Ai chatbot

Made by :

Roopesh gowda R [1DA21IS067]

Likhith Venugopal [1DA21IS065]

Abhishek C R [1DA21IS068]

Varalakshmi M [1DA22IS405]

Submitted to :

Mrs. Malathi P

Assistant Professor Dept of ISE

Dr Ambedkar Institute of Technology

📘 Project Overview

This project involves the design and development of an AI-powered chatbot using **Dialogflow** for **Dr. Ambedkar Institute of Technology (DAIT)**. The chatbot is intended to serve as a virtual assistant that provides instant answers to frequently asked questions by students, faculty, and visitors. It acts as a first-line support system for resolving queries related to academic programs, departments, admissions, campus facilities, placements, and more.

The chatbot uses natural language understanding (NLU) to interpret user queries and provide relevant, conversational responses. It eliminates the need for manual intervention in handling repetitive queries and helps users get quick, accurate information at any time. The system can be accessed through a web-based interface and can be further expanded for integration with platforms like WhatsApp or Telegram.

By automating the process of answering queries, the chatbot enhances user experience, reduces workload on administrative staff, and showcases the institution's adoption of intelligent digital solutions.

🎯 Objectives

The primary objectives of this project are:

1. **To build an intelligent virtual assistant** using Dialogflow that can interpret and respond to natural language queries related to the institution.
2. **To streamline the process of information dissemination** at Dr. Ambedkar Institute of Technology by automating frequently asked questions and student support.
3. **To reduce administrative overhead** by minimizing the number of routine queries that require human response.
4. **To ensure 24/7 accessibility** of institutional information to students, parents, and other stakeholders through a user-friendly web interface.
5. **To demonstrate practical application of AI and NLP** concepts using a real-time project relevant to the academic community.
6. **To provide a scalable foundation** that can later be extended with advanced features such as voice support, multilingual capabilities, and backend database integration.

🛠️ Tools and Technologies Used

The following tools and technologies were used in the development and deployment of the AI Chatbot:

Tool/Technology	**Purpose**
Dialogflow ES (by Google)	To create the chatbot’s conversational agent using intents and entities.
Google Cloud Platform (GCP)	Backend infrastructure support for Dialogflow and optional webhook deployment.
Webhook (Node.js / Python)*(Optional)*	For dynamic responses, data fetching, and custom logic integration.
Firebase *(Optional)*	To host webhook functions and store/retrieve data like FAQs or user logs.
HTML, CSS, JavaScript	To design and embed the chatbot within the college website or a standalone interface.
GitHub	Version control and collaborative development (if used).
VS Code / Any IDE	For code editing and development of backend services.

🧭 System Architecture

The overall system follows a modular, cloud-based architecture to ensure scalability and ease of integration. Here's a breakdown of the architecture:

🔁 Architecture Workflow:

```
pgsql
CopyEdit
User (Student/Visitor)
        ↓
Chat Interface (Website or App)
        ↓
Dialogflow Agent
        ↓
[ Intents + Entities ]
        ↓
(If needed)
Webhook / Firebase Functions
        ↓
Custom Responses or External API Data
        ↓
Response Sent Back to User
```

🔹 Components Explained:

1. **User Interface**
    - A web-based chat window or embedded iframe where users type their queries.
    - Optional integration with platforms like WhatsApp, Telegram, or mobile apps.
2. **Dialogflow Agent**
    - The core NLU engine that identifies user intents (e.g., "What is the admission process?").
    - Uses **intents**, **entities**, and **training phrases** to generate relevant responses.
3. **Intents & Entities**

- **Intents** represent user goals (e.g., asking about departments, placements, contact info).
- **Entities** extract specific data from user inputs (e.g., "CSE", "B.Tech").

4. **Webhook/Fulfillment (Optional)**
    - Custom backend service to handle advanced logic.
    - Used when responses depend on real-time data (e.g., current notices, event dates).
5. **Firebase/Google Cloud (Optional)**
    - Cloud functions or databases to store FAQs, manage logs, or scale webhook functionality.
6. **Response Delivery**
    - The agent sends an appropriate reply back to the user, either directly or through webhook logic.

⚙️ Implementation Steps

The chatbot was implemented in a structured manner following the below steps:

🔷 Step 1: Agent Creation in Dialogflow

- Logged into Dialogflow Console.
- Created a new agent named `DAIT_Chatbot`.
- Set language to English, and time zone to IST (India Standard Time).
- Linked the agent to a Google Cloud Project.

🔷 Step 2: Intent Design

- Defined multiple **intents** to capture different categories of user queries such as:
    - **Welcome Intent** – Greets the user when the chat starts.
    - **Admission Info Intent** – Provides details on admission process.
    - **Department Info Intent** – Gives information about departments in DAIT.
    - **Placement Intent** – Shares placement-related statistics or FAQs.
    - **Contact/Location Intent** – Shares the contact details and college location.

🔷 Step 3: Entity Creation

- Created **custom entities** to extract meaningful data from user inputs.
    - `@department_names` – E.g., “Computer Science”, “Mechanical”, “Civil”
    - `@course_type` – E.g., “B.E.”, “M.Tech”, “Diploma”
- Used **system entities** like `@sys.date` or `@sys.number` where applicable.

