# Customer Emotion Detection

Team members:

Pavan Sai Chandra, Anjali Kumari, Harini Thirunavukkarasan, Prasann Kumar Gupta, Sameera Tazeen, Khushi Agarwal, Hardik Siloiya, Dhruv Joshi, Ajibesh Mohapatra, Samriddhi Agarwal

# Problem Statement

Build a Chat application that can be accessed through any device and aid the customer support executive by showing the emotional state of the customer by using intelligence. The application should also predict sample responses for the support executive and the customer will be able to like or dislike the response received from the support executive whose feedback should be used to better train the model.

Assumptions are as follows:

- One customer support executive works with only one customer at a time.
- If customer logs off by mistake then they cannot connect to previous chat window and will be directed to new chat window.
- If the customer does not provide any input for more than 5 minutes then their chat will be ended.

# Problem Analysis

❖ Scenario 1:

In this scenario a customer has lost their card and is guided by the service executive through the follow up process of reporting the loss of the card and applying for a new card.
The emotions of the customer as detected by the model are shown in the conversation right next to every text from the customer.

**Customer:** I have lost my card. What should I do?

I am sorry for that! You don't need to worry, we have got everything covered. Please provide the card number and card holder's name

**Executive**

Emotion: worried & anxious;
Suggestions provided by model: I'm sorry that you've had to deal with [issue];

**Customer:** ACXXXXXXXX02392, Samriddhi Agarwal

When did you lose your card and had the last transaction on that card?

**Executive**

Emotion: Neutral;
Suggestions provided by model: Please bear with us for a while;

**Customer:** I don't remember. Why are you asking so many questions? Just block my card.

I understand, I just wanted to make sure nobody has used the card since the time it was lost. It seems there hasn't been any transaction in the last 4 days. I would immediately block it.

**Executive**

Emotion: angry & frustrated;
Suggestions provided by model: Sorry about that. We understand your concern, we're looking for the best possible solution;

**Customer:** When will I get a new card?

Within two business days

**Executive**

Emotion: Neutral;
Suggestions provided by model: Please wait, let me check that.

