Hygeia: Fitness Planner

A Project for the Completion of

Introduction to Software Engineering

Presented to:

Prof. Orven Llantos

By

Team Jug-Ah

Angelo Sembrano

Joren Ezra Apal

Silver Gems Serate

Arlene Yosoya

Loraine Faith Paragoso

ABSTRACT

Hygeia Fitness Planner is an application that can give a personal fitness plan to a user. It can generate exercise plan and a diet plan. It allows the user to create personal fitness plan. The plans that are being generated are based on the the user’s health status and on the information of user/users that are gathered such as: gender, age, weight & height.

The fitness plan system offers exercise plan and a diet plan. Different conditions aply in which must be considered on generating a plan. Users can view their personal fitness plan anytime. Once the fitness plan is generated the system is updated and the users can view their personal fitness plan.

INTRODUCTION

The Hygeia Fitness Planner system is the new fitness planner based online. This new system will be able user to acquire personal fitness plan that are suited for their age, gender and health status with just few clicks. This system will also lessen the hassles experienced by the people with it being accessible to World Wide Web.

Hygeia Fitness Planner will also allow users to sign-up for new account for them to be able to log- in and log-out anytime which enables them to view their personal fitness plan.

Chapter 1 : Software Requirements Analysis

In this chapter we are going to dscuss the software requirements that are needed to be fulfilled by the online fitness planner system to make sure that the finished product made by the team followed the users requirements. The chapter includes the use case, features, features acceptance criteria, persona, user stories, user stories acceptance criteria.

* 1. The Charter
     1. Vision

Through an online “Hygeia Fitness Planner”, a

* + 1. Mission

Create a user friendly fitness planner system as a web application.

* + 1. Scope
* The sysem will allow the users to acquire fitness plan such as , exercise plan and diet plan.
* The user will be given a generated fitness plan based on the information by the users that are being gathered. If the
  + 1. Objectives
* To generate fitness plan based on the information of the user that are being gathered.
* To view the generated fitness plan.
* Record the fitness plan that is being acquired by the user.
* To achieve a healthy life style after using the fitness plan.
  + 1. Principles
* User satisfaction is our top priority.
* Generate the right plan to the right person.
  1. Feature Lists

Features for the users.

* New User Sign-up
* User Log-in
* User Log-out
* User Profile/Personal Information
* View Generated Fitness Plan
* View Fitness Plan Records
  1. Feature Acceptance Criteria
     1. Sign-up

User will be redirected to the User Profile page (which still needs to be fill-uped)

* + 1. Log-in

User will be redirected to the homepage with a “logout” option.

* + 1. Log-out

User will be redirected to the homepage with a “log-in” option.

1.3.4. User Profile/Personal Information

The user is obliged to fill-up the user information form( which contains the ff.: name, age, gender, height, weight) in order to acquire plan.

* + 1. View Generated Fitness Plan

The user fitness plan that is being generated according to the users health status is displayed.

* + 1. View Fitness Plan Records

The fitness plan that is being acquired by the user is recorded and displayed.

* 1. Users stories and stories acceptance criteria
     1. As a user I want to sign-up for a new account so that I can sign-in in the fitness planner sytem anytime.
     2. As a user who already has an account , I want to sign-in so that I will be able to acquire fitness plan.
* User is redirected to homepage.
  + 1. As a user who already has signed in the Hygeia fitness planner sysrem, I want to sing-out so that the things that I’ve done with my account remains secure.
* User Health status is being classified and given a a fitness plan.
* User is redirected back to homepage with sign-in option after sign-out.
  + 1. As a user, I can view my Personal fitness plan that is being generated.
* See the exercise plan and diet plan that is being generated.
  + 1. As a user, I can view the fitness plan record.
* User will be shown a page showing the plans that are being acquired and the user health status.

* 1. User Requirements
     1. Register

User is obliged to register in order to have an account and to acquire plan.

* + 1. Log-In

User is required to login if they want to use some functionality of the system like acquiring a fitness plan.the loge in screen will also allow the user to select options.

* + 1. Account Management
       1. New User Registration
* Anyone can register for a new account. During registration the user will have to provide the following information.
* Username, a valid email address and password.
* Personal Information full name, birthday, gender, height and weight.
* The user health status is being determined based on the information inputted.
  + - 1. Personal Profile

The user should have an option to edit their profile anytime.

