

"I certify that this submission is my original work and meets the Faculty's Expectations of Originality"

Daniel Rinaldi	(ID:40010464)	<u>Daniel Rinaldi</u>
Anthony Cappello	(ID:40044215)	<u>Anthony Capello</u>
David Huculak	(ID:27516495)	<u>David Huculak</u>
Rose Dufresne	(ID:40030476)	<u>Rose Dufresne</u>
Lucas Kourouklis	(ID:40063612)	<u>Lucas Kourouklis</u>
Justin Castonguay	(ID:27085605)	<u>Justin Castonguay</u>
Michael Tarantino	(ID:26570070)	<u>Michael Tarantino</u>

**Professor: Mendhurwar Kaustubha
Game Design Document
COMP 376
2019-10-23**

Game Design Document



Ctrl Alt Defeat Developers

Daniel Rinaldi
Anthony Capello
David Huculak
Rose Dufresne
Lucas Kourouklis
Justin Castonguay
Michael Tarantino

R20191128_2

EXECUTIVE SUMMARY

What Is Super Shopper: Closing Time Rush?

Super Shopper: Closing Time Rush (SSCTR) is a wacky maze-like arcade action game that has the player running around a store buying items in order to complete a shopping list whilst taking advantage of coupons and avoiding obstacles and enemies. While the primary goal is to complete your shopping list, other items and coupons appear in the level which contribute to your score. There is also a time limit for each level, making it important to avoid the many obstacles and enemies which stun or hinder the player, causing them to lose time. There is a minimum score for each level that must be reached in order to unlock more levels.

What is the setting?

There's a big sale going down at your local shop but it's almost closing time! Quick, grab your shopping list and grab as much as you can before they kick you out! Be cautious, the stores you go to are more dangerous than anticipated. Other customers are desperate to get what they want and will even go so far as to steal from your cart! The maze-like aisle layout and various obstacles of the stores themselves can prove to be troublesome as well.

Who is the target audience?

This game will appeal to gamers who like to play games in short sessions. It will also appeal to casual gamers and gamers who like classic arcade action games with frantic, addictive gameplay that focuses on the user's reflexes and quick thinking.

What are the key selling points?

Expansive labyrinthian levels coupled with a wacky, fun, and humorous settings will keep users visually interested. Players will enjoy bragging about their high score to their friends and in doing so will compete for the highest score.

TABLE OF CONTENTS

Executive Summary	2
Overview	4
Main Features	4
Compared to Other Games	4
What The Player Should See	5
Related Games	6
Overcooked	6
The Last Guy	7
Pac Man	8
Legend of Zelda: Link's Awakening (2019 Remake)	9
The Sims 4	10
Gang Beasts	11
Player Composites	12
Game World	13
Game Characters	14
Progression Graph	15
Art Direction	16
General	16
Font	16
Graphical User Interface	17
Concept Art	19
UI StoryBoard	22
Dialogue	26
Technology Plan	27
Software & Technologies	28
Assets & Packages	24
Hardware	24
Software Architecture	28
Control Flow	28
Coding Style	28
General Guidelines	28
Syntax	29
Diagrams	30
Controls	32
PC	32
Nintendo Switch	33
Level Design	34
Mechanics Analysis	36
Schedule and Related Elements	38
Budget	39
Change Logs	40

OVERVIEW

1. Main Features

SSCRT was inspired by the success of Overcooked, a popular cooking simulator developed by Ghost Town Games and published by Team17. Whereas Overcooked focuses on the hectic nature of a restaurant kitchen during a rush, SSCRT takes inspiration for the sheer insanity of a store during Black Friday. It combines a maze-like level environment, with an arcade action style of play and a timer to create a hectic, fast-paced experience.

The player must navigate the level, searching for various products on their shopping list, all the while escaping from other shoppers or distracting them by throwing away their items as bait. Hazards such as moving walls and wet floors -can block their path temporarily, or make navigating aisles more difficult.

Bonus points are awarded for checking out with additional items not present in the shopping list. Coupons appear periodically through a round, giving even more bonus points for reaching the checkout with a specific item before the coupon runs out. Additionally, a successfully redeemed coupon grants the player a special powerup to use at their leisure. The player can use these unique advantages to fight their way through the store to the product of their dreams.

The player must balance checking out with their current items or trying to fill the rest of their cart with random items for extra points.

2. Compared to Other Games

To assist the player, the level is presented to them with an overhead camera centered on the player character. Ceilings are kept see-through, similar to the overhead view for indoor locations found Legend of Zelda: Link's Awakening remake, published by Nintendo in the year 2019.

Character models are based on the simple designs found in both Overcooked and Gang Beasts, the latter being developed by Boneloaf, published by Double Fine Presents and released in the year 2017, where players can differentiate individuals by small changes in color, hair and cool hats.

Level design was heavily inspired by the wacky sensibilities of Overcooked, in that an interesting gimmick that creates variety between levels is more important than having a realistic store. For example, Overcooked contains a level where players must work in a kitchen that is split between two trucks driving full speed along a highway.

There is an ever present sense of being hunted by the other shoppers as they try to

chase you down through the aisles, creating a similar sense of anxiety which is present in both *The Last Guy*, developed and published by Sony in the year 2008, and *Pacman*, developed and published by NAMCO in the year 1980.

3. What The Player Should See

The stores in SSCRT are jam packed with aisles of products, excitable and frustrated customers trying to find that one thing that they need before they can finally go home.

Nasty spills, odd design choices and traps (for some reason) litter the stores. Whether the janitor was fired by management to cut costs or the architect of the store just really hated people is a mystery in and of itself.

The player has a list of their own they must complete, but little time to do so. The intercom blares to life with the words “Two minutes till closing” shooting from it. Better get cracking, those coupons expire today!

RELATED GAMES

Overcooked

Developer: Ghost Town Games

Publisher: Team 17

Genre: Action-Simulation

Platform: Windows, PlayStation 4, Xbox One, Nintendo Switch

Year: 2016



Overcooked, like SSCRT, is a top-down simulation game that requires the player to pick up items and bring them to a specified exit point. In Overcooked, the exit point is the service counter, whereas in Super Shopper, it is the cash register.

In both games, the art style was designed with cartoonish and stylized 3D models with fluid character movement. Additionally, Overcooked and SSCRT both have a wide array of wacky stages, with different obstacles and various difficulty levels.

Primarily, what sets SSCRT apart from Overcooked is the enemies present in each stage. Enemies will chase the player down and ram them to knock all the items out of their shopping cart. They do this to steal some of the items the player has acquired, somewhat like the rats from Overcooked that steal food, but significantly more aggressive.

SSCRT utilizes bonuses that don't exist in Overcooked, mostly for purchasing extra items not present in the shopping list or checking out specific items that have active coupons.

Fulfilling the requirements on coupons provides power-ups to the player, with effects such as: a speedboost, a shield, freezing time, an instant checkout, etc.

The Last Guy

Developer: SCE Japan Studio

Publisher: Sony Computer Entertainment

Genre: Maze

Platform: PlayStation 3

Year: 2008



The Last Guy is a maze game where the player works to save citizens from cities overrun with monsters and bring them to an evacuation zone without being caught.

The Last Guy is similar to SSCRT in that they both require the player to navigate through a maze, find some specific objective and return it safely to a specified endpoint while avoiding enemies.

The player is scored in such a way that encourages them to maximize the number of successfully saved objects within a limited amount of time. However, The Last Guy does not provide power-ups to the player. It also doesn't allow the player to distract the monsters in any way, unlike SSCRT which allows them to throw items as a distraction.

The citizen collection system also differs The Last Guy from SSCRT in that citizens will trail behind the player, allowing monsters to more easily catch the player, adding more risks to a specific player strategy. In an effort to add a similar risk factor to SSCRT, the more items collected by the player, the slower they become.

Pacman

Developer: Namco

Publisher: Namco (JP), Midway Games(NA)

Genre: Maze

Platform: Arcade, Atari

Year: 1980



Pac-Man is a classic arcade game where the player, who is controlling the titular Pac-Man, is tasked with eating all the yellow dots in a maze while being chased by ghosts.

This game inspired SSCRT's creation due to its maze game-style and the inclusion of enemies that must be avoided.

Both games have special items that help the player complete their goal or give them bonus points. For instance, the big yellow dots allow Pac-Man to eat the ghosts, somewhat like how some items in the store in Super Shopper can be used to distract other shoppers.

The cherries in Pac-Man, which adds bonus points to the player's score, are similar to the bonus points the player can receive from coupons in SSCRT.

In general, Pac-Man is a very simplistic game that SSCRT attempts to expand by adding new features.