**Customer:** Last time it took more than a week. I should rather open an account in some other bank.

Cont'd:
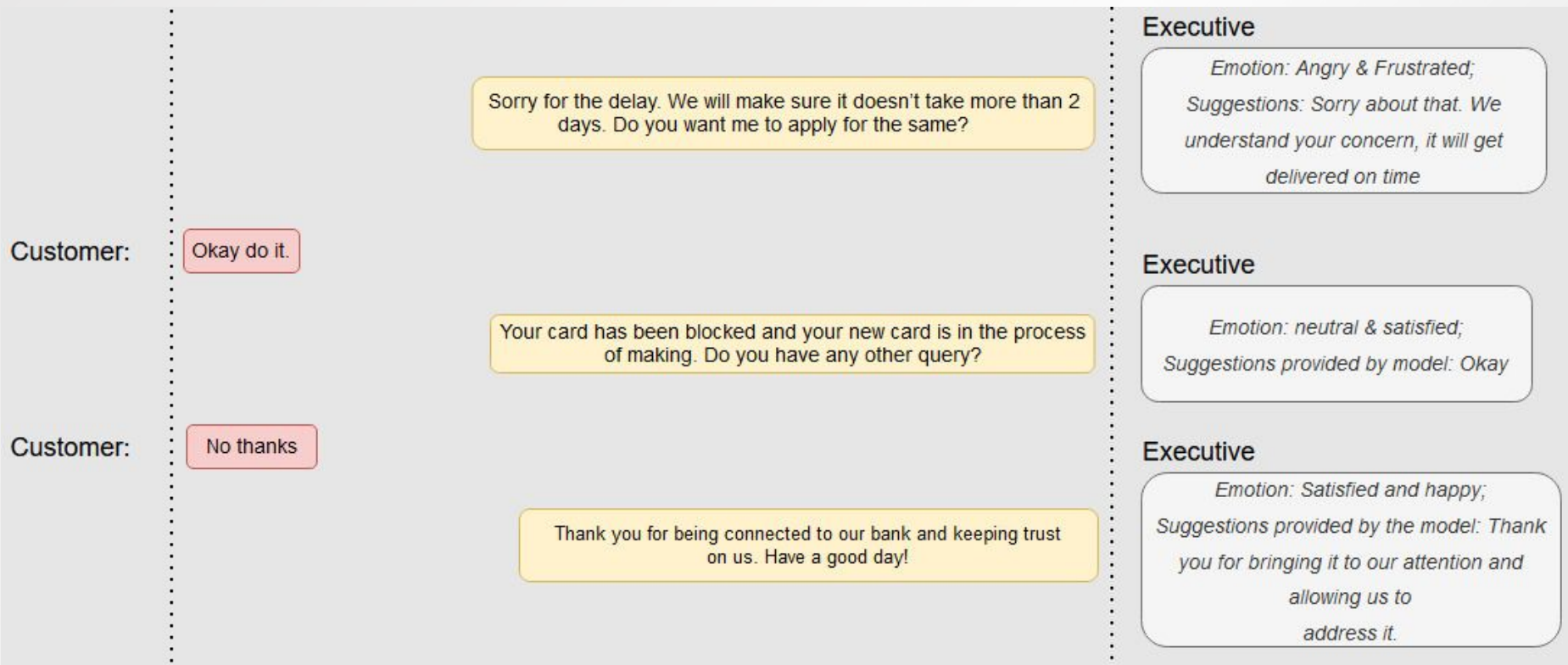
# Problem Analysis

❖ Scenario 2:

In this scenario money was debited from customer's account even after the transaction failed and, has not been refunded back into the account.
The emotions of the customer as detected by the model are shown in the conversation right next to every text from the customer.

# WELLS FARGO

**Customer:** Hey, I made a transaction and money got deducted from my account, but the transaction failed. The money did not get refunded back into my account.

Sorry for the inconvenience faced. We will certainly look into the issue and do our best.

To help you with this situation, we require a set of information. Please provide the transaction date and transaction Id.

**Executive**

*Emotion: worried & anxious;*

*Suggestions provided by model: I'm sorry that you've had to deal with [issue];*

**Customer:** Sure. Transaction date: 5th April 2022, Transaction id:1234567890

Please hold on for 2 mins. We will surely get back to you with the details.

**Executive**

*Emotion: Neutral;*

*Suggestions provided by model: Thank you for contacting Wells Fargo Customer Care Support*

**Customer:** Okay

Don't worry, your money will be refunded back in less than 48 working hours.

Thank you for bearing with us through this incident. If there's anything else I can help you with, please let me know.

