Centenary Celebrated Sharnbasveshwar Vidya Vardhaka Sangha's

ಶರಣಬಸವ ವಿಶ್ವವಿದ್ಯಾಲಯ

SHARNBASVA UNIVERSITY

A State Private University approved by Govt. of Karnataka vide Notification No. ED 144 URC 2016 dated 29-07-2017
Recognised by UGC under Section 2f vide No. F.8-29/2017 (CPP-I/PU), dated 20-12-2017 & AICTE, CoA, PCI New Delhi

A

PROJECT REPORT

ON

"COLLEGE INFORMATION CHATBOT"

Submitted to

Department of Computer Science & Design
in partial fulfilment of the 3rd Semester Project

Submitted by

BHAVANESHKUMAR	**(SG24CSD010)**
GAGAN	**(SG24CSD020)**
KUNAL	**(SG24CSD030)**
ROHIT	**(SG24CSD045)**

Under the Guidance of:

PROF. SOUMYA MITEKAR

DEPARTMENT OF
COMPUTER SCIENCE & DESIGN

FACULTY OF ENGINEERING AND TECHNOLOGY [CO-ED]

SHARNBASVA UNIVERSITY, KALBURAGI

2025-2026

Centenary Celebrated Sharnbasveshwar Vidya Vardhaka Sangha's

ಶರಣಬಸವ ವಿಶ್ವವಿದ್ಯಾಲಯ

SHARNBASVA UNIVERSITY

A State Private University approved by Govt. of Karnataka vide Notification No. ED 144 URC 2016 dated 29-07-2017
Recognised by UGC under Section 2f vide No. F.8-29/2017 (CPP-I/PU), dated 20-12-2017 & AICTE, CoA, PCI New Delhi

CERTIFICATE

This is to certify that the project titled **"COLLEGE INFORMATION CHATBOT"** is a Bonafide work carried out by **BHAVANESHKUMAR (SG24CSD010), GAGAN (SG24CSD020), KUNAL (SG24CSD030), ROHIT (SG24CSD045)** in partial fulfilment of B. Tech 3rd Semester in Computer Science and Design of the Faculty of Engineering and Technology (Co-Education), SHARNBASVA UNIVERSITY, Kalaburagi during the year 2025-2026.

NAME	USN
BHAVANESHKUMAR	**SG24CSD010**
GAGAN	**SG24CSD020**
KUNAL	**SG24CSD030**
ROHIT	**SG24CSD045**

GUIDE **CHAIRPERSON** **DEAN**

Name of the Examiners **Signature with date**

1.

2.

ACKNOWLEDGEMENT

Any achievement, be scholastic or otherwise does not depend solely on the individual efforts but also on the guidance, encouragement and co-operation of intellectuals, elders and friends. A number of personalities, in their own capacities have helped us in carrying out this project work. We would like to take this opportunity to thank them all.

First of all, We would like to express our immense gratitude to Dr. Anilkumar Bidve, Vice Chancellor, Sharnbasva University, Kalaburagi for his help and inspiration during the tenure of the course.

We also extend our sincere thanks to Dr. S G Dollegoudar, Registrar, Sharnbasva University, Kalaburagi.

We also extend our sincere thanks to Dr. S S Honnali, Registrar (Evaluation) Sharnbasva University, Kalaburagi.

We also extend our sincere thanks to Dr. Laxmi Patil Maka, Dean Sharnbasva University, Kalaburagi.

We also extend our sincere thanks to Dr. Kiran Maka, Finance Officer, Sharnbasva University, Kalaburagi.

We also extend our sincere thanks to Dr. Shivakumar Jawalgi, Dean Faculty of Engineering & Technology-Co.Edu. Sharnbasva University, Kalaburagi.

We also extend our sincere thanks to Chairman Dr. Sachinkumar Veerashetty, Department of Computer Science and Design, Faculty of Engineering and Technology-Co-Education, for his constant encouragement.

We also extend our sincere thanks to my Guide Prof. Soumya Mitekar, Assistant Professor, Department of Computer Science & Design, Faculty of Engineering and Technology Co-Education, for his constant encouragement.

We also extend our sense of gratitude and sincere thanks to all the faculty members of Computer Science & Design, Faculty of Engineering and Technology-Co-Education for their constant encouragement and support.

BHAVANESHKUMAR	**(SG24CSD010)**
GAGAN	**(SG24CSD020)**
KUNAL	**(SG24CSD030)**
ROHIT	**(SG24CSD045)**

ABSTRACT

In today's fast-paced digital era, educational institutions face challenges in efficiently managing and responding to a large volume of student and parent enquiries. Traditional enquiry systems, such as help desks, phone calls, and manual responses, are often time-consuming, limited to working hours, and prone to human error. To overcome these limitations, the College Information Chatbot is proposed as an intelligent and automated solution for handling college-related information queries.

