Game Design Document

ATTA: A Toaster's Toasty Adventure

"A journey that will

forever be embreaded

in your heart" - Jeric Baga

CS-583 F21

Table of Contents

Game Development Team Members

PRODUCER

Jeric Baga

PRODUCTION MANAGER

Jeric Baga

PRODUCTION COORDINATOR

Jeric Baga

GAME DESIGNERS

Jeric Baga

SYSTEMS/IT COORDINATOR

Jeric Baga

PROGRAMMERS

Jeric Baga

TECHNICAL ARTISTS

Jeric Baga

AUDIO ENGINEERS

Jeric Baga

UX TESTERS

Jeric Baga

1 Game Overview

Title: ATTA: Toaster's Toasty Adventure

Platform: PC Standalone

Genre: Action-Adventure Shooting Platformer

Rating: (E10+) ESRB

Target: Casual gamers (aging from 10 - 30)

      Hardcore gamers

Release date: November 23rd, 2021

Publisher: Jeric Baga

ATTA: A Toaster's Toasty Adventure is a 2D side-scrolling shooting platformer where the Player is a toaster whose clan has been kidnapped by the evil Queen Kato. The Player must travel across the world of Foodlan, fighting off the various Foodie henchman sent by Queen Kato to stop him/her in their tracks. As a toaster, the Player can fight back by shooting toast and can change forms to change the trajectory of where the toast fires. Throughout the game, the Player can find avocados scattered across Foodlan and collect them for points.

2 High Concept / Genre

ATTA: A Toaster's Toasty Adventure sets the Player in a reimagined world where he/she is bestowed with the duty to save the rest of their kidnapped clan from extinction. The Player must survive through each Foodlan landmark by avoiding Foodie enemy projectiles or not falling off the

land to an impending death. They can earn points by collecting avocados across the land and acts as a competitive aspect when playing with friends.

The genre of my game is an Action-Adventure Shooting Platformer. It is heavily inspired by the Super Mario franchise, which really means a lot to the developer as he grew up playing many Super Mario games and still do to this day. As such, many of the gameplay elements implemented had the 2D Super Mario platformers in mind, as those games are the definitive definition of a 2D platformer to him. The projectile-based combat is also a homage to some of the various powerups Mario has received in his games, including Boomerang Mario and Fire Flower Mario.

The idea of using a toaster as a playable character is a homage to the developer's favorite artist, who got hit on the head by a toaster at a very young age. As a result, there are many jokes regarding her lack of book smarts (although she is very talented at what she does best). Using a toaster as a playable character also allowed more room for creativity in terms of the plot, enemy choice, and overall style of the game, rather than making some "generic" 2D platformer.

3 Unique Selling Points

- Unique playable character
- Highly accessible to players of all skill ranges
- Originally composed music

4 Platform Minimum Requirements

Generally, anything made since 2000 should work.

5 Competitors / Similar Titles

- Super Mario Bros. 2D series

- Contra

- Metroid

6 Synopsis

The evil and somewhat dumb-witted Queen Kato has kidnapped every toaster in Foodlan except you, as a plan to annihilate the entire Toaster clan. You are the last hope of the Toaster clan and must travel across Foodlan to defeat Queen Kato. Along the way, you will meet many of Queen Kato's minions who were tasked to destroy you. Adventure and battle your way through Kato's troops across Foodlan and restore the Toaster clan to its rightful place!

7 Game Objectives

The objective of the game is to make it to the end of each uniquely-designed level (multiple levels pending further development beyond the scope of this class) without running out of health or falling off the stage. Utilize the toaster's abilities to shoot toast and switch forms to destroy any enemies that get in the way of your journey to save your clan!. The player can choose to gain as many points as possible for each level by defeating enemies and finding collectibles such as avocados which are scattered throughout each level.

8 Game Rules

ATTA: A Toaster's Toasty Adventure is filled with open environment yet linear levels where the Player must reach the end of the stage without dying, whether that be running out of health or falling off. There exist many enemies in each level that will actively shoot towards the Player if he/she is in their aggro ranges. The Player can move, jump, shoot toast, and change forms; he/she must strategize their choices to get through each level's obstacles and spawned enemies. Different enemies differ in behavior and stats such as aggro range, movement speed, and health. The Player is prohibited from jumping while on an enemy. The Player scores points by destroying enemies and collecting avocados scattered throughout each level.

