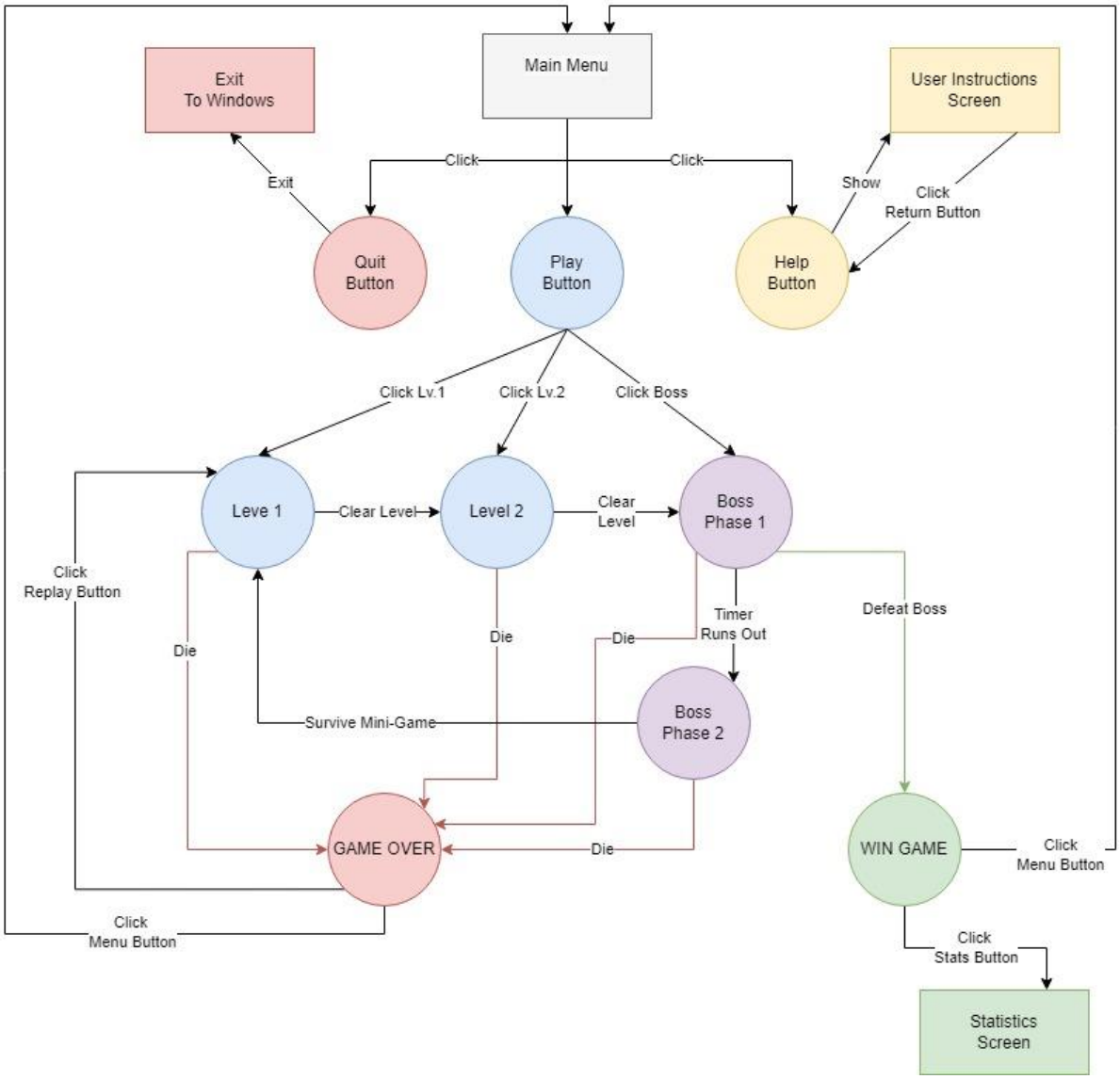

Figure 41: Navigation Map

As shown in the map above, the user can choose to exit the game via the “Quit” button, view the instructions via the “Help” button, where he can return to the main menu by clicking the “Return” button. Finally, the user can start playing the game by pressing the “Play” button in the main menu.

