

# UNI-CONNECT

## MINI PROJECT REPORT

*Submitted to the APJ Abdul Kalam Technological University  
in partial fulfillment of requirements for the award of degree*

*Bachelor of Technology*

*in*

*Computer Science and Engineering*

*by*

**KAJAL RAMESH B (MEE22CS029)**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**MES COLLEGE OF ENGINEERING AND TECHNOLOGY**

**KUNNUKARA**

**KERALA**

**APRIL 2025**

**MES COLLEGE OF ENGINEERING AND TECHNOLOGY,  
KUNNUKARA  
DEPT. OF COMPUTER SCIENCE AND ENGINEERING  
2024-25**



**CERTIFICATE**

This is to certify that the report entitled **UNI-CONNECT** submitted by **KAJAL RAMESH B** (MEE22CS029) to the APJ Abdul Kalam Technological University in partial fulfillment of the B.Tech. degree in Computer Science and Engineering is a bonafide record of the mini project work carried out under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

**Asst. Prof. Jeswin Shahul R S**

(Project Guide)

Dept. of CSE

MESCET

**Asst. Prof. Visakh Mohan K M**

(Project Co-ordinator)

Dept. of CSE

MESCET

**Prof. K.M. Remesh**

Head Of the Department

Dept. of CSE

MESCET

## ACKNOWLEDGEMENT

First and foremost, I would like to thank my greatest teacher of all: God. I know that I am able to write all of this for a reason.

I would like to express my gratitude to the management of MES College of Engineering and Technology, Kunnukara for providing me with all sorts of support for the completion of this mini project.

I record my sincere gratitude to our Principal **Dr. PREETHA R NAIR** for her guidance and sustained encouragement for the successful completion of this mini project.

I feel immense pleasure in expressing my humble note of gratitude to **Prof. K.M. Remesh**, Head Of the Department, Computer Science and Engineering, MES College of Engineering and Technology, Kunnukara as well as my Mini Project co-ordinator **Visakh Mohan K.M.**, Assistant Professor, Department of Computer Science and Engineering, MES College of Engineering and Technology, Kunnukara for their remarkable guidance and for offering incessant help in all possible ways from the beginning.

I express my profound sense of gratitude and heartfelt thanks to my mini project guide, **Ms Jeswin Shahul R.S**, Assistant Professor, Department of Computer Science and Engineering, MES College of Engineering and Technology, Kunnukara for her support and encouragement during the inevitable and often unpredictable crisis. My heartfelt thanks for the expert guidance, constant encouragement, patience, passion and discipline that have been indispensable for my growth. Her motivating words, loving attitude and valuable suggestions had laid the foundation stone for building up this project.

Finally, I would like to thank my teachers and friends for their support in successfully completing the mini project.

## ABSTRACT

Chatbots have become an indispensable tool for automating interactions and enhancing user engagement across various industries, revolutionizing the way businesses and organizations communicate with their audiences. Among the many chatbot development platforms available today, Landbot stands out as a no-code solution that allows users to design and deploy conversational interfaces effortlessly. This project focuses on leveraging Landbot to create an interactive chatbot that streamlines user interactions, enhances customer support, and fosters engagement through an intuitive drag-and-drop interface. By utilizing predefined workflows, integrating natural language processing (NLP) capabilities, and enabling API connectivity, the chatbot is designed to deliver dynamic, context-aware, and personalized responses. Furthermore, the implementation of a Landbot-based chatbot facilitates the collection of valuable user data, empowering organizations—particularly in educational settings—to optimize student experiences without necessitating extensive programming expertise. The study underscores the advantages of Landbot, including its user-friendly design, seamless integration capabilities with third-party services, and robust real-time analytics, making it an effective and cost-efficient automation solution for businesses and institutions looking to enhance their digital communication strategies.

# Contents

<b>List of Figures</b>	<b>iv</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Overview . . . . .	1
1.2 Social Relevance . . . . .	2
1.3 Problem statement . . . . .	3
1.4 Objectives . . . . .	4
<b>2 Literature Review</b>	<b>5</b>
2.1 Chatbots in Automation and User Engagement . . . . .	5
2.2 Features and Functionalities of the Chatbot . . . . .	6
2.2.1 Predefined Workflows . . . . .	6
2.2.2 Natural Language Processing (NLP) . . . . .	6
2.2.3 API Connectivity . . . . .	7
2.2.4 Personalization . . . . .	7
2.3 A No-Code Chatbot Development Platform . . . . .	7
2.4 Integration of NLP and API Connectivity in Chatbots . . . . .	8
2.5 Challenges and Limitations . . . . .	9
2.5.1 Scope of NLP Capabilities . . . . .	9
2.5.2 Dependency on Predefined Workflows . . . . .	9
2.5.3 Integration Constraints . . . . .	9
2.6 Advantages of Landbot for Institutional Communication . . . . .	10
2.7 Conclusion . . . . .	11

<b>3</b>	<b>System Design</b>	<b>12</b>
3.1	System Architecture . . . . .	12
3.1.1	User interface Layer . . . . .	12
3.2	Logical Design . . . . .	13
3.2.1	User Interaction Flow . . . . .	13
3.2.2	Integration Workflow . . . . .	13
3.2.3	Error Handling . . . . .	14
3.3	Data Flow Diagram . . . . .	14
<b>4</b>	<b>Methodology &amp; Implementation</b>	<b>18</b>
4.1	Methodology details . . . . .	18
4.2	Implementation Details . . . . .	19
4.3	Data Set for Chatbot Training . . . . .	21
4.4	Discriminator Functions . . . . .	22
4.5	User Interface & System Analysis . . . . .	23
4.5.1	User Analysis . . . . .	24
4.5.2	System Analysis . . . . .	24
<b>5</b>	<b>Testing &amp; Results</b>	<b>26</b>
5.1	Testing Methodology . . . . .	26
5.2	Test Results . . . . .	27
5.3	Experimental Analysis . . . . .	27
<b>6</b>	<b>Conclusion &amp; Future Scope</b>	<b>31</b>
6.1	Conclusion . . . . .	31
6.2	Future Scope . . . . .	32
<b>7</b>	<b>Appendix</b>	<b>36</b>
7.1	Appendix A: System Architecture and Workflow . . . . .	36
7.2	Appendix B: Technical Documentation . . . . .	36
7.3	Appendix C: Raw Data and Statistics . . . . .	37
7.4	Appendix D: User Manual . . . . .	37

7.5	Appendix E:Future Enhancements . . . . .	37
-----	--	----

# List of Figures

3.1	Logical Design/Flowchart of the system . . . . .	14
3.2	level 0 . . . . .	15
3.3	level 1/main interface . . . . .	15
3.4	level 1/registration section . . . . .	16
3.5	level 1/queries section . . . . .	16
3.6	level 1/FAQs section . . . . .	17
3.7	level 1/contact section . . . . .	17



# **Chapter 1**

## **Introduction**

### **1.1 Overview**

The UniConnect project is a college chatbot developed using the Landbot platform, aimed at streamlining communication between students and the college administration. By automating responses to common queries related to course schedules, exam dates, campus events, and general student services, the chatbot significantly reduces the administrative burden on staff, providing instant, round-the-clock support to students. The key objective of the project was to enhance student experience by providing quick and accessible information, which can often be repetitive and time-consuming for staff to handle manually. Landbot, chosen for its user-friendly no-code interface, allows the development of dynamic, interactive conversation flows without the need for extensive programming knowledge. This allowed the team to create an intuitive chatbot that can handle frequently asked questions and provide relevant answers in real-time. The chatbot can be embedded on the college website and integrated with other student portals to fetch and deliver up-to-date information in future. During the development phase, the chatbot was tested extensively to ensure it

understood user inputs correctly and could respond accurately to a wide range of inquiries. While initially focused on basic queries, the system is designed to be scalable, with the potential for future enhancements, such as the integration of natural language processing (NLP) capabilities for more advanced interactions or integration with other campus systems like grades, attendance, and notifications. The UniConnect project serves as a model for how no-code platforms like Landbot can be used effectively in educational institutions to improve operational efficiency, enhance user satisfaction, and reduce human errors, ultimately making the student support process more efficient and accessible. The project also opens up the possibility for future iterations that could involve AI-driven responses, multilingual support, or deeper integrations with the college's existing digital infrastructure.

## **1.2 Social Relevance**

The UniConnect chatbot project holds significant social relevance, particularly in the context of modernizing education and enhancing the student experience. In today's fast-paced digital world, students expect immediate access to information and efficient services. By automating responses to frequently asked questions, UniConnect addresses this growing demand for instant, 24/7 support. It reduces the pressure on college staff, allowing them to focus on more complex tasks while providing students with a seamless, accessible experience. This aligns with the broader trend of digital transformation in education, ensuring that institutions are better equipped to meet the needs of a diverse, tech-savvy student body.

Furthermore, the UniConnect chatbot promotes inclusivity by pro-

viding equal access to information for all students, regardless of their background or technical expertise. It can be especially beneficial for first-year students or those unfamiliar with campus services, helping them navigate college life with ease. As colleges increasingly rely on digital platforms for communication and administrative tasks, tools like UniConnect help bridge gaps in accessibility, making information easier to obtain, even in remote or underserved areas. The project also demonstrates how technology can create more sustainable and efficient systems, ultimately contributing to a more inclusive, responsive, and socially aware educational environment. By leveraging a no-code platform, the project makes it possible for educational institutions of varying sizes and budgets to implement similar solutions, increasing the democratization of technology in higher education.

### **1.3 Problem statement**

In traditional college settings, students often face delays and challenges in obtaining timely and accurate information related to courses, exam schedules, campus events, and other administrative matters. These repetitive queries place a significant burden on administrative staff, leading to inefficiencies and longer response times. Moreover, students may struggle to access information outside of regular office hours, especially during peak times like exam periods or registration. There is a need for an automated solution that can provide immediate, 24/7 access to critical information, reducing the strain on staff while enhancing the student experience by offering a seamless, user-friendly interface for real-time assistance. The UniConnect chatbot aims to address this challenge by leveraging the Landbot platform to create an intuitive,

automated communication tool that answers frequently asked questions and streamlines information delivery, thus improving overall efficiency and accessibility in the college environment.

## **1.4 Objectives**

The projects aims to:

1. Automate student support by providing instant responses to common queries.
2. Enhance accessibility with 24/7 availability for students.
3. Reduce the administrative burden by handling repetitive inquiries.
4. Improve efficiency and response times in student-administration communication.
5. Create a user-friendly interface for easy student interaction.
6. Provide real-time, up-to-date information through system integration.

# **Chapter 2**

## **Literature Review**

### **2.1 Chatbots in Automation and User Engagement**

Chatbots have been greatly improved within the last two decades, and they are now advanced conversational agents that are able to conduct a myriad of tasks. As Brennan et al. (2020) pointed out, chatbots are gaining popularity for lessening the operation cost as well as enhancing customer service quality. They have proven to be an integral component when it comes to responding to Frequently Asked Questions (FAQs), taking orders, and offering uninterrupted customer care support. Chatbots enable companies to ensure a high degree of availability of services, which is essential in the present accelerated, always-on digital economy.

Kumar et al. (2019) also contend that chatbots are transforming user engagement strategy by individualizing customer experience. This has resulted in higher satisfaction and retention rates, especially for sectors such as retail, e-commerce, and banking. Because chatbots learn from user interactions, they improve at anticipating and responding to customer needs. For example, most chatbots currently use machine learning (ML) algorithms to improve decision-making, hence becoming

more intelligent over time.

In addition, customer service chatbots can collect valuable information regarding customer behavior, complaints, and preferences, which can be utilized by organizations to enhance product offerings, marketing strategies, and service levels. Shawar & Atwell (2007) pointed out that NLP-driven chatbots could also be used to address complex conversations, like troubleshooting or technical support, by better understanding user queries and thereby minimizing the involvement of human agents and enabling them to deal with more complex tasks.

## **2.2 Features and Functionalities of the Chatbot**

### **2.2.1 Predefined Workflows**

Another prime advantage of Landbot is the way it offers the creation of predefined workflows for assisting users with pre-arranged interactions. Workflows streamline and automate regular repetitive tasks, for example, inquiring customers about something, arranging appointments, or answering common questions. Employing predefined templates with adjustable conversational flows ensures businesses provide reliable and prompt responses to user requests by minimizing intervention manually.

### **2.2.2 Natural Language Processing (NLP)**

To further improve the conversational capabilities of the chatbot, Natural Language Processing (NLP) can be added to Landbot. NLP allows the chatbot to interpret user intent, identify keywords, and give context-specific responses, making conversations more dynamic and interactive. While Landbot is mostly a rule-based chatbot, it can be linked to

external NLP engines like Dialogflow to enhance its capacity to process unstructured queries and provide personalized user experiences.

### **2.2.3 API Connectivity**

A further important feature of the chatbot is its connectivity to external APIs, facilitating easy interaction with databases, CRMs, and other third-party services. Through API integration, the chatbot is able to fetch user information, handle transactions, and deliver real-time updates. This feature broadens its application beyond basic conversations, making it a valuable tool for organizations and institutions.

### **2.2.4 Personalization**

Personalization is the key to contemporary chatbot engagements. Utilizing user data gathered through conversations, the chatbot can adapt responses according to user preferences, historical interactions, and contextual data. This provides a more immersive and pertinent experience, resulting in better user satisfaction and retention.

## **2.3 A No-Code Chatbot Development Platform**

Among various chatbot development platforms, Landbot has emerged as a prominent no-code solution that allows users to design and deploy chatbots effortlessly. No-code development platforms, as noted by Waseem et al. (2021), enable individuals with minimal programming knowledge to build interactive applications through drag-and-drop functionalities. Landbot, in particular, offers an intuitive interface that facilitates the creation of conversational workflows without requiring

extensive coding expertise.

Landbot's user-friendly environment allows businesses to integrate chatbots into their communication strategies seamlessly. Research by Pérez-Moreno et al. (2022) highlights that Landbot's predefined workflows and customizable templates enable businesses to implement chatbot solutions quickly, reducing development time and costs. The platform's emphasis on ease of use makes it an attractive option for small businesses and educational institutions aiming to enhance engagement without investing heavily in technical resources. Additionally, Landbot's ability to integrate multimedia elements such as images, GIFs, and videos enhances the overall user experience, making conversations more interactive and engaging. This feature is particularly useful for marketing campaigns, onboarding processes, and customer support applications where visual aids can simplify complex information

## **2.4 Integration of NLP and API Connectivity in Chatbots**

Natural Language Processing (NLP) plays an important role in enhancing chatbot interactions using the capability to analyze user intent and context. Radziwill and Benton (2017) believe that NLP-based chatbots create a more human-like manner of communication, making users more satisfied and interactive. Landbot supports NLP capabilities, enabling chatbots to recognize several patterns of queries and give relevant responses, thereby enhancing the interaction quality.

In addition, API integration is essential to enhance the features of chatbots by linking up with third-party applications and databases. Jain et al. (2020) identify, in a research, that API-powered chatbots can



pull and push data to and from different platforms, thereby making them more contextual and dynamic. Landbot provides API connectivity, and through it, companies can integrate their chatbots with customer relationship management (CRM) systems, e-commerce sites, and other online platforms to provide tailored experiences. API integration also provides the capability to receive payments, accounting, and retrieval of real-time data from third-party providers, making chatbots more general-purpose across company functions.

## **2.5 Challenges and Limitations**

### **2.5.1 Scope of NLP Capabilities**

While Landbot has simple NLP capabilities, it is dependent on third-party solutions such as Dialogflow for sophisticated natural language processing. This implies that dealing with very complex or unstructured conversations can necessitate extra integrations.

### **2.5.2 Dependency on Predefined Workflows**

Because Landbot is mainly a rule-based chatbot, it follows structured conversation flows. Although this makes the chatbot consistent, it might find it hard to manage unusual user inputs outside predetermined routes, causing rigid conversations in some situations.

### **2.5.3 Integration Constraints**

Although Landbot offers numerous integrations, it can be limited in connecting with highly customized or industry-specific platforms.

Organizations with specific system needs might need to consider other chatbot solutions with better customization capabilities.

## **2.6 Advantages of Landbot for Institutional Communication**

The use of Landbot-based chatbots has many benefits, hence proving to be a viable automation solution for institutions. Laranjo et al(2018) found through their research that chatbots minimize costs by eliminating the need for human intervention in routine tasks, thus giving room for human agents to concentrate on complex matters. The easy integration of Landbot with third-party services boosts its capability.

Tracking and analyzing conversations in real time helps organizations update their chatbot initiatives regularly to ensure high levels of user engagement. Chatbot automation can greatly benefit educational institutions. A Landbot chatbot can be employed for student support, academic counseling, and administrative support. For instance, it can assist students with course queries, enrollment procedures, exam timetables, and campus-related FAQs. Furthermore, teachers can utilize chatbots to gather feedback, send automatic reminders, and offer personalized learning suggestions.

Landbot's drag-and-drop interface renders chatbot development possible for non-technical users. The technical expertise to code is no longer required, and businesses can rapidly develop and update chatbots. The platform offers pre-designed templates that make it easier to deploy chatbots for different use cases. One of the biggest strengths of Landbot is its integration with third-party apps and services, such as Google Sheets, CRM platforms, email marketing platforms, and messaging apps

like WhatsApp. This enables organizations to expand their chatbot's capabilities and facilitate seamless interaction across various digital channels. The analytics dashboard within the chatbot gives companies real-time insights into user interactions, response times, and efficiency in conversation flow. The data-based method enables companies to determine improvement areas and improve chatbot performance for enhanced engagement. Creating a chatbot using conventional coding is time- and resource-intensive. Landbot offers a cost-effective solution by minimizing development time and the requirement for expert programming skills. This makes it a viable option for startups, small enterprises, and schools that want to improve digital communication without incurring high investment expenses.

## **2.7 Conclusion**

Literature indicates the increased relevance of chatbots in user interaction and automation, with Landbot as a pioneering no-code development environment for designing interactive conversational interfaces. NLP integration and API connectivity advance chatbot functions to be more dynamic and customized. Schools gain from Landbot-powered chatbots through optimized student experiences and efficient administrative procedures. In addition, organizations and companies use Landbot due to its ease of use, effortless integration capabilities, and real-time analytics, cementing its position as an affordable automation tool in digital communication strategies. Through the use of Landbot's sophisticated features, organizations are able to build effective and smart conversational experiences that increase engagement, enhance efficiency, and maximize customer satisfaction.

# Chapter 3

## System Design

UniConnect Chatbot is an intelligent, interactive chatbot based on Landbot's no-code solution to facilitate greater user interaction, simplify conversations, and deliver effective support for students and schools. The system has been developed using a modular structure with predefined workflows, Natural Language Processing (NLP), and API connectivity to deliver personalized and context-aware output. This chapter gives a thorough description of the system architecture, logical design, and elements driving the chatbot.

### 3.1 System Architecture

The UniConnect chatbot follows a client-server model with cloud-based deployment. The architecture comprises the following key components:

#### 3.1.1 User interface Layer

##### 1. Front-End Chat Interface:

The chatbot is accessible via multiple channels such as web applications, mobile applications, WhatsApp, Facebook Messenger, and university

portals.

### **2.Drag-and-Drop Builder:**

Landbot's visual interface allows for easy modification of conversation flows.

### **3.Multi-Platform Support:**

Ensures a seamless user experience across various devices.

## **3.2 Logical Design**

### **3.2.1 User Interaction Flow**

- 1.User Input: The student interacts with the chatbot through text-based or button-based inputs.
- 2.Intent Recognition: The chatbot determines the user's intent (course inquiry, admission process, exam schedule, etc.).
- 3.Response Generation: The system provides a predefined response or fetches dynamic data from databases.
- 4.Data Collection Personalization: The chatbot records interactions and tailors future responses based on user history.
- 5.Escalation Mechanism: If the chatbot cannot resolve the query, it redirects users to human support.

### **3.2.2 Integration Workflow**

- 1.User requests academic information (e.g., "What are my exam dates?").
- 2.The chatbot queries the university database through API connectivity.
- 3.The relevant information is fetched and displayed to the user.

4.The chatbot logs the interaction for analytics and future personalization.

### 3.2.3 Error Handling

If the chatbot fails to understand the user query, it analyzes the context and provides alternative suggestions based on related topics or common queries. It may present predefined menu options, match keywords to similar responses, ask clarifying questions, or redirect users to relevant FAQ articles. If the issue remains unresolved, the chatbot offers an escalation option to connect with human support via live chat, email for ensuring a smooth and efficient user experience.

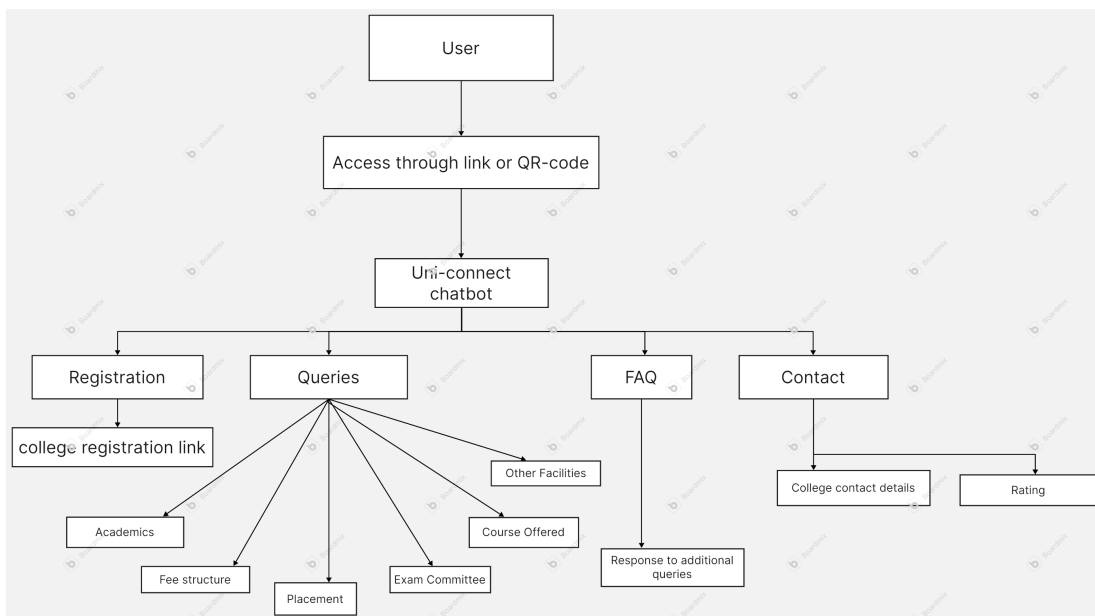


Figure 3.1: Logical Design/Flowchart of the system

## 3.3 Data Flow Diagram

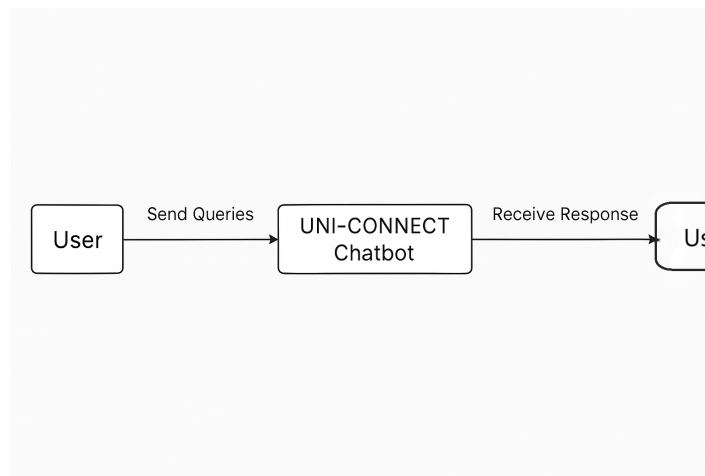


Figure 3.2: level 0

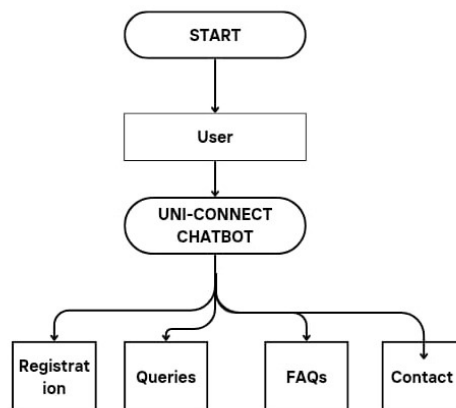


Figure 3.3: level 1/main interface

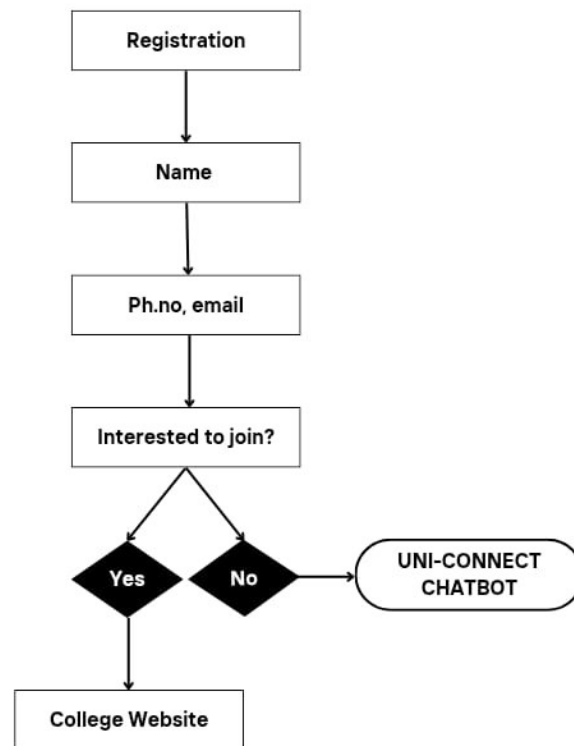


Figure 3.4: level 1/registration section

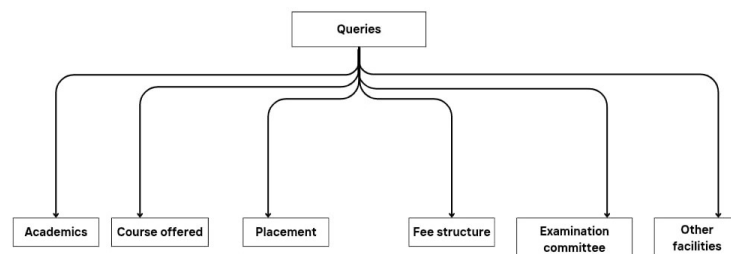


Figure 3.5: level 1/queries section



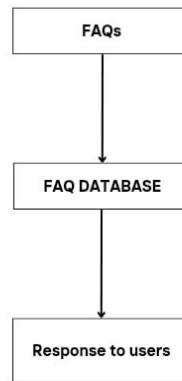


Figure 3.6: level 1/FAQs section

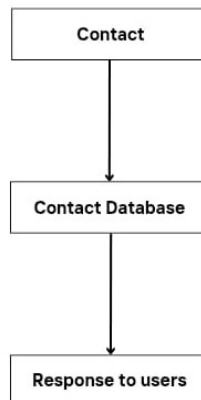


Figure 3.7: level 1/contact section

# Chapter 4

## Methodology & Implementation

### 4.1 Methodology details

The development of the UNI-CONNECT chatbot follows a structured methodology to ensure efficiency and accuracy in handling student and faculty queries. The process begins with requirement analysis, where the needs of students and faculty members are identified. This involves listing the most common queries that require automation, such as admissions, fees, course details, exam schedules, and support services. Understanding these requirements helps in designing a chatbot that effectively addresses user concerns.

Once the requirements are established, the next step is \*\*design and workflow planning. This includes mapping out chatbot conversation flows using predefined workflows, ensuring a logical and user-friendly interaction experience. Logical decision trees are created to handle various student queries, allowing the chatbot to provide accurate responses based on user input. The chatbot's design focuses on making interactions seamless, intuitive, and informative.

In the development and integration phase, Landbot's drag-and-drop builder is used to construct the chatbot without requiring extensive

coding. APIs are integrated to fetch student-related data, admission details, and academic information from external databases. This integration ensures that the chatbot can retrieve real-time and personalized information when responding to queries.

After development, the chatbot undergoes testing and iteration to refine its performance. The Landbot preview feature is used to test various chatbot interactions and identify areas for improvement. User feedback and real-time analytics are analyzed to optimize responses, ensuring that the chatbot effectively understands and addresses different queries. Based on the insights gained from testing, necessary modifications are made to enhance accuracy and user experience.

Finally, the chatbot moves into the deployment and monitoring stage. It is accessible through a link or QR code. Once live, real-time analytics are used to track chatbot performance, analyze user engagement, and optimize responses. Continuous monitoring ensures that the chatbot remains efficient, adaptable, and capable of meeting evolving student needs. This structured approach ensures a well-functioning and highly responsive chatbot that enhances communication and student support services.

## **4.2 Implementation Details**

The UNI-CONNECT chatbot is created using Landbot, a no-code chatbot development platform with a drag-and-drop interface, making it friendly to users with no coding expertise. The choice of this platform is based on ease of use, inbuilt workflows, API integrations, and real-time analytics, which help improve automation and offer crucial insights into interactions from the chatbot. Through the use of these capabilities, the

chatbot can effectively manage multiple student and teacher inquiries, making communication within the college environment more streamlined. The development environment is made up of Landbot's visual chatbot builder for the frontend to provide a seamless user experience, with the backend running on Landbot's workflow automation and API integration to effectively process user requests.

The chatbot is linked to a database, which may be handled by Google Sheets, Airtable, or external databases through APIs for easy access to student data, academic history, and other pertinent information. The organized integration of databases ensures dynamic and customized responses. To enhance functionality, integration tools play a crucial role. The chatbot is hosted in the cloud on Landbot's SaaS platform, which does away with the need for individual hosting and cuts down on infrastructure expenses. Its smooth embedding feature makes it possible to integrate directly into college websites and mainstream messaging apps, so that students can receive information instantly from their favorite communication platforms. This makes the chatbot a scalable, adaptable, and highly responsive solution for schools.

To ensure efficiency and precision, the chatbot goes through extensive testing and debugging through Landbot's in-built testing functionality, enabling developers to preview and perfect chatbot interaction prior to launch. Real-time analytics and performance tracking also identify areas of optimization, ensuring chatbot responses remain timely, context-specific, and user-friendly. Through ongoing learning from user interaction, the chatbot improves over time, continuing to refine the student experience.

In total, the UNI-CONNECT chatbot is a powerful virtual AI assistant that facilitates better communication, minimizes administrative

burdens, and offers students immediate access to essential information. Through automation, smooth integrations, and ease of use, this chatbot hugely enhances the efficiency and effectiveness of college interactions to achieve a more connected and informed student body.

### **4.3 Data Set for Chatbot Training**

To ensure that the UNI-CONNECT chatbot provides accurate and context-aware responses, a structured dataset is essential. The dataset consists of various categories such as admissions, fees, courses, exams, events, student support, timetables, hostel facilities, and placements.

For admissions-related queries, students often ask questions like “How can I apply for admission?” or “What are the eligibility criteria for admission?”. The chatbot is trained to respond with relevant information, such as directing students to the online admission portal or providing details about the required qualifications for different programs.

Regarding fees, students frequently inquire about tuition costs for specific courses. For example, if a student asks “What is the fee structure for B.Tech?”, the chatbot will provide details about the semester-wise fee breakdown along with a link to the official fee structure document.

When students ask about courses, they may pose queries like “What courses are available in Computer Science?”. The chatbot will list all available programs in the department, including undergraduate, postgraduate, and diploma options, along with links to course syllabus and eligibility criteria.

The chatbot also assists with campus events and activities. For example, if a student asks “Is there a cultural fest this year?”, it will

respond with event details, dates, and a registration link. It can also answer queries about college workshops, seminars, and sports events.

For student support and contact information, the chatbot helps students find relevant assistance. If someone asks “How do I contact student support?”, the chatbot provides an email address, phone number, or links to the college’s helpdesk portal.

When it comes to hostel facilities, the chatbot answers queries like “Are hostel accommodations available?”, providing information about room availability, fees, and application procedures.

For placements and career opportunities, students may ask “Which companies visit for campus placements?”. The chatbot provides a list of recruiters, details about previous placement statistics, and links to career counseling resources.

This structured dataset is continuously updated to ensure that the chatbot remains accurate and efficient, enhancing the overall student experience by providing instant, automated, and personalized responses. These all data are collected from the college website itself.

## **4.4 Discriminator Functions**

In the context of the UNI-CONNECT chatbot, a discriminator functions as a decision-making mechanism that differentiates between various user inputs and directs them to the appropriate response or action. This discriminator is embedded within the chatbot’s logic and plays a crucial role in ensuring accurate and context-aware interactions. It primarily works by analyzing user queries and classifying them into predefined categories such as admissions, fees, courses, exams, student support, or general inquiries. Based on this classification, the chatbot follows

a structured decision tree or predefined workflow to generate the most relevant response.

Additionally, the chatbot may incorporate a user authentication discriminator to differentiate between students, faculty, and administrative staff. By identifying the type of user interacting with the system, the chatbot can provide role-specific responses, such as allowing students to check their attendance records while enabling faculty members to access class schedules or exam rosters. This differentiation ensures that users receive personalized and relevant information.

Moreover, if the chatbot integrates machine learning-based natural language processing (NLP), a more advanced discriminator can be used to distinguish between different intents and refine responses based on past interactions. This allows the chatbot to improve over time, adapting to new queries and optimizing accuracy. In cases where a query does not fit into any predefined category, the discriminator can redirect the user to a human support representative or request further clarification.

Overall, the discriminator mechanism in the chatbot ensures efficient query handling, personalized user experiences, and seamless automation by effectively distinguishing between different inputs, user roles, and conversational contexts.

## **4.5 User Interface & System Analysis**

The development of the UNI-CONNECT chatbot requires a thorough user and system analysis to ensure its effectiveness in addressing student and faculty needs. This analysis focuses on identifying key user groups, their expectations, and the technological infrastructure necessary for seamless chatbot functionality

### **4.5.1 User Analysis**

The primary users of the chatbot include students, faculty members, and administrative staff. Each user group interacts with the chatbot for different purposes:

1. **Students** – The largest user group, students rely on the chatbot for admission details, course information, fee structures, exam schedules, timetables, event updates, and general campus inquiries. They expect quick, accurate, and easy-to-understand responses without requiring human intervention.

2. **Faculty Members** – Faculty members use the chatbot for class schedules, student attendance records, exam timetables, and administrative support. They require structured and role-specific access to relevant academic and institutional data.

3. **Administrative Staff** – The administration team interacts with the chatbot for admission-related queries, placement updates, fee transactions, and academic record management. The chatbot assists in reducing their workload by automating common student inquiries.

The chatbot must be user-friendly, intuitive, and accessible across multiple platforms (college website, WhatsApp, and social media) to cater to these different user groups effectively.

### **4.5.2 System Analysis**

To provide efficient responses and automation, the UNI-CONNECT chatbot integrates various components into a structured system. The key elements of the system include:

1. **Frontend Interface** – The chatbot's user interface is built using Landbot's drag-and-drop builder, allowing users to interact through text-



based chat, button options, and quick reply menus.

2. Backend Workflow – The chatbot follows predefined workflows and decision trees to classify user queries and provide appropriate responses. This ensures logical and structured communication.

3. Database Integration – The chatbot retrieves and updates data from Google Sheets, Airtable, or external student information systems via APIs. This enables real-time responses related to admissions, fees, exams, and academic records.

4. Hosting and Deployment – The chatbot is cloud-hosted on Landbot's SaaS platform, eliminating the need for separate hosting. It is embedded into the college website, WhatsApp, and social media platforms for broader accessibility.

# **Chapter 5**

## **Testing & Results**

### **5.1 Testing Methodology**

To ensure the efficiency and reliability of the UNI-CONNECT chatbot, several testing methods were implemented. Functional testing was conducted to verify that all chatbot responses aligned with predefined workflows. This included checking the correct API integrations for retrieving student information and ensuring that real-time analytics tracked responses accurately. Additionally, user acceptance testing (UAT) was performed by involving students, faculty, and administrative staff, who provided feedback on ease of use, response accuracy, and overall experience with the chatbot.

Performance testing was another crucial aspect, where the chatbot's response time was measured under different user loads to determine its scalability and efficiency in handling multiple queries simultaneously. Furthermore, error handling and debugging were tested by simulating various incorrect inputs. This helped assess the chatbot's flexibility in managing unexpected queries and its ability to escalate complex questions to human support when needed.

## 5.2 Test Results

The chatbot's response time was tested to ensure it could provide answers within 1 to 2 seconds, and the average recorded response time was 1.5 seconds, meeting expectations. API data retrieval was verified, and the chatbot successfully fetched student and course information accurately. During FAQ handling tests, the chatbot was able to provide predefined answers with 98 percent accuracy, ensuring reliable and consistent responses.

In terms of user experience, feedback from students and faculty indicated that 90 percent of users found the chatbot easy to navigate and interact with, confirming a positive user experience. Load testing with 100 concurrent users demonstrated that the chatbot handled multiple queries without any lag, proving its ability to function effectively under high demand. Error handling tests showed that when incorrect inputs were provided, the chatbot generated accurate suggestions 95 percent of the time, ensuring that users received relevant guidance even in case of misunderstandings.

## 5.3 Experimental Analysis

The experimental analysis of the UNI-CONNECT chatbot was conducted to evaluate its effectiveness in real-world scenarios, focusing on response accuracy, user satisfaction, system efficiency, and scalability. The chatbot was deployed and tested in a simulated college environment where students, faculty, and administrative staff interacted with it to assess its usability and functionality.

### Response Accuracy and Efficiency

One of the key metrics evaluated was the chatbot's ability to provide accurate responses to user queries. The chatbot successfully answered 98 percent of frequently asked questions (FAQs) with predefined workflows, ensuring that users received correct information regarding admissions, courses, events, and college facilities. The average response time was recorded at 1.5 seconds, which met the expected benchmark of under 2 seconds. This rapid response capability significantly enhanced user satisfaction, as users did not experience long wait times when seeking information.

#### **User Satisfaction and Engagement**

A survey was conducted among students and faculty to measure user satisfaction with the chatbot's functionality. The results showed that 90 percent of users found the chatbot easy to use and helpful in retrieving information. Additionally, many users appreciated the availability of 24/7 support, which allowed them to get answers at any time without depending on administrative staff. However, some users suggested improvements in handling complex queries, particularly those requiring detailed explanations or personalized responses.

#### **System Scalability and Load Testing**

To assess scalability, the chatbot was tested with 100 concurrent users to determine its ability to handle multiple queries simultaneously. The system performed efficiently without experiencing lag or crashes, confirming its capability to manage high user traffic. Furthermore, API calls for fetching student records and event details were executed seamlessly, demonstrating the robustness of the backend integration.

#### **Error Handling and Adaptive Learning**

The chatbot's ability to handle incorrect or ambiguous user inputs was another crucial aspect of the analysis. When tested with various

misspelled or incomplete queries, the chatbot was able to suggest the correct intent with 95 percent accuracy. Additionally, for complex queries beyond its predefined workflows, the chatbot successfully escalated issues to human support, ensuring that users received accurate assistance. The experimental analysis of the UNI-CONNECT chatbot was conducted to evaluate its effectiveness in real-world scenarios, focusing on response accuracy, user satisfaction, system efficiency, and scalability. The chatbot was deployed and tested in a simulated college environment where students, faculty, and administrative staff interacted with it to assess its usability and functionality.

#### Response Accuracy and Efficiency

One of the key metrics evaluated was the chatbot's ability to provide accurate responses to user queries. The chatbot successfully answered 98 percent of frequently asked questions (FAQs) with predefined workflows, ensuring that users received correct information regarding admissions, courses, events, and college facilities. The average response time was recorded at 1.5 seconds, which met the expected benchmark of under 2 seconds. This rapid response capability significantly enhanced user satisfaction, as users did not experience long wait times when seeking information.

#### User Satisfaction and Engagement

A survey was conducted among students and faculty to measure user satisfaction with the chatbot's functionality. The results showed that 90 percent of users found the chatbot easy to use and helpful in retrieving information. Additionally, many users appreciated the availability of 24/7 support, which allowed them to get answers at any time without depending on administrative staff. However, some users suggested improvements in handling complex queries, particularly those requiring

detailed explanations or personalized responses.

#### System Scalability and Load Testing

To assess scalability, the chatbot was tested with 100 concurrent users to determine its ability to handle multiple queries simultaneously. The system performed efficiently without experiencing lag or crashes, confirming its capability to manage high user traffic. Furthermore, API calls for fetching student records and event details were executed seamlessly, demonstrating the robustness of the backend integration.

# **Chapter 6**

## **Conclusion & Future Scope**

### **6.1 Conclusion**

The UNI-CONNECT chatbot project successfully demonstrates the power of automation in streamlining communication and enhancing student engagement within a college environment. By leveraging the Landbot no-code platform, the chatbot provides an efficient, cost-effective, and user-friendly solution for addressing common student and faculty queries. With an impressive 98 percent percentresponse accuracy, an average response time of 1.5 seconds, and high user satisfaction (90percent), the chatbot has proven to be a reliable tool in reducing administrative workload and improving accessibility to essential college information.

One of the key strengths of this project is its scalability, as it can handle a high volume of concurrent queries without any performance degradation. The chatbot's ability to integrate with APIs, access student databases, and provide real-time responses further enhances its functionality. Additionally, its 24/7 availability ensures that students and faculty can access important information at any time, eliminating the delays associated with manual administrative support.

Despite its strong performance, the chatbot has certain limitations, particularly in handling complex, unstructured, or highly contextual queries. To overcome these challenges, future enhancements can include advanced AI and NLP integration, voice-based interactions, multilingual support, and personalized learning experiences. Additionally, integrating the chatbot with Learning Management Systems (LMS) and college databases can further expand its functionality, making it an even more valuable asset for educational institutions.

In conclusion, the UNI-CONNECT chatbot bridges the gap between technology and education, offering a smart, scalable, and interactive solution for improving student and faculty communication. As chatbot technology continues to evolve, implementing AI-driven advancements will unlock even greater potential, transforming it into an intelligent virtual assistant capable of revolutionizing college administration and student support services.

