## TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | **PAGE NO.** |
|  | **ABSTRACT** | **iv** |
|  | **LIST OF FIGURES** | **viii** |
|  |  |  |
| **1** | **INTRODUCTION** |  |
|  | 1.1 Overview | **1** |
|  | 1.2 Objective | **2** |
| **2** | **LITERATURE SURVEY** | **3** |
| **3** | **SYSTEM ANALYSIS** |  |
|  | 3.1 Existing System | **5** |
|  | 3.1.1 Drawbacks of Existing system | **5** |
|  | 3.2 Proposed System | **5** |
|  | 3.2.1 Advantages | **5** |
| **4** | **SYSTEM SPECIFICATIONS**  4.1 Hardware Specification |  |
|  | **7** |
|  | 4.2 Software Specification | **7** |
|  | 4.2.1Unity | **7** |
|  | 4.2.1.1Advantages of Unity | **8** |
|  | 4.2.2Photoshop | **8** |
|  | 4.2.2.1Uses Of Photoshop | **9** |

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | **PAGENO.** |
|  | 4.2.3 Blender | **10** |
|  | 4.2.3.1 Features | **10** |
|  | 4.2.3.2 Advantages | **11** |
| **5** | **SOFTWARE DESIGN** |  |
|  | 5.1 UML | **14** |
|  | 5.1.1 Class Diagram | **14** |
|  | 5.1.2 Dataflow Diagram | **15** |
| **6** | **SYSTEM IMPLEMENTATION** |  |
|  | 6.1 Main menu | **16** |
|  | 6.2 Level Selection | **17** |
|  | 6.3 Game Play | **17** |
|  | 6.4 Level Complete | **17** |
| **7** | **SYSTEM TESTING**  **CONCLUSION & FUTURE ENHANCEMENT** | **20** |
| **8** | **22** |
|  | 8.1 Conclusion | **22** |
|  | 8.2Future Enhancement | **22** |
|  | **APPENDICES** | **23** |
|  | Source Code  Screenshot | **23**  **28** |
|  | **REFERENCES** | **32** |

|  |  |
| --- | --- |
| **LIST OF FIGURES** | |
| **FIGURE NAME** | **PAGE NO.** |
| Use Case Diagram | 14 |
| Data Flow Diagram | 15 |

**CHAPTER 1**

**INTRODUCTION**

* 1. **1.1 OVERVIEW**

Playing games is a mean of relieving stress buster in this period. Also the music plays an important role in relieving stress. This game aims at providing a stress buster with both the integrated concept of game and music. It uses the basics of day to day game play and incorporates them with the energetic music. This android application allows player to play the game of casual and entertainment category. It majorly dwells the fun part with the basic concept of game with music and offline. The system provides a platform for android user.

Every year there are many games launched into the market as a source for entertainment and fun. Since the games are always focused on playing with the friends through online, quite a lot of them choose to play games and relieve stress in the hectic life style but everyone will also feel to be alone but during that time those large games would not be a best choice. Because of this reason this game have important role in providing fun and entertainment. And also, a casual game can’t provide fun. So the idea here is to provide a fun game to relive stress with the entertainment category (music). By using the offline game style concept, people can get into fun all the time when playing this game. The reason for choosing the genre of game category over online and multi-player was because of the factor of necessity a user can be of any age category and he can play this game any time the user wishes to. A user whoever engages with this game can experience the having stress relief anytime.

The aim of this system is to construct an android platform game which is used by almost everyone. This android app is implemented using various technologies such as Unity, Photoshop and blender. Unity is a free game engine used by many industries for production of cool games. Photoshop provides a platform for user to edit the photos whatever the way user needs. Blender is a tool for animation and also sculptingtool.

## 1.2 OBJECTIVE

Beatbox, is an offline game focusing on the category of casual and entertainment. Through this game a user can play without an internet connection. The user can play this game wherever they want and can be played in short period of break the user gets. The player can enjoy the music and play game simultaneously. The game focuses on giving player an immense fun and entertainment with both music and platform running game.

