CSCI222/MCS9222 System Development Spring Session 2014

System Requirement Report

Group 2

Presented by

Junyan **Fan** 3767929

Ruixi **He** 4174458

Zhengyu **Yuan** 4327664

Yuanxin **Xu** 4332969

Yi **Jin** 4370673

Table of Contents

[1. Introduction 4](#_Toc398219362)

[1.1 Purpose 4](#_Toc398219363)

[1.2 Scope 4](#_Toc398219364)

[1.3 Overview 5](#_Toc398219365)

[2. Overall Description 6](#_Toc398219366)

[2.1 Product perspective 6](#_Toc398219367)

[2.1.1 System Interfaces 6](#_Toc398219368)

[2.1.2 User Interfaces 7](#_Toc398219369)

[2.1.3 Hardware Interfaces 7](#_Toc398219370)

[2.1.4 Software Interfaces 7](#_Toc398219371)

[2.1.4.1 External System Interface: 7](#_Toc398219372)

[2.1.4.2 User Interface: 7](#_Toc398219373)

[2.1.5 Communication Interfaces 7](#_Toc398219374)

[2.1.6 Memory Constraints 7](#_Toc398219375)

[2.1.7 Operations 7](#_Toc398219376)

[2.2 Product Functions 8](#_Toc398219377)

[2.3 User characteristics 9](#_Toc398219378)

[2.4 Constraints 9](#_Toc398219379)

[2.5 Assumptions and dependencies 10](#_Toc398219380)

[3. Risks and counter measures 10](#_Toc398219381)

[3.1 Concurrency/Duplication 10](#_Toc398219382)

[3.2 Loss or deletion 10](#_Toc398219383)

[3.3 Errors report 11](#_Toc398219384)

[3.4 Super user Logout 11](#_Toc398219385)

[3.5 Security 11](#_Toc398219386)

[3.6 Operational errors 11](#_Toc398219387)

[3.7 Storage 12](#_Toc398219388)

[4. Software Requirements Specification 12](#_Toc398219389)

[5. Use cases 22](#_Toc398219390)

[6. Domain model with detailed description 32](#_Toc398219391)

[6.1Domain Model 32](#_Toc398219392)

[6.2 Class Description 32](#_Toc398219393)

[7. Meta report 37](#_Toc398219394)

[7.1 Group structure 37](#_Toc398219395)

[7.2 Summary of each group meeting 38](#_Toc398219396)

[7.3 Agenda and report of meeting 39](#_Toc398219397)

[7.4 Work Diaries 41](#_Toc398219398)

[8. Version Control (GitHub) 45](#_Toc398219399)

1. Introduction

# 1.1 Purpose

The purpose of this document is to describe the specifications on the conference management system. It also documents nonfunctional requirements, design constraints and other factors necessary to provide a complete and comprehensive understanding of the system. The intended audience of this document includes the prospective software development team and the potential users of the system.

# 1.2 Scope

The software system to be produced is helping conference organizers to cope with the complexity of the refereeing process, which will be referred to as “Conference Management System” thorough this document.

Conference Management System aims to serve four groups of users – the sponsors, the PC members, the Steering Committee and authors. The purposes of Steering Committee (SC) are the people searches for a location for a conference and it also decides a time frame of the conference. PC members usually consist of the fellow researchers that are well known experts in the research topics of the conference. To form a PC, the members of SC prepare the documents describing a mission and the motivations behind the conference, goals and topics of interest. These documents are sent to the selected researchers together with an invitation to participate in the conference as PC members. Authors submit the research papers. In order to satisfy all potential customers and to effectively manage its functions, Conference Management System is developed into four modules, each of which is an independent but correlated subsystem.

* Users Management Subsystem aims to manage profiles of all users the system.
* Resource Management Subsystem aims to manage profiles of all users of the system.
* Communication Management Subsystem aims to manage the communication among all stakeholders. Documents Managements Subsystem aims to manage all the electronic documents such like text files, Word documents, Excel spread sheets, image, video, audio files a lot of information about PC members, industrial sponsors, contents of conference Web sites, conference participants, their contributions, etc.
* Paper Submit Subsystem aims to submit the paper on the website.

# 1.3 Overview

The rest of this document is divided into two main sections:

* The Overall Description (section 2) describes the general factors that affect the system and its requirements.
* The Specific Requirements (section 3) contains all software requirements that the system must meet in order to satisfy customer’s needs.

2. Overall Description

# 2.1 Product perspective

In the past years, the researchers and the sponsors spent a lot of time and funds in the setting up of scientific conferences. In the last few years experts and sponsors organize at least two conferences per year, eevery time the conference was held complicated and the past experiences indicate that appropriate application of information technologies would simplify this process.

Based on the truth, it is necessary to build a software to help all the stakeholders get benefits from previous huge amounts of works and unnecessary time or cost spending. Conference management system will assist this and achieve the goals.

## 2.1.1 System Interfaces

The Conference Management System to be developed is an application that can be integrated to an organization’s Intranet or deployed on the Internet.

The Clientscan simultaneously log into the system from any PC that supports Internet services and then interact with the system for their purposes, such as list of the latest events, submission of reviews and discussion of reviewing of papers.

## 2.1.2 User Interfaces

The user interfaces provided to client machines must be GUI and must be accessible through any web browser such as IE, Opera etc. conference management system administrators also connect to the system via the client like other users but have higher access right to the system, and they have the rights to change the internal system.

## 2.1.3 Hardware Interfaces

All components must be able to execute on a personal computer.

## 2.1.4 Software Interfaces

### 2.1.4.1 External System Interface:

* The system is able to import existing user accounts from other system.
* The system is able to import existing resource profiles from other system.

### 2.1.4.2 User Interface:

* The user interaction with the system through the client application and the web browser.
* The system supports Windows XP or above.

## 2.1.5 Communication Interfaces

* The client machines must communicate with the Web Server over TCP/IP connection.
* The Web Server and the Database Server are located on different servers.

## 2.1.6 Memory Constraints

* The client machine must be able to operate within 64MB minimum (including memory for browser).
* The Web Server and the Database Server must be able to operate within 128MB minimum.

## 2.1.7 Operations

Conference management system must be easy for all users to use e.g. no specific information or skills (except knowledge on how to access the Internet via Web browser) must be required to use the tool.

The Web Server installation and maintenance should be simple enough for a network administrator to perform and should not require any special technical skills from the administrator.

The software should able to contain backup and recovery operations must be specified in case of network failure, out of power etc.

# 2.2 Product Functions

* The two main functions of conference management system are: all the stakeholders generate the contents through the conference management system. For example, the people who have the relationship with the conference need to communicate each other with the system.
* All the contents (papers etc.) are able to submit by the system.

For SC, conference management system helps them to manage stores resources through the following main functions:

* Use system to get the PC group
* set up the conference time and location
* Collect the paper and submit the papers to a publisher responsible for editing of the conference proceedings.
* To find industry sponsor.

For PC members, conference management system helps them to use the application through the following main functions:

* Update the status of the papers
* Evaluated the paper submitted by researchers and submit the review paper.
* Send the call for paper the selected researchers.
* Use work group subsystem to collection.

For authors, conference management system is controlled by administrators through the following main functions:

* Submit paper to system.
* Pay registration fee
* Get the review about own paper and send own view
* Can check the status of the papers.

For administrator, conference management system is controlled by administrators through the following main functions:

* Post the latest events on the website
* Update the database to store all files and documents
* Use database collect paper and allocate to different kinds of PC members
* Give different permission to different group users

For PC chairs, conference management system is controlled by administrators through the following main functions:

* Collect the paper and the evaluation.
* Reallocate the paper to another program committee.

