NAME: ATOGWE VICTORIA ALOIYE

MATRIC NO.: 18/ENG08/003

DEPARTMENT: BIOMEDICAL ENGINEERING

COURSE TITLE: STRUCTURED COMPUTER PROGRAMMING

COURSE CODE: ENG 224

ASSIGNMENT QUESTION: Design the application following the software development cycle.

Critically discuss the hardware and software features.

Support your answer with a flowchart and an algorithm.

Draw the Top-down or Bottom-up design approach of the application.

SOFTWARE DEVELOPMENT CYCLE.

1. **Conceptualization:** The use of this web-based application is that it will be able to detect, display, rate (degree of infection), store, transmit data obtained wirelessly and access the data via the web together with other features. This development will be of great benefit to the health workers and the general public during this pandemic. It would help to reduce the spread of this virus between individuals as the virus would be detected early by prompting questions for the mobile phone owner on a daily basis. It is also known some carriers of the virus are asymptomatic, the application will help to identify them by virtue of physical proximity information provided to symptomatic people and store this information in its database. Through this people will be able to know which areas are more

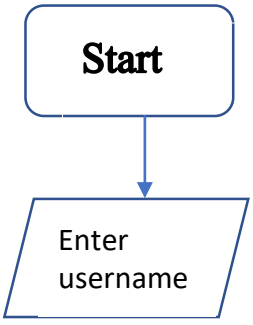
densely populated with infected individuals in order to avoid those areas and help flatten the curve. Health workers will also be able to track the infection and recovery rate efficiently.

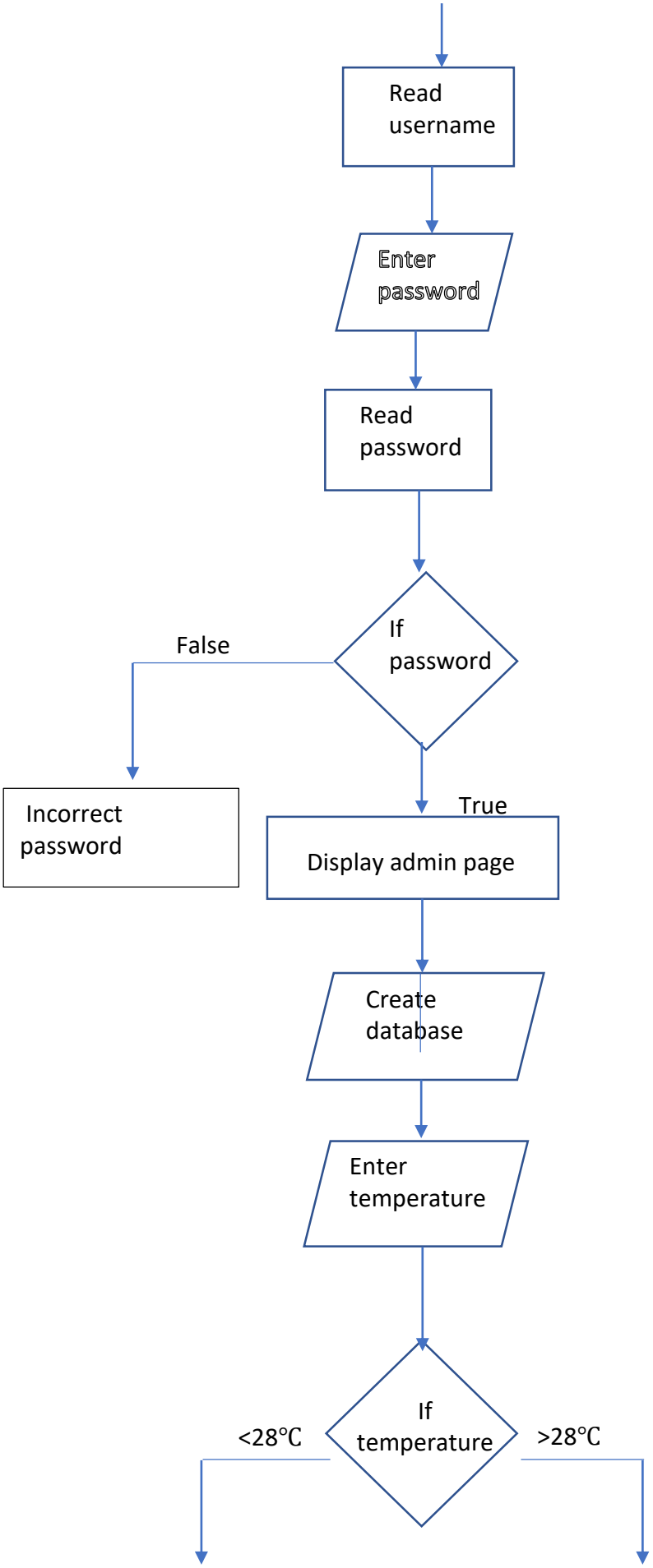
2. **Specification:** The application will require a GUI (Graphical User Interface) that includes push buttons, text boxes, dialogue boxes, text views etc, to ensure that even the individual with the lowest form of education would be able to operate it when taught. Front-end development using languages like HTML, CSS and Java Script would be used to ensure this. The back-end development using languages like Java, PHP or Python will ensure adequate communication between the application and the database. This would make the application fast, accurate and efficient.
3. **Design:**

