

# Gujarat Vidyapith Chatbot

---

**Presented by:**

Falguni Maththar – 240160510026

Akshit Sonani - 240160510049

**Guided by:**

Dr. Neepa Shah



# Agenda

- Introduction
  - Key Features of Chatbot
  - Technology Stack
  - Project Architecture
  - Database Schema
  - UI Design
  - Limitations of the System
  - Proposed Enhancements
-

# Introduction

- **Why Chatbot?**

This project aims to create an efficient and user-friendly chatbot system for Gujarat Vidyapith.

The system helps students, staff and visitors get quick answers to common queries related to the university.

It reduces manual workload and improves information accessibility.

The chatbot uses predefined intents, processes user questions and returns accurate responses.

The goal is to provide a simple, fast and reliable digital helpdesk.

- **What Problem Does It Solve?**

Slow information delivery

Repeated manual queries to staff

Difficulty finding accurate and updated details

Confusion during admissions and department-level inquiries

Lack of a single point of information for the university

# Key Features of Gujarat Vidyapith Chatbot

- Provides instant answers to university-related queries
- Covers departments, courses, admissions and general campus information
- Works on intent-based training to maintain accuracy
- Simple and clean chat interface for students and visitors
- Centralized admin panel to add, update or remove intents
- Responds 24/7 without depending on staff availability
- Reduces repetitive workload for faculty and office staff
- Supports structured responses for important topics like fees, programs and eligibility
- Can be expanded with automatic model training in future
- Lightweight design that works on the web without heavy setup





## User Module – Features

- **User Registration**

New users can create an account using basic details like name, email and password.

- **Secure Login**

Only registered users can log in with valid credentials to access the chatbot.

- **Profile Handling**

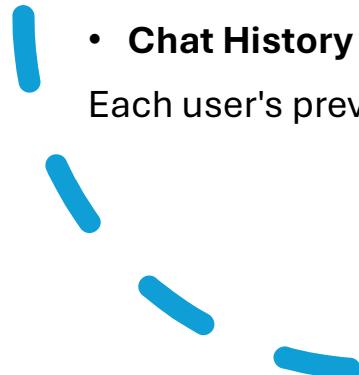
Users can view and update their profile details (name, contact info, etc.) to keep data accurate.

- **Chat with Chatbot**

Logged-in users can ask queries related to Gujarat Vidyapith and get instant responses.

- **Chat History**

Each user's previous conversations are stored so they can review past answers anytime.



## **Admin Module – Features**

- **Admin Dashboard**

Central panel showing overview of total users, active users, total chats and recent activity.

- **User Status Monitoring**

Admin can see which users are active, inactive or newly registered.

- **View User Chat History**

Admin can access user chat logs to:

- Analyze common queries
- Improve intents and responses
- Monitor misuse or inappropriate queries

- **System Control**

Admin can manage intents, update responses and maintain the overall quality of the chatbot.

# Technology Stack

---

- **Frontend**

HTML

CSS

JavaScript

Bootstrap (for responsive UI)

- **Backend**

Python

Django Framework

Chatbot core logic using NLTK / trained intent model

- **Database**

SQLite (or the one you used)

