

Software Requirements Specification

for

KBot for KIITians

Version 1.0 approved

Prepared by:

Nitya Singh

Aditya Meena

Sheetal Mohanty

Vartika

KIIT University

20-04-2023

Table of Contents

Table of Contents

Revision History

1. Introduction

- 1.1 Purpose
- 1.2 Document Conventions
- 1.3 Intended Audience and Reading Suggestions
- 1.4 Product Scope
- 1.5 References

2. Overall Description

- 2.1 Product Perspective
- 2.2 Product Functions
- 2.3 User Classes and Characteristics
- 2.4 Operating Environment
- 2.5 Design and Implementation Constraints
- 2.6 User Documentation
- 2.7 Assumptions and Dependencies

3. External Interface Requirements

- 3.1 User Interfaces
- 3.2 Hardware Interfaces
- 3.3 Software Interfaces
- 3.4 Communications Interfaces

4. Other Nonfunctional Requirements

- 4.1 API

4.1.1 Accuracy

4.1.2 Fast Response

4.1.3 Security

4.2 Web interface

Activity Diagram

Appendix A: Glossary

Revision History

Name	Date	Reason For Changes	Version
Query solver	30-01-23	Creating a prototype and a data pipeline for a model which provides generic response. Obtain OpenAI API key.	1.0

Hostel Boarder helper	06-02-23	Using a smaller dataset, a document of 5 pages, of KIIT hostel's rules and regulations. It responds to the queries of hostel boarders.	2.0
KIITBOT	14-03-23	Expanding the database, document size, to 55 pages. Using a KIITEE brochure. Targeting students aspiring to give the KIITEE exam.	3.0
KIITBOT	20-03-23	Transferring the code from jupyter notebook to a Python File. Hence enabling to run the program at once or to create an API	3.1
KBOT For KIITIANS	25-03-23	Creating a basic Streamlit Front end. Features of the page contain an input box and a response text box.	4.0
KBOT for KIITIANS	05-04-23	Change in CSS elements of the page. Alteration in text, Background, primary and secondary color	4.1
KBOT for KIITIANS	11-04-23	Addition of Sidebar to the web page	4.2
KBOT for KIITIANS	15-04-23	Addition of About and contact tab to the page.	4.3
KBOT for KIITIANS	20-04-23	Deployment ready.	5.0

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the KBot for KIITians, an AI interactive agent. It will explain the purpose and features of the project, its working, what the KBot is capable of and the constraints under which it must operate. This document is intended for the users of this AI interactive agent and also potential developers.

1.2 Document Conventions

This Document was created based on the IEEE template for System Requirement Specification Documents.

1.3 Intended Audience and Reading Suggestions

- *Typical users, students and staff members who want to inquire about the traditional protocols followed across the University.*
- *People who are not yet associated with KIIT but want to learn more about it.*
- *Programmers who are interested in working on the project by further developing it or fixing existing bugs.*

1.4 Product Scope

The KBot for KIITians is an AI interactive agent that receives queries from users, tries to understand the question, and provides appropriate answers. It does this by converting an English sentence into a machine-friendly query, then going through relevant data to find the necessary information, and finally returning the answer in a natural language sentence. In other words, it answers your questions like a human does, instead of giving you the list of websites that may contain the answer. For example, when it receives the question "What time does the gym close today?", it will give a response "The gym closes at 10pm today."

The main objective is creating Web API, sample web, mobile, and text messaging interfaces that demonstrate the use of the API. The goal is to provide the students and faculty of KIIT a

quick and easy way to have their questions answered, as well as to offer other developers the means to incorporate the KBot for KIITians into their projects.

