Flight Management System

Software Requirements Specification

Version 1.0

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 01/Apr/2015 | 1.0 | First Version of SRS  Finished Section 1 | Matthew King  Michael Kong  Jicheng Li  Siyuan Chen |

Table of Contents

1. Introduction 4

1.1 Purpose 4

1.2 Scope 4

1.3 Definitions, Acronyms and Abbreviations 4

1.4 References 5

1.5 Overview 5

2. Overall Description 5

2.1 Product Perspective 5

2.1.1 System Interfaces 5

2.1.2 User Interfaces 5

2.1.3 Hardware Interfaces 5

2.1.4 Memory Constraints 5

2.1.5 Operations 6

2.2 Product Functions 6

2.3 User Characteristics 6

2.4 Constraints 6

2.5 Assumptions and Dependencies 7

3. Specific Requirements 7

3.1 Functional Requirements 7

3.1.1 Users Management Subsystem 7

3.1.1.1 Administrators Side 7

3.1.1.2 Program Committee Chair Side 8

3.1.1.3 User Side 9

3.1.1.4 System Side 10

3.1.2 Paper Management Subsystem 10

3.1.2.1 Paper Submission 10

3.1.2.2 Paper Assignment to Reviewers 11

3.1.2.3 Conference Proceeding 14

3.1.3 Paper Review Management Subsystem 14

3.1.3.1 Program Committee Chairs Side 14

3.1.3.2 Reviewers Side 16

3.1.3.3 Authors Side 17

3.2 Non-Functional Requirement 18

Software Requirements Specification

# Introduction

## Purpose

The purpose of this document is to describe the external behaviour of a Flight Management System. In this document, a detailed overview of the software’s parameters and goals. Optional functions, that may be implemented later, are also described. Also described within this document are non-functional requirements that are necessary in providing a complete and thorough description of all system requirements.

The software development team and the potential users are the main audience of this document.

## Scope

The software system to be produced is a command line based airline management system. This system is aimed at two general groups of users: potential customers of the system who will be performing flight bookings on the system, and the airline management staff who will handle backend roles such as managing flights and services. The entire system will be under the control of the system administrator whose role is to maintain all services and users related to the system.

There are four main components to the system which will be described below:

* Reservation System that manages all flight reservations, seat selection, ticketing, flight availability, flight details, rates and conditions.
* Profile Subsystem that manages individual passengers and travel agency profiles.
* Service Subsystem that manages in-flight services such as food and drinks.
* Reporting Subsystem to generate various summary reports such as Passenger Report, Cash Report and Booking Summaries.

Although the system makes booking flights more convenient for customers, that are some limitations of the system:

* Customers who want to modify the services that they receive on their flight need to contact the airline staff. The airline staff will then change the services accordingly.

## Definitions, Acronyms and Abbreviations

Refer to Glossary document.

## References

Rational Unified Process, SRS sample (Sample - SoftwareRequirementsSpecification.pdf)

## Overview

The remainder of this document is composed of two main sections:

Section 2: Overall Description: Description of the general factors that affect the system and provide high level requirements of the system.

Section 3: Specific Requirements: Description of all software requirements the system must have to meet the requirements and necessities of the client.

# Overall Description

## 2.1 Product perspective

All current activities under the management are done manually at present. Processes that are executed manually, such as booking tickets at the airline ticketing office, are inefficient; considerable amounts of time and manpower are required.

From this ineffiency arises the need to build a Flight Management System to increase efficiency, productivity and convenience for the customers and potential passengers, airline staff and the owners of the airline for which this syste will be built. Processes that are currently inefficient, such as booking tickets and managing flights, will be simplified and made convenient.

### System Interface

The Flight Management System to be developed is a command-line based application. It is to be deployed on the Internet where users are required to login to the system.

Clients can login to the system concurrently from any computer that has supports the command-line application and an Internet connection to interact with the system. Clients can then access the functions and services through the client command-line application.

A server will host the core of the Flight Management System such as the component to handle clients and the database. Client machines will connect to the server and interact with the system via a command line program. The command line program will act as the interface to accept and process client requests, and retrieve data from the central server of the system.

### User interfaces

Client machines will be featuring command line interfaces. Any computer that supports a command-line window will be able to access the Flight Management System. All users will be using the same interfaces; however, users of varying roles will view different interfaces once they have logged into the system.

