SOFTWARE REQUIREMENTS SPECIFICATION

for

iSport

Version 1.0 approved

Prepared by

1650940 Jiang Xiaohu
1650932 Xu Jingnan
1651058 Wang Yicheng

School of Software Engineering
Tongji University

November 21, 2019

Contents

Revision History

Name	Date	Reason For Changes	Version
Jiang Xiaohu	2019.11.1	Finish Introduction Part	v1.0
31	32	33	34

1 Introduction

1.1 SRS Purpose

The purpose of this document is to present a detailed description of iSport. It will explain the purpose and features of iSport, the interfaces of the iSport, the functional and nonfunctional requirements of iSport, what iSport will do, and the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system and will be proposed to the clients for its approval.

1.2 Product Scope

This software system will be a web based system for sports fans, professional athletes, and patients who are under recovering training.

This system will be designed to maximize the exercising efficiency by providing tools to assist in checking and correcting user's wrong postures and recommending training courses customized for users, which would otherwise have to be expensive, time-consuming and labor intensive. By maximizing the user’s training efficiency and convenience the system will meet the needs of sports fans, athletes and injured patients while remaining easy to understand and use.

More specifically, this system is designed to allow a user to imitate the standard exercising postures while observe and correct their mistakes simultaneously with the help of a website.

The software will collect some professional courses in the database, including static and dynamic trainings which means doing exercise according to a set of images or a video and iSport will recommend suitable trainings for users on the basis of their training performance. Courses are classified into exercising courses and recovering courses, aiming to help athletes and patients respectively.

Both visual and audio notification are used in every course of the system to provide eye-catching, user-friendly and clear instructions; the feedback of one's training is proposed once the training is over and the report can be browsed in the report page.

The selection and deletion of one user's favorable course is supported in personal information webpage and one can comment training he/she has taken on comment webpage to provide suggestions to other users.

The personal information registering, changes is allowed via the application options. The system also contains a relational database containing a list of users, training images and videos.

1.3 Intended Audience

This Software Requirements document is intended for:

- Developers who can review project's capabilities and more easily understand where their efforts should be targeted to improve or add more features to it (design and code the application – it sets the guidelines for future development).

- Project testers can use this document as a base for their testing strategy as some bugs are easier to find using a requirements document. This way testing becomes more methodically organized.

- End users of this application who wish to read about what this project can do.

- Clients who delegate the software development to our team, and can check if their requirements are perfectly understood by the developers and whether the functional and nonfunctional requirements are entirely meet. And they can modify some requirements according to this document in later stage.

- Project managers who translate the clients' requirement to the programmers and supervise them to implement the requirements mentioned in the documents and if the clients modify the requirements, the project managers have to negotiate the changes with the clients and modify the document.

- Marketing staff who are responsible to sale and prompt the software should be clear about what iSport can do, what iSport's advantages are, what iSport's competitive power is.

- Service staff who are responsible to solve the customer's problems, they have to know what iSport can do and feedback the technical problems to developers.

- Document writers who are responsible for writing the rest documents in later development stages should follow the requirements specified in this SRS.

The rest part of this SRS document contains the overall description of iSport, and iSport's specific, nonfunctional and other requirements which are shown in chapter2, chapter3, chapter4 and chapter5 respectively. We suggest the readers begin with the overview sections and proceeding through the sections that are most pertinent to them:

- Developers and project testers are recommended to focus on specific and nonfunctional requirements part because theses parts will lead them to build qualified, safe and satisfying application and theses parts are all related to coding (construction and verification stage).

- Clients, project managers and document writers should focus on the entire document since they are responsible for all the requirements specified in this paper.

- Marketing staff have to focus on the functional part of iSport.

1.4 Definitions, Acronyms, and Abbreviations

Table 1.1: Definitions

Definitions, Acronyms, and Abbreviations	
Term	Definitions
User	Someone who interacts with iSport including sports fans, athletes and injured patients who need recovery training.
Sports fans	One of iSport's potential customers who love sports and want to get professional instructions when exercising. Some of them may can't afford the expense of personal coaching or don't have time to go to the gym.
Athletes	One of iSport's potential customers who want to get real-time exercising feedback to improve their performance or who want to get some relaxing training in their spare time to keep a good competitive state.
Injured Patients	One of iSport's potential customers who need recovering training after some treatments, e.g. surgeries. On the one hand, some of them may can't afford the doctor's expensive medical instructions for recovering training. on the other hand, there is no enough doctors or nurses who can instruct and supervise the patients' recovering exercising. But without professional instructions training can be useless or even leads to secondary trauma.
Admin/Administrator	System administrator who is given specific permission for managing and controlling the system, e.g. updating the user's information, uploading new training courses.
Continued . . .	

