

**SOFTWARE DESIGN AND DEVELOPMENT  
(SDD)**

**HSC COURSE**

**DR. ZANG**

**KELLYVILLE HIGH SCHOOL**

**ARAV ZAD**

**TASK 1, PROGRAMMING PROJECT REPORT**

**BRAIN BEAT**



## **TABLE OF CONTENTS**

<b>DEFINING THE PROBLEM</b>	<b>3</b>
<b>UNDERSTANDING THE PROBLEM</b>	<b>7</b>
<b>PLANNING SOFTWARE SOLUTIONS</b>	<b>18</b>
<b>DESIGNING SOFTWARE SOLUTIONS</b>	<b>22</b>
<b>REFERENCES</b>	<b>25</b>

## **DEFINING THE PROBLEM**

### *DESCRIPTION OF THE ENVIRONMENT*

The environment includes all external factors that can interact with the game but are not directly controlled by it. The environment consists of the player, who interacts with the game through the user interface. It includes the device on which the game will be played on, in this case, a computer. In detail, the Operating System of the Computer, and any potential software that could interact with the game, for example screen recording or live streaming. The interface between the game (system) and its environment would be the points of interaction between the game and the player (input and output), and between the game and the device's operating system. For example, the player inputs their response to the game sequence through mouse clicks and the game outputs the sequence and feedback through the screen and speakers. The game also interacts with the operating system for tasks like saving progress or accessing system resources.

### *TARGET AUDIENCE*

Since this game involves visual aspects (Yellow, Green, Red, Blue buttons) and auditory aspects (Different instrumental notes played once a visual aspect is clicked), scientifically it is advised that children, teenagers, adults, and the elderly play this game, making it a game that is suitable and aimed for all ages.

Playing memory games can improve vital brain functions, such as attention, concentration, and focus. Memory games give space to critical thinking and that helps children nurture their attention to detail. This target for children aged from 5-12 is achieved through the **Education Mode**.

Once children have matured into their adolescent age, they require a sense of challenge in their lives amongst peers and themselves individually. To cater for this need, and continue cognitive brain exercise, the target audience for teenagers aged between 13-18 and professional gamers is achieved through a **Challenge Mode**.

Finally, for adults and the elderly, **Normal Mode** is to be developed so that it caters the needs of them going through daily stress, for example providing for family, or other health harming aspects. Studies also show that adults who play memory games frequently every once in a while, lower the chances for dementia in senior citizens. Therefore the target audience of aged between the interval of 25+ is also catered for in this Software Product.

Therefore, the Target Audience for *Brain Beat* is catered for all, due to scientific reasoning, making the Target Audience linked to contemporary issues for this Software Product.

	EDUCATION MODE	NORMAL MODE	CHALLENGE MODE
What?	Cognitive and Instruction Based for Schooling Purposes. No Time Limit in Gameplay Looks more 'kiddish', with "Kid Modules" (Animals, Col..)	For Casual Gameplay. Linear Time Limit. (Consistent) Standard RGBY	For Intense Gameplay. Exponentially Decreasing Time Limit. (After every Sequence, smaller) Flashy/Distracting
Benefits	Teachers can use as an educational tool for fine motor skills and quick thinking activity.	Can be used to de-stress and relax, OR to warm up the brain for other activities.	Socially Captivating, Brag about High Score, Skill Based Game
Age Group	5 -12	Adults and Elderly	Teens, Young Adults

- Here is a Table that shows the benefits of what each game mode is and what it is.

#### *CRITICAL ANALYSIS OF EXISTING PRODUCT*

The Simon Memory Game, a classic electronic game, has been revered for its simplicity, challenge, and entertainment value since its inception in the late '70s. This game, played by a wide audience, primarily targets individuals seeking mental stimulation, memory enhancement, or enjoyable leisure time.

Its strengths lie in its straightforward gameplay and capacity to engage various age groups. The game enhances memory, cognitive skills, and focus by challenging players to recall and replicate complex patterns of light and sound.

While its simplicity is a key attraction, it might present limitations for users seeking more varied or advanced gameplay experiences. The basic pattern repetition may become monotonous over extended play, potentially limiting its appeal to users seeking diverse gaming mechanics or advanced features.

Existing iterations of the Simon Memory Game have modernised with enhanced aesthetics, additional gameplay modes, and digital adaptations for mobile platforms, appealing to tech-savvy users seeking portable gaming experiences.

Socially and ethically, the game promotes healthy competition, mental agility, and social interaction in multiplayer settings. However, overindulgence or obsessive gameplay might lead to addiction or neglect of other responsibilities, warranting moderation and balance.

From a programming perspective, creating the Simon Memory Game involves graphical user interface (GUI) design, sound integration, and coding the logic behind pattern generation and user input.

Overall, while the Simon Memory Game remains an iconic and engaging brain-teaser, its enduring appeal may benefit from occasional updates to cater to evolving gaming preferences and technological advancements.

## *NEEDS, OBJECTIVES, AND BOUNDARIES*

### **Needs -**

A need is an instance in which some necessity or want exists. Needs of the Software Product for the Target Audience and the Developer include:

Potential Customer Needs	Developer of <i>Brain Beat (Game Developer)</i>
To access a Game / Software easily.	To make a game that achieves a high standard
To gain certain scientific value [cognitive exercise] from playing the game consciously (explained in 'Target Audience' pp. 2-3)	To fulfil customers' needs (Minimal Loading Time, High quality user interface, achieve scientific cognitive needs as well)
To gain pure enjoyment from a game	To make sure that the game runs on Mac OS / Windows Operating System
To have minimal loading time	Scoring System which rewards the player
To have (consistent / gradually harder) audio and visuals depending on user	Progression system which increases the game's difficulty as it progresses

### **Objectives -**

Objectives are the short & long term aims & plans for the software being developed.