The main objective is to overcome the usual category of games by an old-school concept offline game. The game contains of energetic music and 3D platform running game.

The objective of player is to continue the game without hitting any obstacle and enjoying the beat with the music by hitting the music chords.

* Provide offline game with fun and entertainment
* Conservation of time and internet cost
* Convenient for people to play the game anywhere and anytime.
* Users are able to hear the music and play game simultaneously.
* Users can easily overcome the stress.
* It is easy to play game as it can be used by any age category.

# CHAPTER 2

## LITERATURE SURVEY

### 2.1 DEFINITION OF GAME:

A **game** is a structured form of [play](https://en.wikipedia.org/wiki/Play_(activity)), usually undertaken for [entertainment](https://en.wikipedia.org/wiki/Enjoyment) or [fun](https://en.wikipedia.org/wiki/Fun), and sometimes used as an [educational](https://en.wikipedia.org/wiki/Education) tool. Games are distinct from [work](https://en.wikipedia.org/wiki/Manual_labor), which is usually carried out for [remuneration](https://en.wiktionary.org/wiki/remuneration), and from [art](https://en.wikipedia.org/wiki/Art), which is more often an expression of aesthetic or ideological elements.

**Playstore:** This provides the games for android of large category.

**Steam:** This will provide games for pc platform.

### 2.2 LACK OF AVAILABILITY:

In above surveys they are providing games for their own platforms of large categories, some of them are providing games with the category to play in online only and mixed. Also they are providing in game purchases for skins and much other stuff. This application provides an experience of fun and entertainment with both game and music.

**2.3 LACK OF PROVIDING RELAXATION:**

The games nowadays are meant for playing with using all sense such as hearing, reflexes, fast movements of hands and eyes. So, initially the game may bring excitement but later on when player loosing game the player becomes stressed but this game in countering those action music are bought in which plays a main role in neutralizing the stress.

**2.4 BRINGING IN COMPETITION:**

The games with usage of internet bring the concept of leader board. Leader Board is nothing but the rank of player among the player’s friends or the entire region. This makes game extra stressful thing to continue to play on. This game has no leader board only based on to provide fun.

**2.5 TEMPTED FOR IN GAME PURCHASE**

As a business part for a game developer he creates the skins and outfits which are available only to them when they buy in game. So, in case of not getting the desired product the player buys again and again until he gets it.

# CHAPTER 3

## SYSTEM ANALYSIS

**3.1 EXISTING SYSTEM**

The existing system gives user in game purchases and online mode which costs. To complete a task in game people are tempted to buy skins and pay for unlocking levels and in online if player gets defeated they are sent into a depression. The latest games follow an online system for maintaining the game from intruders.

* Whenever the player in online game is connected via internet they are vulnerable to get depressed by hackers.
* The game which doesn’t need to have internet connection.

## 3.1.1 DRAWBACKS OF EXISTING SYSTEM

Each and every system has certain drawbacks that lead them in to the improper working. Our offline game only concentrates on providing fun, while online games may contain hackers which results in easy defeat of players may again lead them into depression.

* The player in online are tempted to buy skins and other things.
* The biggest problem in existing games is online and they require a stable internet connection to play the game which also makes player worry.

## 3.2 PROPOSED SYSTEM

## In this proposed system, there is no need of internet connection and need not to be tempted by developer for in game purchases. The game comprises of music system which will relieve stress. Game comprises of many music which is of different languages and different energetic music.

## 3.2.1 ADVANTAGES

* No need of internet connection.
* No need of in game purchases.
* Can be played whenever and wherever.

# CHAPTER 4

## SYSTEM SPECIFICATION

System Requirements Specification (SRS) the requirements work product that formally specifies the system-level requirements of a single system or an application. The System Requirements Specification identifies, defines and clarifies the requirements, that when satisfied through development meet the operational/functional.