**Executive**

*Emotion: Neutral;*

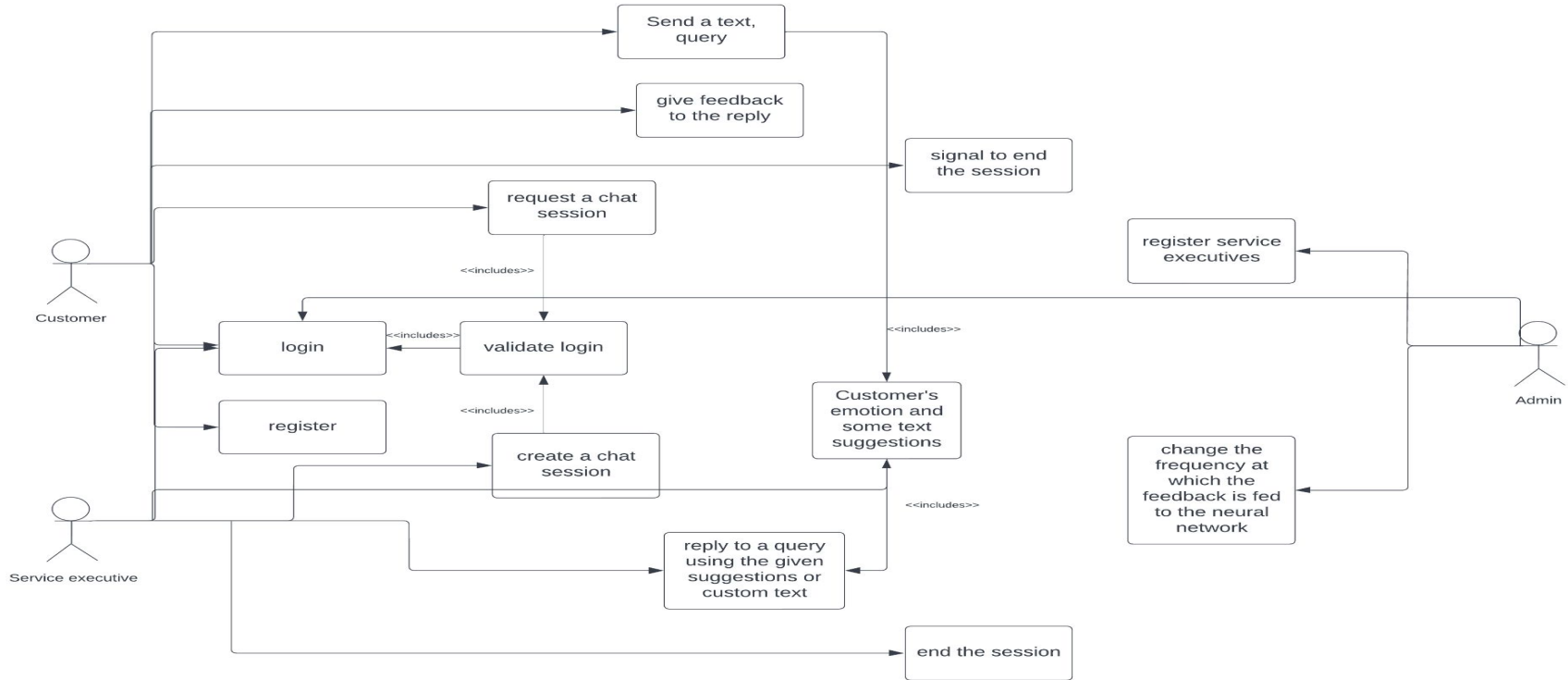*Suggestions provided by model: Okay*

Cont'd:

**Customer:**

No, Thank you

