



Tribhuvan University

Faculty of Humanities and Social Sciences

COLLEGE CHATBOT USING BM25 ALGORITHM

A PROJECT REPORT

Submitted To

Department of Computer Application

Aadim National College

In partial fulfillment of the requirements for the Bachelors in Computer Application

Submitted By:

Sapana Rai (105902063)

(Tu Registration Number: 6-2-1059-11-2020)

Jestha 2082

6th Semester

Under the supervision of

Er. Lokesh Gupta

Abstract

This project aims to design and develop to communication and interact between students and college administration by providing quick, automated response to frequently asked questions. This system is built using React, Express and MongoDB. The primary objective is to provide digital solution streamline the digital communication in real time.

Keywords: *College chatbot application, React.js, Node.js, Express.js, MongoDB, User interface, Cross-platform*

Acknowledgements

I would like to sincerely thank Tribhuvan University (TU) and Aadim National College for giving me the opportunity and support to carry out this project. The guidance, encouragement, and learning environment provided by the institution played a huge role in helping me complete this work successfully.

I'm especially grateful to my supervisor, Er. Lokesh Gupta, for his continuous support, honest feedback, and patient guidance throughout the project. His knowledge and encouragement helped me stay on the right path and complete this work with confidence.

A heartfelt thank you also goes to the Head of Department, Er. Prem Subedi, for creating a supportive and motivating environment that made this learning journey smoother.

I truly appreciate all the IT staff for being so cooperative and helpful whenever I needed information or technical support during the project.

Lastly, I'm incredibly thankful to my friends for their constant encouragement, positive energy, and moral support, which kept me motivated and focused from start to finish.

Table of Contents

Abstract.....	i
Acknowledgements.....	ii
List of Figures	v
List of Tables	vi
Chapter 1: Introduction	1
1.1 Introduction:.....	1
1.2 Problem Statement:.....	1
1.3 Objectives	1
1.4 Scope and Limitations	1
1.4.1 Scope	1
1.4.2 Limitations:	1
1.6 Report Organizations:	3
Chapter 2: Background Study and Literature Review	5
2.1 Background Study:.....	5
2.1.1 Manual Student Assistance Systems:.....	5
2.1.2 Basic Chatbot Systems:	5
2.2 Literature Review:	5
Chapter 3: System Analysis and Design.....	7
3.1 System Analysis	7
3.1.1 Requirement Analysis	7
3.1.2 Feasibility Analysis	8
3.2 System Design.....	9
3.2.1 Architecture Design	9
3.2.3 Interface Design	9
3.3 Algorithm Detail.....	10
Chapter 4: Implementation and Testing.....	11
4.1 Implementation	11
4.2.1 Test Cases for System Testing	12
Chapter 5: Conclusion and Future Recommendations	13
5.1 Conclusion.....	13

5.2 Lesson Learnt/Outcome.....	13
5.3 Future Recommendation	13

List of Figures

Figure 1.1 Chatbot Architecture	2
Figure 3.1 Use Case of College Chatbot.....	7
Figure 3.2 Architecture Design of College Chatbot	9
Figure 3.3 Database Schema Diagram of College Chatbot.....	9

List of Tables

Table 3.1 Functional Requirements of College Chatbot.....	8
Table 3.2 Non-Functional Requirements of College Chatbot.....	8
Table 4.1 College Chatbot System Test Case.....	12

Chapter 1: Introduction

1.1 Introduction:

College chatbot is a system to interact with new clients, students, faculty, staff etc. It provides answer to common questions and makes easy to communicate with new clients over the internet. This chatbot, powered by ML and chatbot architecture like ChatGPT, can answer admission-related query, about college, course fee and more. This system provides an online admission form. It can provide services 24/7 in real time.

1.2 Problem Statement:

Without college chatbot system clients, student, customers and college faces several challenges, lack of information about college, course, faculty, facilities and communication etc. New client, customers, and students often miss updates an exam, admission, fees, deadlines due to slow response times. College struggle to provides quick and accurate information about courses, academic resources, admissions etc.