## 4.1 HARDWARE SPECIFICATION

Processor : Intel Pentium IV and higher versions

Speed : 2.8 GHz

Hard Disk : 32 GB

RAM : 6 GB or more

Internet : 100 Kbps Network

## 4.2 SOFTWARE SPECIFICATION

**4.2.1 UNITY**

**Unity is a** [**cross-platform**](https://en.wikipedia.org/wiki/Cross-platform) [**game engine**](https://en.wikipedia.org/wiki/Game_engine) **developed by** [**Unity Technologies**](https://en.wikipedia.org/wiki/Unity_Technologies)**, first announced and released in June 2005 at** [**Apple Inc.**](https://en.wikipedia.org/wiki/Apple_Inc)**'s** [**Worldwide Developers**](https://en.wikipedia.org/wiki/Apple_Worldwide_Developers_Conference)[**Conference**](https://en.wikipedia.org/wiki/Apple_Worldwide_Developers_Conference) **as a** [**Mac OS X**](https://en.wikipedia.org/wiki/MacOS)**-exclusive game engine. As of 2018, the engine had been extended to support more than 25 platforms. The engine can be used to create** [**three-dimensional**](https://en.wikipedia.org/wiki/Three-dimensional_space)**,** [**two-dimensional**](https://en.wikipedia.org/wiki/Two-dimensional_space)**, virtual reality, and augmented reality games, as well as** [**simulations**](https://en.wikipedia.org/wiki/Computer_simulation) **and other experiences. The engine has been adopted by industries outside video gaming, such as** [**film**](https://en.wikipedia.org/wiki/Film_industry)**,** [**automotive**](https://en.wikipedia.org/wiki/Automotive_industry)**,** [**architecture**](https://en.wikipedia.org/wiki/Architecture)**,** [**engineering**](https://en.wikipedia.org/wiki/Engineering) **and** [**construction**](https://en.wikipedia.org/wiki/Construction)**.**

**Several major versions of Unity have been released since its launch. The latest stable version, 2020.1.16, was released in December 2020.**

## 4.2.1.1 ADVANTAGES OF UNITY

## 

## Platform support

The engine is highly preferred for its extended support to 27 platforms. The app developed and deployed can be easily shared between PC, web and mobile platforms. Besides, the agile methodology enables speedy prototyping and constant releases, which in turn speed up the game development.

### IDE

The text editor is provided by IDE to write the code, but sometimes a distinct code editor is also used by the developers to alleviate confusion. Additionally, the integrated development editor support JavaScript and C# for scripting, and also offers notable features that are ideal for the game development.

### Graphics

The high quality audio and visual effects are supported by the engine that eases the game development. The visuals are adaptable on every screen and device without any distortion or compromise with the image quality.

### Documentation

It’s a must have. The novice developers need the easy-to-understand documentation that’s provided in detail by the Unity engine. The detailed documentation includes the explanation of every small topic.

### Debugging

The debugging and tweaking is amazingly easier with Unity game development because all the game variables are displayed during gameplay, which in turn allow the developers to debug the process at runtime

**4.2.2 PHOTOSHOP**

Adobe Photoshopis a [raster graphics editor](https://en.wikipedia.org/wiki/Raster_graphics_editor) developed and published by [Adobe](https://en.wikipedia.org/wiki/Adobe_Inc) [Inc.](https://en.wikipedia.org/wiki/Adobe_Inc) for [Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) and [macOS](https://en.wikipedia.org/wiki/MacOS). It was originally created in 1988 by [Thomas](https://en.wikipedia.org/wiki/Thomas_Knoll) and [John Knoll](https://en.wikipedia.org/wiki/John_Knoll). Since then, the software has become the industry standard not only in [raster graphics](https://en.wikipedia.org/wiki/Raster_graphics) editing, but in [digital art](https://en.wikipedia.org/wiki/Digital_art) as a whole. Photoshop can edit and compose raster images in multiple layers and supports [masks](https://en.wikipedia.org/wiki/Mask_(computing)), [alpha compositing](https://en.wikipedia.org/wiki/Alpha_compositing) and several [color](https://en.wikipedia.org/wiki/Color_model) [models](https://en.wikipedia.org/wiki/Color_model) including [RGB](https://en.wikipedia.org/wiki/RGB_color_model), [CMYK](https://en.wikipedia.org/wiki/CMYK_color_model), [CIELAB](https://en.wikipedia.org/wiki/CIELAB), [spot color](https://en.wikipedia.org/wiki/Spot_color), and [duotone](https://en.wikipedia.org/wiki/Duotone). Photoshop uses its own PSD and PSB file formats to support these features. In addition to raster graphics, Photoshop has limited abilities to edit or render text and [vector](https://en.wikipedia.org/wiki/Vector_graphics) [graphics](https://en.wikipedia.org/wiki/Vector_graphics) (especially through [clipping path](https://en.wikipedia.org/wiki/Clipping_path) for the latter), as well as [3D](https://en.wikipedia.org/wiki/3D_graphics) [graphics](https://en.wikipedia.org/wiki/3D_graphics) and [video](https://en.wikipedia.org/wiki/Video). Its feature set can be expanded by [plug-ins](https://en.wikipedia.org/wiki/Photoshop_plug-in); programs developed and distributed independently of Photoshop that run inside it offers enhanced features.

