

# EMOTION BASED SONG RECOMMENDATION

By

**Amimul Ehshan**

**142-15-3523**

**Md. Naymul Haque Dider**

**142-15-4041**

And

**Md. Asaduzzaman Sagor**

**142-15-3979**

This Report Presented in Partial Fulfillment of the Requirements for the  
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

**Mr. Mohshi Masnad**

Lecturer

Department of CSE

Daffodil International University

Co-Supervised By

**Nusrat Jahan**

Lecturer

Department of CSE

Daffodil International University



**DAFFODIL INTERNATIONAL UNIVERSITY**

**DHAKA, BANGLADESH**

**MAY 2018**

## **APPROVAL**

This Project/internship titled “**Emotion Based Song Recommendation**”, submitted by Amimul Ehshan, ID No: 142-15-3523, Md. Naymul Haque Dider, ID No: 142-15-4041, Md. Asaduzzaman Sagor, ID No: 142-15-3979 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on MAY 2018.

## **BOARD OF EXAMINERS**

---

**Dr. Syed Akhter Hossain**

**Professor and Head**

Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University

**Chairman**

---

**Dr. Sheak Rashed Haider Noori**

**Associate Professor and Associate Head**

Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University

**Internal Examiner**

---

**Md. Zahid Hasan**

**Assistant Professor**

Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University

**Internal Examiner**

---

**Dr. Mohammad Shorif Uddin**

**Professor**

Department of Computer Science and Engineering  
Jahangirnagar University

**External Examiner**

## DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Mr. Mohshi Masnad, Lecturer and Department of CSE**, Daffodil International University. We also declare that neither this project nor any part of this project has been submitted anywhere for award of any Competition.

### Supervised by:

**Mr. Mohshi Masnad**

Lecturer

Department of CSE

Daffodil International University

### Co-Supervised by:

**Nusrat Jahan**

Lecturer

Department of CSE

Daffodil International University

### Submitted by:

**Animul Ehshan**

ID: 142-15-3523

Department of CSE

Daffodil International University

**Md. Naymul Haque Dider**

ID: 142-15-4041

Department of CSE

Daffodil International University

**MD. Asaduzzaman Sagor**

ID: 142-15-3979

Department of CSE

Daffodil International University

## ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project successfully.

We really grateful and wish our profound our indebtedness to **Mr. Mohshi Masnad, Lecturer** and, **Nusrat Jahan, Lecturer**, Department of CSE, Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor and co-supervisor in the field of “*Human Computer Interaction*” to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to **Dr. Syed Akhter Hossain, Professor & Head, Department of CSE**, for his kind help to finish our project and also to other faculty members and the staffs of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

## **Abstract**

Humans show universal consistency in recognizing emotions but also show a great deal of variability between individuals in their abilities. Emotion recognition is used for a variety of reasons. Emotion recognition is the process of identifying human emotion, most typically from facial expressions. This is both something that humans do automatically but computational methodologies have also been developed. After recognition the emotion we will suggest a song which is based on his/her emotion. If he/she is sad, happy, angry then we will suggest him/her a song which is best for them. And it will change his/her mood if he/she is sad, angry or depressed then it will turn into happy mode. It helps to change the emotion of a person. We use php, JavaScript, MySQL database for this project and D3 for bar chart, clmtracker library for emotion recognition and html, css for design. This application is easy to use and it is user friendly. To use this application user must have a camera in his/her device, without camera it will not work. After finishing all of the part of this application it works properly and it gives us approximately 95% correct result.

## TABLE OF CONTENTS

<b>CONTENTS</b>	<b>PAGE</b>
Board of examiners	i
Declaration	ii
Acknowledgements	iii
Abstract	iv
Table of contents	v-vii
List of figures	viii-ix
List of tables	x

### **CHAPTER**

#### **CHAPTER 1: INTRODUCTION 1-3**

1.1 Introduction	1
1.2 Motivation	1
1.3 Objectives	1
1.4 Expected Outcome	2
1.5 Report Layout	2-3

#### **CHAPTER 2: BACKGROUND 4-6**

2.1 Introduction	4
2.2 Related Works	4
2.3 Advantages and Disadvantages	4
2.4 Comparative Studies	4-5
2.5 Future Scope	5-6
2.6 Challenges	6

<b>CHAPTER 3: REQUIREMENT SPECIFICATION</b>	<b>7-18</b>
3.1 Business Process Modeling	7-9
3.2 Requirement Collection and Analysis	9-10
3.3 Use Case Modeling And Description	10-16
3.4 ER Diagram	16
3.5 Design Requirements	17-18
<b>CHAPTER 4: DESIGN SPECIFICATION</b>	<b>19-33</b>
4.1 Front-end-Design	19-27
4.2 Back-end Design	28
4.3 Interaction Design and UX	28-29
4.4 Implementation Requirements	29-33
<b>CHAPTER 5: IMPLEMENTATION AND TESTING</b>	<b>34-45</b>
5.1 Implementation of Database	34-41
5.2 Implementation of Front-end Design	41
5.3 Implementation of Interaction	41
5.4 Testing Implementation	41-44
5.5 Test Results and Reports	44-45
<b>CHAPTER 6: CONCLUSION AND FUTURE SCOPE</b>	<b>46</b>
6.1 Discussion and conclusion	46
6.2 Limitations	46
6.3 Scope for Further Development	46
<b>REFERENCES</b>	<b>47-48</b>

<b>APPENDIX</b>	<b>49</b>
Appendix A: Project Reflection	49
Appendix B: Related Website	49
<b>PLAGIARISM REPORT</b>	<b>50</b>



## LIST OF FIGURES

<b>FIGURES</b>	<b>PAGE NO</b>
Figure 3.1: Data Flow Diagram	7
Figure 3.2: Business Process Model and Notation	8
Figure 3.3: Waterfall Model	9
Figure 3.4: Use Case Model	11
Figure 3.5: Entity Relationship Model	16
Figure 3.6: Flow Chart	18
Figure 4.1: Home Page (i)	19
Figure 4.2: Home Page (ii)	20
Figure 4.3: Home Page (iii)	20
Figure 4.4: Home Page (iv)	21
Figure 4.5: Log In (i)	21
Figure 4.6: Log In (ii)	22
Figure 4.7: Sign Up (i)	23
Figure 4.8: Sign Up (ii)	24
Figure 4.9: Emotion Recognize	25
Figure 4.10: Watch Video	26
Figure 4.11: Comment	26
Figure 4.12: Post Comment	27
Figure 4.13: Add Song	27
Figure 4.14: Database	28
Figure 4.15: Face Coordinate Output	30
Figure 4.16: Simple Data for Naïve Bayes	31
Figure 4.17: Eigen Vector	33
Figure 5.1: Emotion Table (i)	34
Figure 5.2: Emotion Table (ii)	35
Figure 5.3: Emotion List	35
Figure 5.4: Comment Database (i)	36
Figure 5.5: Comment Database (ii)	36
Figure 5.6: Song Table (i)	37
Figure 5.7: Song Table (ii)	37

Figure 5.8: Vote Table (i)	38
Figure 5.9: Vote Table (ii)	38
Figure 5.10: User Table (i)	39
Figure 5.11: User Table (ii)	39
Figure 5.12: Song Type (i)	40
Figure 5.13: Song Type (ii)	40
Figure: Plagiarism Report	50

## LIST OF TABLES

<b>TABLE</b>	<b>PAGE NO</b>
Table 3.1: Use Case Description of Sign Up	12
Table 3.2: Use Case Description of Log In	12
Table 3.3: Use Case Description of Check Emotion	13
Table 3.4: Use Case Description of Add Song	13
Table 3.5: Use Case Description of Listen Song	14
Table 3.6: Use case Description of Vote	14
Table 3.7: Use case description of React	15
Table 3.8: Use Case Description of Comment	15
Table 3.9: Use Case Description of Log Out	16
Table 5.1: Testing Implementation	42-44

# CHAPTER 1

## Introduction

### 1.1 Introduction

Nowadays, most of the people have smart phone, tablet or personal computer. And they can easily know their emotion using our system.

In our system user can detect his emotion with image processing and if he/she is sad, angry, depressed then our system will recommend them a song which will help to change their emotion and then they will feel better than past. In our system we have log in part, after login user can see how to use this system. If they face any problem they can contact with us and they also can know about the terms and condition of this system. This is an online based web application.

Our system is easy to use and it is safe for everyone.

### 1.2 Motivation of Work

Now a days, people are being for dependent to online. They hardly do anything by themselves. Everyone wants everything done. People also need to listen song for entertainment. Most of them are dependent on YouTube. But, YouTube is just suggest song from most viewed by the user. It cannot detect emotion. If someone is sad and he will listen a romantic song, he will not like it. And he will also don't want to search by himself. He/she needs something prepared for him/her that will find a song after detecting his emotion. So, we are working with emotion and song related data to find the perfect song at a particular time for a particular user.

### 1.3 Objectives:

- It can identify human emotion with image processing.
- After identify human emotion it will recommend a song.
- It is easy to use.
- Just one click need to start.
- It is user friendly.
- To run this program user must have a camera.
- To listen the recommended song user must have internet connection.

- User need to register to log in this system.
- User can add song as like as he/she wants.
- User can contact with us.
- User cannot delete or remove any song from the system.
- Only admin can delete or remove any song from the system.

#### **1.4 Expected Outcome**

The outcome of this project is to recommend a song based on user emotion. We want to make sure that user will get the right song recommendation which will be based on user emotion. If user will face any difficulty he/she can easily contact with us through email. We will try to make happy the user by recommending his/her favorite song which he/she really want to listen when he/she is happy or sad or surprised or angry. This will save time and user do not need to search his favorite song in his/her phone or computer. He/she can easily find his/her favorite song based on his emotion by using our program.

#### **1.5 Report Layout**

- **Chapter 1 : Introduction**

In this chapter we discussed about project introduction, motivation, objectives, expected outcome and report layout.

- **Chapter 2 : Background**

In this chapter we discuss concerning the background circumstances of our project. We tend to conjointly name the connected works, comparison to different candidate systems and the scope of the matter and challenges of the project.

- **Chapter 3 : Requirement Specification**

This chapter is all about the requirements like business process modeling, the requirement collection and analysis, the use case model of the project and their description, the logical data model and the design requirements.

- **Chapter 4 : Design Specification**

In this chapter we discussed about all the designs of this project. Like Front-end design, back-end design, interaction design and the implementation requirements.

- **Chapter 5 : Implementation & Testing**

This chapter all about the implementation of database, front-end designs, interactions, test implementation and the test results of the project.

- **Chapter 6 : Conclusion & Future Plan**

We discussed about the conclusion and the future developments of this project which pretty much best about the project.

## **CHAPTER 2**

### **Background**

#### **2.1 Introduction**

In this chapter we discuss what is different in our project and the benefit, limitation. And also what we face during our project work. And also discuss about future scope and challenges.

#### **2.2 Related Work**

This project mainly based on song recommendation based on emotion. It is easy to use. And we make it very easy for user. It needs internet connection with a camera enable device. At first we make a survey. Which is mainly data collection for our project. We ask them few question and the question is all about his/her emotion and in which emotion he/she likes to listen which song and change his/her mood.

#### **2.3 Advantages & Disadvantages**

##### **Advantages:**

- Easy to use.
- Easily detect the emotion and recommend the best song based on emotion.
- Just one click need to start the system.
- Without login user can use the system.
- User can add song by own based on his/her emotion.

##### **Disadvantages:**

- Without internet connection it do not work.
- Without camera it do not work.
- We might get multiple emotion detection sometimes.

#### **2.4 Comparative Studies**

Nowadays there is so many application which is recognition of emotion. Detect the emotion of human face. But there's easy application to detect face and recognition the emotion but there is not so many application which mainly based on song recommendation and the song is depend

on emotion. Here all the task start after recognition of emotion and then suggest a song based on emotion.

Heart Player is one of the emotion based music player. It is for mobile phone but our application is totally online based application. And there is no restriction in our application, people can access it from their computer or phone but there's only need internet connection.

Nexttone Player is also a music recommendation system based on user behavior. The goal of their system is to recommend songs, which are favored by the user, are fresh to the user's ear, and fit the user's listening pattern. They use "Forgetting Curve" to assess freshness of a song and evaluate "favoredness" using user log. And it needs internet connection to use like our system.

YouTube is one of the best platform to listen song but it always shows the most viewed song and then recommend the song to the user. But here in our application people just log in the system and then they run the application and then the application will recommend them best song according to their current mood.

One more application which is compare text and then they will recommend song. This cannot read the emotion of the user.

Our application can detect four types of emotion and we recommend song based on this four types of emotion. We can detect human happy, sad, surprised and angry face. In future we will try to detect few more facial expression.

## **2.5 Future Scope**

We are focusing to develop this application to feel the people. Nowadays, everybody depends on technology. Apparently we are also getting lazier than past. We also focusing on save time. In our system, people can save their time and they can easily find what they actually need. Now we can only detect four emotion but in future we want to detect few more facial expression and that will be more accurate than our current system. Now in our system it gives us more than 90% accurate result.



People can listen song in YouTube but before listening any song he/she needs to search that song and then he needs to play the song. And it took few times. But in our system user just need to visit our system and then our system will detect which mood actually he/she was. Then our system recommend the user best song for him/her based on his current mood.

Right now we classified the song manually but in future we want me make an automatic song classification which will help the user. User can give us review and through email they will can communicate with us. In future we will try to add live chat option and then they will can give us feedback very easily and if they will face any problem then they can inform us and immediately we will try to solve the problem.

And we will also make a mobile phone version if we success in online platform.

## **2.6 Challenges**

When you need to accomplish something then unquestionably you need to confront a few difficulties and obstructions. We also face some challenges while we do our project.

