

Bank Bot Chat Box Using AI for Bank FAQs

MILEATONE 3

Abstract

Bank Bot is an AI-powered chatbot designed to enhance customer support services in banking systems. The system answers frequently asked questions (FAQs), assists users with account information, helps in balance inquiries, transaction details, and provides general banking assistance. The project integrates machine learning-based Natural Language Processing (NLP), Flask backend, and an interactive web-based chat interface. It reduces workload on human staff and provides 24/7 instant support.

Introduction

Banking industries are rapidly shifting towards digital automation to improve customer experience. Traditional customer service requires long waiting times, manual verification, and human effort. To solve this, AI chatbots are widely adopted.

BankBot provides automated responses to frequently asked banking questions using NLP. It can identify user intent, process queries, retrieve required information, and provide accurate responses.

Problem Statement

Customers often face delays while contacting bank customer service. Common questions like balance inquiry, loan details, KYC updates, and card information consume support time. Manual processes cannot ensure 24/7 availability.

Thus, there is a need for an intelligent system that *automatically* answers FAQs and supports basic customer queries without human involvement.

Objectives

- To design an AI-based chatbot for answering banking FAQs.
- To automate basic customer support operations using NLP.
- To provide instant information about accounts, transactions, and services.
- To improve the accuracy of banking assistance through ML.
- To provide an admin panel to manage FAQs and training data.

Proposed System

The proposed system is an AI chatbot capable of understanding questions, predicting intent, and generating responses. It includes:

- User Login
- Dashboard
- Chatbot with NLP response engine
- FAQ management system
- Admin analytics dashboard
- Training data editor

The system uses Flask for the backend, SQLite database for storage, and machine learning models for intent detection.

System Architecture

User → Web Interface → Flask Backend → NLP Engine → Database → Response

Components:

- Frontend (HTML/CSS/JS): Chat UI
- Flask Backend: API handling, session management
- NLP Module: `generate_bot_response()` predicts intent
- Database: Stores users, chat logs, FAQs, transactions
- Admin Panel: Manage FAQs, analytics, logs

Modules Description

User Login Module

- Users log in using email and password.
- Verifies user identity from the database.

Dashboard Module

- Shows account details, balance, recent transactions.

Chatbot Module

- Receives user's message
- Predicts intent using ML/NLP
- Retrieves required data
- Returns an accurate response

FAQ Manager

- Admin can add/edit/delete FAQ questions & answers.

Admin Panel

- View chatbot accuracy
- Check analytics statistics
- Export logs
- Edit training data

Chat Logs Module

- Stores each conversation into SQLite database
- Can export as CSV

Database Design

Tables:

1. users

account_number
name
email
password
phone
balance

2. chat_logs

id
account
user_message
bot_response
intent
confidence
timestamp

3. faq

id
question
answer

4. transactions

id
account_number
amount
type
timestamp

Technologies Used

- **Frontend:** HTML, CSS, JavaScript
- **Backend:** Python Flask
- **NLP:** Machine Learning-based Intent Detection
- **Database:** SQLite
- **Libraries:** pandas, csv, datetime

