System Design Specification
of
IoT Based Smart Mirror with Human Emotion Recognition System

Coder Comrades		
Name	Registration Number	Index Number
W.G.D.J. Shashipriya	ICT/19/20/106	5042
I.K.G.A.M. Karunarathna	ICT/19/20/060	4998
N.K. Lokusooriya	ICT/19/20/068	5006
U.W.G.R.P. Perera	ICT/19/20/081	5019
Y.M.A.H.B. Yapa	ICT/19/20/124	5059

2024/04/18

Table of Contents

List of Figures

CHAPTER 1 - INTRODUCTION

1.1.Problem to be Addressed

Smart mirrors have emerged as a prominent feature in modern homes, seamlessly integrating into our daily routines. These innovative devices transcend the traditional role of mere reflection, transforming into information hubs that display news updates, weather forecasts, and even personalized schedules. However, despite their growing sophistication, a crucial element remains absent - the ability to understand a user's emotional state. This inability to recognize emotions presents a significant limitation. Our emotional well-being is intricately linked to various aspects of life, from physical health and mental clarity to the quality of our relationships and the effectiveness of our daily activities. A frustrated user rushing out the door might benefit from a reminder to breathe and re-center, while a tired individual might appreciate suggestions for a rejuvenating morning routine. Without the ability to discern emotions, smart mirrors currently lack the capacity to offer truly personalized and impactful support.

1.2.Objectives of the Project

The objectives of our proposed system are,

1) To develop a Smart Mirror prototype using IoT to recognize human emotions using facial expressions.
2) To train a machine learning model for recognizing and analyzing human emotions such as happy, sad, angry, fear, and neutral emotions using the user's facial expressions in real-time.
3) To design a user-friendly smart mirror using web technologies with IoT devices such as an embedded display.
4) To develop a content management system for displaying and delivering emotion-driven content such as music, videos, and information according to that emotional state and playing music.
5) To implement the smart lighting control system for ambient lighting control that aligns with recognizing emotional state.

1.3.Project Deliverables

In this section, we'll outline the project deliverables, focusing on key components essential for project success. These deliverables include the IOT Device and Emotion Recognition Model, serving as foundational elements for system functionality. Additionally, we'll provide the Final Technical Report, Installation Guide, and User Manual to ensure comprehensive understanding and seamless deployment. Progress Reports and Test Plan/Test Cases facilitate ongoing evaluation and quality assurance. Finally, Documentation and Training Materials support user adoption and system maintenance. Each deliverable plays a crucial role in achieving project objectives and ensuring successful implementation.

1. IOT Device
2. Emotion Recognition Model
3. Final Technical Report
4. Final Thesis
5. Installation Guide
6. User Manual
7. Deployment Packages

1.4.System Design Approach

In this system design approach part, we consider about the software development part.

Process Model – SCRUM

The project consists of three implementations, so there are three sprints in this project.

(Daily working time is 10 hours per day) (20 working days in 4 weeks (month))

No of sprints in Implementation And functions to cover in each sprint	**No of Sprint**	**Functions**	**Sub functions**	**Percentage completed**	**No of Hours**
First implementation					
1 (70% of functional and non-functional requirements)	1st	Planning		75%	10
		Research		25%	10
		covered up the functional requirements to 70% before first Implantation (nearly 10% completed)			
			User authentication	5%	15
			Emotion recognition	5%	30
			Ambient light control system		15
			Content recommendation system		15
		train the model			20

		recommendation systems			10
		Database development			20
Second implementation					
No of sprints in Implementation And functions to cover in each sprint	**No of Sprint**	**Functions**	**Sub functions**	**Percentage completed**	**No of Hours**
1 (complete functional and non-functional requirements)	2nd	• Research			10
		Develop the modules and testing - Pre-processing module			30
		Remaining function development			
			User authentication (complete)		0
			Emotion recognition model (60% complete)		10
			Ambient light control system (70% complete)		5
			Content recommendation system (70% complete)		5
		Unit testing on each function			
			User authentication		2
			Emotion recognition model		4
			Ambient light control system		2
			Content recommendation system		2
		debugging			15
		UI designing			15

		Host model in server and test			20
Third implementation					
No of sprints in Implementation And functions to cover in each sprint	**No of Sprint**	**Functions**	**Sub functions**	**Percentage completed**	**No of Hours**
1 (handover the project and finalize the documentation)	3rd	Combining all parts together			10
		System integration and testing			20
		debugging			15
		write the final dissertation			10
		Research			10
		design Project poster			10
		write the Research paper			15
		create the User manual guide			10

Sprint backlogs and allocated times

1st sprint

Duration: 4weeks

Allocated time: 135 hours

Sprint backlog	Allocated time(hours)
Planning	10
Cover functional requirements (60%)	
User authentication	15/15
Emotion recognition model	30/40
Ambient light control system	15/20
Content recommendation system	15/20
Model training	20
Research	10
Data base development	20

2nd sprint

Duration: 3weeks

Allocated time: 120hours

Sprint backlog	Allocated time (hours)
Research	10
Remaining function development	
User authentication	0(completed)
Emotion recognition model	10/40(nearly 60% complete)
Ambient light control system	5/20(nearly 70% complete)
Content recommendation system	5/20(nearly 70% complete)
Module development and testing	30
Unit testing on each function	
User authentication	2
Emotion recognition model	4
Ambient light control system	2
Content recommendation system	2
Debugging	15
Ui designing	15
Hosting model and testing	20

3rd sprint

Duration: 2Weeks

Allocated time: 100hours

Sprint backlog	Allocated time (hours)
Combining all parts together	10
System integration and testing	20
debugging	15
Write final dissertation	10
Design project poster	10
research	10
Write research paper	15
Create training manual guide	10

1.5.Standards to be followed.

Database:

- MySQL workbench (follows ISO/IEC standard "Information Technology – Database Languages – SQL")

Documentation:

- IEEE standards for citations: We use Author-date citation style.
- Figures, Tables, Equations: We use specific formatting and placement guidelines for this.

Coding Standards:

- ❖ Important to maintain consistency and readability in Python codebase.
- ❖ Following the respective language-specific conventions PEP 8 for Python .

1.6.Organization of the SDS

Introduction

In this section, we discuss the problem to be addressed and the objectives of the project. Additionally, we focus on the process model and the system design approach, which includes Object-Oriented Design. Furthermore, we delve into the standards that need to be followed during the creation of the system application. Finally, we provide an overview of the organizational structure of the SDS, concluding this chapter.

Architectural Design

In the second part of this document, we'll talk about how the system is built, the different pieces that work together, and how they talk to each other. We'll also briefly explain the steps the program takes and any special tricks it uses. Finally, we'll cover things like what keeps track of where the program is at, what tools it uses to work, and where it all runs.

UI Design

The third chapter starts by explaining what PACT analysis is and why it's important. Then it dives into how to design the user interface (UI) to make things easy and enjoyable to use. In the middle, it gives designers a toolbox of tricks and templates to make their job easier. It also covers how to design the way users give instructions (input) and get information (output) from the system, to keep things smooth. Finally, it explains where the system will live and how to set it up.

Data Management

This chapter dives into what data the project needs to work properly. Then, it explains the tools used to design the system and why they're important. We'll also look at special diagrams that visually represent how the data is organized. Finally, the chapter wraps up by explaining how the system keeps your information safe.

Research Design

This chapter explains what we're trying to achieve based on what we learned from all the research we did. It then outlines the main parts of the system we're building. In the middle, we'll talk about how we get the information we need, how we design the examples the system will learn from, and how we split this data for testing and training. Finally, we'll briefly touch on how well the system runs beyond its core function, how we'll check if it's working correctly, and how we'll measure how good it really is.

Approval

Finally, there is the approval of the supervisors and the details of the team members.

CHAPTER 2 - ARCHITECTURAL DESIGN

In this chapter, the architectural design of the IoT based Smart Mirror with Human Emotion Recognition System is discussed. The architectural design includes the system architecture, objects and communication, state machines, processes and special algorithms, tools, techniques, libraries, third party tools, and the implementation environment. In this chapter, each of the subsections is separately discussed.

2.1.System Architecture

The system architecture of this proposed system refers to the high-level design and structure of the components, modules, and interactions within the software system. It overviews how different system elements are organized and work together to achieve the objectives.

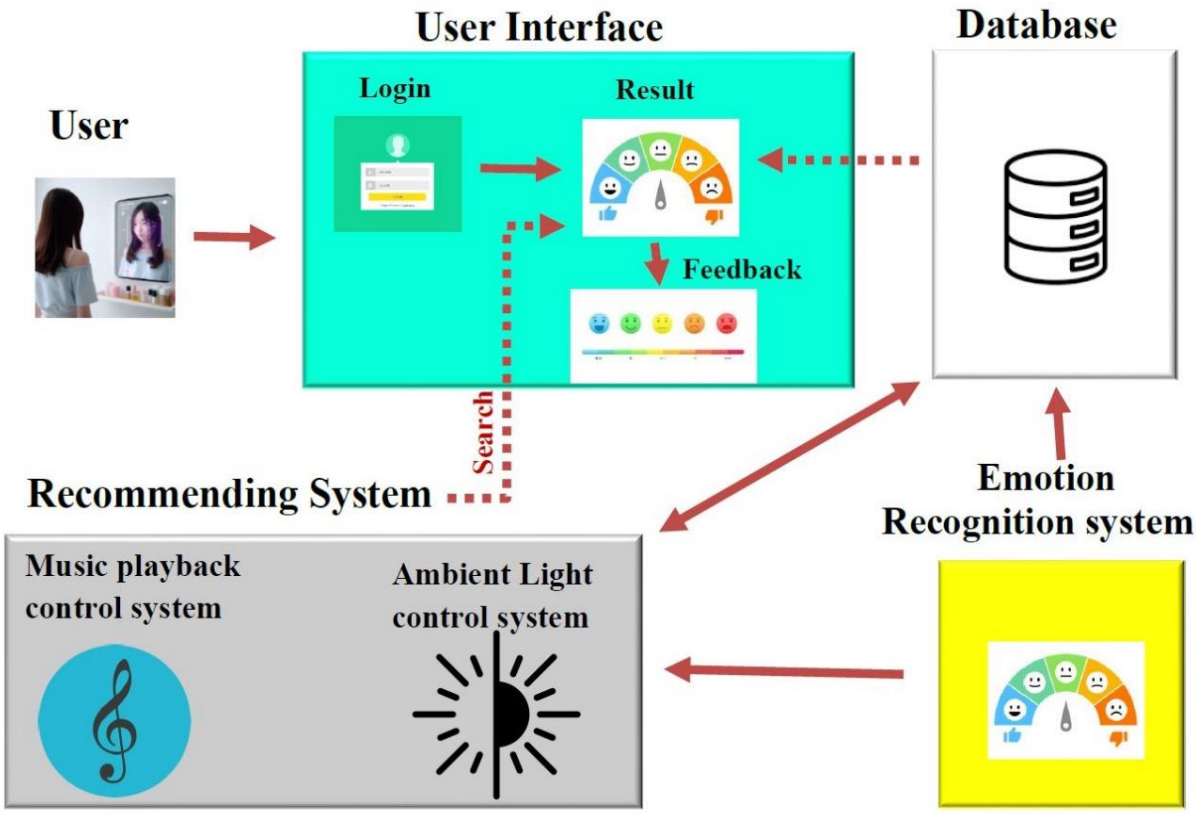


Figure 2.1:Overview of the IoT Based Smart Mirror with Human Emotion Recognition System

2.1.1. Organization of the System

A layered architecture is the most suitable approach to describe the organizational structure of our Smart Mirror System. This breaks down the system into distinct layers, each with specific functions and responsibilities.

Presentation Layer: - This layer displays information and interacts with the user. It focuses on providing a user-friendly interface for showing information, searching, displaying recommendations, playing music, adjusting lighting, and doing updates.

Business Logic Layer: - This layer contains the core business logic of the system. It handles functionalities such as running the operating, controlling hardware components, analyzing facial expressions, selecting recommendations and music, and controlling ambient lighting.

Data Layer: - This layer is responsible for storing and managing data related to the system, such as user information, captured images, emotion recognition results, and recommended content. It provides efficient data retrieval and storage capabilities, ensuring data integrity and consistency.

2.2.Objects and Communication

2.2.1. Sequence Diagrams

This sequence diagram (Figure 2.2) illustrates the interactions between the user interface and the backend system during the sign-up process. It shows the steps involved in registering a new user, including entering registration details, validating inputs, and storing user information in the database. Additionally, it may depict error handling scenarios and confirmation messages upon successful registration.

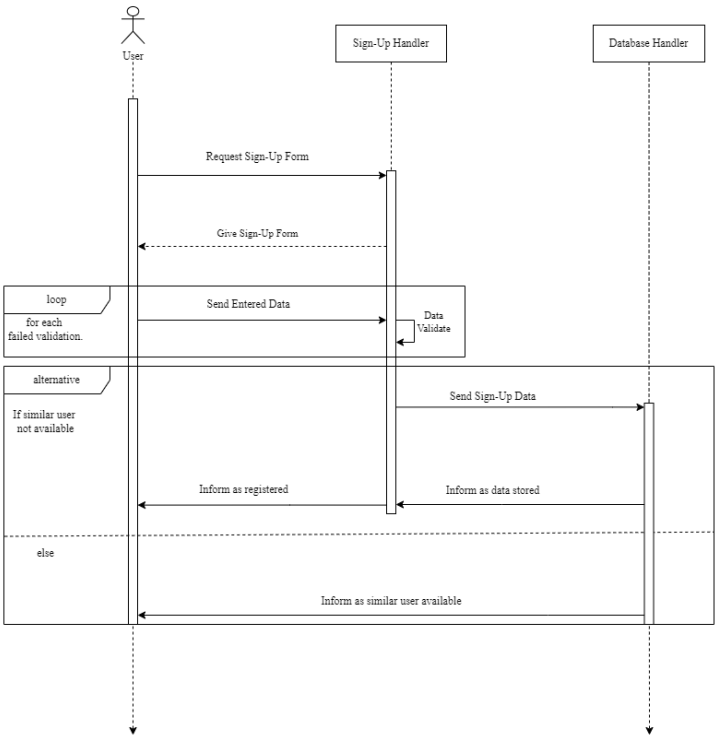


Figure 2.2: Sequence Diagram of Sign-Up

This sequence diagram (Figure 2.3) depicts the flow of interactions between the user and the system when logging in. It outlines the steps for authenticating user credentials, verifying user identity, and granting access to the system's features upon successful login. Error handling for invalid credentials or login failures may also be included in the diagram.