Stores users, chat history, intents and admin data

- **Machine Learning / NLP**

NLTK for text preprocessing

Bag-of-Words model

Custom trained model using json dataset

- **Tools & Libraries**

Django Admin Panel

NumPy

Pickle for model storage

JSON for intents dataset

GitHub for version control

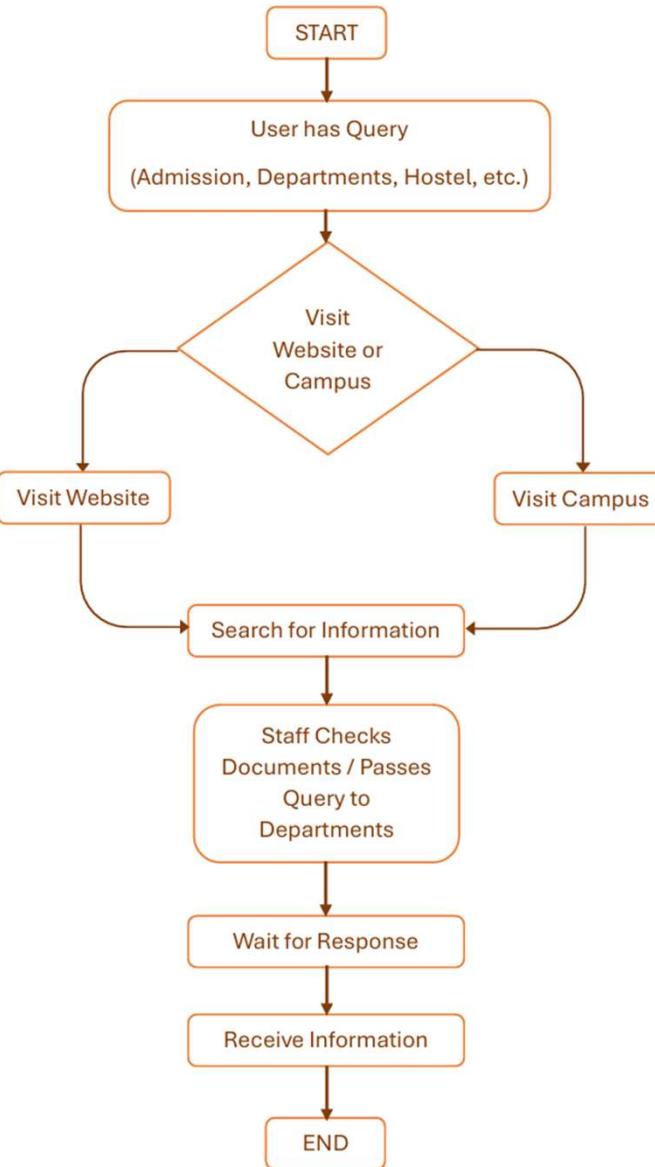
- **Environment**

VS Code

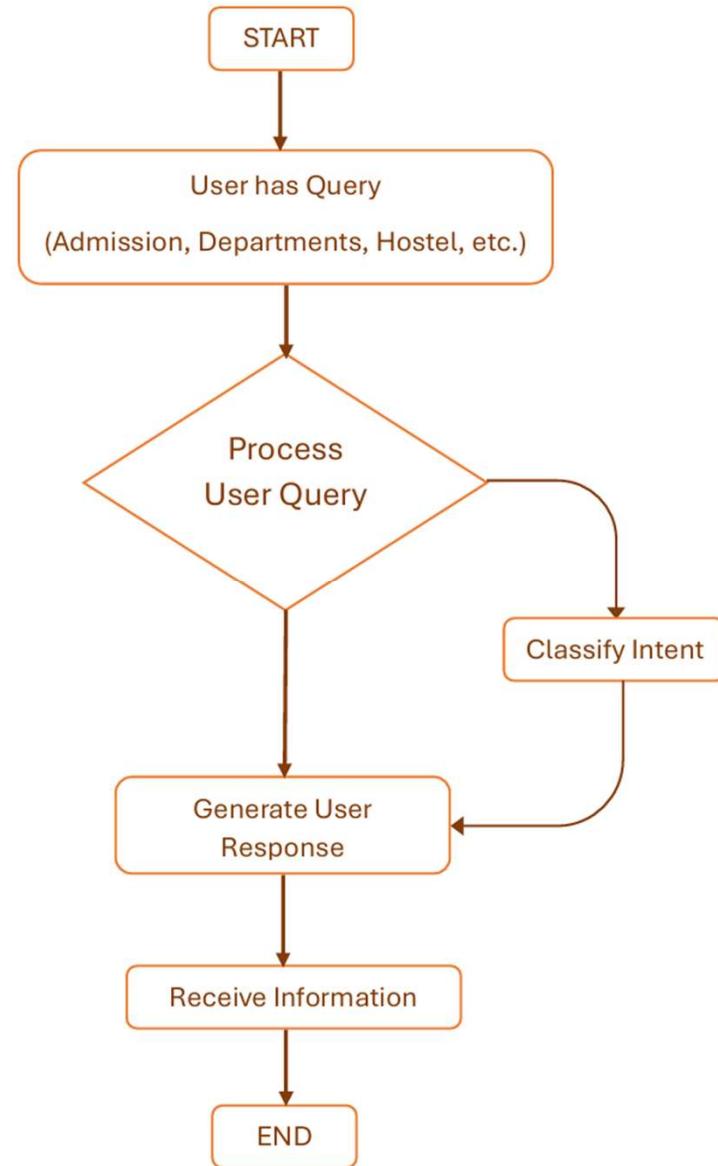
Python Virtual Environment

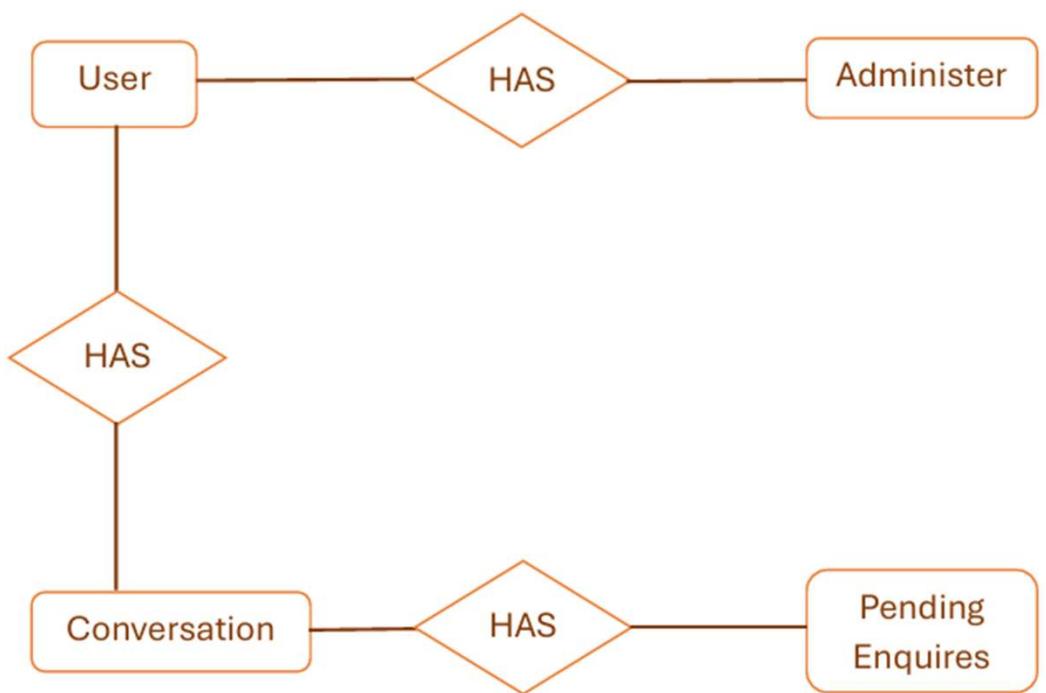
Browser for deployment testing

# System Flow Chart (Existing System)



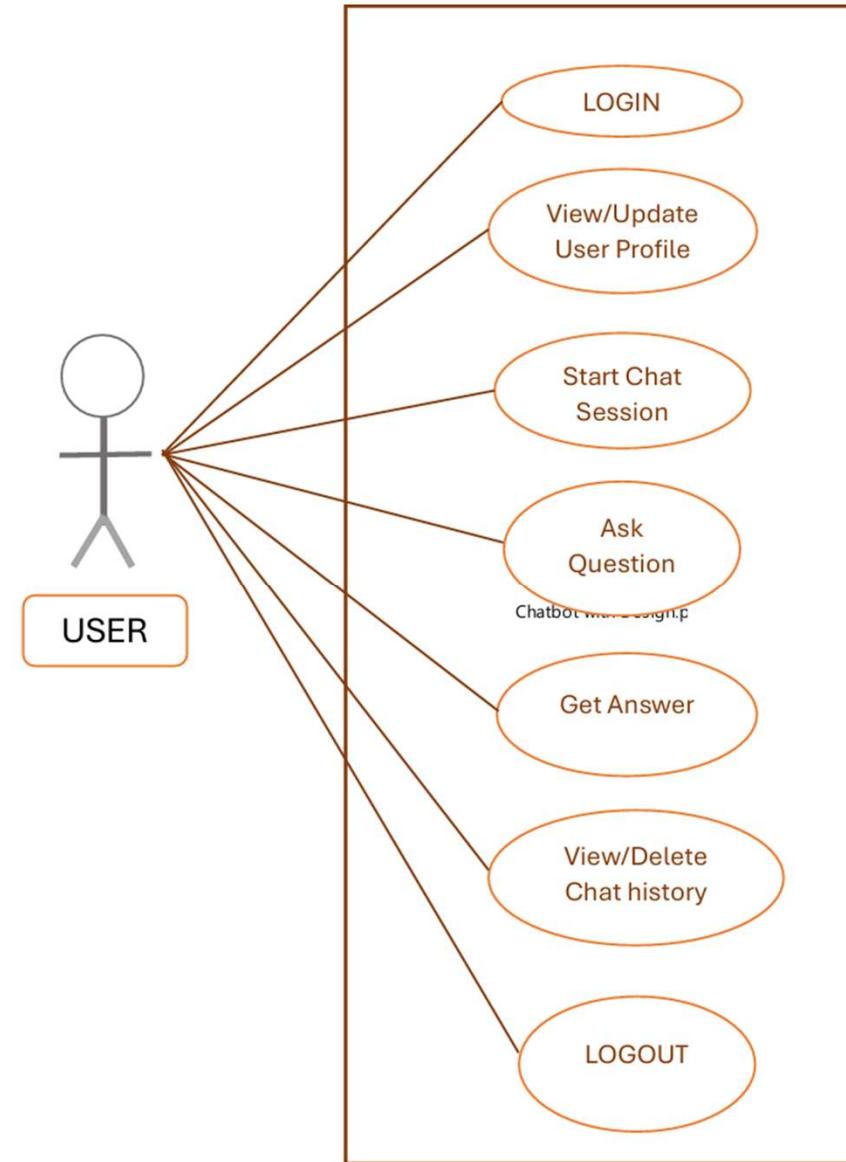
# System Flow Chart (Chatbot System)



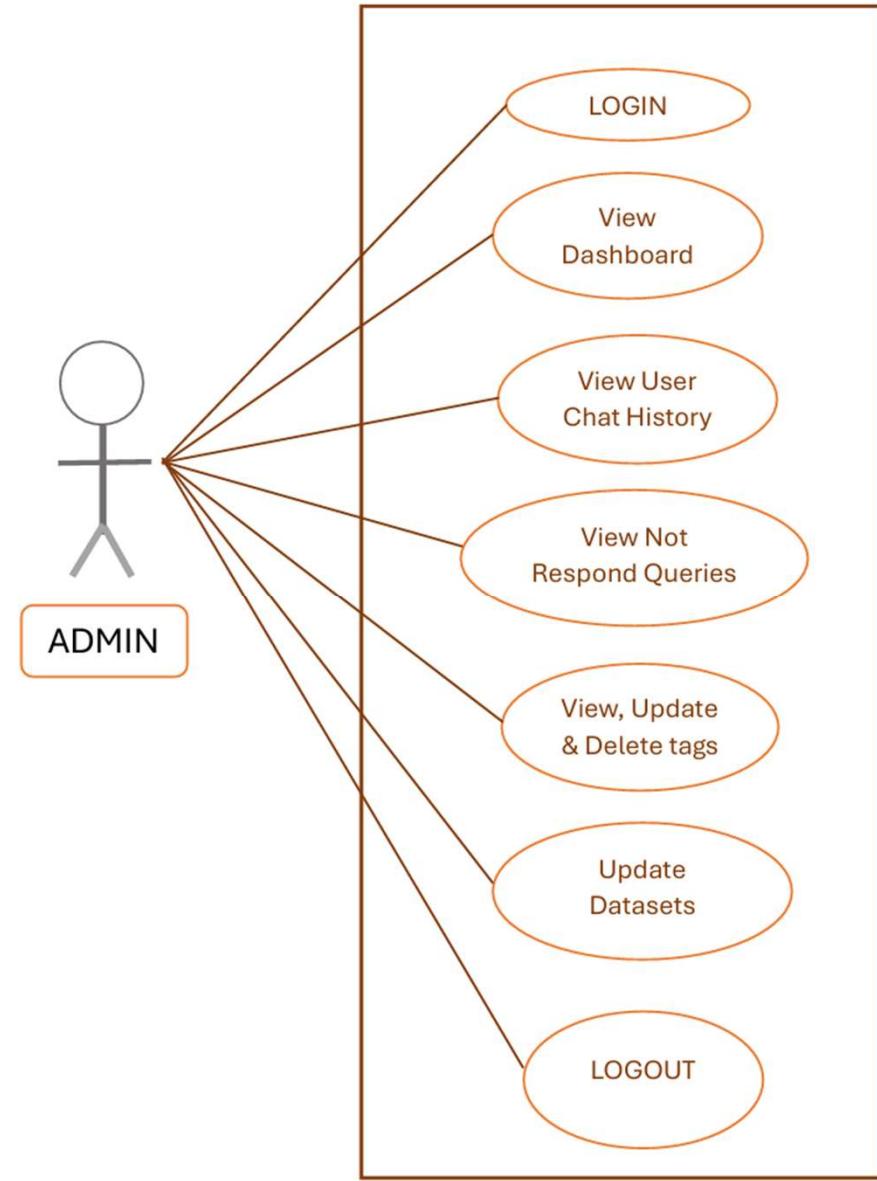


## E-R Diagram(ERD)

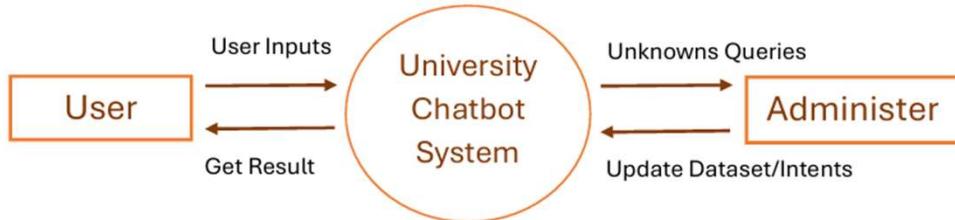
# Use Case Diagram (User)



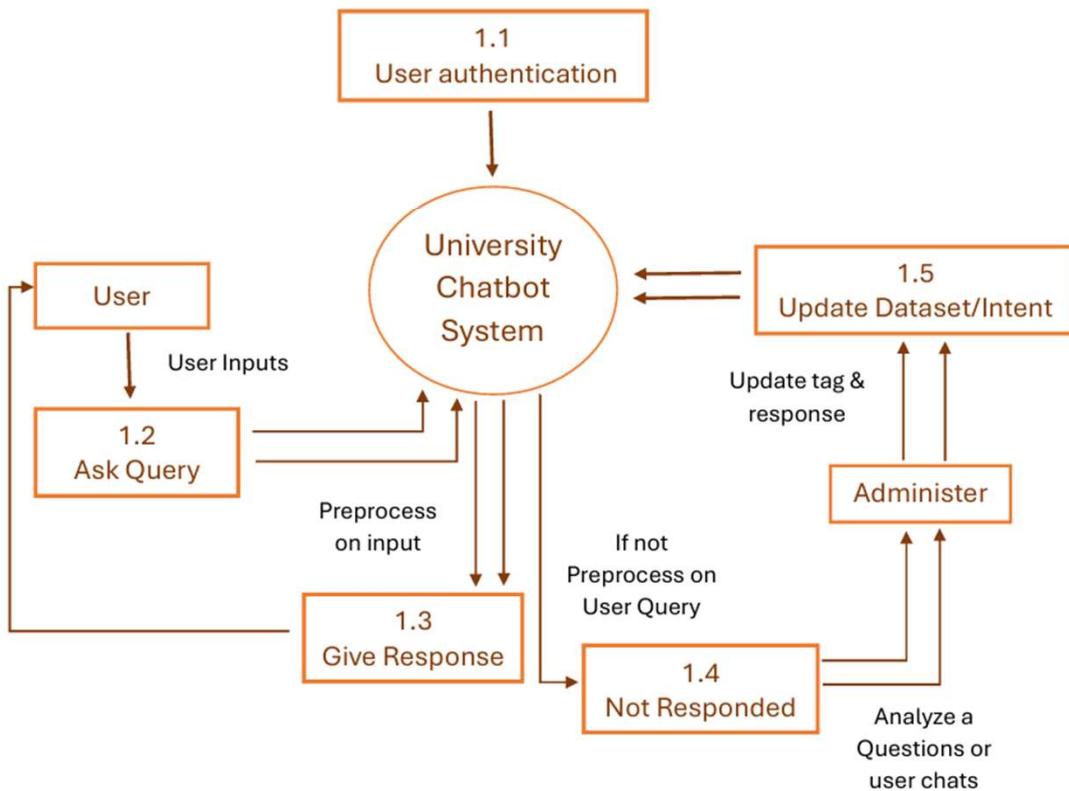
# Use Case Diagram (Admin)



LEVEL 0:

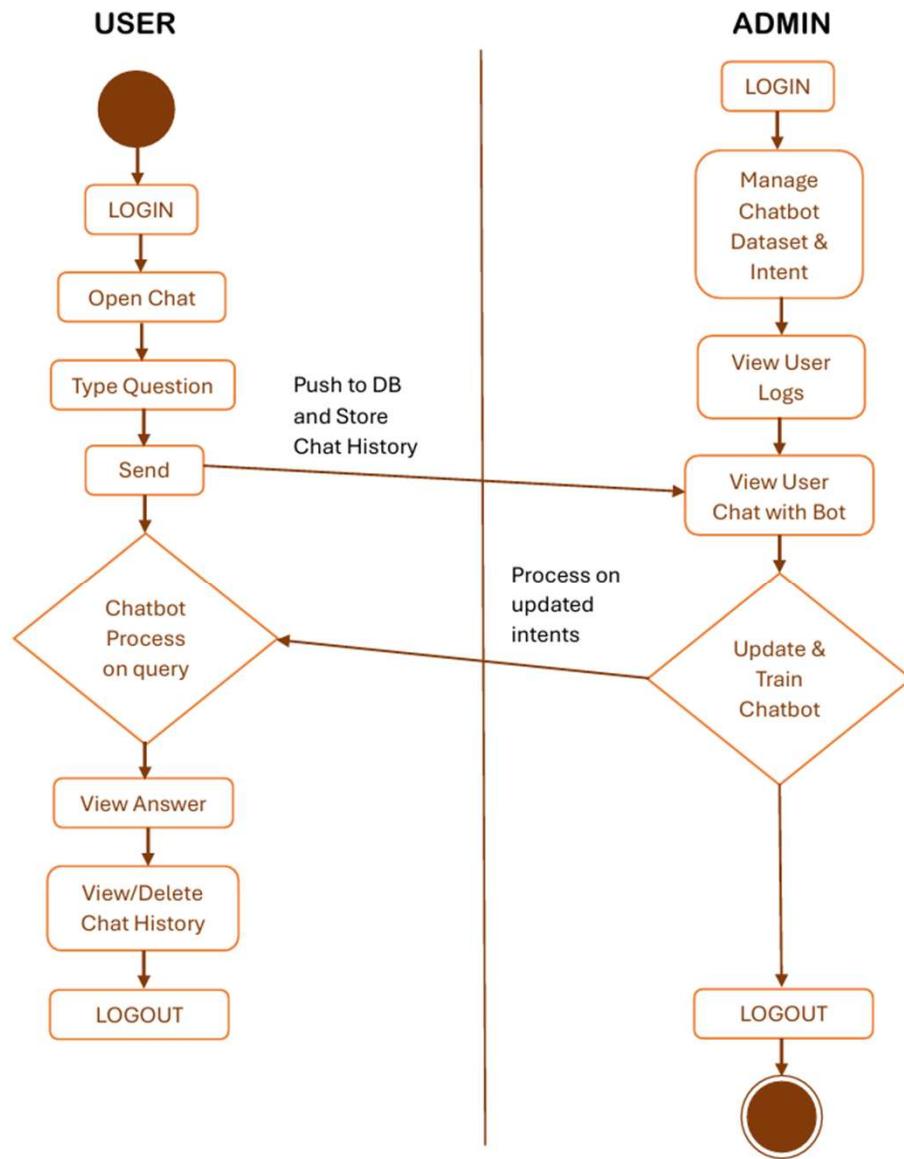


LEVEL 1:



# Data Flow Diagram(DFD)

# Activity Diagram



# Database Schema

---

## 1. User Table Structure (ChatBot\_user)

Column Name	Data Type	Description
<code>_id</code>	ObjectId	MongoDB internal ID
<code>id</code>	Integer	Auto-increment user ID
<code>full_name</code>	String	User's full name
<code>username</code>	String	Unique login username
<code>email</code>	String	User email address
<code>password</code>	String	Plain password (should be hashed in production)
<code>gender</code>	String	male / female
<code>role</code>	String	student / parent / faculty / staff / other
<code>profile_image</code>	String	Profile picture path
<code>joined_date</code>	Datetime	Registration timestamp
<code>status</code>	String	active / inactive

## 2. Admin User Table (ChatBot\_adminuser)

Column Name	Data Type	Description
<code>_id</code>	ObjectId	MongoDB auto ID
<code>username</code>	String	Admin username
<code>email</code>	String	Admin email
<code>password</code>	String	Admin password

### 3. Pending Questions Table (ChatBot\_pendingquestion)

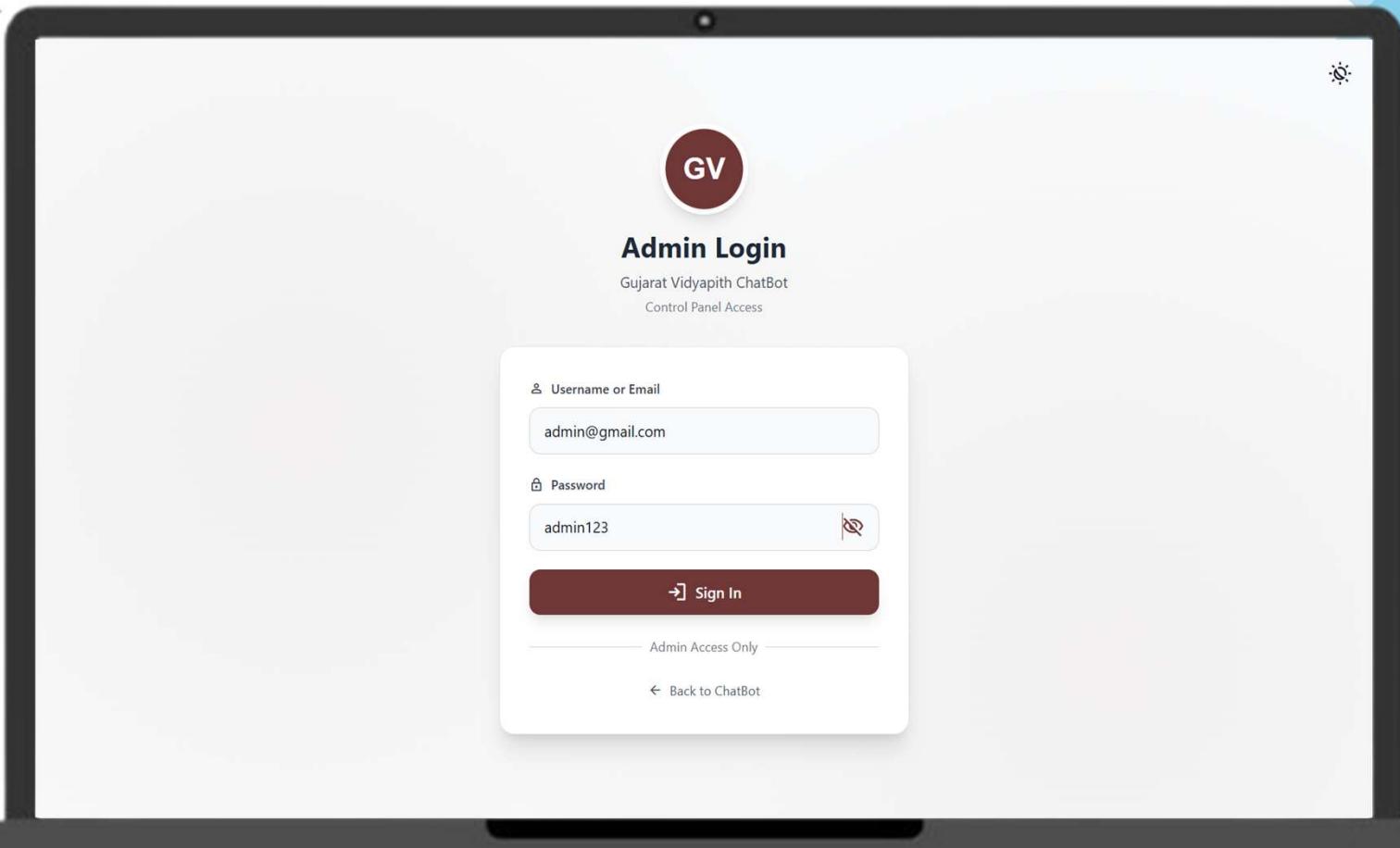
Field Name	Data Type	Description
<b>_id</b>	ObjectId	MongoDB internal ID
<b>id</b>	Integer	Unique question ID
<b>user_id</b>	Integer	FK → <a href="#">User.id</a>
<b>username</b>	String	Username of the person who asked
<b>question_text</b>	String	User's actual question
<b>model_tag</b>	String	Chatbot's predicted tag
<b>confidence</b>	Float	Prediction confidence (0-1)
<b>created_at</b>	DateTime	When question was asked
<b>is_resolved</b>	Boolean	false = pending, true = handled
<b>admin_note</b>	String	Admin explanation / fix note

### 4. Conversation Table (ChatBot\_conversation)

Column Name	Type	Description
<b>_id</b>	ObjectId	Unique conversation ID
<b>id</b>	Integer	Conversation row ID
<b>user_id</b>	Integer	FK → <a href="#">User.id</a>
<b>message</b>	String	Chat message content
<b>is_bot</b>	Boolean	true = bot message, false = user
<b>created_at</b>	DateTime	Message timestamp

### 5. Chat Dataset (intents.json)

Field	Type	Description
<b>tag</b>	String	Intent identifier
<b>patterns</b>	Array	User input examples
<b>responses</b>	Array	Bot output responses



1. Admin Login

The screenshot displays the GVP Admin Dashboard interface. On the left, a vertical sidebar contains the following navigation items:

- GV (User icon)
- Gujarat Vidyapith
- Admin Control Panel
- Dashboard** (highlighted in brown)
- Users
- Chat History
- Pending Questions
- Dataset Manager
- Datasets Browser

At the bottom of the sidebar are Logout and Help links.

The main content area features a title "User Statistics" with a subtitle "Overview of registered users by category". It shows four categories with active counts:

Category	Count	Status
Students	2	Active
Parents	2	Active
Faculty	1	Active
Others	1	Active

A large dark red box highlights the total registered users: **6**, with the note "Across all categories". To the right of this box is a user icon.

Below this are two more sections:

- Total Chats: 1 (View Chats button)
- Pending Questions: 1 (Review Now button)

## 2. Admin Dashboard & Statistics

The screenshot shows the GVP Admin Dashboard with the title "User Management". On the left, there's a sidebar with links: Dashboard, Users (which is highlighted in red), Chat History, Pending Questions, Dataset Manager, Datasets Browser, and Logout. The main area has four categories: Students (2), Parents (2), Faculty (1), and Others (1). Below these are search and filter options: "Search by name, email, or ID...", "All Categories", and "All Status". A table lists user details:

USER	CATEGORY	EMAIL	JOINED DATE	STATUS
hardik mekhiya ID: #36	Parent	hardik@gmail.com	Nov 23, 2025	Inactive
jay heruwala ID: #35	Student	jay@gmail.com	Nov 23, 2025	Inactive
sunny thakor ID: #34	Staff	sunny@gmail.com	Nov 23, 2025	Inactive
hemal rathod ID: #33	Faculty	hemal@gmail.com	Nov 23, 2025	Inactive
Falguni Maththar ID: #32	Parent	falguni@gmail.com	Nov 23, 2025	Inactive
Akshit Sonani ID: #31	Student	sonaniakshit@gmail.com	Nov 23, 2025	Active

### 3. User Management

GVP Admin Dashboard

Admin Portal ☰

**GV**

Gujarat Vidyapith  
Admin Control Panel

- Dashboard
- Users**
- Chat History
- Pending Questions
- Dataset Manager
- Datasets Browser

Logout

## User Management

Students 2

Parents 2

Faculty 1

Others 1

Search by name, email, or ID...