Once started, the user can choose which level he wishes to start from, and continue the game-loop from there. Before starting the Level 2 or Boss Level, the player must choose an **Upgrade** from the list detailed before in the design chapter. While playing any of the levels, including both Boss phases, if the player dies, he gets the option to either return to the main menu or restart the game-loop from Level 1.

The game's HUD elements consists of a Health Bar, Corruption Bar and the icon of the active Upgrade. A timer HUD element is also present when the player is possessed (short duration) and when the player faces the Boss' first phase (longer duration).

Figure 42: HUD - Health & Corruption Bars, Fire Ball Icon (Active Upgrade)

When facing the Boss, during the first phase the player can either die (Game Over) or defeat the Boss (by emptying his health bar). In case of winning, the game shows and end screen with the conclusion of the story of Adam and Azazel's battle.

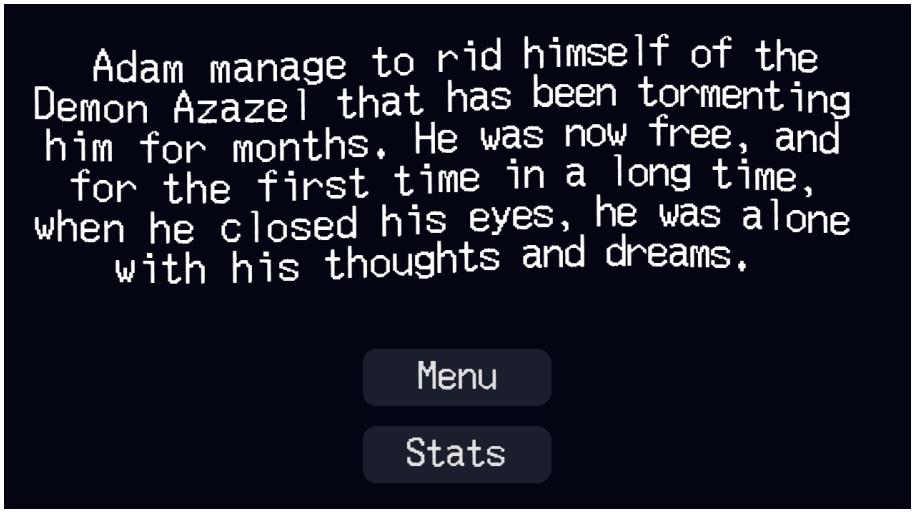

Figure 43: Win Game - End Screen

Here the player can go back to the main menu, or check his statistics for the current game session's results. A pop-up screen will appear showing how many enemies the player has killed, the total duration of the game-play and the number of Boss Fights that were necessary before defeating Azazel.

