كلية الحاسبات والذكاء الإصطناعي
Faculty of Computers & Artificial Intelligence

جامعة حلوان
HELWAN UNIVERSITY

ERG

Gamemotion

WELCOME TO

GAMES WORLD

Supervisor :-
DR\Mohammed Elsaid

Table of contents

01 OVERVIEW

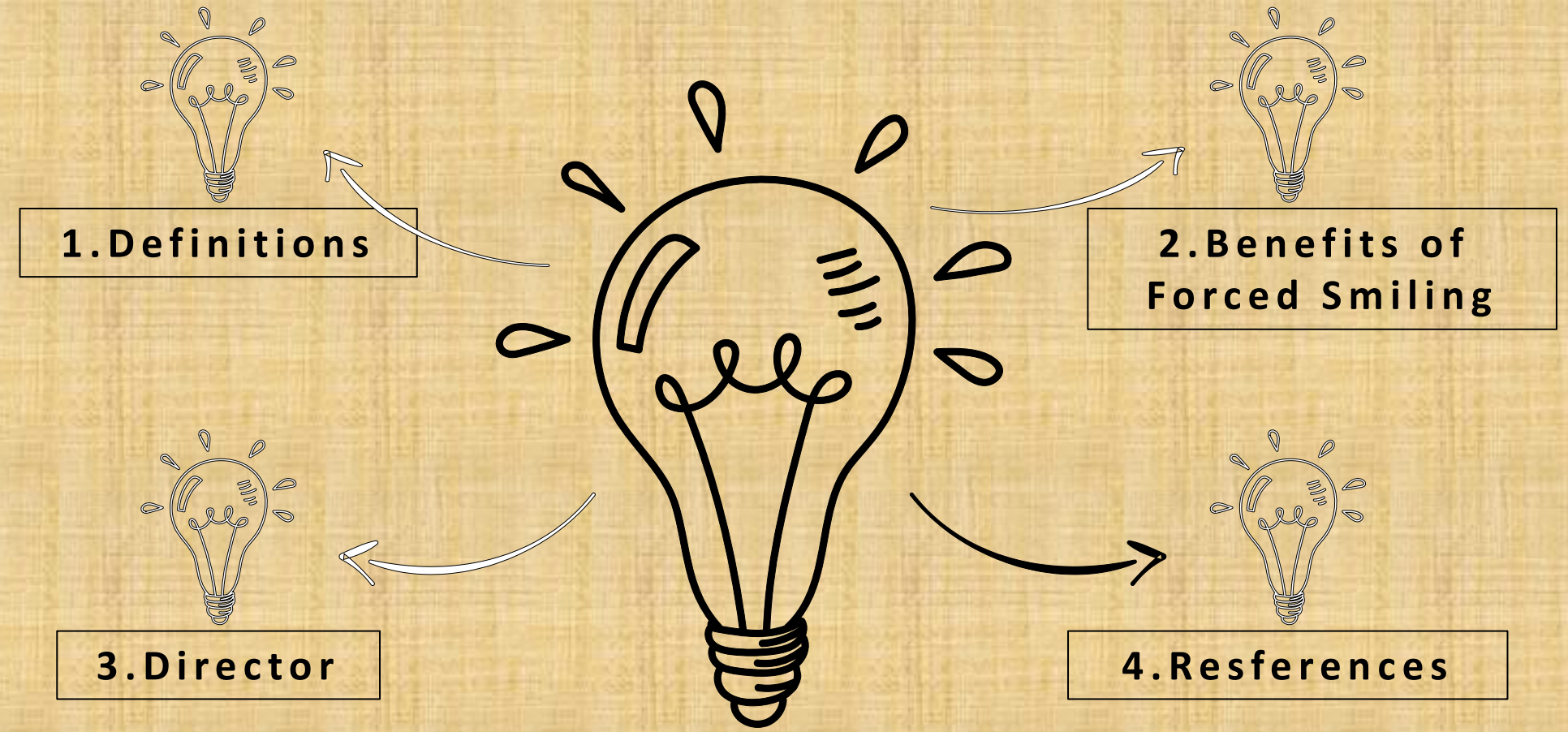
Overview

Facial expression games represent a cutting-edge integration of interactive entertainment and advanced computer vision technology. This project focuses on developing games that utilize facial expression recognition, specifically detecting happy and unhappy expressions, to create engaging and mood-enhancing gameplay experiences. The integration of this technology into game design offers a novel approach to user interaction, making the gaming experience more immersive and emotionally responsive.

02

FORCED SMILE

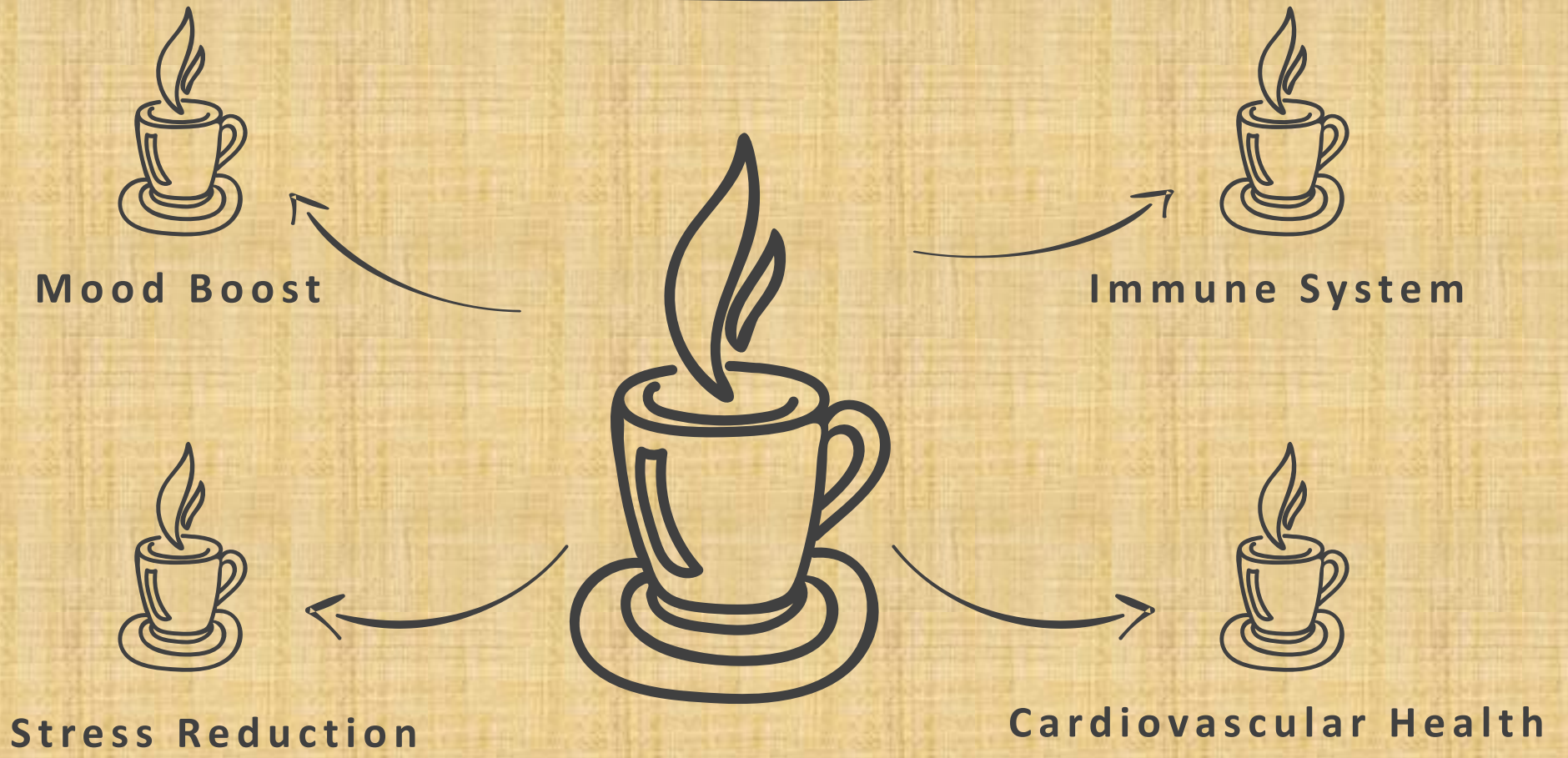
Forced Smile



1.Definition of Forced Smiling

might seem like a simple action, but it can have surprising effects on our well-being.

2.Benefits of Forced Smiling

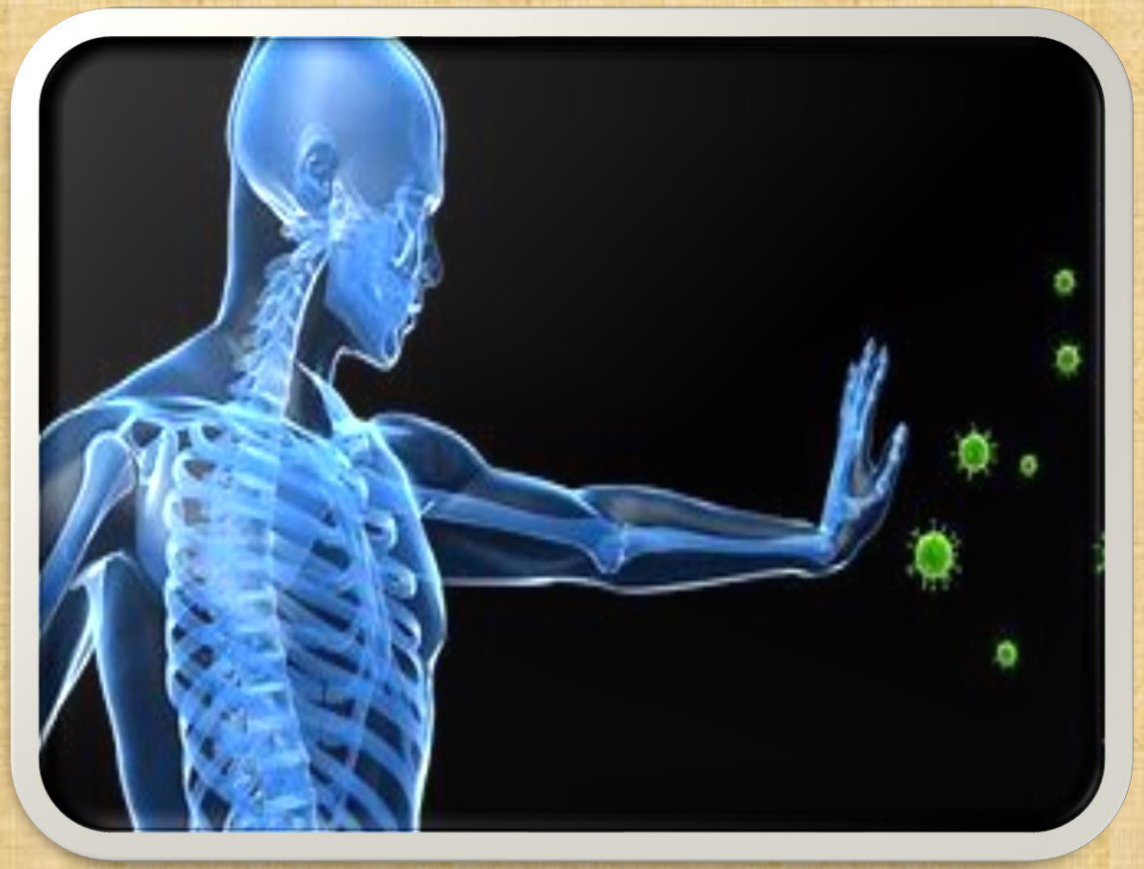


Ref :- Biological Psychology Book

Mood Boost

Forcing yourself to smile can trick your brain into believing you are actually happy. When you smile, your brain releases neurotransmitters like dopamine and serotonin, which are associated with feelings of happiness and reduced stress

Immune System



The physical act of smiling can impact your immune system. Even if you're just pretending, your brain sees the muscle activity and assumes that humor is happening. This can help build your immunity

Stress Reduction

Studies have found that smiling helps reduce the body's response to stress and lowers heart rate in tense situations. So, even if you're not genuinely feeling it, a smile can still have positive effects

Cardiovascular System

Smiling can lower your heart rate and blood pressure, contributing to a healthier cardiovascular system