1.3 Objectives

The main objectives of this project can be enumerated as follows:

- To implement chatbot for college website using BM25 algorithm.
- To develop a system to interact and communicate with student, clients, customers etc.

1.4 Scope and Limitations

College Chatbot System is use students to inquire about or seek help relative to academics, administration, or campus. A few of its uses may be:

1.4.1 Scope

This application is such that it is an online chatbot application for college-related queries. This is the system that the end users (students, faculty, and staff) and administrators can use to perform informative as well as interactive tasks as they interact with the users in obtaining college-related assistance quickly.

- Students and faculty can interact with the system for instant answers about courses, admission, academic calendar, exam schedule, and so on.
- At any given time, this system answers inquiries with regard to admissions, courses, deadlines, fees, and campus facilities.

1.4.2 Limitations:

Some of the limitations of this system are:

- May not be able to answer questions outside the database or syllabus unless updated manually.

- Does not address any emotional or sensitive issues adequately.
- Information about many colleges cannot be provided.

1.5 Development Methodology:

The methodology proposed for the system follows a Chatbot Architecture approach that splits the system, intent recognition, response generation, and data integration. This modular design supports the chatbot to work on user queries and associate the queries with corresponding intents to create responses either from pre-existing response patterns in the system.

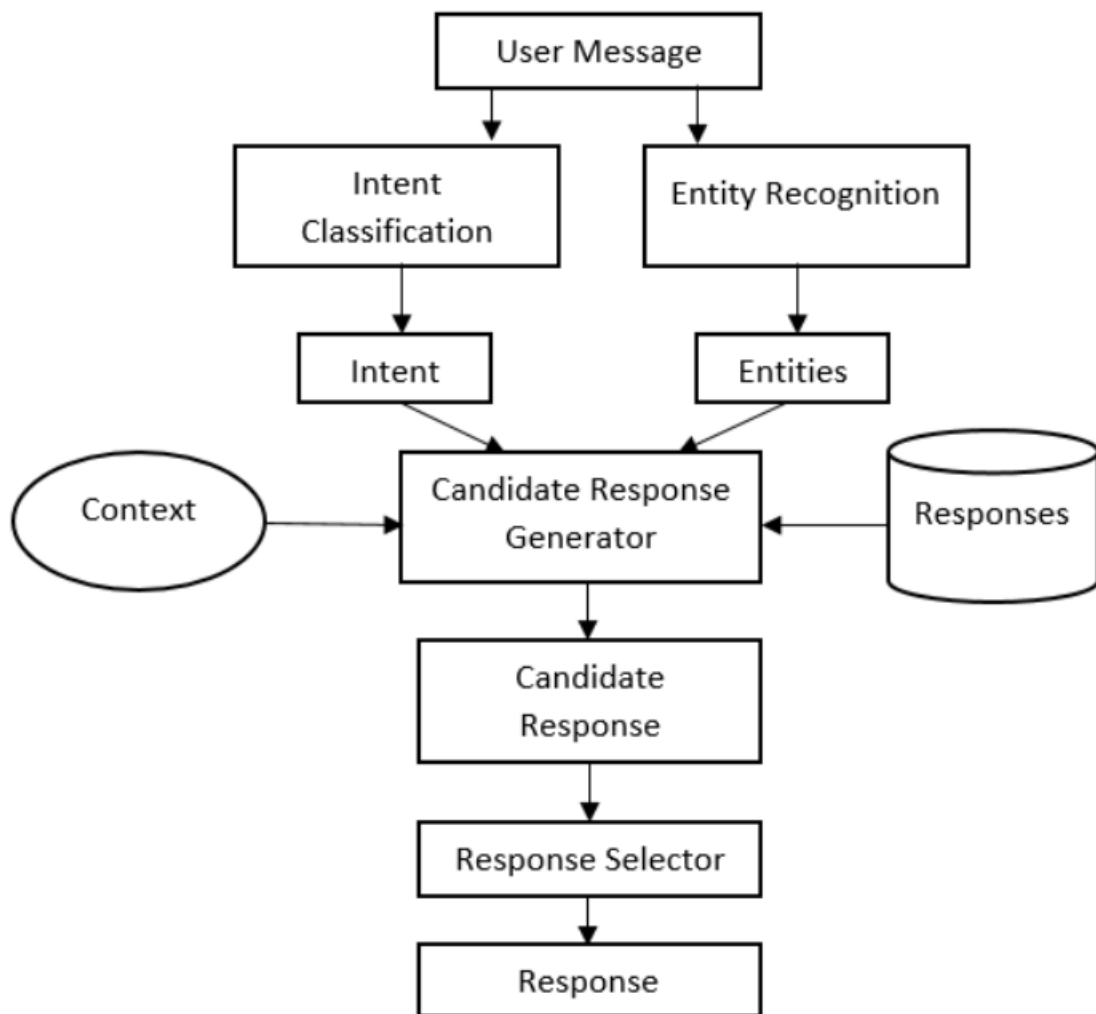


Figure 1.1 Chatbot Architecture

1.6 Report Organizations:

Chapter 1: Introduction

This chapter introduces the projects, showing their project significance and relevance within the field. The problem statement has been defined, indicating the issue the project is particularly targeting. The objectives are clearly stated in order to identify the results the project is supposed to arrive at. The scope and limitations have also been included to show the boundaries of the project. At the end, it gives an overview of the report structure for the readers' sake.

Chapter 2: Background Study and Literature Review

This chapter lays the foundation through a background study that elaborates on the necessary theories, concepts, and terminologies used in this domain. Second, it flows into an extensive literature review that amalgamates prior research, studies, and projects carried out related to the topic. The review provides a base and context for the present undertaking.