🔹 Step 4: Training Phrases and Responses

- Added multiple **training phrases** to teach the agent how users may phrase their queries.
- Set up **static text responses** for each intent (e.g., "The admission process for B.E. includes KCET/COMED-K...").

🔹 Step 5: Fulfillment (Optional)

- Enabled fulfillment for dynamic intents.
- Connected a webhook built using **Node.js** and deployed on **Firebase Functions** or **Glitch**.
- The webhook fetched real-time data such as upcoming events or notices from an external source.

🔹 Step 6: Integration with Web Interface

- Embedded the chatbot into the DAIT website using the Dialogflow web demo iframe:

```
html
CopyEdit
<iframe
  allow="microphone;"
  width="350"
  height="430"
  src="https://console.dialogflow.com/api-
client/demo/embedded/YOUR_AGENT_ID">
</iframe>
```

✅ Features Implemented

The chatbot is equipped with several features aimed at improving user interaction and information accessibility:

Feature	Description
Welcome Greeting	Automatically welcomes users to the DAIT chatbot interface.
Admission Queries	Answers questions about eligibility, entrance exams, and application processes.
Department Details	Provides information about courses, faculty, and infrastructure of each department.
Placement Support	Shares placement stats, recruiters list, and preparation tips.
Contact & Location Info	Shares official college contact, location on map, and email.
Fallback Handling	Uses fallback intent when the bot doesn't understand the user query.
Scalable Architecture	Can be extended to support voice input, multilingual responses, and external API integration.

🧪 Testing and Validation

To ensure the chatbot functions reliably and provides accurate responses, thorough testing was conducted across various parameters. The following testing procedures were followed:

🔷 1. Functional Testing

- Verified that all intents responded correctly to expected queries.
- Each intent was tested with a variety of **training phrases**, including typos, synonyms, and variations in sentence structure.
- Confirmed appropriate triggering of fallback intent when queries did not match any defined intents.

🔷 2. Entity Extraction Validation

- Tested whether custom and system entities were correctly extracted from user inputs.
- Example: "Tell me about **Mechanical Engineering**" correctly identified `@department_names`.

🔷 3. Web Integration Testing

- Verified that the embedded chatbot on the website worked across major browsers (Chrome, Firefox, Edge).
- Tested responsiveness and usability on desktop and mobile interfaces.

🔷 4. Fulfillment Testing (if Webhook used)

- Checked dynamic responses from webhook by simulating backend queries.
- Validated timeout handling and error messaging if webhook services were unreachable.

🔷 5. User Acceptance Testing (UAT)

- Collected feedback from a sample group of students and faculty.
- Made improvements based on suggestions (e.g., adding placement-related queries, refining training phrases).

⚠️ Limitations

While the chatbot meets its core objectives, certain limitations currently exist:

🚫 1. Limited Context Handling

- Dialogflow ES offers only basic context management, limiting the chatbot’s ability to handle multi-turn conversations smoothly.
- For more advanced, flowing dialogues, upgrading to Dialogflow CX is recommended.

🌐 2. Requires Internet Access

- The chatbot is hosted on cloud services, requiring continuous internet connectivity for both users and the backend (if used).

🧠 3. Static Responses for Most Intents

- Many intents use pre-defined static responses. For real-time or frequently updated content (like announcements), backend integration or CMS is needed.

💬 4. Language Limitation

- The current version supports English only. Support for regional languages (Kannada, Hindi) could enhance accessibility.

🔒 5. No User Authentication

- The chatbot does not authenticate users, which may limit personalization and restrict access to sensitive student information (e.g., internal marks).

🚀 Future Enhancements

While the chatbot is functional and serves the basic needs of the users, there are several opportunities to enhance its capabilities in future versions:

🔷 1. Upgrade to Dialogflow CX

- Migrate to Dialogflow CX for handling complex and multi-turn conversations.
- Offers better flow control, state management, and visual representation of dialogue.

🔷 2. Multilingual Support

- Implement support for regional languages like **Kannada** and **Hindi** to improve accessibility for local users.

🔷 3. Voice Integration

- Enable voice-based interaction using **Google Assistant** or **Speech-to-Text API**, especially useful for mobile users or visually impaired students.

🔷 4. Integration with College Database

- Connect to institutional databases to fetch dynamic, real-time data (e.g., exam schedules, student records, notices).
- Requires authentication and backend security.

🔷 5. Feedback Collection System

- Allow users to rate responses or give feedback to improve training data and identify unanswered queries.

