Software Requirements Specification (SRS)

Project: College Chatbot Web Application

1. Introduction

This document describes the requirements for a College Chatbot Web Application developed using Rasa, GPT-2, Python Flask, and deployed on Render.

1.1 Purpose

The chatbot will assist students, faculty, and visitors by answering queries related to admissions, courses, fees, timetable, events, faculty, and placements.

1.2 Scope

The system provides 24/7 automated assistance via a web interface. Rasa handles intent recognition, GPT-2 generates dynamic responses, Flask manages backend operations, and Render hosts the deployed application.

2. Overall Description

The system consists of a frontend (HTML/CSS/JS), backend (Flask API), and AI engine (Rasa + GPT-2).

3. Functional Requirements

- User can type and receive chatbot responses in real-time.
- Rasa identifies user intent and extracts entities.
- Predefined responses for common queries.
- GPT-2 generates responses for unknown queries.
- API Endpoints: / (home), /chat (POST), /health (GET).

4. Non-Functional Requirements

- Response time under 5 seconds.
- Secure input handling and HTTPS support.
- Mobile responsive UI.
- System uptime target: 99%.
- Deployment on Render cloud platform.

5. Deployment Requirements

The application will include a requirements.txt file, Procfile, and optional Docker configuration for deployment on Render.

6. Conclusion

The College Chatbot Web Application integrates Rasa for NLU, GPT-2 for intelligent responses, Flask for backend services, and Render for cloud deployment, providing an efficient automated support system for colleges.