The College Information Chatbot is an AI-based conversational system that uses Natural Language Processing (NLP) and Machine Learning (ML) techniques to interact with users in a human-like manner. It allows users to ask questions in simple natural language and receive instant, accurate, and relevant responses. The chatbot provides information related to admissions, eligibility criteria, courses offered, fee structure, academic calendar, examination schedules, results, faculty details, departments, campus facilities, events, hostel facilities, and placement opportunities.

The system is available 24/7, ensuring uninterrupted access to information without requiring human intervention. By automating repetitive queries, the chatbot significantly reduces the workload of administrative staff and minimizes response time. It also ensures consistency and reliability in the information delivered to users. The chatbot can be deployed on a college website, mobile application, or messaging platforms, making it easily accessible to students, parents, and prospective applicants.

In addition, the proposed system supports scalability and future enhancements such as multilingual support, voice-based interaction, integration with college databases, and personalized responses based on user profiles. Overall, the College Information Chatbot improves communication efficiency, enhances user satisfaction, and contributes to the digital transformation of educational institutions.

CONTENTS

CHAPTER 1

PREAMBLE

1.1 INTRODUCTION

In the digital era, educational institutions are increasingly adopting technology to improve communication and information accessibility. Colleges generate a large amount of information related to admissions, courses, examinations, fees, departments, events, and facilities. Traditionally, students and parents depend on notice boards, websites, or administrative offices to obtain this information, which can be time-consuming and inefficient. To overcome these challenges, a College Information Chatbot is introduced.

A college information chatbot is an intelligent, automated software application designed to interact with users through natural language. It provides instant responses to user queries related to college activities and services. The chatbot uses technologies such as Artificial Intelligence (AI), Natural Language Processing (NLP), and machine learning to understand user questions and deliver accurate information in real time.

The primary purpose of the college information chatbot is to reduce the workload of administrative staff and offer 24/7 assistance to students, parents, and visitors. It ensures quick access to reliable information, improves user experience, and minimizes delays caused by manual enquiry systems. The chatbot can be integrated with college databases and websites, making it a smart and efficient solution for modern educational institutions.Overall, the college information chatbot plays a vital role in enhancing communication, improving efficiency, and supporting digital transformation in colleges.

1.2 OVERVIEW OF JAVA

Java is a high-level, object-oriented, and platform-independent programming language developed by Sun Microsystems and later acquired by Oracle. It follows the principle of “Write Once, Run Anywhere,” meaning that Java programs can run on any system that has a Java Virtual Machine (JVM), regardless of hardware or operating system. Java is known for its simplicity, robustness, and security, as it provides features such as automatic memory management, exception handling, and strong security mechanisms. It supports multithreading, which allows multiple tasks to run simultaneously and improves application performance. Java is widely used for developing web applications, mobile applications (especially Android), enterprise systems, and distributed applications. Due to its reliability, portability, and extensive libraries, Java remains one of the most popular programming languages in the software industry.

1.3 PROBLEM DEFINITION

In many educational institutions, information dissemination still relies on manual processes such as inquiry desks, printed notices, and static websites. These traditional systems present several limitations:

Users often find it difficult to locate specific information quickly.Administrative staff spend a significant amount of time answering repetitive questions.Help desks are available only during working hours, creating inconvenience for students who need information urgently.Websites may not always provide updated or easily navigable information.Large volumes of inquiries during admission periods cause delays and confusion.Due to these challenges, there is a strong need for an intelligent, automated system that can provide accurate information instantly and operate continuously without human intervention.

1.4 MOTIVATION

The motivation behind developing a College Information Chatbot arises from the need to provide quick, accurate, and continuous access to college-related information. In many educational institutions, students and parents frequently approach administrative offices for basic queries regarding admissions, courses, fees, timetables, examinations, and facilities. Handling these repetitive queries manually increases the workload on staff and often results in delays and miscommunication.

With the increasing use of digital platforms and smartphones, users expect instant responses and 24/7 availability of information. Traditional systems such as notice boards, static websites, and help desks are limited in accessibility and efficiency. This creates a gap between information seekers and information providers.

The college information chatbot is motivated by the need to:

1. Automate routine enquiry handling
2. Reduce dependency on administrative personnel
3. Provide instant and reliable responses at any time
4. Improve user experience through conversational interaction
5. Support digital transformation in educational institutions

By integrating technologies like Artificial Intelligence (AI) and Natural Language Processing (NLP), the chatbot can understand user queries and respond effectively. Thus, the motivation of the college information chatbot is to enhance communication efficiency, save time, reduce operational costs, and modernize the college information system.

1.5 OBJECTIVES

- To offer instant, accurate, and reliable information related to college activities, admissions, courses, and academic services.
- To reduce the workload on administrative and support staff by automating repetitive queries.
- To provide 24/7 availability so that users can access information at any time, from anywhere.
- To enhance user experience by enabling interactive, human-like conversation.
- To centralize information and ensure consistent communication across different departments.
- To improve overall efficiency of the college’s information management system.
- To support multiple query types such as text-based questions, FAQs, and guided navigation.

