Software Requirements Specification

for

Jumperman

Version 1.0

Prepared by Lim Wee Boon

Team Only4

02/06/2021

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Lim Wee Boon	02/06/21	Creation of SRS	1.0

1. Introduction

1.1 Product Scope

Jumperman is a 2D pixelated platformer inspired from previous retro platformers but aims to inject unique features into the genre to relish a refreshing gameplay experience for players trying in a tried-and-tested genre. In Jumperman, the player will need to defeat all the enemies in the level and reach a specific tile to clear the level. As the player progresses through the game, the player will have to interact with various types of platform tiles and enemies, overcoming increasingly difficult challenges to advance to the next level.

Aimed towards a more general and casual playerbase, Jumperman is designed with simple but interesting features that can deliver a unique gameplay experience for them without too much effort and investment into the game similar to popular mobile games such as Angry Birds. The product intends to be a simple game with interesting twists in its gameplay features to make it addicting and enjoyable for all.

1.2 Document Conventions

This document will be documented with the following conventions:

1. All documentation will be worded in Arial font black.
2. The header title will be using bolded font size 23 with Heading 1 Style.
3. The subheaders will be using bolded font size 17 with Heading 2 Style.
4. The content under each subheader will be using normal font size 12.
5. Each header title will be numbered in sequence. (e.g. 1. Introduction, 2. Overall Description)
6. Each subheader will append a dot behind the header number and be numbered in sequence (e.g. 1.1 Product Scope, 1.2 Document Conventions, etc.)

1.3 Intended Audience and Reading Suggestions

This document entails the product vision, the product features and the technical and visual specifications of Jumperman. The specifications of the technical and visual requirements is intended to outline the structure and components of the product for the programmers and designers of the development team to be aware of. The document also outlines and describes the core mechanics, gameplay elements and features of the product for the interest of stakeholders.

For marketing staff, documentation writers, testers and project managers. Reading section 2 (Overall description) of this document will more than sufficiently give the reader a gist of the product.

For developing teams, reading sections 2 through 4 is necessary to give the reader an understanding about what the product is. Section 2 will be an overhead glance of the product, section 3 will cover the tools used in the development lifecycle of the product. And finally, section 4 will cover the specifics of each system feature.

This document will first give the overall description of what the product is about, its core features, gameplay mechanics and the unique selling point of the product. It will provide a baseline structure of how the product flows and works, and the different characteristics and functions of the product. It will also go into detail on the specific functions that form the core features of the product and outline the interface designs of the product.

1.4 References

Gustav_k. (2019). Block Steady. Retrieved from https://gustav-k.itch.io/block-steady

Firelight Technologies Pty Ltd. (2020, December 18) FMOD API User Manual 2.00.15. Retrieved from https://fmod.com/resources/documentation-api?version=2.0&page=white-papers-getting-started.html

2. Overall Description

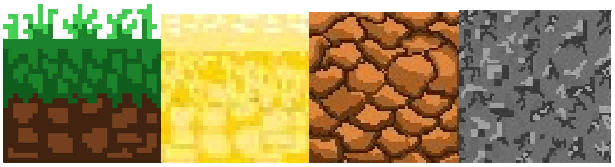
2.1 Product Perspective

Jumperman is a 2D platformer game that is inspired by various retro platformer games and is a standalone product. It is designed to follow the simplistic linear progression of standard platformer genres but adds a unique gameplay element. Jumperman consists of the following features and mechanisms:

Player:

This will be the sprite which the player will control.

Tile designs:



These will be the tiles that Jumperman will use to build its levels. Each tile will serve a different purpose which will be further elaborated later.

Player Movements:

The player will be able to perform basic movements in the game to traverse across the levels. The player can move left, right and jump across platform tiles and on top of enemies. The physics function of the game engine will also allow the player to climb up walls and platform tiles to overcome the level challenges.