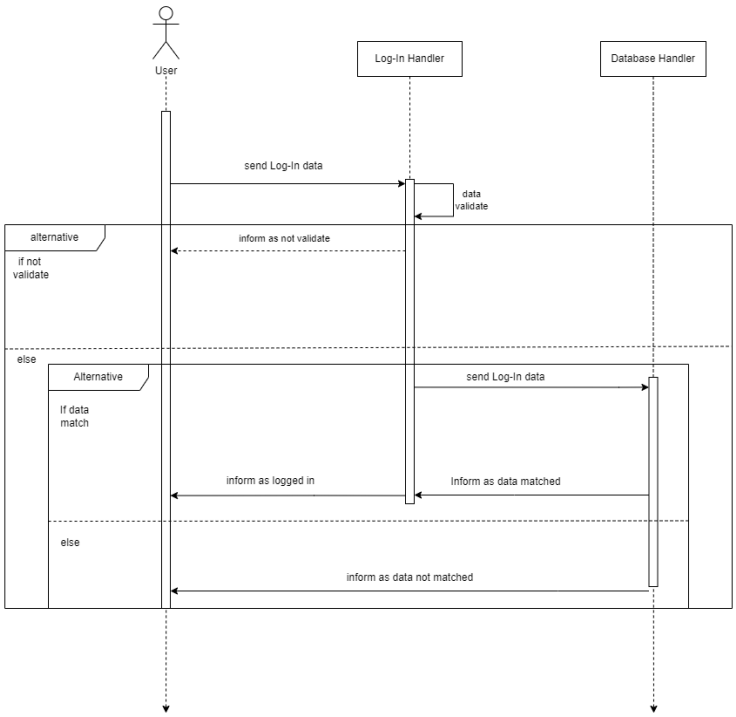


Figure 2.3: Sequence Diagram of Log-In

This sequence diagram (Figure 2.4) illustrates the process of emotion recognition performed by the smart mirror system. It outlines the steps involved in capturing a user's facial expression, processing the image data, and applying emotion recognition algorithms to analyze and classify the detected emotions. The diagram may include interactions with the facial recognition library, image processing modules, and database queries for storing emotion data.

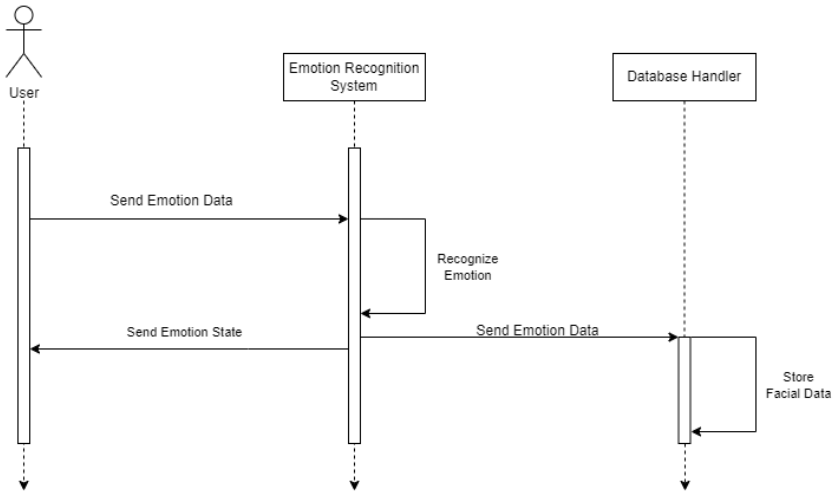


Figure 2.4: Sequence Diagram of Emotion Recognition

This sequence diagram (Figure 2.5) showcases the interactions between the user interface, content management system, and database during content management operations. It outlines the steps for retrieving, updating, and displaying content on the smart mirror, including user preferences, widgets, and notifications. The diagram may illustrate how users interact with the system to customize their content preferences and how the system responds to these interactions through content retrieval and rendering processes

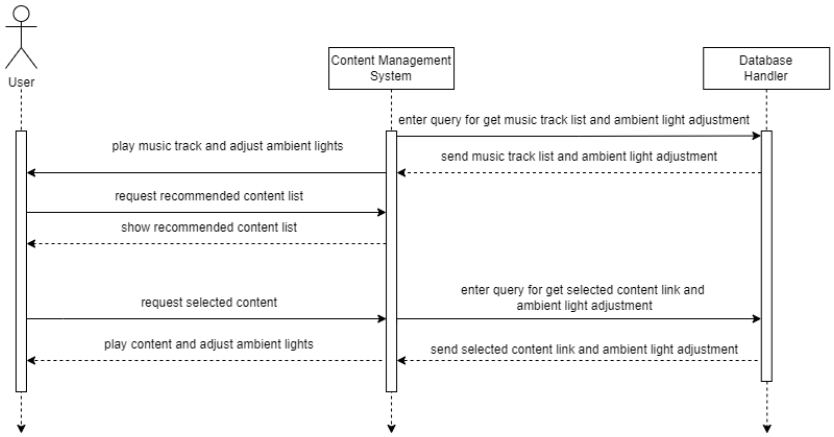


Figure 2.5: Sequence Diagram of Content Management

2.2.2. Class Diagram

The Class Diagram of the System, represented in Figure 2.6, provides a comprehensive overview of the system's structure, detailing the classes, their attributes, methods, and relationships. Each class represents a distinct entity or concept within the system, encapsulating its properties and behaviors. The diagram illustrates how these classes interact and collaborate to fulfill the system's functionalities. It serves as a blueprint for system design and implementation, guiding developers in the creation of robust and maintainable software solutions. Through the Class Diagram, stakeholders gain insights into the system's architecture, facilitating communication, decision-making, and further development efforts.

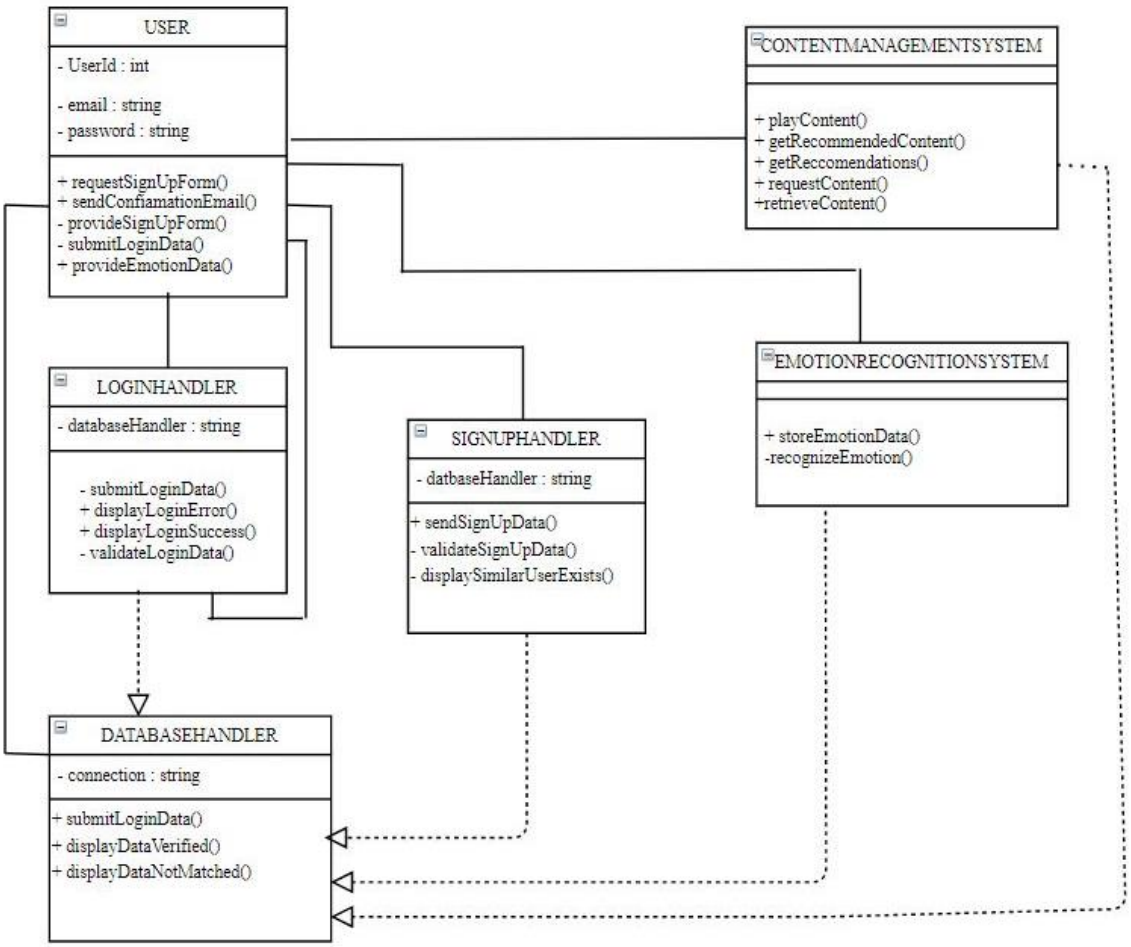


Figure 2.6: Class Diagram of the System

2.3.Processes and Special Algorithms

2.1.1. User Processes:

- User Registration and Authentication:
  - Users initiate the registration process by providing necessary information such as username, email, and password.
  - The system verifies user credentials and authenticates users based on provided information.
  - Upon successful authentication, users gain access to system features and functionalities.

- Emotion Recognition and Analysis:
  - Users interact with the system through the smart mirror interface, which captures facial expressions using built-in camera sensors.
  - Emotion recognition system processes captured images and analyzes facial features to detect emotions such as happiness, sadness, or surprise.
  - Analyzed data is sent to the backend system for further processing and storage.

- Content Recommendation and Personalization:
  - Based on user preferences, past interactions, and historical data, the system recommends personalized content and learning materials.
  - Users receive tailored recommendations through the content management system interface.
  - Feedback mechanisms allow users to provide input on content relevance and satisfaction, which further refines the recommendation algorithm.

- Ambient Light Control and Environment Adaptation:
  - Ambient light sensors integrated into the smart mirror interface detect environmental lighting conditions.
  - The system adjusts light brightness and color temperature dynamically based on detected light levels to optimize visibility and user comfort.
  - Users have the option to manually override automatic adjustments through the smart mirror interface.

2.1.2. Special Algorithms:

- Facial Recognition Algorithm:
  - To facial feature extraction and emotion classification in the emotion recognition system.
  - To analyzes facial expressions captured by the smart mirror camera sensors to detect and classify emotions.

- Proportional-Integral-Derivative (PID) Control Algorithm:
  - To real-time adjustment of display brightness and color temperature based on ambient light sensor readings in the ambient light control system.
  - To precise and dynamic control of display settings to optimize user experience in varying lighting conditions.

2.4.State Machines

When the user submits the registration form, the system should create a new record in the database with the provided form details, including name, email, password, and account status set to 0. The system should change the activation status in the database to 1 when the user activates the account.

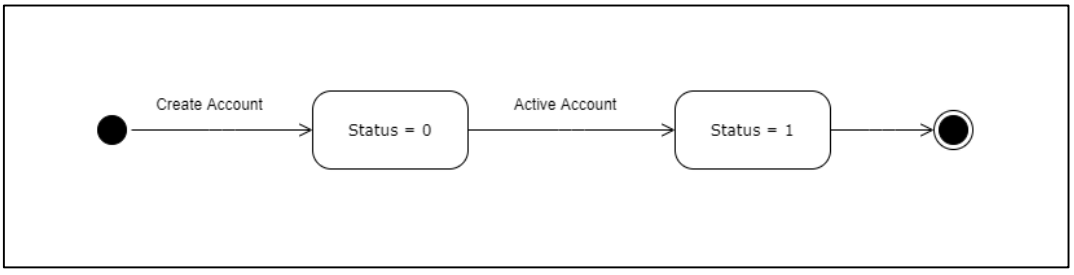


Figure 2.7: State Machine

2.5.Deployment of the System

A component diagram visually illustrates the primary components of a system and their relationships. Each component represents a modular building block encapsulating specific functionality or services. These components, including software modules or libraries, collaborate to achieve the system's objectives. The diagram offers a clear overview of the system's architecture, aiding communication among project teams and facilitating decision-making. It serves as a blueprint for system design and development, providing valuable insights into component relationships and interactions.

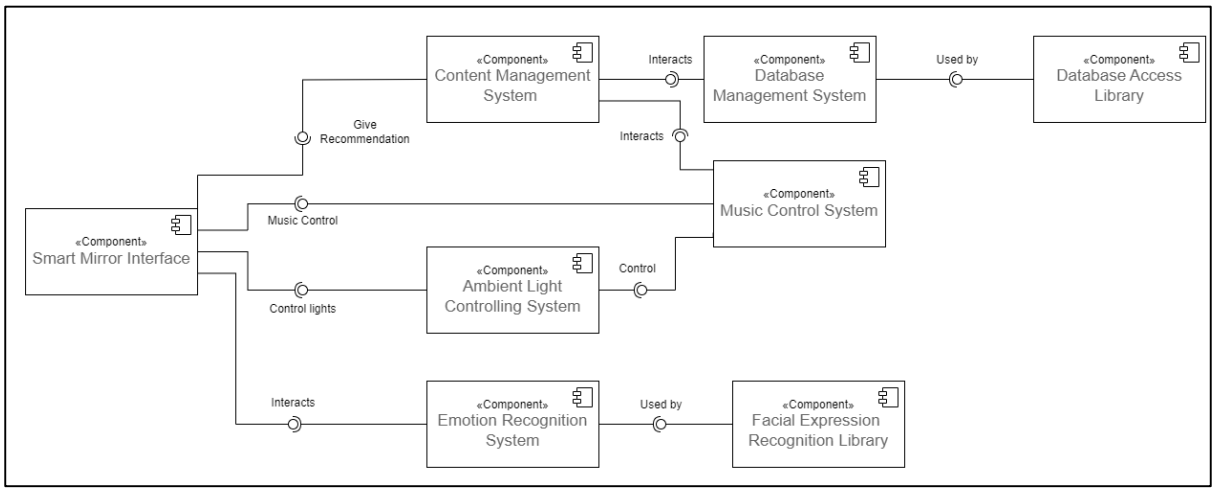


Figure 2.8: Component Diagram

2.6.Integrating and Developing Toolkit

- **Tools and Technologies:**
  - Draw.io:
    Utilize Draw.io to craft comprehensive diagrams illustrating the system architecture and workflow processes. Leverage its intuitive interface and diverse features to create visually appealing representations of various system components and interactions.

  - Adobe XD:
    Employ Adobe XD for designing and prototyping the user interfaces of our application. Collaborate efficiently with designers and developers to refine the user experience and iterate on interface designs using interactive prototypes.

- **Implementation Environment**
  - Visual Studio Code:
    Incorporate Visual Studio Code into our development workflow as the primary text editor for implementing the backend logic of our software. Leverage its extensive feature set, including syntax highlighting and debugging capabilities, to streamline the coding process.

- Anaconda Navigator:
  Utilize Anaconda Navigator to manage Python packages and environments effectively for our machine learning models. Simplify the process of setting up the development environment and ensure compatibility with the required libraries and dependencies.

- **Libraries**
  - TensorFlow:
    Integrate TensorFlow into our project to leverage its robust framework for building and training neural network models. Utilize its flexibility and extensive support for tasks such as image recognition to develop the models required for our application.

  - Keras:
    Simplify the process of building deep learning models using Keras, which offers a high-level API compatible with TensorFlow. Experiment with different architectures and configurations effortlessly to optimize model performance.