Thank you for bringing it to our attention and allowing us to address it. Again, we are deeply sorry this has happened to you. Have a nice day!
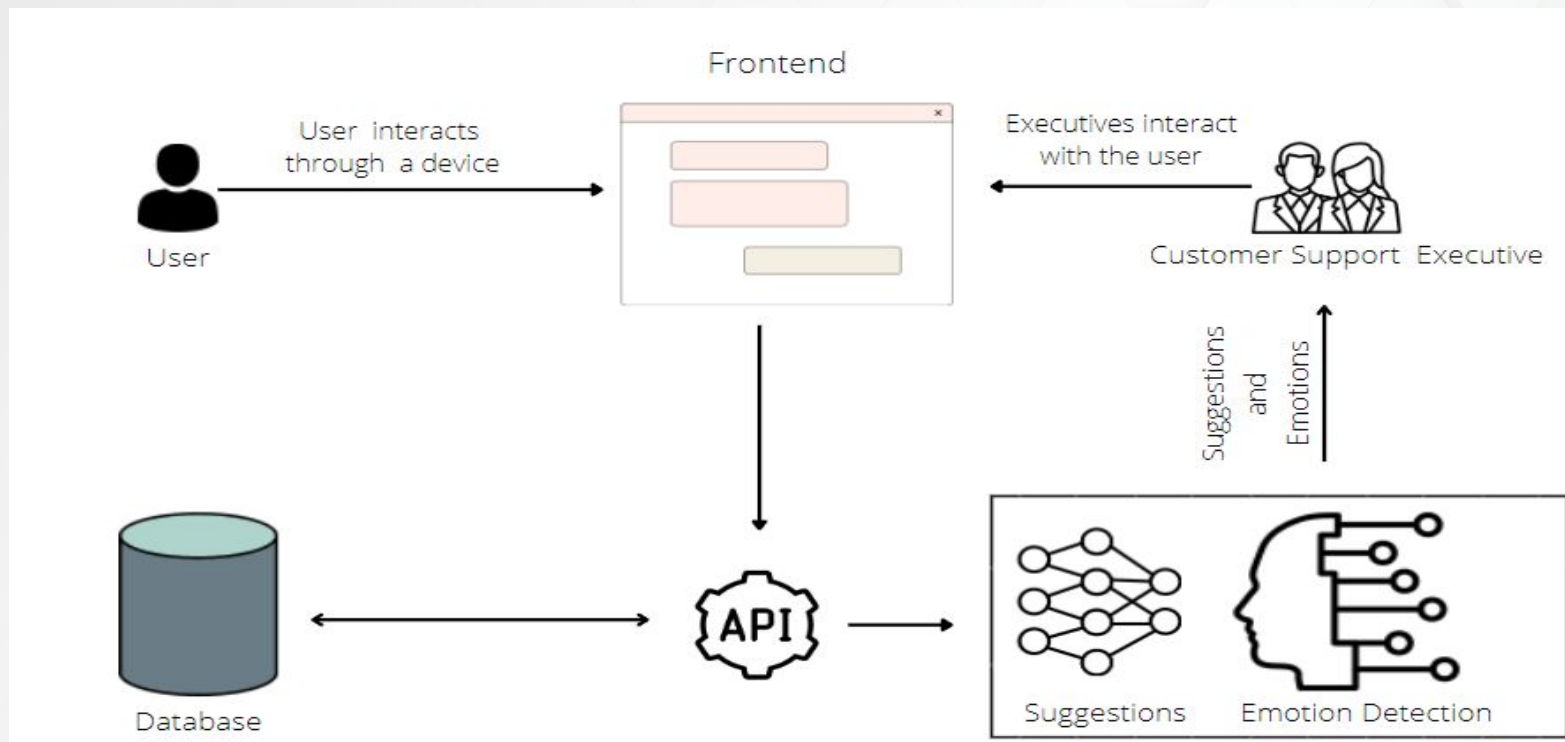
**Executive**