All Categories

Active

USER	CATEGORY	EMAIL	JOINED DATE	STATUS
A Akshit Sonani ID: #31	Student	sonaniakshit@gmail.com	Nov 23, 2025	Active

The screenshot shows the GVP Admin Dashboard interface. On the left, there's a sidebar with a logo (GV), the text "Gujarat Vidyapith Admin Control Panel", and several menu items: Dashboard, Users, Chat History (which is highlighted in red), Pending Questions, Dataset Manager, and Datasets Browser. At the bottom of the sidebar are Logout and Help links. The main content area is titled "All Users Chat History" and features a search bar at the top. Below the search bar are six user profiles, each with a small icon, the user's name, and their email address. The users listed are hardik mekhiya (hardik@gmail.com), jay heruwala (jay@gmail.com), sunny thakor (sunny@gmail.com), hemal rathod (hemal@gmail.com), Falguni Maththar (falguni@gmail.com), and Akshit Sonani (sonaniakshit@gmail.com). The last user, Akshit Sonani, has a "Has Chat" status indicator next to their profile.

#### 4. All Users Chat History

GVP Admin Dashboard

GV

Gujarat Vidyapith  
Admin Control Panel

- Dashboard
- Users
- Chat History
- Pending Questions
- Dataset Manager
- Datasets Browser

Logout

All Users Chat History

Search by name, username or email

Akshit Sonani (akshit001)

hi  
2025-11-23T09:18:55.804004+00:00

Namaste! You can ask me about admissions, courses, departments or campus facilities.  
2025-11-23T09:18:55.982064+00:00

gujarat vidyapith  
2025-11-23T09:20:36.334754+00:00

Gujarat Vidyapith, started by Mahatma Gandhi in 1920, is a Gandhian university with many UG, PG and PhD programs. You can see the full profile on the official site.  
2025-11-23T09:20:36.341148+00:00

fees

hardik mekh  
hardik@gmail.co

hemal ratho  
hemal@gmail.co

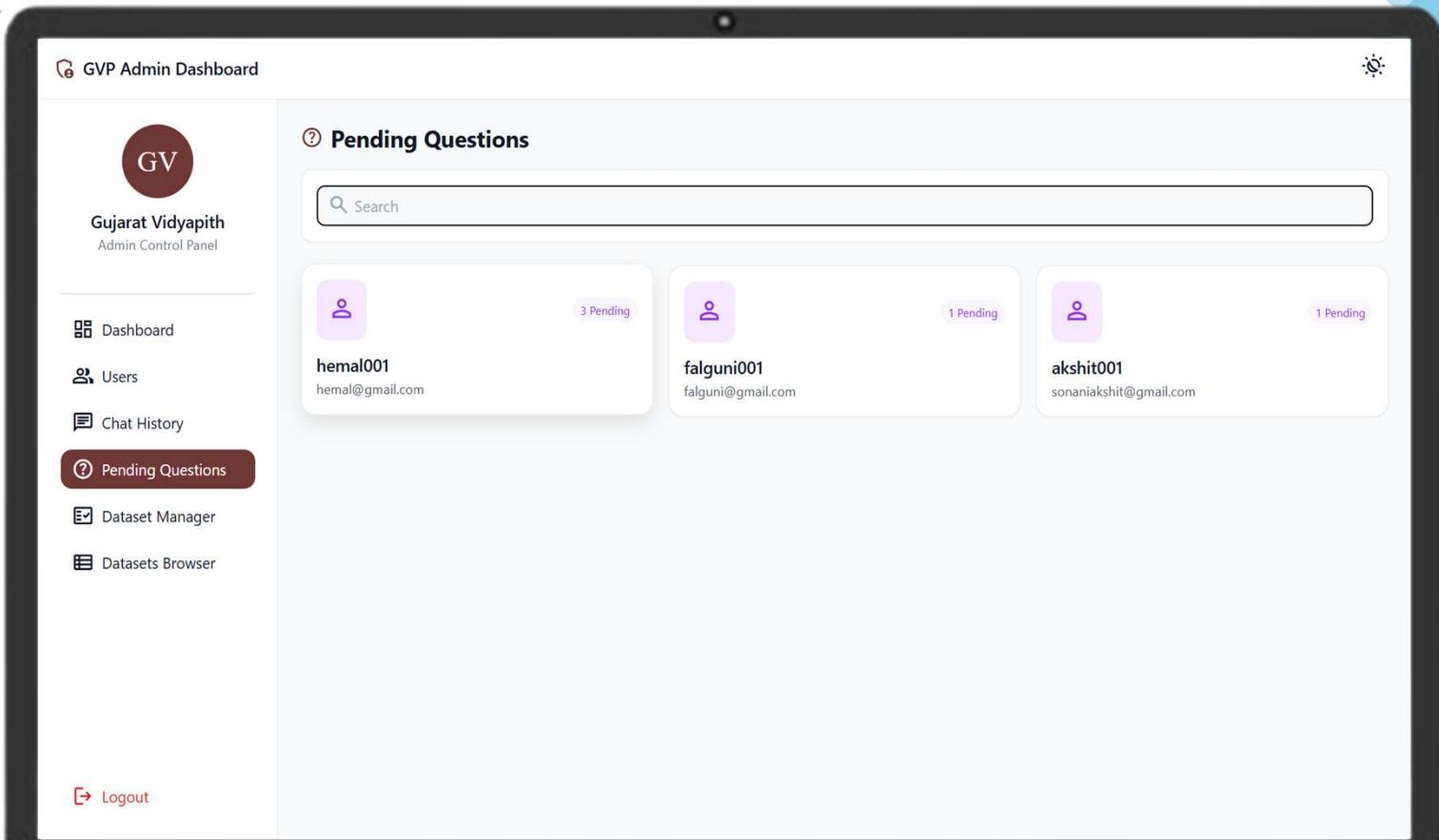
sunny thakor  
sunny@gmail.com

Akshit Sonani  
sonaniakshit@gmail.com

No Chat

Has Chat

Admin Portal

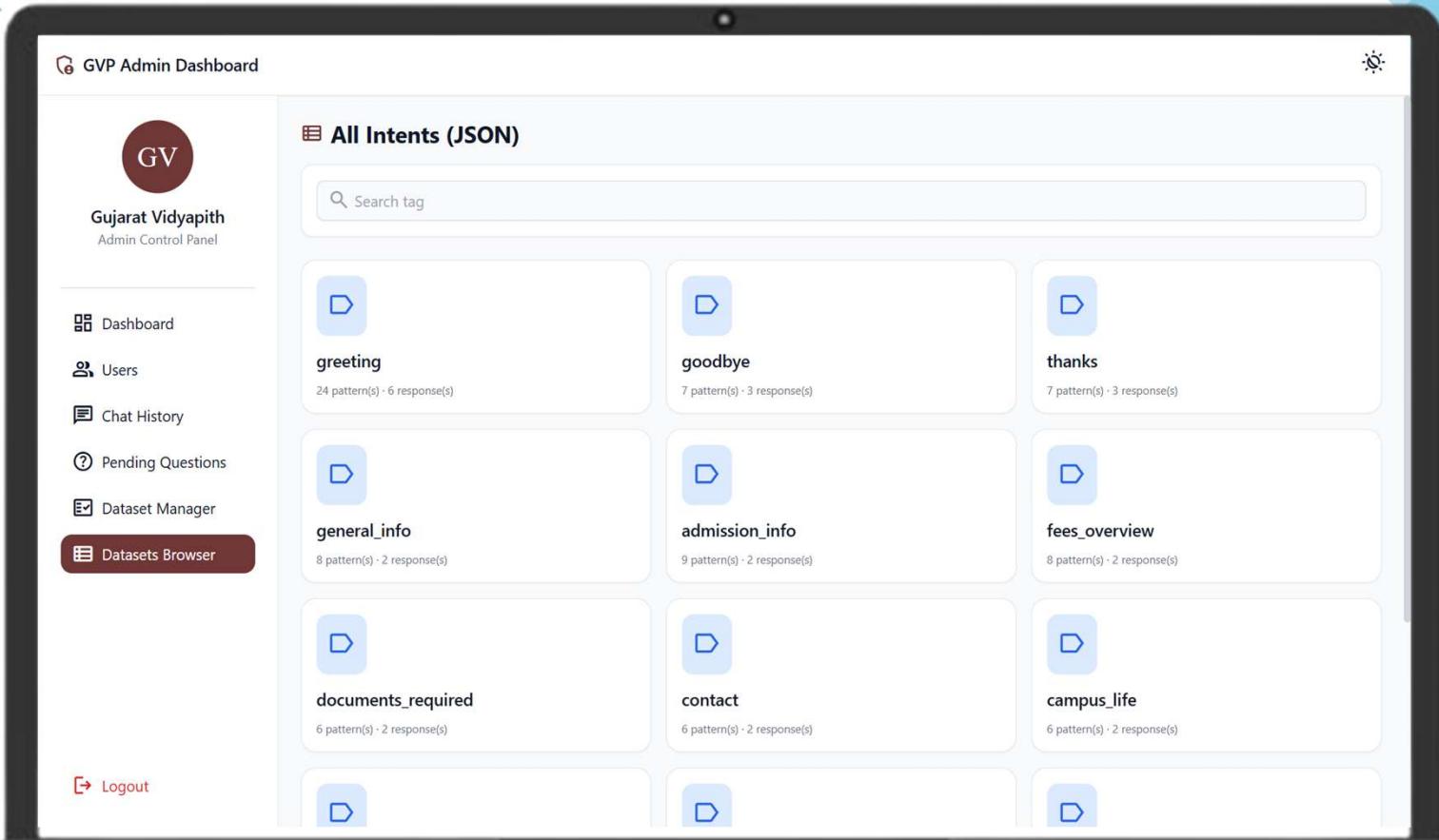


## 5. Pending Questions

The screenshot shows the GVP Admin Dashboard interface. On the left, there's a sidebar with a logo and the text "Gujarat Vidyapith Admin Control Panel". The sidebar includes links for Dashboard, Users, Chat History, Pending Questions (which is currently selected), Dataset Manager, and Datasets Browser. At the bottom of the sidebar is a "Logout" link. The main content area has a header "Pending Questions" with a search bar. A modal window titled "hemal001 - Pending Questions" is open, displaying three pending questions from the user "hemal001" (hemal@gmail.com):

- nearest restaurant from gvp ? (2025-11-23 13:09) with a "Resolve in Intent Manager" button
- nearest hotel from gvp ? (2025-11-23 13:09) with a "Resolve in Intent Manager" button
- Nearest St bus stop name from gvp? (2025-11-23 13:07) with a "Resolve in Intent Manager" button

On the right side of the main content area, there's another user entry for "akshit001" (sonaniakshit@gmail.com) with a status of "1 Pending". The URL in the browser's address bar is "127.0.0.1:8000/admin/intent-manager/?pending\_id=28".



