

Artefacts . . . . .	2
Requirements . . . . .	3
Project brief and proposal . . . . .	4
Functional/Non-Functional Requirements . . . . .	6
Personas and Motivational Modelling . . . . .	8
Persona for the neuroatypical child role - Timmy, an uninterested child . . . . .	9
Persona for the mature-aged user role - Tim, the adult . . . . .	10
Persona for the Parent/Guardian Role - John, a concerned father . . . . .	12
Persona for Masters Student Role - Mary, the Game Integrator . . . . .	13
Persona for Client Role - The Client . . . . .	14
User Stories . . . . .	15
Design . . . . .	18
Sample scenarios . . . . .	19
First Draft visuals . . . . .	20
Game Design Document . . . . .	24
AI Feedback on our project proposal . . . . .	27
Research - Initial Game Brainstorming . . . . .	29
Architectural Design . . . . .	30
Logical view . . . . .	31
Process view . . . . .	32
Development view . . . . .	33
Physical view . . . . .	35
Scenario View . . . . .	36
Coding . . . . .	38
Coding Standards . . . . .	39
Development and Code Reviews . . . . .	40
Code Reviews . . . . .	41
References . . . . .	42
Testing . . . . .	43
Client Feedbacks . . . . .	44
[230914] merging emotions panel into main . . . . .	50
[231004] temporary_branch creation . . . . .	52
[231005] temporary_branch testing . . . . .	55
[231012] update_art testing . . . . .	57
[231014] temporary_branch . . . . .	60
[231015] fixed_typing testing . . . . .	62
[231015] fixed_typing testing 2 . . . . .	65
[231018] main merge . . . . .	67
[231018] database-resilience testing . . . . .	69
[231019] final testing in Unity application . . . . .	72
[231024] finalising game . . . . .	75
Testing Template . . . . .	79
Deployment . . . . .	82
Deployment plan . . . . .	83
Handover . . . . .	91

## Artefacts

### Requirements

 Requirements

### Frontend design

 Design

### Architectural design

 Architectural Design

### Coding

 Coding

### Testing

 Testing

### Deployment

 Deployment

## Requirements

### Project Brief and Proposal

 Project brief and proposal

### Functional/Non-Functional Requirements

 Functional/Non-Functional Requirements

### Personas and Motivational Model

 Personas and Motivational Modelling

### User Stories

 User Stories

# Project brief and proposal

This page details the provided project brief and our general project proposal.

## Project brief

### 1. Project Overview

Our task is to design and develop engaging, intuitive games to be incorporated into an application aimed at teaching emotional granularity. This project is a collaborative effort and requires close coordination with the team developing the main app.

### 2. Objectives

Develop an engaging game that aid in teaching emotional granularity. Ensure that the games are accessible and enjoyable for moderate to severe autistic individuals. Facilitate regular communication with the main app development team to ensure technical compatibility and seamless integration.

### 3. Target Users

The games will be designed for a diverse audience that includes moderate to severe autistic individuals.

### 4. Team Composition and Project Duration

The project will be completed within a university semester. Each game will be developed by a team of 5 computing students.

### 5. Deliverables

A fully functional, standalone game that can be integrated into the main application.

Documentation detailing the game design process, development stages, testing results, and integration guidelines.

### 6. Design and Functionality Guidelines

Games should be simple and intuitive to ensure usability and engagement.

Consider using inclusive design principles to make games accessible and enjoyable for the target audience.

Regularly consult with the main app development team to stay informed about technical constraints.

### 7. Evaluation Criteria

Projects will be evaluated based on:

- The level of engagement and intuitiveness of the game.
- How effectively the game teaches aspects of emotional granularity.
- Compatibility with the main application and ease of integration.
- Quality and comprehensiveness of the accompanying documentation.

## Our project proposal

We aim to a “papa’s pizzeria” style game which involves the player having to serve customers the right order within a specific time limit. Instead of the traditional game where customers order food, the customers will tell the player about a situation they experienced (possibly matched with a facial expression - but leave this to last, and maybe match the emotion with a facial expression as well), and the player will have to decide which emotion is closest to what they must be feeling. If they get it correct, they receive a certain number of hearts. Here are some more specifics:

- The customers will have multiple ways of hinting at the emotion they're feeling
  - They might describe a situation
    - using a text speech bubble
    - through audio
    - through a sign language video
  - They might show a facial expression
- There will be three levels, and it will become progressively more challenging for the player to choose the emotion to match the customer
- Multiple customers will come in each level, and have a time limit (maybe - add at the end) to be served by
- Passing a level will give the player a certain number of hearts
- See example scenarios in Design/Sample Scenarios:  [Design](#)

# Functional/Non-Functional Requirements

## 1. Functional

Business	User	System
<ul style="list-style-type: none"><li>• a game that aligns with the theory of constructed emotions</li><li>• facilitates the development of emotional granularity children with moderate-severe autism</li><li>• simple and to be deployable by week 11 of semester</li><li>• must be integrated into the wider 'application hubs' for emotional granularity developed by the masters' students we are collaborating with</li><li>• should utilise the Emotion-Concept Spreadsheet that is shared between the Masters and Undergraduate students: <a href="#">Copy of Emotion-concept spreadsheet</a></li><li>• adheres to general stylistic standards for the UI set out by masters' students</li><li>• short in length (approximately 3 minutes total: 1 minute per level)</li></ul>	<ul style="list-style-type: none"><li>• user-friendly</li><li>• simple and accessible interface</li><li>• user progress measured by a currency (hearts) system</li><li>• straightforward (non-frustrating) and engaging gameplay</li></ul>	<ul style="list-style-type: none"><li>• made for IOS (more specifically, iPhone) devices-- can be compiled to a .ipa</li><li>• landscape orientation</li><li>• communicates with local database as required</li><li>• utilizes local storage as opposed to global storage</li></ul>

## 2. Quality (Non-Functional)

<b>Usability</b>	<ul style="list-style-type: none"><li>• interface supports a safe environment for children with moderate to severe user-friendly</li><li>• simple and accessible interface</li><li>• user progress measured by a currency (hearts) system</li><li>• straightforward (non-frustrating) and engaging gameplay</li><li>◦ font size and style: minimum font size of 10</li><li>◦ colour scheme</li><li>◦ art design</li><li>◦ simple game instructions</li><li>◦ button size and colour</li></ul>
<b>Reliability</b>	have a low frame rate per second (FPS), probably around 30, to drain less battery
<b>Performance</b>	optimised game performance <ul style="list-style-type: none"><li>• audio and video should have no problems</li></ul>

	<ul style="list-style-type: none"><li>• game shouldn't freeze</li></ul>
<b>Capacity</b>	local IOS storage
<b>Readability</b>	minimum font of 10

# Personas and Motivational Modelling

## Personas

Below are some personas that help elaborate upon our existing user story roles to highlight the key criteria that our application will need to be meeting.

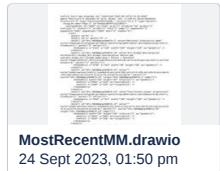
- [Persona for the neuroatypical child role - Timmy, an uninterested child](#)
- [Persona for the mature-aged user role - Tim, the adult](#)
- [Persona for the Parent/Guardian Role - John, a concerned father](#) (Parent/Guardian Role)
- [Persona for Masters Student Role - Mary, the Game Integrator](#)
- [Persona for Client Role - The Client](#)

## Motivational Model

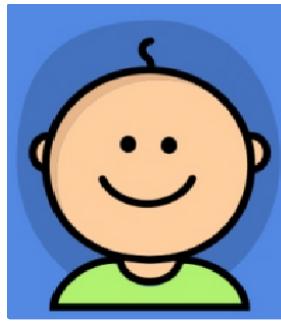
A high-level overview of the application and what it should do, be and make stakeholders feel is below, which has been developed in consultation with the user stories to give a comprehensive overview of what our application will be expected to be able to do, broadly speaking.



The model was made on draw.io-- if you'd like to edit the model at any time feel free to use the below XML file:



# Persona for the neuroatypical child role - Timmy, an uninterested child



"Can I go play something  
else now?"

## Information

Age	10
Gender	Male
Education level	Primary School

## Biography

Timmy is an early primary school student with Level 2 Autism Spectrum Disorder (ASD). On the recommendation of medical experts, Timmy's parents are seeking an engaging game for him to help develop his emotional granularity to better his social skills.

Needs	Motivators
<ul style="list-style-type: none"><li>Emotional growth--Timmy needs to be able to develop in his ability to identify emotions.</li><li>Attention to be retained-- he is bored easily, quickly moving on to the next thing that has pulled his interest.</li></ul>	<ul style="list-style-type: none"><li>Things that are fun-- Timmy loves things that are fun, catchy and novel.</li><li>Things that allow him to feel a sense of achievement</li></ul>
Challenges	Key Success Criteria
<ul style="list-style-type: none"><li>Low attention span – Timmy isn't particularly interested in doing anything he doesn't see as fun or interesting.</li><li>Tech literacy-- As a first-grader, Timmy is not able to use applications that are not designed with ease of use in mind.</li><li>Complexity-- things that are difficult or frustrating are things that lose Timmy's attention.</li></ul>	<ul style="list-style-type: none"><li>Emotional granularity development-- The game should leave Timmy with an enhanced ability to identify and express emotions across a larger continuum.</li><li>High engagement-- The game should be developed to retain Timmy's low attention span by being engaging and enjoyable to ensure he receives maximal benefits.</li></ul>

## Persona for the mature-aged user role - Tim, the adult



"I just need something to help me learn  
a bit more about my emotions."

### Information

Age	32
Gender	Male
Education level	Bachelor Degree

### Biography

Tim is a young & educated adult with autism who has recently been exposed to the Theory of Constructed Emotions. Having learnt about the concept of emotional granularity, he sees this as an opportunity to develop his emotional granularity and improve his social skills for the workplace.

Needs	Wants
<ul style="list-style-type: none"><li>A clear progress indicator-- metrics that allow him to see how far he has progressed.</li><li>Social development-- he needs to be able to gain something from his use.</li><li>Purpose-- the value &amp; functionality should be particularly direct &amp; clear.</li></ul>	<ul style="list-style-type: none"><li>Predictability – James should be able to respond to the game based on preexisting situations he has been exposed to.</li><li>Explicit level of diagnosis-- James should be able to determine where he is with his emotional granularity.</li></ul>
Challenges	Key Success Criteria
<ul style="list-style-type: none"><li>Minimal time for use-- as a working adult, Tim has little time to use applications, and needs measurable results immediately.</li></ul>	<ul style="list-style-type: none"><li>Development-- After having used the game, he should have come away with a heightened emotional granularity.</li><li>Sense of progress – The game should implement a mechanism to ensure that James comes away after having playing feeling as though he was better than before he started.</li></ul>



## Persona for the Parent/Guardian Role - John, a concerned father



"I really hope that I can find something to help Timmy learn a bit more."

### Demographic information

Age	43
Gender	Male
Education level	Bachelor Degree

### Biography

John is a father of two, one of which has recently been diagnosed with Level 2 ASD. Following him having sought medical advice, applications to facilitate his child's emotional granularity growth have been recommended. Whilst skeptical, he is willing to try them if it helps his child out.

Needs	Motivators
<ul style="list-style-type: none"><li>Positive outcomes for his child-- he needs his child's social skills develop.</li><li>Trackability – He likes things that can be clearly measured and tracked.</li></ul>	<ul style="list-style-type: none"><li>Transparency/reputability – John values things that appear reputable and trustworthy through professional banners/visuals.</li></ul>
Challenges	Key Success Criteria
<ul style="list-style-type: none"><li>An uninterested child-- he wants his child's attention to be directed towards things if they're important for them.</li></ul>	<ul style="list-style-type: none"><li>A safe app-- John needs his child to be safe while they are using the app.</li><li>Emotional granularity development-- John needs to be able to see that his child is improving through direct indicators.</li></ul>

# Persona for Masters Student Role - Mary, the Game Integrator



"When can we ship this by?"

## 👤 Demographic information

Age	23
Gender	Female
Education level	Masters Degree

## 📜 Biography

Mary is a Masters' student who is looking to integrate games designed with the theory of constructed emotions (and improving emotional granularity) in mind into an 'application hub'.

Needs	Motivators
<ul style="list-style-type: none"><li>A deliverable executable game that can be easily integrated</li></ul>	<ul style="list-style-type: none"><li>Descriptive documentation on implementing the product</li></ul>
Challenges	Key Success Criteria
<ul style="list-style-type: none"><li>Consistency-- must ensure that any game developed lines up with aesthetics already set.</li><li>Diverse software-- given the differences in software between Mary and any game developed, it may be a challenge to create a game they can integrate</li></ul>	<ul style="list-style-type: none"><li>Being timely-- having efficient and practices that enable their collaborators to not be time-blind are necessary</li><li>Existing requirements met-- any requirements for integrating the application must be met.</li></ul>

## Persona for Client Role - The Client

Note: Some information normally included about demographics/images in Personas has been removed since this is more specific to our real-world client.

---

"The product must align with the theory for it to work."

### 📋 Biography

The Client is a professional skilled with the Theory of Constructed Emotions, and other associated concepts such as emotional granularity. As a result, they are seeking a game they can integrate into an application hub that provides measurable learning benefit for (primarily) children with autism when it comes to improving their emotional granularity.