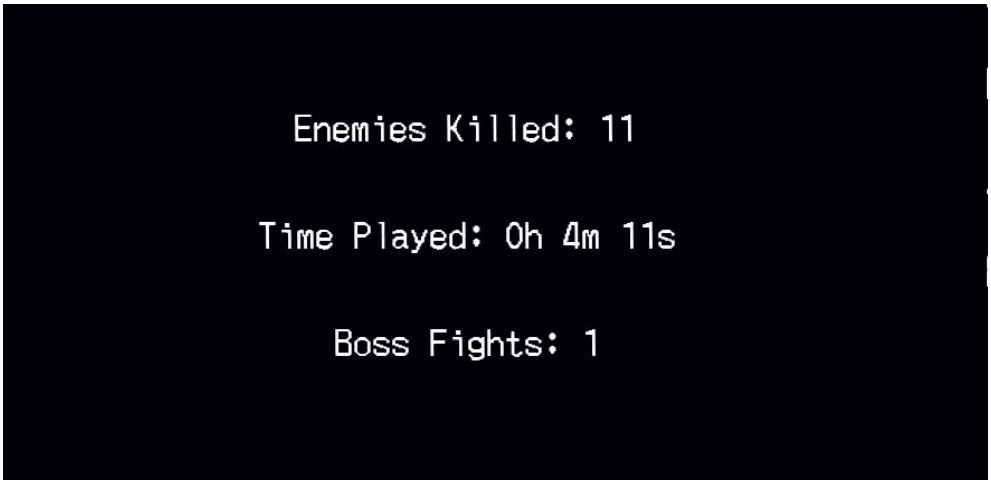

Figure 44: Win Game - Statistics

However, if the player doesn't defeat the boss during the first phase, and survives the fight, this starts the second phase of the boss fight. This phase is a mini-game where the player must click the skulls that appear and disappear in random locations on the screen. The player can only die or survive this mini-game. If the player completes the minigame and escapes the "Nightmare" Boss Fight, the game starts again from Level 1, while the player keeps his current score and the difficulty is slightly increased (the player will take more and more damage with each restart of game-loop after failing to kill the boss and completing the mini-game).

4.4. Other Assets

Most of the videogame's graphical assets as well as non-graphical assets were created by me using software such as GIMP image editor, Microsoft Paint, JSFXR 8-bit sound maker and Godot Engine's various tools, such as Shaders, Hitbox and GDScript.

As for external assets used, these are the ones that have been used, all under a Copyright Free Licenses, with no attribution required for personal and commercial purposes.

Graphical Assets from https://opengameart.org/:

- Background, Foreground & Sky images
- Fire Ball & Explosion sprites

Pixelated Font "Merchant Copy" from https://www.dafont.com/merchant-copy.font.

Audio Assets from https://opengameart.org/:

- Fire Ball sound

- Main music theme
- Evil Laugh Sound
- Player/Enemy “hurt” sound
- Player “death” sound

4.5. Programming Languages & APIs Used

4.5.1. Environment & Programming Language

The engine used for this project is Godot Game Engine, due to its simplicity and ease of use for 2D pixel art games. It’s opensource software and uses scripting languages such as its own GDScript language, while also offering support for other languages such as C#, VisualScript and via its GDNative technology, C & C++. It works on my personal Windows PC and it’s a very light software, making it ideal for this project.

Godot Engine minimum system requirements:

- CPU: dual core or better
- RAM: 4GB or more
- OpenGL: 2.1
- DirectX: 10.1
- Windows: 8 or newer

The scripts written in GDScript attach to a node and extend its behaviour, inheriting all functions and properties of the node they attach to. GDScript is an object-oriented and imperative programming language built specifically for Godot.

The way this works, is that each element in a scene is a “Node” type object, and it has various states and animations associated to it, such as idle, running, attacking, death etc.