Chapter 3: System Analysis and Design

This chapter delves into system analysis, dealing with project requirements as functional and non-functional. Functional requirements are described by use case diagrams or lists; non-functional requirements consider technical, operational, economic, and schedule feasibilities. Data modelling is achieved using the ER diagram, while process modelling uses DFDs to represent system processes. Then, the study moves on to design, from system architecture to the design of a database schema, interface design concerning UI/UX considerations, and finally to physical DFDs, dealing with implementation details.

Chapter 4: Implementation and Testing

This chapter includes the phase of implementation and talks about the tools used, such as the CASE tools, programming languages, and database platforms. It gives a detailed implementation of the modules, detailing the procedures and functions developed for each module of the system. Testing comes next, which starts, initially, with unit testing to validate the functionality of individual components and culminates in system testing to validate overall system reliability and performance.

Chapter 5: Conclusion and Future Recommendations

The chapter assesses project outcomes and lessons learned. It contains a conclusive summary of the project and its achievements, highlighting the contributions that were made in the field. Recommendations for the future are also given, offering areas in which the

project may be enhanced or which merit further investigation to maintain its continued development and relevance.

Chapter 2: Background Study and Literature Review

2.1 Background Study:

When studying existing chatbot systems, it becomes clear that the current landscape is characterized by a mix of basic and advanced AI-powered solutions, each with its own set of strengths and limitations. The analysis covers various aspects, including communication methods, access to platforms, integration with educational systems, and the effectiveness of user interaction.

2.1.1 Manual Student Assistance Systems:

Strengths:

- Traditional student assistance operates by means of a front help desk where students can approach someone or through direct communication with the staff.
- Customization is offered in this manner, while in-person help may be preferred for very difficult queries.

Weaknesses:

- Provided during working hours most of the time.
- Hard to scale when the population grows.

2.1.2 Basic Chatbot Systems:

Strengths:

- Using bm25 algorithms to respond to predefined questions such as schedules of classes, exam dates, or admission requirements.
- Easy to implement and maintain for simple FAQs.

