



SOFTWARE DOCUMENT SPECIFICATION

TWO SEMESTER PROJECT



NAME: SANTOS OMONDI OKELLO

REG: 19/05036

SUPERVISOR: MR COLLINS ONDIEK

UNIT: TWO SEMESTER PROJECT

COURSE: BSC.IT

OCTOBER 24, 2022

DISEASE PREDICTION SYSTEM
KCA UNIVERSITY

DISEASE PREDICTION SYSTEM



INTRODUCTION

Author Note

Data collection and also preliminary research and analysis was conducted and also sponsored by the school of health and science in JKUAT and Faculty of computing and information technology in KCA University respectively. The portion of these information and finding was presented as a poster in 2022 online in Virtual campus KCA University supervised by Mr Collins Ondiek. Response action concerning this article should be addressed to Dean of school of technology Dr Mwendia, Madam Merab Omondi, and also the support and managing staff in KCA University.

First M. Last www.santostimelapse.

DISEASE PREDICTION SYSTEM

Purpose

The primary goal is to develop a prediction engine which will allow the users to check whether they have diseases like malaria, tuberculosis, typhoid, diabetes or heart disease et cetera sitting at home when feeling sick. The user don't need visit the doctor unless he or she has a strong disease that required physical checkup, for further treatment. The prediction engine requires a large dataset and efficient machine learning algorithms to predict the presence of the disease. Pre-processing the dataset to train the machine learning models, removing redundant, null, or invalid data for optimal performance of the prediction engine

Scope

The software Design document would demonstrate how the design will accomplish the functional and non- functional requirement captured in the software requirement specification (SRS) .The document will provide a framework to the programmers through describing the high level components and architecture, sub systems, interface, database design and algorithm design. This is achieved through the use of architectural patterns, design patterns, sequence diagrams, class diagrams, relational models and user interface

DISEASE PREDICTION SYSTEM

Intended audience

This document is mainly for the developers, technical ,garage administrative staff of Blizz hospital and customers /patients of blizz health care system

Glossary

Algorithm design	Specific method to create a mathematics process in solving problems
Architectural design	Establishing the overall structure of software system
Compatible	Capable of ordering efficient intergration and operation with other elements in a system with no modification
Database	A collection of stored related data
Encapsulate	To express or show the most important facts about something
ER diagram	Entity Relationship Diagram, Data model for describing data base in an abstract way
SDS	System design specification
Sequence Diagram	An interaction diagram that show how process interact with one another and in what order.
SRS	Software Requirement Specification