## 4.2.2.1 USES OF PHOTOSHOP

* Several cutting-edge tools at your disposal.
* Adobe Photoshop offers unmatched editing features.
* An integrated stock library that inspires creativity.
* Even the simplest and toughest of editing can be done with ease.
* Can pretty much edit all types of images with ease.
* Easily transforms ordinary photographs into professionally done photographs.
* Easily renders images with high quality.
* Allows graphic designing on a precise scale.
* Adobe Photoshop is all-purpose software.
* Include features that enable editing of video or animation layers.
* Allows transfer of files from one program to another with ease.

## 4.2.3 BLENDER

**Blender** is a [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source) [3D computer graphics](https://en.wikipedia.org/wiki/3D_computer_graphics) software toolset used for creating [animated films](https://en.wikipedia.org/wiki/Animation), [visual effects](https://en.wikipedia.org/wiki/Visual_effects), art, [3D printed](https://en.wikipedia.org/wiki/3D_printing) models, [motion graphics](https://en.wikipedia.org/wiki/Motion_graphics), interactive 3D applications, [virtual reality](https://en.wikipedia.org/wiki/Virtual_reality) and [computer games](https://en.wikipedia.org/wiki/Computer_game). Blender's features include [3D modeling](https://en.wikipedia.org/wiki/3D_modeling), [UV unwrapping](https://en.wikipedia.org/wiki/UV_mapping), [texturing](https://en.wikipedia.org/wiki/Texture_mapping), [raster graphics editing](https://en.wikipedia.org/wiki/Raster_graphics_editor), [rigging](https://en.wikipedia.org/wiki/Skeletal_animation) [and skinning](https://en.wikipedia.org/wiki/Skeletal_animation), [fluid and smoke simulation](https://en.wikipedia.org/wiki/Fluid_simulation), [particle](https://en.wikipedia.org/wiki/Particle_system) simulation, [soft](https://en.wikipedia.org/wiki/Soft_body_dynamics)

[body](https://en.wikipedia.org/wiki/Soft_body_dynamics) simulation, [sculpting](https://en.wikipedia.org/wiki/Digital_sculpting), [animating](https://en.wikipedia.org/wiki/Computer_animation), [match moving](https://en.wikipedia.org/wiki/Match_moving), [rendering](https://en.wikipedia.org/wiki/Rendering_(computer_graphics)), [motion](https://en.wikipedia.org/wiki/Motion_graphics) [graphics](https://en.wikipedia.org/wiki/Motion_graphics), [video editing](https://en.wikipedia.org/wiki/Video_editing_software), and [compositing](https://en.wikipedia.org/wiki/Compositing). This section lists many of the optional features available in blender and gives a brief overview of their use for beginning users.

Audio device support

Support for Jack, OpenAL, or SDL audio can optionally be enabled through their respective USE flags.

Inside Blender, go to the Edit->Preferences->System tab and set the Audio Dev to the desired setting.