## 6. Datasets Browser

GVP Admin Dashboard

Gujarat Vidyapith  
Admin Control Panel

Dashboard

Users

Chat History

Pending Questions

Dataset Manager

Datasets Browser

Logout

### All Intents (JSON)

Search tag

general\_info

greeting

thanks

fees\_overview

campus\_life

documents\_required

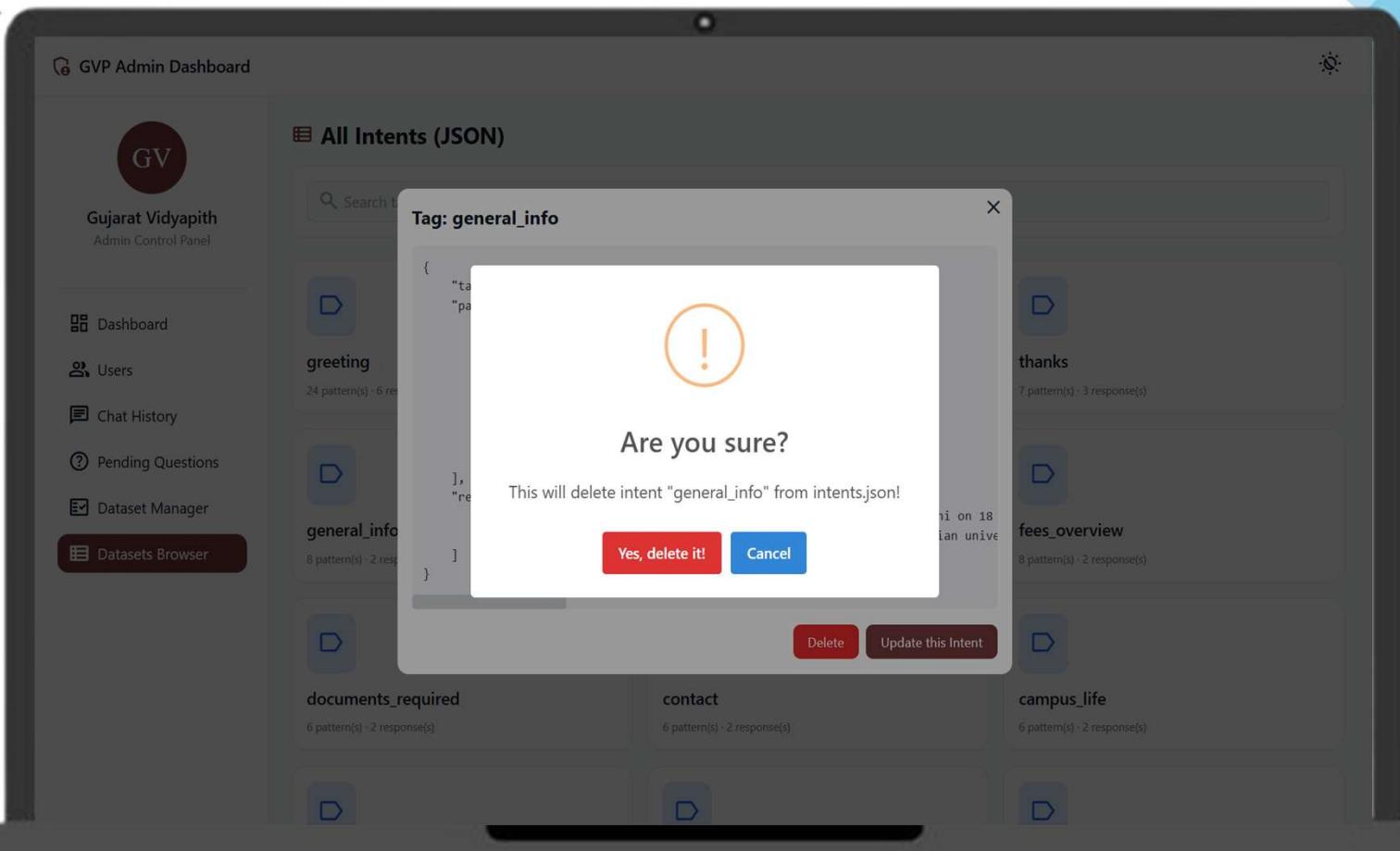
contact

**Tag: general\_info**

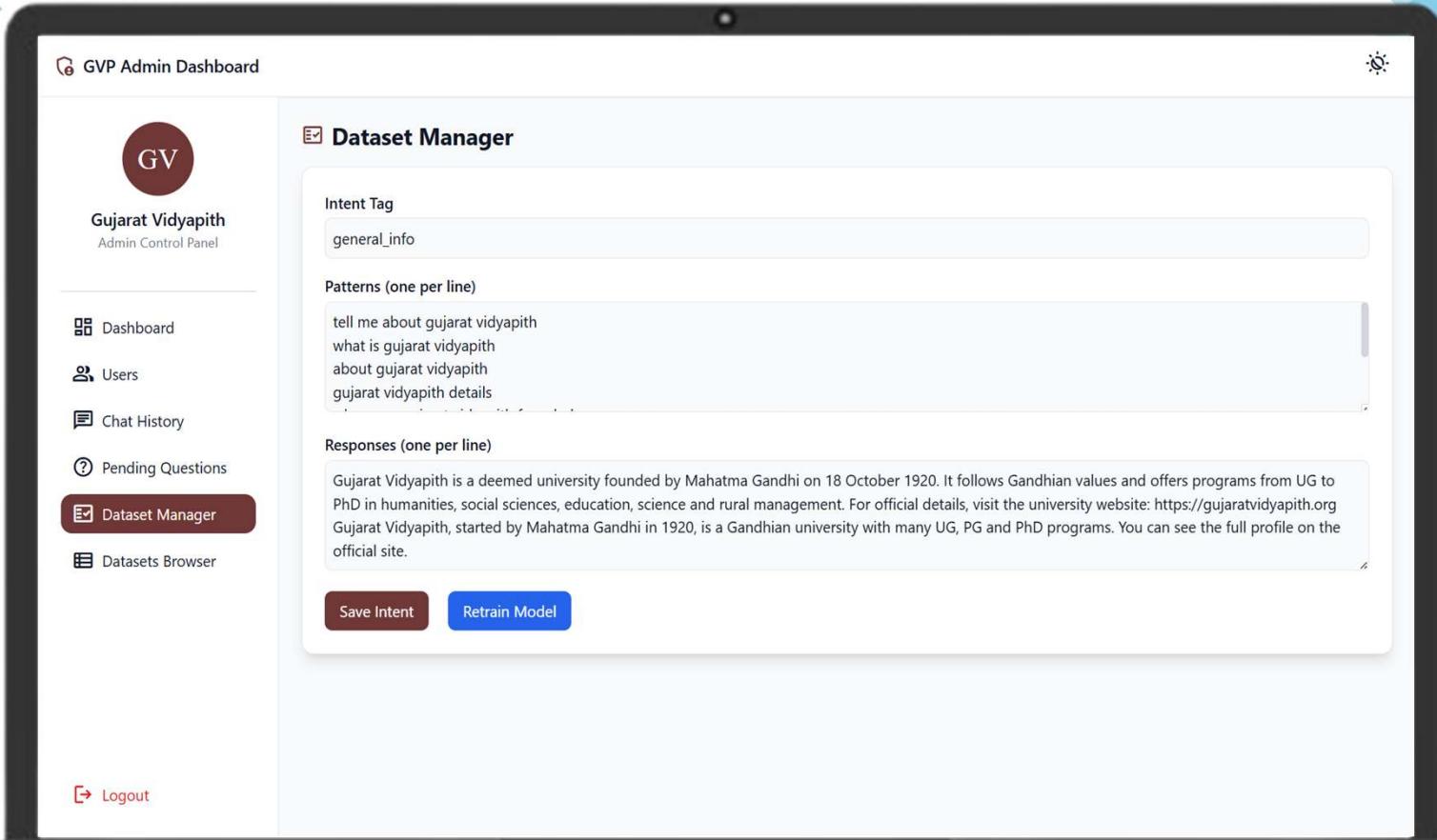
```
{  
  "tag": "general_info",  
  "patterns": [  
    "tell me about gujarat vidyapith",  
    "what is gujarat vidyapith",  
    "about gujarat vidyapith",  
    "gujarat vidyapith details",  
    "when was gujarat vidyapith founded",  
    "who founded gujarat vidyapith",  
    "gvp university information",  
    "gujarat vidyapith profile"  
  ],  
  "responses": [  
    "Gujarat Vidyapith is a deemed university founded by Mahatma Gandhi on 18  
    "Gujarat Vidyapith, started by Mahatma Gandhi in 1920, is a Gandhian unive  
  ]  
}
```