*Emotion: Satisfied, Happy;*

*Suggestions provided by model: Thank you for bringing it to our attention and allowing us to address it.*
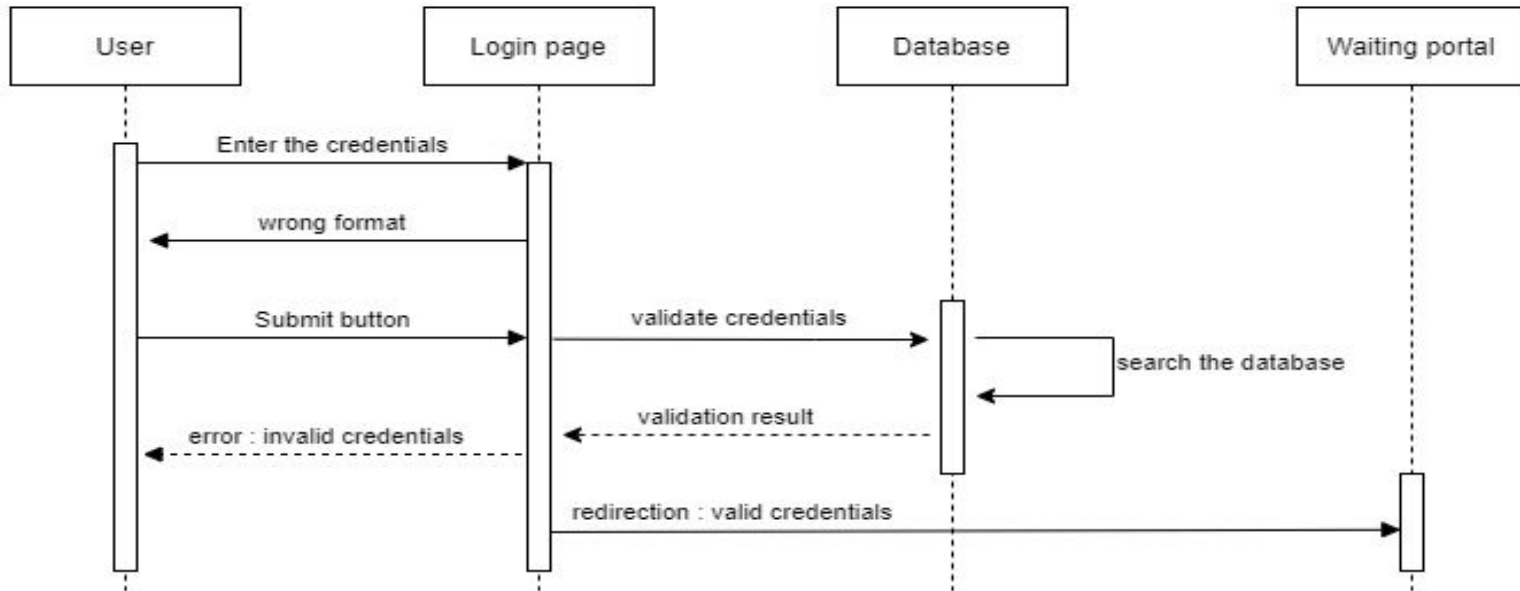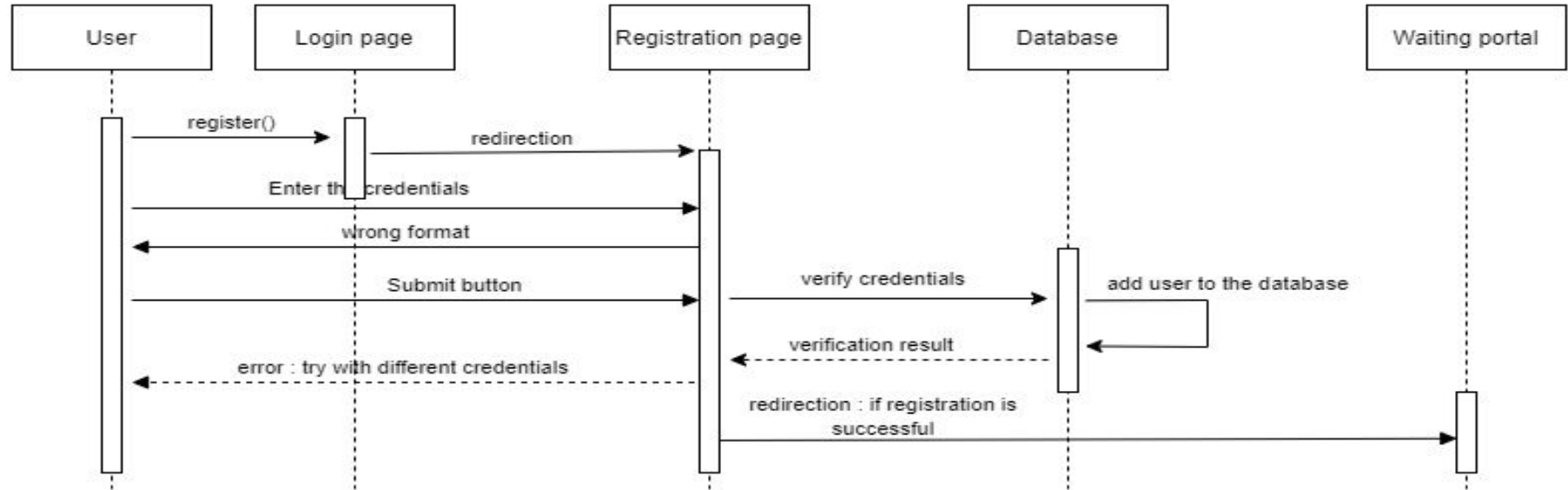
# USE CASE DIAGRAM :

