

Software Requirements Specification (SRS)

Project Name : AI Chatbot for Student Support Services

Author : Sanket Ravindrakumar Shrivastav

Date : 4 Feb 2026

1. Introduction

1.1 Purpose

The purpose of this document is to describe the Software Requirements Specification (SRS) for the **AI Chatbot for Student Support Services**. This chatbot is designed to provide automated responses to common student queries related to academic and administrative services using predefined FAQs.

1.2 Scope

The Student Support Chatbot aims to assist students by:

- Answering frequently asked questions
- Reducing workload on administrative staff
- Providing instant and consistent responses
- Offering 24/7 support through a command-line interface

The chatbot uses basic Natural Language Processing (NLP) techniques such as text preprocessing and keyword matching.

1.3 Definitions, Acronyms, and Abbreviations

Term Description

AI Artificial Intelligence

NLP Natural Language Processing

FAQ Frequently Asked Questions

SRS Software Requirements Specification

1.4 References

- Python Official Documentation
 - Software Engineering IEEE SRS Standards
 - Academic Guidelines for Student Support Systems
-

2. Overall Description

2.1 Product Perspective

The chatbot is a standalone, console-based application developed using Python. It interacts with users through text input and provides responses based on a predefined knowledge base stored in a JSON file.

2.2 Product Functions

- Accept user queries via text input
 - Process user input using text preprocessing
 - Match keywords with stored FAQs
 - Provide appropriate responses
 - Allow users to exit the system
-

2.3 User Classes and Characteristics

User Type	Description
-----------	-------------

Student	Queries related to academics and administration
---------	---

Administrator	Maintains FAQ data
---------------	--------------------

2.4 Operating Environment

- Operating System: Windows / Linux / macOS
 - Programming Language: Python 3.x
 - Interface: Command Line Interface (CLI)
-

2.5 Design and Implementation Constraints

- Internet connection is not required
 - The chatbot relies on predefined FAQs
 - Responses are limited to available data
 - No database connectivity in current version
-

2.6 Assumptions and Dependencies

- Users provide queries in English
 - Python environment is properly installed
 - FAQ data is regularly updated by administrators
-

3. System Requirements

3.1 Functional Requirements

ID Requirement

FR1 The system shall accept user input via text

FR2 The system shall preprocess user input

FR3 The system shall match keywords with FAQs

FR4 The system shall display appropriate responses

FR5 The system shall handle unknown queries gracefully

FR6 The system shall allow the user to exit

3.2 Non-Functional Requirements

3.2.1 Performance Requirements

- The system should respond within 1 second
- The chatbot should handle continuous queries

3.2.2 Reliability

- The system should not crash for invalid input
- Proper error handling must be implemented

3.2.3 Usability

- Simple and user-friendly interface
- Easy to understand responses

3.2.4 Portability

- The system should run on any OS with Python installed

3.2.5 Security

- No sensitive user data is stored
 - Local file access only
-

4. External Interface Requirements

4.1 User Interface

- Command-line based interaction
 - Text input and output
-

4.2 Hardware Interface

- Standard keyboard and display
 - No additional hardware required
-

4.3 Software Interface

- Python Standard Library
 - JSON file for FAQ storage
-

5. System Architecture

5.1 Architectural Design

The system follows a **modular architecture**:

- Input Handling Module
 - Text Preprocessing Module
 - Keyword Matching Module
 - Response Generation Module
-

5.2 Data Flow Diagram (Textual)

1. User enters query
 2. Input is preprocessed
 3. Keywords are extracted
 4. Matching FAQ is found
 5. Response is displayed
-

6. Future Enhancements

- Integration with web interface
 - Machine learning-based intent classification
 - Database support
 - Voice-based chatbot
-

7. Conclusion

The AI Chatbot for Student Support Services provides an efficient and user-friendly solution for addressing common student queries. The system is scalable and can be enhanced with advanced AI techniques in future versions.