Document overview

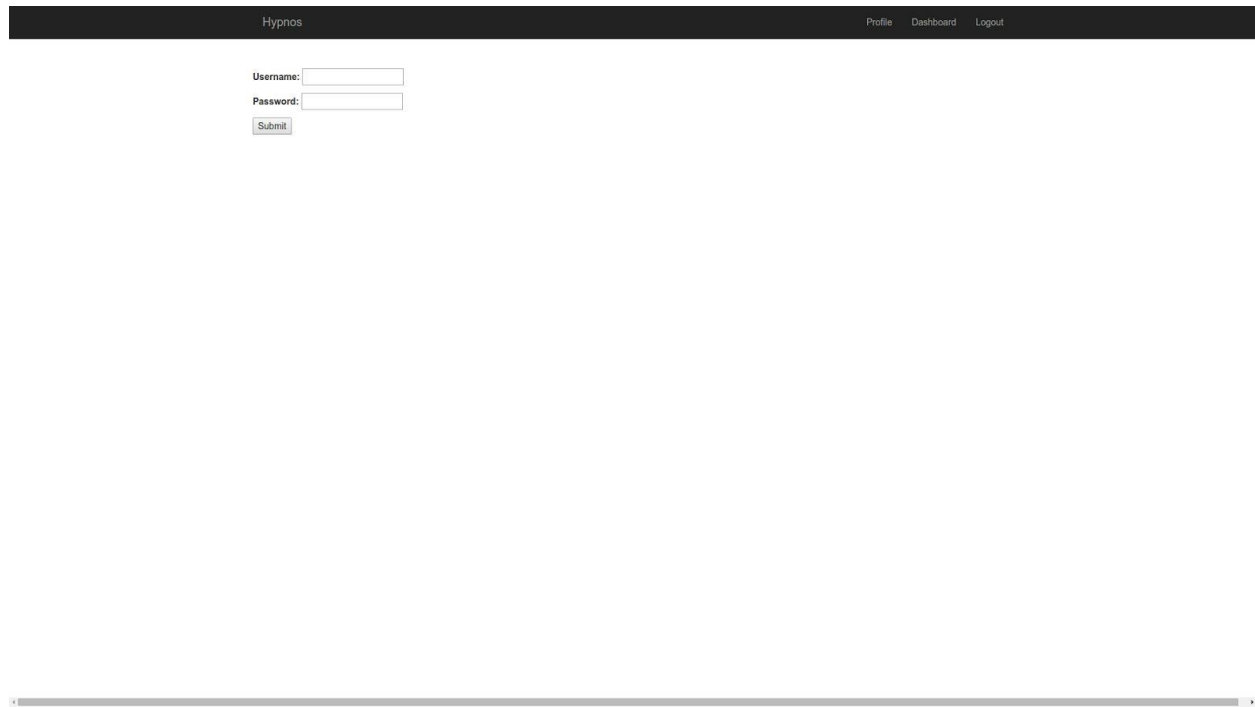
The next chapter of the document has described architectural design of the Disease prediction registration system. The high level components and their interactions, suitable architectural patterns, physical arrangement of components and design applied to the whole system.

The other chapter of this SDS is on components and detailed design, includes design patterns , sequence diagrams, class diagrams, database design in detail and user- interface design with screenshots of the interface

Premium health care

DISEASE PREDICTION SYSTEM

This the screenshot of the user interface that account already existing account



The screenshot shows a web interface for a system named 'Hypnos'. At the top, there is a dark navigation bar with the name 'Hypnos' on the left and links for 'Profile', 'Dashboard', and 'Logout' on the right. Below the navigation bar, the main content area contains a login form. The form has two input fields: 'Username:' and 'Password:', each followed by a text box. Below these fields is a 'Submit' button. The entire form is centered on the page.

This the sign up format of a car owner with no existing account in hand

HIGH LEVEL COMPONENT AND INTERFACE

Premium health care

DISEASE PREDICTION SYSTEM

Components

Patients profile

Disease Predictor

Show Probabilities

The Probable disease are:

Highly probable diseases:
Common Cold

Medium probable diseases:

Low probable diseases:
HIV/AIDS - the First Stage
Dengue Fever
Flu
Hepatitis B
Malaria

Index	
greater than equal to 0.5	Highly Probable
greater than equal to 0.3	Medium Probable
Less than 0.3	Low Probable

This is a key area in disease prediction system, Where the profile of the patient is maintained and looked by the patient and by the database administrator

Authentication and user management components

This is the major sub system that is responsible for the security of the security of the system. It authenticate users and also handles the user management activities such as creating new user account from the system etc. Furthermore this component implements the control access privilege matrix