The objective for Brain Beat consist of the following:

- To provide an entertaining and engaging gaming experience.
- To challenge and improve the player's short-term memory skills.
- To ensure the game runs smoothly and reliably on a computer regardless of Operating System.
- To create a game that is easy to understand but difficult to master, thereby encouraging repeated play, formulating a healthy addiction

## **Boundaries -**

Boundaries define the limits of the problem or system to be developed. Anything outside the system is said to be a part of the environment. The system interacts with its environment via an interface. In this situation, the Boundaries of the ‘Brain Beat’ Program include the following:

- The Sequence Generation Logic (Cannot be altered by the Player)
- The Scoring System
- The Progression System

## *LEGAL, SOCIAL, AND ETHICAL ISSUES*

As Brain Beat is inspired from the Simon Memory Game, it is obvious that some aspects of the GUI and the Game may seem identical. However that is not the case. Social and Ethical Considerations are taken into consideration whilst crafting Brain Beat and here are the following alterations. The changing nature of work for the users is that instead of a normal RGBY game with audio, Brain Beat has an Education Mode, where an Animal Mode is present. This includes Snakes and Lions, etc, making the noise of a “Ssss” or “Roar” once clicked to make it more visually and cognitively appealing. The effects on the Public are the same as Simon. And that is to create a game which depletes all levels of stress and increases overall brain performance. Due to this, the Software is free, Brain Beat falls under the category of **Shareware**. The reason being that the game is not suitable for public modifications and reproduction of the product, where derivatives are made. Therefore, the Software is covered by Copyright, as in the market, there are not many games that explore this type of game mode and functioning. However, copies can be made for archival and distribution purposes. But, as mentioned earlier, modifications and reverse engineering is not permitted as it may defeat the purpose of the game, to be for the goodness of humanity. However, if there are inquiries, the public is allowed to ask for permission, permission may be granted after careful consideration.

## *POSSIBLE PROGRAMMING LANGUAGES TO DEVELOP*

The main programming language which will be used is Python. This is due to my prior skills in the language which have been honed across the years. I am also using Python due to its easy english-like syntax, user-friendly data structures and flexible code modules. Under Python, there will be plenty of add ons which will be required for the code to work, such as Pygame, Pysimple GUI, Tkinter, and PIL for images processing. I will also be using Photoshop, (PSD), to craft and design the PLAN of the user interface for the project. This will include the extensive use of colours, blending and polishing the game experience users will achieve from playing Brain Beat.

## **UNDERSTANDING THE PROBLEM**

### ***DATA DICTIONARY***

**\*Attached on the Google Classroom\* - Named “Data Dictionary”**

### ***IPO TABLES***

#### **Brain Beat**

INPUT	PROCESS	OUTPUT
Player clicks on “BrainBeat.exe”	<u>HomeScreen</u> <u>SelectGameMode</u> Whichever selected- <u>EduMode</u> / <u>NormMode</u> / <u>ChallMode</u>  (Maneuverings and UI-Friendly Options throughout)	Success

#### **HomeScreen**

##### HomeScreen>HowToPlay

I	P	O
“?” Button is clicked	Load “howtoplay.txt”	Instructions on How to Play the Game
	Load “SFX.mp3”	Sound effect once button clicked

##### HomeScreen>Settings

I	P	O
Settings Button Clicked	Load <u>ClearHighScore</u>	
	Load <u>SoundVol</u>	
	Load <u>SFXVol</u>	
	Load <u>ChangeCol</u>	All Settings Options

### HomeScreen>Settings>ClearHighScore

I	P	O
Clicks <u>ClearHighScore</u>	Set <u>HighScore</u> to 0	Pop up GUI “High Score cleared!”

### HomeScreen>Play

I	P	O
Clicks “Play”	Load <u>SelectGameMode</u>	
	Terminate Current Window	<u>SelectGameMode</u>

### HomeScreen>Settings>ChangeCol

I	P	O
Amount of RGB for each button in Normal and Challenge Mode	Change required button according to user preference colour by altering with its HEX RGB Code	Colour of Buttons changed

### SelectGameMode

#### SelectGameMode>EduMode

I	P	O
Player clicks EduMode	Load <u>EduAnimal</u>	
	Load <u>EduColour</u>	
	Choose between 1	Open correct Education Module

#### SelectGameMode>NormMode

I	P	O
Player clicks NormMode	Load <u>NormMode</u>	

	Terminate <u>SelectGameMode</u>	Start playing Normal Mode
--	---------------------------------	---------------------------

#### SelectGameMode>ChallMode

I	P	O
Player clicks ChallMode	Load <u>ChallMode</u>	
	Terminate <u>SelectGameMode</u>	Start playing Challenge Mode

#### SelectGameMode>HomeScreen

I	P	O
Player clicks Home Button	Load <u>HomeScreen</u>	
	Terminate <u>SelectGameMode</u>	PLayer on Home Screen