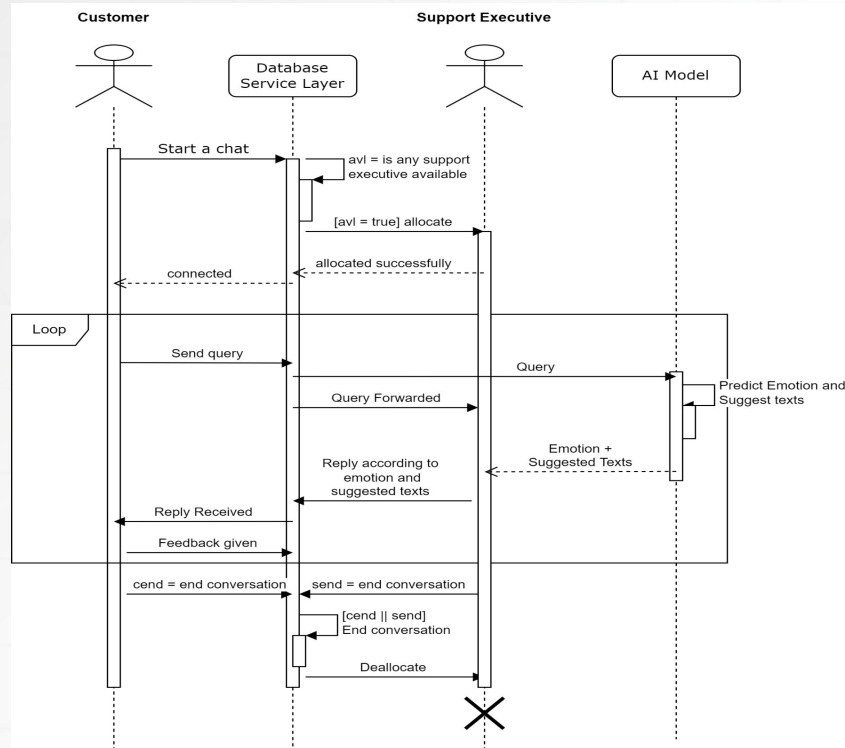# HIGH-LEVEL DESIGN

# Architecture Diagram

# Sequence Diagram (login)

# Sequence Diagram (register)

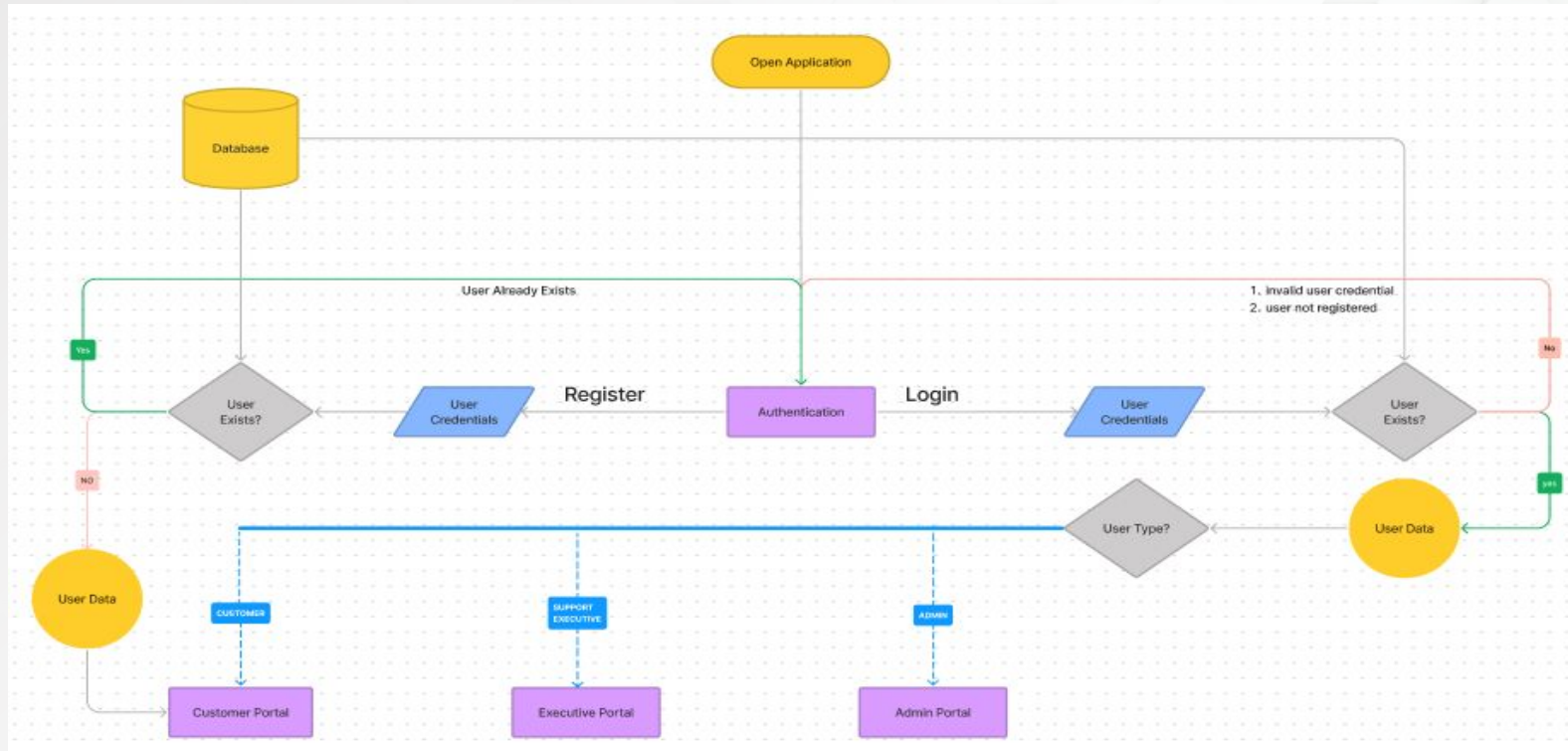# Sequence Diagram (conversation)

