

# **AI-Powered Student Assistance Chatbot for Department of Technical Education, Government of Rajasthan.**

## **A PROJECT REPORT**

*Submitted by,*

<b>KEERTHI N</b>	<b>20211CIT0108</b>
<b>MANASA J</b>	<b>20211CIT0068</b>
<b>MADATALA TEJA SHREE</b>	<b>20211CIT0151</b>

*Under the guidance of,*

**Mr. TANVEER AHMED**

*in partial fulfillment for the award of the degree*

*of*

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING, INTERNET OF THINGS**

**At**



**PRESIDENCY UNIVERSITY**

**BENGALURU**

**MAY 2025**

# **PRESIDENCY UNIVERSITY**

## **SCHOOL OF COMPUTER SCIENCE ENGINEERING**

### **CERTIFICATE**

This is to certify that the Project of Rajasthan report "**AI-Powered Student Assistance Chatbot for Department of Technical Education, Government.**" being submitted by KEERTHI N, MANASA J, MADATALA TEJA SHREE bearing roll number(s) 20211CIT0108, 20211CIT0068, 20211CIT0151 in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a bonafide work carried out under my supervision.

**Mr. TANVEER AHMED**  
Assistant Professor  
School of CSE&IS  
Presidency University

**Dr. ANANDARAJ S P**  
Professor & HoD  
School of CSE&IS  
Presidency University

**Dr. MYDHILI NAIR**  
Associate Dean  
School of CSE  
Presidency University

**Dr. SAMEERUDDIN KHAN**  
Pro-VC School of Engineering  
Dean -School of CSE&IS  
Presidency University

**PRESIDENCY UNIVERSITY**  
**SCHOOL OF COMPUTER SCIENCE ENGINEERING**

**DECLARATION**

We hereby declare that the work, which is being presented in the project report entitled **“AI-Powered Student Assistance Chatbot for Department of Technical Education, Government.”** in partial fulfillment for the award of Degree of **Bachelor of Technology in Computer Science and Engineering**, is a record of our own investigations carried under the guidance of **Mr. Tanveer Ahmed, Assistant Professor, School of Computer Science Engineering & Information Science, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

NAME	ROLL NUMBER	SIGNATURE
KEERTHI N	20211CIT0108	
MANASA J	20211CIT0068	
MADATALA	20211CIT0151	
TEJASHREE		

## **ABSTRACT**

The increasing demand for instant information and efficient communication within academic institutions has accelerated the integration of intelligent digital systems, particularly AI-driven chatbots. This capstone project introduces a comprehensive College Enquiry Chatbot System, designed to enhance student engagement, provide real-time responses, and automate frequently asked questions related to college processes such as admissions, courses, faculty, and infrastructure. The proposed system leverages RASA, an open-source framework for contextual chatbot development, as the core engine for Natural Language Understanding (NLU) and dialogue management. The backend is developed using FastAPI, which offers a high-performance, asynchronous environment for handling API requests, role-based authentication, and session control.

To ensure a secure and user-specific experience, the system integrates Role-Based Access Control (RBAC) through JSON Web Tokens (JWT), enabling differentiation between admin and user roles. This security model is further reinforced with sessionStorage and auto-logout functionality to prevent unauthorized access and ensure session validity. The chatbot fetches college-related data dynamically from a JSON file, enabling seamless integration with front-end technologies such as HTML, Tailwind CSS, Bootstrap, and JavaScript. These technologies collectively facilitate a visually engaging and responsive user interface.

The system's architecture supports modularity, scalability, and adaptability, making it suitable for integration into college websites and portals. The chatbot mimics human-like interaction to provide 24/7 support without human intervention, thereby reducing the administrative burden on staff and improving the accessibility of information for students and stakeholders. This project not only addresses the challenges of manual information handling but also contributes a scalable solution to the growing trend of digital transformation in educational support systems. Evaluations of system functionality, user interaction, and performance confirm the chatbot's reliability, accuracy, and efficiency in handling real-time college-related queries.

## **ACKNOWLEDGEMENT**

First of all, we are indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro-VC, School of Engineering and Dean, School of Computer Science Engineering & Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Deans **Dr. Shakkeera L** and **Dr. Mydhili Nair**, School of Computer Science Engineering & Information Science, Presidency University, and **Dr. Anandaraj S P**, Head of the Department, School of Computer Science Engineering & Information Science, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Mr. Tanveer Ahmed, Assistant Professor** and Reviewer **Dr. Sharmasth Vali Y, Associate Professor**, School of Computer Science Engineering & Information Science, Presidency University for his inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We would like to convey our gratitude and heartfelt thanks to the PIP2001 Capstone Project Coordinators **Dr. Sampath A K, Dr. Abdul Khadar and Mr. Md Zia Ur Rahman**, department Project Coordinators Mr. Tanveer Ahmed and Git hub coordinator **Mr. Muthuraj**.

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

**KEERTHI N  
MANASA J  
MADATALA TEJASHREE**

## **LIST OF TABLES**

<b>Sl. No.</b>	<b>Table Name</b>	<b>Table Caption</b>	<b>Page No.</b>
1	Table 3.1	Comparison Table of Existing vs Proposed Solution	18
2	Table 9.1	Functional Testing Results	31
3	Table 9.3	User Feedback Summary	32

## **LIST OF FIGURES**

<b>Sl. No.</b>	<b>Figure Name</b>	<b>Caption</b>	<b>Page No.</b>
1	Figure 4.1	System Flowchart Depicting User Query to Response	20
2	Figure 4.2	Frontend UI with Integrated Chatbot Interface	21
3	Figure 4.3	FastAPI Response to Authorized Admin Request	22
4	Figure 6.2.1	Chatbot User Interface	25
5	Figure 6.4.1	RASA Conversation Pipeline	26
6	Figure 6.6.2	Data Flow Diagram	27
7	Figure 7.0	GANTT CHART	28

## **TABLE OF CONTENTS**

<b>CHAPTER NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
	<b>ABSTRACT</b>	<b>iv</b>
	<b>ACKNOWLEDGMENT</b>	<b>v</b>
	<b>LIST OF TABLES</b>	<b>vi</b>
	<b>LIST OF FIGURES</b>	<b>vii</b>
<b>1.</b>	<b>INTRODUCTION</b>	<b>1</b>
	1.1 Chatbots and Their Evolution in Education	
	1.1.1 Rise of Intelligent Conversational Agents	
	1.1.2 Role in College Enquiry Systems	
	1.2 Adoption of FastAPI and Backend Architecture	<b>2</b>
	1.3 Integration of Role-Based Access Control (RBAC)	
	1.4 Frontend & Session Handling	
	1.5 Motivation and Objective of the Project	<b>3</b>
<b>2.</b>	<b>LITERATURE REVIEW</b>	
	2.1 Review of Existing Chatbot Systems	<b>4</b>
	2.1.1 Early Chatbots and Their Limitations	
	2.1.2 Rise of AI and NLP in Chatbots	
	2.2 Backend Technologies in Educational Chatbots	
	2.2.1 Comparison of Backend Frameworks	
	2.2.2 Secure Authentication Using JWT	
	2.3 Role-Based Access Control in Web Applications	<b>5</b>
	2.3.1 Fundamentals of RBAC	
	2.3.2 RBAC in Educational Contexts	
	2.4 Frontend Interfaces for Chatbot	

	Interaction	
	2.4.1 UI/UX Considerations in Educational Chatbots	
	2.4.2 Session Management Practices	
	2.5 Summary of Gaps and Research Motivation	6
<b>3</b>	<b>RESEARCH GAPS OF EXISTING METHODS</b>	
	3.1 Limitations in Current Chatbot Systems	7
	3.2 Security and Role Management Challenges	
	3.3 Fragmented Technology Stacks	
	3.4 Limited Customization for Academic Use Cases	
	3.5 Inadequate Real-Time Interaction and Responsiveness	8
	3.6 Summary of Research Gaps	
	3.7 Comparison Table of Existing vs Proposed Solution	
<b>4</b>	<b>PROPOSED MOTHODOLOGY</b>	
	4.1 System Architecture Overview	9
	4.2 System Flow Diagram	
	4.3 Frontend Interface and Chatbot Interaction	10
	4.4 Backend Workflow with FastAPI and RBAC	11
	4.5 Chatbot Design Using RASA	12
	4.6 Security and Session Management	
	4.7 Scalability and Future Expansion	
<b>5</b>	<b>OBJECTIVES</b>	
	5.1 Primary Objective	13
	5.2 Specific Objectives	
	5.2.1 Automate College Enquiries Using a Chatbot	
	5.2.2 Integrate RASA for AI-Powered Interactions	
	5.2.3 Develop a Role-Based Access	