CUDA support

Cycles renderer can work on GPUs, for example Nvidia GTX 970 is about twice as fast as an i5 4690k on traditional BMW benchmark.To enable graphics card rendering with Nvidia graphics cards, install Cuda:Inside Blender, go to the Edit->Preferences->System tab and set Compute Device to CUDA and select the graphics card in the box below. If the graphics card is not supported these options will not be visible.

The first time a render is created with a new version of blender, the CUDA kernels will need to be compiled. This may take over 15 minutes.Now set the renderer to Cycles Renderer and in the renderer panel under the Render options set the Device to GPU Compute.

The first time a render is created with a new version of blender, the CUDA kernels will need to be compiled. This may take over 15 minutes.

File format support

Support OpenCOLLADA (.dae), jpeg2k, sndfile, and tiff image file formats can optionally be enabled through USE flags.

Blender should work with either ffmpeg or libav libraries, although only ffmpeg is officially recommended by the Blender developers

**4.2.3.1 BLENDER FEATURES**

* [Modeling](https://www.blender.org/features/modeling/)
* [Sculpting](https://www.blender.org/features/sculpting/)
* [Animation & Rigging](https://www.blender.org/features/animation/)
* [Grease Pencil](https://www.blender.org/features/grease-pencil/)
* [Rendering](https://www.blender.org/features/rendering/)
* [Simulation](https://www.blender.org/features/simulation/)
* [Video Editing](https://www.blender.org/features/video-editing/)
* [Scripting](https://www.blender.org/features/scripting/)
* [VFX](https://www.blender.org/features/vfx/)
* [Interface](https://www.blender.org/features/interface/)

## 4.2.3.2 ADVANTAGES

* Blender is free and open source. The fact that Blender is free make it much more powerful than other tools that are available on the market. Unlike other free software that have revenue capacity on it, Blender is developed to be free for any purpose, including commercially or for education. It doesn’t require licensing or payment. Even Pixar has developed a version of “Renderman” (a rendering software) base on some of it’s technology and offer a special version of the software called “Blenderman”. Blender is being actively developed by hundreds of contributors from over the world. This include professional 3D designers, animators, artists, VFX designers, ect. In fact, Blender is the
* Blender comes with tons of libraries, free character download, and tutorials. Since 2007, when Blender started to advance, it starts building up its user base. The Blender cloud library serves thousands of textures from animals, humans, eyes to clouds, rocks, trees or even wood. On the same clou, the provide training for character modeling, 3D printing, VFX, rigging and even

advance digital painting. They come as far as featuring user stories interacting with Blender. For a free software, Blender is a strong, powerful and robust tool.

**Unity Remote**

The app connects with Unity while you are running your project in Play Mode from the Unity Editor. The visual output from the Editor is sent to the device’s screen, and the live inputs are sent back to the running project in Unity. This allows you to get a good impression of how your game really looks and handles on the target device, without the hassle of a full build for each test.

### Features

Unity Remote currently supports Android devices (on Windows and OS X via a USB connection) and iOS devices (iPhone, iPad, iPod touch and Apple TV, through USB on OS X and Windows with iTunes).

The Game View of the running Unity project is duplicated on the device screen, but at a reduced framerate. The following input data from the device is also streamed back to the Editor:

* Touch and stylus input
* Accelerometer
* Gyroscope
* Device **camera** streams
* Compass
* GPS
* Joystick names and input

### Advantages

With Unity Remote 5, you can use an Android device to view and test your game live, right inside the Unity Editor 5.4 or later. Unity Remote 5 makes your

android device act as remote control. It streams touch, accelerometer, gyroscope, webcam and screen orientation change events back to Unity Editor

