HEALTH AND WELLNESS INFORMATION SYSTEM

Monica Rangel

Follow this and additional works at: https://scholarworks.lib.csusb.edu/etd

Part of the Computer Engineering Commons

Recommended Citation
Rangel, Monica, "HEALTH AND WELLNESS INFORMATION SYSTEM" (2019). Electronic Theses, Projects, and Dissertations. 943.
https://scholarworks.lib.csusb.edu/etd/943

This Project is brought to you for free and open access by the Office of Graduate Studies at CSUSB ScholarWorks. It has been accepted for inclusion in Electronic Theses, Projects, and Dissertations by an authorized administrator of CSUSB ScholarWorks. For more information, please contact scholarworks@csusb.edu.
HEALTH & WELLNESS INFORMATION SYSTEM

A Project
Presented to the
Faculty of
JHB College of Business and Public Administration,
California State University
San Bernardino

In Partial Fulfilment
of the Requirements for the Degree
Master of Information Systems and Technology

by
Monica Rangel
June 2019
HEALTH & WELLNESS INFORMATION SYSTEM

A Project
Presented to the
Faculty of
JHB College of Business and Public Administration,
California State University
San Bernardino

by
Monica Rangel
June 2019

Approved by:

Jay Varzandeh, PhD., Chair
Date

Conrad Shayo, PhD, Reader

Jay Varzandeh, PhD., Chair, Information & Decision Sciences Department
ABSTRACT

The greatest wealth is health. It is sometimes said your health is a function of what you are not doing, not what you are currently doing. The degree to which individuals can attain, process, and comprehend the necessary health information and services they need to make proper health decisions is vital for optimal health and well-being.

This project documents the analysis, design, development, and implementation of a prototype web-based data-driven health & wellness system targeted for college students. The architecture for this system uses business intelligence to develop a smart online platform for real-time analysis based on inputs entered by its users.

The objective is to develop modules that can be used to provide meal plan options that dietitians can recommend to students, while also providing a standard wellness health check. This also promotes constant awareness for students with specialized health diets. User-health and wellness history of each Student is collected and stored for generating progress and wellness reports for end users. The dietitian can monitor the user in real time through the data collected and stored in the data server. Users can monitor their own progress. The system incorporates user context and feedback to personalize each user's lifestyle.

Implementation of this system provides a complete and easy to use integrated system that promotes the process of analyzing wellness and improving the user’s overall health. The system is designed to be in a non-clinical setting and hence more lifestyle-oriented compared to other health-oriented systems. It is thus more relevant and convenient to student's everyday life context.
ACKNOWLEDGEMENTS

I would like to express my special thanks and sincere gratitude to my advisor Javad Varzandeh, who supported me in completing and reviewing my project. I would also like to thank my graduate professor Dr. Conrad Shayo, for his valuable suggestions and support.

Finally, I want to thank my family Monica Torres and Michael Garaysi, who stood by my side in every situation, supporting me at all hard times and for making me stronger. I am grateful to have them in my life.
# Table of Contents

**ABSTRACT** ............................................................................................................................. 4  
**ACKNOWLEDGEMENTS** ........................................................................................................... 5  
**LIST OF TABLES** ..................................................................................................................... 9  
**CHAPTER ONE: INTRODUCTION** ........................................................................................... 11  
  1.1 Chapter Overview .................................................................................................................. 11  
  1.2 Problem Statement ................................................................................................................ 11  
  1.2 Problem Proposed Solution .................................................................................................... 12  
  1.3 Project Scope ......................................................................................................................... 14  
**CHAPTER TWO: REQUIREMENTS AND SPECIFICATIONS** ...................................................... 15  
  2.1 Chapter Overview .................................................................................................................. 15  
  2.2 End Users ............................................................................................................................... 15  
  2.3 System Requirement Specification .......................................................................................... 16  
  2.4 Actors and use cases ............................................................................................................... 16  
  2.5 Cost Estimation ...................................................................................................................... 17  
  2.6 Modules .................................................................................................................................... 18  
    Admin Login .............................................................................................................................. 18  
    User Login ............................................................................................................................... 18  
    Dietitian Login ......................................................................................................................... 20  
  2.7 Resource Requirements .......................................................................................................... 21  
    2.6.1 Hardware Requirement: .................................................................................................................. 21  
    2.6.2 Software Requirement: .................................................................................................................. 21  
    2.6.3 Language Requirements for Building the System ........................................................................ 21  
**CHAPTER THREE: SYSTEM DESIGN & ARCHITECTURE** ............................................................ 22  
  3.1 Chapter Overview .................................................................................................................. 22  
  3.2 System Architecture & Design ............................................................................................... 22  
    3.2.1 3- Tier Architecture ..................................................................................................................... 22  
  3.3 User Interface ......................................................................................................................... 23  
  3.4 Communication Interface ....................................................................................................... 23  
  3.4 Use Case Diagrams .................................................................................................................. 23  
    3.4.1 Use Case Diagram for the Admin ................................................................................................. 24  
    3.4.1 Use Case Diagram for the Student ............................................................................................... 25  
    3.4.1 Use Case Diagram for the Dietitian ............................................................................................... 26  
  3.5 E-R Diagram ........................................................................................................................... 27  
**CHAPTER 4: IMPLEMENTATION** ............................................................................................ 28
4.1 Account Interface Design ........................................................................................................ 28
   4.1.1 Static Pages .................................................................................................................. 28
   4.1.2 Account Login ............................................................................................................ 29
   4.1.3 Login .......................................................................................................................... 30
   4.1.4 Student Details ......................................................................................................... 31
   4.1.5 BMI Calculator ......................................................................................................... 32
   4.1.6 Edit User Information ............................................................................................. 33
   4.1.7 User Registration .................................................................................................... 34
   4.1.8 Create Diet Plan ........................................................................................................ 35
   4.1.9 Diet Plan Scheduling ............................................................................................... 36
   4.1.10 Sleep Tips ............................................................................................................... 37
   4.1.11 Calorie Intake .......................................................................................................... 38
   4.1.12 Feedback ................................................................................................................ 38
   4.1.13 View Request ......................................................................................................... 39
   4.1.14 Create Diet Plan ..................................................................................................... 40
   4.1.10 Create Diet Plan ..................................................................................................... 41

4.2 Technical Feasibility ............................................................................................................ 43

4.3 Economic Feasibility ......................................................................................................... 44

4.4 Operational Feasibility ...................................................................................................... 44

CHAPTER 5: TESTING ............................................................................................................ 46

5.1 Black box testing .............................................................................................................. 46
5.2 White box testing ............................................................................................................. 46
5.3 String Testing .................................................................................................................... 46
5.4 Levels of Testing .............................................................................................................. 48
5.5 Unit Testing ...................................................................................................................... 48
5.6 Integration Testing .......................................................................................................... 48
5.7 System testing ................................................................................................................ 49
5.8 Validation Testing .......................................................................................................... 49
5.9 Output Testing ................................................................................................................ 49

CHAPTER 6: CONCLUSION ................................................................................................. 51

6.1 Summary ........................................................................................................................ 51

6.2 Conclusion ....................................................................................................................... 51

APPENDIX ............................................................................................................................ 53

   Mission ............................................................................................................................ 53
   Vision ............................................................................................................................... 53
   Values ............................................................................................................................... 53
   Competitive Advantage ................................................................................................. 54

