**Software Requirements**

**Specification**

**for**

**<SZABIST AI Chatbot>**

**Version 1.0 approved**

**Prepared by <author>**

**<SZABIST> <date created>**

**Copyright © 1999 by Karl E. Wiegers. Permission is granted to use, modify, and distribute this document.**

**Software Requirements Specification for <Project> Page ii**

# Table of Contents

[Table of Contents 2](#_Toc57275226)

[Revision History 3](#_Toc57275227)

[1. Introduction 1](#_Toc57275228)

[1.1 Purpose 1](#_Toc57275229)

[1.2 Document Conventions 1](#_Toc57275230)

[1.3 Intended Audience and Reading Suggestions 1](#_Toc57275231)

[1.4 Product Scope 1](#_Toc57275232)

[1.5 References 2](#_Toc57275233)

[2. Overall Description 2](#_Toc57275234)

[2.1 Product Perspective 2](#_Toc57275235)

[2.2 Product Functions 2](#_Toc57275236)

[2.3 User Classes and Characteristics 2](#_Toc57275237)

[2.4 Operating Environment 2](#_Toc57275238)

[2.5 Design and Implementation Constraints 3](#_Toc57275239)

[2.6 User Documentation 3](#_Toc57275240)

[2.7 Assumptions and Dependencies 3](#_Toc57275241)

[3. External Interface Requirements 4](#_Toc57275242)

[3.1 User Interfaces 4](#_Toc57275243)

[3.3 Software Interfaces 4](#_Toc57275244)

[3.4 Communications Interfaces 4](#_Toc57275245)

[4. System Features 4](#_Toc57275246)

[4.1 System Feature 1 4](#_Toc57275247)

[4.2 System Feature 2 5](#_Toc57275248)

[4.3 System Feature 3 6](#_Toc57275249)

[4.4 System Feature 4 7](#_Toc57275250)

[4.5 System Feature 5 8](#_Toc57275251)

[4.6 System Feature 6 9](#_Toc57275252)

[5. Other Nonfunctional Requirements 10](#_Toc57275253)

[5.1 Performance Requirements 10](#_Toc57275254)

[5.2 Safety Requirements 10](#_Toc57275255)

[5.3 Security Requirements 10](#_Toc57275256)

[5.4 Software Quality Attributes 10](#_Toc57275257)

[5.5 Business Rules 11](#_Toc57275258)

[Appendix A: Glossary 11](#_Toc57275259)

[Appendix B: Analysis Models 11](#_Toc57275260)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# 1. Introduction

## 1.1 Purpose

The purpose of this SRS to provide a detail of our product, gather the information and analyze the information of and preview the insight of the AI Chatbot. The main objective of our project is building an AI Chatbot to solve the real-life use case, our main focus is built a Chatbot with tenser flow that they solved the issue of students and faculty in zabdesk. We use this use case to solve the situation of academic to provide solution in pandemic situation.

## 1.2 Document Conventions

* Times New Roman, font 18 is used for Main headings
* Times New Roman, font 14 is used for sub-headings
* Times New Roman, font 12 is used for sub-sub headings
* Times New Roman, font 12 is used for the paragraphs and writing of the entire document

## 1.3 Intended Audience and Reading Suggestions

**This SRS documentation is intended for:**

* Developers who can review our product because they easily understand our goals and put own efforts to implement more feature in our Chatbot and scale our product to run reliable on users’ systems
* The tester can use this document to focus on the main features and base of our product and fixes the bug easier because single bug challenges our Chatbot.
* Front-End Developer read this document to learn and understand our Back-End to create a better user interface to make life easier for our student and faculty members.

## 1.4 Product Scope

* Chat-bot will help the user with his/her admission related queries
* Chat-bot will help the students get a class schedule for his/her day.
* Chat-bot will help the student get his/her financial outstanding and help them with their fee related quires.
* Chat-bot will help the students schedule a meeting with a professor or a SZABIST official body by letting them know their free slots.
* Chat-bot will help the student schedule an appointment with the SZABIST therapist.
* Chat-bot will help the pass out students with their degree clearance procedure
* Chat-bot will help the students get a record of the current books they’ve issued from the library.
* Our chat-bot will also help the faculty members get their class schedule and will also help them reschedule their classes (if required).

## 1.5 References

1. <https://flask.palletsprojects.com/en/1.1.x/>
2. <https://www.python.org/>
3. <https://www.tensorflow.org/learn>
4. <https://www.w3schools.com/html/default.asp>
5. https://heartbeat.fritz.ai/learn-a-conversational-chatbot-with-nltk-and-tensorflow

# 2. Overall Description

Our main idea is basically to work on Chatbot Technology

* To explore AI, machine learning and text processing
* To solve the real-world use case

## 2.1 Product Perspective

We create an AI chatbot for zabdesk with a custom functionality so developer enhance the product to next level, add a path deep layer to get a better response in user end

## 2.2 Product Functions

The product functions are summarized below as:

## 2.3 User Classes and Characteristics

* Developer, Users and Organization
* All features delivered by Developer.
* User and Organization use this product implement own use cases.

## 2.4 Operating Environment

There are following tools need to operate or work in our Back-End product and Frontend.

* Linux, Mac OS or Windows
* Visual Studio Code
* Python
* Tenser flow
* NLP
* Flask
* HTML
* CSS
* JavaScript
* jQuery
* AJAX

## 2.5 Design and Implementation Constraints

**Design Constraint:**

We are using Visual Studio Code to implement the Smart Contract to set functionality of our blockchain framework and to work according to our main objective of this project.