1.6 FUTURE SCOPE

- The College Information Chatbot has significant potential for expansion and improvement. Some future enhancements may include:
- Voice-enabled interaction, allowing users to speak directly to the chatbot using speech recognition.
- Personalized student services by integrating with internal databases to provide attendance, marks, timetables, and notifications.
- Machine learning capabilities to help the chatbot learn from user queries and improve accuracy over time.
- Multi-language support to assist users from different linguistic backgrounds.
- Mobile application integration for more accessible and user-friendly interaction.
- Automated form submission, such as admission forms, feedback forms, and complaint registration.
- Predictive assistance, guiding students in career choices, course planning, and exam preparation.
- Integration with social media platforms for broader communication and quick updates.

CHAPTER 2

LITERATURE SURVEY

2.1 RELATED WORKS

With the rapid growth of artificial intelligence and natural language processing, chatbots have emerged as an effective solution for automating information services in educational institutions. Several researchers have proposed and implemented college information and enquiry chatbots to improve accessibility, reduce administrative workload, and enhance student interaction.

[1]Ingale et al. (2024) developed a college enquiry chatbot using the RASA framework, focusing on intent recognition and entity extraction. Their system demonstrated efficient handling of student queries related to admissions, courses, and general college information. The study highlighted the advantage of using open-source NLP frameworks for building scalable and customizable chatbot systems.

[2]Gayathri et al. (2022) proposed a college enquiry chatbot system based on artificial intelligence techniques. The chatbot was designed to provide accurate responses to frequently asked questions and reduce dependency on manual enquiry systems. Their work emphasized improved response time, better user experience, and reduced human effort in handling repetitive queries.

[3]Pawar et al. (2018) presented a web-based college enquiry chatbot integrated with result-related information. The system enabled students to access academic and administrative details through a conversational interface. The authors concluded that chatbots significantly enhance efficiency compared to traditional enquiry methods and can be easily integrated with web applications.

[4]Sneha et al. (2019) introduced a college enquiry bot aimed at assisting students by providing instant responses to common queries. The system focused on ease of use and accuracy, demonstrating that chatbots can serve as reliable virtual assistants in educational environments.

[5]A systematic review published in AIJR Proceedings (2025) analyzed various chatbot applications in higher education institutions. The review concluded that chatbots play a crucial role in student support services, particularly in information dissemination, admission assistance, and administrative communication. It also highlighted challenges such as limited contextual understanding and the need for continuous learning.

[6]Peyton et al. (2025) reviewed university chatbots used for student support, including FAQs and administrative services. Their study emphasized the growing adoption of chatbots in universities and identified future directions such as personalization, multilingual support, and integration with institutional databases.

2.2 EXISTING SYSTEM

- Students visit the college website and search pages manually for information.
- They call or email the administration / help desk / departments for doubts.
- For simple queries (fees, course details, bus timings, exam schedules), students often rely on notices, circulars, or seniors.
- Limitations of the existing (manual / normal website) system:
- No 24×7 human support.
- Repeated answering of same questions by staff.
- Students spend more time searching on website or waiting for replies.
- No single interactive platform where a student can ask any college-related question in natural language.
- Difficult to personalize responses (e.g., branch-wise, year-wise information).

ADVANTAGES

- 24×7 Availability – Students can ask queries anytime, from anywhere, using just internet access.
- Instant Responses – Reduces waiting time compared to visiting offices or sending emails.
- Reduces Staff Workload – Frequently asked questions (FAQ) are answered automatically, so administrative staff can focus on more important tasks.
- User-Friendly Interface – Natural language interaction (type like chatting with a person) instead of searching multiple web pages.
- Consistency of Information – All students get the same, updated information from a centralized knowledge base.
- Scalability – Can handle a large number of students at the same time (peak times: admissions, exam results, etc.).
- Data Analytics – Admin can see most frequently asked questions and improve services or update FAQs accordingly.

DISADVANTAGES

- Technical Requirements – Needs internet connection, hosting, maintenance, and integration with college website / databases.
- Privacy & Security Concerns – If integrated with student data (Marks, profile, etc.), security and access-control must be handled carefully.

2.3 PROPOSED SYSTEM

Proposed system presents a College Information Chatbot designed to provide instant, accurate, and user-friendly information related to college activities using modern conversational technology. The system aims to reduce manual effort and improve accessibility of information for students, parents, and visitors. chatbot system is developed using Artificial Intelligence (AI) and Natural Language Processing (NLP) techniques, which enable it to understand user queries and respond in a human-like manner. The chatbot acts as a virtual assistant capable of answering frequently asked questions regarding admissions, courses offered, departments, faculty details, fee structure, examination schedules, placement information, and campus facilities.

Proposed system consists of two major components: Frontend Interface and Backend Processing Unit. The frontend provides a simple and interactive chat interface through a web or mobile application, where users can type their queries. The backend uses a trained chatbot model that processes the input, identifies user intent, and fetches relevant information from the database. Chatbot utilizes a centralized database to store structured college-related information. Whenever a user submits a query, the system matches the input with predefined intents or keywords using NLP techniques and retrieves the most appropriate response. If the user query is unclear, the chatbot requests additional input to improve response accuracy.