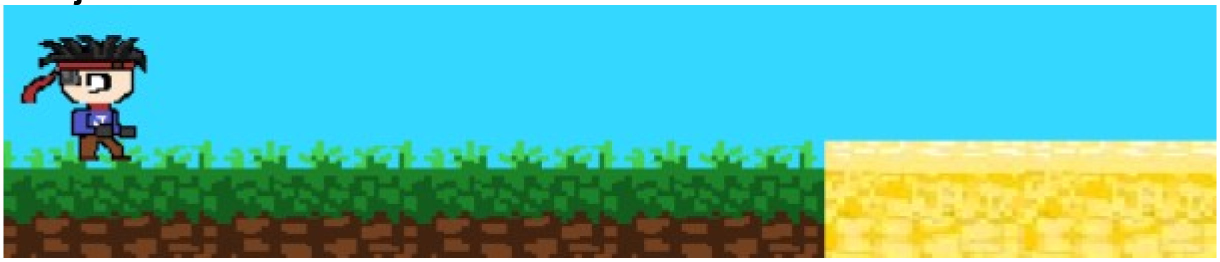
Game Objective:

The player needs to traverse across the level and reach the level objective. The level can only be cleared if the player is able to kill all the enemies in the level, survive the collapsing tiles and reach the objective. While there will be various gimmicks in different levels, the objective across all levels in the game will remain the same.

Different Platform Tiles:

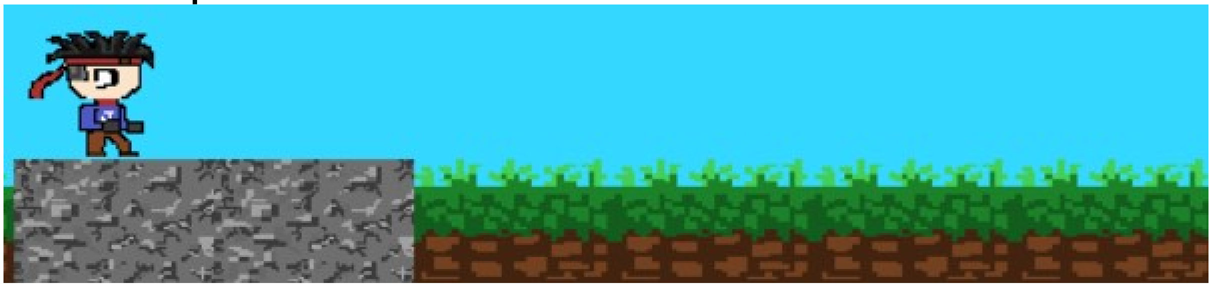
The game will consist of multiple levels that are built based using different platform tiles with different functionalities. They are categorized into three types: Firstly, the "Objective Tile". Second is a family of "Collapsible Tiles" and lastly "Non-Collapsible Tiles".

1. Objective Tile:



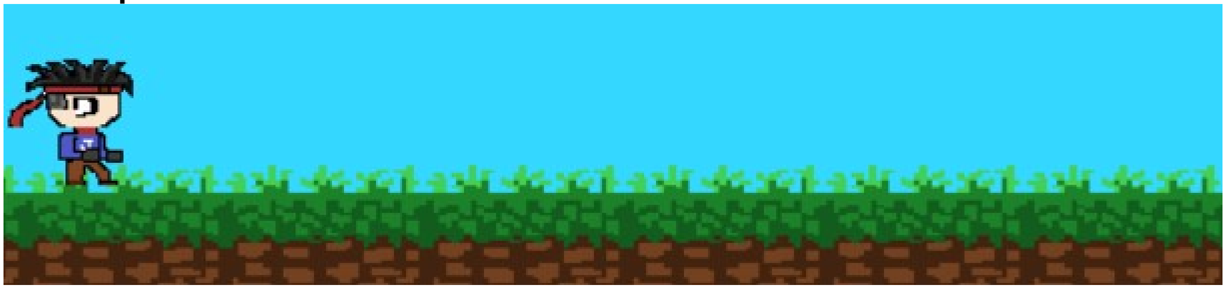
A gold-colored tile that acts as the level objective that the player has to reach to clear the level.