**CHAPTER 5**

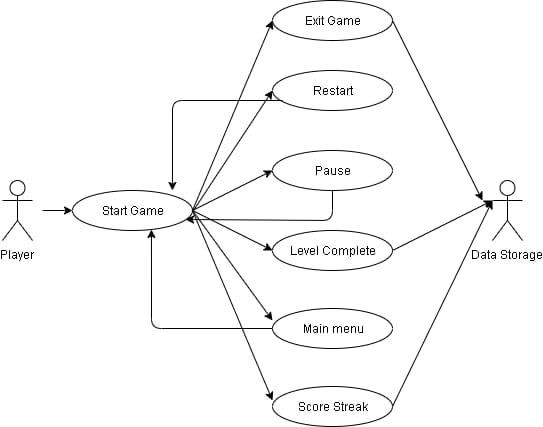
## SOFWARE DESIGN

## 5.1 UML

In the field of software engineering, the **Unified Modeling Language** (**UML**) is a standardized visual specification language for object modeling. UML is a general-purpose modeling language that includes a graphical notation used to create an abstract model of a system, referred to as a UML model

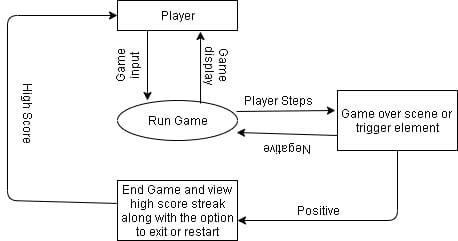
## 5.1.1 USECASE DIAGRAMS

It shows a set of use cases and actors and their relationships. These diagrams illustrate the static use case view of a system and are important in organizing and modeling the behaviors of a system.



Here in this use case diagram the player’s involvement with the game and UI is described. The player has the role on nearly every process in the game. Whereas the data storage has the functionality to store the high score streak value in the game and saves it retrive the data.

## 5.1.2 DATAFLOW DIAGRAM:

A data flow diagram (DFD) is a graphical representation of the “flow” of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated.

The data flow diagram explains the flow of data in the proposed system. The player first chooses the level which one to be played. Then the user goes for play window and plays the game. The player gives input and displays back, then based upon the input either of the scenes will be displayed. If positive then it leads to end game scene or else turns back to run game.

# CHAPTER 6

## SYSTEM IMPLEMENTATION

Implementation is the stage in the project where the theoretical design is turned into a working system. It involves careful planning, investigation of the current system and its constraints on implementation, methods to achieve the changeover, an evolution of the methods implemented.

Apart from planning, major task of preparing the implementations are education and training of users. The implementation process begins with preparing a plan for the implementation system. After the implementation training of the user is important task of the developer.

## MODULES

### 6.1 MENU

This is the section where the user will be navigated from this selection. This contains UI elements which directs player to level (Song).

The menu module is the opening window of the game. Which has the following buttons:

* Play Button
* Option Button
* Quit Button

**PLAY BUTTON**

The play button directs into the next window which is LevelSelection window

**OPTION BUTTON**

The option button controls the sensitivity.

**QUIT BUTTON**

The quit button closes the application.

### 6.2 LEVEL SELECTION

This section of UI contains the song selection buttons which directs into 3D world.

The level selection for now contains 3 buttons which directs the player on clicking into its scene. The 3 buttons are the song containing container. The songs are Spectre, Sirikalam parakalam and Vaathi Coming which are music with increasing difficulty.

### 6.3 GAME PLAY

This section involves the world where the player involves with the character (cube), and hits the music chords to produce the sound of beats. The Music is played in the background, and the music chords are placed in such a manner that it syncs with the music beats.

So, the music and beats in the game produces the sound of beats which resonates with the music’s beat and musical chord in the game.

The Game play window contains the panel with score and pause button. The score displays the number of chords have been hit by the player which acts as a point.

If player lose the game by hitting on with any of the obstacle then it directs the player into window which displays the buttons: restart and home button.

### In Game Menu

This section involves the in-game pause option which sends player to the UI menu with the buttons containing resume, home menu and exit option

* The pause button directs into the window containing the options menu, resume button, restart button and Home button.
* Resume button plays the game where player have paused the game.
* Home button redirects you to the window containing levels.

### 6.4 LEVEL COMPLETION

The level completion is displayed when the player have completed without hitting any obstacle and displays the UI containing restart and main menu button.