Legend of Zelda: Link's Awakening (2019 Remake)**Developer:** Grezzo**Publisher:** Nintendo**Genre:** Adventure**Platform:** Nintendo Switch**Year:** 2019

The Legend of Zelda: Link's Awakening tasks the player with, as usual for the series, saving the land for the forces of darkness.

This has absolutely nothing to do with SSCRT.

The main common point between the two games is the placement of the player character in relation to the main camera. That is, having the player character located, generally, at the center of the screen with the camera hovering at an angle above them.

The Sims 4

Developer: Maxis

Publisher: Electronic Arts

Genre: Simulation

Platform: PC, PS4, Xbox One

Year: 2014



The Sims is a series of games where the player takes control of various “people”, telling them when to eat, clean, work, sleep, and other activities that should remain private.

The player can design and create all aspects of the sims’ living space, peering inside at them through invisible ceilings.

It’s this, combined with bright interior decor and architecture that inspired some of the design choices in SSCRT.

Gang Beasts

Developer: Boneloaf

Publisher: Double Fine Presents

Genre: Beat 'em up, Party

Platform: PC, PS4, Xbox One

Year: 2017



Gang Beasts is a game where players, usually two to four, compete against one another to be the last man standing in a variety of odd locations.

SSCRT uses this game to inspire the design of the player character and enemies: short, pudgy and mostly differentiated by bright colors and cool, or sometimes silly, hats and hair.

Furthermore, SSCRT was inspired to use simple animations for the characters to keep gameplay fast and easy to follow.

PLAYER COMPOSITES

Ricky Bobby

Age: 10 - 16

Job status: unemployed

Education: currently attending high school or lower

Ricky plays games alone or with friends, usually after school hours and at home. Likes to play games that make him laugh and that are high energy action paced. Plays on his parent's computer or his nintendo switch console. His parents are the ones who buy games for Ricky and as such have rules for when he is allowed to play games. He is only allowed to play video games after his homework and chores are completed and he must respect his 11pm curfew. Ricky makes use of his remaining time during school days and on the weekend playing video games and occasionally going to the park with friends to play outside. A typical gaming session for Ricky would be about 2 hours on a weekday and 5 hours on a weekend or holiday. Ricky likes games like Overcooked, Mario Bros., and PacMan. Ricky also really likes watching cartoons like SpongeBob Squarepants, The Fairly Odd Parents, Naruto, and Pokemon.

Barbara Roberts

Age: 38-50

Job status: School teacher

Education: has a degree in education

Barbara doesn't consider herself a gamer, but loves to play games on her mobile phone during her lunch breaks at work. She tends to buy games herself on the app store. She likes casual games that don't require too much focus or use of complex game controllers because when games have too many buttons, she tends to get overwhelmed. For this reason, she avoids playing competitive games with her husband or her kids because they are too good at these games. However, she loves to play family friendly games with the rest of her family occasionally. This, in addition to the time she plays at work, would mean she plays on average 30 minutes to 1 hours per day. She also doesn't like to play violent games because these games usually have too much going on on the screen at once and she doesn't like the use of guns in games. On the other hand, what she does like about casual games is their colourful nature and relatable themes. Barbara loves to watch sitcoms and shows on netflix such as Friends, The Office and Grey's Anatomy, which is the activity that takes up most of the free time she has. She will usually spend about 5\$ on average for new games she finds on the app store, but she could spend over 120\$ on special occasions like her kid's birthdays. The rest of the extra money she earns will go into food, clothes and other necessities for her family. Barbara would love games like Mario Party, Angry Birds and Clash of Clans.

GAME WORLD

Super Shopper is set in a wacky world where anything can happen. Unlike our world, architectural and typical company norms are non-existent. You won't be able to find a store that is "plain" in the current sense of the word. Because of this, you will find shops that have absolutely no security standards, obstacles, hazards and dangerous customers. The structure and organization of the aisles are completely impractical as well. They are maze-like and have their checkout stations at the weirdest places. People in this world believe this is the companies' ploy to trap customers inside their stores for the longest time possible as they attempt to find their way out, which would induce them to buy more. Bobbidy, a witness of the store layout complexity of this world even said this: "I got stuck in a store for months and had to rely on the in-store food to survive!".

The employees seem to follow company standards as well because they will not hesitate to make the player's shopping a challenging experience. Trolley boys will move through the entire shop and block aisles. Samplers will randomly suck you into conversations from which you can't leave without seeming impolite. Company managers will launch fire sails without warning, which will attract hordes of people to go inside the store. In a few words, it's chaos.

Additionally, every store is different. Examples of stores you will explore are: grocery stores, hardware stores and pet stores. All of them have their unique characteristics, such as unique items, maze layout and sometimes holidays.⁸ For example, there may be grocery shops decorated for new years, which distinguishes itself by having more themed decorations and more customers. The shops have no view of the outside world. This is to reflect the feeling you get when you are lost inside huge stores such as IKEA and you feel trapped. To contrast the lack of view of the outside world, these shops are colorful and eye-catching. This is also a sneaky way for companies to gain more money by luring customers to their colorful items, just like slot machines in casinos.

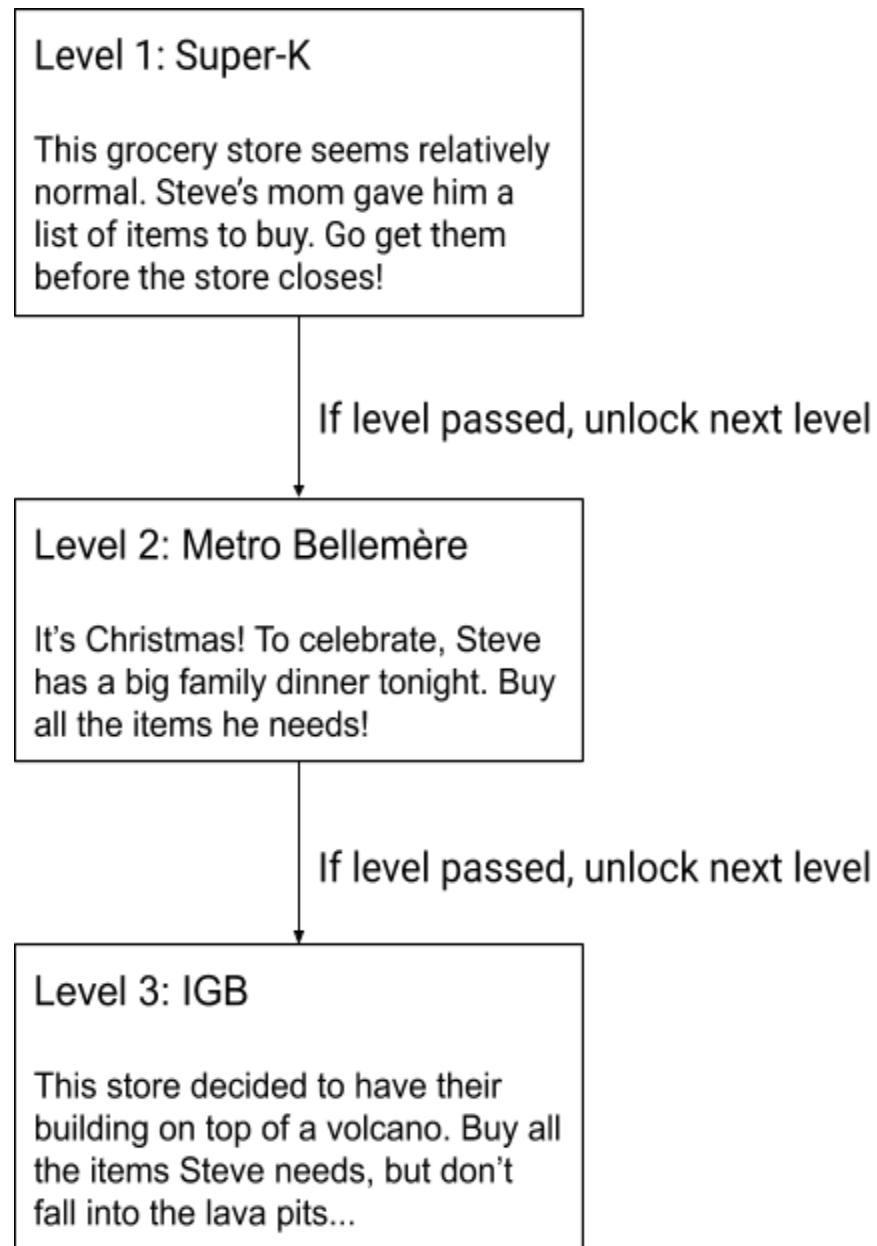
GAME CHARACTERS

Main character: Player's assume the role of 17 year old **Steve Robinson**. Steve's family is very wealthy. In consequence to their wealth, Steve's mom suffers from a condition known as monomania (a condition in which the subject is addicted to shopping). However, lately she has been unable to go shopping due to a recent head injury she got while spelunking with Bill Gates. As a compromise, she makes Steve a shopping list every week and he must get everything on them and more! However, this week, Steve lost track of time and hastily needs to go shopping for his mom before she goes crazy. He's ready to shop, but it's the end of the week and almost closing time for the local shops around his neighborhood. Can he get everything on his list in time?