1.5 References

1. Open Ai <https://platform.openai.com/docs/introduction>
2. LangChain <https://docs.langchain.com/docs/>
3. PyPDF <https://pypdf2.readthedocs.io/en/3.0.0/>
4. Streamlit <https://docs.streamlit.io/>
5. KIIT <https://kiit.ac.in/>
6. KIITEEE <https://kiitee.kiit.ac.in/>

2. Overall Description

2.1 Product Perspective

This Software is developed to act as a one-stop destination for solving all the queries of the target audience. It provides faster troubleshooting or clearing queries as the user doesn't need to search for any brochure, documentation or FAQs page. This is made possible using the state of the art Machine Learning Large Language Model (LLM) of OpenAI, GPT da-vinci 002, which simulates the learning of a document by an AI and provides response to the query based on the knowledge base.

2.2 Product Function

Window

- **Home** - Main homepage of the software. Introduces the user to the software with graphics(images) and textual messages. This page contains the main query input content as well as the reply box.
 1. Query Input - *Takes the query as a text input from the user and sends it to the backend code.*
 2. Response output - *Returns the processed response to the user's query.*
- **About** - Introduces the user to the University. Contains infographics and information about the college.

- **Contact us** - Holds the official support channels of the college.
 1. *Link to the official website for KIITEE*
 2. *E-mail Address of the Admission Department*
 3. *Contact Number of the Admission Department*

Sidebar

Introduces the user to the software and describes its capabilities. Contains three Hyperlink Buttons.

- Homepage Button - Redirects the user to the official KIIT homepage.
- Admissions Button - Redirects the user to the website of the Admission Department of KIIT.
- Academics button - Redirects the user to the Academics Page of KIIT University.

2.3 User Classes and Characteristics

- *Typical users, students and staff members who want to inquire about the traditional protocols followed across the University.*
- *People who are not yet associated with the institution but wish to learn more about it.*
- *Programmers who are interested in working on the project by further developing it or fixing existing bugs.*

2.4 Operating Environment

- Operating System : Windows 7, 8, 8.1, 10, 11, Ubuntu, Mac Os.
- Python
- Environment : Conda environment (Anaconda)

2.5 Design and Implementation Constraints

- The Software, the Frontend User Interface, and even the Backend have been implemented using Python and its libraries or solely functional on python environments. Hence, running on the local machine requires the user to fulfill the version and dependencies of python and the used libraries.
- **Limited Question Scope-** *Creating a chatbot able to answer every single question about KIIT University isn't practically possible to implement with current technology and resources, and within the duration of the project, so the system will be able to answer questions about limited topics.*
- **Language-** *The system will only support questions in standard English.*

2.6 User Documentation

- Input your query as a text in the dialog box available on the homepage. Wait for the Response to be available below the dialog box.
- The response time varies proportionally to the input query length and the grammatical complexity.
- Use the tabs at the upper margin of the page for navigating from one subpage to another.
- Access the sidebar to get introduction of the software along with the hyperlinks to various official KIIT web pages.

2.7 Assumptions and Dependencies


- User has a Python environment : 3.9.x, 3.10.x, 3.11.x
- The Conda environment Python is used .
- Following libraries are installed in the environment :
 - Langchain
 - PyPDF2
 - Os
 - Streamlit

- User has an OpenAI API key to use the LLM of OpenAI of one of the following type
 - Da-vinci - 002
 - Da-Vinci- 003
 - GPT 3.5 turbo
 - GPT 4

3. External Interface Requirements

3.1 User Interfaces


Home page



Welcome to the KIIT University chatbot! Our bot is designed to help you quickly and easily find the information you need about the university. Whether you're a student, faculty member, or staff member, our bot is here to assist you with a wide range of tasks, from course selection to IT support. Our bot is built using Large language model, which is a popular and sophisticated machine learning model, providing you with an interactive and user-friendly experience. We hope our chatbot can help make your experience at KIIT University as smooth and successful as possible.