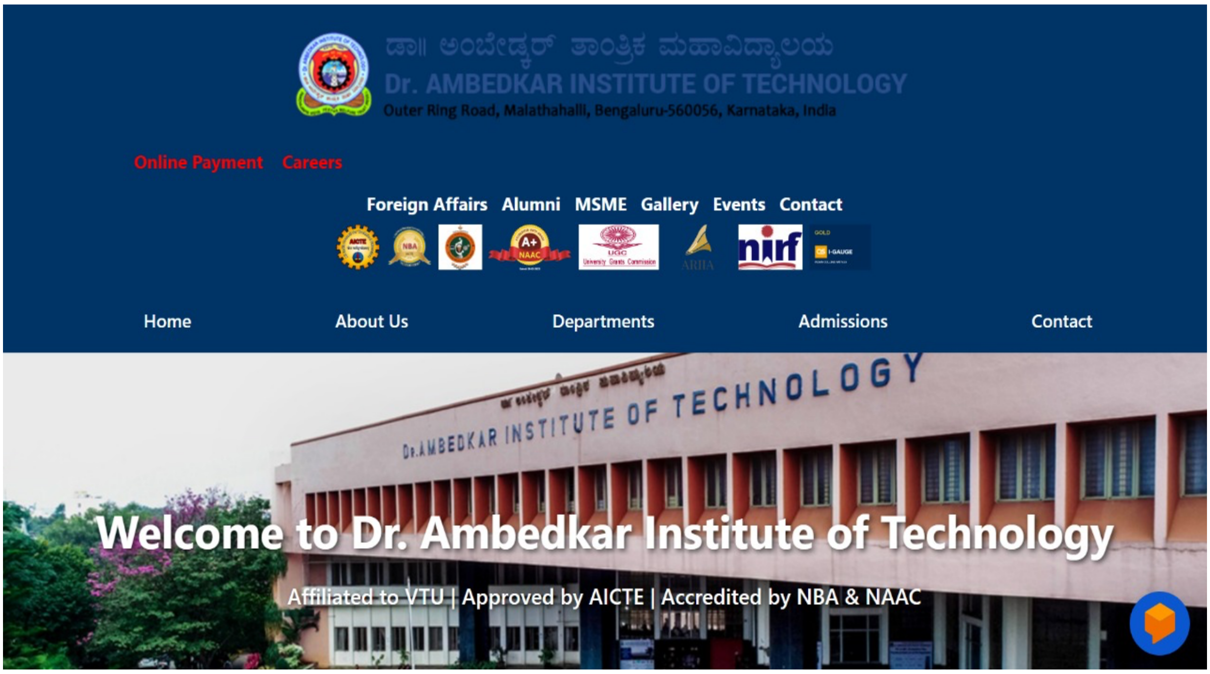
🔷 6. Mobile App Integration

- Integrate the chatbot within a college mobile app for broader accessibility and push notification support.

🔷 7. Analytics Dashboard

- Implement usage analytics (number of users, common queries, unhandled intents) to monitor performance and make data-driven improvements.

Screenshots :