6.2 Project Advantages ......................................................................................................... 54

6.3 Disadvantages ................................................................................................................. 54

6.4 Applications: ............................................................................................................... 55
LIST OF TABLES

Table 1. Actors ...................................................................................................................... 16
Table 2: Use Cases .............................................................................................................. 16
Table 3. Functional and Non-Functional Requirements..............................................59
LIST OF FIGURES

Figure 1: Use Case Point Analysis.................................................................17
Figure 2. Tier Architecture.............................................................................22
Figure 3: Use case diagram for Admin............................................................24
Figure 4. Use case diagram for Student..........................................................25
Figure 5: Use case diagram for Dietitian.........................................................26
Figure 6. E-R Diagram....................................................................................27
Figure 7. Home Page .....................................................................................29
Figure 8: Login Page.......................................................................................35
Figure 9. Admin Login Page ..........................................................................36
Figure 10: User Login Page............................................................................36
Figure 11: Dietitian Login Page.......................................................................36
Figure 12. Add Dietitian ..............................................................................37
Figure 13. Student Details ............................................................................38
Figure 14: BMI Calculator............................................................................39
Figure 15. User Registration ........................................................................40
Figure 16: Edit User Information.................................................................41
Figure 17: Diet Plan Scheduling...................................................................42
Figure 18. Sleep Tips ..................................................................................43
Figure 19: Calorie Intake.............................................................................44
Figure 20. Feedback.......................................................................................36
Figure 21: View Request .............................................................................46
Figure 22: Create Diet Plan .........................................................................47
Figure 23: View Feedback...........................................................................48
Figure 24: Project Schedule.........................................................................62
Figure 25: Gantt Chart................................................................................62
Figure 26: Database Dietitian........................................................................77
CHAPTER ONE: INTRODUCTION

1.1 Chapter Overview

This chapter identifies and summarizes the problem that the project addresses. This includes: lack of sleep, poor eating habits, and stress among college students. The Health and Wellness System conceived, designed, and implemented for this project, provides one way students can monitor and minimize the impact of such problems. The last section of the chapter describes the scope of the work that will be implemented throughout the project. The key take away from this section highlights the absolute need for a student Health & Wellness System: a platform that supports wellness and health.

1.2 Problem Statement

College students frequently let their health fall by the curb, permitting academics to overshadow excellent and bad health habits. It can get relatively easy to skip a meal or cave into a cheap fast food option when working and attending school. Increased fast food consumption can affect health because it has been linked to a diet that is “high in calories, saturated fat, sugar, and sodium, as well as body fatness, weight gain, and increased body mass index” [8]. The underlying consequences are not as easy. Physical health and nutrition connect straightforwardly to one's mental health, effective study habits, and regular sleep patterns.

Nutrition and exercise can seem relatively insignificant while young, but constructing a routine of healthy behaviors during this time can aid well in the long stretch. The
following are issues many students face as they balance coursework with other obligations and responsibilities:

- **Lack of Sleep:** Regrettably, when students make time to study and socialize, they lose sleep. Pulling "all-nighters," or staying up too late, consuming a lot, leaving one with low levels of energy and incentive. With no sleep accountability and college freedom, students are easily substitutable to less sleep.

- **Poor Eating Habits:** While usual fast food and parallel dining selections appear like time and money savers, accumulating too many of these meals to one’s diet deprives one of the nutrients required to keep physically healthy and mentally aware. This can become a lifestyle and consequently cause long term health issues, from diabetes to obesity.

- **Stress:** Throughout the students' academic career, anxiety will likely appear from the onset of exams, assignments, and other requirements. With the accumulation of a challenging work schedule and other accountabilities, stress may reach unprecedented heights. Acquiring the knowledge about how to mitigate stress takes practice and patience, and doing so would result in a healthier tactic to achieving goals.

**1.2 Problem Proposed Solution**

This system aims at providing easily accessible health & wellness services for the university community and all its students. This may include those who may not have the resources to become self-aware or would like to monitor better health choices. The following are the proposed solutions that will help mitigate student health and wellness issues:
BMI Calculator – The proposed application will use a calculator to calculate the range of healthy weight based on the users Body Mass Index (BMI).

  - Benefit: Students have accessible scales on campus via the student health center and recreational center. Conveniently, being able to track your progress within the system will allow constant awareness of a healthy weight target. This aims at providing a convenient way to reduce health risks and develop a healthier lifestyle.

Sleep Routine: The system will provide insights on creating a balance sleep routine. Insights include tips, benefits, and FAQ.

  - Benefit: Sometimes, getting a good night’s rest is a lot easier said than done. Providing tips may help sleep-deprived students get back on track while also benefiting from better sleep patterns.

Stress Survey: Although detecting stress may not always be easily determined for students. Providing a “How do you know you’re stressed” survey will point the student to the correct resource whether it’s eating & weight management, time management, depression/ anxiety, and finances, etc.

  - Benefit: For many students, the first few weeks of college can create a stressful environment. There are several resources available on campus to help mitigate student stress. Using this tool will help point you in the right direction to find help on campus.

The proposed Health & Wellness System is expected to provide self-awareness and support dietitians in conjunction with the student health center to provide a resource platform to a virtual audience. This system also asks you similar questions related to
body and health. The system will also advise you about what should your intake in your
diet and what should you ignore to keep yourself healthy via your diet.

1.3 Project Scope

The Health & Wellness system is an end user support and an online resource tool
project. Here we propose a system that allows users to receive instant guidance on
health & wellness through an online platform.

When you go to a nutrition specialist, he or she will ask you your details related to body
and health such as your age, your height, your weight and how much water do your
consumer in a day and how much work do you do regularly. Just like this nutrition
specialist, the system also asks you similar questions in your platform, and you have to
answer all those questions. These include questions such as age, gender, weight, and
height. This system will also advise you about what should your intake in your diet and
what should you ignore to keep yourself healthy via your diet. The scope of the system
will be limited to the following aspects:

- BMI calculator for students who are enrolled in the system.
- Sleep Insights include tips, benefits, and FAQ.
- Stress Survey: “How do you know you’re stressed” survey.

In this system, using the technique of artificial intelligence, you will get access to all
resources via this web application, which is provided by the system. The main
advantage of using this web application is a healthier lifestyle that provides awareness.
Also, this system offers more than one diet plan also, for customized preferences.
CHAPTER TWO: REQUIREMENTS AND SPECIFICATIONS

2.1 Chapter Overview

This chapter is first focused on the three types of end users Admin, Student users, and dietitians via the student health center. Then shown is a list of the systems requirement and specifications for the system. These include the functional and non-function requirements that the system must provide. full description of the requirements and specifications for the health and wellness system development in which is described in detail in this chapter.

2.2 End Users

There are three types of system users:

a. Admin: They will be responsible and reliable for any maintenance, updates, adding and removing users, searching users, adding details, and health options to the system and, analyzing feedback.

b. Student Users: They are the user of the system that has access to all of the functionalities of the system with respect to admin and dietitian functionalities.

c. Dietitians: They are users that support the student's meal plans and health and wellness reports.
2.3 System Requirement Specification

A list of all functional and non-functional requirements are listed and referenced in the appendix. This contains a table with a structured collection of information that embodies the requirements of the H&W System.