Weaknesses:

- Flexibility is limited and cannot accommodate questions outside predefined answer patterns.
- Poor user experience emanating from being repetitive and rigid.

2.2 Literature Review:

Leading ISP of Nepal, WorldLink Communications Ltd., has recently installed an AI-enabled chatbot for customers. It can be reached via the company's official website and WhatsApp. The chatbot assists users with tasks ranging from checking for Internet connectivity, retrieving passwords for those who have forgotten, solving issues with NETTV, to handling account inquiries. The chatbot functions with an unwavering time, so whatever the issue, there is concerned service almost immediately, which saves time as direct human intervention requirements lessen. It offers personalized recommendations and can entertain many simultaneous requests, ensuring services are not slowed down even

during busy times. Currently, the only language offered by the chatbot is English, which again limits some accessibility[1].

Chapter 3: System Analysis and Design

3.1 System Analysis

The chapter discusses the proposed model for the system, the methodology used to build the system, the tools and techniques, requirement analysis, and requirement specification. It further explains the functional requirements of the system and system design. This chapter shall also expose the architectural views of the application, use case diagram, data design etc.

3.1.1 Requirement Analysis

The following section presents the complete set of functional requirements of the college chatbot system.

i. Functional Requirements

Following figure shows the requirement of college chatbot system

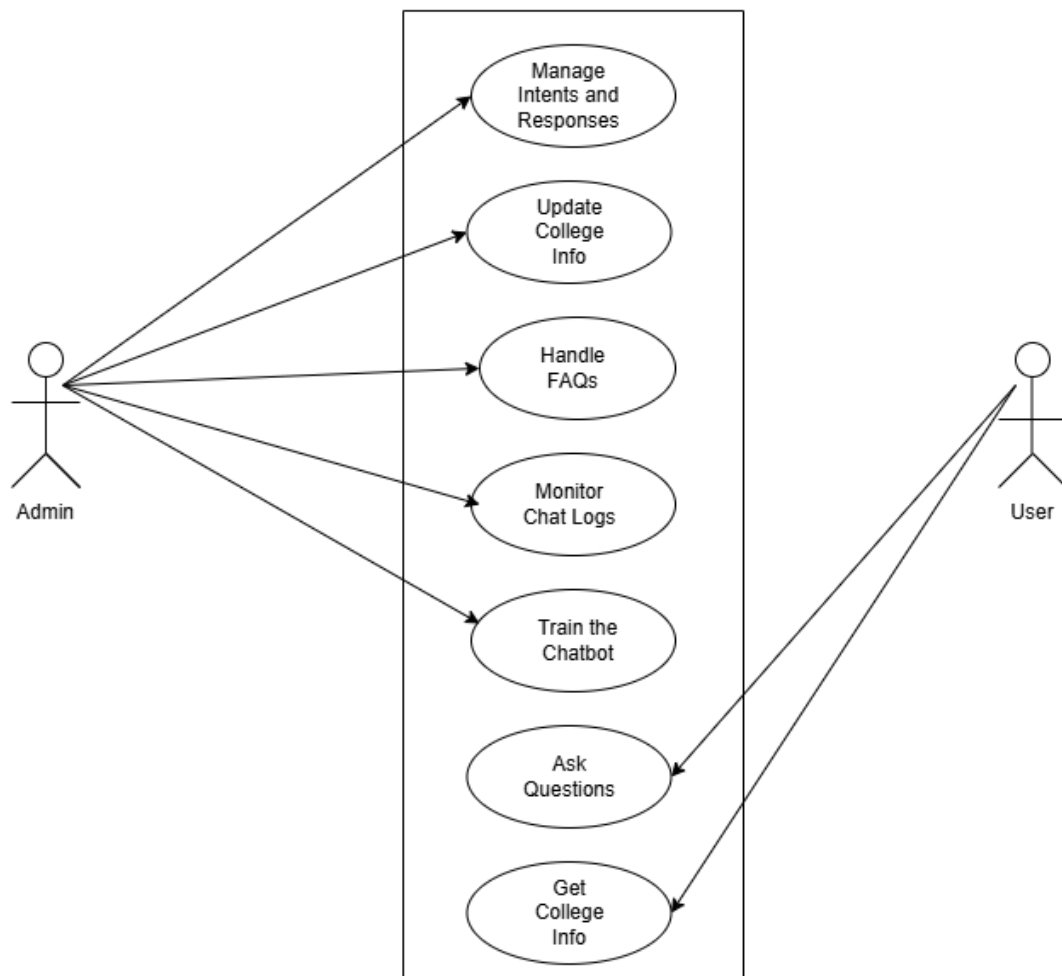


Figure 3.1 Use Case of College Chatbot

Table 3.1 Functional Requirements of College Chatbot

Req.No	Description	Type
R-101	The web application will contain user interface.	Function
R-102	The web application will require internet to operate.	Configuration
R-103	The web applications will allow navigating menus.	Function
R-104	The web applications will allow admin users to manage (add) about college information.	Function

ii. Non-Functional Requirements

Some of the contents of non-functional requirements are shown table below.

Table 3.2 Non-Functional requirements of College Chatbot

Req.No	Description	Type
NR-101	The web application shall ensure sensitive information is secure	Security
NR-102	The web application will be user friendly	Usability
NR-103	Other than admins will not be able to access the system	Security
NR-104	The web application shall run well on desktop and mobile devices	Configuration

3.1.2 Feasibility Analysis

Some important feasibility studies are mentioned bellow:

i. Technical Feasibility

This project is technical feasibility as already have hardware and software needed for a development-based type of software are available temporarily. Apart from that, my technical knowledge stands in how projects are made using certain programming languages like JavaScript, python etc.

ii. Operational Feasibility

It is operationally feasible that making this system by removing the threats and weakness of existing non manageable system which is reliable for the users.

iii. Economic Feasibility

The system does not require extra software and hardware. So, there is no recurring cost than just the internet connection.