  - OpenCV:
    Utilized in combination with Python for implementing various image processing techniques. OpenCV's comprehensive set of functions and algorithms facilitate tasks such as image filtering and object detection, enhancing the application's image processing capabilities.

- **Languages**
  - MySQL:
    Employed for storing and retrieving structured data in the application's backend. MySQL's reliability and performance make it suitable for managing data effectively, ensuring seamless interaction with the application's database.

  - Python:
    Utilized for implementing the backend logic and data processing components of the system. Python's simplicity and readability, along with its versatility, make it well-suited for tasks ranging from web development to machine learning, enhancing the application's functionality and performance.

CHAPTER 3 - UI DESIGN

3.1.PACT (People, Activities, Contexts, Technologies) analysis

PACT analysis is a framework used to understanding the user's needs the preferences in terms of their activities, context, and technology. Here is a PACT analysis of the user interface in Human emotion recognizing smart mirror.

3.1.1. People

There are many different ways in which people differ from each other. That mainly depends on their interests. Hence it is important to ensure that the system is using clear language and providing easy navigation. The user interface should be designed with the users in mind, including their knowledge level, language, and cultural background.

With our emotion recognition smart mirror mainly target the young generation and people who should care their health because positive emotions are highly affecting human health many ways. And to use this system user need only very low computer literacy.

3.1.2. Activities

When the system power on the users should log in/sign in to the system. The typical tasks and goals of the users when interacting with smart mirror include identifying user's current emotional state, and overcome negative emotions or improve positive emotions.

The typical tasks and goals of the smart mirror may include the improver user's health with making user happy and make user positive emotions. The tasks must possess special characteristics in order to have a better interactive application.

Some of the main characteristics of the tasks that need to be considered are

- **Load time of the application**

The application should be simple enough to be able to launch within a short period of time and we can reduce it to 5 seconds in our mobile application.

- **Processing time of the application**

The user-friendliness of our application is enhanced by reducing the processing time to less than 5 seconds which leads to higher customer satisfaction.

- **Grab the data from the built-in camera**

The photos capturing time is reduced to 5 seconds by capturing image like 1 second so that the user won't get exhausted and change his mind to quit the process.

3.1.3. Context

System context analysis is the study of the environments in which the system operates. System boundary, inputs to the system from the environment, the components inside it, data processing, movement between components, and providing outputs to the environment are basically discussed under this topic. When we consider the physical environment that affects our project. The Google drive, ram, and mobile storage of our system mainly affect the running of applications. Applications will operate more slowly if system has very little RAM or storage, and very slow internet connection when content recommendation.

People in countries like Sri Lanka often experience various stresses. Our goal is to alleviate this stress through innovative methods. This system is designed to assist individuals in maintaining their emotional well-being. It will provide support akin to having a trusted friend readily available whenever needed.

3.1.4. Technology

Inputs

- **Text box:** When the user needs to enter the Name or anything a user can use those text boxes
- **Camera:** The camera is provided through the system to capture user images to identify user's current emotional state.
- **Pointing device:** To select contents system should use pointing device like touch screen in this prototype we use mouse and keyboard to reduce cost.

Outputs

- **Phone display**: almost all the outputs are given from the in-built display
- **Size of the screen:** In here we use 15inch screen to show contents inside the mirror.
- **In-built speaker:** this will use to play music that when user need to ask songs or when system recommend user to listen songs or music.

	Sign up	Log in	Identify emotions	Music recommendation based on current emotion	Changing ambient light based on current emotion
People					
Cognitive characteristics					
Level and duration of attention	✓	✓	✗	✗	✗
Educational level	✓	✓	✗	✗	✗
Fears	✗	✗	✗	✗	✗
Personality characteristics	✗	✗	✗	✓	✓

Activity					
Task	✓ Entering their personal information, create user name and password	✓ Enter their user name and password	✓ Siting in front of mirror then power on and click to scan face	✓ Select Music from recommendation list or own favourite list	✓ Click to change ambient light based on current emotion or select from color map.
Individual	✓	✓	✓	✓	✓
Continuous	✓	✓	✓	✓	✓
Frequently	✗	✗	✗	✓	✓
Context					
Lightning	✓	✓	✓	✓	✓ .

Background color	✓ (Background color: #111111)	✓ (Background color: #111111)	✓ (Background color: #111111)	✓ (Background color: #111111)	✓ (Background color : #111111)
Storage	✓	✓	✓	✓	✗
User friendly	✓	✓	✓	✓	✓
Used any time	✓	✓	✓	✓	✓
Cultural context	✗	✗	✗	✗	✗
Technology					
Input					
Camera	✗	✗	✓	✗	✗ .
Pointing device	✓	✓	✓	✓	✓
Text box	✓ To enter signup details	✓ To enter login details	✗	✗	✗
GUI	✓	✓	✓	✓	✓
Security	✓	✓	✗		
Output					
Display	✓ Showing details that user enter in signup process.	✓ In login process showing details that user entered	✓ Showing result and information when doing emotion recognition	✓ Showing Music Player , recommended list and user's favourite song list	✓ Showing recommend color and color map to select color to user
Speaker	✗	✗	✗	✓	✗

3.2.UI Design consideration and approaches

- **Search Functionality:**
  The app should have a search functionality that allows users to search songs and music to add their favourite list.

- **Clear and Simple:**
  The UI should be clear and simple to use, with easy-to-understand instructions and clear visual cues. The interface should be clear and straightforward, making it easy for users to identify what kind of situation that he facing

- **Consistency:**
  The interfaces should be made so that similar actions can be activated in the same way across all of them, and the related elements should all have a consistent style and look.

3.3.Design tools, techniques, templates

3.3.1. UI design tools

We use Adobe XD to create a UI design. Adobe XD is a powerful tool for designing and prototyping user interfaces. Its features include wireframing, designing, prototyping, and collaboration capabilities, making it suitable for creating interactive and visually appealing UI designs.

3.3.2. Techniques

User-Centered Design (UCD) - UCD involves understanding the needs and preferences of users through research and iterative design processes. It focuses on creating interfaces that are intuitive, efficient, and enjoyable to use.

Accessibility Design - Accessibility design ensures that UI designs are inclusive and accessible to users with disabilities. Techniques include using semantic HTML, providing alternative text for images, and ensuring keyboard navigation.

