  
  
**Assignment Cover Sheet**

|  |  |
| --- | --- |
| **Subject Code:** | **CSIT321** |
| **Subject Name:** | Capstone Project |
| **Submission Type:** | Online |
| **Assignment Title:** | Design Document |
| **Student Name:** | Kaye Lim, Mahmoud Hamdan, Maryam Ahmed, Rabia Sahr, Zeyad Hasan |
| **Student Number:** | 6525325, 6320764, 6623426, 6142540, 6287591 |
| **Student Phone/Mobile No.** | 0504719659,0506172148,0559922741,0563645404,0553391145 |
| **Student E-mail: @uowmail.edu.au** | kdl601, mh770, mas869, rs740, zhah235 |
| **Lecturer Name:** | Dr. Mai El Barachi |
| **Due Date** | 05/01/2023 |
| **Date Submitted:** | 05/01/2023 |

|  |  |
| --- | --- |
| **PLAGIARISM:** The penalty for deliberate plagiarism is FAILURE in the subject. Plagiarism is cheating by using the written ideas or submitted work of someone else. UOWD has a strong policy against plagiarism.  The University of Wollongong in Dubai also endorses a policy of non-discriminatory language practice and presentation.  **PLEASE NOTE:**STUDENTS MUST RETAIN A COPY OF ANY WORK SUBMITTED | **DECLARATION:** I/We certify that this is entirely my/our own work, except where I/we have given fully-documented references to the work of others, and that the material contained in this document has not previously been submitted for assessment in any formal course of study. I/we understand the definition and consequences of plagiarism.  **Signature of Student:** Text, letter  Description automatically generated Text, letter  Description automatically generatedText, letter  Description automatically generated |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | **Optional Marks:** | | **Comments:** | |

https://my.uowdubai.ac.ae/images/scissors.gif

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Lecturer Assignment Receipt**(To be filled in by student and retained by Lecturer upon return of assignment) | | | **Subject:** | **Assignment Title:** | | **Student Name:** | **Student Number:** | | **Due Date:** | **Date Submitted:** | | **Signature of Student:** | | |

https://my.uowdubai.ac.ae/images/scissors.gif

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Student Assignment Receipt** (To be filled in and retained by Student upon submission of assignment) | | | **Subject:** | **Assignment Title:** | | **Student Name:** | **Student Number:** | | **Due Date:** | **Date Submitted:** | | **Signature of Lecturer** | | |

Logo

Description automatically generated with medium confidence

**CERTIFICATION**

We, Team Techbenders, declare that this technical document, submitted in fulfilment of the requirements for the completion of the graduation project subject CSIT321, in the Faculty of Engineering & Information Sciences, University of Wollongong in Dubai, is wholly our own work unless otherwise referenced or acknowledged. The document has not been submitted for qualifications at any other academic institution.

Zeyad Hesham *Zeyad Hesham*

Kaye Lim *Kaye Lim*

Sahr Rabia *Sahr Rabia*

Maryam Syed *Maryam Syed*

Mahmoud Hamdan *Mahmoud Hamdan*

**EXECUTIVE SUMMARY**

DiabetesCare is a web app designed to assist individuals with diabetes in managing their condition by providing recommendations for diet, physical activity, and mental health based on their activity and tracking their blood glucose and blood pressure. The app will use machine learning techniques, including natural language processing (NLP), to analyze data from social media platforms and journal entries. The app aims to help users develop the confidence, awareness, and skills needed to properly manage their diabetes, encourage them to develop a healthier lifestyle, and improve their mental health. The app will be free to use and will be accessible on various devices.

This report is a comprehensive document that outlines the feasibility and requirements for developing an app designed to assist individuals with diabetes in managing their condition. The report begins by outlining the scope of the project, including the goals and objectives of the app and a PESTEL analysis to assess the potential impact of external factors on its success. The report also includes a competitor analysis, a work breakdown structure, a project budget with time and cost estimates, and performance measurement baselines. Additionally, it includes a list of milestones and associated dates, a staffing plan, and a division of work among team members, as well as a SWOT analysis and risk management plan. The design of the app includes descriptions and visualizations of the overall architecture of the app, the specific components and features of the app, and how the app will appear and function for the user. This is then followed by showing the implementation of the important technologies and requirements of the application. Lastly, the testing of the app includes functional testing to ensure all features are working properly, and maintenance includes adapting the system to cope with changes and implementing new user requirements.

**ACKNOWLEDGEMENT**

We would like to take this opportunity to thank everyone who has supported us at every level of our project. We would like to express our sincerest gratitude and appreciation to our faculty mentor, Dr. Ali Zalzala who has been with us for two semesters, providing us with invaluable insights and expertise that were crucial in the successful completion of our work. We are deeply grateful for his mentorship and assistance, and for the time and effort he invested in our project. We would also like to thank Dr. Zeenath Khan and Dr. Mai El Barachi for their assistance and support throughout the project. Their help and guidance were essential to the success of our work.

Lastly, we would like to express our sincerest gratitude and appreciation to our families and friends for their support during the course of our final project. The love and encouragement they provided were instrumental in helping us stay motivated and focused throughout the project. We are deeply grateful for their unwavering support and for the belief they had in us.

Table of Contents

[I. Introduction 11](#_Toc123843459)

[II. Feasibility 11](#_Toc123843460)

[1. Project Charter 11](#_Toc123843461)

[2. Scope Statement 12](#_Toc123843462)

[3. Goals and Objectives 12](#_Toc123843463)

[4. PESTEL Analysis 13](#_Toc123843464)

[5. Competitor Analysis 13](#_Toc123843465)

[6. Work Breakdown Structure 14](#_Toc123843466)

[7. Project Budget, including cost and time estimates 19](#_Toc123843467)

[7.1 Time Estimate 19](#_Toc123843468)

[7.2 Cost Estimate 20](#_Toc123843469)

[8. Performance Measurement Baselines 20](#_Toc123843470)

[9. Milestones and Associated Dates 21](#_Toc123843471)

[10. Staffing 22](#_Toc123843472)

[11. SWOT Analysis 23](#_Toc123843473)

[12. Risk Management Plan 23](#_Toc123843474)

[12.1 Risk Register 25](#_Toc123843475)

[12.2 Risk Resolution 25](#_Toc123843476)

[12.3 Risk Breakdown Structure 26](#_Toc123843477)

[13. Open Issues 26](#_Toc123843478)

[III. Requirements 27](#_Toc123843479)

[1. Introduction 27](#_Toc123843480)

[a. Overview 27](#_Toc123843481)

[b. Definitions 27](#_Toc123843482)

[c. Document Conventions 28](#_Toc123843483)

[d. Assumptions 28](#_Toc123843484)

[2. Design Constraints 28](#_Toc123843485)

[2.1 User Characteristics 28](#_Toc123843486)

[2.2 Mandated Constraints 29](#_Toc123843487)

[2.3 Potential System Evolution 29](#_Toc123843488)

[3. Nonfunctional Requirements 29](#_Toc123843489)

[3.1 Usability Requirements 29](#_Toc123843490)

[3.2 Operational Requirements 30](#_Toc123843491)

[3.3 Performance Requirements 30](#_Toc123843492)

[3.4 Security Requirements 31](#_Toc123843493)

[3.5 Safety Requirements 31](#_Toc123843494)

[3.6 Legal Requirements 31](#_Toc123843495)

[3.7 Documentation and Training 31](#_Toc123843496)

[3.8 External Interface 32](#_Toc123843497)

[4. System Features 32](#_Toc123843498)

[4.1 Twitter Analysis 32](#_Toc123843499)

[4.2 Feature: <Physical Activity Recommender System > 33](#_Toc123843500)

[4.3 Feature: <Diet Recommender System> 34](#_Toc123843501)

[4.4 Feature: Mental Health Recommender System 35](#_Toc123843502)

[4.5 Feature: <Sign up> 36](#_Toc123843503)

[4.6 Feature: <Login> 36](#_Toc123843504)

[4.7 Feature: <Self-Assessment> 37](#_Toc123843505)

[4.8 Feature: <Glucose Level Tracker> 37](#_Toc123843506)

[4.9 Feature: <Blood Pressure Tracker> 38](#_Toc123843507)

[4.10 Feature: <Journal Entry> 38](#_Toc123843508)

[5. Project Matters 38](#_Toc123843509)

[5.1 Preliminary Schedule 38](#_Toc123843510)

[5.2 Preliminary Budget 39](#_Toc123843511)

[IV. Design 40](#_Toc123843512)

[1. Introduction 41](#_Toc123843513)

[2. System Overview 41](#_Toc123843514)

[3. Design Map 41](#_Toc123843515)

[4. Supporting Materials 41](#_Toc123843516)

[5. Definitions and Acronyms 42](#_Toc123843517)

[6. Design Considerations 42](#_Toc123843518)

[6. 2 Assumptions 42](#_Toc123843519)

[6.3 Constraints 43](#_Toc123843520)

[6.3 System Environment 44](#_Toc123843521)

[6.4 Design Methodology 44](#_Toc123843522)

[6.5 Risks and Volatile Areas 44](#_Toc123843523)

[7. Architecture 45](#_Toc123843524)

[7.1 System Architecture 45](#_Toc123843525)

[7.2 Network Architecture 46](#_Toc123843526)

[8. Low Level Design 47](#_Toc123843527)

[8.1 Module 47](#_Toc123843528)

[9. User Interface Design 48](#_Toc123843529)

[9.1 Application Control 48](#_Toc123843530)

[10. High Level Design 48](#_Toc123843531)

[10.1 User Twitter Analysis (Mental, Nutrition, Physical) 48](#_Toc123843532)

[10.2 Mental Health Recommender System 49](#_Toc123843533)

[10.3 Physical Activity Recommender System 49](#_Toc123843534)

[10.4 Diet Recommender System 50](#_Toc123843535)

[10.5 Package Diagram 51](#_Toc123843536)

[10.6 Class Diagram 52](#_Toc123843537)

[10.7 Conceptual Diagram 53](#_Toc123843538)

[10.8 Deployment Diagram 53](#_Toc123843539)

[11. High-Fidelity Design (Screen) 54](#_Toc123843540)

[V. Implementation 61](#_Toc123843541)

[1. Recommender Systems 61](#_Toc123843542)

[1.1 Mental Health 61](#_Toc123843543)

[1.2 Physical Activity 62](#_Toc123843544)

[1.3 Nutrition 63](#_Toc123843545)

[2. Twitter Analysis 63](#_Toc123843546)

[2.1 Mental Health Model 63](#_Toc123843547)

[2.2 Physical Activity Model 65](#_Toc123843548)

[2.3 Nutrition Model 65](#_Toc123843549)

[VI. Testing 66](#_Toc123843550)

[1. Functional Testing Goals 66](#_Toc123843551)

[2. Functional Test Plan Scope 66](#_Toc123843552)

[3. Functional Test Plan Assumptions and Constraints 67](#_Toc123843553)

[3.1 Assumptions 67](#_Toc123843554)

[3.2 Constraints 67](#_Toc123843555)

[4. Functional Test Entry Criteria, Approach and Tools 67](#_Toc123843556)

[4.1 Testing Criteria 67](#_Toc123843557)

[4.2 Testing Methodology 67](#_Toc123843558)

[4.3 Testing Tools 68](#_Toc123843559)

[5. Functional Test Entry Criteria, Approach and Tools 68](#_Toc123843560)

[VII. Maintenance 78](#_Toc123843561)

[Appendices 78](#_Toc123843562)

[1. Glossary 78](#_Toc123843563)

[2. Version Index 79](#_Toc123843564)

[3. References 79](#_Toc123843565)

[4. Progress Reports 80](#_Toc123843566)

[5. Meeting Minutes 81](#_Toc123843567)

[6. Division of work among team members 84](#_Toc123843568)

**Tables**

[Table 1: Project Charter 11](#_Toc123842977)

[Table 2: Competitor Analysis 13](#_Toc123842978)

[Table 3: Work Breakdown Structure Tables 14](#_Toc123842979)

[Table 4: Time Estimate 19](#_Toc123842980)

[Table 5: Cost Estimate 20](#_Toc123842981)

[Table 6: Milestones 21](#_Toc123842982)

[Table 7: Staffing 22](#_Toc123842983)

[Table 8: Risk Management 24](#_Toc123842984)

[Table 9: Register 25](#_Toc123842985)

[Table 10: Risk Resolution 25](#_Toc123842986)

[Table 11: Budget 39](#_Toc123842987)

[Table 12: Definitions 42](#_Toc123842988)

[Table 13: Test Plan Scope 66](#_Toc123842989)

[Table 14: Test Case - Sign up 68](#_Toc123842990)

[Table 15: Test Case - Self-Assessment 69](#_Toc123842991)

[Table 16:Test Case - Log in (authentication) 70](#_Toc123842992)

[Table 17: Test Case - Recommendations for Mental Health 71](#_Toc123842993)

[Table 18: Test Case - Recommendations for Physical Activity 73](#_Toc123842994)

[Table 19: Test Case -Recommendations for Nutrition 74](#_Toc123842995)

[Table 20: Test Case - Twitter Analysis 75](#_Toc123842996)

[Table 21: Version Index: Final Document 79](#_Toc123842997)

[Table 22: Division of Work 84](#_Toc123842998)

**Figures**

[Figure 1: Work Breakdown Structure Diagram 19](#_Toc123842930)

[Figure 2: Risk Breakdown Structure 26](#_Toc123842931)

[Figure 3: Sequence Diagram: <User Twitter Analysis (Mental, Nutrition, Physical)> 33](#_Toc123842932)

[Figure 4: Sequence Diagram: <Physical Activity Recommender System> 34](#_Toc123842933)

[Figure 5: Sequence Diagram: <Diet Recommender System > 35](#_Toc123842934)

[Figure 6: Sequence Diagram for <Mental Health Recommender System> 35](#_Toc123842935)

[Figure 7: Use Case Diagram: <Sign up> 36](#_Toc123842936)

[Figure 8: Use Case Diagram: <Login> 36](#_Toc123842937)

[Figure 9: Use Case Diagram: <Self-Assessment> 37](#_Toc123842938)

[Figure 10: Sequence Diagram for <Glucose Level Tracker> 37](#_Toc123842939)

[Figure 11: Sequence Diagram for <Blood Pressure Tracker> 38](#_Toc123842940)

[Figure 12: Use Case Diagram for Journal Entry 38](#_Toc123842941)

[Figure 13: Gantt Chart: Preliminary Schedule 39](#_Toc123842942)

[Figure 14: Design Methodology on Diabetes Management 44](#_Toc123842943)

[Figure 15: System Architecture of DiabetesCare 46](#_Toc123842944)

[Figure 16: Network Architecture of DiabetesCare 47](#_Toc123842945)

[Figure 17: Web Application flowchart 47](#_Toc123842946)

[Figure 18: Wireframe Design 48](#_Toc123842947)

[Figure 19: Sequence Diagram: Twitter Analysis 48](#_Toc123842948)

[Figure 20: Sequence Diagram: Mental Health 49](#_Toc123842949)

[Figure 21: Sequence Diagram: Physical Activity 49](#_Toc123842950)

[Figure 22: Sequence Diagram: Diet 50](#_Toc123842951)

[Figure 23: Package Diagram 51](#_Toc123842952)

[Figure 24: Class Diagram 52](#_Toc123842953)

[Figure 25: Conceptual Diagram 53](#_Toc123842954)

[Figure 26: Deployment Diagram 54](#_Toc123842955)

[Figure 27: Login 54](#_Toc123842956)

[Figure 28: Sign up 55](#_Toc123842957)

[Figure 29: Consent Form 55](#_Toc123842958)

[Figure 30: Self-Assessment 56](#_Toc123842959)

[Figure 31: Home Page 56](#_Toc123842960)

[Figure 32: Diet Page 58](#_Toc123842961)

[Figure 33: Physical Activity Page 59](#_Toc123842962)

[Figure 34: Mental Health Page 60](#_Toc123842963)

[Figure 35: Mental Health Data 61](#_Toc123842964)

[Figure 36: ML Model 61](#_Toc123842965)

[Figure 37: Algorithm 62](#_Toc123842966)

[Figure 38: Physical Activity Data 63](#_Toc123842967)

[Figure 39: Mental Health Model 1 64](#_Toc123842968)

[Figure 40: Mental Health Model 2 64](#_Toc123842969)

[Figure 41: Nutrition Model 65](#_Toc123842970)

[Figure 42: Testing Day 1 77](#_Toc123842971)

[Figure 43: Testing Day 5 77](#_Toc123842972)

[Figure 44: Semester 1 Progress Report 80](#_Toc123842973)

[Figure 45: Semester 2 Progress Report 81](#_Toc123842974)

[Figure 46: Spring 2022 81](#_Toc123842975)

[Figure 47: Autumn 2022 83](#_Toc123842976)

# Introduction

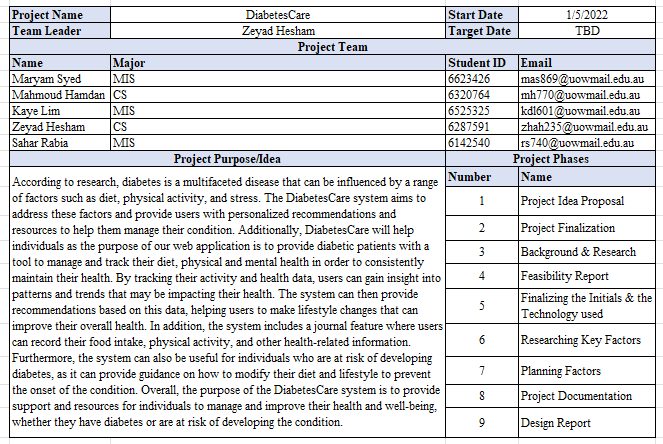
Diabetes is a serious condition that requires responsible management, as high blood glucose levels can have serious consequences. It is a complex disease that is influenced by various factors, including diet, physical activity, and stress. The prevalence of diabetes has been increasing globally, particularly in the US, Europe, and the Middle East and North Africa, due in part to low levels of physical activity and unhealthy diet (Wardian 2017).

To address this, our team at TechBenders has developed DiabetesCare, a system that helps diabetic individuals manage their diet, physical and mental health by providing recommendations based on their activity. DiabetesCare also tracks users' blood glucose and blood pressure to ensure consistent health. It uses a neural network algorithm to analyze user activity on platforms like Twitter and journaling to track progress.

# Feasibility

## 1. Project Charter

Table 1: Project Charter



## 2. Scope Statement

As per the findings, the seventh leading cause of death in the United States, diabetes costs a total estimated $327 billion in medical costs and lost work and wages (The facts, stats, and impacts of diabetes, 2022). In fact, people with diagnosed diabetes have more than twice the average medical costs that people without diabetes have. The main reason for this is that people are not aware of the seriousness of diabetes and how to control it.

This is where our application will come to help. The app includes glucose and blood pressure entries to keep track of both, the user can update it constantly throughout the day and compare the readings. It will help diabetics and pre-diabetic people to control or prevent diabetes by providing them with a specific caloric plan that they should follow. It will also track their progress to let them know how well they are doing. Furthermore, users will also be given daily workouts that they require to do every day to keep their blood sugar level in control which will come hand in hand with the dietary plan to reverse the effects of type 2 diabetes and keep it under control.

The app will also have a journaling feature where a user can write down what’s going through their minds, or how they feel to which the app will recommend specific activities based on the detected emotion in the journal entry.

Attracting people to use this app would be a difficult task, so a better initiative would be if the app were suggested by a doctor or healthcare professional. Moreover, it will help the doctor to use this data to track how well the patient is doing and what his progress is after the last appointment. This could provide them with better insights and would help them decide on what medications to prescribe next.

To accomplish all these functionalities, DiabetesCare will apply 2 machine learning techniques. First, we will implement NLP (Natural Language Processing) to collect social media data from sample community users using Twitter APIs. Afterward, the data will be captured using NLP to look for trends which will be converted into three charts showing the user’s well-being in mental health, nutrition, and physical activity. The second is text classification to be able to understand and classify the information written down on social media and journal entries.

## 3. Goals and Objectives

The goals and objectives of DiabetesCare are:

* Assist people in better managing and controlling their diabetes from the comfort of their own homes.
* Help users develop the confidence, awareness and skills they need to properly manage their diabetes.
* Use AI to keep track of user lifestyle habits based on social media activity and ensure they are on the right track.
* Encourage the users to develop a healthier lifestyle by suggesting personalized exercises and dietary options.
* Help users improve their mental health as it affects their blood sugar levels.

## 4. PESTEL Analysis

**Political**

Political factors include that any changes in government policies regarding application development and launch will have a direct impact on the application's release.

**Economical**

The web application is currently free to use, but it can be released on a larger scale in the future, which could incorporate certain premium features that would generate revenue.

**Social**

Some people might be hesitant to use the web application because they would rather make an appointment and visit the doctor or dietician in person. However, the system was designed to help individuals manage their diabetes in their daily lives rather than treat it, thus it is not a replacement for regular doctor’s checkups.

**Technological**

Technology has advanced significantly in recent years, therefore different changes in the technology is used may occur throughout the development of this application. Finding the most recent versions of technology and implementing them while maintaining compatibility with all devices used can help in the development of better software. Also, understanding and analyzing how technology is used by the market will help create more up-to-date software.

**Environmental**

Climate change, carbon footprint, or any other weather and environmental-related issue will not be having any effect on the web application. Therefore, environmental factors don’t play an important role in this situation.

**Legal**

Government regulations will have a direct impact on the web application both internally and externally since it will be accessible to a global audience. Understanding which rules and regulations apply in each country and then determining which ones should be applied is important.