3.Advisor

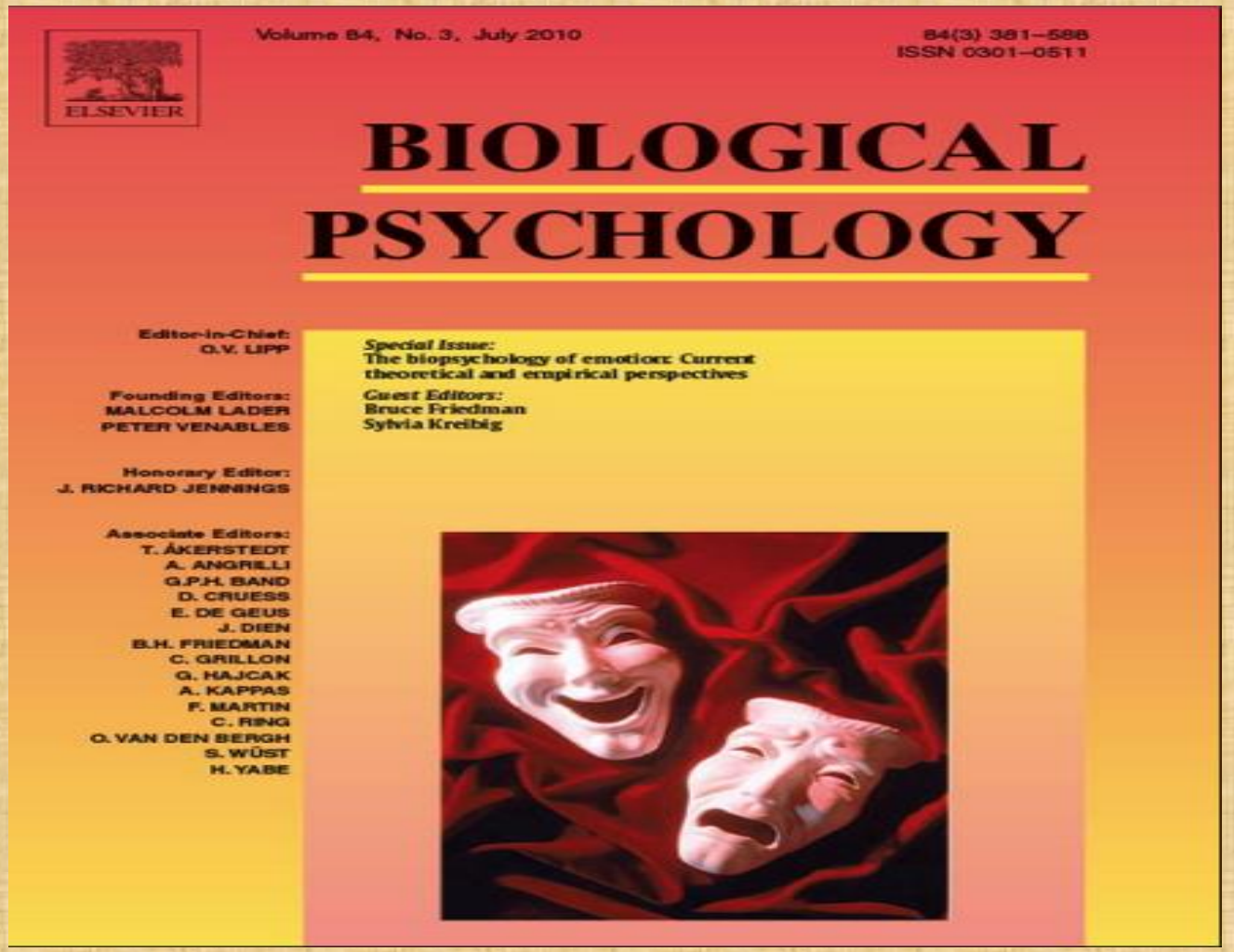
Dr\Heba Labna

Helwan University

Doctor at Helwan University, Faculty of Arts, Department of Psychology

4.References

Books :-



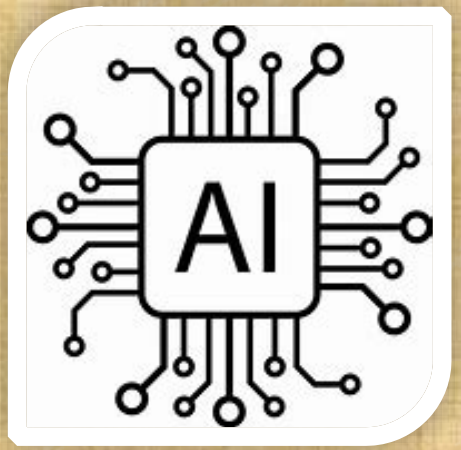
References

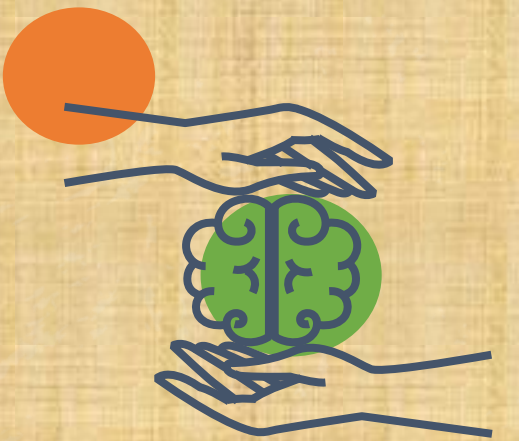
Papers :-

Paper topic	Link
The subjective value of a smile alters social behaviour - PMC	https://www.bing.com/ck/a?!&&p=6fc66ee6db09d856JmltdHM9MTcxNzExMzYwMCZpZ3VpZD0wZDQxNzdiYS0zOTNmLTZmMDktM2RlNi02MzlkMzg0MzZlZjYmaW5zaWQ9NTIxNg&ptn=3&ver=2&hsh=3&fclid=0d4177ba-393f-6f09-3de6-639d38436ef6&psq=Paper+about+forced+Smiling&u=a1aHR0cHM6Ly93d3cubmNiaS5ubG0ubmloLmdvdi9wbWMvYXJ0aWNsZXMvUE1DNjg4NjgwNi8&ntb=1
The Impact of Smile on Human Interactions: A Psychological Perspective	https://www.researchgate.net/publication/334625351_The_Impact_of_Smile_on_Human_Interactions_A_Psychological_Perspective
The Psychological Study of Smiling	The Psychological Study of Smiling – Association for Psychological Science – APS
Let the Avatar Brighten Your Smile: Effects of Enhancing Facial Expressions in Virtual Environments	https://doi.org/10.1371/journal.pone.0161794

03

THE AI MODEL





INTRODUCTION

Facial Expression Recognition :–

is a computer vision task aimed at automatically detecting and categorizing human facial expressions from images or videos. This documentation presents a comprehensive overview of a Facial Expression Recognition system developed using various deep learning models and datasets .

Datasets

The system utilizes several datasets for training and testing :

1. FER 2013

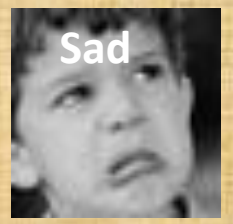
The FER 2013 dataset contains grayscale images of facial expressions categorized into seven classes: anger, disgust, fear, happy, neutral, sad, and surprise.

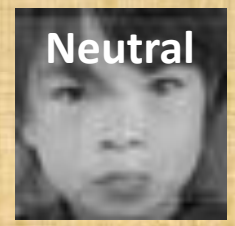
2. AffectNet

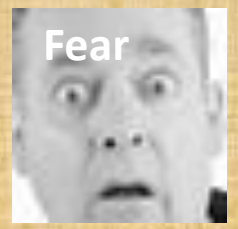
AffectNet is a large-scale dataset that contains images annotated with facial expressions. For this system, a subset of AffectNet with either 2 or 8 facial expressions was used, depending on the experiment.

Datasets

1. FER 2013 samples

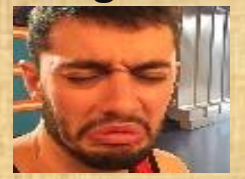


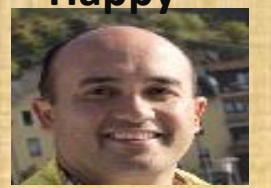




grayscaled (48,48)

2. AffectNet samples





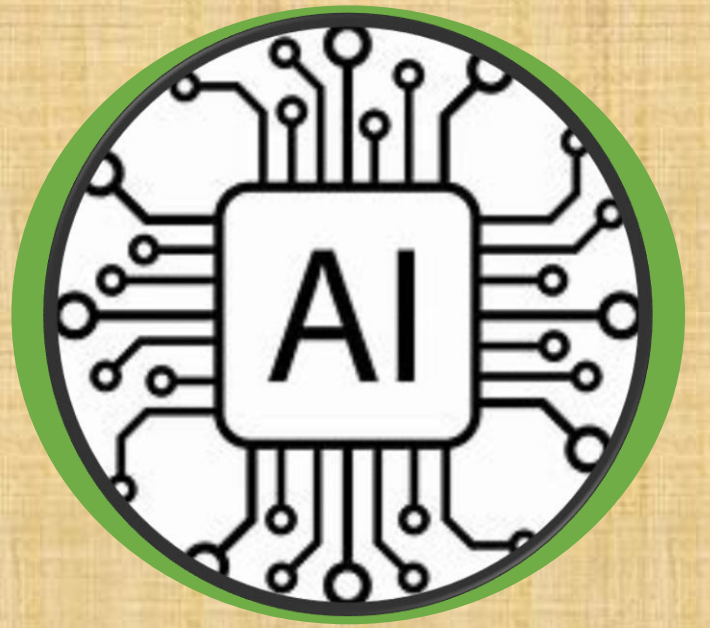
RGB (128,128)

Models

Several deep learning models were employed for facial expression recognition :

1. CNN

A custom CNN architecture was designed for the task. This CNN model consists of multiple convolutional layers followed by max-pooling and dropout layers to extract and learn hierarchical features from facial images



2. MobileNet

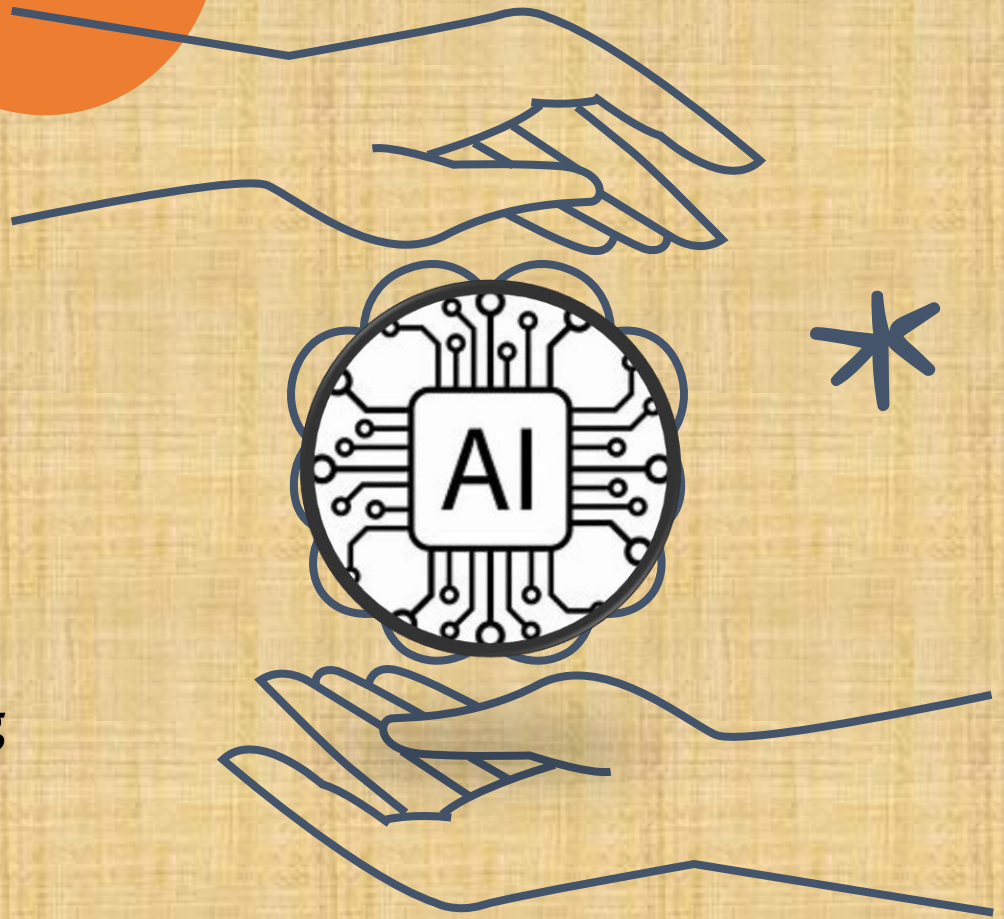
MobileNet is a lightweight convolutional neural network architecture optimized for mobile and embedded vision applications. In this system, MobileNet was used as a base model with additional fully connected layers for fine-tuning.

- Training and Evaluation:

The system was trained using a combination of training and validation data.

The training process involved data augmentation techniques such as rotation, shifting, and flipping to improve model generalization.