**Implementation Constraint:**

We will be using JavaScript and Solidity as our programming language.

**Hardware constraints:**

Laptop or PC

**Cultural constraints:**

English

## 2.6 User Documentation

* SRS
* SDS
* Test Cases
* Use Cases

## 2.7 Assumptions and Dependencies

**Assumption:**

* For connection Ethernet will be required.
* Electricity for charging
* Ethereum Coins

**Dependencies:**

* Tenser flow
* Bootstrap
* Word processing

# 3. External Interface Requirements

## 3.1 User Interfaces

The user will be using the website will be able to view the home page where the different options will appear. The organization and user or developer who wants details and download our Chatbot and check the demo as well how chatbot

**3.2 Hardware Interfaces**

* Since the application must run over the internet, all the hardware shall require to connect internet will be the hardware interface for the system. As for e.g. Modem, WAN – LAN, Ethernet Cross-Cable.
* The system has a direct connection with the network and with software Visual Studio Code.

## 3.3 Software Interfaces

* Linux, Mac OS or Windows
* Visual Studio Code
* Python
* Tenser flow
* NLP
* Flask
* HTML
* CSS
* JavaScript
* jQuery
* AJAX

## 3.4 Communications Interfaces

The chat bot uses local server to perform functionality and for the intranet communication will be through TCP/IP protocol suite

# 4. System Features

## 4.1 System Feature 1

Zabdesk web UI:

Description and Priority

* Student and faculty interact with web
* It is a web user interface
* Priority is high for this functionality

Stimulus/Response Sequences

* authentication
* view data
* chatbot

Functional Requirements

REQ 1:

Actor:

User.

Feature:

Zabdesk web UI.

Pre-condition:

Zabdesk web UI must work on a all browser

Action:

Show how to work zabdesk and how user interect

Software reaction:

interaction Complete.

REQ 2:

Alternate scenarios:

User can’t interact with web

Post conditions:

## 4.2 System Feature 2

Chatbot UI:

Description and Priority

* Student and faculty interact with chat bot
* It is a chat bot user interface
* Priority is high for this functionality

Stimulus/Response Sequences

* authentication
* view data
* chatbot

Functional Requirements

REQ 1:

Actor:

User.

Feature:

Chatbot UI.

Pre-condition:

Chatbot UI must work on an all browser

Action:

Show how to work chat boy and how user interact

Software reaction:

interaction Complete.

REQ 2:

Alternate scenarios:

User can’t interact with chatbot because of irrelevant questions

Post conditions:

## 4.3 System Feature 3

Chatbot Json data:

Description and Priority

* chat bot give response for relevant questions
* It is a chat bot response data
* Priority is high for this functionality

Stimulus/Response Sequences

* authentication
* view data
* chatbot

Functional Requirements

REQ 1:

Actor:

Developer.

Feature:

Chatbot Json data.

Pre-condition:

Chatbot Json data must have valid academic related responses

Action:

How data become a model to train with

Software reaction:

interaction Complete.

REQ 2:

Alternate scenarios:

User can’t get response

Post conditions:

## 4.4 System Feature 4

Chatbot Model Preparation:

Description and Priority

* prepare a model through machine learning algorithm
* Priority is high for this functionality

Stimulus/Response Sequences

* authentication
* view data
* chatbot

Functional Requirements

REQ 1:

Actor:

User.

Feature:

Chatbot Model Preparation

Pre-condition:

Chatbot Model must be prepared

Action:

Data modeling

Software reaction:

interaction Complete.

REQ 2:

Alternate scenarios:

Model is not good enough

Post conditions:

## 4.5 System Feature 5

Chatbot Training:

Description and Priority

* chat bot data pass through tenser flow to train
* Priority is high for this functionality

Stimulus/Response Sequences

* Chat bot get data
* Become model
* And then train the responses

Functional Requirements

REQ 1:

Actor:

System.

Feature:

Chatbot Training.

Pre-condition:

Model must be prepared for training

Action:

Train by tenser flow

Software reaction:

interaction Complete.

REQ 2:

Alternate scenarios:

Training not working

Post conditions:

## 4.6 System Feature 6

Chatbot Implementation with UI:

Description and Priority

* Chat bot implement with UI through Ajax
* Priority is high for this functionality

Stimulus/Response Sequences

* authentication
* view data
* chatbot

Functional Requirements

REQ 1:

Actor:

User.

Feature:

Chatbot Implementation with UI.

Pre-condition:

Chatbot Implementation with UI data must come

Action:

Data get from backend and send to through route on frontend

Software reaction:

interaction Complete.

REQ 2:

Alternate scenarios:

Route not working

Post conditions:

# 5. Other Nonfunctional Requirements

## 5.1 Performance Requirements

* Search feature: the search feature should be prominent and easy to find for the user
* Response time: the quickness of the search feature will indicate about the response time of the website
* Verification Notification: the user should be able to receive an e-mail notification for verification immediately

## 5.2 Safety Requirements

All unit test must be passed because little bugs effect in the framework and challenge his immutability.

## 5.3 Security Requirements

* The only academic zabdesk user can interact with authentic accounts.
* And Roots are protected.
* Add library to protect web attacks

## 5.4 Software Quality Attributes

* Easy to use.
* Intermediate Knowledge about AI and word processing Easy to operate.
* Simple English language easy for the user to understand its terms and conditions.
* Linux, Mac OS or Windows 10.
* Visual Studio Code Anaconda

## 5.5 Business Rules

NA

# Appendix A: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

# Use case diagram