## 5. Competitor Analysis

Table 2: Competitor Analysis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Features | DiabetiesCare | Diabetes:M | DiabTrend | Glucose Tracker – Diabetic Diary |
| Allows food entry | ✅ | ✅ | ✅ | ✅ |
| Provides specific calorie plan with recipes | ✅ | ❌ | ✅ | ❌ |
| Allows exercise log in | ✅ | ✅ | ✅ | ✅ |
| Provides personalized workout routine | ✅ | ❌ | ❌ | ❌ |
| Mental health journaling | ✅ | ❌ | ❌ | ❌ |
| Provides daily mental health activities | ✅ | ❌ | ❌ | ❌ |
| Glucose and BP tracker | ✅ | ✅ | ✅ | ✅ |
| Medication entry | ❌ | ✅ | ✅ | ✅ |

## 6. Work Breakdown Structure

The work breakdown structure that is located in (Appendix) illustrates a visual representation of our key deliverables and focuses on other important tasks throughout our development life cycle.

**Project Title:** Diabetes Management Application for Rural Communities

Table 3: Work Breakdown Structure Tables

|  |  |
| --- | --- |
| WBS Item Number: | 1 |
| WBS Item Name: | Conceptual Framework for application |
| Description: | What we are expecting to find during our research phase, that can vary from architectures to different platforms to create the application on, and even different design ideas that might suit or target users |

|  |  |
| --- | --- |
| WBS Item Number: | 1.1 |
| WBS Item Name: | Architectures |
| Description: | The first step of our framework research, we’ll be looking at different architectures, comparing their pros and cons and finding which is the most convenient and most effective to use |

|  |  |
| --- | --- |
| WBS Item Number: | 1.1.1 |
| WBS Item Name: | MVC |
| Description: | MVC is an architectural pattern consisting of three parts: Model, View, Controller. Model: Handles data logic. View: It displays the information from the model to the user. Controller: It controls the data flow into a model object and updates the view whenever data changes |

|  |  |
| --- | --- |
| WBS Item Number: | 1.1.2 |
| WBS Item Name: | MVP |
| Description: | Model–view–presenter (MVP) is a derivation of the model–view–controller (MVC) architectural pattern and is used mostly for building user interfaces. |

|  |  |
| --- | --- |
| WBS Item Number: | 1.1.3 |
| WBS Item Name: | MVVM |
| Description: | Model-View-View Model (MVVM) is a software design pattern that is structured to separate program logic and user interface controls. MVVM is also known as model-view-binder |

|  |  |
| --- | --- |
| WBS Item Number: | 1.1.4 |
| WBS Item Name: | Viper |
| Description: | VIPER (View, Interactor, Presenter, Entity and Router) is a design pattern for software development that develops modular code based on clean design architecture. The modules in VIPER are protocol-oriented and each function, property input and output are performed by way of specific sets of communication rules. |

|  |  |
| --- | --- |
| WBS Item Number: | 1.2 |
| WBS Item Name: | User Design |
| Description: | This part of the research will look deeper into different guidelines/platform and user experience factors that will play a role in the application. |

|  |  |
| --- | --- |
| WBS Item Number: | 1.2.1 |
| WBS Item Name: | Guidelines |
| Description: | This part will be specifically for looking into the different platforms we can apply this application on, for example IOS, android, a hybrid, or even a web application |

|  |  |
| --- | --- |
| WBS Item Number: | 1.2.1.1 |
| WBS Item Name: | IOS |
| Description: | Products that use Apple's iPhone operating system, including the iPhone, iPod touch and iPad. |

|  |  |
| --- | --- |
| WBS Item Number: | 1.2.1.2 |
| WBS Item Name: | Android |
| Description: | Android is a mobile operating system based on a modified version of the Linux kernel and other open-source software, designed primarily for touchscreen mobile devices such as smartphones and tablets. |

|  |  |
| --- | --- |
| WBS Item Number: | 1.2.1.3 |
| WBS Item Name: | Hybrid |
| Description: | A hybrid cloud—sometimes called a cloud hybrid—is a computing environment that combines an on-premises datacenter (also called a private cloud) with a public cloud, allowing data and applications to be shared between them. |

|  |  |
| --- | --- |
| WBS Item Number: | 1.2.1.4 |
| WBS Item Name: | Web Application |
| Description: | A Web application is an application program that is stored on a remote server and delivered over the Internet through a browser interface. |

|  |  |
| --- | --- |
| WBS Item Number: | 1.2.2 |
| WBS Item Name: | Concepts |
| Description: | This Part of the research will look into the different concepts when it comes to user requirements in terms of design and usability from the most optimal user experience, where the design is centered around the target user. |

|  |  |
| --- | --- |
| WBS Item Number: | 1.2.2.1 |
| WBS Item Name: | User Experience |
| Description: | User experience focuses on having a deep understanding of users, what they need, what they value, their abilities, and also their limitations. |

|  |  |
| --- | --- |
| WBS Item Number: | 1.2.2.2 |
| WBS Item Name: | User Centered Design |
| Description: | User-centered design is an iterative design process in which designers focus on the users and their needs in each phase of the design process. In User-centered design, design teams involve users throughout the design process via a variety of research and design techniques, to create highly usable and accessible products for them. |

|  |  |
| --- | --- |
| WBS Item Number: | 1.2.2.3 |
| WBS Item Name: | Designing for Society |
| Description: | Social design is the application of design methodologies in order to tackle complex human issues, placing the social issues as the priority. |

|  |  |
| --- | --- |
| WBS Item Number: | 2 |
| WBS Item Name: | App Implementation |
| Description: | This is the second stage of our project; we’re going to be applying all the concepts and theoretical work we did to an application |

|  |  |
| --- | --- |
| WBS Item Number: | 2.1 |
| WBS Item Name: | Study of Other Applications |
| Description: | Looking into similar applications and finding what they already have which helps us understand user requirements, and what we might add in addition to what already exists |

|  |  |
| --- | --- |
| WBS Item Number: | 2.1.1 |
| WBS Item Name: | Use Case Requirements |
| Description: | All the functionalities required for the application are based on the research done |

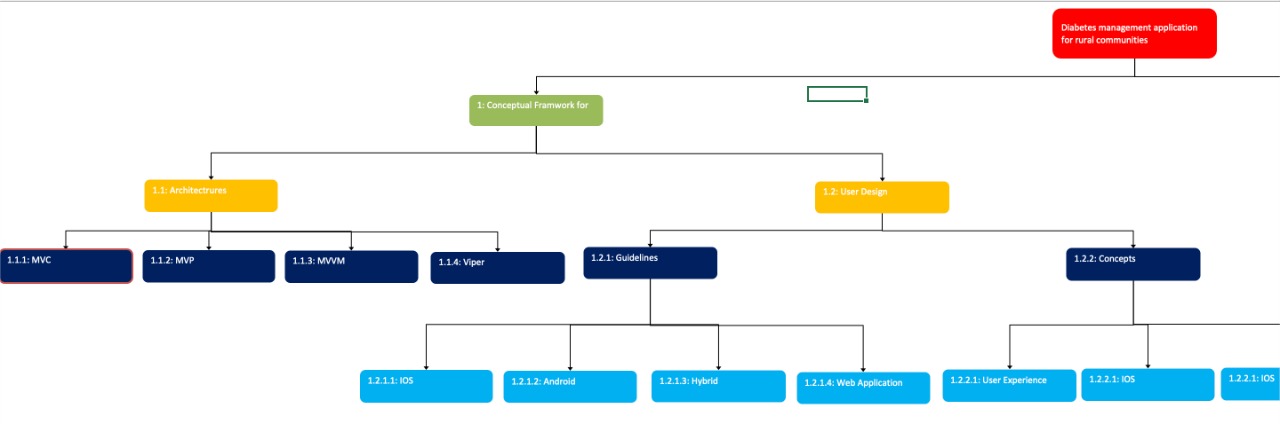
|  |  |
| --- | --- |
| WBS Item Number: | 2.1.2 |
| WBS Item Name: | Applying the design concepts |
| Description: | Applying all the design concepts and building it (Architecture, interface, platform). |

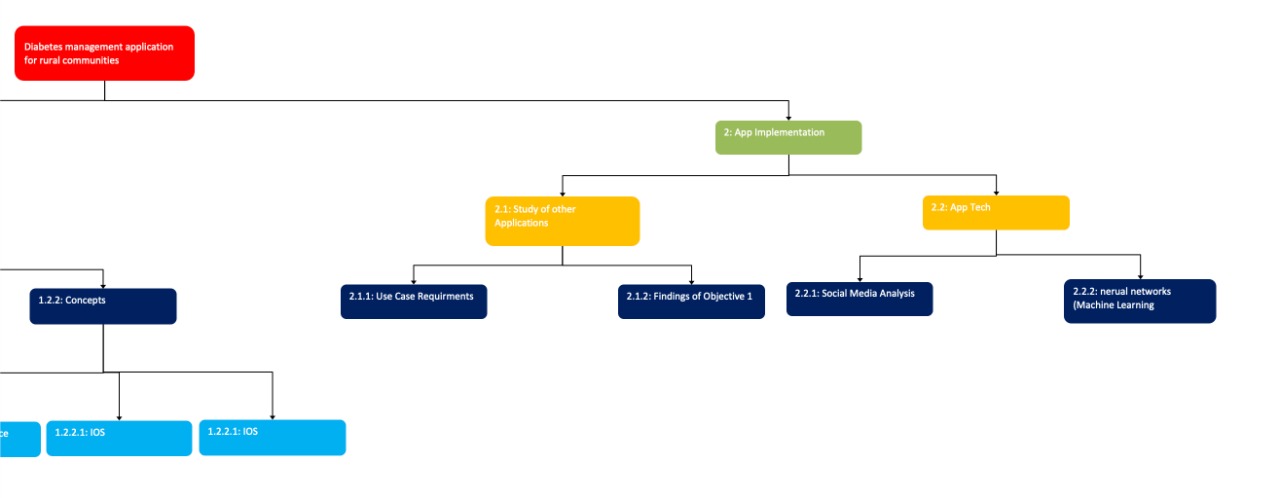
|  |  |
| --- | --- |
| WBS Item Number: | 2.2 |
| WBS Item Name: | App Tech |
| Description: | Applying machine learning and manual searching to scout for information regarding diabetes management. |

|  |  |
| --- | --- |
| WBS Item Number: | 2.2.1 |
| WBS Item Name: | Social Media Analysis |
| Description: | Using NLP on twitter’s open APIs to recognize specific phrases from tweets, retweets, following list, comments, likes. |

|  |  |
| --- | --- |
| WBS Item Number: | 2.2.2 |
| WBS Item Name: | Recommender System |
| Description: | Using a ML model to detect different emotions to suggest appropriate activities, and a rule-based system to suggest caloric intake and physical activities for users |

Figure 1: Work Breakdown Structure Diagram





## 7. Project Budget, including cost and time estimates

### 7.1 Time Estimate

The estimated time for each section is shown below according to the different items of the WBS. The decided time depends on the work done for each section.

Table 4: Time Estimate

|  |  |
| --- | --- |
| 1. **Conceptual Framework for architecture (Total 3 weeks)** | |
| 1.1 Architectures (Research and Analysis) | Total 1 weeks |
| 1.1.1 MVC | 2 days |
| 1.1.2 MVP | 2 days |
| 1.1.3 MVVM | 2 days |
| 1.1.4 Viper | 1 days |
| 1.2 User Design | Total 2 weeks |
| 1.2.1 Guidelines | 1 week in total |
| 1.2.1.1 IOS | 2 days |
| 1.2.1.2 Android | 2 days |
| 1.2.1.3 Hybrid | 1 day |
| 1.2.1.4 Web Application | 2 days |
| 1.2.2 Concepts | 1 week in total |
| 1.2.2.1 User Experience | 1 week |
| 1. **App Implementation (Total 15 weeks)** | |
| 2.1 Study of other applications | Total 4 weeks |
| 2.1.1 Use case requirements | 3 weeks |
| 2.1.2 Applying the design concepts | 1 weeks |
| 2.2 App Tech | Total 11 weeks |
| 2.2.1 Social Media Analysis | 6 weeks |
| 2.2.2 Recommender Systems | 4 weeks |

### 7.2 Cost Estimate

These are our cost estimates, these figures were obtained after we had done the research on the cost of the hardware and software required to develop the system successfully. As mentioned below, the main hardware we need is a laptop that has a minimum of 2GB RAM in order to create and run the web application smoothly. Moreover, the software we need is Google chrome/Firefox, HTML editor- Atom, free FTP client and for design, we would need Figma and Visual Studio.

Table 5: Cost Estimate

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Value** | **Quantity** | **Total Cost** |
| HP Envy 13.3" Intel i7 Laptop | AED4,620.00 | 1 | 4,620.00 |
| Google Chrome/Firefox | - | 1 | - |
| HTML Editor - Atom | - | 1 | - |
| FTP client (free) | - | 1 | - |
| Figma | - | 1 | - |
| Visual Studio | - | 1 | - |
| TOTAL COST | AED4,620.00 | **-** | AED4,620.00 |

## 8. Performance Measurement Baselines

**Schedule Baseline:** The time we have decided for the completion of the project is 18 weeks. The WBS is used to split the project into different sections so that the developers may finish each of the tasks in the set time frame. The 18 weeks will be divided for the conceptual framework for architecture and the app implementation.

**Cost Baseline:** We’ve chosen the necessary hardware and software required to develop the diabetes management system successfully and in an effective manner. Most of the items are free of cost aside from the laptop and Visual Studio, which in total would cost us approximately AED4,620.00.

**Scope Baseline:** The scope baseline we’ve set for developing and creating a successful diabetes management system is based on our scope statement and the WBS. In order to have the system implemented properly, we have distributed the tasks to the project manager, UI/UX designer/developer, back-end developers and the front-end developers.

## 9. Milestones and Associated Dates

Table 6: Milestones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Milestone number** | **Milestone name** | **Due date** | **Description** | **Status** |
| 1 | Project proposal | Sem 1  Week 1 | A project proposal with three proposals evaluated from one to three depending on the feedback | Completed |
| 2 | Project finalization | Sem 1  Week 2 | The three ideas were presented to the head of subject and the one that had the best feedback was chosen | Completed |
| 3 | Scope research | Sem 1  Week 3-6 | To explore the feasibility, development, and other elements of the project, research was taken by team members | Completed |
| 4 | Feasibility report | Sem 1  Week 3 | Research was done to determine the project's strengths, limitations, opportunities for improvement, and other open issues. | Completed |
| 5 | Finalization of the technologies used | Sem 1  Week 4 | The front end and back-end technologies required for the project was chosen | Completed |
| 6 | Research on the main features of the system | Sem 1  Week 4 | The features added to the system were discussed and researched on | Completed |
| 7 | Planning report | Sem 1  Week 4 | The report consisted of the project’s plan and other procedures | Completed |
| 8 | Requirement report | Sem 1  Week 6 | The report consisted of what is needed by the system, and it explains the finished function and what it must accomplish | Completed |
| 9 | Design report | Sem 1  Week 9 | The report includes the project's design, primary features, and project deliverables | Completed |
| 10 | Proof of concept presentation | Sem 2  Week 1 | To assess the project's potential, a prototype demo is shown to the professor | Completed |
| 11 | Updating scope | Sem 2  Week 2 | Changes made to the scope of the project after feedback was given from the professor | Completed |
| 12 | Test plan | Sem 2  Week 9 | Testing is done on the different functions to assess if their working properly | Completed |
| 13 | Final documentation | Sem 2  Week 10 | Final documentation of the entire project is done after the extension was given | Completed |

## 10. Staffing

Table 7: Staffing

|  |  |  |  |
| --- | --- | --- | --- |
| **Required Team Members** | **Role** | **Monthly Wage (AED)** | **Availability** |
| Zeyad Hasan | Project Manager/ Developer | 27,000 | Full Time |
| Kaye Lim | UX/UI Designer / Developer | 23,500 | Full Time |
| Maryam Syed | Front-End Lead Developer | 22,450 | Full Time |
| Rabia Sahr | Front-End Developer | 22,200 | Full Time |
| Mahmoud Hamdan | Back-End Developer | 24,350 | Full Time |
| **Consultant/s** | | | |
| Dr. Ali Zalzala | Main Mentor | N/A | Once a week |

The Project Manager oversees the full web development process, including planning, organizing, executing, and monitoring. While UX/UI designers’ responsibilities is to provide a seamlessly user experience. They are responsible for designing a user-friendly web application. In addition, a front-end web developer is in charge of developing visual components in a web application that users view and interact with. Lastly, Back-end developers deal directly with databases, data processing, and third-party service interaction.

Moreover, Zeyad Hesham and Mahmoud Hamdan will be our main developers, focusing on the back end of our web application. Whereas Kaye Lim, Maryam Syed, and Rabia Sahr will be specializing on the web application's front-end and assisting our main developers. After developing the web application, all members will test it, as website testing is critical for identifying the website's key problems. Testing all the features of the website is important to guarantee that they function properly and that end users do not encounter any difficulties when using the website.

## 11. SWOT Analysis

The team has produced a SWOT analysis model to help the team get visibility on the status of the project in addition to understand and measure overall performance.

**Strengths**:

* Flexible and innovative
* Ensures Customer satisfaction
* Prior experienced web developers
* Effective learning environment
* Effective communication
* Simplified data entry

**Weaknesses:**

* Data challenges
* No clinical history
* May require ongoing updates and maintenance to remain effective

**Opportunities**:

* Opportunities for partnerships and collaborations with healthcare providers and organizations
* Potential to generate revenue through subscriptions or partnerships
* Gaining in-depth knowledge about developing a fully functional software
* Opportunity for Self-diagnosis, self-management

**Threats**:

* Hack or breach of the diabetes application
* Technical illiteracy risks
* Changes in healthcare policies or regulations that could impact the market for the app
* Competition from other diabetes management apps or tools
* Dependence on external factors such as social media platforms and their policies

## 12. Risk Management Plan

As the team analyses the requirements of our system it is essential to manage and identify the risks that our team may face. Despite all the benefits a diabetes management system may possess, the risks shown could be alarming. The team has furthermore proceeded to perform numerous risk assessments.

We have followed 4 essentials steps to managing the risks:

1. Identify the risks
2. Asses the risks
3. Treat the risks
4. Monitor and report the risk.

Table : Risk Management

|  |  |
| --- | --- |
| **Risks** | **Risk Management strategy** |
| Data security risks | |
| Privacy and security | Unauthorized users gain access to private information about patients’ medical data. |
| Technical risks | |
| Hardware incompatibility | Hardware that supports medical and public health practices or an old OS device they are using may not be compatible. |
| Software management | Have a plan for software maintenance and management |
| Technical illiteracy risks | Individuals who live in rural areas will have no knowledge on how to use the software |
| Team Risks |  |
| Complexity constraint | All team members need to research about the various methods for developing a diabetes management system as well as the steps needed to complete the tasks within a given time frame. |
| Communication | Without communication, the development of the game could get delayed and cause some development confusion. |
| Usability | Usability of the software must be user friendly and focused on completing the tasks assigned. |
| Scheduling Risks |  |
| Crashing | Why does a system crash? Is it poor planning? Team needs to research and try to find ways to reduce the chances of a system crashing |
| Delay | The team need to work and research about the past IT projects and see what the average delay and the reasons behind it is. |

### 12.1 Risk Register

Table 9: Register

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Rank | Risk | Category |
| R1 | 1 | Complexity Constraint | Team Risk |
| R2 | 2 | Software Management | Technical Risk |
| R3 | 3 | Delay | Scheduling Risk |
| R4 | 4 | Communication | Team Risk |
| R5 | 5 | Technical illiteracy risk | Technical risk |
| R6 | 6 | Usability | Team Risk |
| R7 | 7 | Crashing | Scheduling Risk |
| R8 | 8 | Hardware Incompatibility | Technical Risk |
| R9 | 9 | Privacy and Security | Data Security Risk |

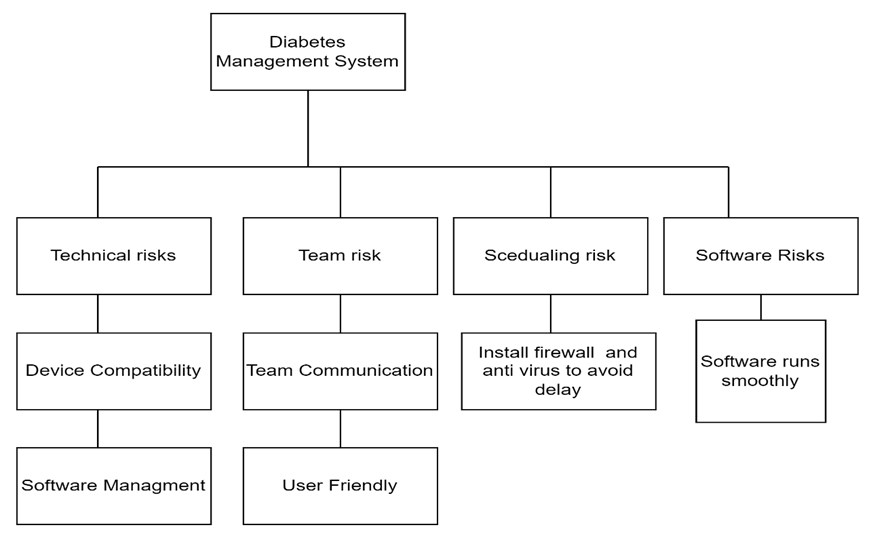
### 12.2 Risk Resolution

Table 10: Risk Resolution

|  |  |
| --- | --- |
| Problem | Contingency plan |
| Complexity Constraint | Team will be making extensive research about the project and use all the information gained |
| Software Management | The team will have a plan of how to manage the software and how to maintain it in case of any difficulties |
| Delay | The team will research about the average delay in an IT project usually is and use the knowledge to develop the system |
| Communication | The team will be able to communicate and elaborate about any problem they have and will be solved accordingly within the group regularly |
| Technical illiteracy Risk | The team will research on how to distribute and give knowledge to rural areas |
| Usability | The team will ensure that the software is user friendly so users at all ages will be able to use and manage themselves without any further assistance. |
| Crashing | The team will research about what is the main reasons for a software to crash and find ways to avoid it. |
| Hardware Incompatibility | Users will be notified about what is compatible with the system as sensors will be needed. |
| Privacy and Security | The CS students will work to make sure that there will be no unauthorized access to the software or any sort of breach to access confidential information about a patient. |

### 12.3 Risk Breakdown Structure

Figure 2: Risk Breakdown Structure



## 13. Open Issues

Type 2 Diabetes is the most common disease in the world, with patients having difficulty meeting their targets. A research article explained the challenges of Diabetes self-management. Participants were interviewed using a content analysis approach, the main issues that were discovered regarding self-management were the lack of family support, lack of resources, challenges to lifestyle modification, and mental health issues. Furthermore, according to the evidence provided, diabetes self-management is widely needed and is required to be more accessible, patients with mental health and financial challenges will have extra support from the team to help manage their disease (Whittemore et al, 2019).

