|  |
| --- |
|  |
| Vitarca |
| Lawrence Bennett 40216550  Joseph Mitchell 40442074 |
|  |

# Table of Contents

[**Overview and Vision Statement**](#_heading=h.c0gywjyd6ksr) **4**

[**Inspirations**](#_heading=h.w266m8ag5ibm) **4**

[**Player Experience Goals**](#_heading=h.l8q8ojh7dlv8) **5**

[**Audience and Platform**](#_heading=h.89ed94xtq5m6) **5**

[Legal Analysis](#_heading=h.bnnniyam8ucx) 6

[**Gameplay**](#_heading=h.77rywanxad8v) **6**

[Overview](#_heading=h.cyoo036cqd8n) 6

[Player Mode](#_heading=h.tkcicjq0iy3c) 6

[Screen Mockup(s)](#_heading=h.54pkb0pvjn95) 6

[Formal Elements](#_heading=h.drg9tnbm7r2u) 7

[Players](#_heading=h.s6dxvoyweztw) 7

[Objectives](#_heading=h.agiywtc3su63) 7

[In each level the player's goal is to reach the exit without being detected by any enemies, whilst solving puzzles that make the terrain navigable. Outwitting the enemy is key here.](#_heading=h.tbi7ihnetm5t) 7

[Procedures](#_heading=h.rncj5myvj1s9) 7

[Rules](#_heading=h.1hjrzkjjcaw0) 7

[Resources](#_heading=h.6kg7anyektfu) 7

[Conflict](#_heading=h.qa9nab43ijgp) 7

[Boundaries](#_heading=h.h3nxy570z5l7) 7

[Outcome](#_heading=h.yovfn9igbriq) 7

[Controls](#_heading=h.88q1vmb3xk0e) 7

[Levels](#_heading=h.4vnkemj5wvx5) 8

[Flowchart](#_heading=h.wrhh4z7yegqn) 8

[Editor](#_heading=h.lpfxfzk2pys4) 9

[Characters](#_heading=h.p257um50rujl) 9

[Story](#_heading=h.aful5pk1mc2m) 9

[Game World](#_heading=h.tfnfih9gyag1) 9

[**Required Media List**](#_heading=h.vjl6b9q6ciux) **9**

[User Interface Assets](#_heading=h.jcg0qas6ky8) 9

[Environment Assets](#_heading=h.t2upjtlw5c9v) 9

[Character-based Assets](#_heading=h.rwqlz7eumel8) 9

[Animation Assets](#_heading=h.edqmpvbuw7y2) 9

[Music Assets](#_heading=h.fmrzesglpicx) 9

[Sound Effect Assets](#_heading=h.2ywnkkazji2x) 10

[**Technical Specification**](#_heading=h.btsi89g6dzpm) **10**

[Development Platform and Tools](#_heading=h.dyyqkjn6uxg3) 10

[Delivery Mechanism](#_heading=h.gq5i4drbjrsv) 10

[Game Engine](#_heading=h.1born7ocvxo) 10

[User Interface Technical Requirements](#_heading=h.pwgp9t2t3ykv) 10

[Controls Technical Specification](#_heading=h.ot2996q2edmg) 10

[Network Requirements](#_heading=h.6hypuplpchv3) 10

[System Parameters](#_heading=h.bvbo2pjrk17k) 10

[Flowchart](#_heading=h.ixywidtw8dg) 11

# Document History

|  |  |
| --- | --- |
| Document Created | 11/03/21 |
| Document Update 01 | 05/05/21 |
| Document Update 02 | 06/05/21 |
| Document Update 03 | 10/05/21 |

# Overview and Vision Statement

The purpose of the game to be created is to showcase a system for very basic 2D procedural animation. This will give life to the player character without having to create complex and large amounts of artwork.

The mechanics of this game revolve around stealth and platforming elements similar to Rainworld and Stealth Bastards. The player controls a tentacle monster that has inhabited the torso of one of the evil robots. The player must make their way through an environment littered with climbable areas and solve minor puzzles to progress. Although the player must keep a keen eye out for the enemy surveillance systems and sentries. If you are caught by these systems for longer than a few split seconds, it is game over.

# Inspirations

* Rain World



Rain world has many examples of procedural animation, but the rain-deer above were the main influence for the player characters limb movement.

* Stealth Bastard



Stealth bastard already had some overlapping features with the original idea of our game, such as possible stealth mechanics, the enemy characters being robots and having each level be a separate room for simplicity. This made it a good game to find inspiration from.

# Player Experience Goals

The player will take control of a tentacle monster to progress through a complex environment that no normal player character can. The tentacles that the player has are sticky, and can be used to grapple onto surfaces. These limbs can also be used to activate levers and move objects to get to new areas.

The player must also avoid the enemy surveillance systems and sentries at all costs. Any unwanted interaction with these enemies will result in death. The player can avoid these areas by staying out of view range. This is indicated via a coloured cone for the CCTV.