The performance of the models was evaluated using metrics such as accuracy.



EXPERIMENTATION AND RESULTS

Several experiments were conducted to explore different combinations of models and datasets:

Experiments compared by :

1. Used Dataset
2. Num Of Expressions
3. Model
4. Training Model
5. Testing and Validation Accuracy

EXPR_5

EXPR_4

EXPR_3

EXPR_2

EXPR_1

Num Of EXP

7 SEVEN EXP

Happy , Sad , Anger , Surprise , Fear , Confused , Nature

Training Accuracy

Model

Custom CNN Architecture

Testing and Validation Accuracy

DATASET:

-:: FER 2013

EXPR_5

EXPR_4

EXPR_3

Num Of EXP

3 Three EXP

Happy , Sad ,Nature

Training Accuracy

Model

Custom CNN Architecture

Testing and Validation Accuracy

DATASET:

-:: FER 2013

EXPR_2

EXPR_1

EXPR_5

EXPR_4

EXPR_3

EXPR_2

EXPR_1

Num Of EXP

2 Two EXP

Happy , Sad

Training Accuracy

Model

MobileNet architecture with additional fully connected layers

Testing and Validation Accuracy

DATASET:

-:: AffectNet

EXPR_5

EXPR_4

EXPR_3

EXPR_2

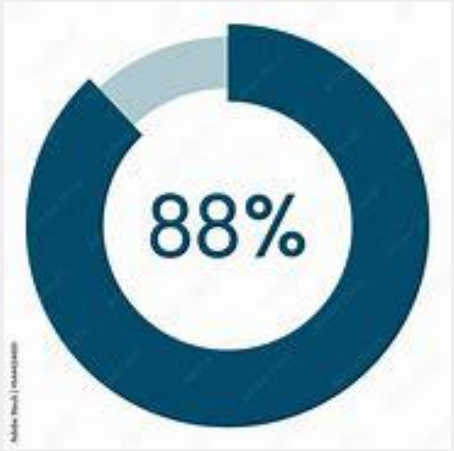
EXPR_1

Num Of EXP

8 Eight EXP

Happy , Sad ,
Anger , Surprise ,
Fear , Confused ,
Nature , disgust

Training Accuracy



Model

MobileNet architecture with additional fully connected layers

Testing and Validation Accuracy

DATASET:

-:: AffectNet

Training Accuracy

Testing and Validation Accuracy

Num Of EXP

2 Two EXP

Happy , Else (considered as Unhappy)

Model

MobileNet architecture with additional fully connected layers

DATASET:

-:: AffectNet

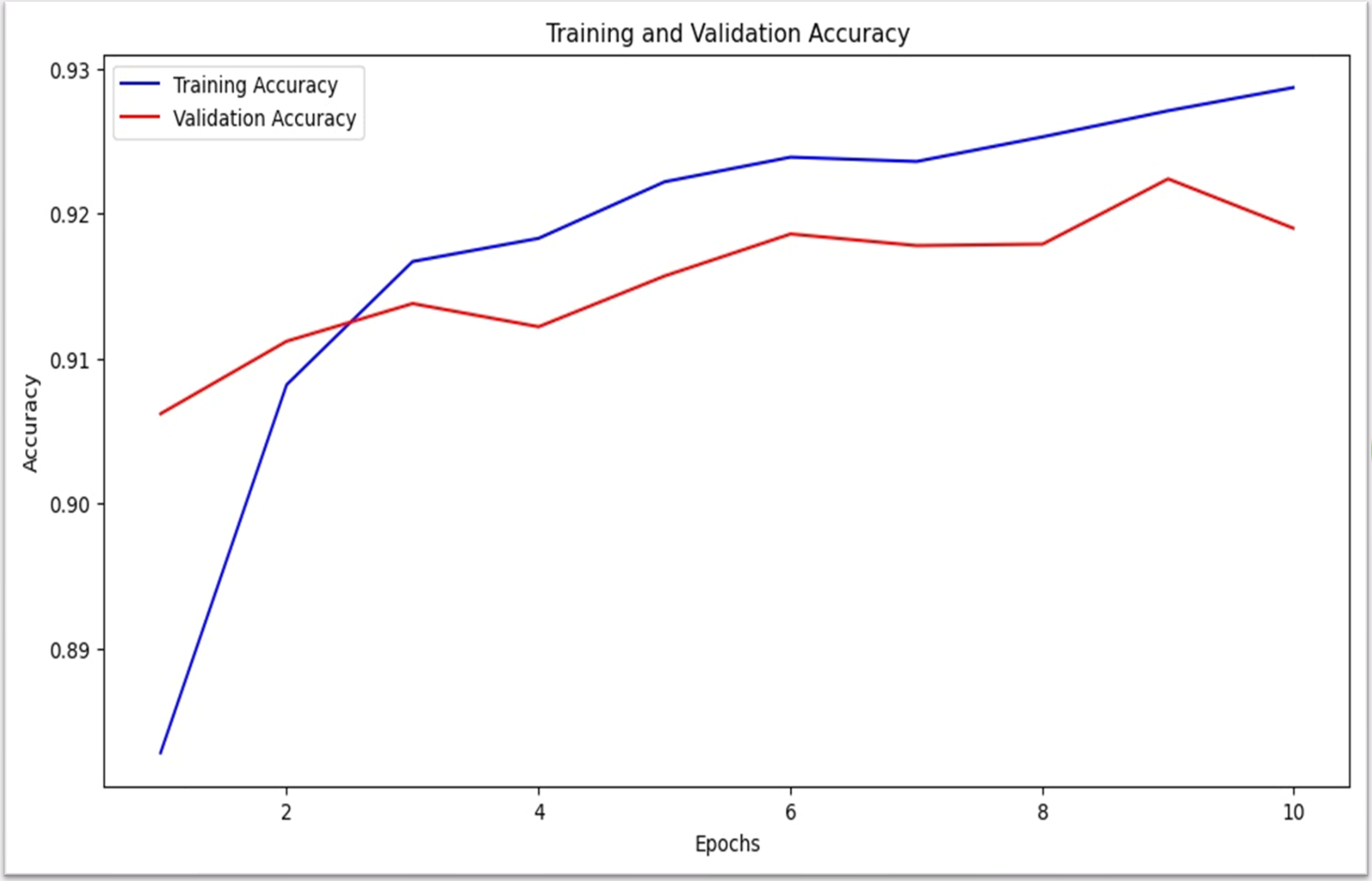
EXPR_5

EXPR_4

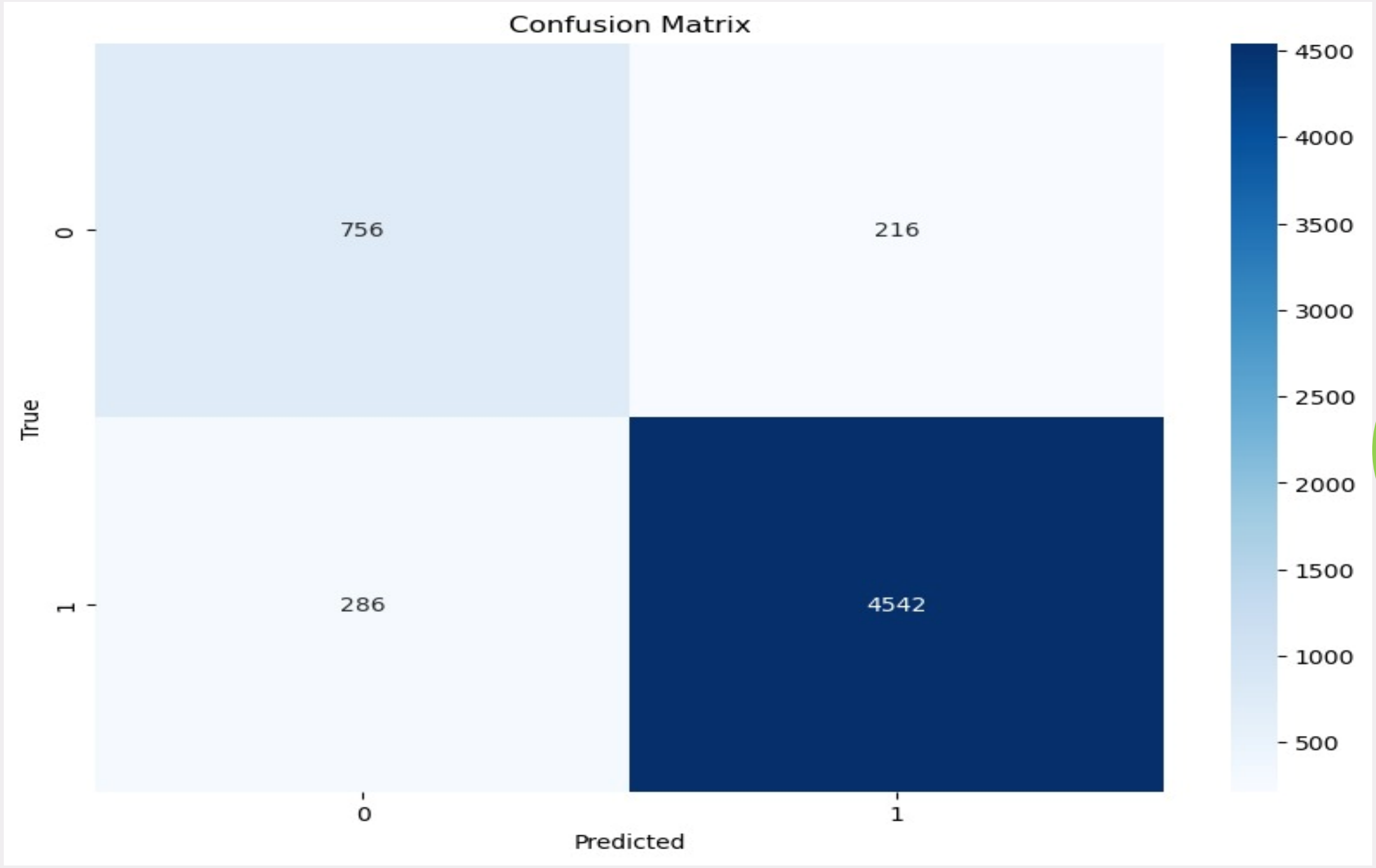
EXPR_3

EXPR_2

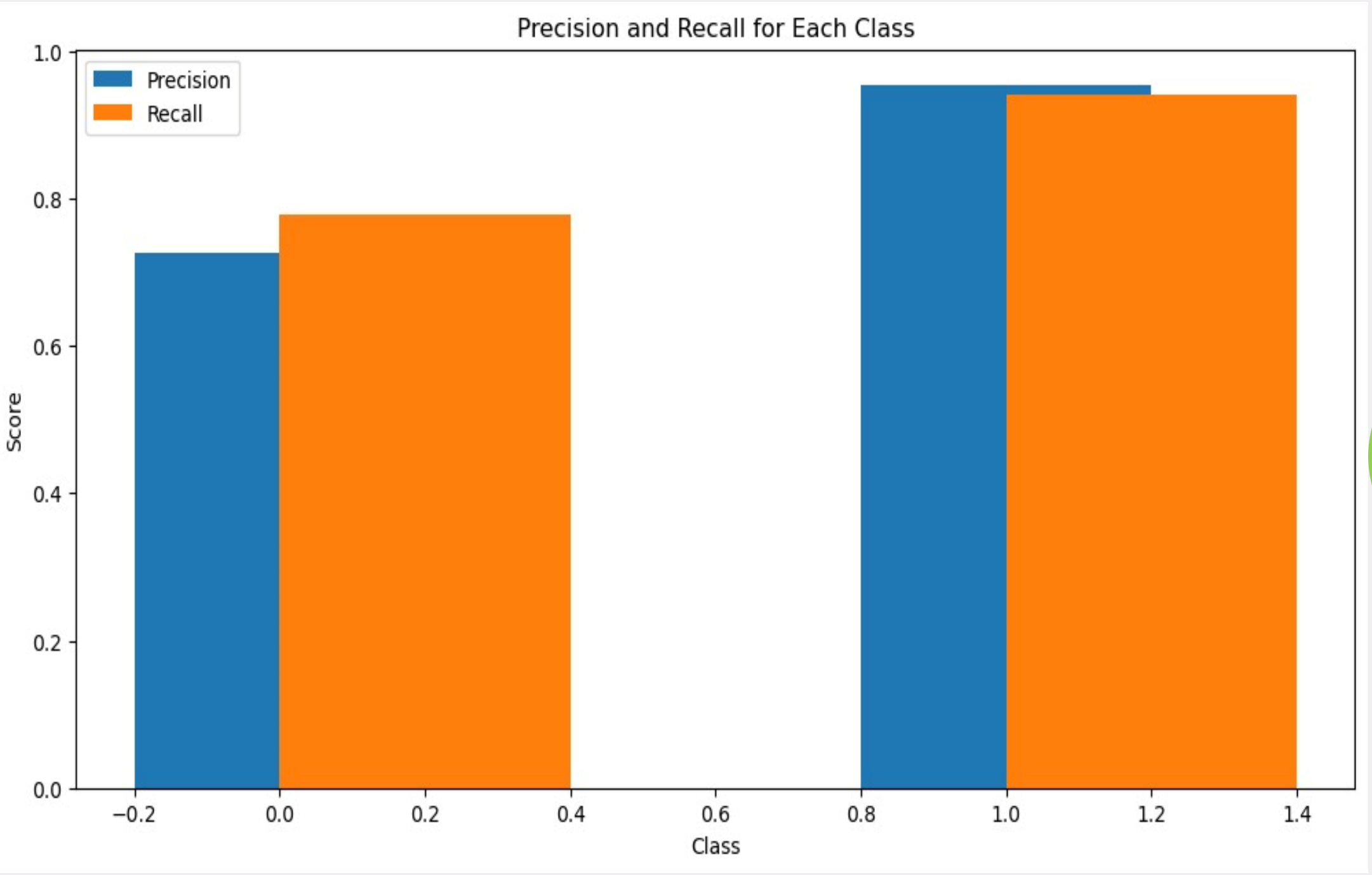
EXPR_1


Training and Validation Accuracy
Training Accuracy
Validation Accuracy
0.93
0.92
0.91
0.90
0.89
Accuracy
2
4
6
8
10
Epochs