All access to data and information such as flight schedules, user info and services will be restricted to the command line program provided in the Flight Management System.

### Hardware interfaces

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

### Memory Constraints

All client machines running the command-line application require a minimum of 100MB.

### Operations

The Flight Management System must have high ease of usability. No specific information or skills are required to use the system.

System data storage files should be well managed by the system so that the network administrator can have easy access to them and would not require and additional information or technical skills.

Backup and recovery operations must be specified in case of power failure, file corruption, or any other uncertain circumstances that may disrupt the system.

## Product Functions

The two main functions of the Flight Management System are to allow customers to book flights and allow the airline managers to manage the various aspects of the system effectively and efficiently.

For customers and travel agencies, the Flight Management System allows them to perform flight bookings and account management through following main functions:

* List available flights with details about each flight such as destination, origin and price
* Make flight bookings and allow payment by credit card or frequent flier points

For the staff and managers, the Flight Management System allows them to perform flight management services through the following main functions:

* Edit flights that the airline can take and all aspects of a flight including routes, planes, schedule and ticket prices
* Edit services that are available on the flights including availability and prices
* Cancel flights and all bookings that were made for that flight
* Print summary reports
* Manage customer and travel agency accounts including seat reassignment, flight reassignment, customer watch/no-fly list management and account closure

A system administrator controls all aspects of the system through the following main functions:

* Manage all user accounts including manager, customer and travel agency accounts
* Maintain the system through the various system configuration settings

The functionalities listed here are centered on the business needs and requirements of the airline that requested this system. In a further section of this document, these functionalities will be subdivided into smaller parts and be covered in more detail. The functionalities that make up the Flight Management System are aimed to make tasks and transactions simpler and more convenient.

## User Characteristics

There exist three main users of the Flight Management System that are the customers and travel agencies, system managers and system administrator.

* Customers and travel agencies need to have a basic understanding of how to use the system
* System managers should have strong working knowledge of the features of the system that they have control over
* System administrator should have strong knowledge on how a flight management system is supposed to work, and is able to install and deploy the system

## Constraints

The system should obey and satisfy the following constraints:

* Authentication: The system should only allow users to login when the right email and password and entered
* Access Control: The appropriate access rights should be given to the various users of the system. (Eg: Customers should not be allowed to edit flights)
* Back up and recovery: System files should be easily recoverable through back up in the event of data corruption or data loss
* The system must be developed by June 2015
* The core of the system must be developed using either Java or C++ programming languages

## Assumptions and dependencies

All potential users must have an existing email address.

# Specific Requirements

Each requirement specified in this section of the Flight Management System, whether it is a functional or non-functional one, is categorized into two main groups:

* Mandatory: Requirements of highest importance. These requirements are core functionalities of the Flight Management System and must be implemented.
* Optional: Requirements of lowest importance. Optional requirements are enhanced functionalities of the Flight Management System. These requirements should only be implemented after all mandatory requirements are implemented.

## 3.1 Functional Requirements

*3.1.1 Reservation Subsystem*

This section decribes the functionalities that the Flight Management System allows management for adding, changing and modifying all flight reservations, seat selection, tickets, flight availability, flight details, and flight rates and conditions. These are functions are provided to the Customer, Travel Agency, Normal Staff, Flight Manager and Reservation System Manager.

3.1.1.1 Administrator Side

This section includes functions for the system admin to manage the configuration of the system and assign different roles to the users.

|  |  |  |
| --- | --- | --- |
| **Requirement #:** R\_3.1.1.1\_01 | **Requirement Type:** Functional | **Use Case ID**: - |
| **Description:** The system should allow the administrator to edit Subsystem Managers | | |
| **Rationale:** System Admin wants to edit Subsystem Managers | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** System Administrator successfully edit Subsystem Managers | | |
| **Dependencies:** None. | | |
| **Classification:** Mandatory | | |
| **History:** Created by Michael Kong on 10/4/2015 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** R\_3.1.1.1\_02 | **Requirement Type:** Functional | **Use Case ID**: - |
| **Description:** The system should allow the administrator to edit an existing non-administrator user profile. A profile consists of Title, First Name, Last Name, Gender, Date of Birth, Phone, Email (cannot be edited), Password (cannot be empty), Street Address, State, City, Country, Credit Card Type, Credit Card Number, Frequent Flier Points, Passport Holder, Status (can be empty), Travel Agent (can be empty) | | |
| **Rationale:** System Admin wants to edit an existing non-administrator user profile. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** System Admin successfully edits an existing non-administrator user profile. | | |
| **Dependencies:** None. | | |
| **Classification:** Mandatory | | |
| **History:** Created by Michael Kong on 10/4/2015 | | |