2.4 Actors and use cases

<table>
<thead>
<tr>
<th>Actors</th>
<th>Semantics</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dietitian</strong></td>
<td>A user who provides a student diet plan and health wellness.</td>
</tr>
<tr>
<td><strong>Student</strong></td>
<td>A user who is looking for a health consult.</td>
</tr>
<tr>
<td><strong>The H&amp;WS</strong></td>
<td>The system that manages a platform that is used by dietitians and students.</td>
</tr>
<tr>
<td><strong>University Pay Processing Company</strong></td>
<td>The system redirects to the student login and student center where the student pays for any additional support.</td>
</tr>
</tbody>
</table>

Use cases

<table>
<thead>
<tr>
<th>Use cases</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CreateAccount</strong></td>
<td>For both the Dietitian actor and Student actor create an account</td>
</tr>
<tr>
<td><strong>EntersBMI</strong></td>
<td>The user enters values to create BMI score</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------</td>
</tr>
<tr>
<td>UserDetails</td>
<td>Displays student users’ details</td>
</tr>
<tr>
<td>DietChart</td>
<td>Displays insights on the food intake for a healthy lifestyle</td>
</tr>
<tr>
<td>DietPlan</td>
<td>Request a Personalized Diet Plan to Dietitian</td>
</tr>
<tr>
<td>ViewFeedback</td>
<td>View student feedback regarding the system.</td>
</tr>
<tr>
<td>RequestPlan</td>
<td>Student requests a personalized diet plan</td>
</tr>
<tr>
<td>ValidatePayment</td>
<td>Student will be redirected to the student center where they must process their payment.</td>
</tr>
<tr>
<td>Login</td>
<td>Users enter the H&amp;WS</td>
</tr>
<tr>
<td>Logout</td>
<td>Users quit the H&amp;WS</td>
</tr>
<tr>
<td>ViewDietPlans</td>
<td>The dietitian views all requested plans</td>
</tr>
</tbody>
</table>

### 2.5 Cost Estimation

**Figure 1: Use Case Point Analysis**

<table>
<thead>
<tr>
<th>Points</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Minimum Hours per Use Case Point</td>
</tr>
<tr>
<td></td>
<td>Maximum Hours per Use Case Point</td>
</tr>
<tr>
<td></td>
<td>Average Hours per Use Case Point</td>
</tr>
<tr>
<td>UUCW - Unadjusted Use Case Weight</td>
<td>15</td>
</tr>
<tr>
<td>UAW - Unadjusted Actor Weight</td>
<td>25</td>
</tr>
<tr>
<td>UUCP - Unadjusted Use Case Points:</td>
<td>20</td>
</tr>
<tr>
<td>TCF - Technical Complexity Factor</td>
<td></td>
</tr>
<tr>
<td>EF - Environmental Factor</td>
<td></td>
</tr>
<tr>
<td>UCP - Use Case Points</td>
<td></td>
</tr>
</tbody>
</table>

The Cost estimation effort for this system is calculated by Use Case Point Analysis. The
Unadjusted Use case points are based on the quality and complexity of the system actors and use cases. The formula used is UUCP= UAW+UUCW. The Effort per use case is shown above. This is taking the total effort and dividing it by the total unadjusted use case weight to calculate the effort per UUCW. The results show the minimum hours per UCP is 15 with a Max of 25.

### 2.6 Modules

A Modular design, or "modularity in design", is a design approach that subdivides a system into smaller parts called modules. This website has three significant modules with its sub-modules as follows:

**Admin Login**

- **Add Dietitian:**
  - The system allows admin to add dietitian details into the system who can create the diet plan for the users. This can be used for new resources which are attached to support the system.

- **View/Students:**
  - Can view and edit the student details whenever required.
  - Can see the list of registered users with their details.

- **View Feedback**
  - Can view feedback provided by the registered users.

**User Login**

- **Register:**
- To continue with the diet plan details, the user first needs to fill up all the required details.

  o My Details:
    - The user views account details that are currently used to register into the system.

  o Calculate BMI:
    - Based on details provided by the user, the system automatically calculates the BMI of the user.

  o Get Diet plan:
    - Based on details provided by the user, the system asks the user if they would like to get a diet plan.

  o Timings:
    - Based on details provided by the user, the system asks the user what times they typically eat throughout the day. Including Breakfast, lunch, snacks, and Dinner.

  o Plan:
    - The System then creates a diet plan based on proteins and Carbohydrates based on the different times of the day.

  o View Diet Plan:
    - The diet plan for the user is generated by the system itself using artificial intelligence.
- Request Diet Plan:
  - If the user is unsatisfied with the diet plan provided by the system, then he/she can raise a request to generate the proper diet plan him/herself.
  - The diet plan request is forwarded to a dietitian.
- Calorie Intake:
  - User can use a 3rd party widget to calculate suggested daily calorie intake for weight loss, maintenance, and extreme fat loss.
- Survey:
  - User can use a self-question survey to ask questions regarding stress.
- Write Feedback:
  - Registered users can provide feedback and the feed

**Dietitian Login**

- View Diet Plan Request:
  - Here, a dietitian can view the diet plan request from the users.
- Create a Diet Plan:
  - Based on the user's request, dietitian creates a diet plan for him/her.
- Update Diet Plan:
  - The user resends the request to a dietitian about unsatisfied diet plan. So, as per the user's request, dietitian regenerates the diet plan and sends to the user.
- View User:
2.7 Resource Requirements

The Project is loaded in Visual Studio 2019. Visual Studio is used for the design and coding of the project. Developed, maintained, and updated all databases into SQL Server 2017, tables and queries are created to store data and record of the project.

2.6.1 Hardware Requirement:

- i3 Processor-Based Computer
- 1GB-Ram
- 5 GB Hard Disk
- Internet Connection

2.6.2 Software Requirement:

- Operating System: Windows 10 or higher
- Google Chrome or Safari

2.6.3 Language Requirements for Building the System

- Programming Language: .NET
- Web- Technology: ASP.NET 2.0
- Front- End: C#. NET, Visual Studio 2019
- Back-End: SQL Server 2017
3.1 Chapter Overview

This chapter discusses the designing analysis phase of the system, which includes use cases, and diagrams.

3.2 System Architecture & Design

The Health and Wellness system implements a client-server web-based system. The front-end pages and business tier are written in C#. The whole project is built on a .NET framework.

3.2.1 3- Tier Architecture

The Health and Wellness system is organized into a 3-tier architecture. The following tiers are used within the design:

- Application layer or Presentation Layer is the front-end layer in a 3 – tier system that consists of the user interface. This is built using Asp.Net web forms.

Business tier contains the business logic, validations, or calculations related to the data.
This was done using C# classes and calculations.

Data Layer contains methods that help the business layer to connect the data and perform required actions, whether to return, update, or manipulate data [4].

These three layers are beneficial in scalability, ease of management, flexibility and Security. You can secure each tier separately using different methods. It’s easy to manage separately modifying or updating a tier without having any effects on existing tiers. Finally, the architecture is flexible in which it’s able to expand each tier.

3.3 User Interface
The H&WS categorizes users into three major components: Dietitian, Administrator, and Student user. All of the system's interfaces include a login interface page.

3.4 Communication Interface
The H&W System uses .NET framework and the operating system to manage contact between the student and the server.