Type 2 diabetes is a worldwide health epidemic with countries in the middle class as well as the low class affected excessively. According to a study people with diabetes are 17% more in rural areas than in urban areas, this can cause an issue as diabetes self-management system is less available than in high income urban countries. Adults that are diagnosed with T2D in Mexico lack immense knowledge and understanding about healthcare self-management systems, very few people understand the importance of self-management in Mexico as well as other rural areas (Whittemore et al, 2019).

In conclusion adults with T2D need to have more access to resources such as the diabetes management system to improve early diagnosis as well as to treat individuals with T2D globally.

# Requirements

## 1. Introduction

### Overview

Diabetes affects around 422 million people globally, with the majority residing in low- and middle-income countries, and diabetes is directly responsible for 1.5 million fatalities per year. Diabetes has been continuously growing in both the number of cases and the prevalence during the last few decades (Diabetes 2022). A greater understanding and awareness of the health condition is required to control type 2 diabetes. Unfortunately, there are still many people that don’t have enough knowledge about the condition and how to manage it in their daily lives.

This section contains the requirements for the DiabetesCare web application. It will concentrate on the design constraints, functionalities, and non-functionalities in order to understand the components and features that will be applied to the web application.

### Definitions

**Diabetes** - is a chronic condition that develops when blood glucose, often known as blood sugar, becomes excessively high.

**Machine Learning (ML)** - a subset of artificial intelligence (AI) that concentrates on using data and algorithms to mimic how people learn, progressively improving its accuracy.

**Deep Learning (DL)** - a sub-field of machine learning.

**Natural Language Processing (NLP)** - refers to systems capable of understanding language.

**Multi-factor Authentication (MFA)** – is a type of authentication that require users to give two or more verification factors in order to have access to an application.

**Encryption** - is a method of concealing information by transforming it into a random data.

**Terms of Service Agreement** - is a set of rules and guidelines that need to be followed by the users when using a website.

**Privacy Policy** - is a statement or legal document that describes how a company or website gathers and manages user and visitor data.

**OTP** - A One-Time Password is a secured PIN-code or a set of characters that is given to users through registered mobile phone or e-mail and it is only given for one single login attempt.

### Document Conventions

**Introduction** - This will give the person a brief introduction of the report. It will help them understand the goals, objectives and scope of the system. It also provides an overview of our system.

**General Design constraints** - This section will provide the characteristics and product environment for the users using the system.

**Nonfunctional requirement** - This will discuss the non-functional requirements of the system.

**System Features** - this will discuss features and functionalities of our system.

### Assumptions

* It is assumed that users must be connected to the internet in order to access the application.
* It is assumed that the web app will be of free of charge, therefore, all users can use it.
* It is assumed that the web app does not require downloading or installation.
* It is assumed that the user will have a social media.
* It is assumed that accessing the app will require the use of a laptop, a computer, and a mobile phone.

## 2. Design Constraints

### 2.1 User Characteristics

The app is directed towards people all around the world, including those from low-income societies, and those who have less education, and it is also aimed at those who are less aware of how to manage diabetes as well as the elderly, who may lack technological expertise. Furthermore, it also focuses on people who have busy lives and can’t spend much time researching diabetes management.

For the people that have less education, we’ve focused on making the language as simple as possible and provided a variety of images for better understanding. In addition to that, the user interface has been made simple, straightforward and easy to use for those who lack technical knowledge. The elderly, as well as certain racial and ethnic groups, are at a greater risk of developing type 2 diabetes. Few of these include the people of Alaska Native, American Indian, African American, Hispanic, and Asian or Pacific Islander origin, which are more vulnerable to developing diabetes (Goad, 2022).

Diabetes is not taken seriously in many parts of the world; people from rural communities are unable to care for it because many of them are unaware of how to manage it, but there are also people from more developed parts of the world who do not take it seriously because some of them are too lazy to research how to manage it, others find healthy meals unappealing, and so on. For example, in the UK, a study done in 2018 shows that many diabetic patients don’t take their condition very seriously, as evidenced by the fact that diabetes affects more people there than any other major health condition, and around 12.3 million people there are at risk of developing it (Coleman, 2018).

In order to deal with that, we’ve made our application easy for everyday use, we’ve provided a whole diabetic friendly lifestyle guide for the users as we not only provide a variety of personalized diabetic friendly meal options for them for breakfast, lunch and dinner, but also personalized workout routines and a section where they can journal about what's going on in their minds. Stress can affect the blood sugar levels; therefore, it is also something that needs to be managed. Our app provides this option, where depending on what the user writes down in the journal, they will be provided with certain activities that will help them overcome their negative emotion and relax to reduce stress levels.

### 2.2 Mandated Constraints

DiabetesCare is constricted to mobile applications for both Android and iOS since the app will be developed as a web application. Thus, users who use mobile phones may still use the app through web browser, such as Google Chrome, which they can install in their mobile app. Furthermore, the user must have a social media account for the app to examine their personal data more accurately. Without social media, the data collected by the app would be less accurate than it would be otherwise.

### 2.3 Potential System Evolution

Our web application can be used anywhere in developed countries and developing countries as well. Mainly we are trying to implement it in developing countries because they do not have access to healthcare readily compared to developed countries. However, since it is a health application privacy policies and standards need to be maintained for every country. The web applications need to have a proper firewall to protect the data of our patients so proper security measures should be implemented. We will also try to implement using Instagram and Facebook to capture data since it will be more accurate compared to Twitter. Once the website is running, we will try to implement a desktop application along with launching our diabetes care app on iOS and Android.

## 3. Nonfunctional Requirements

### 3.1 Usability Requirements

**Effective to use**

The effectiveness of a system refers to how good it is at performing what it is designed to do. This assists the user to become familiar with the web application's functionality. Furthermore, if the user receives clear feedback that the action, they took resulted in a certain outcome, they are far more inclined to understand and remember this action. To demonstrate, if a user enters the valid password, the app will send them to the home page; however, if the password is not valid, the app will display a warning message stating that the user entered the incorrect password.

**Safe to use**

When it comes to safety measures, DiabetesCare’s vision is to have a secure web application. We will ensure that when users will use the web app, their private information are protected and secure. As a result, data encryption will be implemented. Encryption will assist in encrypting user data to prevent it from anyone who is not authorized to access their account. In addition, the implementation of multi-factor authentication will be applied to the web application. An example would be the phone-based authentication where the users will receive one-time password (OTP). Hence, multi-factor authentication will support in the security layers of the web application.

**Learnability**

Nothing is worse for a user experience than an overloaded page with too many competing features, thus the app will employ a simple design for the interface. This is vital because the simpler the interface, the easier it is for users to immediately understand how the system operates. Additionally, the application ensures that consistency is maintained. Consistency is applied to the app's functionality such as icons, color, and button sizes. As a result, there will be a sense of familiarity to the app, allowing the user to quickly grasp the most basic functions.

**Good utility**

The utility of an app is determined by how easily the user can access its functions. This may be accomplished by grouping similar features together. For example, when users want to sign in, the log in and sign-up buttons will be near to each other, with no extra distracting features. Moreover, the user should never be in a scenario where there are no suitable options for carrying out the required task. As an example, the web app should provide back and next buttons to allow users to return to past actions.

### 3.2 Operational Requirements

In order to use the application, the user must have either computer desktop or laptop. DiabetesCare is a web application that may be accessed using a web browser such as Google Chrome, Mozilla Firefox, or Safari. As a result, using DiabetesCare will benefit users since they will have access to a wide range of devices to operate. Furthermore, in order to access their social media accounts, the user must be connected to the internet.

### 3.3 Performance Requirements

* Capacity: It must be able to fulfill the storage requirements of the system without crashing or affecting the speed.
* Response Time: The system should be able to deliver a response within a second at maximum after the request has been sent.
* Accuracy: It must be able to provide accurate results without any errors.
* Accessibility: It should be available 24 hours a day without any issues regarding bandwidth.

### 3.4 Security Requirements

The system must conduct security restrictions linked to access rights as part of the security criteria.

The following are some of these requirements:

* Unauthorized users should not be able to log in, and all user accounts in the system should be unique. User login acknowledgment should also be displayed.
* The system will only provide patients with the bare minimum of access or privileges.
* The system will separate tasks linked to granting access and use role-based access restrictions to prevent patients from accessing other patients’ data.
* Password processing will be done in a secure way by the system.
* Users will be appropriately invalidated by the system after several minutes of inactivity ensuring that no one else may impersonate them and utilize the system in their absence
* In order to comply with end-users’ privacy rules, the system will not leak any sensitive and personal information to outside companies
* The website's description, terms of service, and privacy policy will be carefully designed so the user can understand what sort of personal data is collected, utilized, and shared.
* The website will make sure it aligns with the country’s privacy laws and data protection legalization and will continuously be updated according to the changing web standards.

### 3.5 Safety Requirements

The system will not be prescribing any medications to the users, nor will it detect diabetes itself; instead it will just show their risk level and recommend a healthy lifestyle as its purpose is only to help the people who are already diabetic. It will enable them to understand and manage diabetes better by providing exercise and diet recommendations.

### 3.6 Legal Requirements

Users must agree to the terms and conditions after registering into the web app. It is a method of establishing rules and restrictions for users using the website (Privacy Policy vs Terms & Conditions 2022). Furthermore, because the app will apply machine learning to collect or use the user's personal information, a privacy policy will be implemented. The privacy policy is required by law and will help in the protection of users. Furthermore, DiabetesCare will not collect any private information about users, such as medical records. Instead, DiabetesCare will only collect data from the user's public data on their social media account.

### 3.7 Documentation and Training

Since the self-management diabetes app is for people all over the world, it will emphasize on the simplicity of use, the final web application will not require training to be utilized effectively and efficiently. The web application will be simple to use, and the user will not require any training or coaching. The application's official website will provide further information on how the application works.

### 3.8 External Interface

**User Interface**

The user interface should be designed to be clear, simple, and minimalistic. The features must be easy to understand since it is developed for people which also includes those who are more likely to not have much technical knowledge. Performing tasks must demand the least amount of effort from the user, and each page should serve a single purpose. The design must be consistent throughout the system. The colors must be soft and calming as they have an influence on the mood and wellbeing of the patient. Therefore, the colors used must convey a friendly and calming attitude and not pull unnecessary attention. In addition to that, color palette and font style must be consistent throughout and there must be enough contrast between text and colors in order to make it more readable. The fonts must have a clear and legible size and the text must be properly aligned to help the users understand better. The text should be well-organized and readable with adequate spacing between the characters. Buttons must be the correct size for the users to interact with them and placed far enough apart so that the users may interact with them without accidentally tapping them. The use of icons may be beneficial for the users to comprehend. The icons should be simple enough for users to be able to understand what they represent. Additionally, illustrations and photographs may also create a positive impression when using the diabetes management system and help provide better guidance about the information. To be aesthetically appealing, high-resolution photos must be used.

**Software Interface**

For this application, we will be using HTML and CSS to design the web pages. We will also use JSON text files to store our data and google analytics to continuously monitor the patients on our website. Our application will be an easy interface and will run both on desktops and on mobile phones. It will be easy to navigate the users from one web page to another but for the web application to work the user will need to have a stable Wi-Fi connection to connect and load the web pages.

## 4. System Features

### 4.1 Twitter Analysis

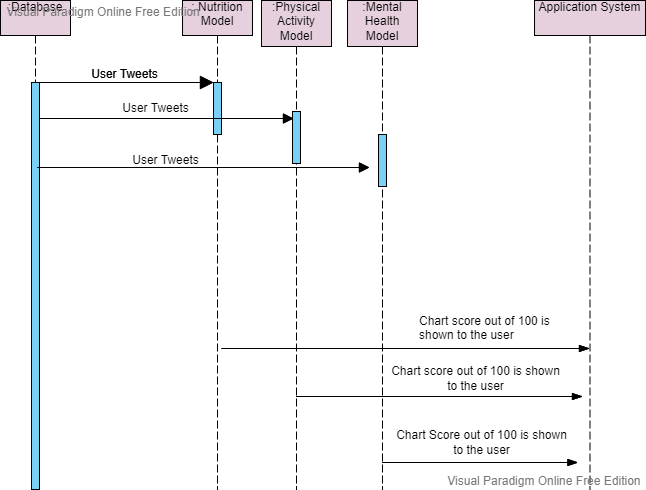
This is the number one priority requirement in our project, this is the technical part that will show how this app is unique and different. The application will be using machine learning, in specific NLP. The application will be capturing data from Twitter’s APIs and taking things such as tweets and retweets into our three machine learning models to get analyzed on the different criteria. The analysis is shown as chart score to the user from 0-100 where 0 is the worst a user can do and 100 is the best.

**Cost**: Free

**Risk**: Semi

**Value**: High

Figure : Sequence Diagram: <User Twitter Analysis (Mental, Nutrition, Physical)>



### 4.2 Feature: <Physical Activity Recommender System >

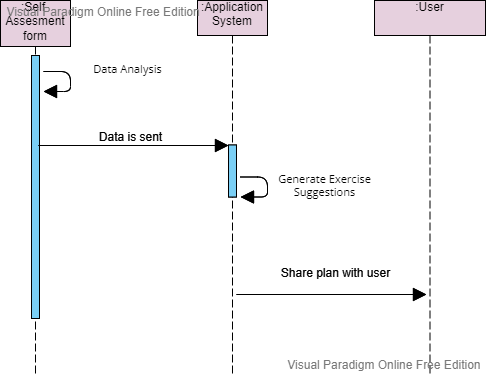
This feature depends on the user’s self-assessment data, it will check whether the user has access to equipment such as a gym, dumbbells, treadmill, or not. It will also check the user’s level in terms of physical activity (Beginner – Intermediate – Expert) to suggest appropriate workouts from the array of workouts included.

**Cost**: none

**Risk**: low

**Value**: high

Figure 4: Sequence Diagram: <Physical Activity Recommender System>



### 4.3 Feature: <Diet Recommender System>

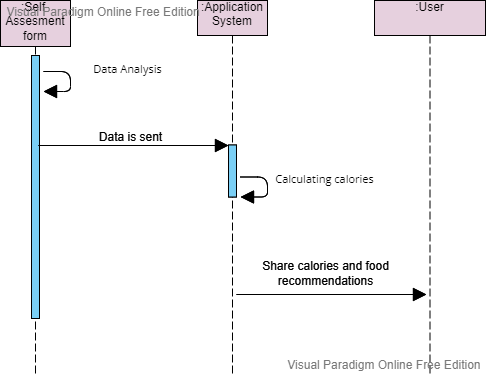
This feature depends on the user’s self-assessment data, it will take the user’s age, gender, weight and height into consideration in order to calculate the appropriate caloric intake the user should have. The user can then log meal from our array of recipes which will mention the caloric intake per meal, which gives users the ability to space out their calories whichever way they’d like to.

**Cost**: None

**Risk**: Low

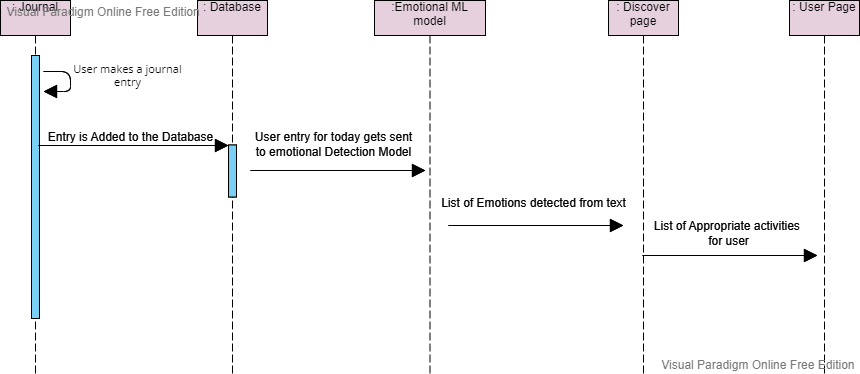
**Value**: High

Figure 5: Sequence Diagram: <Diet Recommender System >



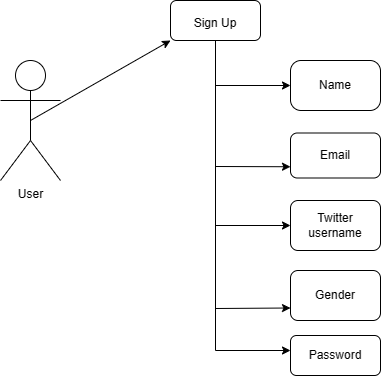
### 4.4 Feature: Mental Health Recommender System

Figure : Sequence Diagram for <Mental Health Recommender System>



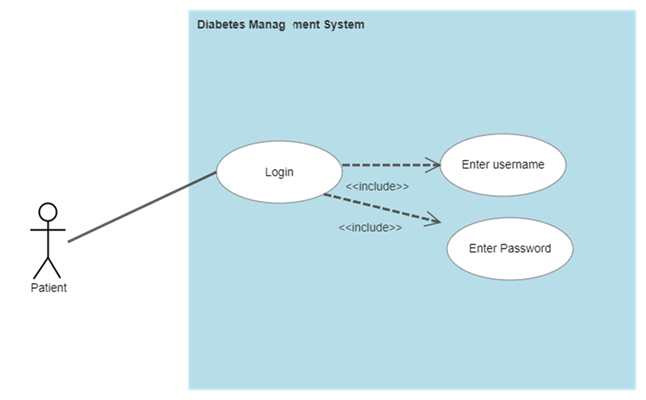
### 4.5 Feature: <Sign up>

Figure : Use Case Diagram: <Sign up>



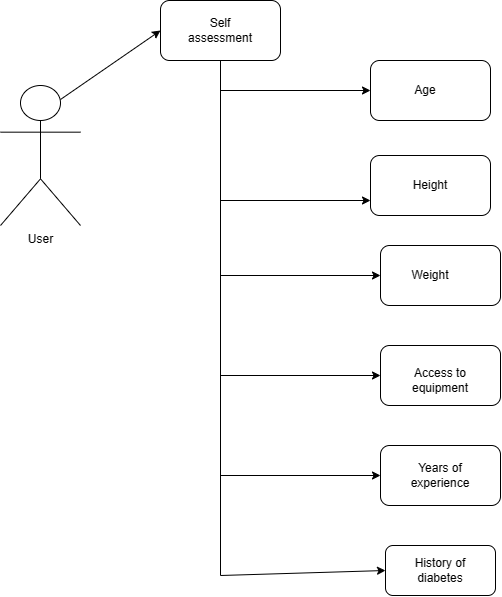
### 4.6 Feature: <Login>

Figure : Use Case Diagram: <Login>



### 4.7 Feature: <Self-Assessment>

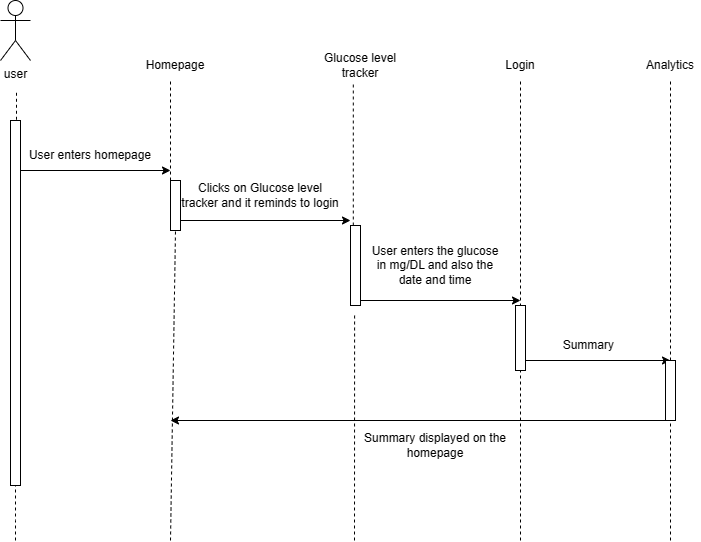
Figure 9: Use Case Diagram: <Self-Assessment>



### 4.8 Feature: <Glucose Level Tracker>

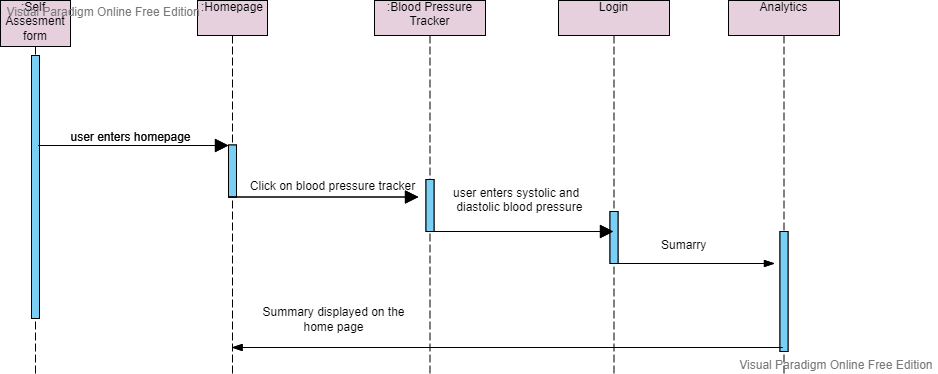
The glucose tracker allows users to record and track their blood glucose levels on a daily basis. When using the tracker, users will be prompted to enter their glucose readings and the time at which they were taken.