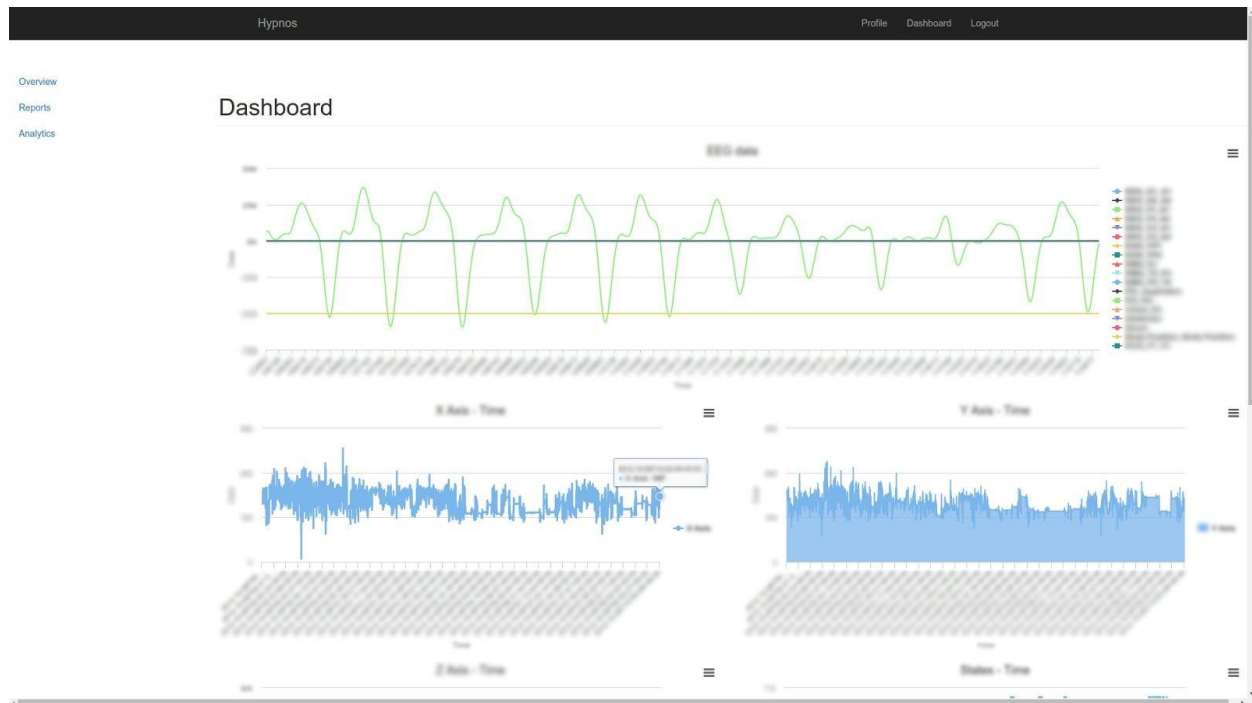
Subject component

This is the key component that implements the functions related to the subject operations of administrator such as adding a new body mass, Temperature, sleeping hour etc and removing the previous unwanted data in the system

Publish components

DISEASE PREDICTION SYSTEM

Dashboard



Authentication and user management component

Authenticate user: This is the interface that allows the users to login to the system. This will guide the user to the relevant home page

Publish component

Get notice: This is the interface where notifications are published. It is connected with the user component

Public component

Dashboard: This is the interface which shows the result of the disease and graph of health history to user, therefore connected with the user component

Premium health care

DISEASE PREDICTION SYSTEM

Architectural styles/Patterns

The disease prediction system will be developed under two main architectural styles/patterns. Development of the project will be done in MVC architectural style and also 3 tier Client/Server architecture. Client can browse the internet and access the health system provided they are within the local area network.

MVC Architecture (Model – View – Controller)

MVC Style separates presentation and interaction from the system data. The system is structured into three logical components that interact with each other

The model component –Manages the system data and associated operations on that data.

The View component – Defines and manages how the data is presented to the user.

The Controller components – Manages user interaction and passes these interaction to the View and the model

We will use this MVC style for the disease prediction system because, there are multiple ways to view and interact with data. Also used when the future requirements for interaction and presentation of data are unknown. In some software systems the code between the process logic and interface are mixed. This will reduce the modularity of application and make the make the system more difficult to maintain. To avoid this problem we have decided to use MVC architectural style to separate the application logic with the interface. The main advantage of this style allows the data to change independently of its representation and vice versa. Support presentation of the same data in different ways with changes made in one representation shown all of them

Three-tier Client/Server Architecture

In a client server architecture, the functionality of the system is organized into services, with each service delivered from separate server. Client are users of these services and access servers to make use of them. We will use this 3- Tier Client are Architecture because, When data in a shared database has to be accessed from a range of locations. Because server can be replicated, may also be used when the load on a system is a variable.

DISEASE PREDICTION SYSTEM

Data tire

The data tire maintains the applications data such as User' data, disease details, chatbot and comments and the SQL queries. It stores these data in a relational database management system (RDBMS). All the connections with the RDBMS are managed in the tier.