#### EduMode

##### EduMode>EduAnimal

StepNo.	I	P	O
1	Player clicks EduAnimal	Load <u>EduAnimalGUI</u>	
2		Order = 1	
3		Randomise 1st sequence	
4		Store 1st sequence in CompSet(Order)	Display 1st sequence with medium of <u>EduAnimalGUI</u>
5	Get Player Sequence	Store Player Sequence in PlayerSet(Order)	
6		WHILE PlayerSet(Order) = CompSet(Order): Increment Order	

7		REPEAT PROCESS FROM StepNo. 3	
8		IF PlayerSet(Order) != CompSet(Order):	"Game Over"

### EduMode>EduColour

StepNo.	I	P	O
1	Player clicks EduAnimal	Load <u>EduColourGUI</u>	
2		Order = 1	
3		Randomise 1st sequence	
4		Store 1st sequence in CompSet(Order)	Display 1st sequence with medium of <u>EduColourGUI</u>
5	Get Player Sequence	Store Player Sequence in PlayerSet(Order)	
6		WHILE PlayerSet(Order) = CompSet(Order): Increment Order	
7		REPEAT PROCESS FROM StepNo. 3	
8		IF PlayerSet(Order) != CompSet(Order):	"Game Over"

### NormMode

#### NormMode

StepNo.	I	P	O
1	Player clicks	Load <u>NormModeGUI</u>	

	EduAnimal		
2		Order = 1	
		Time = 6.5 SECONDS	
3		Randomise 1st sequence	
4		Store 1st sequence in CompSet(Order)	Display 1st sequence with medium of <u>NormModeGUI</u>
5	Get Player Sequence	Store Player Sequence in PlayerSet(Order)	
6		WHILE PlayerSet(Order) = CompSet(Order): Increment Order	
7		REPEAT PROCESS FROM StepNo. 3	
8		IF (PlayerSet(Order) != CompSet(Order)) AND Player Sequence not responsive by Time:	“Game Over”