Proposed system operates 24/7 and eliminates the need for human operators for routine enquiries. It ensures quick response time, consistency in information delivery, and scalability to handle multiple users simultaneously. The system can be further enhanced by integrating voice input, multilingual support, and real-time notifications.

Overall, the proposed College Information Chatbot system improves communication efficiency, reduces administrative workload, and enhances the user experience by providing reliable and instant information through an automated conversational platform.

CHAPTER 3

SYSTEM REQUIREMENT SPECIFICATION

3.1 SOFTWARE REQUIREMENTS

Front-end Requirements:

HTML, CSS, JavaScript – For designing user interface (web-based chatbot).

Mobile App Framework (optional) – Flutter / React Native if deploying on mobile.

Back-end Requirements:

Java (Spring Boot) – for running the backend code

3.2 HARDWARE REQUIREMENTS

Server-side Hardware :

If hosted locally or on a private server.

Processor: Minimum Intel i5 or equivalent

RAM: 8 GB (16 GB recommended for ML-based chatbot)

Storage: 250 GB HDD or 128–256 GB SSD

Network: Stable internet connection for real-time responses

If hosted on cloud, hardware requirements for server are minimal.

Client-side Hardware (Users) :

Any student or faculty member accessing the chatbot requires:

Smartphone / Laptop / Desktop

Processor: Dual-core CPU or better

RAM: 2–4 GB

Storage: No special requirement

Internet Connectivity: 3G/4G/5G or Wi-Fi

CHAPTER 4

IMPLEMENTATION

4.1 PROJECT IMPLEMENTATION

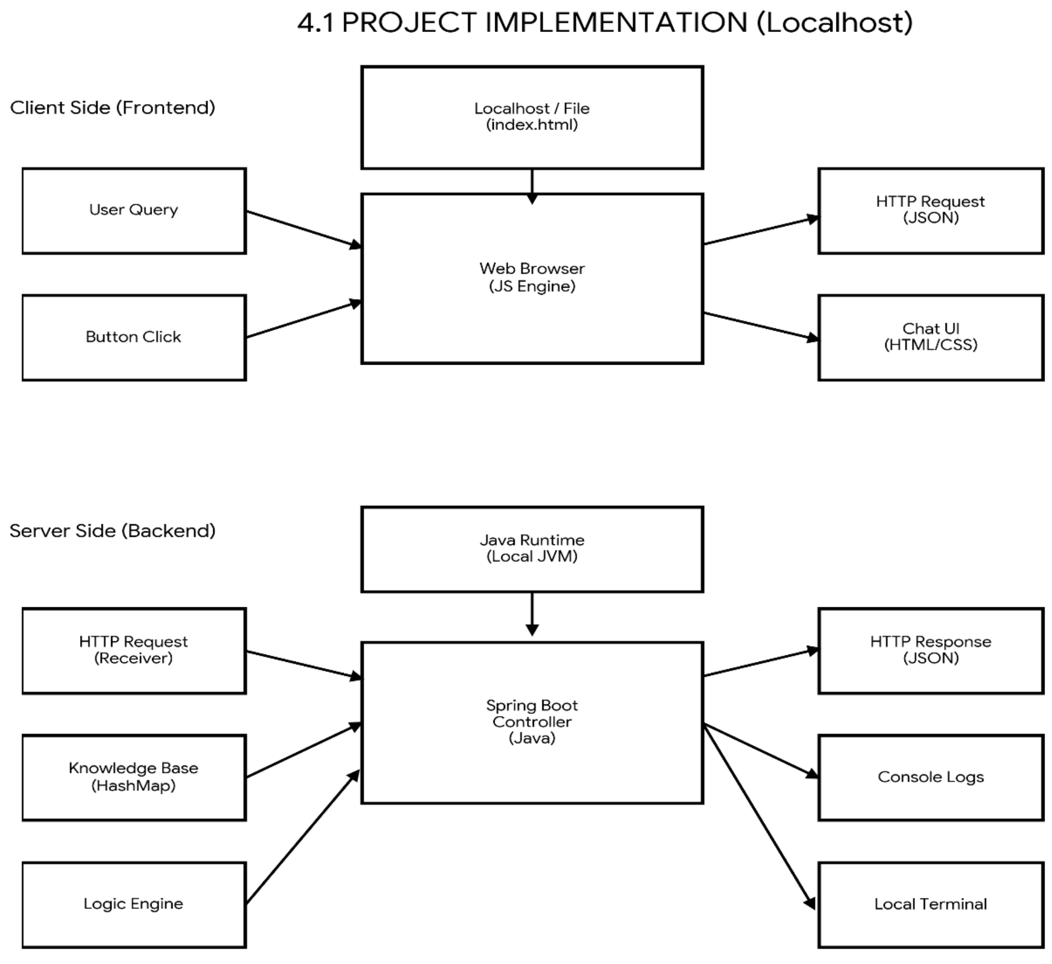


Fig 4.1: Implementation Diagram of College Chatbot

Implementating of the College Information Chatbot on a local environment involves setting up a Client-Server architecture where both the "Frontend" and "Backend" run simultaneously on the same machine. The browser acts as the client, communicating with the Java application running in the background via the local loopback address (127.0.0.1).