*3.1.1.2 Program Committee Chair Side*

This section includes functions which allows the Program Committee (PC) Chairs to oversee and manage users of the system

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.1.2\_01 | **Requirement Type:** Functional | **Use Case ID:** 12 |
| **Description:** The system should allow the PC chairs to assign Program Committees. By going through the list of users, assignment of Program Committees are based on the users interests and how closely tied the user’s interest is to the conference topic. | | |
| **Rationale:** PC chairs needs to assign Program Committees | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** PC chairs successfully assign a normal user the role of Program Committee | | |
| **Dependencies:** Users who are being assigned the role and the PC chair must have an existing account. Refer to F\_3.1.1.3\_01. | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.1.2\_02 | **Requirement Type:** Functional | **Use Case ID:** 13 |
| **Description:** The system should allow the PC chairs to monitor the program committee. Types of information required when monitoring the program committee includes   * Number of papers assigned to each Program Committee * Papers assigned to each PC * Papers which has been reviewed by each PC * Papers which has not been reviewed by each PC * Reviews on paper wrote by each PC * Preferences on each paper | | |
| **Rationale:** PC chairs want to monitor the Program Committees. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** PC chairs are able to monitor the Program Committee successfully. | | |
| **Dependencies:** PC chair and program committee both must have an existing account. Refer to F\_3.1.1.3\_01. | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.1.2\_03 | **Requirement Type:** Functional | **Use Case ID:** 16 |
| **Description:** The system should allow the PC Chairs to check on the latest events. The type of events include   * Submission of a paper by an author * An upcoming deadline * Submission of review by a reviewer * An event that recently expired * Creation of new accounts | | |
| **Rationale:** PC chairs want to check out the latest events. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** PC chairs are able to view the list of latest events successfully. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

*3.1.1.3 User side*

This section includes functions which allow users to interact with the system.

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.1.3\_01 | **Requirement Type:** Functional | **Use Case ID:** None |
| **Description:** The system should allow creation of new accounts by new users. Types of information required when creating a new account includes   * First Name * Last Name * User Name * Email * Password * Expertise * University | | |
| **Rationale:** New users need to create a new account. | | |
| **Source:** None | | |
| **Fit Criterion:** New users successfully create a new account. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.1.3\_02 | **Requirement Type:** Functional | **Use Case ID:** None |
| **Description:** The system should allow users to login to their accounts. | | |
| **Rationale:** Users to login to the system. | | |
| **Source:** None | | |
| **Fit Criterion:** Users successfully login into their accounts. | | |
| **Dependencies:** Users who log in must have an existing account. Refer to F\_3.1.1.3\_01. | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 6/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.1.3\_03 | **Requirement Type:** Functional | **Use Case ID:** 4 |
| **Description:** The system should allow users to change their own account passwords. | | |
| **Rationale:** Users to change their account password. | | |
| **Source:** None | | |
| **Fit Criterion:** Users successfully change their account password. | | |
| **Dependencies:** Users who modify their account passwords must have an existing account. Refer to F\_3.1.1.3\_01. | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 3/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.1.3\_04 | **Requirement Type:** Functional | **Use Case ID:** 3 |
| **Description:** The system should allow users to modify their personal details. | | |
| **Rationale:** Users to modify their personal details. | | |
| **Source:** None | | |
| **Fit Criterion:** Users successfully modifies their personal details. | | |
| **Dependencies:** Users who modify their personal details must have an existing account. Refer to F\_3.1.1.3\_01. | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 3/9/2014 | | |

**3.1.2 Paper Management Subsystem**

This section captures the functionalities that the Conference Management System provides to the users to manage research papers, including submission of research papers, assignment of research papers to the reviewers.

*3.1.2.1 Paper submission*

This section includes functionalities for the users to submit papers to the system.