3.4 Use Case Diagrams
A use case diagram is a representation of a user’s interaction with the system in which shows the relationship between the user and different use cases in which the user is involved. There are three primary end users: Admin, Dietitians, and Students. In this section will discuss, separate use case diagrams for the three types of users will be discussed in this section.
3.4.1 Use Case Diagram for the Admin

A detailed use case diagram for the Admin is shown in figure 3.

Figure 3: Use case diagram for Admin
3.4.1 Use Case Diagram for the Student

A detailed use case diagram for the Student is shown in figure 4.

Figure 4: Use case diagram for Student
3.4.1 Use Case Diagram for the Dietitian

A detailed use case diagram for the Dietitian is shown in figure 5 shows the dietitian login, view, create a diet plan, update, and view details.

Figure 5: Use case diagram for Dietitian
3.5 E-R Diagram
The entity relationship diagram shows the relationships of entities and how each relates within the H&W System.

Figure 6: E-R Diagram
4.1 Account Interface Design

The H&W System is a fully web-based system programmed with Asp.net. Asp.net provides services to allow the creation, deployment, and execution of web applications and web services [3]. Asp.net supports popular browsers such as Safari, Firefox, google chrome, internet explorer etc. All user interfaces can be reached dynamically on the server side and can fully update the client pages in real time, giving the user an instant response.

4.1.1 Static Pages
These pages make the website for the general audience, who are not part of this application. This page is designed to pick a desired the login if you are either a student, admin, or dietitian. The main user buttons are the home page, About, BMI Calculator, Survey, and login page. The BMI calculator allows anyone to use the calculator, once information is inserted, but redirects to the login page to create a meal plan.
4.1.2 Account Login

The login page is the main page that will be used by all the users of the system. For each different end user, the login page provides user logins based on the following attributes:

- Admin requires admin ID and Password
- Student requires User ID and Password
- The Health Advisor Dietitian requires an Email ID and Password for security purposes.
4.1.3 Login

Figure 9: Admin Login Page

Figure 10: User Login Page
4.1.4 Add Dietitian

The Admin has the ability to add a new dietitian to the system by adding a few variables including, name, contact number, email, gender, and password. The system then validates the creation by sending a message saying Dietitian has been created.
4.1.5 Student Details

The admin can view all student details and look at information about Student ID, Name, Address, Contract number, Email, and Health History. The purpose of this is to create a detailed analysis and record of all users.
4.1.6 BMI Calculator

The Admin and Student user have access to the calculator in which the user enters the weight and height and develops an average BMI score. A user must login to create a custom meal plan.
4.1.7 User Registration

The student must register by clicking the sign-up button before logging into the system. The registration form is mandatory to view resource tools within the system. The Registration form consists of basic information about personal information to complete an account and also health information. Once the registration is complete, the system redirects the user to login with given account information.
4.1.8 Edit User Information

The student has access to personal information regarding the student account. This is in case of any change of personal information the student may call the admin to adjust these changes.
4.1.9 Diet Plan Scheduling

Once the student generates a BMI score, the system then asks the student to create a diet plan. The schedule below reflects what a diet plan schedule may consist of for a student requesting a meal plan.
4.1.10 Sleep Tips

[2] Sleep plays an essential role in your health and wellbeing throughout your life. Getting enough good quality sleep has many benefits, including protecting your physical and mental health, quality of life, and personal safety. As a part of the Health and Wellness System mission, we bring awareness and share Tips weekly.
4.1.11 Calorie Intake

The Health & Wellness system also uses a 3rd party resource widget from calculator tech.com. The tool lets user enter user information pertaining to their health and activity and lets the user know what they should be eating to maintain the weight and lose weight. This tool is a resource that contributes to monitoring daily activities.
4.1.12 Feedback

Feedback usually compares current behavior with predetermined goals and gives back information describing the gap between actual and intended performance. [1] The H&WS provides a designated link for many of its users to write any feedback or suggestions. This is then routed to the proper administrator.
4.1.13 View Request
The Dietitian receives a notification regarding and requests to create a meal plan. The list is continuously updated to provide up to date information and real-time analysis on food consumption and feedback driven results.
4.1.14 Create Diet Plan
Once the dietitian receives a request for a diet plan, the dietitian updates the plan to include the following meal intake per each breakfast, lunch, snack, and dinner. This will help prep the user for a week full of meal prep with the appropriate nutrition for the user’s body. Once the dietitian completes the list, it is then created and sent to the student.
4.1.10 Create Diet Plan

Once the dietitian receives a request for a diet plan, the dietitian updates the plan to include the following meal intake per each breakfast, lunch, snack, and dinner. This will help prep the user for a week full of meal prep with the appropriate nutrition for the user’s body. Once the dietitian completes the list it is then created and sent to the student.
4.2 Technical Feasibility

In this step, the system is verified whether the proposed system is technically feasible or not. i.e., all the technologies required to develop the system are available readily or not. Technical Feasibility determines whether the organization has the technology and skills necessary to carry out the project and how this should be obtained. The system can be feasible because of the following grounds[5]:

- All necessary technology exists to develop the system.
- This system is too flexible, and it can be expanded further.
This system can give guarantees of accuracy, ease of use, reliability, and data security [5.]

This system can give an instant response to inquire.

The H&W system is technically feasible because all the technology needed for the project is readily available.

4.3 Economic Feasibility

Economically, this project is entirely feasible because it requires no additional financial investment, and concerning time, it's entirely possible to complete this project in 6 months. In this step, we verify which proposal is more economical. It must determine whether it is worthwhile to process with the entire project or whether the benefits obtained from the new system are not worth the costs [6]. Economic benefits must be equal or exceed the costs. In this issue, we should consider:

- The cost to conduct a full system inspection.
- The development tools needed.
- The cost of maintenance for the system.

The Health & Wellness system is economically feasible because the cost of development is extremely minimal when compared to the financial benefits of the application.

4.4 Operational Feasibility

In this step, the system is verified for different operational factors of the proposed systems like workforce, time, etc., whichever solution uses less operational resources, is the best operationally feasible solution. The solution should also be operationally possible to
implement. Operational Feasibility determines if the proposed system satisfied user objectives could be fitted into the current system operation [6].

- The users wholly accept the methods of processing and presentation since they can meet all user requirements.
- The users have been involved in the planning and development of the system [6].
- The proposed system will not cause any problem under any circumstances.

The system is operationally feasible because the time requirements and personnel requirements are satisfied.
CHAPTER 5: TESTING

5.1 Black box testing
In clearing house across various modules, this testing was performed to check the following:

a) Establishing communication with the database for handling request and response.

b) Verification of OLE-DB providers (ADO) in functionality

c) Parameters passing and report generation used from the application with crystal report.

5.2 White box testing
All the statements included in the code across various modules were tested to find none of the statements where overlooked or skipped from execution [9]. This enabled isolating of errors that would have otherwise occurred and would have resulted in abnormal terminal or exceptions thrown. The test was deeply tested in end user’s responsibility, the insured party, ailments, procedures, and applied payment modules.

5.3 String Testing
The application was tested for inputs about student data, responsible party, insured party for strings such as name, relation, student details, health details, meal plan details, and dietitian information, was tested for the following:

A. null data

B. string length

C. data format

D. alphanumerical characters
Also, numeric inputs were tested for invalid characters, invalid data format, size of the input data, and the data type being handled.