The goal of the demo is to reach the end point of the dungeon, at this point the tentacle monster can rest easy in a safe zone while it gathers its strength and decides on what to do next.

# Audience and Platform

The core demographic is ages between 18 and 34. This audience equates to 38% of video game enthusiasts. The target platform is Steam which has 120 million monthly users. There is a large pool in which this game could potentially reach thanks to the target platform of Steam and PC. Games such as Rainworld and Stealth Bastards have their popularity centered around Steam. This is another reason why the PC and Steam have been chosen. Due to the unique story, setting and gameplay, the game has potential to be picked up by influencers. This would help other people find out about the game after release. We would have to compete with other indie-style games releasing at similar times, as well as games with similar gameplay such as the aforementioned Stealth Bastards, however very few games share the theme of a fully medieval fantastical world alongside elements of modern technology.

Hardware requirements: a windows 10 PC with at least 4gb Ram.

## Legal Analysis

The game is an original IP, so it is not a clone of any other game, thus will not suffer any copyright considerations. A potential issue is the tile set for the dungeon environment. This tile set may be too much to complete in the small amount of time that is allocated for development. If that is the case, then a website called https://opengameart.org/ will be used to procure free to use assets for the environment.

At this stage in development and due to lack of professional experience of the developers, it cannot be said for certain which library terms and conditions will need to be met. This will be updated alongside the development of the project.

# Gameplay

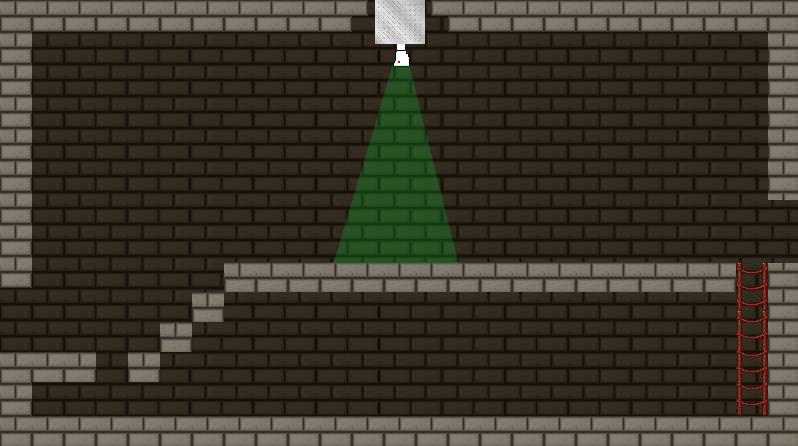
## Overview

Maneuver a level using special abilities unique to the player. These abilities will allow you to climb on certain surfaces. The player can also use their limbs to depress levers and move objects in order to solve puzzles. avoid the enemy CCTV and Robots. If you are caught, it starts a small timer until game over, you must escape from the enemy vision to keep playing.

### Player Mode

The demo will be single player for now.

## Screen Mockup(s)



## Formal Elements

### Players

The player interacts with the game via Single-player versus the game. The role of the player is to stealthily navigate an environment without getting caught, so is somewhat of an explorer.

### Objectives

### In each level the player's goal is to reach the exit without being detected by any enemies, whilst solving puzzles that make the terrain navigable. Outwitting the enemy is key here.

### Procedures

Player moves left and right and can jump. The player's arms can be extended and then will point in the direction of the mouse. The arms can be used to manipulate physics objects. The objects can be latched onto, to either push or pull. These objects will have limited space to move. The player can then jump on these boxes to access new areas.

### Rules

If the player manages to get to the end point of the level, then the win condition of the game will be met. If the player is spotted (in detection range) for longer than 1 second, the player is defeated. If the player grabs onto an object with LMB, the player now can push or pull the object.

### Resources

In the game the player will have to make use of objects to reach previously out of bounds areas in order to progress. the player will also have to press buttons and levers. to open up doors in certain sequences in order to progress.

### Conflict

Conflict in the game will take place as a detection zone around enemy sentries and CCTV. If the player is caught they will have a small time frame in which to escape this detection zone and continue the demo without being immediately defeated and have to restart. But the player must get past this detection zone in order to progress.

### Boundaries

In the tile set, there will be a static wall tile that will act as an impassable object, whenever the player tries to move past the bounds of this tile, it will stop the player from moving in that direction.

### Outcome

By reaching the end point of the game, the player is awarded with a text message that details their small victory over the machines and can now rest easy in the safe zone until the development of the full game is complete.