9 Game Structure
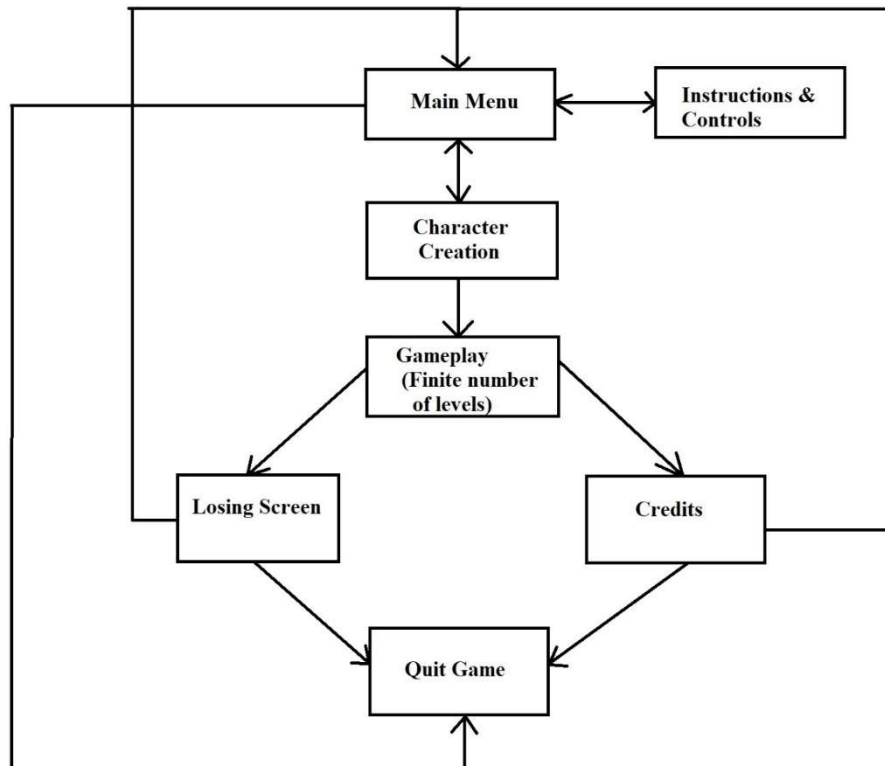
10 Game Play

10.1　　Game Controls

Fire
| A |

Switch Forms
| S |

Jump
| ⬆ |

Move Left
| ⬅ |