3.2 System Design

3.2.1 Architecture Design

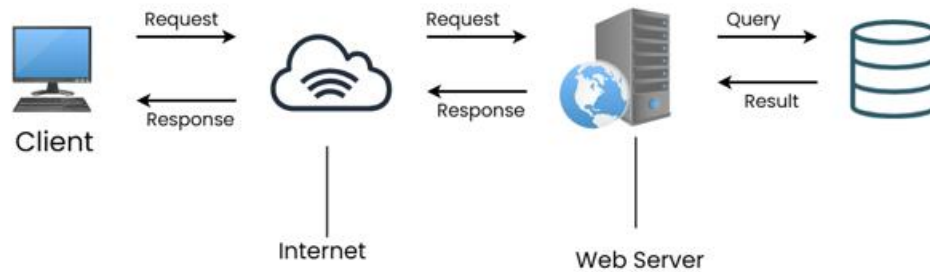


Figure 2.2 Architecture Design of College Chatbot

3.2.2 Database Schema Design

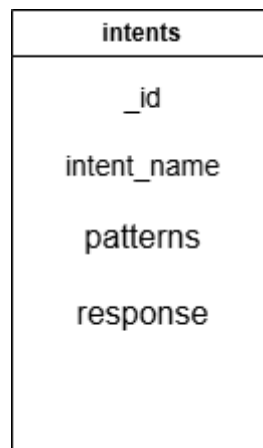
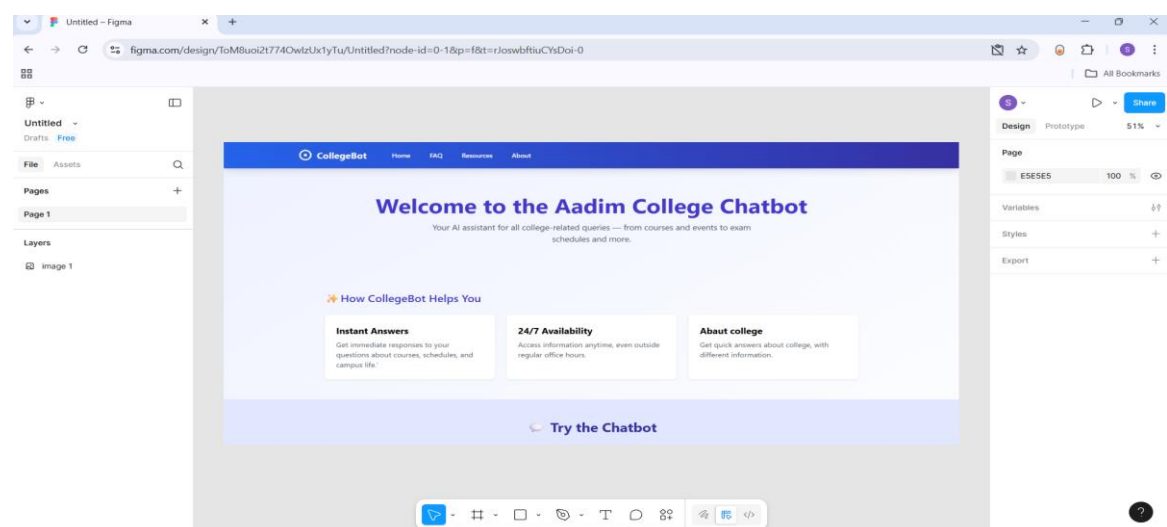
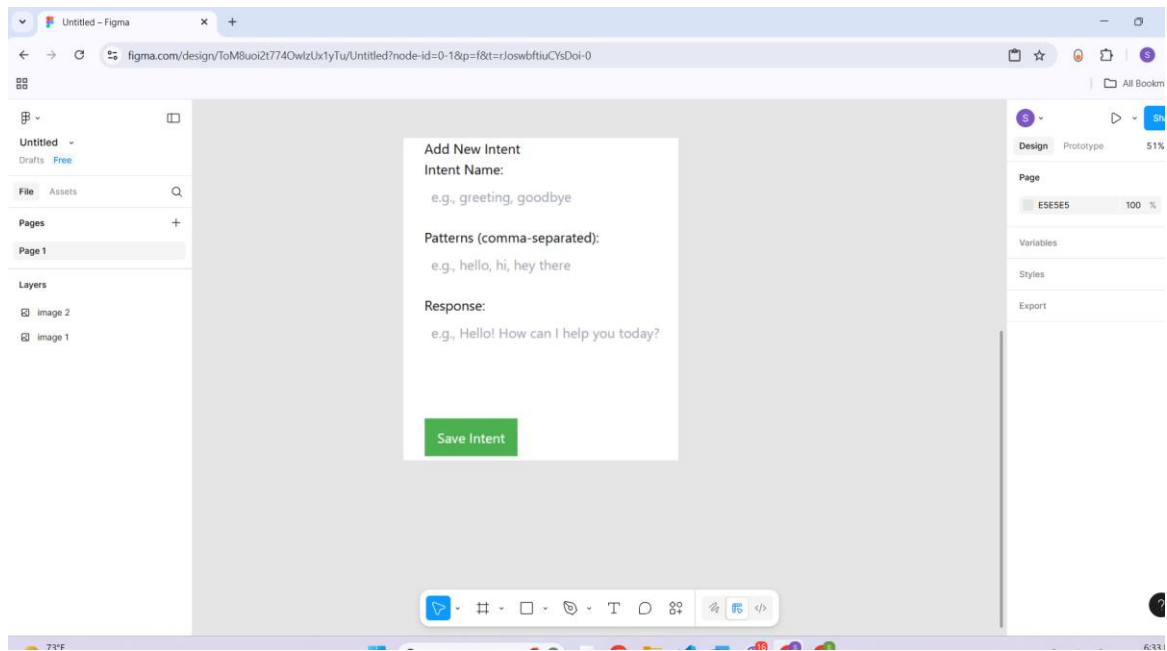


Figure 3.3 Database Schema Diagram of College Chatbot

3.2.3 Interface Design

Some of the screenshots of interface design of college chatbot System is shown below:





3.3 Algorithm Detail

BM25, also known as Okapi BM25, is a probabilistic information retrieval model used to rank documents based on their relevance to a given search query. It's a widely used ranking algorithm in search engines and other platforms where retrieving relevant information is crucial.