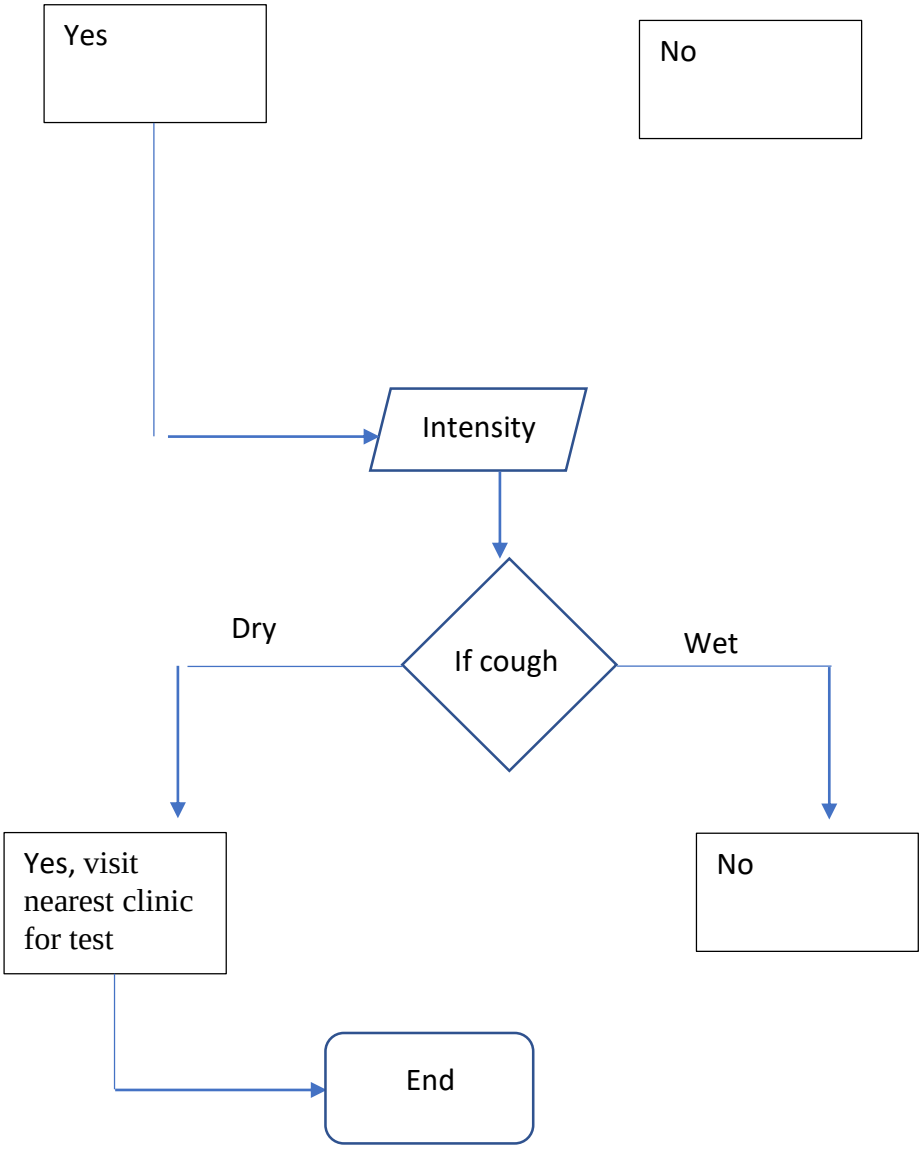
A. Algorithm: 1. Start

2. Read Username
3. Enter Password
4. Read Password
5. If Password=True, display admin page
   Admin page: Enter Name, Age, Address, Medical history, Symptoms (S)
   Display "Input Name, Age, Address, Medical history, Symptoms (S)"
   Read "Name, Age, Address, Medical history, Symptoms (S)"
   Create Database
   Else
   If Password = False, display "incorrect password"
6. Symptoms (S)=Display "Temperature, sore throat, cough"
7. Temperature = Display "Enter Temperature"
   If Temperature < 28 ℃, display "Yes"
   Else
   If Temperature > 28℃, display "No, you're safe"
8. Cough= Display "Cough Intensity"
   If Cough = Dry, display "Yes, visit nearest clinic for test"
   Else,
   If Cough = Wet, display "No
9. End

B. Flowchart




Read username
Enter password
Read password
If password
False
Incorrect password
True
Display admin page
Create database
Enter temperature
If temperature
<28°C
>28°C



4. **<u>Implementation:</u>** The architecture of the program will be in two phases i.e. a front-end and back-end development.

<u>Front-end Development:</u> This manages the body and looks of the application. Programming languages such as HTML (Hypertext Markup Language) which is the standard markup language for applications designed to be displayed in a web browser. It can be assisted by scripting languages such as JavaScript to create the interphase and add functionalities, and technologies such as Cascading Style Sheets (CSS).

<u>Back-end Development:</u> This refers to the server side of the application and everything that communicates between the database and the browser. It is simply the working of the application. This phase development is divided into three parts: the server, the application and the database. Ruby on Rails language would be used to achieve this as it is more cost-effective while being able to reach more people.

5. **Testing and debugging:** Although the application is tested at every stage of its development, after the front-end and back-end development. The final integrated testing is carried out over the web to fix final bugs before it is introduced to the general public to ensure its smooth running and user-friendly interface.

6. **Release and update:** This is to ensure the application is in the market using real time interfacing to maintain it fixes bugs as they appear. It also involves getting review from users and updating the application when and where necessary. To deploy and maintain the application a domain and hosting site are needed. The domain such as .com,.io,.co provide the site's location whereas the host site stores the information and codes used to build the website.

HARDWARE AND SOFTWARE FEATURES.

Software features should include;

- An interface that indicates number of recorded cases according to the location of the user.
- An administration interface which can only be accessed by specified users through an approved password, an admin can modify and make changes to various part of the application.
- An interface that indicates number of recorded cases worldwide.
- An access control management software to ensure that the data isn't altered or tampered with by an unauthorized third-party.
- A section to display the pandemic curve which forecasts the rise or fall of the spread of the pandemic.
- A questionnaire with preset questions on the symptoms to check if the user is infected with the disease.
- User choice in language of the application and the preferred units of temperature etc.
- A Database Management System.

Hardware features of the application are basically any device capable of accessing the internet e.g. mobile phones, tablets, modems, etc.