Firstly, we face some problem while we develop our project because we are new in this platform. We tried our best to make this project error free and try to make easy for the user. To get 100% accurate result we need to do test it as much as we can.

Secondly, our project is “**Emotion Based Song Recommendation**”. It is totally based on emotion and if I can’t detect the emotion correctly then it can’t recommend the song properly. For this reason we need to do a survey and then add song based on that survey. So if people do not write the song which he/she listen when he/she is happy or sad or surprised or angry then we cannot add the correct song to him/her. If people write that correctly then our program can easily give song recommendation and people will be happy.

# CHAPTER 3

## Requirement Specification

### 3.1 Business Process Modeling

Business process modeling is mapping out regular business processes and finding ways to improve them. It is a part of the practice of business process management.

Process modeling software gives an analytical representation of 'as-is' processes in an organization and contrasts it with 'to-be' processes for making them more efficient.

### Data Flow Diagram

Here we discuss about our business process model using DFD (Data Flow Diagram). Data Flow Diagram described us how the data is process in our program. Below, in the following figure we draw a Data Flow Diagram for our system. Figure 3.1 shows Data flow diagram of our system.

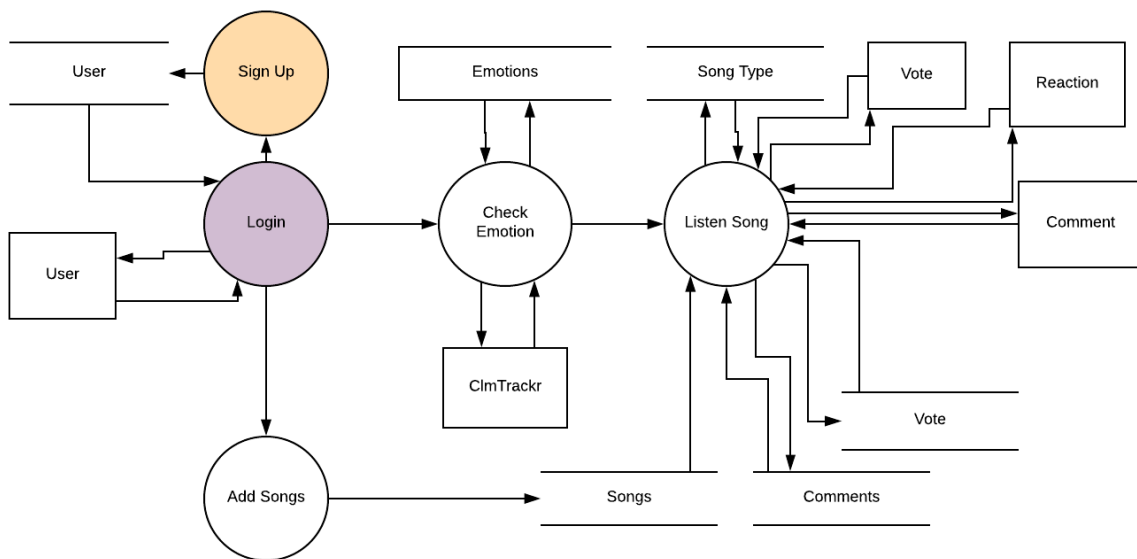


Figure 3.1: Data Flow Diagram

### Business Process Model and Notation

Here we discuss about our Business Process Model and Notation (BPMN).

A standard Business Process Model and Notation (BPMN) will provide businesses with the capability of understanding their internal business procedures in a graphical notation and will

give organizations the ability to communicate these procedures in a standard manner. Furthermore, the graphical notation will facilitate the understanding of the performance collaborations and business transactions between the organizations.

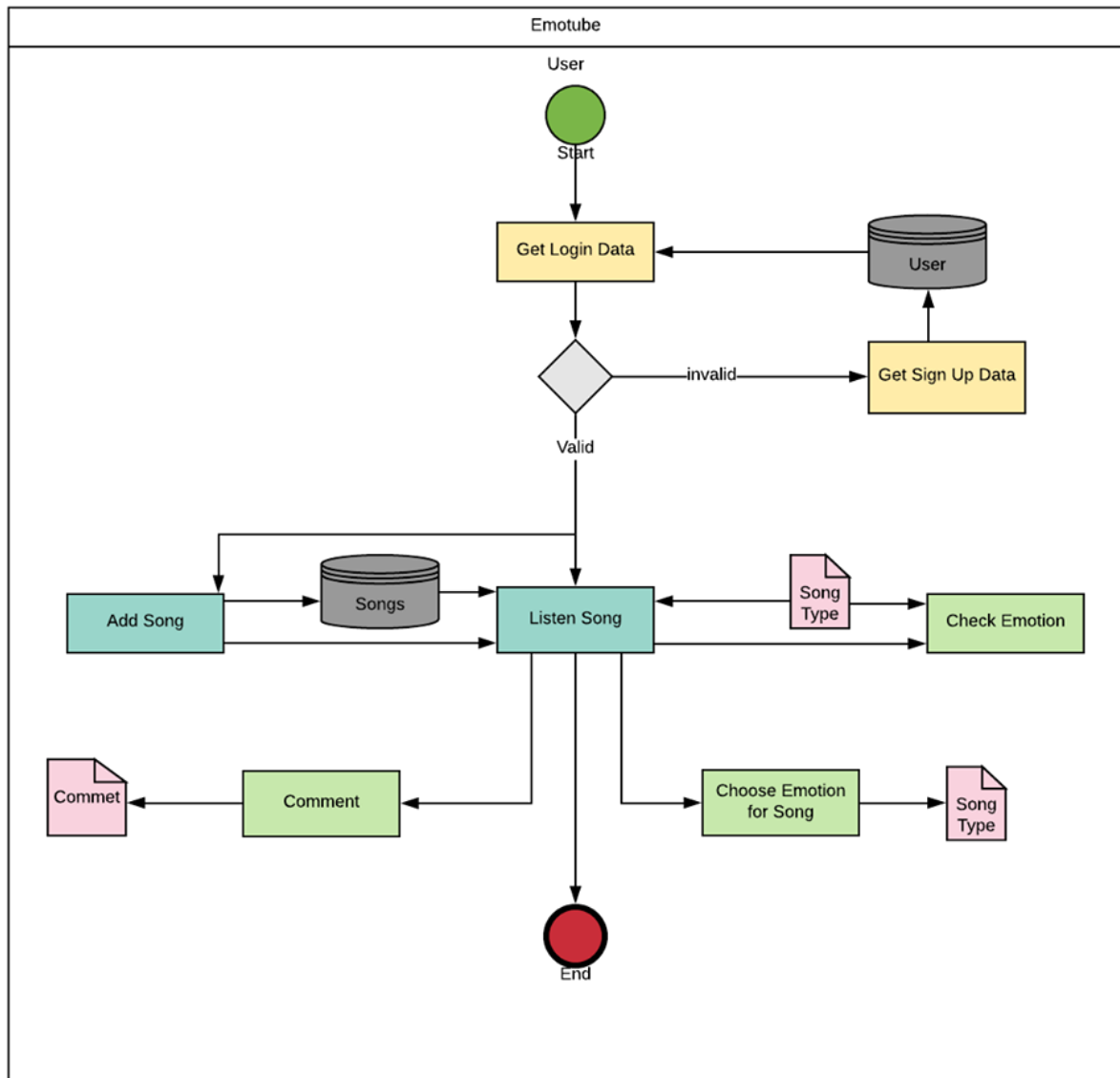


Figure 3.2: Business Process Model & Notation

This will ensure that businesses will understand themselves and participants in their business and will enable organizations to adjust to new internal and B2B business circumstances quickly.

In the following figure we draw a Business Process Model and Notation for our system. Figure 3.2 shows Business Process Model and Notation of our system. Now it will clear how our system will work.

## Waterfall Model

The Waterfall Model was the first Process Model to be introduced. It is also referred to as a **linear-sequential life cycle model**. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

The Waterfall model is the earliest SDLC approach that was used for software development.

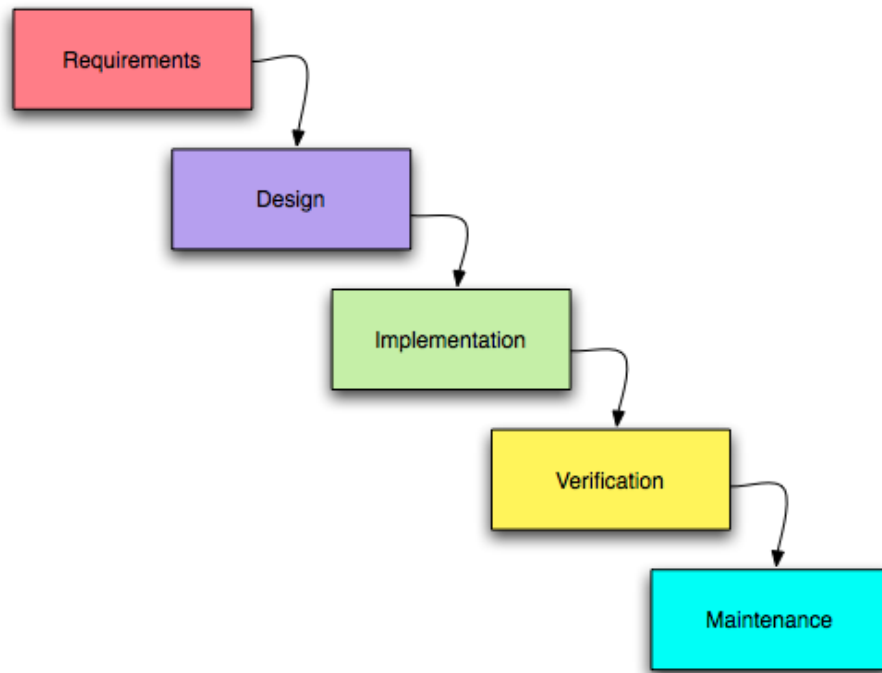


Figure 3.3: Waterfall Model

The waterfall Model illustrates the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete. In this waterfall model, the phases do not overlap.

Our system follow all the steps of waterfall model and it consider all the steps of waterfall model.

### 3.2 Requirement Collection and Analysis

Requirement collection and analysis is one of the very essential conditions for any application advancement process. In application arrangement, there are two types of requirements needed, one is the functional requirement and the other is nonfunctional requirement. Functional

requirements are the activities that's the application software can easily perform. Nonfunctional requirements define the personality of the application, which results the application that is how much efficient, performance issues of the application and many more.

### **3.2.1 Functional Requirement**

According to our system, there should have so many functional requirements. Such as

- Login
- Signup
- Add video
- Categorized video
- Watch video
- Detect emotion

And so on.

### **3.2.2 Non-functional Requirement**

Non-functional requirements in our application are help to optimize performance, memory consuming, being more efficient; smoother operation, and load on quickly as possible to our application. Application UI should be user friendly and gorgeous for excellent user experience. We have used Object Oriented Mysqli for database query as a faster query. We have all the java script file like Ajax or others uploaded in the website, so it don't need other website to access or any third party ads. And most importantly the website design is tested several time by some listeners.

### **3.3 Use Case Modeling and Description**

Figure 3.4 shows the Use Case Modelling.

A use case has these characteristics:

- Use to organize functional requirements.
- Modelling the goals of the system/actor interactions.
- Record path from trigger event to goals.

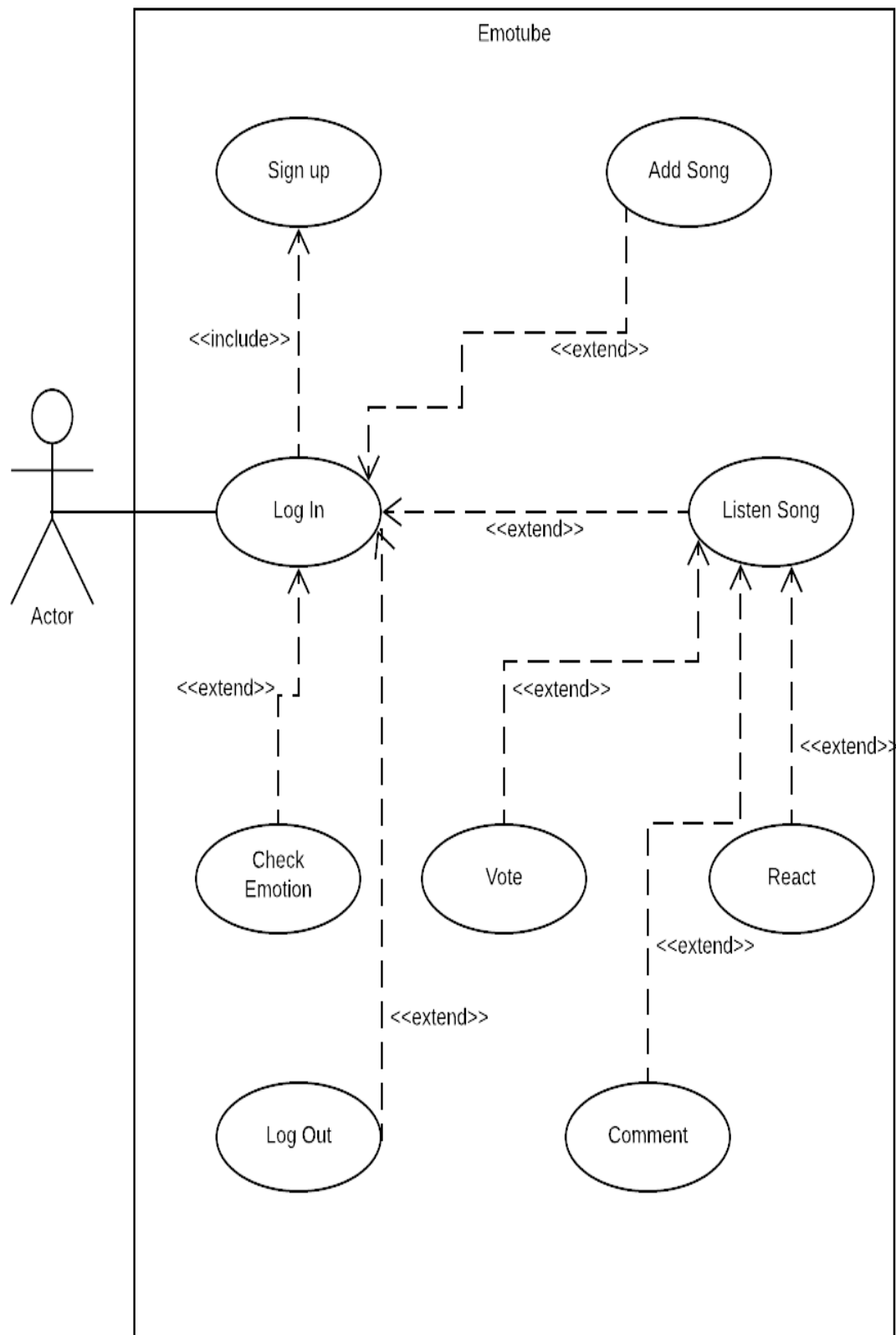


Figure: 3.4 Use Case Model

Table 3.1: Use case description of Sign Up

Use Case #01	Sign Up
Primary Actor	User
Secondary Actor	Null
Pre-condition	You must have and email and phone number
Scenario	<ul style="list-style-type: none"> <li>▪ Enter valid full name</li> <li>▪ Enter valid email address</li> <li>▪ Enter password minimum of 6 character</li> <li>▪ Enter Phone number</li> </ul>
Post-condition	<ul style="list-style-type: none"> <li>▪ Successfully Registration or Not</li> </ul>