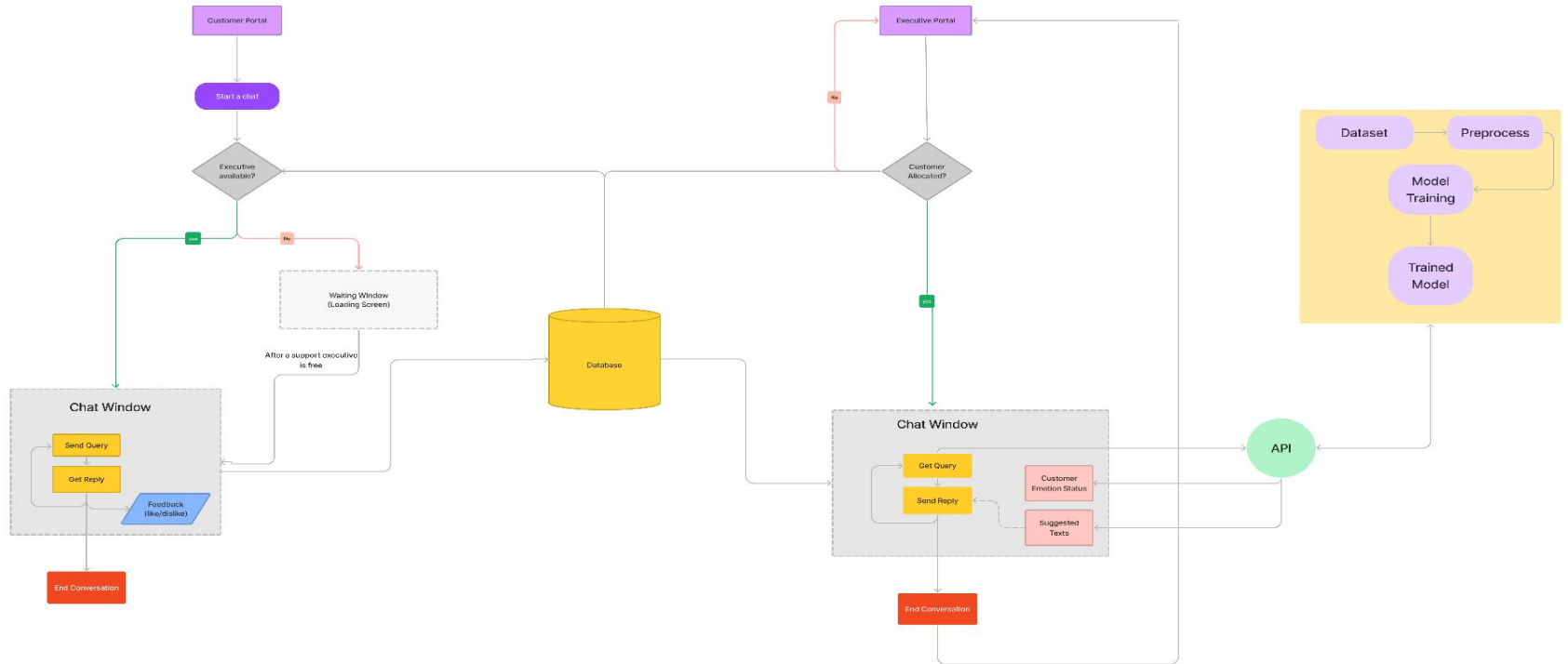# Flow Diagram (Hyperlink)
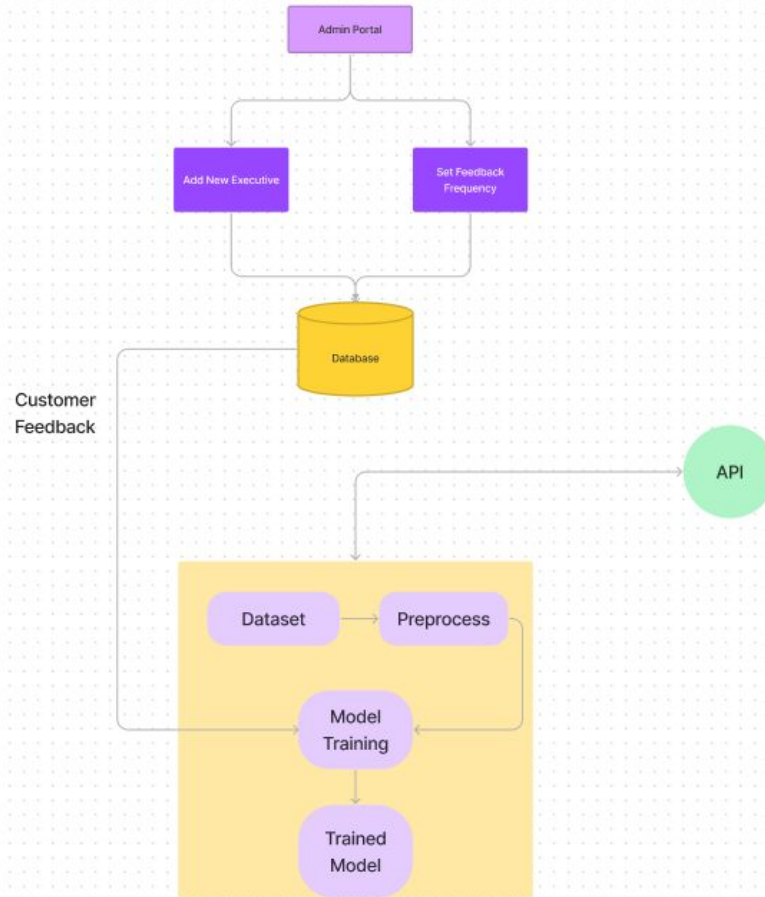
1. Authentication/Login Flow Diagram

## 2. Conversation Flow Diagram

3. Admin Portal Flow Diagram

# The Tech Stack

# Tech Stack

Frontend : React

Why React to be preferred over Angular ?

- It is lightweight javascript library and gives faster and better performance.
- It allows us to use other javascript libraries.
- Better to build UI components as it uses Virtual DOM.
- It is cross platform and comparatively easier to learn.
- Enhance support for server-side-rendering, making it robust for content focussed applications.

# Tech Stack

Cons of Angular JS:

- An angular feature can be confusing for newcomers.
- There is no clear manual and extensive, all-inclusive documentation.
- Steep learning curve
- Angular some time becomes slow with pages embedding interactive elements.
- Third party integration is very difficult.
- While switching from the older versions to the newer ones, one can face several issues.

# Tech Stack

Why a NoSQL Database is preferred?

- This is because SQL requires the use of predefined schemas to determine the structure of data before you work with it and changing the structure can be quite confusing.
- NoSQL databases have dynamic schemas.
- SQL databases are vertically scalable whereas NoSQL is horizontally scalable. Hence for large datasets horizontal scalability is better as we can add more servers easily.
- NoSQL databases are much better suited for big data as flexibility is an important requirement which is fulfilled by their dynamic schema.