3.4.Input Design aspects

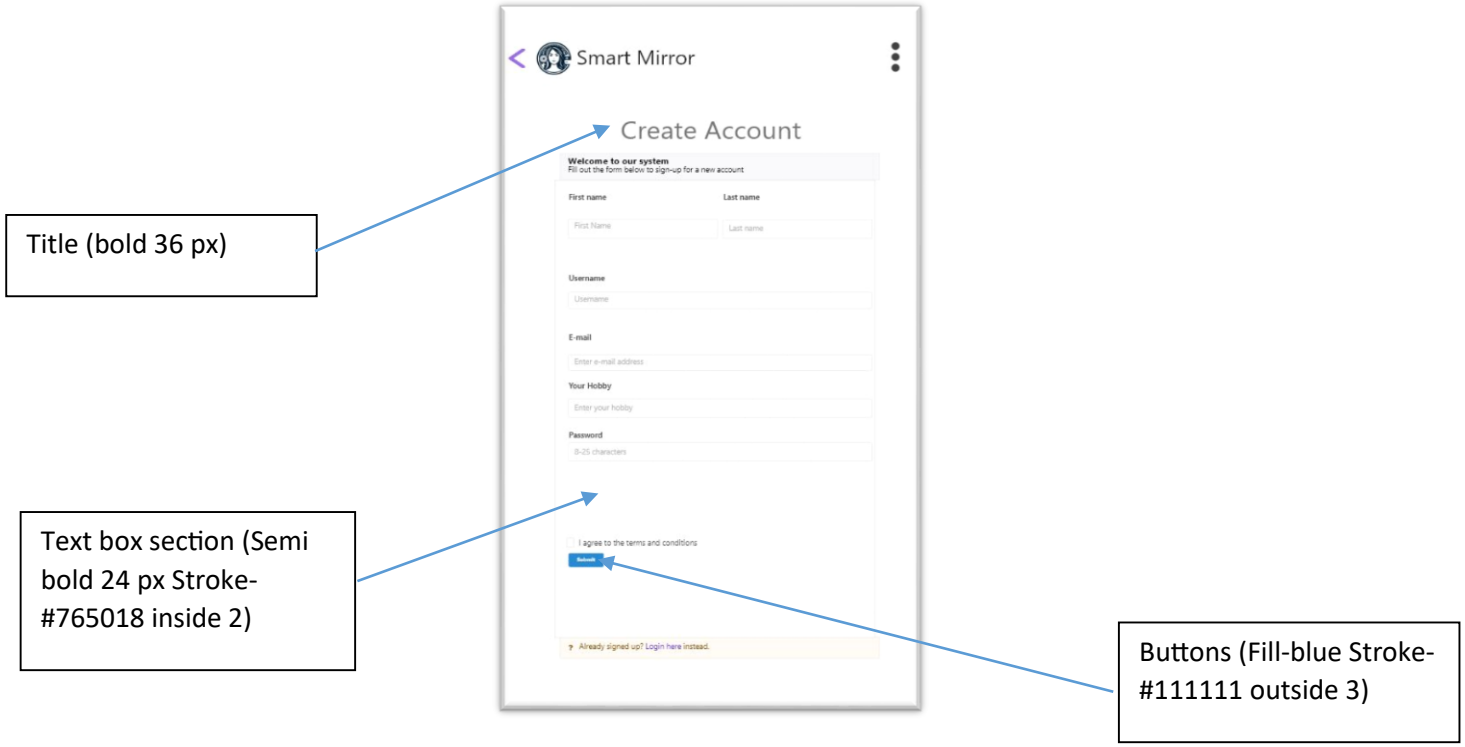


Figure 3.1: UI Element Format

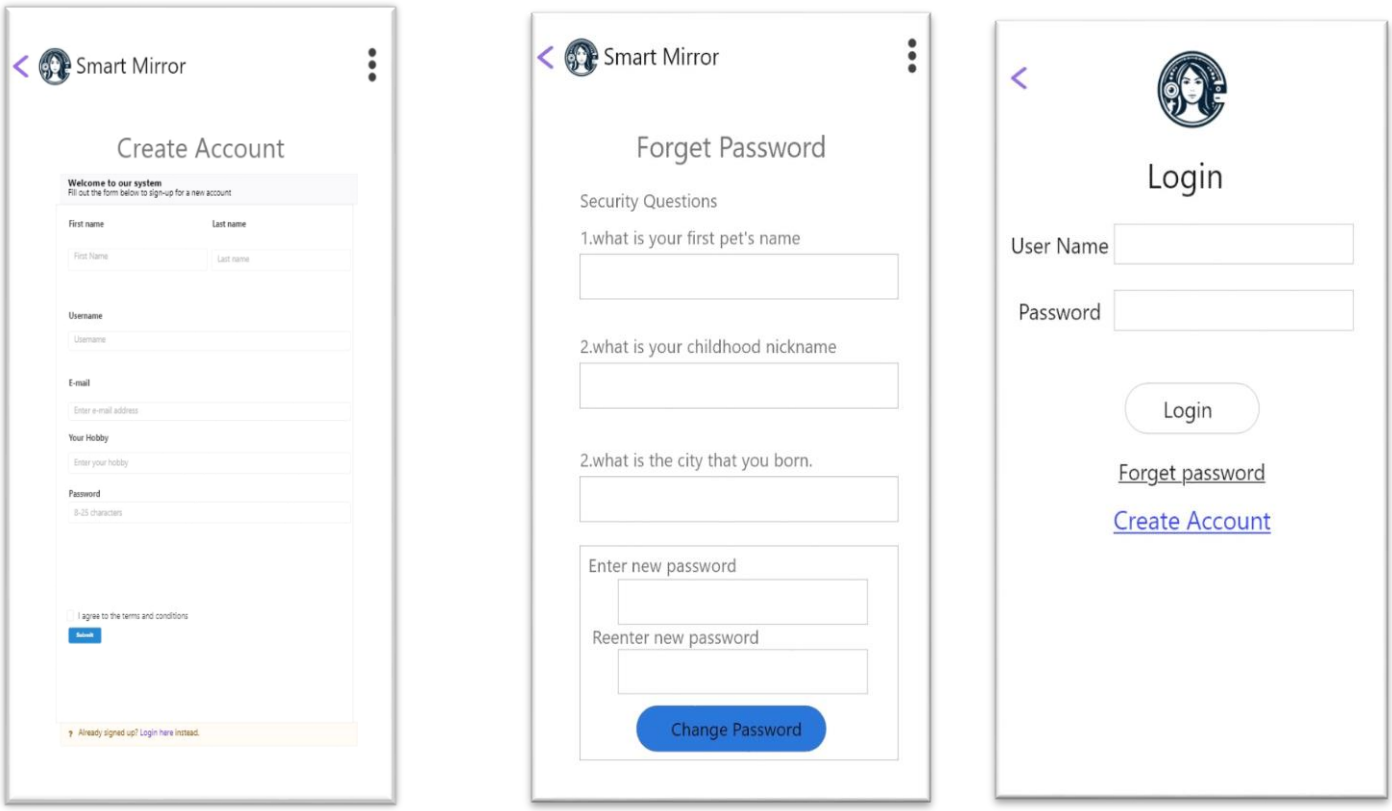


Figure 3.2: Input Design aspects

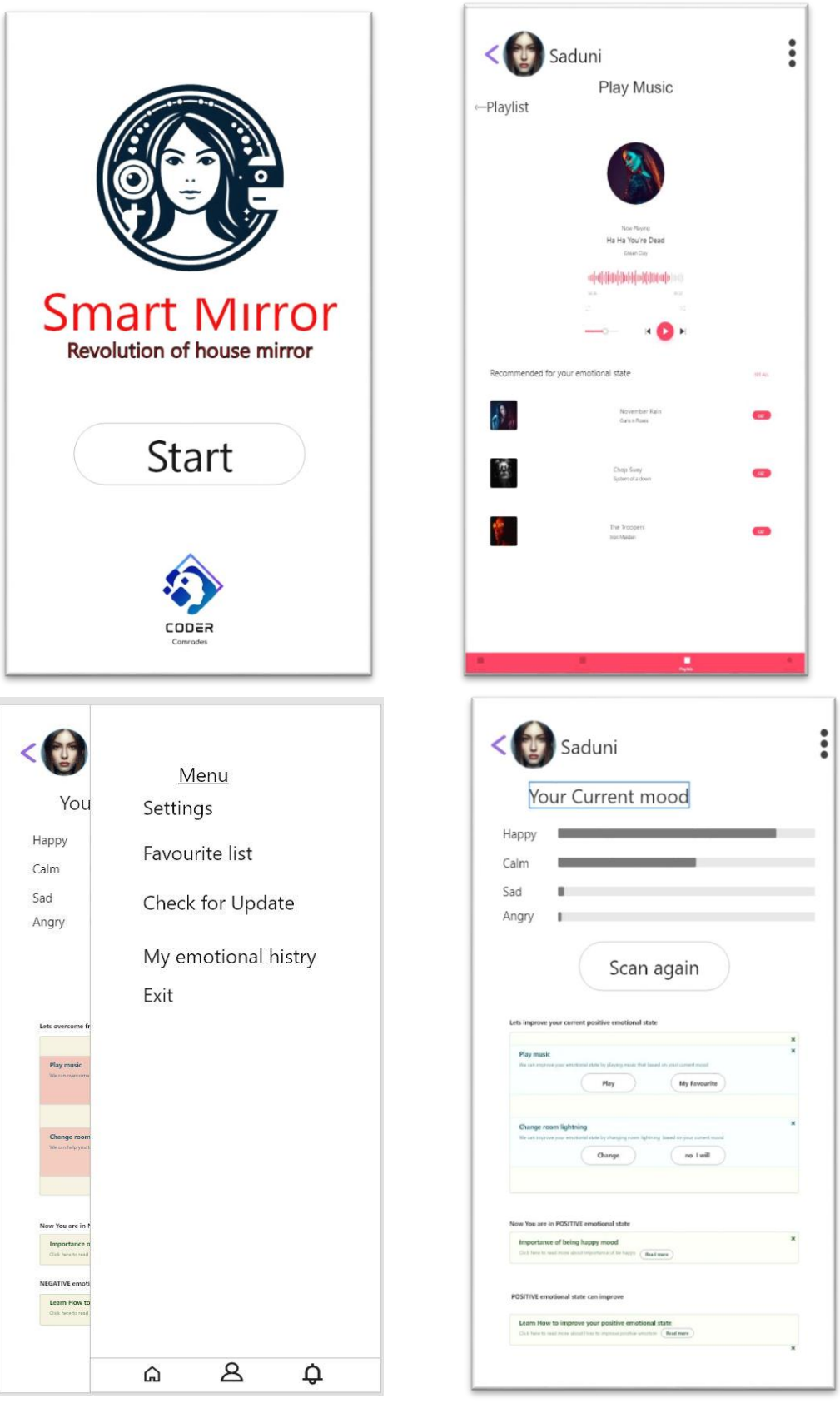


Figure 3.3: Input Design aspects

3.5.Output Design aspects

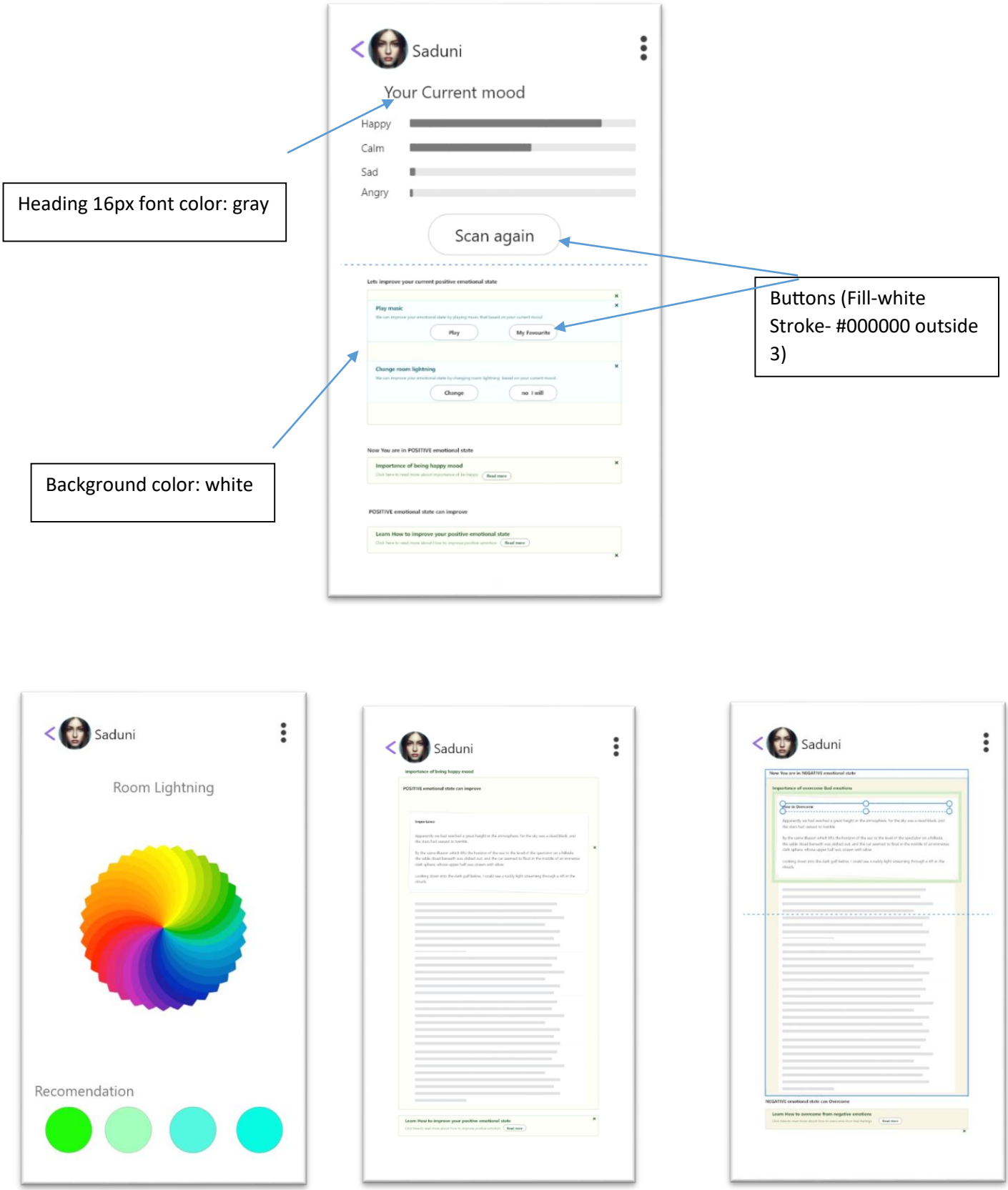


Figure 3.4: Output Design aspects

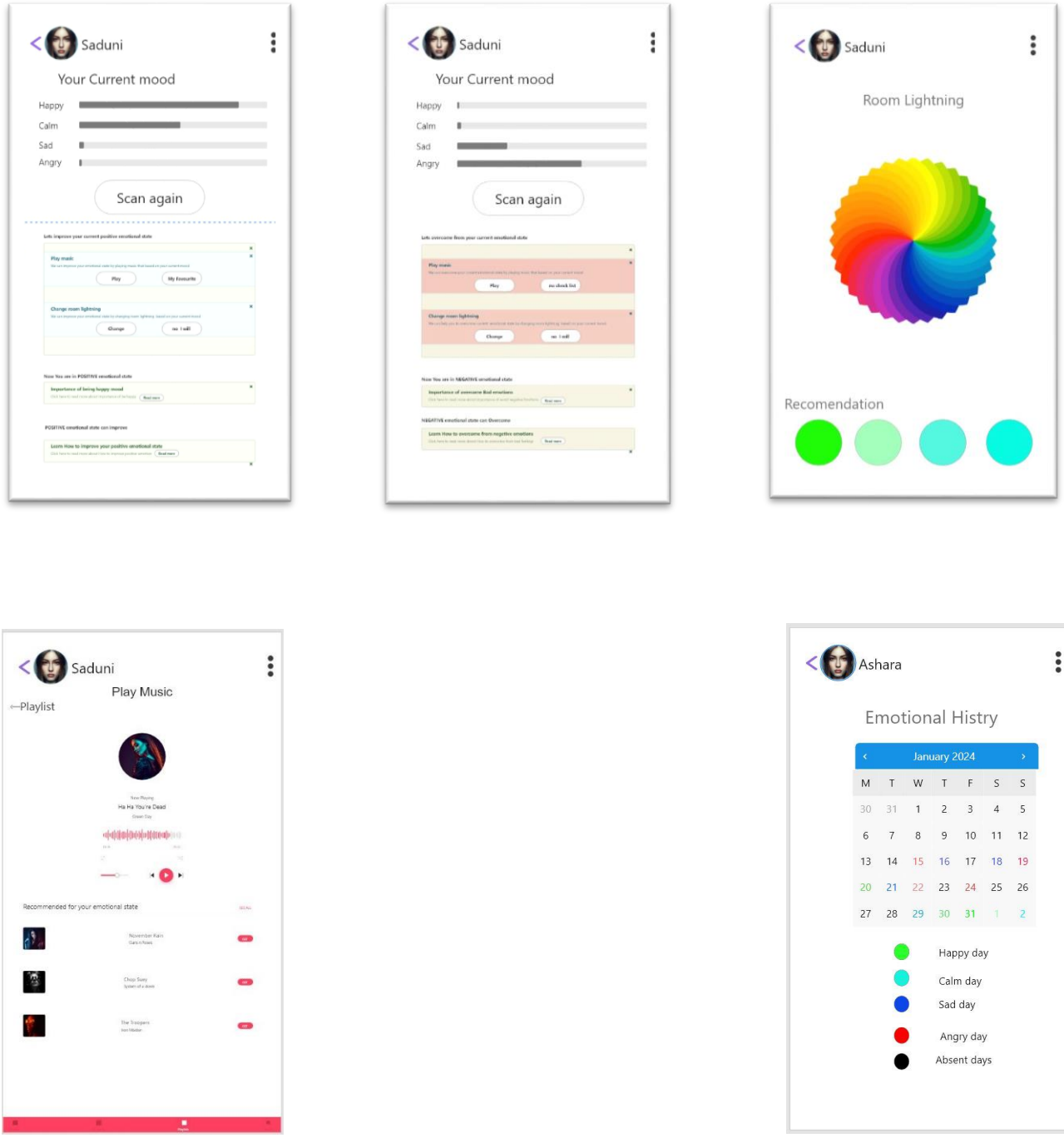


Figure 3.5: Output Design aspects

3.6.Dialogue design aspects

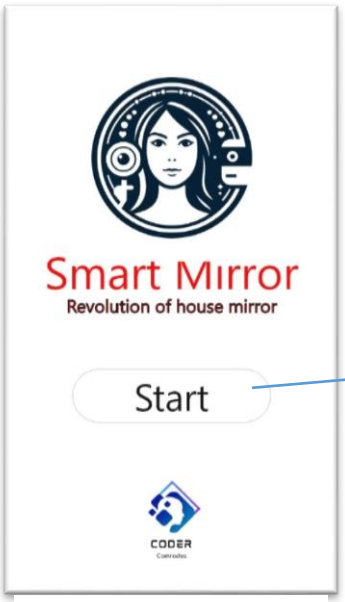


Figure 3.8: Front Interface

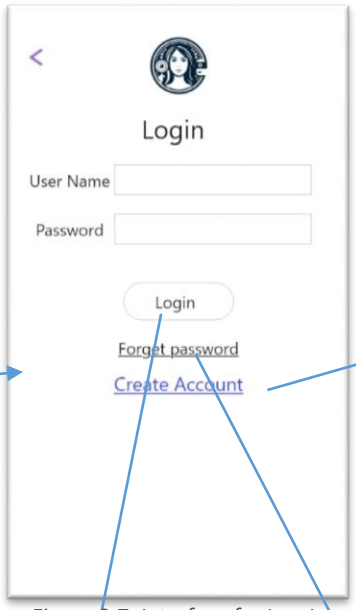


Figure 3.7: Interface for Log-In

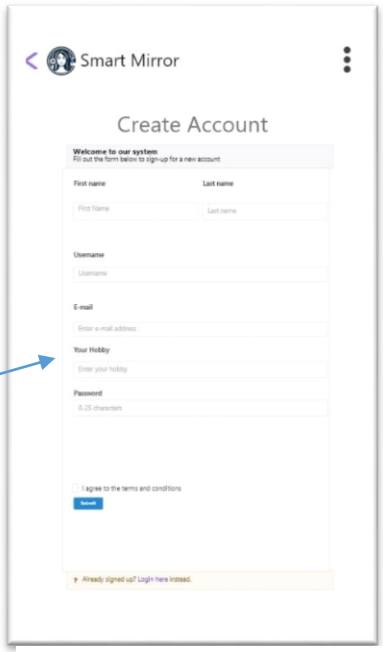


Figure 3.6: Interface for Create Account

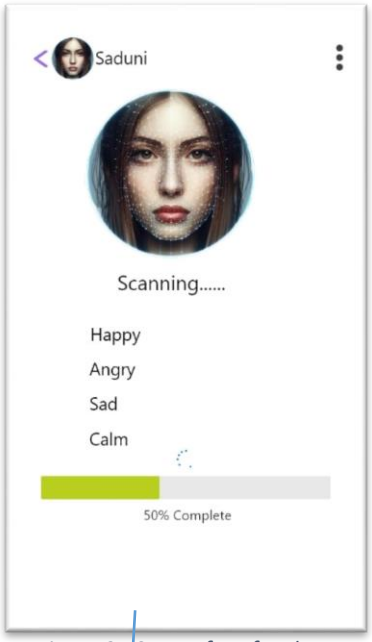


Figure 3.10: Interface for show emotion state

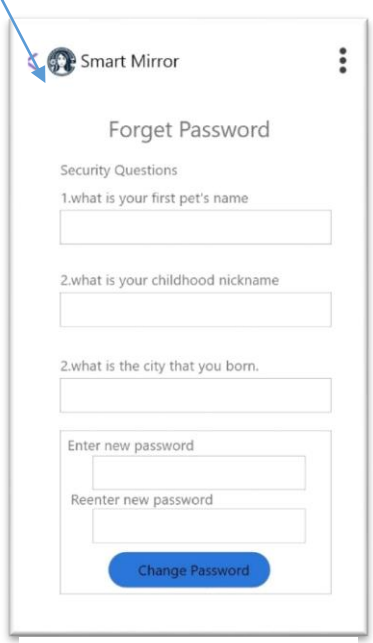


Figure 3.9: Interface for Forget Password

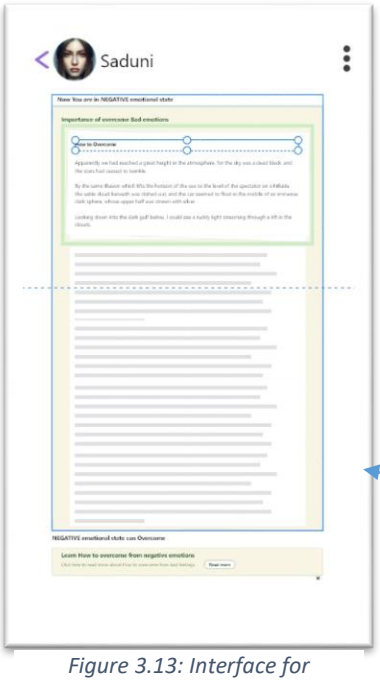


Figure 3.13: Interface for Recommendation

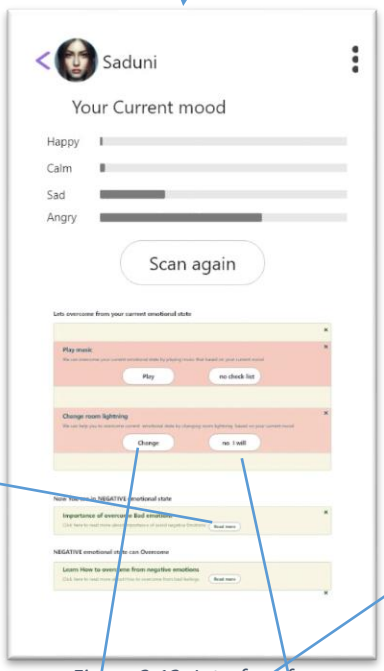


Figure 3.12: Interface for Dashboard

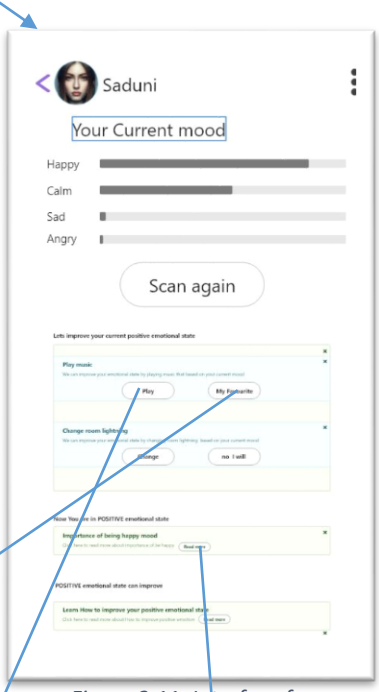


Figure 3.11: Interface for Dashboard

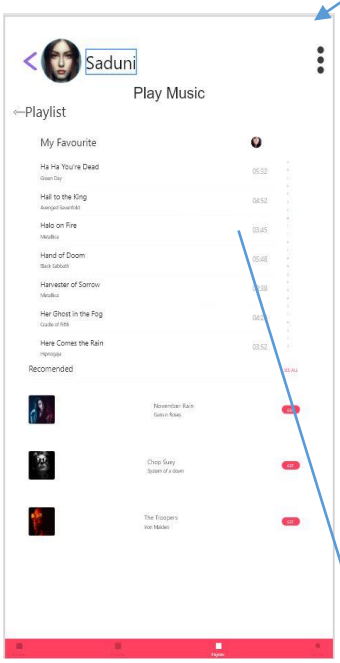


Figure 3.14: Interface for Favourite List

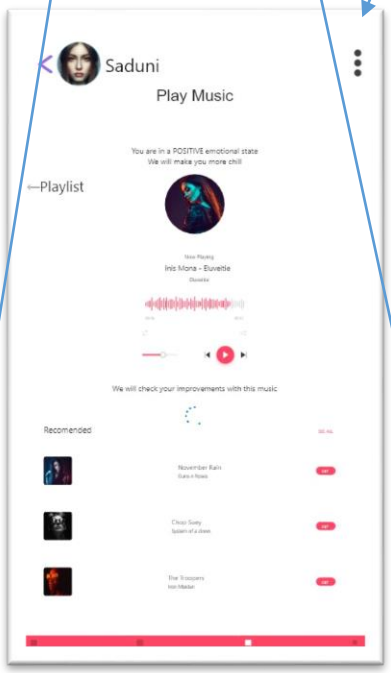


Figure 3.16: Interface for Playlist

Figure 3.15: Interface for Recommendation

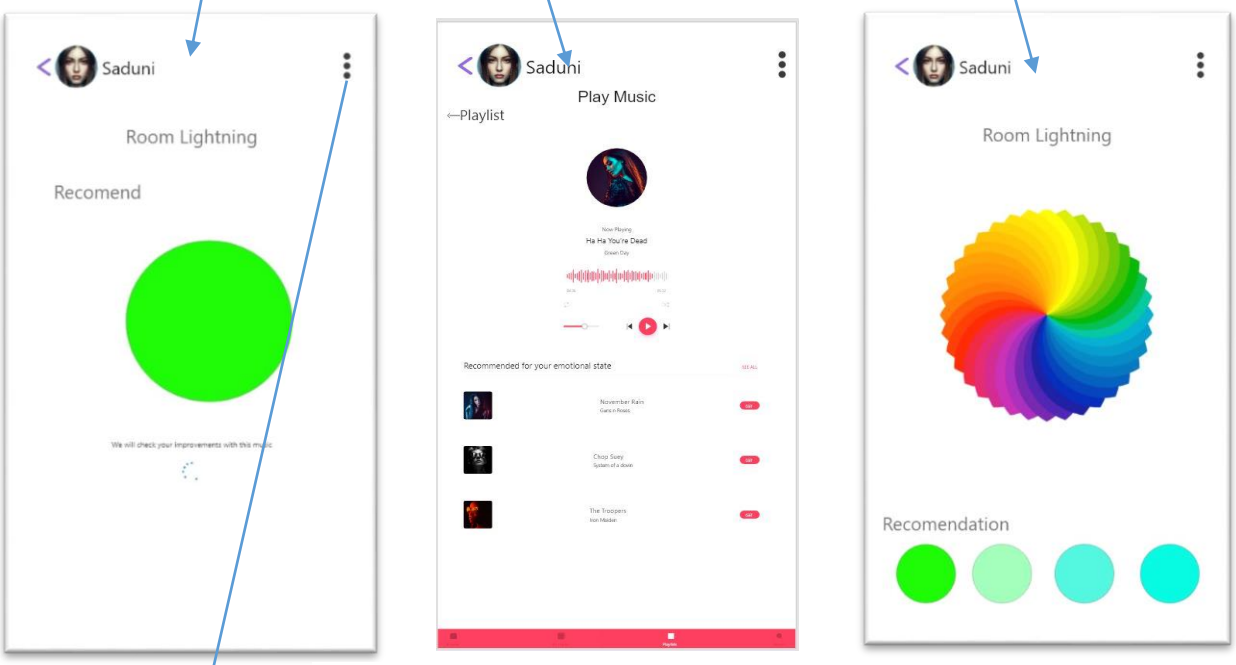


Figure 3.17: Music and Light System Adjustment Interfaces

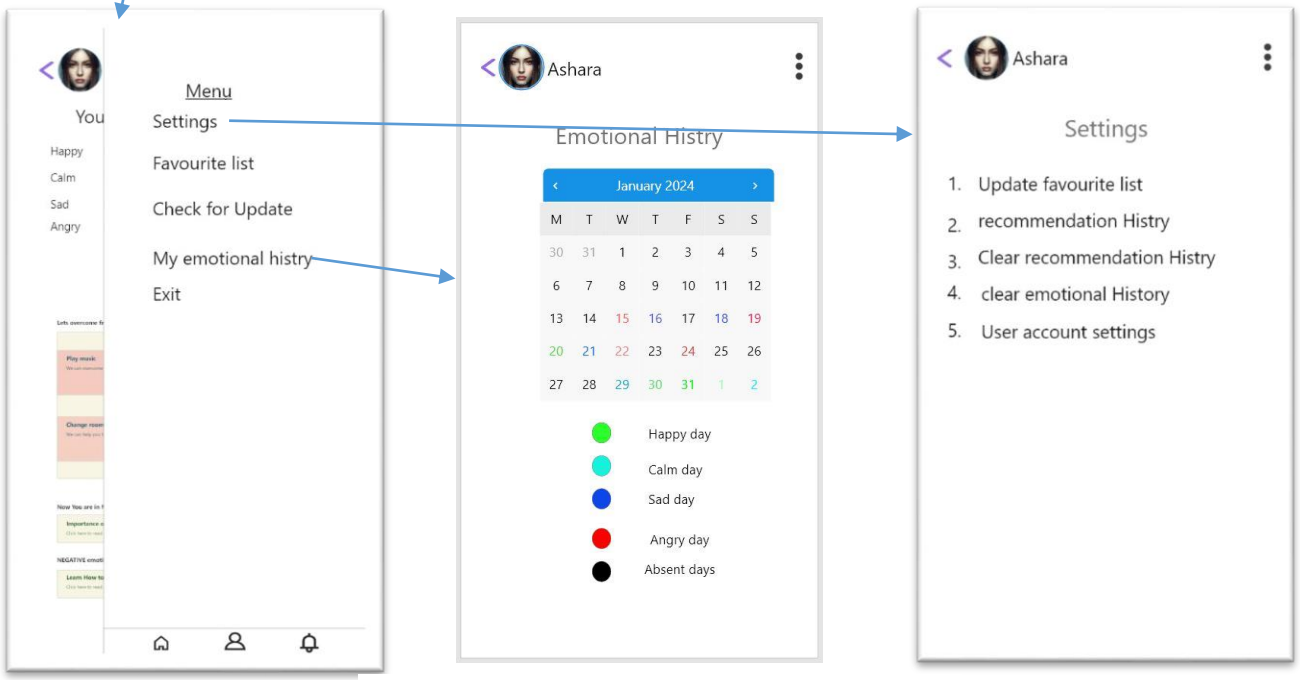


Figure 3.18: Emotion History and Setting Interfaces

3.7.Hosting/ installation environment

Cloud platforms are a great choice for running emotion recognition model for smart mirror project, especially if you prioritize processing speed over running the model directly on the Raspberry Pi. Here's why:

- Scalable Resources: Cloud platforms offer on-demand resources that can scale up or down based on processing needs. This ensures your model runs efficiently without compromising performance.

- Faster Training and Processing: Cloud platforms have powerful computing resources that can significantly accelerate the training process for emotion recognition model. This translates to faster and more accurate results when running the model on live data.

- Simplified Management: Cloud platforms offer tools and services specifically designed for machine learning model deployment and management. This simplifies the process of getting AI application up and running on the cloud.

Several cloud platforms can host our emotion recognition model for the smart mirror project:

- Amazon Web Services (AWS)
- Microsoft Azure
- Google Cloud Platform (GCP)

These platforms offer scalable resources, fast processing, and tools for model deployment and management, making them ideal for running your AI application.