As the project is on a bit large scale, it’s crucial to test to make successful. If each component works appropriately in all respect and gives the desired output for all kind of inputs, then the project is said to be successful. So, the conclusion is to make the project successful, it needs to be tested.

The testing done was system testing, checking whether the user requirements were satisfied. The code for the new system has been written entirely using ASP .NET with C# as the coding language, C# as the interface for front-end designing. The new system has been tested, and all the applications have been verified from every nook and corner of the user.

Although some applications were found to be in error, these applications have been corrected before being implemented. The flow of the forms is very much following the actual flow of data.
5.4 Levels of Testing
To reveal the mistakes, present in various stages, levels of testing was used. The key dimensions of testing are:

- Client Needs
- Requirements
- Design
- Code

→ Acceptance Testing
→ System Testing
→ Integration Testing
→ Unit Testing

A progression of testing is accomplished for the proposed framework before the framework is prepared for user acceptance testing.

The steps involved in Testing the H&W System are:

5.5 Unit Testing
Unit testing focuses verification efforts around the littlest unit of the application structure, the module. This is otherwise called "Module Testing." The modules are tried independently [9]. This testing completed amid the programming stage. In this testing, every module is observed to work acceptably as respects to the normal output from the module.

5.6 Integration Testing
Information can be netted over an interface; one module can have unfriendly endeavors on another [9]. Integration testing is methodical trying for development of the program
structure while in the meantime directing tests to reveal errors related to the interface. The goal for this system is to take unit tried modules and assemble a program application. Every one of the modules in the system is joined and tried all in all.

5.7 System testing
System testing was accurately performed efficiently for live operation commences. System testing is vital to the success of the H&W system. It makes a logical assumption that if all the parts of the system are correct, the goal was successfully achieved [9].

5.8 Validation Testing
After integration testing application is assembled as a package, interfacing errors have been uncovered and corrected, the validation test begins [9]. When testing for validation, it can be defined in many ways. However, the simple definition is that validation succeeds when the software function in a manner that can reasonably be expected by the customer. Once the validation test has been finalized, one of two possible conditions exists. One is the function or performance of characteristics that conform to specifications and are then accepted, and the other is a deviation from the specification is uncovered, and a deficiency list is created. The system has been tested by using validation testing and found to be working satisfactorily.

5.9 Output Testing
After performing validation testing, the following stage is output testing of the proposed framework since no framework could be helpful on the off chance that it doesn't deliver the required output in the predetermined format [9]. Receiving information about the configuration required tests the outputs created by the framework. Here the output format is analyzed in two different ways, one is on the screen, and the other is through
visualization. The output format on the screen is observed to be accurate as the format was designed in the framework design and planning stage as according to the student needs and requirements. The output comes as the functional requirements by the users. Thus, output testing does not result in any rectifications in the system.

5.10 User Acceptance Testing

User acceptance of a system is the critical factor of the success of any system. The system under study is tested for Student acknowledgment and acceptance by consistently staying in contact with the prospective H&W system users at the time of developing and adjusting updates as required.

Different Test Cases

<table>
<thead>
<tr>
<th>Action</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Admin Login</td>
<td>Here, the admin of dietitian website put his credentials in the Login page.</td>
</tr>
<tr>
<td>2. Login</td>
<td>Here when User enter a user name and password it shows the main application on the window.</td>
</tr>
<tr>
<td>3. New Registration</td>
<td>Here, the user put his personal details to get registered on this website.</td>
</tr>
<tr>
<td>4. View Diet</td>
<td>Here, user can view appropriate diet as per specification</td>
</tr>
<tr>
<td>5. Feedback Form</td>
<td>Here, the user fills in a feedback</td>
</tr>
</tbody>
</table>
CHAPTER 6: CONCLUSION

This chapter summarizes the project’s achievements. It also describes the conclusions derived from it.

6.1 Summary

This project was completed to apply practical and technical skills learned in the MSIST program. It helped develop the framework for a solution to a business problem that happens every year when students lose track of their wellness and health. This is commonly referred to as the “freshman 15”. This weight gain can contribute to a risk of being overweight and obese later in life. Due to the difficulty associated with weight loss, strategies are needed to combat weight gain in college students.[7] It’s important to advise food choices according to health.

6.2 Conclusion

The Health and Wellness System is a web-based application in which has been developed with end users consisting students of the California State University, San Bernardino. The purpose of this application has been to provide an efficient and useful method to develop resource tools within one platform for the University. Students and the University can benefit tremendously from this web application, since they will be able to monitor their health, provide weekly insight tips, etc. The application is planned to be funded through advertisements and in partnership with the student health center and recreation center.

The web application has been developed to provide a user-friendly interface that can be easily navigated through a mobile device or computer. Students do not need to be experts
to use this application as the many with no knowledge in technology are easily able to access the application. Web-based health and wellness is the new era of personal lifestyle decisions.

Several open sources technologies such as .NET, SQL, C# have been used throughout the development process and can be easily installed. This application results in the completion of the mission to provide an exceptional online system that improves awareness of the health of the community. The H&W System can help any individual gain more insights into how they can make wise food choices, and support staying healthy.

Creating a healthy environment can help an individual while guiding healthy behaviors. The next step of this project will be to enhance the projects ability to capture a broader audience. This project will take into consideration surrounding restaurants and fitness centers. A recommendation for capturing one’s health is QR codes. Scanning a QR code at each restaurant or after each workout will allow the user to track and monitor any calories or weight loss data. Considering that sometimes we forget to enter data, this will eliminate the possibility of inaccurate or missing data. In addition, further research and analysis should be evaluated when a dietitian is developing a specific tailor-made meal plan according to changing behaviors towards a healthy lifestyle. In conclusion, Individual consumers need to be more aware and educated about their individual dietary needs and devise their dietary strategies for food choice according to their health [8].
APPENDIX

Mission

The mission of the Health & Wellness System is to provide an exceptional online system that improves awareness of the health of the university community.

Vision

Improving lifestyle, empowering the university through health, wellness, and healing.

Values

- **Initiative**
  - Continuously learn and improve our system with the latest technology. Proactive in earning and maintaining the trust of all users.

- **Communication**
  - Constant communication and alerts regarding compliance and privacy.

- **Service**
  - The H&W System is committed to providing superior user satisfaction that ensures the user an exceptional experience, by making the experience unique and rewarding.

- **Stewardship**
  - Responsible stewardship of the financial, human, and technological resource of the system.

- **Quality & Safety**
  - Quality care is expected; the H&W System strives to offer an unparalleled level of care by continually seeking ways to improve Student’s data.
Competitive Advantage

*Delivery:* Traditionally, one has to travel for consulting in his/her campus clinic, which consumes time as well as it proves frenetic in transportation. Implementation of this system in the real world will reduce the efforts of traveling for students who commute and will save time.

*Security:* Student data is in one, easily accessible place.

*Value:* New, cost-effective delivery to resources and students that don’t have it. Focus on maximizing value (lower cost and higher quality) of services delivered.

*User-friendly:* Enable live chats, virtual visits, and online blog form. Creates a user-friendly website and builds a connection that engages users.