Enemies:

- **Scooter Joe:** Scooter Joe frequently visits the shops that Steve goes to. He and Steve never got along and he will stop at nothing to make sure Steve is miserable. Watch out for his Scooty Puff Junior! I heard those things can reach a top speed of 10km/h , yikes!
- **Sneaky Sam:** If Sam sees you, he will try to steal half your items from you. This guy just wants to ruin your day. However, he is easily distracted. If you drop an alluring item for him to steal instead, he will leave you alone for a while.
- **Mobs:** The managers of the stores you go to sure knows how to launch fire sales. It seems like the whole city goes to their shops in hordes. These people have no care of what's in front of them and will push anyone in their way, including you.
- **Sample Stand:** Employees that work at these stands have the most boring job in these stores. For this reason, they will go the extra mile to try to sell you the item they are sampling. Steve just wants to eat their samples and move on, but has to tediously rush through long conversations with sample stand employees if he enters their field of view.
- **Trolley boys:** Trolley boys just do their job. It's not their fault if the stores they work in are made in such an impractical way. Because of the conditions they are forced to work with, they have to carry their rows of trolleys through the inside of the entire shop! This may block access to certain aisles temporarily.

PROGRESSION GRAPH



ART DIRECTION

General

Color palette:

Hex and RGB values for colors are pulled from [Material Design](#) color palette (2014).

Default clear color: #212121 

Font

The font used for text in game is [Permanent Marker](#), a free font designed by Font Diner and licensed under the [Apache License, Version 2.0](#). This font represents the look and feel of using a permanent marker which is a common writing instrument used in many stores for marking labels and prices of goods.

Characters

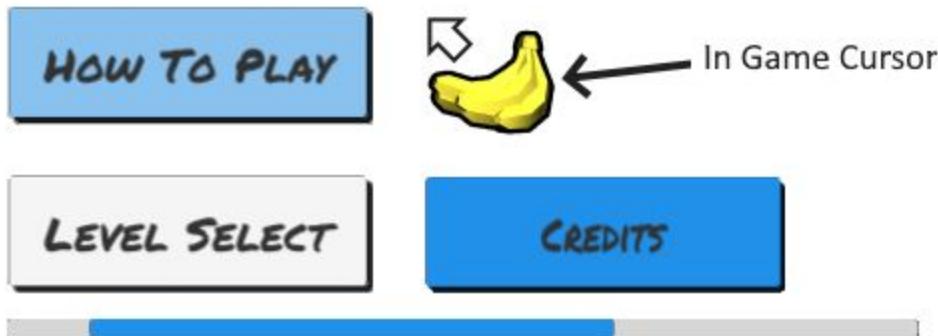
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z A B C D E
F G H I J K L M N O P Q R S T U V W X Y Z 1 2 3 4 5 6 7 8 9 0 ? ! %
) [#] { @ } / + | < - + ÷ x = > @ @ \$ € € ¥ ¢ ; : . * *

Styles	Glyph (shows upper and lowercase)
REGULAR REGULAR ITALIC BOLD BOLD ITALIC	

Graphical User Interface

System UI:

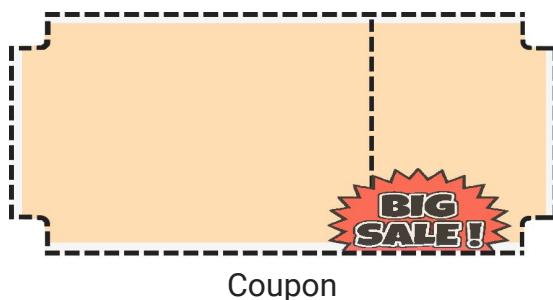
- Default Unity UI system sprites (use Permanent Marker font for any text)
- Button 'highlighted' and 'selected' states use #90CAF9 
- Button 'pressed' state use #2196F3 
- Normal state uses #F5F5F5 
- Scroll bars use the same coloring scheme as buttons



Gameplay UI:



Item sprites (example banana and beer) make use of outline around sprite to make silhouette more apparent



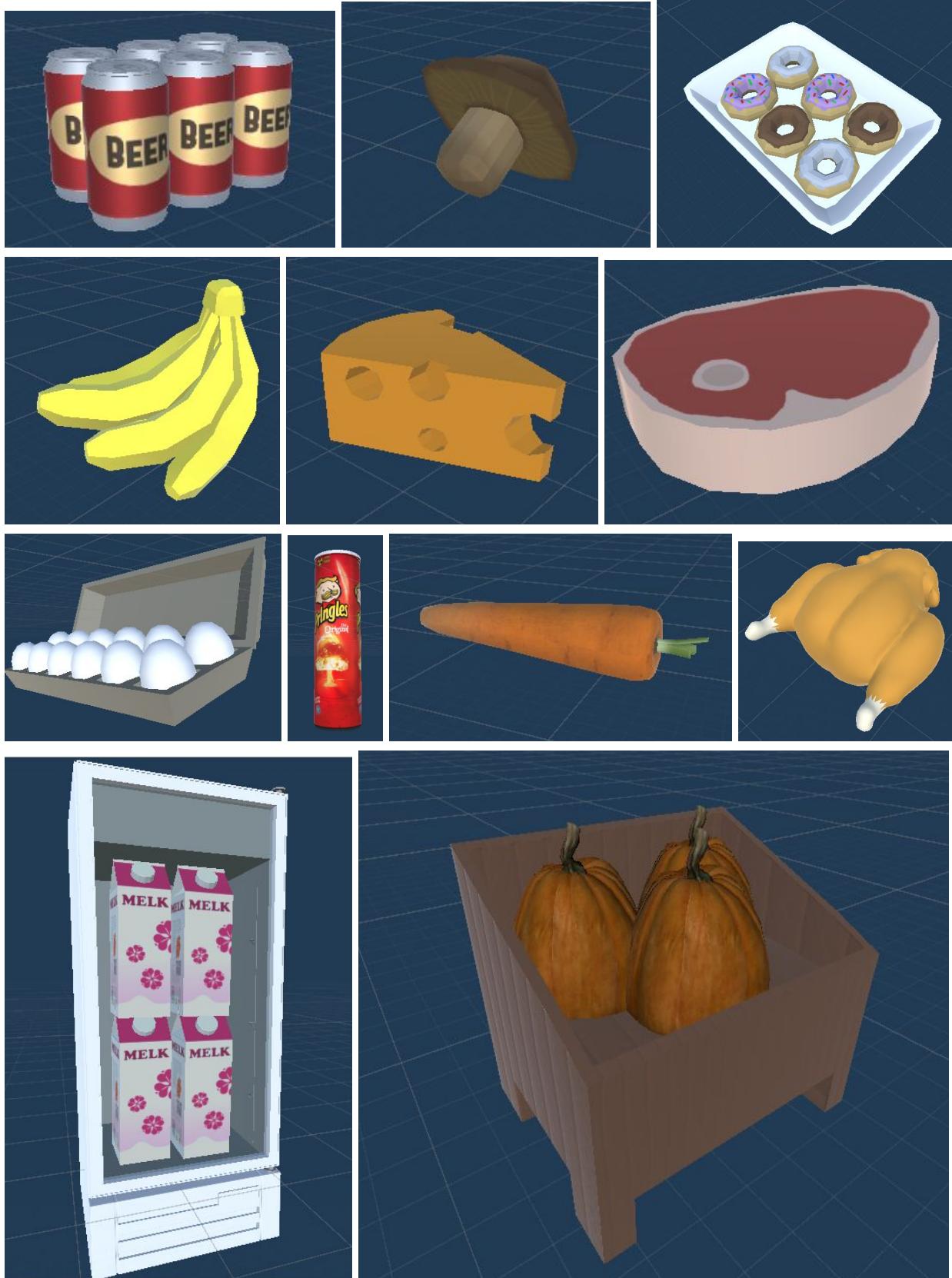


Powerups

Concept Art



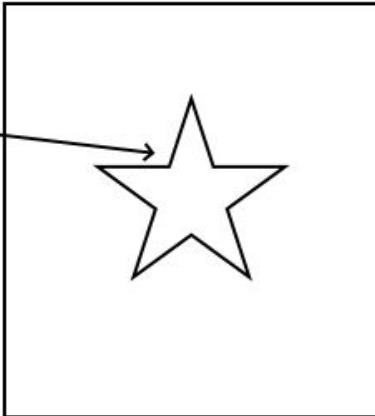




UI STORYBOARD

UI : Splash Scene

Team Logo expands slowly and then fades to next scene (Welcome)



