Software Requirements Specification

for

UIC Teamwork Contribution Assessment System

Version 1.0 approved

Prepared by YuePeng LONG, YiTao QIU, HongYue SHEN and RongKai LIU

AIR

Friday, March 6, 2020

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Project Scope 1

1.5 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Features 2

2.3 User Classes and Characteristics 3

2.4 Operating Environment 3

2.5 Design and Implementation Constraints 3

2.6 User Documentation 3

2.7 Assumptions and Dependencies 3

3. System Features 3

3.1 System Feature 1 3

3.2 System Feature 2 (and so on) 4

4. External Interface Requirements 4

4.1 User Interfaces 4

4.2 Hardware Interfaces 4

4.3 Software Interfaces 4

4.4 Communications Interfaces 5

5. Other Nonfunctional Requirements 5

5.1 Performance Requirements 5

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

6. Other Requirements 5

Appendix A: Glossary 5

Appendix B: Analysis Models 6

Appendix C: Issues List 6

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| YuePeng LONG, YiTao QIU, HongYue SHEN, RongKai LIU | 2020-03-06 | The first version | 1.0 |
|  |  |  |  |

# Introduction

## Purpose

This document is for the project named “UIC Teamwork Contribution Assessment System” which is developed by Group AIR. The document will cover the introduction of the document, overall description, external requirements, main features and other non-functional requirements of the product.

This document aims to help developers to conveniently develop the application. And for developers, it specifies the major features this product will provide.

## Document Conventions

This document will use “Times New Roman” as font for both first-level and second-level titles, of which the font sizes are 18 and 14 respectively. And titles will be highlighted by bold.

The main body of this document is written in the font “Arial” with the font size 11. Also, this document will use **bold text** to emphasize the important content when necessary, and Italic text to indicate those company names and product names mentioned.

For the acronyms mentioned in this document, please refer to Appendix A for details.

## Intended Audience and Reading Suggestions

The document is mainly written for clients, developers and testers.

* For developers, please read the whole document
* For clients, we suggest that they can refer to the content in **part 2** and **main features of the product** in **part 3**
* For testers, the content in **part 2** and **non-functional requirements** in **part 5** are recommended

## Project Scope

This product aims to bring UIC staffs and students a better and more convenient platform to do the record, management and assessment about teamwork contribution. With this application, teachers can easily specify the group allocation, manage the statistics uploaded by students and get the reference of teamwork contribution for grading students, while students can do submission of each assignment or group project, and assess their classmates’ work clearly.

