

---

# 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 10, 2019

# Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	SRS Purpose . . . . .	4
1.2	Product Scope . . . . .	4
1.3	Intended Audience . . . . .	4
1.4	Definitions, Acronyms, and Abbreviations . . . . .	4
1.5	Document Conventions . . . . .	4
1.6	References and Acknowledgements . . . . .	5
<b>2</b>	<b>Overall Description</b>	<b>6</b>
2.1	Product Perspective . . . . .	6
2.2	Product Functionality . . . . .	6
2.3	Users and Characteristics . . . . .	6
2.4	Operating Environment . . . . .	6
2.5	Design and Implementation Constraints . . . . .	7
2.6	User Documentation . . . . .	7
2.7	Assumptions and Dependencies . . . . .	7
<b>3</b>	<b>Specific Requirements</b>	<b>8</b>
3.1	External Interfaces Requirements . . . . .	8
3.2	Functional Requirements . . . . .	8
3.3	Behavior Requirements . . . . .	8
<b>4</b>	<b>Other Nonfunctional Requirements</b>	<b>9</b>
4.1	Performance Requirements . . . . .	9
4.2	Safety and Security Requirements . . . . .	9
4.3	Software Quality Attributes . . . . .	9
<b>5</b>	<b>Other Requirements</b>	<b>10</b>
5.1	Data Dictionary Requirements . . . . .	10
5.2	Appendix A: Glossary . . . . .	10
5.3	Appendix B: Analysis Models . . . . .	10
5.4	Appendix C: To Be Determined List . . . . .	10

## Revision History

Name	Date	Reason For Changes	Version
21	22	23	24
31	32	33	34

# 1 Introduction

## 1.1 SRS Purpose

<Identify the product whose software requirements are specified in this document, including the revision or release number. Describe the scope of the product that is covered by this SRS, particularly if this SRS describes only part of the system or a single subsystem.>

## 1.2 Product Scope

<Provide a short description of the software being specified and its purpose, including relevant benefits, objectives, and goals. Relate the software to corporate goals or business strategies. If a separate vision and scope document is available, refer to it rather than duplicating its contents here.>

## 1.3 Intended Audience

<Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers. Describe what the rest of this SRS contains and how it is organized. Suggest a sequence for reading the document, beginning with the overview sections and proceeding through the sections that are most pertinent to each reader type.>

## 1.4 Definitions, Acronyms, and Abbreviations

< >

## 1.5 Document Conventions

<Describe any standards or typographical conventions that were followed when writing this SRS, such as fonts or highlighting that have special significance. For example, state whether priorities for higher-level requirements are assumed to be inherited by detailed requirements, or whether every requirement statement is to have its own priority.>

## 1.6 References and Acknowledgements

<List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document. Provide enough information so that the reader could access a copy of each reference, including title, author, version number, date, and source or location.>

## 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 usability. 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: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

### 5.3 Appendix B: Analysis Models

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

### 5.4 Appendix C: 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.>