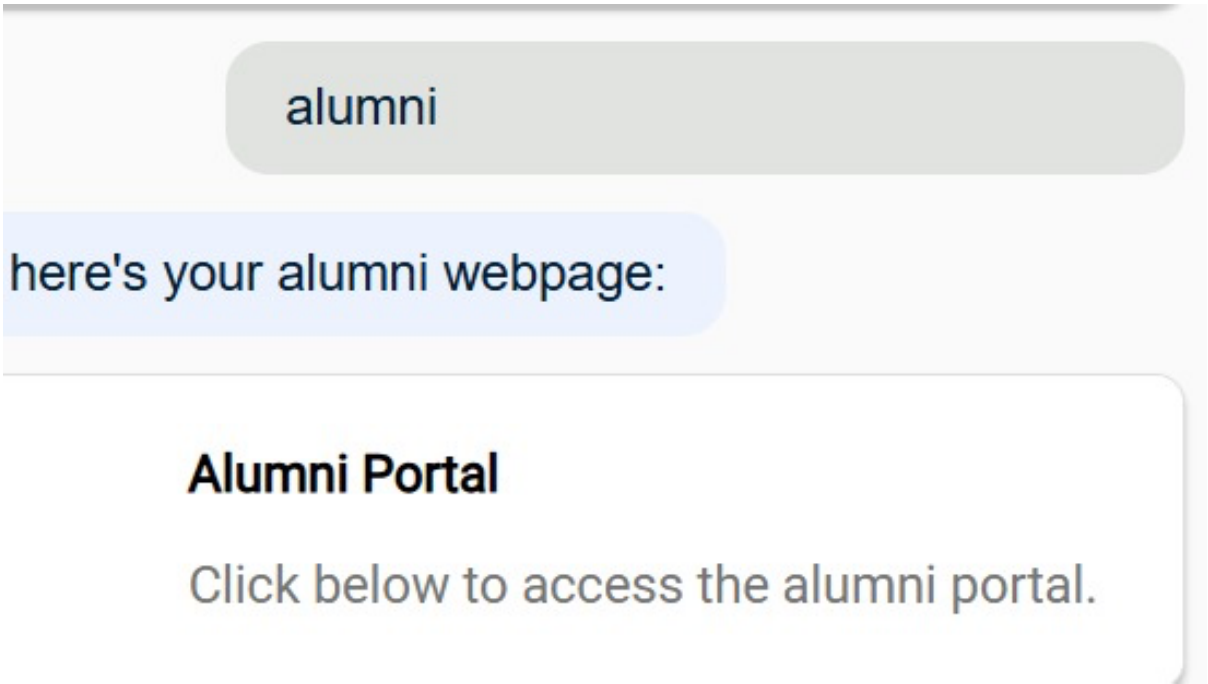

ಡಾ॥ ಅಂಬೇಡ್ಕರ್ ತಾಂತ್ರಿಕ ಮಹಾವಿದ್ಯಾಲಯ

Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY

Outer Ring Road, Malathahalli, Bengaluru-560056, Karnataka, India

Welcome to Dr. Ambedkar Institute of Technology

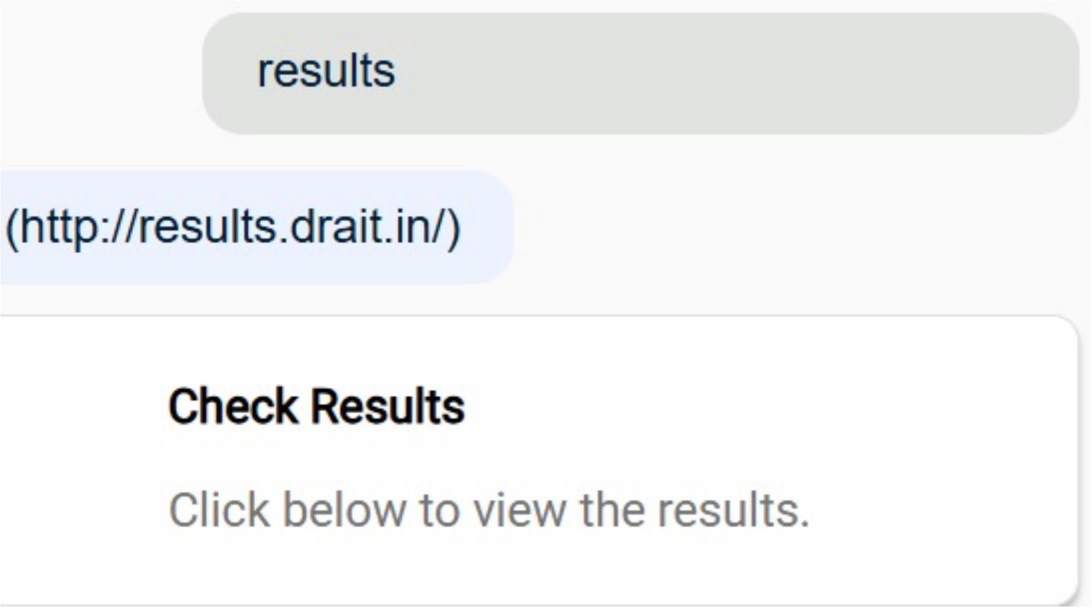
Affiliated to VTU | Approved by AICTE | Accredited by NBA & NAAC


alumni

here's your alumni webpage:

Alumni Portal

Click below to access the alumni portal.

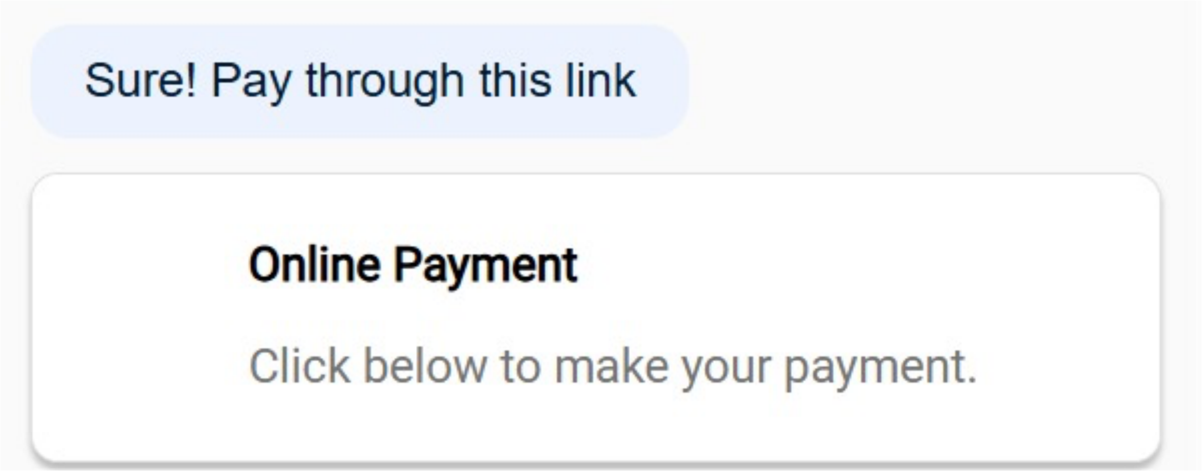


results

(http://results.drait.in/)