Move Right
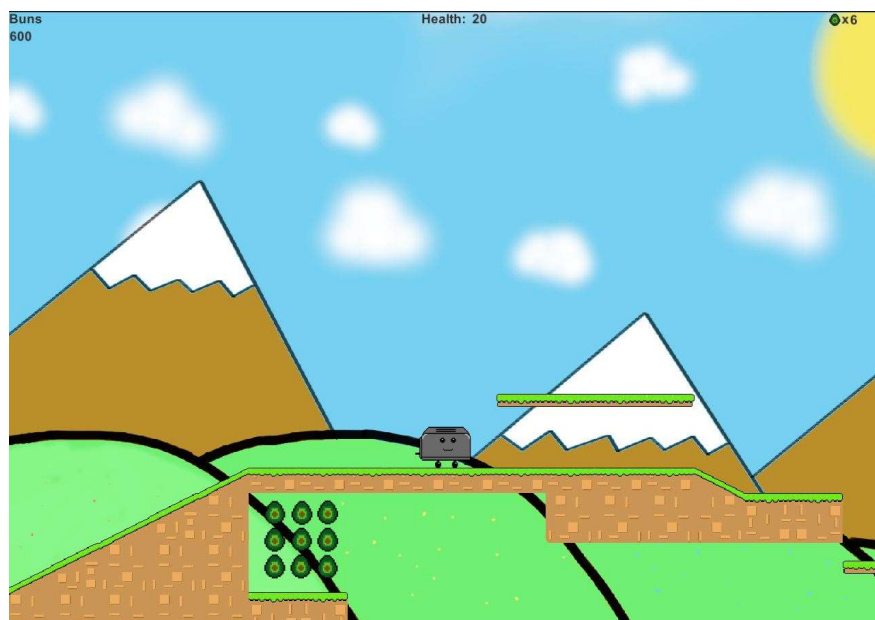| ➡ |
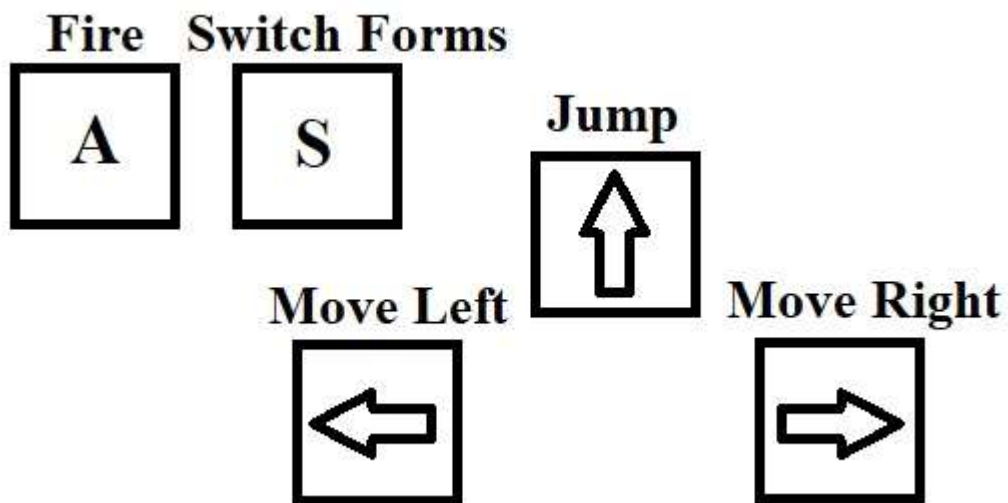
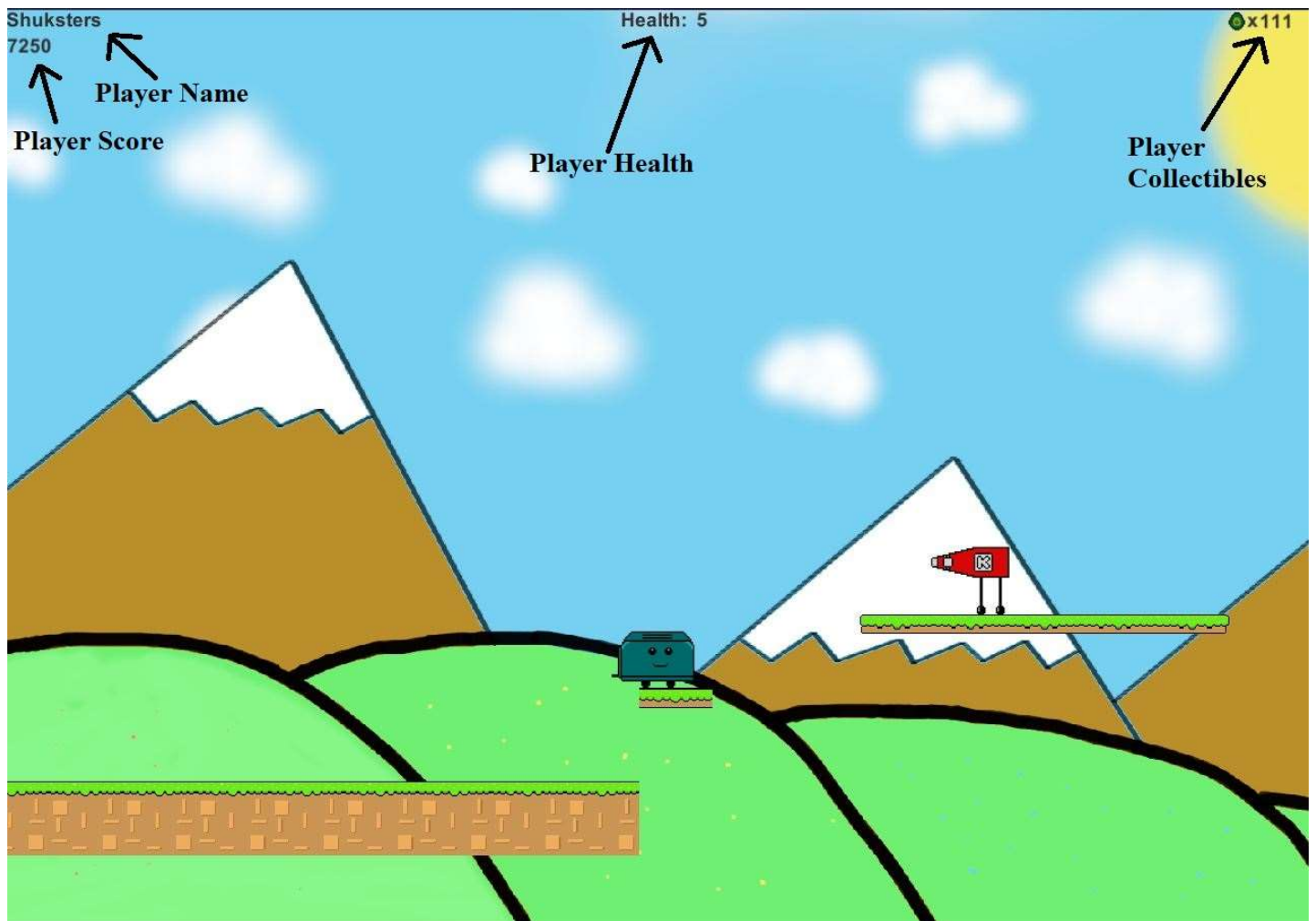## 10.2　　Game Camera