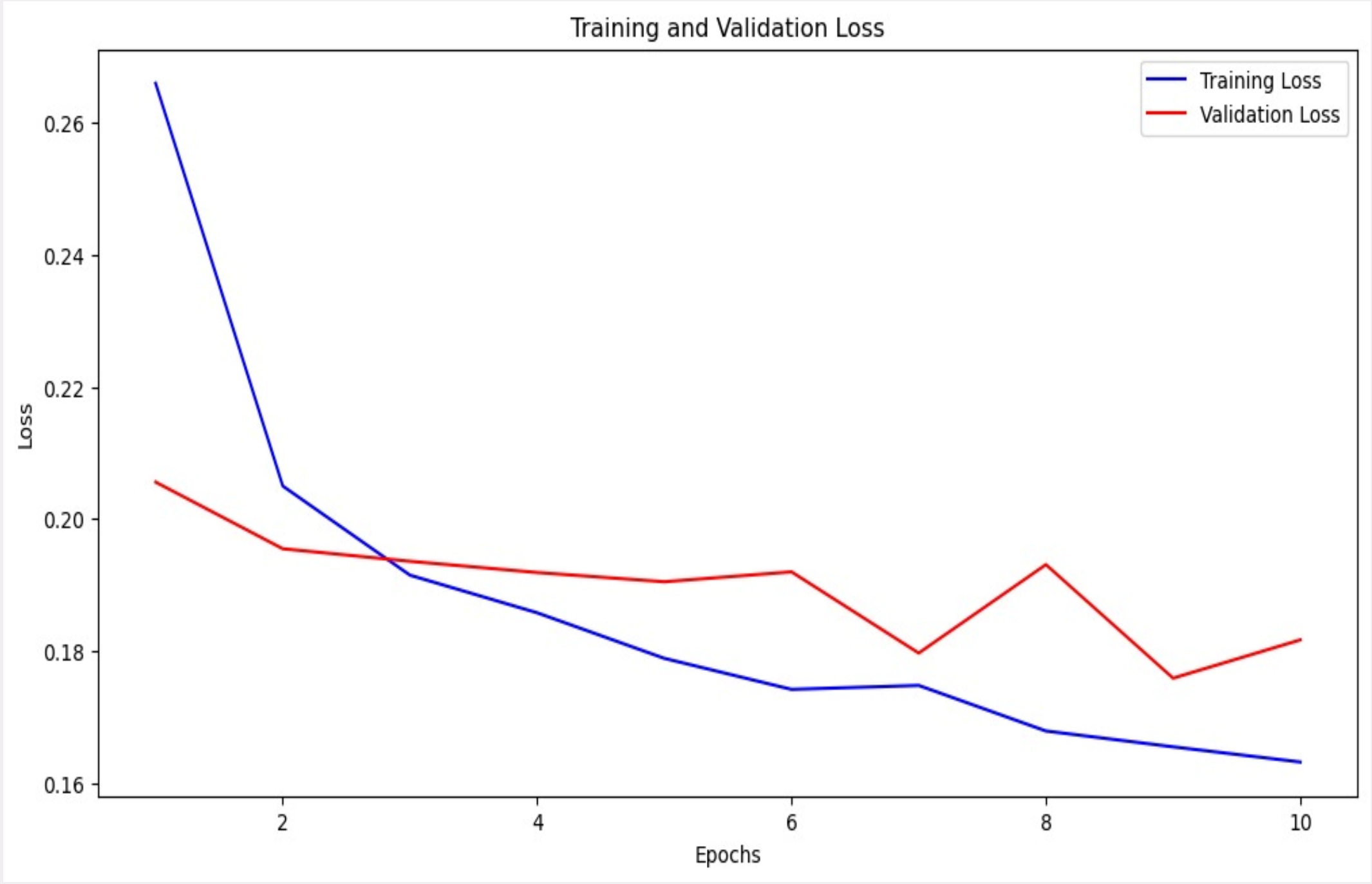
EXPR_5


Confusion Matrix
True
Predicted
0
1
756
216
286
4542
4500
4000
3500
3000
2500
2000
1500
1000
500

EXPR_5


Precision and Recall for Each Class
Precision
Recall
Score
Class
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4

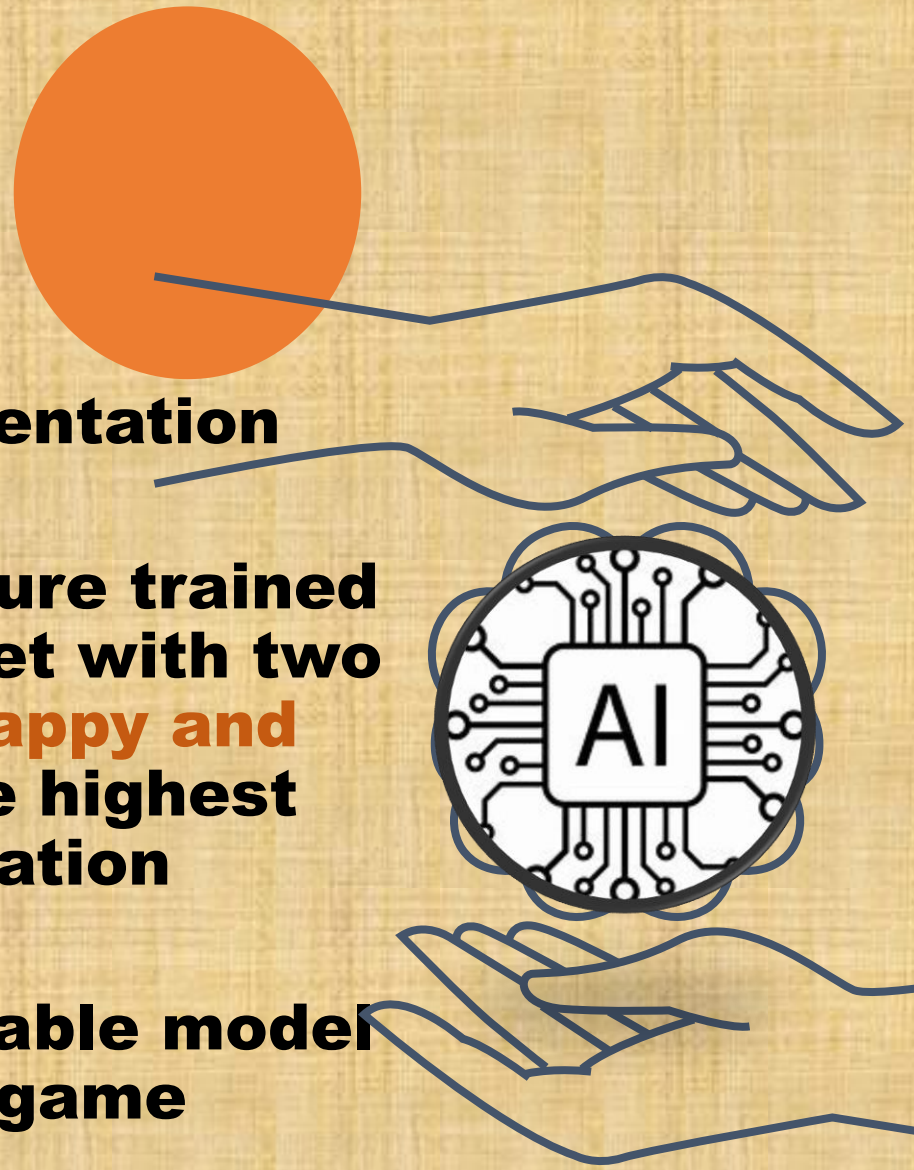
EXPR_5


Training and Validation Loss
Training Loss
Validation Loss
Loss
0.26
0.24
0.22
0.20
0.18
0.16
2
4
6
8
10
Epochs

EXPR_5

EXPR_5
EXPR_4
EXPR_3
EXPR_2
EXPR_1

-:: CONCLUSION



- **Based on the experimentation and evaluation,**

the MOBILENET architecture trained on the AFFECTNET dataset with two primary expressions (happy and not happy) achieved the highest accuracy and generalization performance,

making it the most suitable model for integration into our game application.

04

INTEGRATION THE MODEL WITH GAME

Integration

1.Computer Vision With Unity

2.connect the ML model to the game

3.Tools

4.Technologies

Integration

1.Computer Vision With Unity

Unity Computer Vision is a set of free tools and content for creating synthetic data12. Synthetic data is data that is artificially generated to simulate real-world scenarios. Synthetic data can be used to train and test computer vision models more efficiently and accurately. Unity Computer Vision offers open source tools that allow users to create high-quality synthetic data faster

Integration

1.Computer Vision With Unity

There are several ways to connect tensor AI model with Unity Game :-

The most popular ways is :

1. ML Agents

2. ONNX

2.How did we connect the ML model to the game

Steps :-

1. We have a TensorFlow model and convert it into ONNX model just because integrating it into Unity can be straight forward using ONNX Runtime for Unity.

2. We have faced some challenges like some additional setup and scripting to Integrate ONNX models into Unity and that's more complex compared to other tools like (ML agent).

3. Up to this point we have an ONNX model and script to connect it to Unity but that not all.

4. We have to install OpenCv plus for Unity and install Barracuda (is a lightweight cross-platform Neural Networks inference library for Unity).