2. Non-Collapsible Tiles:

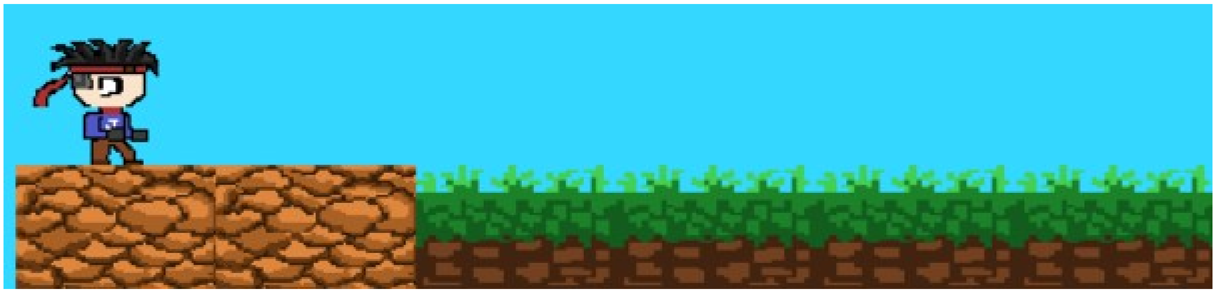


Gray tiles that are used on the level's starting point and several parts of the platforms to act as a checkpoint in the level. They allow the player to traverse across without any changes.

3. Collapsible Tiles:



Grass tiles are tiles that will collapse whenever the player touches it upon killing an enemy in the level, and they form the core part of the level challenges and the gameplay. The player will have to jump over to other platforms before the tiles completely collapse and reach the objective tile.

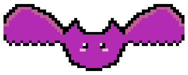


Special Tiles are tiles that players can choose if he/she wants to make it behave like a regular collapsible tile. The twist is that this tile is collapsible on demand by pressing the key down or ‘s’ key.

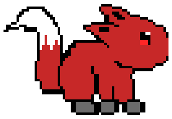
Enemies:
Jumperman will consist of a multitude of enemies that will form part of the level challenges. The player will have to defeat the enemy by jumping atop of it while avoiding getting killed by it. There will be three enemy types present in the game, and each type will have their own movement patterns and behaviours:

Slime: Ground enemy that moves left and right

Flying Bat: Flying enemy that flies in “V” shape pattern



Jumping Squirrel: Ground enemy that moves left, right and can jump



2.2 Product Functions

The following lists below entails the brief descriptions of the major functions and mechanics that will be presented in the game.

2.2.1 Movement

A major function in the game that allows and dictates the movements of objects in the game. The function can be categorized into two components:

Player Movement

Allows the player to move left, right, jump and interact with special tiles via movement controls.

Enemy Movement

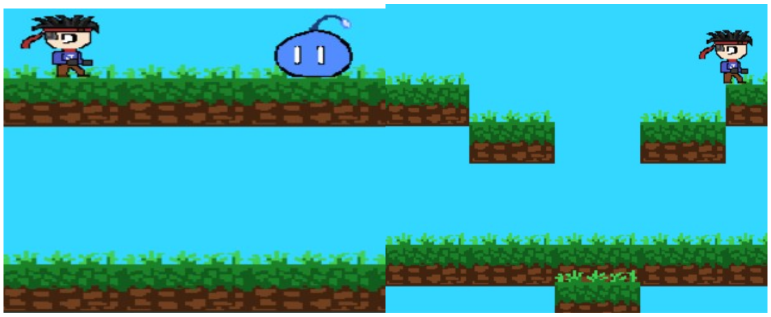
Dictates the movements of different enemy types in the game. Each enemy type has unique movement patterns:

- **Slime:** An enemy in the game can move left and right. Movement is constrained on platform tiles.
- **Flying Bat:** An enemy in the game that flies in a "V" shaped pattern. Movement is not constrained to any object.
- **Jumping Squirrel:** An enemy that can move left and right and jump at fixed intervals. Movement is constrained on platform tiles.

2.2.2 Tile Platform

Jumperman consists of multiple tiles with different functionalities and roles in the game. Below details specifies the types of tiles that will be functioning in the game:

- **Non-Collapsible Tiles**: Tiles that allows the player to traverse across without any change.
- **Collapsible tiles:** Tiles that allows the player to traverse for a brief moment before they collapse.