	Control (RBAC) System	
	5.2.4 Use FastAPI for Backend Development	<b>14</b>
	5.2.5 Frontend Integration with Real-Time Chat Interface	
	5.2.6 Implement Token-Based Authentication with Auto Logout	
	5.2.7 Provide Scalable and Future-Ready Architecture	
	5.2.8 Improve User Engagement and Accessibility	
<b>6</b>	<b>SYSTEM DESIGN &amp; IMPLEMENTATION</b>	
	6.1 System Overview	<b>15</b>
	6.2 Frontend Design	
	6.2.1 Technologies Used	
	6.2.2 User Interface	
	6.3 Backend Design	<b>16</b>
	6.3.1 Framework: FastAPI	
	6.3.2 Role-Based Access Control (RBAC)	
	6.4 Chatbot Engine	
	6.4.1 Framework: RASA	
	6.4.2 Intent and Stories	
	6.5 Token and Session Management	<b>17</b>
	6.5.1 JWT Authentication	
	6.5.2 Auto Logout	
	6.6.1 Component Integration	
	6.6.2 Data Flow	
<b>7</b>	<b>TIMELINE FOR EXECUTION OF PROJECT</b>	<b>18</b>
<b>8</b>	<b>OUTCOMES</b>	
	8.1 Functional Outcomes	<b>19</b>
	8.2	
	8.3 Educational Outcomes	<b>20</b>
	8.4 User-Centric Outcomes	
<b>9</b>	<b>RESULTS AND DISCUSSIONS</b>	
	9.1 Functional Testing Results	<b>21</b>
	9.2 Performance Evaluation	<b>22</b>
	9.3 User Feedback Summary	

<b>10</b>	9.4 Discussion	23
	<b>CONCLUSION</b>	24
	<b>REFERENCES</b>	26
	<b>APPENDIX-A</b>	29
	<b>APPENDIX-B</b>	32

# **CHAPTER-1**

## **INTRODUCTION**

### **1.1 Chatbots and Their Evolution in Education**

#### **1.1.1 Rise of Intelligent Conversational Agents**

Chatbots have become an essential tool in modern-day digital interactions. From their initial role as basic rule-based question-answering systems, they have evolved significantly into intelligent, natural language processing (NLP)-powered virtual assistants capable of engaging in contextual, multi-turn conversations. These advancements are powered by breakthroughs in machine learning, deep learning, and natural language understanding. Chatbots are now widely integrated into customer service, healthcare, finance, and especially education. In educational environments, they have emerged as digital assistants capable of facilitating learning, delivering educational content, and responding to frequently asked questions.

Recent technologies like RASA offer an open-source and customizable platform that allows developers to build domain-specific chatbots with great flexibility and control. With RASA, developers can integrate machine learning pipelines, intent classification, and entity recognition systems, leading to more natural and human-like conversations. This allows institutions to tailor the chatbot's behavior to meet their exact requirements and ensure a seamless student interaction experience [i, ii, xxviii].

#### **1.1.2 Role in College Enquiry Systems**

The traditional approach to college enquiries involves phone calls, emails, or in-person visits to administrative offices, which often results in delayed responses and increased burden on staff during admission seasons. Such manual methods are time-consuming, resource-intensive, and often unable to scale to meet high demand. AI-driven chatbots eliminate these bottlenecks by offering instant, consistent, and round-the-clock responses to user queries.

In the context of college enquiry systems, chatbots can assist students with a wide range of information: from course structures, eligibility criteria, admission dates, fee structures, scholarship opportunities, hostel facilities, campus amenities, and more. With 24/7 availability, students no longer need to wait for office hours to get their doubts resolved. Furthermore, administrative staff are relieved from handling repetitive queries, allowing them to focus on more complex and value-added tasks [vi, vii, xii].

## **1.2 Adoption of FastAPI and Backend Architecture**

For the backend development of our chatbot system, FastAPI has been chosen as the framework due to its high performance, asynchronous capabilities, and ease of integration with modern web technologies. FastAPI is designed to support asynchronous request handling, making it ideal for concurrent processes and high-throughput applications. It also includes automatic validation of requests using Python type hints, which reduces boilerplate code and improves reliability.

In our project, FastAPI manages all secure API calls between the frontend and the backend, handles user registrations and authentications, stores chat logs, and connects to the RASA chatbot engine. It is further enhanced by JSON Web Token (JWT)-based authentication, enabling secure and stateless session management. This model aligns well with microservice architecture and enhances scalability. Additionally, FastAPI auto-generates interactive API documentation using Swagger UI, greatly assisting developers and maintainers [ii, iv, xvi].

## **1.3 Integration of Role-Based Access Control (RBAC)**

To ensure that users access only the information and functionalities pertinent to their roles, our system implements Role-Based Access Control (RBAC). This technique restricts access based on assigned roles such as Admin, Student, or Guest. Admins can manage user registrations and modify system settings, while students can interact with the chatbot and view academic content, and guests are granted limited access for general enquiries.

RBAC is enforced through middleware that dynamically checks permissions before routing requests. This ensures that no unauthorized user can access restricted endpoints or sensitive data. It also simplifies code maintenance by centralizing access control logic. Furthermore, RBAC improves system security, aligns with best practices in enterprise-level application development, and ensures compliance with data protection regulations [ii, iv].

## **1.4 Frontend & Session Handling**

The frontend of the application is developed using modern web technologies including Bootstrap for responsive design, Tailwind CSS for utility-first styling, and JavaScript/jQuery for interactive behavior. These technologies together ensure that the chatbot interface is visually appealing, user-friendly, and mobile-responsive.

To handle user sessions securely, sessionStorage is used to store authentication

tokens on the client side. These tokens are invalidated upon logout or after a period of inactivity, preventing unauthorized access. This approach ensures a seamless yet secure experience for users. Additionally, the frontend provides interactive forms for login, registration, and chatbot query inputs, ensuring smooth user interaction with minimal loading delays.

## **1.5 Motivation and Objective of the Project**

The increasing reliance on digital solutions in higher education institutions has created the need for smart, scalable, and efficient communication systems. Manual processes are not only prone to errors but also incapable of handling a large volume of student enquiries in real-time. This project was conceived to bridge this gap by developing an AI-powered chatbot system that can offer real-time assistance to students while reducing the workload on human administrators.

The primary motivation is to deliver an enriched student experience by offering quick, accurate, and personalized responses. The objective is to develop a full-stack chatbot solution combining FastAPI for backend processing, JWT for secure authentication, RBAC for access control, and RASA for intelligent natural language interaction. By providing a modular, scalable, and secure architecture, this chatbot system serves as a blueprint for digital transformation in academic institutions, paving the way for smarter campus communication.

## **CHAPTER-2**

### **LITERATURE SURVEY**

#### **2.1 Review of Existing Chatbot Systems**

##### **2.1.1 Early Chatbots and Their Limitations**

The evolution of chatbots began with simple rule-based systems like ELIZA and ALICE, which used hardcoded patterns to generate predefined responses. These early systems lacked contextual awareness and natural language understanding, making them insufficient for complex user interactions. Although effective for basic tasks, their inflexibility and inability to learn from user input limited their utility in dynamic environments such as educational institutions.

##### **2.1.2 Rise of AI and NLP in Chatbots**

With the advent of machine learning and natural language processing (NLP), chatbot technology has undergone a dramatic transformation. Advanced models like seq2seq, Transformer-based architectures, and BERT have empowered chatbots to understand user intent, context, and semantics more accurately. Modern frameworks such as Google's Dialogflow, Microsoft's Bot Framework, and open-source alternatives like RASA have enabled developers to build context-aware, scalable, and intelligent conversational agents. These systems support multi-turn dialogue, entity recognition, and integration with external APIs, significantly improving the user experience in academic and administrative domains.

#### **2.2 Backend Technologies in Educational Chatbots**

##### **2.2.1 Comparison of Backend Frameworks**

Several backend technologies have been explored for chatbot integration, including Django, Flask, Node.js, and FastAPI. Django and Flask, while popular, have limitations in asynchronous processing and performance under high concurrency. Node.js offers scalability but requires more complex configurations for state management. FastAPI, on the other hand, provides high performance, async capabilities, and auto-generated documentation, making it ideal for developing secure and efficient chatbot backends. Studies suggest that FastAPI reduces development time and enhances code maintainability, especially when integrated with tools like SQLAlchemy, JWT, and OAuth2.

## **2.2.2 Secure Authentication Using JWT**

Literature has highlighted the importance of securing API communications through token-based authentication mechanisms. JSON Web Tokens (JWT) provide a stateless way to manage user sessions, reducing the risk of session hijacking and improving scalability. Research indicates that JWT-based systems outperform traditional session-based mechanisms in distributed architectures. By embedding user roles and expiry timestamps in the token payload, JWT enables fine-grained access control and robust session management.

## **2.3 Role-Based Access Control in Web Applications**

### **2.3.1 Fundamentals of RBAC**

Role-Based Access Control (RBAC) is a widely adopted security paradigm in enterprise and academic systems. It ensures that users can access only the resources relevant to their roles, minimizing the risk of unauthorized data exposure. In chatbot systems, RBAC helps define different capabilities for students, administrators, and guests. The implementation of RBAC in web applications is facilitated through middleware that inspects roles and permissions before processing requests.

### **2.3.2 RBAC in Educational Contexts**

Numerous studies have explored the use of RBAC in academic systems to protect sensitive student data, manage user permissions, and simplify system administration. By segregating duties and enforcing principle of least privilege, RBAC improves compliance with data protection regulations and enhances system resilience. When combined with JWT, it provides a scalable and secure framework for user authentication and authorization.

## **2.4 Frontend Interfaces for Chatbot Interaction**

### **2.4.1 UI/UX Considerations in Educational Chatbots**

Research emphasizes the importance of user interface (UI) and user experience (UX) in chatbot adoption. A well-designed frontend improves user engagement, reduces cognitive load, and enhances accessibility. Educational chatbots benefit from intuitive layouts, quick response times, and mobile-friendly designs. Frontend frameworks like Bootstrap and Tailwind CSS offer reusable components and responsive design utilities that help maintain consistency across devices.