Table 3.2: Use case description of Login

Use Case #02	Login
Primary Actor	User
Secondary Actor	Null
Pre-condition	Sign Up
Scenario	<ul style="list-style-type: none"> <li>▪ Enter valid email address</li> <li>▪ Enter password</li> </ul>
Post-condition	<ul style="list-style-type: none"> <li>▪ Successfully Login or Failed</li> </ul>

Table 3.3: Use case description of Check Emotion

Use Case #03	Check Emotion
Primary Actor	User
Secondary Actor	Null
Pre-condition	Login
Scenario	<ul style="list-style-type: none"> <li>▪ Detect current emotion</li> <li>▪ Able to recognize emotion</li> <li>▪ Shows the percentage of four emotion</li> </ul>
Post-condition	<ul style="list-style-type: none"> <li>▪ Recommend Songs based on emotion percentage</li> </ul>

Table 3.4: Use case description of Add Song

Use Case #04	Add Song
Primary Actor	User
Secondary Actor	Null
Pre-condition	Login
Scenario	<ul style="list-style-type: none"> <li>▪ Song Name</li> <li>▪ Select Name</li> <li>▪ Upload Thumbnail</li> <li>▪ YouTube link</li> <li>▪ Song details</li> <li>▪ Upload</li> </ul>
Post-condition	<ul style="list-style-type: none"> <li>▪ Successfully Song Added or Not</li> </ul>



Table 3.5: Use case description of Listen Song

Use Case #05	Listen Song
Primary Actor	User
Secondary Actor	Null
Pre-condition	Login
Scenario	<ul style="list-style-type: none"> <li>▪ Listen Song</li> <li>▪ Listen emotion based songs</li> <li>▪ Listen most viewed songs</li> <li>▪ Listen most liked songs</li> <li>▪ Listen latest songs</li> </ul>
Post-condition	<ul style="list-style-type: none"> <li>▪ Song will be played</li> <li>▪ User can vote for the song as Like or Dislike.</li> <li>▪ User also can React and Comment</li> </ul>

Table 3.6: Use case description of Vote

Use Case #06	Vote
Primary Actor	User
Secondary Actor	Null
Pre-condition	Listen Song
Scenario	<ul style="list-style-type: none"> <li>▪ Like</li> <li>▪ Dislike</li> </ul>
Post-condition	<ul style="list-style-type: none"> <li>▪ Successfully Voted or Not</li> </ul>

Table 3.7: Use case description of React

Use Case #07	React
Primary Actor	User
Secondary Actor	Null
Pre-condition	Listen Song
Scenario	<ul style="list-style-type: none"> <li>▪ React sad</li> <li>▪ React angry</li> <li>▪ React happy</li> <li>▪ React surprised</li> </ul>
Post-condition	<ul style="list-style-type: none"> <li>▪ Successfully Reacted or Not</li> </ul>

Table 3.8: Use case description of Comment

Use Case #08	Comment
Primary Actor	User
Secondary Actor	Null
Pre-condition	Listen Song
Scenario	<ul style="list-style-type: none"> <li>▪ Write Comment</li> <li>▪ Post Comment</li> </ul>
Post-condition	<ul style="list-style-type: none"> <li>▪ Successfully Comment Added or Not</li> </ul>

Table 3.9: Use case description of Log Out

Use Case #09	Log Out
Primary Actor	User
Secondary Actor	Null
Pre-condition	Log in
Scenario	<ul style="list-style-type: none"> <li>▪ Use case begins when user clicks on “Logout”</li> <li>▪ The system will destroy the user session and the user is logged out</li> </ul>
Post-condition	<ul style="list-style-type: none"> <li>▪ Successfully Log Out or Not</li> </ul>

### 3.4 ER Diagram

In software engineering, an ER (Entity Relationship) model is commonly formed to represent things that a business needs to remember in order to perform business processes.

Consequently, the ER model becomes an abstract data model, that defines a data or information structure which can be implemented in a database is typically a relational database.

Figure 3.5 shows the ER diagram.

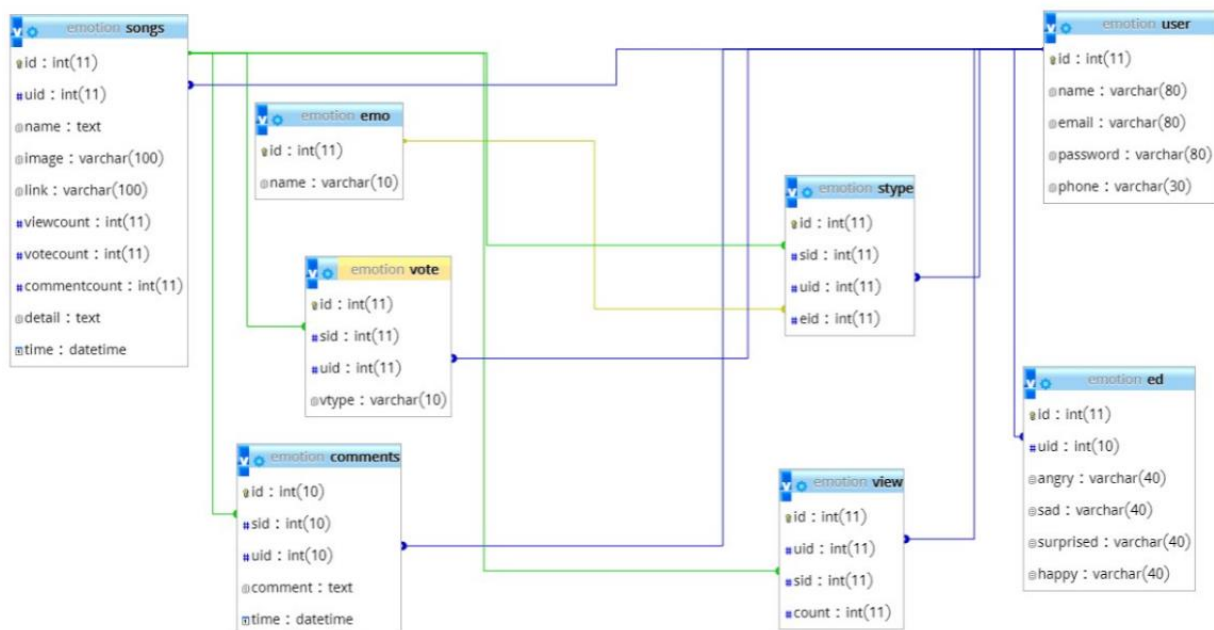


Figure: 3.5 Entity Relationship Model

### **3.5 Design Requirements**

The design requirement is the one of the most important part of an application which makes an application unique from any other existing application. In our application we mainly focused on better user experience and user friendly. We working on our own specific problem statement, system or user experience which we are designing. We also provide some new features by which the user can easily operate our system.

- In our application we design a sign up section so that any user can be sign up in our application with their name, email and phone number.
- We design emotion detection so that user can detect his/her emotion.
- We design song recommendation based on user emotion.
- We design an option for user that he/she can add song after log in in our program.
- In our application user also can contact with us.

### 3.5.1 Flow Chart

A flowchart is a diagram that represents an algorithm, workflow or process, showing the step by step as boxes of various kinds, and their order by connecting them with arrows. This diagrammatic representation shows a solution model to a given problem.

Figure 3.6 shows a Flow chart

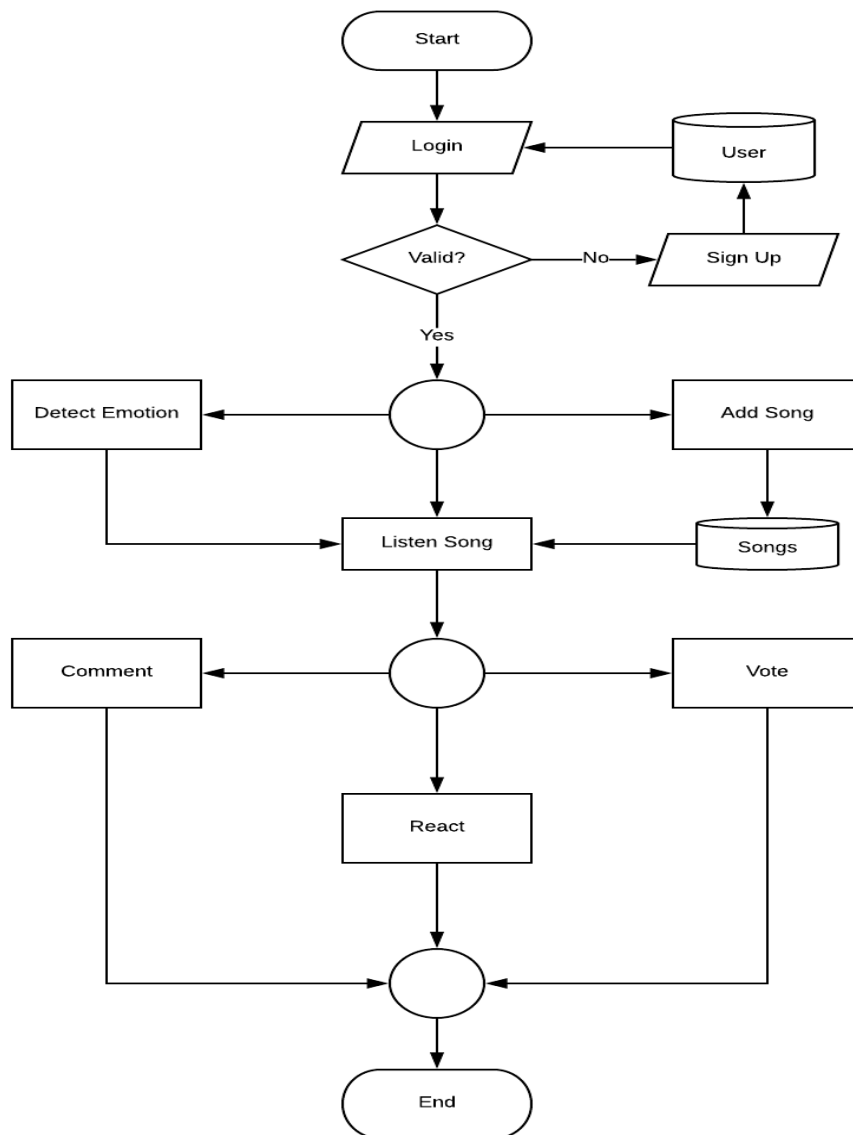


Figure: 3.6 Flow Chart

# CHAPTER 4

## Design Specification

### 4.1 Front-end Design

In any application the front end design is the visual part of a program. By which the user interacts with. In the perspective of designing, Front-end design is one of the most essential segments for the program. It represents the introduction layer and user can directly communicate with this.

This is the home page of our project and this page is visible to the user on anybody.