Table 1.1: (continued)

Definitions, Acronyms, and Abbreviations (continued)	
User Info	User's basic information including user's avatar, account name, tel-number and email address.
Courses	Training courses including normal exercising training and recovering training.
Normal Courses	Training courses which serve the sports fans and athletes.
Recovering Courses	Training courses which serve the injured patients.
Static Courses	Training courses which instruct the users photo by photo.
Dynamic Courses	Training courses which instruct the users according to a standard video.
Appraisal Subsystem	Remark the user's performance by using a grade from 0 -100
Comment Subsystem	User comment on the training courses they have taken to provide reference for other users.
Recommendation Subsystem	A subsystem which will provide some courses for users according to their recent performance.
Exercise Tips	There will be sports tips in the webpage of iSport to prevent users from athletic injuries.
Sport Report	A web page to feedback the user's exercising performance.
Audio Notification	An audio notification will be shown when the user is doing exercise to encourage the user to hold on or notify the user to correct their postures.
Continued . . .	

Table 1.1: (continued)

Definitions, Acronyms, and Abbreviations (continued)	
Visual Notification	A visual notification will be shown when the user is doing exercise, if the user's posture is standard, then the web-frame will turn green to suggest the user to hold on, otherwise the web-frame will be red.
DataBase	A relational database containing a list of user info, training images and videos.
Detection Subsystem	Subsystem to detect the user's postures and draw the user's skeleton. The main model of detection subsystem is PoseNet.
Comparison Subsystem	Subsystem to compare the postures of the user and that of the standard. The subsystem aims to check if the user pass the posture.
Correction Subsystem	Subsystem to calculate where the postures' wrong part are, e.g. left-arm, right-leg, head.
Clients	Group who delegate the development of iSport to the developers and will take charge of the later management of iSport.
Developers	Develop team including project managers, programmers, testers who are responsible for the development of iSport and its later mainteinance and updating.
The End	

1.5 Document Conventions

This document follows MLA Format[1]. Bold-faced text has been used to emphasize sectionand sub-section headings. Highlighting is to point out the references of tables and figures. And italicized text is used to label and recognize special characters or terminologies

The SRS paper is written by latex[4], the packages we used to format the document including $longtable, graphicx, subfig, utf8 - inputenc, hyperref$

and so on.
And we modified the IEEE SRS latex template from Jean-Philippe Eisenbarth's github[5] according to the IEEE standard[2] mentioned in Frank F Tusi'book[3].

1.6 References and Acknowledgements

Standard Reference

The standard we have followed are as follows:

[1] T. Russell, A. Brizee, E. Angeli, and R. Keck, “Mla formatting and style guide,” The Purdue OWL, 2010.

[2] I. S. E. S. Committee et al., “Ieee recommended practice for software re- quirements specifications,” IEEE organization, 1998.

[3] F. F. Tsui, O. Karam, and B. Bernal, Essentials of software engineering. Jones Bartlett Learning, 2016.

Writing Tools Reference

The writing tools we have used are as follows:

[4]L. Lamport, LATEX: a document preparation system: user’s guide and reference manual. Addison-wesley, 1994.

[5] J.-P. Eisenbarth, “Srs latex template under ieee standard,”
http://https://github.com/jpeisenbarth/SRS-Tex.

2 Overall Description

2.1 Product Perspective

<Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. A simple diagram that shows the major components of the overall system, subsystem interconnections, and external interfaces can be helpful.>

2.2 Product Functionality

<Summarize the major functions the product must perform or must let the user perform. Details will be provided in Section 3, so only a high level summary (such as a bullet list) is needed here. Organize the functions to make them understandable to any reader of the SRS. A picture of the major groups of related requirements and how they relate, such as a top level data flow diagram or object class diagram, is often effective.>

2.3 Users and Characteristics

<Identify the various user classes that you anticipate will use this product. User classes may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience. Describe the pertinent characteristics of each user class. Certain requirements may pertain only to certain user classes. Distinguish the most important user classes for this product from those who are less important to satisfy.>

2.4 Operating Environment

<Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must peacefully coexist.>

2.5 Design and Implementation Constraints

<Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer's organization will be responsible for maintaining the delivered software).>

2.6 User Documentation

<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.>

2.7 Assumptions and Dependencies

<List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project, unless they are already documented elsewhere (for example, in the vision and scope document or the project plan).>

3 Specific Requirements

3.1 External Interfaces Requirements

< >

3.2 Functional Requirements

< >

3.3 Behavior Requirements

< >

4 Other Nonfunctional Requirements

4.1 Performance Requirements

<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.>

4.2 Safety and Security Requirements

<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product's design or use. Define any safety certifications that must be satisfied.> <Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements. Refer to any external policies or regulations containing security issues that affect the product. Define any security or privacy certifications that must be satisfied.>

4.3 Software Quality Attributes

<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usabil-

ity. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.>

5 Other Requirements

<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

5.1 Data Dictionary Requirements

< >

5.2 Appendix A: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

5.3 Appendix B: To Be Determined List

<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>