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.2.1\_01 | **Requirement Type:** Functional | **Use Case ID:** 5 |
| **Description:** The system should allow existing users to submit papers to the system. User who submits a paper will automatically be assigned the role of author. During submission of papers, authors are to fill in forms about the details of the paper and also information about the author.   * Information of author includes  1. First Name 2. Last Name 3. Email 4. University  * Information of paper includes  1. Title 2. Abstract 3. Keywords   Authors who submit the paper will fill in details about other authors who have contributed to the paper. Other authors who have contributed to the paper must have an existing account. Authors are permitted to submit an unlimited amount of papers. | | |
| **Rationale:** Existing user needs to submit a paper to the system. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** User successfully submits a paper to the system. | | |
| **Dependencies:** User who submits a paper must have an existing account. Other authors who have contributed to the paper must also have an existing account. Refer to F\_3.1.1.3\_01. | | |
| **Classification:** Mandatory | | |
| **History:**   * Created by Ng Shien Wee on 29/8/2014 * Edited by Ng Shien Wee on 3/9/2014 * Edited by Ng Shien Wee on 6/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.2.1\_02 | **Requirement Type:** Functional | **Use Case ID:** 6 |
| **Description:** The system should allow authors to modify their submissions. Modification includes changing the details of the authors and resubmission of paper. This must be done before the submission deadline. | | |
| **Rationale:** Authors to modify their submissions. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** Authors successfully modify their submissions. | | |
| **Dependencies:** User must have submitted a paper. Refer to F\_3.1.2.1\_01. | | |
| **Classification:** Mandatory | | |
| **History:**   * Created by Ng Shien Wee on 29/8/2014 * Edited by Ng Shien Wee on 6/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.2.1\_03 | **Requirement Type:** Functional | **Use Case ID:** None |
| **Description:** The system should be able to assign users who have contributed to the papers automatically. This is related to requirement F\_3.1.2.1\_01. This process is only done when multiple author details has been added to a paper submission. Based on the email address of the authors (excluding the author who submitted the paper), the system should be able to link the paper to the respective user accounts. | | |
| **Rationale:** System to assign users to their papers. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** System successfully assigns user to their papers. | | |
| **Dependencies:** When linking paper to user accounts, user account must exist. Refer to F\_3.1.1.3\_01. | | |
| **Classification:** Mandatory | | |
| **History:**   * Created by Ng Shien Wee on 29/8/2014 * Edited by Ng Shien Wee on 3/9/2014 | | |

*3.1.2.2 Paper assignment to reviewers*

This section includes functionalities for the Program Committees, which are also known as reviewers to be assigned a paper for review.