First scene the player sees after launching the app

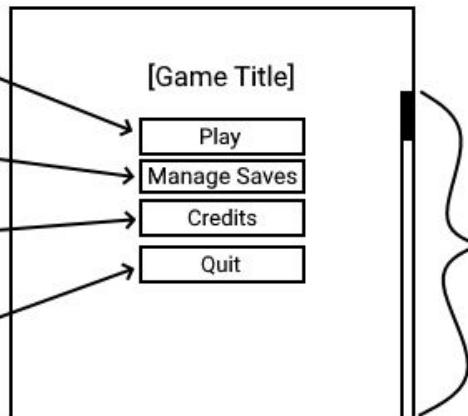
UI : Welcome Scene

Play button : goes to level select scene

Manage Saves Btn : Opens Saves Menu

Credits btn : goes to credits scene

Quit btn : shuts down the application

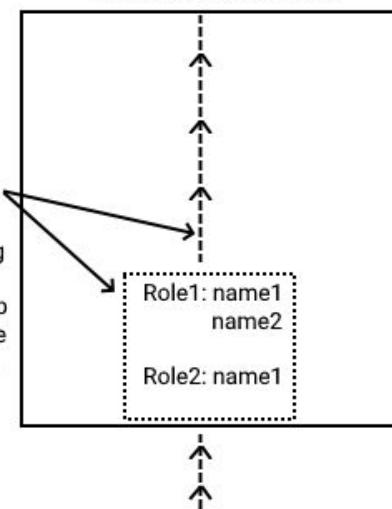


This scene contains the game's title and 3 buttons which gives user choice of what happens next

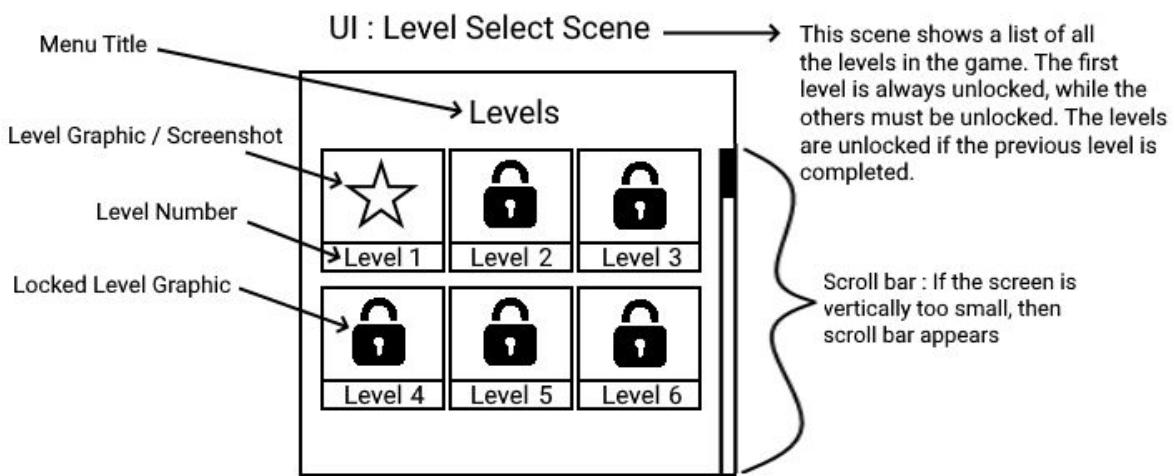
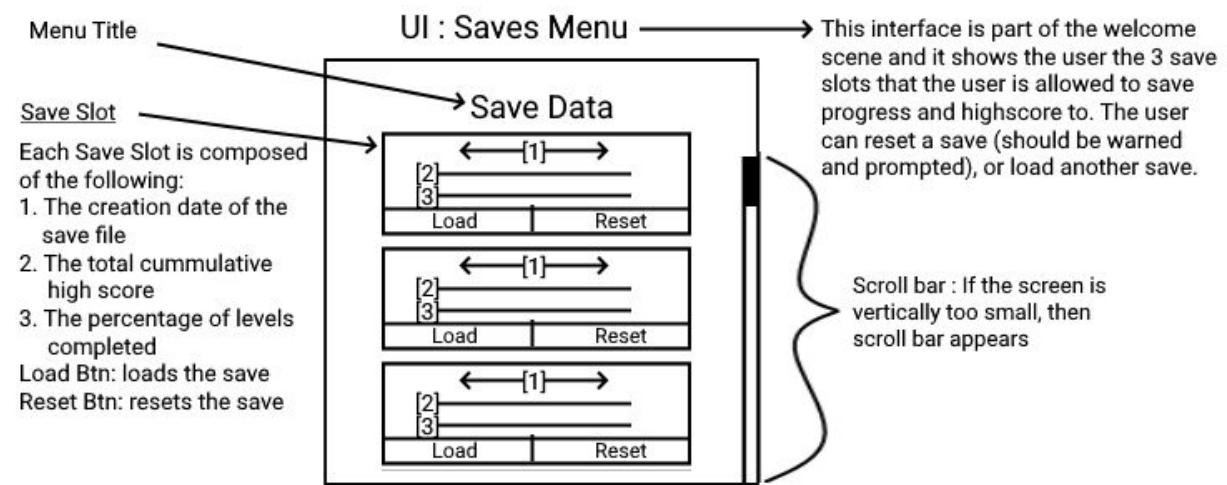
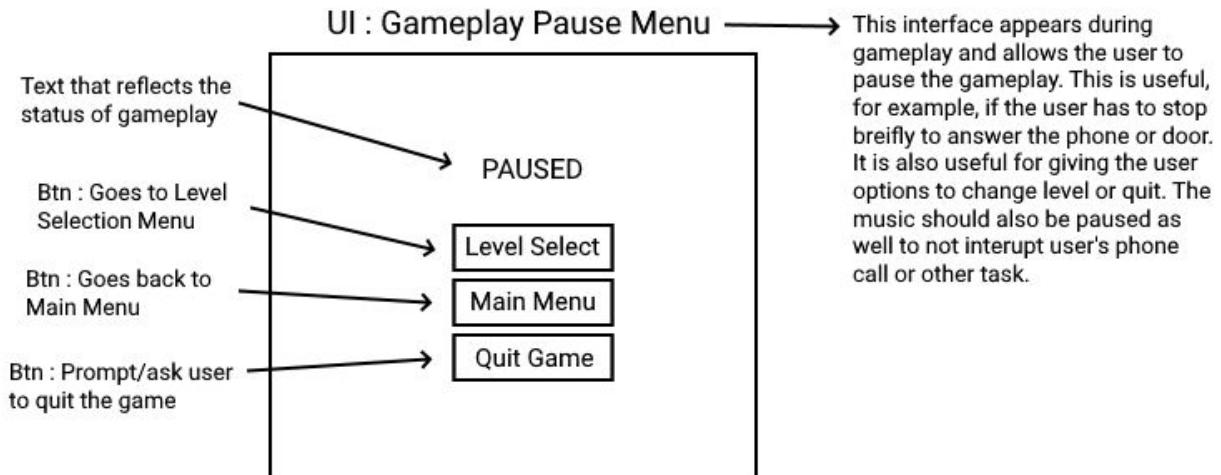
Scroll bar : If the screen is vertically too small, then scroll bar appears