1. Project Overview

Objective: To develop and run a locally hosted chatbot that simulates an intelligent university enquiry system, providing instant responses about Sharnbasva University directly from the developer's machine.

Key Components:

- **Backend:** Java Development Kit (JDK 17+), Spring Boot Framework, Apache Maven.
- **Frontend:** HTML5, CSS3, JavaScript (ES6).
- **Server:** Embedded Tomcat Server (running on port 8080).
- **IDE:** IntelliJ IDEA / Eclipse.

2. System Components (Local Environment)

- **Local Server (The Engine):** The Spring Boot application initializes an embedded Tomcat web server. It listens for incoming data on port 8080.
- **Controller Logic (The Brain):** A Java class ChatController that intercepts HTTP POST requests sent to http://localhost:8080/api/chat.
- **Static File Host (The Display):** The index.html file is served directly from the local file system or the static resources folder of the Spring Boot app.

3. Backend Implementation (Java)

Setup:

- A Maven project structure was created with dependencies for spring-boot-starter-web.
- The pom.xml file was configured to manage build plugins and dependencies.

Logic Programming:

- An internal HashMap (Knowledge Base) was populated with university data (Fees, Courses, Faculty) upon application startup.
- A REST Controller was programmed to parse JSON input. Regular Expressions (Regex) were used to detect keywords like "Placement" or "Holiday".
- CORS Configuration: Cross-Origin Resource Sharing was enabled to allow the HTML file (running as a file) to talk to the Java server (running on localhost).

4. Frontend Implementation (HTML/JS)

Interface Design:

- A modern chat interface was built using CSS Grid and Flexbox.
- "Quick Action Buttons" were implemented using JavaScript onclick events to send predefined text (e.g., "Courses") to the input field.

Local Communication:

- The JavaScript fetch() API was hardcoded to point to the local address: http://localhost:8080/api/chat.
- Asynchronous functions (async/await) were used to prevent the browser from freezing while waiting for the Java server to reply.

5. Execution Workflow (Running the Project)

Starting the Backend:

- The developer opens the project in the IDE (IntelliJ).
- The DemoApplication.java file is executed.
- The console displays Started DemoApplication in X.XX seconds, confirming the local server is active.

Launching the Frontend:

- The index.html file is opened via a standard web browser.
- The browser loads the UI and establishes a connection to the local backend.

Interaction:

- When the user types "Hello", the browser sends a JSON packet {"message": "Hello"} to localhost:8080.
- The Java application processes this, finds the greeting logic, and returns {"reply": "Hello! I am..."}.
- The browser receives this and appends the text to the chat window.

6. Testing & Validation

- **Connectivity Test:** Verified that the frontend successfully receives a "200 OK" status code from the local server.
- **Logic Test:** Confirmed that specific queries like "Who is the VC?" return the correct name (Dr. Anilkumar G. Bidve) from the local HashMap.
- **Error Handling:** Tested the system behavior when the Java server is stopped; the frontend correctly displays a "Server not reachable" error message.

7. Future Scope

- **Database Integration**: Replace the static HashMap with a local MySQL or H2 database for persistent data storage.
- **Voice Module**: Implement the Web Speech API in the frontend to allow microphone input on the local machine.
- **Deployment**: The working local project can be easily packaged into a JAR file and deployed to a cloud provider (like AWS or Render) for public access.

4.2 METHODOLOGY

The College Chatbot project methodology consists of 3 parts:

1. **Input Acquisition:** Capturing user intent through text or button clicks (replacing physical sensors).
2. **Processing & Decision:** The Server analyzes the input against the University Knowledge Base to find the best match.
3. **Response & Action:** The Server constructs a formatted HTML response and sends it back to the client for display.

4.3 SOURCE CODE

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sharnbasva University Assistant</title>
    <link
href="https://fonts.googleapis.com/css2?family=Inter:wght@300;400;500;600;700&display=swap" rel="stylesheet">
    <style>
        :root {
            --primary: #2563eb;
```

```
--primary-dark: #1e40af;
          --bg-gradient: linear-gradient(135deg, #f0f9ff 0%, #e0f2fe 100%);
          --chat-bg: #ffffff;
          --bot-msg-bg: #f3f4f6;
          --user-msg-bg: #2563eb;
          --text-main: #1f2937;
       }

       * { margin: 0; padding: 0; box-sizing: border-box; font-family: 'Inter', sans-serif; }

       body {
          background: var(--bg-gradient);
          height: 100vh;
          display: flex;
          justify-content: center;
          align-items: center;
       }
.chat-card {
          width: 420px;
          height: 700px;
          background: var(--chat-bg);
          border-radius: 20px;
          box-shadow: 0 20px 40px rgba(0,0,0,0.1);
          display: flex;
          flex-direction: column;
          overflow: hidden;
```