Methodology

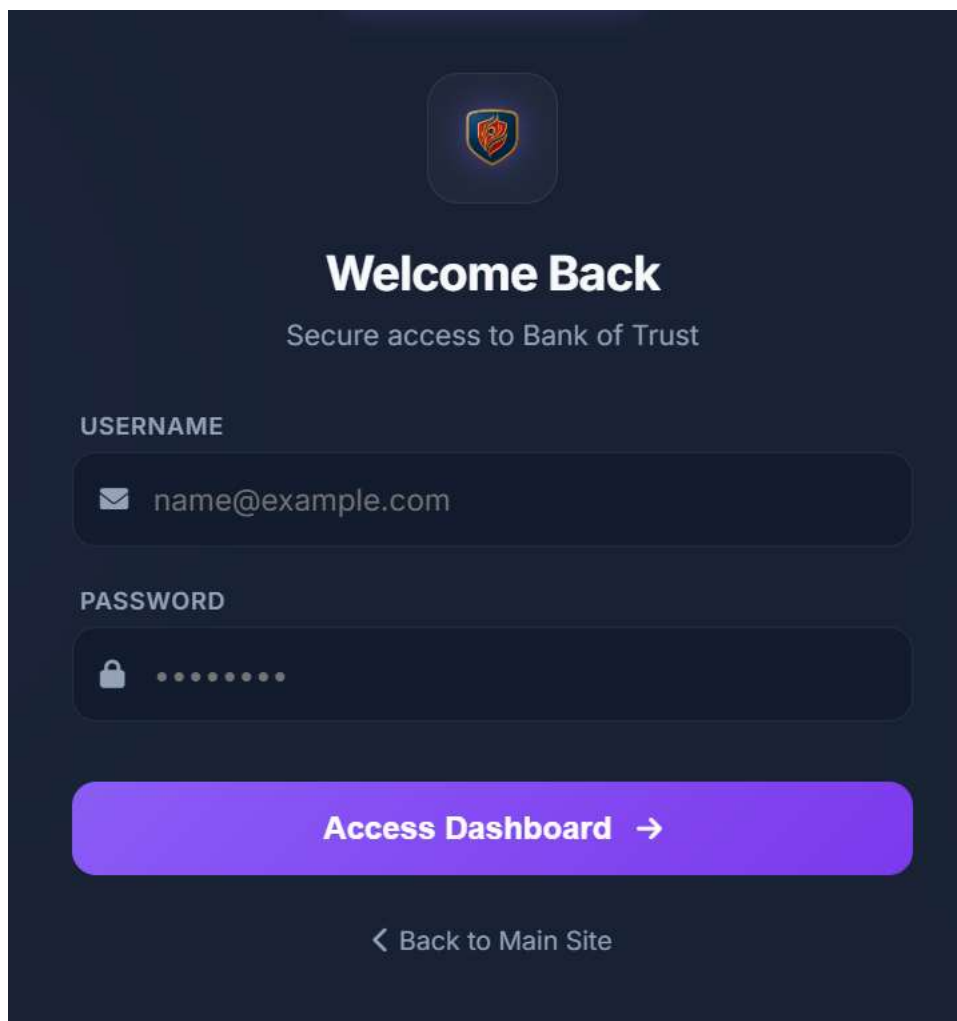
- User sends message
- Message processed by NLP module
- Intent and entities extracted
- Database queried if required
- Response generated