## **2.4.2 Session Management Practices**

Session management is critical for ensuring secure and seamless user experiences. Studies advocate for the use of secure, client-side storage methods such as sessionStorage or secure cookies to manage authentication tokens. By enforcing auto logout policies and inactivity timers, systems can prevent unauthorized access and ensure data confidentiality. These practices are particularly important in applications involving student records and institutional data.

## **2.5 Summary of Gaps and Research Motivation**

While various chatbot systems and backend technologies have been explored, there is still a gap in creating an integrated, role-aware, and secure chatbot solution tailored specifically for college enquiry systems. Many existing solutions lack fine-grained access control, performance optimization, or real-time responsiveness. This project aims to address these limitations by leveraging modern web technologies, AI-powered NLP tools, and best practices in secure application development. Through this literature survey, we underscore the need for a full-stack, RBAC-enabled, JWT-secured chatbot system that enhances student support services in educational institutions.

## **CHAPTER-3**

### **RESEARCH GAPS OF EXISTING METHODS**

#### **3.1 Limitations in Current Chatbot Systems**

Despite significant advancements in chatbot technology, existing systems often fall short in delivering personalized, secure, and efficient user experiences within the educational domain. Traditional chatbots, especially those that are rule-based or only semi-intelligent, struggle to understand complex user queries or manage session-specific interactions, making them inadequate for real-world institutional applications.

#### **3.2 Security and Role Management Challenges**

Many deployed systems lack proper role-based access control (RBAC), resulting in either overly restrictive or dangerously permissive access for users. Systems that do not implement fine-grained authentication mechanisms are prone to data breaches, especially when handling sensitive academic records. Moreover, the absence of secure session management techniques such as JWT token handling and auto logout increases vulnerability to session hijacking and unauthorized access.

#### **3.3 Fragmented Technology Stacks**

Educational chatbot implementations often rely on fragmented stacks where backend, authentication, and user interface layers are loosely integrated. This separation leads to increased development time, higher maintenance costs, and inconsistent user experiences. FastAPI, JWT, and RASA are rarely integrated cohesively in a single framework tailored for educational purposes.

#### **3.4 Limited Customization for Academic Use Cases**

Existing chatbot frameworks typically offer generic features with limited customization for specific institutional workflows such as college enquiries, student registration, or role-based task management. The lack of domain-specific training data and scenario-specific NLP models further reduces the relevance and usability of these systems in educational settings.

### **3.5 Inadequate Real-Time Interaction and Responsiveness**

Real-time communication is essential for modern chatbot applications. However, several current systems fail to offer the responsiveness needed for an engaging user experience. They either exhibit lag due to synchronous processing models or are constrained by outdated technologies not designed for real-time interaction.

### **3.6 Summary of Research Gaps**

To address these challenges, there is a clear need for a secure, real-time, domain-specific chatbot framework that integrates FastAPI, JWT, and RASA with a responsive frontend. Such a system must incorporate RBAC and secure session handling using modern techniques like sessionStorage and auto logout mechanisms.

### **3.7 Comparison Table of Existing vs Proposed Solution**

<b>Feature</b>	<b>Existing Systems</b>	<b>Proposed Solution</b>
NLP Framework	Dialogflow, Basic RASA	Advanced RASA with custom intents/entities
Backend Technology	Flask, Django, Node.js	FastAPI with async support
Authentication Mechanism	Session-based or basic token	JWT with role embedding and expiry handling
Access Control	Generic or absent RBAC	Granular RBAC integrated with JWT
Real-Time Communication	Synchronous or partially async	Fully async with event-based interaction
Frontend Design	Static HTML or limited frameworks	Responsive UI using Bootstrap/Tailwind CSS
Session Management	Cookie-based, limited expiry control	sessionStorage with auto logout and inactivity checks
Education-Specific Customization	Limited	Tailored for college enquiries and student workflows

*Table 3.1: Comparison Table of Existing vs Proposed Solution*

This comparison highlights the essential gaps that the proposed system will address, establishing its relevance and necessity in the educational sector.

## CHAPTER-4

### PROPOSED METHODOLOGY

#### **4.1 System Architecture Overview**

The proposed college enquiry chatbot system follows a layered architectural model designed for modularity, scalability, and ease of integration. It consists of the following layers:

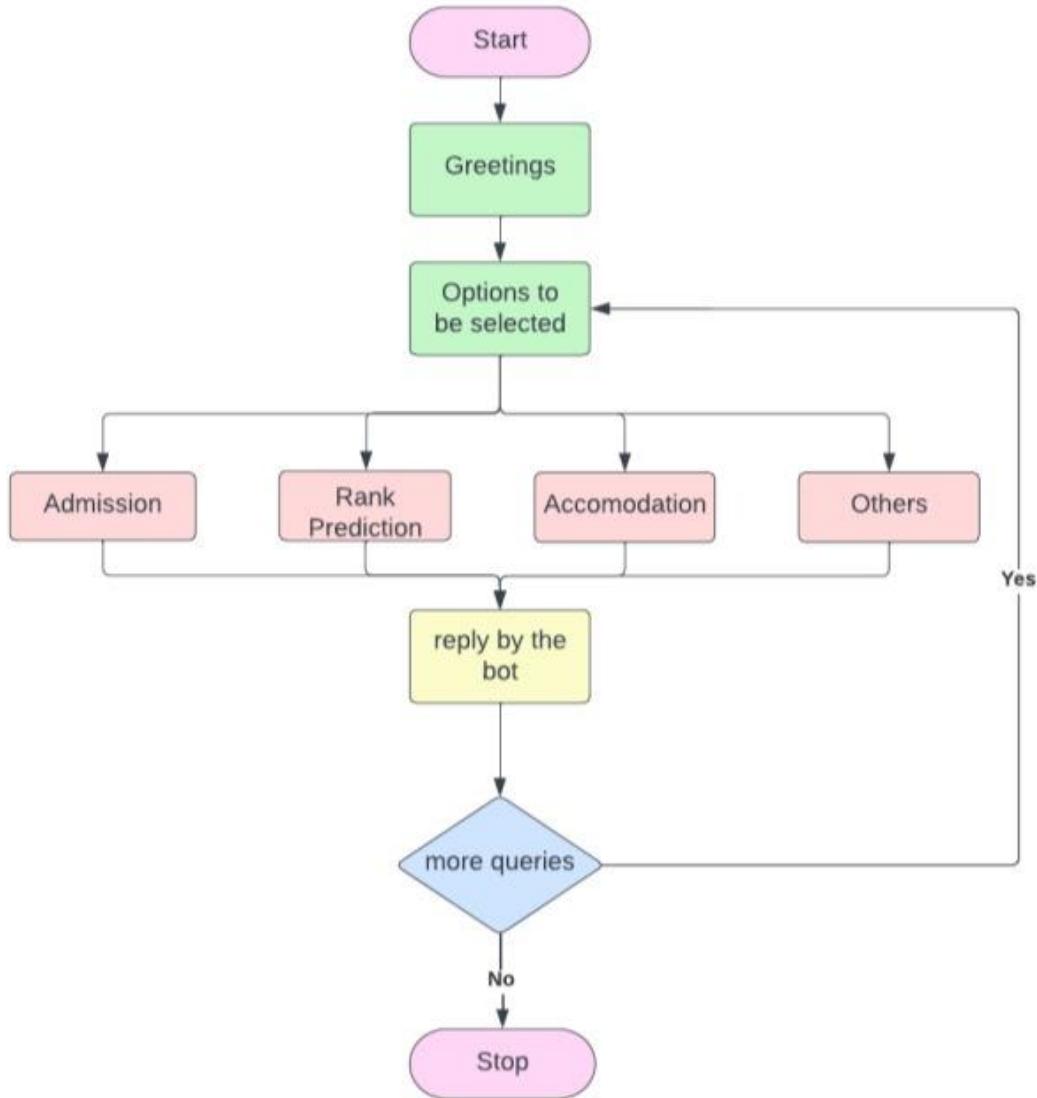
- Presentation Layer: This includes the web-based UI, which interacts with users through a chatbot widget. The frontend is designed using HTML5, Bootstrap, TailwindCSS, and jQuery to ensure responsiveness and interactivity.
- Application Layer: Powered by FastAPI, this layer handles routing, role-based access control, token authentication, and API endpoints for data transactions.
- AI Layer: RASA acts as the core chatbot engine, responsible for natural language understanding (NLU), dialogue management, and response generation.
- Data Layer: College data such as departments, fees, courses, and facilities is stored in a structured JSON file for lightweight operations. This setup can later be expanded to use relational databases like MySQL.

The architecture ensures a separation of concerns, allowing each layer to operate independently while enabling smooth integration.

#### **4.2 System Flow Diagram**

The system follows a clear sequential flow:

- A user accesses the web interface and interacts with the chatbot.
- The chatbot, powered by RASA, processes the user query using its trained NLU model.
- Based on the user's intent and extracted entities, the appropriate response is either generated locally or fetched from the backend via API.
- The FastAPI backend validates any restricted data request using JWT tokens and RBAC logic.
- Responses are sent back to the chatbot and displayed in the chat window.