```
position: relative;
        }

        /* HEADER */
        .header {
            background: #ffffff;
            padding: 20px;
            border-bottom: 1px solid #e5e7eb;
            display: flex;
            align-items: center;
            gap: 15px;
        }
        .avatar {
            width: 45px;
            height: 45px;
            background: var(--primary);
            color: white;
            border-radius: 12px;
            display: flex;
            align-items: center;
            justify-content: center;
            font-weight: 700;
            font-size: 18px;
        }
        .header-text h2 { font-size: 16px; color: var(--text-main); font-weight: 700;
}
        .header-text p { font-size: 12px; color: #10b981; font-weight: 500; display:
```

```
flex; align-items: center; gap: 5px; }
    .dot { width: 8px; height: 8px; background: #10b981; border-radius: 50%; }

    /* CHAT AREA */
    .chat-box {
      flex: 1;
      padding: 20px;
      overflow-y: auto;
      display: flex;
      flex-direction: column;
      gap: 15px;
    }
    .message {
      max-width: 85%;
      padding: 12px 16px;
      border-radius: 16px;
      font-size: 14px;
      line-height: 1.5;
      animation: fadeIn 0.3s ease;
    }
    .bot { background: var(--bot-msg-bg); color: var(--text-main); align-self: flex-start; border-bottom-left-radius: 4px; }
    .user { background: var(--user-msg-bg); color: white; align-self: flex-end; border-bottom-right-radius: 4px; box-shadow: 0 4px 10px rgba(37,99,235,0.2); }

    /* Headers in messages */
    .bot h3 { margin-bottom: 8px; color: var(--primary); font-size: 15px; }
```

```
/* QUICK ACTION BUTTONS (The Chips) */
.quick-actions {
    padding: 10px 15px;
    background: #ffffff;
    display: flex;
    gap: 8px;
    overflow-x: auto;
    border-top: 1px solid #f3f4f6;
}
.quick-actions::-webkit-scrollbar { height: 4px; }
.quick-actions::-webkit-scrollbar-thumb { background: #e5e7eb; border-radius: 4px; }

.action-btn {
    white-space: nowrap;
    padding: 8px 14px;
    background: #eff6ff;
    color: var(--primary);
    border: 1px solid #dbeafe;
    border-radius: 20px;
    font-size: 13px;
    font-weight: 500;
    cursor: pointer;
    transition: all 0.2s;
}
.action-btn:hover { background: var(--primary); color: white; transform: translateY(-2px); }
```

```
/* INPUT AREA */
.input-area {
   padding: 15px;
   background: white;
   display: flex;
   gap: 10px;
   border-top: 1px solid #e5e7eb;
}
input {
   flex: 1;
   padding: 12px;
   border: 2px solid #e5e7eb;
   border-radius: 12px;
   outline: none;
   transition: 0.3s;
}
input:focus { border-color: var(--primary); }
.send-btn {
   background: var(--primary);
   color: white;
   border: none;
   padding: 0 20px;
   border-radius: 12px;
   cursor: pointer;
   font-weight: 600;
}
```

```
.send-btn:hover { background: var(--primary-dark); }

        @keyframes fadeIn { from { opacity: 0; transform: translateY(10px); } to { opacity: 1; transform: translateY(0); } }
    </style>
</head>
<body>

<div class="chat-card">
    <div class="header">
        <div class="avatar">SU</div>
        <div class="header-text">
            <h2>Sharnbasva University</h2>
            <p><span class="dot"></span> Official Assistant</p>
        </div>
    </div>

    <div class="chat-box" id="chat-box">
        <div class="message bot">
            Hello! 👋 Welcome to Sharnbasva University.<br><br>
            I can help you with <b>Admissions, Courses, Fees</b>, and more. Select an option below!
        </div>
    </div>