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.2.2\_01 | **Requirement Type:** Functional | **Use Case ID:** 8 |
| **Description:** The system should allow reviewers to look through the list of paper submitted and specify their preferences on each paper. When looking through the list of papers, reviewers are only allowed to look at the following   * Keywords * Abstract * Title   Preferences that can be specified by the reviewers include “Yes”, “No”, “Maybe” and “Conflict of Interest”. | | |
| **Rationale:** Reviewers specify their preference on each paper. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** Reviewers successfully specify their preference on each paper. | | |
| **Dependencies:** Reviewers must have an existing account. Refer to F\_3.1.1.3\_01. | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.2.2\_02 | **Requirement Type:** Functional | **Use Case ID:** None |
| **Description:** The system should allow PC chairs to automate the process of specifying user preferences on papers. This process is only done if a user has not specified any preferences on a paper. This automation process involves looking through the keywords of the paper submitted with the expertise of the reviewer. A percentage value would be given on how closely related they are. Closely related being how many keywords match the expertise of the reviewer. The higher the percentage, the more closely related they are. | | |
| **Rationale:** System needs to automatically assign preferences to paper. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** System successfully assigns preferences to paper. | | |
| **Dependencies:** None. | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.2.2\_03 | **Requirement Type:** Functional | **Use Case ID:** 14 |
| **Description:** The system should allow PC chairs to automate the process of assigning papers to the reviewers. This process involves looking at the preferences (Preferences being “Yes”, “No”, “Maybe” or “Conflict of Interest”) of the reviewer and assigns the paper to him/her. There is a possibility where a reviewer can be assigned a paper where his/her preference is “No”. This happens under a certain condition being   * A paper has reached its maximum number of reviewers * A reviewer has not been assigned the maximum number of papers. | | |
| **Rationale:** System needs to assign papers to users | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** System successfully assign papers to users. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.2.2\_04 | **Requirement Type:** Functional | **Use Case ID:** 15 |
| **Description:** The system should allow PC Chairs to manually assign paper to reviewers. The automation process may have assigned the paper to reviewers who have a conflict of interest towards the paper. | | |
| **Rationale:** PC Chairs need to manually assign paper to reviewers. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** PC Chairs successfully assign paper to reviewers. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.2.2\_05 | **Requirement Type:** Functional | **Use Case ID:** 21 |
| **Description:** The system should allow PC Chairs to set the number of reviewers a paper can receive. The number is dependent on the number of reviewers available and the amount of papers assigned. Default is set at 5. | | |
| **Rationale:** PC Chairs need to set the number of reviewers per paper. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** PC Chairs successfully set the number of reviewers per paper. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:**   * Created by Ng Shien Wee on 29/8/2014 * Edited by Ng Shien Wee on 6/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.2.2\_06 | **Requirement Type:** Functional | **Use Case ID:** 17 |
| **Description:** The system should allow PC Chairs to set the number of papers reviewers can receive. The number set by the PC chair is dependent on the number of papers submitted and the number of reviewers available. Default is set at 3. | | |
| **Rationale:** PC Chairs need to set a global limit for the number of papers all reviewers receive. Global limit being the number of papers all reviewers receive. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** PC Chairs successfully set a global limit for it for the number of papers all reviewers receive. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:**   * Created by Ng Shien Wee on 29/8/2014 * Edited by Ng Shien Wee on 6/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.2.2\_07 | **Requirement Type:** Functional | **Use Case ID:** 18 |
| **Description:** The system should also allow manual assignment of the number of papers a reviewer can receive. Some reviewers may be busier than the others and assigned lesser amount of papers for reviewing or vice versa. | | |
| **Rationale:** PC Chairs need to manually assign the number of papers a reviewer can receive. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** PC Chairs successfully assign the number of papers a reviewer can receive. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:**   * Created by Ng Shien Wee on 7/9/2014 | | |

*3.1.2.3 Conference Proceeding*

This section includes functionalities for the preparation of conference proceedings.

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.2.3\_01 | **Requirement Type:** Functional | **Use Case ID:** 20 |
| **Description:** The system should allow the PC Chairs to automate the process to generate the conference proceeding. This is done by looking through the list of papers, then generating a list of accepted papers. List of accepted papers are grouped according to the topics. | | |
| **Rationale:** System automatically generates conference proceeding. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** System successfully generates the conference proceeding. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.2.3\_02 | **Requirement Type:** Functional | **Use Case ID:** 26 |
| **Description:** The system should allow Program Committee Chairs to manually group the list of papers based on the topics for the conference proceeding. The automation process may have grouped the paper incorrectly. | | |
| **Rationale:** PC chairs manually group papers based on their topics. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** PC Chairs successfully group papers based on the topics. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

**3.1.3 Paper Review Management Subsystem**

This section captures the functionalities that the Conference Management System provides for users to execute different actions related to paper reviews. Different roles include Program Committee Chairs, Program Committee and the authors.

*3.1.3.1 Program Committee Chairs Side*

This section includes functionalities for the Program Committee Chairs to manage paper reviews.

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.3.1\_01 | **Requirement Type:** Functional | **Use Case ID:** 25 |
| **Description:** The system should allow Program Committee Chairs to enable or disable paper review discussions among the reviewers. | | |
| **Rationale:** Program Committee Chair to enable or disable review discussions. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** Program Committee Chairs successfully enables or disable review discussions. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.3.1\_02 | **Requirement Type:** Functional | **Use Case ID:** 19 |
| **Description:** The system should allow Program Committee Chairs to accept or reject a paper based on the reviews and discussion submitted by the reviewers. PC chairs will have a page listing all the papers submitted, choose a paper and get more details about the paper. Details about the paper include   * Information about the contributed authors * Paper information * Reviewers * Completed Reviews * Discussions | | |
| **Rationale:** Program Committee Chair to enable or disable review discussions. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** Program Committee Chairs successfully enables or disable review discussions. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.3.1\_03 | **Requirement Type:** Functional | **Use Case ID:** 24 |
| **Description:** The system should allow Program Committee Chairs to enable or disable the functionality where the authors are able to leave a respond about the reviews on his/her paper. | | |
| **Rationale:** Program Committee Chair to enable or disable author response on the review. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** Program Committee Chairs successfully enables or disable author response on the review. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.3.1\_04 | **Requirement Type:** Functional | **Use Case ID:** 23 |
| **Description:** The system should allow Program Committee Chairs to enable or disable paper review submissions by the Program Committee. | | |
| **Rationale:** Program Committee Chair to enable or disable review submissions. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** Program Committee Chairs successfully enables or disable review submissions. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 6/9/2014 | | |

