Technical Design Document for:

Candyboiz

Written by Visceral Tilt

Yellow text - written by Ashley Richmond
Purple text - written by Joshua Ferguson
Diagrams by Joshua Ferguson

Table of Contents

1.0 Revision History

Version	Description
1.0	Initial document
1.1	2.0 Development Environment and 6.0 Artificial Intelligence
1.2	3.0 Game Overview, 4.0 Mechanics, 5.0 Graphics, 8.0 Items
1.3	9.0 Game Flow
1.4	The rest of it
1.5	Fixed stuff based on feedback

2.0 Development Environment

2.1 Game Engine

Unity 2018.3.12f1

2.2 IDE

2.2.1 Coding Guidelines

Use Microsoft's coding conventions for C#

- https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/inside-a-program/coding-conventions

2.2.2 Naming Conventions

Naming conventions are specified in above section.

2.2.3 Source Control procedures

Perforce integrated with Unity

2.2.4 Memory limits per system

2GB

2.3 Other Software

Textures/2D graphics:

- Photoshop CC 2019 20.0.5
- Paint.net 4.1.6

3D Models:

- Maya 2019

Audio:

- Audacity 2.3.2

3.0 Game Overview

2.1 Technical Goals

2.1.1 60fps

Keeping a consistent 60fps should be prioritised over fancy graphics. A high frame-rate is important for smooth and satisfying experience within a fast-paced game.

2.1.2 Time-based Events

Frequent scheduled updates to the game to give players unique objectives and unlockables for a limited time.

2.1.3 Multiplayer

8 player matches.

2.2 Game Objects and Logic

2.2.1 Characters

Playable Characters

- 5 unique characters with different abilities
- Controlled by player input, both keyboard and gamepad

Gingerbread Soldiers (minions)

- AI-controlled soldiers that fight for their given team
- Spawn in small groups at timed intervals at their team’s base
- Randomises which lane to take, then moves along it until a target is reached
- Once an opposing team character is within targeting radius, they will move towards them until stopping distance is reached and attack

2.2.2 Weapons

Weapons

- Each playable character starts with their own standard weapon
- Melee weapons and ranged weapons

Bullets

- Travels forward from the spawn point (weapon) until a character of the opposing team is hit or lifetime timer is up
- Deals damage to an opposing character when hit

2.2.3 Arenas

Basic Arenas

- Has two bases - one for each team
- Bases can be destroyed by opposing team
- Spawn points for minions and items

Event Arenas

- Available to players during set timeframes
- Has special events/objectives that can occur

Turrets

- Destroyable by opposing characters
- Shoots at any opposing character in range (one target at a time), prioritising whichever character entered the range first