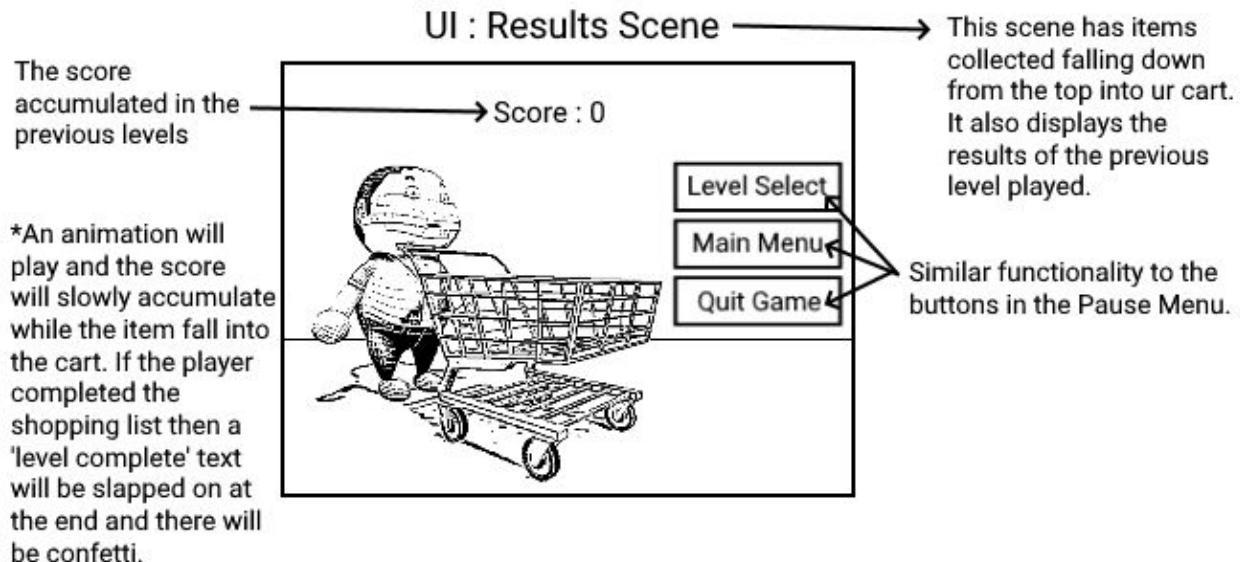
UI : Credits Scene

List of credits, starts off screen at the bottom and slowly moves up to the top. User is able to increase speed by holding down left mouse btn. Once credits reach the top and are fully hidden above the top edge of the screen, fade to Welcome scene.

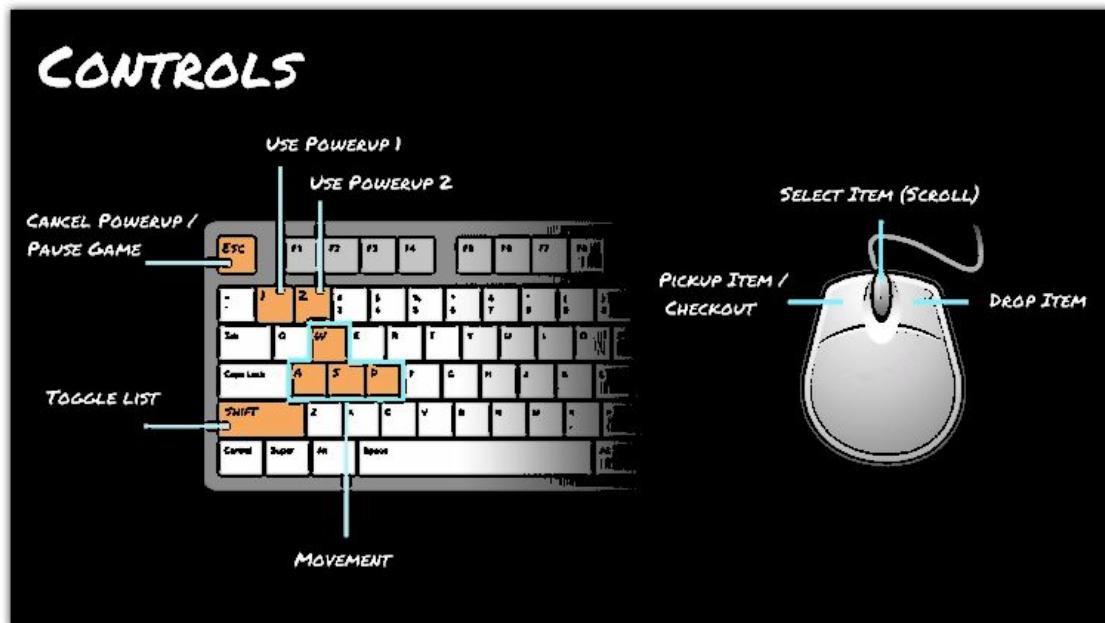


This scene gives proper accreditation to all contributors. It also has falling items in the background



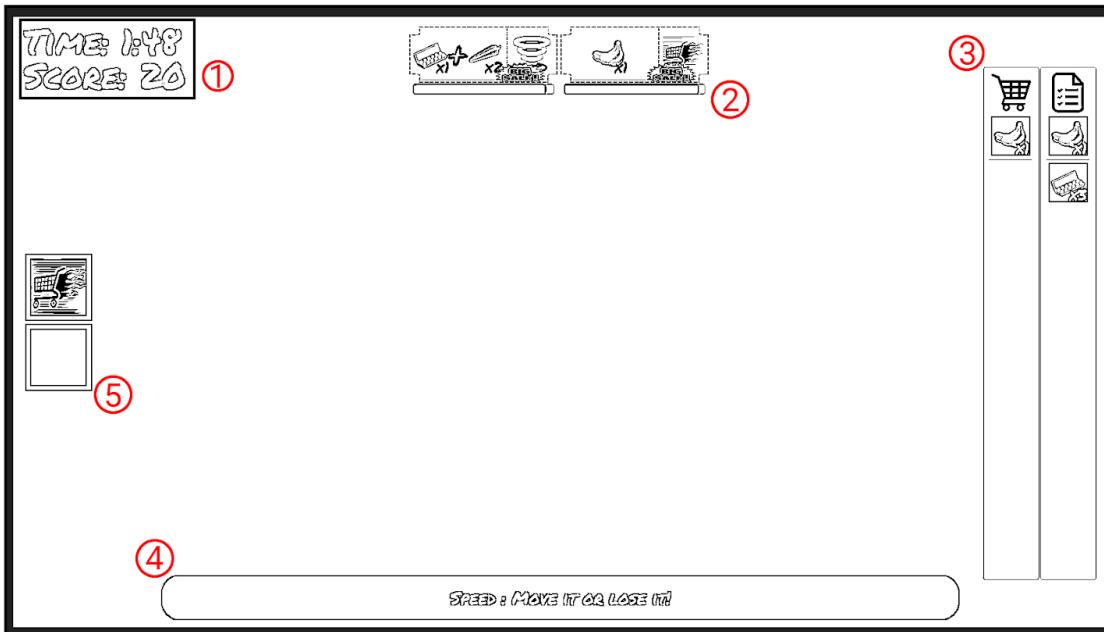


HOW TO PLAY



This is an extra menu that has been added after release date that player can access from the main menu and the pause menu via button click.

UI : Gameplay HUD



This interface shows various info to the user including the player's inventory, shopping list, score, remaining time, coupons, and powerups.

1. Time and Score:
Shows the remaining time and the player's current score.
2. Coupons:
Coupons will appear and show their required items on the left hand side (up to 2 items) and a power up on the right.
3. Inventory and List:
The inventory (left side) shows all the items the player has picked up until they check them out. The list (right side) is not always visible. If the player chooses to see the list it will appear, and present all the items remaining on the list for the player to purchase.
4. Powerup toast:
When a powerup is obtained from a coupon, a toast will show up to provide insight into what the powerup does.
5. Powerup:
Each slot shows up as the player acquires power ups.

DIALOGUE

The only dialogues in the game are for the sample stand character, she has 4 lines of dialogue which are:

"Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. " ,

"Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor." ,

"in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa." ,

" qui officia deserunt mollit anim id est laborum."

TECHNOLOGY PLAN

Software & Technologies:

- [Unity](#) - Cross-platform game engine.
- [Visual Studio](#) - IDE developed by Microsoft, used to edit cs and shader files
- [GitLab](#) - A web-based DevOps lifecycle tool that provides a Git-repository manager, a wiki, and issue-tracking.
 - The main reason for using this instead of Github is that they allow private repositories with up to 10 developers whereas Github's limit is less than 7.
- [GitKraken](#), [SourceTree](#), [GitAhead](#) - GUI git clients used by our developers.
- [C#](#) - General-purpose, multi-paradigm programming language.
- [ShaderLab](#) - All shader files in Unity are written in a language called "ShaderLab".
- [GoogleDrive](#) - Cloud hosted storage.
 - Used to store text documents(GoogleDocs), work-in-progress assets, graphic design files, images, notes, and anything else that should not be included in the project's git repository but that needs to be shared.
- [GoogleDocs](#) - A free web-based tool in which documents can be created, edited stored (GoogleDrive), and shared online.
- [Blender](#) - free and open source 3D creation suite.
- [AutoDesk 3dsMax](#) - Paid 3D creation suite used and already owned by one of the developers.
- [Krita](#), [Paint.net](#) - Free raster graphics editors. Used for graphic design.
- [LogicPro](#) - digital audio workstation and MIDI sequencer software application for the macOS platform.

Assets & Packages:

- [Unity Asset Store](#):
 - [Adventure Game Assets](#) - Free and Open Source Assets created by the Unity team for demonstration. Some of the assets are related to our game's theme and using them will save time on asset creation.
 - [Legacy Image Effects](#) - Package that contains Image Effect shaders that was previously shipped by default with Unity. Used mostly for camera effects such as vignette, blur and chromatic aberration.