6.2 Project Advantages

1. Real-time health and wellness professional consulting for diet plans, wellness, and health.

2. This system provides the user details and awareness of the nutrient constitution in the body if required with charts.

3. Saves money and very useful and give accurate results as it is coded with keeping diet chart in mind.

4. There is an alternative diet chart provided by the system if the user doesn't like any.

6.3 Disadvantages:
The manual effort, students may forget to enter data for a prolonged period delaying progression results.
It consumes time more for processing activities for student’s wellness efforts.

There is a probability of getting incorrect results within the application.

Sometimes it may not be user-friendly; for example, students may not have access while offline.

Maintenance of the system is challenging concerning updates and constant monitoring.

The end user must have accurate details while entering fields like age, height, weight.

Contrarily this system would give results that are not suitable for a user of uncertainty about what they entered.

6.4 Applications:

- Schools can use this system to make appropriate wellness recommendations to Students.
- This system can be used in colleges technology classes for teaching and practicing purposes so that many students can learn from it.
- This system can also be utilized in the gym, particularly for calculating the student’s calories and diet plans.

6.5 Features

1) **Load Balancing:**

   The server will be limited to the time of admin access since the system will be available only the admin logs.

2) **Easy Accessibility:**

   Records can be effectively retrieved easily and used to store other data, individually.

3) **User-Friendly:**
The Website will be giving a very user-friendly approach for all user

4) **Efficient and reliable:**
   Keeping up the all verified and database on the server, which will be available as indicated by the client necessity. Along with no upkeep cost will be productive when contrasted with putting away all the client information on a spreadsheet or physically in the record books.

5) **Easy maintenance:**
   Artificial Intelligence Dietitian website is designed as a natural way. So, maintenance is also easy.
6.6 Project Planning

Figure 24: WBS

<table>
<thead>
<tr>
<th>Task Mode</th>
<th>Task Name</th>
<th>Duration</th>
<th>Start</th>
<th>Finish</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Brain Storming - Tentative Topics</td>
<td>7 days</td>
<td>Sat 3/30/19</td>
<td>Mon 4/8/19</td>
</tr>
<tr>
<td></td>
<td>Requirement Gathering #1</td>
<td>5 days</td>
<td>Tue 4/9/19</td>
<td>Sun 4/14/19</td>
</tr>
<tr>
<td></td>
<td>Project write up submit fist draft of paper</td>
<td>11 days</td>
<td>Mon 4/15/19</td>
<td>Sat 4/27/19</td>
</tr>
<tr>
<td></td>
<td>Design build &amp; UML Diagram with Use cases</td>
<td>3 days</td>
<td>Sun 4/28/19</td>
<td>Tue 4/30/19</td>
</tr>
<tr>
<td></td>
<td>Development database and coding</td>
<td>4 days</td>
<td>Wed 5/1/19</td>
<td>Sat 5/4/19</td>
</tr>
<tr>
<td></td>
<td>Testing Quality Assurance fix errors</td>
<td>4 days</td>
<td>Sun 5/5/19</td>
<td>Wed 5/8/19</td>
</tr>
<tr>
<td></td>
<td>Implementation Final</td>
<td>4 days</td>
<td>Thu 5/9/19</td>
<td>Tue 5/14/19</td>
</tr>
<tr>
<td></td>
<td>Revise first draft of paper</td>
<td>16 days</td>
<td>Wed 5/15/19</td>
<td>Wed 6/5/19</td>
</tr>
<tr>
<td></td>
<td>Oral Presentation</td>
<td>1 day</td>
<td>Sat 6/8/19</td>
<td>Sat 6/8/19</td>
</tr>
</tbody>
</table>

Figure 25: Gantt Chart
6.7 Costs

The H&WS system will be commercialized to all audiences at no cost. With the competitive advantage for online affordable health and wellness, the cost per analysis, unlike other competitors averaging from $50 per visit. The H&WS system will be provided as an open source tool for active students on campus and the community. The system will generate revenue through advertisements paid ads, google analytics and partnerships with school departments.
### 6.8 Functional and Non-Functional Requirements

#### Table 3: Requirements

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Type</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>The H&amp;WS shall create new user accounts.</td>
<td>• Registration</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Functional</td>
<td>Must-Have</td>
</tr>
<tr>
<td>R2</td>
<td>The H&amp;WS shall log on users both the nutrition specialist, admin, and student to their prospective account.</td>
<td>• Authentication</td>
<td>Must-Have</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Functional</td>
<td></td>
</tr>
<tr>
<td>R3</td>
<td>The H&amp;WS shall validate credentials.</td>
<td>• Authentication</td>
<td>Must-Have</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Functional</td>
<td></td>
</tr>
<tr>
<td>R4</td>
<td>The H&amp;WS shall monitor the dietitian users log.</td>
<td>• Product</td>
<td>Must-Have</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Functional</td>
<td></td>
</tr>
<tr>
<td>R5</td>
<td>The H&amp;WS shall set a minimum student requirement.</td>
<td>• User Interface</td>
<td>Should-Have</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Functional</td>
<td></td>
</tr>
<tr>
<td>R6</td>
<td>The H&amp;WS shall set a static page for non-registered users.</td>
<td>• User Interface</td>
<td>Should-Have</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Functional</td>
<td></td>
</tr>
<tr>
<td>R7</td>
<td>The H&amp;WS shall search for specific BMI ratios by using user inputs.</td>
<td>• Product</td>
<td>Should-Have</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Functional</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
</tbody>
</table>
| **R8** | The H&WS shall display health analysis options by student inputs. | • Product  
• Functional | Should-Have |
| **R9** | The H&WS shall display the proper diagnosis and analysis. | • Product  
• Functional | Must-Have |
| **R10** | The H&WS shall allow users to provide any relevant health history, including allergy’s, diabetes, etc. | • Product  
• Functional | Must-Have |
| **R11** | The H&WS shall require user details access to health resource tools. | • User Interface  
• Functional | Must-Have |
| **R12** | The H&WS shall calculate a BMI and determine users BMI score | • User Interface  
• Functional | Must-Have |
| **R13** | The H&WS shall allow the user to “Calculate BMI” and then ask to log in for a custom meal plan. | • User Interface  
• Functional | Must-Have |
| R14 | The H&WS shall notify the user, admin, and dietitian when the order is complete. | • Order  
• Functional | Must-Have |
|-----|---------------------------------------------------------------------------------|----------------|--------------|
| R15 | The H&WS shall send a copy of the current BMI score and request to order a custom meal plan. | • Order  
• Non-Functional | Must-Have |
| R16 | The H&WS shall allow the user to view fitness calorie intake. | • Order  
• Non-Functional | Must-Have |
| R17 | The H&WS shall enable allow the user to cancel an order before the order is being dispatched. | • Order  
• Non-Functional | Must-Have |
| R18 | The H&WS shall notify the admin if the order is canceled | • Order  
• Non-Functional | Must-Have |
| R19 | The H&WS shall refund the user the order. | • Order  
• Non-Functional | Must-Have |
| R20 | The H&WS shall accept and process the payment with 20 seconds in 95% of cases. This might not always be achievable because of the Internet delay by the card processing company. | • Performance  
• Non-Functional | Should-Have |
<table>
<thead>
<tr>
<th>R21</th>
<th>The H&amp;WS shall log in the user and the dietitian within 5 seconds.</th>
<th>Performance</th>
<th>Non-Functional</th>
<th>Should-Have</th>
</tr>
</thead>
<tbody>
<tr>
<td>R22</td>
<td>The H&amp;WS shall display research result in 6 seconds.</td>
<td>Performance</td>
<td>Non-Functional</td>
<td>Should-Have</td>
</tr>
<tr>
<td>R23</td>
<td>The H&amp;WS shall the copy of order confirmation to the Student within 30 seconds after the payment.</td>
<td>Performance</td>
<td>Non-Functional</td>
<td>Should-Have</td>
</tr>
<tr>
<td>R24</td>
<td>The H&amp;WS shall monitor and update user information.</td>
<td>Performance</td>
<td>Non-Functional</td>
<td>Must-Have</td>
</tr>
<tr>
<td>R25</td>
<td>The H&amp;WS shall store all transaction data.</td>
<td>Availability</td>
<td>Non-Functional</td>
<td>Must-Have</td>
</tr>
<tr>
<td>R26</td>
<td>The H&amp;WS shall be accessible 24 hours a day 360 days.</td>
<td>Availability</td>
<td>Non-Functional</td>
<td>Must-Have</td>
</tr>
<tr>
<td>R27</td>
<td>The H&amp;WS shall allow 20,000 transactions per day.</td>
<td>Capacity</td>
<td>Non-Functional</td>
<td>Must-Have</td>
</tr>
<tr>
<td>R28</td>
<td>The H&amp;WS shall allow 20 transactions per second at peak time.</td>
<td>Capacity</td>
<td>Functional</td>
<td>Should-Have</td>
</tr>
<tr>
<td>R29</td>
<td>The H&amp;WS shall use a browser at its user interface.</td>
<td>Compliance To-Standards</td>
<td>Must-Have</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R30</td>
<td>The H&amp;WS shall support the latest version of web browsers.</td>
<td>• Non-Functional</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R31</td>
<td>The H&amp;WS shall organize the list of illnesses by symptoms.</td>
<td>• Compliance To-Standards Functional</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R32</td>
<td>The H&amp;WS shall allow users to recover the password by clicking &quot;forgot password.&quot;</td>
<td>• Product Functional</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R33</td>
<td>The H&amp;WS shall be suitable enough for various devices.</td>
<td>• Authentication Functional</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R34</td>
<td>The H&amp;WS shall accept all primary forms of payment.</td>
<td>• Payment Functional</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R35</td>
<td>The H&amp;WS shall validate with the card processing company.</td>
<td>• Payment Functional</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R36</td>
<td>The H&amp;WS shall check the Student’s personal information before processing the payment</td>
<td>• Payment Functional</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R37</td>
<td>The H&amp;WS shall automatically calculate taxes and fees.</td>
<td>• Payment Functional</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| R38 | The H&WS shall calculate a monthly quota. | • Payment  
• Functional | Must-Have |
|-----|------------------------------------------|----------------|----------|
| R39 | The H&WS shall secure all forms of locations and communication. | • Security  
• Non-Functional | Must-Have |
| R40 | The H&WS shall provide two-step verification. | • Security  
• Non-Functional | Must-Have |
| R41 | The H&WS shall provide a security audit trail | • Security  
• Non-Functional | Must-Have |
6.9 Source Code