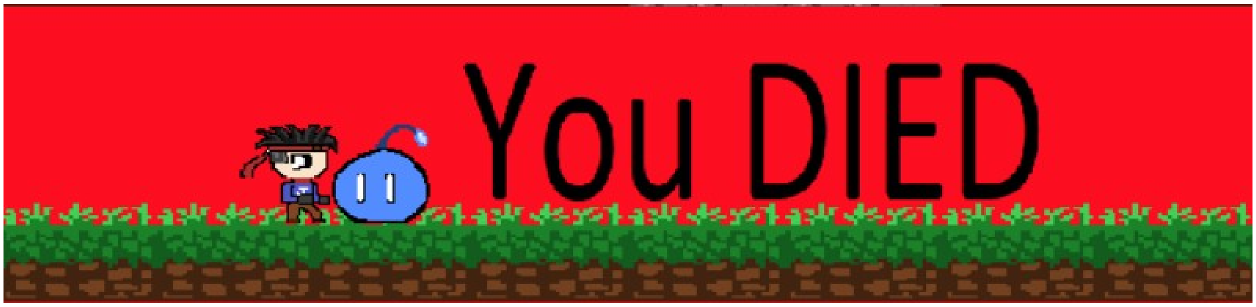
Example of the tiles collapsing after the player jumps on top of the enemy.

- **Objective Tile:** Tiles that serve as the objective of the level for the player upon contact. The tile acts as a Non-Collapsible Tile and will trigger the victory screen.

- **Special Tiles:** Tiles that act like Collapsible Tiles, but may collapse on demand by pressing the down input.

2.2.3 Defeat conditions

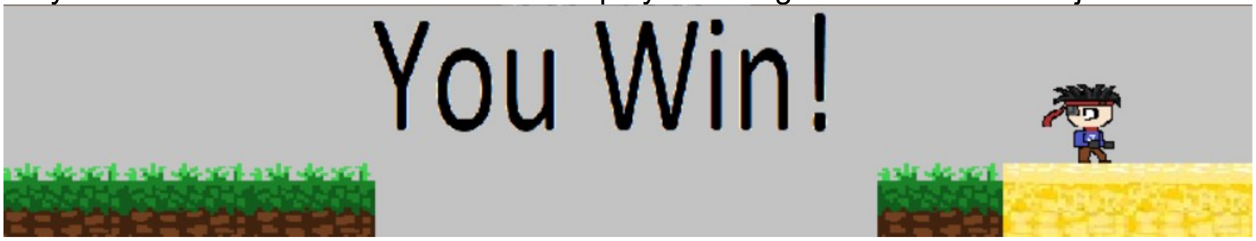
Player's stats: Player is dead once it falls off the tiles or collides with enemy.
Enemy's stats: Enemy is dead once player jumps on it.



You DIED

2.2.4 Victory condition

Player wins when all tiles are cleared and player manage to stand on the objective tile.



You Win!

2.2.5 Level Progression

As the player progresses past each level, the difficulty of the levels will increase as well. The number of challenges such as the number of enemies standing in between the player and the goal tile, and the complexity of the level will increase proportionally.

2.3 User Classes and Characteristics

The simplistic design of Jumperman is intended to draw a casual and general playerbase into the product. As the game boasts simple-to-learn mechanics and intuitive gameplay elements, the user base generally do not need to invest time and effort into learning or mastering the controls as the game does not possess any deep or complex mechanics.

2.3.1 User classes:
The expected user classes that will be playing Jumperman are:

- Casuals players
- Hardcore players

Casual players are users who are less competitive users who expect less serious and less mechanically intensive gameplay. The general demographics of casual players are those who do not have time or are disinterested in investing too much effort in learning the mechanics of games. Jumperman offers a puzzle-like gameplay experience added on top of the platformer genre that rewards players who move and defeat their enemies strategically to advance into the later levels.

Hardcore players are users who are competitive and seek to obtain the highest possible scores in most games that they play. Jumperman offers a leaderboard system that keeps track of the highest scores obtained when playing on the local device. This score will be scaled off enemy kills, tiles collapsed and the time taken to clear the stage. These elements incentivise the more hardcore players to replay the stage and optimise their strategies for the highest possible score.

2.4 Assumptions and Dependencies

Assumptions:
It is assumed that within the core audience that Jumperman is designed for, the majority of the intended audience possesses Windows OS and are familiar with the system to a usable degree. Jumperman will be developed with Windows OS compatibility in mind but will not be integrated into other OS such as Linux.

External dependencies:
This section of the document will cover dependencies which are required for the development of this game.

- **Alpha Engine**
  A C++ external library which consists of certain basic functionalities such as Graphics and Math amongst others to aid in the development of the game. This library is provided by the DigiPen Institute of Technology.