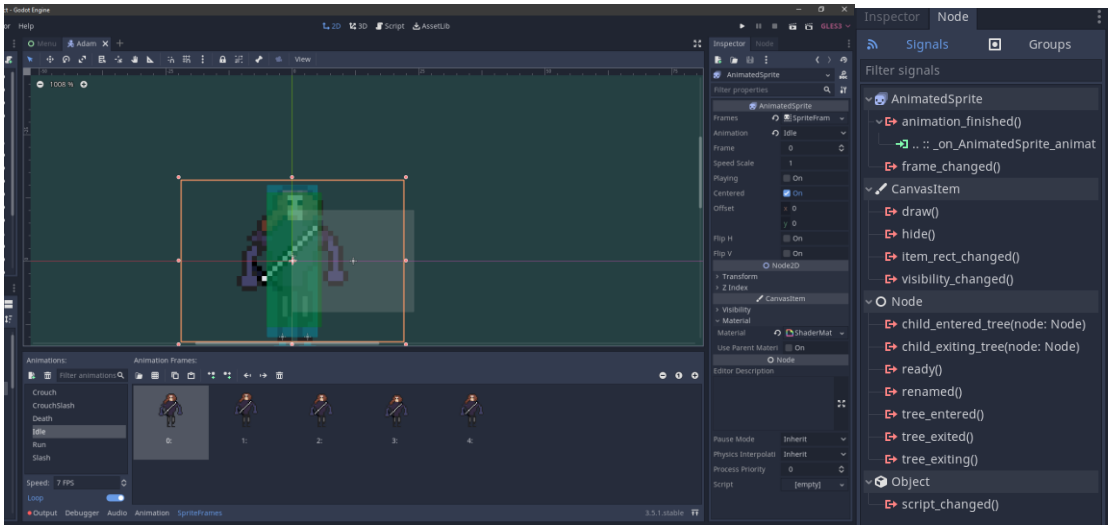

Figure 45: Godot Node Object

To this node we can attach a GDScript file, which contains the code necessary for all this object's functionality and interactions. In this example, we see the main player of the game, Adam, is created as a Node object, and it has various types of movements, animations and interactions with the levels and other objects such as Enemies or the Boss.

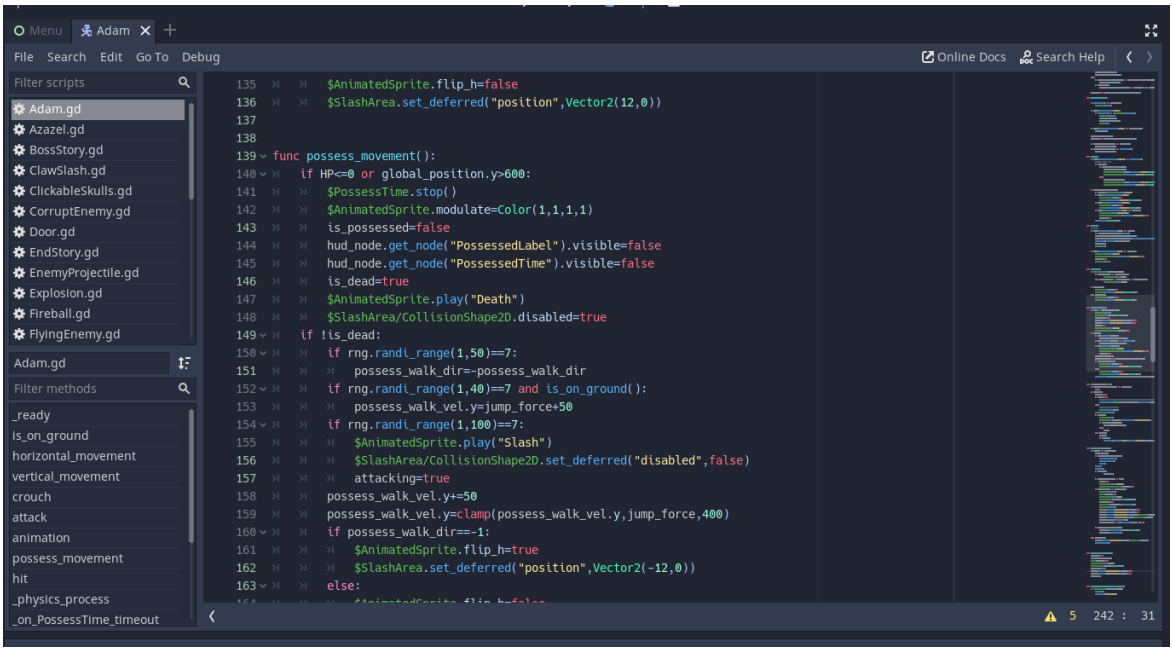

Figure 46: Godot GDScript

Therefore, the Node for Adam, defined in the file Adam.tscn, has a GDScript file attached to it. This file is called "Adam.gd" where all the animations and actions of the main character are defined and coded. Among these functions, we can see the animation where we tell the node witch state of the character to show depending on if the player is standing idle, running, attacking etc.

In the case of the character's movement, I have implemented acceleration so the longer the movement button is being held pressed down the more acceleration the character accumulates, up to a maximum range, to avoid unrealistic or hard to control speeds.