Upon gameplay, the camera focuses on the Player toaster the entire game. It is fixated in such a position that the Player is always at the same position relative to the screen. One can imagine it as a camera man capturing the player and is always mimicking the exact movements and speed of the Player.

10.2.1 HUD

10.2.2 Maps

Level 1: Condiment Countryside

Level 2: Condiment Battlefield
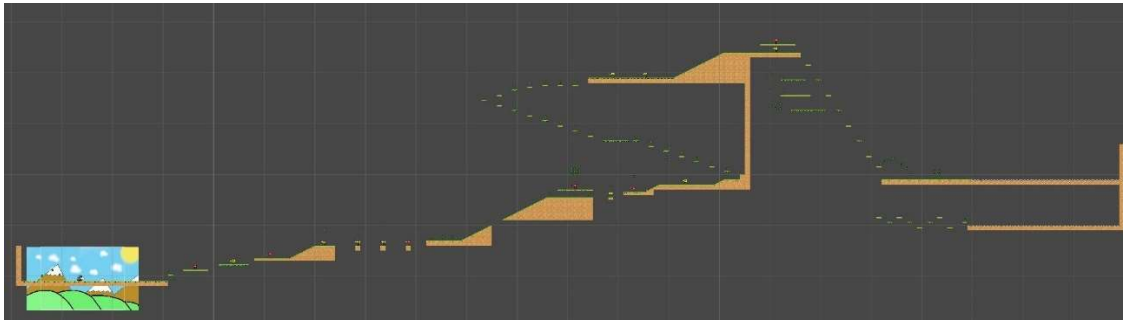
11 Players

11.1 Characters

The toaster represented by the Player is one of the many living toasters in the Toaster clan in the realm of Foodlan. However, Queen Kato has been slowly plotting revenge on the Toaster clan after an unidentified clan member bonked her on the head when she was young. With Queen Kato finally enacting her vengeance on the Toaster clan, it is up to the Player toaster to hero up and rescue his family!

Upon character creation, the Player can choose the toaster's name and color, which will persist during the lifetime of gameplay.

11.2 Metrics

Movement Speed: 4

Jump Height: 5.5

Max Health: Varies based on chosen game difficulty (30, 20, 10, 5, 1).

Attack Damage: 1 (All projectiles are weighed the same amount of damage)

Attack Cooldown: 0.35 seconds

11.3 States

Idle: The idle state is triggered when the Player is not moving. In the idle state, the Player moves up and down in place.