Master Page.aspx

```xml
<%@ Master Language="C#" AutoEventWireup="true" CodeFile="MasterPage.master.cs" Inherits="MasterPage" %>
<head runat="server">
<title></title>
</head>
<asp:ContentPlaceHolder id="head" runat="server">
</asp:ContentPlaceHolder>

<link rel="stylesheet" href="style.css" type="text/css" />

<style>
    .sidebar
    {
        -webkit-border-radius: 10px;
        -moz-border-radius: 10px;
        background-color:White;
    }
    .auto-style2 {
        height: 16px;
        font-family: "Times New Roman", Times, serif;
        font-size: Medium;
        font-variant: normal;
        font-style: italic;
        color: #000080;
    }
    .newStyle1 {
        font-family: "Times New Roman", Times, serif;
        font-size: small;
        font-variant: normal;
        font-style: italic;
        color: #000080;
    }
    .auto-style3 {
        height: 59px;
    }
    .auto-style4 {
        font-size: medium;
    }
</style>
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class MasterPage : System.Web.UI.MasterPage
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (Session["fname"] == "admin")
        {
            Panel1.Visible = false;
            Panel2.Visible = true;
            Panel3.Visible = false;
            Panel4.Visible = false;
        }
        else if (Session["fname"] == "user")
        {
            Panel1.Visible = false;
            Panel2.Visible = false;
            Panel3.Visible = true;
            Panel4.Visible = false;
        }
        else if (Session["fname"] == "Dietitian")
        {
            Panel1.Visible = false;
            Panel2.Visible = false;
            Panel3.Visible = false;
            Panel4.Visible = true;
        }
        else
        {
            Panel1.Visible = true;
            Panel2.Visible = false;
            Panel3.Visible = false;
            Panel4.Visible = false;
        }
    }
}
protected void Button1_Click(object sender, EventArgs e)
{
}

protected void Button2_Click(object sender, EventArgs e)
{
    Session.Abandon();
    Response.Redirect("Login.aspx");
}
<Columns>
    <asp:BoundField HeaderText="Type" DataField="Type"/>
    <asp:BoundField HeaderText="Time" DataField="Time"/>
    <asp:BoundField HeaderText="Food Item" DataField="FoodItem"/>
    <asp:BoundField HeaderText="Note" DataField="Note"/>
</Columns>
</asp:GridView>
</td>
</tr>
<tr align="center" style="color: #990033">
    <asp:Label ID="Label2" runat="server" style="font-family: gadugi; color: #CC0000" Text="Request a Personalized Diet Plan to Dietitian" Visible="False"></asp:Label>
</td>
</tr>
<tr align="center" valign="top" class="auto-style4">
    <br />
    Note;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp; </span>
    &nbsp;<asp:TextBox ID="TextBox1" runat="server" Height="80px" TextMode="MultiLine" Width="30%"></asp:TextBox>
    <br />
</td>
</tr>
<tr align="center">
    <asp:Button ID="Button1" runat="server" Height="49px" OnClick="Button1_Click" style="font-family: gadugi; font-size: large" Width="172px" />
</td>
</tr>
<tr align="center">&nbsp;</td>
</tr>
</table>
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Data.SqlClient;
using System.Data;
using System.Configuration;

public partial class PersonalisedPlan : System.Web.UI.Page
{
    SqlConnection con = new SqlConnection(@"Data Source=.;Initial Catalog=Diet;Integrated Security=True");
    protected void Page_Load(object sender, EventArgs e)
    {
        if(Session["Add"] == "Data")
        {
            Page.ClientScript.RegisterStartupScript(GetType(), "msgbox", "alert('Success. Diet Plan request sent');", true);
            Session["Add"] = "";
        }
        string str = "select type, time, fooditem, note from dietplan where uid='" + Session["Uid"].ToString() + "'";
        SqlDataAdapter da = new SqlDataAdapter(str,con);
        DataSet ds = new DataSet();
        da.Fill(ds);
        if(ds.Tables[0].Rows.Count>0)
        {
            GridView1.DataSource = ds;
            GridView1.DataBind();
            GridView1.Visible = true;
            Button1.Text = "Resend Request";
        }
        else
        {
            GridView1.Visible = false;
            Button1.Visible = true;
            Label2.Visible = true;
            Button1.Text = "Request";
        }
    }

    protected void Button1_Click(object sender, EventArgs e)
    {
    }
string str = "select uid,name,email,mob,HealthHistory,date from dietrequest where uid = '" + Session["Uid"].ToString() + '"';
SqlDataAdapter da = new SqlDataAdapter(str, con);
DataSet ds = new DataSet();
da.Fill(ds);
if (ds.Tables[0].Rows.Count > 0)
{
    string dele = "delete from dietrequest where uid='" + Session["Uid"].ToString() + '"';
    SqlCommand cmd2 = new SqlCommand(dele, con);
    con.Open();
    cmd2.ExecuteNonQuery();
    con.Close();
}
else
{
    string ins = "insert into dietrequest values('" + Session["Uid"].ToString() + "," + Session["Name"].ToString() + "," + Session["Email"].ToString() + "," + Session["Mob"].ToString() + "," + Session["HealthHistory"].ToString() + "," + Session["Ftype"].ToString() + "," + DateTime.Now.ToString("yyyy-MM-dd HH:mm") + "," + TextBox1.Text + ",");