2.2.4 UI

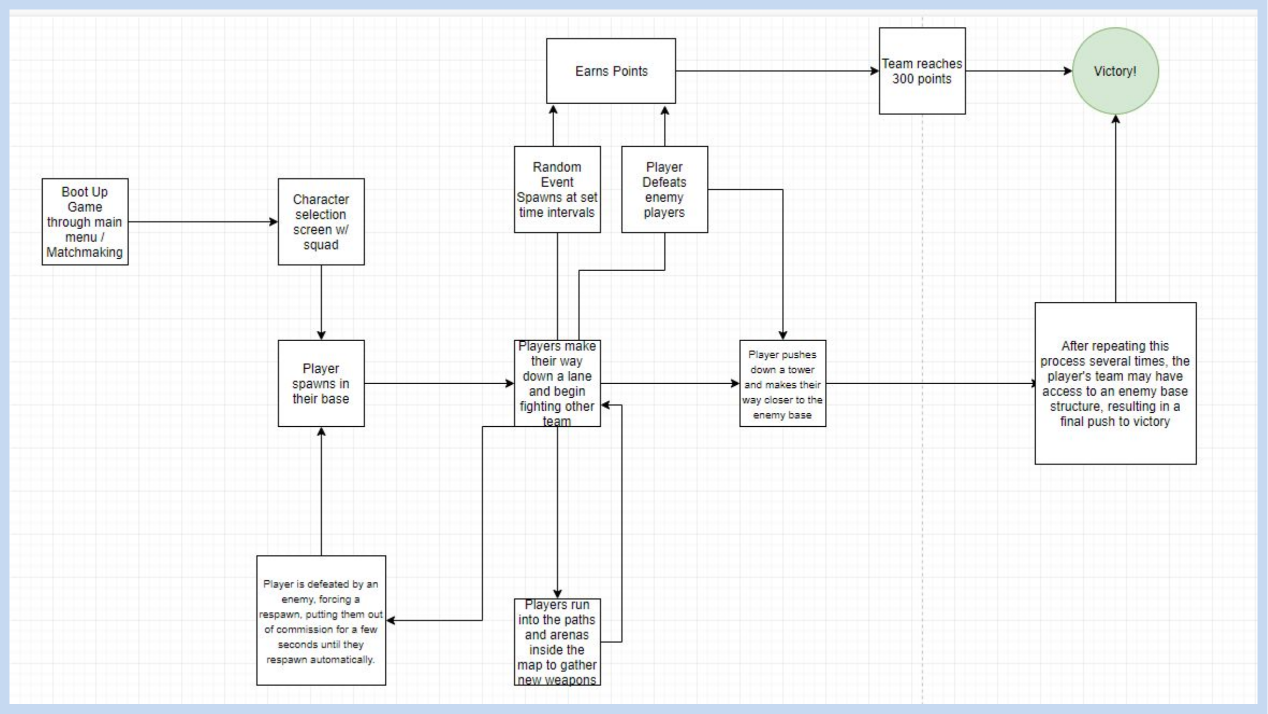
HUD

- Has a map that shows the player’s team positions in real time
- Shows player’s current weapon, ammo, health bar and special bar
- Match timer

Menus

- Uses Unity’s UI GameObjects
- Buttons are clickable, but are also selectable with a gamepad

2.3 Game Flow


Earns Points
Team reaches 300 points
Victory!
Boot Up Game through main menu / Matchmaking
Character selection screen w/ squad
Random Event Spawns at set time intervals
Player Defeats enemy players
Player spawns in their base
Players make their way down a lane and begin fighting other team
Player pushes down a tower and makes their way closer to the enemy base
After repeating this process several times, the player's team may have access to an enemy base structure, resulting in a final push to victory
Player is defeated by an enemy, forcing a respawn, putting them out of commission for a few seconds until they respawn automatically.
Players run into the paths and arenas inside the map to gather new weapons

4.0 Mechanics

4.1 Weapons

Weapons spawn randomly in the arena (at predetermined spawn points), and as drops from Gingerbread Soldiers. Initially this will be pseudo - random to insure no favouritism towards either side. This will move to a more true random algorithm towards the end of the match.

4.2 Abilities

Special abilities unique to each playable character. You need a certain amount of the Special Bar filled to use it. The Special Bar fills slowly over time, and with each kill. See *Game Design Doc Section 6.1 Characters* for more info.

4.3 XP

XP is gained during matches and is used to level up. Upon levelling up, the player is rewarded with a Loot Box. The amount of XP will be stored server side.

4.4 Currency

Regular money for buying cosmetics is earned at the end of each match. The amount gained depends on a bunch of different variables, eg. how many kills, how helpful you were to your team in a match, certain special objectives. Currency like XP will be stored server side

4.5 Loot Boxes

Loot Boxes contain random cosmetics and can be earned by levelling up or can be purchased. Loot Boxes will be mostly random with a built in counter system to guarantee occasionally something worthwhile will drop.

5.0 Graphics

Use shaders/lighting off the Unity Asset Store because the team is inexperienced and there are plenty of good ones already out there. Use Unity's deferred shading rendering path.

6.0 Artificial Intelligence

6.1 Gingerbread Soldiers (Minions)

X amount of Gingerbread Soldiers will spawn in timed intervals at their designated base.

Movement uses Unity's NavMesh navigation to move toward the enemy base until a target is within follow range. When there is a target, the waypoint will become the target's position. Once the Gingerbread Soldier is close enough to the target, it will stop and attack the target until either the target is dead, the Soldier is dead, or the target has exited attacking range.

If the target has died or exited attacking range, the Soldier will check for any other potential targets within follow range and move towards them. If there are none, it will continue to move toward the enemy base.