*3.1.3.2 Reviewers Side*

This section includes functionalities for the Program Committee to submit paper reviews and discuss reviews done by other reviewers.

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.3.2\_01 | **Requirement Type:** Functional | **Use Case ID:** 9 |
| **Description:** The system should allow the reviewers to choose from a list of assigned papers and review the paper based on the given evaluation form. The reviewer submits the review when he/she has completed the review. | | |
| **Rationale:** Reviewers to submit paper reviews. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** Reviewers successfully submits the paper review. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.3.2\_02 | **Requirement Type:** Functional | **Use Case ID:** 10 |
| **Description:** The system should allow the reviewers to modify the reviews he/she has submitted. | | |
| **Rationale:** Reviewers to modify paper reviews | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** Reviewers successfully modifies the paper review. | | |
| **Dependencies:** The reviewer must have submitted a paper review to modify it. Refer to F\_3.1.3.2\_01 | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.3.2\_02 | **Requirement Type:** Functional | **Use Case ID:** 11 |
| **Description:** The system should allow reviewers to discuss about the reviews submitted by different reviewers, who have been assigned the same paper. Authors are not allowed to see the discussion among the reviewers. | | |
| **Rationale:** Reviewers to discuss with other reviewers about the paper reviews | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** Reviewers successfully discuss with other reviewers about their paper reviews. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

*3.1.3.3 Authors Side*

This section includes functionalities for authors who have submitted papers to respond to reviews provided to them.

|  |  |  |
| --- | --- | --- |
| **Requirement #:** F\_3.1.3.3\_01 | **Requirement Type:** Functional | **Use Case ID:** 7 |
| **Description:** The system should allow authors to respond to reviews on their paper. This is also known the rebuttal phase. Authors are allowed to read the reviews and leave a comment. | | |
| **Rationale:** Authors are able to leave a comment regarding the reviewers review. | | |
| **Source:** Client – Hoa Dam | | |
| **Fit Criterion:** Authors successfully leaves a comment. | | |
| **Dependencies:** User must have submitted a paper. Refer to F\_3.1.2.1\_01 | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee on 29/8/2014 | | |