    <!-- Quick Action Buttons -->
    <div class="quick-actions">
```

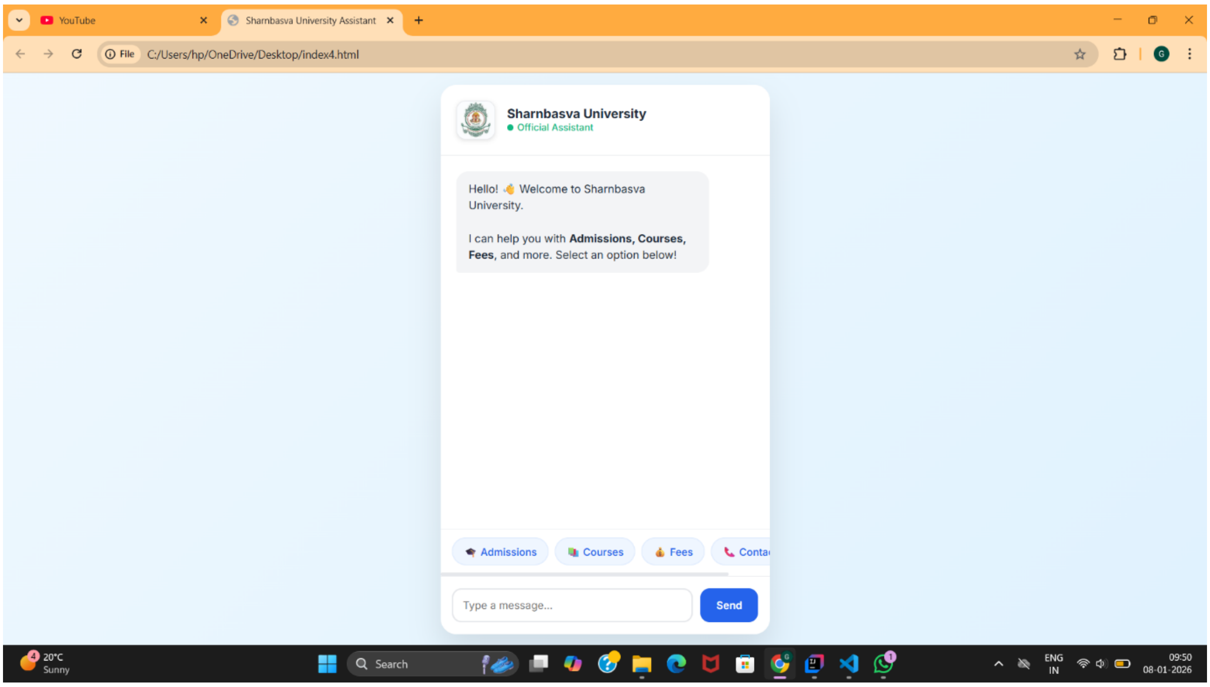
```
        <button class="action-btn" onclick="sendQuickMessage('🎓 Admissions')">🎓 Admissions</button>

        <button class="action-btn" onclick="sendQuickMessage('📚 Courses')">📚 Courses</button>

        <button class="action-btn" onclick="sendQuickMessage('💰 Fees')">💰 Fees</button>

        <button class="action-btn" onclick="sendQuickMessage('📞 Contact Us')">📞 Contact Us</button>

    </div>

<div class="input-area">

        <input type="text" id="user-input" placeholder="Type a message..." onkeypress="handleEnter(event)">

        <button class="send-btn" onclick="sendMessage()">Send</button>

    </div>

</div>

<script>

    // Handles clicking the Quick Action buttons

    function sendQuickMessage(text) {

        document.getElementById("user-input").value = text;

        sendMessage();

    }

async function sendMessage() {

        const inputField = document.getElementById("user-input");

        const chatBox = document.getElementById("chat-box");

        const message = inputField.value.trim();

 if (message === "") return;