    SqlCommand cmd = new SqlCommand(ins, con);
    con.Open();
    cmd.ExecuteNonQuery();
    con.Close();
    Session["Add"] = "Data";
    Response.Redirect("PersonalisedPlan.aspx");
}

SLEEP.ASPX


<asp:Content ID="body" runat="server" ContentPlaceHolderID="ContentPlaceHolder1">
    <div>
        <br />
        <asp:Label ID="Lab1l" runat="server" Font-Size="X-Large" Font-Underline="True" Text="Sleep Tips"></asp:Label>
        <br />
        <br />
    </div>
</asp:Content>
VIEWUSER.ASPX


<asp:Content ID="Content1" runat="server" contentplaceholderid="head">
  <style type="text/css">
    .auto-style4 {
      margin: 0px;
    }
  </style>
</asp:Content>

<asp:Content ID="Content1" runat="server" contentplaceholderid="head">
  <style type="text/css">
    .auto-style4 {
      margin: 0px;
    }
  </style>
</asp:Content>

<asp:Content ID="Content1" runat="server" contentplaceholderid="head">
  <style type="text/css">
    .auto-style4 {
      margin: 0px;
    }
  </style>
</asp:Content>


<asp:Content ID="Content1" runat="server" contentplaceholderid="head">
  <style type="text/css">
    .auto-style4 {
      margin: 0px;
    }
  </style>
</asp:Content>

<asp:Content ID="Content1" runat="server" contentplaceholderid="head">
  <style type="text/css">
    .auto-style4 {
      margin: 0px;
    }
  </style>
</asp:Content>
< PagerStyle BackColor="#000066" ForeColor="#330099" HorizontalAlign="Center" />  
< RowStyle BackColor="White" ForeColor="#000066" />  
< SelectedRowStyle BackColor="#FFCC66" Font-Bold="True" ForeColor="#000066" />  
< SortedAscendingCellStyle BackColor="#000066" />  
< SortedAscendingHeaderStyle BackColor="#AF0101" />  
< SortedDescendingCellStyle BackColor="#F6F0C0" />  
< SortedDescendingHeaderStyle BackColor="#0066CC" BorderColor="#000066" />  
< Columns >  
< asp:BoundField DataField="ID" HeaderText="ID" />  
< asp:BoundField DataField="Name" HeaderText="Name" />  
< asp:BoundField DataField="Address" HeaderText="Address" />  
< asp:BoundField DataField="Mobile" HeaderText="Contact No." />  
< asp:BoundField DataField="Email" HeaderText="Email ID" />  
< asp:BoundField DataField="Healthhistory" HeaderText="Health History" />  
</ Columns >  
</ asp:GridView>  
<br />  
<br />  
</ div>  
</ asp:Content>  
< asp:Content ID="Content1" runat="server" contentplaceholderid="head">  
<style type="text/css">  
    .auto-style5 {  
        background-color: #CCCCCC;  
        height: 60px;  
    }  
</ style>  
</ asp:Content>

VIEWUSER.ASPX.CS

using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Web;  
using System.Web.UI;  
using System.Web.UI.WebControls;  
using System.Data.SqlClient;  
using System.Data;  

public partial class ViewUser : System.Web.UI.Page  
{  
    SqlConnection con = new SqlConnection("Data Source=.;Initial Catalog=Diet;Integrated Security=True");  
    
    protected void Page_Load(object sender, EventArgs e)  
    {  
        if (!IsPostBack)  
        {  
            SqlCommand cmd = new SqlCommand("SELECT * FROM User", con);  
            SqlDataAdapter da = new SqlDataAdapter(cmd);  
            DataSet ds = new DataSet();  
            da.Fill(ds);  
            GridView1.DataSource = ds;  
            GridView1.DataBind();  
        }  
    }  
}
string s = "SELECT Id,Name,Address,Mobile,Email,Healthhistory FROM User";
SqlDataAdapter da = new SqlDataAdapter(s, con);
DataSet ds = new DataSet();
da.Fill(ds);
if (ds.Tables[0].Rows.Count > 0)
{
    GridView1.DataSource = ds;
    GridView1.DataBind();
    GridView1.Visible = true;
    con.Close();
}
else
{
    GridView1.Visible = false;
    lblMsg.Visible = true;
    lblMsg.Text = "Currently, No User Registered !!";
}

protected void Button1_Click(object sender, EventArgs e)
{
    SqlCommand cmd = new SqlCommand("Select Name from User where Name='' +
TextBox1.Text + ''", con);
    con.Open();
    SqlDataReader dr = cmd.ExecuteReader();
    if (dr.HasRows)
    {
        con.Close();
        string s = "SELECT Id,Name,Address,Mobile,Email,Healthhistory,Ftype FROM User
where Name like '%" + TextBox1.Text + '%'';
        SqlDataAdapter da = new SqlDataAdapter(s, con);
        DataSet ds = new DataSet();
da.Fill(ds);
        GridView1.DataSource = ds;
        GridView1.DataBind();
        lblMsg.Visible = false;
    }
    else
    {
        lblMsg.Text = "Name doesn't Exist !!";
        lblMsg.Visible = true;
    }
}

Third Party Widget Code

<%@ Page Language="C#" MasterPageFile="~/MasterPage.master" AutoEventWireup="true"
CodeFile="Calorie.aspx.cs" Inherits="Calorie" %>

<asp:Content ID="body" runat="server" ContentPlaceHolderID="ContentPlaceHolder1">
<br />
</div>
Figure 26: Database Dietitian
Figure 25: Web Forms
RESOURCES

   https://www.w3computing.com/systemsanalysis/feedback-users-hci/

[2] Why sleep is important | Health Navigator NZ.
   https://www.healthnavigator.org.nz/healthy-living/sleep/why-sleep-is-important/


[4] Unit-1 Introduction to ASP.NET web programming & IDE.
   http://www.darshan.ac.in/Upload/DIETDS/Documents/CE/AWT_Unit_1_20042018_092751AM.pdf

   https://www.ijsr.net/archive/v6i1/ART20164529.pdf


https://www.testingexcellence.com/types-of-software-testing-complete-list/