3.2 Non-Functional Requirement

|  |  |  |
| --- | --- | --- |
| **Requirement #:** NFREQ01 | **Requirement Type:** Performance | **Use Case #:** None |
| **Description:** The system should be able to work 24/7 except during maintenance. | | |
| **Rationale:** All users want to access the system 24/7. | | |
| **Source:** All users who interact with the system | | |
| **Fit Criterion:** Authorized users can use the system at any time (24/7). | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee and Girit on 3/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** NFREQ02 | **Requirement Type:** Performance | **Use Case #:** None |
| **Description:** The system should be responsive. The system must respond to actions by any users within 20 seconds. | | |
| **Rationale:** All users want to receive a quick respond from the system. | | |
| **Source:** All users who interact with the system | | |
| **Fit Criterion:** Response by the system towards the user’s action is given within 20 seconds. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee and Girit on 3/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** NFREQ03 | **Requirement Type:** Performance | **Use Case #:** None |
| **Description:** The system should generate the conference proceeding within no more than 1 minute. | | |
| **Rationale:** The PC chairs wants to view the conference proceeding as quickly as possible. | | |
| **Source:** PC chairs | | |
| **Fit Criterion:** The conference proceeding is generated within 1 minute. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee and Girit on 3/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** NFREQ04 | **Requirement Type:** Performance | **Use Case #:** None |
| **Description:** The system should display details about an author and paper when prompted by the PC chairs within 20 seconds. | | |
| **Rationale:** The PC chairs want to obtain more information about the author or the paper as quickly as possible. | | |
| **Source:** PC chairs | | |
| **Fit Criterion:** The details about authors and the paper are displayed for the PC chair within 20 seconds. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee and Girit on 3/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** NFREQ05 | **Requirement Type:** Usability | **Use Case #:** None |
| **Description:** The system should provide a simple “Help” page for new users or existing users at the main page of the system. The help page provides simple tips to learn more about the system and provide contact details so that the user can ask for more advanced help. | | |
| **Rationale:** To ease users who are new in using the system. | | |
| **Source:** All users who interact with the system. | | |
| **Fit Criterion:** The help is accessible to all users of the system. | | |
| **Dependencies:** None | | |
| **Classification:** Optional | | |
| **History:** Created by Ng Shien Wee and Girit on 3/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** NFREQ06 | **Requirement Type:** Usability | **Use Case #:** None |
| **Description:** The system should be easy to use for existing users or new users. It should not require any specific knowledge or technical skills. No additional training should be provided for the users so that within 2 days of exploring the system, the user should have a good overview of the system. | | |
| **Rationale:** User wants to use the system without going through any additional training. | | |
| **Source:** All users who interact with the system. | | |
| **Fit Criterion:** The user is able to use the Conference Management System within a 2 days period. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee and Girit on 3/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** NFREQ07 | **Requirement Type:** Usability | **Use Case #:** None |
| **Description:** The system should run on command line and in English. | | |
| **Rationale:** User wants to use a familiar language when using the system. | | |
| **Source:** All users who interact with the system. | | |
| **Fit Criterion:** The system runs on command line and in English. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee and Girit on 3/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** NFREQ08 | **Requirement Type:** Reliability | **Use Case #:** None |
| **Description:** The system should recover from failure no more than 2 days. | | |
| **Rationale:** System to recover from a failure as fast as possible. | | |
| **Source:** All users who interact with the system. | | |
| **Fit Criterion:** The system not taking more than 2 days to recover. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee and Girit on 3/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** NFREQ09 | **Requirement Type:** Reliability | **Use Case #:** None |
| **Description:** The system should support more than 30 simultaneous users at a time. | | |
| **Rationale:** To support multiple users to use the system. | | |
| **Source:** All users who interact with the system. | | |
| **Fit Criterion:** The system should support more than 30 users at a time. | | |
| **Dependencies:** None | | |
| **Classification:** Mandatory | | |
| **History:** Created by Ng Shien Wee and Girit on 3/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** NFREQ10 | **Requirement Type:** Security | **Use Case #:** None |
| **Description:** The system should automatically log out after 10 minutes of idle time. | | |
| **Rationale:** To control the amount of users who are currently logged on to the system. | | |
| **Source:** All users who interact with the system. | | |
| **Fit Criterion:** The system logs out a user who has been idle for more than 10 minutes. | | |
| **Dependencies:** None | | |
| **Classification:** Optional | | |
| **History:** Created by Ng Shien Wee and Girit on 3/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** NFREQ11 | **Requirement Type:** Performance | **Use Case #:** None |
| **Description:** The system should be able to store up to 200 user profiles and 200 different papers submitted by users. | | |
| **Rationale:** System to store as many papers and user profiles as possible. | | |
| **Source:** All users who interact with the system. | | |
| **Fit Criterion:** 200 user profiles and 200 different papers are stored in the system | | |
| **Dependencies:** None | | |
| **Classification:** Optional | | |
| **History:** Created by Ng Shien Wee and Girit on 3/9/2014 | | |

|  |  |  |
| --- | --- | --- |
| **Requirement #:** NFREQ12 | **Requirement Type:** Usability | **Use Case #:** None |
| **Description:** The system should have user documentation and manuals to instruct users on how to use the Conference Management System. | | |
| **Rationale:** To shorten the amount of time needed to get used to the system | | |
| **Source:** All users who interact with the system. | | |
| **Fit Criterion:** The documentation helped the users understand better of the system. | | |
| **Dependencies:** None | | |
| **Classification:** Optional | | |
| **History:** Created by Ng Shien Wee and Girit on 3/9/2014 | | |