CHAPTER 4 - DATA MANAGEMENT

Data management is the section that describes data requirements, design tools, techniques, conceptual database design, logical database design, schema refinement, physical database design, and security design.

4.1. Data requirements

The main objective of the data management section is to collect details about types of emotions, contents, and ambient light required according to a human's emotional state and store them in the database. User details are also collected through the interface, and based on personal recommendations of expertise. Captured facial expressions we get as inputs and compared them with the data that we stored in the database.

To get these results precisely, data requirements should be done very carefully. Here are the data requirements of the project.

- Collect user data for login purpose
- Emotion recognition data (emotion type, facial data)
- Information about recommended content (Type, link)
- Emotional associations of the content (uplifting music, calming videos)

4.2. Design tools, techniques

MySQL workbench is being used for the proposed system because it provides a reliable and scalable database solution for storing and managing relevant data due to following reasons,

- Data Storage - MySQL can store data efficiently, allowing the application to store various types of data in large amounts and accommodate an increasing number of users and devices as the system grows.

- Performance and Scalability -known for its performance and scalability capabilities. It can handle large amounts of data and concurrent user interactions efficiently, making it suitable for applications that require real-time tracking and analytics.

- Database Administration - MySQL Workbench provides tools for managing the database system itself. Therefore, can perform tasks such as creating and modifying database schemas, security settings, and monitoring database performance. These administration features are useful for ensuring the system's reliability and security.

- Visualizing Data -MySQL Workbench facilitate visualizing and analysing data enabling to create custom reports. This can be useful for displaying analytics and generating reports.

- Data Modelling - MySQL Workbench provides a user-friendly visual interface for designing and modelling databases.

4.3. Conceptual database design

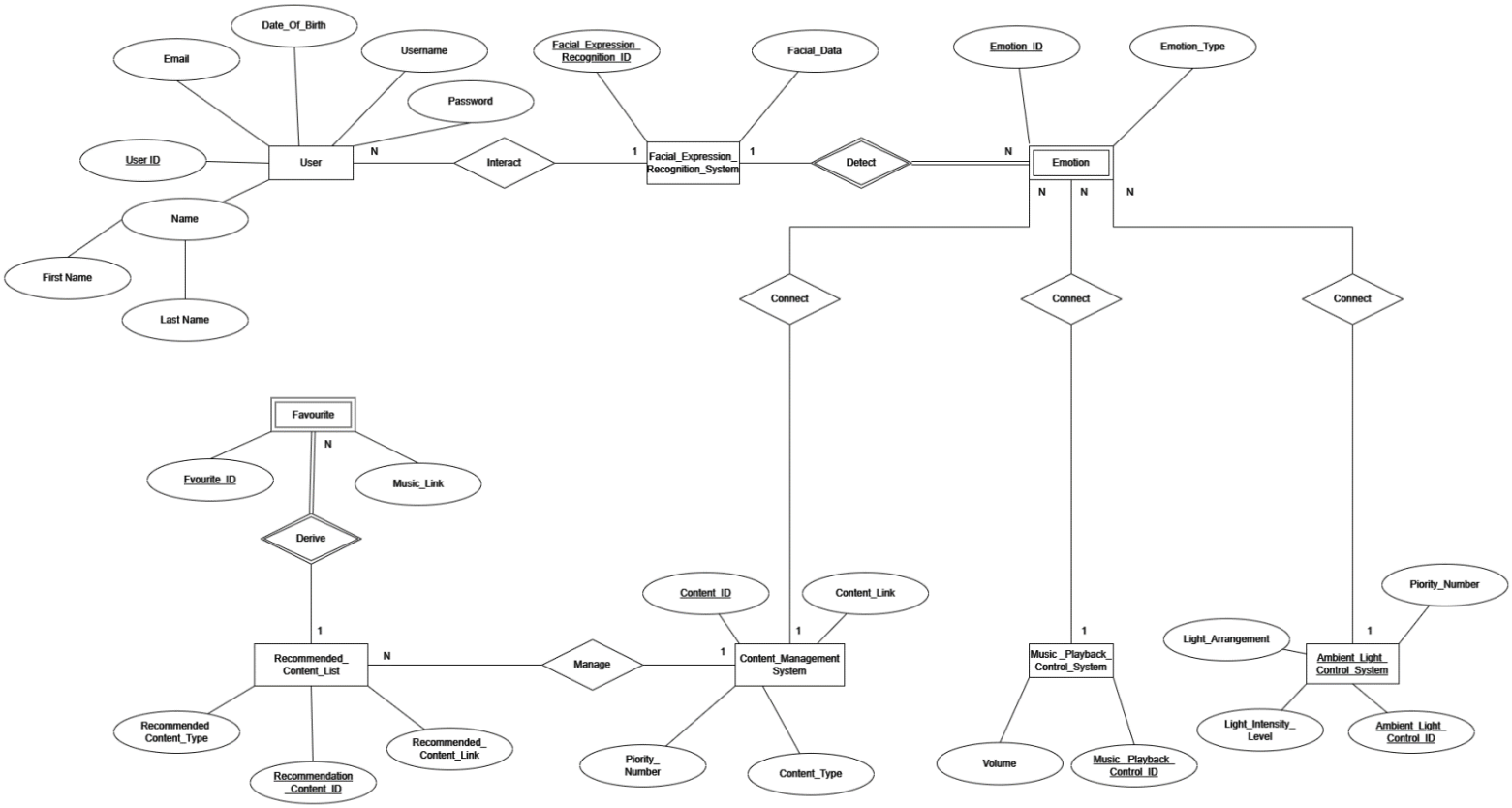


Figure 4.1: Conceptual database design

4.4. Logical database design

To convert the conceptual database design into a logical database design, need to refine the entities, attributes, and relationships and represent them in a structured manner.