All functionalities of the system are built based on the conference needs can make all activities and arrangements carried out by the all users easier and more convenient.

# 2.3 User characteristics

The users of conference management system include the sponsors, the PC members, the Steering Committee and authors.

* Administrators have strong knowledge on networks and web applications to be able to install and maintain Online CD Services system.
* Sellers are knowledgeable of the CD stores activities and have solid understanding of all tasks, processes and resources they are working with.
* Customers are people who have enough understanding on the use of Internet to use the system.

2.4 Constraints   
The system should strictly obey and satisfy the following constraints:

* Authentication security: the system should enforce user authentication security.
* Access control: the system must provide appropriate access right and user interface to each type of user.
* Backup and recovery: the backup and recovery of all the system’s database must be easy to perform to prevent databases from corruption and loss risks.
* Integrity control: since the system consists of many databases that are correlated with each other, integrity among these databases must be strictly maintained.

# 2.5 Assumptions and dependencies

The following assumptions and dependencies for the system are stated:

* All participants must book their accommodation in one of the available hotels via the website which the system has the functions to allow users to search them.
* Because the conferences have to be handled with a lot of electronic documents, all participants should have a standard email address.

3. Risks and counter measures

# 3.1 Concurrency/Duplication

Risk:

There might be a case of duplication of application entry where there are more than one same paper belongs to one applicants and submitted to different field. This may increase unnecessary redundancy of the system and program committee may not realize it.

Countermeasure:

System should have specific ID for each applications form even if the applicants have the same name, since a case where two persons have the same name and surname might happen.

Acceptable Level of Risk after Countermeasure: ACC

# 3.2 Loss or deletion

Risk:

There also might be a case where the applications details may be lost or accidently delete somewhere in the process.

Countermeasures:

A method to notify their application is process through a certain stage (so they are certain that their details are updated to a certain stage).

Acceptable Level of Risk after Countermeasure: ACC

# 3.3 Errors report

Risk:

A functional failure (Technical reasons cause unknown error).

Countermeasures:

Replacement technology and consult a professional and sending the error message to the developer directly.

Acceptable Level of Risk after Countermeasure: ACC

# 3.4 Super user Logout

Risk:

Staff/Admin may forget logout which cause other person can control background system.

Countermeasures:

The system should indicated that person are no longer has the using action should be logout with a set time or logout automatically.

Acceptable Level of Risk after Countermeasure: ACC

# 3.5 Security

Risk:

Hacker attacks the system.

Countermeasures:

The system should use famous fire-wall open-source software with to avoid the intrusion of background and database (especially relate to the financial issues).

Acceptable Level of Risk after Countermeasure: ACC

# 3.6 Operational errors

Risk:

While using unfamiliar interface may cause operation faults.

Countermeasures:

Some crucial operations will pop-up warning message and give suitable suggestion. If operation fault is caused by continuous multiple pressing operations, the system should handle it and rollback gracefully.

Acceptable Level of Risk after Countermeasure: ACC

# 3.7 Storage

Risk：

Long time running system will cause not enough space to store papers.

Countermeasures:

Some of old papers can be archived or destroyed decided by program committee or set appropriate storage such as a growing file system such as ZFS and add new disks if it is necessary.

Acceptable Level of Risk after Countermeasure: ACC

4. Software Requirements Specification

**Specific Requirements**

Each requirement (either functional or non-functional one) of conference management system is ranked based on its level of importance.

● Critical: highest importance level. Critical requirements are those that reflect core functionalities of conference management system and must be firstly implemented

● Essential: second highest importance level. Essential requirements are those that reflect important functionalities of conference management system and must be covered when Critical requirements have successfully implemented

● Optional: lowest importance level. Optional requirements are those that reflect enhanced functionalities of conference management system and should be considered only when all Critical and Essential requirements are completed