Delete

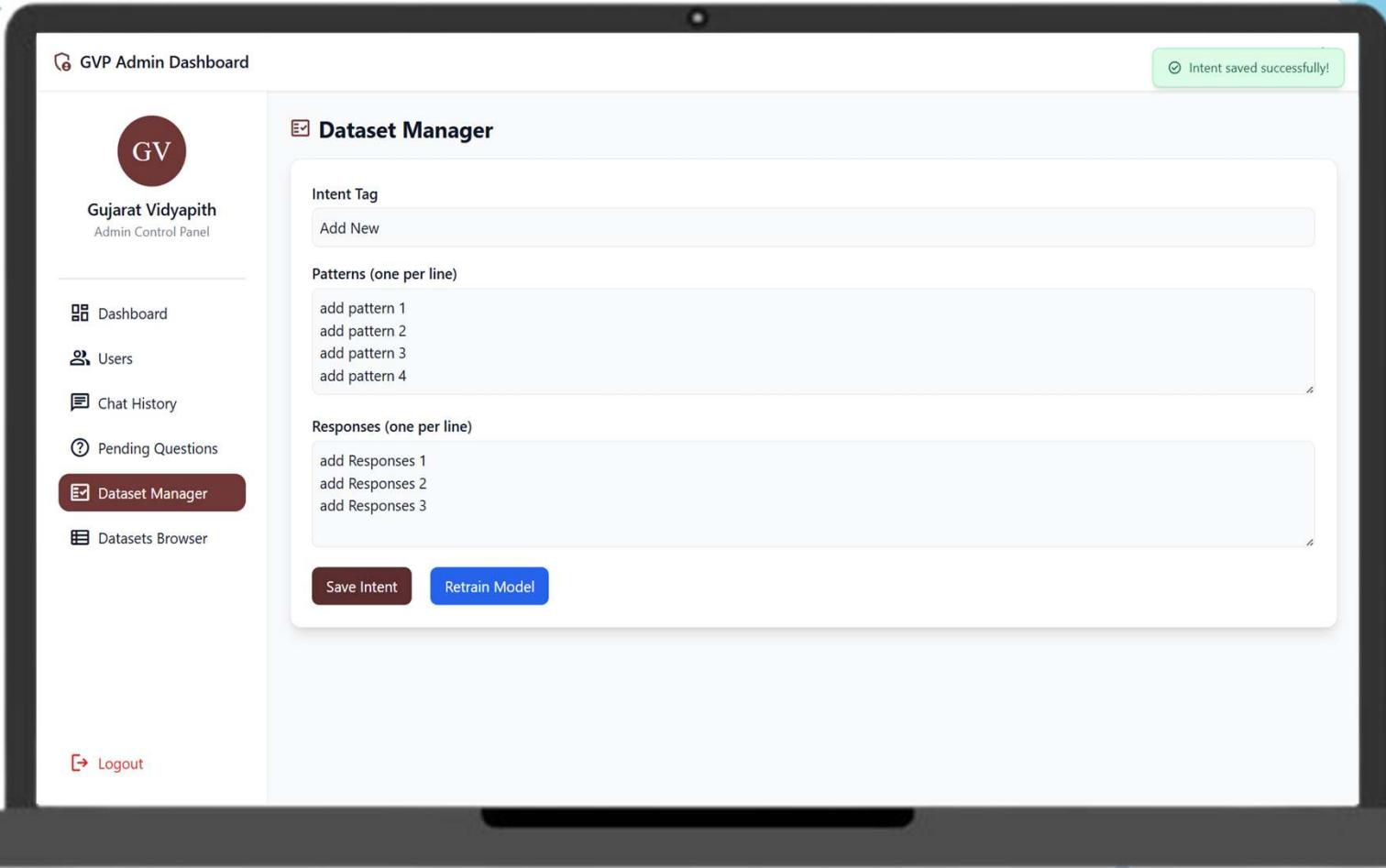
Update this Intent



7.Delete above(general\_info) intent from JSON file



## 8. Update above(general\_info) intent from JSON file



## 9. Add new intent into JSON file

## >User Side Pages

The image shows a tablet screen displaying a user registration form for "GVP University Help Desk AI Registration". The form is titled "Gujarat Vidyapith" and is founded by Mahatma Gandhi. It includes fields for Full Name (Sunny Thakor), Username (sunny001), Email (sunny@gmail.com), Password, Confirm Password (both fields are masked with dots), Gender (Male selected), and Role (Student selected). A red error message "Passwords must match." is displayed below the password fields. A "Register" button is at the bottom, and a "Login" link is provided for existing users.

GVP University Help Desk AI Registration

Gujarat Vidyapith  
Founded by Mahatma Gandhi

Full Name: Sunny Thakor    Username: sunny001

Email: sunny@gmail.com

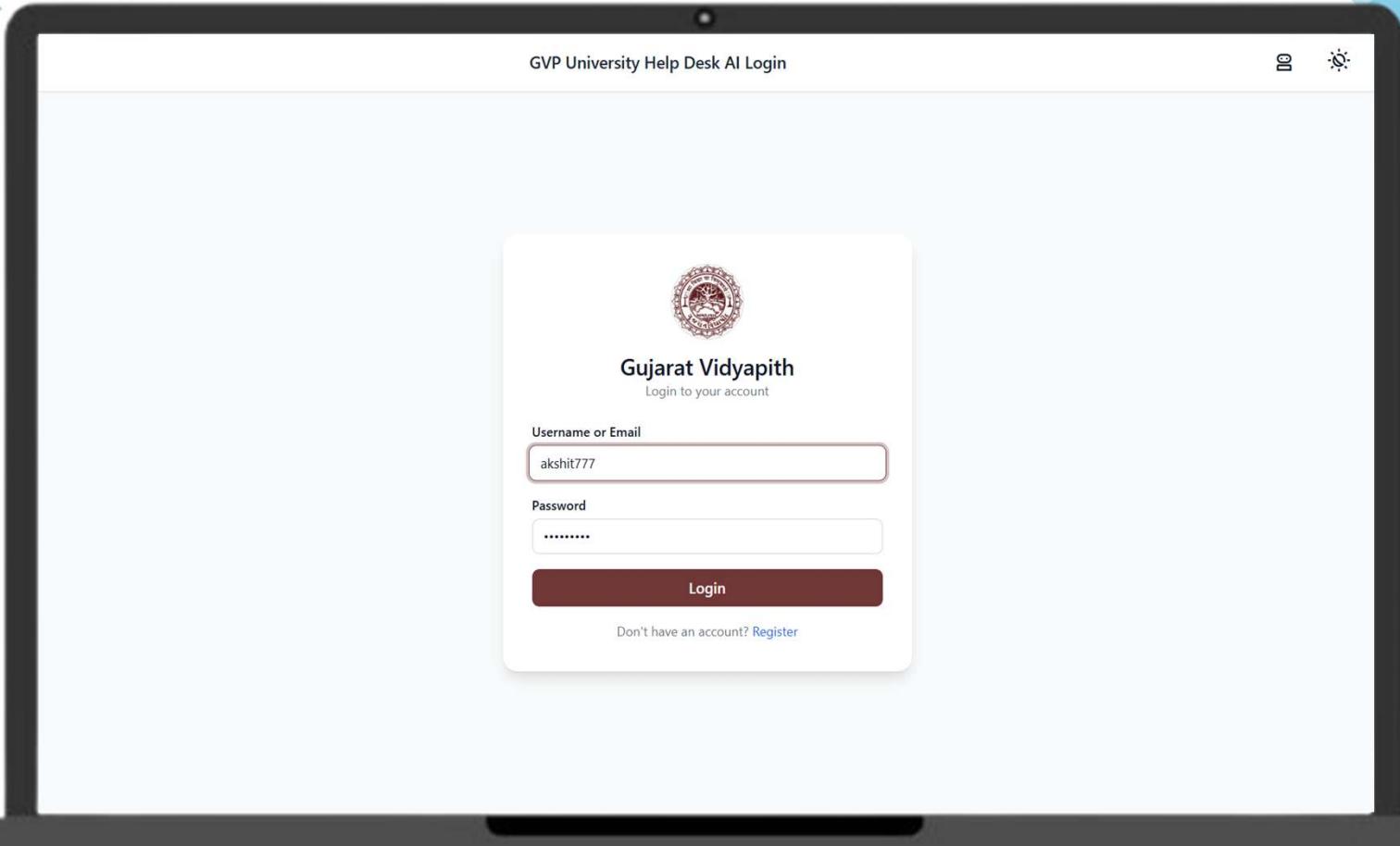
Password: .....    Confirm Password: .....

Gender: Male    Role: Student

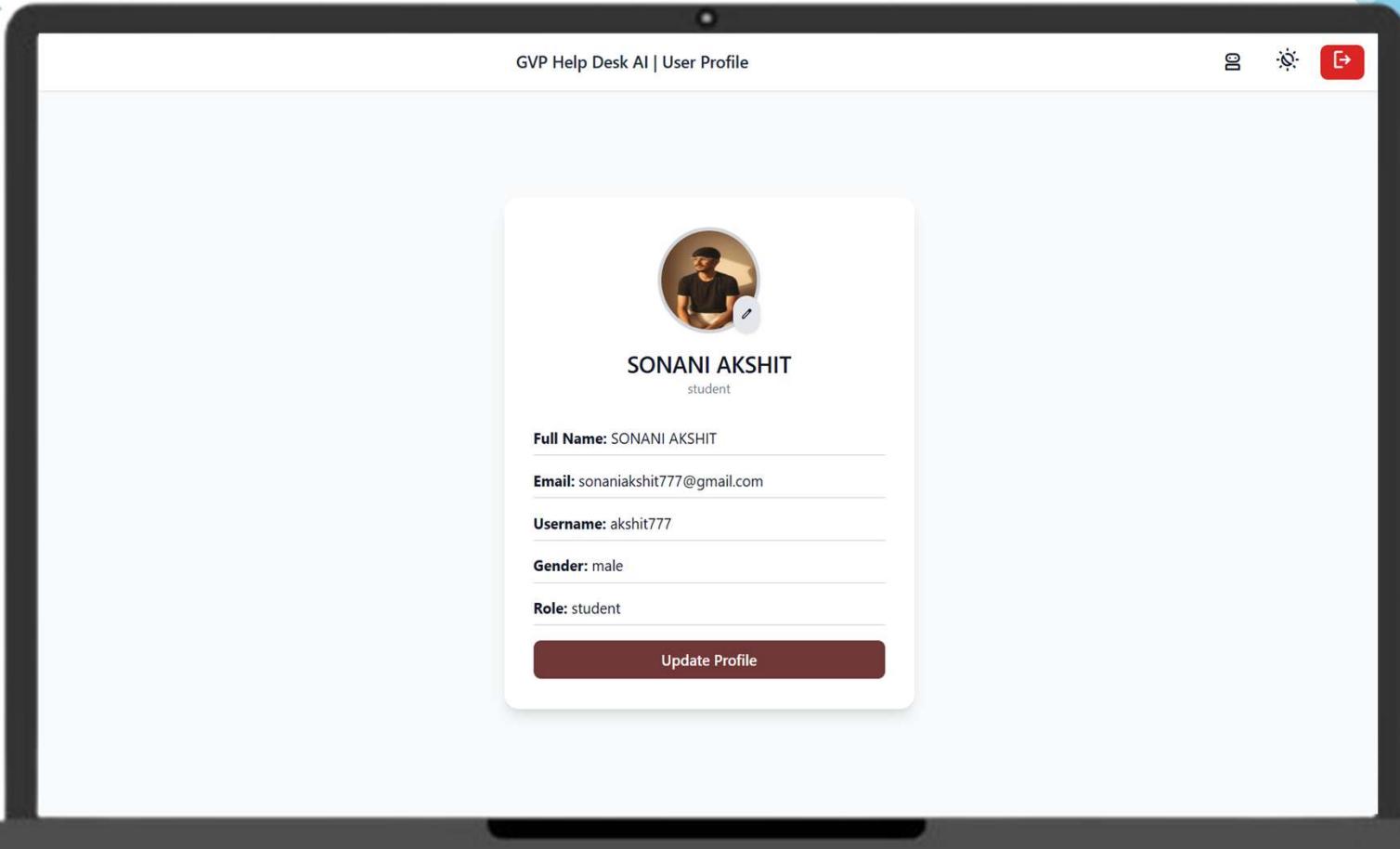
Register

Already have an account? [Login](#)

