

# **Edu Guard: Unlocking Your Path to the Perfect College**

## **PROJECT REPORT**

**21AD1513- INNOVATION PRACTICES LAB**

*Submitted by*

**SANTHOSH KUMAR V S     - 211422243285**

**PAVITHRAN M                 - 211422243232**

**NIRANJAN K                 - 211422243219**

*in partial fulfillment of the requirements for the award of degree of*

**BACHELOR OF TECHNOLOGY**

in

**ARTIFICIAL INTELLIGENCE AND DATA SCIENCE**



**PANIMALAR ENGINEERING COLLEGE, CHENNAI-600123**

**ANNA UNIVERSITY: CHENNAI-600 025**

October, 2024

## **BONAFIDE CERTIFICATE**

Certified that this project report titled “**Edu Guard: Unlocking Your Path to the Perfect College**” is the bonafide work of **SANTHOSH KUMAR V S, PAVITHRAN M, NIRANJAN K** Register No. **211422243285, 211422243232, 211422243219** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

### **INTERNAL GUIDE**

**Dr.S.Hemamalini M.E.,Ph.D**  
**Associate Professor,**  
**Department of AI &DS**

### **HEAD OF THE DEPARTMENT**

**Dr.S.MALATHI M.E., Ph.D**  
**Professor and Head,**  
**Department of AI &DS**

Certified that the candidate was examined in the Viva-Voce Examination held on .....

**INTERNAL EXAMINER**

**EXTERNAL EXAMINER**

## ABSTRACT

Selecting the best college can be a challenging and overwhelming process, particularly for students who are unable to visit all the institutions to gather necessary information. Edu Guard addresses this issue by offering a one-stop platform that provides detailed and comprehensive information on key factors such as college fees, placement opportunities, hostel amenities, and more. With everything available in a single location, Edu Guard makes it easier for students to access the data they need without extensive research. The platform, powered by a chatbot, enhances decision-making during the crucial counseling process by offering real-time updates, comparison tools, and personalized recommendations tailored to individual student profiles. Edu Guard's natural language processing (NLP) capabilities ensure seamless communication, allowing students to ask questions and receive accurate, up-to-date information effortlessly. The chatbot makes it easy for users to navigate through various options, helping them make well-informed decisions. In the future, Edu Guard is set to introduce AI-driven recommendations, further refining the personalization of college suggestions. Additionally, real-time conversations with alumni will provide students with firsthand insights into the experiences at different institutions, giving them an even stronger foundation to make knowledgeable choices about their academic paths.

***Keywords:*** Artificial Intelligence, Machine Learning, Natural Language Processing , Wordnet and Chatbot

## ACKNOWLEDGEMENT

I also take this opportunity to thank all the Faculty and Non-Teaching Staff Members of Department of Artificial Intelligence and Data Science for their constant support. Finally I thank each and every one who helped me to complete this project. At the outset we would like to express our gratitude to our beloved respected Chairman, **Dr.Jeppiaar M.A.,Ph.D**, Our beloved correspondent and Secretary **Mr.P.Chinnadurai M.A., M.Phil., Ph.D.**, and our esteemed director for their support.

We would like to express thanks to our Principal, **Dr. K. Mani M.E., Ph.D.**, for having extended his guidance and cooperation.

We would also like to thank our Head of the Department, **Dr.S.Malathi M,E.,Ph.D.**, of Artificial Intelligence and Data Science for her encouragement.

Personally we thank << **Dr.S.Hemamalini M.E, Ph.D**>>, Department of Artificial Intelligence and Data Science for the persistent motivation and support for this project, who at all times was the mentor of germination of the project from a small idea.

We express our thanks to the project coordinators **DR.S.RENUGA M.E., Ph.D.**, Associate Professor & **Ms.K.CHARULATHA M.E.**, Assistant Professor in Department of Artificial Intelligence and Data Science for their Valuable suggestions from time to time at every stage of our project.

Finally, we would like to take this opportunity to thank our family members, friends, and well-wishers who have helped us for the successful completion of our project. We also take the opportunity to thank all faculty and non-teaching staff members in our department for their timely guidance in completing our project.

**SANTHOSH KUMAR V S**  
**(211422243285)**

**PAVITHRAN M**  
**(211422243232)**

**NIRANJAN K**  
**(211422243219)**

## TABLE OF CONTENTS

<b>CHATER NO</b>	<b>TITLE</b>	<b>PAGE NO</b>
	<b>ABSTRACT</b>	<b>iii</b>
	<b>LIST OF FIGURES</b>	<b>vii</b>
	<b>LIST OF TABLES</b>	<b>viii</b>
	<b>LIST OF ABBREVIATIONS</b>	<b>ix</b>
<b>1</b>	<b>INTRODUCTION</b>  1.1 Complex AI Systems and NLP Advancements 1.2 AI-Powered support 1.2.1 AI in College Administration 1.2.2 User friendly web application 1.3 AI College Advisor 1.3.1 AI-Powered Response Systems 1.3.2 User Enrollment Access 1.3.3 Interactive Support Sections 1.3.4 Efficient User Interface 1.3.5 Role of chatbots 1.3.6 Mimicking human interactions  1.4 Architecture Diagram 1.4.1 Application	<b>x</b>  x x xi xi xi xi xii xii xii xii xii xii xii xiii  xiii xiv
<b>2</b>	<b>LITERATURE REVIEW</b> 2.1 Chatbot Defintion 2.2 Chatbot for College management System using AI 2.3 Chatbot for Education System 2.4 College Information Chatbot System 2.5 Developing a Chatbot using Machine Learning 2.6 Chatbot using a Knowledge in Database: Human to Machine Learning	<b>xv</b> xv  xvi  xvii xvii  xviii  xix

<b>3</b>	<b>SYSTEM DESIGN</b> 3.1 System Architecture 3.2 Data Flow Diagram 3.2.1 DFD-1 3.2.2 DFD-2 3.3 Sequence Diagram	<b>xx</b> xx xxi xxi xxiii  xxv
<b>4</b>	<b>MODULES</b> 4.1 User Interface (UI) Module 4.2 Chatbot Engine Module 4.3 Natural Language Processing (NLP) Module 4.4 Database Module 4.5 User Profile Management Module	<b>xxviii</b> xxviii  xxix  xxix   xxx xxxi
<b>5</b>	<b>SYSTEM REQUIREMENT</b> 5.1 Introduction 5.2 Requirement 5.2.1 Hardware requirement 5.2.2 Software requirement	<b>xxxii</b> xxxii  xxxii xxxii xxxiii
<b>6</b>	<b>CONCLUSION &amp; REMARK</b> 6.1 conclusion	xxxvii
	<b>REFERENCES</b> <b>APPENDIX</b>	<b>xxxviii</b>  xxxviii

## LIST OF FIGURES

<b>FIGURE NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
<b>1.4</b>	Architecture Diagram	<b>11</b>
<b>3.1.1</b>	System Architecture	<b>18</b>
<b>3.1.2</b>	Data Flow Diagram	<b>19</b>

## **LIST OF TABLES**

<b>TABLE NO.</b>	<b>TITLE NAME</b>	<b>PAGE NO.</b>
1.	LITRATURE REVIEW	15



## LIST OF ABBREVIATIONS

ABBREVIATIONS	MEANING
ML –	Machine Learning
SHA-1 –	Secure Hash Algorithm 1
SHA-256 –	Secure Hash Algorithm 256-bit
UI –	User Interface
NLP –	Natural Language Processing
AI –	Artificial Intelligence
QA –	Question Answering
Bot –	Chatbot
WN –	WordNet
TF-IDF –	Term Frequency-Inverse Document Frequency
SQL –	Structured Query Language
API –	Application Programming Interface

# **CHAPTER 1**

## **INTRODUCTION**

### ***1.1 "COMPLEX AI SYSTEMS AND NLP ADVANCEMENTS"***

Artificial Intelligence (AI) systems are becoming more intricate and challenging to operate, driven by rapid advancements in information technology and interconnected networks. With the rise of natural language processing (NLP), robotics, and decision-support networks, AI is increasingly intertwined with human-like activities, enabling machines to interact more intelligently and intuitively. The integration of hybrid approaches and adaptive techniques has revolutionized AI, allowing for more refined and advanced operations across diverse sectors. In today's world, the development of cutting-edge NLP systems has empowered AI to not only comprehend but also process human language with remarkable accuracy. By continuously learning from vast digital resources available online, AI systems are able to extract valuable insights and enhance their functionality, paving the way for more innovative applications in both industry and daily life.

### ***1.2 AI-POWERED SUPPORT***

A chatbot, also known as a chatterbox or Artificial Conversational Entity, is an AI-driven program designed to simulate human conversations. Utilizing advanced techniques like speech analysis, image processing, and natural language processing (NLP), chatbots can interact with users seamlessly. In the

college administration system, AI algorithms power a chatbot that helps answer user inquiries. This web-based application allows users to select a question category, submit their query, and receive accurate responses from the bot. By incorporating AI, the chatbot efficiently provides relevant answers, streamlining the user experience.

### ***1.2.1 AI IN COLLEGE ADMINISTRATION***

Artificial intelligence algorithms have been utilized to construct a chatbot specifically for the college administration system, which analyzes user inquiries efficiently.

### ***1.2.2 USER FRIENDLY WEB APPLICATION***

This chatbot system functions as a web application that addresses user inquiries in an organized manner. Users simply select a category for their questions and pose their queries to the bot, which records them for processing.

## ***1.3 AI COLLEGE ADVISOR***

An AI algorithm system provides relevant answers, allowing users to make requests without visiting the college or its website. After enrolling and logging in, users can access help pages and interact with support sections about college activities. The efficient Graphical User Interface (GUI) ensures quick responses regarding academics, admissions, and social events.

The chatbot communicates in natural language, taking on some human tasks by answering user questions. Utilizing Artificial Intelligence Markup Language (AIML), it mimics human interaction for a conversational experience.

While effective, chatbots have limited real-time learning abilities and may struggle to retain conversation details.

### ***1.3.1 AI-POWERED RESPONSE SYSTEMS***

A system of artificial intelligence algorithms provides relevant answers to user inquiries. Users can make requests without needing to visit the college or its website.

### ***1.3.2 USER ENROLLMENT ACCESS***

Both system enrollment and login are required for users. Once logged in, users gain access to various help pages tailored to their needs.

### ***1.3.3 INTERACTIVE SUPPORT SECTIONS***

Users can interact and ask questions about college activities through multiple support sections, ensuring they have the information they need.

### ***1.3.4 EFFICIENT USER INTERFACE***

Thanks to an efficient Graphical User Interface (GUI), the system quickly responds to user requests. This web application allows users to inquire about college-related activities, including academics, admissions, intake, and social events.

### ***1.3.5 ROLE OF CHATBOTS***

An AI program, known as a chatbot, can communicate with humans in natural language, effectively taking on some human responsibilities for answering questions. Chatbots, designed to assist users, utilize Artificial Intelligence Markup Language (AIML) to facilitate communication.

### 1.3.6 MIMICKING HUMAN INTERACTION

One of the primary goals of chatbots is to simulate human-like conversation, creating the illusion of an engaging interaction. While they are designed to be convincing, their ability to learn in real-time is limited, and they may not retain all details of the conversation.

## 1.4 ARCHITECTURE DIAGRAM

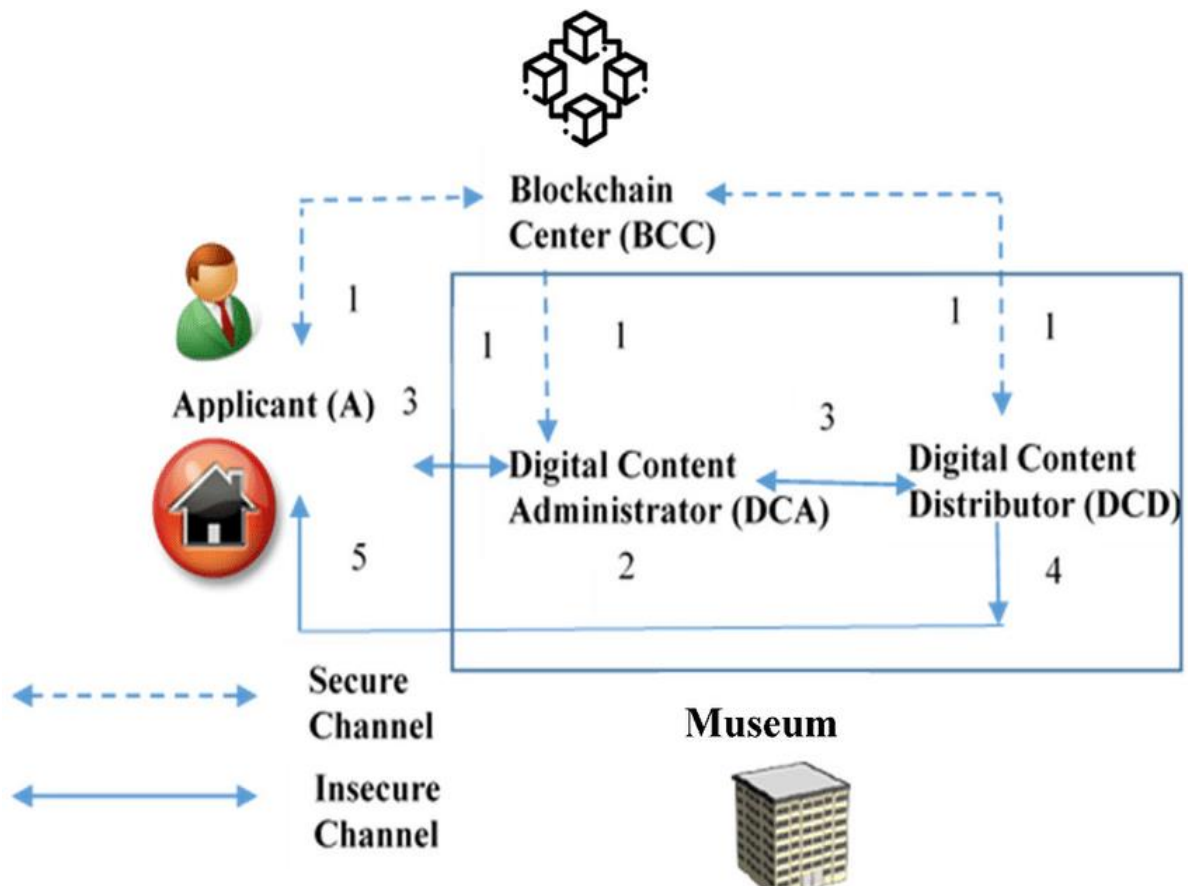


Fig 1.4: Architecture diagram of social media

It illustrates a blockchain-based digital content distribution system designed for secure access to museum resources. An applicant (A) initiates a request to access digital content, which is processed through the Blockchain Center (BCC) for verification and tracking. The Digital Content Administrator (DCA) manages this request, ensuring that permissions are correctly assigned and content remains secure. Verified requests are passed to the Digital Content Distributor (DCD), which handles the secure distribution of the digital resources. The museum's content is then made accessible to the applicant through a Secure Channel, while an Insecure Channel may be used for non-sensitive interactions. This architecture ensures secure and controlled access to digital museum content, leveraging blockchain for transparency and integrity in content management.

## **APPLICATION**

It represents a system that uses blockchain for the secure distribution and management of digital content, such as in a museum context. Here are the applications or roles represented in this diagram:

- (i) Applicant (A)
- (ii) Blockchain Center (BCC)
- (iii) Digital Content Administrator (DCA)
- (iv) Digital Content Distributor (DCD)
- (v) Museum

## CHAPTER 2

### LITERATURE REVIEW

A [scholarly](#), which includes the current knowledge including substantive findings, as well as theoretical and methodological contributions to a particular topic. Literature reviews are [secondary sources](#), and do not report new or original experimental work. Most often associated with academic-oriented literature, such reviews are found in [academic journals](#), and are not to be confused with [book reviews](#) that may also appear in the same publication. Literature reviews are a basis for research in nearly every academic field. A narrow-scope literature review may be included as part of a [peer-reviewed](#) journal article presenting new research, serving to situate the current study within the body of the relevant literature and to provide context for the reader. In such a case, the review usually precedes the methodology and results sections of the work.

#### ***2.1 Chatbot Definition***

This article defines chatbots, also known as chatterbots or Artificial Conversational Entities, and explains how they use AI to simulate human-like conversations. The piece explores the evolution of chatbots, highlighting key techniques such as speech analysis, image processing, and Natural Language Processing (NLP). It focuses on the role of AI in enabling chatbots to handle auditory and textual communication.

The article examines the workings of chatbots, describing how they leverage AI technologies to understand user inputs and generate appropriate responses. It also breaks down the chatbot development process, from creating decision trees and rule-based systems to implementing machine learning algorithms for improved interaction. The methodology includes examples of chatbot deployment in customer support, user assistance, and entertainment.

*AUTHOR* : Michael Garland

*YEAR* : 18 May 2016

## ***2.2 Chat-Bot For College Management System Using A.I***

This paper presents the design and development of a chatbot system for college management using Artificial Intelligence (AI). The chatbot is designed to assist students with routine queries related to academics, admissions, and other administrative activities. The goal is to create an interactive system that can effectively mimic human communication and provide accurate information without requiring manual intervention.

The chatbot is developed using a combination of AI techniques, including Natural Language Processing (NLP) for understanding user inputs, and Artificial Intelligence Markup Language (AIML) to structure responses. The system architecture includes a backend database for storing query responses, an NLP engine for interpreting user questions, and a user interface for interaction.

*AUTHOR* : Prof. K. Bala, Mukesh Kumar, Sayali Hulawale, Sahil Pandita

*YEAR* : November 2017



## ***2.3 Chatbot for Education System***

This paper focuses on the development of a chatbot for the education system, designed to assist students by providing relevant academic and administrative information. The chatbot leverages AI technologies to automate responses and help users navigate the complexities of an educational institution's systems. The primary aim is to reduce human involvement in routine tasks and enhance user experience through quick and accurate interactions.

The chatbot is built using Natural Language Processing (NLP) and machine learning algorithms to understand user queries. The system includes a training dataset for the chatbot to learn from, enabling it to improve its responses over time. The chatbot interacts with users via a web interface, where they can ask questions related to course details, exam schedules, admissions, and other educational activities. The chatbot's performance is evaluated based on its ability to provide accurate information and its response time.

*AUTHOR : Guruswami Hiremath, Aishwarya Hajare, Priyanka Bhosale, Rasika Nanaware, Dr. K. S. Wagh*

*YEAR : March 2018*

## ***2.4 College Information Chat Bot System***

This paper proposes a chatbot system aimed at improving the dissemination of college-related information. The system provides instant answers to queries regarding college admissions, fees, exam schedules, and other

academic activities. The chatbot aims to streamline the information retrieval process for students, making it easier and faster to get responses.

The chatbot uses an AI-driven approach, with the implementation focusing on NLP techniques to process user queries. The system is trained on a variety of typical questions asked by students, and responses are stored in a backend database. The chatbot's architecture includes a user-friendly interface, a query processor, and a response generator. Testing involved user interactions to assess the chatbot's ability to provide accurate and relevant information in real-time.

*AUTHOR : Amey Tiwari, Rahul Talekar, Prof. S. M. Patil*

*YEAR : March-April 2017*

## ***2.5 Developing a Chatbot using Machine Learning***

This paper explores the development of a chatbot using machine learning techniques, focusing on how AI can improve the way chatbots understand and respond to user input. The chatbot is designed to provide automated assistance in various domains, with a special focus on education and customer service. The goal is to create a more intuitive and efficient communication tool.

The chatbot employs machine learning algorithms, particularly supervised learning, to train the system to recognize and respond to a variety of queries. A dataset of typical user interactions is used to train the model, and the chatbot uses NLP to process language inputs. The system includes a learning loop, enabling it to refine its answers based on user feedback. The chatbot's

performance is evaluated through real-world testing, focusing on accuracy, response time, and user satisfaction.

*AUTHOR: K. Jwala, G. N. V. G. Sirisha, G. V. Padma Raju*

*YEAR : June 2019*

## **2.6 Chatbot Using a Knowledge in Database: Human-to-Machine Conversation Modeling**

This paper proposes a new chatbot model that uses a knowledge base to generate more natural and engaging conversations with users. The proposed model is based on the idea that chatbots should be able to access and understand the information in a knowledge base in order to provide more relevant and informative responses.

The proposed model was implemented in a prototype system that was able to generate conversations with users about a variety of topics. The system was able to generate conversations that were both accurate and informative. The proposed model is a promising approach for developing chatbots that can engage in more natural and meaningful conversations with users.

*AUTHOR:: Setiaji Bayu, Wibowo Ferry*

*YEAR : January 2016*

## CHAPTER 3

### SYSTEM DESIGN

#### 3.1 SYSTEM ARCHITECTURE

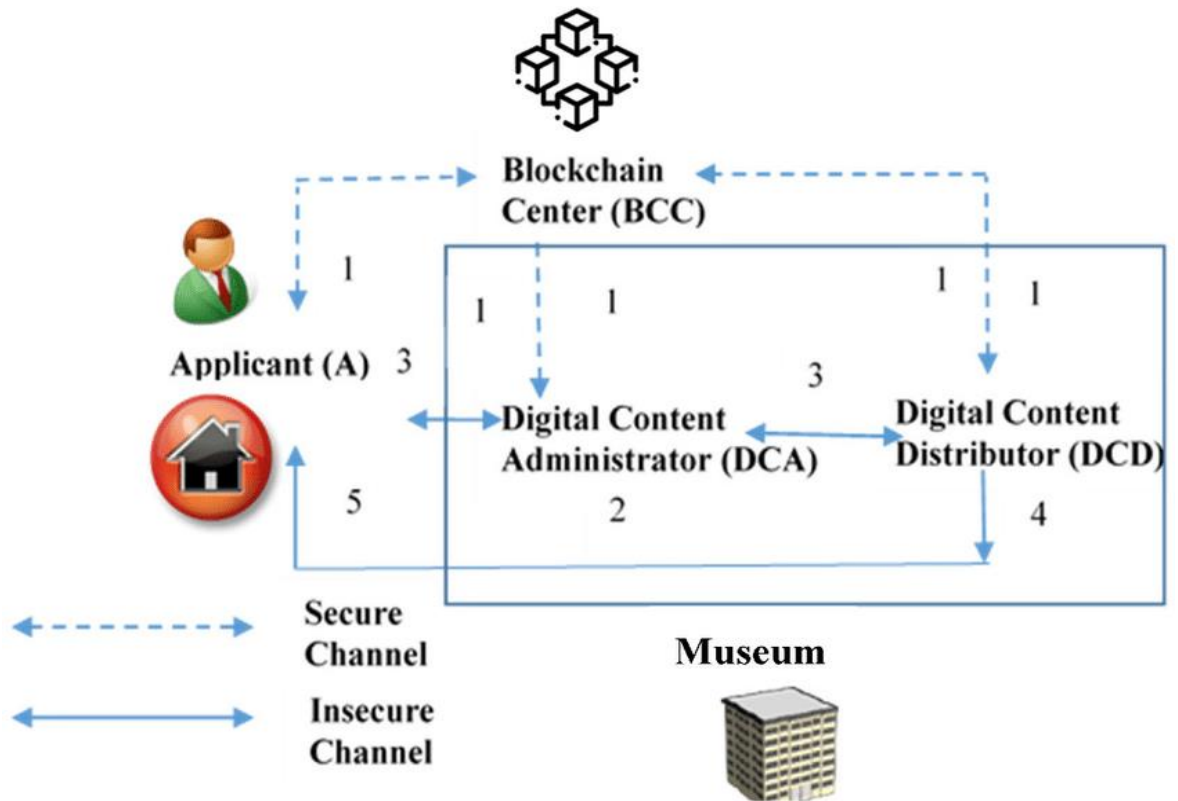


fig 3.1 : system architecture

The diagram illustrates a blockchain-based digital content distribution system designed for secure access to museum resources. An applicant (A) initiates a request to access digital content, which is processed through the

**Blockchain Center (BCC)** for verification and tracking. The **Digital Content Administrator (DCA)** manages this request, ensuring that permissions are correctly assigned and content remains secure. Verified requests are passed to the **Digital Content Distributor (DCD)**, which handles the secure distribution of the digital resources. The museum's content is then made accessible to the applicant through a **Secure Channel**, while an **Insecure Channel** may be used for non-sensitive interactions. This architecture ensures secure and controlled access to digital museum content, leveraging blockchain for transparency and integrity in content management.

### **3.2 DATA FLOW DIAGRAM**

. A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A “DFD” is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated.

#### ***3.2.1 DFD-1***

The College Chatbot System Flowchart outlines a systematic process for assisting users with college-related queries. The interaction begins with the chatbot greeting the user, followed by a prompt for the user's email. Upon receiving the email, the chatbot displays various college categories to help the user navigate relevant topics. The system then analyzes the user's actions to determine their intent, ensuring that responses are tailored to the specific query.

If the initial response provided by the chatbot is deemed useful by the user, the interaction concludes. However, if the response is not satisfactory, the chatbot restates the question and proceeds to process the user's query in more detail. It identifies key terms to refine the response, aiming to enhance the accuracy and relevance of the information provided. This flowchart highlights the chatbot's approach to creating a responsive and helpful experience for users seeking college information.

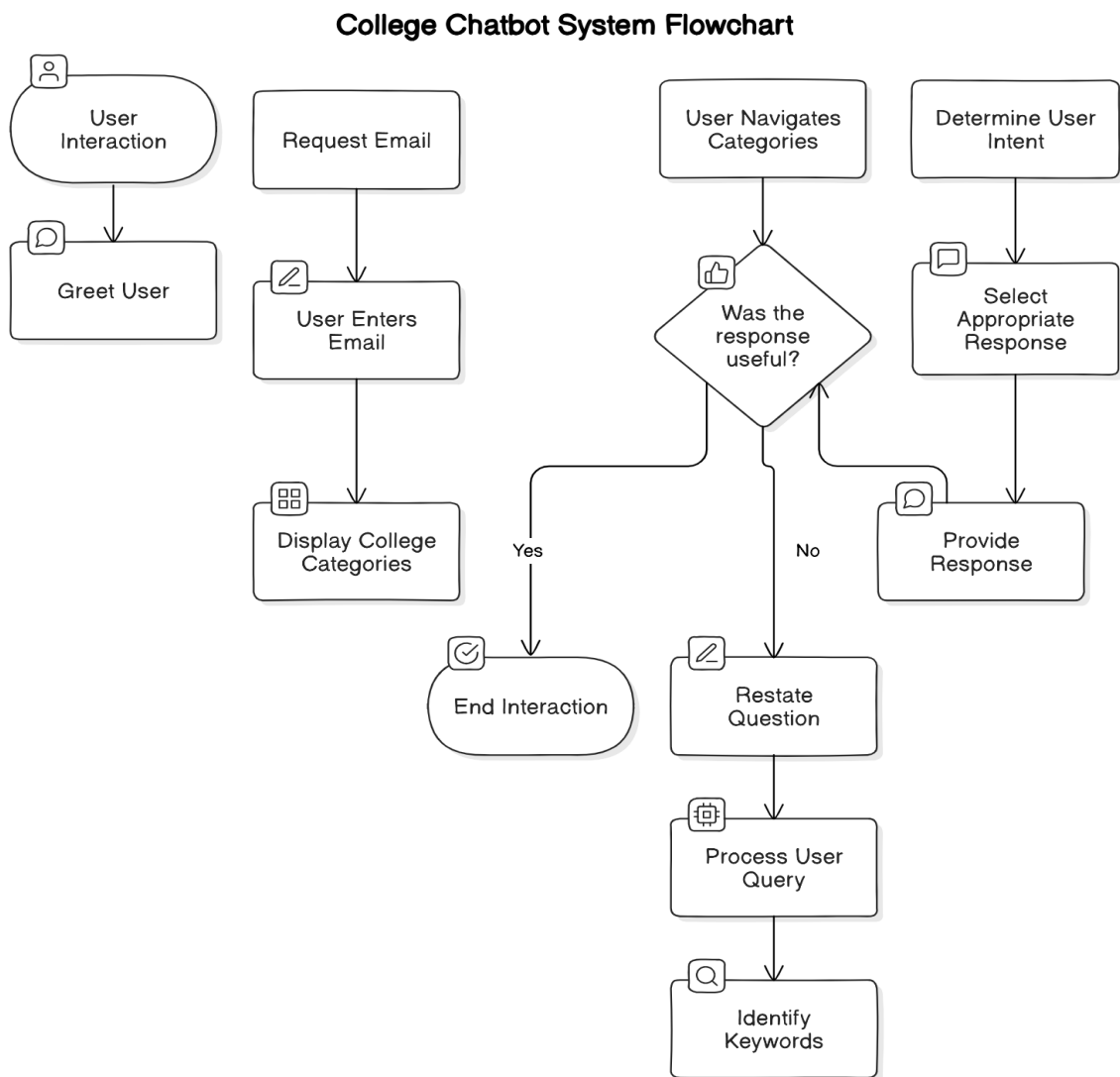


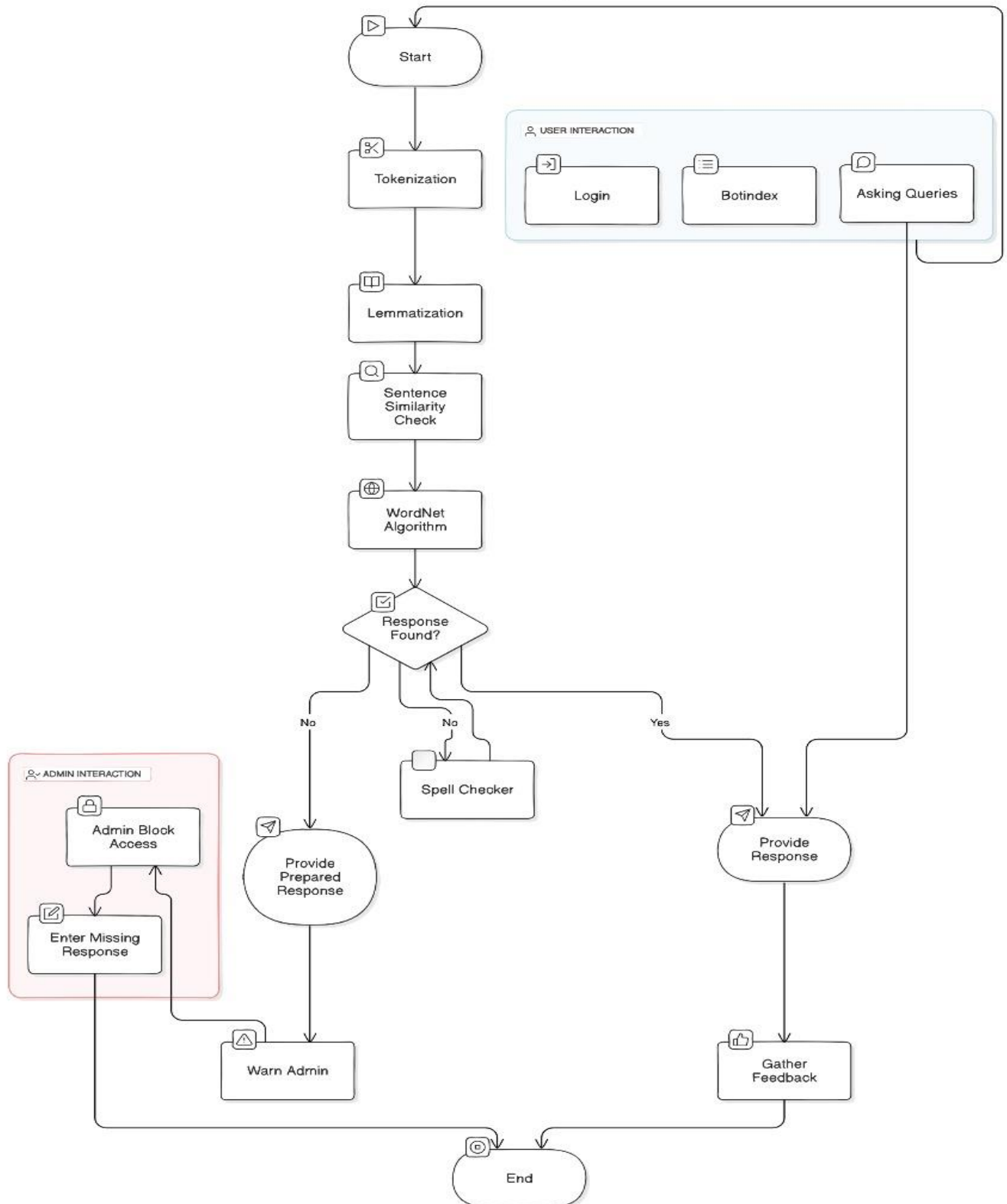
Fig 3.2.1: DFD-1

### **3.2.2 DFD-2**

The diagram outlines the **Chatbot Interaction and Response Process** designed to handle user queries efficiently. It begins with the **Tokenization** and **Lemmatization** of user input, breaking down text into understandable parts and reducing words to their base forms. The chatbot then checks sentence structure and uses a **WordNet Algorithm** to find relevant responses. If a matching response is identified, it moves directly to providing the response and gathering feedback from the user.

If no suitable response is found, the chatbot performs a **Spell Check** and retries; if still unsuccessful, the system triggers an **Admin Interaction**. In this case, an admin is alerted to either address the issue directly or add a new response to the database for future use. This process allows the chatbot to continually improve while ensuring users receive accurate information, with admin support as a backup for more complex inquiries.

## Chatbot Interaction and Response Process





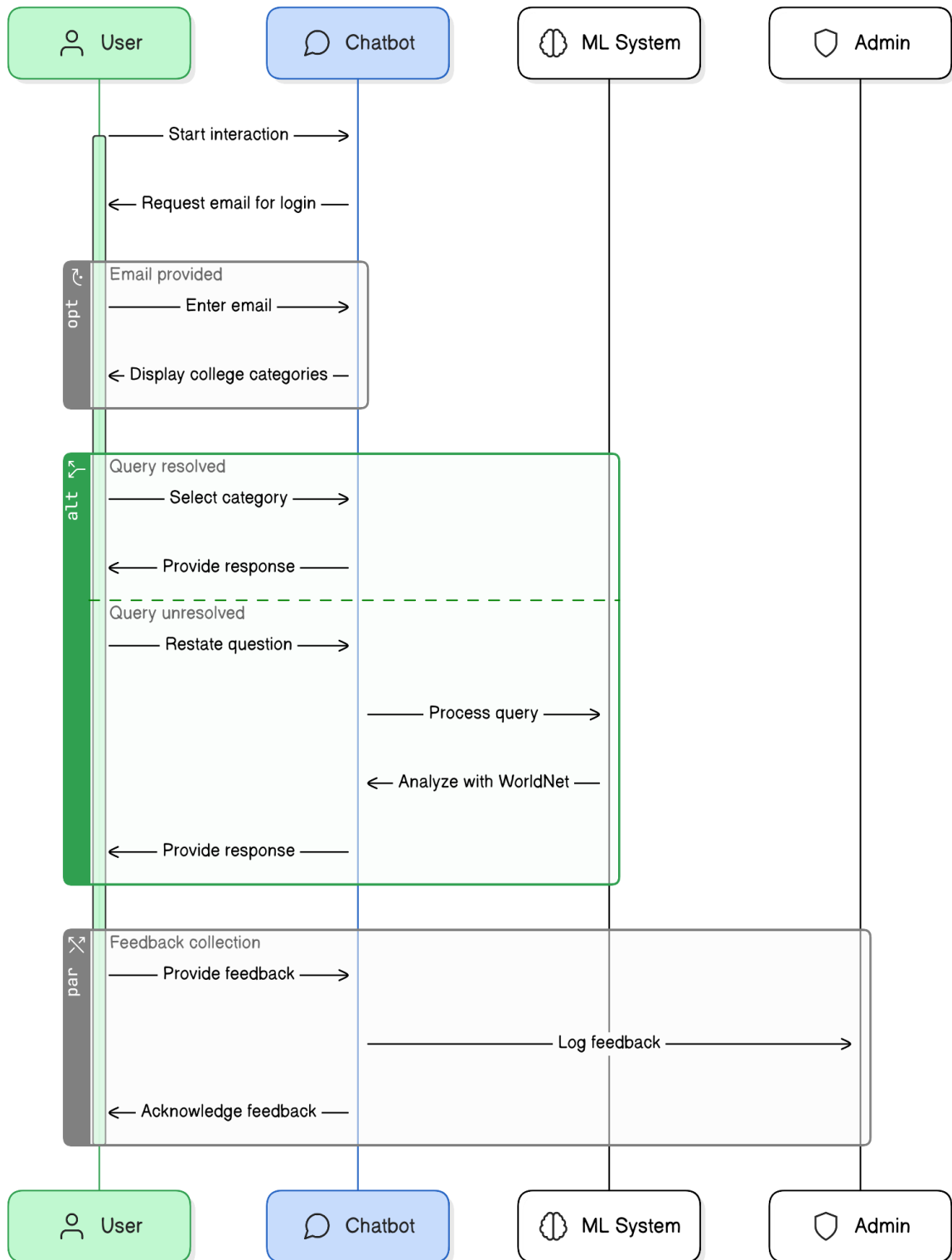
### 3.3 SEQUENCE DIAGRAM

The "College Chatbot System Interaction" sequence diagram illustrates the flow of interactions between a User, Chatbot, ML System, and Admin. The process begins with the User initiating an interaction, after which the Chatbot requests the User's email for login. If the User provides their email, the Chatbot displays a list of available college categories.

The main flow then branches into two possibilities for handling the User's query: **\*\*resolved\*\*** or **\*\*unresolved\*\***. If the query is resolved, the User selects a category, and the Chatbot responds directly with relevant information. If the query remains unresolved, the Chatbot restates the question for clarity and sends it to the ML System for further processing. The ML System analyzes the query with WorldNet, then returns the processed response to the Chatbot, which relays it to the User.

The diagram also includes a feedback collection process, where the User provides feedback that the Chatbot acknowledges. Simultaneously, the ML System logs this feedback, allowing the system to improve responses in future interactions. This sequence diagram effectively maps out the user journey through query handling and feedback within the college chatbot system.

## College Chatbot System Interaction



## **CHAPTER 4**

### **PROJECT MODULES**

#### **5 MODULES**

The project consists of Four modules. They are as follows,

1. User Interface (UI) Module
2. Chatbot Engine Module
3. Natural Language Processing (NLP) Module
4. Database Module
5. User Profile Management Module

##### ***4.1 USER INTERFACE(UI) MODULE***

The User Interface (UI) module of CustomBot is designed to provide a seamless and engaging user experience, featuring an intuitive layout that simplifies navigation and interaction. Users can effortlessly input topics or keywords through a streamlined form, allowing for quick content generation tailored to their needs. The UI offers real-time feedback, enabling instant viewing of generated articles, which can be easily modified for optimal results. Additionally, it incorporates visually appealing analytics dashboards that present key performance metrics, such as user engagement and content accuracy, empowering users with valuable insights into their journalism efforts. With its focus on accessibility and efficiency, the UI module

transforms the process of automated news reporting into a dynamic and user-friendly experience.

## ***4.2 CHATBOT ENGINE MODULE***

The Chatbot Engine module is the core of CustomBot, driving its ability to generate coherent and contextually relevant news articles. Powered by advanced natural language processing algorithms, this module utilizes tokenization and custom tagging to analyze input data effectively, ensuring a deep understanding of language nuances. The engine leverages machine learning techniques to adapt and improve over time, producing high-quality content that meets journalistic standards. With its capability to synthesize information from diverse sources, the Chatbot Engine not only accelerates content creation but also enhances the accuracy and reliability of generated articles. Its robust architecture allows for real-time processing, enabling instant responses to user queries and a dynamic storytelling experience, making it an indispensable tool for the future of automated journalism.

## ***4.3 NATURAL LANGUAGE (NLP) PROCESSING***

The Natural Language Processing (NLP) module is a vital component of CustomBot, enabling it to understand, interpret, and generate human language with remarkable precision. Utilizing cutting-edge algorithms, this module performs tasks such as tokenization, named entity recognition, and sentiment analysis, allowing for a nuanced comprehension of context and meaning. By transforming raw text into structured data, the NLP module ensures

that CustomBot can generate coherent and relevant news articles that resonate with readers. Its ability to process large volumes of information quickly and accurately enhances the system's responsiveness and adaptability, making it an essential tool for automated journalism. With continuous learning capabilities, the NLP module evolves over time, refining its language understanding and ensuring that CustomBot remains at the forefront of innovation in content generation.

#### ***4.4 DATABASE MODULE***

The Database module serves as the backbone of CustomBot, efficiently managing and storing vast amounts of data critical for content generation and analysis. This module is designed for optimal performance, ensuring quick retrieval and updating of information, which is essential for maintaining the system's responsiveness. It supports various data types, including news articles, user interactions, and metadata, facilitating comprehensive analytics and insights. By leveraging advanced indexing and querying techniques, the Database module enables seamless access to historical data and trends, empowering CustomBot to generate timely and contextually relevant news content. Its robust security features also ensure data integrity and privacy, making it a reliable foundation for automated journalism. Ultimately, the Database module enhances the overall functionality and reliability of CustomBot, positioning it as a leader in the evolving landscape of digital news.

## ***4.5 USER PROFILE MANAGEMENTMODULES***

The User Profile Management module enables users to create, update, and manage their personal profiles within the system. Users can easily register by providing essential information such as name, email, and password. They also have the flexibility to edit their profiles, adjust privacy settings, and control who can view their information.

Additionally, the module offers options for deactivation or permanent deletion of profiles, complete with clear instructions to prevent accidental actions. An intuitive user interface ensures easy navigation, while robust security measures protect user data through encryption and secure password storage. This module enhances user experience by promoting autonomy and maintaining privacy.

## **CHAPTER 5**

### **SYSTEM REQUIREMENTS**

#### **5.1 INTRODUCTION**

This chapter involves the technology used, the hardware requirements and the software requirements for the project .

#### **5.2 REQUIREMENTS**

##### **5.2.1 Hardware Requirements**

##### **5.1.2 RAM**

- Minimum: 8 GB
- Recommended: 16 GB
- Type: DDR4
- Justification: For a web-based platform like Edu Guard that involves handling moderate user traffic, basic NLP tasks, and database operations, 8 GB of RAM is sufficient. Upgrading to 16 GB provides smoother multitasking and improved performance, particularly during peak usage times.

##### **5.1.3 Storage**

- Type: Solid State Drive (SSD)
- Capacity:
  - Minimum: 256 GB

- Recommended: 512 GB
- **Justification:** A 256 GB SSD is enough for storing website content, databases, and application code. Opting for 512 GB provides extra space for data growth and media files, ensuring there's enough headroom for backups and scalability.

## 5.2.2 Software Requirements

### 5.2.1 Operating System

- **Choices:**
  - **Server:** Ubuntu 20.04 LTS or newer, Windows Server 2019/2022.
  - **Development:** Windows 10/11, macOS, or Ubuntu.
- **Justification:** Compatibility with a wide range of development tools, libraries, and server environments ensures flexibility and ease of deployment.

### 5.2.2 Web Server

- **Options:** NGINX or Apache
- **Justification:** Reliable and high-performance web servers that can handle HTTP requests efficiently and support load balancing.