|  |  |  |
| --- | --- | --- |
| Requirement#:r1 | Requirement type: Functional | Use case #:u17 |
| ­­­Description: the system should manage the access of PC members and referees to papers and conflicts of interests. | | |
| Rationale: the system should allowed multiple members access the database at the same time | | |
| Source: PC member and referees | | |
| Fit Criterion: the PC member and referee can access the database without conflicts | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: none | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r2 | Requirement type: Functional | Use case #:u2 |
| ­­­Description: the system should let the SC member send the CFP to the researchers who want to submit a paper and post CFP on the website | | |
| Rationale: the SC need to attract more researchers submit papers | | |
| Source: SC member | | |
| Fit Criterion: the researcher can get the CFP paper. | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: CFP | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r3 | Requirement type: Functional | Use case #:u17 |
| ­­­Description: the system should provide the reminder email and automatically extend one week for the PC member who did not submit their evaluation paper on time, and an ability to reallocate the paper to another PC member | | |
| Rationale: the PC member need to know they did not submit evaluation on time, if they do not want to evaluation it anymore, the paper should be gave to another PC member | | |
| Source: SC | | |
| Fit Criterion: the PC can get the reminder and extra one week, and the paper will be evaluated. | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: paper | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r4 | Requirement type: Functional | Use case #: u15 |
| ­­­Description: the system should let PC member change the status of the paper, if the paper is accepted, the system should send the review paper to author | | |
| Rationale: the author should know the their paper’s status and get the review paper | | |
| Source: the author | | |
| Fit Criterion: the author knows their paper’s status and if their paper is accepted, the author can receive the review paper. | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: review paper | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r5 | Requirement type: Functional | Use case #:u7 |
| ­­­Description: the system should provide a submission function for letting accepted paper’s author submit two paper, one has header, another not | | |
| Rationale: this is for “blind process”. | | |
| Source: PC member | | |
| Fit Criterion: system received two kinds of paper | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: the two kinds of final version of accepted paper | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r6 | Requirement type: Functional | Use case #:u7 |
| ­­­Description: the system should send a reminder to the accepted paper’s author for reminding them that they have 2-3 weeks for preparing final version | | |
| Rationale: give time for accepted paper’s author to prepare final version | | |
| Source: the accepted paper’s author | | |
| Fit Criterion: the accepted paper’s author received reminder email | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: none | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r7 | Requirement type: Functional | Use case #:u7 |
| ­­­Description: the system should send a reminder to the accepted paper’s author let them know the final version should be a PDF file | | |
| Rationale: the paper format should be PDF file | | |
| Source: the PC member | | |
| Fit Criterion: the PC member received the PDF format paper | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: none | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r8 | Requirement type: Functional | Use case #:u13 |
| ­­­Description: the system should provide a payment information for industry sponsor and participant | | |
| Rationale: the organization need money for operation | | |
| Source: the organization | | |
| Fit Criterion: the organization received money | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: none | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |
| Requirement#: r9 | Requirement type: Functional | Use case #:u9 |
| ­­­Description: the system should make the registration fees of the accepted paper’s author lower than others. | | |
| Rationale: the author’s paper has been accepted | | |
| Source: the accepted paper’s author | | |
| Fit Criterion: the accepted paper’s author pay lower than others for registration fees | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: none | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#: r10 | Requirement type: Functional | Use case #:u7 |
| ­­­Description: the system should let at least one of the accepted author register via website | | |
| Rationale: the author need to register for letting their paper being included in the conference proceeding | | |
| Source: SC | | |
| Fit Criterion: the system has the record of the registration of the accepted paper’s author | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: none | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#: r11 | Requirement type: Functional | Use case #:u7 |
| ­­­Description: the system should exclude the paper from conference proceeding if the author did not register | | |
| Rationale: the author need to register for letting their paper being included in the conference proceeding | | |
| Source: SC | | |
| Fit Criterion: if the accepted paper’s author did not register, their paper will not be founded from the conference proceeding | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: none | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |
| Requirement#: r12 | Requirement type: Functional | Use case #:u9 |
| ­­­Description: the system should send the copy of the conference proceeding to the people who have already registered. | | |
| Rationale: the registration fee contain the cost of conference proceeding | | |
| Source: people who registered | | |
| Fit Criterion: people who register received the copy of the conference proceeding | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: the copy of the conference proceeding | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#: r13 | Requirement type: non-Functional | Use case #:none |
| ­­­Description: the system should provide a link to an accommodation website, let the participant book the accommodation | | |
| Rationale: the participant need to find the living place during the conference time | | |
| Source: participant | | |
| Fit Criterion: participant found their living place | | |
| Dependencies: none | | |
| Rank of importance: optional | | |
| Supporting Materials: | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#: r14 | Requirement type: Functional | Use case #:u18 |
| ­­­Description: the system should provide a payment address for letting the participant pay the registration fee and conference banquet fee | | |
| Rationale: the organization need money for the operation | | |
| Source: SC | | |
| Fit Criterion: organization received the money | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: none | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#: r15 | Requirement type: Functional | Use case #:u3 |
| ­­­Description: the system should send the schedule for the presentation to the participant and published on their website. | | |
| Rationale: the participant needs to know the schedule for the presentation | | |
| Source: participant | | |
| Fit Criterion: participant know the schedule for the presentation | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: schedule for the presentation | | |
| History: Create by Zhengyu Yuan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r16 | Requirement type: Functional | Use case #:u20 |
| ­­­Description: The system should have a database to manage all the text files, Word documents, Excel spreadsheets, image, video, audio files a lot of information about PC members, industrial sponsors, contents of conference  Web sites, conference participants, their contributions | | |
| Rationale: all file need be stored, there are lots of files in it. If we use the database, we can find these easily. Get a copy for all file. | | |
| Source: administration | | |
| Fit Criterion: didn’t lost file and easy to find file which u want. | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: all file | | |
| History: Create by Junyan Fan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r17 | Requirement type: Functional | Use case #:none |
| ­­­Description: The system should provide communicate with PC members, sponsors, participants, and to receive submissions, and to distribute the outcomes of the evaluations. Use email to inform participants. | | |
| Rationale: different user need have communication in the system, if we have some outcome, we need tell user using email. | | |
| Source: administration | | |
| Fit Criterion: let them communication easily on system , can email to user if have result | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: outcomes of the evaluation | | |
| History: Create by Junyan Fan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r18 | Requirement type: Functional | Use case #:u22 |
| ­­­Description: system need provide different level of access to system for four group of user | | |
| Rationale: different user have different limits of authority | | |
| Source: administration | | |
| Fit Criterion: let different user can access fit area of the system | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: none | | |
| History: Create by Junyan Fan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r19 | Requirement type: non-Functional | Use case #:u1 |
| ­­­Description: system need use history to get the PC member | | |
| Rationale: system can search the history and judgment best PC and list them | | |
| Source: SC | | |
| Fit Criterion: get the best PC member from the history | | |
| Dependencies: none | | |
| Rank of importance: optional | | |
| Supporting Materials: history information of PC | | |
| History: Create by Junyan Fan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r20 | Requirement type: Functional | Use case #:u21 |
| ­­­Description: system can get all paper and store them, distribute all paper to reviewer | | |
| Rationale: all paper need to store, and send to reviewer. | | |
| Source: administration | | |
| Fit Criterion: every paper can be stored and every paper can be distributed to reviewer | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: paper | | |
| History: Create by Junyan Fan on 08/09/2014 | | |
| Requirement#:r21 | Requirement type: Functional | Use case #:u14 |
| ­­­Description: system need let reviewer use work group sub system to collect review | | |
| Rationale: use work group sub system is good for review’s collect | | |
| Source: administration | | |
| Fit Criterion: all reviewers can use work group sub system to collect review. | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: review | | |
| History: Create by Junyan Fan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r22 | Requirement type: Functional | Use case #:u15 |
| ­­­Description: system will provide paper’s informed automatically (acceptance or rejection) to author. | | |
| Rationale: the author need know their paper whether be accepted | | |
| Source: author | | |
| Fit Criterion: the author can get their paper’s inform | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: none | | |
| History: Create by Junyan Fan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r23 | Requirement type: non-Functional | Use case #:u6 |
| ­­­Description: system provide the industry sponsors to SC with the history information | | |
| Rationale: system have the history information about industry sponsors, if we need industry sponsors, we can judge the history and get best industry sponsors to SC | | |
| Source: SC | | |
| Fit Criterion: SC get the best industry sponsors | | |
| Dependencies: none | | |
| Rank of importance: optional | | |
| Supporting Materials: history information about industry sponsors | | |
| History: Create by Junyan Fan on 08/09/2014 | | |

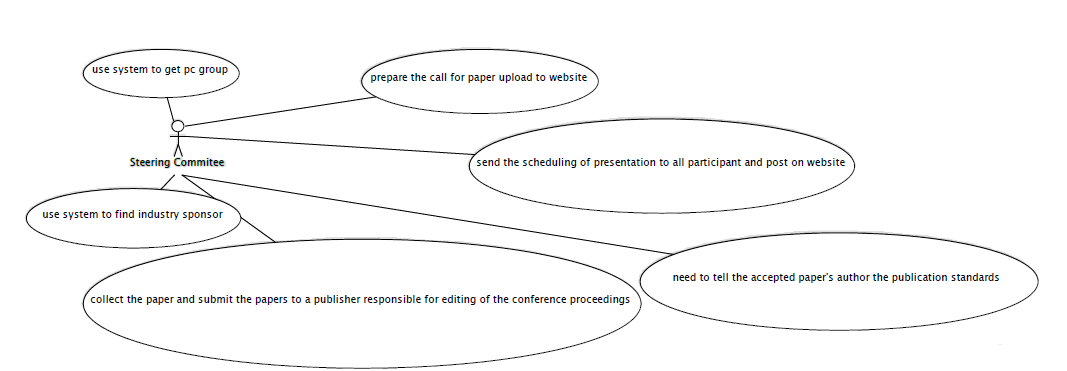
|  |  |  |
| --- | --- | --- |
| Requirement#:r24 | Requirement type: Functional | Use case #:u19 |
| ­­­Description: system need provide list of the latest events. | | |
| Rationale: the system need let every participants know the latest event and put it on the system interface | | |
| Source: all user | | |
| Fit Criterion: every user can know the latest event | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: none | | |
| History: Create by Junyan Fan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r25 | Requirement type: Functional | Use case #:u5 |
| ­­­Description: system can provide automatic preparation of conference proceedings, collect all paper. | | |
| Rationale: system need collect all paper about conference proceeding, we will need them and send to every users | | |
| Source: conference organize | | |
| Fit Criterion: let all paper about conference proceeding together, don’t lost | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: paper about conference proceeding | | |
| History: Create by Junyan Fan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r26 | Requirement type: optional | Use case #:u10 |
| ­­­Description: system will let the author response (aka rebuttal) phase, when the author can respond to the reviews | | |
| Rationale: system will let author know the response, and let them have chance to say their view. | | |
| Source: author | | |
| Fit Criterion: author can get response in system and successful say own view | | |
| Dependencies: none | | |
| Rank of importance: Essential | | |
| Supporting Materials: paper | | |
| History: Create by Junyan Fan on 08/09/2014 | | |

|  |  |  |
| --- | --- | --- |
| Requirement#:r27 | Requirement type: non-functional | Use case #:none |
| ­­­Description: system will monitoring email | | |
| Rationale: system can know every email sent and got, and collect all information about the email | | |
| Source: administration | | |
| Fit Criterion :get all information about email and store their time and object | | |
| Dependencies: none | | |
| Rank of importance: optional | | |
| Supporting Materials: email | | |
| History: Create by Junyan Fan on 08/09/2014 | | |