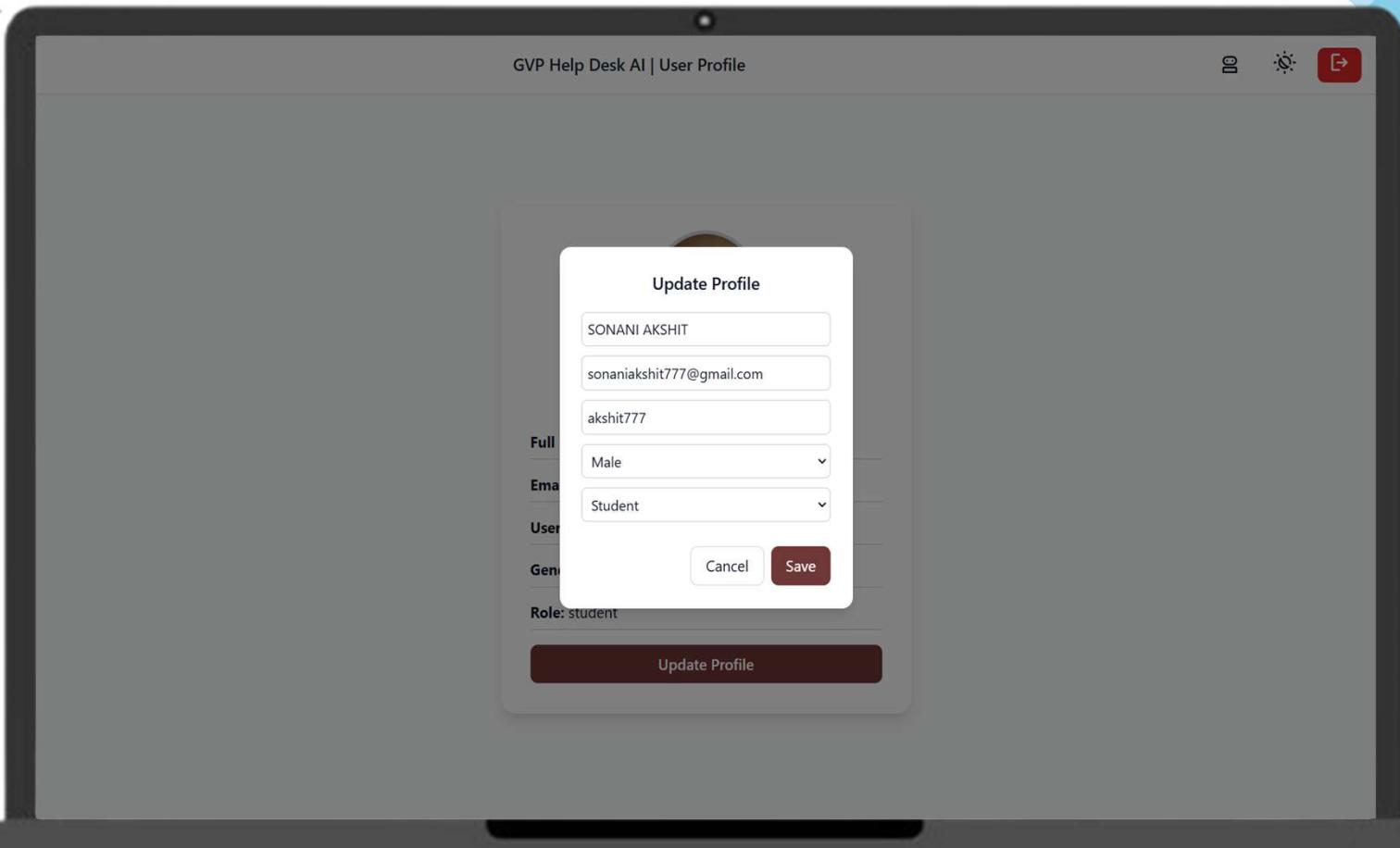
### 1. User Registration



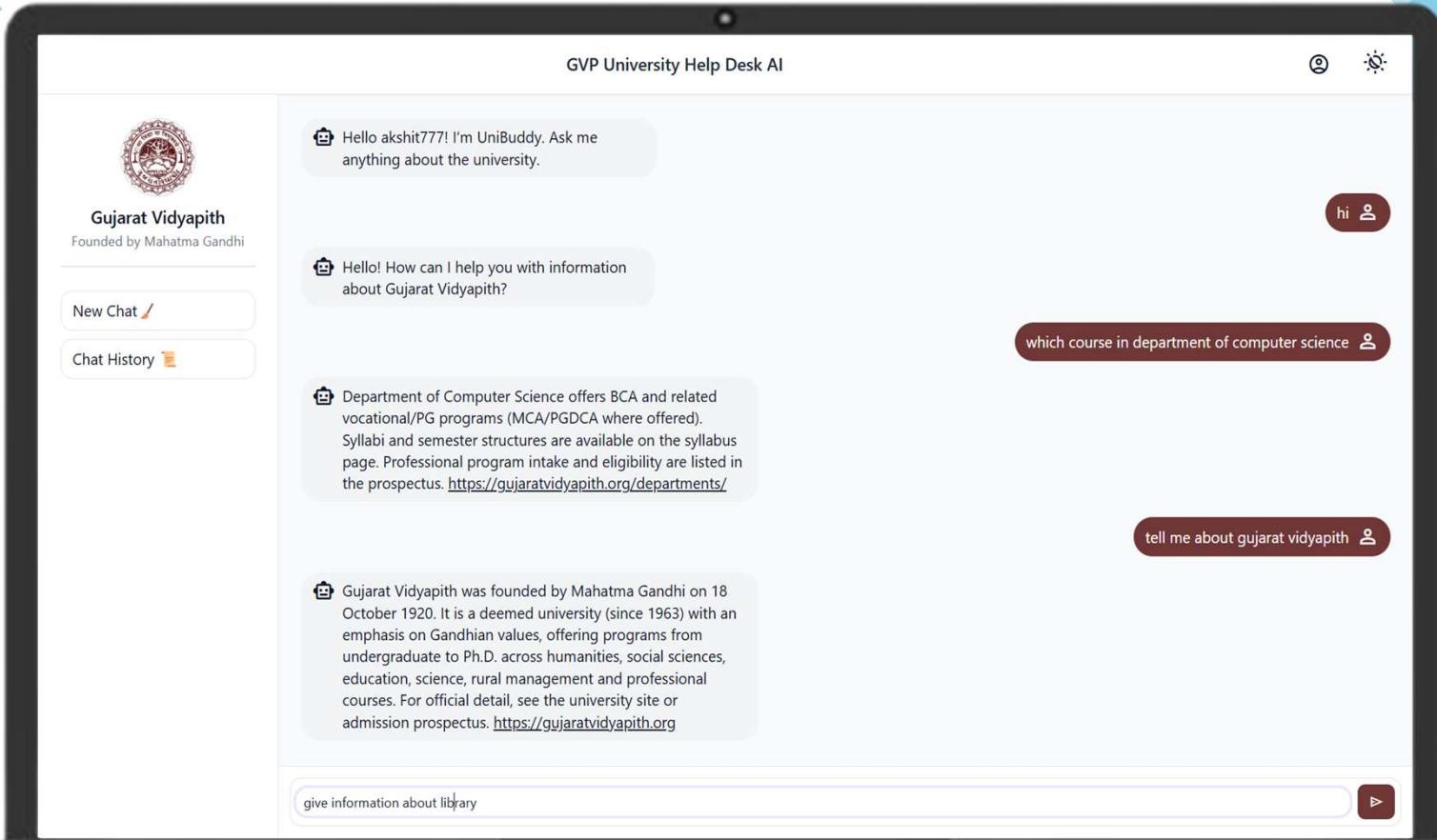
## 2. User Login



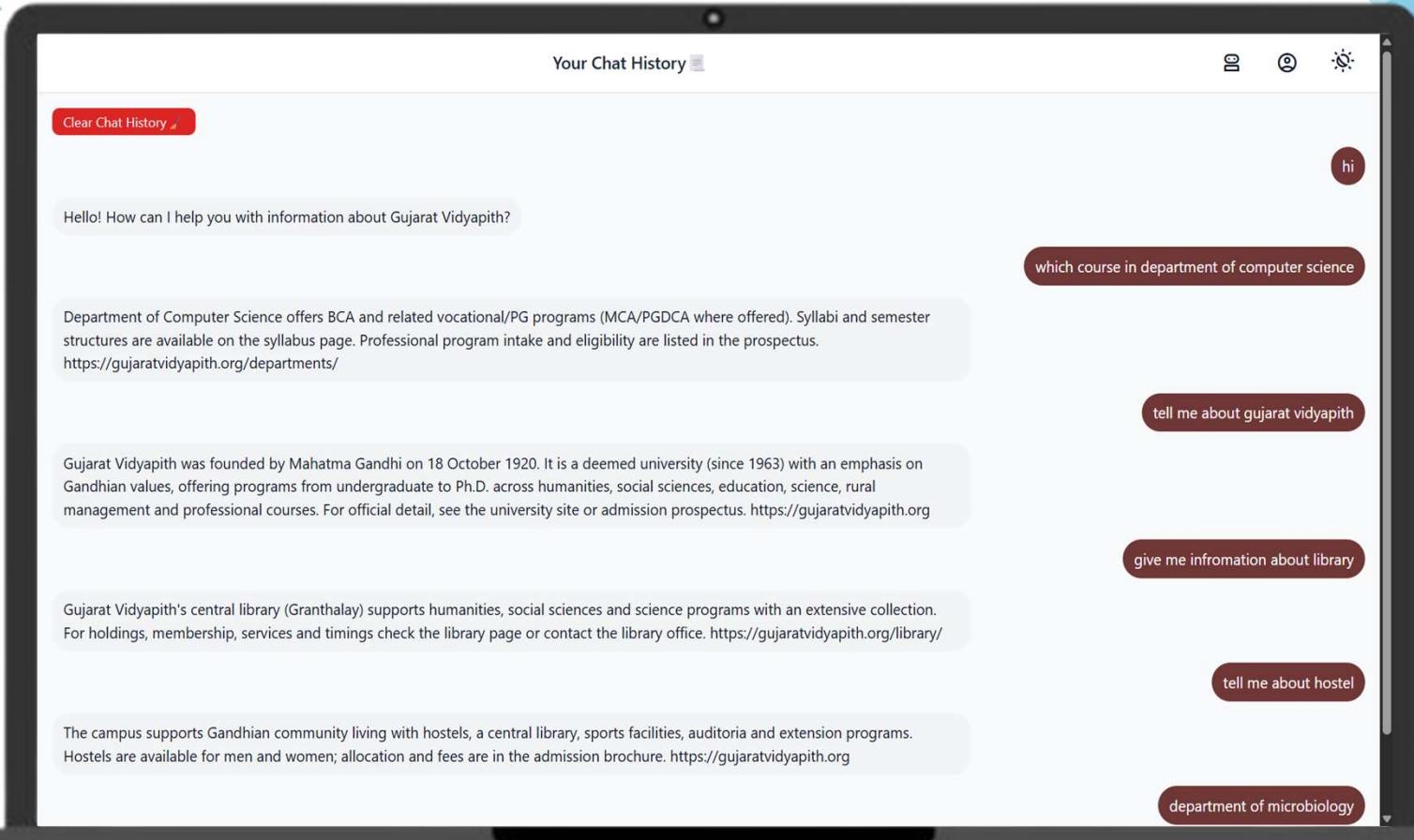
### 3. User Profile



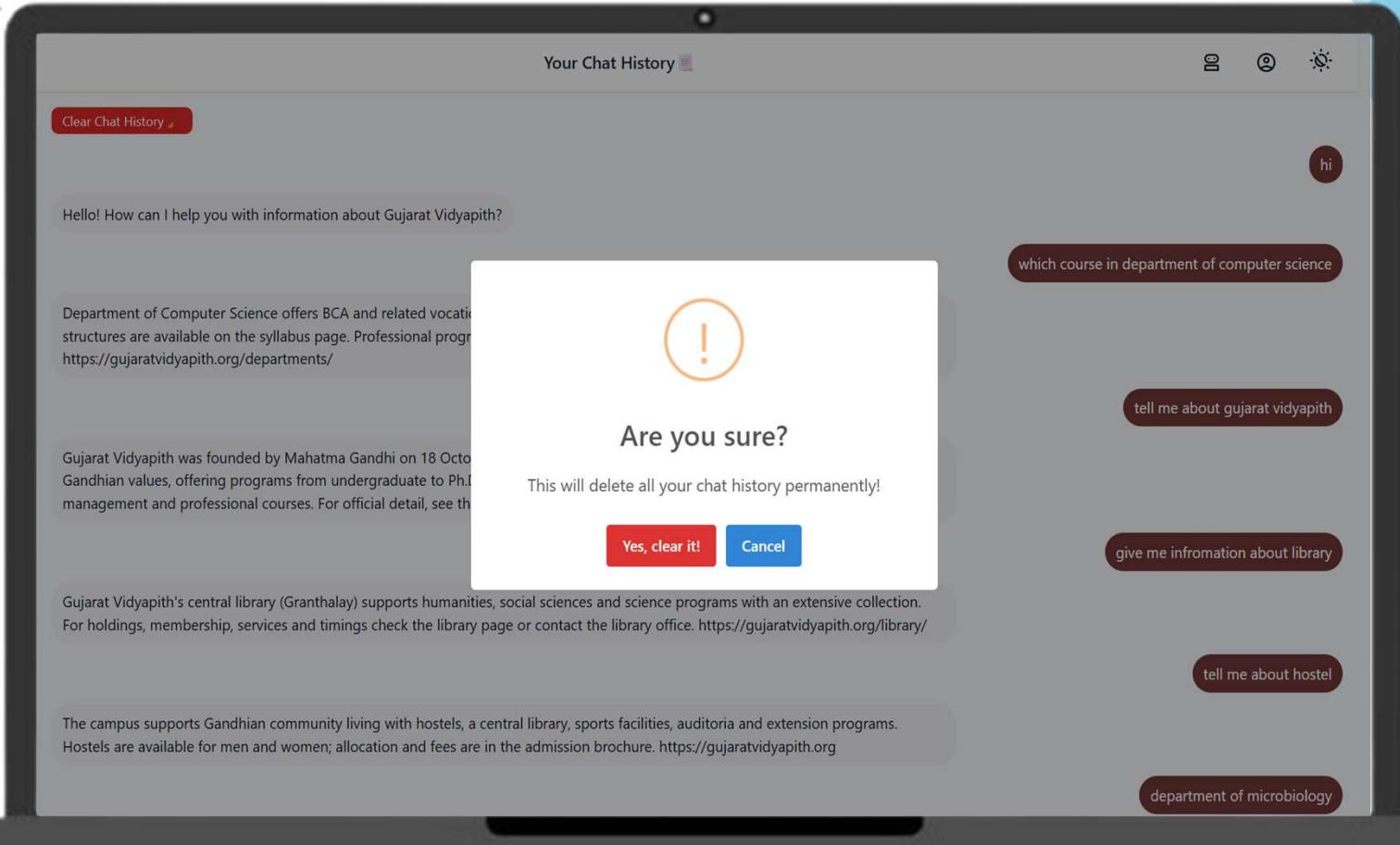
#### 4. Update user profile



## 5. Chatbot Conversation



## 6. User Chat History



## 7. Clear Chat History

# Limitations of the System

---

## **Not a Real-Time or Fully Accurate Source**

The chatbot provides high-level guidance based on predefined data. It cannot give real-time updates such as seat availability, changing fee structures, or last-minute admission rules unless the dataset is manually updated.

## **No Official Decision-Making Ability**

The system cannot make admission decisions, verify documents, allocate hostels, or process applications. It only provides general information and links to official sources.

## **Limited Handling of Complex or Ambiguous Queries**

If the user asks a vague, mixed, or highly detailed question, the chatbot may fail to interpret it correctly and revert to fallback messages or clarification prompts.

## **No Automatic Content Updates**

The system does not crawl the university website or prospectus. When university information changes, the dataset and responses must be updated manually.

## **Language and Input Limitations**

The chatbot performs best with clean English queries. Highly informal messages, spelling errors, mixed languages (like Gujarati + English), or slang can reduce accuracy unless additional training data is provided.

# Proposed Enhancements

---

## **Automatic Model Retraining**

At present, the chatbot is trained manually using a fixed set of intents and responses. In the future, the system can be upgraded to automatically retrain the model based on real chat logs. This would include:

## **Multilingual Support**

Currently, the chatbot primarily supports English. A major enhancement will be adding support for multiple languages such as Gujarati, Hindi and Hinglish.

Future features may include:

- Automatic language detection
- Separate datasets for additional languages
- Translation-based response generation this will make the chatbot more accessible to local users and improve usability for non-English speakers.

## **Voice Interaction**

In future versions, voice input and output can be integrated using speech-to-text and text-to-speech APIs.

This will allow users to communicate with the chatbot verbally, making the system easier to use for visually impaired students or users who prefer voice-based interaction.

# Bibliography

---

- **MongoDB Documentation**

MongoDB Inc. *MongoDB Manual and Developer Guide.*

<https://www.mongodb.com/docs>

- **PyTorch Documentation**

Meta AI Research. *PyTorch Machine Learning Framework Documentation.*

<https://pytorch.org/docs>

- **NLTK Documentation**

Bird, S., Klein, E., & Loper, E. *Natural Language Processing with Python.*

<https://www.nltk.org>

- **Django Documentation**

Django: *MongoDB Connector for Django ORM.*

<https://www.djongomapper.com>

- **University Website (for informational training data)**

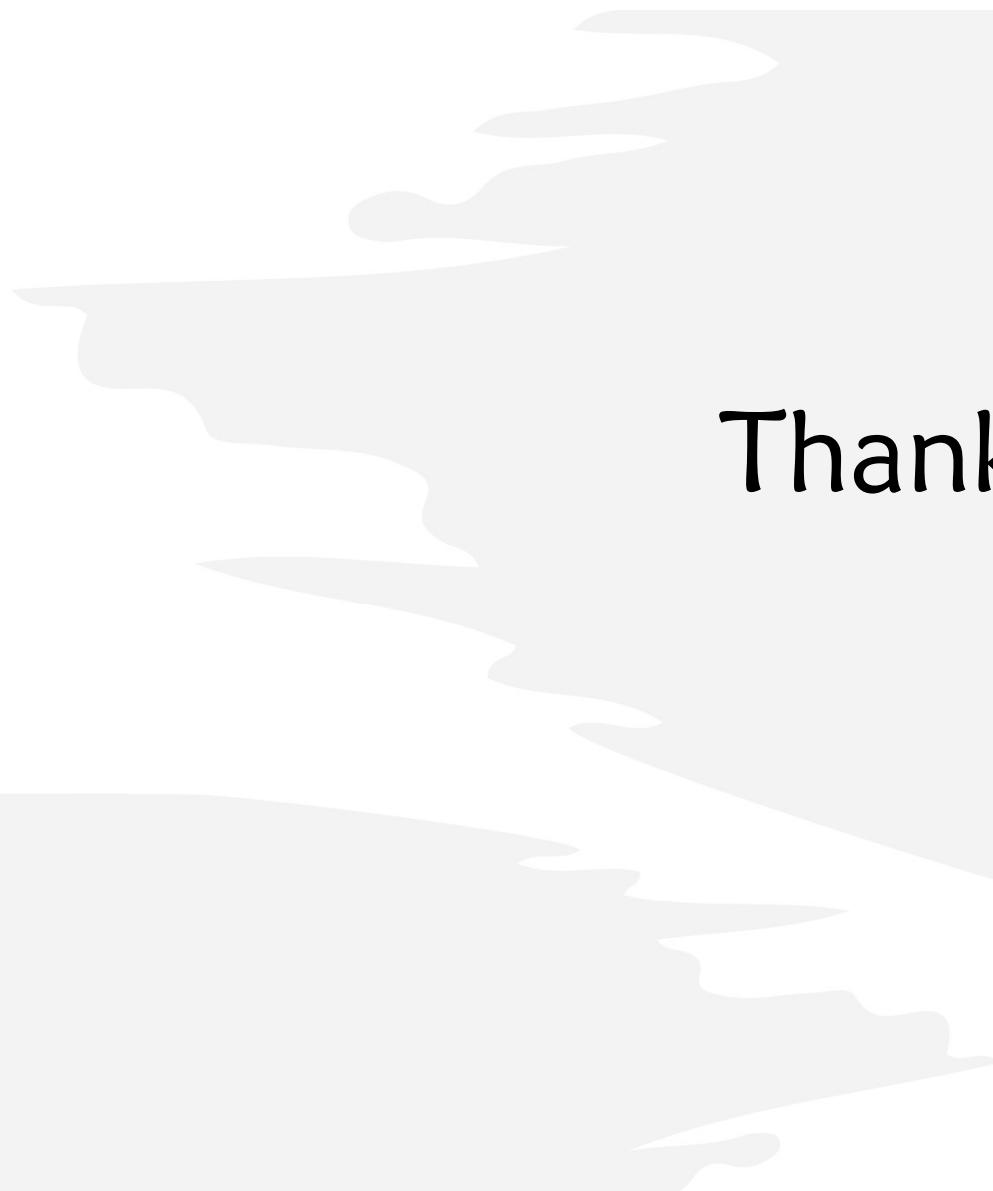
Gujarat Vidyapith. *Official University Website:*

<https://gujaratvidyapith.org>

- **GitHub Repositories (Open Source Chatbot Examples)**

Various developers. *Chatbot implementations using Python, NLP and Django* (referenced for understanding patterns and structure).

<https://github.com>



Thank You...