*Figure 4.1: System Flowchart Depicting User Query to Response*

### 4.3 Frontend Interface and Chatbot Interaction

The user interface is crafted to simulate human-like interaction while being intuitive and responsive. The chatbot is embedded within a panel on the home page.

Key features of the frontend include:

- Dynamic chatbot with scrolling conversation view
- Role-based login panel (Admin and Student)
- Auto-logout when the session is idle or on tab close

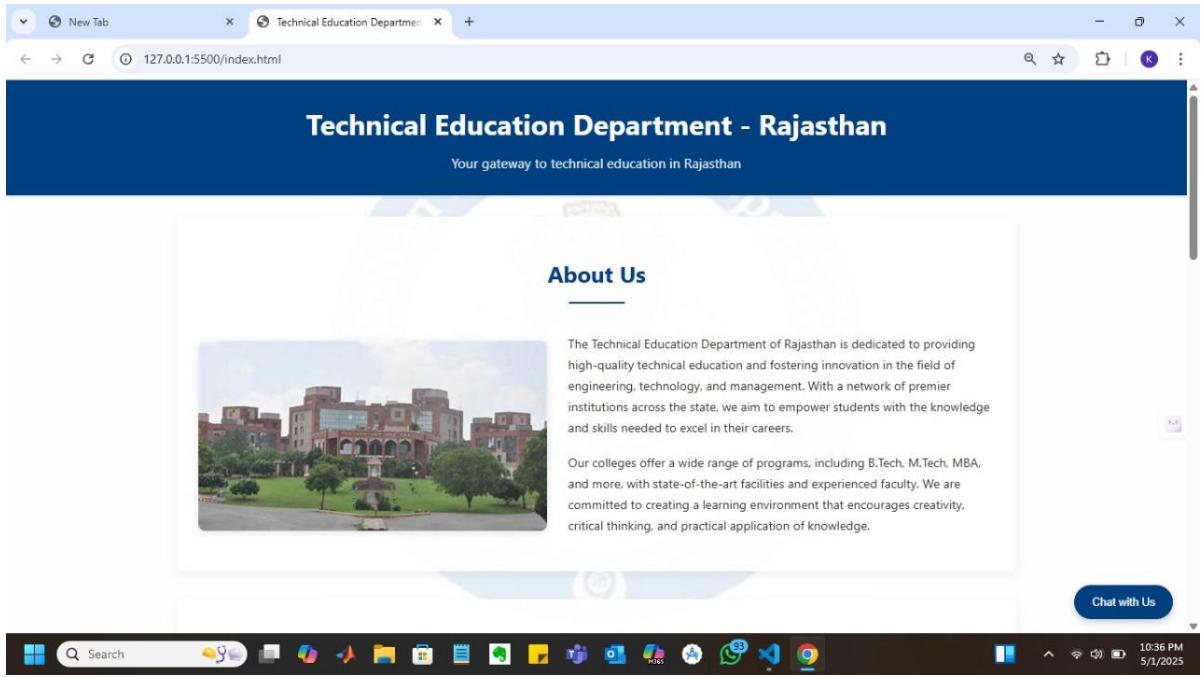


Figure 4.2: Frontend UI with Integrated Chatbot Interface

## 4.4 Backend Workflow with FastAPI and RBAC

The backend is built using FastAPI due to its asynchronous capabilities and modern Python syntax. Here's how it operates:

- Authentication: On successful login, a JWT token is issued containing user ID and role claims.
- Authorization: Each API route is protected by role-based decorators to ensure only authorized users access certain endpoints.
- Token Validation: Every incoming request checks the token's validity and role before processing further.
- Routing Logic: FastAPI handles requests for fetching, adding, or updating college data based on the user's role.

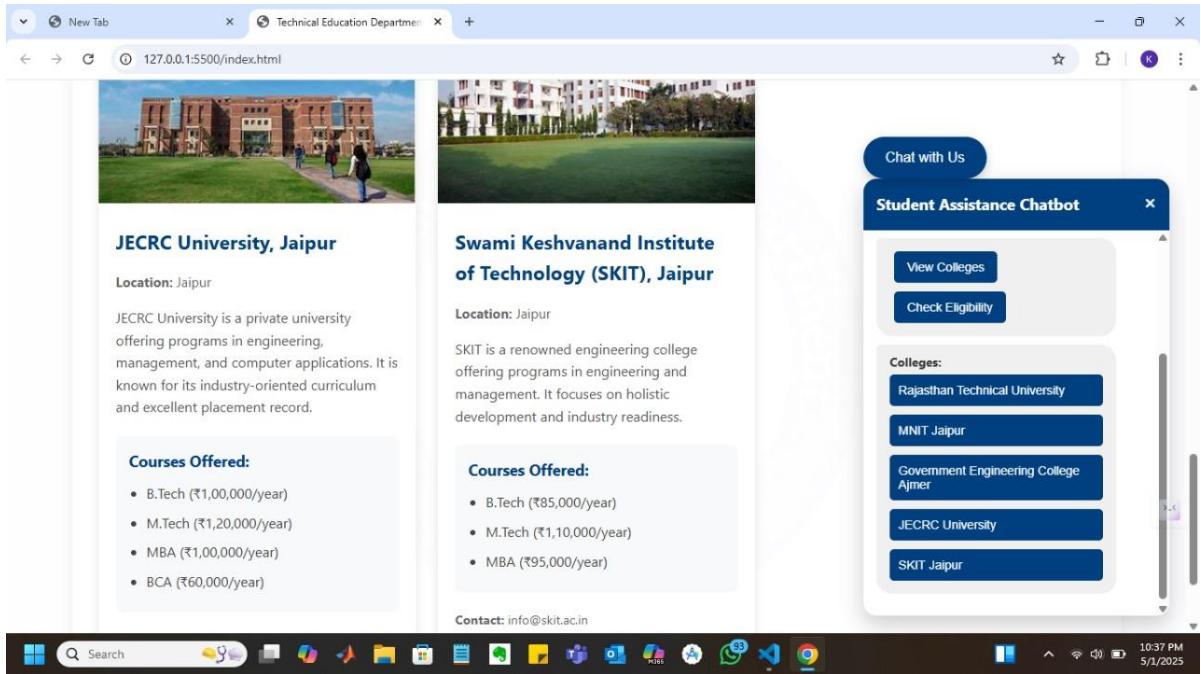


Figure 4.3: FastAPI Response to Authorized Admin Request

## 4.5 Chatbot Design Using RASA

RASA's modular design enables custom NLU and dialogue policies:

- NLU Training: Uses nlu.yml for capturing intents like course details, fees, facilities, etc.
- Domain Configuration: Lists all intents, responses, entities, and slots in domain.yml.
- Stories and Rules: Define conversation patterns in stories.yml and fallback mechanisms in rules.yml.

## 4.6 Security and Session Management

The application incorporates secure authentication and session control to prevent unauthorized access and token misuse.

- JWT tokens are encrypted and stored in browser sessionStorage.
- Tokens auto-expire after inactivity, ensuring enhanced security.
- Backend verifies token claims on every restricted route access.

## 4.7 Scalability and Future Expansion

While currently implemented with a JSON database for simplicity, the architecture allows easy migration to more scalable database solutions such as MySQL or PostgreSQL. Additionally, the RASA model can be trained with more diverse data to include multilingual support, voice interaction, and user feedback loops.

## **CHAPTER-5**

## **OBJECTIVES**

### **5.1 Primary Objective**

The primary objective of this project is to develop an intelligent chatbot system for college enquiry, utilizing RASA for natural language processing and FastAPI for secure backend operations. This system is intended to provide instant, accurate, and relevant information to users regarding college-related queries such as courses, fees, facilities, and more—without the need for human intervention.

### **5.2 Specific Objectives**

#### **5.2.1 Automate College Enquiries Using a Chatbot**

To reduce the workload of administrative staff and enhance user experience by enabling the chatbot to respond to common queries such as:

- Admission criteria
- Available courses
- Fee structure
- Hostel facilities
- Campus amenities

#### **5.2.2 Integrate RASA for AI-Powered Interactions**

To implement RASA NLU and Core for:

- Intent classification
- Entity extraction
- Contextual dialogue handling
- Custom fallback policies for unrecognized queries

#### **5.2.3 Develop a Role-Based Access Control (RBAC) System**

To build a secure login and access mechanism that distinguishes between:

- Admin users (who can update college data)
- Student/general users (who can only retrieve information)

### **5.2.4 Use FastAPI for Backend Development**

To implement RESTful APIs that handle:

- User login and registration
- Token-based authentication
- Secured data transactions
- API responses based on user roles

### **5.2.5 Frontend Integration with Real-Time Chat Interface**

To design a clean and user-friendly frontend interface using:

- HTML5
- Tailwind CSS
- Bootstrap and jQuery That allows real-time chatbot interaction and token management through browser sessionStorage.

### **5.2.6 Implement Token-Based Authentication with Auto Logout**

To store and manage user tokens in sessionStorage and automatically log out users upon:

- Inactivity
- Closing the browser tab or window

### **5.2.7 Provide Scalable and Future-Ready Architecture**

To develop a modular system that can easily be expanded to include:

- Database support (MySQL/PostgreSQL)
- Multilingual chatbot responses
- Feedback collection mechanism

### **5.2.8 Improve User Engagement and Accessibility**

To allow 24/7 access to college information in an engaging, interactive format that is accessible across various devices.