5. Use cases



|  |  |
| --- | --- |
| Name: get PC member | ID: u1 |
| Stakeholder and goals: SC – want to form the PC group | |
| Description: the system can help SC form the PC group | |
| Actors: SC | |
| Trigger: the SC ask for the recommend PC group members from system | |
| Normal flow: 1. SC log in the system  2. SC press generate the PC group  3. End | |
| Sub-flow: none | |
| Alternative/Exceptional flows: none | |

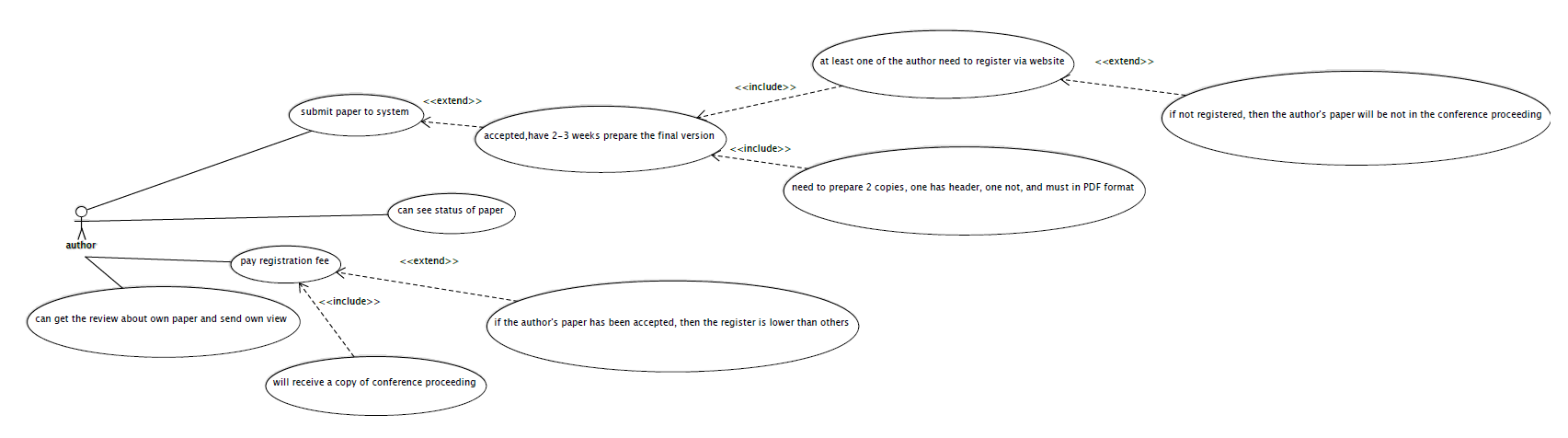
|  |  |
| --- | --- |
| Name: upload the CFP | ID: u2 |
| Stakeholder and goals: SC – upload the CFP | |
| Description: the SC upload the CFP on website | |
| Actors: SC | |
| Trigger: the SC upload the CFP to the website | |
| Normal flow: 1. SC log in the system  2. SC upload the CFP  3. End | |
| Sub-flow: none | |
| Alternative/Exceptional flows: none | |

|  |  |
| --- | --- |
| Name: send the scheduling of presentation | ID: u3 |
| Stakeholder and goals: SC and all participant– let all participant have scheduling of presentation and post to website | |
| Description: the SC need to send the scheduling to all participant via system and post it on website | |
| Actors: SC | |
| Trigger: the SC send the scheduling to all participant and post on website | |
| Normal flow: 1. SC log in the system  2. SC send the scheduling to all participant  3. SC post the scheduling to website  4. End | |
| Sub-flow: none | |
| Alternative/Exceptional flows: none | |

|  |  |
| --- | --- |
| Name: tell the publication standards | ID: u4 |
| Stakeholder and goals: SC and author –let the accepted paper's author know publication standards | |
| Description: the SC tell the accepted paper's author publication standards via system | |
| Actors: SC | |
| Trigger: the SC send the information of publication standards to the accepted paper’s author | |
| Normal flow: 1. SC log in the system  2. SC send the publication standards to author  3. End | |
| Sub-flow: none | |
| Alternative/Exceptional flows: none | |

|  |  |
| --- | --- |
| Name: edit conference proceeding | ID: u5 |
| Stakeholder and goals: SC, author, and all participant – create the conference proceeding | |
| Description: the SC send all the collected paper to publisher for create the conference | |
| Actors: SC | |
| Trigger: the SC send the all the collected papers to publisher | |
| Normal flow: 1. SC log in the system  2. SC use system collect all accepted papers  3. SC send all accepted papers to publisher via system  4. End | |
| Sub-flow: none | |
| Alternative/Exceptional flows: none | |

|  |  |
| --- | --- |
| Name: find sponsor | ID: u6 |
| Stakeholder and goals: SC, participant, industry sponsor– find the industry sponsors | |
| Description: the SC need to find the industry sponsor by using the system | |
| Actors: SC | |
| Trigger: the find the sponsor by using the information provided by system | |
| Normal flow: 1. SC log in the system  2. SC let the system recommend the sponsors  3. SC find the sponsor  4. End | |
| Sub-flow: none | |
| Alternative/Exceptional flows: none | |

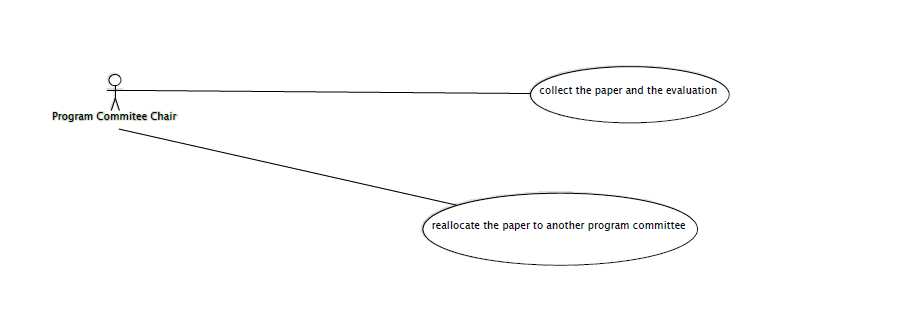


|  |  |
| --- | --- |
| Name: submit paper | ID: u7 |
| Stakeholder and goals: author, all participant- submit the paper to system | |
| Description: author submit the paper via system | |
| Actors: author | |
| Trigger: author submit the system via system | |
| Normal flow: 1. Author log in the system  2. Author upload the paper to system and submit  3. End | |
| Sub-flow: 2.a. if the paper is accepted, the author needs to prepare final version in 2-3 weeks.  b. At least one of the author need to register via website  c. Need to prepare 2 copies, one has header, one not, and must in PDF format | |
| Alternative/Exceptional flows: b.1. If not registered, then the author's paper will be not in the conference proceeding | |

|  |  |
| --- | --- |
| Name: see status | ID: u8 |
| Stakeholder and goals: author-check the status of their submitted papers | |
| Description: the author can see the status of their paper on system | |
| Actors: author | |
| Trigger: the author found the status of their paper | |
| Normal flow: 1. Author logs in the system  2. Author find the status of their paper  3. End | |
| Sub-flow: none | |
| Alternative/Exceptional flows: none | |