Home About Contact us

KIITEE 2023



KIITEE query solver


enter your query

what are the subject weightage

For B.Tech.(4 years)/ B.Arch/B.Sc.Nursing/Biotechnology Dual Degree(5years)/B.Sc. Computer Science, the subject weightage is Physics (40 questions), Chemistry (40 questions), and Mathematics (40

About Page :

Home About Contact us



About KIIT

KIIT is a private academic institution dedicated to providing educational opportunities for the intellectual, social, and professional development of a diverse student population. To achieve this purpose, the

Contact Us :

Contact us



Website :

<https://www.kiitee.com/>

Email :

admission@kiit.ac.in

Phone :

+918080735735

Sample Query Response Example :

KIITEE query solver

enter your query

what are the subject weightage

For B.Tech.(4 years)/ B.Arch/B.Sc.Nursing/Biotechnology Dual Degree(5years)/B.Sc. Computer Science, the subject weightage is Physics (40 questions), Chemistry (40 questions), and Mathematics (40 questions). For B.Tech.(Lateral Entry) the subject weightage is Mathematics (40 questions). For MCA the subject weightage is Computer Awareness (30 questions), Mathematics (30 questions), and age preference (Preference to older candidate). For M.Tech the subject weightage is Branch Specific (120 questions) and preference to older candidates. For M.Sc. (Applied Microbiology/Biotechnology) the subject weightage is Biology (50 questions), Chemistry (30 questions), Mathematics (20 questions), and Physics (20 questions). For LL.M the subject weightage is Multiple Choice questions (120 questions) and preference to older candidates. For M.Com/M.A in Economics/M.A in English/M.A in Sociology/M.Sc in Computer Science/Master of Mass communication the subject weightage is preference to older candidates. For M.Sc. (2 years) (Physics/Chemistry/Mathematics and Data Sciences) the subject

3.2 Hardware Interfaces

The user must have more than 500 MHz CPU and more than 500MB of available RAM for running the software. The secondary storage must have necessary capacity to store the

Document to be parsed. The browser must also be up to date and compatible with the streamlit interface.

4. Other Nonfunctional Requirements

4.1 API

1. Accuracy

The overall accuracy of the Program's response will be measured using a developer-made testing set. The overall accuracy is calculated by dividing the total number of correct answers by the number of questions asked..

The Accuracy bars for certain supported topics will be as follows:

- *KIIT facilities' locations and schedules will have accuracy greater than 90%.*
- *KIIT staff's office locations, contact information, and positions will have accuracy greater than 90%.*
- *KIIT policies including Academics, Admissions etc. will have accuracy greater than 90%.*
- *Queries regarding the subjects, venues, and dates regarding KIITEE will have accuracy greater than 90%.*

2. Fast Response

The average time for the server to respond, over the question testing set, will be less than or equal to 3 seconds due to resource constraints. Considering the fact that the response time is directly proportional to the query length and depth, the average response time is a rough estimation and the actual results may vary.