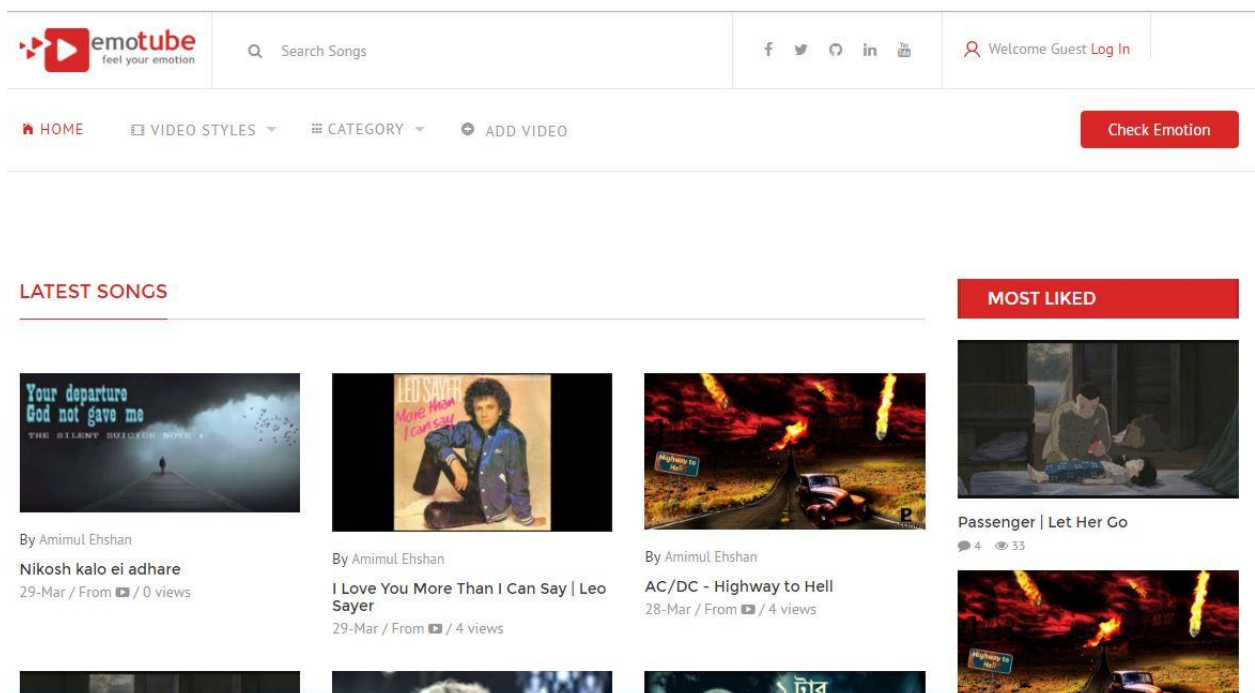


Figure 4.1: Home Page (i)

In Figure 4.1 it is home page and in home page we can see a log in button and an emotion detection button. User can detect his/her emotion after clicking that button.

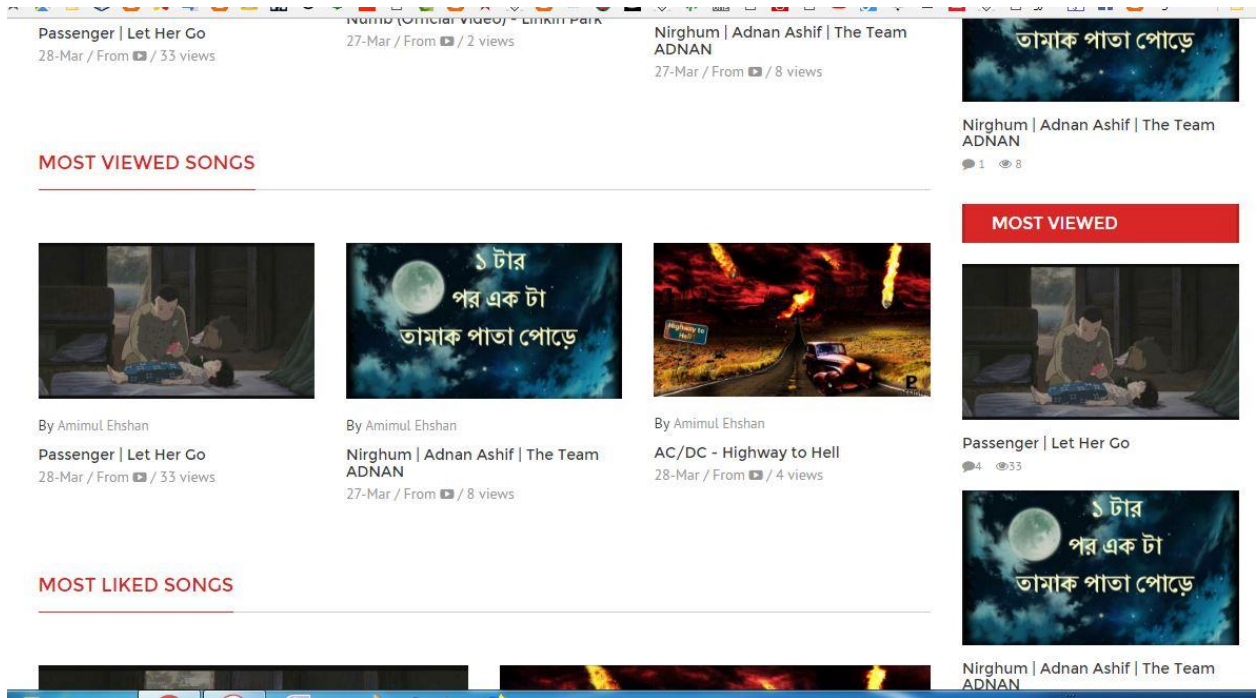


Figure 4.2: Home Page (ii)

In Figure 4.2 here we can see the most viewed, liked song.

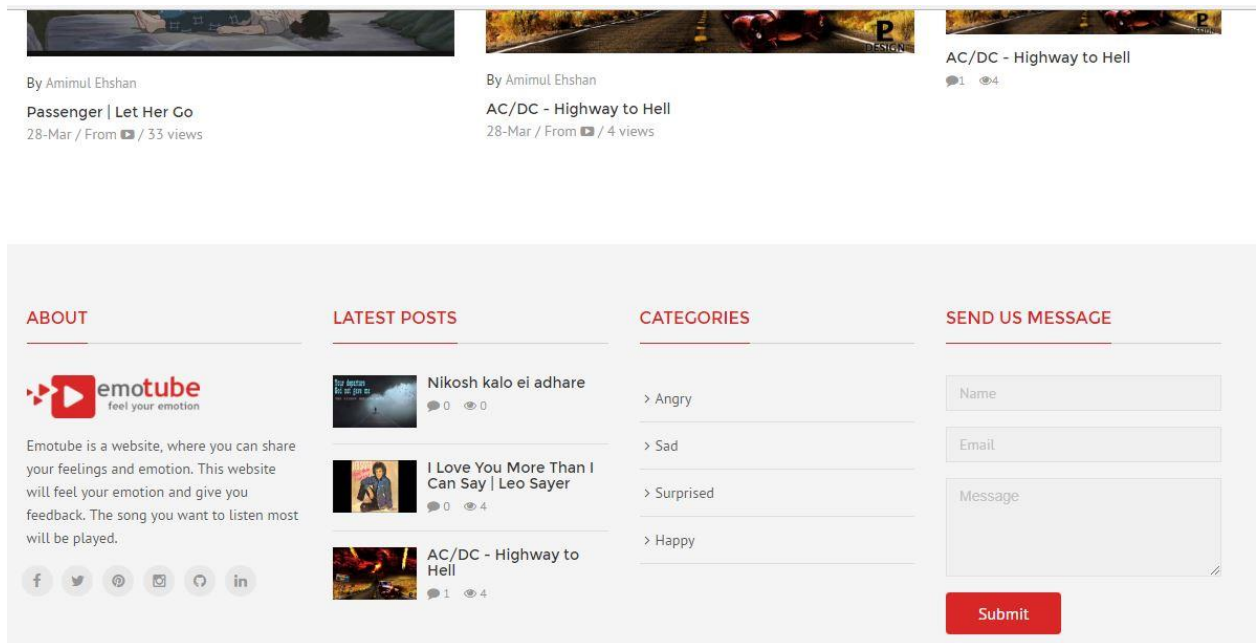


Figure 4.3: Home Page (iii)

In the last part of home page we can see there is categories and also a contact option with us. There is four types of categories. They are Sad, Happy, Surprised and Angry.

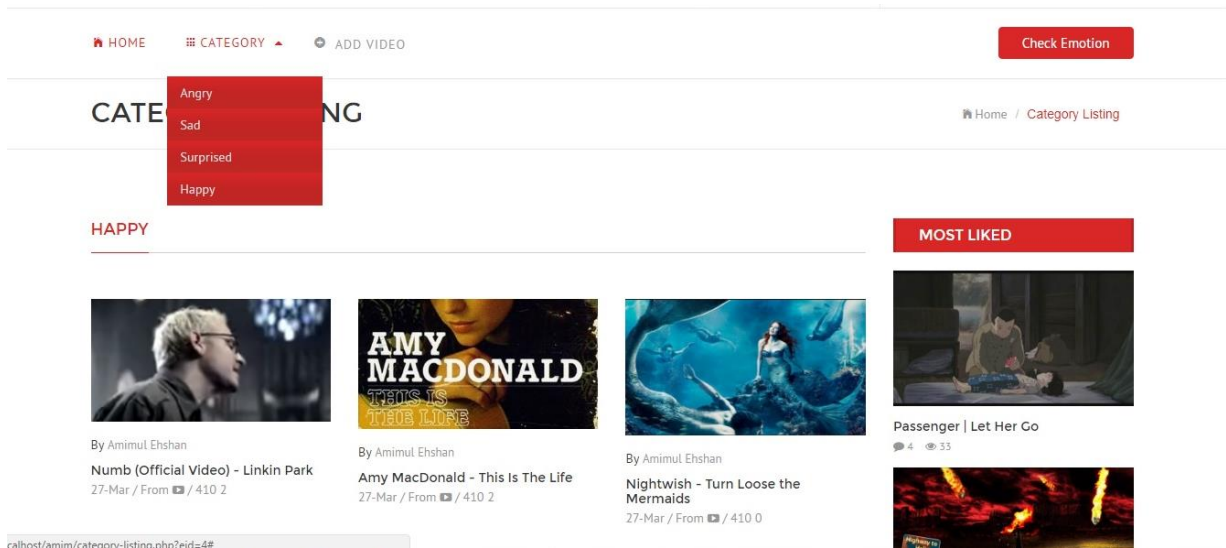


Figure 4.4: Home Page (iv)

To contact with us, user just need to type there name, email and their text message and then they just need to click the send button.

Our project have log in and sign up option. User can log in in our system and then he/she can add song and also he/she can check his/her emotion. In figure 4.5 we show our log in page.

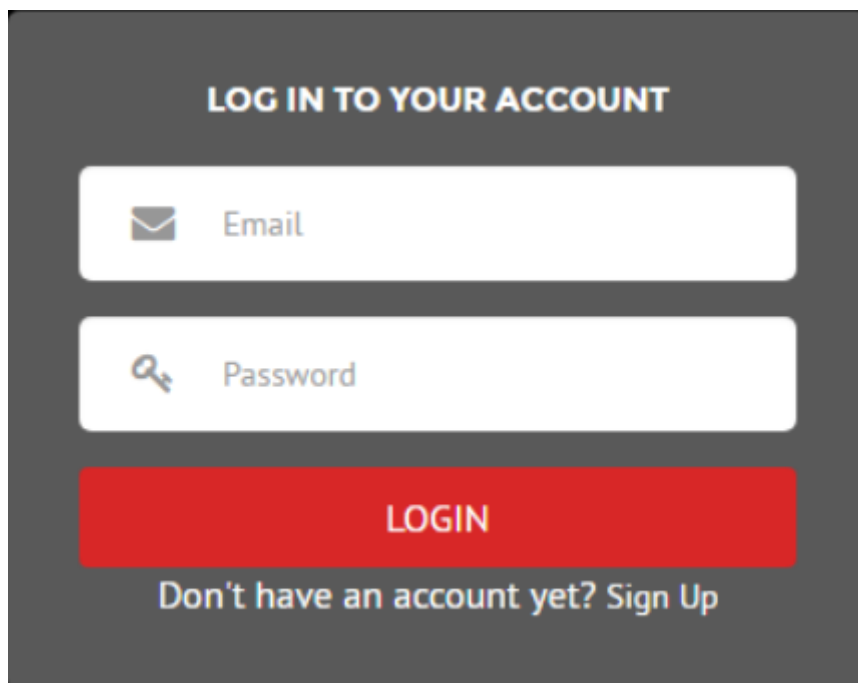
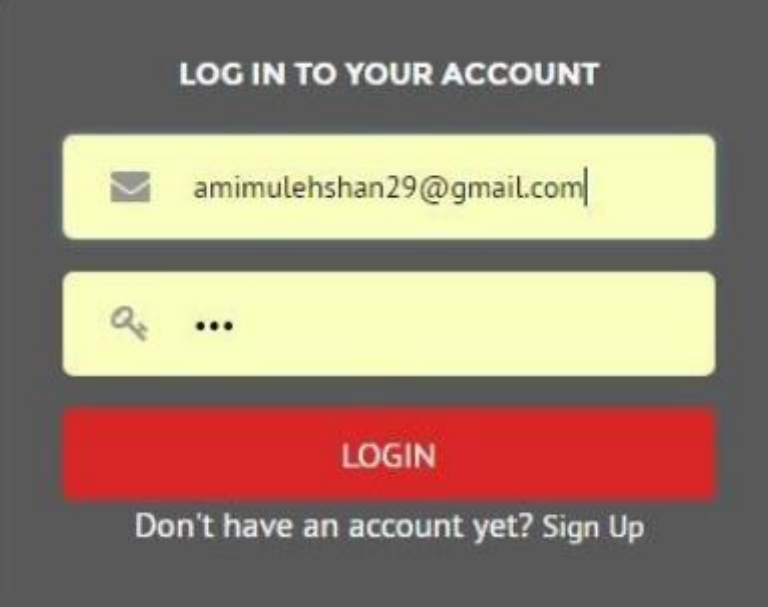


Figure 4.5: Log in (i)



After clicking the login button user just need to give his email and password then he/she just need to click button. In figure 4.6 we can see that the user input his/her email and password for login.



