Software Requirement Specification Document
Review Report

Reviewed Document	Document Title: Software Requirements Specification - FH Mobile Application. Version 1.5 Prepared by: By: Omar Rivera, Andrew Poirier, Daven Amin, Rick Rejeleene, Brian Strattard
Reviewers	Qutaiba Albluwi, Younghun Chae, Tripti Garg, Zach Oliveira

Document Layout, Format and Structure:

Strengths:
The document is well formatted. There is also an overall consistency in the format of headers, contents and bullets. The document is easy to read and navigate through.

Recommendations and Suggestions:
1. Headers for sections 2.1, 2.5, and 2.6 are provided without any content. Either these headers could be deleted or some content could be added
2. There are numerous spelling and grammatical mistakes. Examples include:
    a. Bullet 3, Section 3.1.3.2
    b. Sentence 1, paragraph 1, section 6.1
    c. Sentence 2, paragraph 2, section 6.1
    d. Sentence 1, paragraph 3, section 6.1
3. Spacing between consecutive sections is not consistent. See for example pages 11, 14 and 15.
4. The table under section 1.4 does not have a label.
5. It is not clear if "Appendix A" is an appendix attached to the SRS document or a future deliverable.
6. Traceability issue: The revision table does not specify the actual changes. It just references the section headers. As the project progresses, it would be difficult to trace back changes using this mechanism.

Introduction

Strengths:

The overview of the project is well described in this section, such as the versions of tools, the description of each term, and the references.

Recommendations and Suggestions:

1. The introduction provides description of the product and its main functionality. However, the need for the system is not
2. The term "HTTPS" is referenced in Section 4.4; however, the acronym was not defined in section 1.4
3. It might be easier to find a term if the table is sorted in order of alphabets.

Overall Description

Strengths:

Since this section specified the required hardware/software with versions, the constraints are clear.

Recommendations and Suggestions:

1. The SRS provides a single Data Flow Diagram (DFD) which depicts a specific flow of data. There is a need to provide a higher level diagrams that illustrate the system components, higher level communication and high level architecture.
2. It seems that the Data Flow Diagram represents only one scenario of making bills (the penalty). The SRS is missing a depiction of other important scenarios such as creating and updating workout schedules by clients, and validating schedules by trainers.
3. The following are in reference to the Data Flow Diagram (DFD)
    1. The notation of the relationship between the Bank Records and the Calculate New Balance is missing
    2. The notation of the relationship between the Bank Record and the Update Balance is missing.
    3. The notation of the relationship between Get User File and the User Record is missing.
    4. It is not clear which part is the role of the server and which part is the role of the client.

Functional Requirements

Strengths:

Overall, the use cases were very descriptive and easy to follow. It made it simple to understand how the application will run. The uses cases were exhaustive- there was only one possible suggestion for an additional use case (adding a payment method as its own case).

1. Section 2.2 lists the functional requirements in a paragraph. It is better to list the requirements in bullets/numbers to facilitate design, testing and traceability. it will also enhance the readability of the document.
2. The following points are related to the Use Cases:
    a. In the use cases, the preconditions are stated between parentheses. It is better to highlight it with the tag: pre-condition.
    b. In the exception scenarios for each case are missing the situation when the network is not valid.
    c. A case of Schedule modification is not presented.
    d. In section 3.1.1.1, the brief description is very similar to the step by step descriptions of the use case. They seem redundant.
    e. In section 3.1.1.2, do you want the user to verify the password they have chosen? Many websites and apps do this, and it may be something to consider. This would also result in an additional exception scenario, where the passwords do not match.
    f. In section 3.1.3, you are assigning a payment method to a specific workout schedule. Should the ability to add a payment method be it's own case? Then you could have people choose a pre-existing method instead of adding a new one every time.
    g. In section 3.1.3.2, the limit of number of hours user can schedule their workout is missing.
    h. In section 3.1.4.3, the situation in which user don't accept the workout schedule is missing.
    i. In Use Cases it is not clear which user will perform the task. e.g. "The user selects the option to validate another user's workout.", "the user specifies the login name of another user who will provide verification that workout metrics have been met."

External Interface Requirements

Strengths:

The descriptions of how the interfaces work with other parts of the project are very clear (e.g. how the software interface connects to the database.

Recommendations and Suggestions:

Database Requirements

In section 4.5.2, it should say that *correct* data must be entered before a credit or penalty should be assessed. Otherwise it implies that any credit card data (including incorrect data) can be entered and allow the user to continue

Non Functional Requirements

Strengths:

The performance requirements specified the actual performance time.

Recommendations and Suggestions:

Security

1. The security of customers finance records is not clear. If all trainers have authority to access their customer finance details, then it could be potentially loss of highly confidential customer information.
2. Non-functional requirement: security. The system mentions the need to save login credentials in a secure location, but does not mention the need to secure credit card and bank information. This should be highlighted in section 5.3

Portability:

There is a reference to portability in the last sentence of Section 5.4. Portability is the ability of the software to work under different environments. However, the description addresses the speed and storage limitations.

Sections 6-8:

Strengths:

Section 7 clearly describes the milestones and schedules with a graphical depiction. Moreover, the descriptions about the inventory and team member background are impressive and give the project more reliability.

Recommendations and Suggestions:

1. Section 6.1 mentions: "system configuration parameters". However, this term has not been defined.
2. The heading of Section 8 is "Disaster Issues". The term "risk management" or "crisis management" are more conventional terms.