Hardware:

- Workstations with keyboard + mouse peripherals and speakers.

Workstations	1	2	3	4	5	6	7
CPU	i7-3770k	i7-6700k	i7-4790k	i7-3770k	i3-6100	i5-6600k	AMD Ryzen 5 2600
GPU	Geforce GTX 1080	Geforce GTX 1080	Geforce GTX 980	-	Geforce GTX 780	Radeon RX 480	GeForce GTX 1660 Ti
RAM	8GB DDR4	16GB DDR4	32GB DDR3	8GB	16GB DDR4	16.0GB	16GB DDR4
Storage	256GB SSD	500GB SSD	250GB SSD	500GB	2TB	120GB SSD 1TB HDD	240GB SSD 1TB HDD
Display	1920x1080	1920x1080	2560x1440	1280x1024	2560x1440	1920x1080	1920x1080

SOFTWARE ARCHITECTURE

Target Unity Editor Version: 2019.2.5f1

Scripting Language: C# for most purposes and ShaderLab for shaders

Control Flow

The game must always start with **Main.unity** (main scene). This is because there are game objects here that need to perform initial setup tasks and also there are objects that remain persistent across scenes (do not get destroyed on scene load). For this scene, any setup that needs to be done before next scene load should be done in scripts' **Awake()** method to ensure it runs before the GameManager's **Start()** method which will load the first scene.

The **GameManager.cs** file contains the logic for loading the first viewable scene in its **Start()** method and, during normal play, that scene is always **Splash.unity**. For testing and development purposes, an editor extension has been created called **AutoLoadMainScene.cs**. Developers may now choose to automatically load the main scene while playing in the editor by going to **File > AutoLoadMainScene > Load Main Scene on Play** and selecting the main scene to load. The default functionality of pressing the play button in the editor will now be overwritten and instead will load the main scene first and then the previously open scene will load afterwards. This is useful and needed because if the developer is working on a scene that requires an object of the main scene to be initialized then they do not have to go through multiple scenes and menu to get to the scene they are working on just to test it.

During normal play, the first scene that is visible to the user is **Splash.unity**. This scene just shows the development team's logo and then transitions to **Welcome.unity**. The Welcome scene just opens the main menu and allows the user to begin playing. From here, the normal control flow is applied to all monobehaviours in the scene. For more information, consult Unity's official manual: [script lifecycle flowchart](#).

Coding Style

General Guidelines:

- choose easily readable identifier names.
 - Ex: **vertical_alignment** is better than **alignment_vertical**
- favor readability over brevity.
 - Ex: **user_group** is better than **usr_grp**
 - Exceptions: **user_id** (id is a well known abbreviation)
- DO NOT use **var** unless the type is apparent from right side of the assignment.
 - Ex: **Good Use: for(var i = 0; x < 10; i++)** **Bad Use: var person = GetCulprit();**
- Prefix interfaces with the letter **I**.
 - Ex: **IObservable**
- AVOID using [Egyptian style brackets](#)
 - Brackets always have their own line or a pair are on the same line.
- TRY to name delegates with "EventHandler" or "Callback" suffix.

Syntax:

```

using UnityEngine;      // <--|
                      // |--- group using statements together by the first
using System.Collections; // <--|   word before the dot(.) in the namespace names

/*
 * Multiline comments should use "/**/" although multiple uses of "//" are
 * permitted with reasonable justification.
*/
namespace ExampleNamespace // namespaces use PascalCase
{
    // classes use PascalCase
    class ExampleClass : MonoBehaviour, IObserver
    {
        public const int EXAMPLE_CONSTANT = 7; // const or readonly use SNAKE_CAPS

        // enums use PascalCase and the enumerations should be in SNAKE_CAPS
        public enum ExampleEnum { NONE, EXAMPLE, MULTI_WORD_EXAMPLE }

        // member variables that are public/exposed in the inspector use camelCase
        [SerializeField] string exposedFieldInInspector = "";

        // Unity exposes public members in inspector by default. If this is not
        // desired use [HideInInspector], otherwise still use [SerializeField]
        [HideInInspector] public int hidden_public_field;
        [SerializeField] public string exposedPublicField = "";

        float example_count; // private/protected member variables use snake_case

        // C# shorthand syntax for getter/setter should use PascalCase
        public float ExampleSpeed { get; set; }

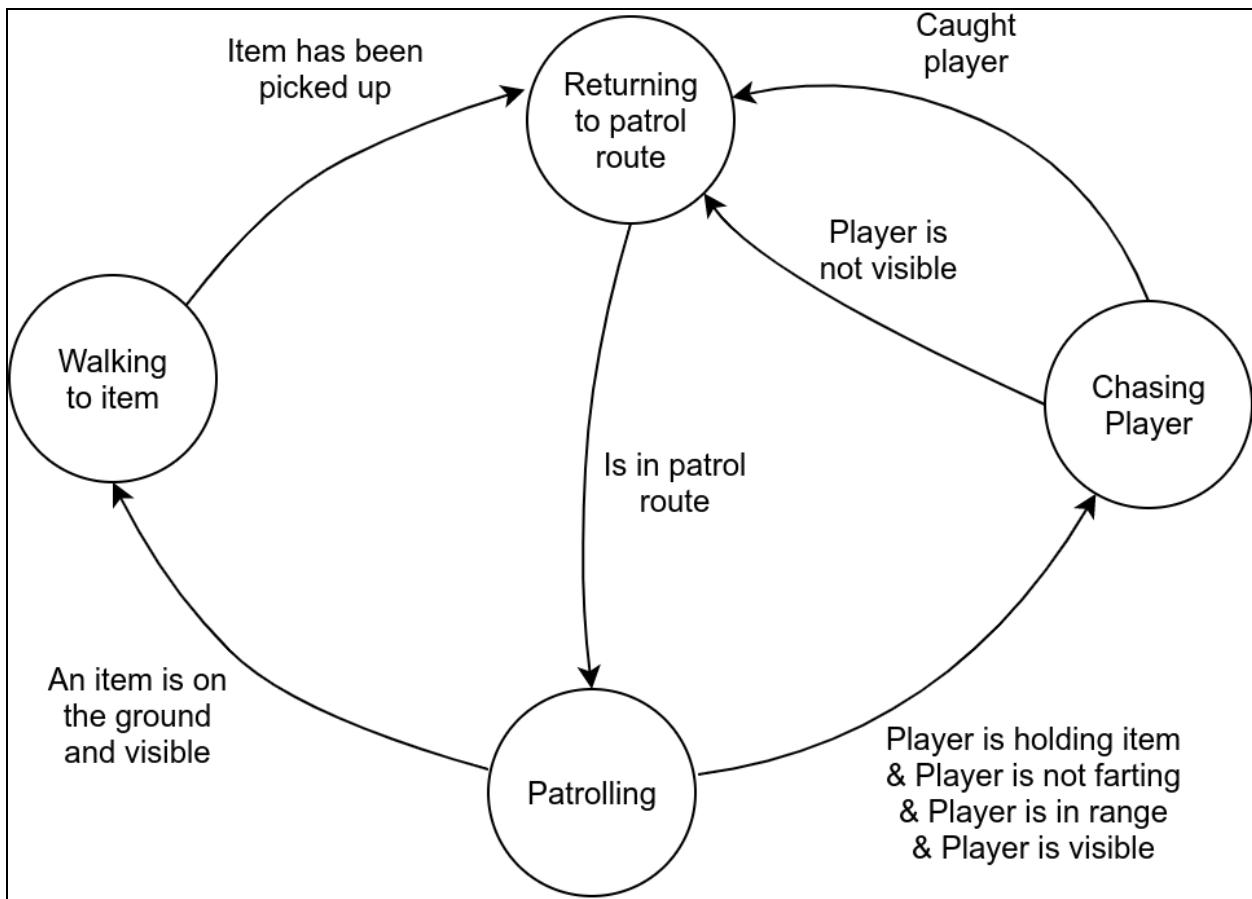
        // method names use PascalCase but parameters use snake_case
        void MethodName(string example_name)
        {
            int example_count = 0; // local variables use snake_case

            // only use "this" where necessary and when referencing a
            // member that is not visibly present in the same file.
            example_count = this.GetComponent<Transform>().ChildCount();
        }
    }
}

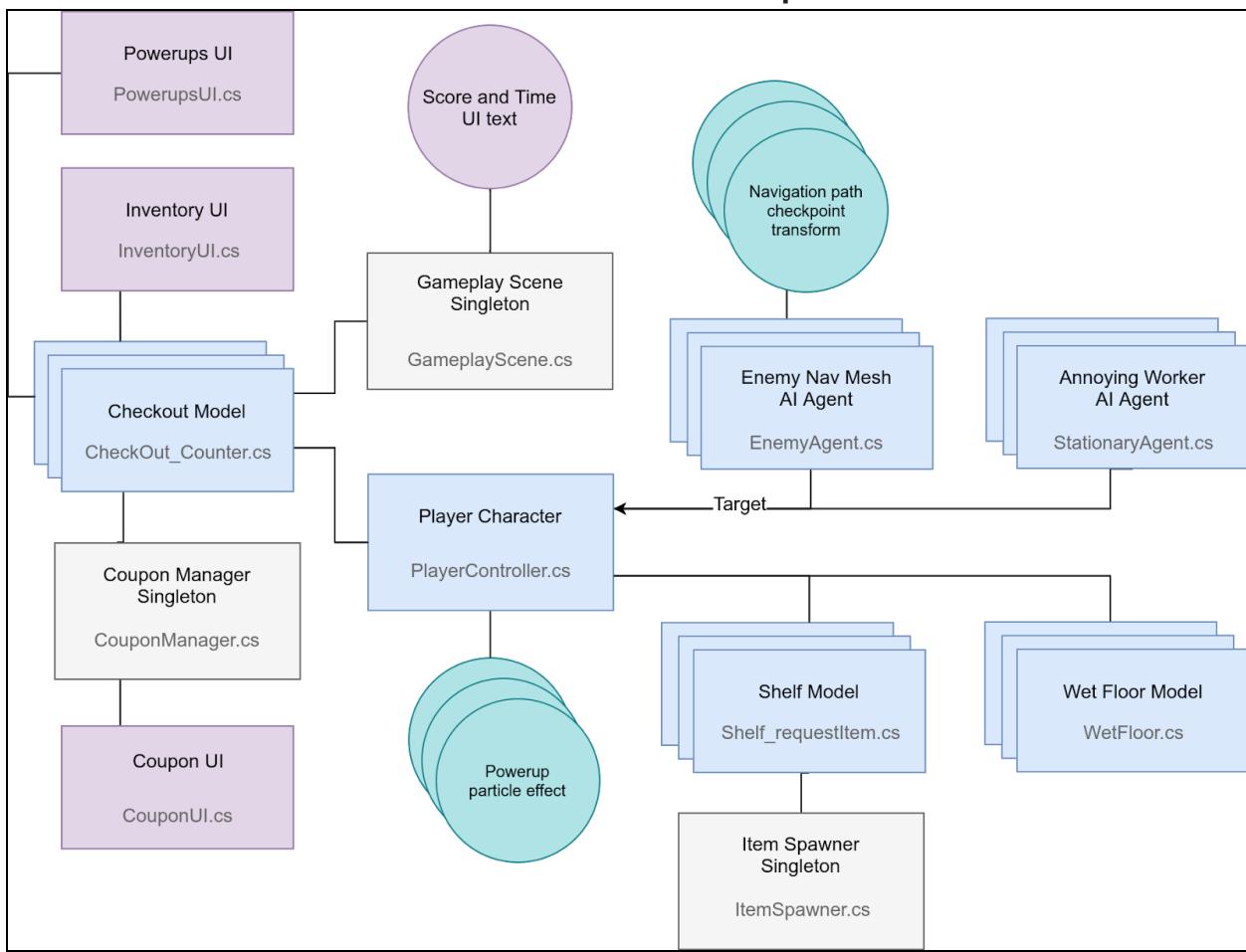
```