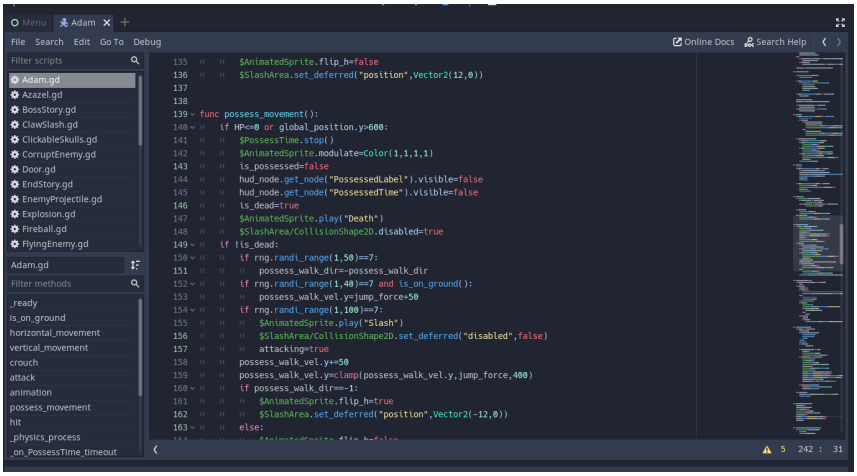

Figure 47: Movement Code Snippet

We can also see the implementation of input buffering and coyote time mechanics, used for the vertical movement (player jumping). Coyote time was comically named after the character Wile E. Coyote (from Warner Bros. Looney Tunes) who routinely had issues with the laws of physics while chasing the Road Runner. This feature refers to an "invisible" (uncountable) delay between an action (in this case the player jumping vertically) and the consequences of that action (taking off or landing) that has no physical cause, in order to provide the player with a very brief extra window to react to the action and make the game's mechanics a little more forgiving.

The implementation of these features, albeit simple, gives the player the feeling of experiencing realistic physics when it comes to the game's characters and objects. Overall, the main aim is to provide an enjoyable experience to the player.

4.5.2. Tools Used

- Godot (v.3.5.1): Game engine used for the creation of the video game.
- Gimp & Paint: Used for image creation and editing
- GitHub & SourceTree: Used for hosting the source code, release & versions control.
- Draw.io: Used to generate diagrams
- Google Docs & Notepad++: Used to generate the documentation of the project.
- JSFXR & Audacity: Used for audio creation and editing.
- Shortcut: Used for video editing.

- Personal Computer: Equipped with the minimum requirements necessary in order to use all previously mentioned tools.

5. Implementation

5.1. Installation Requirements

Game Minimum Requirements:

- **Operating System**: Windows 8 or newer
- **CPU**: Dual Core or better
- **OpenGL**: 2.1
- **DirectX**: 10.1
- **Disk Space**: 60 MB available space

5.2. Installation Instructions

In order to run the game, the user needs to extract the files form the compressed zip file “Playable_Game.zip”.

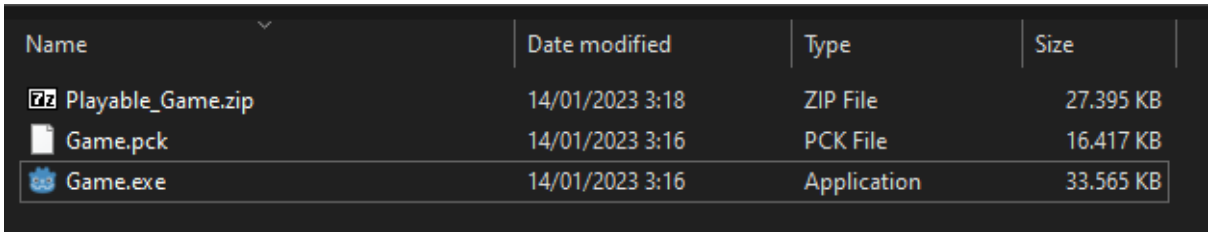

Figure 48: Installation

Once extracted, the user must run the executable file “Game.exe” and the game will start up, showing the main menu.

En el caso de ser necesaria una instalación, especificar los pasos detallados sobre cómo se tiene que instalar/implantar la aplicación.

