**Software Requirements Specification**

***for***

***Machine Learning in Diabetes***

***Version 0.01***

***Prepared by***

***Donna R***

Table of Contents

1. OBJECTIVES AND SCOPE
2. INTENDED AUDIENCE
3. DEFINITIONS, ACRONYMS AND ABBREVIATIONS
4. REFERENCES
5. REQUIREMENTS OVERVIEW
   1. Project Perspective
   2. Project functions
   3. Operating Environment
   4. Customer enforced technology choices
   5. User Documentation
   6. Assumptions ,dependencies and external risks
6. EXTERNAL INTERFACE REQUIREMENTS
   1. User Interfaces
   2. Hardware interfaces
   3. Software interfaces
   4. Communication interfaces
7. FUNCTIONAL REQUIREMENTS
   1. Graphical User Interface
   2. Encoder/Decoder
   3. Communication module
8. PERFORMANCE REQUIREMENTS
9. ACCEPTANCE REQUIREMENTS
10. MAINTAINABILITY REQUIREMENTS
11. OTHER REQUIREMENTS
    1. Portability Requirements
    2. Reliability requirements
    3. Scalability requirements
12. OPEN ISSUES

***1. Objectives and Scope***

The purpose of this Software Requirements Specification (SRS) document is to outline the requirements and specifications for a Diabetes Prediction System. The system will utilize Machine Learning (ML) algorithms to predict the likelihood of a person developing diabetes based on blood pressure, and glucose levels.

***2. Intended Audience***

This document is intended for the development team, stakeholders, and anyone involved in the design, development, and testing of the Diabetes Prediction System.

***3. Definitions, Acronyms, and Abbreviations***

SRS: Software Requirements Specification

ML: Machine Learning

***4. References***

Hospital documents(patient data)

***5. Requirements Overview***

**5.1 Project Perspective**

The Diabetes Prediction System is a standalone web-based application that aims to assist healthcare professionals in early identification and prediction of diabetes in patients. It will provide a user-friendly interface for entering patient information and generate accurate predictions using ML algorithms.

**5.2 Project Functions**

The key functions of the Diabetes Prediction System include:

* *User Management*: Allow healthcare professionals to register, log in, and update their profile information.
* *Patient Management*: Enable healthcare professionals to add, edit, and delete patient records, including personal details and medical data.
* *Data Preprocessing*: Clean, normalize, and handle missing values in the input data.
* ML Model Training: Implement and evaluate ML algorithms to train a diabetes prediction model using a training dataset.
* *Diabetes Prediction*: Provide prediction functionality to determine the likelihood of a patient developing diabetes based on input data.
* *Reporting and Visualization*: Generate reports and visualizations to present prediction results to healthcare professionals.
* *System Administration*: Include an administrative interface for managing user roles and access permissions.

**5.3 Operating Environment**

The Diabetes Prediction System will operate in a web-based environment, accessible through standard web browsers.

**5.4 Customer Enforced Technology Choices**

The system shall be developed using Python for ML algorithm implementation and web development frameworks like Django or Flask.

The ML algorithms used for training the diabetes prediction model shall be based on well-established and validated methodologies.

**5.5 User Documentation**

The Diabetes Prediction System shall provide user documentation, including user guides and tutorials, to assist healthcare professionals in utilizing the system effectively.

**5.6 Assumptions, Dependencies, and External Risks**

The system assumes that healthcare professionals have the necessary knowledge and expertise to interpret and utilize the prediction results.

The availability and quality of relevant patient data are dependencies for accurate predictions.

The ML algorithms used depend on the quality and representativeness of the training dataset.

***6. External Interface Requirements***

**6.1 User Interfaces**

The user interface of the Diabetes Prediction System shall be intuitive, user-friendly, and accessible to healthcare professionals. It shall include screens for user registration, login, patient management, data input, prediction results, and report generation.

**6.2 Hardware Interfaces**

The Diabetes Prediction System does not have any specific hardware interface requirements. It should be accessible through standard computing devices such as desktop computers, laptops, or tablets.