Figure 4.2: Logical database design

4.5. Schema refinement

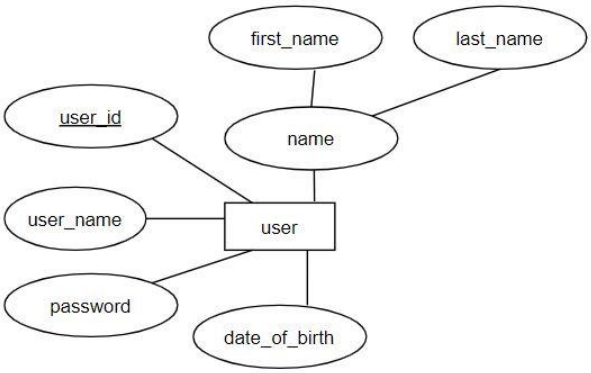
During the refinement of the schema, tables are refined. It is the final phase before moving on to physical design and tweaking with normal workloads. This technique is mostly used to decrease anomalies and redundant information. It will organize the database's contents. Because it lacks dependencies, a normalized database is more like the relational model.

Techniques for normalization include,

- Determine functional dependencies
- Avoiding anomalies
- Normalizing characteristics with multiple values

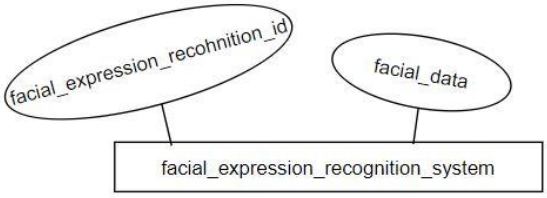
as seen by the database's physical layout.

4.5.1. Table Creation



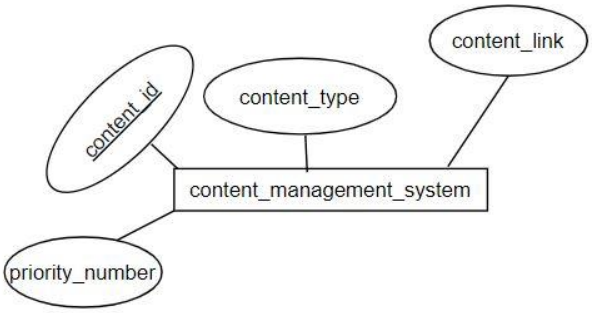
user

user_id	first_name	last_name	user_name	email	date_of_birth	password



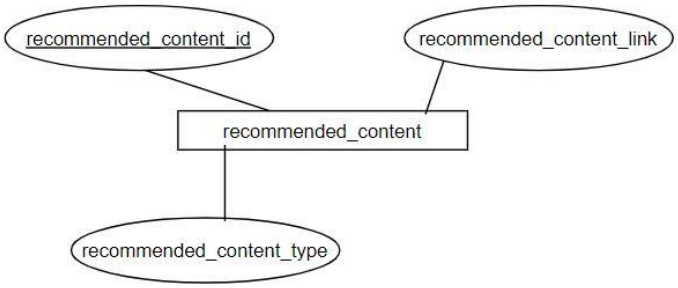
facial_expression_recognition_system

facial_expression_recognition _id	facial_data



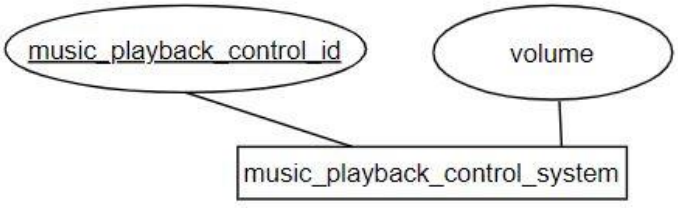
content_management_system

content_id	content _link	content_type	priority_number



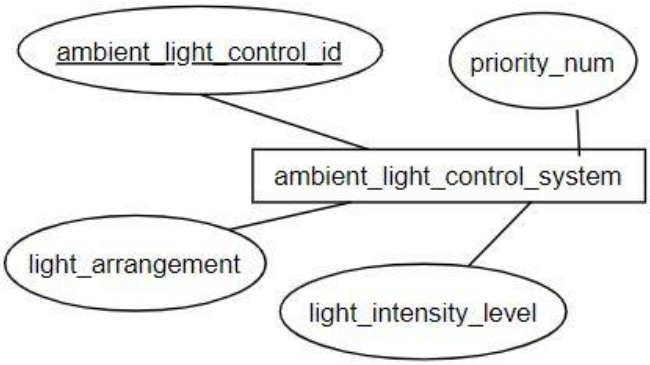
recommended_content

recommended_content _id	recommended_content _type	recommended_content _link



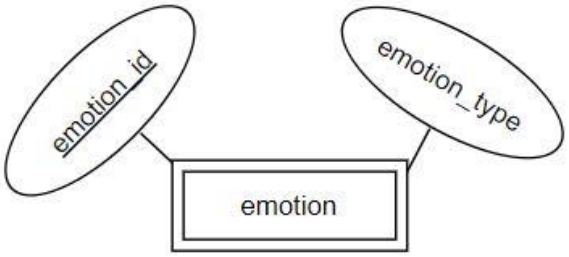
music_playback_control_system

music _ playback_control_id	volume



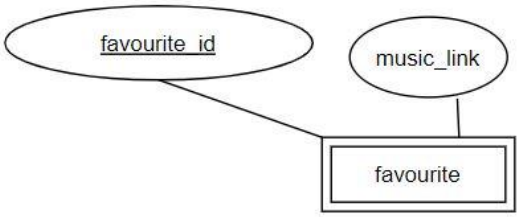
ambient_light_control_system

ambient_light_control_id	priority_number	light_intensity_level	light_arrangement



emotion

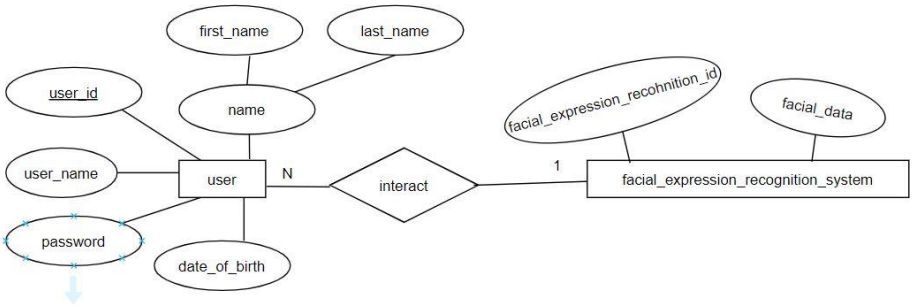
emotion_id	emotion_type



favourite

favourite_id	music_link

4.5.2. Define Relationships

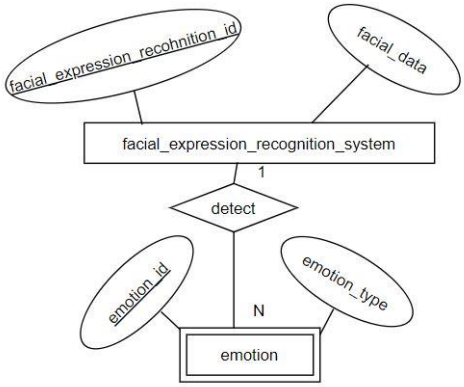


user

user_id	first_name	last_name	user_name	email	date_of_birth	password	facial_expression_recognition _id

facial_expression_recognition_system

facial_expression_recognition _id	facial_data

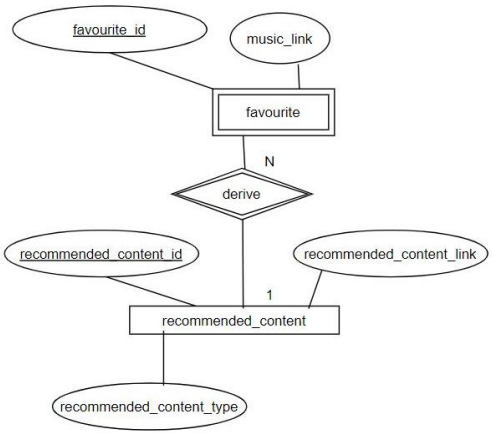


facial_expression_recognition_system

facial_expression_recognition _id	facial_data

emotion

emotion_id	emotion_type	facial_expression_recognition _id

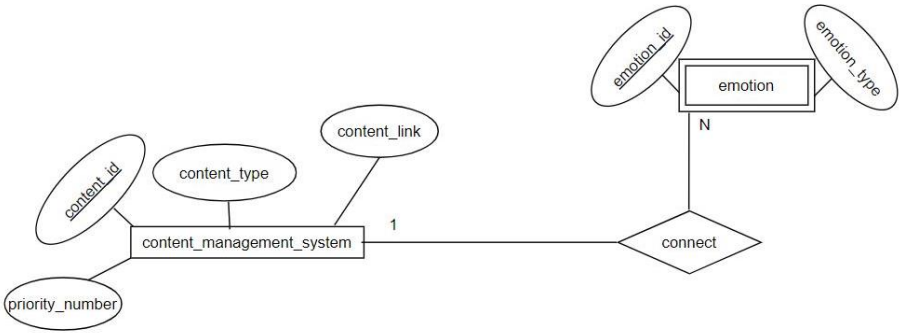


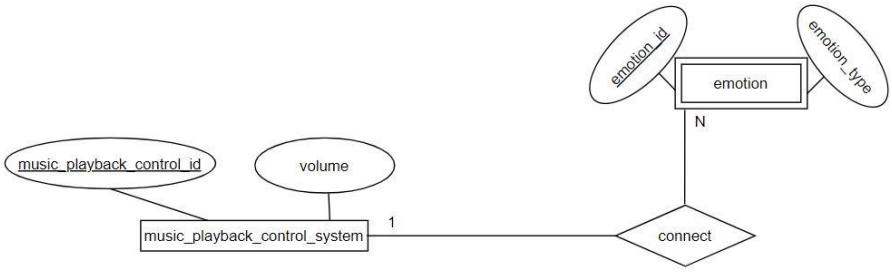
favourite

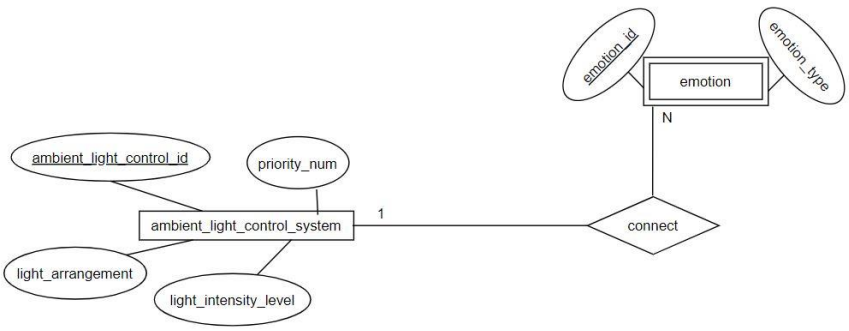
favourite_id	music_link	recommended_content _id

recommended_content

recommended_content _id	recommended_content _type	recommended_content _link







emotion

emotion_id	emotion_type	facial_expression_recognition _id	content _id	music_playback_control_id	ambient_light_control_id

content_management_system

content_id	content _link	content_type	priority_number

music_playback_control_system

music _ playback_control_id	volume

ambient_light_control_system

ambient_light_control_id	priority_number	light_intensity_level	light_arrangement

4.5.3. Schema

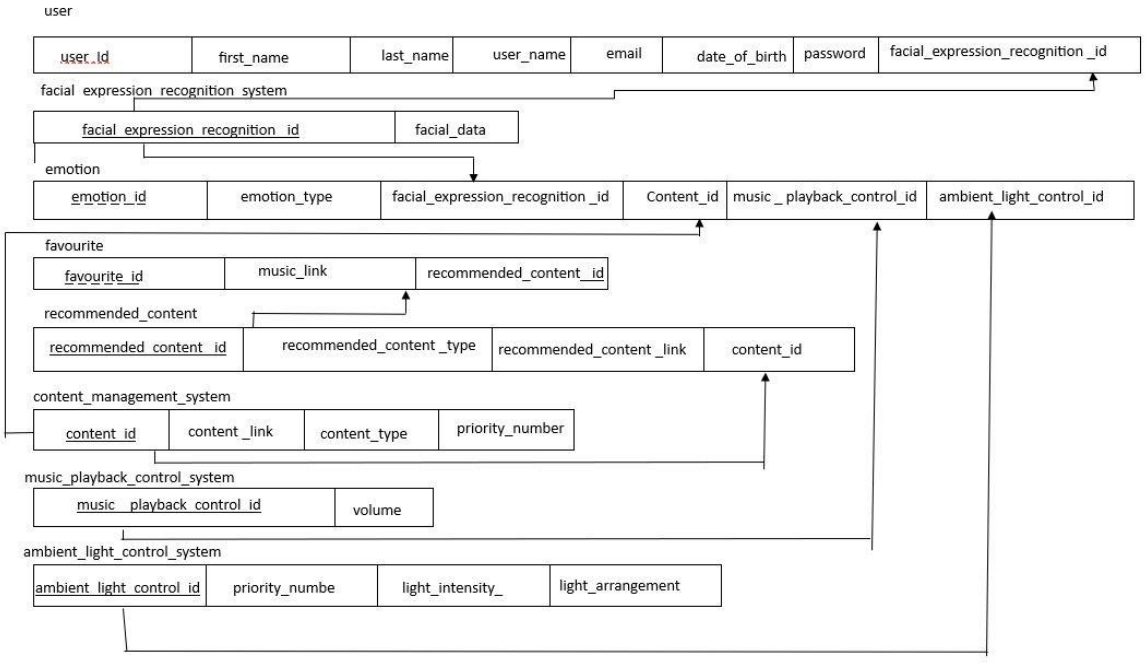


Figure 4.3:Schema Refinement

4.6. Physical database design

The physical design of a database is how a database is implemented into a working system. Many physical database attributes are dependent on the details and semantics of the target database management system (DBMS), but logical design can be carried out regardless of the final database platform.