Testing

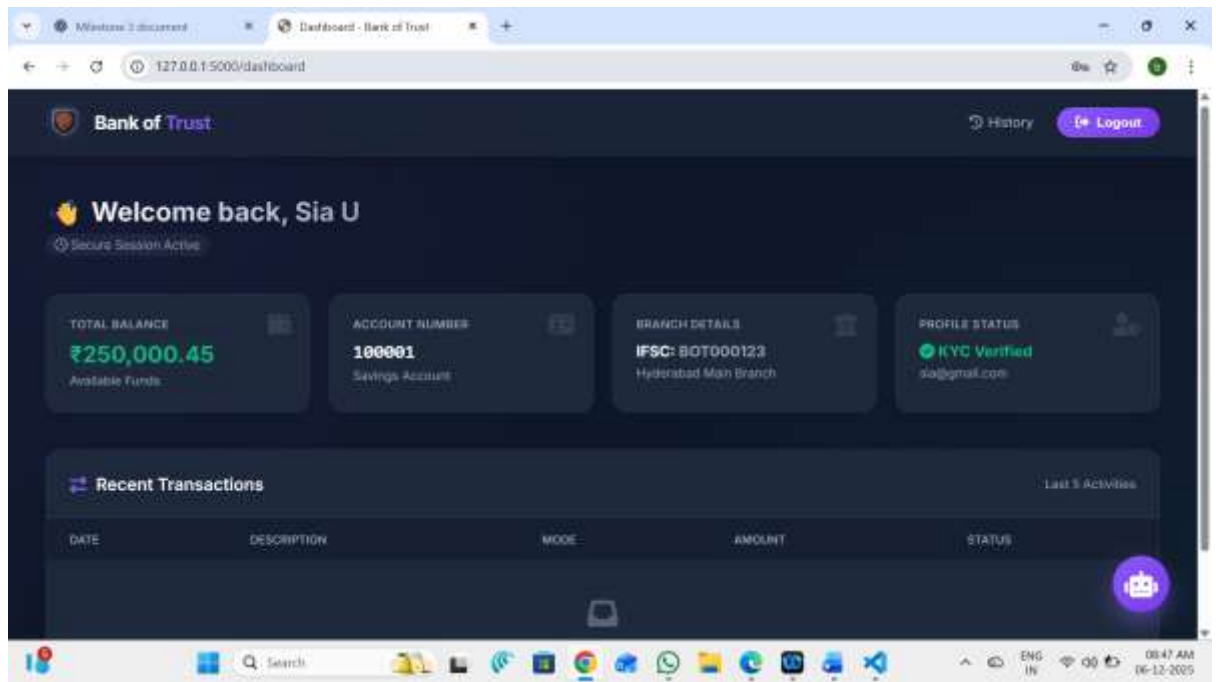
- Functional Testing
- Intent Accuracy Testing
- Admin Panel Testing
- UI Testing

13. Screenshots (To be added later)

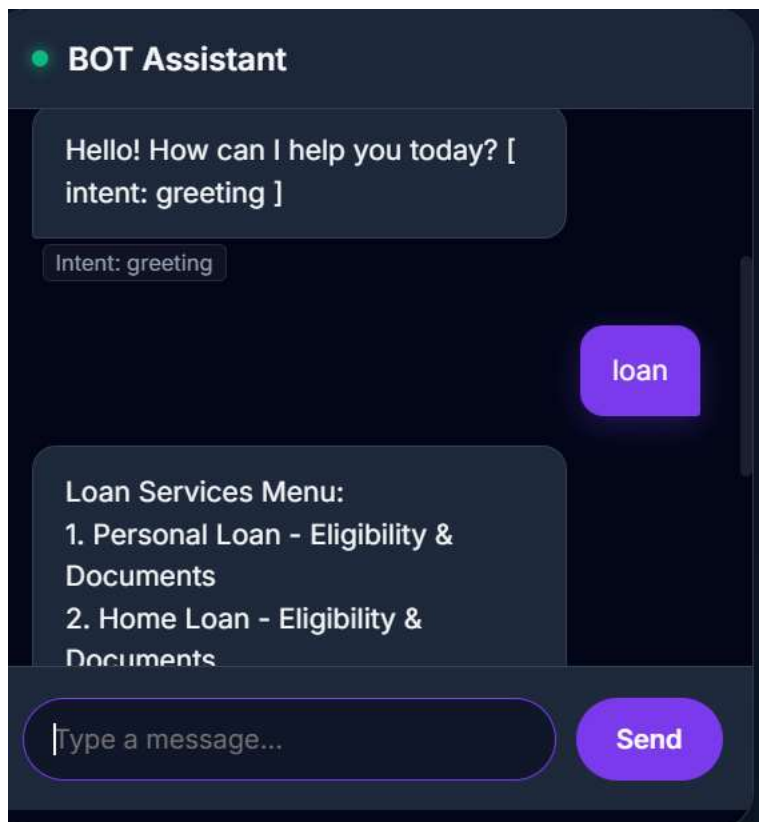
- Login Page



- Dashboard



- Chatbot Interface



Conclusion

The BankBot AI system improves customer service efficiency in banking by providing real-time automated responses. It reduces workload on staff, ensures 24/7 availability, and offers a scalable system for future enhancements.