Software Requirements Specification

Event Management System

Blithe Fan, CY Lim, Loo Yee, Matthew Boroczky, Michael Kong, Sandon Joubert, Toby He

TABLE OF CONTENTS

SOFTWARE REQUIREMENTS SPECIFICATION

1. INTRODUCTION

GLOSSARY - DEFINITIONS, ACRONYMS AND ABBREVIATIONS

Manager / Organiser - Individuals at the top of the hierarchy within the event. They have the ability to change all details about the event

Conference - a formal meeting for discussion. When considering Conferences the system is not limited to small academic conferences. It should support the management of large conferences such as Apple's WWDC conference in which thousands attend (~ 5000)

Event – An event is similar to a conference but can generally be much larger. Typically events also have multiple presentations.

Presentation - Presentations are held within conferences or events. One event/conference must have one or more presentations

Author – Refers to an author of a paper. Also refers to event speakers since authors are potential speakers.

System – Refers to the Event Organiser and its subsystems.

User – Refers to any user of the system.

COI – Conflict of Interest

Drawing Board - The graphical user interface provided to managers to enable then to draw seats , types of seat, accessory items and position object. This may also be used in the context of visitors/guests having the ability to select seats.

PURPOSE

The purpose of this document is to describe the external behaviour of the Event Organiser. In this document, a detailed overview of the software's parameters and goals are presented. 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 web based event organiser system. This system is aimed at 4 groups of users: event organisers, event attendees, paper authors and paper reviewers. 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 9 main subsystems that comprise the Event Organiser:

- User Management Subsystem that manages user profiles and access rights.
- Paper Management Subsystem that manages paper submissions, deadlines, sorting, filtering and storage.
- Paper Review Subsystem that manages reviewer requests, provides features for review procedures and protocols, and tracks reviews of papers.
- Event Creation Subsystem that allows the creation of events, choosing of venues and setting of ticket prices.
- Venue Creation Subsystem that allows venue building via a "room builder" feature.
- Event Logistics Subsystem that manages the various resources such as microphones and projectors during an event, and allow users to specify meal preferences.
- Ticketing and Payment Subsystem that allows users to purchase tickets.
- Analytics and Tracking Subsystem that tracks various aspects of the system such as the demographics of the event attendees and produces financial reports of events.
- In-house Event Subsystem that contains timetables of events, provides venue maps, facilitates check-in and sends post-event questionnaires to event attendees.

Although the system makes event organizing and booking convenient for all users, there are some limitations of the system:

- Usage is limited to newer browsers that support HTML5 specifications
- Only mobile devices that are running iOS7 and above can run the mobile application of the Event Organiser

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.

REFERENCES

– NIL –

2. OVERALL DESCRIPTION

2.1 PRODUCT PERSPECTIVE

All current activities under the present system are done manually. Processes that are executed manually, such as the collection and organization of event attendee particulars, are inefficient. To perform such tasks, considerable amounts of time and manpower are required.

This inefficiency in performing such tasks creates the need to build an Event Organiser system to increase efficiency, productivity and convenience for the event organisers, event attendees, paper authors and paper reviewers. The system will make processes that are currently inefficient, quicker and more convenient.

SYSTEM INTERFACE

The Event Organiser system will be built for web and mobile. It will be deployed on the Internet so that users can simultaneously access the system from any personal computer or mobile device that supports Internet services.

Servers on which the Event Organiser is deployed are connected to the Internet to allow client machines to access the Event Organiser system to utilize the various features provided by the system. The servers will receive and process client requests, retrieve, store and compute data from database servers, and send data back to clients.

USER INTERFACES

The user interface (UI) for the Event Organiser system is based on graphical user interfaces. The UI can be accessed from any personal computer or mobile device that supports Internet services. All users will access a similar UI with differences depending on the users' profile privileges. Users can only view the UI after logging into the system.

Access to the database and the servers will be performed through the UI provided by the web hosting company.

HARDWARE INTERFACES

Server modules must be able to execute on the servers. Web modules must be able to execute on Google Chrome and Mozilla Firefox on personal computers. Mobile modules must be able to execute on iOS7 and above.

MEMORY CONSTRAINTS

The web server machines must have a minimum of 5GB of storage and the database server machines must have a minimum of 20GB of storage. The personal computers must have a minimum of 500MB of free space.

OPERATIONS

The Event Organiser system should be easy for all users to use. No specific information or technical skills are required to use the system.

Access to the web server should be familiar to the system administrator such that they do not require any additional information or technical skills.

The database of the system and access to it should be familiar to the system administrator such that they do not require any additional information or technical skills.

Backup and recovery operations must be specified in case of database corruption, power outage or any other incidents which may disrupt the system.