Move: The movement state is triggered when the Player moves in either the left or right direction (Player is pressing down on either Left or Right arrow key). In the moving state, the wheels of the Player toaster rotate to visualize the movement.

Damaged: The damaged state is triggered when the Player is hit by an enemy projectile. In the damaged state, the Player toaster blinks for 1 second and is temporarily immune to damage.

Shooting: The shooting state is triggered when the Player presses the "A" key. During the shooting state, the lever of the Player toaster shoots towards the top to help visualize the toaster projectile shooting out of it.

Death: The death state is triggered when the Player runs out of health or falls off the stage. During the death state, the Player is unable to make any actions as it explodes into flames.

Because there are two playable forms of the Player toaster, each one shares different idle, moving, damaged, and shooting animations.

11.4   Weapons

The Player controls the weapon itself: the toaster! Like a real-life toaster, the Player shoots toast to attack the Foodie enemies summoned by Queen Kato. The cooldown for shooting is 0.35 seconds, and the speed at which the toast moves is 10.

12 Player Line-up
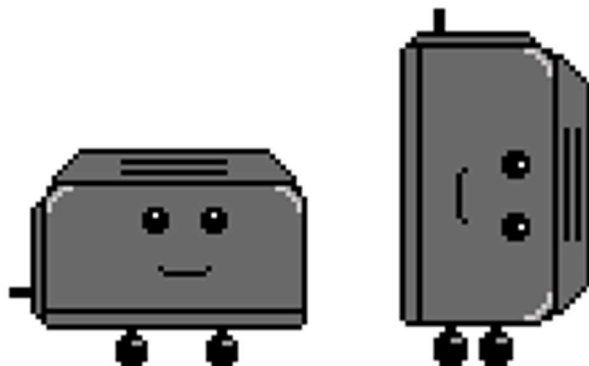
13 NPC

13.1   Enemies

The Foodie enemies are spawned across each level in well-thought out spawn points. Each unique Foodie enemy has a unique amount of health. They patrol a specific amount to the left and right, briefly stopping at both end points. The patrol speed and length depend on where they are spawned. When the Player gets close enough (within its aggro range), an exclamation point will show above the Foodie enemy to let the Player know he/she is in that enemy's aggro range. Aggro range is also dependent on where the Foodie enemy is positioned in the level. From there, the Foodie enemy will shoot in hopes it will hit the player (See 15.3 Artificial Intelligence NPCs).

Ketchup Karen

Health: 2

Move Speed: Varies on location

Patrol Distance: Varies on location

Aggro Distance: Varies on location

Missile Speed: Varies on location

Mustard Mike
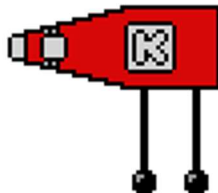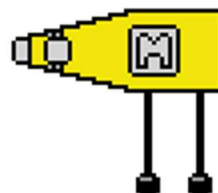
Health: 4

Move Speed: Varies on location

Patrol Distance: Varies on location

Aggro Distance: Varies on location

Missile Speed: Varies on location

13.1.1 Enemy States

Idle: The idle state is triggered when the Foodie enemy is not moving. That is, the Foodie enemy has gotten to either its left or right patrol point and is briefly looking in that direction before turning in the opposite direction. In the idle state, the Foodie enemy moves up and down in place.

Move: The movement state is triggered when the Foodie enemy is patrolling in either the left or right direction. In the moving state, the wheels of the Foodie enemy rotate to visualize the movement.

Damaged: The damaged state is triggered when the Foodie enemy is hit by the Player's projectile. In the damaged state, the Foodie enemy blinks for 1 second.

Shooting: The shooting state is triggered when the Player is within the aggro range of the Foodie enemy. Although there is no visual indication within the Foodie enemy itself, it will shoot it's projectile in the direction of the Player.

Death: The death state is triggered when the Foodie enemy runs out of health or. During the death state, the Foodie enemy explodes into flames.

13.1.2 Enemy Spawn Points

Rather than having Foodie enemy spawn points, enemies were strategically placed across each level in a way that didn't cause any bugs based on the functionality that was able to be produced before the deadline. As such, each Foodie enemy's stats (aggro range, move speed, patrol distance) were carefully edited to ensure they wouldn't encounter any visual/mechanical mishaps during gameplay (See 16. Wishlist).