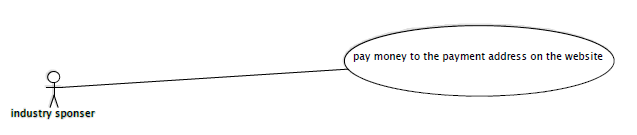
|  |  |
| --- | --- |
| Name: pay registration fee | ID: u9 |
| Stakeholder and goals: author, all participant- make the conference runs well | |
| Description: author needs to pay the registration fee to the payment address on the system | |
| Actors: author | |
| Trigger: author pay the registration fee | |
| Normal flow: 1. Author log in the system  2. Author pay the registration fee  3. Author will receive the copy of the conference proceeding  4. End | |
| Sub-flow: none | |
| Alternative/Exceptional flows: b.1. if the author's paper has been accepted, then the register is lower than others | |

|  |  |
| --- | --- |
| Name: send own view | ID: u10 |
| Stakeholder and goals: author-argue their result | |
| Description: the author can send their opinion about the review of their papers | |
| Actors: author | |
| Trigger: the author send their opinions via system | |
| Normal flow: 1. Author logs in the system  2. Author send their opinions  3. End | |
| Sub-flow: none | |
| Alternative/Exceptional flows: none | |

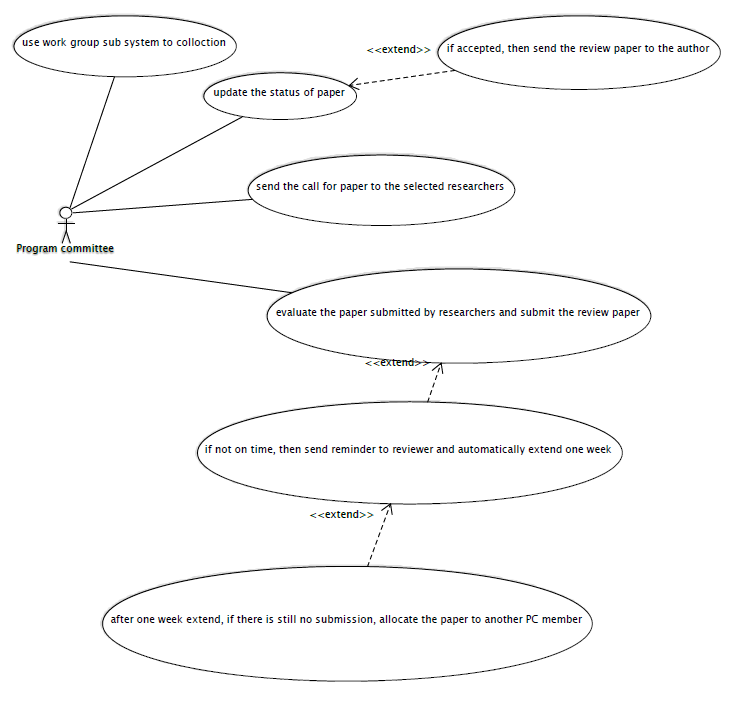


|  |  |
| --- | --- |
| Name: collection the result | ID: u11 |
| Stakeholder and goals: PC chair, author--- need collect the paper and evaluation | |
| Description: the PC chair need use the system to collect all paper and evaluation | |
| Actors: Program Committee Chair | |
| Trigger: when author upload the paper and get evaluation, pc chair will need this | |
| Normal flow: 1. PC chair log in the system  2. Go in the system’s database and search all paper and get their evaluation.  3. End | |
| Sub-flow: none | |
| Alternative/Exceptional flows : none | |

|  |  |
| --- | --- |
| Name: reallocate the paper | ID: u12 |
| Stakeholder and goals: PC chair, PC—reallocate the paper to PC | |
| Description: the PC chair need use the system get the paper and reallocate the paper to every PC member | |
| Actors: Program Committee Chair | |
| Trigger: when PC chair find pc cannot do work , it will reallocate | |
| Normal flow: 1. PC chair log in the system  2.PC go in the database and if he find the PC didn’t work  3. he will reallocate paper to PC  4.end | |
| Sub-flow: none | |
| Alternative/Exceptional flows : none | |



|  |  |
| --- | --- |
| Name: pay money | ID: u13 |
| Stakeholder and goals: industry sponsor—need pay money on website | |
| Description: the industry sponsor need pay money to the conference, they can pay it on website | |
| Actors: industry sponsor | |
| Trigger: when conference need to be prepared, it need money, money comes from industry sponsor | |
| Normal flow: 1. industry sponsor log in the system  2 industry sponsor find the payment website  3. pay it  4.end | |
| Sub-flow: none | |
| Alternative/Exceptional flows : none | |

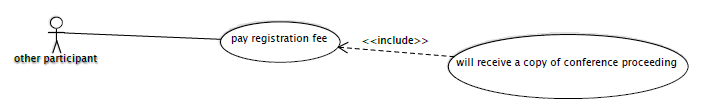


|  |  |
| --- | --- |
| Name: collect paper | ID: u14 |
| Stakeholder and goals: PC—need collect the paper | |
| Description: PC should use the work group sub system to collect the paper , it will be easy | |
| Actors: PC | |
| Trigger: when PC start to work about the paper | |
| Normal flow: 1. PC log in the system  2. find the paper  3. use the work group sub system to collect it  4.end | |
| Sub-flow: none | |
| Alternative/Exceptional flows : none | |

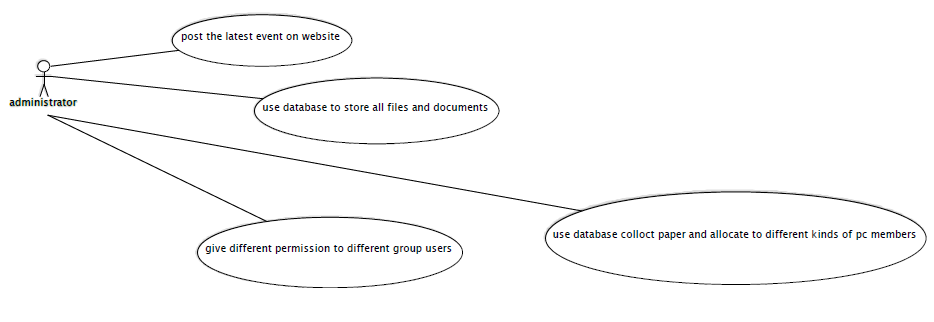
|  |  |
| --- | --- |
| Name: update the status | ID: u15 |
| Stakeholder and goals: PC—update the status of paper | |
| Description: PC should update the status of paper(accept or reject), it can let author know their whether be accept | |
| Actors: PC | |
| Trigger: when PC review all paper , after that | |
| Normal flow: 1. PC log in the system  2. find the paper  3.he will change the paper ‘s status base on the comment about the review  4.end | |
| Sub-flow: none | |
| Alternative/Exceptional flows : if accepted, then send the review paper to the author | |

|  |  |
| --- | --- |
| Name: send the call for paper | ID: u16 |
| Stakeholder and goals: PC—send the call for paper to the selected researchers, | |
| Description: PC will give call for paper to the selected researchers, let them know information about the conference. | |
| Actors: PC | |
| Trigger: when PC start to prepare the conference | |
| Normal flow: 1. PC log in the system  2. get the call for paper and use email to send to researchers  3.end | |
| Sub-flow: none | |
| Alternative/Exceptional flows : if accepted, then send the review paper to the author | |

