

FitNoQuit - Software Requirement Specification

1814007 - Bhavik Bhatt, 1814010 - Piyush Chavda,
1814011 - Bharat Choithani, 1814020 - Muskaan Nandu

26 September 2021

1 INTRODUCTION

1.1 Product Overview

Every human being has a different body with different requirements and thus as the saying goes “one size may not fit all”, a single diet and workout routine would not serve all. To solve this issue, “FitNoQuit” aims to provide its users with personalized diet and workout recommendations. Using machine learning algorithms, we aim to provide our users with a customized diet and workout routine that best suits them. These algorithms take into context a ton of user details like height, weight, age, gender, food preference, medical conditions etc. These details enable the algorithm to formulate a plan that is accurate and easy to follow for the users. The users can also browse through the blog section where in a plethora of information with respect to current diet and workout trends, myths regarding health and so on is available. The user is recommended with related blogs using content-based filtering which aims to recommend users with blogs from a category/author that the users previously showed interest in.

To sum up, “FitNoQuit” is a knowledge-based system that aims to provide diet and workout services to its users that is personalized and most apt to the user.

2 SPECIFIC REQUIREMENTS

2.1 External Interface Requirements

2.1.1 User Interfaces

The screen formats and menu structure should be in such a way that users will find easy to use. The product must be user-friendly and inter-active. The interface must be easy to understand. The user interface includes

- **SCREEN FORMATS/ORGANIZATION:** The introductory screen will be the user dashboard from where the user can access different modules of the web-app like diet recommendation, workout recommendation, blog section etc.

- **WINDOW FORMAT/ORGANIZATION:** When the user chooses some other option, then the information pertaining to that choice will be displayed in a new window which ensures multiple windows to be visible on the screen and the users can switch between them.

- **DATA FORMAT:** The data entered by the users will be alphanumeric.

- **END MESSAGES:** When there are some exceptions raising error like entering invalid details, then error messages will be displayed prompting the users to re-enter the details.

2.1.2 Hardware Interfaces

Are as follows:

Source of input: Keyboard, Mouse

To accept data from user like height, weight, age, user preferences and other personal and medical information

Destination of Output: Desktop Screen

Display the personalized workout and diet recommendations along with the option of browsing through blogs via the desktop screen.

2.1.3 Software Interfaces

- Languages: Python, HTML, CSS, JavaScript, PHP
- Framework: Django, BootStrap

- Database: SQLite

2.1.4 Communication Protocols

Usage of simple electronic forms which make use of SMTP and HTTPS protocols.

2.2 Software Product Features

2.2.1 Functional Requirements

The requirements/features that end-users shall expect and will be incorporated in the system are:

1. Registration/Sign Up

New users will be able to access the system by registering or creating an account with “FitNoQuit”.

2. Login/Sign In

Existing users should be able to access the system using their username and password. Premium features will be available to premium registered users.

3. User Profile

Users will be able to edit and view their personal information from the User Profile section.

4. Workout Recommendation

The user will be recommended with appropriate workout and exercise based on their end goal (weight gain, weight loss maintain weight or manage a health condition)

5. Diet Recommendation

Suggest Diet Meals to the users in categories of Breakfast, Lunch and Dinner based on user’s food preferences like Veg/Non Veg/Jain/Vegan, health condition like Thyroid, PCOS, Heart Conditions, etc

6. Blog Suggestions

The users will be suggested with relevant blogs in the Community Blog section using Content-Based filtering based on the previously liked blogs.

7. Logout

The user should be able to safely log out of the system without any loss of information.

2.2.2 Non-Functional Requirements

The basic quality constraints that will be satisfied by the system is as follows:

1. Authorization

The user data will be confidential and only authorized nutritionists will be made available with the data.

2. Speed

The application will respond to user's queries as soon as possible with a minimum waiting time.

3. Portability

The system will be highly portable as it is a web-app and will work efficiently even if the user changes the device.

4. Compatibility

Since the system is a web-app it can be accessed on any devices like laptop, tablet or mobile phones, irrespective of its Operating System.

5. Security

Sensitive information like user passwords will be encrypted by using algorithms like SHA and then stored on the database for verification.

6. Availability

The web-app will be available to all users with internet-connectivity as we plan to host the website.

7. Reliability

The recommendation system would be reliable as we would try to achieve maximum accuracy and get the recommendation model verified by a certified nutritionist.

2.3 Software System Attributes

2.3.1 Reliability

The factors needed to establish the software expected reliability are

- The user inputs should be valid and within the given range.
- Normal execution of the system without glitches
- Appropriate navigation and restricting access to unauthorized users

2.3.2 Availability

The factors guarantee the software's availability includes proper termination and correct input details. Also the resources used for the project development

are Python Certified which speaks of its high quality standards.

2.3.3 Security

Are as follows:

- It must be ensured that access will be provided to the authorized persons through user ID and password.
- Passwords must be Strong.
- No loss of data for any user must be ensured.
- Checks can be performed at regular intervals to ensure data integrity.

2.3.4 Maintainability

The software will be developed by implementing the concept of modularity which in turn reduces the complexity involved in maintaining it. The administrator should have a sound technical knowledge about maintaining the software and further enhancements will be undertaken by the developer

2.3.5 Portability

The web application is portable to any device-mobile or desktop and is adaptable for use on different browsers with different device models and standards.

2.3.6 Performance

FitNoQuit application should be able to respond to the queries submitted by the customer without much delay.

- When a user inputs personal information in order to obtain workout and diet recommendations, the application should return the results in minimum duration.
- The user should be displayed blog recommendations that are aligned with their liking and the loading time for blogs should be less than 5 seconds.

2.4 Database Requirements

The system requires the use of SQLite database to maintain the customer's personal details and their medical information. An entity should be used to denote all the workout plans, diet classes (Veg, Non-Veg, Jain and Vegan) and the number of calories associated with them