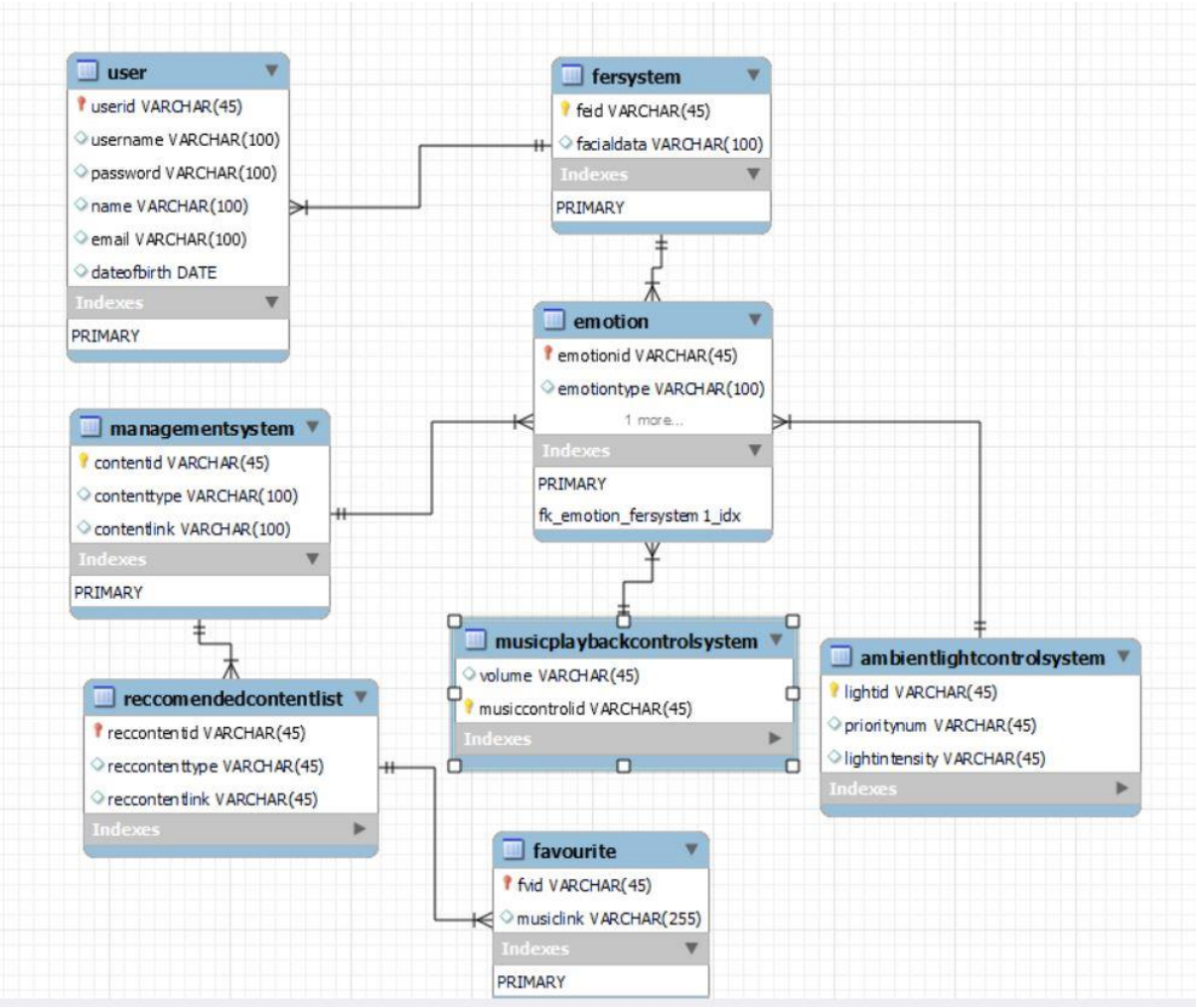


Figure 4.4: Physical database design

```
CREATE TABLE `db1`.`user` (
  `UserId` VARCHAR(45) NOT NULL,
  `username` VARCHAR(255) NULL,
  `name` VARCHAR(255) NULL,
  `email` VARCHAR(45) NULL,
  `dateofbirth` DATETIME NULL,
  `password` CHAR(45) NULL,
  `feid` VARCHAR(45) NULL,
```

```
  PRIMARY KEY (`UserId`));

CREATE TABLE `db1`.`fersystem` (
  `FeId` VARCHAR(45) NOT NULL,
  `facialdata` VARCHAR(255) NULL,
  INDEX `FeId_idx` (`FeId` ASC) VISIBLE,
  CONSTRAINT `FeId`
    FOREIGN KEY (`FeId`)
    REFERENCES `db1`.`user` (`UserId`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION);

CREATE TABLE `db1`.`emotion` (
  `EmotionId` VARCHAR(45) NOT NULL,
  `emotiontype` VARCHAR(255) NULL,
  PRIMARY KEY (`EmotionId`));

CREATE TABLE `db1`.`favourite` (
  `FvId` VARCHAR(45) NOT NULL,
  `musiclink` VARCHAR(255) NULL,
  PRIMARY KEY (`FvId`));

CREATE TABLE `db1`.`reccomendedcontentlist` (
  `RecContentId` VARCHAR(45) NOT NULL,
  `reccontenttype` VARCHAR(255) NULL,
  `reccontentlink` VARCHAR(255) NULL,
  PRIMARY KEY (`RecContentId`));

ALTER TABLE `db1`.`emotion`
ADD CONSTRAINT `fe_id`
  FOREIGN KEY (`EmotionId`)
  REFERENCES `db1`.`fersystem` (`FeId`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION;
```

```
ALTER TABLE `db1`.`favourite`
ADD COLUMN `reccontentid` VARCHAR(45) NULL AFTER `musiclink`,
ADD INDEX `reccontent_id_idx` (`reccontentid` ASC) VISIBLE;

ALTER TABLE `db1`.`favourite`
ADD CONSTRAINT `reccontent_id`
  FOREIGN KEY (`reccontentid`)
  REFERENCES `db1`.`reccomendedcontentlist` (`RecContentId`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION;

CREATE TABLE `db1`.`contentmanagementsystem` (
  `ContentId` VARCHAR(45) NOT NULL,
  `contentlist` VARCHAR(255) NULL,
  `contenttype` VARCHAR(255) NULL,
  `prioritynum` VARCHAR(255) NULL,
  PRIMARY KEY (`ContentId`));

ALTER TABLE `db1`.`contentmanagementsystem`
ADD CONSTRAINT `contentid`
  FOREIGN KEY (`ContentId`)
  REFERENCES `db1`.`emotion` (`EmotionId`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION;

CREATE TABLE `db1`.`musicplaybackcontrolsystem` (
  `musiccontrolsystem` VARCHAR(45) NOT NULL,
  `volume` VARCHAR(35) NULL,
  PRIMARY KEY (`musiccontrolsystem`));
```

```
ALTER TABLE `db1`.`musicplaybackcontrolsystem`
CHANGE COLUMN `musiccontrolsystem` `musiccontrolid` VARCHAR(45) NOT NULL ;
ALTER TABLE `db1`.`musicplaybackcontrolsystem`
ADD CONSTRAINT `musiccontrolid`
  FOREIGN KEY (`musiccontrolid`)
  REFERENCES `db1`.`emotion` (`EmotionId`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION;

CREATE TABLE `db1`.`ambientlightcontrolsystem` (
  `LightId` VARCHAR(45) NOT NULL,
  `lightarrangement` VARCHAR(255) NULL,
  `lightintensity` VARCHAR(255) NULL,
  `prioritynum` VARCHAR(35) NULL,
  PRIMARY KEY (`LightId`));

ALTER TABLE `db1`.`ambientlightcontrolsystem`
ADD CONSTRAINT `lightid`
  FOREIGN KEY (`LightId`)
  REFERENCES `db1`.`emotion` (`EmotionId`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION;
```

4.7. Security design

This outlines the security design considerations for the IoT based smart mirror with Human Emotion Recognition system. The system stores user data, potentially including facial data and emotional recognition results, and interacts with content.

- **User Authentication and Authorization**

  Define user roles and access controls to restrict unauthorized access to user data and system functionalities.

- **Data Security**

  Store user data, particularly facial data securely. Consider using encryption at rest and in transit.
  Implement appropriate access controls to user data based on roles and permissions.
  Regularly review and update data retention policies to ensure user data is not stored unnecessarily.

- **Content Security**

  Validate and sanitize content uploaded or accessed by users.

- **System Security**

  Regularly conduct security audits to identify and address vulnerabilities.

- **Privacy**

  Obtain informed consent from users regarding the collection, storage, and use of their facial data and emotion recognition results.
  Provide users with control over their data, including the ability to view, modify, or delete it.
  Be transparent about how user data is used and ensure it aligns with user expectations.

CHAPTER 5 - HARDWARE DESIGN

5.1. Hardware components and detailed design

The Smart Mirror is an IoT device that uses a Raspberry Pi 3 to process complex data and run AI models. It features a high-resolution LED monitor, a webcam for facial expressions, speakers for audio feedback, and a Wi-Fi module for connectivity. A proximity sensor detects user presence. Data is securely stored on internal or cloud-based solutions, adhering to strict privacy protocols.

5.1.1. Define the requirements

The Smart Mirror requires a Raspberry Pi 3 processor, a high-resolution LED monitor, a webcam for facial expression capture, speakers for audio feedback, a Wi-Fi module for internet connection, a proximity sensor for activation/deactivation, and internal or cloud-based data storage. The operating system should be compatible with Raspberry Pi 3, and the AI framework should run specific facial expression recognition models. Additional requirements include a power supply, a two-way mirror, and specific software applications for desired functionality.

5.1.2. Characteristics of the custom peripherals in circuit

The image shows a circuit with custom peripherals, including a Raspberry Pi 3 Model B+, an LED display, a two-way mirror, a HD camera, a proximity sensor, a Wi-Fi router, and a power supply. The Raspberry Pi is a single-board computer with a Broadcom BCM2837 CPU, 1GB of RAM, and various I/O ports. The LED display can be any LED or OLED panel, with resolution, size, and power consumption varying depending on the model. The two-way mirror allows a display to be placed behind the mirror, allowing users to see the display through the mirror. The HD camera can capture video or images. The proximity sensor detects proximity, and the Wi-Fi router provides Wi-Fi connectivity to the Raspberry Pi and other devices. The power supply is necessary to meet the requirements of the components.

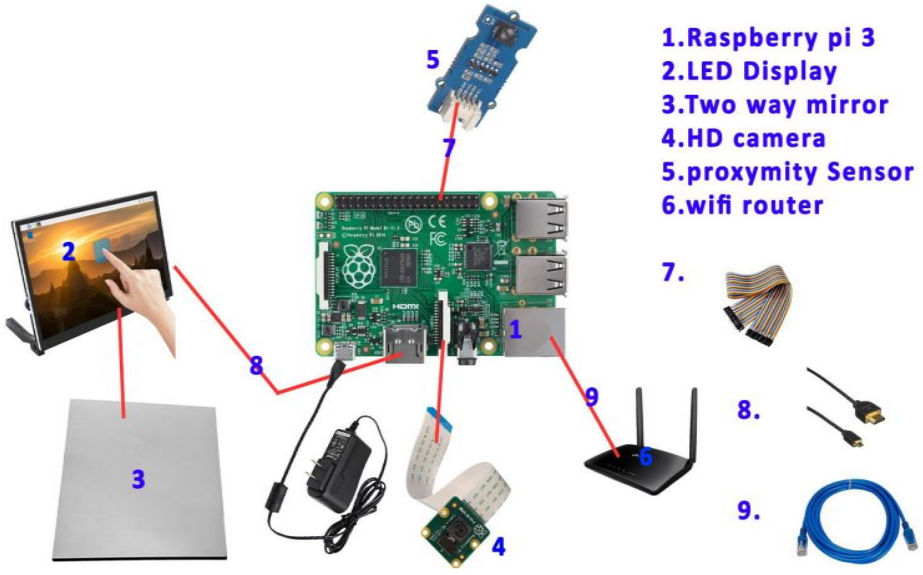


Figure 5.1: Hardware Design

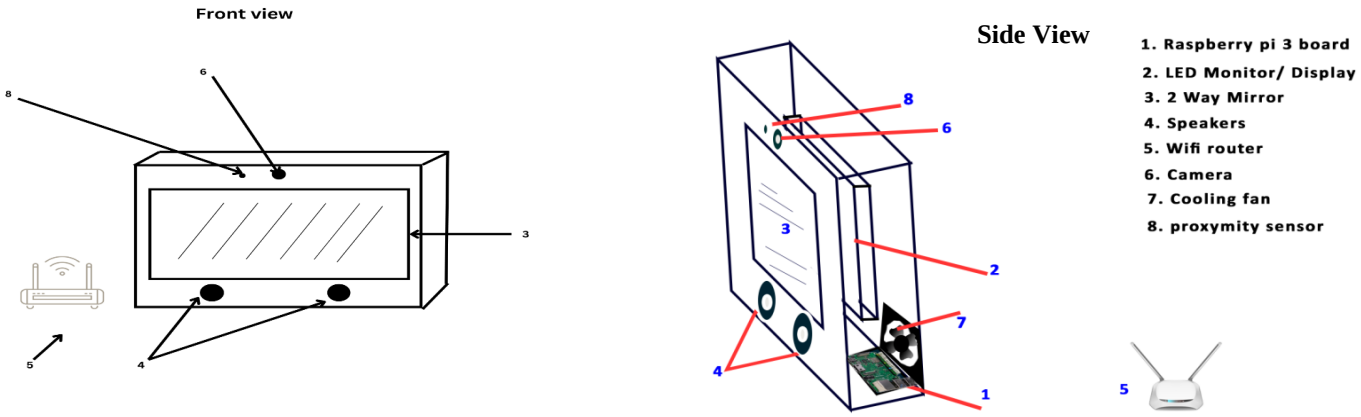


Figure 5.2: Hardware Design

5.1.3. Components of Hardware Design

- **Raspberry PI 3**
  The Raspberry Pi 3 is the hub of an emotional detection system using a smart mirror. It extracts facial features and emotions from sensors using software libraries like OpenCV or TensorFlow. The Pi displays relevant information, including emotional feedback and interactive elements, and may connect to other devices.

- **LED Monitor**
  The LED monitor uses a smart mirror to display user interaction data, emotional feedback, personalized content, interactive elements, and a reflective surface, allowing users to see their reflection.

- **Wi-Fi module**
  The Wi-Fi module is crucial for emotional detection, facilitating communication, data transfer, and machine learning analysis. It enables interaction with other smart devices, remote configuration, and data visualization.

- **Proximity sensor**
  The proximity sensor is crucial in an emotional detection system, detecting the user's physical presence and distance to the mirror, triggering activation and personalized responses. It contextualizes emotions, improving analysis accuracy and sensitivity. Data fusion enhances understanding.

5.2. Hardware component communication and interfaces

The Smart Mirror uses various hardware components for communication, with the Raspberry Pi acting as the central hub. It connects to the LED display via HDMI, the camera via USB, and potentially other peripherals through its GPIO pins. A Wi-Fi module is connected for internet connectivity. Sensors like the proximity sensor communicate digitally using protocols like I2C or SPI. Common interfaces for display, camera, and peripherals are HDMI, USB, and GPIO.

CHAPTER 6 - RESEARCH DESIGN

The majority of the research design chapter relates to the literature review survey that was done on the proposed system. The literature study part provides an overview of the technology and methodology in previous studies related to the project proposal. A summary of the project's non-functional and high-level implementation components is also given in this chapter..