The image shows a login form on a dark grey background. At the top, the text "LOG IN TO YOUR ACCOUNT" is displayed in white. Below this, there are two light yellow input fields. The first field contains an email address "amimulehshan29@gmail.com" and has a small envelope icon on the left. The second field is empty and has a key icon on the left. Below the input fields is a red button with the word "LOGIN" in white. At the bottom of the form, the text "Don't have an account yet? Sign Up" is displayed in white.

Figure 4.6: Log in (ii)

If user don't have any account then he/she can easily create an account in our system. For sign up user just need give his/her name, email, phone and password.

In figure 4.7 we can see the sign up page design.

The image shows a sign-up form on a dark grey background. At the top, the text "SIGN UP" is centered in white. Below this are four white input fields, each with a small icon on the left and a label on the right: "Name" with a capital letter 'A' icon, "Email" with an envelope icon, "Password" with a key icon, and "Phone Number" with a mobile phone icon. Below the input fields is a prominent red button with the text "SIGN UP" in white. At the bottom of the form, the text "You have an account allready? Login" is displayed in white, where "allready" is a misspelling of "already".

Figure 4.7: Sign Up (i)

After signing up account user can now log in by clicking log in button and they just need to give their email and password. If the password is not valid then they cannot log in to the system. Figure 4.8 shows us the details about sign up.

The image shows a mobile application sign-up screen. At the top, the text "SIGN UP" is centered in white on a dark grey background. Below this, there are four input fields stacked vertically. The first field is white and contains the name "Md Naymul Haque Dider" next to a grey 'A' icon. The second field is yellow and contains the email address "naymul4041@diu.edu.bd" next to a grey envelope icon. The third field is yellow and contains six dots "....." next to a grey key icon, representing a password. The fourth field is yellow and contains the phone number "01752804620" next to a grey phone icon. Below these fields is a red button with the text "SIGN UP" in white. At the bottom, there is a link that says "You have an account allready? Login" in white text.

Figure 4.8: Sign Up (ii)

Now we are going to emotion detection design and figure 4.9 shows us the design of this. And it needs to enable camera to work properly. Without camera it will not work and also if there is no light it cannot recognize any face and it cannot give the emotion.

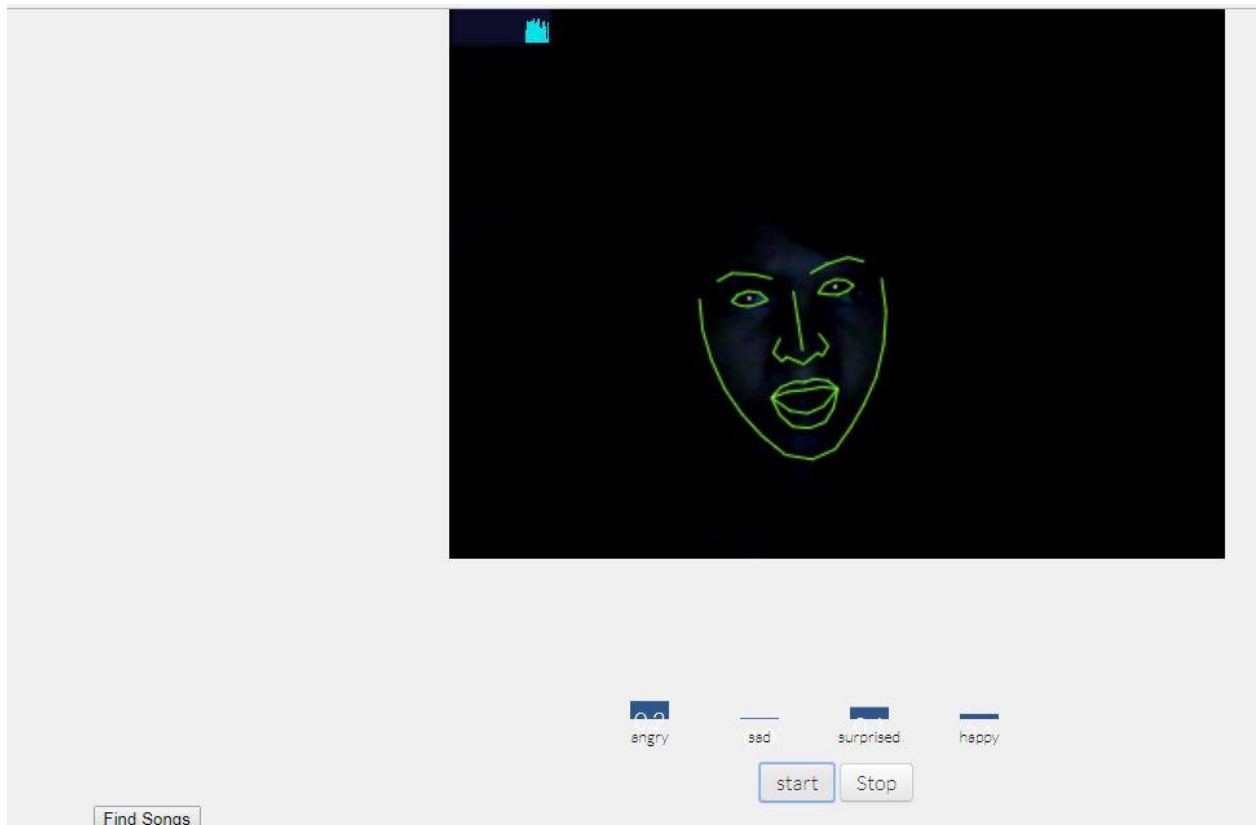


Figure 4.9: Emotion Recognize

Then we can listen song or watch the videos. Here are some screenshot and that will represent by figure 4.10, 4.11 & 4.12, how it looks. In this page, user can listen songs, react angry, sad, happy or surprised. Or vote like or dislike.

## WATCH VIDEO

Home / Watch Video



### MOST LIKED



Passenger | Let Her Go

4 34



Figure 4.10: Watch Video



By Guest

Passenger | Let Her Go

Mar 28, 2018

35

2

Angry

Sad

Surprised

Happy

Like

Dislike

'Let Her Go' from the album 'All the Little Lights' by Passenger

### COMMENTS

test

Commented By Amimul Ehshan

another test

Commented By Amimul Ehshan

test 3

Commented By Amimul Ehshan

test by another id

Commented By Naymul Haque Dider

Nirghum | Adnan Ashif | The Team  
ADNAN

1 8



I Love You More Than I Can Say | Leo  
Sayer

0 4

### MOST VIEWED



Passenger | Let Her Go


4 34




Figure 4.11: Comment

Comment

**POST COMMENT**




Numb (Official Video) - Linkin Park  
0 12



AC/DC - Highway to Hell  
2 9

**ABOUT**      **LATEST POSTS**      **CATEGORIES**      **SEND US MESSAGE**



**Titanic - Nearer My God To Thee**      > Angry

0 2

Name

Figure 4.12: Post Comment

User can add song after log in. Figure 4.13 shows us the design of that page.

## ADD VIDEO

**A** Song Name

**😊** Choose Emotion

**📺** Youtube Address for the Song

**🖼️** Select Thumbnail to upload  
 No file chosen

**📝** Enter the song details

Figure 4.13: Add Song

## 4.2 Back-end Design

The Back-end design is the part which is working behind of the project. The user can't see the back-end part. There is only one-way user can interact with the program by front-in design. And the user cannot see the back-end design and they don't know how this part is working. In every program back-end does almost everything that happens on the server of the program. We tried to keep our back-end as simple as possible.

We use a database in our system for storing data and figure 4.14 represent the components of data base. We use 8 table.



Table	Action	Rows	Type	Collation	Size	Overhead
comments	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16 K1B	-
ed	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 K1B	-
emo	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16 K1B	-
songs	Browse Structure Search Insert Empty Drop	10	InnoDB	latin1_swedish_ci	16 K1B	-
stype	Browse Structure Search Insert Empty Drop	11	InnoDB	latin1_swedish_ci	16 K1B	-
user	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 K1B	-
view	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16 K1B	-
vote	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 K1B	-
8 tables	Sum	40	InnoDB	latin1_swedish_ci	128 K1B	0 B

Figure 4.14: Database

## 4.3 Interaction Design and UX

Interaction design represents interaction between users and the application. Interaction design considered the process such as the user issue space, processing the issues and discovering the outcomes. It can also perform activities by the regarding outcomes about and take care of the issue in the application.

Actually in any website user experience mainly focuses on the overall experiences between the users and the website. In our website, most of all the features can interact with the user. We are mainly focusing the data mining of user data and there is an option where the user needs to register and logging in the website by using his/her verified information such like email and password.

In our website for UX we have tried to give our user some great experiences by adding some extra features like reaction, comment, like or dislike. For better performance and experience for the user, we tried to keep our website simple and easier to use.

#### **4.4 Implementation Requirements**

To implement a project, we need to use different types of tool, components those help us to developed a project successfully. So in our website development we also used some tools and components. In the Implementation Requirement section, we mainly discussed about all the tools and components that we have used to develop our project.

##### **4.4.1 PHP**

PHP is a server scripting language, and a powerful tool for making dynamic and interactive Web pages. PHP is a widely-used, free, and efficient alternative to competitors such as Microsoft's ASP. We have used PHP as our backend programming language.

##### **4.4.2 MYSQL**

MySQL is an open-source relational database management system. Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language. We have used MySQL query for our SQL.

##### **4.4.3 Java Script**

JavaScript, often abbreviated as JS, is a high-level, interpreted programming language. It is a language which is also characterized as dynamic, weakly typed, prototype-based and multi-paradigm. We have mainly used JavaScript for detecting the emotion and other design.

##### **4.4.4 jQuery Ajax**

Ajax is a set of Web development techniques using many Web technologies on the client side to create asynchronous Web applications. We have used ajax to save data or check data in database without a page refresh.

##### **4.4.5 D3.js**

D3.js is a JavaScript library for producing dynamic, interactive data visualizations in web browsers. It makes use of the widely implemented SVG, HTML5, and CSS standards. It is the



successor to the earlier Protovis framework. In this website we have used D3 to show the emotion percentage wise bar chart.

#### 4.4.6 Clmtrackr

Clmtrackr is a JavaScript library for fitting facial models to faces in videos or images. It currently is an implementation of constrained local models fitted by regularized landmark mean-shift, as described in Jason M. Saragih's paper. Clmtrackr tracks a face and outputs the coordinate positions of the face model as an array, following the numbering of the model below in figure 4.15.

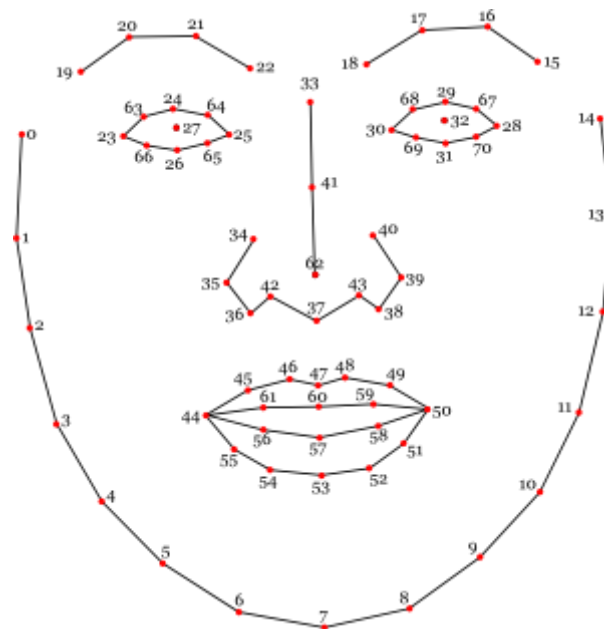


Figure 4.15: Face Coordinates Output

#### 4.4.7 The MUCT Face Database

The MUCT database consists of 3755 faces with 76 manual landmarks. The database was created to provide more diversity of lighting, age, and ethnicity than currently available landmarked 2D face databases.

#### 4.4.8 Naive Bayes Classifier

The Naive Bayes algorithm is based on conditional probabilities. It uses Bayes' Theorem, a formula that calculates a probability by counting the frequency of values and combinations of values in the historical data.

Bayes' Theorem finds the probability of an event occurring given the probability of another event that has already occurred.

Bayes' theorem: Probability of event A given evidence B

$$P(A|B) = \frac{P(A|B)}{P(A)}$$

Where:

- A represents the dependent event: A target attribute and
- B represents the prior event: A predictors attribute
- P(A) is a priori probability of A (The prior probability) Probability of event before evidence is seen. The evidence is an attribute value of an unknown instance.
- P(A|B) is a posteriori probability of B. Probability of event after evidence is seen. Posteriori = afterwards, after the evidence

#### 4.4.8.1 Train Data

There are some data given as training data for naive bayes classification.

```

73     "path": {
74         "normal": [
75             [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14], //fac
76             [15, 16, 17, 18], //right eyebrow
77             [19, 20, 21, 22], //left eyebrow
78             [23, 63, 24, 64, 25, 65, 26, 66, 23], //left eye
79             [28, 67, 29, 68, 30, 69, 31, 70, 28], //right eye
80             [34, 35, 36, 42, 37, 43, 38, 39, 40], //nasal
81             [33, 41, 62], //nose
82             [44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 44, 56,
83             27, //left eyeball
84             32 //right eyeball
85         ],
86         "vertices": [
87             // jawline
88             [0, 1, 23, 0],
89             [1, 23, 66, 1],
90             [1, 2, 66, 1],
91             [2, 66, 26, 2],
92             [2, 26, 35, 2],
93             [2, 35, 36, 2],
94             [2, 36, 3, 2],
95             [36, 44, 45, 36],
96             [3, 4, 44, 3],
97             [3, 4, 45, 3]

```

Figure 4.16: Sample Data for Naïve Bayes

#### 4.4.8.2 Prediction Data

Naive bayes will analyze the training data and predict the result with percentage and the final output will be analyzed by D3 js and our website will show the bar chart and emotion percentage.