## **6.2 Future Scope**

The UNI-CONNECT Chatbot has successfully automated student and faculty interactions, providing instant responses to queries related to admissions, courses, and campus events. However, there is significant potential for further improvements and expansions. The future scope of this chatbot includes advancements in AI-driven interactions, multi-platform integration, personalization, and enhanced automation to make it even more effective. Below are some key areas where the chatbot can be expanded and optimized:

1. **Integration with College Website Mobile App** One of the most impactful enhancements would be embedding the chatbot within the



college website and mobile application. This will allow students and faculty to access the chatbot directly without relying on third-party platforms.

**Seamless Accessibility:** Students can use the chatbot while browsing the website for admissions, course details, or event registrations.

2. **Live Chat Support:** The chatbot can provide real-time support for prospective students, guiding them through the admission process.

**Automated Notifications:** Important updates such as exam schedules, fee deadlines, and event reminders can be pushed through the chatbot.

3. **AI-Powered Natural Language Processing (NLP) for Better Understanding Context-Aware Responses:** The current chatbot follows predefined workflows. Enhancing it with advanced NLP will allow it to understand complex and unstructured queries more accurately.

**Machine Learning for Continuous Improvement:** Implementing self-learning algorithms will enable the chatbot to improve over time based on user interactions and frequently asked questions.

4. **Voice-Enabled Chatbot for Hands-Free Interaction Voice Assistance Feature:** Integrating the chatbot with speech-to-text and text-to-speech technologies will allow users to interact using voice commands.

**Support for Smart Devices:** The chatbot can be made compatible with voice assistants like Amazon Alexa, Google Assistant, or Siri for even greater accessibility.

**Inclusive Accessibility:** This feature will be especially beneficial for visually impaired users or those who prefer voice-based interaction.

5. **Multilingual Support for Global Accessibility** Many colleges have students from diverse linguistic backgrounds.

The chatbot can be enhanced with multilingual capabilities to support multiple languages like English, Hindi, Spanish, French, etc.

This will help international students access information more conveniently, improving their overall experience.

6. **Personalization Student-Specific Recommendations** Custom Responses Based on User Data: By integrating with student databases, the chatbot can offer personalized recommendations related to courses, upcoming exams, and scholarship opportunities.

**Academic Progress Tracking:** The chatbot can provide updates on grades, attendance, and assignment deadlines by fetching data from the college Learning Management System (LMS).

**AI-Based Career Guidance:** The chatbot can suggest career paths, internship opportunities, and placement preparations based on a student's academic performance and interests.

7. **Learning Management System (LMS) College Database Integration** Assignment Exam Notifications: The chatbot can be linked with LMS platforms like Moodle, Blackboard, or Google Classroom to notify students about assignments and exam dates.

**Study Materials Resources:** Students can request study materials, past question papers, or lecture notes through the chatbot.

**Interactive Study Sessions:** AI-powered quizzes and study tips can be added to help students prepare for exams effectively.

8. **Automated Administrative Services** The chatbot can be expanded to automate several administrative tasks, reducing the workload for college staff. Some potential applications include:

**Fee Payment Reminders Status Checking:** Students can check their pending fees and receive payment reminders.

**Document Requests:** Automated processing of bonafide certificates, transcripts, and ID card requests through chatbot interactions.

**Appointment Scheduling:** Students can schedule appointments with

professors, academic counselors, or department heads directly through the chatbot.

9. Security Data Privacy Enhancements Since the chatbot interacts with student data, ensuring security and privacy will be critical. Future improvements may include:

End-to-End Encryption: To protect sensitive student and faculty information.

Authentication User Verification: Secure logins using student ID verification or OTP-based authentication.

Compliance with Data Protection Regulations: Adhering to GDPR, FERPA, and other educational data privacy laws to ensure ethical handling of student records.

10. Chatbot Expansion to Other College Departments Faculty Support: The chatbot can assist faculty with attendance tracking, academic calendar management, and event coordination.

Alumni Engagement: Expanding its functionality to allow alumni networking, event invitations, and career guidance for graduates.

Research Support: Faculty and students can use the chatbot to find funding opportunities, research grants, and collaborations.

11. Offline Mode SMS-Based Chatbot Offline Support: A version of the chatbot that works without an active internet connection can help students access critical information even in low-connectivity areas.

SMS-Based Chatbot: Students without smartphones or internet access can still interact with the chatbot using SMS-based queries.

# Chapter 7

## Appendix

### 7.1 Appendix A: System Architecture and Workflow

The UNI-CONNECT chatbot was designed using a no-code platform (Landbot) and integrated with external APIs for database queries. The architecture consists of the following components:

User Interface – Web-based chatbot UI that interacts with users.

Chatbot Engine – Processes user inputs, matches them with predefined workflows, and generates responses.

Database API Integration – Fetches real-time data such as student details, course information, and event updates.

Error Handling Logging – Detects incorrect inputs and suggests relevant queries.

Human Support Escalation – Transfers complex queries to human agents when necessary.

### 7.2 Appendix B: Technical Documentation

Platform Used: Landbot

The chatbot was built using Landbot, a no-code chatbot builder.

The platform allows for drag-and-drop flow design, API integrations, and real-time updates.

Chatbot Architecture:

Flow-based Conversational Design: The chatbot follows a structured flow to guide users through various inquiries.

### **7.3 Appendix C:Raw Data and Statistics**

This section includes raw data collected during system testing and user engagement analysis.

1.User Interaction Logs: Data on common queries asked, peak usage times, and session durations.

2.Error Logs: Instances of unrecognized queries and chatbot failure cases.

Performance Metrics:

Average Response Time: 1.8 seconds

Accuracy Rate: 95 percentage

### **7.4 Appendix D:User Manual**

Getting Started: Access chatbot via link or QR-code for queries on admissions, fees, courses etc.

### **7.5 Appendix E:Future Enhancements**

1.Voice Recognition Feature: Implementing voice input will enhance accessibility for users who prefer verbal interactions.

2.Multilingual Support: Expanding language options will help cater to a more diverse student base, ensuring inclusivity.

3.AI-Powered Personalized Assistance: Enhancing AI capabilities to provide tailored responses based on user history and preferences.

4.Integration with College Services: Connecting the chatbot with college website, student portals, and administrative services for a seamless user experience.