Check Results

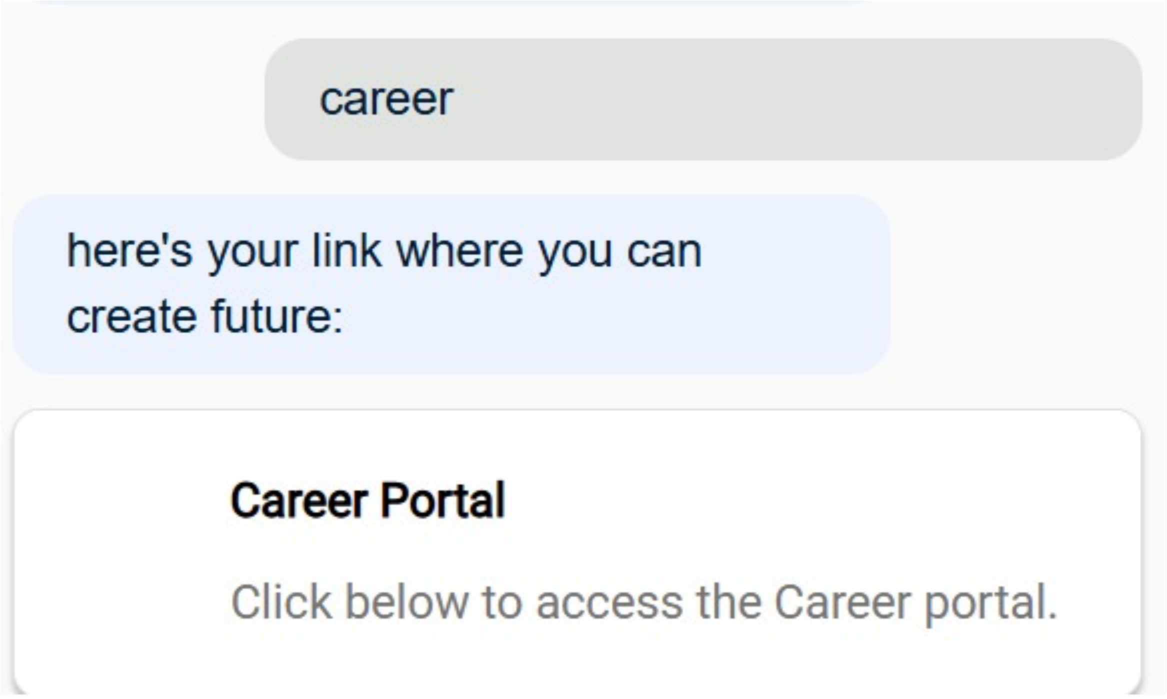
Click below to view the results.



Sure! Pay through this link

Online Payment

Click below to make your payment.

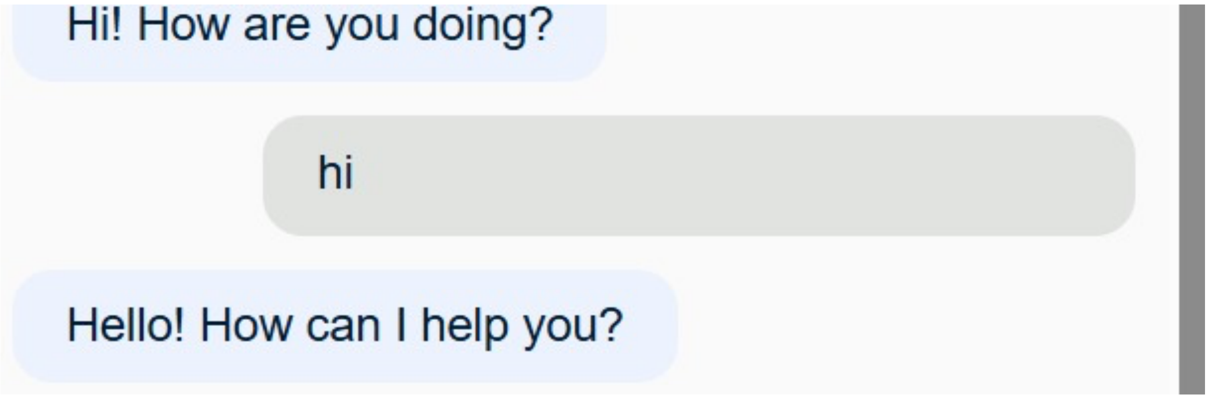


career

here's your link where you can create future:

Career Portal

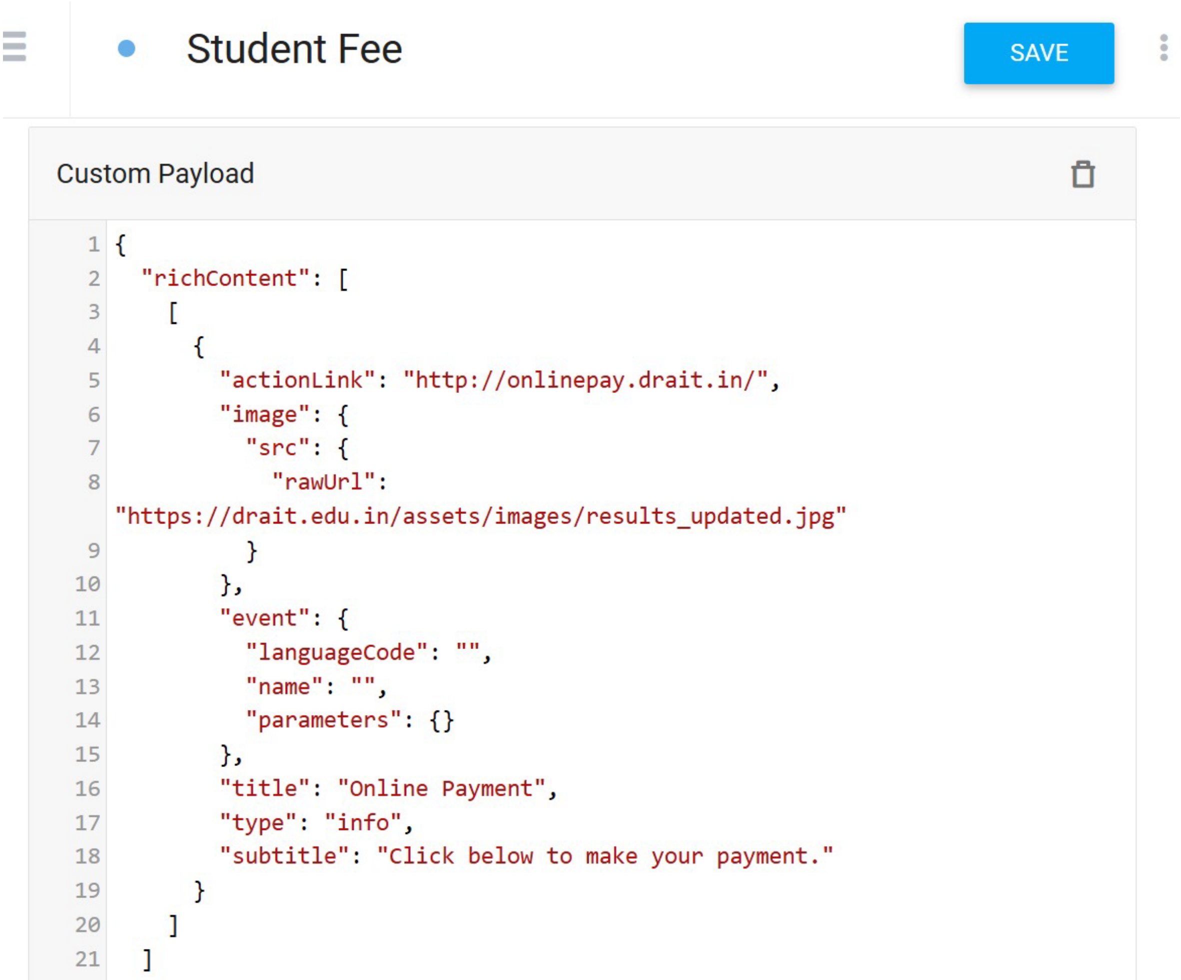
Click below to access the Career portal.



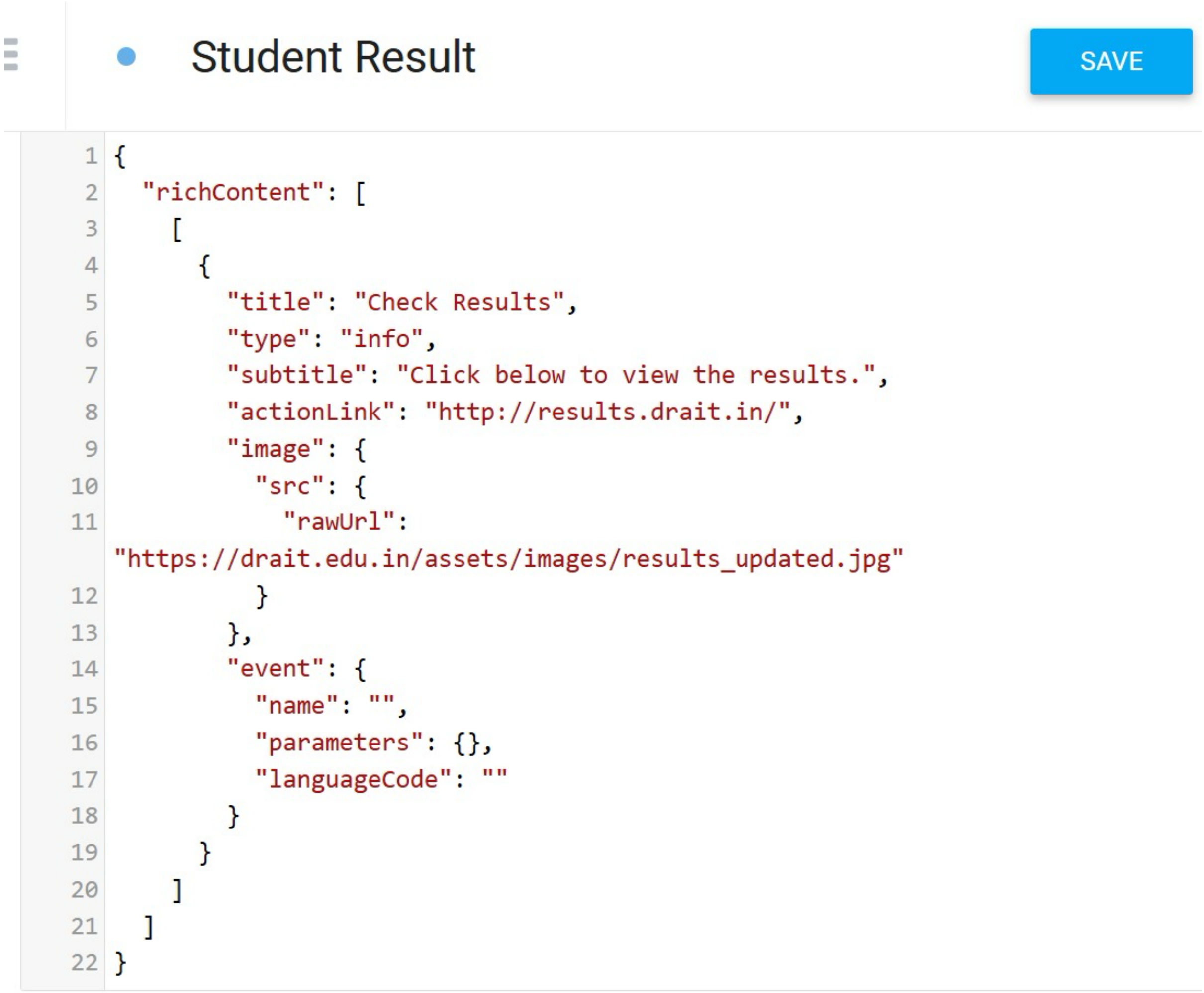
Hi! How are you doing?