7.0 Physics

Physics are handled by Unity. Collision detection is handled by Unity.

7.1 Collisions

Players to arena boundaries/objects

- Players should not be able to move past/through solid environmental objects and boundaries

Players and minions to opposing team's players and minions

- Players and minions can pass through each other
- Hit detection for melee attacks between player and minion, checks for if subject is on opposing team before dealing damage

Bullets

- Bullets store what team it originated from
- They will pass through a player unless the player is from the opposing team
- If it hits an opposing player/minion, it deals damage, emits particles, and is destroyed
- Are automatically destroyed after a set amount of time. What that time is is determined by its originating weapon's firing range.
- Emits particles and is destroyed upon hitting a wall or solid environmental object

8.0 Items

Stats are just descriptions until further testing.

Weapon	Damage	Fire Rate	Bullet Speed	Bullet Range	Knockback	Bonus
Hand-Gum	High	Slow	Slow	Medium	None	-
Flossinator	Low	Fast	High	Far	A bit (enemy)	Large ammo clip
Shmak	Low	Medium	-	-	Either high knockback vs an enemy, or knock self high into the air when targeting terrain	Increases player movement speed

9.0 Game Flow

9.1 Objectives

Within one match of the standard game mode there are conditions in which either can be met in order to end the game and win the match:

1. The team's point goal is reached. These points are gained collectively by each team member normally with destroying opponents (both human players and cpu soldiers), or the opposing team's structures.
2. The opposing team's main base structure is destroyed.

Unique objectives will be added as limited-time events in regular updates.

10.0 Levels

10.1 Pseudo-random Spawning of Events and Weapons

Events and weapons will follow a pseudo random algorithm for spawning. This is to issue fairness in there placement and spawn timing. The algorithm chooses one of the many set spawn points in the arena (separate spots for weapons and events) to spawn appropriate object at.

10.2 Minion Spawning System

Minions will be spawned at the beginning of lanes on a timer system to make sure the spawns are consistent.

10.3 Minions Move Down Lanes

Minions will travel down the lanes only stopping when they meet the wave coming from the opposite base.

10.3.1 Aggro on Players

Minions will aggro (target) on players when players attack other players within a certain radius. They will also aggro on players if players are too near and there are no friendly minions around.

10.4 How Structures are Damaged

Structures can be damaged by both players and minions

10.5 Win Condition

There are 2 possible win conditions listed below. Meeting either will end the game