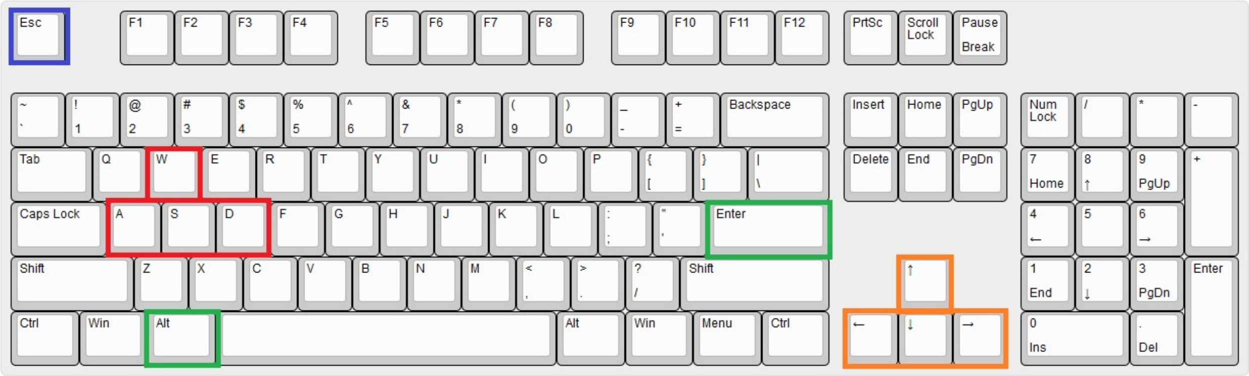
- **PixilArt**
  An online pixel art editor used for designing and developing art assets and spritesheets for the game.

- **FMod**
  Open source library written in C++ for the introduction of audio inside the game.

3. External Interface Requirements

3.1 User Interfaces

3.1.1 Controls and Key Configurations



Keybinds:

Label	Keybind/Shortcuts	Function
1	Up Arrow Key	Player Jump
2	Left Arrow Key	Move Left
3	Right Arrow Key	Move Right
4	Down Arrow Key	Interact with Special Tiles
5	W	Secondary Player Jump
6	A	Secondary Move Left
7	D	Secondary Move Right
8	S	Secondary Interact with Special Tiles
9	ESC	Pause Game
10	Left Alt + Enter	Toggle Fullscreen Mode

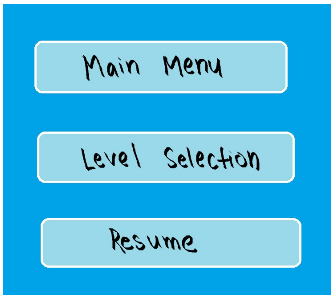
3.1.2 Gameplay Interfaces

The user interfaces in Jumperman are connected to one another. The player is able to traverse from the menu screen to level selection and then to the gameplay itself. The player also has the option to return to the previous screens. Refer to Diagram 1, Appendix B for the full UI flowchart.

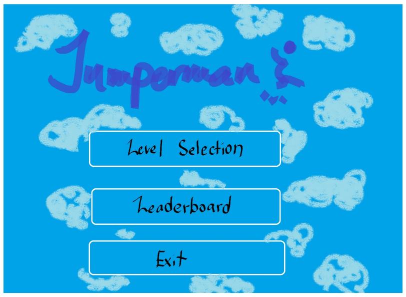
Jumperman's UI will encompass menus of the following:

- Main Menu that navigates to Level Selection, Leaderboard or Quit.
- Level Selection that allows the user to move any level that has been unlocked.
- Gameplay UI.
- Pause Menu that allows user to navigate back to main menu, level select or resume the game.
- Game Over Menu that allows the user to restart or to go back to the main menu.

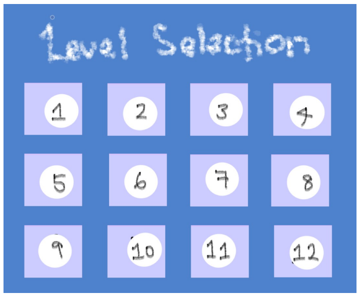
The game will also consist of a Leaderboard interface to display high scores.