Figure 10: Sequence Diagram for <Glucose Level Tracker>



### 4.9 Feature: <Blood Pressure Tracker>

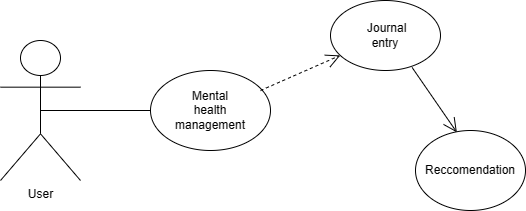
Figure : Sequence Diagram for <Blood Pressure Tracker>



### 4.10 Feature: <Journal Entry>

The journal entry feature allows users to document their thoughts, feelings, and experiences. It also enables the app to provide personalized recommendations based on the user's emotions and behaviors, helping them to manage their diabetes in a more holistic and effective manner. Users can access their journal entries anytime and edit or delete them as needed.

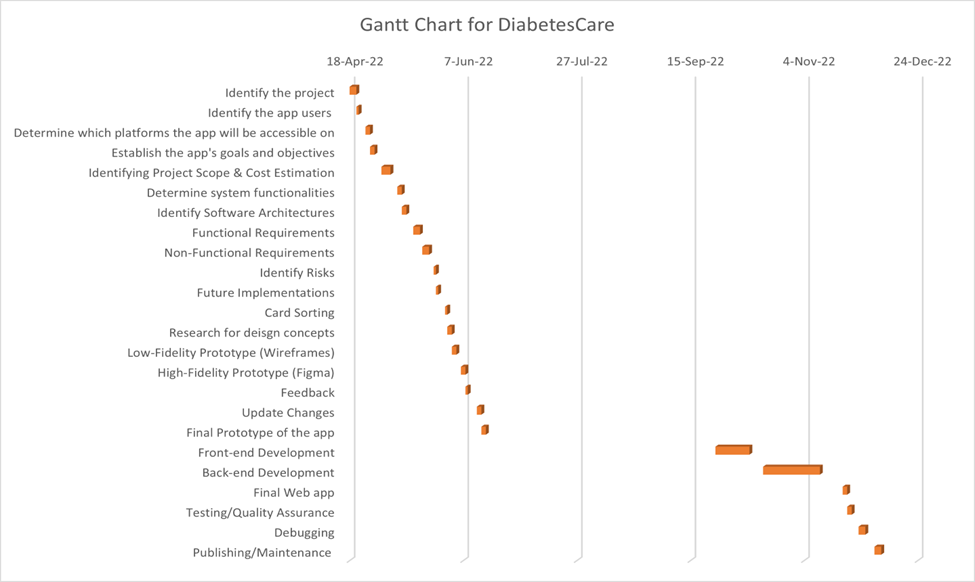
Figure 12: Use Case Diagram for Journal Entry



## 5. Project Matters

### 5.1 Preliminary Schedule

Figure 13: Gantt Chart: Preliminary Schedule



### 5.2 Preliminary Budget

The figures provided below were developed by researching similar projects and understanding how the money is distributed across each component, from strategy development to application development.

Table 11: Budget

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GANTT CHART (SPRING 2022 - AUTUMN 2022) | | | | |
| WORKING DAYS: MONDAY-FRIDAY | | | | |
| **TASK NAME** | **START DATE** | **END DATE** | **DURATION** | **ESTIMATE (AED)** |
| **Web App Strategy - 3 weeks** |  |  |  |  |
| Identify the project | 18-Apr-22 | 21-Apr-22 | 3 | 2000 |
| Identify the app users | 21-Apr-22 | 22-Apr-22 | 1 | 1000 |
| Determine which platforms the app will be accessible on | 25-Apr-22 | 27-Apr-22 | 2 | 4500 |
| Establish the app's goals and objectives | 27-Apr-22 | 29-Apr-22 | 2 | 12000 |
| Identifying Project Scope & Cost Estimation | 2-May-22 | 6-May-22 | 4 | 7000 |
| **Define Requirements -3 weeks** |  |  |  |  |
| Determine system functionalities | 9-May-22 | 11-May-22 | 2 | 4000 |
| Identify Software Architectures | 11-May-22 | 13-May-22 | 2 | 5000 |
| Functional Requirements | 16-May-22 | 19-May-22 | 3 | 5500 |
| Non-Functional Requirements | 20-May-22 | 25-May-22 | 3 | 2000 |
| Identify Risks | 25-May-22 | 26-May-22 | 1 | 2500 |
| Future Implementations | 26-May-22 | 27-May-22 | 1 | 2000 |
| **Design - 3 weeks** |  |  |  |  |
| Card Sorting | 30-May-22 | 31-May-22 | 1 | 3000 |
| Research for design concepts | 31-May-22 | 2-Jun-22 | 2 | 1500 |
| Low-Fidelity Prototype (Wireframes) | 2-Jun-22 | 4-Jun-22 | 2 | 15000 |
| High-Fidelity Prototype (Figma) | 6-Jun-22 | 8-Jun-22 | 2 | 22500 |
| Feedback | 8-Jun-22 | 9-Jun-22 | 1 | 1000 |
| Update Changes | 13-Jun-22 | 15-Jun-22 | 2 | 15250 |
| Final Prototype of the app | 15-Jun-22 | 17-Jun-22 | 2 | 26000 |
| **Implementation - 10 Weeks** |  |  |  |  |
| Front-end Development | 26-Sep-22 | 14-Oct-22 | 15 | 27000 |
| Back-end Development | 17-Oct-22 | 18-Nov-22 | 25 | 35000 |
| Final Web app | 21-Nov-22 | 23-Nov-22 | 2 | 10000 |
| Testing/Quality Assurance | 23-Nov-22 | 25-Nov-22 | 2 | 15000 |
| Debugging | 28-Nov-22 | 1-Dec-22 | 3 | 20000 |
| Publishing/Maintenance | 5-Dec-22 | 7-Dec-22 | 3 | 20500 |
|  |  |  | Total | 259250 |

The total estimate for the project came to AED 259,250.

# Design

## 1. Introduction

The purpose of this document is to provide a comprehensive overview of the design of Diabetes Care, an application that uses machine learning to help people with pre-diabetes or type 2 diabetes manage their condition.

## 2. System Overview

DiabetesCare is an application that makes use of machine learning to calculate a person’s risk in either getting diabetes or worsening his health/diabetes. The application is calculating this risk based on the user’s twitter activity, which is constantly being captured. The application runs on a ngrok server, which is being ran on a flask app (which allows us to connect the back-end code written in python to the front-end code). The user’s important data is saved on a mongo dB server, which can be read by the algorithm and updated by the user at any time to give the most accurate results. The application will focus on dietary habits and needs first then will move on to exercise and mental health habits and needs.

## 3. Design Map

**Target Users**: The users who we targeted are those who are pre-diabetic or those with type 2 diabetes. They are who we shaped this application around, giving them easy tools and ways to be able to track their diabetes condition.

**Backend**: The backend includes multiple libraries that work together to make this application possible. For example, the “tweepy” library is what allows us to scrape the user’s twitter data. “Pymongo” allows us to connect our application to a database. “Flask” gives us a way to connect the backend to the frontend. “Ngrok” allows us to have the application running on a server which can be accessible to anyone. “TensorFlow” and “scklearnkit” will give us the tools needed to build and train machine learning models.

**Frontend**: The frontend includes HTML, CSS, and JavaScript, which all work together to give the user an interface to interact with. The forms are what allows the application to fetch information inputted by the user to the back end which then sends it to the database.

## 4. Supporting Materials

**Figma**: The wireframe design of the web app was created using Figma. Furthermore, before coding the front-end, we used Figma to create a design and flow for the web app's interface.

**Draw.io**: The flow charts and diagrams were used using draw.io.

## 5. Definitions and Acronyms

Table 12: Definitions

|  |  |
| --- | --- |
| Words | Definitions |
| API | Application Programming Interface is a set of programming code that allows one software product and another to communicate and exchange data with each other. Allowing them to interact with each other without any user intervention. |
| Twitter API | The Twitter API provides unique and advanced programmatic access to Twitter. The API allows you to find, access, and interact with data from twitter such as tweets, direct messages, lists, users and trends. |
| Tweepy | An open-source Python package that provides a very convenient way to access the Twitter API with Python. |
| Machine Learning (ML) | A subset of artificial intelligence (AI) that concentrates on using data and algorithms to mimic how people learn, progressively improving its accuracy. |
| Ngrok | “Ngrok” is a globally distributed reverse proxy fronting your web services running in any cloud or private network, or your machine. This is one of the main libraries which helps build this app. |
| Flask | Flask is a micro web framework written in Python. This is one of the main libraries which helps build this app. |

## 6. Design Considerations

### 6. 2 Assumptions

**User Account**

Users must register an account and approve the consent form before they can use our application. Its purpose is to authenticate the users in order to secure their private information and provide them with a personalized experience.

**Users Activity on social media (Twitter)**

A user must be active on Twitter in order for program to be able to give weekly or monthly suggestions to the user based on their tweets, retweets, likes, following etc. If the software cannot detect any activity on their social media, it is impossible to deliver any sort of suggestions of managing their diabetes.

**User Analysis (BP&Glucose)**

Users must manually enter their data in order to track their glucose and blood pressure. Users will receive no analysis results if they do not enter their data as they won’t be any data given to the system.

**Web Application**

· The application must be able to run autonomously, there should be no need for a programmer to interact with the backend for this application to work.

· The application should be able to read and write into the database at any time depending on user needs.

### 6.3 Constraints

**Internet Connection**

Users are required to have access to the internet in order to use the web application as it allows them to connect, send and receive data. Without a proper internet connection, users won’t be able to send their data to the web app, nor connect to twitter to receive suitable results on the web application.

**Account Availability**

For the application to work, the user must have a Twitter account. They must have past data on their Twitter account that will help us in monitoring their activities. To extract useful information, essential data on food and diet must be posted.

**Permissions**

The application requires access to a user’s Social Media data such as Twitter. This can lead the user to object to any of the access rights. The program won't function as intended or be useful to the user if any of the aforementioned permissions is denied.

**Input Quality**

DiabetiesCare uses the user’s Twitter Data to give suggestions on how a user can manage their Health. This would require information that is extracted to not be misinterpreted. For example, sarcastic comments would not be analyzed effectively as information given would not be legitimate, which will lead for the user to not be able to fully experience the application’s features.

**Data Privacy**

DiabetesCare uses Twitters API and Machine learning to assess and calculate the user’s Health. This would need to store the user’s personal tweets for insights and feedback. Since data privacy is a sensitive topic, storing these user data is important for future research.

**Web Application**

· The maximum number of tweet extractions allowed by tweepy per account is limited to 5000.

· A powerful PC/laptop is required to keep this running 24/7.

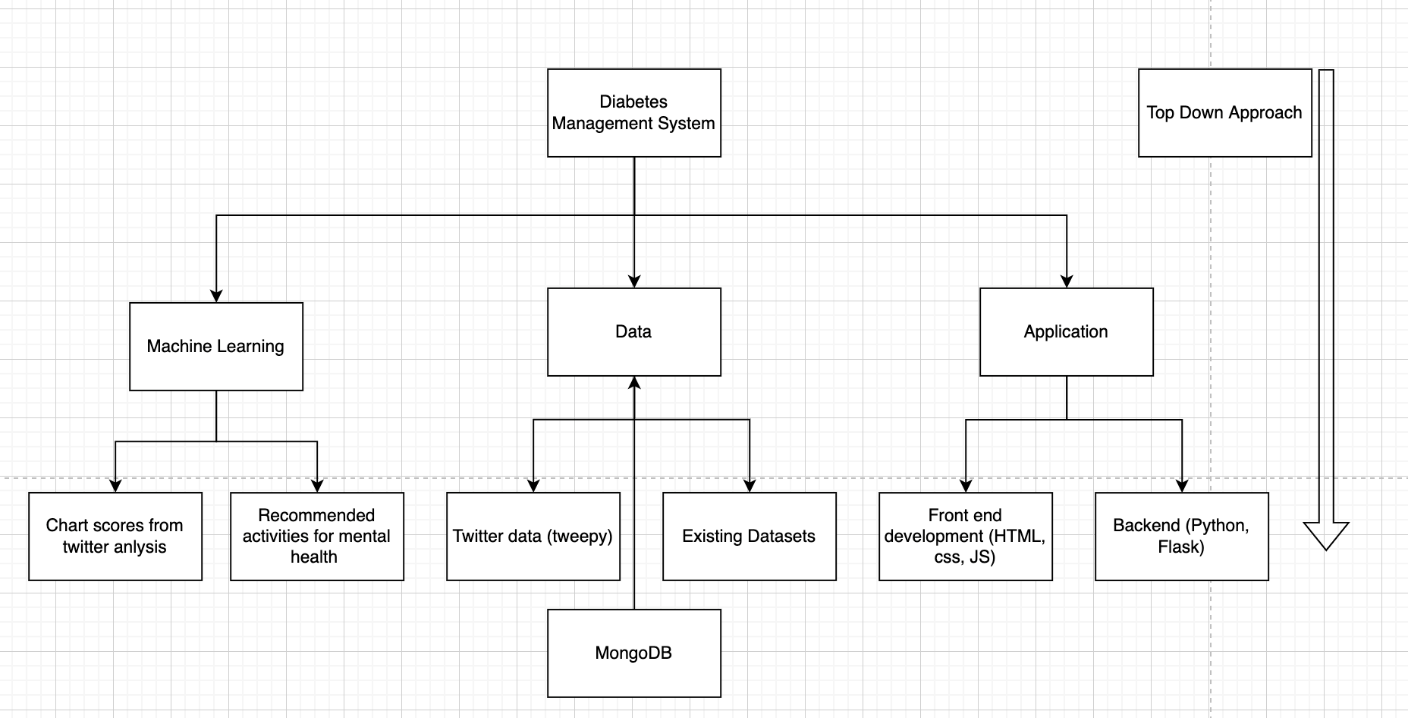
### 

### 6.3 System Environment

The hardware (laptop/tablet/phone) is what the user will need to interact with the interface (software’s frontend), the interface is the tool which will allow the user to make use of the functionalities (software’s backend), which is constantly working in the background simultaneously with the database (pymongo/mongo dB), the twitter data scraping tools (tweepy), the API (Flask) and the server (ngrok).

### 6.4 Design Methodology

Figure 14: Design Methodology on Diabetes Management



The top-down design technique is one of the design techniques used by our system. The goal of this process was to breakdown the entire concept into smaller, less complex components from the initial overall problem. char

This design starts from the end solution and works backwards. The overall idea was presented and as soon as these base elements were identified, the tasks were broken up into very few extensive subtasks.

The overall problem was too complex to solve in just a few steps. So we have divided the overall problem into smaller manageable pieces, solve each one of them and then finally bring everything back together.

### 6.5 Risks and Volatile Areas

**Database size insufficiency:**

​​Database size might reach a limit if application is used on a larger number of users (free mongo dB account).

**Insufficient data:**

One of the risks for this application could be insufficient data on the twitter account. Them not having a twitter account which will lead to inaccurate data.

**Internet Connectivity:**

One of the main risks in order for the user to experience the features of the program is internet connection, without internet connection there could be a risk in managing a user’s health.

**Diagnostic errors:**

If users do not have enough data, their diabetes results will not be accurate. This can affect the lifestyles of the potential users.

**Weak Password**

By creating a weak password, the user increases the risk that their account will be easily hacked, giving hackers access to their account.

## 7. Architecture

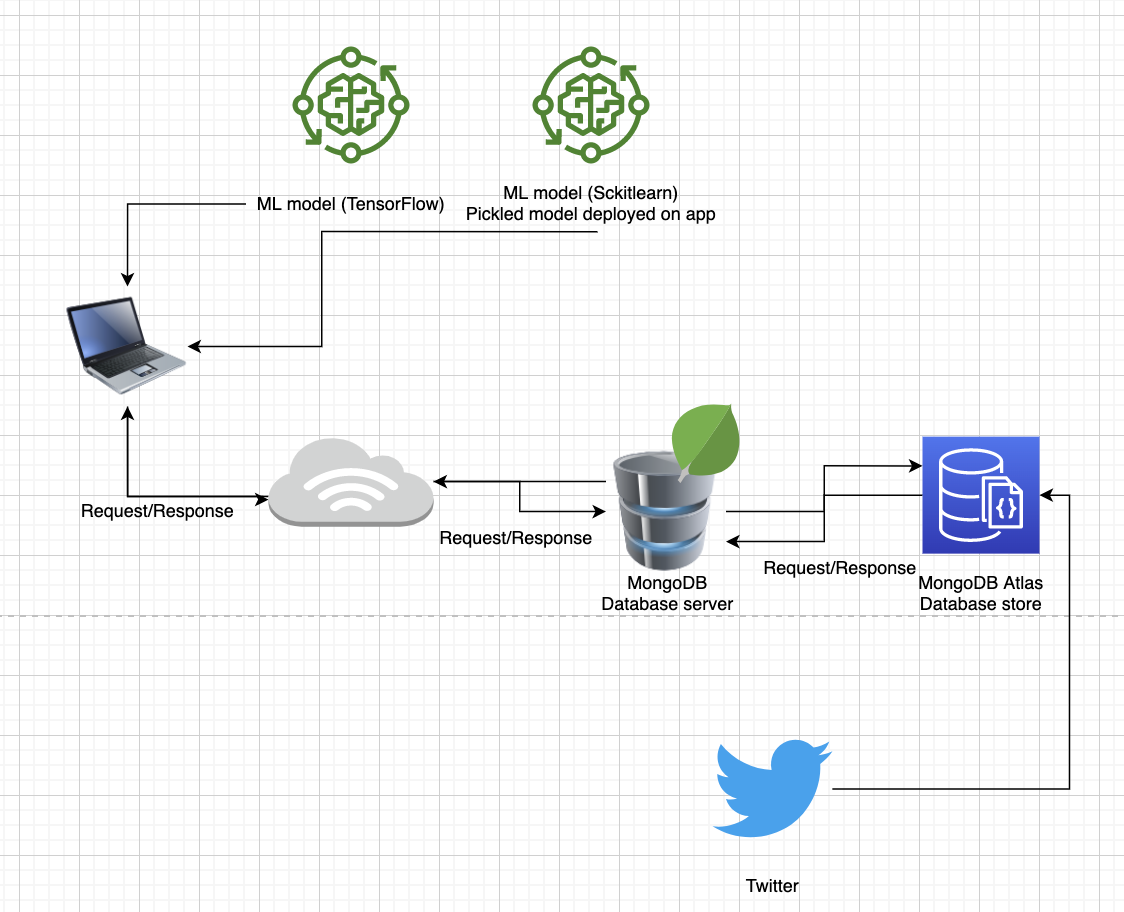
### 7.1 System Architecture

DiabetesCare's System Architecture is illustrated in the flow chart below. To start, the user will need to connect to a mobile or computer device in order to access our web application. To access DiabetesCare, the user must first register. It will then send a request to the web server, and the information will be saved in the database.

DiabetesCare can access the user's data and fetch data from Twitter API by connecting through the user's social media account, which is twitter. DiabetesCare will be able to retrieve data from the user's Twitter account through Twitter APIs. It will then analyze the extracted data using NLP (Natural Language Processing) and capture key phrases and trends that we can potentially apply in our next phase of implementations using DL (Deep Learning).

Following that, the algorithm will examine the user's data to determine whether or not the user's data are leading to diabetes or a healthy lifestyle. Additionally, all user activity, such as liking tweets, retweets, comments, follow hashtags, and following twitter handles, will be tracked. Depending on the user's statistics, it will show if the user is normal, pre-diabetic, or diabetic. DiabetesCare would then recommend a healthy lifestyle to the user to help them prevent or manage their diabetes. All of these results will display on web application. Furthermore, all of user’s information will be stored and saved in the database.

Figure 15: System Architecture of DiabetesCare



**Rationale:**

· Registering using the user’s twitter handle will help DiabetesCare in collecting the user’s data.

· Each user will have a unique account identification in the database, which will allow each user to be identified and interacted with.

### 7.2 Network Architecture

The diagram shown below depicts how the different parts of the network, which include the connections, the devices and the data flow interact with each other in order to make the system work.

Figure 16: Network Architecture of DiabetesCare



**Rationale:**

· Through twitter API, data gets captured from the users’ twitter accounts which then passes on to the machine learning

· Machine learning is used to help produce more accurate results, suggestions and predictions on the web application without the need of any human intervention.