13.1.3 Enemy AIs

The AI of the Foodie enemies is a simple implementation of a patrol system. However, rather than programming the Foodie enemies to check when its hitting the edge of a platform and turn around, they simply move a certain distance left and right with respect to the original positions they were placed in each level. This made it easier to place enemies on the same platform and not have to potentially spend more time implementing a more intricate AI system, which could've cost the developer time not solidifying other aspects of the game. Although it works, it has its drawbacks, such as having to manually input an enemy's stats depending on where they are in the level such that it doesn't cause any undesirable results.

The Foodie enemies were also given an aggro system that is based on a 2D vector distance between the Player and the Foodie enemy. The AI also ensured that for a Foodie enemy to be aggro'd, it must be facing the actual Player along with that 2D vector distance. The Player's vertical position was also checked to make sure when the Foodie enemies detect the player, it appears as if the Player is actually in their field of view and not just nearby solely due to the horizontal distance. Aggro distances for each enemy were checked and adjusted accordingly to ensure there weren't any interactions that shouldn't happen. For example, a short wall that separates the Player and Foodie enemy such that the Player is still in that Foodie enemy's aggro range. The Foodie enemy shouldn't when it is "facing" the character because it shouldn't see it due to the wall in between them.

Upon being aggro'd, the Foodie enemy shoots a projectile. Rather than stopping it's patrol to shoot, the Foodie enemy just shoots as it patrols. In other words, the patrol system for all Foodie enemies is not interrupted by aggro; they shoot as long as the conditions previously mentioned are met up until they need to around to patrol in the opposite direction, which will automatically fail one of the conditions for them to get aggro'd.

13.2   Allies / Companions

Currently, there are no NPCs that exist to aid the Player during gameplay.

13.2.1 Ally States

See 13.2 Allies / Companions.

13.2.2 Ally Spawn Points

See 13.2 Allies / Companions.

14 Art

The background arts of the level were made in tandem with its theme and name. As the first landmark was planned to be something thematically-related to simple, grassy plains, the developer used his limited art skills to create a visual that captured the aspect of that specific theme. Pending further development, the developer would continue brainstorming ideas for future levels with themes and background arts that capture his ideas.

The developer originally wanted to draw the player, enemies, and tiles, but he eventually realized it would be difficult for him to animate them, so he switched to making them 2D sprites instead. Since the developer wanted to make everything from scratch, he spent a lot of time (especially for the tiles) making sure the sprites, their animations, and the tiles were perfectly aligned so it would look as clean as possible during gameplay. Making the tiles as sprites also saved time in making the levels in Unity using the tile map feature.

The background art for the menus were also made by the developer, using art programs such as GIMP and paint.net. Albeit fairly basic, the simplicity captures the simplicity of the game itself.

14.1   Setting

The game takes place in the world of Foodlan, where landmarks are differentiated by the different types of species that live there. For example, Condiment Countryside and Condiment Battlefield all feature condiments as the Foodie enemies. The grassy theme of these levels was heavily inspired by games the developer has played growing up (See 2. High Concept / Genre). Further development of this game would feature different landmarks with unique Foodie enemies to each one.
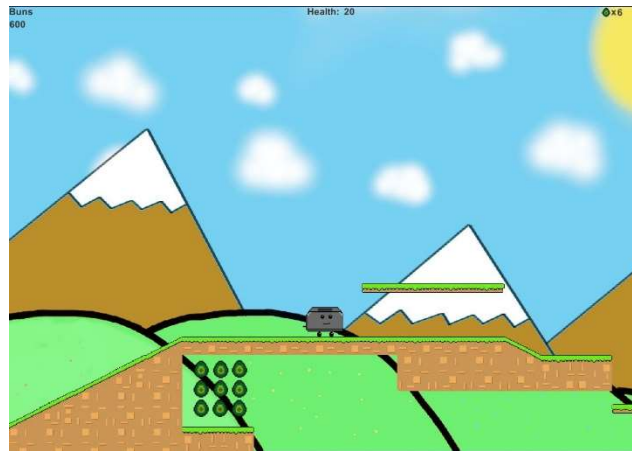
14.2   Level Design

Each level in the game consists of a unique mapping of various platforms and terrain and Foodie enemies to act as obstacles for the Player. Although the objective is linear, each level has its own unique secrets and design that help immerse the Player into the diverse world that is Foodlan. Each unique landmark also carries its own unique level background and tile set (pending further development, see 16. Wish List). The Foodie enemies are spawned only once per level and are strategically placed depending on the design of the level.