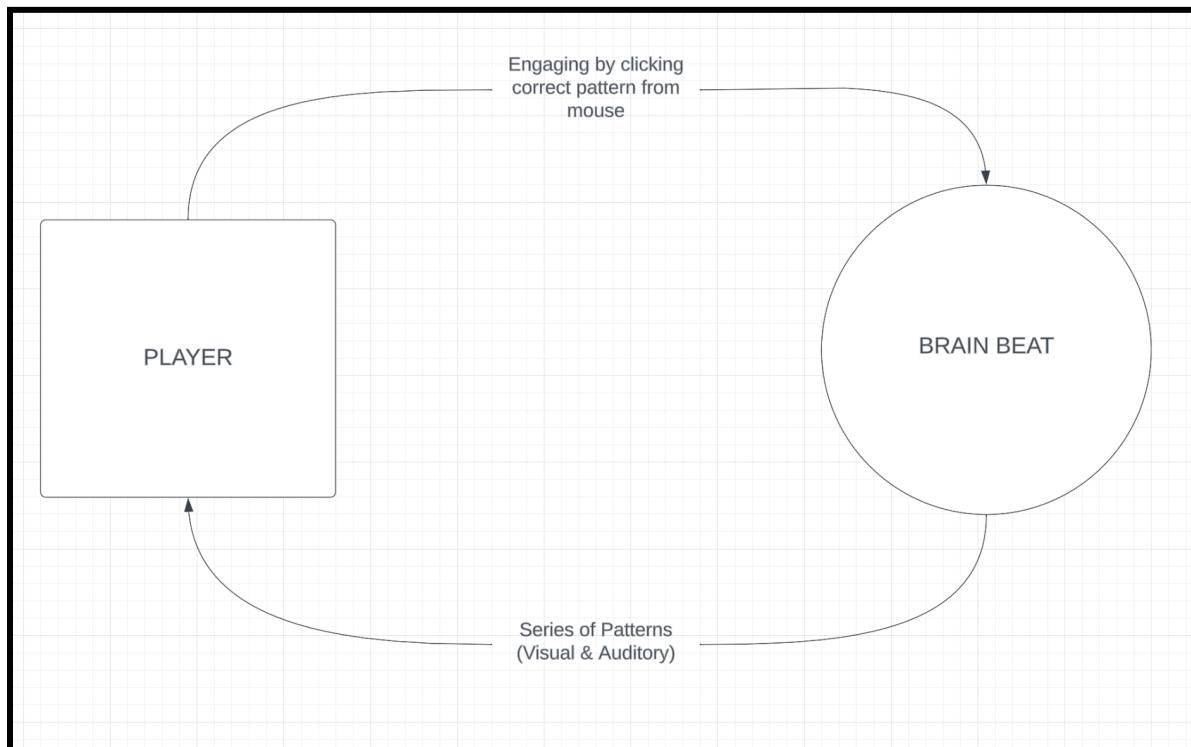
### ChallMode

#### ChallMode

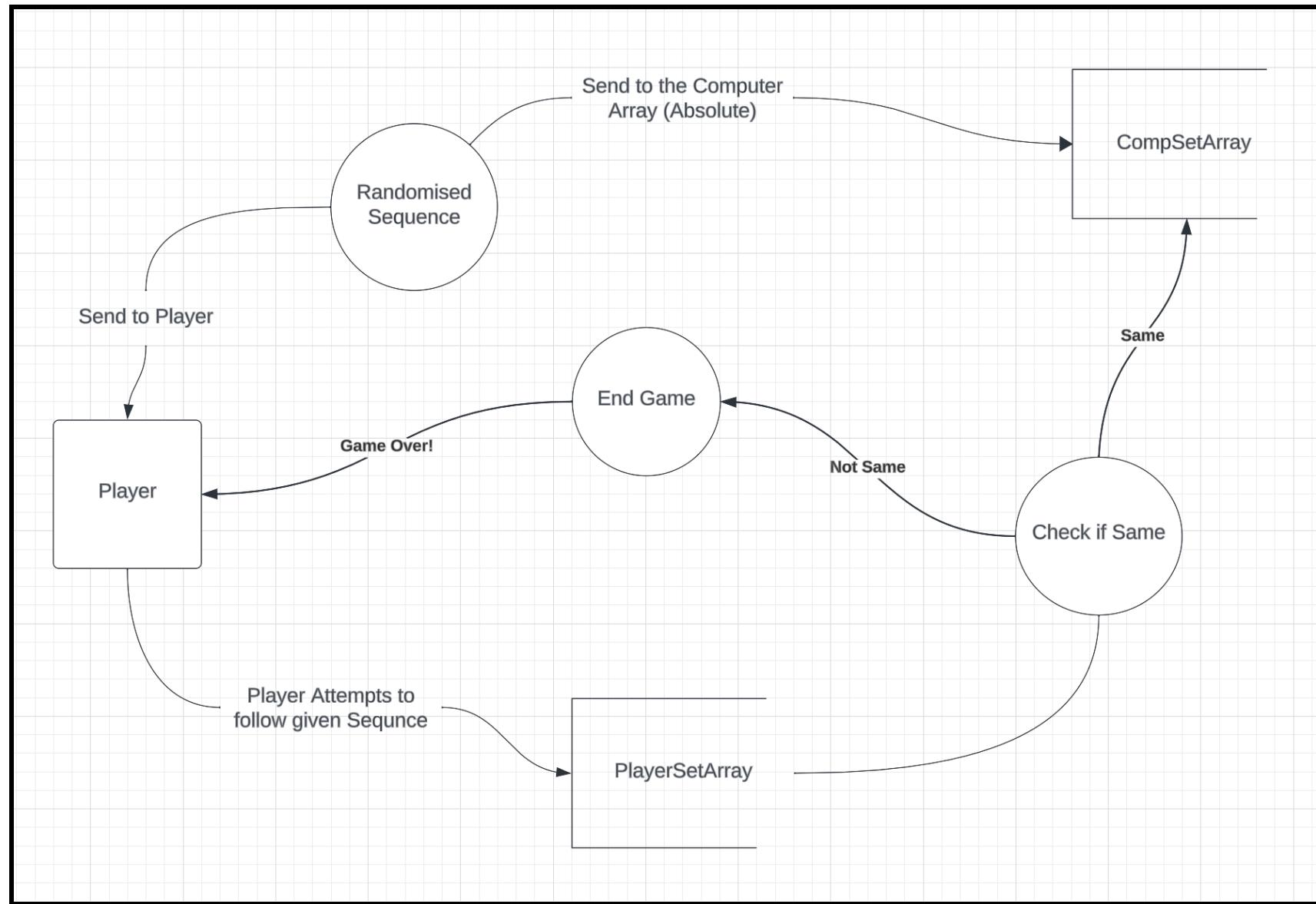
StepNo.	I	P	O
1	Player clicks EduAnimal	Load <u>ChallModeGUI</u> (DISTRACTIONS)	
2		Order = 1	
		Time = 3 SECONDS	
3		Randomise 1st sequence	
4		Store 1st sequence in	Display 1st sequence

		CompSet(Order)	with medium of <u>ChallModeGUI</u>
5	Get Player Sequence	Store Player Sequence in PlayerSet(Order)	
6		WHILE PlayerSet(Order) = CompSet(Order): Increment Order	
7		REPEAT PROCESS FROM StepNo. 3	
8		IF (PlayerSet(Order) != CompSet(Order)) AND Player Sequence not responsive by Time:	“Game Over”

### CONTEXT DIAGRAM



## DATA FLOW DIAGRAM (DFD)



## PRELIMINARY SKETCHES WITH ANNOTATIONS

### UI DESIGN 1 (HomeScreen)

This hand-drawn sketch illustrates the initial user interface. At the top left is the 'Brain Beat Logo' featuring stylized letters 'B' and 'B'. To the right is the title 'UI DESIGN 1 (HomeScreen)'. On the far right is a 'Settings' icon. The central area contains the text 'Brain Beat' and a large, hand-drawn 'PLAY' button with an arrow pointing right. In the corners are icons: a question mark in the top right and a gear in the bottom right. A red circle highlights the 'PLAY' button. A red line connects the 'PLAY' button to the 'How To Play' annotation below.

**Brain Beat Logo**  
Captivating, with Normal Mode Theme

**PLAY BUTTON**  
Proceeds to terminate Current Window, then open SelectGameMode

**QUIT APPLICATION**  
Allows for easy access to exit the App

**Settings**  
For User Friendly handling, options such as: Clear High Score, Sound and SFX Volume, etc...

**How To Play**  
Saved on a Text File which will appear as a Pop Up once clicked so that Players are informed of Brain Beat's purpose and function as a game

### UI DESIGN 2 (SelectGameMode)

This hand-drawn sketch shows the 'Select Game Mode' screen. The title 'Select Game Mode:' is at the top. Below it are three cards representing different game modes: 'EduMode Thumbnail' (with colorful shapes), 'Normal Mode' (with a bell and flower), and 'Challenge Mode' (with a skull and fire). A red circle highlights the 'Normal Mode' card. A red line connects the 'Normal Mode' card to the 'Modes' annotation below.