The game play window contains the end trigger which will be at the end of the music which depicts the player he has completed the level.

Score Streak

The score streak is nothing but continuously how many music chords have been hit by the player.

### Unity Asset Store

The unity asset store is for downloading assets like sprites, lighting effects, and characters

# CHAPTER 7

## SYSTEM TESTING

Software testing is a critical element of software quality assurance and represents the ultimate reviews of specification, design and coding. Testing represents interesting anomaly for the software. During earlier definition and development phases, it was attempted to build software from an abstract concept to tangible implementation. The testing phase involves the testing of the developed system using various test data. Preparation of the test data plays a vital role in the system testing. After preparing the test data the system under study was tested using those test data. While testing the system, errors were found and corrected by using the following testing steps and corrections are also noted for future use. Thus, a series of testing is performed for the proposed system, before the system was ready for the implementation. Testing is the process of detecting errors. Testing performs a very critical role for quality assurance and for ensuring the reliability of software. The results of testing are used later on during maintenance also the aim of testing is often to demonstrate that a program works by showing that it has no errors. The basic purpose of testing phase is to detect the errors that may be present in the program. Hence one should not start testing with the intent of showing that a program works, but the intent should be to show that a program doesn’t work.

* **Functionality testing** – this requires the testers to play the game while at the same time looking for general problems with the game itself or the user interface.
* **Compliance testing** – this type of testing is usually done by a game testing lab. It can also refer to testing by a regulatory body, for example, if the game requires a certain rating. Testers in this instance are looking for inappropriate or objectionable content.
* **Compatibility testing** – this type of testing is usually performed for PC titles. Major functions of the game will be tested using a variety of hardware. It is performed to make certain that a game will run the same regardless of the hardware and software being used.
* **Soak testing** – this type of test requires the game to be left running for long periods of time. Various modes will be used, as in pause, idle or running the title screen. Once the initial setup has been performed there is no need for any user interaction.
* **Regression testing** – this is performed once a bug has been found and the programmers have fixed it. The aim is to check whether the bug is still there and whether the fix caused something else to break.
* **Load testing** – this is designed to test heavy activity and whether the application is able to function properly under load.
* **Mobile game testing** – this testing speaks for itself. All the above tests play a part but in relation to the popular mobile game platforms such as Android and iOS.

# CHAPTER 8

## CONCLUSION & FUTURE ENHANCEMENT

## CONCLUSION

The Beatbox game provides fun and entertainment to the users or players. The main purpose of this proposed system is to compensate the online games and to be used all age category. The game also has some energetic songs and obstacle which pumps adrenaline. Each module provides a feature to find out the game’s process. All the modules at last are combined to get the solution of the complete system.

## 

## FUTURE ENHANCEMENT

The proposed work has a wide scope for development by including some extra features such as

* Skins which can be unlocked by completing levels with respective score streak.
* The game can also be included with the user’s song request box.
* More and more songs can be included to bring out player’s excitement.
* Environment can be built so the game looks complete.
* Color changing mechanism with the hit of the musical chords.

## APPENDICES

## SOURCE CODE

### Player movement

using System.Collections;

using System.Collections.Generic; using UnityEngine;

public class playermov : MonoBehaviour

{

public Rigidbody rb; public float ff = 1000f; public float sf = 500f; Touch touch;

float speedModifier; void Start()

{

rb.GetComponent<Rigidbody>(); speedModifier = 100f;

}

private void Update()

{

transform.position= new Vector3(Mathf.Clamp(transform.position.x, -6.5f, 6.5f),transform.position.y,transform.position.z);

}

void FixedUpdate()

{

rb.AddForce(0, 0, ff \* Time.deltaTime); if (Input.touchCount>0)

{

touch = Input.GetTouch(0);

if (touch.phase == TouchPhase.Moved)

{

transform.position = new Vector3(transform.position.x + touch.deltaPosition.x \* speedModifier, transform.position.y, transform.position.z);

}

}

if (rb.position.y < -1f) { FindObjectOfType<gamemanager>().endgame();