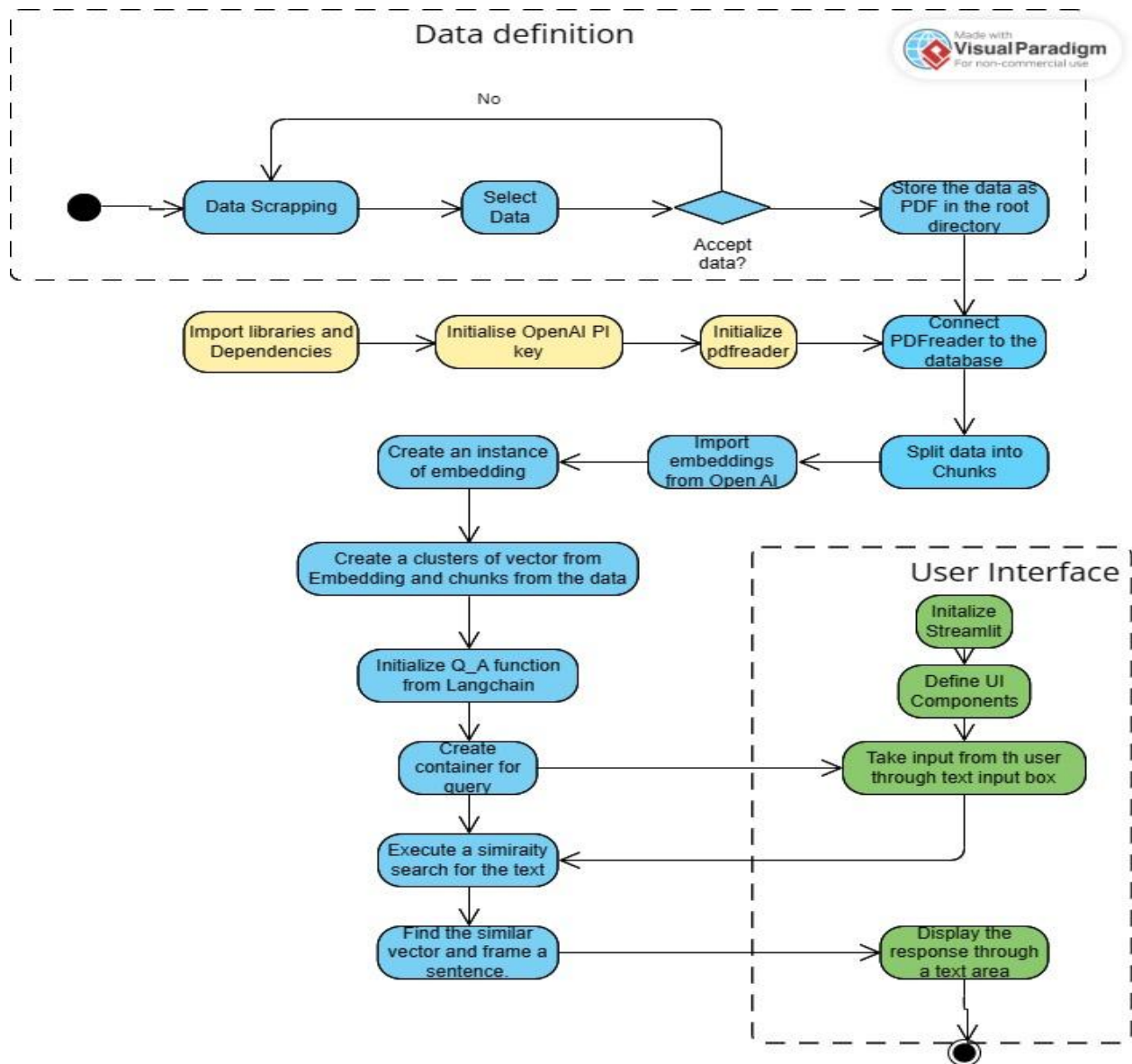
3. Security

The connection between the Web API and the programs will use HTTPS, for security.

4.2 Web Interface / Mobile Application

Ease of Use: A new user will make less than 3 mistakes within the first 5 minutes of use.

Activity Diagram



Glossary:

Few of the technical terms used throughout the project:

1. **LangChain:** A framework used for developing applications powered by language models.

2. **Streamlit:** *A framework used for developing quick web applications.*
3. **PyPDF2:** *An open-source pure-python PDF library capable of splitting, merging, cropping, and transforming the pages of a PDF file.*
4. **text-davinci-002:** *An InstructGPT model based on code-davinci-002.*
5. **text-davinci-003:** *An upgrade over the traditional text-davinci-002, capable of responding concisely and more accurately without the need for any examples given in the prompt.*
6. **GPT 3.5 Turbo:** *A neural network-based language model that has been trained on a massive amount of data, and is capable of performing a wide range of language tasks, including language translation, text completion, and question answering.*
7. **GPT 4:** *An upgrade over the GPT 3.5 Turbo model but can only be differentiated when the task complexity reaches a sufficient threshold. More reliable, creative, and able to handle much more nuanced instructions than GPT 3.5.*