**EduMode Thumbnail**  
Colourful Shapes represent the Edu Mode Module

**Hover for More**  
Once User Hovers over, gets more of a Description as to WHAT each Mode is (Target Audience explained)

**Home Screen**  
For the User to go "Back" to the Home Screen and view other Options, E.g. (Settings, HTP)

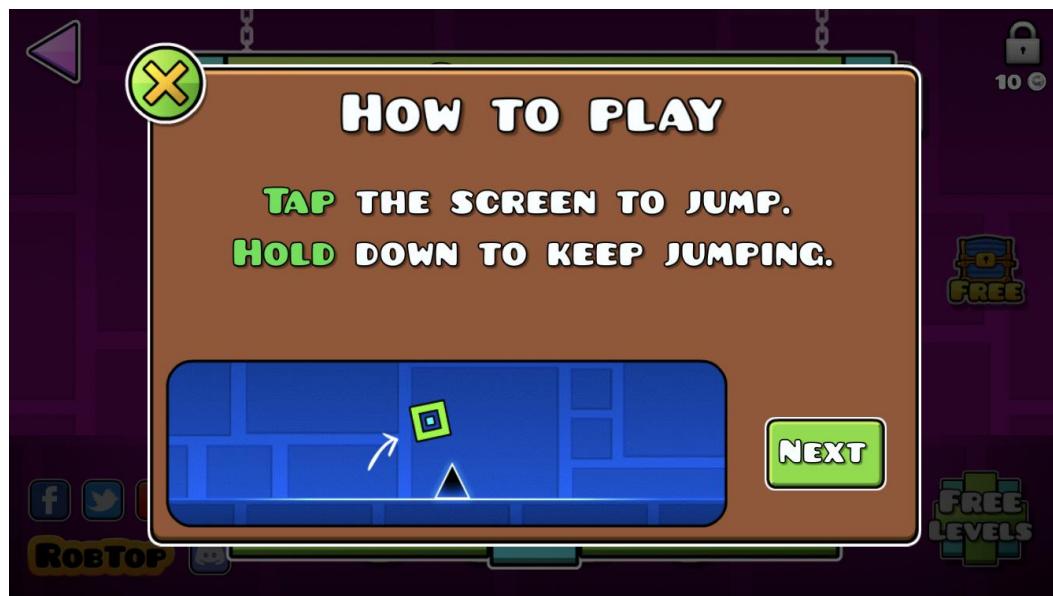
**Modes**  
Middle is Normal Mode, whilst last represents Challenge Mode with Skulls and Fire

## UI DESIGN 1 -

- The Settings app is following the GUI from a popular existing system named - “Geometry Dash Subzero”. This is known as a Graphical User Interface Menu Option. Once the Settings Button is clicked, it shall display the SoundVol and SFXVol items with the trackbar for easy slide options.



- The “HOW TO PLAY” button is similar to my “?” How to Play Button and will display similarly as well. As the Player will keep tapping the “NEXT” button, similarly BrainBeat will also adhere and talk about how to play as well with relevant images.