|  |  |
| --- | --- |
| Name: evaluate and submit the review | ID: u17 |
| Stakeholder and goals: PC—evaluate the paper and submit the review | |
| Description: PC will get a paper and evaluate the paper, when finish evaluate, they need submit the review to the PC chair | |
| Actors: PC | |
| Trigger: when PC chair give PC paper, and PC get the paper | |
| Normal flow: 1. PC log in the system  2. get paper from the PC chair  3.start work  4. end | |
| Sub-flow: none | |
| Alternative/Exceptional flows: 3.a. if not on time, then send reminder to reviewer and automatically extend one week  3.b. after one week extend, if there is still no submission, allocate the paper to another PC member | |



|  |  |
| --- | --- |
| Name: pay registration fee | ID: u18 |
| Stakeholder and goals: other participant—need pay registration fee | |
| Description: other participant need pay registration fee for conference | |
| Actors: other participant | |
| Trigger: when other participant want to know this conference | |
| Normal flow: 1. other participant log in the system  2. go in the payment web page  3.pay it  4. end | |
| Sub-flow: will receive a copy of conference proceeding | |
| Alternative/Exceptional flows: none | |



|  |  |
| --- | --- |
| Name: post the event | ID: u19 |
| Stakeholder and goals: administrator—need post the latest event on website | |
| Description: the administrator need post the latest event on website, it can let other participant know the event | |
| Actors: administrator | |
| Trigger: when prepare will be end, the conference will start | |
| Normal flow: 1. Administrator log in the system  2.find latest event  3.post it on website  4. end | |
| Sub-flow: none | |
| Alternative/Exceptional flows: none | |

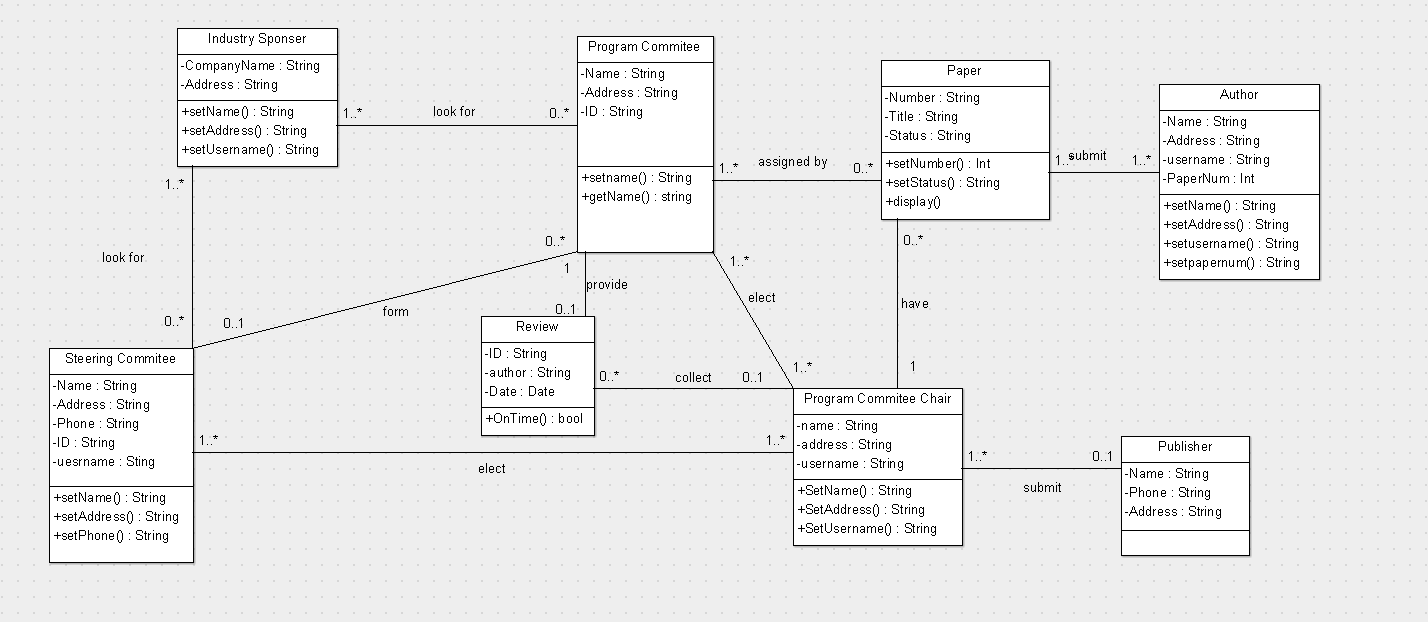
|  |  |
| --- | --- |
| Name: store the file | ID: u20 |
| Stakeholder and goals: administrator—use the database store all files | |
| Description: the administrator need use the database store the file that author upload | |
| Actors: administrator | |
| Trigger: when author upload the paper already | |
| Normal flow: 1. Administrator log in the system  2.use database get the file and manage it  3. end | |
| Sub-flow: none | |
| Alternative/Exceptional flows: none | |

|  |  |
| --- | --- |
| Name: collect paper and allocate to pc | ID: u21 |
| Stakeholder and goals: administrator—use the database collect paper and allocate to PC | |
| Description: the administrator need use database collect paper and allocate to different kinds of PC, make sure every paper have PC review | |
| Actors: administrator | |
| Trigger: when author upload the paper already | |
| Normal flow: 1. Administrator log in the system  2.use database find all paper and allocate to PC  3. end | |
| Sub-flow: none | |
| Alternative/Exceptional flows: none | |

|  |  |
| --- | --- |
| Name: set permission | ID: u22 |
| Stakeholder and goals: administrator—let different user have different permission | |
| Description: the administrator set the permission for different users | |
| Actors: administrator | |
| Trigger: when author upload the paper already | |
| Normal flow: 1. Administrator log in the system  2. select different user and set their permission  3. end | |
| Sub-flow: none | |
| Alternative/Exceptional flows: none | |

6. Domain model with detailed description

# 6.1Domain Model

Class Diagram

# 6.2 Class Description

Class name: Steering Committee

Superclasses:

Attributes:

* Name: Alphanumeric.
* Address: Address of the steering committee.
* Phone: Digit number of the phone number of steering committee.
* ID: Digit number of the id
* Username: The login name of the steering committee.

Methods:

setName()

Set the name of steering committee.

setAddress()

Set the home address of steering committee.

setPhone()

Set the phone number of steering committee.

Class name: Industry Sponsor

Superclasses:

Attributes:

* CompanyName: the company of the industry sponsor.
* Address: the address of the industry sponsor.

Methods:

setName()

Set the name of industry sponsor.

setAddress()

Set the address of industry sponsor.

setUsername()

set the login username of industry sponsor.

Class name: program committee

Attributes:

-Name: Alphanumeric.

-Address: Alphanumeric.

-ID: Alphanumeric.

Method:

Set name(name)

This method allows us to set the PC member’s name.

Get name()

This method returns the name for PC member.

Class name: review

Attributes:

-ID: Alphanumeric.

-Author: Alphanumeric.

-Date: date format.

Method:

+onTime(): bool.

Set whether the PC member submit their review on time.

Class name: Paper

Superclass: N/A

Attributes:

-Number. The unique ID of the paper

-Title. The name of the paper

-Status. The level of the paper such as the “Update” version and “Modified”.

Method

+setNumber() : Int

The method can set the number.

+setStatus() : String

The method can set the status of the paper.

+display()

Display the information of the paper.