USER CHARACTERISTICS

The Event Organiser consists of 2 types of users with possible 7 user states.

The 2 types of users found in the system are:

- **Public Visitor** - A public viewer is any user who enters the website and has the ability to view events, times, seat selection and etc. These forms of users are not stored in the database
- **Guests** - Guests are the default user of the system. Upon purchasing a ticket or creating an account the user becomes a guest. Guests inherits the ability to view the website as a Visitor does.

Below are the different states a Guest can have within the system:

- States Specific to a conference a Conference:
    - **Authors** – Users who submit papers into the system for reviewing. They should have some knowledge of online submissions and online communication.
    - **Reviewers** – Users who are eligible to review papers based on their expertise and knowledge. They should be knowledgeable in their individual fields because they might be selected to be part of a program committee to review papers.
    - **Chair** – Reviewers with additional privileges. They should know the processes that encompass paper reviewing so that the integrity of paper reviewing process is maintained. They have the ability to moderate conversations between Authors and Reviewers and have the final review on the paper.
- States for Conferences and Events:
    - Event **Attendees** – Users who will purchase tickets to events. Their knowledge of online systems can be extremely varied from novice users to expert users.
    - Event **Organisers/Managers** – Users who will create, manage and monitor events. They should have knowledge of event organization and event logistics.
    - **Presenters** – Users who either represent Authors that are unable to attend the event, or the authors themselves. They have a close association with the Author and the paper that they are presenting.
    - **Security** Personnel – Users who will be maintaining security at an event. They should have basic knowledge of Internet-enabled mobile devices and the features of the Event Organiser that are specific to their role.

Disclaimer: The below functionalities are just a small snippet of some of the functionalities our system provides. For an in depth view please see the Functional Requirements section.

PRODUCT FUNCTIONS

The main functions of the Event Organiser are listed below and categorized based on different types of users.

For Event Organisers/Managers, the Event Organiser system would provide the functionalities to:

- create and manage events
- monitor and manage submission of papers
- select a conference Chair from a pool of Reviewers
- manage and monitor a program committee

For Event Attendees, the Event Organiser system would provide the functionalities to:

- select seats
- purchase tickets
- view event timetable
- connect with other attendees that have the same interests
- receive notifications via email or mobile notifications

For Authors, the Event Organiser system would provide the functionalities to:

- upload and submit papers in PDF or LaTeX formats
- receive reviews on their papers
- elect a Presenter if their paper is selected and they are not able to attend the event

For Chair, the Event Organiser system would provide the functionalities to:

- elect and manage a program committee from a pool of Reviewers
- mediate a conversation between an Author and a Reviewer
- manage Reviewer access to papers with conflicts of interests
- assign papers to Reviewers based on Reviewer preferences

For Reviewers, the Event Organiser system would provide the functionalities to:

- review a given paper
- reject proposal to review a paper
- converse with an Author about their paper
- select preferences

For Presenters, the Event Organiser system would provide the functionalities to:

- upload their credentials
- act as a representative as the Author for an event
- accept or reject the offer to be a presenter for an Author

For Security Personnel, the Event Organiser system would provide the functionalities to:

- receive requests for assistance
- communicate with other Security Personnel

The functionalities listed here are centered on the business needs and requirements of the client that has 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 Event Organiser system are aimed to make tasks and transactions simpler and more convenient.

CONSTRAINTS

The system should obey and satisfy the following constraints:

- Authentication: The system should only allow users to login when the correct email and password are entered.
- Access Control: The appropriate access rights should be given to the various users of the system. (E.g.: Event attendees should not be allowed to edit event details)
- Backup and recovery: System files should be easily recoverable through back up in the event of data corruption or data loss.
- The system must have all Core and Main requirements implemented by June 2016.

ASSUMPTIONS AND DEPENDENCIES

a) All potential users must have an existing email address.
b) All potential users must have access to the Internet.
c) All potential users must have a device containing an HTML5 and JavaScript enabled web browser.
d) If users wish to access additional on-the-day features, which is recommended, they must have an apple device with access to the internet and have downloaded our application from the Apple app store.

3. SPECIFIC REQUIREMENTS

Each requirement (either functional or non-functional requirements) of Event Management System is ranked based on its level of priority:

1. Core: these requirements are the core functionalities of Event Management System. Without these requirements there will not be a functional Event or Conference Management system.