## 8. Low Level Design

### 8.1 Module

Figure 17: Web Application flowchart

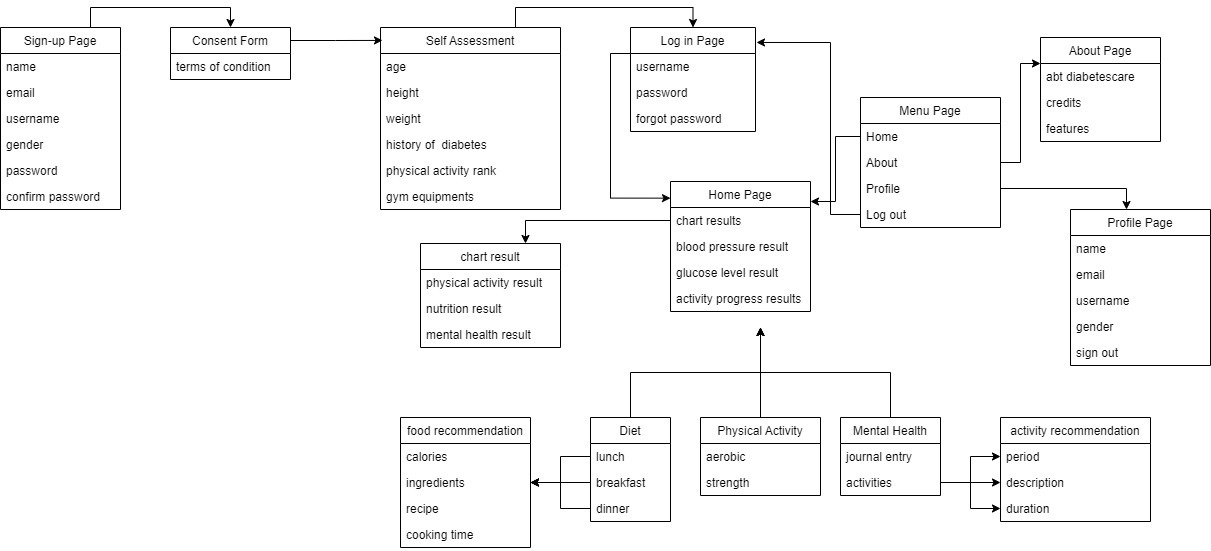


Figure 18: Wireframe Design



## 9. User Interface Design

### 9.1 Application Control

**Navigation bar**

Our web application will include a navigation bar on all pages and a consistent navigation scheme. On a larger screen, such as a computer desktop, the navigation bar will be displayed on the header. While mobile users will see the navigation bar on the sidebar, this will be help users to instantly access the most essential pages. If a website's navigation is straightforward and easy to use, users will know exactly where to go.

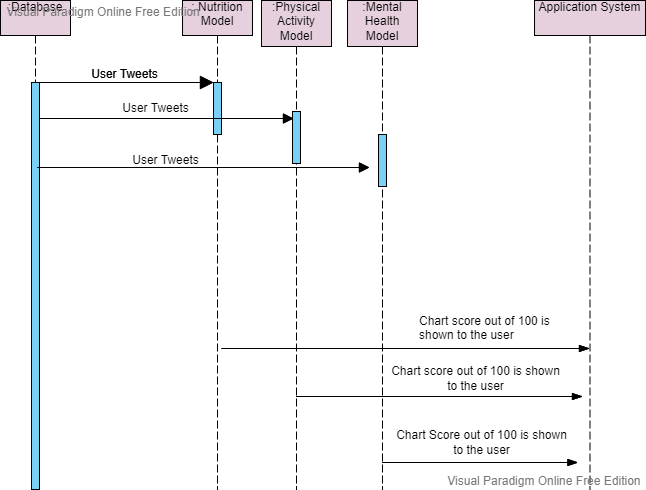
**Layout**

The layout will remain consistent throughout the online application. Aside from the forms, each page will include a header with a brief description of what the page is about and a relevant image. The functions that the given page offers will be situated below the header. The colors will also remain the same, which are white, light blue, and blue. The form pages will be similar as well, with the text in the center.

## 10. High Level Design

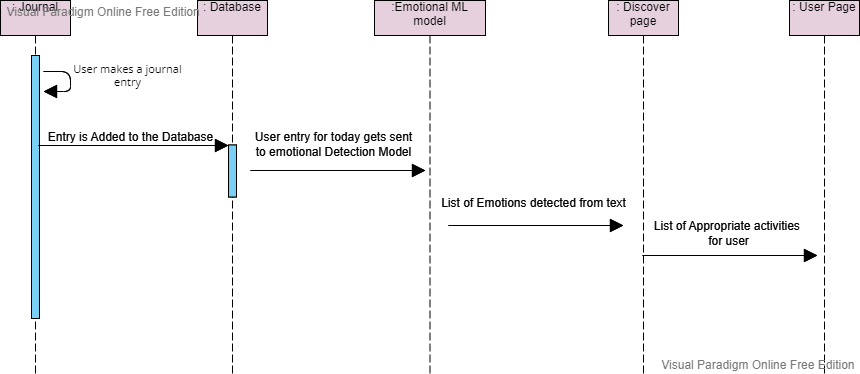
### 10.1 User Twitter Analysis (Mental, Nutrition, Physical)

Figure : Sequence Diagram: Twitter Analysis



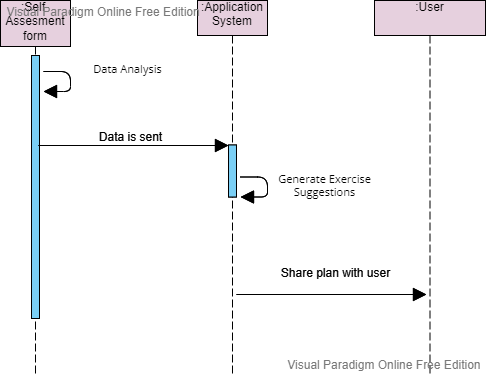
### 10.2 Mental Health Recommender System

Figure : Sequence Diagram: Mental Health



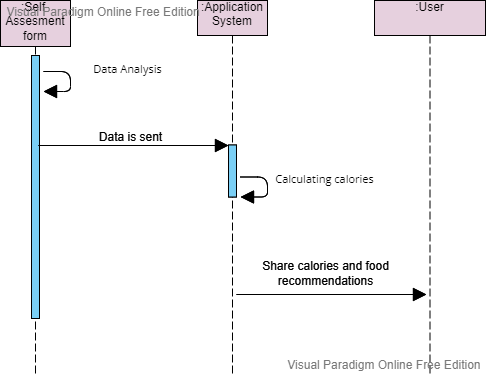
### 10.3 Physical Activity Recommender System

Figure : Sequence Diagram: Physical Activity



### 10.4 Diet Recommender System

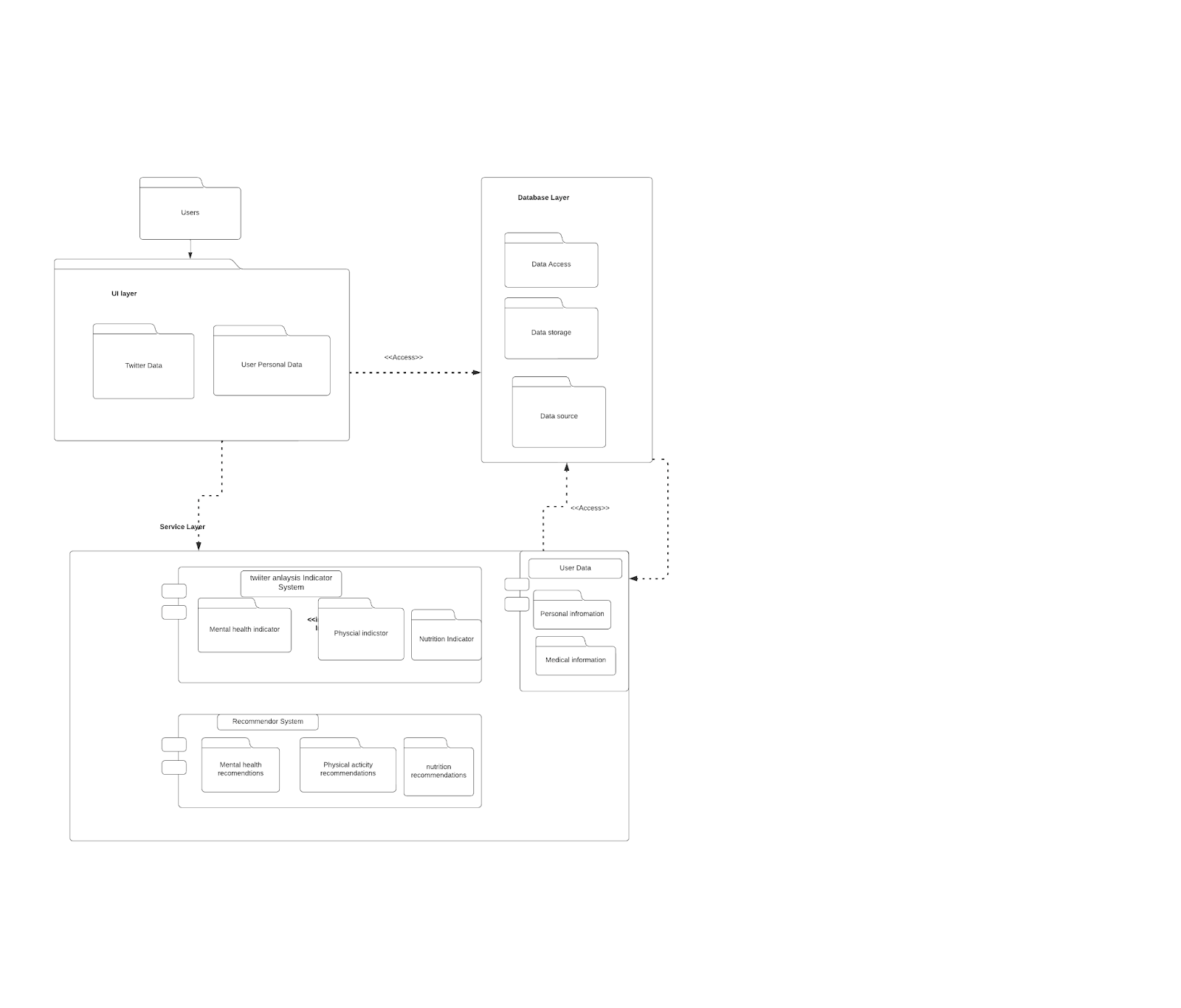
Figure : Sequence Diagram: Diet



### 

### 10.5 Package Diagram

Figure 23: Package Diagram



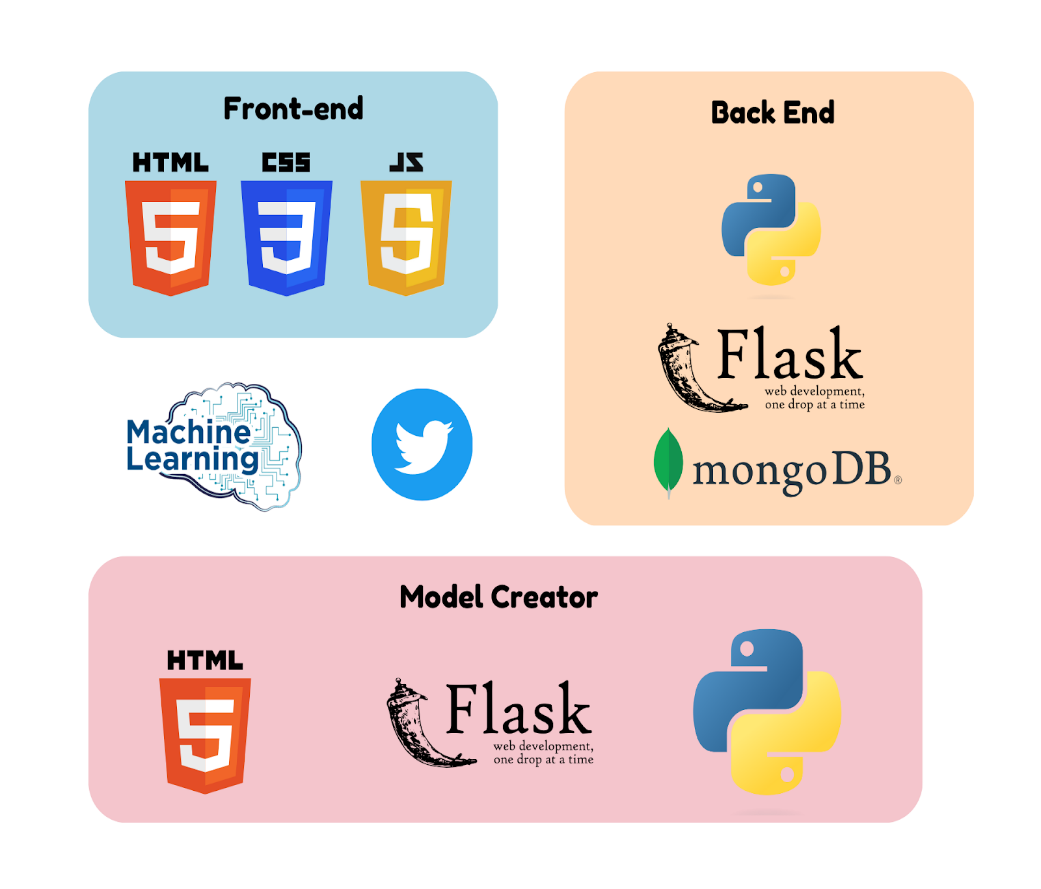
### 10.6 Class Diagram

Figure 24: Class Diagram

#### 

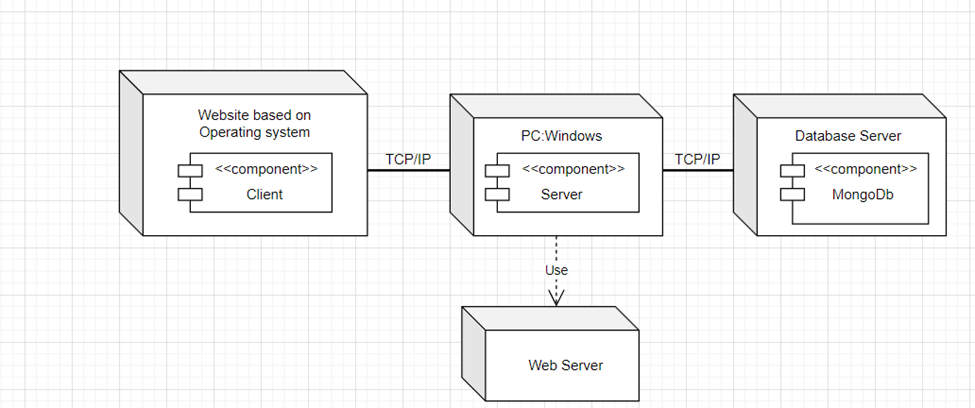
### 10.7 Conceptual Diagram

Figure 25: Conceptual Diagram



### 10.8 Deployment Diagram

Figure 26: Deployment Diagram



The Deployment Diagram is used to illustrate the system's operating environment. The website application will be written in HTML, CSS and JavaScript and will be available for windows. Flask will be used to create our Application Programming Interface (API). Moving on, because of our knowledge of the DB Language, we will create our Database Instance using MongoDB. Our twitter analysis will be done using Python

## 11. High-Fidelity Design (Screen)

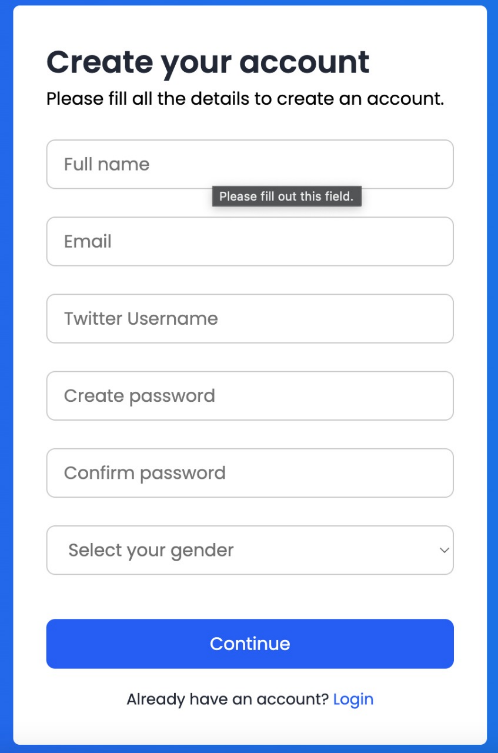
Figure 27: Login



**Description**

If users already have a registered account, they can log in using the username and password they provided when they initially signed up.

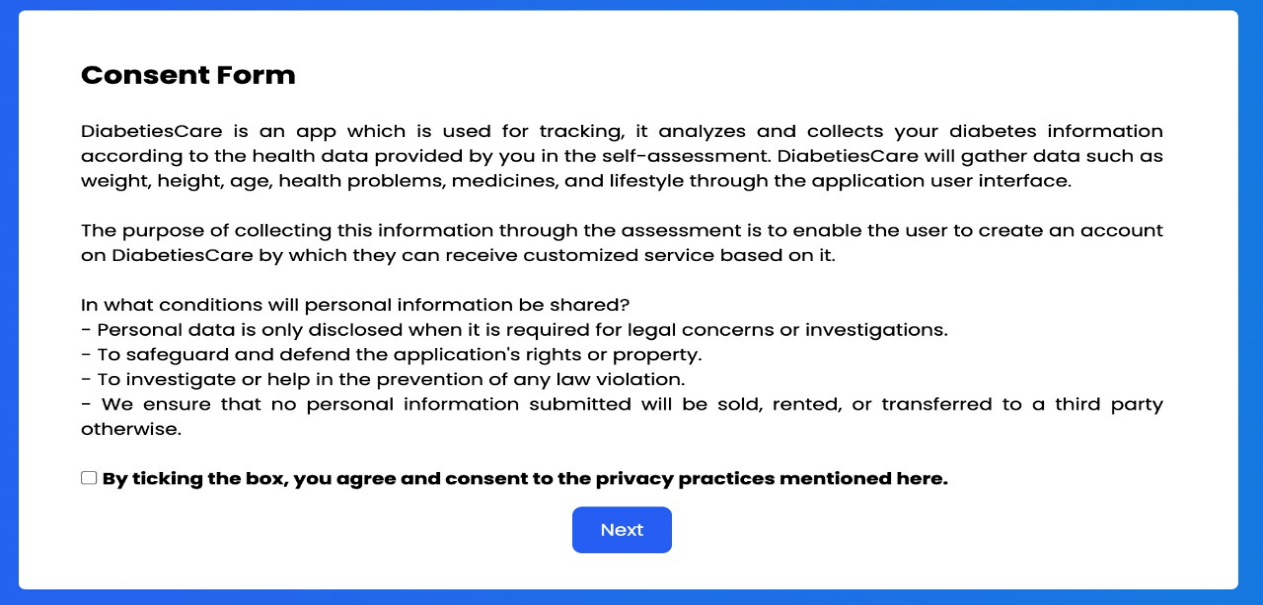
Figure 28: Sign up



**Description**

When users use our web application for the first time, they must sign up in order to create an account. To access the app's features, users must enter their name, email, username, password, and gender.

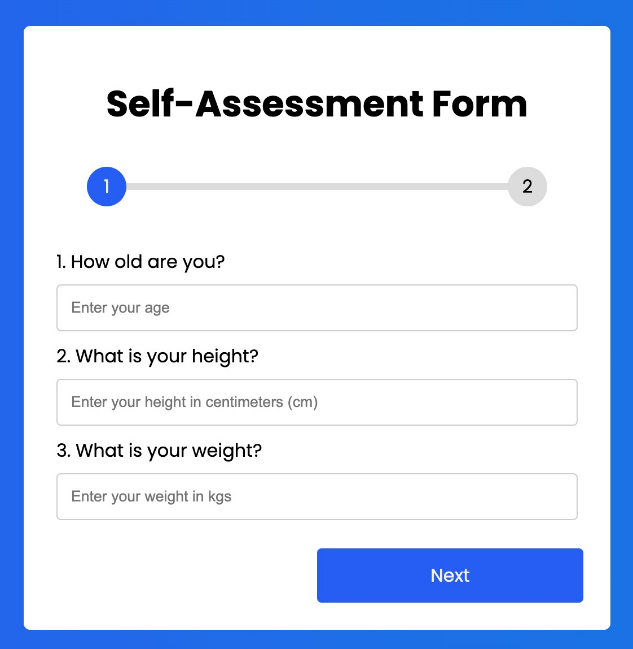
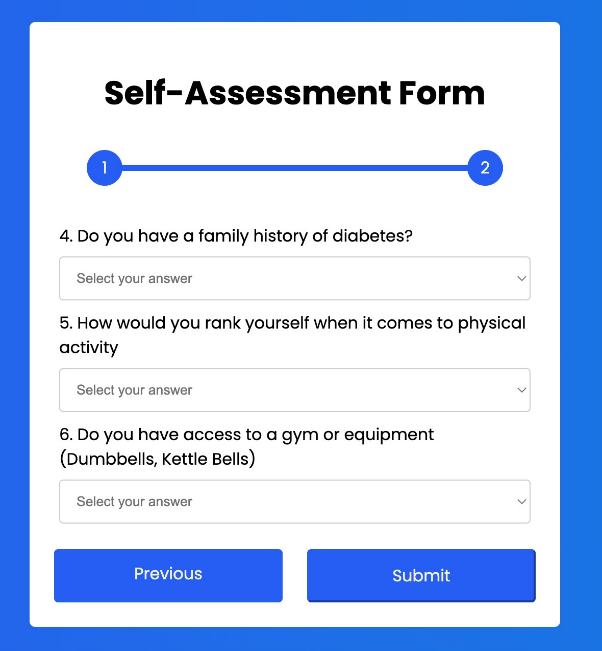
Figure 29: Consent Form



**Description**

This page is the consent page, this is where users will be provided information about the web application, its purpose and what it will do. It also informs the users about how it will be using the data that they provide and how it intends on protecting it. The users give permission by ticking the box.