Needs	Motivators
<ul style="list-style-type: none"><li>A deployable game to improve emotional granularity</li></ul>	<ul style="list-style-type: none"><li>Clear implementation of emotional granularity</li><li>Learning benefit for children produced by the application</li></ul>
Challenges	Key Success Criteria
<ul style="list-style-type: none"><li>Understanding of the theory-- if the game does not adhere to the theory, then it does not produce value for the client.</li><li>Complexity of mapping emotions-- linearly mapping emotions is complex, and the game must achieve this for client satisfaction.</li></ul>	<ul style="list-style-type: none"><li>Well-aligned to theory-- any game developed needs to adhere to the theory of constructed emotions</li><li>Producing value-- Above all, meeting the client needs something that produces measurable value, and would prefer something simple that works over something that is complex but fails.</li></ul>

## User Stories

### Epic: The game increases emotional granularity

User Story IDs		As a <ROLE>	I want (to) <Do something>	So that <achieve some goals>	Priority (must have ...)
US_01		Parent / Guardian / Client	The game to teach how to improve emotional granularity	The player can be educated on how to articulate emotion-concepts	Must have
US_02		Client	The game to not directly address / link emotions with set / concrete feelings	The game doesn't mislead the player to think that facial expressions always map to the same emotion thereby reducing emotional granularity	Must have
US_03		Client	The game to utilize the theory of constructed emotions and distinguish between affect / emotion	The game effectively captures more emotions that should map to the player's feelings (emotional granularity)	Must have
US_04		Parent/Guardian	To be able to configure the emotion concepts the game teaches	The child is able to learn specific emotion-concepts across particular parts of the emotional granularity axis	Won't have

### Epic: The game integrates into broader application hub designed by masters' students

User Story IDs		As a <ROLE>	I want (to) <Do something>	So that <achieve some goals>	Priority (must have ...)
US_05		Masters Student	To have the game be executable as a .ipa	The game is optimised for iPhone which uses iOS, the OS the	Must have

			game is expected to run under	
US_06	Client	The game to utilize local rather than global storage	Game does not require too much investment into global storage	Should have
US_07	Parent/Guardian	The game to send performance data to the main application about how the child is performing in the games so far	The progress of the child can be tracked in real-time	Won't have
US_08	Client	The game to have a banner and a scroll	The player can look at the summary and the banner to get details of game flow / play	Must have
US_09	Player	To be able to track the emotion-concepts learnt in the game via the greater 'application hub'	To determine how much progress has been made	Could have
US_10	Masters Student	The game to implement any unified arts that are prescribed	Keep the aesthetic style consistent across all minigames	Must have

## Epic: The game facilitates player progression

User Story IDs	As a <ROLE>	I want (to) <Do something>	So that <achieve some goals>	Priority (must have ...)
US_11	Player	The game to have a flow that is predictable	The game is reliable enough that concepts can be learnt	Should have
US_12	Player	The game to be easy and engaging enough for me to finish	The game can be finished in a reasonable amount of time without getting too frustrated	Must have
US_13	Player	The game to have some hints and guides for me to	The game does not become confusing or difficult	Could have

		follow when I'm struggling		
US_14	Player	The game to be engaging and interesting enough	The game will be played voluntarily	Should have
US_15	Player	The game to have 3 stages that is clear in what it is trying to achieve	The game can be completed without losing focus or getting frustrated	Must have
US_16	Client	The game to award players with a currency (hearts) based on their engagement	The player will have a sense of achievement and motivation to play the game / acknowledge the game is finished	Must have
US_17	Player	Complete each level in approximately 1 minute	The game is not too challenging/time consuming	Must have
US_18	Player	To be able to use the existing heart currency earnt in order to unlock items in the game	To further engage with the game	Won't have
US_19	Parent/Guardian	To be able to set the difficulty of the game	To train a child in simpler emotion-concepts	Won't have
US_20	Parent/Guardian	To view information about the child's gameplay in the game as opposed to the main application	To view current progress in the specific game	Won't have

## Design

### Sample Scenarios

 Sample scenarios

### Draft Visuals

 First Draft visuals

### Game Design Document

 Game Design Document

### ChatGPT Feedback on Game Proposal

 AI Feedback on our project proposal

### Research - Initial Game Brainstorming

 Research - Initial Game Brainstorming

## Sample scenarios

Below are some draft sample situations generated by chatGPT that we could use for the game.

1. **Situation:** Sarah successfully finished a challenging puzzle her grandmother had given her. The puzzle had many small pieces that formed a beautiful picture.
  - **Emotion:** Feeling proud
2. **Situation:** Alex was exploring a new park with their family and became separated from them while playing. They found themselves alone and unsure of where their family was.
  - **Emotion:** Feeling worried
3. **Situation:** Maya's best friend invited her to a sleepover party. At the sleepover, they planned to eat popcorn, share stories, and stargaze.
  - **Emotion:** Feeling excited
4. **Situation:** Liam went on stage for a school play but forgot the lines he was supposed to say. The audience looked surprised as he struggled to remember.
  - **Emotion:** Feeling embarrassed
5. **Situation:** Emily had a long day at school and felt tired. Her mom surprised her with her favorite cake and a candle at home.
  - **Emotion:** Feeling loved
6. **Situation:** Max saw other kids talking about a party, but he didn't receive an invitation to it.
  - **Emotion:** Feeling left out
7. **Situation:** Mia got an ice cream cone at the park, and a dog accidentally bumped into her, causing her ice cream to fall.
  - **Emotion:** Feeling frustrated
8. **Situation:** Emma got sick, and her younger brother made a special card to help her feel better.
  - **Emotion:** Feeling cared for
9. **Situation:** Daniel's family had to move to a new city because of his dad's job, leaving behind his old friends and familiar places.
  - **Emotion:** Feeling sad
10. **Situation:** During a science experiment, Ava accidentally spilled colorful stuff, creating an unexpected mess on the floor.
  - **Emotion:** Feeling surprised

## First Draft visuals

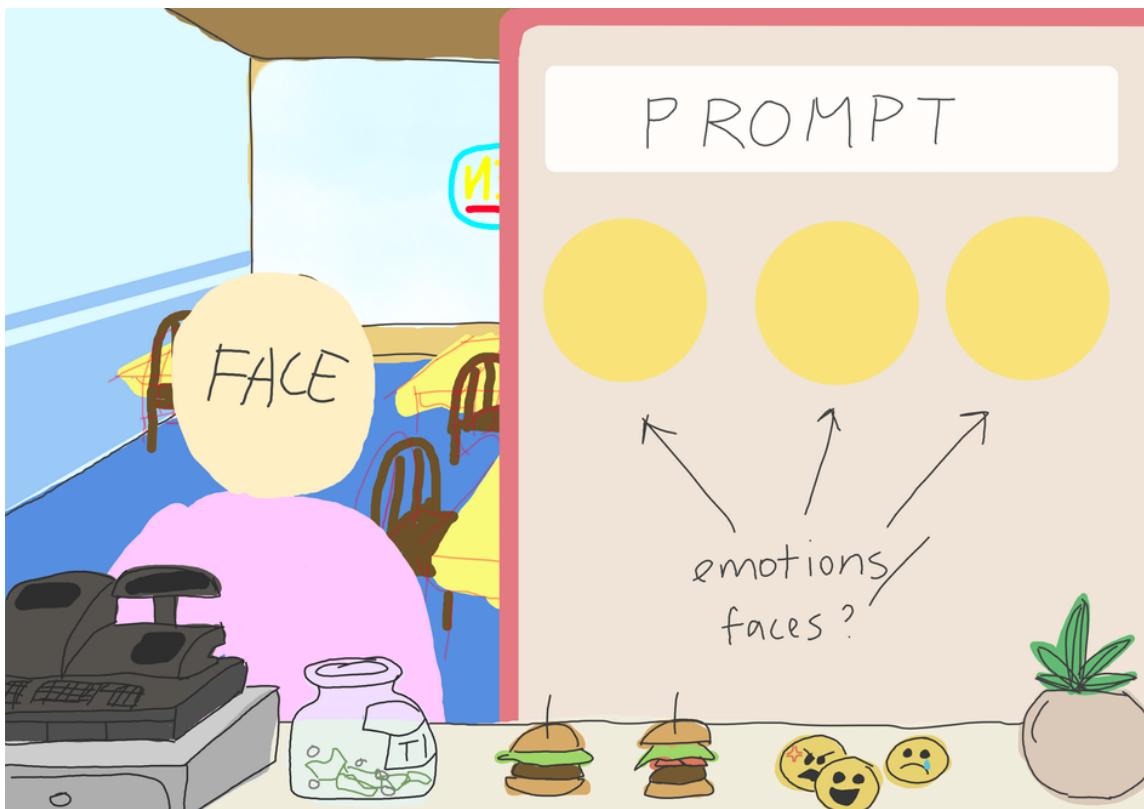
Background Image:



When level starts and customer appears with a written scenario and a facial expression:



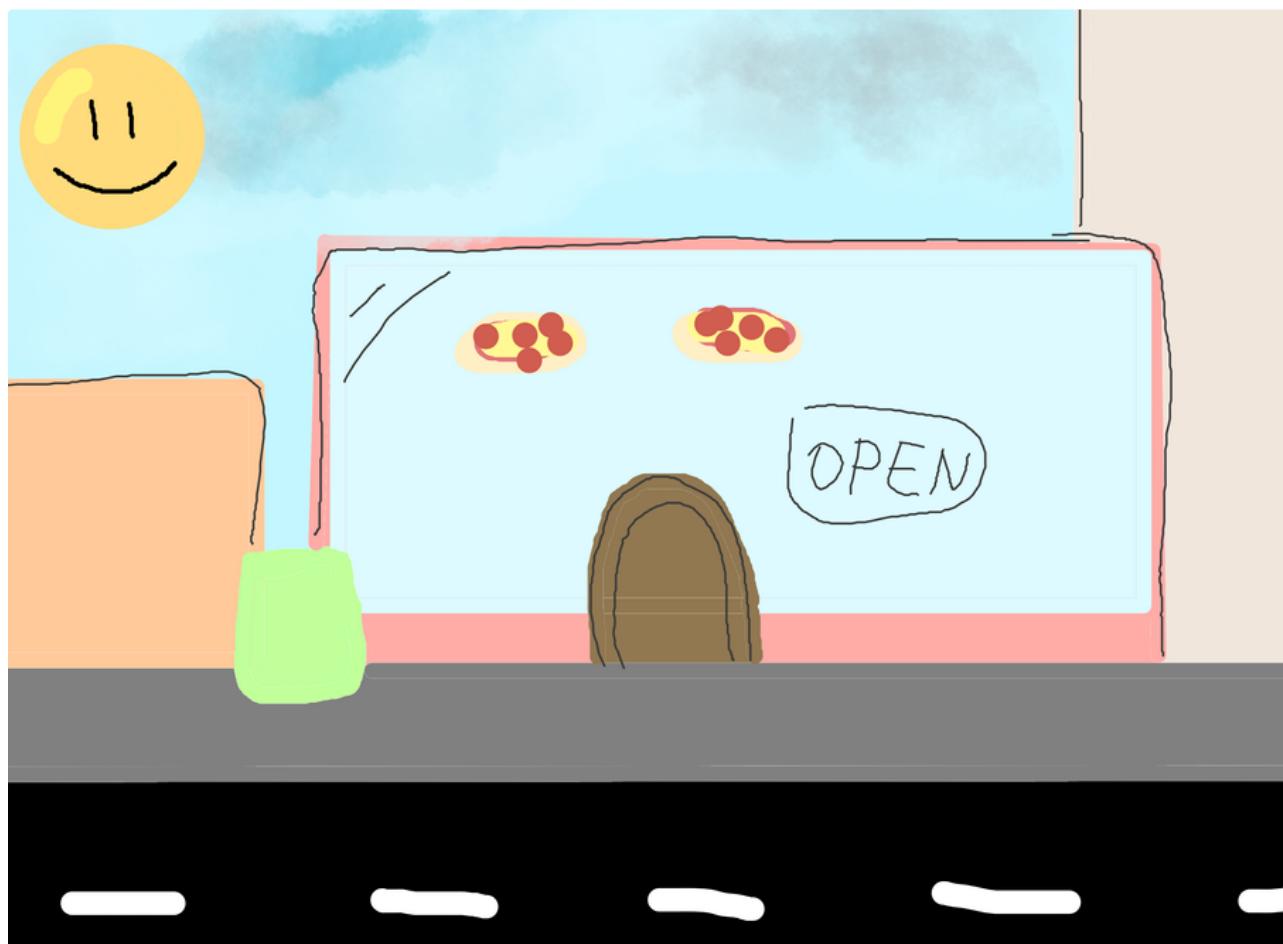
After user taps "NEXT", user will be able to select emotions/faces to match the scenario/prompt:



User interface: also add a settings button, so user can exit



Start screen:





# Game Design Document

## Game Overview

**Core Concept:** A “papa’s pizzeria” style game which involves the player having to serve customers the right order within a specific time limit. Instead of the traditional game where customers order food, the customers will tell the player about a situation they experienced, and the player will have to decide which emotion is closest to what they must be feeling. If they get it correct, they receive a certain number of hearts.

**Genre:** Children's game, restaurant management game

**Target Audience:** The game will be designed for a diverse audience that includes neurotypical and autistic individuals.

**Unique Selling Points (USPs):** Restaurant concept, emotional granularity, levels of increasing difficulty.

## Gameplay and Mechanics

**Player Perspective:** Player's perspective will be in first-person, with a fixed camera, invisible player character on screen



Player's general perspective

**Controls:** Players will only need to tap on the screen to interact with the game world

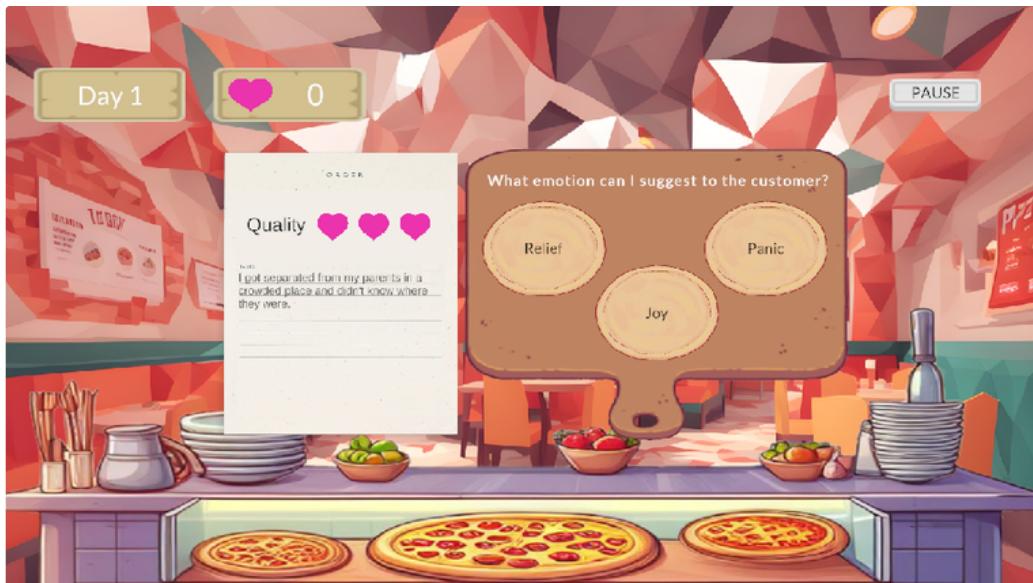
**Progression:** There will be 3 levels, each level having an increase in scenarios that are more complex (level 1: 3 scenarios, level 2: 6 scenarios, level 3: 9 scenarios). Complexity may entail more difficult emotions. The player cannot “lose” the game, however there is a hearts system in place to reward correct answers (level 1: 10 hearts, level 2: 30 hearts, level 3: 60 hearts). Player may also be able to collect emotions in their “cookbook” which entice the player continue to play.