**6.3 Software Interfaces**

The system shall integrate with the following software interfaces:

* Web Browsers: The system shall be compatible with popular web browsers such as Chrome.
* ML Libraries: The system shall utilize ML libraries like scikit-learn for implementing ML algorithms.

**6.4 Communication Interfaces**

The Diabetes Prediction System does not require any specific communication interfaces.

***7. Functional Requirements***

**7.1 Graphical User Interface**

The Graphical User Interface (GUI) of the Diabetes Prediction System shall provide the following functionalities:

* *User Registration*:
  + Allow healthcare professionals to register by providing their name, email address, and password.
  + Validate and store the user registration details in the system.
* *User Login*:
  + Provide a login screen for healthcare professionals to enter their credentials.
  + Authenticate user credentials and grant access to the system.
* *User Profile Management*:
  + Allow healthcare professionals to update their profile information, including name, email address, and password.
* *Patient Management*:
  + Provide screens for healthcare professionals to add, edit, and delete patient records.
  + Capture personal details such as name, age, gender, and contact information for each patient.
* *Data Input*:
  + Enable healthcare professionals to enter medical data for patients, including blood pressure, and glucose levels.
  + Perform data validation and handle missing values.
* *Prediction Results*:
  + Display the prediction outcome for each patient, indicating the probability or a binary classification (e.g., "diabetic" or "non-diabetic").
* *Reports and Visualization*:
  + Generate reports and visualizations to present prediction results, including patient details, input data, and the prediction outcome.

**7.2 Encoder/Decoder**

The Encoder/Decoder module shall handle data preprocessing and encoding/decoding of patient information. It shall perform the following tasks:

* *Data Preprocessing:*
  + Clean and normalize input data.
  + Handle missing values appropriately.
  + Ensure data compatibility with ML algorithms.
* *Feature Encoding:*

Encode categorical features, such as gender, using appropriate encoding techniques (e.g., one-hot encoding).

* *Feature Scaling:*

Scale numerical features, such as blood pressure, and glucose levels, to ensure consistency and compatibility with ML algorithms.

* *Feature Decoding:*

Decode the encoded features for presentation in reports and visualizations.

**7.3 Communication Module**

The Communication Module shall facilitate communication between the user interface and the ML model. It shall perform the following tasks:

* *Input Data Validation*:
  + Validate input data for completeness, correctness, and compatibility with ML model requirements.
  + Display error messages for invalid or missing data.
* *ML Model Integration*:
  + Pass the validated input data to the ML model for prediction.
  + Receive prediction results from the ML model.
* *Prediction Result Presentation*:

Return the prediction results to the user interface for display to healthcare professionals.



***8. Performance Requirements***

The Diabetes Prediction System shall provide real-time or near real-time prediction results.

The prediction process shall not exceed a maximum response time of 3 seconds.

***9. Acceptance Requirements***

The acceptance criteria for the Diabetes Prediction System shall include the following:

The system should accurately predict the likelihood of a patient developing diabetes based on input data.

The system should generate reports and visualizations that effectively present prediction results to healthcare professionals.

The user interface should be intuitive, user-friendly, and responsive.

The system should meet all the functional and non-functional requirements specified in this document.

***10. Maintainability Requirements***

The Diabetes Prediction System should be designed and developed in a modular and maintainable manner to facilitate future enhancements, bug fixes, and

updates. The codebase should be well-documented, and appropriate version control practices should be followed.

***11. Other Requirements***

**11.1 Portability Requirements**

The Diabetes Prediction System should be platform-independent and compatible with various operating systems, including Windows, macOS, and Linux.

The system should be deployable in different environments, such as local servers or cloud platforms.

**11.2 Reliability Requirements**

The system should be available and operational 24/7 with minimal downtime.

Backup and recovery mechanisms should be implemented to prevent data loss in case of system failures.

**11.3 Scalability Requirements**

The system should be designed to handle a large volume of patient records and user traffic.

The system architecture should support horizontal scaling to accommodate future growth.

***12. Open Issues***

This concludes the modified version of the Software Requirements Specification (SRS) document for the Diabetes Prediction System. The document covers the functional and non-functional requirements of the system, along with its constraints, assumptions, and dependencies.