Class name: Program Committee Chair

Superclass: N/A

Attributes:

-Name. The name of the member

-Address. The name of the address

-username. The name of the system login.

Method

+setName() : Int

The method can set the name.

+setAddress() : String

The method can set the address.

+SetUsername()

Set the user name for login the system.

Class name: Author

Attributes:

Name. Alphanumeric

Address: Alphanumeric

Usename: letter and number

paperNum: number

Methods

setName()

this method can let use set our name

setAddress()

this method can allows us to set address

setusename()

this method can allows us to set user name

setpapernum()

this method can allows us to set number of paper

Class name: publisher

Attributes:

Name: Alphanumeric

Phone: this is 10 digit numbers

Address: Alphanumeric

7. Meta report

# 7.1 Group structure

|  |  |  |
| --- | --- | --- |
| **Group member** | **Roles** | **Artifacts** |
| Zhengyu Yuan | Manager  Systems Architect  Requirements Engineer  Designer | 1. Distribute job to each member 2. Plan work schedule 3. Design the structure of the system 4. Elicit requirements from the client 5. Analysis the requirements 6. Document the requirements 7. Create use case diagram and the detailed description 8. Combine each parts together to the final version report |
| Junyan Fan | Requirements Engineer  Designer | 1. Analysis the requirements 2. Document the requirements 3. Create Use Case Diagram and the detailed description |
| Yuanxin Xu | Designer  Analyst  Documentation | 1. Introduce the Business case which specify the benefit of the system 2. Describe the overview of the project 3. Produce the detailed plans for the whole project |
| Yi Jin | Designer  Analyst  Documentation | 1. Elicit requirements from the client 2. Analyze the risks 3. Create domain model which include Class Diagram and detailed of each class |
| Ruixi He | Analyst  Documentation | 1. Create domain model which include Class Diagram and detailed of each class 2. Maintain documents of the group 3. Record the detail of each group meeting |

# 7.2 Summary of each group meeting

Meeting1, Week3 (12/08/2014)

In the first meeting we read the project specification, basic understand what the assignment want us to do. We also make an agreement about the work division and assigns different jobs to each group member.

Meeting2, Week3 (15/08/2014)

In the second meeting, first we discussed the initial requirements including the functional and non-functional requirements of the project, and prepare the questions about the system to ask the client (tutor).

Meeting3, Week4 (22/08/2014)

In the third meeting, Zhengyu Yuan and Junyan Fan (the designers) provided the Use Case Diagram, after that we discussed about the Use Case of this project and modified the Use Case Diagram to the final version.

Meeting4, Week5 (29/08/2014)

In this meeting, the class designer Ruixi He finished the class diagram, and all the members confirmed the final architecture of the system.

Meeting5, Week6 (11/09/2014)

In the last meeting, each group member provide the final version of their works and hand in the work diaries to Zhengyu Yuan, the manager to turn out the final report of this project. And finally, all the members discussed what we had missed for the whole project.

# 7.3 Agenda and report of meeting

|  |  |  |
| --- | --- | --- |
| Meeting 1 |  | Date: 12/08/2014  Time: 2hours |
| Meeting called by  Attendance: | Zhengyu Yuan – Manager  Zhengyu Yuan,  Junyan Fan,  Yuanxin Xu,  Yi Jin,  Ruixi He |  |
| Place: | Library |  |
|  |  |  |
| 10:30 – 11:30 | Read the specification of the project. |  |
| 11:30 – 12:30 | Distributed roles and assigned works to each group member. |  |

|  |  |  |
| --- | --- | --- |
| Meeting 2 |  | Date: 15/08/2014  Time: 2hours |
| Meeting called by  Attendance: | Zhengyu Yuan – Manager  Zhengyu Yuan,  Junyan Fan,  Yuanxin Xu,  Yi Jin,  Ruixi He |  |
| Place: | Building 3.127 |  |
| 13:30 – 14:30  14:30 – 15:30 | Discussed the functional and non-functional requirements of the project.  Prepare the questions to elicit the requirements from the client (tutor). |  |
| Meeting 3 |  | Date: 22/08/2014  Time: 2hours |
| Meeting called by  Attendance: | Zhengyu Yuan – Manager  Zhengyu Yuan,  Junyan Fan,  Yuanxin Xu,  Yi Jin,  Ruixi He |  |
| Place: | Building3.125 |  |
| 11:30 – 12:30  12:30 – 13:30 | Create the initial Use Case Diagram by the designers.  Discussed and modified the Use Case by all the members. |  |

|  |  |  |
| --- | --- | --- |
| Meeting 4 |  | Date: 29/08/2014  Time: 2hours |
| Meeting called by  Attendance: | Zhengyu Yuan – Manager  Zhengyu Yuan,  Junyan Fan,  Yuanxin Xu,  Yi Jin,  Ruixi He |  |
| Place: | Building3.125 |  |
| 11:30 – 12:30  12:30 – 13:30 | Ruixi He created the domain model included the class diagram and detail description.  Determined the final architecture of the project. |  |

|  |  |  |
| --- | --- | --- |
| Meeting 5 |  | Date: 11/09/2014  Time: |
| Meeting called by  Attendance: | Zhengyu Yuan - Manager  Zhengyu Yuan,  Junyan Fan,  Yuanxin Xu,  Yi Jin,  Ruixi He |  |
| Place: | Library |  |
| 12:00 – 12:30  12:30 – 13:00  13:00 – 14:00 | All members present their works to manager.  Combine the work together to make the final version of the report.  Final discussion to find the missing thing or deficiency. |  |

# 7.4 Work Diaries

Ruixi He:

Meeting1 12/08/2014

Today, we have the first meeting of the assignment1 of CSCI222. This is the first group assignment for me, which is the reason making me feel a bit confuse in the meeting time. In this meeting we do not do too much thing, just read the specification of the assignment and try to understand what we should do and how we should assign the work to each member. After read the specification, we know that we should design a conference management system to help the conference organizers manage the complexity of the conference process. Then, we make an agreement with the work distribution as elect Zhengyu Yuan as the manager to manage the whole teamwork, and also be the system architect to design the whole structure of the system. And I am the analyst and documentation to record all the information in each meeting.

Meeting2 15/08/2014

Today, in the second meeting, we discuss and provide total 27 functional and non-functional requirements. After that, we also make a discussion about what questions should we ask the client to elicit the potential or missing requirements.

Meeting3 22/08/2014

In today’s meeting, the Use Case designers, Zhengyu Yuan and Junyan fan, provide us the initial Use Case Diagram and the detail description of that to other member. Then we have a discussion to find out whether there is question about the Use Case Diagram, and modify then two the second version.

Meeting4 29/08/2014

In this meeting, I need to present the domain model, and I create the class diagram within the detail description about it. And Zhengyu Yuan, the manager of our group, try to determine the final architecture conference manager system.

Meeting5 11/09/2014

This is the final meeting before submit this assignment. At the beginning of the meeting, we present our work to the manager, and we try to combine them together to compose the report. And also, we will have a big discussion to find whether there is something missing or failure.

Yuanxin Xu:

Meeting 1 12/08/2014

Today we have the first group meeting for the arrangement of the first assignment. We just talk about the specification of this assignment.

Meeting 2 15/08/2014

Today we talk about the detailed specification such like the requirements and the whole plan.

Meeting 3 22/08/2014

Zhengyu Yuan and Junyan Fan, provide us the initial Use Case Diagram and the detail description of that to other member. I clearly know that what should I do.

Meeting 4 29/08/2014

I work out the whole plan and the overall requirements review.

Meeting 5 11/09/2014

Combine our works and print out the paper.