2.How did we connect the ML model to the game

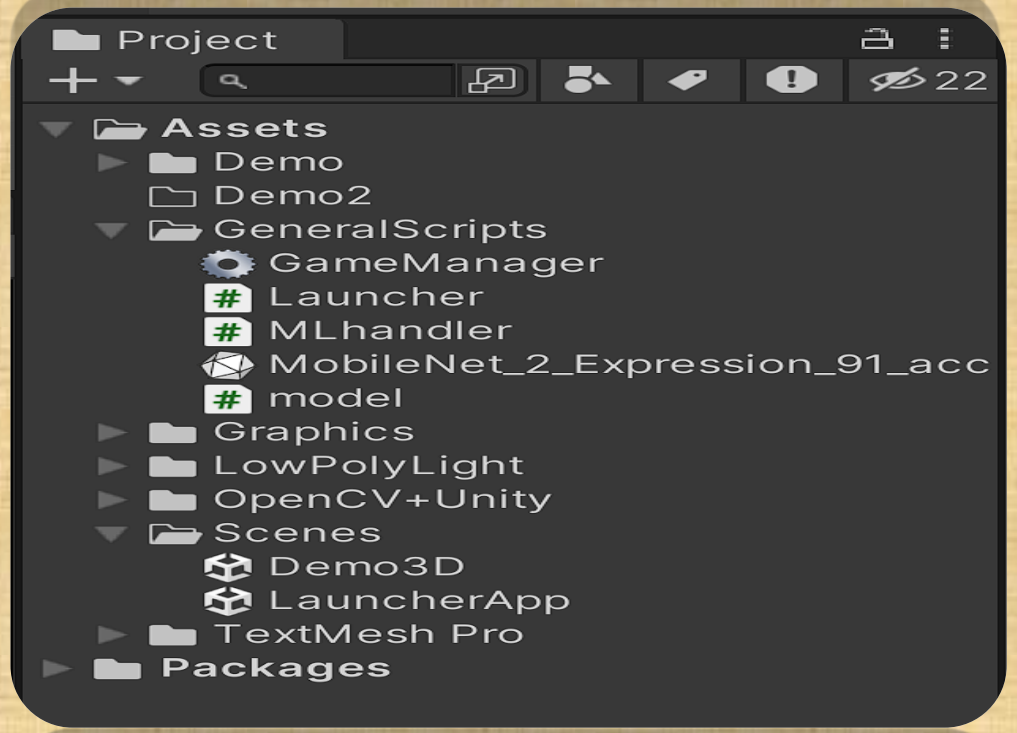
Steps :-

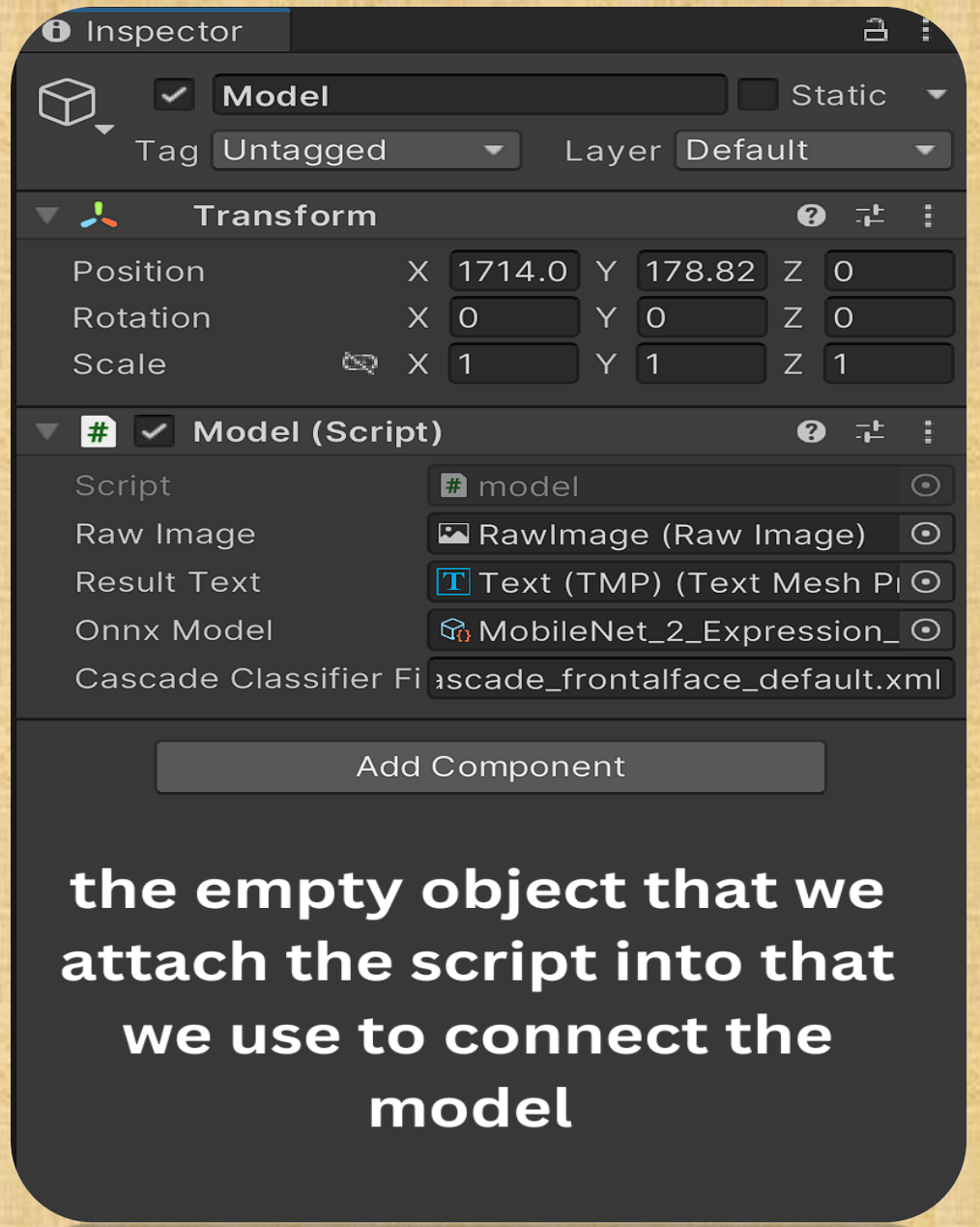
5. We connected all three the Barracuda, OpenCV plus and ONNX model through one script and attach it to an empty game object in Unity

6. So the attached script requires 4 things, firstly a raw image to display the image captured by the webcam, secondly the ONNX model, thirdly the text that displays the words (Happy, Not Happy), fourthly the path on the system to get the required file from OpenCv plus folder.

7. Now we have the model connected to Unity but how does the model interact with the game.

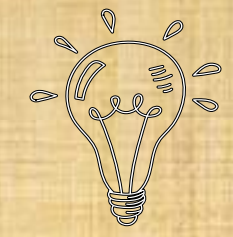
8. To make the model interact with the game we write a script and add condition in it that if the text (the required one previously in the script we use to connect the model) display "Happy" do that (anything we want for the gameplay).





the empty object that we attach the script into that we use to connect the model

Integration



The Format We used in our Games

ONNX :- format to convert the (model.h5) to (model.ONNEX)

```
!pip install onnx tf2onnx
import tensorflow as tf

# Load the Keras model
keras_model =
tf.keras.models.load_model('/content/sample_data/MobileNet_2_Expression_91_accura
cy.h5')



import tf2onnx
import onnx

# Convert the Keras model to ONNX format
spec = (tf.TensorSpec((None, 128, 128, 3), tf.float32, name="input_2"),)  #
Specify input shape
# Use tf2onnx to convert the model
onnx_model, _ = tf2onnx.convert.from_keras(keras_model, input_signature=spec,
opset=12)

# Save the ONNX model to a file
onnx.save(onnx_model, 'MobileNet_2_Expression_91_accuracy2.onnx')
```

Integration

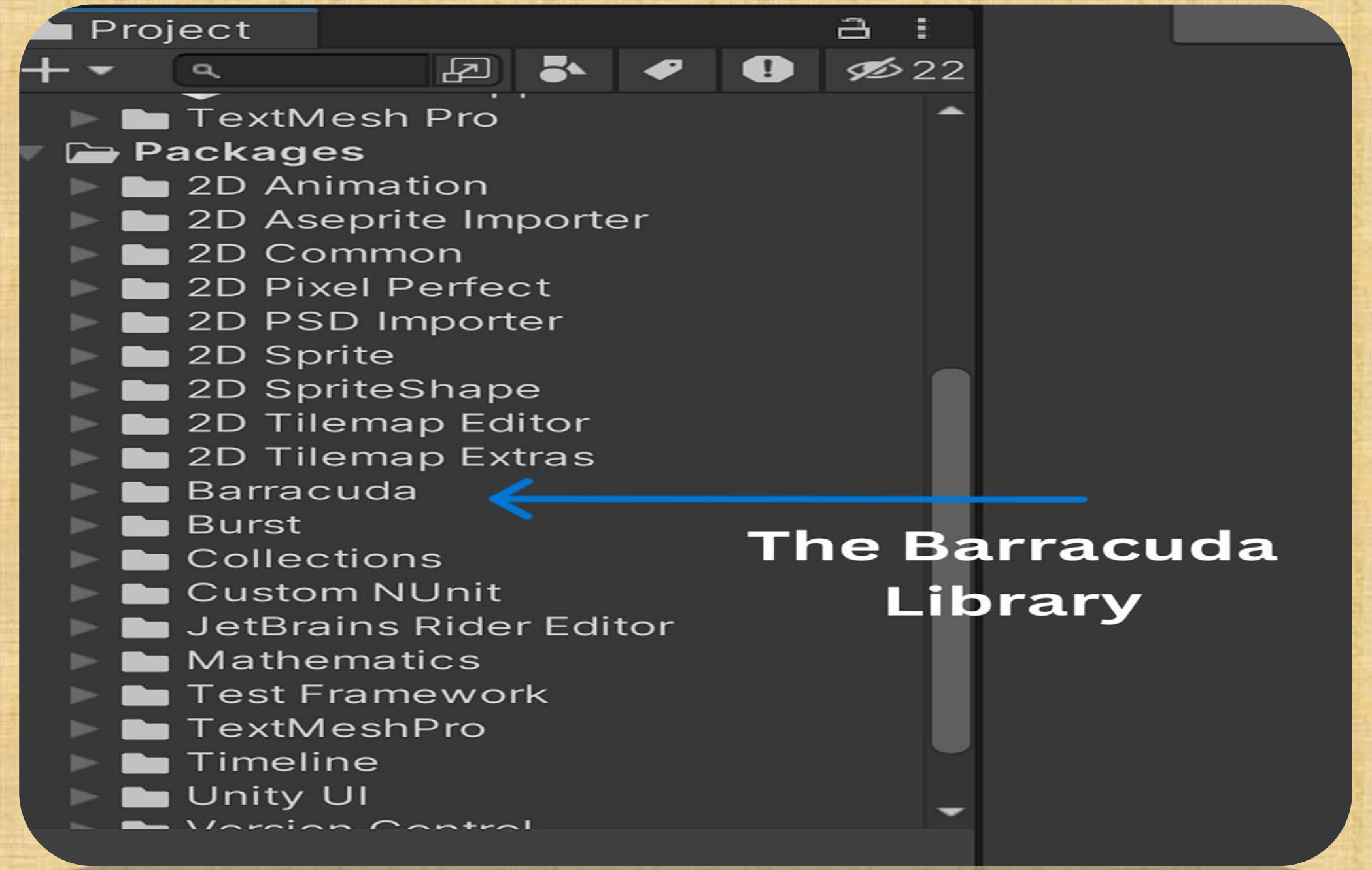
2.1 Packages Needed

1. Barracuda

2. OpenCvSharp

2.1.1.Barracuda

The Unity Barracuda package is a comprehensive library designed for neural network inference within the Unity environment.
It allows developers to run pre-trained neural network models, such as those exported in ONNX (Open Neural Network Exchange) format, directly within Unity applications.
Barracuda supports the ONNX format, which is an open standard for representing machine learning models.
This compatibility allows you to export models from various machine
learning frameworks (like TensorFlow, PyTorch, etc.) and run them in Unity. ONNX support ensures that models can be seamlessly integrated into Unity applications without extensive re-implementation.


Project
22
TextMesh Pro
Packages
2D Animation
2D Aseprite Importer
2D Common
2D Pixel Perfect
2D PSD Importer
2D Sprite
2D SpriteShape
2D Tilemap Editor
2D Tilemap Extras
Barracuda
Burst
Collections
Custom NUnit
JetBrains Rider Editor
Mathematics
Test Framework
TextMeshPro
Timeline
Unity UI
The Barracuda Library

2.1.1.Barracuda

Unity-Technologies/
barracuda-release

2 Contributors · 30 Issues · 503 Stars · 71 Forks

```
using Unity.Barracuda;
using OpenCvSharp; // instead of OpenCV
```

Code:

```
// Path to the ONNX model file
    public NNModel onnxModel;

// Load the ONNX model using Barracuda
    var model = ModelLoader.Load(onnxModel);
```





2.1.1.OpenCvSharp

OpenCvSharp plays a critical role in our Unity script by providing the necessary tools for image processing, including converting between different image formats, detecting faces, and manipulating images.

It works alongside Barracuda, which handles the neural network inference, to create a complete facial expression detection pipeline within Unity.

