|  |
| --- |
| Technical Design Document |
| Coffee Mania |
| By William Tjang and Pasqual Fletcher |

|  |
| --- |
|  |

Contents

[1.0 Revision History Version 2](#_Toc11750920)

[2.0 Development Environment 2](#_Toc11750921)

[2.1 Game Engine 2](#_Toc11750922)

[2.2 IDE 2](#_Toc11750923)

[2.3 Source Control procedures 2](#_Toc11750924)

[2.4 Third Party Libraries 2](#_Toc11750925)

[2.5 Other Software 2](#_Toc11750926)

[3.0 Game Overview 2](#_Toc11750927)

[3.1 Technical Goals 2](#_Toc11750928)

[3.2 Game Objects and Logic 2](#_Toc11750929)

[3.3 Game Flow 2](#_Toc11750930)

[4.0 Mechanics 2](#_Toc11750931)

[5.0 Graphics 3](#_Toc11750932)

[6.0 Artificial Intelligence 3](#_Toc11750933)

[7.0 Physics 3](#_Toc11750934)

[8.0 Items Item 3](#_Toc11750935)

[9.0 Game Flow 3](#_Toc11750936)

[9.1 ‘Mission’ / ‘Level’ structure 3](#_Toc11750937)

[9.2 Objectives 3](#_Toc11750938)

[10.0 Levels 3](#_Toc11750939)

[11.0 Interface 4](#_Toc11750940)

[11.1 Menu 4](#_Toc11750941)

[11.2 Camera 4](#_Toc11750942)

[11.3 Controls 4](#_Toc11750943)

[12.0 Asset List 4](#_Toc11750944)

[13.0 Technical Risks 4](#_Toc11750945)

# Revision History Version

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Description** |
| 1.0 | 14 Oct 2019 | Initial Document |
| 1.1 | 15 Oct 2019 | Added some pickup item mechanics |
| 1.2 | 16 Oct 2019 | Added UML diagrams |
|  |  |  |

# Development Environment

## Game Engine

Unity v2018.3.8f1

## IDE

Windows Visual Studios 2017

## Source Control procedures

GitKraken, Unity Collab

## Third Party Libraries

## Other Software

# Game Overview

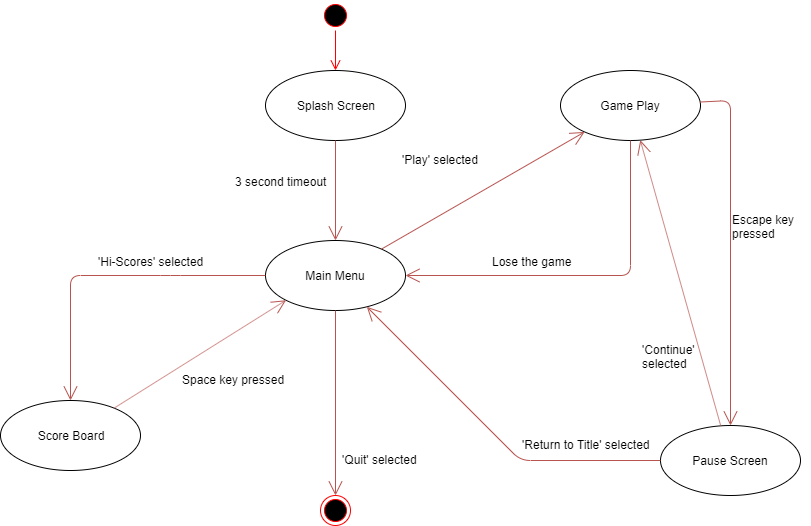
## Technical Goals

* To procedural generate the level for the endless runner as game is being played, to allow for infinite levels and game play.
* To optimize the game to run at 60 frames a second on standard hardware.
* To be played in a 3D environment with a 3D camera view

## Game Objects and Logic

|  |  |
| --- | --- |
| **Game Object** | **Description** |
| Player | Represents the controllable player character |
| Tile Set(s) | The tile sets randomly generated for each level |
| Obstacle(s) | Will cause the player to lose upon frontal collision |
| Pickup(s) | Grants a buff or debuff on the player (refer to section 8) |