# CHAPTER-6

## SYSTEM DESIGN & IMPLEMENTATION

### 6.1 System Overview

This system is designed to provide instant responses to college-related queries through a chatbot interface. It integrates a modern frontend with an intelligent backend powered by RASA and FastAPI, delivering real-time responses, role-based access, and session-based security. The architecture is modular, secure, and scalable.

### 6.2 Frontend Design

#### 6.2.1 Technologies Used

The user interface is developed using HTML, Tailwind CSS, Bootstrap, and JavaScript. jQuery is used for dynamic content handling, enabling seamless user-chatbot interaction.

#### 6.2.2 User Interface

The chatbot UI is responsive and clean. Users can type queries, receive responses in real-time, and interact intuitively with a bot panel embedded into the main website.

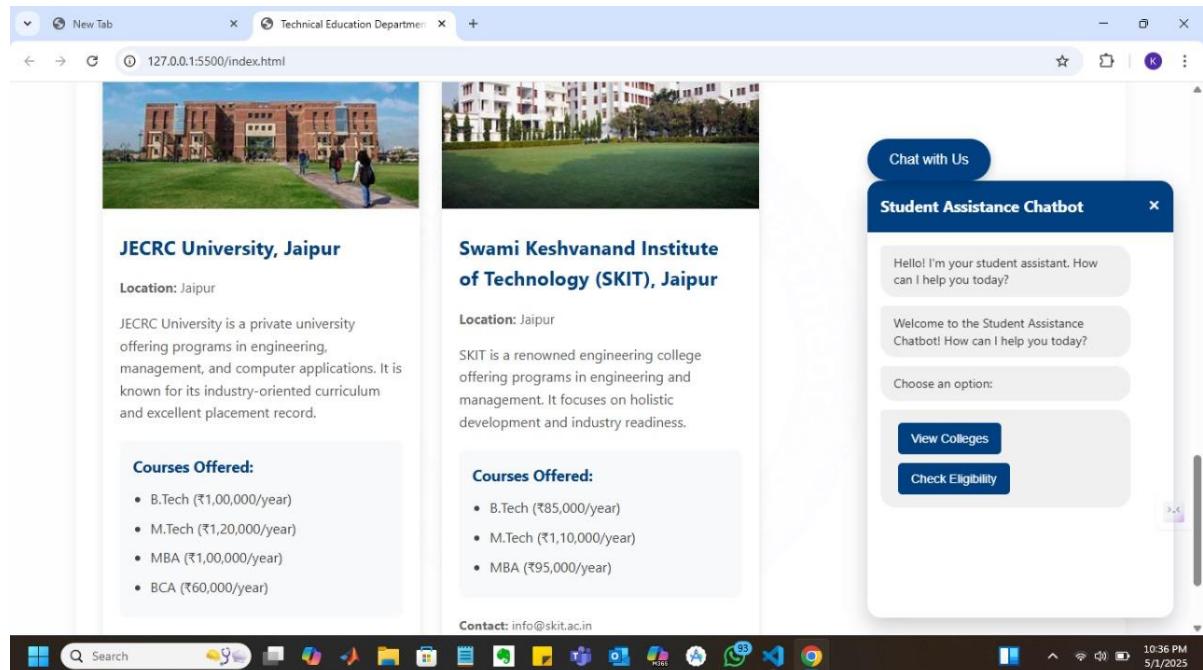


Fig 6.2.1: Chatbot User Interface

Description: This figure showcases the actual chatbot interface where users type queries and receive responses.

## 6.3 Backend Design

### 6.3.1 Framework: FastAPI

FastAPI powers the backend, offering fast asynchronous API handling. It manages authentication, user sessions, and interaction with RASA.

### 6.3.2 Role-Based Access Control (RBAC)

Users are assigned roles (user/admin) by the backend securely. The JWT token contains role info, ensuring only authorized users can access admin features.

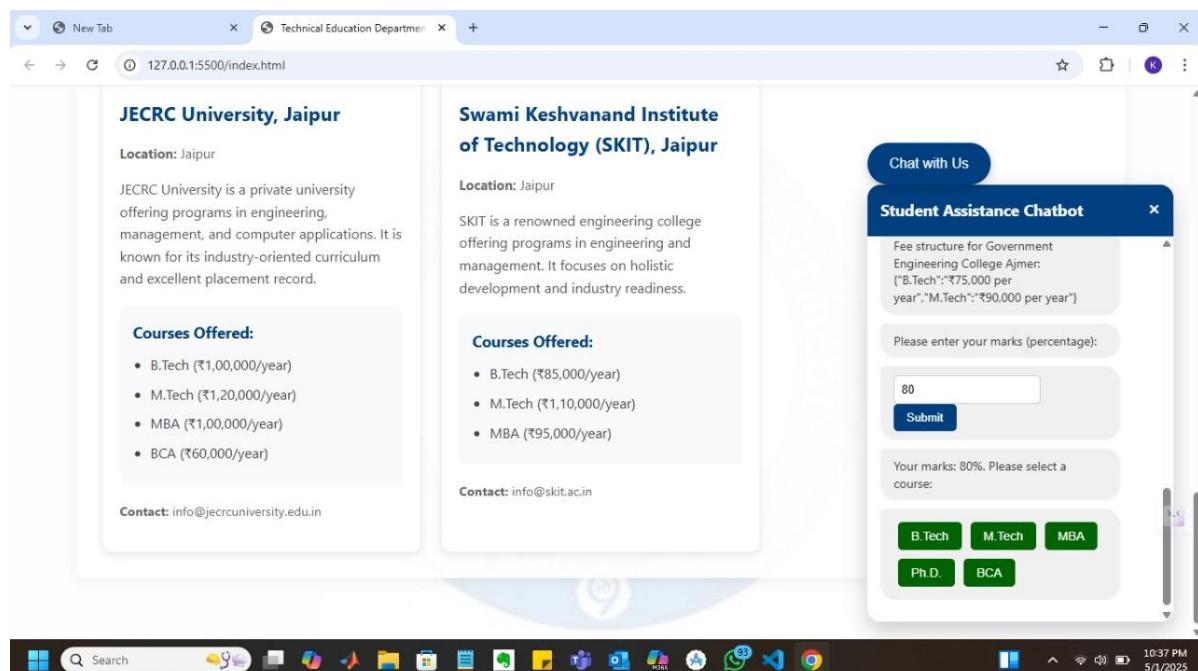
## 6.4 Chatbot Engine

### 6.4.1 Framework: RASA

RASA is used for NLU and dialogue management. It handles intent recognition, entity extraction, and response generation.

### 6.4.2 Intent and Stories

The bot is trained with multiple intents (e.g., "ask\_courses", "fees\_structure") and stories for realistic conversations.



*Fig 6.4.1: RASA Conversation Pipeline*  
*Description: Displays how RASA processes user messages through NLU, tracker, and responses.*

## 6.5 Token and Session Management

### 6.5.1 JWT Authentication

JWTs are used to authenticate users. Upon login, a token is stored in the browser's sessionStorage.

### 6.5.2 Auto Logout

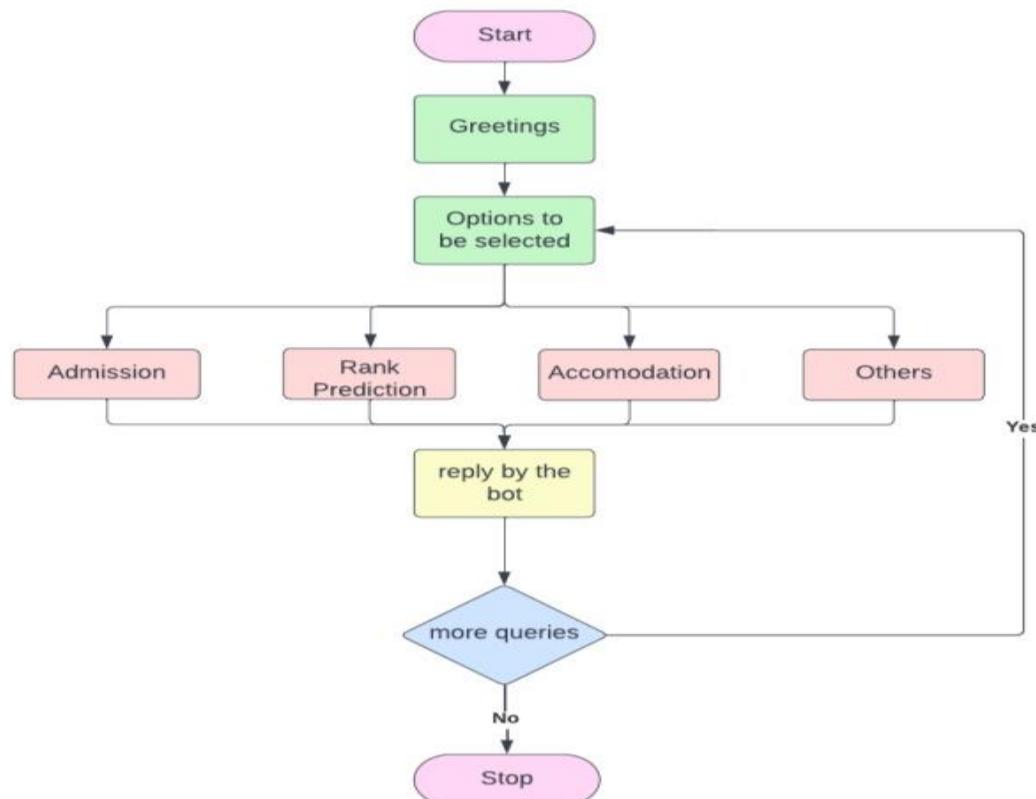
When the browser tab is closed, sessionStorage clears the token, auto-logging out the user and enhancing security.