### 5.2.3 Programming Languages and Frameworks

- **Backend:**
  - **Language:** Python (v3.8 or newer) or Node.js (v14 or newer).



- **Frameworks:** Django or Flask (for Python), Express.js (for Node.js).
- **Frontend:**
  - **Technologies:** HTML5, CSS3, JavaScript.
  - **Frameworks:** React.js, Angular, or Vue.js.
- **Justification:** Modern frameworks offer a scalable and maintainable structure for both client and server-side development, enhancing performance and user experience.

#### 5.2.4 Database Management System (DBMS)

- **Relational Database:** PostgreSQL or MySQL.
- **NoSQL Database (Optional):** MongoDB for handling flexible data structures.
- **Justification:** Relational databases are suitable for structured data such as college profiles and student information, while NoSQL is useful for unstructured data and scalability.

#### 5.2.5 Development Tools

- **Integrated Development Environment (IDE):**
  - **Options:** Visual Studio Code, PyCharm, IntelliJ IDEA.
- **Version Control System:** Git with platforms like GitHub, GitLab, or Bitbucket.
- **Justification:** IDEs streamline the development process, while Git ensures effective version control and collaboration.

### 5.2.6 Chatbot Engine

- **Options:**
  - Rasa Open Source or Dialogflow for building interactive chatbot capabilities.
- **Justification:** Provides robust NLP features for user interactions, enhancing the counseling process and delivering real-time responses.

### 5.2.7 Natural Language Processing (NLP) Tools

- **Libraries:** SpaCy, NLTK, Hugging Face Transformers.
- **Justification:** Essential for processing and understanding user queries effectively.

### 5.2.8 Security Tools

- **Data Encryption:** SSL/TLS certificates for data transmission security.
- **User Authentication:** OAuth 2.0 or JSON Web Tokens (JWT) for secure user sessions.
- **Firewall:** Software-based firewalls and intrusion detection tools (e.g., Fail2Ban, UFW).
- **Justification:** Ensures secure communication and protects user data.

### 5.2.9 Monitoring and Logging

- **Monitoring Tools:** Prometheus, Grafana.
- **Log Management:** ELK Stack (Elasticsearch, Logstash, Kibana).
- **Justification:** Helps track system performance, identify issues, and monitor user activity effectively.

### 5.2.10 Containerization and Virtualization (Optional)

- **Containerization:** Docker for creating lightweight, reproducible environments.
- **Orchestration:** Kubernetes (for large-scale projects).
- **Justification:** Ensures consistent application deployment and facilitates scaling.

### 5.2.11 Backup Solutions

- **Tools:** Automated backup software such as Veeam or Bacula.
- **Justification:** Provides data recovery options and protects against data loss.

These software requirements ensure that Edu Guard runs smoothly, provides robust performance, and maintains high security and reliability.

For a comprehensive system to run the Edu Guard platform, the system requirements should cover both software and hardware to ensure optimal performance across the different modules

## **CHAPTER 6**

### **CONCLUDING REMARKS**

#### **6.1 CONCLUSION**

Our project involved creating a chatbot system tailored specifically for a college. This chatbot system can be customized to fit the needs of the education domain and improve user interaction on the college website by providing precise answers to user queries. Additionally, we looked into the design stages of the chatbot system and several methods to improve its accuracy. The administrator must expand the chatbot system's knowledge base and provide it with additional college-related information in order to enhance the meaning and accuracy of the responses the chatbot provides. However, getting input from the prospective user can be useful for creating the college chatbot system, which would then answer user inquiries.

## REFERENCES

- [1] S. Chatbot definition, Available at [https://medium.com/@mg/bot-is-ahilariously-over-simplified-buzzword-let-s-fix-thatfl63abb8ba7#:~:text=A%20chatbot%20\(also%20known%20as,via%20auditory%20or%20textual%20methods.](https://medium.com/@mg/bot-is-ahilariously-over-simplified-buzzword-let-s-fix-thatfl63abb8ba7#:~:text=A%20chatbot%20(also%20known%20as,via%20auditory%20or%20textual%20methods.)
- [2] Introduction to Artificial Intelligence Markup Language, Available at [https://www.tutorialspoint.com/aiml/aiml\\_introduction.htm](https://www.tutorialspoint.com/aiml/aiml_introduction.htm)
- [3] Prof.K.Bala, Mukesh Kumar, SayaliHulawale, SahilPandita,“Chat-Bot For College Management System Using A.I” International Research Journal of Engineering and Technology (IRJET) Volume: 04, Issue: 11, Page no: 2030-2033| Nov 2017.
- [4] Porter Stemmer Algorithm, Available at <http://snowball.tartarus.org/algorithms/porter/stemmer.html>
- [5] Guruswami Hiremath, AishwaryaHajare, PriyankaBhosale, RasikaNanaware, Dr. K. S. Wagh, “Chatbot for education system” International Journal of Advance Research, Ideas and Innovations in Technology (IJARIIT) ISSN: 2454-132X, Volume: 4, Issue: 3, Page no: 37-43|2018.
- [6] Amey Tiwari, Rahul Talekar, Prof.S.M.Patil, “College Information Chat Bot System” International Journal of Engineering Research and General Science (IJERGS) Volume: 5, Issue: 2, Page no: 131-137| March-April 2017.
- [7] K. Jwala, G.N.V.G Sirisha, G.V. Padma Raju, “Developing a Chatbot using Machine Learning” International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume: 8 Issue: 1S3, Page no: 89-92| June 2019.
- [8] Basics of Natural Language ToolKit, Availilable at <https://www.nltk.org/>
- [9] ] Naeun Lee, Kirak Kim, Taeseon Yoon, “Implementation of Robot Journalism by Programming Custombot using Tokenization and Custom Tagging” International Conference on Advanced Communications Technology (ICACT) Page no: 566-570| Feb 2017.
- [10] Fundamentals of Natural Language Processing - Tokenization, Lemmatization, Stemming and Sentence Segmentation, Available at [https://colab.research.google.com/github/dairai/notebooks/blob/master/notebooks/2020-03-19nlp\\_basics\\_tokenization\\_segmentation.ipynb#scrollTo=H7gQFbUxOQt b](https://colab.research.google.com/github/dairai/notebooks/blob/master/notebooks/2020-03-19nlp_basics_tokenization_segmentation.ipynb#scrollTo=H7gQFbUxOQt b)