#### 4.4.9 Sobel-lbp

Sobel-is a process to represent image in binary form. In this project, about 3766 face is used as training data. All of the faces data is collected from MUCT face Database. And then, the faces are used as sobel-lbp.

#### 4.4.10 Eigen Vector

Eigen Vector is the main function to detect the emotion of face components like eyebrow or eyeball or others.

Consider  $n$ -dimensional vectors that are formed as a list of  $n$  scalars, such as the three-dimensional vectors

$$A = \begin{bmatrix} 1 \\ 3 \\ 5 \end{bmatrix} \text{ and } B = \begin{bmatrix} -12 \\ -15 \\ -6 \end{bmatrix}$$

These vectors are said to be scalar multiples of each other, or parallel or collinear, if there is a scalar  $\lambda$  such that

$$x = \lambda y$$

In this case  $\lambda = -1/20$ .

Now consider the linear transformation of  $n$ -dimensional vectors defined by an  $n$  by  $n$  matrix  $A$ ,

$$Av = w$$

Or

$$\begin{bmatrix} A_{11} & \cdots & A_{1n} \\ \vdots & \ddots & \vdots \\ A_{n1} & \cdots & A_{nn} \end{bmatrix} \begin{bmatrix} v_1 \\ \vdots \\ v_n \end{bmatrix} = \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix}$$

Where, for each row,

$$w_i = A_{i1}v_1 + A_{i2}v_2 + \dots + A_{in}v_n = \sum_{j=1}^n A_{ij}v_j$$

If it occurs that  $v$  and  $w$  are scalar multiples, that is if

$$Av = w = \lambda v \dots \dots (1)$$

Then  $v$  is an eigenvector of the linear transformation  $A$  and the scale factor  $\lambda$  is the eigenvalue corresponding to that eigenvector. Equation (1) is the Eigen value equation for the matrix  $A$ .

Equation (1) can be stated equivalently as

$$(A - \lambda I)v = 0$$

Where  $I$  is the  $n$  by  $n$  identity matrix.

The manually calculated Eigen values are used in this emotion detecting project. Here is the matrix array in figure 4.17, where we have used this.

```

441 "shapeModel": {
442   "eigenVectors": [
443     [-0.25918455544134111, 0.049312213852896816, -0.13803814349503329
444     [-0.015370973540248745, 0.27558869894124727, 0.043057403787850584
445     [-0.27438982091500386, 0.03985289630911687, -0.13450394922360431,
446     [-6.0611116520893973e-05, 0.2907441742190161, 0.06116467954949145
447     [-0.28061079598099281, 0.028500046273409352, -0.13348176497662428
448     [0.014069445433529176, 0.25865071929613015, 0.065608406647573517,
449     [-0.26489741712968851, 0.016037983258916477, -0.15478385413970339
450     [0.046354114624841534, 0.20340754175093087, 0.067448564665621949,
451     [-0.20187217065452481, -0.003452715117513167, -0.1452661188972032
452     [0.05991940055025699, 0.17456494999571184, 0.045546642557409402,
453     [-0.11406660762823516, -0.0077005020180582872, -0.097128049401724
454     [0.050548778821860645, 0.13771378257629771, -0.008012026254443966
455     [-0.010279976005766552, 0.0013953711369666829, -0.042500498536138
456     [0.020506937371916192, 0.080791931896572633, -0.07617795791826975
457     [0.040972435984073616, 0.0012967065227411402, -0.0020002072259834
458     [-0.00080695403900046543, 0.065839220606613527, -0.10155889748442
459     [-0.011079460967186745, 0.0017868787382064075, 0.0394680464852986

```

Figure 4.17: Eigen Vector

# CHAPTER 5

## Implementation and Testing

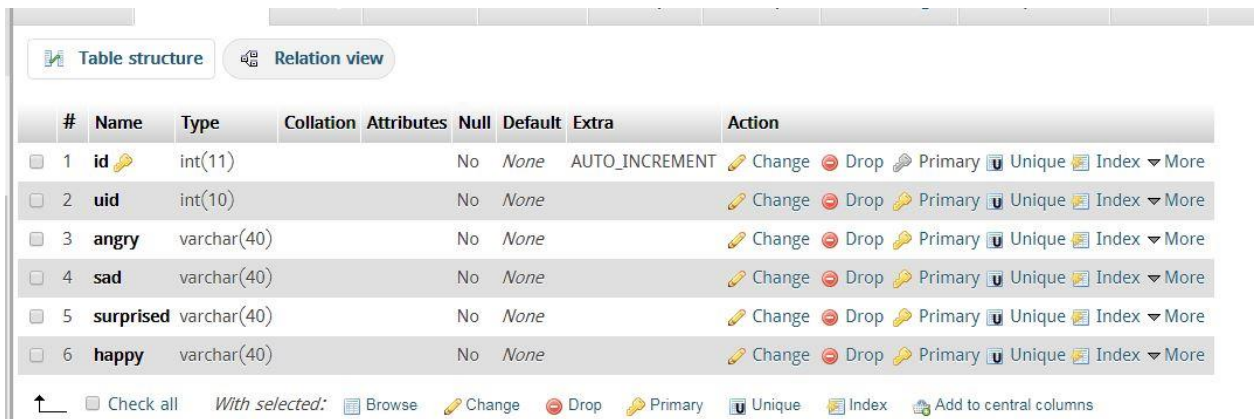
### 5.1 Implementation of Database

This is the chapter which we are focused on how we implement our database. As we mention earlier for our project we use Firebase database. Firebase is a real time database. It has also some other features like Crash reporting, Cloud storage Authentication, Hosting and so on. Since Firebase is a real time database, so there is no need of direct SQL query to perform the actions.

#### 5.1.1 Database Design

Our main database is shown in figure 4.14 in chapter 4. Here we will discuss about the tables created in the database.

We have emotion data from face tracking and we are storing them in emotion table for future use like, in which mood user randomly is. Figure 5.1 and 5.2 represents the emotion table.



#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1	id			No	None	AUTO_INCREMENT	Change  Drop  Primary  Unique  Index  More
<input type="checkbox"/>	2	uid			No	None		Change  Drop  Primary  Unique  Index  More
<input type="checkbox"/>	3	angry			No	None		Change  Drop  Primary  Unique  Index  More
<input type="checkbox"/>	4	sad			No	None		Change  Drop  Primary  Unique  Index  More
<input type="checkbox"/>	5	surprised			No	None		Change  Drop  Primary  Unique  Index  More
<input type="checkbox"/>	6	happy			No	None		Change  Drop  Primary  Unique  Index  More

↑  Check all With selected: Browse Change Drop Primary Unique Index Add to central columns

Figure 5.1: Emotion Table (i)

+ Options		id		uid		angry		sad		surprised		happy	
<input type="checkbox"/>				45	0	1.5743080883587943	36.40009378541974	4.477568705369977	6.287003120711897				
<input type="checkbox"/>				46	1	5.076861829630722	5.987214598775838	18.380165086426402	22.019894922049623				

Figure 5.2: Emotion Table (ii)

Figure 5.3 represents the emotion list. In future, we may increase this emotion list.

+ Options		id		name	
<input type="checkbox"/>				1	angry
<input type="checkbox"/>				2	sad
<input type="checkbox"/>				3	surprised
<input type="checkbox"/>				4	happy

Figure 5.3: Emotion List

We have added some comment for user. But user doesn't know the back-end of comments section. All comment is connected with users and songs by the secondary key uid and sid as user id and songs id. Figure 5.4 represents the comment table.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(10)			No	None	AUTO_INCREMENT	Change Drop More
2	sid	int(10)			No	None		Change Drop More
3	uid	int(10)			No	None		Change Drop More
4	comment	text			No	None		Change Drop More
5	time	datetime		on update CURRENT_TIMESTAMP	No	CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP		Change Drop More

Figure 5.4: Comment Database (i)

Here are some test comments in figure 5.5.

				id	sid	uid	comment	time
<input type="checkbox"/>	Edit	Copy	Delete	1	2	1	test	2018-03-28 00:54:20
<input type="checkbox"/>	Edit	Copy	Delete	2	2	1	another test	2018-03-28 00:56:27
<input type="checkbox"/>	Edit	Copy	Delete	3	2	1	test 3	2018-03-28 00:57:51
<input type="checkbox"/>	Edit	Copy	Delete	4	2	2	test by another id	2018-03-28 01:10:22

Figure 5.5: Comment Database (ii)

We also do work for add song. In database song add part will represent by figure 5.6  
 The songs are connected with user by uid. View count is increased when a user plays a song.  
 Vote count is increased or decreased in every like or dislike. Comment count increases for each comment.

Table structure   Relation view

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	<b>id</b>	int(11)			No	None	AUTO_INCREMENT	Change
2	<b>uid</b>	int(11)			No	None		Change
3	<b>name</b>	text			No	None		Change
4	<b>image</b>	varchar(100)			No	None		Change
5	<b>link</b>	varchar(100)			No	None		Change
6	<b>viewcount</b>	int(11)			No	None		Change
7	<b>votecount</b>	int(11)			No	None		Change
8	<b>commentcount</b>	int(11)			No	None		Change
9	<b>detail</b>	text			No	None		Change
10	<b>time</b>	datetime			No	CURRENT_TIMESTAMP		Change

Check all   With selected:   Browse   Change   Drop   Primary   Unique   Index   Add to

Figure 5.6: Song Table (i)

Here some song added sample in figure 5.7.

Browse   Structure   SQL   Search   Insert   Export   Import   Privileges   Operations   More

	id	uid	name	image	link	viewcount	votecount	commentcount	detail	time
Copy Delete	2	0	Passenger   Let Her Go	20161115005364.jpg	RBumgq5yVrA	32	2	4	'Let Her Go' from the album 'All the Little Lights...	2018-03-28 01:
Copy Delete	3	1	Numb (Official Video) - Linkin Park	linkinpark numb.JPG	kXYiU_JCYtU	1	0	0	Linkin Park "Numb" off of the album METEORA. Direc...	2018-03-27 22:
Copy Delete	4	1	AC/DC - Highway to Hell	highway to hell.JPG	I482T0yNkeo	1	0	0	Highway to Hell is AC/DC's fifth internationally r...	2018-03-28 01:
Copy Delete	5	1	The Dark Element - Here's To You	The dark element here to you.JPG	ogcFCxaSvAA	0	0	0	The DARK ELEMENT is a musical project fronted by t	2018-03-27 22:

Console

Figure 5.7: Song Table (ii)

User can give vote in our system. It represents by figure 5.8.



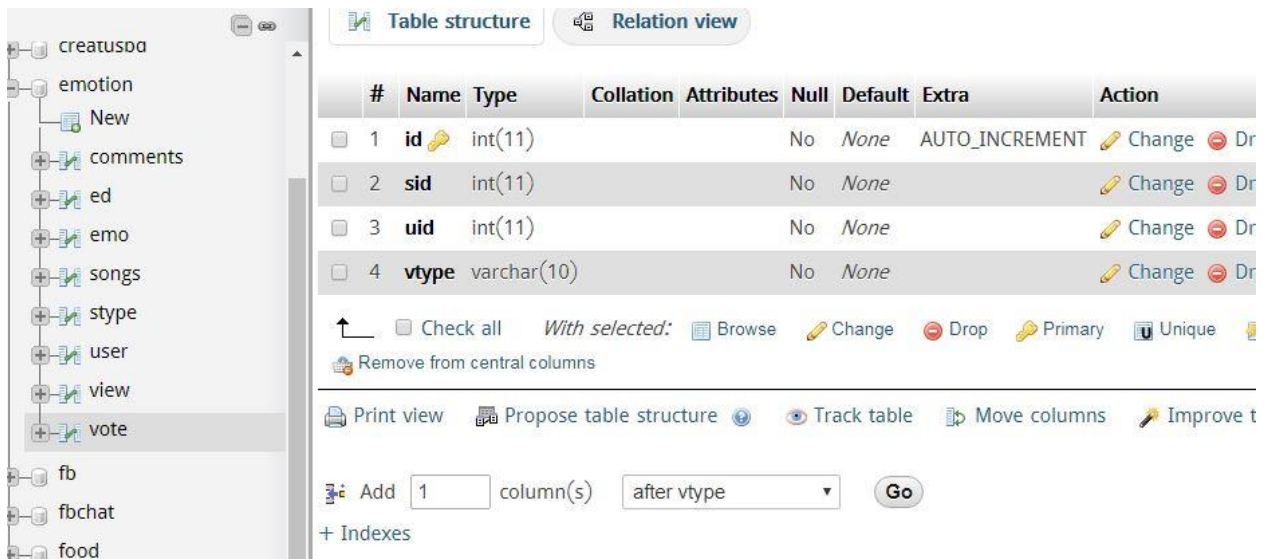


Figure 5.8: Vote Table (i)

There are only two types of vote. LIKE or DISLIKE.