**Gameplay Mechanics:** There may be an option for a time limit which can make the game more fun and tense. Prizes for having correct answers - hearts, a face that pops up and can be destroyed by tapping on it aggressively.

## Levels and World Design

**Game World:** Game world is 2.5D isometric . It will be a single screen. There will be three levels that will utilise the same scene. The player navigates through the game world by answering questions correctly - exploring different customers and emotions.

**Objects:** Player, customer, emotion.



Player selects the desired emotion (no faces)



Customer enters restaurant with a scenario

## Art and Audio

**Art Style:** Cartoonish and colourful. Aiming for an overall aesthetic of a popular fast food restaurant. Inspirations include: Papa's Pizzeria, and Good Pizza Great Pizza. Colours - shapes - textures - font (TBD).

**Sound and Music:** What is the sound design of the game? What sounds are used? What music is used? How do the sounds and music fit in with the overall aesthetic of the game? [?](#)

**Assets:** Are we going to be using any artistic assets? Should we start creating/resourcing these assets? [?](#)

## User Interface

User Interface will include a level indicator, heart counter and pause button.



UI buttons have been highlighted

## Technology and Tools

The main software we will use is Unity, where we consolidate all elements — models, textures, and sounds. In Unity, we will code essential game functions, including NPC interactions and so on. Furthermore, we also employ Unity for rigorous testing, enabling early playtesting for valuable user feedback.

We will utilise GitHub for version control, simplifying the process of document sharing and collaboration among team members: <https://github.com/SiRong-github/-TODO-Create-team-name> [Connect your Github account](#) [Connect your Github](#)

## AI Feedback on our project proposal

### Prompt

Hey, could you please give me feedback on this game proposal? It's supposed to be a game for iphones for autistic children in the age range from 8 to 18, and should help them improve emotional granularity. Don't be afraid to be critical!

Here's the proposal: We aim to a "papa's pizzeria" style game which involves the player having to serve customers the right order within a specific time limit. Instead of the traditional game where customers order food, the customers will tell the player about a situation they experienced (possibly matched with a facial expression - but leave this to last, and maybe match the emotion with a facial expression as well), and the player will have to decide which emotion is closest to what they must be feeling. If they get it correct, they receive a certain number of hearts.

Here are some more specifics: The customers will have multiple ways of hinting at the emotion they're feeling They might describe a situation

- using a text speech bubble,
- through audio
- or through a sign language video

They might show a facial expression There will be three levels, and it will become progressively more challenging for the player to choose the emotion to match the customer. Multiple customers will come in each level, and have a time limit (maybe - add at the end) to be served by.

### Response

Your game proposal is an interesting concept for helping autistic children improve emotional granularity. However, there are several points to consider and potentially improve upon:

- 1. Clarity of Objective:** The core objective of the game should be clearer. While you mentioned that the player needs to match emotions, it would be beneficial to explicitly state how this helps with emotional granularity and why it's important for the target audience.
- 2. Age Appropriateness:** The age range of 8 to 18 is quite broad, and the game's difficulty level should be adjustable to cater to different age groups within this range. You might want to consider offering multiple difficulty settings or having different game modes for younger and older players.
- 3. Accessibility:** You mentioned the use of text speech bubbles, audio, and sign language videos for communication. Ensure that these elements are accessible to all players, including those with hearing or visual impairments. Consider providing alternative ways to convey information for players who may have difficulty with certain modes of communication.
- 4. Learning Curve:** You mentioned that the game becomes progressively more challenging, which is good, but be careful not to make it too difficult too quickly. Provide a smooth learning curve to ensure that players gradually build their emotional understanding skills.
- 5. Feedback Mechanism:** It's crucial to have a clear feedback mechanism. When a player makes a choice, provide feedback on whether it was correct or not and offer explanations if needed. Positive reinforcement can motivate players to continue improving.
- 6. Time Limit:** The inclusion of a time limit should be carefully considered. While time pressure can add excitement, it may also increase stress, which might not be suitable for all autistic children. You might want to provide the option to turn off or adjust the time limit based on the player's preference.
- 7. Diversity and Inclusion:** Ensure diversity among the customers and their emotional situations to reflect the real world. This can help players practice recognizing a wide range of emotions in various contexts.
- 8. Visuals and Interface:** Consider the visual design and interface. It should be user-friendly and appealing to children. If you plan to include facial expressions, ensure they are clear and easy to understand.
- 9. Research and Expert Input:** It's important to collaborate with experts in autism and child psychology to ensure that the game effectively meets its educational and therapeutic goals. Conduct research to understand the specific needs of your target audience.

10. **Privacy and Safety:** If the game includes any interactive features, consider privacy and safety measures, especially when dealing with children.
11. **Monetization:** If the game is intended to be a commercial product, think about the monetization strategy. Will it be a one-time purchase, free with in-app purchases, or ad-supported? Make sure that any monetization model chosen is fair and doesn't hinder the game's educational value.

Overall, your idea has potential, but it requires careful execution to ensure it effectively serves its purpose while providing an enjoyable gaming experience for autistic children. Collaborating with experts in the field and conducting user testing with your target audience can be invaluable in refining the concept and gameplay.

## Research - Initial Game Brainstorming

This was an initial collaborative space used early on in the semester to gather and showcase games we found that we believed improve emotional granularity.

The underlying goal in all of the games shown in the comments was to improve emotional granularity. We sought to analyze and understand game paradigms for how we might better help develop the emotional maturity of the children who would likely play our game.

Essentially, you will find descriptions of features we found useful that we ultimately implemented in our final game.

## Architectural Design

We have opted for a 4+1 architectural view of our design for the product.

### Logical View

 [Logical view](#)

### Process View

 [Process view](#)

### Development View

 [Development view](#)

### Physical View

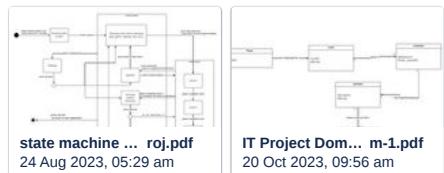
 [Physical view](#)

### Scenarios

 [Scenario View](#)

## Logical view

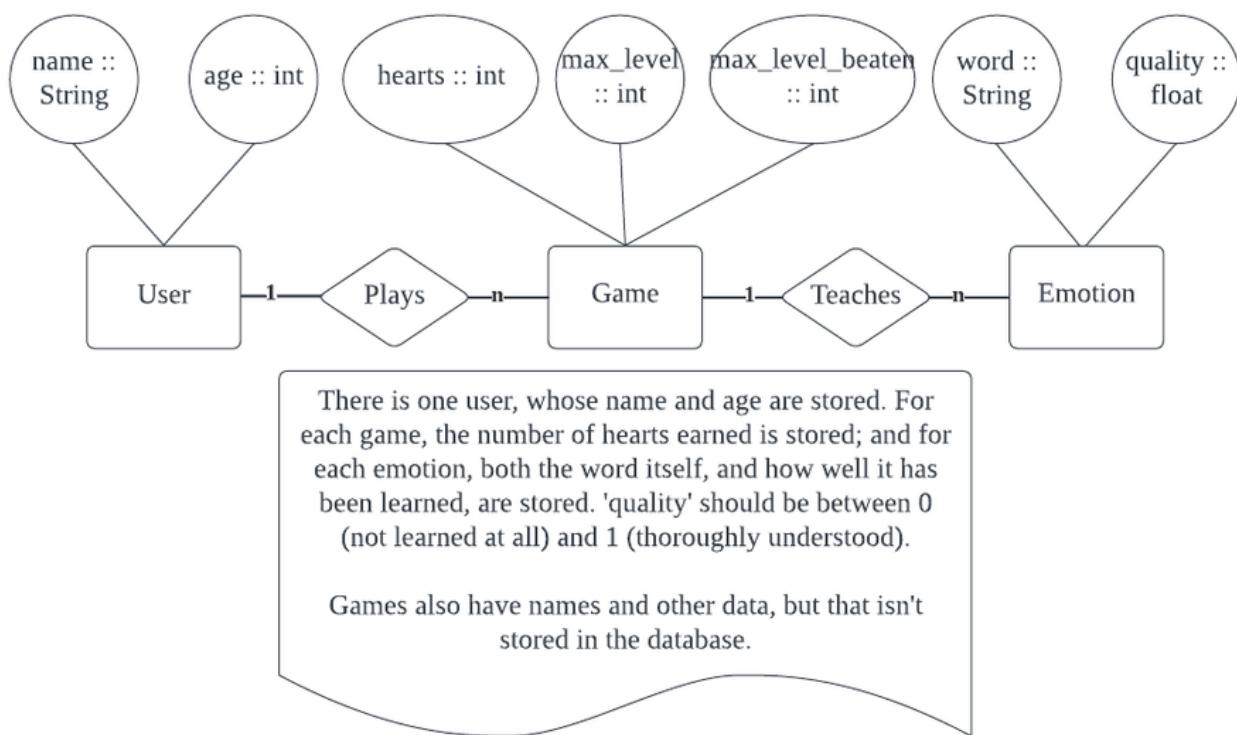
Here is a link to a Domain class diagram: [IT Project: Domain class diagram | Lucidchart](#)



## Process view

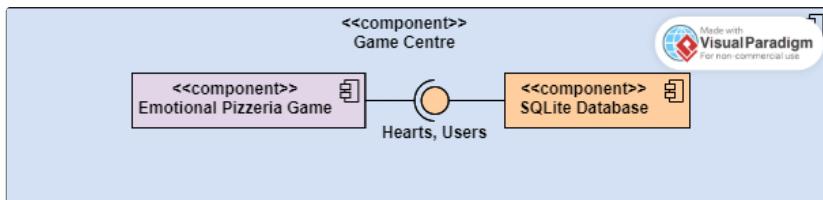
Here is a link to a State machine diagram: [IT Project: State machine diagram | Lucidchart](#)

Additionally, here is an ER diagram created by the Master's students that demonstrates how the games will interact with the database:



## Development view

Component diagram below. See also the attached pdf.



Additionally, we have provided some of the architectural goals and constraints to further describe this view:

Requirements	What	Why	How
Usability	Has a straightforward and simple interface for users.	To prevent confusion for target audience of children and maximize gameplay enjoyability.	Consistent interface logic that does not deceive the user (e.g., buttons function as expected)
Reliability	Is able to work in the absence of the main application.	To ensure the game can still work as a standalone application.	By performing a safety check to see if the main application (game center) by the Masters' stakeholders is installed on the device.
Supportability	Works across multiple iPhone (iOS devices)	To ensure users can use the application across a variety of different iOS devices.	Automatic resolution resizing logic implemented that adjusts resolution based on the device being used.
Extensibility	Supports future modifications to the database by the client	The client in future may choose to migrate the emotion database to a cloud database	Modify existing Unity code to pull information from online database as opposed to referring to a local directory.
Security	Does not compromise user data, keeps players personal information safe	<ul style="list-style-type: none"> <li>• Data protection laws</li> <li>• Client trust</li> <li>• Personal user safety</li> </ul>	<ul style="list-style-type: none"> <li>• Keep gameplay functionality offline to prevent interaction with other player devices</li> </ul>

- Do not take information from the player that is not necessary

## Physical view

We created a deployment diagram to highlight the relationship between our software designed and the Masters Students.

Displayed below:

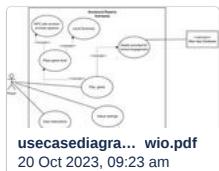


Here is the XML file for the diagram if anyone wishes to edit it:

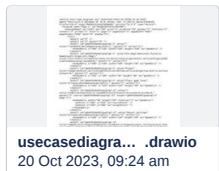


# Scenario View

Use case diagram detailing overall system



A drawio file is also provided here in case edits are required:



**Use case descriptions:**

**UC-01: Customer scenario**

## Summary

Provide the user with a customer that they can engage with in the game

## Rationale

One of the core components of the emotional pizzeria is the concept of customers coming in and telling the player how they are currently feeling. This part of the game essentially underpins the overall game concept

## Users

All players of the game.

## Preconditions

The player has started playing the game.

## Steps

1. The user begins the game level by pressing the start button.
2. The game will have the 'door' of the pizzeria open as an audio cue that a customer is appearing, and footsteps will sound.
3. A 'character' will appear in front of the player, and begin describing an experience they had with reference to the particular emotional state during that experience. Their face will also be somewhat related to their emotional state.
4. Following the character completing their description, the game will create an 'Okay!' button.
5. The user presses the 'Okay!' button.
6. The game will present an 'emotion prompt' to the player, with three specific emotions that they can select. The face of the customer and the situation they described will also be available in the prompt to allow them to review the event.
7. The player selects the emotion that most closely represents the emotion concept the customer is feeling.
8. The game will produce a 'ding!' sound to highlight the player on their correct choice. The other emotions will become greyed out to prevent their use, leaving the player to move on to the next part of the game.
9. The player will click the button to move on.

10. The game will have the customer disappear, with walking sounds to indicate this to them and the 'door' of the pizzeria shutting via audio cues.

#### **Alternative Paths:**

In step 7, it is possible for a player to click an incorrect emotion, and the game will signal this to the user, remaining on this step until a correct emotion is chosen.

7b. The player incorrectly picks the emotion.

8b. The game will make a negative 'beeping' sound to indicate an incorrect option. The game will also grey out the incorrect option chosen, removing an option.

#### **Postconditions**

The player's "heart count" will be updated depending on the number of guesses needed by the player to guess the emotion concept. If they have reached the end of the level, an 'end level' summary will appear.

# Coding

Coding Standards

 Coding Standards

Development and Code Reviews

 Development and Code Reviews

Unity References

 References

# Coding Standards

These are a general list of coding principles we have decided to follow below, though we have also opted to use the Google Style Guide for C# ([🔗 C# at Google Style Guide](#)) as a reference for any C#-specific constructs that are not covered below that we may not yet be familiar with.