Middle Tire

The middle tier (web/ application server) implements the business logic, controller logic and presentation logic to control the interaction between the application' client and data. Business rules enforced by the business logic dictate how client and cannot access application data and how application process data.

Client Tire

The client tire is the application user interface connecting data entry forms and client side applications. It displays data to the user server .User interact directly with the application through user interface. The client tier interacts with the web/application server to make requests and to retrieve data from database. It displays to the user the data retrieved from the server.

Different process and Their communication

In the disease prediction system, there are number of different processes, such as database server process, web server process, connections between above server likewise. When sending mails there should run a mail server. HTTP protocol is using to communicate with web server, SMTP protocol is using to communicate with mail servers. They should communicate each other well to perform the functions of whole application.

DISEASE PREDICTION SYSTEM

Arrangement of devices and servers

Disease prediction system needs some specific set of servers and devices: Such as:

Server to host web applications and web services service applications.

Personal computer, note book, smart phone etc... to access the website

Mode /router/switch/hub/WI-FI network/ cable network etc And also need an internet Service Provider to have the Internet connectivity.

Communication among components

Below devices are communicating with each other. Personal computer communicates with web server and the database through HTTP protocol. It communicates with mail server through SMTP protocol. Cable network or wi-Fi network is also a communication method using in connecting different network components

Three-Tier Client Server Architecture

Reasons:

- As more user access the system a three tier solution is more scalable than the other solutions because you can add as many middle tiers as needed to ensure good performance.
- Security is also the best in three-tier architecture because the middle layer protects the database tier
- Ease to coding and provide well defined interfaces within each logic

DISEASE PREDICTION SYSTEM

Techniques Used

Prototyping

In designing the disease prediction system prototyping will be used to demonstrate underpinning concept of the designing and for user interface. This techniques will provide the opportunity for the system users to experiment the software to a certain extent during the development process.

Architecture

The architecture provides the top level design view of a system and provides a basis for more detailed design work

Strategy 1

Database schema

Tables, Fields and Relationships

Provide a description of any new tables, fields and relationships that need to be created for the design

Databases

Following are the software used for the disease prediction system

	Software Product	Source
Client on internet	Web Browser	SE
Cloud OS	Ubuntu 14.04, window 10 pro	http://www.ubuntu.com/
Web Server	Django 1.9	https://www.djangoproject.com/
Database Server	PostgreSQL 9.4 , mysql	http://www.postgresql.org/
Development End	HTML5, JS, Bootstrap, Android, MPAndroidChart, Highcharts	Latest & Stable Version