 // 1. Add User Message
 chatBox.innerHTML += <div class="message user">${message}</div>;
        inputField.value = "";
        chatBox.scrollTop = chatBox.scrollHeight;
// 2. Send to Java Backend
        try { const response = await fetch("http://localhost:8080/api/chat", {
                method: "POST",
                headers: { "Content-Type": "application/json" },
                body: JSON.stringify({ message: message })
            });
              const data = await response.json();
            // 3. Add Bot Response
            // We use a small timeout to make it feel natural
            setTimeout(() => {chatBox.innerHTML += <div class="message
bot">${data.reply}</div>;
                chatBox.scrollTop = chatBox.scrollHeight;
            }, 300);
        } catch (error) {
            chatBox.innerHTML += <div class="message bot"
style="color:red">Error: Server not reachable.</div>;
        }
    }
    function handleEnter(e) {
        if (e.key === 'Enter') sendMessage();
    }
</script>
</body>
</html>
```

CHAPTER 5

RESULT INTERPRETATION

5.1 OUTCOMES

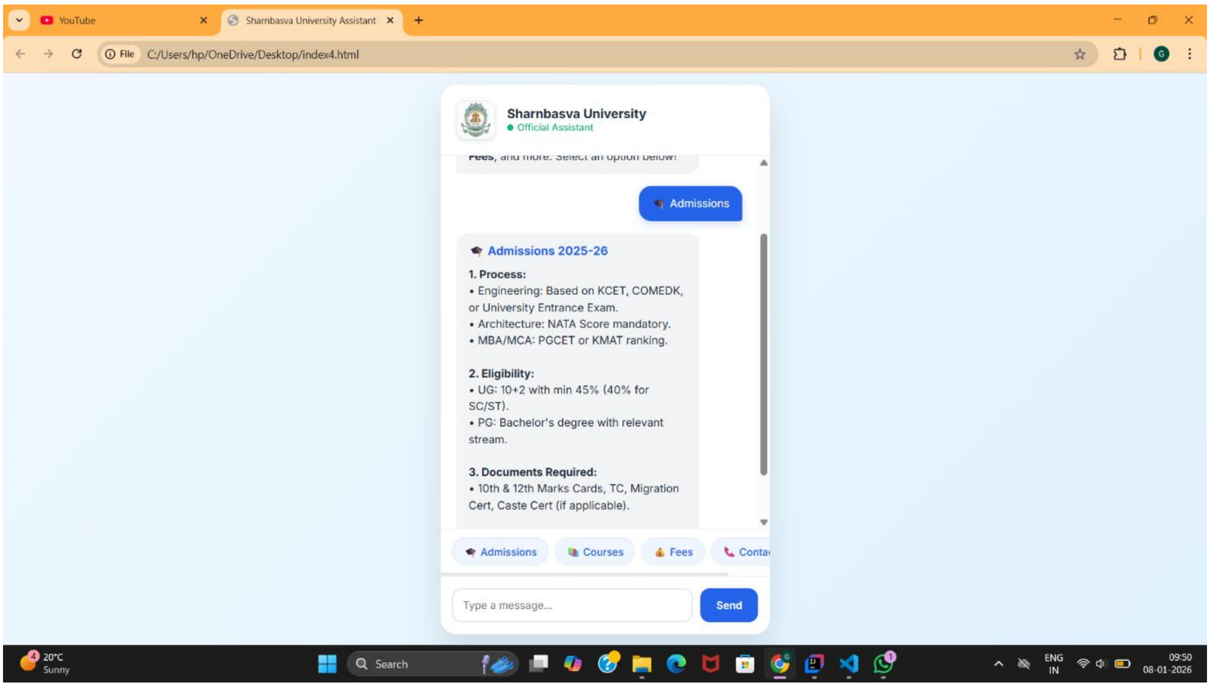
SNAPSHORT 01 :WEB PAGE



Hello! 👋 Welcome to Sharnbasva University.

I can help you with **Admissions, Courses, Fees**, and more. Select an option below!

SNAPSHORT 02 : ANSWERING THE QUESTIONS



Fees, and more. Select an option below!

Admissions

Admissions 2025-26

1. Process:
- Engineering: Based on KCET, COMEDK, or University Entrance Exam.
- Architecture: NATA Score mandatory.
- MBA/MCA: PGCET or KMAT ranking.

2. Eligibility:
- UG: 10+2 with min 45% (40% for SC/ST).
- PG: Bachelor's degree with relevant stream.

3. Documents Required:
- 10th & 12th Marks Cards, TC, Migration Cert, Caste Cert (if applicable).

CONCLUSION

The development of the College Information Chatbot represents a significant step toward modernizing communication and information-delivery systems within educational institutions. Traditional methods of addressing student queries—such as manual enquiry counters, phone calls, and emails—are often time-consuming, resource-intensive, and unable to provide real-time support. The chatbot overcomes these limitations by offering an automated, intelligent, and always-available platform that responds instantly to student queries with high accuracy.Through the integration of Artificial Intelligence, Natural Language Processing, and rule-based or machine-learning techniques, the chatbot understands user questions and delivers relevant information related to admissions, academic programs, fee structures, campus facilities, faculty details, placement information, and various college services. This not only enhances the accessibility of information but also ensures consistency in responses, reducing the chances of misinformation.From the institution's perspective, the chatbot significantly reduces the administrative burden on staff members who would otherwise need to handle repetitive queries. It ensures faster query resolution, improves operational efficiency, and allows staff to focus on more critical tasks. For students, especially new and prospective students, the chatbot acts as a reliable virtual guide that simplifies navigation through college systems and helps them make informed decisions.Moreover, the chatbot enhances user engagement by supporting a conversational interface, making interactions feel more natural and intuitive. As technology continues to advance, the system has the potential to incorporate additional features such as voice-based interaction, multilingual support, personalized recommendations, integration with academic portals, attendance tracking, and real-time notifications. These improvements can transform the chatbot into a comprehensive digital assistant that supports both academic and administrative processes.In conclusion, the College Information Chatbot is an innovative and future-ready solution that aligns with the growing need for digital transformation in higher education. It contributes to improved communication, better student satisfaction, and more efficient management practices. With continuous development and integration of advanced AI capabilities, the chatbot can evolve into a powerful and indispensable tool for both students and educational institutions.

REFERENCES

Ingale, N., Jha, T. A., Dixit, R., & Borate, V. (2024). College Enquiry Chatbot Using RASA. International Journal of Scientific Research in Science and Technology, 5(8), 210–215.

Gayathri, V., Saranya, V., Vijetha, A., Vijey, A., & SriRagavi, M.; Malarvizhi, K. (2022). College Enquiry Chatbot System using Artificial Intelligence. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 8(3), 289–293.

Pawar, S., Rane, O., Wankhade, O., & Mehta, P. (2018). A Web Based College Enquiry Chatbot with Results. International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET), 7(4).

Sneha, C., Srilekha, C., Prathusha, G. N., Ramana Murthy, B. V., & Kishor Kumar Reddy, C. (2019). College Enquiry Bot. Global Journal of Engineering Science and Researches (GJESR), ICITAIC-2019.

“A Systematic Review of Chatbot Applications in Higher Education Institutions”. (2025). AIJR Proceedings.

Peyton, K. et al. (2025). A review of university chatbots for student support: FAQs, administration and information services. (Journal, 2025) — discusses recent chatbot implementations in university settings.