hi

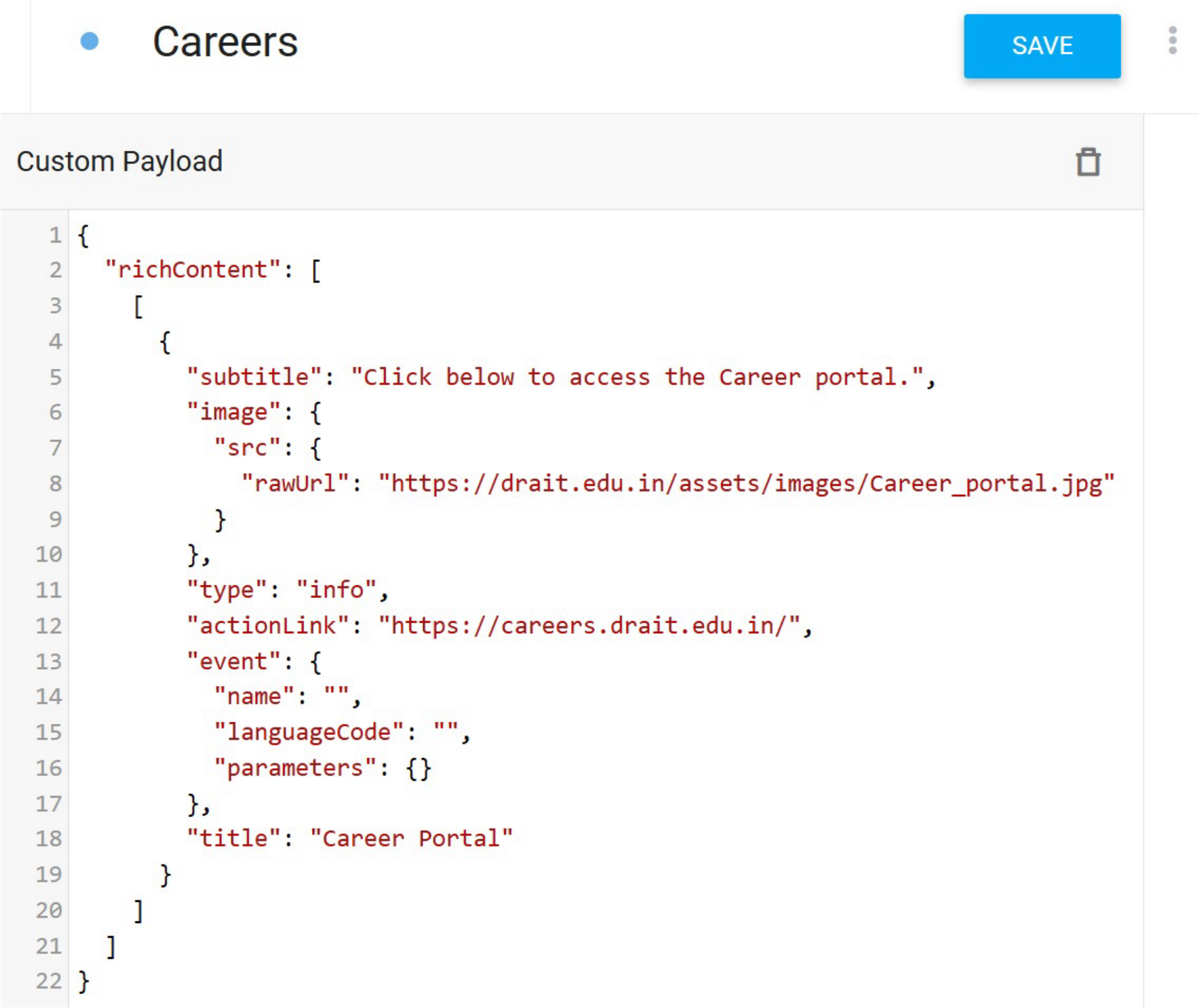
Hello! How can I help you?