# Tech Stack

Database: Firebase

Firebase is a NoSQL database. Firebase is a Backend-as-a-Service containing identity management, realtime data views and a document database. It runs in the cloud. Firebase includes two data stores: the Real-Time Database and Cloud Firestore, each optimized for a different part of application development.

## Why Firebase over MongoDB?

Speaking about Firebase and MongoDB, both are scalable modern database platforms for application developers. Both are post-relational databases with similar JSON-like document data models and schemas. But for our project we chose firebase over MongoDb keeping these noted advantages in mind:

- We won't be dealing with a huge set of Data to be stored nor process on;
- And at the same time, we require high performance level as we arent dealing with complex scenarios, hence for  full back-end as a service (BEaaS) and the least possible effort, Firebase is ideal.
- Firebase provides real time change reflection
- It provides easy integration of authentication mechanism.
- Access to data can be provided from any place in the firebase platform.
- The firebase platform can be easily integrated with other firebase services and is easily manageable. There is no such feature supported in the MongoDB database.

As and when we are dealing with highly secured and highly scalable data and more functionality, we can shift to MongoDB but considering the good features which include authentication, storage and easy usability, we have opted for firebase.

Backend: Python & Node.js

Python will be used to train and export the required models and Node.js will be used to import these models and use them for making the API.

Dataset:

Emotion Detection model -
https://www.kaggle.com/datasets/praveengovi/emotions-dataset-for-nlp

Intent Classification Dataset -
https://huggingface.co/datasets/banking77

# Testing

- Integration testing:
→ We would run some scenarios to make sure all the modules are working fine and interacting as they are supposed to.

# Unit Testing for Authentication

***Positive Scenarios***:

- New Users are only validated or allowed to access the application, after Registration process, where user must provide their respective credentials including user's Email ID, Password and Name.

- Existing users are only validated when their login credentials are already present in the Database, which includes Email ID and Password.

- Only admins have access to create credentials for the customer support executive.

## *Negative Scenarios*:

- <u>User not Registered</u> - If users try to login, but their credentials are not present in the database, they cannot be validated. They must first complete the registration process.

- <u>Invalid Email Format</u> - If users try to register using invalid email format, they cannot be validated.

- <u>Invalid Email ID or Password</u> - If the login credentials provided by the user does not match that of the database, then the user cannot be validated.

- <u>Weak Password</u> - If Password set by the user is weak enough, user must create a new password to register.

- <u>Login as support Executive</u> - Users can only login as a support executive if admin had already added their credentials in the database.

# Unit testing for Neural Network

**Confusion Matrix**: The performance of the model can be tested by plotting the confusion matrix

**Acceptance Criteria:** The text fed to the neural network must be meaningful and grammatically correct for accurate analysis. The expected language of the texts is English.

**Positive test cases**: Below are some positive testing scenarios for emotion detection and text suggestion.

The given statements are the chats sent by the customer which would be sent for emotion detection

1. What is the minimum account balance to be maintained?
2. Why is it taking so long to send my cheque book?
3. I have lost my ATM card.
4. It was helpful. Thank you!

**Negative test cases:** Below are some positive testing scenarios for emotion detection and text suggestion .

1. Ahfgsye ahdsiof fhgin (Meaningless text sent)

2. 27489 292876 28366 (Only numbers are sent)

3. $*^^# *&%#$#@@ ( Only Special characters are sent)

4. When happens what (Text with grammatical error)

5. quand viendra mon cheque (Text in French)

# Waiting Queue

## Positive Scenario:

The proper functioning of the waiting queue will include the following features:

- Whenever a customer requests for a chat with the support executive, they are added to the end of the waiting queue.

- Whenever a support executive becomes available they are allotted a customer from the front of the waiting queue.

- The waiting queue has the proper count of the number of customers waiting for their request of a session with a support executive to be completed.

# Negative Scenario:

- **Improper count of waiting customers:**
  if the waiting queue is unable to keep track of exact number of customers waiting to be serviced by the support executive ,it will lead to a delay in resolving customer query.

- **Improper allotment of support executive:**
  This an error that will occur when a customer waiting at the front of the queue is not allotted a support executive instead query for some other customer waiting at the back of the queue is being serviced.

- **Improper allotment of customer in the waiting queue:**
  Here we can have two scenarios possible:
    - Firstly it might be possible that a customer is totally not being added to the waiting queue whenever there is no support executive available. Hence there wouldn't be any session to resolve the query of this customer.
    - Secondly it might be possible that a customer who has to wait for a support executive to become available is not added to the end of the queue(*non-empty)* but assigned some random order.

# Tentative UI Design

# WELLS FARGO

## Customer View of a conversation