2.1.1.OpenCvSharp

```
using OpenCvSharp;   // Uploading the backage

// Step 1: Create a Texture2D from the WebCamTexture
Texture2D webcamTexture2D = new Texture2D(webcamTexture.width,
webcamTexture.height, TextureFormat.RGB24, false);
webcamTexture2D.SetPixels32(webcamTexture.GetPixels32());
webcamTexture2D.Apply();
```

2.1.1.OpenCvSharp

```
// Step 2: Convert the Texture2D to an OpenCV Mat
Mat frameMat = OpenCvSharp.Unity.TextureToMat(webcamTexture2D);
// Convert frame to grayscale for face detection
Mat grayMat = new Mat();
Cv2.CvtColor(frameMat, grayMat, ColorConversionCodes.BGR2GRAY);

// Detect faces in the frame
OpenCvSharp.Rect[] faces = cascadeClassifier.DetectMultiScale(grayMat,
scaleFactor: 1.1, minNeighbors: 5, minSize: new Size(30, 30));

 // Resize faceMat to 128x128 (adjust size according to your model's input size)
 Mat resizedFaceMat = new Mat();
 Cv2.Resize(faceMat, resizedFaceMat, new Size(128, 128));

 // Convert the resized faceMat to a Texture2D
 Texture2D faceTexture = OpenCvSharp.Unity.MatToTexture(resizedFaceMat);

// Draw a rectangle around the face and add the emotion text
Cv2.Rectangle(frameMat, face, Scalar.Green, 2);
Cv2.PutText(frameMat, emotion, new Point(face.X, face.Y - 10),
HersheyFonts.HersheySimplex, 0.9, Scalar.Green, 2);
```

3.Tools

unity

Google colab

OpenAI

ChatGPT

jupyter

4.Technologies

C#

OpenCV

Unity-Technologies/
barracuda-release

2 Contributors · 30 Issues · 503 Stars · 71 Forks

TensorFlow

Keras

05 THE GAMES

“Computer Vision in gaming"

- The early 2000s and PlayStation Eye
- VR & AR
- Ray tracing to add realistic Graphics

“Emotion Recognition games“

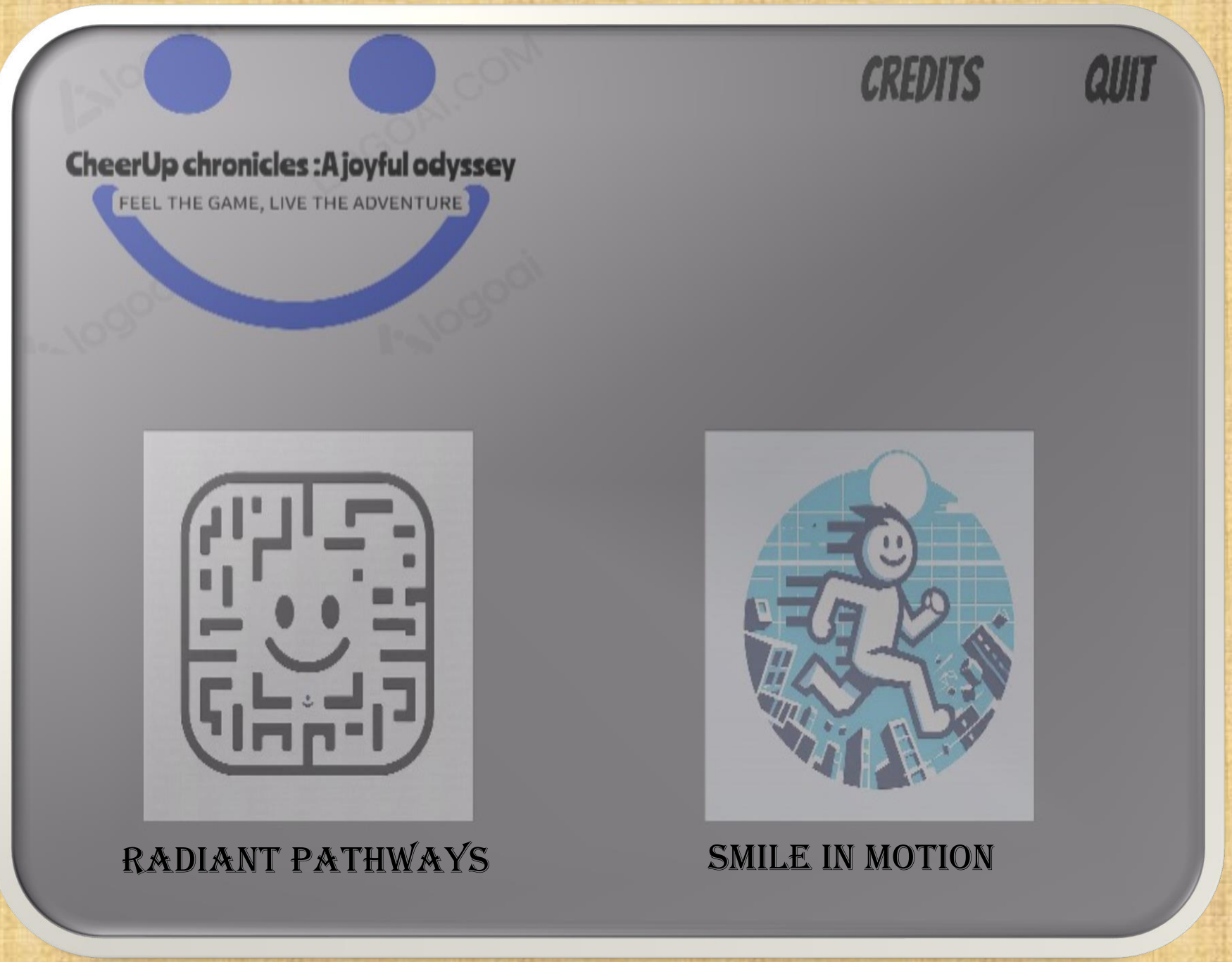
- Nevermind 2015

- Mindlight 2015

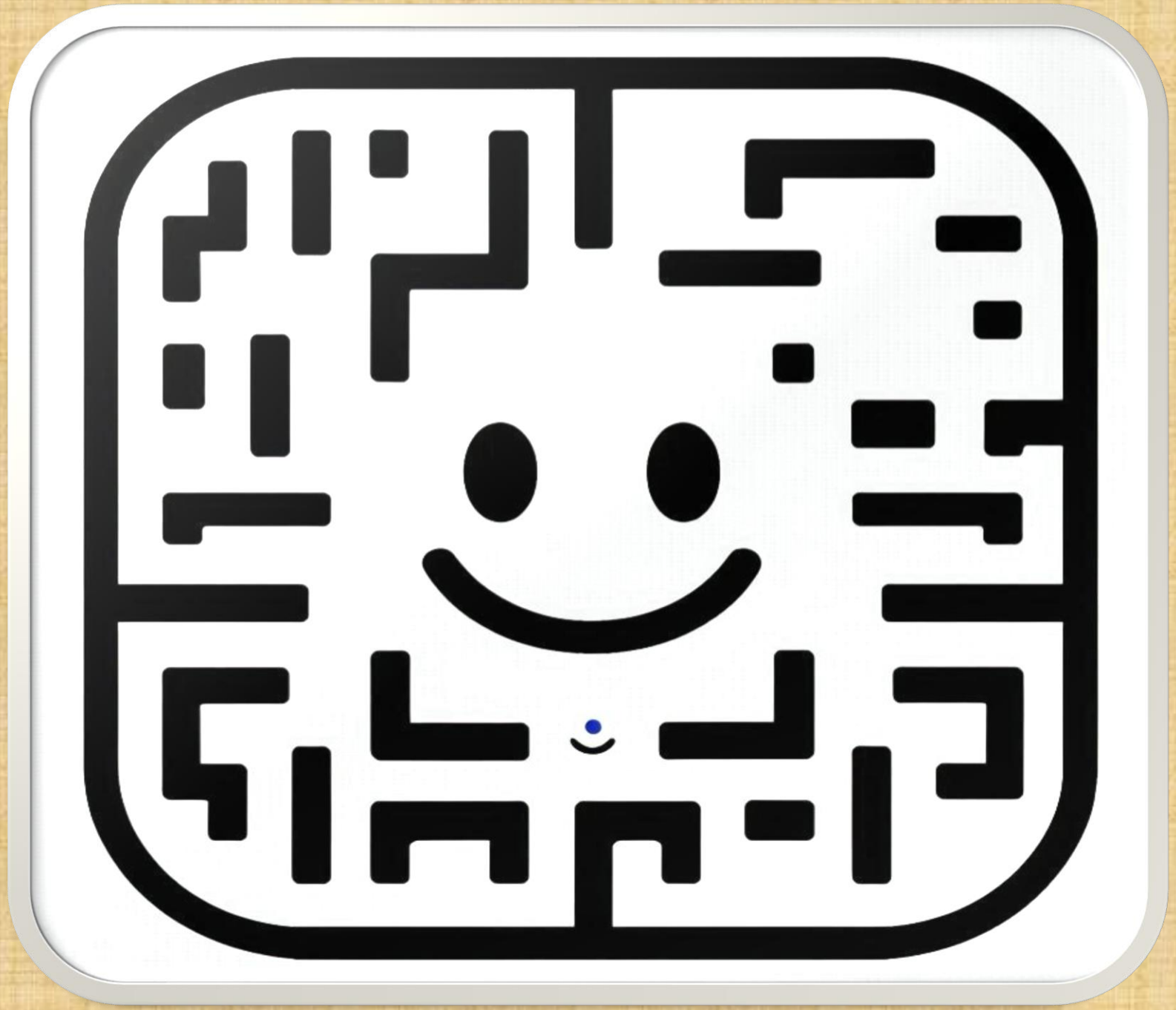
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4488062/


CHEER UP
CHRONICLES
A JOYFUL ODYSSSEY

Our game consists of two minigames Showed in simple menu We named the overall game CheerUp Chroconicles



1.Frist Game is Radiant Pathways



“Radiant Pathways"

is a computer-vision-based video game that leverages the player's smile to influence gameplay mechanics. The player controls a character to move around the maze and solve the puzzles on it, trying to survive and find End-point where enemies try to kill the character .The unique aspect of this game is the integration of real-time smile detection, which aids the character surviving in avoiding enemies in maze , and trying to solve the maze puzzle .

Game Concept

1

Player Control

You control a

A man wakes up in a dark maze

in this

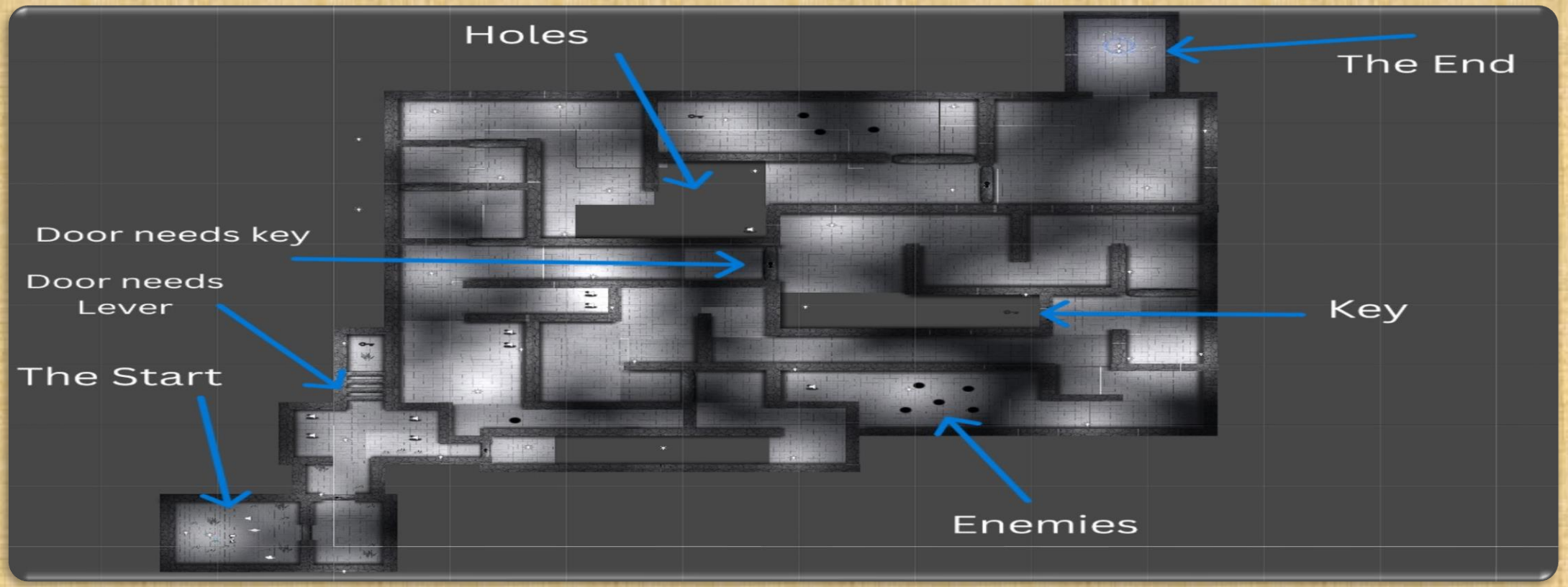
Top-down / Isometric game

where

WASD / Player's smile

makes the player

Move around the maze / solve the puzzles in it



Game Concept

1

Player Control

You control a *A man wakes up in a dark maze* in this *Top-down / Isometric* game

where *WASD / Player's smile* makes the player *Move around the maze / solve the puzzles in it*

2

Basic Gameplay

During the game,

Some obstacles, Gates and Traps	appear	*The depths of the darkness of the maze*

and the goal of the game is to

collect some memories of him and solve all the puzzles and not to die trying to get out of the maze

2

Basic Gameplay

During the game,

Some obstacles, Gates and Traps	appear	*The depths of the darkness of the maze*

and the goal of the game is to

collect some memories of him and solve all the puzzles and not to die trying to get out of the maze