## References

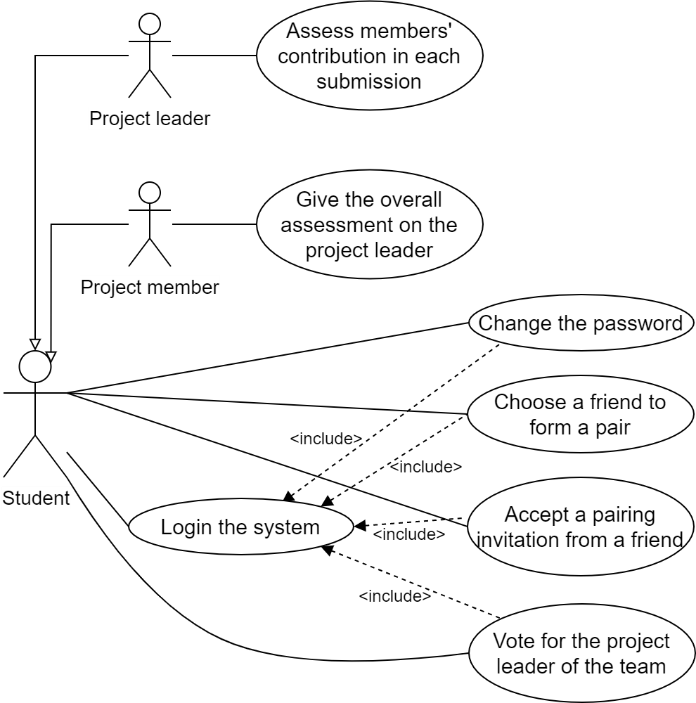
**TBD**

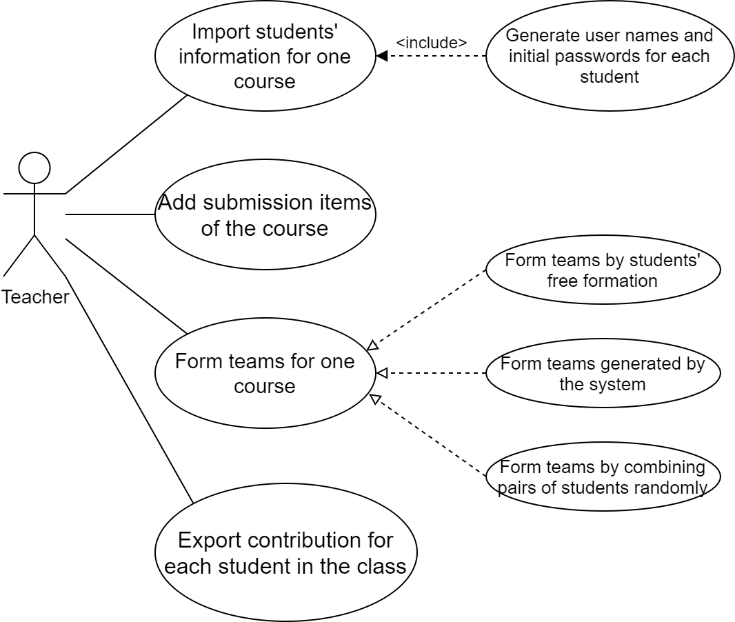
# Overall Description

## Product Perspective

The system should be used as an assistant tool for grade assessment. User accounts can be imported from other systems such as an account management system. Teamwork contribution can be exported and be used in another grading system.

## Product Features

The use case diagrams are shown below.



For the Teacher user case diagram:

Actors: Teacher

* Teacher can import students’ information for a course.
* Teacher can generate students’ basic information (including usernames and passwords).
* Teacher can add the submission items of a course.
* Teacher can form teams for a course.
* Teacher can export a file which lists the contribution.

For the Student user case diagram:

Actors: Student (including Project leader and Project member)

* Student can login the system.
* Student can change the password, send a pairing invitation, accept a pairing invitation and vote for the team leader.
* Project leader can assess members’ contribution.
* Project member can give an assessment to the leader.

## User Classes and Characteristics

There are two kinds of user classes. One class is teachers and the other class is students. Teachers and students are both members in the same university and they need to use this system to finish their work with higher quality and efficiency. Additionally, they all have some basic knowledge about computer and Internet. For example, they know how to use browser in Windows, Mac or Linux desktop to open and use our web application.

## Operating Environment

The application is based on web browser. The application is able to run on Linux-based operating system or Microsoft Windows.

## Design and Implementation Constraints

Language requirements: Web application uses English as the language of the user interface.

Other constrains are TBD.

## User Documentation

An independent user manual is needed to provide for users. This manual should be written in English version. The user manual will also be placed on the application's web page for user to read online.

## Assumptions and Dependencies

Users should have a device that can connect to the Internet and use a browser.

# System Features

<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>

## System Feature 1

<Don’t really say “System Feature 1.” State the feature name in just a few words.>

3.1.1 Description and Priority

<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>

3.1.2 Stimulus/Response Sequences

<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.>

3.1.3 Functional Requirements

<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.>

<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>

REQ-1:

REQ-2:

## System Feature 2 (and so on)

# External Interface Requirements

## User Interfaces

TBD

## Hardware Interfaces

The software does not have the ability of controlling the hardware.

## Software Interfaces

* **Database**

The software needs MySQL-like database (we recommend MariaDB) to store all the information. Typically, the software queries the database to display the information (account information, courses, assignments, assessment, etc.) on the user interface; all the users can operate the user interface (web pages) to perform database-level operations, such as database addition, deletion, queries and modification.

* **Web server**

A web server, Nginx, is needed to proxy all the information between users and the backend of the software. Typically, when users send requests to Nginx, Nginx directs the requests to the correct interface of the backend; when the backend sends back responses according to the requests, Nginx directs the responses and sends them back to the corresponding users.

## Communications Interfaces

The software interacts with users on web browsers using HTTPS protocol. Modern web browsers (we recommend Google Chrome) are required.

# Other Nonfunctional Requirements

## Performance Requirements

The response time of each inquiry operation should be less than one second and other operation’s response time should be less than two seconds.

## Safety Requirements

This system does not concern any safety issues.

## Security Requirements

The system will the keep the privacy of the user safe. The teacher can only view and manage the data of students from his/her own class. The password protection is not provided in the system, so the user should carefully keep his/her password. Other security requirements are TBD.

## Software Quality Attributes

Maintainability: Some documents should be written to maintain the system in the future. There should be comments to help other developer to understand the system.

Usability: The GUI should be easy to learn and use by the user of any technical background. A concise documentation should be provided to the administrator.

Portability: the website will ensure to support latest versions of the web browsers.

Security: The information should be available to user according to their identities.

Robustness: The system has a Mean time to Failures (MTTF) for 3000 hours per failure.

Reusability: some functions of the system could be reused on other websites, such as being part of iSpace.

# Other Requirements

TBD

Appendix A: Glossary

* TBD: To be determined
* T.C.A.S.: Teamwork Contribution Assessment System
* MTTF: Mean time to Failures

Appendix B: Analysis Models

class diagram

Appendix C: Issues List

TBD