## Naming conventions - camelCase

- local variables: start with lower case
- function names & script (class) names should start with upper case
- constant variables: capitalized
- avoid digits in names
- names of functions and variables should be *concise and self-explanatory*

## Indentation

```
1 namespace CoolStuff.AwesomeFeature
2 {
3     using Azure;
4
5     public class Awesome
6     {
7         public void Stuff()
8         {
9             WaitUntil wait = WaitUntil.Completed;
10            // ...
11        }
12    }
13 }
```

## Documentation / Comments

- include comments to indicate author of function  
e.g., *author* - @Jiwon / *last modified* - August 24th, 2023

# Development and Code Reviews

## Using Git

### Commits

- commit comments should be 1-2 sentences - should be descriptive of what component was updated / modified

### Merging / resolving conflicts

- have team members of merge conflicts present at the same time to resolve conflicts

### Branching

- create a separate branch for each component of the module
- branch names should reflect what component of the program is being worked on / modified
- do not commit to the main branch - this should be done when all / majority of team members are present

### Code Reviews

The following is a step by step procedure we should undertake any time we are developing:

1. Make sure your local main branch is updated (do a git pull)
2. Create a new branch off main to do your development
3. Test your changes
  - a. Once we have a testing framework, check the functionality of your changes by applying all the tests
4. Push the branch to the remote repository, create a pull request to main, and ask at least one other team member to review your pull request
5. Once the reviewer approves your changes, merge your branch to main on GitHub
6. Pull down main and test again to ensure nothing broke

### Key takeaways:

- avoid committing and pushing directly to main unless other team members are present
- Someone must always review your code
- Testing is essential to make sure your changes don't break things

### Code Review Documentation

 [Code Reviews](#)



## References

- Pause Menu  
[!\[\]\(b918af89b3d29b3fd1d13eb4ddf791b5\_img.jpg\) 6 Minute PAUSE MENU Unity Tutorial](#)
- Speech Bubble Text Effect  
[!\[\]\(da9ce1d6b0f2a0fcf0d0a74eb777d743\_img.jpg\) Nick Hwang](#)

# Testing

## Client Feedbacks

We have compiled the client feedbacks. Since some epics and user stories could not be tested via acceptance testing (i.e. Epic: The game increases emotional granularity), we included these here. Additionally, we wanted to prove that our client was satisfied with our product.

### Client Feedbacks

## Testing

 [230914] merging emotions panel into main

 [231004] temporary\_branch creation

 [231005] temporary\_branch testing

 [231012] update\_art testing

 [231014] temporary\_branch

 [231015] fixed\_typing testing

 [231015] fixed\_typing testing 2

 [231018] main merge

 [231018] database-resilience testing

 [231019] final testing in Unity application

 [231024] finalising game

## Client Feedbacks

Date	Game Feature/s	Client Feedback	Summary	Epics and User Stories	Outcome
11th August	<p>Our initial game idea was to have a diary-like game where:</p> <ul style="list-style-type: none"> <li>• players can customise faces (i.e. multiple eyes and noses) to indicate how they're feeling</li> <li>• the game asks how they're feeling and suggest what they could add to the drawing</li> </ul>	<ul style="list-style-type: none"> <li>• different matching facial expressions</li> <li>• okay in principle, matching faces have to be done carefully as it should align with the theory <ul style="list-style-type: none"> <li>◦ people don't have same facial expressions</li> <li>◦ you can't tell a person can be smiling and furiously angry, could be neutral</li> <li>◦ unless you have a concept for the emotion, you can't feel the emotion</li> <li>◦ prolly not have time to do this</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• There might not be enough time for the game idea.</li> <li>• Should be done cautiously to ensure it aligns with the theory.</li> </ul>	<p><b>Epic:</b> The game increases emotional granularity</p> <p><b>User Stories:</b></p> <ul style="list-style-type: none"> <li>• As a Parent / Guardian / Client, I want to the game to teach how to improve emotional granularity, so that the player can be educated on how to articulate emotion-concepts.</li> <li>• As a Client, I want to the game to not directly address / link emotions with set / concrete feelings so that the game doesn't mislead the player to think that facial expressions always map to the same emotion</li> </ul>	Fail

				<p>thereby reducing emotional granularity.</p> <ul style="list-style-type: none"> <li>As a Client, I want to the game to utilize the theory of constructed emotions and distinguish between affect / emotion so that the game effectively captures more emotions that should map to the player's feelings (emotional granularity)</li> </ul>	
18th August	<p>Our final game idea was to have a Papa's Pizzeria-like game where:</p> <ul style="list-style-type: none"> <li>customers appear and talk about their situation (context of the emotion)</li> <li>player chooses which emotion the customer might be feeling (emotion)</li> <li>customers will have generic</li> </ul>	<ul style="list-style-type: none"> <li>can also have something like a depression pizza <ul style="list-style-type: none"> <li>cheese, tomatoes could have different components of depression (?)</li> <li>or new pizza that describes the feeling of getting feedback</li> </ul> </li> </ul>	<p>The client approves of the game idea and has given some suggestions to make it more fun.</p>	<p><b>Epic:</b> The game increases emotional granularity</p> <p><b>User Stories:</b></p> <ul style="list-style-type: none"> <li>As a Parent / Guardian / Client, I want to the game to teach how to improve emotional granularity, so that the player can be educated on how to articulate emotion-concepts.</li> </ul>	Success

	<p>expressions (i.e. enthusiastic and happy have the same expression)</p> <ul style="list-style-type: none"> <li>• emotions player chooses are in the form of pizzas</li> </ul>	<p>in a meeting that feels slightly unpleasa nt</p> <ul style="list-style-type: none"> <li>■ eleva ted → on the spotli ght</li> <li>■ but it feels some what pleas ant as you'r e given feedb ack</li> </ul>	<ul style="list-style-type: none"> <li>• As a Client, I want to the game to not directly address / link emotions with set / concrete feelings so that the game doesn't mislead the player to think that facial expressions always map to the same emotion thereby reducing emotional granularity.</li> <li>• As a Client, I want to the game to utilize the theory of constructed emotions and distinguish between affect / emotion so that the game effectively captures more emotions that should map to the player's feelings (emotional granularity)</li> </ul>		
8th September	We have shown a game demo	<ul style="list-style-type: none"> <li>• appropriate visuals and audio</li> </ul>	Client has not commented on the gameplay.	N/A	N/A

with completed gameplay.	<ul style="list-style-type: none"> <li>• can you do the faces as pizzas</li> <li>• scenarios could come from database <ul style="list-style-type: none"> <li>◦ Mark (or anyone else) to buy the chatgpt pro and James will just pay him back</li> </ul> </li> <li>• move from wireframe prototype</li> <li>• this one has low fidelity graphics <ul style="list-style-type: none"> <li>◦ when you start combinin g high and low fidelity graphics, it's hard to tell which ones are going to be switched out (i.e. +1 Hearts)</li> <li>◦ high fidelity faces in the background (is this one what it's actually going to look like)</li> </ul> </li> </ul>	Instead, he talked about moving from the wireframe prototype and how our game has low fidelity graphics.
--------------------------	---	--

22nd September	We have shown a game demo with updated graphics.	<ul style="list-style-type: none"> <li>• excellent</li> <li>• when it comes to graphics and fonts, don't non-proportionally scale them (squash them to fit)</li> <li>• need to start sourcing graphics asap to show how the graphics will look like</li> <li>• have a style frame to for graphics</li> <li>• you're better off sourcing already made work instead designing it yourself unless there's a good reason</li> </ul>	The client has urged us to improve on the graphics.	N/A	N/A
29th September	We have made further improvements on the graphics.	<p><b>Background Image</b></p> <p><b>Comment:</b> Are they intended to be part of the final design?</p> <p><b>Feedback:</b> I am not sure if this is what we want. It looks like a background image of a city street. I think we should have a more abstract background image.</p> <p><b>Input:</b> I think we should have a more abstract background image. It's better for the game.</p> <p><b>Design:</b> I think we should have a more abstract background image. It's better for the game.</p> <p><b>Output:</b> I think we should have a more abstract background image. It's better for the game.</p> <p><b>Color</b></p> <p><b>Comment:</b> The colors are not consistent.</p> <p><b>Feedback:</b> I think we should have a more consistent color palette. The colors in the game are not consistent.</p> <p><b>Input:</b> I think we should have a more consistent color palette. The colors in the game are not consistent.</p> <p><b>Design:</b> I think we should have a more consistent color palette. The colors in the game are not consistent.</p> <p><b>Output:</b> I think we should have a more consistent color palette. The colors in the game are not consistent.</p> <p><b>UI</b></p> <p><b>Comment:</b> The UI is not consistent.</p> <p><b>Feedback:</b> I think we should have a more consistent UI. The UI in the game is not consistent.</p> <p><b>Input:</b> I think we should have a more consistent UI. The UI in the game is not consistent.</p> <p><b>Design:</b> I think we should have a more consistent UI. The UI in the game is not consistent.</p> <p><b>Output:</b> I think we should have a more consistent UI. The UI in the game is not consistent.</p> <p><b>Props</b></p> <p><b>Comment:</b> The props are not consistent.</p> <p><b>Feedback:</b> I think we should have a more consistent set of props. The props in the game are not consistent.</p> <p><b>Input:</b> I think we should have a more consistent set of props. The props in the game are not consistent.</p> <p><b>Design:</b> I think we should have a more consistent set of props. The props in the game are not consistent.</p> <p><b>Output:</b> I think we should have a more consistent set of props. The props in the game are not consistent.</p> <p><b>Environment</b></p> <p><b>Comment:</b> The environment is not consistent.</p> <p><b>Feedback:</b> I think we should have a more consistent environment. The environment in the game is not consistent.</p> <p><b>Input:</b> I think we should have a more consistent environment. The environment in the game is not consistent.</p> <p><b>Design:</b> I think we should have a more consistent environment. The environment in the game is not consistent.</p> <p><b>Output:</b> I think we should have a more consistent environment. The environment in the game is not consistent.</p> <p><b>Model</b></p> <p><b>Comment:</b> The models are not consistent.</p> <p><b>Feedback:</b> I think we should have a more consistent set of models. The models in the game are not consistent.</p> <p><b>Input:</b> I think we should have a more consistent set of models. The models in the game are not consistent.</p> <p><b>Design:</b> I think we should have a more consistent set of models. The models in the game are not consistent.</p> <p><b>Output:</b> I think we should have a more consistent set of models. The models in the game are not consistent.</p> <p><b>Lighting</b></p> <p><b>Comment:</b> The lighting is not consistent.</p> <p><b>Feedback:</b> I think we should have a more consistent set of lighting. The lighting in the game is not consistent.</p> <p><b>Input:</b> I think we should have a more consistent set of lighting. The lighting in the game is not consistent.</p> <p><b>Design:</b> I think we should have a more consistent set of lighting. The lighting in the game is not consistent.</p> <p><b>Output:</b> I think we should have a more consistent set of lighting. The lighting in the game is not consistent.</p> <p><b>Image Sources</b></p>	The client has given us a more detailed feedback for each aspect of the game visuals.	N/A	N/A
13th October	We have made further		The client said that we have a	Epic:	Success

	improvements on the graphics.	"massive update".	The game integrates into broader application hub designed by masters' students  As a Client, I want the game to have a banner and a scroll so that the player can look at the summary and the banner to get details of game flow / play.	As a Masters Student, I want the game to implement any unified arts that are prescribed so that keep the aesthetic style consistent across all minigames.	
19th October (Team Presentation)	We have shown a demo of our completed game.	As the client was absent for most of our presentation, we did not receive a detailed feedback on our game. However, he said that our presentation was excellent and did not seem to have further comments.	The client seemed to be satisfied with our game.	Epic:  The game facilitates player progression  As a player, I want the game to be easy and engaging enough for me to finish so that the game can be finished in a reasonable amount of time without getting too frustrated.	Success

## [230914] merging emotions panel into main

### Unit Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
U_01	Valid choice of scenarios with expression sprites	N/A	File structure valid under Assets\Resources\Scenarios	Scene showing expression and text when game is simulating	Sprite present	Failure	Sprite not linked with scenario
U_02	Valid and randomized choice of person sprite	N/A	Correct sprite set in Customer Component - Character Script	Person sprite seen when level is started	Not implemented	Failure	N/A
U_03	Connection with database for hearts	Press +1 Heart button in Main Menu	Database file valid / space on local storage	Log showing incrementing number of hearts in database	Not connected	Failure	Requirements for database not yet finalised
U_04	Correct implementation of typewriter for scenario	Left click on mouse	Valid access and open of text file	Start with typewriter skip ahead with left-click of mouse → should have full text shown	As expected	Success	N/A
U_05	Valid options for selecting emotions from prompts	N/A	File listing valid / appropriate emotions	No duplicate choices, all choices valid and appropriate emotions	Not implemented	Failure	Only just made prompt panel
U_06	Main menu buttons / end level screen buttons with correct behavior	Click on buttons	Scenes linking to exists	Link to correct scenes / prompts showing	Same as expected	Success	N/A
U_07	Pause button correctly working	Click on pause button	Level is running	Status of typewriter is constant,	Not implemented	Failure	N/A

				disabled when end prompt is layered			
--	--	--	--	---	--	--	--

## Integration Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
I_01	Scenarios and expression sprites are kept constant / validly showing	N/A	U_01	Scenario, expression matches on customer and order summary (in emotions prompt)	As expected	Success	Scenario yet to be matched to emotion options
I_02	Correct layering of prompts (customer, emotions prompt, pause and end prompt)	Clicking on show prompt button / pause button / finishing level	U_06, U_07	Layering is correct in the order of hierarchy (customer < emotions prompt < pause / end prompt )	Character appears beside prompts	Failure	Layering to be adjusted - speech bubble and character needs to be adjusted for emotions panel
I_03	Choice of answers from emotions prompt linked with correct score	Clicking on options from emotions prompt	U_05	Increment hearts obtained for correct answer & correctness of choice logged in console	Not implemented	Failure	Default settings for scoring has been made, but yet to be decided
I_04	Correct sequence for game flow	Playing the game and choosing options	I_02, U_01/2/4/5	Customer enters → Scenario is explained → Prompt is shown and options chosen → summary of level shown	As expected	Success	As of right now, we only expect one customer

# [231004] temporary\_branch creation

Date : 4th October, 2023

Tested by : Jiwon Bae

Branch : *temporary\_branch*

Commit Name (Hash) : N/A (creating branch)