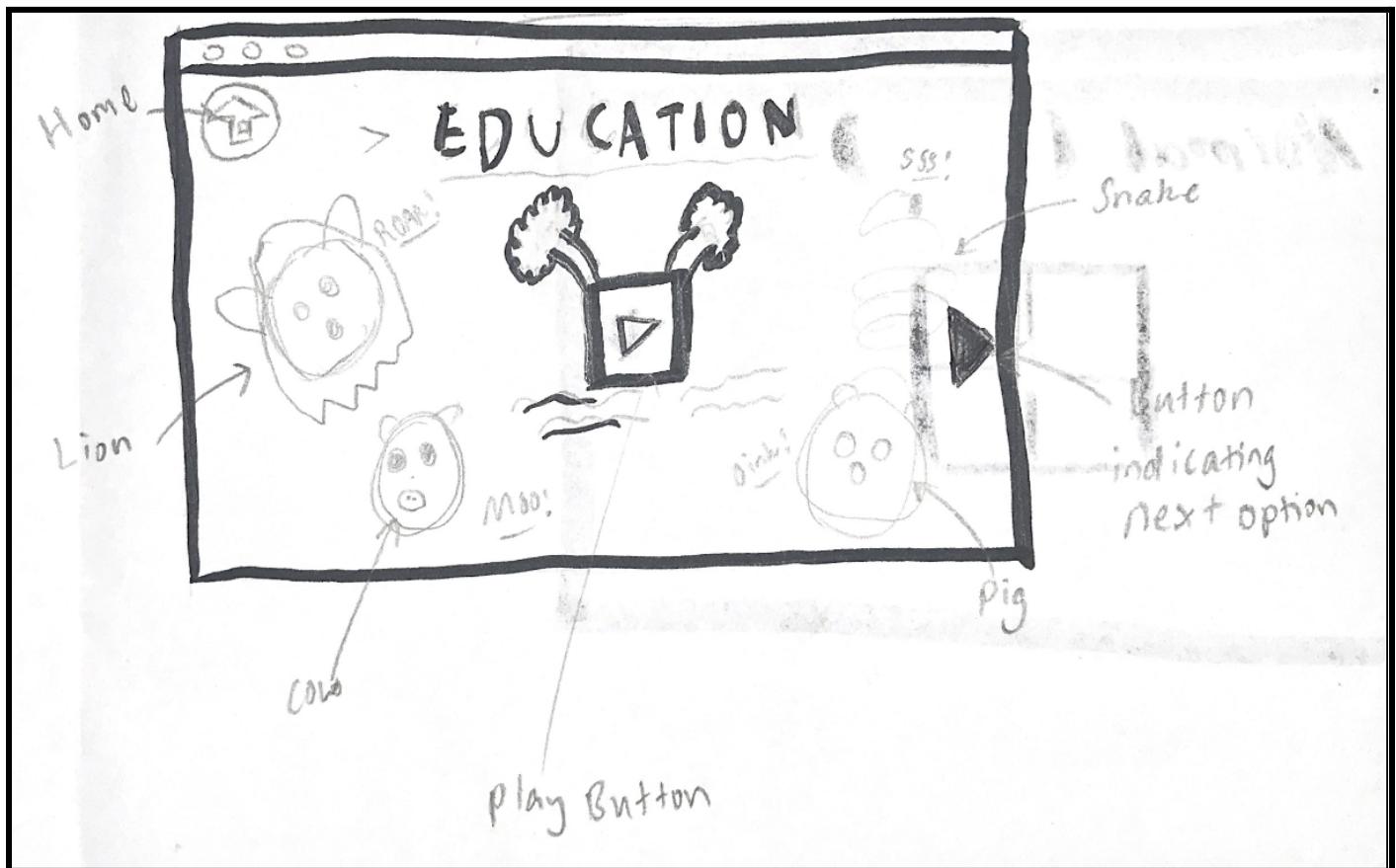
## UI Design 2-

- By Hovering above the "i" logo, one will be able to see more information about the Game Mode, with its suggested target audience as well.

ANOTHER POSSIBLE SOLUTION CONSISTS OF THE FOLLOWING IDEA:

Another possibility, to increase user friendliness is to make each screen have its own notion to support and promote the game mode. This includes the extensive use of colours and graphics.

**N.B.** This idea was thought of fairly close to the due date, therefore not being able to be explained to its fullest potential and not being included in the marketing pitch. STILL DECIDING WHETHER TO USE EduColour or EduMath (Report conducted under EduColour)



**CHALLENGE**

- Animated Fire to Show Hardness
- Demonic to Show Hard.
- No button indicating end of Options

**EDUCATION**

- Play Button
- Mathematical Operations

**Normal (Simon)**

- Play Button
- (courtesy of inspired by Simon)

## **PLANNING SOFTWARE SOLUTIONS**

*ALGORITHM, PSEUDOCODE, MODULE BY MODULE*

Brain Beat - Sequential

BEGIN BrainBeat:

    Display HomeScreen  
    Display SelectGameMode  
    Display:  
        EduMode OR NormMode OR ChallMode

END BrainBeat

HomeScreen - Multiway Selection

BEGIN HomeScreen:

    CASWHERE User clicks:  
        “?”, Pop Up GUI, Load and Display “htp.txt”  
        “SettingsButton”, Pop Up GUI, Load and Display Settings  
        “Play!”, Load and Display SelectGameMode

END HomeScreen

Settings - Sequential

BEGIN Settings:

    Load SoundVol  
    Load SFXVol  
    Load ClearHighScore  
    Load ChangeCol

END Settings

SoundVol - Sequential

BEGIN SoundVol:

    Get desired sound volume  
    Set current sound volume to desired sound volume

END SoundVol

SFXVol - Sequential

BEGIN SoundVol:

    Get desired sound effect volume  
    Set current sound effect volume to desired sound effect volume

END SFXVol

ClearHighScore - Binary Selection

BEGIN ClearHighScore:

    IF Player clicks “Clear High Score” THEN  
        Get which Game Mode  
        Set HighScore of that Game Mode to 0

END ClearHighScore

ChangeCol - Sequential

BEGIN ChangeCol:

    Get Game Mode to be altered (EduColour, NormMode, ChallMode)  
    Display Preview of how the Game looks  
    Get RGB code for each button  
    Change required button according to user preference colour by altering with its RGB Code

END ChangeCol

### SelectGameMode - Multiway Selection

BEGIN SelectGameMode

CASEWHERE User clicks:

    “HomeButton”, Load and Display HomeScreen  
    “Education Mode”, Load and Display EduMode  
    “Normal Mode”, Load and Display NormMode  
    “Challenge Mode”, Load and Display ChallMode

END SelectGameMode

### EduMode - Binary Selection

BEGIN EduMode:

    IF player clicks “Animal” THEN  
        Load EduAnimal  
    ELSE  
        Load EduColour

END EduMode

### EduAnimal - Pre Test Iteration

BEGIN EduAnimal:

    Order = 1  
    WHILE Order > 0  
        Randomly choose Animal of Order  
        Add Animal(Order) to CompSet Arr  
        Display Ordered Sequence along with (Load “Animal.mp3”)  
        Get Player Sequence  
        Store Player Sequence in PlayerSet(Order)  
        IF PlayerSet(Order) = CompSet(Order) THEN  
            Increment Order  
        ELSE  
            Order = 0  
    ENDWHILE  
    Display “Game Over”  
END EduAnimal