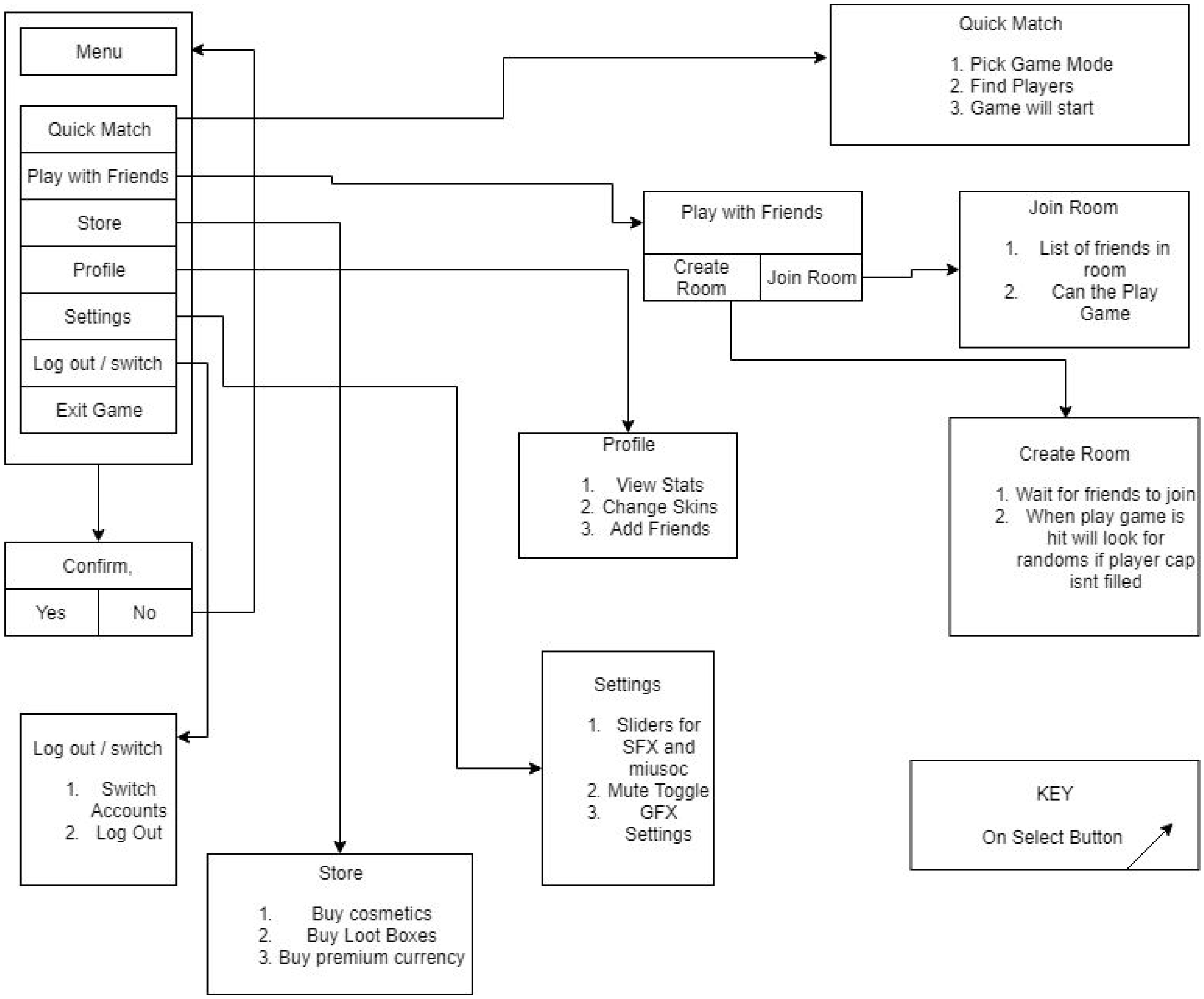
10.5.1 Points

Points are collected from winning events and player kills

10.5.2 Base Destruction

Destroying the enemy base

11.0 Interface



11.1 Menu

Menus will be created with Unity’s UI GameObjects. Menu buttons can be selected by using a mouse or the right joystick of a controller.

11.2 Camera

Each player will have a camera following behind their character in a third-person perspective as their main view. A static camera is placed above each arena. This is rendered to a texture in the HUD to be used as the arena map. This camera will see objects hidden in the main camera that are used as markers so the player can easily identify where they are and where their team is.

11.3 Controls

Control	**Controller**	**Keyboard**

	IN GAME	
Movement	Left Analog Stick	W,A,S,D
Firing	Right Trigger	Left Mouse Click
Special Ability	Left Trigger	Right Mouse Click
Interact	A button	E
	MENU	
Selecting	A button	Left Mouse Click
Scrolling menu	Left Analog Stick	Mouse over

12.0 External File Formats

Textures: .png, max size 2048 x 2048
Models: .fbx, max 20K poly count
Audio: .wav, 96kHz

13.0 Audio

Audio will be handled with Unity's Audio Module. This allows for easy implementation of 3D directional sound and mixing.

14.0 Asset List

This is a list of scripts needed for the game. For graphical and audio assets, see the Art Bible.

1. Character x 5
   a. Scripts
      i. ScriptableObject for character stats
      ii. Motor for movement
      iii. Controller for control inputs
      iv. Ability controller
   b. Components

i. Rigidbody
ii. Capsule collider

2. Weapons
   a. Scripts
      i. Scriptable object for stats
      ii. Controller for control inputs
      iii. Object spawner for bullets
3. Bullets
   a. Scripts
      i. Scriptable objects for stats
      ii. Motor
   b. Components
      i. RigidBody
      ii. Collider
4. SpawnPoint
   a. Scripts
      i. Spawner script
   b. Components
      i. N/A - empty gameobject
5. Minion
   a. Scripts
      i. Controller script
      ii. Motor
      iii. Stats Script
   b. Components
      i. Capsule Collider
      ii. RigidBody
      iii. Navmesh Agent