DISEASE PREDICTION SYSTEM

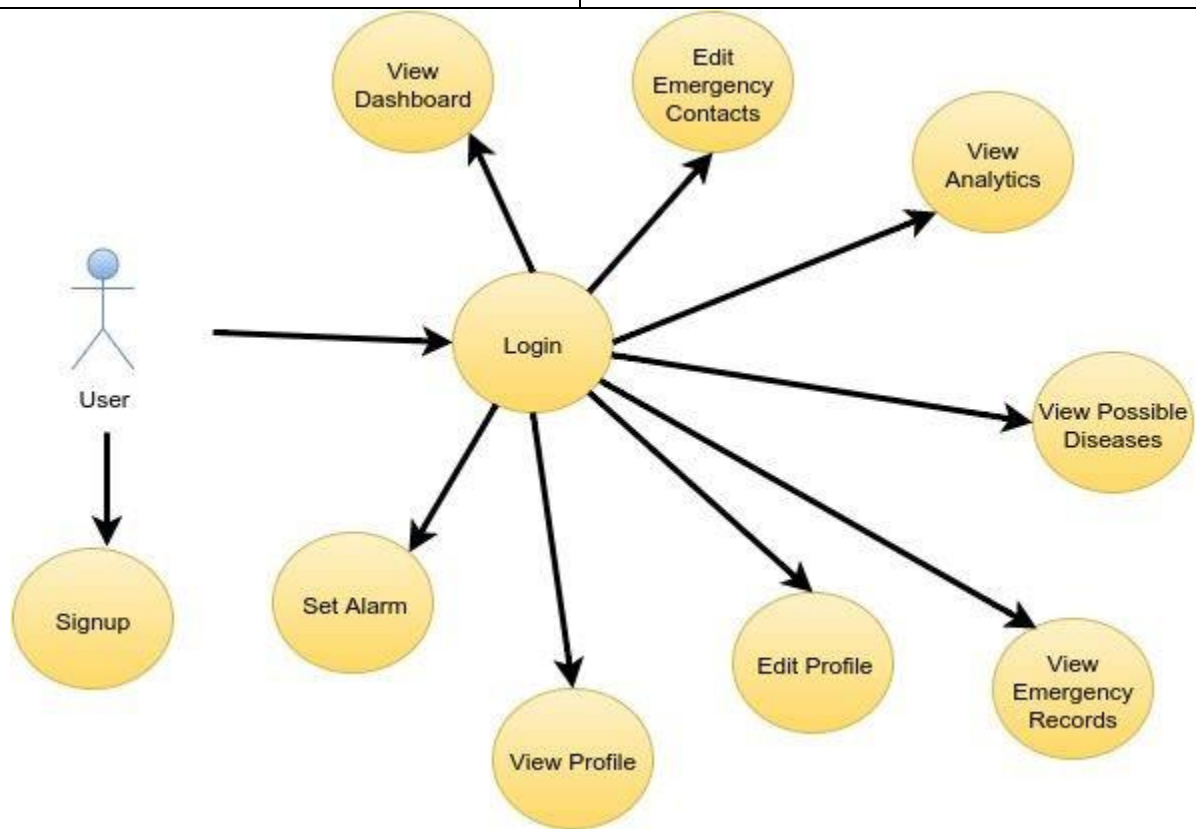
Machine Learning	Python, Scikit, Numpy, Scipy, Matplot	Latest & Stable Version
------------------	---------------------------------------	-------------------------

SYSTEM OVERVIEW

Use Case	Description
Login	Users need to login to start using system.
Register	Users need to register themselves and their devices to system.
Check profile	Check disease profile of previous prediction
View Profile	Users can see their profile which includes health information.
Edit Profile	Users can change their personal & health information.
View Emergency Records	Users can see history of emergency situations.
View Disease Predictions	Users can see possible diseases predicted by Naïve Bayes algorithm using the system.
Naïve	Users can see their information filtered by time and sensor choice.
View Dashboard	View the recent predicted disease by chatbot .

DISEASE PREDICTION SYSTEM

View Analytics	Users can see the results of the applied machine learning algorithms.
Edit emergency contacts	Users can edit contacts to be informed in emergency situations



User Interface

This how the login and sign up form is going to look like

The two pictures shows how the user will be able to sign up for a new account to get in touch with the services offered by the system

Testing

The administrative will test the system expecting pros and cons after the test list the rating performance influence that it will have to the society

Expected software response

Expected software response is required unless otherwise

Identification of critical components

Premium health care

DISEASE PREDICTION SYSTEM

The key error and wrong placed and written components during testing are exposed to open and rectified

References

IEEE Guide for Software Requirements Specifications," in IEEE Std 8301984 , vol., no., pp.126, Feb. 10 1984, doi: 10.1109/IEEESTD.1984.119205,
URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=278253&isnumber=6883>

Appendix C of Don Widrig, Dean Leffingwell, "Managing Software Requirements: A Unified Approach," AddisonWesley Professional, Release Date: October 1999, ISBN: 0201615932.

Marko Borazio, Eugen Berlin, Nagihan Kücükıldiz, Philipp M Scholl and Kristof Van Laerhoven, Towards Benchmarked Sleep Detection with Inertial Wristworn Sensing Units", ICHI 2014, Verona, Italy, IEEE Press, 09/2014.

Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for LowRate Wireless Personal Area Networks (WPANs), URL:
<http://standards.ieee.org/getieee802/download/802.15.4d2009.pdf>