Example of Main Menu UI

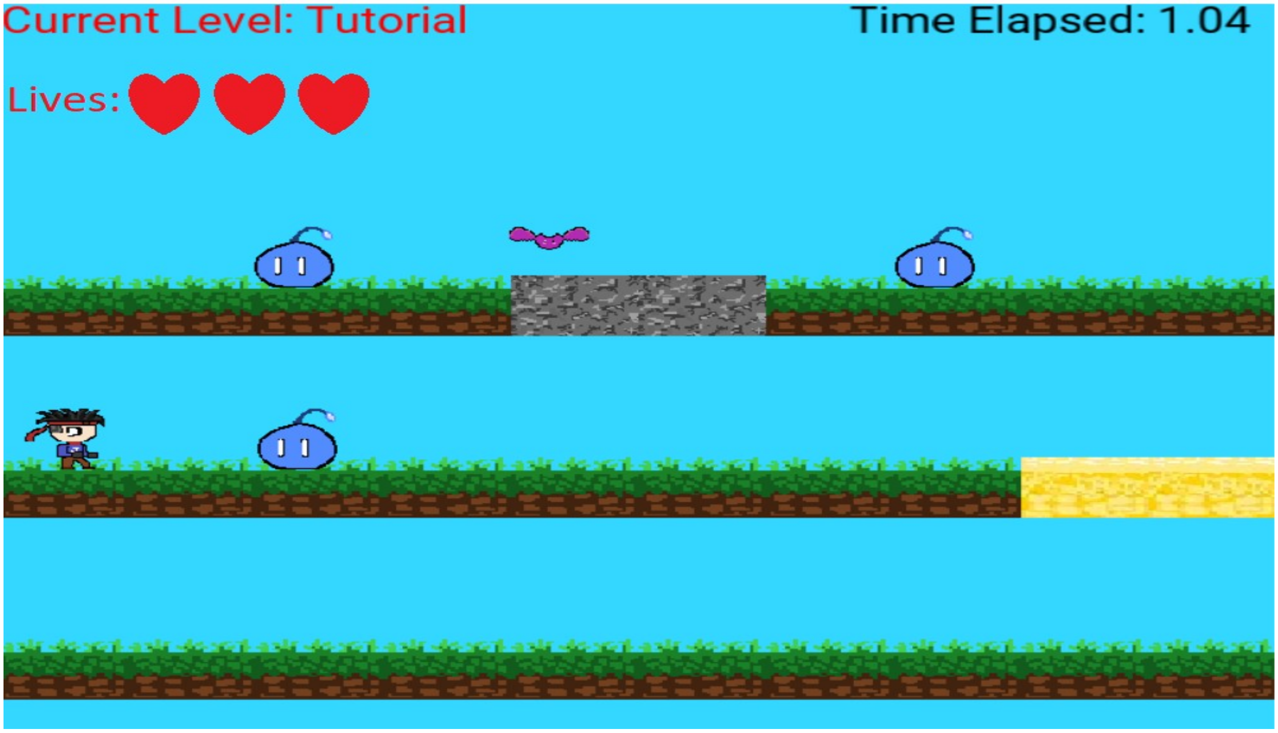


Example of Pause Menu UI



Example of Level Selection UI

Gameplay UI



During gameplay, there will be sprites loaded in that represent the player, the platform tiles and the enemies. The current level the player is in, the number of lives the player has and the timer will also be shown during the gameplay. Displayable data will be shown at the top corners of the screen as shown in this example.

4. System Features

4.1 Platform Tiles System

4.1.1 Description

The game features four major types of platform tiles: Non-Collapsible Tiles, Non-Collapsible Tiles, Special Tiles and Objective Tiles. Non-Collapsible Tiles are tiles that will not react upon player's contact, acting as a spawn tile and checkpoint tiles in the game level. Collapsible Tiles are tiles that will fall off the level map when an enemy is defeated on it due to player contact. Special Tiles are Collapsible Tiles that will allow the player to manually collapse it. The tiles are all designed to be a gameplay challenge for the player to overcome. The player will have to jump to other platforms before it collapses, or the player will fall to his death. Objective Tiles are tiles that serve as the objective area for the player to reach to clear the level and the game. This feature will be on the high priority of the development as the feature serves to be one of the key components of its gameplay.