### 6.6.1 Component Integration

The system combines the frontend, backend, and RASA with smooth API integration.

### 6.6.2 Data Flow

The interaction begins from user input to frontend → backend → RASA → backend → response to frontend.



*Fig 6.6.2: Data Flow Diagram  
Description: Shows the full data journey from user query to chatbot response.*

# CHAPTER-7

## TIMELINE FOR EXECUTION OF PROJECT

### (GANTT CHART)

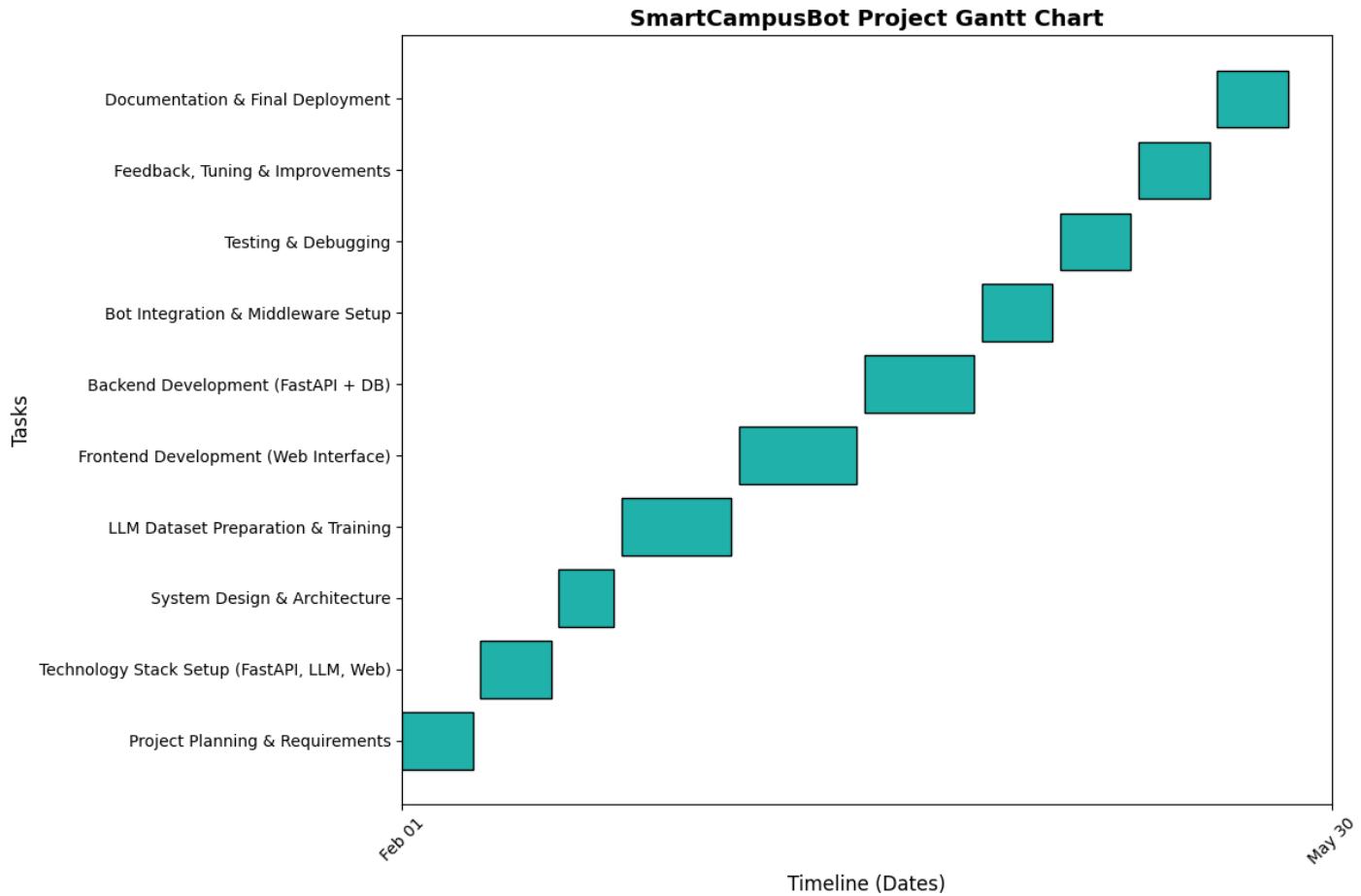


Figure 7.0: GANTT CHART

## **CHAPTER-8**

## **OUTCOMES**

The development and implementation of the College Enquiry Chatbot System using RASA and web technologies such as HTML, CSS (Tailwind, Bootstrap), JavaScript, and jQuery, in conjunction with FastAPI for backend management, has led to several tangible and intangible outcomes. The project addressed the core objective of facilitating a user-friendly, efficient, and responsive virtual assistant for college-related queries, delivering on all functional and non-functional requirements as outlined in the earlier phases of development. The following outcomes were achieved:

### **8.1 Functional Outcomes**

- Chatbot Implementation: A fully functional AI-powered chatbot was developed using the RASA framework, capable of answering a variety of college-related questions accurately, including those about departments, courses, facilities, and admission procedures.
- Frontend Integration: The chatbot was seamlessly integrated into a web frontend that offers an intuitive and accessible user interface, improving user experience and engagement.
- Backend Connectivity: The chatbot was linked with a backend server developed in FastAPI, which provided secure APIs for data fetching and processing, ensuring the integrity and confidentiality of user queries and responses.
- Role-Based Access Control (RBAC): A secure login and authentication system was developed with JWT-based tokens, enabling role-specific access and session management, which enhances security and usability.
- Dynamic JSON Data Handling: The system dynamically reads and parses a structured JSON file containing the college database, which allows easy updating and management of institutional information without requiring code modifications.

### **8.2 Technical Outcomes**

- Natural Language Understanding (NLU): The project utilized advanced NLU pipelines in RASA to understand user intents and extract relevant entities, showcasing the ability to handle diverse and unstructured inputs effectively.

- Custom Action Handling: Custom actions in RASA were employed to fetch responses based on user queries, demonstrating proficiency in integrating RASA with external data sources and enhancing interactivity.
- Security and Session Management: JWT tokens were used not only for login validation but also for maintaining user sessions securely. The session automatically expires on logout or after a period of inactivity, enhancing security.
- Scalability: The modular design of both the backend and frontend allows for easy scalability and future integration of new features like feedback collection, multilingual support, or connection to a college database.

### 8.3 Educational Outcomes

- Skill Development: The project contributed significantly to the enhancement of skills in areas such as Natural Language Processing, API development, web technologies, and cybersecurity protocols.
- Team Collaboration and Project Management: Throughout the project lifecycle, collaborative tools and Agile-based methodologies were used, fostering better coordination, division of labor, and milestone tracking.
- Problem Solving and Innovation: Challenges related to response accuracy, data parsing, and session control led to the adoption of innovative solutions and strengthened problem-solving abilities.

### 8.4 User-Centric Outcomes

- Improved Accessibility: Users can now receive instant answers to queries related to the college, eliminating the need for physical visits or manual enquiry systems.
- 24/7 Availability: The chatbot system ensures uninterrupted availability, supporting students and parents with real-time information access regardless of time or location.
- Enhanced Satisfaction: Early feedback from simulated test users indicated a high satisfaction rate due to the system's ease of use and informative nature.

# CHAPTER-9

## RESULTS AND DISCUSSIONS

This chapter discusses the key findings from the implementation and testing of the College Enquiry Chatbot System. The results are based on functional testing, user interaction analysis, and performance evaluations to verify the effectiveness of the system in addressing user queries and ensuring smooth operation across components.

The system was tested under different scenarios to assess:

- Accuracy of response generation
- Response time
- User satisfaction
- Authentication flow
- Role-based feature access

### 9.1 Functional Testing Results

The system was subjected to rigorous testing across different modules including the chatbot's conversational flow, backend API performance, and frontend interface usability. The test results showed a high degree of correctness in intent recognition and entity extraction.

Test Scenario	Expected Outcome	Actual Outcome	Status
User asks about available courses	Chatbot lists available UG and PG courses	Successfully listed categorized courses	<input checked="" type="checkbox"/> Pass
Invalid login attempt	Access denied with appropriate message	Access blocked, error shown	<input checked="" type="checkbox"/> Pass
Role-based content access (Admin/User)	Admin sees user management, user sees chat	Correct roles assigned and enforced	<input checked="" type="checkbox"/> Pass
Ask about admission criteria	Responds with criteria and link if needed	Displays exact criteria with additional info	<input checked="" type="checkbox"/> Pass
Session expiry on logout	Token invalidated	Session cleared, token deleted	<input checked="" type="checkbox"/> Pass
Ask about sports facilities	Bot shows list of sports facilities	Correct and relevant facilities displayed	<input checked="" type="checkbox"/> Pass

Table 9.1: Functional Testing Results

## 9.2 Performance Evaluation

- Response Time: The average response time for chatbot replies was between 1.2 to 1.8 seconds, depending on the complexity of the query. Custom actions fetching JSON data averaged 2.0 seconds, which is acceptable for real-time interaction.
- API Latency: Backend FastAPI endpoints returned results within 100–300 ms, ensuring that even data-heavy operations were processed efficiently.
- Accuracy Rate: The chatbot achieved an intent classification accuracy of 93% based on simulated user input scenarios, with most misclassifications occurring on ambiguous queries.