The above style/syntax rules are meant to be a guideline and are not enforced. But it's nice if everyone uses a similar syntax. ☺

AI State Machine Diagram



C# Class References Graph



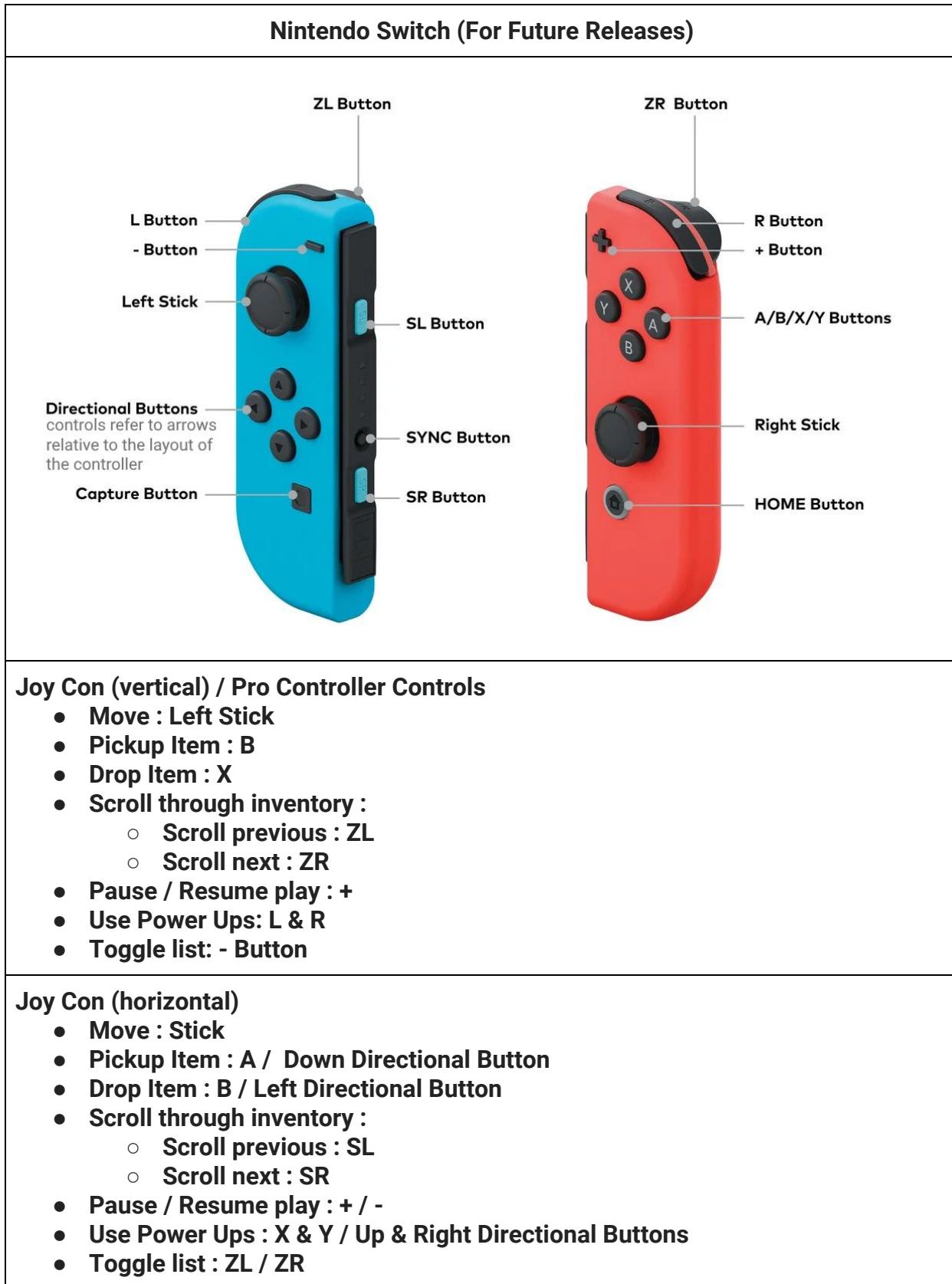
CONTROLS

PC

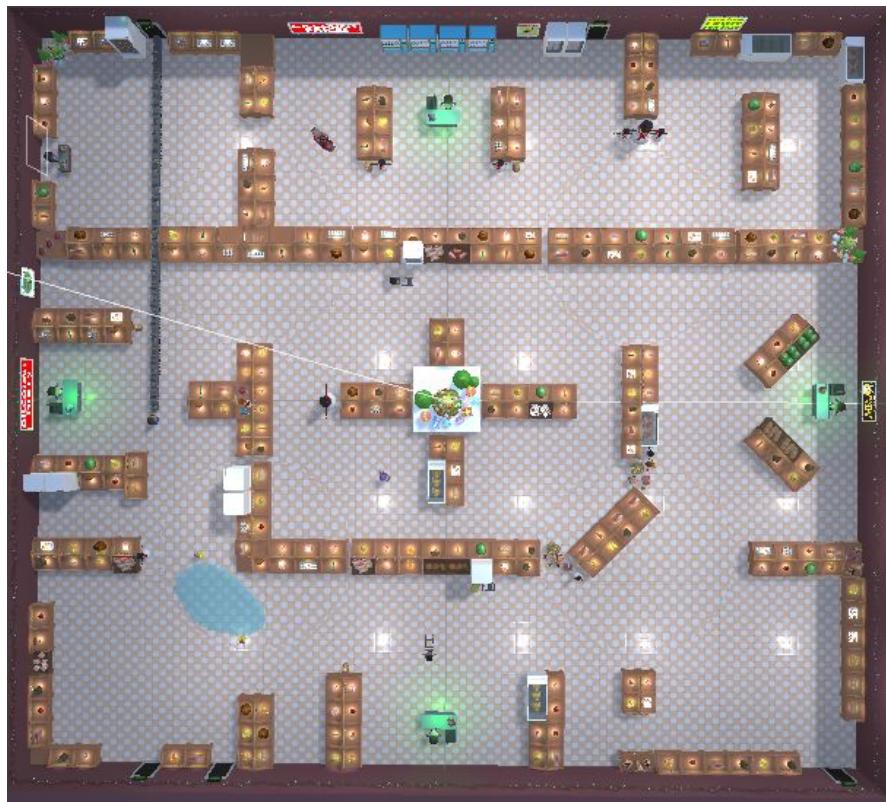
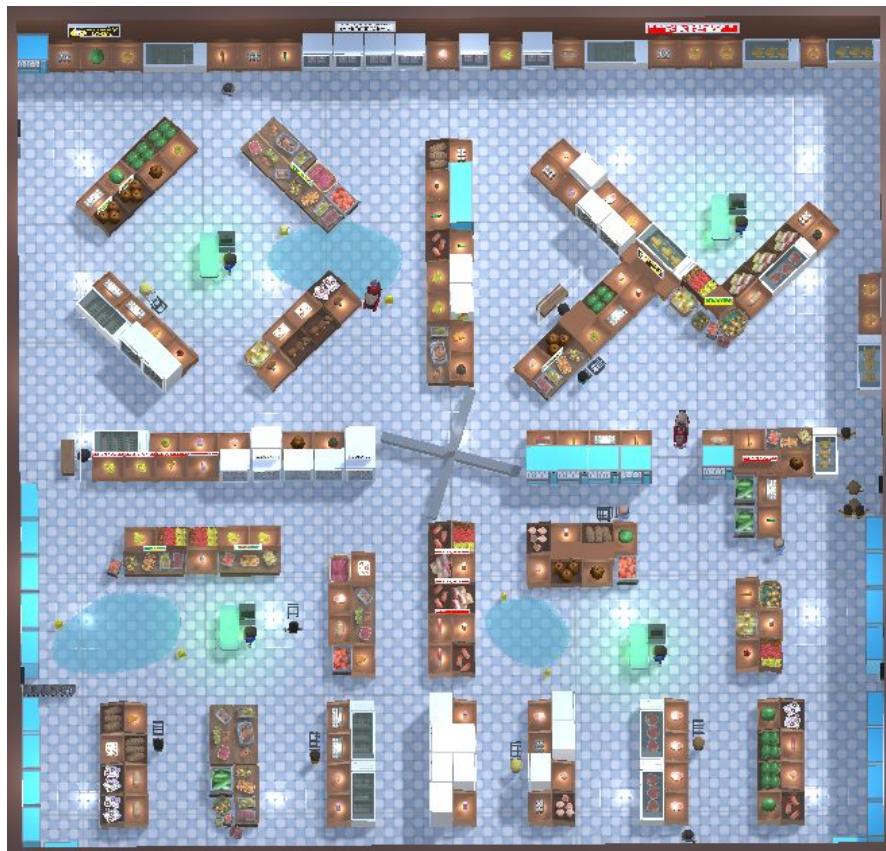


Mouse & Keyboard

- **Move : WASD**
- **Pickup Item : Left Mouse Button**
- **Drop Item : Right Mouse Button**
- **Scroll through inventory :**
 - **Scroll previous : Scroll wheel down**
 - **Scroll next : Scroll wheel up**
- **Pause / Resume play : esc key**
- **Use Power Ups: 1 & 2**
- **Toggle list : Shift**



LEVEL DESIGN





MECHANICS ANALYSIS

Grab items: Grab items from shelves in the store that you need to complete your shopping list. This is similar to looting mechanics present in games like *Thief* where the player gets close to the item in question and presses a button to take it (the difference is that our player *is* paying for their items). The animation for picking up items is also very similar to the picking up mechanism seen in *Animal Crossing*. This is the core mechanic of the game and will drive the player to complete their objectives for each play session.



Checkout: The player can go to the checkout to purchase items in their current inventory and cross them off their shopping list. The player will be able to make multiple trips to the various checkouts throughout the store. However, any checkout used will become inactive until all checkouts have been visited, at which point they will all reactivate.

Dropping Items: Dropped items will distract other shoppers to make a path for yourself. Similar to the mechanic in *Pokemon Snap* where the player can throw an apple on the ground to attract a Pokemon, this will have the same effect on other NPC shoppers in the store. Dropping items also lifts weight off of your shopping cart which can be useful if you are in sticky situations.