}

}

}

### Transition

using System.Collections;

using System.Collections.Generic; using UnityEngine;

using UnityEngine.SceneManagement;

public class levelloader : MonoBehaviour

{

public Animator transition; public float transtime = 1f; void Update()

{if (Input.GetMouseButtonDown(0))

{

LoadNextLevel();

}

}

public void LoadNextLevel()

{

StartCoroutine( Loadlevel(SceneManager.GetActiveScene().buildIndex + 1));

}

IEnumerator Loadlevel(int levelIndex) {

transition.SetTrigger("start");

yield return new WaitForSeconds(transtime); SceneManager.LoadScene(levelIndex);

}

}

### Camera following

using System.Collections;

using System.Collections.Generic; using UnityEngine;

public class camfollow : MonoBehaviour

{

public Transform player; public Vector3 offset;

// Update is called once per frame void Update()

{

transform.position = player.position+offset;

}

}

### Collision

using System.Collections;

using System.Collections.Generic; using UnityEngine;

public class collidsonp : MonoBehaviour

{

public playermov pm;

void OnCollisionEnter(Collision collinfo)

{

if (collinfo.collider.tag == "Obstacle") { pm.enabled = false; FindObjectOfType<gamemanager>().endgame();

} }

}

### End trigger

using System.Collections;

using System.Collections.Generic; using UnityEngine;

public class endtrigger : MonoBehaviour

{

public gamemanager gamemanager;

private void OnTriggerEnter(Collider other)

{

gamemanager.completelevel();

}

}

### Game Manager

using System.Collections;

using System.Collections.Generic; using UnityEngine;

using UnityEngine.SceneManagement;

public class gamemanager : MonoBehaviour

{

bool gameended = false; public float restartdelay = 1f;

public GameObject completelevelui; public void completelevel() {

completelevelui.SetActive(true);

}

public void endgame()

{ if (gameended == false)

{

gameended = true; Debug.Log("go"); Invoke("restart", restartdelay);

}

}

void restart() { SceneManager.LoadScene(SceneManager.GetActiveScene().name);

}

}

### ScoreUI

using System.Collections;

using System.Collections.Generic; using UnityEngine;

using UnityEngine.UI;

public class score : MonoBehaviour

{

public Transform player; public Text scoret;

void Update()

{

scoret.text = player.position.z.ToString("0");

}

}

### Levelcompletion

using System.Collections;

using System.Collections.Generic; using UnityEngine;

using UnityEngine.SceneManagement;

public class levecomplete : MonoBehaviour

{

public void loadnextlevel()

{

SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex+1);

}

}

### PlayButton

using System.Collections;

using System.Collections.Generic; using UnityEngine;

using UnityEngine.SceneManagement;

public class credits : MonoBehaviour

{

public void startgame()

{

SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex + 1);

}

}

## SCREENSHOTS

**Main Menu**



Main menu of the game, which is the starting window of the game.

### Level Selection



The Song selection Menu of the game, this window is used to select the song from the list.

### Transition window



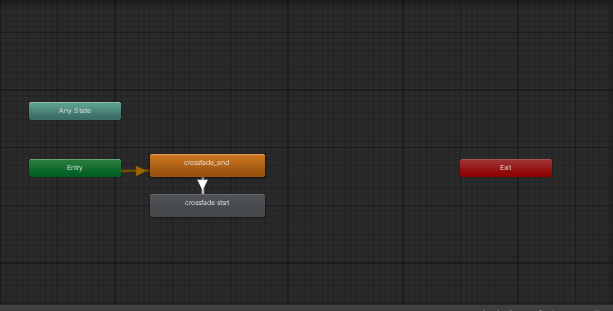
The transition window of the game which is used to deliver effect of smoothness.

**Game Window**



The Gameplay picture of the beatbox game.

### Animator window



The animator Window of the game which is for implemented in the level selection.

# REFERENCES

1. Adobe® Photoshop® CC Help, Adobe Inc, 2018.
2. Blender Documentation, Blender organization.
3. Unity Documentation, Unity Technologies.