2. Main: these requirements are the important functionalities of Event Management System and should be implemented when all the Core requirements have been finished

3. Stretch: these requirements are the necessary functionalities of Event Management System and should be covered when Core and Main requirements have implemented

4. Dream: these requirements are the enhanced functionalities of Event Management System and should be considered only when all Core, Main and Stretch requirements are completed

REQUIREMENT DEPENDENCIES

DEPENDENCY TABLE

ID	Requirement	Platform	Type	Priority	Dependencies
F_U1	Create Profiles	Web	Functional	Core	
F_U2	Remove Profile	Web	Functional	Core	F_U1
F_U3	Profile Details	Web	Functional	Core	F_U1
F_U4	Update Profile	Web	Functional	Core	F_U1
F_U5	Unique ID and Password	Web	Functional	Core	F_U1
F_U6	Change Password	Web	Functional	Core	F_U1, F_U5
F_U7	Upgrade Account	Web	Functional	Core	F_U1
F_U8	Password Encryption	Web, IOS	Functional	Main	
F_U9	Create a Conference	Web	Functional	Core	F_U3, F_U7
F_U10	Enable Paper Submission for Event	Web	Functional	Main	F_U3, F_U7
F_U11	Create Email	Web	Functional	Main	F_U3, F_U7
F_U12	Event Reminders	Web	Functional	Main	F_U3
F_U13	View Presenter and Presentation Times	Web, IOS	Functional	Stretch	F_U3
F_U14	Multilingual support	Web	Functional	Dream	F_U3, F_U7, F_U16
F_U15	Default Mandarin Support	Web, IOS	Functional	Dream	F_U3
F_U16	Database Type	Web	Functional	Dream	
F_U17	Facebook Integration	Web, IOS	Functional	Stretch	F_U3
F_U18	Login via Facebook	Web, IOS	Functional	Stretch	
F_U19	Invite to Event via Facebook	Web, IOs	Functional	Stretch	F_U17
F_U20	Linkedin Integration	Web, IOS	Functional	Stretch	F_U3
F_U21	Invite to Event via Linkedin	Web, IOS	Functional	Stretch	F_U20
F_U22	Invite to Event via Contacts	IOS	Functional	Stretch	F_U3
F_PM1	Upload PDF Documents	Web	Functional	Core	

F_PM2	Upload LaTex Documents	Web	Functional	Main	
F_PM3	Upload Word Documents	Web	Functional	Main	
F_PM4	Upload ZIP Files	Web	Functional	Main	
F_PM5	Submit Paper	Web	Functional	Core	F_PM1, F_ PM2, F_PM3, F_PM4
F_PM6	Modify Submission	Web	Functional	Core	F_PM5
F_PM7	Match Accounts to Paper Contribution	Web	Functional	Main	F_PM5
F_PM8	Allow Automatic Detection of Paper Contribution	Web	Functional	Stretch	F_PM2
F_PM9	Restrict Upload Types	Web	Functional	Main	F_PM1, F_ PM2, F_PM3, F_PM4
F_PM10	Setup Categories and Subcategories	Web	Functional	Main	
F_PM11	Automatic Detection of Paper Abstract	Web	Functional	Stretch	F_PM1, F_ PM2, F_PM3, F_PM4
F_PM12	Set Limit on File Size	Web	Functional	Stretch	F_PM1, F_ PM2, F_PM3, F_PM4
F_PR1	Assign Papers to Reviewers	Web	Functional	Core	F_U3
F_PR2	Multiple Papers to Reviewers	Web	Functional	Core	F_U3
F_PR3	Delete Reviewer	Web	Functional	Core	F_U15
F_PR4	Distribute Papers to Reviewers	Web	Functional	Core	F_U3, F_U7
F_PR5	Blind Review	Web	Functional	Core	F_PR4
F_PR6	Double Blind Review	Web	Functional	Main	F_PR4
F_PR7	Scaling of Reviews based on Reviewer Competence	Web	Functional	Main	F_PR5
F_PR8	Accept/Reject Paper Review	Web	Functional	Main	F_PR1
F_PR9	Automatically Flag Reviews with a difference	Web	Functional	Main	F_PR5
F_PR10	Communicate between Author and Reviewer Moderated by Chair	Web	Functional	Main	F_U3, F_U7