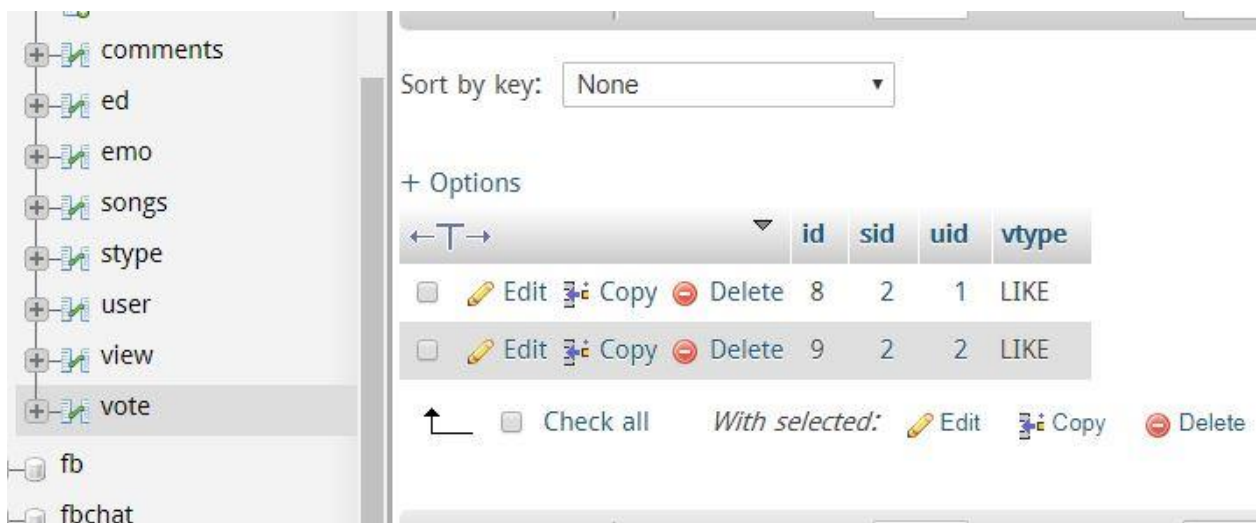


Figure 5.9: Vote Table (ii)

User's table helps to validate login. And the password is saved with Encoded form. Figure 5.10 and 5.11 represents this.

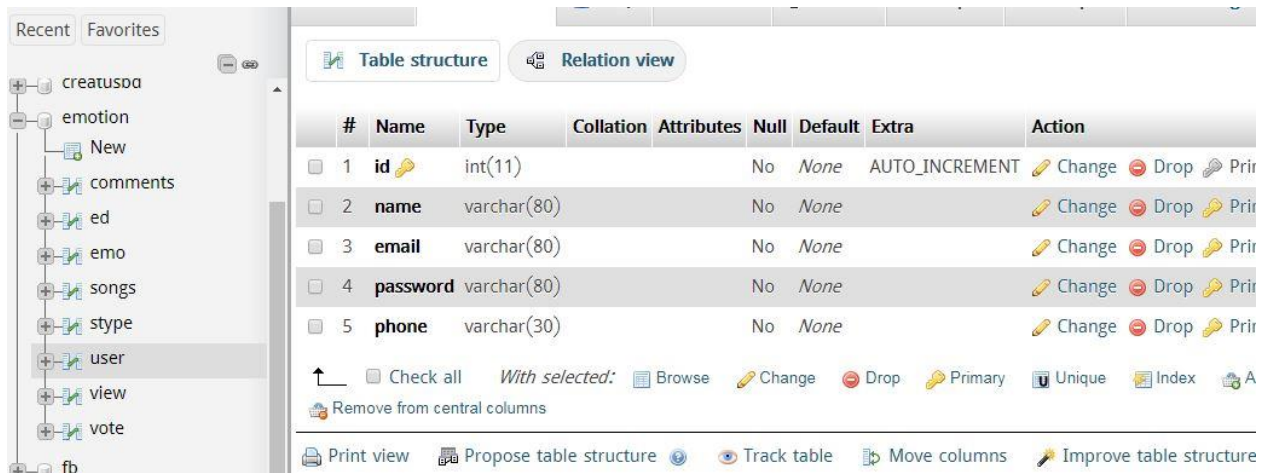


Figure 5.10: User Table (i)

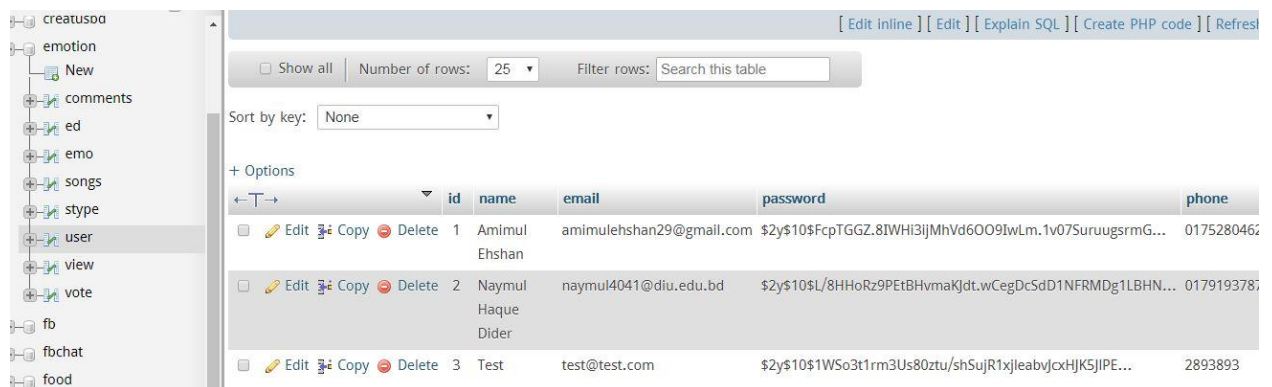


Figure 5.11: User Table (ii)

We also add song type and the Database Design of this part will represent by figure 5.12 & 5.13.

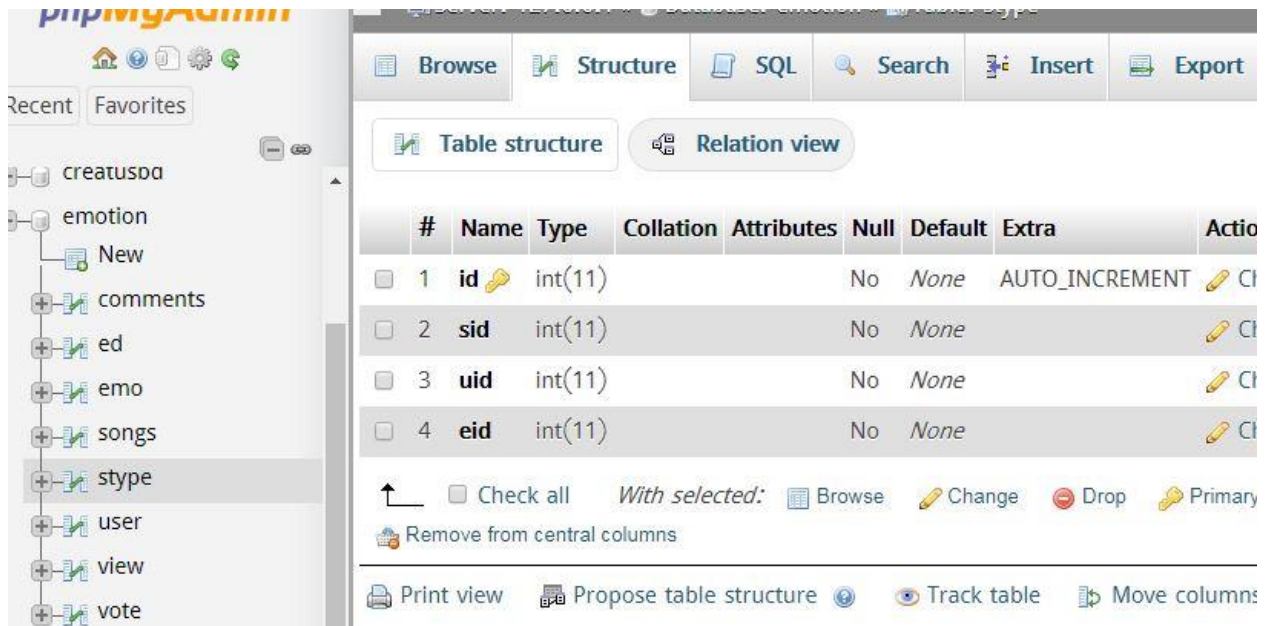


Figure 5.12: Song Type (i)

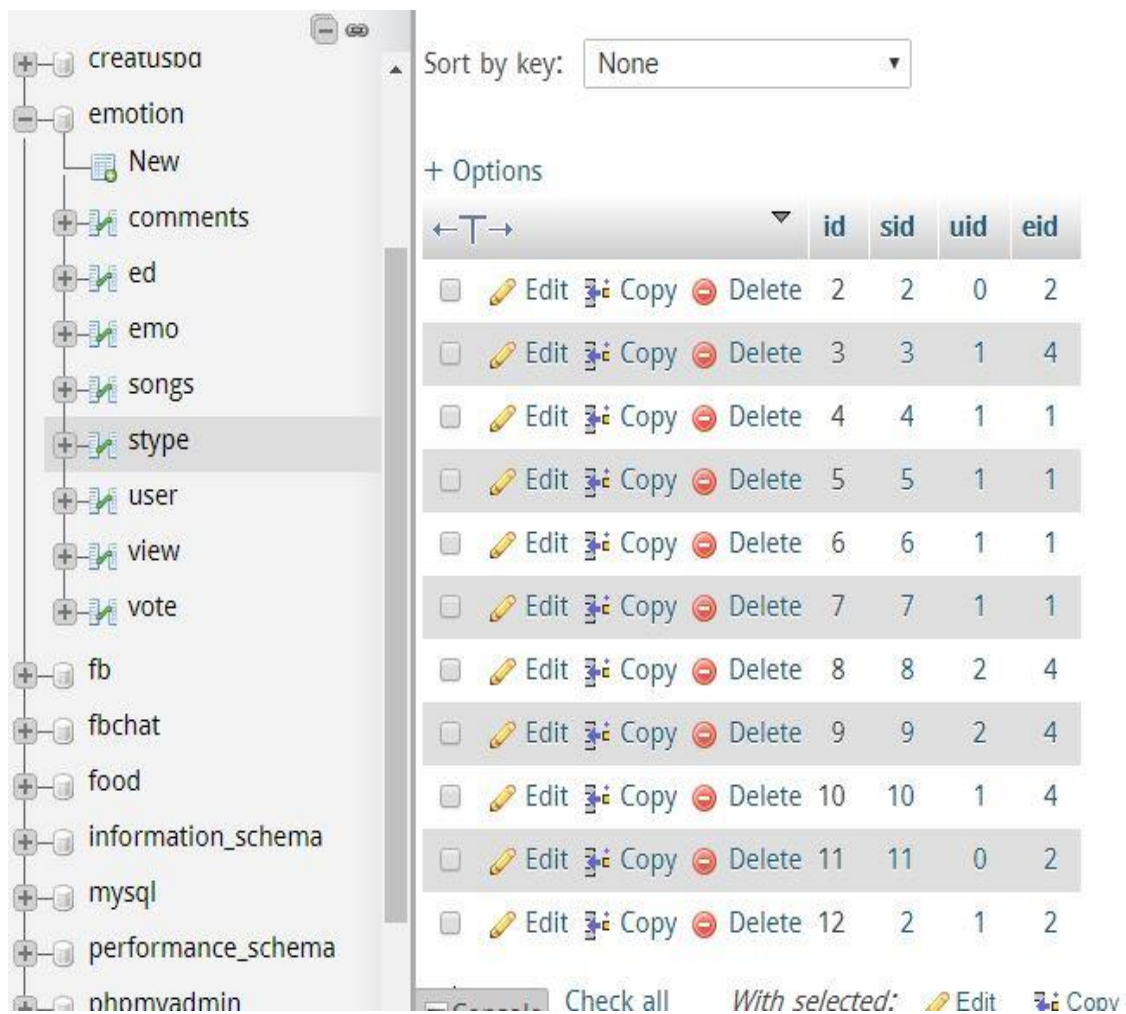


Figure 5.13: Song Type (ii)

## **5.2 Implementation of Front-end Design**

How many user will use a system, is completely depend on the easy design and usability of the system. When a user opens our website, they will firstly see the front end design. Device size is varying from one device to another device. Some device is equipped with heavy hardware, some has large screen, and some are too small. For this we have to make our design device independent and it was very challenging. We tried our best to keep our design gorgeous and also user friendly and responsive. It looks gorgeous both from mobile and computer.

## **5.3 Implementation and Interaction**

The successfulness of a system is completely depends on the interaction with the user. Now a days, interaction is found everywhere. Interaction is which makes a system popular and attractive to a user. So, it is very important to make a system interactive. We developed some unique feature to interact our system with user (like – auto play and repeat)

We implement our system with responsive UI for better user experience from any device like mobile, tablet computer or desktop computer. For this, we have used easy icon, text link and button. The system is totally user friendly and user can easily interact with our system.

Our application is successfully implemented and has impressive interaction with the users.

## **5.4 Testing Implementation**

Testing implementation is process of testing the implementation of a system, where tester or system architect will see various cases and specification, is it implementable or it has limitations.

There are several types of testing:

- Localization
- Disaster recovery
- Installation/ upgrade.
- Scalability
- Usability
- System interoperability
- Functionality
- Regression
- Security

- Performance

Here is the localization testing implementation bellow.