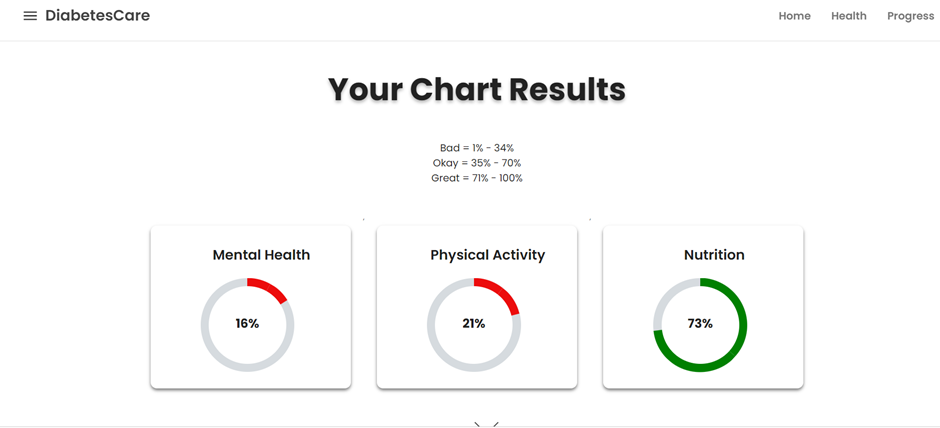
Figure 30: Self-Assessment

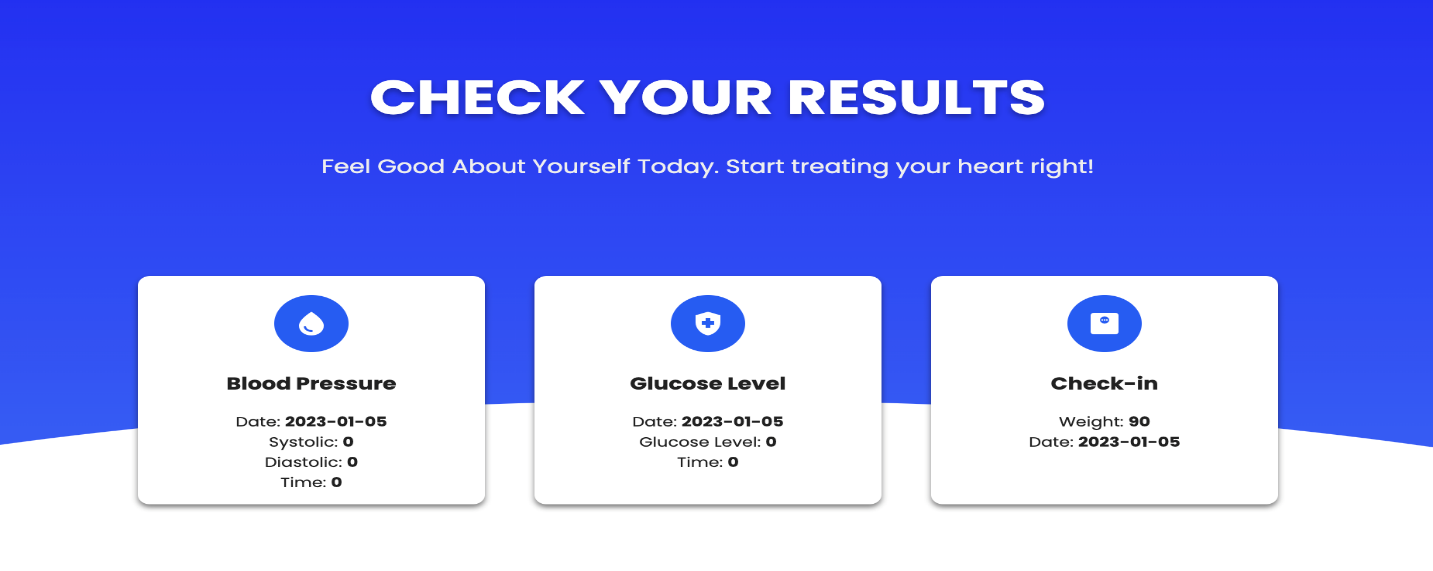
 

**Description**

This page is a self-assessment questionnaire for users to provide information about themselves and their health status. This will allow the web application to have a basic understanding of a new user's profile in order to provide more tailored recommendations for improving their lifestyle.

Figure 31: Home Page





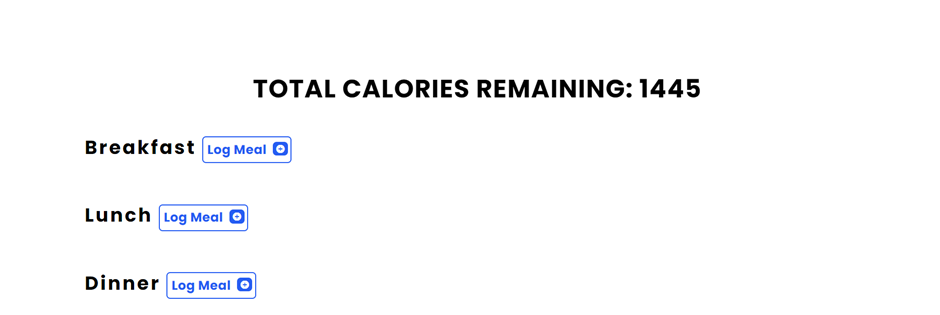


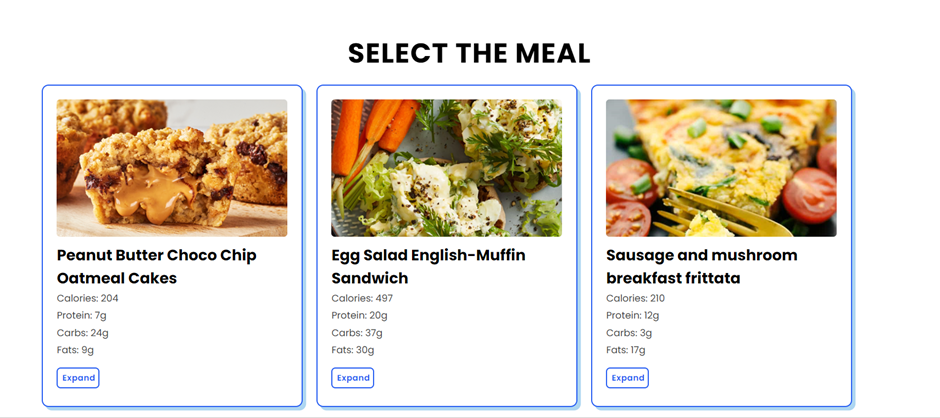
**Description**

The home page is designed to provide users with an overview of their findings on mental health, physical activity, and nutrition based on their Twitter activity. It includes a chart that displays this information visually, allowing users to easily track their progress and identify areas for improvement. The page is intended to be a central hub for users to access the various features and tools provided by the app, such as the ability to track and analyze their glucose and blood pressure data, as well as being able to update their weight so the application can update their caloric intake required. Additionally, it includes a progress section that allows users to track their progress in managing their diabetes. It also includes buttons for mental health, diet, and physical activity, which when pressed will take the user to a designated page for each respective topic. These pages provide users with more detailed information and tools related to each aspect of managing their diabetes, such as tips and resources for improving mental health, recommendations for healthy eating, and ideas for staying active. By providing these resources, the app aims to help users make informed decisions about their care and improve their overall health.

Figure 32: Diet Page



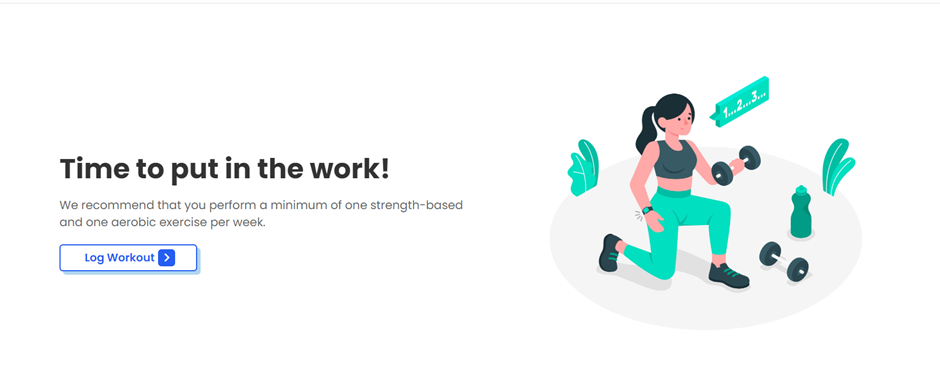
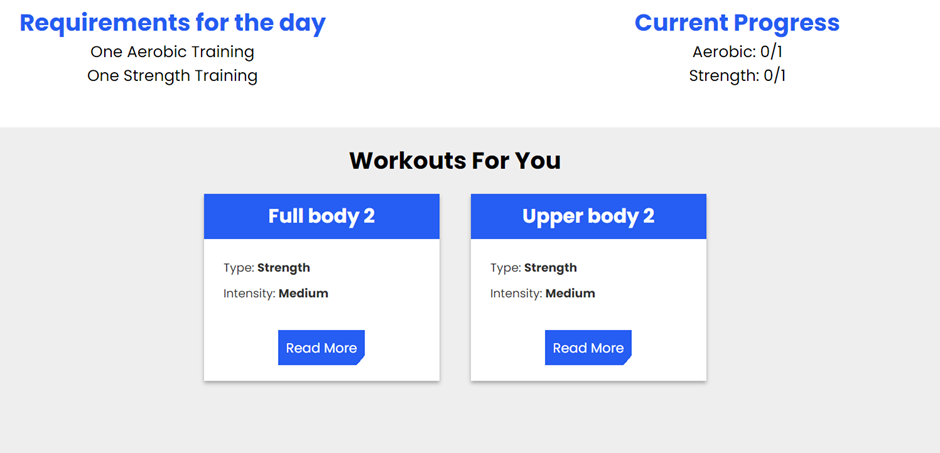




**Description**

The diet page help users make healthy choices about their nutrition and manage their calorie intake. The calorie intake is determined by the person's age, height, and weight, and a defined calorie is given to the user.  The individual is given recommendations on what to eat in order to maintain their calorie intake, as well as instructions on how to prepare the meal if the user does not know how.

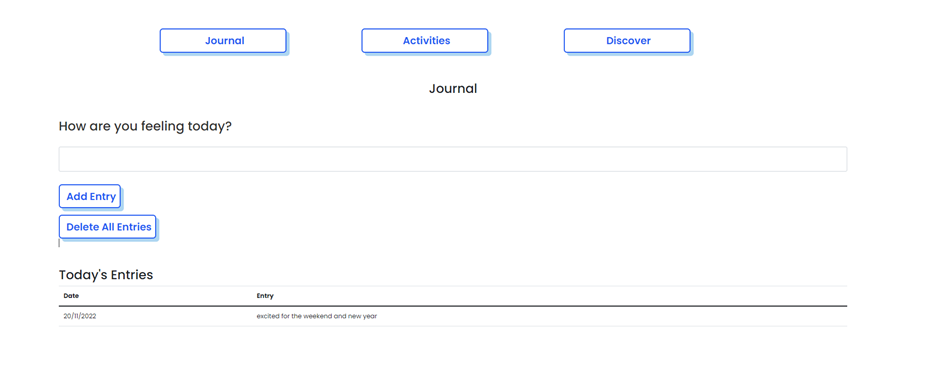
Figure 33: Physical Activity Page

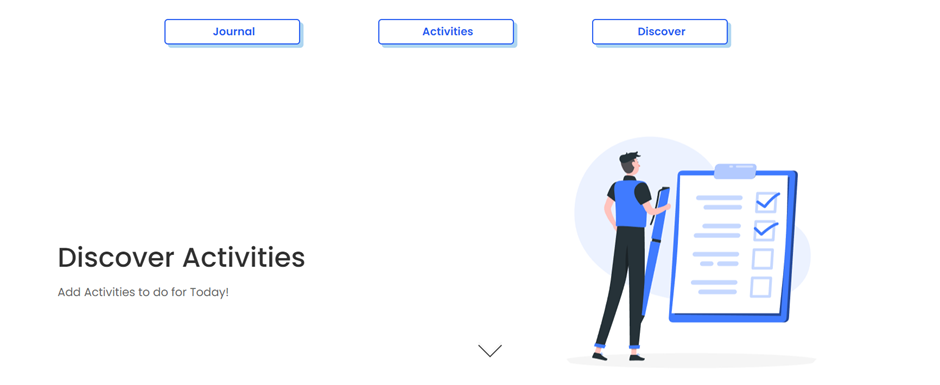
 

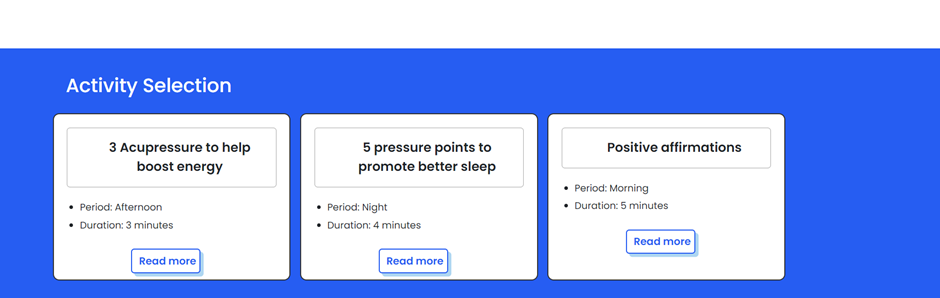
**Description:**

The physical activity page is designed to help users incorporate exercise into their routine in a safe and effective manner. It provides users with a workout plan based on whether they have access to equipment and their fitness level. Providing users with appropriate workouts and guidance on how to do movements, the physical activity page aims to help users improve their physical health and manage their diabetes more effectively.

Figure 34: Mental Health Page







**Description**:

The mental health page help users prioritize and maintain their mental well-being. It includes a self-journal feature that allows users to record their thoughts and feelings. A machine learning model goes through each entry to detect emotion, based on that the page will recommend activities to do each day to improve their mental health and help improve their mood. Additionally, there is a separate page that shows the user how many of these activities they have completed.

# Implementation

## 1. Recommender Systems

### Mental Health

The Mental health recommender system has two main functionalities, a journal where the user can put down what’s on their mind, what they must do throughout the day, how they feel, and based on that the system will do the second functionality which is the actual recommendation.

Once the user has at least one journal entry an algorithm which is shown below will detect one of 5 emotions: Joy, Sadness, Fear, Anger, and Disgust. The user can check which activity fits his time and schedule, do it and log it into the system.

Figure 35: Mental Health Data

The data comes from a dataset (the figure below) that took chunks of text and matched It with an emotion (like the ones mentioned above).

Application, table

Description automatically generated

Figure 36: ML Model

This was an NLP text classification model which was creating using the Scikit-learn library in python, as it contains all the required algorithms for classifications. The data was preprocessed with the help of the following libraries: nltk (used to download stop words, as well as tokenize the words), re (used to access regular expressions we needed to get rid of).

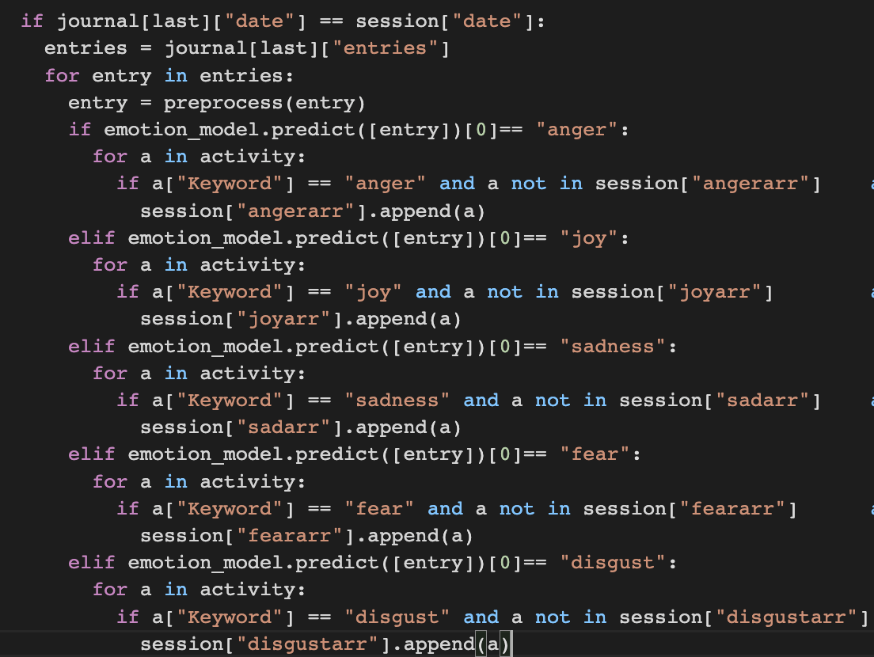
Once data was preprocessed the model was trained use the clean data and the features (emotions).

Graphical user interface, text

Description automatically generated

Figure 37: Algorithm

The below code shows how it works, the system checks the user’s journal entries from the current date, and the prediction model check which labels is predicted for each entry.



### 1.2 Physical Activity

The Physical Activity recommender system is a simple rule-based recommender system. Initially when the user signs up, during the self-assessment form they are asked about their level of experience (beginner, intermediate, expert) in training, and if they have access to equipment or not. Based on the answer the recommender system will show that their options when they decide to log a workout

Figure 38: Physical Activity Data

Text

Description automatically generated

### 1.3 Nutrition

The Nutrition recommender system is a simple rule-based recommender system. Initially when the user signs up, during the self-assessment form they are asked about their gender, age, height, and weight, which are all the factors required to be able to calculate their required calories to lose weight assuming they are following the training plan.

For women, it's: 655.1 + (9.563 \* weight in kg) + (1.850 \* height in cm) - (4.676 \* age). For men, the formula is: 66.5 + (13.75 \* weight in kg) + (5.003 \* height in cm) - (6.75 \* age)

## 2. Twitter Analysis

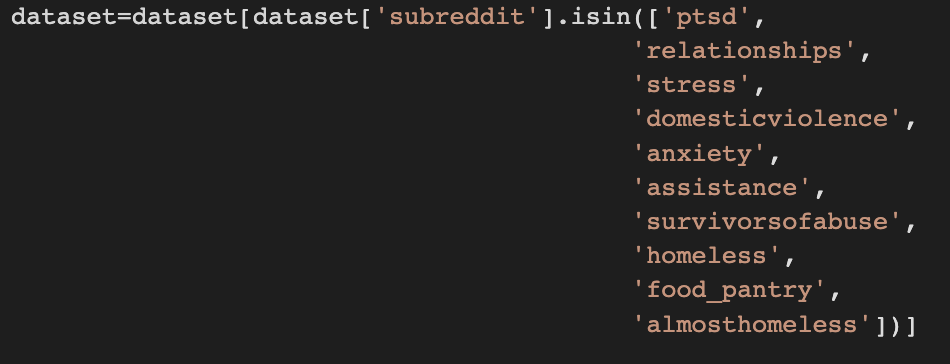
The twitter analysis models were trained using TensorFlow, a library on python which is used to build and train machine learning models. Each model had their own dataset.

The twitter analysis is done once a day open the user logging in for the first time, allowing the user to see their day-to-day progress, as seeing it multiple times a day can get overwhelming or maybe sometimes demotivating for the user.