## Controls

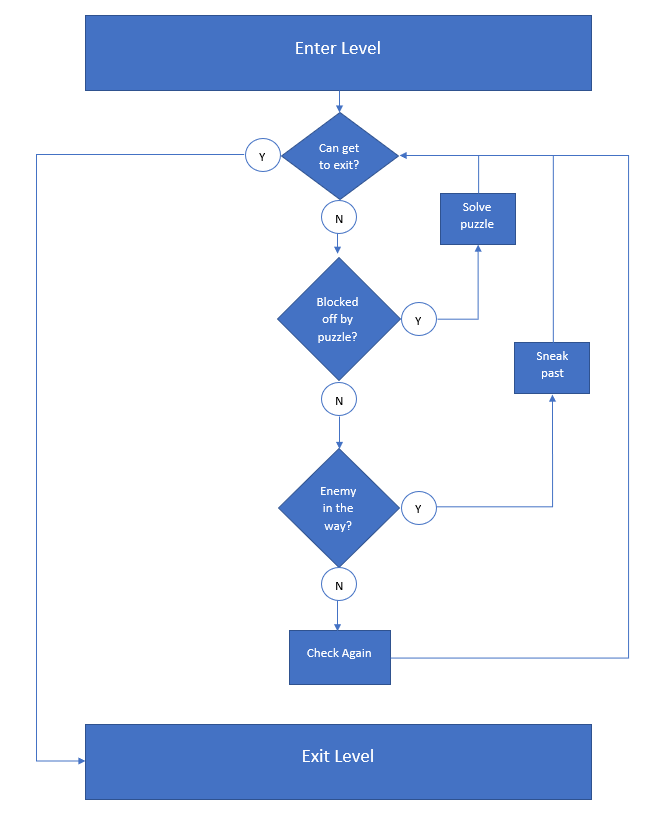
There will be a main menu that allows exiting the application and starting the levels. In the game there will be a pause menu that allows the player to exit back to the main menu. Also a level select screen to select levels to play that have been unlocked.

The win condition is to reach the end of the levels and find the safe zone.

## Levels

The game would include a wide array of levels and so at this stage of development we cannot be sure of the specific levels to be developed. It is likely there would be some levels made to introduce new mechanics to a player, such as a small level with two possible progression routes, one of which contains a cctv camera which sends out robots when the player moves past its vision range.

## Flowchart



## Editor

No editor will be created in the engine for the game. The dungeons will be static and easily created from a tile set.

## Characters

* Player: The player controls an unknown entity contained within the body of a destroyed robot, with only its arms and legs able to protrude. It can walk around using its legs, and move objects with its arms.
* Cultist: The cultists are mentioned in the synopsis as having created the player by infusing a demonic entity into the empty shell of a destroyed robot.
* AI: The AI is mentioned in the synopsis of the game as having appeared in the world and begun to destroy humanity.
* Robots: The robots are controlled by the AI and are used by it to end any human life encountered by them.

## Story

A futuristic AI arrives in a medieval fantasy world and begins to attack the human population using its robotic army. A group of cultists come together in order to summon a powerful demonic presence into the body of a destroyed robot as the last chance to defeat the AI.

## Game World

The majority of the game takes place within a dungeon that the player was created in. This dungeon resides in a medieval style world, although it contains arcane elements such as the existence of higher powers which humans interact with. This world has been invaded by an artificial intelligence from the future, and as such has some futuristic elements within it.

# Required Media List

## User Interface Assets

a font will be required to display text to the screen for the UI.

## Environment Assets

img\TileSet-7-420x60:



Created with:

<https://opengameart.org/content/32x32-stone-brick-sprite> under CC0 licence

by Luckius

<https://opengameart.org/content/rope-ladder> under OGA-BY 3.0 licence

by AntumDeluge

<https://opengameart.org/content/protruded-metallic-texture> under CC0 licence

by looneybits

img\crate:



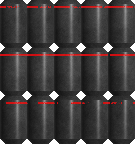
Created with:

<https://opengameart.org/content/3-crate-textures-w-bump-normal> under CC0 licence

by Lucas Miller

## Character-based Assets

img\player\_spritesheet:



Created with:

<https://3dtextures.me/2021/03/18/metal-scratched-008/> under CC0 licence

<https://3dtextures.me/2019/10/25/metal-painted-001a/> under CC0 licence

by Joao Paulo

## Music Assets

sounds/soundtrack.ogg under CC BY 3.0 licence by Kris Klavenes

## Sound Effect Assets

sounds/buttonclick.wav under CC0 1.0 licence by dotY21

sounds/enemywalk.wav under CC by 3.0 license by PsychoPancake

sounds/playerwalk.wav under CC0 1.0 license by Archos

sounds/cratedrag.wav under CC by 3.0 licence by et\_graham

# Technical Specification

## Development Platform and Tools

* Visual studio
* C-make
* SFML library
* Box2D library
* Engine library
* Tile\_Level\_Loader library

## Delivery Mechanism

The user will be provided with an installer which will generate the required files.

## Game Engine

having a windows 10 capable PC with Visual studio 19, using the SFML library and Latest version of CMake.

## User Interface Technical Requirements

1920x1080 will be the resolution for the game

## Controls Technical Specification

Mouse and Keyboard

## Network Requirements

Single player, offline game.

## System Parameters

1 player

## Flowchart