### EduColour

```
BEGIN EduColour:
    Order = 1
    WHILE Order > 0
        Randomly choose Colour of Order
        Add Colour(Order) to CompSet Arr
        Display Ordered Sequence along with (Load "Colour.mp3")
        Get Player Sequence
        Store Player Sequence in PlayerSet(Order)
        IF PlayerSet(Order) = CompSet(Order) THEN
            Increment Order
        ELSE
            Order = 0
        ENDWHILE
        Display "Game Over"
    END EduColour
```

### NormMode

```
BEGIN NormMode:
    Order = 1
    Time = 6.5 (SECONDS)
    WHILE Order > 0
        Randomly choose Colour of Order
        Add Colour(Order) to CompSet Arr
        Display Ordered Sequence along with (Load "Colour.mp3")
        Get Player Sequence
        Store Player Sequence in PlayerSet(Order)
        IF PlayerSet(Order) = CompSet(Order) THEN
            Increment Order
        ELIF (PlayerSet(Order) != CompSet(Order)) OR (Player Sequence response > Time)
            Order = 0
        ENDWHILE
        Display "Game Over"
    END NormMode
```

### ChallMode

```
BEGIN ChallMode:
    Randomly Display Dummy Flashing Lights
    Order = 1
    Time = 8 (SECONDS)
    WHILE Order > 0
        Randomly choose Colour of Order
```

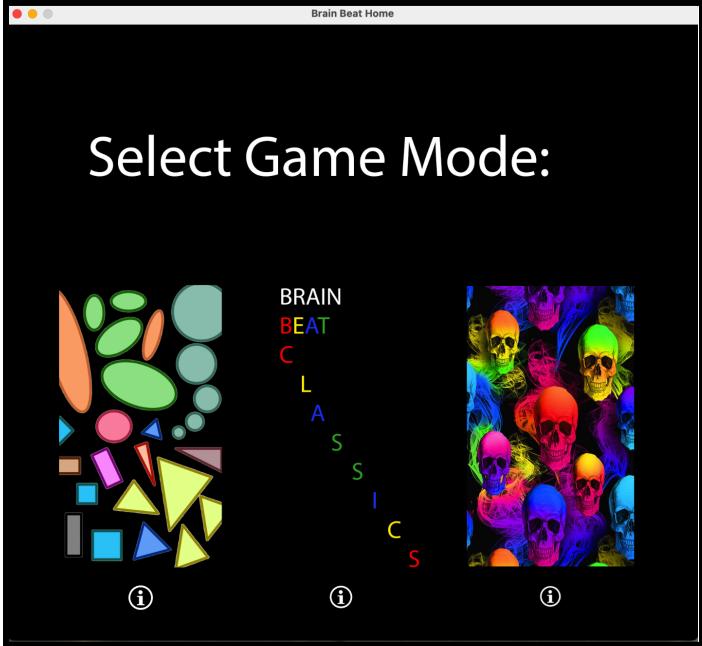
```

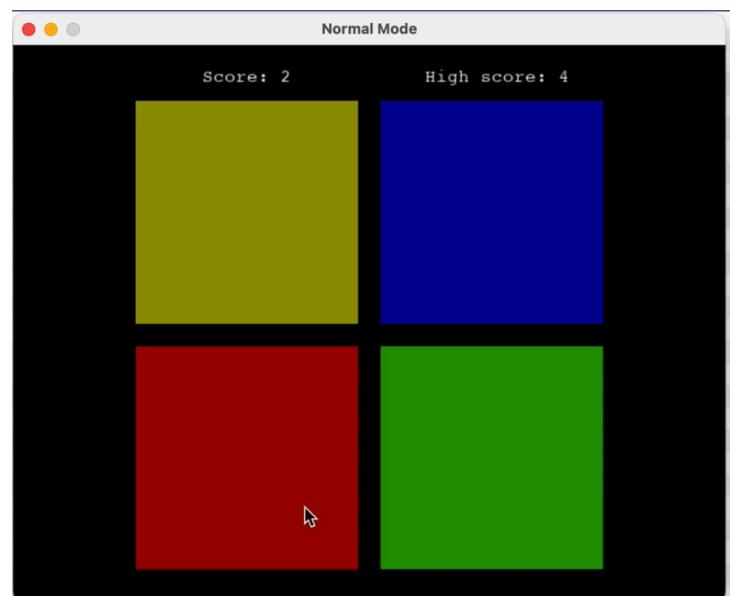
Add Colour(Order) to CompSet Arr
Display Ordered Sequence along with (Load "Colour.mp3")
Get Player Sequence
Store Player Sequence in PlayerSet(Order)
IF PlayerSet(Order) = CompSet(Order) THEN
    Increment Order
    Time = Time - 0.25 SECONDS
ELIF (PlayerSet(Order) != CompSet(Order)) OR (Player Sequence response > Time)
    Order = 0
ENDWHILE
Display "Game Over"
END NormMode