## Unit Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
U_01	Valid choice of scenarios with expression sprites	N/A	File structure valid under Assets\Resourcess\Scenarios	Scene showing expression and text when game is simulating	Same as expected	Success	N/A
U_02	Valid and randomized choice of person sprite	N/A	Correct sprite set in Customer Component - Character Script	Person sprite seen when level is started	Same as expected	Success	N/A
U_03	Connection with database for hearts	Press +1 Heart button in Main Menu	Database file valid / space on local storage	Log showing incrementing number of hearts in database	Same as expected	N/A	Temporary fix of creating a new database and incrementing heart
U_04	Correct implementation of typewriter for scenario	Left click on mouse	Valid access and open of text file	Start with typewriter skip ahead with left-click of mouse → should have full text shown	Same as expected	Success	N/A
U_05	Valid options for selecting emotions from prompts	N/A	File listing valid / appropriate emotions	No duplicate choices, all choices valid and appropriate emotions	Invalid options for wrong choices	Failure	Need to hardcode the wrong options into json file
U_06	Main menu buttons / end level screen buttons with correct behavior	Click on buttons	Scenes linking to exists	Link to correct scenes / prompts showing	Same as expected	Success	N/A

U_07	Pause button correctly working	Click on pause button	Level is running	Status of typewriter is constant, disabled when end prompt is layered	Same as expected	Success	N/A
------	--------------------------------	-----------------------	------------------	---	------------------	---------	-----

## Integration Testing

Epic	User Story	Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
		I_01	Scenarios and expression sprites are kept constant / validly showing	N/A	U_01	Scenario, expression matches on customer and order summary (in emotions prompt)	Printing only "I" when exiting to main menu and returning	Failure	Research on issues of object not deleting on deletion of parent
		I_02	Correct layering of prompts (customer, emotions prompt, pause and end prompt)	Clicking on show prompt button / pause button / finishing level	U_06, U_07	Layering is correct in the order of hierarchy (customer < emotions prompt < pause / end prompt )	Same as expected	Success	N/A
		I_03	Choice of answers from emotions prompt linked with correct score	Clicking on options from emotions prompt	U_05	Increment hearts obtained for correct answer & correctness of choice logged in console	Scoring not implemented yet	Failure	Jiwon - implement by week 12
		I_04	Correct sequence for game flow	Playing the game and choosing options	I_02, U_01/2/4/5	Customer enter → Scenario is explained → Prompt is shown and options	Same as expected	Success	5 customers showing

					chosen → summary of level shown			
--	--	--	--	--	---------------------------------------	--	--	--

- No acceptance testing as we have not produced a complete executable

# [231005] temporary\_branch testing

Date : 5th October, 2023

Tested by : Jiwon Bae

Branch : temporary\_branch

Commit Name (Hash) : hints and other things implemented (1e9e3326c8902e22309b89ce58ee9abb19ac94b8)

## Unit Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
U_01	Valid choice of scenarios with expression sprites	N/A	File structure valid under Assets\Resourc es\Scenarios	Scene showing expression and text when game is simulating	Same as expected	Success	N/A
U_02	Valid and randomized choice of person sprite	N/A	Correct sprite set in Customer Component - Character Script	Person sprite seen when level is started	Same as expected	Success	N/A
U_03	Connection with database for hearts	Press +1 Heart button in Main Menu	Database file valid / space on local storage	Log showing incrementing number of hearts in database	Same as expected	Success	Temporary fix of creating a new database and incrementing heart
U_04	Correct implementation of typewriter for scenario	Left click on mouse	Valid access and open of text file	Start with typewriter skip ahead with left-click of mouse → should have full text shown	Same as expected	Success	N/A
U_05	Valid options for selecting emotions from prompts	N/A	File listing valid / appropriate emotions	No duplicate choices, all choices valid and appropriate emotions	Same as expected	Success	Make decision on format of invalid emotions
U_06	Main menu buttons / end level screen buttons with correct behavior	Click on buttons	Scenes linking to exists	Link to correct scenes / prompts showing	Same as expected	Success	N/A
U_07	Pause button correctly working	Click on pause button	Level is running	Status of typewriter is	Same as expected	Success	N/A

				constant, disabled when end prompt is layered			
--	--	--	--	--	--	--	--

## Integration Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
I_01	Scenarios and expression sprites are kept constant / validly showing	N/A	U_01	Scenario, expression matches on customer and order summary (in emotions prompt)	Printing only "I" when exiting to main menu and returning	Failure	Research on issues of object not deleting on deletion of parent
I_02	Correct layering of prompts (customer, emotions prompt, pause and end prompt)	Clicking on show prompt button / pause button / finishing level	U_06, U_07	Layering is correct in the order of hierarchy (customer < emotions prompt < pause / end prompt )	Same as expected	Success	N/A
I_03	Choice of answers from emotions prompt linked with correct score	Clicking on options from emotions prompt	U_05	Increment hearts obtained for correct answer & correctness of choice logged in console	Scoring not implemented yet	Failure	Jiwon - implement by week 12
I_04	Correct sequence for game flow	Playing the game and choosing options	I_02, U_01/2/4/5	Customer enter → Scenario is explained → Prompt is shown and options chosen → summary of level shown	Same as expected	Success	5 customers showing

- No acceptance testing as there is no complete executable

## [231012] update\_art testing

### Unit Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
U_01	Valid choice of scenarios with expression sprites	N/A	File structure valid under Assets\Resourc es\Scenarios	Scene showing expression and text when game is simulating	As expected	Success	N/A
U_02	Valid and randomized choice of person sprite	N/A	Correct sprite set in Customer Component - Character Script	Person sprite seen when level is started	As expected	Success	Sprite of characters yet to be updated
U_03	Connection with database for hearts	Press +1 Heart button in Main Menu	Database file valid / space on local storage	Log showing incrementing number of hearts in database	No database connection	Failure	Not connected as of yet
U_04	Correct implementation of typewriter for scenario	Left click on mouse	Valid access and open of text file	Start with typewriter skip ahead with left-click of mouse → should have full text shown	As expected	Success	N/A
U_05	Valid options for selecting emotions from prompts	N/A	File listing valid / appropriate emotions	No duplicate choices, all choices valid and appropriate emotions	As expected	Success	N/A
U_06	Main menu buttons / end level screen buttons with correct behavior	Click on buttons	Scenes linking to exists	Link to correct scenes / prompts showing	As expected	Success	Font and size to be updated
U_07	Pause button correctly working	Click on pause button	Level is running	Status of typewriter is constant, disabled when end prompt is layered	As expected	Success	N/A

### Integration Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
I_01	Scenarios and expression sprites are kept constant / validly showing	N/A	U_01	Scenario, expression matches on customer and order summary (in emotions prompt)	As expected	Success	Characters to be altered
I_02	Correct layering of prompts (customer, emotions prompt, pause and end prompt)	Clicking on show prompt button / pause button / finishing level	U_06, U_07	Layering is correct in the order of hierarchy (customer < emotions prompt < pause / end prompt )	As expected	Success	Transparency of prompts to be discussed
I_03	Choice of answers from emotions prompt linked with correct score	Clicking on options from emotions prompt	U_05	Increment hearts obtained for correct answer & correctness of choice logged in console	Scoring not yet implemented	Failure	To be implemented by Jiwon by week 12
I_04	Correct sequence for game flow	Playing the game and choosing options	I_02, U_01/2/4/5	Customer enter → Scenario is explained → Prompt is shown and options chosen → summary of level shown	As expected	Success	N/A

## Acceptance Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
A_01	As a player, finish each level to obtain certain / correct number of hearts.	Answers chosen from emotions prompt	U_03, U_05	Correctly shows number of hearts obtained in level & game summary			
A_02	Game should be composed of three stages, sequential to the previous level	N/A	N/A	Game runs sequentially			

A_03	When playing the game, can access hints and guides	Press on hints and guides button	Hints and guide button should be implemented (and updated on Unit testing)	Correct and valid hints and guides are shown appropriate to the situation given			
A_04	Game able to be compiled in .ipa file and deployable on iOS device	N/A	Scripts compile without errors	Game able to be run on iOS device with .ipa file as input			
A_05	When finishing games and obtaining hearts, local database is correctly modified to update hearts	Game is successfully finished with certain number of hearts obtained	U_03, A_01	When running game again, number of hearts is recorded and shown correctly			

# [231014] temporary\_branch

Date : 14th October, 2023

Tested by : Jiwon Bae

Branch : temporary\_branch

Commit Name (Hash) : heart implemented, halfway working through summary (1ab1dd1ab20ba51e239b9218f2c4354752484154)

## Unit Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
U_01	Valid choice of scenarios with expression sprites	N/A	File structure valid under Assets\Resourc es\Scenarios	Scene showing expression and text when game is simulating	Same as expected	Success	N/A
U_02	Valid and randomized choice of person sprite	N/A	Correct sprite set in Customer Component - Character Script	Person sprite seen when level is started	Same as expected	Success	N/A
U_03	Connection with database for hearts	Press +1 Heart button in Main Menu	Database file valid / space on local storage	Log showing incrementing number of hearts in database	Same as expected	Success	Temporary fix of creating a new database and incrementing heart
U_04	Correct implementation of typewriter for scenario	Left click on mouse	Valid access and open of text file	Start with typewriter skip ahead with left-click of mouse → should have full text shown	Same as expected	Success	N/A
U_05	Valid options for selecting emotions from prompts	N/A	File listing valid / appropriate emotions	No duplicate choices, all choices valid and appropriate emotions	Same as expected	Success	N/A
U_06	Main menu buttons / end level screen buttons with correct behavior	Click on buttons	Scenes linking to exists	Link to correct scenes / prompts showing	Same as expected	Success	N/A
U_07	Pause button correctly working	Click on pause button	Level is running	Status of typewriter is	Same as expected	Success	N/A

				constant, disabled when end prompt is layered			
--	--	--	--	--	--	--	--

## Integration Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
I_01	Scenarios and expression sprites are kept constant / validly showing	N/A	U_01	Scenario, expression matches on customer and order summary (in emotions prompt)	Printing only "I" when exiting to main menu and returning	Failure	Shanaia's followup on fixing this issue
I_02	Correct layering of prompts (customer, emotions prompt, pause and end prompt)	Clicking on show prompt button / pause button / finishing level	U_06, U_07	Layering is correct in the order of hierarchy (customer < emotions prompt < pause / end prompt )	Same as expected	Success	N/A
I_03	Choice of answers from emotions prompt linked with correct score	Clicking on options from emotions prompt	U_05	Increment hearts obtained for correct answer & correctness of choice logged in console	Scoring implemented successfully	Success	N/A
I_04	Correct sequence for game flow	Playing the game and choosing options	I_02, U_01/2/4/5	Customer enter → Scenario is explained → Prompt is shown and options chosen → summary of level shown	Same as expected	Success	N/A

# [231015] fixed\_typing testing

Date : 15th October, 2023

Tested by : Shanaia

Branch : *fixed\_typing*

Commit Name (Hash) : Customer SFX

(c34f98d843468a0387640043796cc879f1cab441)

## Unit Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
U_01	Valid choice of scenarios with expression sprites	N/A	File structure valid under Assets\Resourc es\Scenarios	Scene showing expression and text when game is simulating	Same as expected	Success	N/A
U_02	Valid and randomized choice of person sprite	N/A	Correct sprite set in Customer Component - Character Script	Person sprite seen when level is started	Same as expected	Success	N/A
U_03	Connection with database for hearts	Press +1 Heart button in Main Menu	Database file valid / space on local storage	Log showing incrementing number of hearts in database	Same as expected	Success	Temporary fix of creating a new database and incrementing heart
U_04	Correct implementation of typewriter for scenario	Left click on mouse	Valid access and open of text file	Start with typewriter skip ahead with left-click of mouse → should have full text shown	Same as expected	Success	N/A
U_05	Valid options for selecting emotions from prompts	N/A	File listing valid / appropriate emotions	No duplicate choices, all choices valid and appropriate emotions	Same as expected	Success	Make decision on format of invalid emotions
U_06	Main menu buttons / end level screen buttons with correct behavior	Click on buttons	Scenes linking to exists	Link to correct scenes / prompts showing	Same as expected	Success	N/A

U_07	Pause button correctly working	Click on pause button	Level is running	Status of typewriter is constant, disabled when end prompt is layered	Same as expected	Success	N/A
------	--------------------------------	-----------------------	------------------	---	------------------	---------	-----

## Integration Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
I_01	Scenarios and expression sprites are kept constant / validly showing	N/A	U_01	Scenario, expression matches on customer and order summary (in emotions prompt)	Same as expected	Success	The solution was to change WaitForSeconds to WaitForSecondsRealtime in TextScript.c
I_02	Correct layering of prompts (customer, emotions prompt, pause and end prompt)	Clicking on show prompt button / pause button / finishing level	U_06, U_07	Layering is correct in the order of hierarchy (customer < emotions prompt < pause / end prompt )	Same as expected	Success	N/A
I_03	Choice of answers from emotions prompt linked with correct score	Clicking on options from emotions prompt	U_05	Increment hearts obtained for correct answer & correctness of choice logged in console	Scoring not implemented yet	Failure	Jiwon - implement by week 12
I_04	Correct sequence for game flow	Playing the game and choosing options	I_02, U_01/2/4/5	Customer enter → Scenario is explained → Prompt is shown and options chosen → summary of level shown	Same as expected	Success	5 customers showing
I_05	Destroy customer and customer sfx before showing End Level Panel	Winning the level	I_04	Sounds should stop playing before the end level is shown	End Level shown while customer sfx is heard	Failure	N/A

- No acceptance testing as there is no complete executable



## [231015] fixed\_typing testing 2

Date : 15th October, 2023

Tested by : Shanaia

Branch : *fixed\_typing*

Commit Name (Hash) : Destroyed customer (and sfx) at the end level (7dbcde1fbc13f053efc3e2307f42a7d3d26100fa)