## Game Flow



# Mechanics

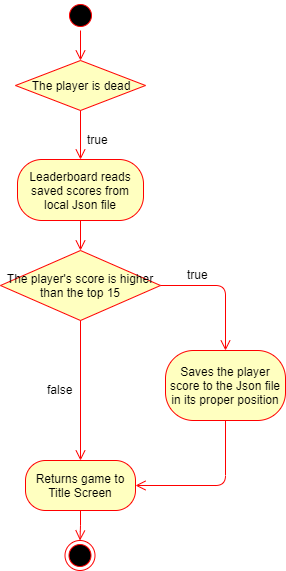
Moving

* Side to side controls
* Strafe: The player switches to an adjacent lane either left or right using the A or D keys\
* Turns: The player turns 90 degrees to the left or right using the arrow keys <- or ->
* Jump: The player jumps a certain altitude using the SPACE key
* Slide: The player slides a certain distance using the S key

Currency

* Beans: The player can spend the bean currency for various benefits in game

Leaderboard



# Graphics

3D environment, stylized art

# Artificial Intelligence

<Describe how AI works, i.e. state machine, fuzzy logic, GOAP. Describe the various behaviours and how they change behaviour, how do the ‘creatures’ in the game evaluate the world>  
<include diagrams/flowcharts showing decision making processes>

# Physics

<if needed>  
<What engine are you using, what features from it (spring? Colliders?) how will physics be handled for objects? (box or sphere collider for objects, capsule for player) need to record specific locations for any reason? Potential slowdowns and how to mitigate.>

# Items

|  |  |
| --- | --- |
| **Item** | **Description** |
| Coffee Bean | Adds 1 currency to the player’s overall account |
| Sugar | Speeds up the player character and grants invincibility for 5 seconds |
| Ice Cubes | Slows down the level speed for 10 seconds (may be subject to future changes) |
| Net | Allows the player to catch all nearby coffee beans in an AoE |
| Barista Knife | Allows player to ignore the next obstacle they crash into and destroy it |
| Milk | Grants a score multiplier buff |
| Tea | Inverts the player’s control for 10-15 seconds |

# Game Flow

## ‘Mission’ / ‘Level’ structure

<Are all levels stored in memory? what data is saved across levels, are levels loaded synchronously to prevent pauses?>

## Objectives

* The player’s objectives are to collect as many coffee beans by running over them, adding them to the players total, to earn rewards.
* The player’s aim is to get as far as they can through the endless runner and score high.
* Their score is calculated by how far they have traveled amongst other factors (e.g. pickups) and evaluated against a local leaderboard.

# Levels

<If any of the Levels require specific behaviours, describe those here>

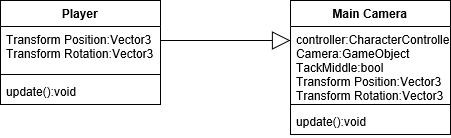
# Interface

## Menu

|  |  |
| --- | --- |
| **Option** | **Description** |
| Play | Starts the main gameplay |
| Hi-Score | Displays the leaderboard |
| Quit | Closes the game window |

## Camera

* 3rd person perspective
* Positioned slightly higher than player
* Angled slightly downwards
* Centered at the middle lane



## Controls

|  |  |  |
| --- | --- | --- |
| **KEYBOARD** | | |
| **Action** | **Command** | |
| Strafe | A (left) | D (right) |
| Turn | <- (left) | -> (right) |
| Jump | ‘Space’ | |
| Slide | ‘S’ | |

# Asset List

<List all files needed, along with known attributes >

# Technical Risks

Risks

* Having 2+ programmers in the project can lead to style and/or data structure inconsistencies
* Some pickup mechanics may not be implemented due to time limit or complexity issues
* Source control issues can severely delay the project, especially with 2+ programmers

Measures

* Pre-planning variable naming, style and data structures conventions between programmers
* Having good time management and prioritization of tasks to minimize time limit issues
* Using multiple source controls in case one fails