6. Demonstration

6.1. Use Instructions

The game implements a “Help” menu accessible from the main menu, indicating the game’s control scheme.

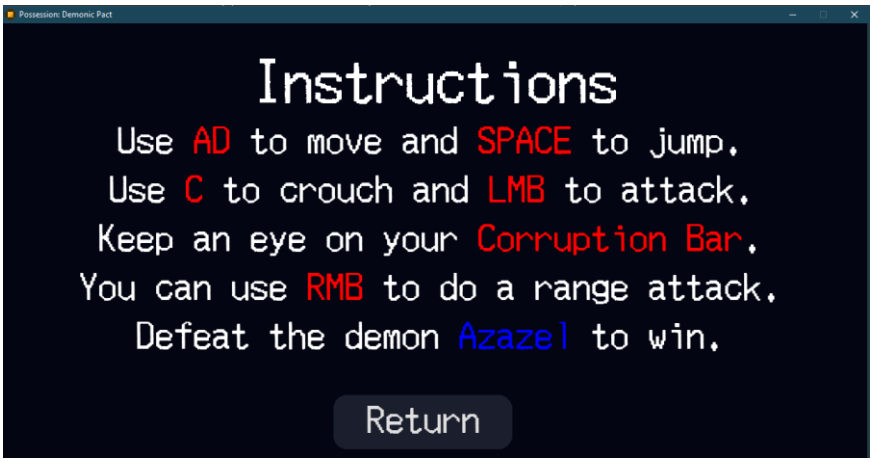

Figure 49: Installation

Movement:

- “**AD**” to move left/right
- “**W**” to jump. Use in combination with “AD” to jump left/right
- “**C**” to crouch

Attack:

- Left-Mouse Click (**LMB**) for sword attack. Also used to click on interactable items during the Boss 2nd Phase Mini-Game.
- Right-Mouse Click (**RMB**) for ranged attack (when available).

7. Conclusions & Future Lines

7.1. Conclusions

Reaching the end of this project, the main conclusion to take away from this experience is that developing a videogame from scratch alone is very difficult. A videogame, even the simplest ones, require an immense effort especially when one does all the tasks related to its development, planning, finding the resources, marketing and all minor aspects and challenges. However. With time, perseverance and an effective objective layout and planning, it can be done. And it can be a fulfilling experience, full of trials & errors and a lot of learning and improving, both academically/professionally and personally.

Another valuable conclusion extracted from this project is the importance of time-management and prioritization of the main objectives. It is very easy for one to fall into trying to perfect a certain aspect of the game or try to implement as many features as possible and falling behind schedule and even endangering the project's release date because of these issues. Therefore, sticking to the pre-defined time margins, and not hesitating to sacrifice certain optional objectives, so as to not endanger the viability of delivering a fully working and polish product by the deadline, were paramount in managing to finish the project in time and deliver a quality product.

7.2. Reflection On Proposed Objectives

As far as the objectives of the project are concerned, I've managed to implement the majority of the main objectives that I proposed at the start of the project. Furthermore, I manage to squeeze in some secondary objectives in this first full iteration of the game, such as the implementation of pick-ups and character upgrades between levels as well as including a little bit of this fictional world's lore and character background in the form of text prompts upon completing levels and beating the game.

On the other hand, there are quite a few secondary objectives or interesting aspects of the game that I would have loved to explore, such as implementing challenges, collectibles and unlockable skins or weapons or the inclusion of images generated with A.I. (Artificial Intelligence) to offer the game a unique look and offer a different alternative to asset generation that does not depend on personal knowledge of image editing or rely on license-free assets available online. However due to time constraints and other challenges encountered during the development process, were discarded and set up as future objectives.

Overall, I am happy with the end result and I am excited for the potential future iterations of this videogame. I also learned many interesting facts about the vast field that is videogame development

and acquired numerous valuable skills that will benefit me in my professional career as a programmer.