14.3   Audio

Having a musical background, the developer was excited to work on the audio aspect of the game. As a musician, he enjoyed composing the background music and sound effects and combining it with his career path.

The developer wanted to compose the level music in such a way that the name of the levels can be imagined while listening to the song. The theme for Condiment Battlefield was originally supposed to be a theme for a boss fight, but the limited timeframe dissuaded the developer from attempting to create a boss fight, although he already had the music in mind for it composed.

The menu music and game over music were also crafted by the developer in a similar manner. Although these sound more generic, the developer is proud to have made multiple pieces of music in a video game he created.

The sound effects were made by the developer randomly trying out different instruments within GarageBand and eventually choosing a recording that he thought was a good sound for each effect.

Although the audio took more time than the developer thought it would, he enjoyed every second of it because he was able to combine his interests in music and video games.

Name	Category	Description
Menu Theme 1	Background Music	Plays during non-gameplay scenes, excluding the Game Over scene.
Game_Over	Background Music	Plays during the Game Over scene, which is triggered when player dies.
Condiment Country	Background Music	Plays during Level 1: Condiment Country
Condiment Battlefield	Background Music	Plays during Level 2: Condiment Battlefield
Level Complete	SFX	Plays when player reaches the end of the level.
Jumping	SFX	Sound when player jumps.
Player Damaged	SFX	Sound when player is damaged.
Player Death	SFX	Sound when player dies.
Enemy Damaged	SFX	Sound when Foodie enemy is damaged.
Enemy Death	SFX	Sound when Foodie enemy dies.
Enemy Detection	SFX	Sound when Foodie enemy detects player.
Enemy Projectile Collision	SFX	Sound when projectile hits an object (Player, projectile, ground)
Collectible Get	SFX	Sound when player collects avocado.

15 Procedurally Generated Content

There exists no content in the game that was automatically generated by any algorithms.

15.1   Environment

See 15. Procedurally Generated Content.

15.2   Levels

See 15. Procedurally Generated Content.

15.3   Artificial Intelligence NPC

See 15. Procedurally Generated Content.

15.4   Visual Arts

See 15. Procedurally Generated Content.

15.5   Audio

See 15. Procedurally Generated Content.

15.6 Minimum Viable Product (MPV)

- Create toaster that moves around and shoots toast from itself.
- Box environment for toaster to move in.

16 Wish List

Powerups: In any future development, add Powerups to give Player more states/behaviors and mechanics not previously possible with the current version of the game.

Enemies: In any future development, add more Foodie enemies so that the player is faced with a diverse set of NPCs. This includes any type of boss enemy.

Enemy AI: In any future development, add more types of enemy AI to make different Foodie enemy species have unique behaviors and states. Improve on previous enemy AI to make it "smarter."

Levels: In any future development, add more levels, which come with its own theme, background art, and tiles.

Platform Mechanics: In any future development, add more complexity to the platform challenges the Player must beat that scales as the game progresses. Examples of new complexity are moving platforms, switches, and blinking platforms.

Score System: In any future development, keep track of the 10 most recent highest scores.

Multiplayer: In any future development, implement a multiplayer feature where players take turns playing each level, very much akin to Super Mario Bros.

17 Bibliography

Every asset used to create the game was made by the developer, Jeric Baga.

Game:

Scenes - All by Jeric Baga

Scripts - All by Jeric Baga

Art:

Animations - All by Jeric Baga, using Unity

Background - All by Jeric Baga, using GIMP and Paint.net

Sprites - All by Jeric Baga, using Piskel

Tile Maps - All by Jeric Baga, using Piskel

Audio:

SFX and BGM - All by Jeric Baga, using MuseScore3, GarageBand, and Audacity