Key components of chatbot architecture are bellow:

- i. User interface
- ii. Natural Language Understanding (NLU)
- iii. Dialogue management
- iv. Backend system
- v. Response generation

Chapter 4: Implementation and Testing

4.1 Implementation

This chapter is implemented to document the development lifecycle of the key features of the system. It goes into detail on the problems and how they were resolved. The chapter will also go into the test plan and test report of the project to verify the proper functioning of the functionalities. In this chapter, the implementation phase of the project will be presented. But before that, we will give an analysis of the software and hardware components guiding the implementation of this project.

4.1.1 Tools Used

React Js

React is an open-source JavaScript library for front-end development based on component architecture. React is maintained by Meta and an ecosystem of individual developers and companies. React is capable of developing single-page and server-rendered applications for mobile or web platforms using frameworks such as Next.js.

Node Js

Node.js is a cross-platform, open-source JavaScript runtime environment that can run on Windows, Linux, Unix, macOS, and more. The V8 JavaScript engine powers Node.js and executes JavaScript code outside of the web browser. Node.js allows the use of JavaScript to write command line tools and server-side scripting.

MongoDB

MongoDB is a cross-platform, document-database program. It is considered to be a NoSQL database. MongoDB utilizes JSON-like documents with optional schemas. MongoDB is developed by MongoDB Inc., and the current versions are licensed under the Server Side Public License.

Github

GitHub is a web-based platform for version control and collaborative software development. GitHub uses Git, a distributed version control system, to keep track of different changes in source code over time so that other developers can collaborate on projects.

4.2 Testing

4.2.1 Test Cases for System Testing

Table 4.1 College Chatbot System Test Case

Field	Description
Test Case ID	TC-01
Test Case Title	Chatbot Responds to the Admission Query
Precondition	User should be on the chatbot interface (web or app).
Steps for Testing	1. Open the interface for the chatbot. 2. Enter the query: "What is the admission process?" 3. Press Enter or Send key.
Test Data	User query: "What is the admission process?"
Expected Result	Chatbot should provide the relevant information about the college admission process.
Actual Result	Chatbot responded with a detailed stepwise admission procedure.

Chapter 5: Conclusion and Future Recommendations

5.1 Conclusion

The college chatbot is a helpful tool that gives quick answers to common questions students have, like admission details, course info, exam schedules, and more. It saves time by giving instant replies, anytime and anywhere, without needing to visit the office. It makes communication easier for students, teachers, and staff. Overall, the chatbot helps make college life smoother and more convenient for everyone.

5.2 Lesson Learnt/Outcome

While developing this college chatbot system, learned a lot about creating modular system, which made the code easier to manage and test. Also realized how implement it is organize data efficiently, especially when using express to handle database. Connecting the front-end (React) and backend (node/express) using REST APIs helped to understand how to sync data smoothly between them.

5.3 Future Recommendation

In the future, the college chatbot system can be improved by adding student-performance demonstration through simple reports with regard to time. Adding NLP-based access would be an excellent idea so that students, teachers, and admins can exercise the access rights they need. It could also track attendance and behavior to provide a view of student progress. Moving the chatbot to the cloud would add performance and ease of management, especially in situations with many users using the chatbot simultaneously. A mobile app would be another step in convenience toward on-the-go use.

References

- [1] WorldLink Communications Ltd. "WorldLink Chatbot." worldlink.com.np.
https://worldlink.com.np/worldlink-chatbot/?utm_source=chatgpt.com (accessed on May 22, 2025)