### Unit Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
U_01	Valid choice of scenarios with expression sprites	N/A	File structure valid under Assets\Resourc es\Scenarios	Scene showing expression and text when game is simulating	Same as expected	Success	N/A
U_02	Valid and randomized choice of person sprite	N/A	Correct sprite set in Customer Component - Character Script	Person sprite seen when level is started	Same as expected	Success	N/A
U_03	Connection with database for hearts	Press +1 Heart button in Main Menu	Database file valid / space on local storage	Log showing incrementing number of hearts in database	Same as expected	Success	Temporary fix of creating a new database and incrementing heart
U_04	Correct implementation of typewriter for scenario	Left click on mouse	Valid access and open of text file	Start with typewriter skip ahead with left-click of mouse → should have full text shown	Same as expected	Success	N/A
U_05	Valid options for selecting emotions from prompts	N/A	File listing valid / appropriate emotions	No duplicate choices, all choices valid and appropriate emotions	Same as expected	Success	Make decision on format of invalid emotions
U_06	Main menu buttons / end level screen buttons with correct behavior	Click on buttons	Scenes linking to exists	Link to correct scenes / prompts showing	Same as expected	Success	N/A
U_07	Pause button correctly working	Click on pause button	Level is running	Status of typewriter is	Same as expected	Success	N/A

				constant, disabled when end prompt is layered			
--	--	--	--	--	--	--	--

## Integration Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
I_01	Scenarios and expression sprites are kept constant / validly showing	N/A	U_01	Scenario, expression matches on customer and order summary (in emotions prompt)	Same as expected	Success	The solution was to change WaitForSeconds to WaitForSecondsRealtime in TextScript.c
I_02	Correct layering of prompts (customer, emotions prompt, pause and end prompt)	Clicking on show prompt button / pause button / finishing level	U_06, U_07	Layering is correct in the order of hierarchy (customer < emotions prompt < pause / end prompt )	Same as expected	Success	N/A
I_03	Choice of answers from emotions prompt linked with correct score	Clicking on options from emotions prompt	U_05	Increment hearts obtained for correct answer & correctness of choice logged in console	Scoring not implemented yet	Failure	Jiwon - implement by week 12
I_04	Correct sequence for game flow	Playing the game and choosing options	I_02, U_01/2/4/5	Customer enter → Scenario is explained → Prompt is shown and options chosen → summary of level shown	Same as expected	Success	5 customers showing
I_05	Destroy customer and customer sfx before showing End Level Panel	Winning the level	I_04	Sounds should stop playing before the end level is shown	Same as expected	Success	N/A

# [231018] main merge

Date : 18th October, 2023

Tested by : Jiwon Bae

Branch : *main*

Commit Name (Hash) : successful merge of main

(01cb9ee48762072a7132ec2e2e37095fb2b06543)

## Unit Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
U_01	Valid choice of scenarios with expression sprites	N/A	File structure valid under Assets\Resourc es\Scenarios	Scene showing expression and text when game is simulating	Same as expected	Success	N/A
U_02	Valid and randomized choice of person sprite	N/A	Correct sprite set in Customer Component - Character Script	Person sprite seen when level is started	Same as expected	Success	N/A
U_03	Connection with database for hearts	Press +1 Heart button in Main Menu	Database file valid / space on local storage	Log showing incrementing number of hearts in database	No database connection alarm	Fail	Temporary fix of creating a new database and incrementing heart
U_04	Correct implementation of typewriter for scenario	Left click on mouse	Valid access and open of text file	Start with typewriter skip ahead with left-click of mouse → should have full text shown	Same as expected	Success	N/A
U_05	Valid options for selecting emotions from prompts	N/A	File listing valid / appropriate emotions	No duplicate choices, all choices valid and appropriate emotions	Same as expected	Success	N/A
U_06	Main menu buttons / end level screen buttons with correct behavior	Click on buttons	Scenes linking to exists	Link to correct scenes / prompts showing	Same as expected	Success	N/A

U_07	Pause button correctly working	Click on pause button	Level is running	Status of typewriter is constant, disabled when end prompt is layered	Same as expected	Success	N/A
------	--------------------------------	-----------------------	------------------	---	------------------	---------	-----

## Integration Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
I_01	Scenarios and expression sprites are kept constant / validly showing	N/A	U_01	Scenario, expression matches on customer and order summary (in emotions prompt)	Printing "I" issue disappeared.	Success	N/A
I_02	Correct layering of prompts (customer, emotions prompt, pause and end prompt)	Clicking on show prompt button / pause button / finishing level	U_06, U_07	Layering is correct in the order of hierarchy (customer < emotions prompt < pause / end prompt )	Same as expected	Success	N/A
I_03	Choice of answers from emotions prompt linked with correct score	Clicking on options from emotions prompt	U_05	Increment hearts obtained for correct answer & correctness of choice logged in console	Scoring implemented successfully	Success	N/A
I_04	Correct sequence for game flow	Playing the game and choosing options	I_02, U_01/2/4/5	Customer enter → Scenario is explained → Prompt is shown and options chosen → summary of level shown	Same as expected	Success	N/A

## [231018] database-resilience testing

### Unit Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
U_01	Valid choice of scenarios with expression sprites	N/A	File structure valid under Assets\Resourc es\Scenarios	Scene showing expression and text when game is simulating	As expected	Success	NA
U_02	Valid and randomized choice of person sprite	N/A	Correct sprite set in Customer Component - Character Script	Person sprite seen when level is started	As expected	Success	NA
U_03	Connection with database for hearts	Press +1 Heart button in Main Menu	Database file valid / space on local storage	Log showing incrementing number of hearts in database	No output	Failure	Have disconnected this part, as is unnecessary
U_04	Correct implementation of typewriter for scenario	Left click on mouse	Valid access and open of text file	Start with typewriter skip ahead with left-click of mouse → should have full text shown	As expected	Success	NA
U_05	Valid options for selecting emotions from prompts	N/A	File listing valid / appropriate emotions	No duplicate choices, all choices valid and appropriate emotions	As expected	Success	NA
U_06	Main menu buttons / end level screen buttons with correct behavior	Click on buttons	Scenes linking to exists	Link to correct scenes / prompts showing	As expected	Success	NA
U_07	Pause button correctly working	Click on pause button	Level is running	Status of typewriter is constant, disabled when end prompt is layered	The typewrite/foot step sounds all keep playing even when I press pause	Failure	Need to ensure pause stops everything in the game.

### Integration Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
I_01	Scenarios and expression sprites are kept constant / validly showing	N/A	U_01	Scenario, expression matches on customer and order summary (in emotions prompt)	As expected	Success	NA
I_02	Correct layering of prompts (customer, emotions prompt, pause and end prompt)	Clicking on show prompt button / pause button / finishing level	U_06, U_07	Layering is correct in the order of hierarchy (customer < emotions prompt < pause / end prompt )	As expected	Success	NA
I_03	Choice of answers from emotions prompt linked with correct score	Clicking on options from emotions prompt	U_05	Increment hearts obtained for correct answer & correctness of choice logged in console	As expected	Success	NA
I_04	Correct sequence for game flow	Playing the game and choosing options	I_02, U_01/2/4/5	Customer enter → Scenario is explained → Prompt is shown and options chosen → summary of level shown	As expected	Success	NA

## Acceptance Testing

Epic	User Story	Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
The game facilitates player progression	As a client, I want the game to award players with a currency (hearts) based	A_01	As a player, finish each level to obtain certain / correct number of hearts.	Answers chosen from emotions prompt	U_03, U_05	Correctly shows number of hearts obtained in level & game summary	Does not show end number of hearts obtained	Failure	Are we implementing this?

	on their engagement								
The game facilitates player progression	As a player, I want the game to have 3 stages that is clear in what it is trying to achieve	A_02	Game should be composed of three stages, sequential to the previous level	N/A	N/A	Game runs sequentially	Only one level	Failure	Not yet implemented
The game facilitates player progression	As a player, I want the game to have some hints and guides for me to follow when I'm struggling	A_03	When playing the game, can access hints and guides	Press on hints and guides button	Hints and guide button should be implemented (and updated on Unit testing)	Correct and valid hints and guides are shown appropriate to the situation given	No hints and guide button	Failure	Not yet implemented
The game integrates into broader application hub designed by masters' students	As a Masters Student (Stakeholder), I want to have the game be executable as a .ipa	A_04	Game able to be compiled in .ipa file and deployable on iOS device	N/A	Scripts compile without errors	Game able to be run on iOS device with .ipa file as input	NA	Not tested	NA
The game integrates into broader application hub designed by masters' students	As a parent/guardian, I want the game to send performance data to the main application about how the child is performing in the games so far	A_05	When finishing games and obtaining hearts, local database is correctly modified to update hearts	Game is successfully finished with certain number of hearts obtained	U_03, A_01	When running game again, number of hearts is recorded and shown correctly	As expected	Success	NA

## [231019] final testing in Unity application

### Unit Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
U_01	Valid choice of scenarios with expression sprites	N/A	File structure valid under Assets\Resourc es\Scenarios	Scene showing expression and text when game is simulating	As expected	Success	NA
U_02	Valid and randomized choice of person sprite	N/A	Correct sprite set in Customer Component - Character Script	Person sprite seen when level is started	As expected	Success	NA
U_03	Connection with database for hearts	Press +1 Heart button in Main Menu	Database file valid / space on local storage	Log showing incrementing number of hearts in database	No output	Failure	Have disconnected this part, as is unnecessary
U_04	Correct implementation of typewriter for scenario	Left click on mouse	Valid access and open of text file	Start with typewriter skip ahead with left-click of mouse → should have full text shown	As expected	Success	NA
U_05	Valid options for selecting emotions from prompts	N/A	File listing valid / appropriate emotions	No duplicate choices, all choices valid and appropriate emotions	As expected	Success	NA
U_06	Main menu buttons / end level screen buttons with correct behavior	Click on buttons	Scenes linking to exists	Link to correct scenes / prompts showing	As expected	Success	NA
U_07	Pause button correctly working	Click on pause button	Level is running	Status of typewriter is constant, disabled when end prompt is layered	As expected	Success	NA

### Integration Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
I_01	Scenarios and expression sprites are kept constant / validly showing	N/A	U_01	Scenario, expression matches on customer and order summary (in emotions prompt)	As expected	Success	NA
I_02	Correct layering of prompts (customer, emotions prompt, pause and end prompt)	Clicking on show prompt button / pause button / finishing level	U_06, U_07	Layering is correct in the order of hierarchy (customer < emotions prompt < pause / end prompt )	As expected	Success	NA
I_03	Choice of answers from emotions prompt linked with correct score	Clicking on options from emotions prompt	U_05	Increment hearts obtained for correct answer & correctness of choice logged in console	As expected	Success	NA
I_04	Correct sequence for game flow	Playing the game and choosing options	I_02, U_01/2/4/5	Customer enter → Scenario is explained → Prompt is shown and options chosen → summary of level shown	As expected	Success	NA

## Acceptance Testing

Epic	User Story	Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
The game facilitates player progression	As a client, I want the game to award players with a currency (hearts) based	A_01	As a player, finish each level to obtain certain / correct number of hearts.	Answers chosen from emotions prompt	U_03, U_05	Correctly shows number of hearts obtained in level & game summary	Correct scoring system	Success	Have removed game summary part (but shows in level)

	on their engagement								
The game facilitates player progression	As a player, I want the game to have 3 stages that is clear in what it is trying to achieve	A_02	Game should be composed of three stages, sequential to the previous level	N/A	N/A	Game runs sequentially	As expected	Success	NA
The game integrates into broader application hub designed by masters' students	As a Masters Student (Stakeholder), I want to have the game be executable as a .ipa	A_04	Game able to be compiled in .ipa file and deployable on iOS device	N/A	Scripts compile without errors	Game able to be run on iOS device with .ipa file as input	NA	Not tested	NA
The game integrates into broader application hub designed by masters' students	As a parent/guardian, I want the game to send performance data to the main application about how the child is performing in the games so far	A_05	When finishing games and obtaining hearts, local database is correctly modified to update hearts	Game is successfully finished with certain number of hearts obtained	U_03, A_01	When running game again, number of hearts is recorded and shown correctly	As expected	Success	NA

## [231024] finalising game

### Unit Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
U_01	Valid choice of scenarios with expression sprites	N/A	File structure valid under Assets\Resources\Scenarios	Scene showing expression and text when game is simulating	As expected	Success	 See screenshots: Scenario and sprite appropriately appears.
U_02	Valid and randomized choice of person sprite	N/A	Correct sprite set in Customer Component - Character Script	Person sprite seen when level is started	As expected	Success	NA
U_04	Correct implementation of typewriter for scenario	Left click on mouse	Valid access and open of text file	Start with typewriter skip ahead with left-click of mouse → should have full text shown	As expected	Success	NA
U_05	Valid options for selecting emotions from prompts	N/A	File listing valid / appropriate emotions	No duplicate choices, all choices valid and appropriate emotions	As expected	Success	See screenshot: All choices valid and relevant 
U_06	Main menu buttons / end level screen buttons with correct behavior	Click on buttons	Scenes linking to exists	Link to correct scenes / prompts showing	As expected	Success	NA
U_07	Pause button correctly working	Click on pause button	Level is running	Status of typewriter is constant, disabled when end prompt is layered	As expected	Success	 See video--music appropriately pauses/unpauses

U_08	Volume settings correctly working	Slide on volume sliders	Music is set at some volume	Volume adjusts as expected	As expected	Success	Background volume successfully adjusts 
------	-----------------------------------	-------------------------	-----------------------------	----------------------------	-------------	---------	---

## Integration Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
I_01	Scenarios and expression sprites are kept constant / validly showing	N/A	U_01	Scenario, expression matches on customer and order summary (in emotions prompt)	As expected	Success	NA
I_02	Correct layering of prompts (customer, emotions prompt, pause and end prompt)	Clicking on show prompt button / pause button / finishing level	U_06, U_07	Layering is correct in the order of hierarchy (customer < emotions prompt < pause / end prompt )	As expected	Success	Some sprites intersect slightly with the emotion prompt, but functionally emotion prompt is appropriately layered and never obscured 
I_03	Choice of answers from emotions prompt linked with correct score	Clicking on options from emotions prompt	U_05	Increment hearts obtained for correct answer & correctness of choice logged in console	As expected	Success	NA
I_04	Correct sequence for game flow	Playing the game and choosing options	I_02, U_01/2/4/5	Customer enter → Scenario is explained → Prompt is shown and options chosen →	As expected	Success	NA

					summary of level shown			
--	--	--	--	--	------------------------	--	--	--

## Acceptance Testing

Epic	User Story	Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
The game facilitates player progression	The game to award players with a currency (hearts) based on their engagement	A_01	As a player, finish each level to obtain certain / correct number of hearts.	Answers chosen from emotions prompt	U_03, U_05	Correctly shows number of hearts obtained in level & game summary	Correct scoring system	Success	Have removed game summary part (but shows in level)
The game facilitates player progression	The game to have 3 stages that is clear in what it is trying to achieve	A_02	Game should be composed of three stages, sequential to the previous level	N/A	N/A	Game runs sequentially	As expected	Success	NA
The game integrates into broader application hub designed by masters' students	To have the game be executable as a .ipa	A_04	Game able to be compiled in .ipa file and deployable on iPhone device	N/A	Scripts compile without errors	Game able to be run on iOS device with .ipa file as input	NA	Success	NA
The game integrates into broader application hub designed by masters' students	The game to send performance data to the main application about how the child is performing in the games so far	A_05	When finishing games and obtaining hearts, local database is correctly modified to update hearts	Game is successfully finished with certain number of hearts obtained	U_03, A_01	When running game again, number of hearts is recorded and shown correctly	As expected	Success	NA
The game integrates into broader application hub	To have the game be executable as a .ipa	A_06	Game scales to any iPhone device	N/A	Game device set to specific iPhone	Game scales to device correctly	As expected	Success	NA

designed by masters' students					type (e.g. SE)				
The game facilitates player progression	The game to have 3 stages that is clear in what it is trying to achieve	A_07	The game runs for only 3 specific stages that increase progressively in difficulty	N/A	Game being played	Game finishes after 3 stages which each get slightly more complex	As expected	Success	N/A