7.3. Future Lines

The videogame Possession: Demonic Pact implements most of my main concepts and ideas, however, there is always room for improvement and there are a few secondary objectives that I would love to explore in the future. I also have many exciting new concepts and ideas for a future version of the game, and maybe even a rough idea for a prequel, exploring Adam's Monster Hunter past and build on this dark world the game takes place in.

Some of the most interesting features and ideas for the future of this game are:

- Implementation of A.I. Generation Tools (such as Midjourney, Dall.E 2, etc.) to generate the graphical assets for the videogame:
    - This used to be an objective for the current iteration of the game, however upon investigating the nature of how the A.I. generated the graphics and the lackluster and confusing information and details about the licensing and ownership of these images, I eventually decided to discard this as a possibility early on during the planning stage of the project. That being said, I would like to continue to research this interesting new alternative to asset generation that does not depend on personal knowledge of image editing or rely on license-free assets available online
- Improving the User Interface, HUD and implement a Pause Menu to allow the player to pause the game without fear of failing or losing progress.
- Improve the feedback the player receives when receiving damage.
- Improve Enemy A.I.
- Improve the Statistics End Screen and add more stats.
- Implement a Skill Tree to improve stats, abilities and add new special abilities and powers.
- Use softer camera that's not snapped to the player. Whenever a player gets near the bounds of a level it should scroll to show a bigger image of the level until the player moves away from the limits of the level.
- Add Screen-shake mechanic when taking damage, attacking, etc.
- Expand both phases of the Boss Fight to make it a more complete and entertaining experience. I want to delve deeper into the idea that the Boss fight is a nightmare that the player must escape.
- Adding new levels, enemies, pick-ups and abilities
- Implementing unlockable skins and weapons, that would provide the player with game modifiers (such as improved health, corruption immunity, etc.).
- New Game Modes:

    - Challenges: Challenge the player to do various objectives or challenges during the normal playthrough, such as timed challenges, defeat a certain number of enemies etc. By completing these challenges, the player would unlock achievements and even unlockable gear, skins and other items.
    - Survival Mode: Infinite spawning enemies coming in waves.
    - Various Difficulty Modes
    - Multiplayer Modes: Versus mode Adam vs Azazel, Asymmetric multiplayer where a player is Adam and various players play as Enemies, Co-Op Story Mode, etc.
- Adapt the game to mobile platforms, such as Android and/or iOS.
- Implement procedural generation of the game's levels, traps and enemy spawning points.
- Add hidden collectibles. They would be items hard to find hidden somewhere in the levels, that when found would unlock bit-sized snippets of story, providing a more in depth look on the history of the world and characters of the game.
- Add achievements that the player can unlock by completing certain objectives in the game and allow the player to view them in a dedicated showcase menu
- Develop an expansion or sequel exploring Adam's Monster Hunter past and build on this dark world the game takes place in. This world could house a wide variety of stories, both featuring the original protagonist Adam, as well as other characters and creatures that roam the world:
    - One interesting idea would be to develop a game where the protagonist is a were-wolf, further exploring the uncontrollable "possession" idea introduced in the original game. The player could fight to avoid turning into the werewolf, or use it to their advantage. The werewolf form would come with advantages and disadvantages, similar to the possession in the original game.
    - Another idea is to further explore and expand the "Nightmare" levels where the human fights against its demons for control over their body.
    - I also would love to make the story of the world and its history more present in future game, since I believe a crucial part of the players enjoyment is the inmersion in the world we create for them.

Bibliography

UOC Resources
Various articles, books and other resources linked in the virtual classroom

"**The Art Of Screen-shake**" by Jan Willem Nijman
https://www.youtube.com/watch?v=AJdEqssNZ-U

Godot Docs – Documentation & Tutorials
https://docs.godotengine.org/en/stable/

Godot Reddit – Community Support
https://www.reddit.com/r/godot/

YouTube Tutorials:
https://www.youtube.com/@GodotTutorials/videos
https://www.youtube.com/@SquidGodDev/videos
https://www.youtube.com/@Mizizizi z/videos
https://www.youtube.com/@uheartbeast/videos
https://www.youtube.com/@duongtuan4379/videos

Stack Overflow – Technical Questions & Answers
https://stackoverflow.com/

Gitflow – Manual
https://gfourmis.co/gitflow-sin-morir-en-el-intento/

Github Docs – Documentation
https://docs.github.com/en

Annexes

Annex A: Glossary

2D

Two-dimensional object, only featuring a visible height and width.