Dropping items will help the player navigate the store/play area and to get by other shoppers without being interfered with.

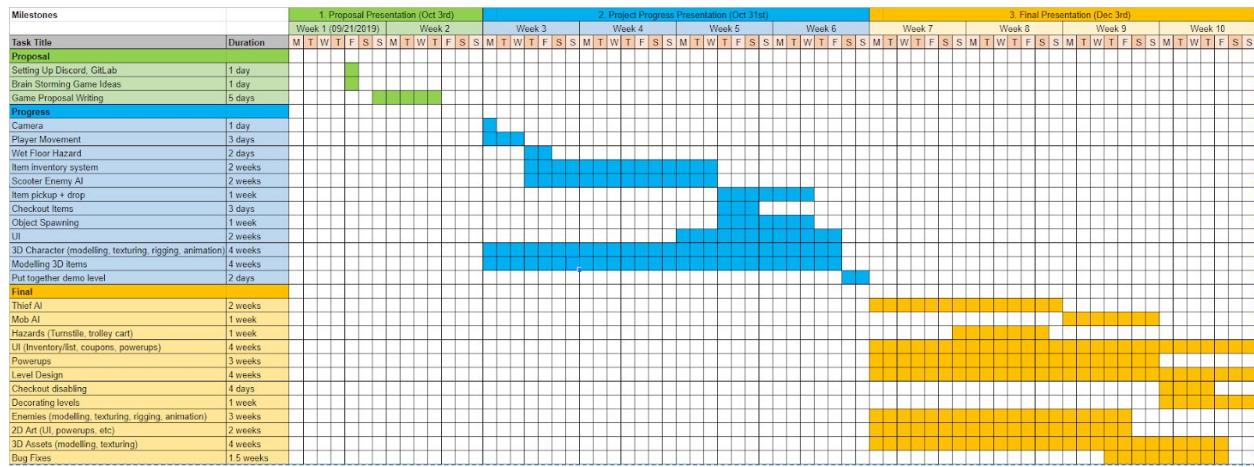
Time Pressure: You need to complete your shopping list before the time runs out. The more items you buy, the better your score.

Power-Ups: If you manage to make use of coupons that show up on screen, not only do you get bonus points but some will reward power ups.

Hazard Avoidance: There are pitfalls, traps, enraged customers, thieves, and many more hazards to avoid in each level. Quick reflexes and forward thinking are necessary.

Memory: You can only see part of the store. You will need to move around and remember where things, traps, and enemies are.

SCHEDULE AND RELATED ELEMENTS



To see a larger chart please follow this link:

<https://docs.google.com/spreadsheets/d/1jnjBC--v1Pz965wAd78yMnLxcQtVGcxDOzjrmif5bGY/edit?usp=sharing>

BUDGET

We did not spend any money except for 6.00\$ on donuts for one of the meetings,



as well as 2.00\$ on bananas for celebrating the completion of the game.



CHANGE LOGS

2019-10-31 :

- Changed UI Layout (reduced amount of screen real estate used)
- Changed the model for shelves. After testing the proposal version of the shelves we noticed that some items were not very visible to the player, so we changed the model to a design that clearly shows the items on top.

2019-11-28 :

- Changed the way coupons and powerups work. PowerUps are now the reward for completing certain coupons.
- Minor adjustment to coding style - syntax
- Modified Software Architecture (Addition of AutoLoadMainScene)
- Cosmetic changes to the GDD

2019-12-05 :

- Gave Scooter Joe a cooldown for balance
- Made Sam more easily distracted
- Reduced polygon count on some models which were causing performance issues
- Added banana cursor
- Other visual changes
- Added diagrams: AI State Machine & C# Class References Graph
- Added controls menu to game