# 🧪 Testing Template

## Unit Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
U_01	Valid choice of scenarios with expression sprites	N/A	File structure valid under Assets\Resourses\Scenarios	Scene showing expression and text when game is simulating			
U_02	Valid and randomized choice of person sprite	N/A	Correct sprite set in Customer Component - Character Script	Person sprite seen when level is started			
U_04	Correct implementation of typewriter for scenario	Left click on mouse	Valid access and open of text file	Start with typewriter skip ahead with left-click of mouse → should have full text shown			
U_05	Valid options for selecting emotions from prompts	N/A	File listing valid / appropriate emotions	No duplicate choices, all choices valid and appropriate emotions			
U_06	Main menu buttons / end level screen buttons with correct behavior	Click on buttons	Scenes linking to exists	Link to correct scenes / prompts showing			
U_07	Pause button correctly working	Click on pause button	Level is running	Status of typewriter is constant, disabled when end prompt is layered			

U_08	Volume and typing speed settings correctly working	Slide on volume and typing sliders	Music is playing	Volume and typing speed are adjusted as expected			
------	--	------------------------------------	------------------	--	--	--	--

## Integration Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
I_01	Scenarios and expression sprites are kept constant / validly showing	N/A	U_01	Scenario, expression matches on customer and order summary (in emotions prompt)	As expected	Success	NA
I_02	Correct layering of prompts (customer, emotions prompt, pause and end prompt)	Clicking on show prompt button / pause button / finishing level	U_06, U_07	Layering is correct in the order of hierarchy (customer < emotions prompt < pause / end prompt )	As expected	Success	NA
I_03	Choice of answers from emotions prompt linked with correct score	Clicking on options from emotions prompt	U_05	Increment hearts obtained for correct answer & correctness of choice logged in console	As expected	Success	NA
I_04	Correct sequence for game flow	Playing the game and choosing options	I_02, U_01/2/4/5	Customer enter → Scenario is explained → Prompt is shown and options chosen → summary of level shown	As expected	Success	NA

## Acceptance Testing

Test Case ID	Test Case Description	Input Values	Precondition	Expected Outputs	Actual Outputs	Test Result (Success / Failure)	Comments
A_01	As a player, finish each level to obtain certain /	Answers chosen from	U_03, U_05	Correctly shows number of hearts			

	correct number of hearts.	emotions prompt		obtained in level & game summary			
A_02	Game should be composed of three stages, sequential to the previous level	N/A	N/A	Game runs sequentially			
A_04	Game able to be compiled in .ipa file and deployable on iOS device	N/A	Scripts compile without errors	Game able to be run on iOS device with .ipa file as input			
A_05	When finishing games and obtaining hearts, local database is correctly modified to update hearts	Game is successfully finished with certain number of hearts obtained	U_03, A_01	When running game again, number of hearts is recorded and shown correctly			
A_06	Game scales to any iPhone device	N/A	Game device set to specific iPhone type (e.g. SE)	Game scales to device correctly			

## Deployment

### Deployment Plan

 Deployment plan

## Handover

 Handover

# Deployment plan

Our game is meant to be played on an iPhone, but it is also useful to test on a desktop. Therefore, we have included instructions on how to deploy the game to an iPhone, a Mac and a Windows desktop

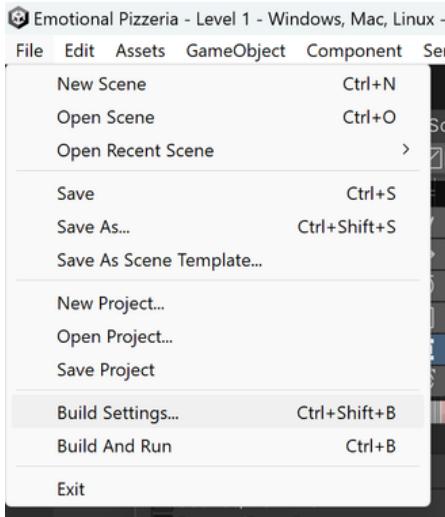
## Deployment to iPhone

To be able to create a .ipa file and submit it to the Apple App Store, please ensure you have:

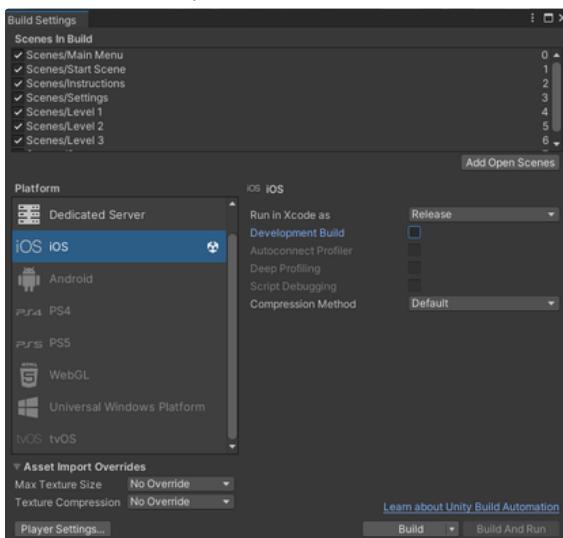
- a Mac with XCode installed
- an Apple Developer Account
- Enrolled in the Apple Developer Program

This will result in a .ipa file. These steps assume you don't have a 'IOSBuild' folder in the provided codebase. If you do, skip to step 7

1. Download Unity at [Download](#), making sure to install the latest editor version
2. Clone the project at <https://github.com/SiRong-github/-TODO-Create-team-name>
3. Through Unity, open the project by opening the Emotional Pizzeria file
4. Go to File → Build Settings

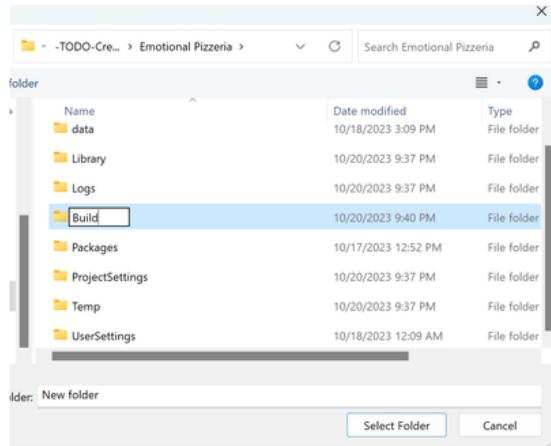


5. Choose IOS as the platform, and click the Switch Platform button if not already done



You may need to press the 'Switch Platform' button before being able to build!

6. Click the Build button, name the destination folder 'IOSBuild' and place it in the Emotional Pizzeria folder



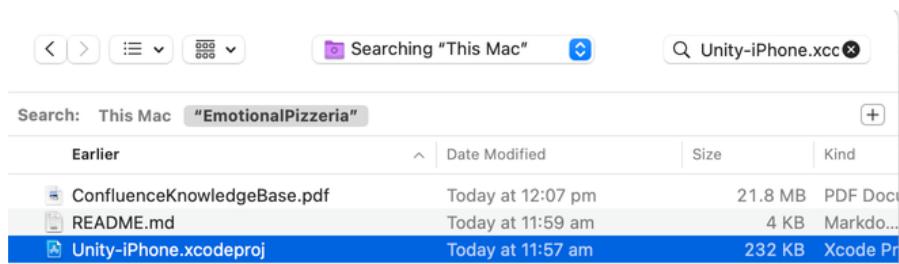
7. In your Mac, open the XCode application.

8. In the application, click "Open Existing Project..." and head to the downloaded GitHub project.

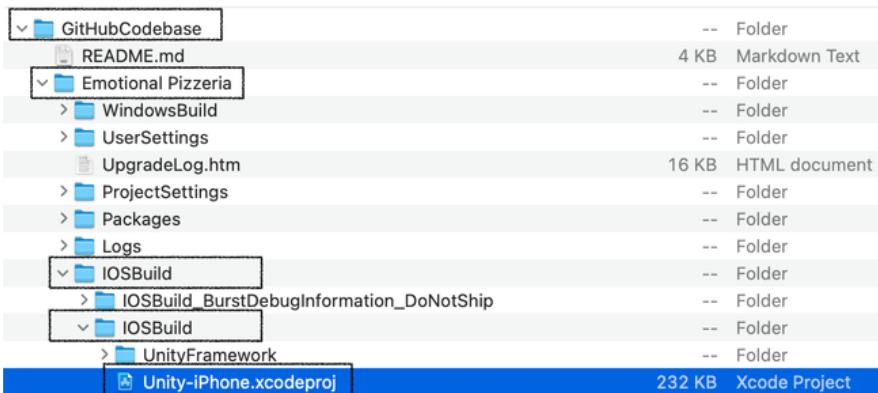


9. In the EmotionalPizzeria folder, search for and click on the Unity-iPhone.xcodeproj.

- You may do this by searching directly using the search bar. However, you must ensure you are searching inside "EmotionalPizzeria" and not "This Mac" or "Network".

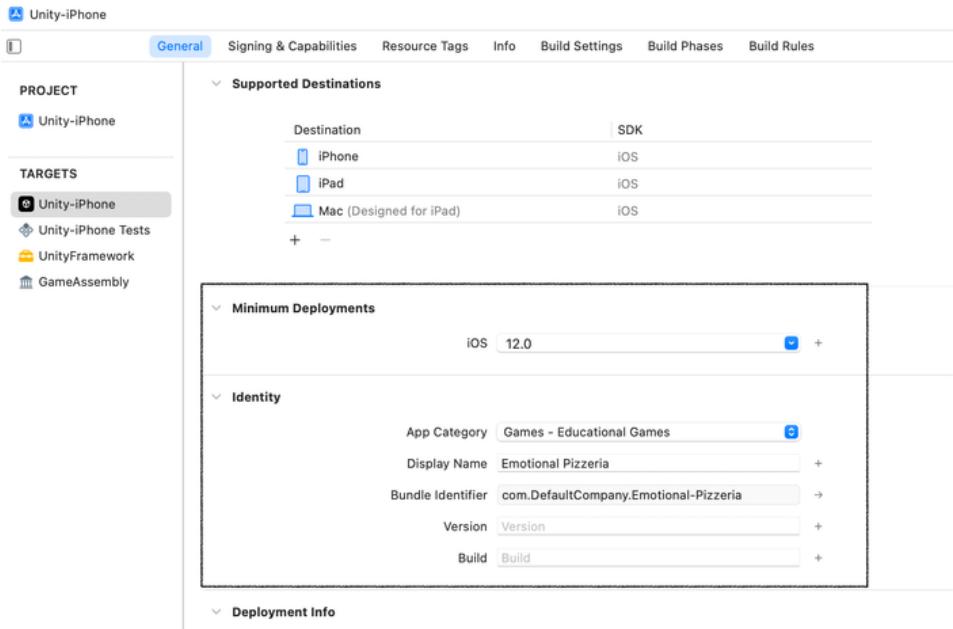


- Or you may also do this by going to GitHubCodebase → Emotional Pizzeria → IOSBuild → IOSBuild → Unity-iPhone.xcodeproj

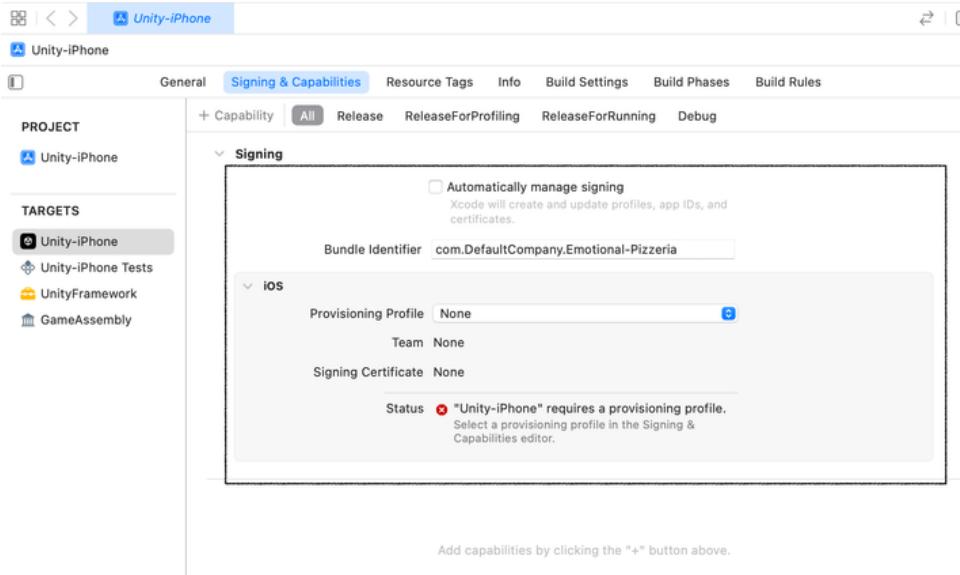


10. In XCode, if the editor is empty ("No Editor"), head to the Project Navigator which is at the leftmost side of the window and click on Unity-iPhone