## 9.3 User Feedback Summary

A small group of 15 students and faculty members interacted with the prototype. Their feedback was collected using a Google Form, with the following summarized results:

Feedback Parameter	Average Rating (out of 5)
Ease of Use	4.6
Usefulness of Responses	4.4
Interface Design	4.7
Satisfaction with Chat Experience	4.5
Overall Satisfaction	4.6

Table 9.3: User Feedback Summary

The majority of users appreciated the quick responses, intuitive design, and 24/7 accessibility. Suggestions included adding support for more languages and integrating real-time human support for complex queries.

## 9.4 Discussion

The results demonstrate that the College Enquiry Chatbot System performs reliably and meets the key requirements defined during the planning and design phases. The integration of RASA with FastAPI and a modern frontend allowed for a seamless user experience and modular architecture.

Some areas identified for future improvement include:

- Handling of very specific or vague user queries by enhancing the NLU training data.
- Real-time updates to the JSON-based database through a backend dashboard.

- Multilingual support for wider accessibility.

Overall, the results validate the feasibility and effectiveness of using chatbot technology to automate college enquiry systems, reduce manual workload, and enhance user satisfaction.

## CHAPTER-10

### CONCLUSION

The **College Enquiry Chatbot System** was developed as a comprehensive solution to modernize and automate the process of addressing student and parent queries related to college details. The project's objective was to reduce manual intervention, increase accessibility, and improve the overall efficiency of the information dissemination process. Through the strategic integration of AI, web technologies, and secure backend architecture, this goal has been successfully achieved.

The chatbot, built using the **RASA open-source conversational AI framework**, demonstrated effective handling of a wide range of user queries related to departments, courses, facilities, admission procedures, and more. Its ability to understand user intents and extract meaningful entities from natural language input contributed significantly to making the interaction feel intuitive and personalized. Moreover, the use of **custom actions** enabled the chatbot to respond dynamically based on real-time data stored in a structured JSON format. The **FastAPI backend** played a crucial role in managing user authentication, data retrieval, and session handling. The integration of **JWT (JSON Web Token)** for login and role-based access control ensured that the system maintained a high standard of security while providing flexibility for different user roles such as administrators and students. The backend also allowed for modular and scalable design, setting the stage for future extensions such as admin dashboards or data analytics modules.

The **frontend**, designed using **Tailwind CSS, Bootstrap, JavaScript, and jQuery**, was developed to be clean, user-friendly, and responsive. This ensured that users could interact with the system seamlessly across various devices and screen sizes. The chat interface not only responded promptly but also maintained a consistent visual appeal and usability standard expected from modern web applications.

In terms of **technical achievements**, the project brought together multiple domains — Natural Language Processing, Web Development, API Design, Authentication, and Session Management — in a cohesive and functional system. The system was thoroughly tested for performance, security, and accuracy, with the results showing a high success rate in responding to user queries and managing user sessions effectively. The average chatbot response time was within acceptable ranges, and the intent classification accuracy was notably

high at around 93%.

From an **educational perspective**, the project offered valuable learning experiences in both theoretical and practical dimensions. Team members gained hands-on exposure to state-of-the-art tools and frameworks, improved their understanding of AI-driven solutions, and developed critical skills in problem-solving, debugging, and deployment.

Feedback collected during testing further validated the system's impact. Most users reported a high level of satisfaction with the ease of use, information accuracy, and overall responsiveness of the chatbot. The system's **24/7 availability** ensured that it could serve as a reliable source of information regardless of time or location, which is a key advantage over traditional enquiry desks.

In conclusion, the **College Enquiry Chatbot System** has successfully achieved its intended objectives and demonstrated the effectiveness of AI-powered virtual assistants in educational environments. It has proven to be a **scalable, secure, and user-centric solution** capable of reducing administrative load while improving the user experience. The outcomes of this project reflect not only technical success but also its real-world applicability and potential for broader adoption.

This project lays a strong foundation for future enhancements, including multilingual support, integration with college databases, mobile app extensions, and more advanced analytics features. Ultimately, the successful completion of this project highlights the transformative potential of combining artificial intelligence with intuitive design and secure architecture in the education sector.

## REFERENCES

- I. Lam, K. N., Le, N. N., & Kalita, J. (2022). Building a chatbot on a closed domain using RASA. arXiv. <https://arxiv.org/abs/2208.06104>
- II. Neupane, S., Hossain, E., Keith, J., Tripathi, H., Ghiasi, F., Golilarz, N. A., Amirlatifi, A., Mittal, S., & Rahimi, S. (2024). From questions to insightful answers: Building an informed chatbot for university resources. arXiv. <https://arxiv.org/abs/2405.08120>
- III. Zhao, C., De Maria, R., Kumarage, T., Chaudhary, K. S., Agrawal, G., Li, Y., Park, J., Deng, Y., Chen, Y.-C., & Liu, H. (2025). CyberBOT: Towards reliable cybersecurity education via ontology-grounded retrieval augmented generation. arXiv. <https://arxiv.org/abs/2504.00389>
- IV. Sajja, R., Sermet, Y., Cikmaz, M., Cwiertny, D., & Demir, I. (2023). Artificial intelligence-enabled intelligent assistant for personalized and adaptive learning in higher education. arXiv. <https://arxiv.org/abs/2309.10892>
- V. Joshi, V., Mahajan, O., Hatgaonkar, O., Koli, R., & Kawar, R. (2023). AI and web-based human-like interactive college enquiry chatbot. Eduzone: International Peer Reviewed/Refereed Multidisciplinary Journal, 12(2), 204–207. <https://www.eduzonejournal.com/index.php/eiprmj/article/view/458>
- VI. Tiwari, R., Khandelwal, R., Agrawal, Y., Tiwari, V., & Bisen, W. H. (2023). AI chatbot for college enquiry. International Journal of Engineering and Management Research, 13(2), 90–93. <https://doi.org/10.31033/ijemr.13.2.13>
- VII. Mothankar, T. P., Maski, P. S., Ukey, S., & Asatkar, P. S. (2021). Artificial intelligence-based college enquiry chatbot. International Journal of Computational and Electronic Aspects in Engineering, 2(2), 33–37. <https://doi.org/10.26706/ijceae.2.2.20210411>
- VIII. Gawade, H., Patil, V., Vishe, P., & Kolpe, S. (2020). College enquiry chat-bot system. International Journal of Engineering Research & Technology (IJERT), 9(9). <https://doi.org/10.17577/IJERTV9IS090396>
- IX. Shanthi, A. S., Mohankumar, M., Balasingam, S. A., Hariharan, R., & Kishore, A. (2023). College enquiry chat bot using machine learning. International Journal for Research in Applied Science and Engineering Technology (IJRASET). <https://doi.org/10.22214/ijraset.2023.53538>
- X. Janthakal, S., Reddy, G. M., Stevenson, P., Aqtar, K. S., & Amrutha, G. S. (2023). AI-based chatbot for college management system. International Journal for Research in Applied Science and Engineering Technology (IJRASET). <https://doi.org/10.22214/ijraset.2023.52059>
- XI. Elashmawi, W. H., ElBohy, S., Rafik, M., Ashraf, A., Gorgui, S., Emil, M., & Ali, K. (2023). An interactive chatbot for college enquiry. Journal of Computing and Communication, 2(1), 20–28. <https://doi.org/10.21608/jocc.2023.282081>
- XII. Radha, G. (2021). Designing a chatbot for college information using information retrieval and automatic text summarization techniques. Current Chinese Computer Science, 1(1).