Co-Op

A cooperative mode of a videogame where two or more players can play together in a cooperative effort to finish the game.

Hack-and-slash

Hack-and-slash or hack-and-slay is a term used to define a videogame genre that emphasizez hand-to-hand close combat.

HUD

HUD or Head-up Display, is used to describe the information that is shown on the screen at all times during gameplay, usually showing text, numbers and icons related to the gameplay's mechanics.

Pixel

A minute area of illumination on a display screen, one of many from which a full image is composed.

Pixel Art

Pixel art is a form of digital art created sing software, where the images are thought at the pixel level. This aesthetic of this type of graphics comes from 8-bit and 16-bit computer based and retro video game consoles.

Side-scroller

A side-scroller is a type of videogame that uses a side-view camera angle. These games are usually 2D with game characters that move from the left to the right side of a screen.

Sprite

Bitmap image usually used for graphical representations in 2D videogames.

Annex B: Version Registry

Version 0.1 - Initial version, main character and movement (no animation yet), basic platforms and basic fire-ball animation.

Version 0.2 - Added walking enemy that follows the player, movement animation and mele attack.

Version 0.3 - Added flying enemy and traps and corruption meter filling.

Version 0.4 - First level alpha design, only the two previous created enemies spawn at the start, need to add more in different parts of the level.

Version 0.5 - Implemented Help Menu & Power-up screen. Finished first level, with pick-ups, enemies, traps and patrolling flying enemies. Corruption only works near skulls corruption traps in this version.

Version 0.6 - Implemented full second level & previous problem fixes. Added Possessed Enemies.

Version 0.7 - Implemented Boss Phase 1 & previous problem fixes: Removed Possessed Enemies glow and changed it for the same red smoke of the Skull Traps. Corruption enemies/traps always drop pick-ups. Increased player health. Implemented Choose Start Level (the player can start the game from Level 1, 2 or Boss and continue the game-loop from there as usual).

Version 0.8 - Full **Alpha** Version: Level 1, 2 and Boss Phase 1 complete & Various Minor Changes.

To-Do List For Next Version:

p1. Finish Boss Phase 2 implementation.

p2. Fix Gameplay loop when escaping the boss (instead of restarting the game).

p3. Add story text screen previous to the Boss fight.

p4. Implement Score screen (if the player WINS).

Version 1.0 - Full Playable Version (**Beta**): Reduced Boss Timer, fixed gameplay loop. Added Story Text Screen & Score Screen. Finished Boss Phase 2. Various Minor Changes.

To-Do List For Next Version:

p1. Add smoke on the ground for Boss Phase 2 (symbolizing Adam being "Trapped")

p2. Reduce Boss Phase 1 Timer AGAIN!

p3. Increase difficulty once the game-loop re-starts at level 1 after escaping Phase 2.

p4. Add mele attack sound to Boss.

p5. Add number of times you fought the boss to End Screen Stats.

p6. Fix restart game when dying. It currently keeps power-ups from previous fight.

Version 1.1 (Final) - Minor Fixes & Changes: (Final Deliverable Version) - Complete gameplay loop, all enemies, upgrades, pick-ups, attacks and main modifiers described in the main objectives are in working order.

- Some secondary objectives have been discarded due to time constraints.
- This is the final version of the game for the TFG Delivery.
- Further implementation of secondary objectives (or new ones) could be added in a future version (2.0+) outside the scope of the TFG's current project planning and delivery.

Annex C: Project Deliverables

Final Degree Project Worksheet (Memoria TFG):

Game Executable (ZIP):

Game Trailer Video:

Project Defence Video:

Self-assessment Report:

Cession Of Rights Document: