

DEPARTMENT OF AI, ML & DS

(AI742E05)

AI for Business

B. Tech

(Artificial Intelligence and Machine Learning)

*CIA 3 Mini Project on
Library Book Availability chatbot*

School of Engineering and Technology,

CHRIST (Deemed to be University),

Kumbalagodu, Bengaluru-560 074

September 2025

Noel Tony (2263048) has successfully submitted the Mini Project for the course **AI742E05 – AI for Business** as part of **CIA 3, Component** Assessment, during the academic year 2025–2026.

FACULTY- IN CHARGE

Name :

Register No. :

Examination Center : SoET, CHRIST (Deemed to be University)

Date of Submission :

TABLE OF CONTENTS

Sl. No	Description	Page No.
1	Introduction	4
2	Literature Review / Existing System	5
3	Proposed System / Methodology	6
4	Implementation & Result	7
5	Conclusion & Future Scope	8
6	References	9

Introduction

The Library Chatbot System is an innovative solution designed to enhance user experience in academic libraries by providing instant, automated responses to common library-related queries. This system has been designed in two versions: a simple implementation and an advanced implementation. The simple version is a lightweight, rule-based chatbot that uses keyword matching and is built with Streamlit to provide a user-friendly interface. It focuses on handling common library queries with predefined responses and is ideal for small to medium-sized libraries. On the other hand, the advanced version uses sophisticated natural language understanding (NLU) with zero-shot classification, context-aware conversation management, and entity extraction capabilities, making it more suitable for larger institutions with complex query patterns.

The problem this system aims to solve arises from the limitations of traditional library information services. Human staff availability is limited, repetitive queries consume staff time, and students or users often require instant answers to basic inquiries. Furthermore, traditional methods of providing information are prone to inconsistencies. The objectives of the chatbot system are to provide 24/7 automated library assistance, reduce the workload on staff, ensure consistent and accurate information delivery, improve the user experience, and handle multiple user queries simultaneously.

The **objectives** of this project are therefore defined as:

1. Design a chatbot to handle multiple library-related queries
2. Combine rule-based methods with transformer models for accurate intent recognition.
3. Deliver context-aware responses based on user queries
4. **user-friendly interface** using Streamlit, making the chatbot accessible and practical .

By achieving these objectives, this project contributes to building a tool that not only reduces the effort involved in internship preparation but also increases the chances of students securing quality opportunities.

Literature Review / Existing System

Career assistance systems have evolved significantly, starting from rule-based chatbots to modern AI-driven conversational agents. A review of existing solutions highlights the **strengths and limitations** of each:

1. **Multi-Intent Handling:**

Design a chatbot capable of addressing multiple library-related queries such as book availability, borrowing rules, membership details, and events.

A structured intent-response system with at least 15 predefined intents.

2. **Hybrid Intent Recognition:**

Integrate both rule-based methods and transformer-based NLP models for improved and flexible intent recognition.

A dual-layer architecture combining keyword matching for simple queries and Hugging Face zero-shot classification for complex ones.

3. **Context-Aware Responses:**

Provide personalized responses that adapt based on user queries, such as book titles, authors, or borrowing history.

Incorporation of entity extraction (book names, authors) and conversation context tracking.

4. **User-Friendly Interface:**

Build an accessible and intuitive chatbot interface for seamless user interaction.

A Streamlit-based frontend with conversation history, real-time responses, and mobile responsiveness.

Identified Gaps

- Traditional methods are limited by staff availability and operational hours.
- Existing chatbots mostly provide static, fixed responses with little adaptability.
- Lack of personalization or context-awareness in query handling.
- Scalability issues in handling multiple users simultaneously.
- Limited integration with modern NLP techniques for dynamic understanding.