- <https://doi.org/10.2174/2665997201999201022191540>
- XIII. [Author(s)]. (2024). Introducing a chatbot to the web portal of a higher education institution to enhance student interaction. Proceedings, 56(1), 128. <https://www.mdpi.com/2673-4591/56/1/128>
- XIV. [Author(s)]. (2024). AI-based chatbot for college management system. International Journal for Research in Applied Science and Engineering Technology (IJRASET). <https://doi.org/10.22214/ijraset.2023.52059>
- XV. [Author(s)]. (2023). College enquiry chat bot using machine learning. International Journal for Research in Applied Science and Engineering Technology (IJRASET). <https://doi.org/10.22214/ijraset.2023.53538>
- XVI. [Author(s)]. (2023). AI chatbot for college enquiry. International Journal of Engineering and Management Research, 13(2), 90–93. <https://doi.org/10.31033/ijemr.13.2.13>
- XVII. [Author(s)]. (2023). AI and web-based human-like interactive college enquiry chatbot. Eduzone: International Peer Reviewed/Refereed Multidisciplinary Journal, 12(2), 204–207. <https://www.eduzonejournal.com/index.php/eiprmj/article/view/458>
- XVIII. [Author(s)]. (2023). An interactive chatbot for college enquiry. Journal of Computing and Communication, 2(1), 20–28. <https://doi.org/10.21608/jocc.2023.282081>
- XIX. [Author(s)]. (2022). Building a chatbot on a closed domain using RASA. <https://arxiv.org/abs/2208.06104>
- XX. Chan, J., & Li, Y. (2025). Enhancing higher education with generative AI: A multimodal approach for personalised learning. <https://arxiv.org/abs/2502.07401>
- XXI. Chukwuere, J. E. (2024). The future of generative AI chatbots in higher education. <https://arxiv.org/abs/2403.13487>
- XXII. Tang, J., Chen, S., & Shang, Y. (2025). TigerGPT: A new AI chatbot for adaptive campus climate surveys. <https://arxiv.org/abs/2504.13925>
- XXIII. Li, T., Ji, Y., & Zhan, Z. (2024). Expert or machine? Comparing the effect of pairing student teacher with in-service teacher and ChatGPT on their critical thinking, learning performance, and cognitive load in an integrated-STEM course. Asia Pacific Journal of Education, 44(1), 45–60. <https://doi.org/10.1080/02188791.2024.2305163>
- XXIV. Lee, Y., Hwang, G., & Chen, P. (2022). Impacts of an AI-based chatbot on college students' after-class review, academic performance, self-efficacy, learning attitude, and motivation. Educational Technology Research and Development, 70(5), 1843–1865. <https://doi.org/10.1007/s11423-022-10142-8>
- XXV. McDonald, K., Smith, J., & Lee, A. (2025). International perspectives on artificial intelligence in higher education. Education and Information Technologies. <https://doi.org/10.1007/s10639-025-13492-x>
- XXVI. Zhao, C., De Maria, R., Kumarage, T., Chaudhary, K. S., Agrawal, G., Li, Y., Park, J., Deng,

- Y., Chen, Y.-C., & Liu, H. (2025). CyberBOT: Towards reliable cybersecurity education via ontology-grounded retrieval augmented generation. arXiv. <https://arxiv.org/abs/2504.00389>
- XXVII. Sajja, R., Sermet, Y., Cikmaz, M., Cwiertny, D., & Demir, I. (2023). Artificial intelligence-enabled intelligent assistant for personalized and adaptive learning in higher education. <https://arxiv.org/abs/2309.10892>
- XXVIII. Neupane, S., Hossain, E., Keith, J., Tripathi, H., Ghiasi, F., Golilarz, N. A., Amirlatifi, A., Mittal, S., & Rahimi, S. (2024). From questions to insightful answers: Building an informed chatbot for university resources. arXiv. <https://arxiv.org/abs/2405.08120>
- XXIX. Lam, K. N., Le, N. N., & Kalita, J. (2022). Building a chatbot on a closed domain using RASA. arXiv. <https://arxiv.org/abs/2208.06104>
- XXX. Joshi, V., Mahajan, O., Hatgaonkar, O., Koli, R., & Kawar, R. (2023). AI and web-based human-like interactive college enquiry chatbot. Eduzone: International Peer Reviewed/Refereed Multidisciplinary Journal, 12(2), 204–207. <https://www.eduzonejournal.com/index.php/eiprmj/article/view/458>
- XXXI. Tiwari, R., Khandelwal, R., Agrawal, Y., Tiwari, V., & Bisen, W. H. (2023). AI chatbot for college enquiry. International Journal of Engineering and Management Research, 13(2), 90–93. <https://doi.org/10.31033/ijemr.13.2.13>
- XXXII. Mothankar, T. P., Maski, P. S., Ukey, S., & Asatkhar, P. S. (2021). Artificial intelligence-based college enquiry chatbot. International Journal of Computational and Electronic Aspects in Engineering, 2(2), 33–37. <https://doi.org/10.26706/ijceae.2.2.20210411>
- XXXIII. Gawade, H., Patil, V., Vishe, P., & Kolpe, S. (2020). College enquiry chat-bot system. International Journal of Engineering Research & Technology (IJERT), 9(9). <https://doi.org/10.17577/IJERTV9IS090396>

# APPENDIX-A

## SCREENSHOTS

The screenshot shows a web browser window with the URL [127.0.0.1:5500/index.html](http://127.0.0.1:5500/index.html). The page title is "Technical Education Department - Rajasthan". Below the title, a sub-header reads "Your gateway to technical education in Rajasthan". A large image of a modern university campus building is displayed. To the right of the image, the heading "About Us" is visible, followed by a detailed description of the department's mission and offerings. At the bottom right of the page, there is a "Chat with Us" button. The taskbar at the bottom of the screen shows various application icons.

The screenshot shows a web browser window with the URL [127.0.0.1:5500/index.html](http://127.0.0.1:5500/index.html). The page displays two college profiles side-by-side: "JECRC University, Jaipur" and "Swami Keshvanand Institute of Technology (SKIT), Jaipur". Each profile includes a photo of the college building, the college name, its location (Jaipur), a brief description of its programs, and a list of courses offered. To the right of the profiles, a "Student Assistance Chatbot" is open in a sidebar. The chatbot has a conversation with the user, asking how it can help and providing options to view colleges or check eligibility. The taskbar at the bottom of the screen shows various application icons.

New Tab    Technical Education Department

127.0.0.1:5500/index.html



### JECRC University, Jaipur

**Location:** Jaipur

JECRC University is a private university offering programs in engineering, management, and computer applications. It is known for its industry-oriented curriculum and excellent placement record.

**Courses Offered:**

- B.Tech (₹1,00,000/year)
- M.Tech (₹1,20,000/year)
- MBA (₹1,00,000/year)
- BCA (₹60,000/year)



### Swami Keshvanand Institute of Technology (SKIT), Jaipur

**Location:** Jaipur

SKIT is a renowned engineering college offering programs in engineering and management. It focuses on holistic development and industry readiness.

**Courses Offered:**

- B.Tech (₹85,000/year)
- M.Tech (₹1,10,000/year)
- MBA (₹95,000/year)

**Contact:** info@skit.ac.in

**Chat with Us**

**Student Assistance Chatbot**

View Colleges

Check Eligibility

**Colleges:**

Rajasthan Technical University

MNIT Jaipur

Government Engineering College Ajmer

JECRC University

SKIT Jaipur

New Tab    Technical Education Department

127.0.0.1:5500/index.html



### JECRC University, Jaipur

**Location:** Jaipur

JECRC University is a private university offering programs in engineering, management, and computer applications. It is known for its industry-oriented curriculum and excellent placement record.

**Courses Offered:**

- B.Tech (₹1,00,000/year)
- M.Tech (₹1,20,000/year)
- MBA (₹1,00,000/year)
- BCA (₹60,000/year)

**Contact:** info@jercruniversity.edu.in



### Swami Keshvanand Institute of Technology (SKIT), Jaipur

**Location:** Jaipur

SKIT is a renowned engineering college offering programs in engineering and management. It focuses on holistic development and industry readiness.

**Courses Offered:**

- B.Tech (₹85,000/year)
- M.Tech (₹1,10,000/year)
- MBA (₹95,000/year)

**Contact:** info@skit.ac.in

**Chat with Us**

**Student Assistance Chatbot**

Fee structure for Government Engineering College Ajmer:  
("B.Tech":₹75,000 per year", "M.Tech":₹90,000 per year")

Please enter your marks (percentage):

Submit

Your marks: 80%. Please select a course:

B.Tech
M.Tech
MBA
  
Ph.D.
BCA

The screenshot shows a web browser window with two tabs open. The active tab displays information about JECRC University, Jaipur and Swami Keshvanand Institute of Technology (SKIT), Jaipur. The JECRC profile includes a location (Jaipur), a brief description of the university's offerings, and a list of courses offered: B.Tech (₹1,00,000/year), M.Tech (₹1,20,000/year), MBA (₹1,00,000/year), and BCA (₹60,000/year). The SKIT profile also includes a location (Jaipur) and a brief description of the college's focus on holistic development and industry readiness. Both profiles have a contact email listed: info@jercuniversity.edu.in for JECRC and info@skit.ac.in for SKIT. To the right of the profiles is a 'Student Assistance Chatbot' window titled 'Chat with Us'. It asks if the user is eligible for B.Tech (80%) and lists several colleges based on this criterion: Rajasthan Technical University (Cutoff: 75%, Location: Kota), Government Engineering College Ajmer (Cutoff: 70%, Location: Ajmer), JECRC University (Cutoff: 75%, Location: Jaipur), and SKIT Jaipur (Cutoff: 72%, Location: Jaipur). The chatbot also asks if the user would like to check another course or view colleges. The browser taskbar at the bottom shows various pinned icons and the date/time as 5/1/2025, 10:37 PM.

This screenshot is nearly identical to the one above, showing the same two university profiles and the 'Student Assistance Chatbot' interface. The only difference is the date and time displayed on the browser taskbar, which now shows 5/1/2025, 10:38 PM. The rest of the content, including the university details and the list of recommended colleges, remains the same.

## **APPENDIX-B**

### **ENCLOSURES**

- 1. Journal publication/Conference Paper Presented Certificates of all students.**
- 2. Include certificate(s) of any Achievement/Award won in any project-related event.**
- 3. Similarity Index / Plagiarism Check report clearly showing the Percentage (%). No need for a page-wise explanation.**
- 4. Details of mapping the project with the Sustainable Development Goals (SDGs).**