### Mental Health Model

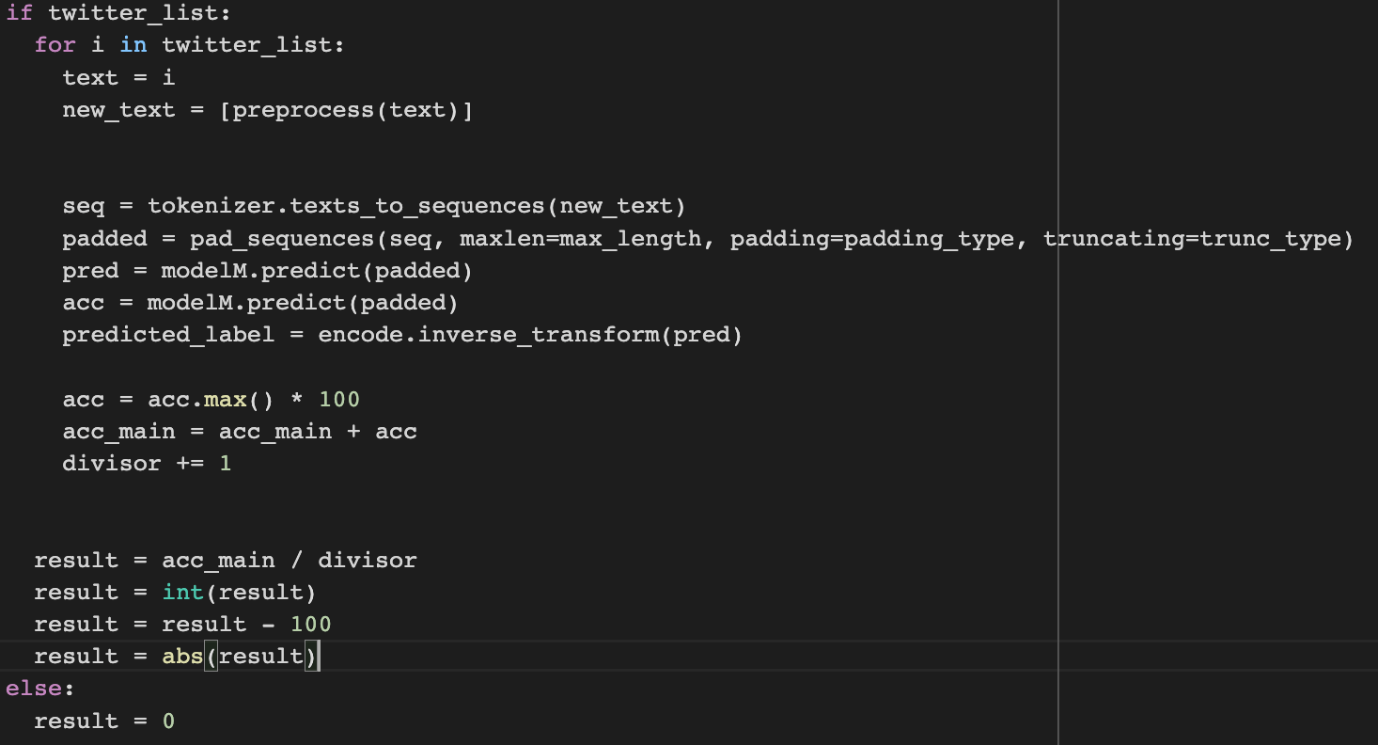
The model was trained to detect 10 different negative states that are shown in the image below, the data was taking negative reddit posts and the subreddit it relates to. The way the model works is that an accuracy of the prediction is given, since all the labels are negative, we can assume that the lower the accuracy the less negative the tweets are as they cannot relate to any of the labels.

Figure 39: Mental Health Model 1



The application goes through the user’s tweets which are captured and saved into the database, proceeds to make a prediction as to which category the tweets fall into and those accuracies are then added together and their mean is calculated to give a result, that result is then subtracted by a hundred, and the absolute value is used as the result shown to the user.

Figure 40: Mental Health Model 2



### Physical Activity Model

The model was trained to detect if someone’s tweets can relate to strength training or aerobic training. The dataset included multiple “logs” of users mentioning workouts they did and referring to specific exercises which can get captured by the model to understand that this given tweet relates highly to one of the two labels.

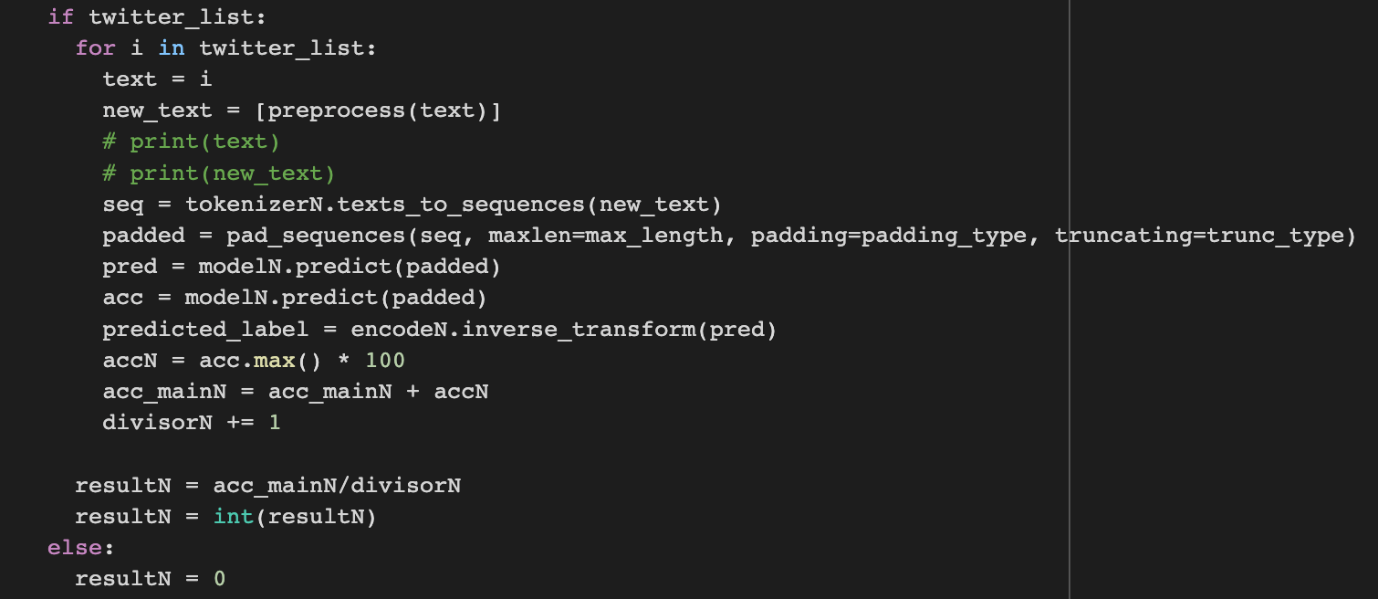
Like the Mental Health Model, it goes through all the user’s tweets that are in the database, but since it is only detecting the positive side of it the accuracy is taken as it is and divided by the number of tweets analyzed to give us the chart score for the user.

### Nutrition Model

The model was trained to detect if someone’s tweets can relate to “nutritious” and “healthy” food. The model takes tweets and tries to relate them to label to give the accuracy score, the higher the better.

Like the Mental Health Model, it goes through all the user’s tweets that are in the database, but since it is only detecting the positive side of it the accuracy is taken as it is and divided by the number of tweets analyzed to give us the chart score for the user.

Figure 41: Nutrition Model



# Testing

## 1. Functional Testing Goals

The aim of the functional testing is to test all the features of the application works in line with the expectations of the user. The main functionalities of the DiabetesCare application are as follows:

1. User Authentication
2. Creating User Profile
3. Inputting glucose level and blood pressure
4. Giving user diet recommendations
5. Giving user physical activity recommendations
6. Giving user mental health recommendations
7. Updating user charts based on social media analysis (twitter)

The sub-functionalities of the application will be activated based on the testing of the primary functionalities. To ensure the validity of the application, functional testing will be conducted, and the observed results will be compared to the expected outcomes. The testing period is 5 days, and all testing will be completed within this given time frame. The functional testing will be deemed successful if all features of the application have been thoroughly tested and no exceptions are discovered during the test cases.

## 2. Functional Test Plan Scope

Table 13: Test Plan Scope

|  |  |
| --- | --- |
| **In Project Scope** | **Out of Project Scope** |
| 1. User Authentication 2. Creating User Profile 3. Inputting glucose level and blood pressure 4. Giving user diet recommendations 5. Giving user physical activity recommendations 6. Giving user mental health recommendations 7. Updating user charts based on social media analysis (twitter) | 1. **User Honesty:** User could simply input wrong information, we as the developers have no way of making sure the user has cooked a meal with the ingredients given, or followed a workout as advised, or completed activities mentioned. 2. **User social media activity:** The app’s chart analysis depends on the user’s activity on twitter, we cannot have control over that. |

## 3. Functional Test Plan Assumptions and Constraints

### Assumptions

1. **Social Media activity**

DiabetesCare needs the user to have an active twitter account which they interact with daily, that will give the indication of how well the user is doing in his/her everyday life. This information can then be taken to have a medical professional question the patient/user to assure that they are sticking to the plan.

1. **User Is using application Honestly**

DiabetesCare required the user to be honest about their diet, physical activity, and mental health activities logs. If a user decides to lie or not follow through with the recommendations and plan, then the app will not work correctly.

### 3.2 Constraints

1. **Twitter Account**

DiabetesCare did not account for other social media platforms, nor does it work unless a user has an active twitter account, as without that the application fails to do its goal.

## 4. Functional Test Entry Criteria, Approach and Tools

### Testing Criteria

* Test Cases have been defined with the expected output.
* Complete application has to be developed to perform test cases.
* A group of testing users have been gathered in order to test the applications (6 members.
* Members must all agree to give us access to analyze their Twitter data.
* Members must have required tools such as an active Twitter account.

### Testing Methodology

Two testing approaches will be used to evaluate the completed application. First, alpha testing will be performed using predefined test cases and both black box and white box techniques. These tests will be conducted by the application developers, acting as real users.

Secondly, beta testing will involve real users testing the application in their everyday lives. The main benefit of this is to tell us as the developers what they think should be added or changed, as there are using it with no bias whatsoever.

### Testing Tools

Testing will be carried out on a laptops, phones, and tablets in order to make sure the app can work any time and any place.

For beta testing our target users will be asked questions on their experience with the app, giving us feedback which help us understand if the app is helping them in the right direction.

The application's efficiency will be verified and confirmed through the collection of continuous user feedback. This feedback will be utilized in a constructive manner to enhance the user experience and improve the application further.

## 5. Functional Test Entry Criteria, Approach and Tools

***1. Sign up***

Table 14: Test Case - Sign up

|  |  |
| --- | --- |
| Requirement Name | 1.1 |
| Test Case Description: | This test case checks if a user’s information exists in the database. It checks if the format of the information entered is valid then it will register the user. |
| Pre-Requirements: | The user matches all sign-up data requirements. |
| Input: | Name =” Yuno Syk, Email=” yunosyk@gmail.com”, Twitter Username =” @yuno, Password =”123456abc”, Confirm Password =”123456abc”, Gender= “male” |
| Expected Output: | Sign-up Confirmation |
| Observed Output | Sign-up Confirmation, loading consent form |
| Verdict: | Pass |
| Comments: | The user has an account in the database. |

|  |  |
| --- | --- |
| Requirement Name: | 1.2 |
| Test Case Description: | This test case checks if a user’s information exists in the database. It checks if the format of the information entered is valid then it will register the user. |
| Pre-Requirements: | The user matches all sign-up data requirements. |
| Input: | Name =” Kaye Lim”, Email=” klim@gmail.com”, Twitter Username =” @kayelim, Password =”12345”, Confirm Password =”12345”, Gender= “female” |
| Expected Output: | Sign-up Confirmation |
| Observed Output | Sign-up Confirmation, loading consent form |
| Verdict: | Pass |
| Comments: | The user has an account in the database. |

|  |  |
| --- | --- |
| Requirement Name: | 1.3 |
| Test Case Description: | This test case checks if a user’s information exists in the database. It checks if the format of the information entered is valid then it will register the user. |
| Pre-Requirements: | The user matches all sign-up data requirements. |
| Input: | Name =”Mahmoud Hamdan”, Email=” yunosyk@gmail.com”, Twitter Username =” 7mdan254“, Password =”123”, Confirm Password =”123”, Gender= “male” |
| Expected Output: | Sign-up Confirmation |
| Observed Output | Error message “Username does not exist on twitter or is not a public account” |
| Verdict: | Fail |
| Comments: | The user needs a twitter account that is public |

|  |  |
| --- | --- |
| Requirement Name: | 1.4 |
| Test Case Description: | This test case checks if a user’s information exists in the database. It checks if the format of the information entered is valid then it will register the user. |
| Pre-Requirements: | The user matches all sign-up data requirements. |
| Input: | Name =”Mahmoud Hamdan”, Email=” yunosyk@gmail.com”, Twitter Username =” mahmoud7mdan25“, Password =”123”, Confirm Password =”123”, Gender= “male” |
| Expected Output: | Sign-up Confirmation |
| Observed Output | Sign-up Confirmation, loading consent form |
| Verdict: | Pass |
| Comments: | The user has an account in the database. |

***2. Self- Assessment***

Table 15: Test Case - Self-Assessment

|  |  |
| --- | --- |
| Requirement Name: | 2.1 |
| Test Case Description: | This test case displays various questions where users fill their personal information and their current health status. |
| Pre-Requirements: | The user must successfully sign up. |
| Input: | Age: 23  Height: 165  Weight: 62  Access to equipment: yes Ranking in terms of training: Beginner |
| Expected Output: | User data will be saved, and home page will display. |
| Observed Output | User data will be saved, and home page will display. |
| Verdict: | Pass |
| Comments: | Personalized user recommendations (nutrition, workout) |

|  |  |
| --- | --- |
| Requirement Name: | 2.2 |
| Test Case Description: | This test case displays various questions where users fill their personal information and their current health status. |
| Pre-Requirements: | The user must successfully sign up. |
| Input: | Age: 22  Height: 156  Weight: 45  Gender: F  Access to equipment: no Ranking in terms of training: Beginner |
| Expected Output: | User data will be saved, and home page will display. |
| Observed Output | User data will be saved, and home page will display. |
| Verdict: | Pass |
| Comments: | Personalized user recommendations (nutrition, workout) |

|  |  |
| --- | --- |
| Requirement Name: | 2.3 |
| Test Case Description: | This test case displays various questions where users fill their personal information and their current health status. |
| Pre-Requirements: | The user must successfully sign up. |
| Input: | Age: 21  Height: 172  Weight: 82  Access to equipment: yes Ranking in terms of training: Intermediate |
| Expected Output: | User data will be saved, and home page will display. |
| Observed Output | User data will be saved, and home page will display. |
| Verdict: | Pass |
| Comments: | Personalized user recommendations (nutrition, workout) |

***3. Login***

Table 16:Test Case - Log in (authentication)

|  |  |
| --- | --- |
| Requirement Name: | 3.1 |
| Test Case Description: | This test case checks if the system allows the user to log in to the application using their account. |
| Pre-Requirements: | User Information existed in database,  User’s Twitter Username” @yuno” created with Password ”123456abc” |
| Input: | Twitter Username=” @yuno”, User password =”123456abc” |
| Expected Output: | Login successful, home page displays |
| Observed Output | Login successful, home page displays |
| Verdict: | Pass |
| Comments: | The user successfully logs in and can access the web application features. |

|  |  |
| --- | --- |
| Requirement Name: | 3.2 |
| Test Case Description: | This test case checks if the system allows the user to log in to the application using their account. |
| Pre-Requirements: | User Information existed in database,  User’s Twitter Username =” @kayelim, Password =”12345” |
| Input: | Twitter Username =” @kayelim, Password =”12345” |
| Expected Output: | Login successful, home page displays |
| Observed Output | Login successful, home page displays |
| Verdict: | Pass |
| Comments: | The user successfully logs in and can access the web application features. |

|  |  |
| --- | --- |
| Requirement Name: | 3.3 |
| Test Case Description: | This test case checks if the system allows the user to log in to the application using their account. |
| Pre-Requirements: | User Information existed in database,  User’s Twitter Username =” mahmoud7mdan25“, Password =”123” |
| Input: | Twitter Username =” mahmoud7mdan25“, Password =”123” |
| Expected Output: | Login successful, home page displays |
| Observed Output | Login successful, home page displays |
| Verdict: | Pass |
| Comments: | The user successfully logs in and can access the web application features. |

***4. Mental Health***

Table 17: Test Case - Recommendations for Mental Health

|  |  |
| --- | --- |
| Requirement Name: | 4.1 |
| Test Case Description: | This test case checks if appropriate recommendations (activities) are being given to the user |
| Pre-Requirements: | User logged in, data in journal |
| Input: | Journal entries from user: - “excited for the weekend and new year”  - “I feel really nervous I have a big meeting coming up” |
| Expected Output: | Recommended Activities |
| Observed Output | Recommended Activities |
| Verdict: | Pass |
| Comments: | The user data that has been saved will be used to display recommended activities. |

|  |  |
| --- | --- |
| Requirement Name: | 4.2 |
| Test Case Description: | This test case checks if appropriate recommendations (activities) are being given to the user |
| Pre-Requirements: | User logged in, data in journal |
| Input: | Journal entries from user: - “Starting my day off a really tired I need energy”  - “I have a very busy day ahead of me it’s getting me stressed” |
| Expected Output: | Recommended Activities |
| Observed Output | Recommended Activities |
| Verdict: | Pass |
| Comments: | The user data that has been saved will be used to display recommended activities. |

|  |  |
| --- | --- |
| Requirement Name: | 4.3 |
| Test Case Description: | This test case checks if appropriate recommendations (activities) are being given to the user |
| Pre-Requirements: | User logged in, data in journal |
| Input: | Journal entries from user: - “I woke up feeling great! Ready to seize the day” |
| Expected Output: | Recommended Activities |
| Observed Output | Recommended Activities |
| Verdict: | Pass |
| Comments: | The user data that has been saved will be used to display recommended activities. |

***5. Physical Activity***

Table 18: Test Case - Recommendations for Physical Activity

|  |  |
| --- | --- |
| Requirement Name: | 5.1 |
| Test Case Description: | This test case checks if appropriate recommendations (Workout) are being given to the user |
| Pre-Requirements: | User logged in; self-assessment completed |
| Input: | Self-Assessment:  1. Do you have access to equipment? (yes) 2. How would you rank yourself when it comes to training (Beginner) |
| Expected Output: | Workouts Recommendations (Beginner workouts, with equipment) |
| Observed Output | Workouts Recommendations (Beginner workouts, with equipment) |
| Verdict: | Pass |
| Comments: | The recommendations displayed are personalized based on user self-assessment data |

|  |  |
| --- | --- |
| Requirement Name: | 5.2 |
| Test Case Description: | This test case checks if appropriate recommendations (Workout) are being given to the user |
| Pre-Requirements: | User logged in; self-assessment completed |
| Input: | Self-Assessment:  1. Do you have access to equipment? (no) 2. How would you rank yourself when it comes to training (Beginner) |
| Expected Output: | Workouts Recommendations (Beginner workouts, without equipment) |
| Observed Output | Workouts Recommendations (Beginner workouts, without equipment) |
| Verdict: | Pass |
| Comments: | The recommendations displayed are personalized based on user self-assessment data |

|  |  |
| --- | --- |
| Requirement Name: | 5.3 |
| Test Case Description: | This test case checks if appropriate recommendations (Workout) are being given to the user |
| Pre-Requirements: | User logged in; self-assessment completed |
| Input: | Self-Assessment:  1. Do you have access to equipment? (yes) 2. How would you rank yourself when it comes to training (Intermediate) |
| Expected Output: | Workouts Recommendations (Intermediate workouts, with equipment) |
| Observed Output | Workouts Recommendations (Intermediate workouts, with equipment) |
| Verdict: | Pass |
| Comments: | The recommendations displayed are personalized based on user self-assessment data |

***6. Nutrition***

Table 19: Test Case -Recommendations for Nutrition

|  |  |
| --- | --- |
| Requirement Name: | 6.1 |
| Test Case Description: | This test case checks if appropriate recommendations (Food Options and calories) are being given to the user |
| Pre-Requirements: | User logged in; self-assessment completed |
| Input: | Self-Assessment test  Age: 23  Height: 165  Weight: 62  Gender: F |
| Expected Output: | Recommended Breakfast – Lunch – Dinner options with recipes and exact macronutrients  Exact Calories: 1445 |
| Observed Output | Recommended Breakfast – Lunch – Dinner options with recipes and exact macronutrients  Exact Calories: 1445 |

|  |  |
| --- | --- |
| Requirement Name: | 6.2 |
| Test Case Description: | This test case checks if appropriate recommendations (Food Options and calories) are being given to the user |
| Pre-Requirements: | User logged in; self-assessment completed |
| Input: | Self-Assessment test  Age: 22  Height: 156  Weight: 45  Gender: F |
| Expected Output: | Recommended Breakfast – Lunch – Dinner options with recipes and exact macronutrients  Exact Calories: 1271 |
| Observed Output | Recommended Breakfast – Lunch – Dinner options with recipes and exact macronutrients  Exact Calories: 1271 |

|  |  |
| --- | --- |
| Requirement Name: | 6.3 |
| Test Case Description: | This test case checks if appropriate recommendations (Food Options and calories) are being given to the user |
| Pre-Requirements: | User logged in; self-assessment completed |
| Input: | Self-Assessment test  Age: 21  Height: 172  Weight: 82  Gender: M |
| Expected Output: | Recommended Breakfast – Lunch – Dinner options with recipes and exact macronutrients  Exact Calories: 1912 |
| Observed Output | Recommended Breakfast – Lunch – Dinner options with recipes and exact macronutrients  Exact Calories: 1912 |