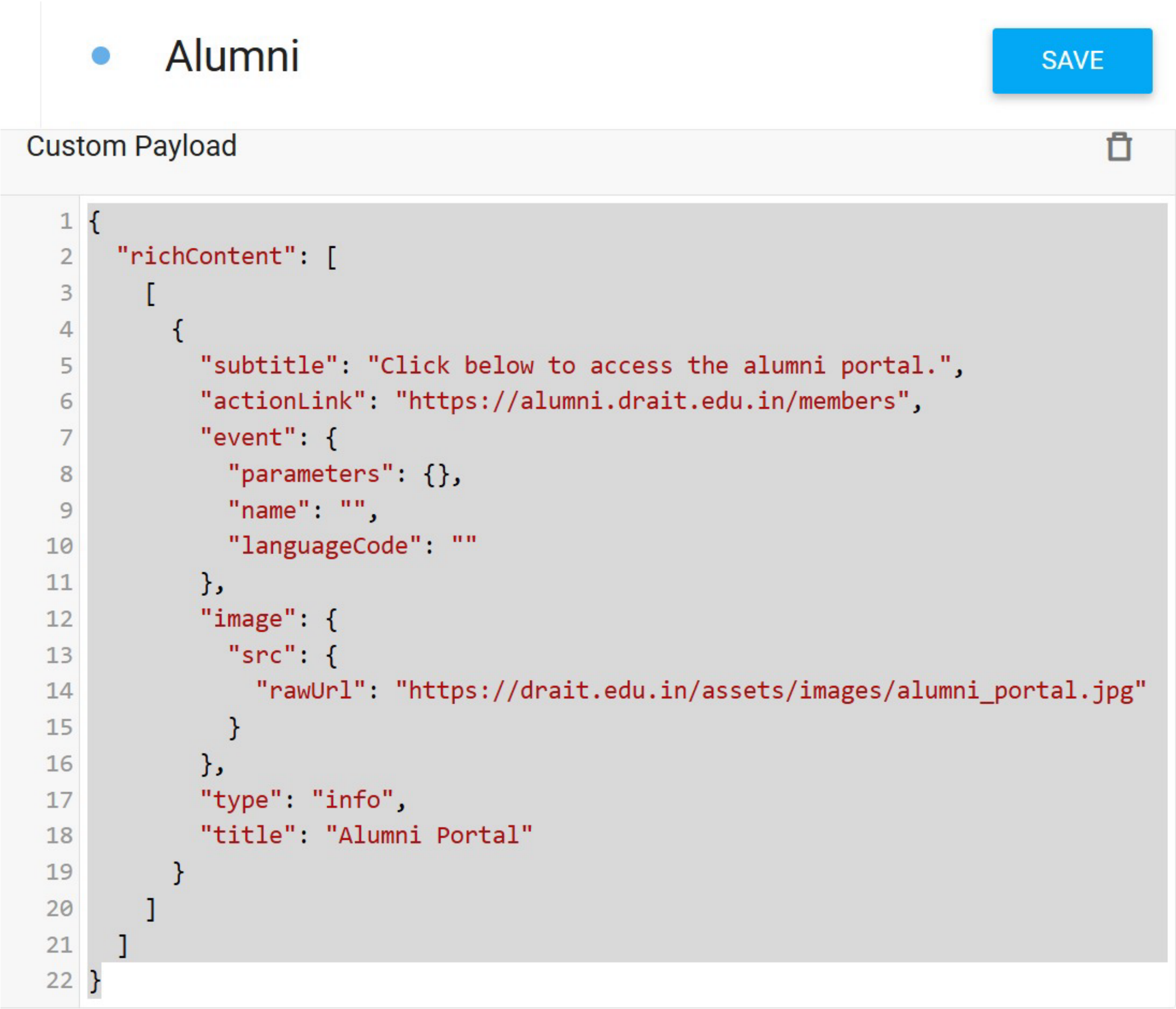
```
{
  "richContent": [
    [
      {
        "actionLink": "http://onlinepay.drait.in/",
        "image": {
          "src": {
            "rawUrl":
"https://drait.edu.in/assets/images/results_updated.jpg"
          }
        },
        "event": {
          "languageCode": "",
          "name": "",
          "parameters": {}
        },
        "title": "Online Payment",
        "type": "info",
        "subtitle": "Click below to make your payment."
      }
    ]
  ]
```



```
{
  "richContent": [
    [
      {
        "title": "Check Results",
        "type": "info",
        "subtitle": "Click below to view the results.",
        "actionLink": "http://results.drait.in/",
        "image": {
          "src": {
            "rawUrl":
"https://drait.edu.in/assets/images/results_updated.jpg"
          }
        },
        "event": {
          "name": "",
          "parameters": {},
          "languageCode": ""
        }
      }
    ]
  ]
}
```



```
{
  "richContent": [
    [
      {
        "subtitle": "Click below to access the Career portal.",
        "image": {
          "src": {
            "rawUrl": "https://drait.edu.in/assets/images/Career_portal.jpg"
          }
        },
        "type": "info",
        "actionLink": "https://careers.drait.edu.in/",
        "event": {
          "name": "",
          "languageCode": "",
          "parameters": {}
        },
        "title": "Career Portal"
      }
    ]
  ]
}
```



● Alumni

Custom Payload

```json
{
  "richContent": [
    [
      {
        "subtitle": "Click below to access the alumni portal.",
        "actionLink": "https://alumni.drait.edu.in/members",
        "event": {
          "parameters": {},
          "name": "",
          "languageCode": ""
        },
        "image": {
          "src": {
            "rawUrl": "https://drait.edu.in/assets/images/alumni_portal.jpg"
          }
        },
        "type": "info",
        "title": "Alumni Portal"
      }
    ]
  ]
}
```