InfoBot: An AI-Powered Academic Management System

1. Executive Summary

InfoBot is an AI-powered academic management system developed as a Semester 6 mini-project at Amal Jyothi College of Engineering. The system is designed to streamline student-faculty operations by leveraging intelligent automation. Its core feature is a chatbot integrated with Claude Sonnet, capable of answering academic queries, generating automated timetables, managing multi-batch scheduling, and tracking exams. The primary goal of InfoBot is to reduce manual administrative workload, eliminate scheduling conflicts, and provide students and faculty with instant, centralized access to academic resources through an intuitive web interface.

2. Introduction

In a traditional academic setting, administrative tasks often create significant operational bottlenecks. At Amal Jyothi College of Engineering, the process of compiling semester timetables was a manual, painstaking effort that consumed **two to three weeks** of staff time each cycle. This manual approach was not only inefficient but also prone to errors, leading to scheduling conflicts. Furthermore, tracking deadlines for syllabus completion and arranging necessary extra classes was a decentralized and challenging task. InfoBot was conceived as a direct solution to these problems—a smart tool designed to replace weeks of manual labor with intelligent automation, ensuring accuracy, efficiency, and clarity for the entire institution.

3. Project Objectives

- To develop an AI-powered chat assistant using the Claude Sonnet API to provide instant answers to academic queries regarding syllabi, course plans, and timetables.
- To design and implement an automated timetable generator that creates conflict-free class and faculty schedules for up to 8 semesters, drastically reducing generation time.
- To build a centralized system for managing and tracking test schedules, submission deadlines, and academic events, including the scheduling of extra classes for syllabus completion.
- To create a unified portal for easy access to essential resources like syllabi, study materials, and official notices.

- To reduce the manual workload for faculty and provide a seamless, user-friendly interface for both students and staff.

4. Project Scope

The project focuses on developing a web-based application for academic management within a single college environment. The scope includes an AI chatbot, an automated timetable generator for multiple batches, an exam and deadline tracker, and a resource portal. The system is designed to handle overlapping classes and labs for different student groups. Out of scope for this mini-project are features like student admissions, fee processing, grade management, and integration with external university-wide ERP systems.

5. Methodology

The project followed an agile development approach. The process began with identifying the core pain points in the existing academic system. This was followed by system design, where the architecture for the frontend, backend, database, and AI integration was planned. The development was executed in sprints, focusing on individual features like the database schema, backend API endpoints, timetable algorithm, and frontend UI. The Claude Sonnet API was then integrated to power the chatbot. Testing was conducted throughout the development cycle using Postman for the backend APIs and manual testing for the user interface to ensure functionality and usability.

6. Technology Stack

- **Frontend:** HTML, CSS, JavaScript
- **Backend:** Node.js, Express.js
    - **Justification:** Chosen for its non-blocking, asynchronous architecture, which is highly efficient for handling API requests and database I/O. The team's proficiency in JavaScript allowed for a unified language across the stack, accelerating development.
- **Database:** MySQL (managed via phpMyAdmin)
    - **Justification:** Selected for its reliability and structured nature. As the core data consisted of well-defined records for students, faculty, and courses, a relational database like MySQL ensured data integrity and ACID compliance.
- **AI Integration:** Claude Sonnet API
    - **Justification:** Specifically chosen for its advanced problem-solving and reasoning capabilities. The project required more than simple Q&A; it needed

an AI that could understand and resolve complex logical constraints, making it the ideal choice for the timetable generation module.

- **Development Tools:** Visual Studio Code, Postman, GitHub

7. Detailed Description

InfoBot is a comprehensive platform aimed at simplifying academic logistics. The workflow begins when a user interacts with the system through its web interface.