4.1.2 Function Stimulus

On Non-Collapsible Tiles, the player will be spawning on one and will not move even upon contact. Non-Collapsible Tiles allow the player to stand and traverse over without risk of falling off the level map, but also allows in-game enemies to move on it as well.

On Collapsible Tiles, the player will be able to traverse and jump over to other platforms before the tile collapses. The tiles will act like Non-Collapsible Tiles until an enemy in the level is killed. Upon defeating the first enemy, all surrounding Collapsible Tiles in the level will begin to collapse after the player makes contact with it. The player loses when he falls out of the level map with the Collapsible Tiles.

On Objective Tile, the player will be able to clear the level upon reaching the Objective Tile. The Objective Tile will act like a Non-Collapsible Tile throughout the level.
Special Tiles are tiles that the player will be able to interact with the down button, which will cause the tile to collapse as a manual collapse feature. The Special Tiles will behave like Collapsible Tiles would in any other scenario.

4.1.3 Functional Requirements

The detailed specifications of the functions are as followed:

User Case for Non-Collapsible Tile

- Allow the player to move across the tile without any effects.
- Must be unaffected by the ripples of surrounding collapsible tiles.

User Case for Collapsible Tile

- **Condition:** Tile only collapses if an enemy dies in the level.
- Before Condition: Allows the player to move across the tile without any effects.
- After Condition: Upon defeat of an enemy, the tile will collapse if the enemy was defeated on it or it is affected by the ripples from a previous surrounding Collapsible Tile.
- After Condition: There is a buffer time before the tile collapses if the enemy was defeated on the tile, allowing the player to move across for a moment.

User Case for Objective Tile

- Upon contact, the tile will enable the player to clear the level.
- Must be unaffected by the ripples of surrounding collapsible tiles.

User Case for Special Tile

- **Condition:** Allows the player to move across the tile without any effects and allows interaction using the down key/ ‘s’ button.
- Before Condition: Tile acts as Collapsible Tile if an enemy dies in the level.
- After Condition: Upon interaction with the player, the tile will collapse and fall off the level map.

4.2 Enemies

4.2.1 Description

Jumperman will feature 3 types of enemies: ground, flying and jumping enemies. The enemies in Jumperman serve as level challenges that the player must overcome in order to clear the level and progress in the game. Each enemy type will feature different movement patterns, functionalities and AI. All enemy types will have fixed behaviours and movement patterns pertaining to their type.

The first being a grounded enemy that moves along a straight line of tiles. The second kind being an aerial enemy that hovers over an area on the map in a ‘V’ shaped pattern. The third kind of enemy will move along a set path of tiles and jump at set intervals.

4.2.2 Function Stimulus

The only way for the player to kill the enemies is to use the jump feature and land on top of the enemy. Upon successfully killing any enemy, a ripple effect will be triggered, causing all surrounding collapsible tiles to collapse starting from the one where the enemy was killed. Should the player come into contact with the enemy by any other means, the player will die and will be forced to restart from the start point**.**

4.2.3 Functional Requirements

Common enemy requirements:

- Kills the player upon contact.
- Dies if the player lands on top of it.
- Produces a ripple effect which causes any surrounding collapsible tiles to fall after a delay.

The specifics of the traits unique to the types of enemy are as follows:

Land Slimes:
Will only traverse along the tile along its x-axis.

Aerial Bats:
Floats over tiles following a ‘V’ shaped trajectory.

Jumping Squirrel:
Movement is constrained to left and right. Jumps at fixed intervals.

4.3 Player

4.3.1 Description

The game features basic movement mechanics for the player. In the game, the player is able to move left and right across platforms, jump over to other platforms and interact with special tiles in certain levels.

4.2.2 Function Stimulus

Upon left and right key input, the player is able to move the player sprite left and right in the level. Upon the up key input, the player is able to jump. Pressing a combination of directional arrow keys and jump input allows the player to jump across platforms. Pressing the down key allows the player to interact with the Special Tiles.

The player is able to kill the enemy upon jumping on top of it. The player is able to die upon contacting the enemy in any other way. The player is affected by gravity and will fall if the player is not standing on any tile. The player will die upon falling off the level map.

4.2.3 Functional Requirements

User Case: Player Movement

- The player can move left and right on a platform tile.
- The player can jump between platforms.
- The player can interact with a Special Tile.
- The player can fall off the level map if he is not standing on any tile.