***7. Twitter Analysis***

Table 20: Test Case - Twitter Analysis

|  |  |
| --- | --- |
| Requirement Name: | 7.1 |
| Test Case Description: | This test checks if data is being appropriately analyzed using the user’s social media activity from twitter |
| Pre-Requirements: | User logged in; user is using twitter |
| Input: | Twitter data (tweets – retweets) |
| Expected Output: | 3 separate percentage indicators that show how well the user is doing in the three respective categories (Mental Health – Nutrition – Physical) Activity). Zero showing horrible indication when it comes to their habits, and 100 being excellent |
| Observed Output | 3 separate percentage indicators that show how well the user is doing in the three respective categories (Mental Health – Nutrition – Physical) Activity). Zero showing horrible indication when it comes to their habits, and 100 being excellent |
| Verdict: | Pass |
| Comments: | The users will view 3 different charts that illustrate their results (Mental Health, Nutrition, and Physical). |

|  |  |
| --- | --- |
| Requirement Name: | 7.2 |
| Test Case Description: | This test checks if data is being appropriately analyzed using the user’s social media activity from twitter |
| Pre-Requirements: | User logged in; user is using twitter |
| Input: | Twitter data (tweets – retweets) |
| Expected Output: | 3 separate percentage indicators that show how well the user is doing in the three respective categories (Mental Health – Nutrition – Physical) Activity). Zero showing horrible indication when it comes to their habits, and 100 being excellent |
| Observed Output | 3 separate percentage indicators that show how well the user is doing in the three respective categories (Mental Health – Nutrition – Physical) Activity). Zero showing horrible indication when it comes to their habits, and 100 being excellent |
| Verdict: | Pass |
| Comments: | The users will view 3 different charts that illustrate their results (Mental Health, Nutrition, and Physical). |

|  |  |
| --- | --- |
| Requirement Name: | 7.3 |
| Test Case Description: | This test checks if data is being appropriately analyzed using the user’s social media activity from twitter |
| Pre-Requirements: | User logged in; user is using twitter |
| Input: | Twitter data (tweets – retweets) |
| Expected Output: | 3 separate percentage indicators that show how well the user is doing in the three respective categories (Mental Health – Nutrition – Physical) Activity). Zero showing horrible indication when it comes to their habits, and 100 being excellent |
| Observed Output | 3 separate percentage indicators that show how well the user is doing in the three respective categories (Mental Health – Nutrition – Physical) Activity). Zero showing horrible indication when it comes to their habits, and 100 being excellent |
| Verdict: | Pass |
| Comments: | The users will view 3 different charts that illustrate their results (Mental Health, Nutrition, and Physical). |

***8. User Results (5-day testing)***

Figure 42: Testing Day 1

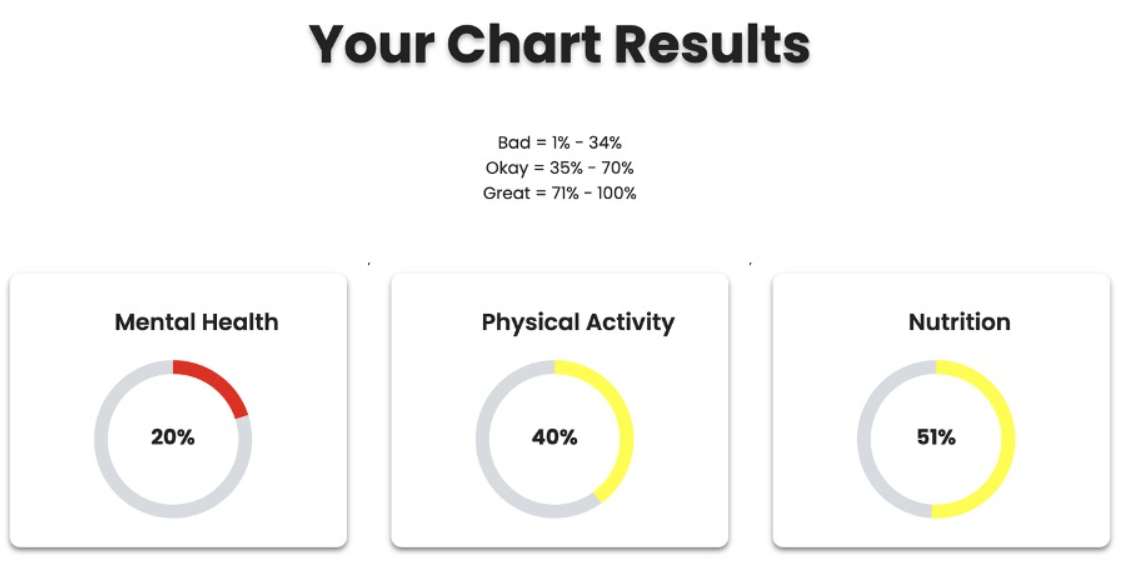
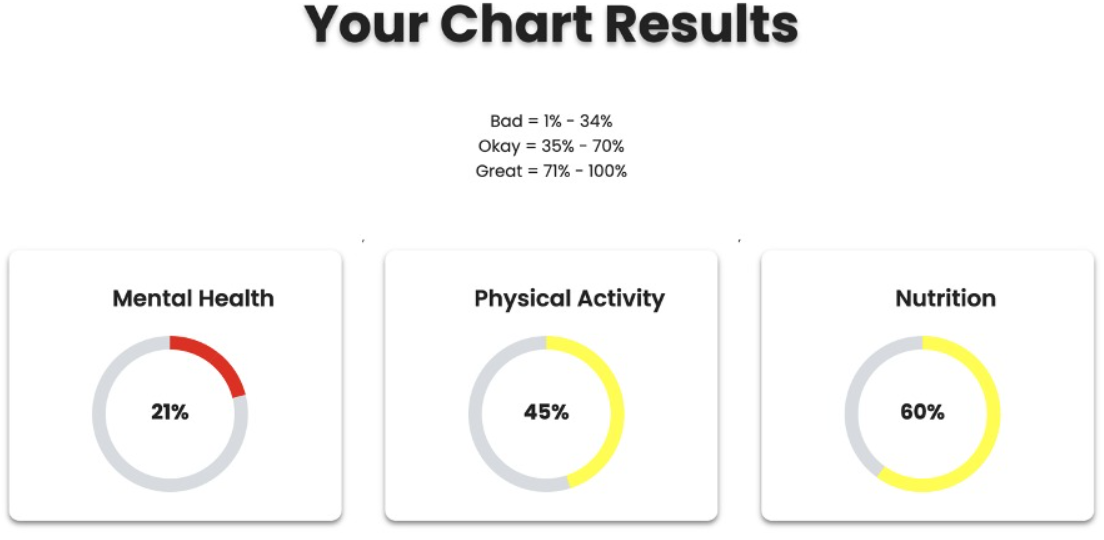
**

Figure 43: Testing Day 5

**

# Maintenance

In this we will look at how we can maintain the system in a long run. We will do this by following 4 methods to continuously maintain the system.

* Adaptive - When changes are required in the system such as software updates. We have put a development plan that will improve the features of our health app and will also take into consideration of the market demands and changes and improve our app according to those guidelines and demands
* Perfective - We will continuously monitor our app and will also monitor our competitors and will analyze what they are doing and try to upgrade our app accordingly. We will also do market research to understand the needs and our target audience to improve the app and features accordingly to the needs of our customers.
* Corrective - We will get feedback and reviews from our end users and executive members and try to improve any bugs, broken links and performance issues in the application. We will also improve any userability issues by continuously testing the application and upgrading the web application host.
* Preventative - We will have multiple layers of testing to prevent any errors. If any bug found our cyber security will try to fix the issue and find solutions to prevent the issue from happening again.

## Appendices

### 1. Glossary

*Blood glucose*: The amount of sugar (glucose) present in a person's blood. Blood glucose levels are an important factor in the management of diabetes, as high levels can lead to complications.

*Blood pressure*: The force of blood against the walls of the arteries as it is pumped around the body. Blood pressure is an important indicator of overall health and is often monitored in people with diabetes.

*Deep learning*: A subfield of machine learning that involves the use of algorithms based on artificial neural networks to analyze and interpret data.

*Diabetes*: A condition in which the body is unable to properly use and store glucose, a type of sugar that is the main source of energy for the body's cells.

*Flask*: A microweb framework written in Python that is used to develop web applications.

*Machine learning (ML):* A subset of artificial intelligence (AI) that focuses on using data and algorithms to mimic how people learn, progressively improving its accuracy.

*MongoDB*: A popular open-source database management system that uses a document-oriented data model.

*Multi-factor authentication (MFA*): A type of authentication that requires users to provide two or more verification factors in order to have access to an application.

*Ngrok*: A tool that allows users to expose a local development server to the internet.

*Natural language processing (NLP):* The analysis and interpretation of human language in order to extract meaning and information.

*Neural network*: A type of machine learning algorithm that is designed to simulate the way the human brain works.

### 2. Version Index

Table 21: Version Index: Final Document

|  |  |  |
| --- | --- | --- |
| Date | Version# | Description |
| 01/1/2022 | 1.0 | Final Report Document |

### 3. References

Coleman, N 2018, *Diabetes not taken seriously by UK public*, *Diabetes UK*. Diabetes UK, viewed 26 December 2022, <https://www.diabetes.org.uk/about\_us/news/be-in-the-know-diabetes-complications>.

Diabetes 2022, World Health Organization, viewed 20 December 2022, <https://www.who.int/news-room/fact-sheets/detail/diabetes>.

Goad, K 2022, *How does race affect diabetes risk?*, *AARP*, viewed 25 December 2022, <https://www.aarp.org/health/healthy-living/info-2018/role-of-race-in-diabetes.html#:~:text=African%20Americans%2C%20Hispanics%2C%20American%20Indians,American%20Diabetes%20Association%20>.

Nakamoto, P, 2017. ‘Neural networks and deep learning: deep learning explained to your granny A visual introduction for beginners who want to make their own Deep Learning Neural Network’, CreateSpace Independent Publishing Platform.

Sivasubramaniam, S., Dlabolová, D.H., Kralikova, V. and Khan, Z.R., 2021, ‘Assisting you to advance with ethics in research: an introduction to ethical governance and application procedures’, *International Journal for Educational Integrity*, vol.17, no.1, pp.1-18.

The facts, stats, and impacts of diabetes 2022, *Centers for Disease Control and Prevention*, viewed 25 December 2022, <https://www.cdc.gov/diabetes/library/spotlights/diabetes-facts-stats.html>.

Wardian, J.L, 2017, ‘Diabetes cannot be controlled, but it can be managed’, *Clinical Diabetes*, vol.*35*, no.5, pp.329-330.

Wisdom, V. and Gupta, R, 2016, ‘An Introduction to Twitter Data Analysis in Python’. *Artigence Inc*, pp1-5.

Whittemore,R., Vilar-Compte, M., De La Cerda, S., Marron, D., Conover, R., Delvy, R., Lozano-Marrufo, A. and Pérez-Escamilla, R., 2019, ‘Challenges to diabetes self-management for adults with type 2 diabetes in low-resource settings in Mexico City: a qualitative descriptive study’, *International journal for equity in health*, vol.18, no.1, pp.1-10.

Vidvadharan, V., Hamdan, M. and Zalzala, A.M. 2021, ‘An Evidence-Based Study of Diabetes Prevention and Management with NLP and Deep Learning’. *IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 1-8.

### 4. Progress Reports

Figure 44: Semester 1 Progress Report

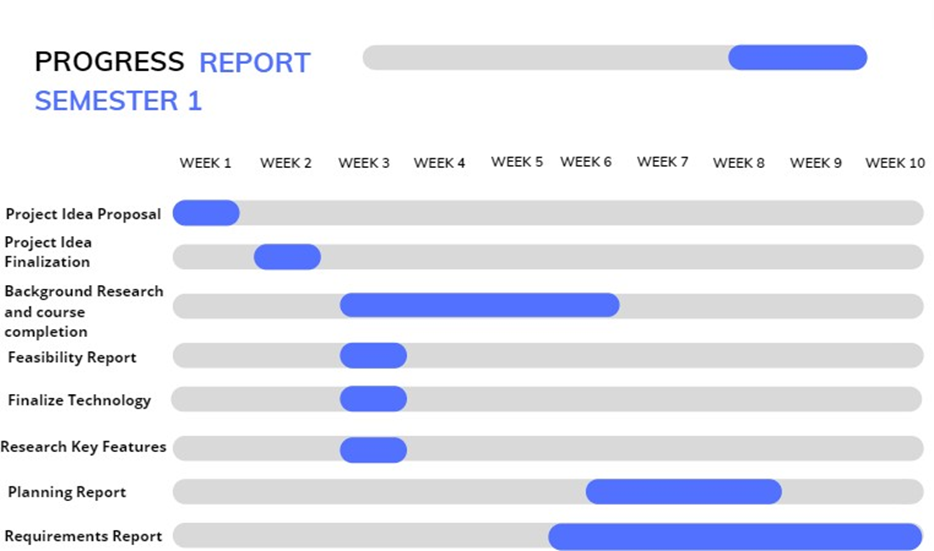


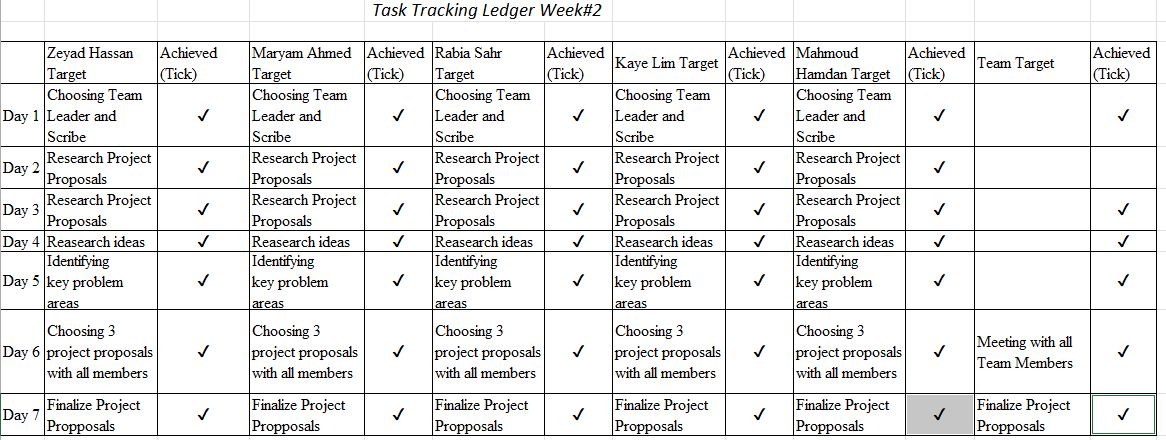
Figure 45: Semester 2 Progress Report

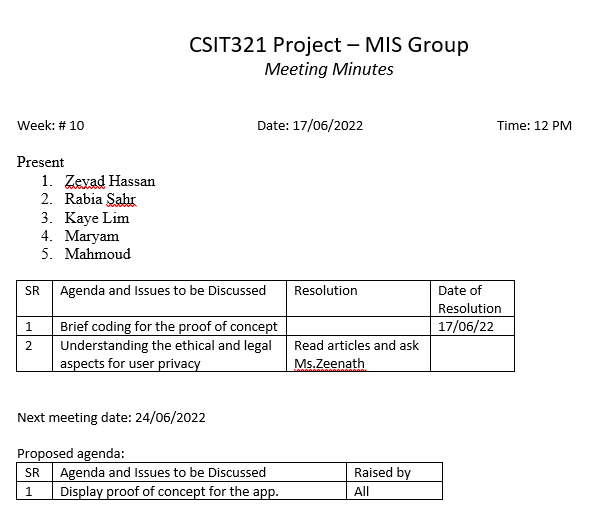


### 5. Meeting Minutes

Figure 46: Spring 2022







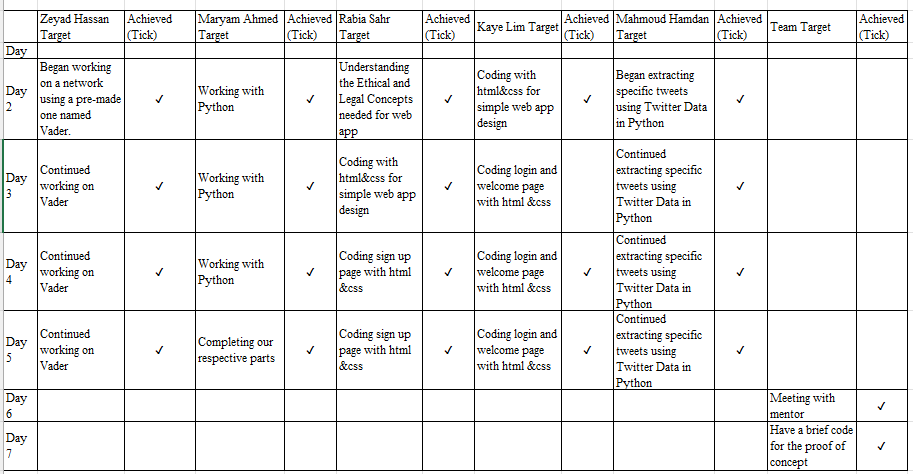
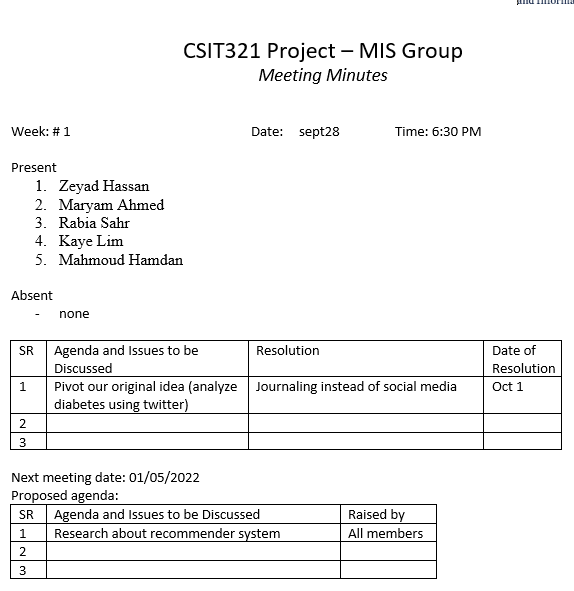
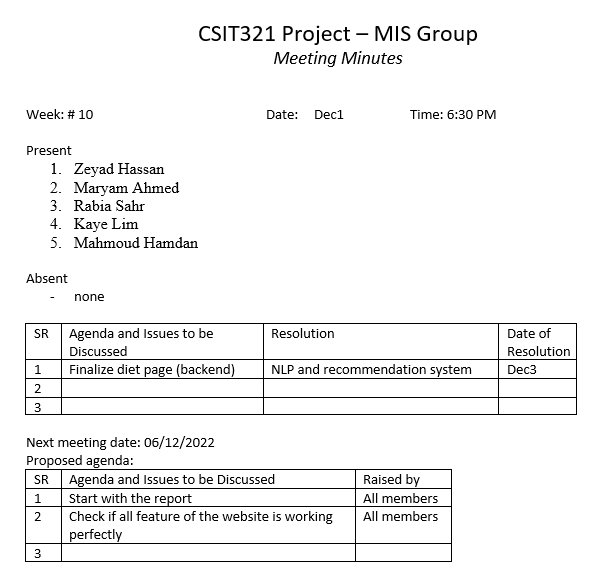
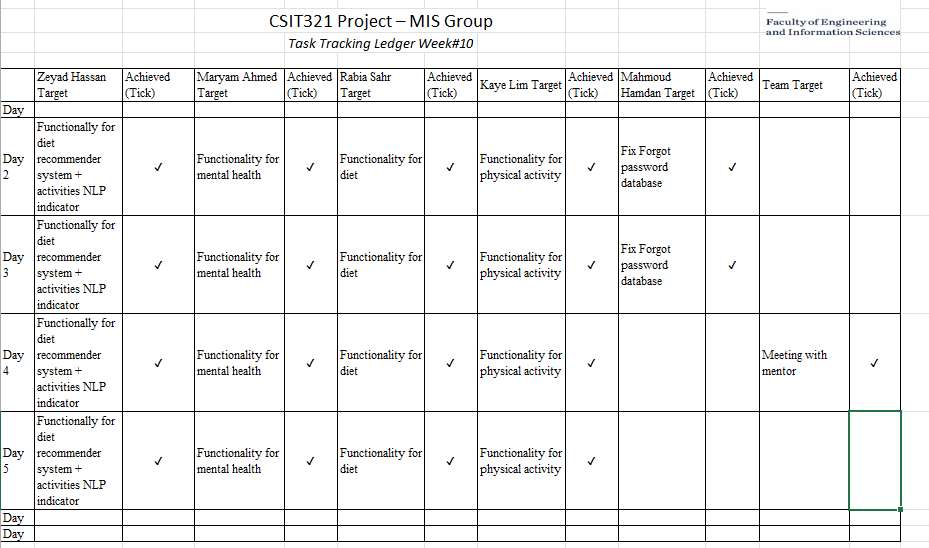


Figure 47: Autumn 2022









### 6. Division of work among team members

Table 22: Division of Work

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tasks | Zeyad | Mahmoud | Kaye | Sahar | Maryam |
| UI/UX Design | Check mark, Wingdings font, character code 252 decimal. |  | Check mark, Wingdings font, character code 252 decimal.Check mark, Wingdings font, character code 252 decimal.Check mark, Wingdings font, character code 252 decimal. | Check mark, Wingdings font, character code 252 decimal. |  |
| UI/Front end Implementation |  |  | Check mark, Wingdings font, character code 252 decimal.Check mark, Wingdings font, character code 252 decimal.Check mark, Wingdings font, character code 252 decimal. | Check mark, Wingdings font, character code 252 decimal.Check mark, Wingdings font, character code 252 decimal. | Check mark, Wingdings font, character code 252 decimal. |
| Dataset Collection | Check mark, Wingdings font, character code 252 decimal. |  |  |  |  |
| Machine Learning training/testing | Check mark, Wingdings font, character code 252 decimal. |  |  |  |  |
| Deploying ML model | Check mark, Wingdings font, character code 252 decimal. |  |  |  |  |
| Inputting database information | Check mark, Wingdings font, character code 252 decimal. | Check mark, Wingdings font, character code 252 decimal. | Check mark, Wingdings font, character code 252 decimal. | Check mark, Wingdings font, character code 252 decimal. | Check mark, Wingdings font, character code 252 decimal. |
| Twitter authentication and tweet extraction |  | Check mark, Wingdings font, character code 252 decimal. |  |  |  |
| Integrating the 3 systems (Python, mongo, flask) | Check mark, Wingdings font, character code 252 decimal. |  |  |  |  |
| Back-end development (functionalities) | Check mark, Wingdings font, character code 252 decimal. |  |  |  |  |