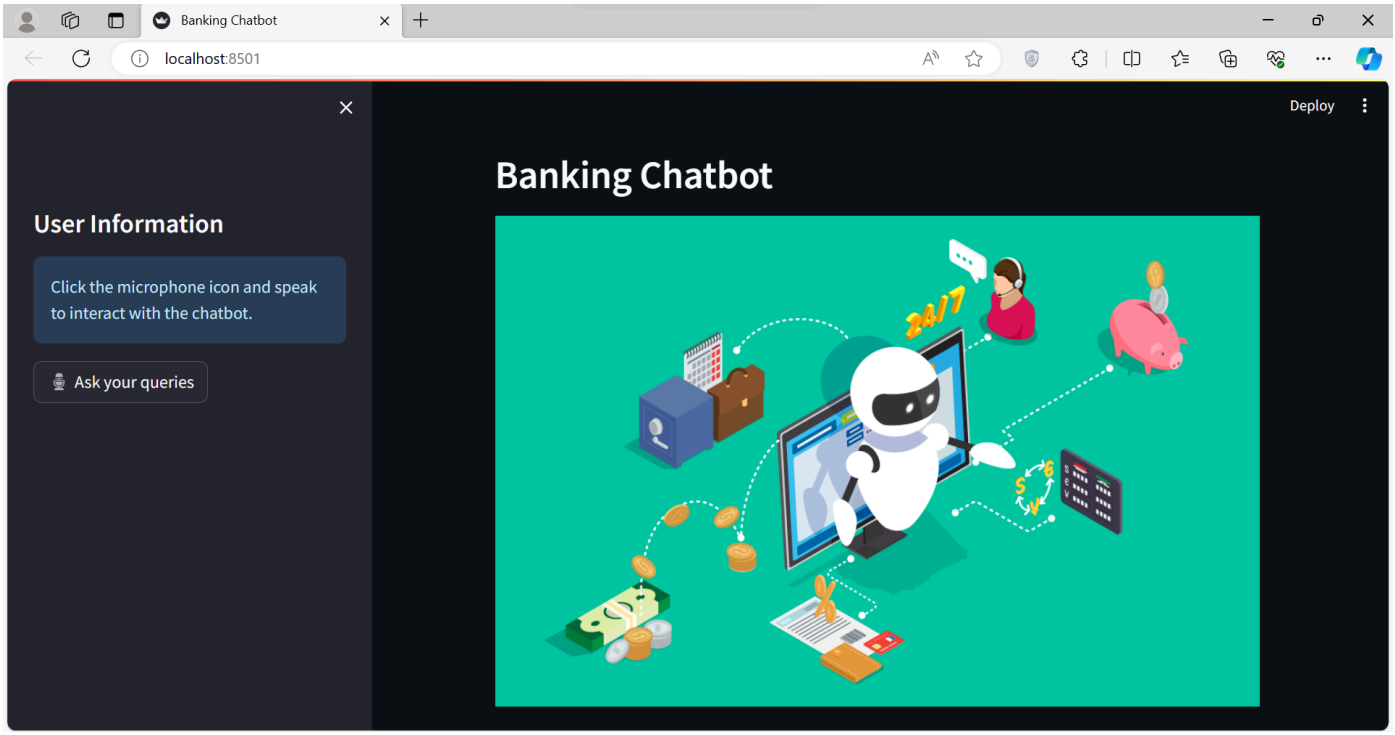
Banking Chatbot



The above picture shows the interface of the working chatbot

Team Name: Phineas and Ferb

Team Members: R Adarsh(12141350) , R Chiranjeevi Srinivas (12141290)

Objective:

This report provides a comprehensive overview of the Banking Chatbot code, its functionalities, and an analysis of its output.

Overview:

The code is designed to create a conversational chatbot for banking interactions, targeting users in rural areas. It utilizes various libraries and frameworks, such as Vosk for speech recognition, Pyttsx3 for text-to-speech conversion, Transformers for intent recognition using a pre-trained BERT model, and Streamlit for the user interface.

Key Features:

1. Speech Recognition:

- Users can input messages via typing or speech recognition using the Vosk library.

- The code captures audio input, converts it into text, and uses the text for further processing.

2. User Registration:

- Users are prompted to register by providing their name.

- A unique authorization code is generated for each user.

3. Intent Recognition:

- BERT-based intent recognition categorizes user input into predefined intents.

- Recognized intents include:

- `TRANSFER_MONEY`: Initiates a money transfer process.

- `GET_BALANCE`: Checks the account balance.

- `GET_DUE`: Checks the dues.

4. Account Handling:

- Account details are stored in a JSON file, including name, account number, and balance.

- Users can check their account balance and dues.

- Money transfer functionality allows users to transfer funds between accounts.

5. Conversational Chatbot:

- Utilizes the ChatOpenAI model for generating conversational responses.

- Maintains a flow of messages to create a natural conversation with the user.

User Interaction Flow:

1. User Registration:

- Users provide their name, and an authorization code is generated.

- If already registered, the existing authorization code is reused.

2. Intent Recognition and Processing:

- User input is processed to determine the intent.

- Based on the intent, the code triggers appropriate actions:

  - Transfer money prompts for recipient details and amount.

- Get balance and Get due prompt for the account number.

3. Speech Interaction:

- Users can use the microphone to speak to the chatbot.

- Speech is converted to text for processing.

4. Conversational Chatbot:

- Generates responses in a conversational manner using a custom chatbot model (ChatOpenAI).

Output Analysis:

1. User Registration Output:

- Displays the user's authorization code upon successful registration.

- Provides a success message or a warning for invalid inputs.

2. Intent Recognition Output:

- Triggers appropriate actions based on the recognized intent.

- Prompts for additional details for specific intents.

3. Money Transfer Output:

- Success or error message displayed based on the outcome of the money transfer operation.

- Provides feedback to the user on the success or failure of the transaction.

4. Balance and Dues Output:

- Displays the account balance or dues based on user input.

- Provides feedback to the user regarding the requested information.

5. Conversational Chatbot Output:

- Displays responses from the ChatOpenAI model based on user input.

- Provides a conversational flow for a natural interaction.

Areas for Improvement:

1. Optional Features:

- The code does not implement optional features such as speaker identification. If required, integrating voice-based user identification would be beneficial.

Conclusion:

The output from the Banking Chatbot code demonstrates a user-friendly conversational interface for banking interactions. Users receive clear feedback on their actions, whether it's registration, money transfer, or account inquiries. The conversational flow, combined with speech and text inputs, enhances the user experience, providing a promising foundation for further development.