3

Sound & Effects

There will be sound effects

when the gates open or when the obstacles come to the player

and particle effects

when the player smile

4

Gameplay Mechanics

As the game progresses,

the puzzles get difficult and obstacles become difficult to overcome

making it

hard to defeat the maze

[*optional*] There will also be

puzzles depend on player's smile only to increase the engagement

4

Gameplay Mechanics

As the game progresses,

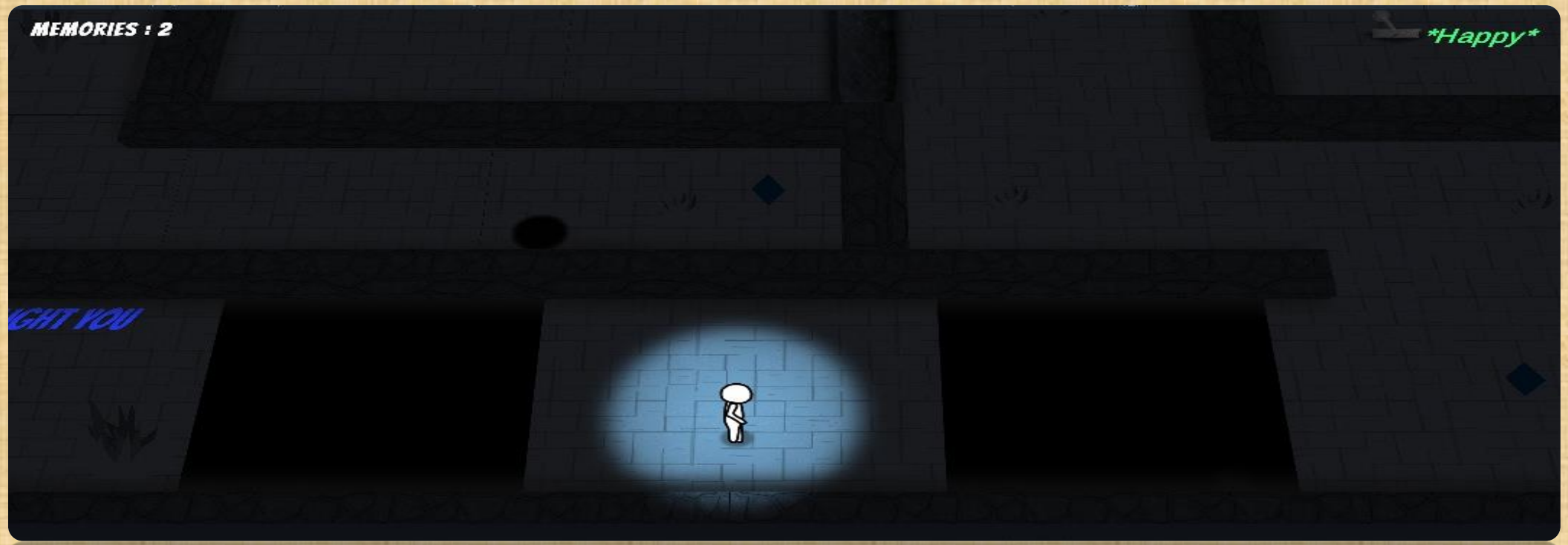
the puzzles get difficult and obstacles become difficult to overcome

making it

hard to defeat the maze

[*optional*] There will also be

puzzles depend on player's smile only to increase the engagement

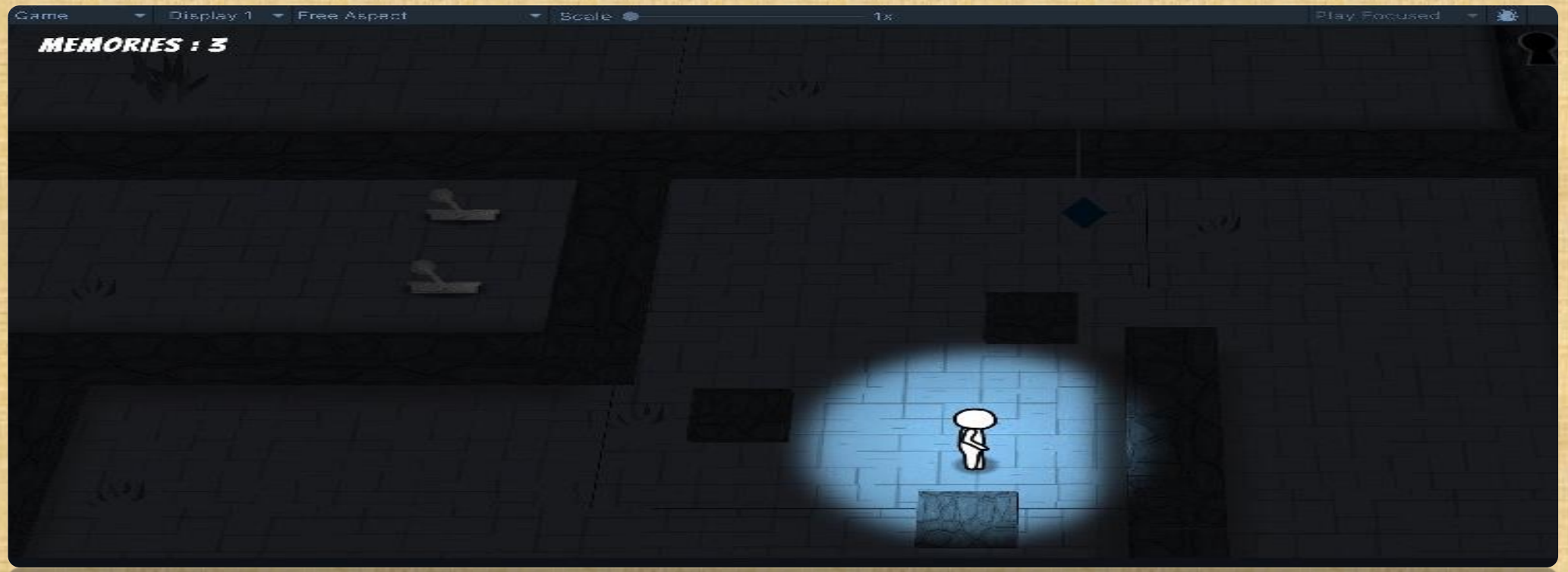


5

User Interface

The	will	whenever
Scor of memory *player's emotion*	*Increase* *Appear*	*the player takes a memory in his way* *the player smile*

At the start of the game, the title		and the game will end when
Radiant Pathways Echoes of Light Brighten the Darkness	will appear	*The player gets out of the maze with an optional quote in the credits*



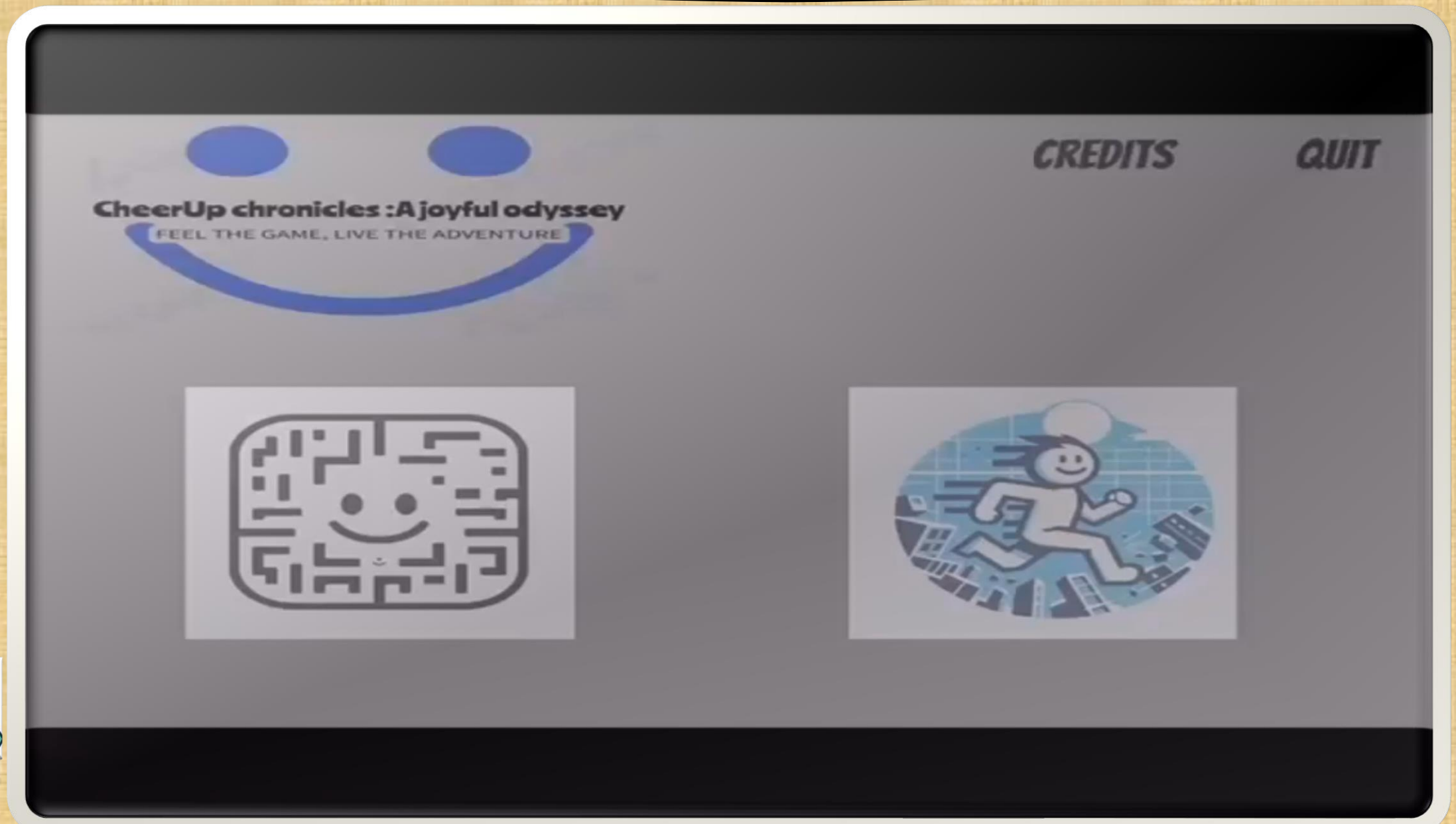
5

User Interface

The	will	whenever
Scor of memory *player's emotion*	*Increase* *Appear*	*the player takes a memory in his way* *the player smile*

At the start of the game, the title		and the game will end when
Radiant Pathways Echoes of Light Brighten the Darkness	will appear	*The player gets out of the maze with an optional quote in the credits*

Radiant Pathways Demo



2.Second Game is
“Smiles in Motion”

"Smiles in Motion"

is a computer-vision-based video game that leverages the player's smile to influence gameplay mechanics.
The player controls a character running at full speed over collapsing buildings, attempting to escape an unknown fate.
The unique aspect of this game is the integration of real-time smile detection, which aids the character in avoiding obstacles and surviving in this chaotic environment.