Table 5.1: Testing Implementation

<b>Test Case</b>	<b>Test Input</b>	<b>Expected Output</b>	<b>Actual Output</b>	<b>Result</b>	<b>Tested On</b>
1. Visit website	Tested on various web browser – - Firefox -Google Chrome -Microsoft edge -Safari -Opera -Maxton	Successfully visited from all those versions.	The homepage	Passed	30/03/2018
2. Sign Up	Blank Input.	Restrict access to the features.	Imposed the Restriction.	Passed	30/03/2018
3. Email or Password	Blank or incorrect email or password	Showing a Warning that correct email or password must be entered.	Showed the warning.	Passed	30/03/2018

4. Password	Blank or incorrect password.	Showing a warning that correct password must be entered.	Showed the warning.	Passed	30/03/2018
5. Homepage	Click on the login and provide username/password	To show the Homepage.	Showed the homepage.	Passed	30/03/2018
6. Check Emotion	Click on the Check Emotion Button	Show the current Emotion.	Current Emotion has been shown.	passed	30/03/2018
7. Emotion list	Click on the Category list and select any emotion	To show the Selected emotion type songs.	Showed the songs successfully.	passed	30/03/2018
8. Add Video	Blank Input	Add video will be failed.	Add video was failed.	passed	30/03/2018
9. Add video	Correct input	Video will be added	Video was added successfully.	passed	30/03/2018

10. Watch Video	Select any video and click on the play button on it	Video will be played	Successfully played the video	passed	30/03/2018
11. React	Click on any reaction under the watch video frame	Reaction will be accepted in database	Reaction accepted successfully.	passed	30/03/2018
12. Vote	Click on thumbs up or thumbs down.	Action will be stored in database.	The action stored successfully.	passed	30/03/2018
13. Vote	Click on same vote again	Action will not be saved.	Action saving failed	passed	30/03/2018
14. Log out	Click on the Log out button.	To logout from that account.	Logged out Successfully.	passed	30/03/2018

## 5.5 Test Results and Reports

Test Report represents the result of the test in a formal way. Report contains the data which we evaluated in the professional and organized manner. Report describes the operating condition and shows the test result with test objective.

By seeing the report we can estimate if the system is ready for launch or not.

In chapter 5.4 we have shown the test case, test input, expected output, actual output and finally we find out our results and the test result of this application was successful. We perform

usability test to check the user satisfaction. Usability testing check the following feature of the website.

- How easy it is to use the website?
- How easy it is to learn the website?
- How convenient is the website to end-user?

So at the end we can carry out the results as the benefits of usability testing to the end of the user or learner.

- Good website quality.
- Website is easy to use.
- Website is positively accepted by users.
- All type of information for the new users.
- Easy and understandable UI for interaction.



## CHAPTER 6

### Conclusion and Future Scope

#### 6.1 Discussion and Conclusion

This is the system which people can choose their suitable songs on the demand of their emotions. The design of this application is very user friendly, for this who has minimum knowledge about English language can easily use it.

The main reason behind the developing this application because we see that people are so much confused about choosing a song for the person's demand in basis of his mood. This era is full of modern technology and internet is so much available to a huge percentage of people those who are listening songs of various genre. So we try to solve this problem by technology. Our Application has been successfully tested and implemented. People who has smart device which are connected with internet can use this application.

#### 6.2 Limitations

Like all the things in our world nothing is 100% perfect. Our application also has some limitation. We will try our best to reduce the limitation of our application in future version.

Some of the limitations are:

- Because of using YouTube, users constantly need to connect with internet for playing a song.
- Now, the system recommends only four songs based on the percentage of four type of emotion. But we need to develop it later.
- It needs a lot of data for data mining. As starting, we have very little data in it. The more data we will get, the accurate song will be recommended.

#### 6.3 Scope for Future Developments

- Will improve database for songs quantity.
- Will add more emotions categories.
- Will provide update for this application.
- Will try to make android/ios app about this web application.
- Will improve the accuracy about the emotions.

## References

- [1] Learn about Emotion Recognition <[https://en.wikipedia.org/wiki/Emotion\\_recognition](https://en.wikipedia.org/wiki/Emotion_recognition)> last access on 17-05-2017
- [2] Learn about BPMN <<https://kissflow.com/bpm/business-process-modeling/>> last access on 13-09-2016
- [3] Learn about BPMN <<http://www.bpmn.org/>> last access on 13-09-2016
- [4] Learn about Waterfall Model <[https://www.tutorialspoint.com/sdlc/sdlc\\_waterfall\\_model.htm](https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm)> last access on 02-02-2017
- [5] Learn about ER Diagram <[https://en.wikipedia.org/wiki/Entity%E2%80%93relationship\\_model](https://en.wikipedia.org/wiki/Entity%E2%80%93relationship_model)> last access on 07-06-2016
- [6] Learn about Naïve bayes <[https://gerardnico.com/data\\_mining/naive\\_bayes](https://gerardnico.com/data_mining/naive_bayes)> last access on 14-11-2017
- [7] Learn about Naïve bayes <<https://php-ml.readthedocs.io/en/latest/machine-learning/classification/naive-bayes/#train>> last access on 14-11-2017
- [8] Create DFD <<https://www.lucidchart.com/invitations/accept/41b61870-fbb1-4309-8339-b1681b91fddd>> last access on 24-05-2017
- [9] Create BPMN <<https://www.lucidchart.com/invitations/accept/9148e0ae-5785-46cd-9e5f-03274e13f6a2>> last access on 24-05-2017
- [10] Create Flow Chart <<https://www.lucidchart.com/invitations/accept/3aed2eaa-9283-45a5-ae58-adcbc801e3b4>> last access on 24-05-2017
- [11] Learn about EDA activity <<https://liu.academia.edu/LarsThorell>> last access on 25-11-2017
- [12] Yajie Hu, Mitsunori Oghihara “NEXTONE PLAYER: A MUSIC RECOMMENDATION SYSTEM BASED ON USER BEHAVIOR” ISMIR 2011
- [13] Learn about PHQ9 <<https://patient.info/doctor/patient-health-questionnaire-phq-9>> last access on 24-11-2017
- [14] Mandal MK, Pandey R, Prasad AB “Facial expressions of emotions and schizophrenia: a review” [Research Support, Non-U.S. Gov 1982]
- [15] Survey Form<[https://docs.google.com/forms/d/e/1FAIpQLSftxljb87JBpYC\\_k0\\_UeVeb5EQ9rP3EsCrw7Ag0uOTIjitiwWw/viewform](https://docs.google.com/forms/d/e/1FAIpQLSftxljb87JBpYC_k0_UeVeb5EQ9rP3EsCrw7Ag0uOTIjitiwWw/viewform)> last access on 27-03-2018
- [16] Roberto Zangróniz, Arturo Martínez-Rodrigo, José Manuel Pastor, María T. López, and Antonio Fernández Caballero “Electrodermal Activity Sensor for Classification of Calm/Distress Condition” [MDPI, Basel, Switzerland, Oct, 2017]
- [17] Learn about lbp-sobel <<http://ieeexplore.ieee.org/iel5/4667700/4711669/04712212.pdf>> last access on 26-04-2017
- [18] Shiv Ram Dubey “Face Retrieval using Frequency Decoded Local Descriptor” [IEEE 2007]
- [19] Learn about Eigen vector <<http://mathworld.wolfram.com/Eigenvector.html>> last access on 16-07-2016
- [20] Matti Pietikäinen, Abdenour Hadid, Guoying Zhao, Timo Ahonen “Local Binary Patterns for Still Images” [Springer, London 2011]

- [21] Learns about js <<https://www.w3schools.com/js/default.asp>> last access on 21-03-2016
- [22] Learn about js <<https://www.javascript.com/>> last access on 21-03-2016
- [23] Learn about js <<https://developer.mozilla.org/en-US/docs/Web/JavaScript>> last access on 21-03-2016
- [24] Learn about php <<https://www.youtube.com/watch?v=iCUV3iv9xOs&list=PL442FA2C127377F07>> last access on 27-05-2016
- [25] Learn about php <<https://www.w3schools.com/php/default.asp>> last access on 27-05-2016
- [26] Learn about php <<http://www.php.net/>> last access on 27-05-2016
- [27] Learn about php <<https://www.tutorialspoint.com/php/index.htm>> last access on 27-05-2016
- [28] Learn about Eigen vector <<http://mathworld.wolfram.com/Eigenvector.html>> last access on 16-07-2016
- [29] Learn about Eigen values <[https://en.wikipedia.org/wiki/Eigenvalues\\_and\\_eigenvectors](https://en.wikipedia.org/wiki/Eigenvalues_and_eigenvectors)> last access on 16-07-2016
- [30] Learn about Eigen vector <<http://ipsa.swarthmore.edu/MtrxVibe/EigMat/MatrixEigen.html>> last access on 16-07-2016
- [31] Learn about eigen vector <<http://math.mit.edu/linearalgebra/ila0601.pdf>> last access on 16-07-2016
- [32] Learn about mysql <<https://www.mysql.com/>> last access on 27-05-2016
- [33] Learn about mysql <<https://en.wikipedia.org/wiki/MySQL>> last access on 27-05-2016
- [34] Learn about php mysql <[https://www.w3schools.com/php/php\\_mysql\\_intro.asp](https://www.w3schools.com/php/php_mysql_intro.asp)> last access on 27-05-2016
- [35] Learn about mysql <<https://www.oracle.com/mysql/index.html>> last access on 27-05-2016
- [36] Learn about XAMPP <<https://www.apachefriends.org/index.html>> last access on 11-02-2016
- [37] Learn about ajax <[https://www.w3schools.com/php/php\\_ajax\\_intro.asp](https://www.w3schools.com/php/php_ajax_intro.asp)> last access on 04-06-2016
- [38] Learn about jquery ajax <<https://api.jquery.com/category/ajax/>> last access on 04-06-2016
- [39] Learn about D3 <<https://d3js.org/>> last access on 24-03-2017
- [40] D3 sources <<https://github.com/d3/d3>> last access on 24-03-2017
- [41] Learn about MUCT <<http://www.milbo.org/muct/>> last access on 15-11-2017
- [42] Learn about MUCT <<http://www.milbo.org/muct/The-MUCT-Landmarked-Face-Database.pdf>> last access on 15-11-2017
- [43] Learn about MUCT <<http://www.milbo.org/muct/muct-details.html>> last access on 15-11-2017
- [44] MUCT Database <<https://github.com/StephenMilborrow/muct>> last access on 15-11-2017
- [45] Learn about Image Processing <[http://www.ri.cmu.edu/pub\\_files/2009/9/CameraReady-6.pdf](http://www.ri.cmu.edu/pub_files/2009/9/CameraReady-6.pdf)> last access on 27-03-2017
- [46] Learn about Clmtrackr <<https://github.com/auduno/clmtrackr>> last access on 29-10-2017

## **APPENDIX**

### **Appendix A: Project Reflection**

The purpose of this appendix is about project reflection. From Summer 2017 semester we started our journey to make this website. The main feature of our application is detecting emotion. Also using our website people can easily find the song he likes. In the very first days, we studied a lot of papers about emotion and others. There we knew about relation of human behavior with EDA and Face. We have studied a lot to detect the emotion. And we also tried to do it with EDA or Electro Dermal Activity. But, it will not be user friendly. So, we have not used it. Then we studied PHP and JavaScript to use those effectively. We also had to study a lot about D3 js and Clmtrackr to use the MUCT Face Database, where are about 3755 faces. Firstly we built a model for our project. Then we implemented our project step by step. We also took a survey to find which type of song people listen in four kind of emotion. After many hard works and spending a lot of time finally we were able to reach our goal.

The project "Emotion Based Song Recommendation - Emotube" will be very helpful for every normal person. People will be satisfied very easily.

So we believe that our "Emotion Based Song Recommendation - Emotube" application will be very positive and effective for users.

### **Appendix B: Related Website**

We did not find any related website like this, but there are an android app for emotion based song player. But, this is an offline application. Our website is online based and it updates her database always. So, the result of our project will be more accurate day by day.

# PLAGIARISM REPORT

Checked By: [www.plagamme.com](http://www.plagamme.com)

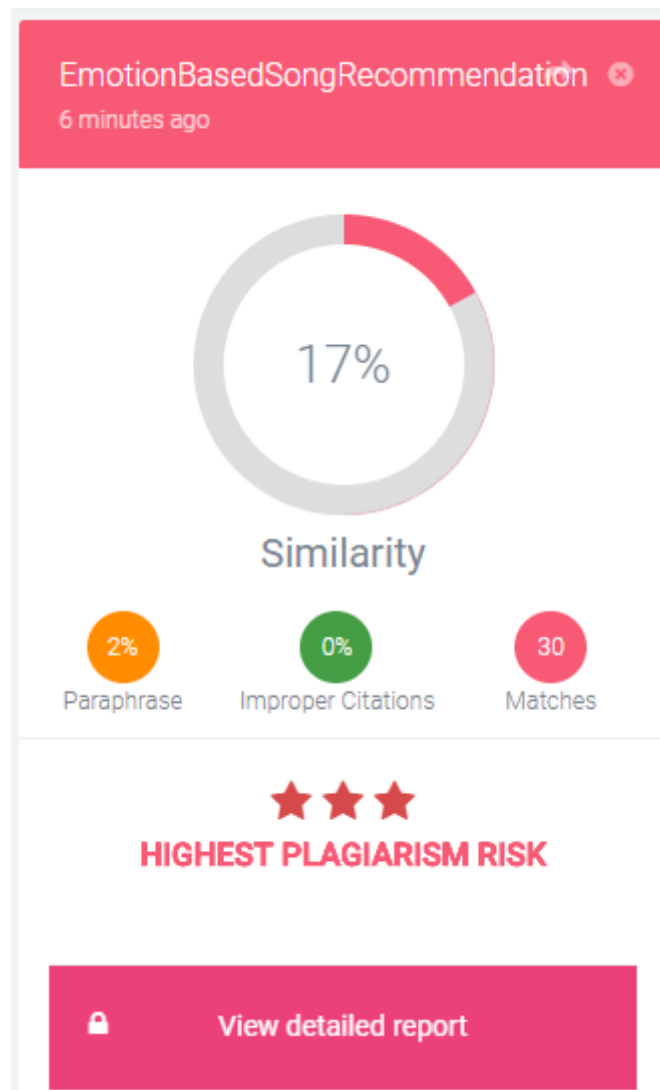


Figure: Plagiarism Report