Yi Jin:

Meeting 1 12/08/2014

Spent 2 hours attend the group meeting, assign different work for each group members

Meeting 2 15/08/2014

Spent 3 hours on researching risk analysis and assignment specification.

Meeting 3 22/08/2014

Spent 3 hours finish the risk analysis

Meeting 4 29/08/2014

I work out the whole plan and the overall requirements review.

Meeting 5 11/09/2014

Combine our works and set contents and layout of the final report.

Junyan Fan:

I joined in group, I start to focus on this assignment. We plan the group meeting is every week. I think I need read this assignment note before the first meeting.

Meeting1, Week3 (12/08/2014)

We have a first meeting, every member arrived there. In this meeting, we discuss this assignment, and discuss who best role in every part is. My task is use case, requirement and part of the document. My leader told me that try to do my part if I have problem, we will assign task again. After the meeting, I think about my task, I think this task is fit for me.

Meeting2, Week3 (15/08/2014)

someone‘s role have change, I didn’t change, I discuss the all process with other group member, that can let me know the process of program and list all requirement, we will use this build use case. After the meeting, zhengyu Yuan and I start to build use case.

Meeting3, Week4 (22/08/2014)

We finish the use case, and send to other group member. This use case can help other member‘s work.

Meeting4, Week5 (29/08/2014)

Ruixi show his work for team mate, and explain it, discuss this diagram whether is right, modify some mistakes. After that, we put these in document.

Meeting5, Week6 (11/09/2014)

We help other member do to their task, everyone’s task will be end, I and other group member put all things to report. All the members discussed what we had missed for the whole project. We will get the final vision.

Zhengyu Yuan

29.08.2014:

Spent 2 hours attend the group meeting, assign different work for each group members

08.09.2014:

Spent 5 hours finish the first SRS (software requirements specifications) with Junyan Fan.

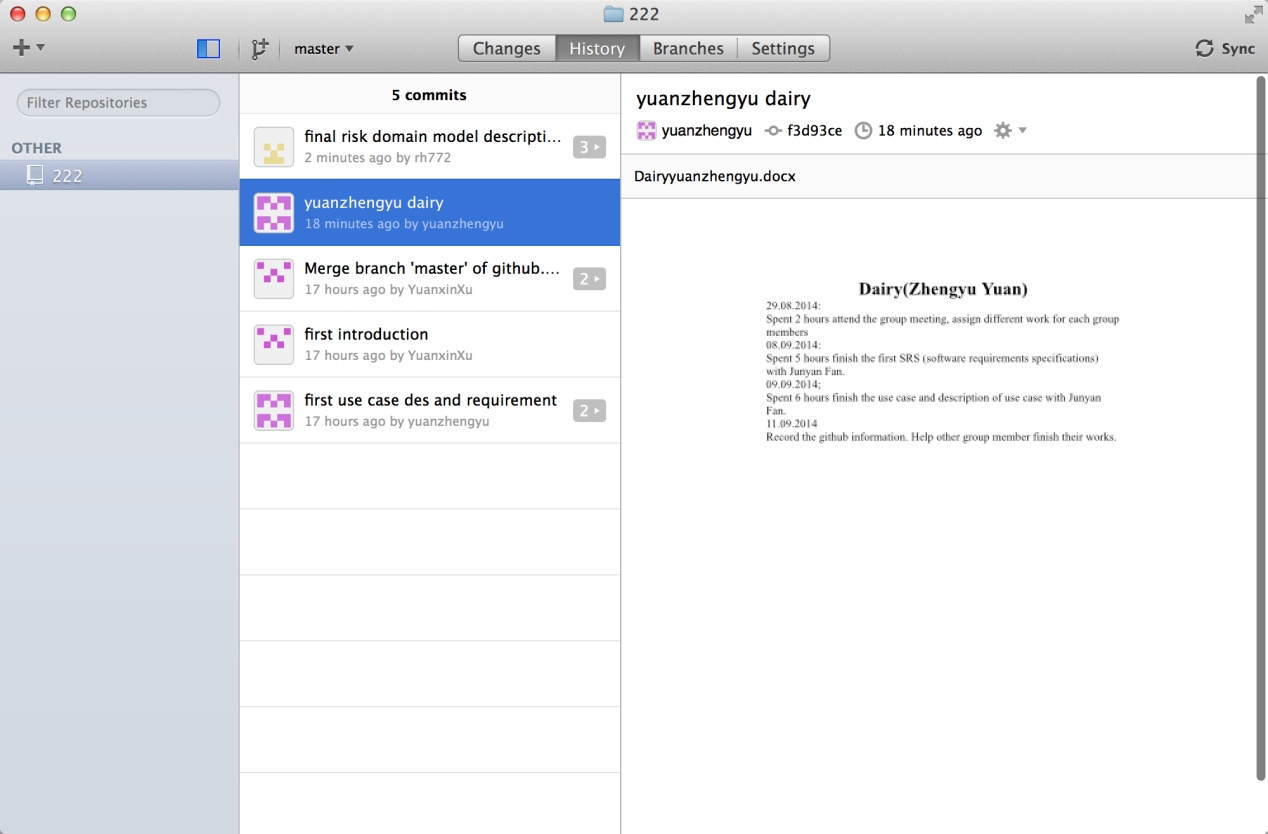
09.09.2014;

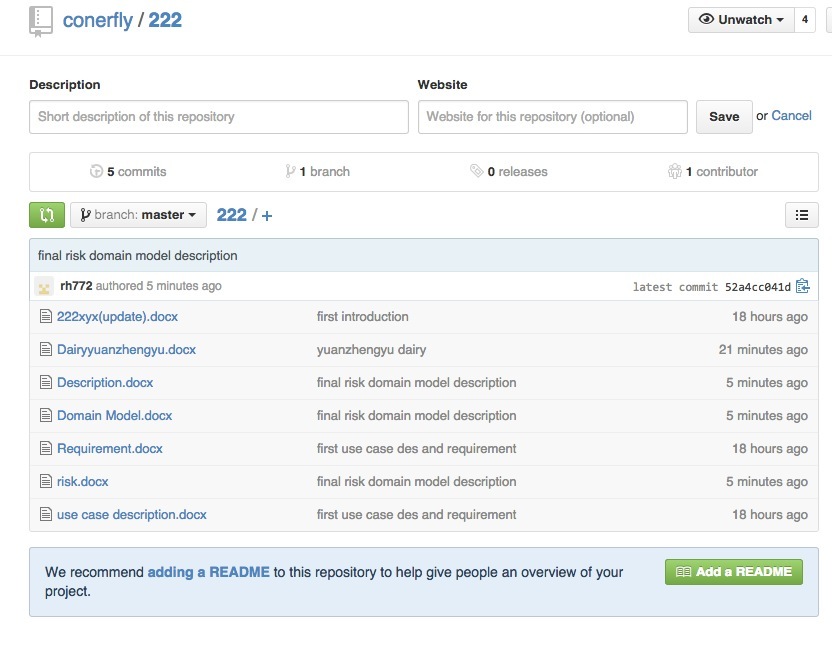
Spent 6 hours finish the use case and description of use case with Junyan Fan.

11.09.2014

Record the github information. Help other group member finish their works.

8. Version Control (GitHub)





**Member contribution assessment**

|  |  |
| --- | --- |
|  | Contribution |
| Junyan Fan 3767929 | 100% |
| Ruixi He 4174458 | 100% |
| Zhengyu Yuan 4327664 | 100% |
| Yuanxin Xu 4332969 | 100% |
| Yi Jin 4370673 | 100% |