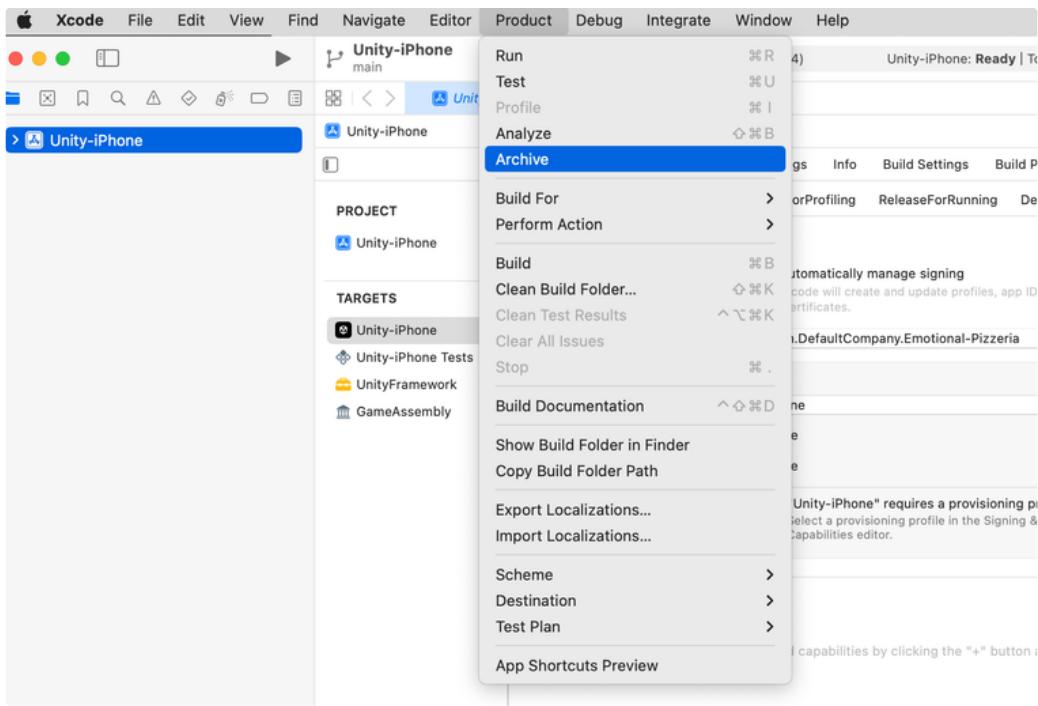
11. In the editor under the General tab, fill in your details in the Identity section.
- In the Minimum Deployments section, select the minimum required IOS version.
  - In the Identity section, fill in the Bundle Identifier, Version and Build.



12. In the same editor, head to the Signing & Capabilities tab. Add your Apple Developer Account and fill in the necessary details.
- You may tick on the Automatically manage signing checkbox for XCode to handle the provisioning profiles and certificates.
  - You may enter these manually according to your Apple Developer Accounts



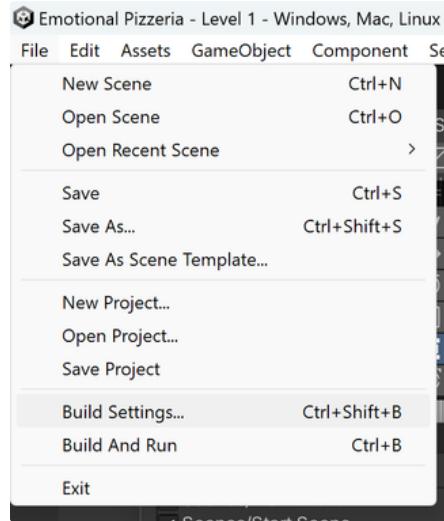
13. In the toolbar, head to Product → Archive.



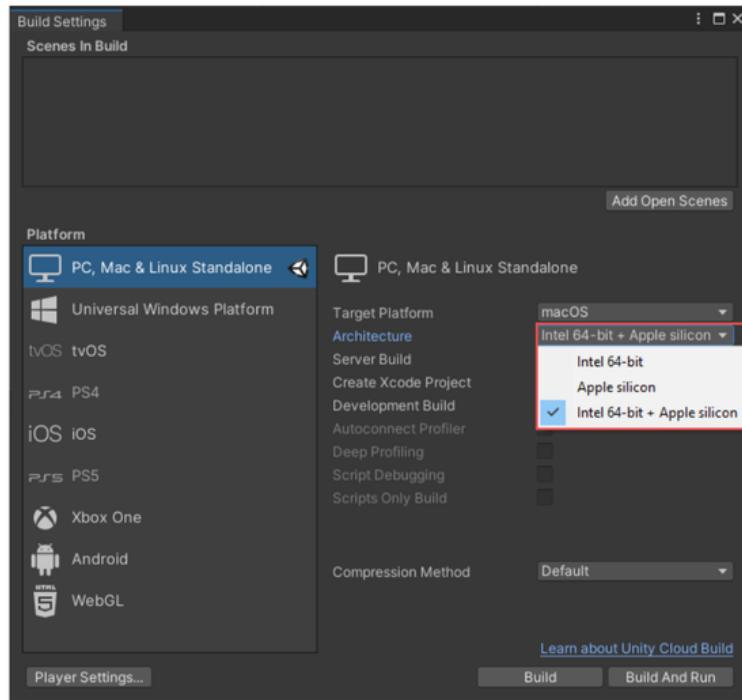
## Deployment to a Mac

This will result in an Application file. These steps assume you don't have a 'MacBuild' folder in the provided codebase. If you do, skip to step 7

1. Download Unity at [Download](#), making sure to install the latest editor version
2. Clone the project at <https://github.com/SiRong-github/-TODO-Create-team-name> Connect your Github account
3. Through Unity, open the project by opening the Emotional Pizzeria file
4. Go to File → Build Settings

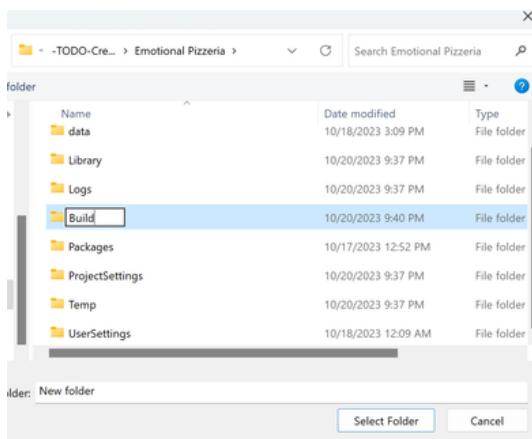


5. Choose 'PC, Mac, and Linux Standalone' as the platform, and click the Switch Platform button (bottom right) if not already done



You may need to press the 'Switch Platform' button before being able to build!

6. Click the Build button, name the destination folder 'MacBuild' and place it in the Emotional Pizzeria folder



7. Inside the Build folder, you will find the Application 'MacBuild'

Name	Date Modified	Size	Kind
MacBuild	Yesterday at 8:35 am	170.4 MB	Application
MacBuild_BurstDeb...formation_DoNotShip	Yesterday at 8:36 am	--	Folder

You will notice some additional files aside from the Application-- the only thing you need to do to test the game as an Application is simply run this Application

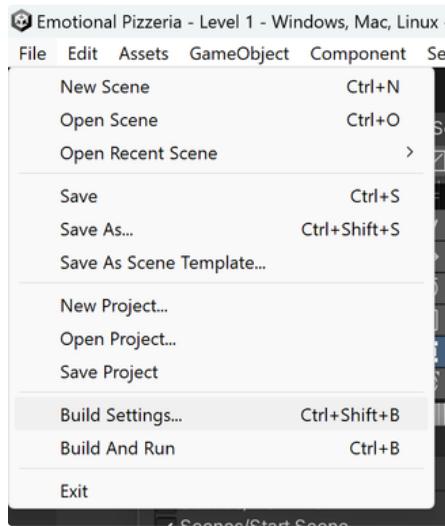
## Deployment to a Windows desktop

This will result in a .exe file. These steps assume you don't have a 'WindowsBuild' folder in the provided codebase. If you do, skip to step 7

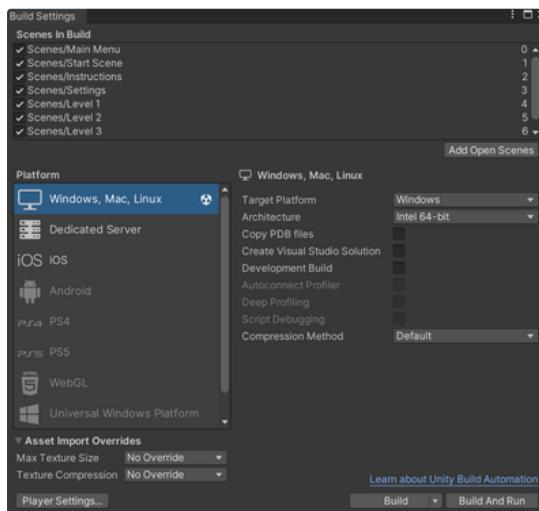
1. Download Unity at [Download](#), making sure to install the latest editor version
2. Clone the project at <https://github.com/SiRong-github/-TODO-Create-team-name> Connect your Github account

3. Through Unity, open the project by opening the Emotional Pizzeria file

4. Go to File → Build Settings

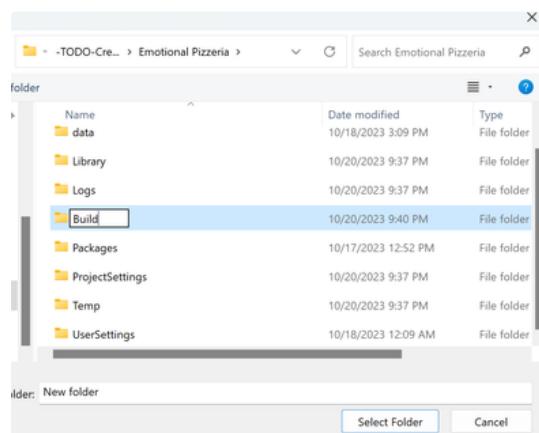


5. Choose 'PC, Mac, and Linux Standalone' as the platform, and click the Switch Platform button (bottom right) if not already done



You may need to press the 'Switch Platform' button  
before being able to build!

6. Click the Build button, name the destination folder 'WindowsBuild' and place it in the Emotional Pizzeria folder



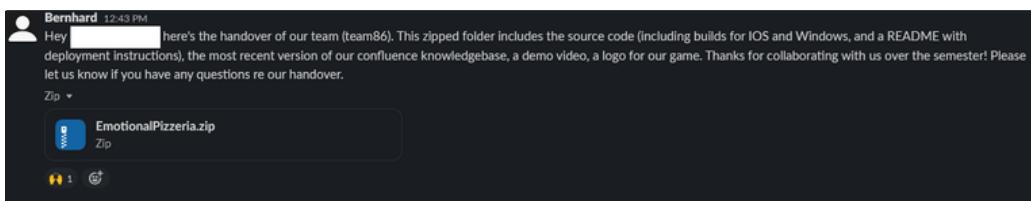
7. Inside the Build folder, you will find the executable 'EmotionalPizzeria.exe'

Name	Date modified	Type
Emotional Pizzeria_BurstDebugInformati...	13/09/2023 11:05 PM	File folder
Emotional Pizzeria_Data	13/09/2023 11:05 PM	File folder
MonoBleedingEdge	13/09/2023 11:05 PM	File folder
Emotional Pizzeria.exe	13/09/2023 11:05 PM	Application
UnityCrashHandler64.exe	13/09/2023 11:05 PM	Application
UnityPlayer.dll	13/09/2023 11:05 PM	Application exten...

You will notice some additional files aside from the executable-- the only thing you need to do to test the game as an executable is simply run this .exe

# Handover

As of the 25th of October, we have handed over all significant artefacts to the main point of contact for the Masters' students in our collaborative Slack, who have submitted our core deliverables to the client, James Marshall.



## Handover Documents

The below `.zip` file from the Google drive link contains the documents handed over in the above message:

[https://drive.google.com/file/d/1pZdk\\_c5cXVI936QLF7SzBkiPj8VKcFHs/view?usp=share\\_link](https://drive.google.com/file/d/1pZdk_c5cXVI936QLF7SzBkiPj8VKcFHs/view?usp=share_link)

This included the following:

- A `.pdf` with referring to all pages within this Confluence (including any documentation produced).
- A demo video of the game being played and a `.png` of a game logo which can be used when submitting the game to the app store.

Further, it also contains the folder `GitHubCodebase`, which contains a copy of our final game build.

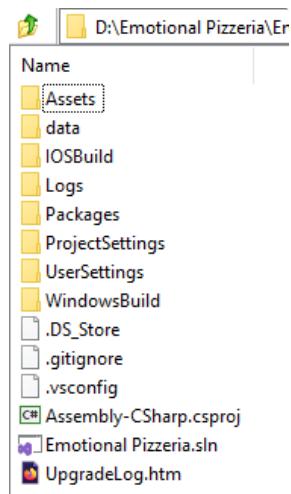
D:\Emotional Pizzeria\EmotionalPizzeria.zip\EmotionalPizzeria\					
Name	Size	Packed Size	Modified	C	
GitHubCodebase	1 381 305 351	553 017 422	2023-10-25 11:59		
GameLogo.png	313 766	312 905	2023-10-25 11:07		
DemoVideo.mp4	11 909 165	11 073 362	2023-10-25 11:07		
ConfluenceKnowledgeB...	21 798 159	21 482 228	2023-10-25 12:07		

All documents should be viewable in the `.zip` file.

Inside of `GitHubCodebase` is a folder `Emotional Pizzeria` which can be loaded as a Unity project and edited as required.

Inside of this folder, you will find:

- An iOS build of the game.
- A Mac build of the game (for bug testing and stability purposes) under the folder 'MacBuild'. If you would like to test the final game version, simply run the `MacBuild` Application file in this folder (see screenshots in the deployment plan for deploying to a Mac, step 7).
- A Windows build of the game (for bug testing and stability purposes) under the folder 'Windows Build'. If you would like to test the final game version, simply run the `Emotional Pizzeria.exe` file in this folder (see screenshots in the deployment plan for deploying to a Windows desktop, step 7).
- All associated assets (sprites, emotion prompts, backgrounds etc...)
- A data base file pointing to the local storage of the 'hearts' currency as per our requirements. As of now, the game will automatically update this database with hearts as the game is played for integration with the larger application hub. This is located in the `data` folder.
- Any other associated C# scripts or miscellaneous components of the game.



Opening the 'Emotional Pizzeria' folder in the 'GitHubCodebase' folder should greet you with the above.