Game Concept

1
Player Control

You control a

A man running over falling buildings

in this

Side view / platformer game

where

Space / Player's smile

makes the player

Jump over the building / getting sharp things away of him / get another chance in game over menu

2

Basic Gameplay

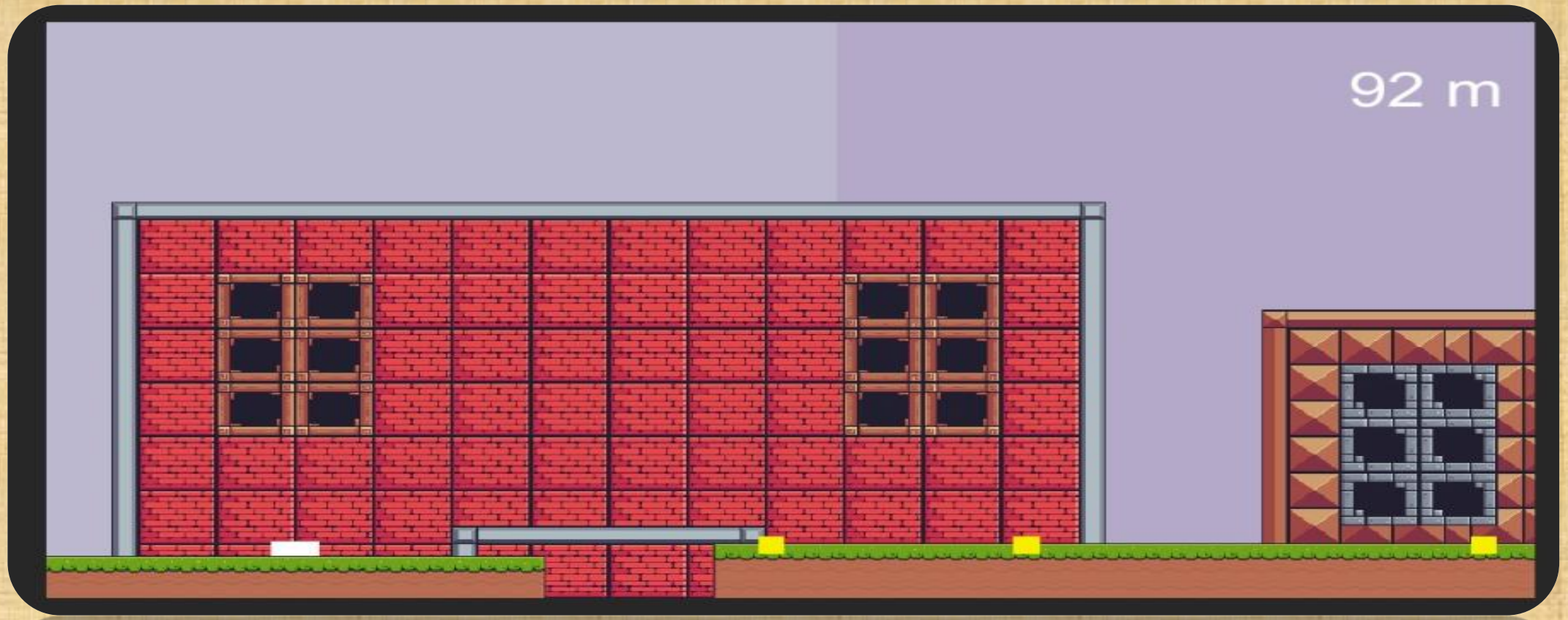
During the game,

Some cubes and sharp things	appear	*The right side of the screen*

from

and the goal of the game is to

Runs for miles as far as he can and try not to die early



3

Sound & Effects

There will be sound effects

when the building is falling
when the player jump

and particle effects

None and not needed

4

Gameplay Mechanics

As the game progresses,

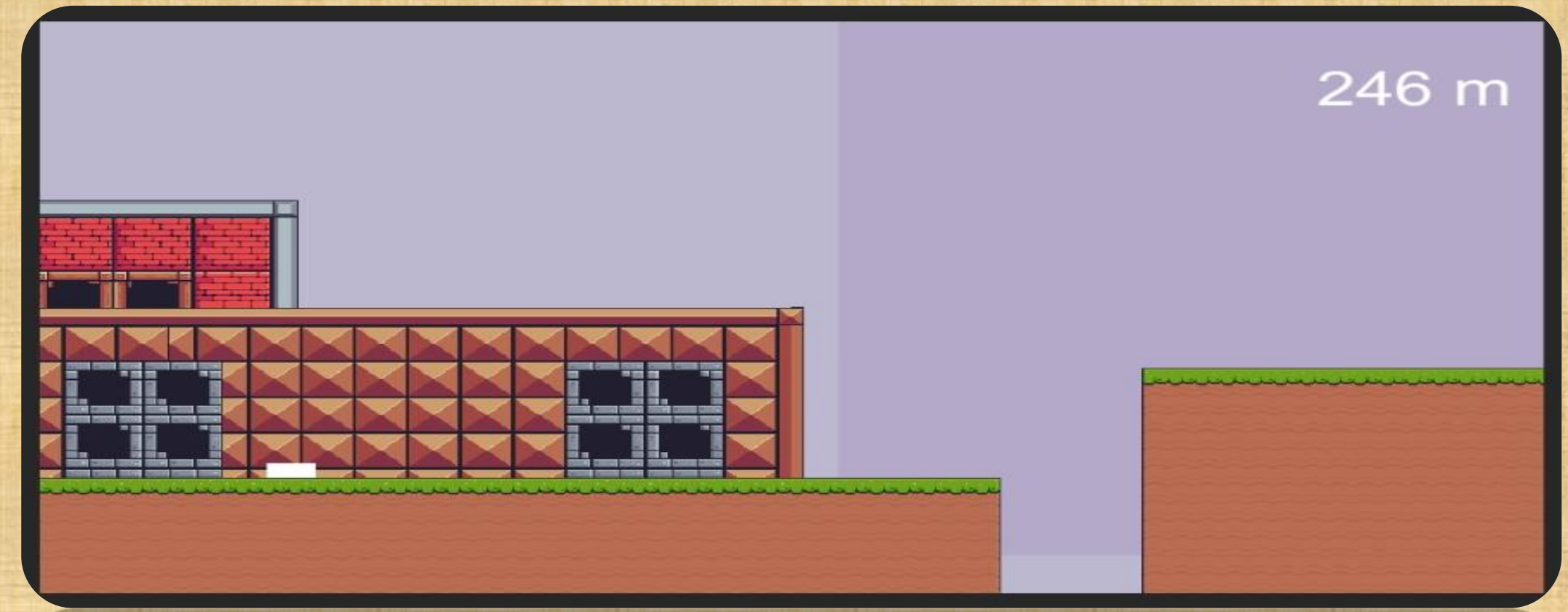
The buildings falls quickly and the number of evil cubes that make you slow increase

making it

More difficult to cut a long distance

[*optional*] There will also be

None



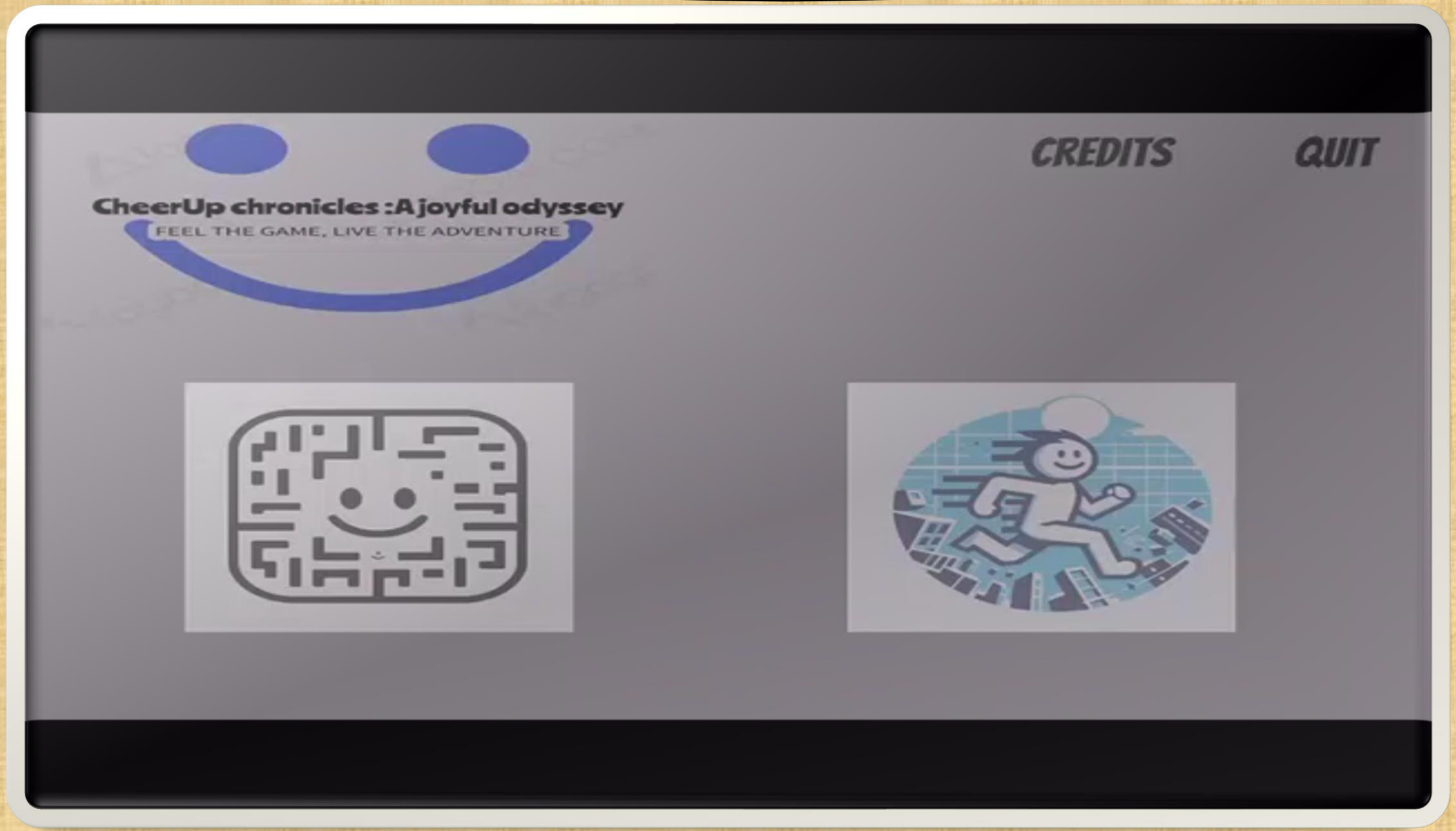
5

User Interface

The	will	whenever
miles he ran	*Increase*	*The player continues to run without falling*

At the start of the game, the title		and the game will end when
Smiles in motion	will appear	*The player falls off buildings and the miles he ran shows on a screen appear*

Smile Motion Demo



06 FUTURE WORK

Future Work

1- Broader Emotion Recognition

2- Improved Model Accuracy

3- Enhanced Game Design

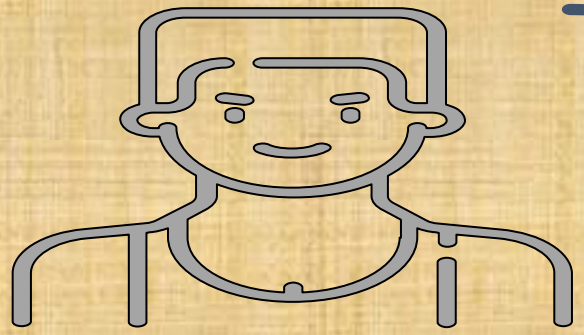
4- User Experience Studies

5- Cross-platform Integration

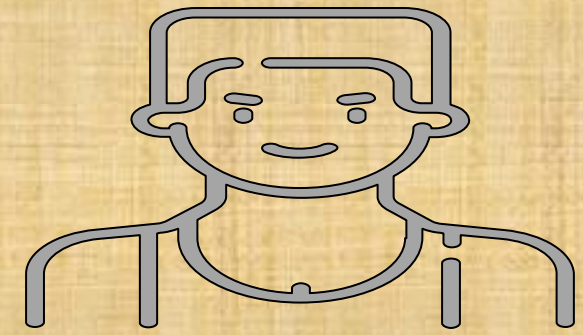
6- Therapeutic Applications

07 ABOUT US

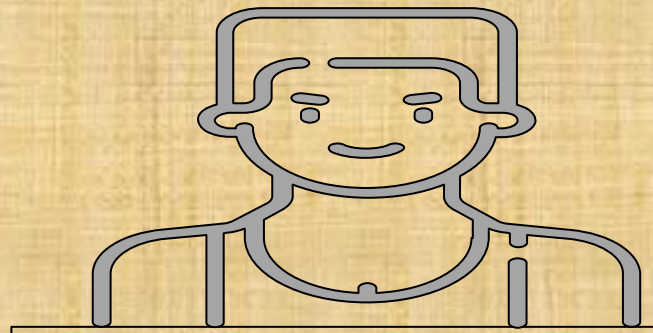
Our Team



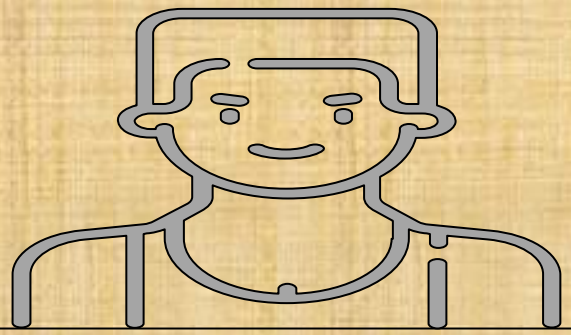
Shehab AlDin Mokhtar



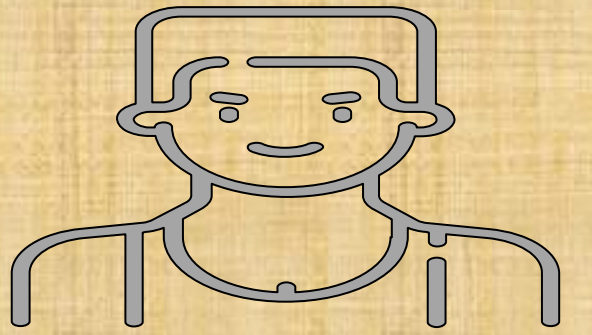
AbdelRahman Yasser



Youssef Mohammed



AbdelRahman Hamdy



Abdallah Shaaban