Proposed System / Methodology

The proposed chatbot follows a **hybrid methodology** that blends both rule- based and AI-driven approaches:

Simple Implementation

- Uses **Streamlit UI**, keyword-based intent matching, predefined response dictionary, and session state management.
- Supports **16 predefined intents** and applies basic pattern matching for query understanding.
- Maintains **conversation history** for better interaction.
- **Easy to deploy** and suitable for small-scale library operations.

Advanced Implementation

- Incorporates **NLU module (Hugging Face Transformers)**, **entity extraction (spaCy)**, a conversation manager, response generator, and a mock database.
- Features **zero-shot classification** for dynamic intent recognition.
- Provides **context-aware responses** with follow-up question handling.
- Extracts entities such as **book names, authors, and user details**.

Technology Stack

- **Frontend:** Streamlit
- **Backend:** Python 3.8+
- **NLP Tools:** Hugging Face Transformers (NLU), spaCy (entity recognition)
- **Development Tools:** VS Code, Git, virtual environments, batch scripts for deployment


Implementation & Result

The chatbot system was developed in two forms, combining both simplicity and advanced capabilities. The simple implementation relied on an intent dictionary, pattern matching, a response generator, and a Streamlit-based chat interface to handle common queries such as book availability, library timings, fine policies, and membership information. Building on this foundation, the advanced implementation introduced natural language understanding (NLU), entity extraction, context management, and response templates. This allowed the chatbot to dynamically recognize intents, extract specific entities like book titles and authors, and maintain a smooth, context-aware conversation flow.


Implemented Features:

- Supported book availability checks, timing inquiries, fine calculations, membership details, and event updates.
- Provided a chat-like interface with conversation history and real-time responses.
- Included error handling, session management, and configurable responses.


Sample Output

 You: What are the lost_books?


🗨️ **LibraryBot:** Lost books must be reported immediately and replaced or paid for at the circulation desk.

 You: What are the membership?


🗨️ **LibraryBot:** Membership renewal is done annually at the help desk or online via the library portal.

 You: What are the damaged_books?

🗨️ **LibraryBot:** The fine for damaged books depends on severity, usually the replacement cost is charged.

 You: What are the fiction_section?

🗨️ **LibraryBot:** The fiction section is located on the second floor, near the east wing.

 You: What are the contact?

🗨️ **LibraryBot:** Contact the reference librarian for help with research and citations.

Conclusion and Future scope

The **Internship Application Assistant** chatbot successfully demonstrates the potential of combining rule-based and AI-driven NLP techniques to provide meaningful and context-aware career guidance. It offers students personalized support for resumes, cover letters, interview preparation, and application tracking.

Key Achievements:

- Successfully developed chatbot implementations: a rule-based version with an advanced NLU-powered version.
- Automated responses for common library queries, reducing staff workload.
- Delivered a user-friendly, scalable, and maintainable architecture.
- Demonstrated the feasibility of context-aware and interactive conversations in library services.

Future Enhancements:

- Integration with **real library databases** for live book availability.
- **Multi-language support** and **voice interface** for better accessibility.
- Advanced features like **user authentication, book reservation, fine payment, and real-time updates**.
- **Cloud deployment, mobile application integration, analytics dashboards, and API extensions** for broader usability and scalability.

6. References

- Streamlit Documentation: <https://docs.streamlit.io/>
- Hugging Face Transformers: <https://huggingface.co/docs/transformers/index>
- spaCy Documentation: <https://spacy.io/api/doc>
- Python Documentation: <https://docs.python.org/3/>
- Jurafsky, D., & Martin, J. H. (2023). *Speech and Language Processing*.
- Bird, S., Klein, E., & Loper, E. *Natural Language Processing with Python*.
- Raj, S. (2019). *Building Chatbots with Python*.
- Vajjala, S., et al. (2020). *Practical Natural Language Processing*.
- Tedd, L. A. *Digital Libraries: Principles and Practice in a Global Environment*.
- Smith, L. C. *Modern Library Technologies for Data Storage, Retrieval, and Use*.