Technical Design Document Template

# Revision History

|  |  |  |
| --- | --- | --- |
| Version | Description | Date |
| 1.0 | Initial document | 6/08/20 |

# Development Environment

## Game Engine

## IDE

Visual Studio

## Source Control procedures

Git

## Third Party Libraries

Raylib

## 2.5 Other Software

Sonic the Hedgehog Sprites

Shadow the Hedgehog Sprites

Chaos Emeralds and Master Emerald sprites

Classical Temple tiles

# Game Overview

## Technical Goals

* 60 FPS
* Animation
* Challenging AI

## Game Objects and Logic

* Sonic the Hedgehog
  + Behaviour:
    - Keyboard (If the player chooses to play as Sonic)
    - Seek
    - Attack
  + Purpose:
    - A playable character
* Shadow the Hedgehog
  + Behaviour:
    - Keyboard (If the player chooses to play as Shadow)
    - Seek
    - Attack
  + Purpose:
    - A playable character
* Chaos Emeralds
  + Purpose:
    - Main collectable item in the game
* Master Emerald
  + Purpose:
    - Finishing point for the game

## Game Flow

When the game starts, the player can choose whether they want to play as Sonic or Shadow. Then when the game starts the player can then move their character with the arrow keys. The player must gather all the Chaos Emeralds and then go to the Master Emerald first to win the game. If the AI gathers all the Chaos Emeralds and then go to the Master Emerald first, then the player loses. If all the Chaos Emeralds have been collected, and the player and the AI don’t have 7 Chaos Emeralds in total, then the player will have to hit the AI to steal a Chaos Emerald from it but the only way to hit the AI is with an attack that charges overtime. Once the attack is charged then the player can press the spacebar to attack the AI if it is in range.

# Mechanics

* Movement
  + Left Arrow Key to move left
  + Right Arrow Key to move right
  + Up Arrow Key to move up
  + Down Arrow Key to move down
* Attack
  + Spacebar to attack once the attack is ready
* Collect
  + Once the player is within reach of a Chaos Emerald, the Chaos Emerald will disappear and the player’s Chaos Emerald count will go up by 1.

# Graphics

This game will be a top-down 2D race game.

# Artificial Intelligence

# Physics

I will be using Raylib’s collision methods to prevent the player and AI from going off the map.

# Items

|  |  |  |
| --- | --- | --- |
| Item | Description | Amount |
| Chaos Emerald | The main item that needs to be collected to win the game. | 7 |
| Master Emerald | The finish point when the Chaos Emeralds are collected. | 1 |

# Game Flow

## ‘Level’ structure

The map will be loaded from a .png file. The borders will be created from Raylib’s rectangle, but they will not have any colour to conceal them from the player.

## Objectives

The players objective is to collect the 7 Chaos Emeralds and get to the Master Emerald.

# Levels

# Interface

## Menu

## Camera

## Controls

# Asset List

# 13.0 Technical Risks