User Case: Interacting with Enemy

- The player can defeat enemies by jumping and landing on enemies.
- The player will be defeated if they make contact with the enemy in any other way.

User Case: Interacting with Tiles

- The player can move across tiles.
- The player can interact with the Special Tile.

4.4 Level System

4.4.1 Description

The game features a multitude of levels with different challenges, complexities and designs. The player can unlock levels through a linear progression of the game by clearing one level to unlock the next. The game will also allow the player to navigate and select which level the player wishes to play if the level is unlocked.

4.4.2 Function Stimulus

In the level navigation menu, the player is able to select and play any levels he has unlocked. Levels are unlocked if the player has cleared the level prior. The player is unable to select locked levels in the navigation menu if he did not clear the previous level.

Each level will feature various platform tiles and enemies that are arranged according to the level design. The player will be able to unlock the next level upon clearing the level objective.

4.4.3 Functional Requirements

User Case: Navigation

- Player can freely select and play any levels they unlocked.
- Player cannot select levels that are locked.

User Case: Gameplay

- Level should load the required level assets for gameplay.
- Player will be faced with different challenges depending on the design of the level.

Appendix A: Glossary

User Interface: An easy-to-read graphical interface that the player can read and interact with via the monitor screen and keyboard inputs.

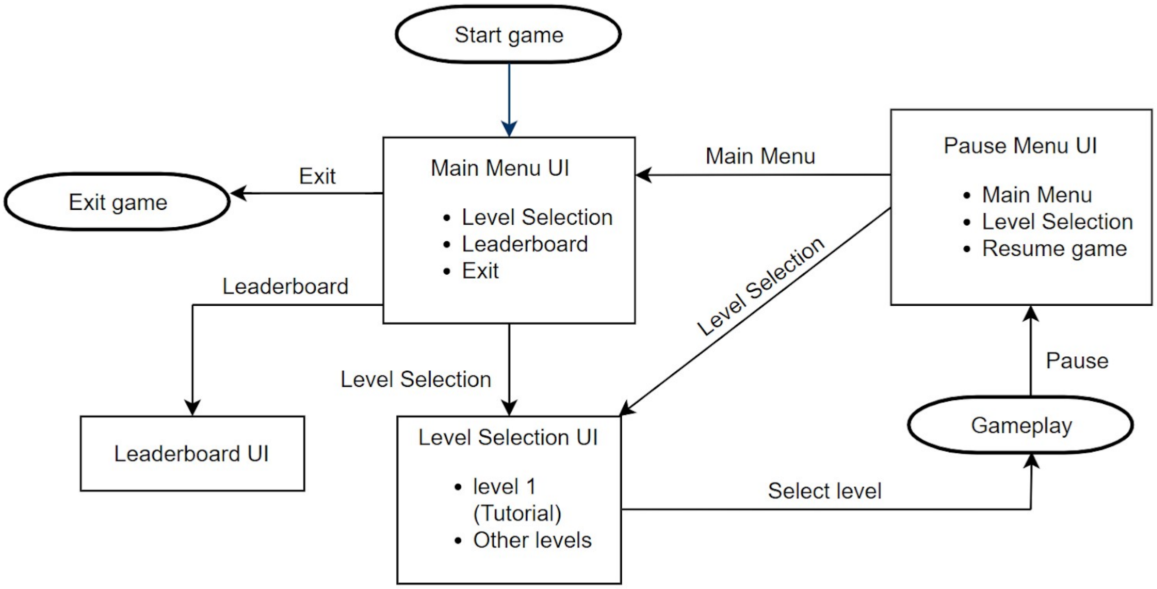
Keybind: A function or operation of the product assigned to a specific key input from the keyboard. The function or operation activates when the assigned key is pressed.

OS: Operating System.

Appendix B: Analysis Models

Diagram 1

User Interface Flowchart


Start game
Main Menu UI
• Level Selection
• Leaderboard
• Exit
Exit
Exit game
Leaderboard
Leaderboard UI
Level Selection
Level Selection UI
• level 1 (Tutorial)
• Other levels
Main Menu
Pause Menu UI
• Main Menu
• Level Selection
• Resume game
Level Selection
Pause
Gameplay
Select level