```

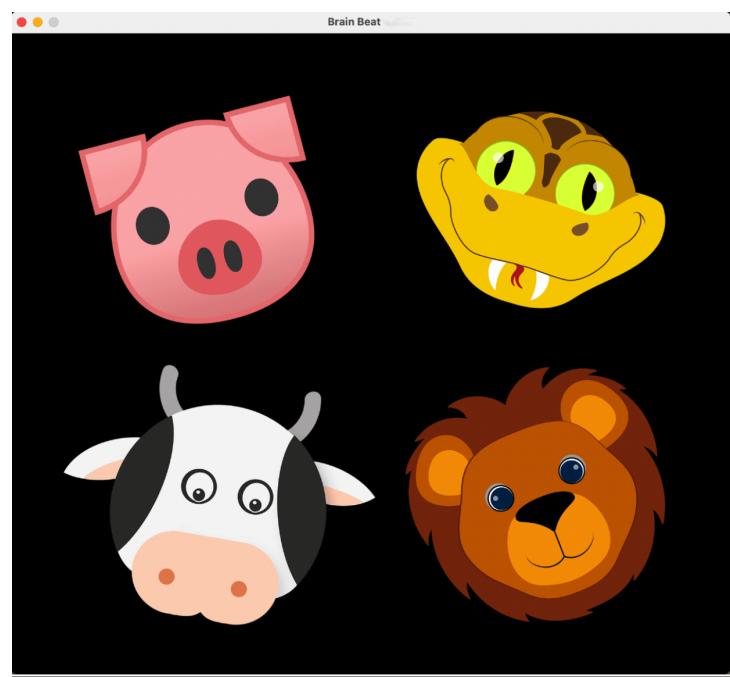
## DESIGNING SOFTWARE SOLUTIONS

### *CREATING DESIGN INTERFACES*

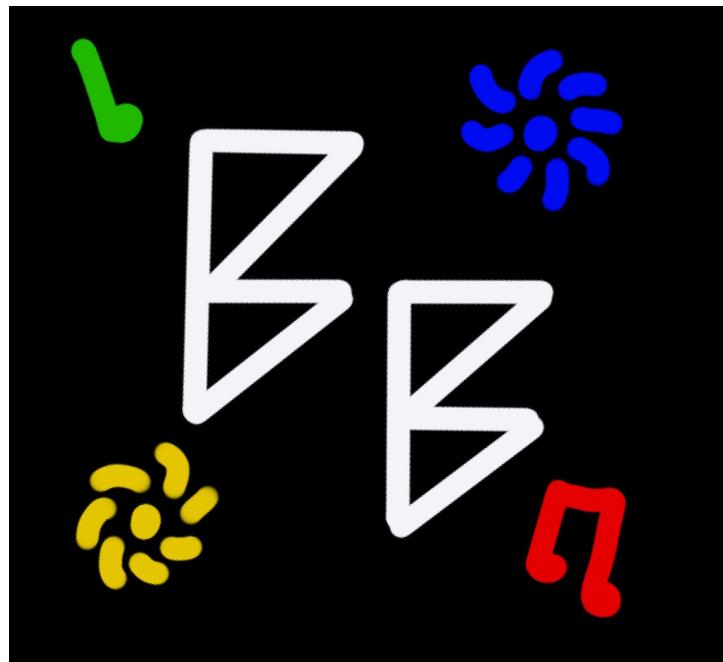
PHOTO OF GUI	DESCRIPTION
	<ul style="list-style-type: none"> <li>- Original Planned GUI for <u>SelectGameMode</u></li> <li>- Includes clear Thumbnailing for all modes</li> <li>- "I" logo is to be hovered over so that users can see the audience and additional info on the mode.</li> <li>- (PROTOTYPE WAS BEFORE MULTIPLE EDUCATION MODE MODULE)</li> </ul>



- NormMode GUI of Game Play.
- Still need to add the HomeScreen Button and Pause Button so that it is more User-Friendly
- Has standard RGBY
- Has Score and High Score Integer Counters
- Pays respect to Simon Memory Game



- EduAnimal from the Education Mode.
- Still need to animate so that once called upon, each head of the animal pops up with an animation to it.
- Audio is saved from youtube to mp3 converters and will be able to be implemented under the pygame add on to python
- MAY add option of choosing from a variety of Animals
- Still need to add the HomeScreen Button and Pause Button so that it is more User-Friendly

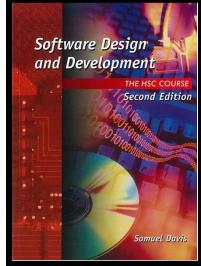


- Crafted on iPad with ApplePen
- Pays respect to Simon Memory Game
- Has yellow and blue light themed strikes
- Includes Red and Green Musical Notes
- Is to be IMPLEMENTED EVERY SCREEN DESIGN EXCEPT GAME TO FORMULATE A BRANDING TYPE SCHEME
- Two Bs represent Brain Beat



- For the Kids Thumbnail, found this on the web, however, needed the background to be black for the GUI
- USED Photoshop (Psd) to change and ended up with this by changing all white coloured cells to black ,also known as screen colour inverting.

## REFERENCES

Resource Type	Source	Purpose
Website	<a href="#">Simon (game)</a> , Wikipedia	Critical Analysis on existing product
Website	<a href="#">19 Captivating Facts About Simon (electronic Memory Game)</a> , Facts.net	Critical Analysis on existing product
Website	<a href="#">Team Gantt</a> , Team Gantt	Planning of Software Solution
Website	<a href="#">5 Great Memory Games for Adults</a> , The Clare	Target Audience (Adults + Elderly)
Website	<a href="#">The Importance of Playing Memory Games</a> , Curious World	Target Audience (Children)
Book	Author: Sam Davis Date Published: 25/12/2011 <a href="#">"Software Design and Development, THE HSC COURSE, Second Edition"</a>	TASK 1, PROGRAMMING PROJECT REPORT (Whole Assignment, Multiple Aspects) 
Website	<a href="#">Chat GPT</a> , “do a critical analysis on the Simon Memory Game - roughly 300 words” (PROMPT)	Critical Analysis of Existing Product (HOWEVER, DID MODIFY)