* + 1. Acquire Plan

With the information given by the user. The health status of the user is being determined. The user can now have a generated plan based on its health status and age and gender. Now the user already acquires plan.

* + 1. Log-off

When a user wants to log-out he can click the logout button. The system should manage to save the changes that occurred in the account of a certain user.

* 1. Use Case Scenario
     1. Sign-up/Register

The purpose of this use case is to describe the procedure to of creating an account in the system

Pre-condition:

* None

Post-condition

* An account is created for the user.

Basic flow:

1. The user is in the homepage
2. The user clicks on “sign-up” button to create a new account
3. The customer enters all the information needed and clicks “ok” button
4. An error message appears if the email address is not valid.
5. An error message appears if the ID number already exist.
   * 1. Log-in

* The purpose of this use is to describe the procedure of logging into the system.

Pre-condition:

The user has created an account already

Post-condition:

The user is logged into the system

Basic Flow:

Under the login section, the user enters their user\_id, email address and password and clicks the “Login” button

* + - If the customer enters an incorrect combination of user\_id and password , and invalid email adsress an error will appear saying that the user\_id and password does not match or the email address you’ve enter s invalid and gives the customer the ability to login again
    1. View Acquired Fitness Plan

The purpose of this use case is to describe the procedure of how user acquire fitness plan.

Pre-condition:

The user has logged in to the system already

Post-condition:

The user is presented with a form to be fill up in order to acquire plan.

Basic Flow:

1. The user will fill up the data needed.

2. After clicking submit. The user health status will be then classified. Weather its underweight, normalweight or overweight. The user age status will also be shown, weather it young adult, adult, etc.

3. The user will then be redirected to the view personal fitness plan page where he can view his exercise plan and diet plan.

* + 1. View Fitness Plan Record

The purpose of this use is to show the history of transaction made by the customer

Pre-conditions:

The customer is logged into the system

Post-condition:

The customer able to see the history of reservation

Basic Flow:

Click on the “Fitness Plan Record” button to see the history/ record of fitness plan being acquired by the user.

* 1. Software/Technologies used in the System

* Python - is a [high-level programming language](http://en.wikipedia.org/wiki/High-level_programming_language). Supports multiple [programming paradigms](http://en.wikipedia.org/wiki/Programming_paradigm), including [object-oriented](http://en.wikipedia.org/wiki/Object-oriented_programming), [imperative](http://en.wikipedia.org/wiki/Imperative_programming) and [functional programming](http://en.wikipedia.org/wiki/Functional_programming) or [procedural](http://en.wikipedia.org/wiki/Procedural_programming" \o "Procedural programming)styles. It features a [dynamic type](http://en.wikipedia.org/wiki/Dynamic_type) system and automatic [memory management](http://en.wikipedia.org/wiki/Memory_management) and has a large and comprehensive [standard library](http://en.wikipedia.org/wiki/Standard_library).

* pgAdmin - is a comprehensive design and management interface for PostgreSQL database.
* Apache HTTP Server - commonly referred to as Apache is a web server application notable for playing a key role in the initial growth of the World Wide Web.
* Github -  is a [Git](http://en.wikipedia.org/wiki/Git_(software)" \o "Git (software)) repository [web-based hosting service](http://en.wikipedia.org/wiki/Shared_web_hosting_service), which offers all of the [distributed revision control](http://en.wikipedia.org/wiki/Distributed_revision_control) and [source code management](http://en.wikipedia.org/wiki/Source_code_management) (SCM) functionality of Git as well as adding its own features. GitHub provides a [web-based graphical interface](http://en.wikipedia.org/wiki/Web_application) and desktop as well as mobile integration. It also provides [access control](http://en.wikipedia.org/wiki/Access_control) and several collaboration features such as [wikis](http://en.wikipedia.org/wiki/Wiki), [task management](http://en.wikipedia.org/wiki/Task_management), and [bug tracking](http://en.wikipedia.org/wiki/Bug_tracking_system) and [feature requests](http://en.wikipedia.org/wiki/Software_feature) for every project.