6.1. Study objectives based literature review

The objective of this research is to design and implement an IoT-based Smart Mirror that can recognize and respond to human emotions, thereby enhancing user interaction and experience. Drawing from the literature, including [1] and [2], the study will review existing smart mirror technologies and emotion recognition systems, focusing on the integration of IoT for real-time processing and feedback. The objectives of our proposed system are,

1. To develop a Smart Mirror prototype using IoT to recognize human emotions using facial expressions.
2. To train a machine learning model for recognizing and analyzing human emotions such as happy, sad, angry, fear, and neutral emotions using the user's facial expressions in real-time.
3. To design a user-friendly smart mirror using web technologies with IoT devices such as an embedded display.
4. To develop a content management system for displaying and delivering emotion-driven content such as music, videos, and information according to that emotional state and playing music.
5. To implement the smart lighting control system for ambient lighting control that aligns with recognizing emotional state.

By reviewing relevant literature on,

1) To develop a Smart Mirror prototype using IoT to recognize human emotions using facial expressions.

Tater et al. (2020) [1] introduced an “IoT-based Assistive Smart Mirror with Human Emotion Recognition System”. Their system utilized IoT technology to detect emotions through facial expressions. Bianco et al. (2021) [2] developed a “Smart Mirror for Emotion Monitoring in Home Environments”. They use sensors and technologies to detect emotions. Lakshmi et al. (2018) [12] developed an “IoT-Based Smart Mirror Using Raspberry Pi”. This study focuses on creating an IoT-based smart mirror using Raspberry Pi.

Raspberry Pi serves as the hardware platform for building the mirror. IoT architectures allow seamless integration of sensors, data processing, and cloud services. IoT can achieve high accuracy in emotion detection (up to 92% in some cases). Various sensors can be used for emotion recognition, including cameras for facial expression analysis, heart rate monitors, skin conductance sensors, and temperature sensors. We hope to use a camera sensor with Raspberry Pi to detect emotions using facial expressions.

2) **To train a machine learning model for recognizing and analyzing human emotions such as happy, sad, angry, fear, and neutral emotions using the user's facial expressions in real-time.**

Facial emotion recognition (FER) is an emerging field in pattern recognition. Non-verbal communication plays a significant role in daily life, with facial expressions contributing around 55% to 93% of overall communication. Researchers have explored both conventional machine learning and deep learning approaches for FER. Computer vision techniques, AI, and image processing are widely used to enhance automated facial recognition systems for security and healthcare applications. Face detection serves as the initial step in locating or detecting faces in videos or images during the FER process. Various machine learning algorithms have been employed for emotion recognition, including k-NN, decision trees, random forests, convolutional neural networks, and multilayer perceptron. To improve performance, researchers evaluate different machine learning classifiers on benchmark emotion datasets. We hope to use a machine learning model for recognizing and analyzing human emotions such as happy, sad, angry, fear, and neutral emotions using the user's facial expressions in real time.

3) **To design a user-friendly smart mirror using web technologies with IoT devices such as an embedded display.**

We will research user interface design principles for smart mirrors such as interactive design, Visual output, user-centered interaction, feedback mechanisms, and personalization, ensuring an intuitive and user-friendly experience on an embedded display. We will use web technologies to design the interface integration with IoT devices like cameras, sensors, and routers.

4) **To develop a content management system for displaying and delivering emotion-driven content such as music, videos, and information according to that emotional state and playing music.**

Understanding the connection between human emotions and content preferences is crucial for designing an effective content management system. Research has shown that emotional states influence user behavior, including content consumption. For example, positive emotions (such as happiness) may lead users to prefer uplifting music or motivational videos, while negative emotions (such as sadness) may prompt them to seek comforting content. The relationship between emotions and content preferences can inform the design of personalized recommendation algorithms for our smart mirror. We hope to train recommendation models using historical data.

5) **To implement the smart lighting control system for ambient lighting control that aligns with recognizing emotional state.**

Smart lighting systems have evolved beyond energy-saving features. They offer opportunities to enhance user comfort and well-being. Research in this field has predominantly focused on non-residential environments, but the potential impact on residential spaces remains largely unexplored. To create a holistic smart mirror experience, integrating emotion recognition technology with smart lighting can enhance user satisfaction and mental comfort.

Lighting significantly influences our mood, cognitive performance, and overall well-being. According to our research, colors play a big role in emotional state and glare-free lighting enhances comfort and reduces eye strain. Smart lighting control can minimize glare by adjusting intensity and direction. Smart lighting systems incorporate sensors and connectivity to adapt lighting conditions dynamically. Integrating emotion recognition allows real-time adjustment of lighting based on the user's emotional state

6.2. Formalizing high-level implementation components

The implementation will involve a Raspberry Pi as the core computing unit, as suggested by [4], along with a camera module, display, and sensors. The software will include operating system for Raspberry Pi, emotion recognition algorithms, machine learning libraries for model development as discussed by [6] and [7], web technologies for user interface creation, content management system for recommendation delivery, and ambient lighting control software. And the Smart Mirror system will consist communication protocols for data transfer between the camera, emotion recognition software, user interface, content management system, and ambient lighting system.

6.3. Data extractions, sample design, test data sets, training data sets

6.3.1. Data Extractions

- **Emotion Data Sources -** Gather data related to human emotions using image frames with individuals displaying various facial expressions representing emotions like happy, sad, angry, fear, and neutral.

- **Public Dataset -** Use existing datasets containing labeled emotional expressions. Resources like KDEF (The Karolinska Directed Emotional Faces): - https://www.kaggle.com/datasets/msambare/fer2013

6.3.2. Sample Design

- **Participant Selection -** Decide on the demographics of participants like age, and gender. Just as the actors' characteristics influence the performance, the demographics of participants can significantly impact the data collected.

- **Sample Size -** Determine an appropriate sample size based on statistical power and significance. Because of too few participants, and model might not generalize well to unseen faces, and with too many, data collection can become expensive and time-consuming.

- **Randomization -** Randomly assign participants to different emotional scenarios or conditions.

6.3.3. Test Data Sets

- **Purpose -** Test data sets are used to evaluate the performance of the emotion recognition system.

- **Emotion Labels -** Each sample should be labeled with the corresponding emotion such as happy, sad, angry, fear, or neutral. This allows us to compare the model's predicted emotion with the actual emotion, providing valuable insights into its accuracy.

- **Metrics -** Use metrics like accuracy, precision, recall, and F1-score to assess system performance. These metrics quantify the model's performance, telling how often it correctly identifies emotions, how good it is at avoiding false positives, and how well it balances these factors.

6.3.4. Training Data Sets

- **Purpose -** Training data sets are used to train the machine learning model.

- **Balancing -** Ensure the training data has a balanced representation of different emotions. If the dataset is not balanced, it can lead to the model favoring frequently occurring emotions.

- **Feature Extraction -** Extract relevant facial features from the images.

- **Machine Learning Models** - Train models using labeled training data.

6.4. Non-functional aspects

6.4.1. Product Requirements

- **Personalization and Memory -** The system should have the ability to learn and remember the user's emotional responses and preferred adjustments. Over time, the system should be able to make more accurate and personalized recommendations for the user based on their emotions and preferences.

- **User Interface -** The user interface should be intuitive and user-friendly, allowing the user to manually override the recommendations made by the system if desired.

- **Reliable and Accurate Emotion Recognition -** The system should be capable of accurately recognizing and interpreting emotions to provide effective recommendations and adjustments.

- **Accessibility Features -** The smart mirror should be designed to accommodate users of all abilities, including those with user interface customization. This may include features like adjust font size and style, colour contrast.

- **Product Support and Updates -** The manufacturer should provide ongoing support, updates, and maintenance for the smart mirror to ensure its continued functionality and relevance.

- **Integration with Smart Home Systems -** The smart mirror should be compatible with other smart home devices and systems to create a seamless and integrated experience for the user.

6.4.2. Organizational Requirements

- **Implementation requirement:**
  IoT based smart mirror with human emotion recognition system requires a Raspberry Pi hardware setup, a two-way mirror, a webcam, and sensors for processing power, display, and data collection. Emotion recognition algorithms are used to analyse facial expressions and identify emotions like happiness, sadness, anger, fear and neutral. An interactive interface allows users to access information, reminders, and control features based on their emotional state. This combination of hardware, software, and user

interface enables the smart mirror to display information and improve the user's emotional well-being.

- **Standard requirement:**
  The proposed IoT based Smart Mirror with Human Emotion Recognition System uses IoT technology to recognize human emotions. This allows the mirror to display information and provide personalized responses, such as mood-lifting music or ambient lighting adjustments, based on the user's emotional state. This approach aims to enhance overall well-being by improving emotional understanding and offering personalized support within the familiar environment, enhancing comfort for users.

6.4.3. External Requirements

- **Interoperability Requirement -** Enable the smart mirror to effortlessly connect with and manage Bluetooth-enabled devices.
- **Ethical Requirement -** Safeguard user privacy and ensure ethical data handling practices. Obtain explicit user consent for emotion data collection, implement secure and anonymous storage of emotion data.
- **Privacy Requirement -** Comply with data protection regulations and protect user privacy. Adherence to GDPR (General Data Protection Regulation) standards, Secure storage, and anonymization of emotional data.
- **Safety Requirement -** Ensure the system does not cause harm or discomfort to users. No reported incidents of harm or discomfort caused by the system.

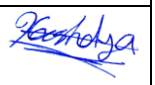
6.5. Proposed validation methods and measurements

Validation is used in projects to see whether the project has met its requirements and specifications or the desired outcomes. By implementing these validation methods and measurements, we can assess the effectiveness, usability, and performance of our Smart Mirror System. This validation process helps ensure that the system meets user expectations, drives engagement, and provides seamless and satisfactory recommendations.

- **Accuracy and Precision Metrics -** Measure the accuracy and precision of the recommendation system by comparing the recommended content with the user preferences using metrics like F1 score, confusion matrix, and precision/recall.
- **User Studies -** User studies with recruited participants will assess the usability, user experience, and effectiveness of personalized recommendations based on emotional state. Questionnaires can be used to gather user feedback.
- **System Performance Testing -** Response times, processing power usage, and network latency will be measured to ensure the system meets performance expectations.

CHAPTER 7 - APPROVAL

Signature of the team members:

Registration Number	Index Number	Name	Signature
ICT/19/20/106	5042	W.G.D.J. Shashipriya	
ICT/19/20/060	4998	I.K.G.A.M. Karunarathna	
ICT/19/20/068	5006	N.K. Lokusooriya	
ICT/19/20/081	5019	U.W.G.R.P. Perera	
ICT/19/20/124	5059	Y.M.A.H.B. Yapa	

Date: 03-05-2024

Approval of the supervisor

I agree / ~~disagree~~ with the scope stipulated in this Software Design Specification

Name: Mrs.M.M.A.Nisansala

Department/Organization: Department of Computing

University/Organization: Rajarata University of Sri Lanka

Signature: 

REFERENCES

[1] L. Tater, S. Pranjale, S. Lade, A. B. Nimbalkar, and P. N. Mahalle, “IoT based Assistive Smart Mirror with Human Emotion Recognition System,” International Journal of Engineering Research and Technology, vol. V9, no. 02, Feb. 2020, doi: 10.17577/ijertv9is020127.

[2] S. Bianco et al., “A smart mirror for emotion monitoring in home environments,” Sensors, vol. 21, no. 22, pp. 7453, Nov. 2021, doi: 10.3390/s21227453.

[3] H. Yu, J. H. Bae, J. Choi, and H. S. Kim, “LUX: Smart Mirror with Sentiment Analysis for Mental Comfort,” Sensors, vol. 21, no. 9, pp. 3092, Apr. 2021, doi: 10.3390/s21093092.

[4] B.A. Rani, R. Vinay, C. Darshan, H.S. Shashank, H.B. Jain, “Design of Smart Mirror Based on Raspberry Pi”, International Journal of Research in Engineering, Science and Management, vol. 3, no. 5, May 2020.

[5] P.S. Tondewad, H. Parate, P. Awalkonde, A. Mule, “Smart Mirror Based on Raspberry Pi”, vol. 6, no. 2, June 2019.

[6] A.R. Khan, “Facial Emotion Recognition Using Conventional Machine Learning and Deep Learning Methods”, May 2022.

[7] T. Nguyen, I. Bass, M. Li, I.K. Sethi, “Investigation of combining SVM and decision tree for emotion classification”, Seventh IEEE International Symposium on Multimedia (ISM’05), 2005, doi:10.1109/ism.2005.72.

[8] S. Bianco, “A smart mirror for emotion monitoring in Home Environments”, Sensors, vol. 21, no. 2, pp. 7453, 2021, doi:10.3390/s21227453.

[9] Y. Andreu, “Wize Mirror - a smart, multisensory cardio-metabolic risk monitoring system”, Computer Vision and Image Understanding, 148, pp. 3–22, 2016, doi: 10.1016/j.cviu.2016.03.018.

[10] P. Henriquez, “Mirror Mirror on the wall... an unobtrusive intelligent multisensory mirror for well-being status self-assessment and visualization”, IEEE Transactions on Multimedia, vol. 19, no. 7, pp. 1467–1481, 2017, doi:10.1109/tmm.2017.2666545.

[11] B. Vesnicer, “Face recognition using simplified probabilistic linear discriminant analysis”, International Journal of Advanced Robotic Systems, vol. 9, no. 5, pp. 180, 2012, doi:10.5772/52258.

[12] N.M. Lakshmi, M. S. Chandana, P. Ishwarya, N. Meena, R. R. Patil, “IoT based Smart Mirror using Raspberry Pi”, International Journal of Engineering Research and Technology, vol. 6, no. 13, Apr. 2018, doi: 10.17577/IJERTCONV6IS13131.

[13] G. Boateng, “Towards real-time multimodal emotion recognition among couples”, Proceedings of the 2020 International Conference on Multimodal Interaction, pp. 748-753, Oct. 2020, doi:10.1145/3382507.3421154.

[14] R.J. Davidson, P. Ekman, C.D. Saron, J. A. Senulis, W.V. Friesen, "Approach-withdrawal and cerebral asymmetry: Emotional expression and brain physiology. I", Journal of personality and social psychology, vol. 58, no. 2, pp. 330-341, 1990, doi: 10.1037/0022-3514.58.2.330.

[15] D. Hagemann, E. Naumann, A. Lurken, G. Becker, S. Maier, D. Bartussek, “EEG asymmetry, dispositional mood and personality”, Personality and Individual Differences, vol. 27, no. 3, pp. 541–568, Nov. 1998, doi:10.1016/s0191-8869(98)00263-3.

Appendix 01

Team members and Contributions

Chapter	Name	Index
Introduction	**All members**	**All members**
Architecture design		
UI design	W.G.D.J. Shashipriya I.K.G.A.M. Karunarathna U.W.G.R.P. Perera	5042 4998 5019
Data management	W.G.D.J. Shashipriya N.K. Lokusooriya Y.M.A.H.B. Yapa	5042 5006 5059
Hardware design	I.K.G.A.M. Karunarathna U.W.G.R.P. Perera	4998 5019
Research design	N.K. Lokusooriya Y.M.A.H.B. Yapa	5006 5059