- **AI Chat Assistant:** Users can type natural language questions into the chat interface. The backend processes these queries and sends them to the Claude Sonnet API. The AI model interprets the query and returns a relevant answer based on the academic data, covering topics like syllabus details, faculty information, or exam dates.
- **Automated Timetable Generator:** This is the cornerstone of InfoBot. The system tackles the immense complexity of academic scheduling, which involves variables like semester type (even/odd), unique subject lists, specific faculty assignments, and teachers who may be shared across multiple batches. Instead of a traditional hard-coded algorithm, we pioneered a novel approach:
    - **Prompt Engineering:** We developed a highly structured and detailed prompt that encapsulates all scheduling constraints—faculty availability, subject requirements, classroom assignments, and batch allocations.
    - **AI-Powered Conflict Resolution:** This master prompt is sent to the Claude Sonnet API. The AI leverages its inherent problem-solving capabilities to process these constraints and generate a complete, conflict-free timetable. This method outsources the complex combinatorial logic to the LLM, resulting in a flexible and powerful scheduling solution.
- **Exam & Deadline Tracker:** Faculty can input schedules for tests, assignments, and other academic events. This information is stored in the database and displayed in a centralized calendar or list view, allowing students to stay informed about upcoming deadlines.
- **Resource Access Portal:** This feature functions as a digital repository where faculty can upload, and students can access, documents such as course syllabi, study materials, and official notices.

8. Resources Used

- **API & Documentation:** Claude Sonnet API documentation for AI integration.
- **Development Resources:** Official documentation for Node.js, Express.js, and MySQL.

- **Sample Data:** Anonymized academic data (e.g., course codes, faculty lists, semester structures) from Amal Jyothi College of Engineering was used for development and testing purposes.

9. Implementation Plan

1. **Phase 1: Foundation and Database Design:** Set up the Node.js/Express.js backend server and designed the MySQL database schema to store student, faculty, course, and schedule information.
2. **Phase 2: Backend Logic and API Development:** Created REST API endpoints for core functionalities like fetching course details, managing schedules, and structuring data for the AI.
3. **Phase 3: Prompt Engineering & AI Integration:** Developed and refined the master prompt for timetable generation and integrated the Claude Sonnet API.
4. **Phase 4: Frontend Development:** Built the client-side interface using HTML, CSS, and JavaScript to allow users to interact with the system's features.
5. **Phase 5: Testing and Deployment:** Conducted end-to-end testing of the application to identify and fix bugs, ensuring all components worked together seamlessly.

10. Results and Deliverables

- A fully functional prototype of the InfoBot web application.
- A successful integration of the Claude Sonnet AI, creating an intelligent chatbot and a powerful scheduling engine.
- A demonstrated ability to automatically generate complex, conflict-free timetables, **reducing a 2-3 week manual process to a matter of minutes.**
- A centralized dashboard for tracking academic events and accessing resources.
- The complete source code hosted on a GitHub repository.

11. Challenges and Risks

- **Technical Challenge:** The primary challenge was perfecting the prompt engineering for the timetable generator. It required numerous iterations to create a prompt that was detailed enough for the AI to understand all constraints without ambiguity.
- **Integration Challenge:** Integrating the third-party Claude Sonnet API required robust error handling to manage API downtime, rate limits, and unexpected response formats.

- **Data Management:** Structuring and managing the large volume of academic data in the MySQL database effectively was crucial for the system's performance.
- **Risk:** The project's core scheduling functionality is dependent on the availability and performance of the external Claude Sonnet API.

12. Lessons Learned

The InfoBot project provided invaluable experience in leveraging Large Language Models for complex logistical problem-solving, going beyond typical chatbot applications. Key takeaways include the power of advanced prompt engineering, the importance of selecting the right technology for the data structure (relational vs. non-relational), and the necessity of designing resilient systems that can handle external API dependencies. The project was a comprehensive learning experience that enhanced skills across multiple technology domains.

13. Recommendations

- **Mobile Application:** Develop a native mobile application (iOS/Android) for InfoBot to improve accessibility for students and faculty.
- **Personalization and Notifications:** Add features for personalized user dashboards and push notifications for upcoming deadlines or timetable changes.
- **Direct Calendar Integration:** Allow users to sync their academic schedules from InfoBot directly with personal calendars like Google Calendar or Outlook Calendar.
- **Expansion of AI Capabilities:** Train the AI model on a wider range of institutional data to handle more complex and personalized queries, such as providing tailored study resource recommendations.

14. Conclusion

InfoBot successfully demonstrates the transformative potential of integrating advanced AI to modernize academic management. By automating the highly complex and time-consuming task of scheduling and centralizing information, the system directly addresses critical institutional pain points. This project not only serves as a practical, high-impact solution but also pioneers a novel approach to logistical challenges by harnessing the reasoning power of modern LLMs.

15. References

- Official documentation for Node.js (nodejs.org)
- Official documentation for Express.js (expressjs.com)

- Official documentation for MySQL (mysql.com)
- API reference and documentation for the Claude Sonnet model.