F_PR11	Select Reviewer Preferences	Web	Functional	Core	F_U7
F_PR12	Send Request to Review Paper	Web	Functional	Core	F_U7
F_PR13	Accept/Reject Assigned Paper Request	Web	Functional	Core	F_PR12
F_PR14	View the Reviewer List	Web, IOS	Functional	Core	F_U3
F_PR15	Add Reviewer to Reviewer List	Web	Functional	Core	F_PR14
F_PR16	Check COI Manually	Web	Functional	Core	F_PR1
F_PR17	Check COI through Facebook and Linkedin	Web	Functional	Stretch	F_U17, F_U20
F_PR18	Customise Flag Value of Review	Web	Functional	Main	
F_PR19	Automatically Suggest Reviewers Based on Linkedin Skills	Web	Functional	Dream	F_U20
F_PR20	Send Paper Link to a Friend for Review	Web	Functional	Stretch	F_PM1, F_ PM2, F_ PM3, F_PM4
F_PR21	Comment on Papers	Web	Functional	Core	F_U3, F_U7
F_PR22	Create Best Paper Award	Web	Functional	Main	F_PM10
F_PR23	Set Minimum and Maximum number of Reviews per Paper	Web	Functional	Main	
F_PR24	Set Maximum number of Papers each Reviewer can Review	Web	Functional	Main	
F_PR25	Accept/Reject Review of Paper	Web	Functional	Core	F_PR21
F_PR26	Multiple Reviewers for a paper	Web	Functional	Main	F_U3, F_U7
F_E1	Create an Event	Web	Functional		F_U3, F_U7
F_E2	Upload Poster	Web	Functional		
F_V1	Build a Conference Room	Web, IOS	Functional	Dream	~F_V15
F_V2	Drag and Drop for Utilities	Web, IOS	Functional	Dream	F_V1
F_V3	Drag and Drop for Seat	Web, IOS	Functional	Dream	F_V1

	Location				
F_V4	Select Number of Seats in X and Y Axis	Web	Functional	Dream	F_V1
F_V5	Fan-Out option for Seats	Web	Functional	Dream	F_V1
F_V6	Display Unavailable Seats	Web, IOS	Functional	Dream	F_V1
F_V7	Seat Types	Web, IOS	Functional	Main	
F_V8	Seat Classes	Web, IOS	Functional	Main	
F_V9	Selecting Seats for Types	Web, IOS	Functional	Main	F_V7
F_V10	Selecting Seats for Classes	Web, IOS	Functional	Main	F_V8
F_V11	Automatic Legend Generation	Web, IOS	Functional	Dream	F_V9, F_V10
F_V12	Seating Release Conditions	Web, IOS	Functional	Dream	F_V8
F_V13	Ability to Draw Venue with Line Tools	Web, IOS	Functional	Dream	F_V1
F_V14	Ability to Add Labels	Web, IOS	Functional	Dream	F_V1
F_V15	3D Venue Builder	Web	Functional	Dream*	
F_V16	Virtual Walk through	Web	Functional	Dream*	F_V15
F_V17	Selecting, Moving and Modifying Objects	Web, IOS	Functional	Dream	F_V2, F_V3
F_V18	Booking Accommodation for Guests	Web	Functional	Stretch	
F_V19	Pricing options for Accommodation	Web	Functional	Stretch	
F_V20	Upload Birds-Eye-View Image of Venue	Web, IOS	Functional	Stretch	
F_L1	Book Accommodation for Guests	Web	Functional	Stretch	
F_L2	Room Selection for Speakers	Web	Functional	Main	F_U9
F_L3	Create a presentation	Web	Functional	Core	F_U9
F_L4	Determine Available Rooms for Presenters	Web	Functional	Main	
F_L5	Filter Criteria for Room Booking	Web	Functional	Stretch	

F_L6	Timetable Generation	Web	Functional	Main	F_L2, F_L3
F_L7	User Specific Meal Preferences	IOS, Web	Functional	Stretch	
F_L8	Video Conference Capabilities	Web	Functional	Dream	
F_L9	Ability to have Private or Public Events	Web	Functional	Main	F_E1
F_L10	Ability to have Private or Public Presentations/Performances	Web	Functional	Stretch	F_E1
F_L11	Automatically generate Visa Letters	Web	Functional	Stretch	
F_L12	Ability to have a presenter that is not an author of a paper	Web	Functional	Main	
F_L13	Ability to have backup presenters	Web	Functional	Main	F_L12
F_L14	Ability to transfer access to a event to new organisers	Web	Functional	Dream	F_E1, F_U7
F_L15	Ability to copy event data as a template for others user	Web	Functional	Stretch	F_E1
F_L16	Ability to publish conference/event proceedings before and after the event	Web	Functional	Stretch	F_U1, F_E1, F_PM1, F_PM2, F_PM3, F_PM4
F_TP1	User purchase ticket	IOS, Web	Functional	Core	F_TP9
F_TP2	Interface with external payment system	Web	Functional	Main	
F_TP3	Select seat type and class of seat	IOS, Web	Functional	Stretch	
F_TP4	Viewing birds eye view of the event system	Web	Functional	Main	F_V20
F_TP5	Optional accommodation for guests	Web	Functional	Stretch	
F_TP6	Managers create categories of tickets (other than general admission)	Web	Functional	Main	

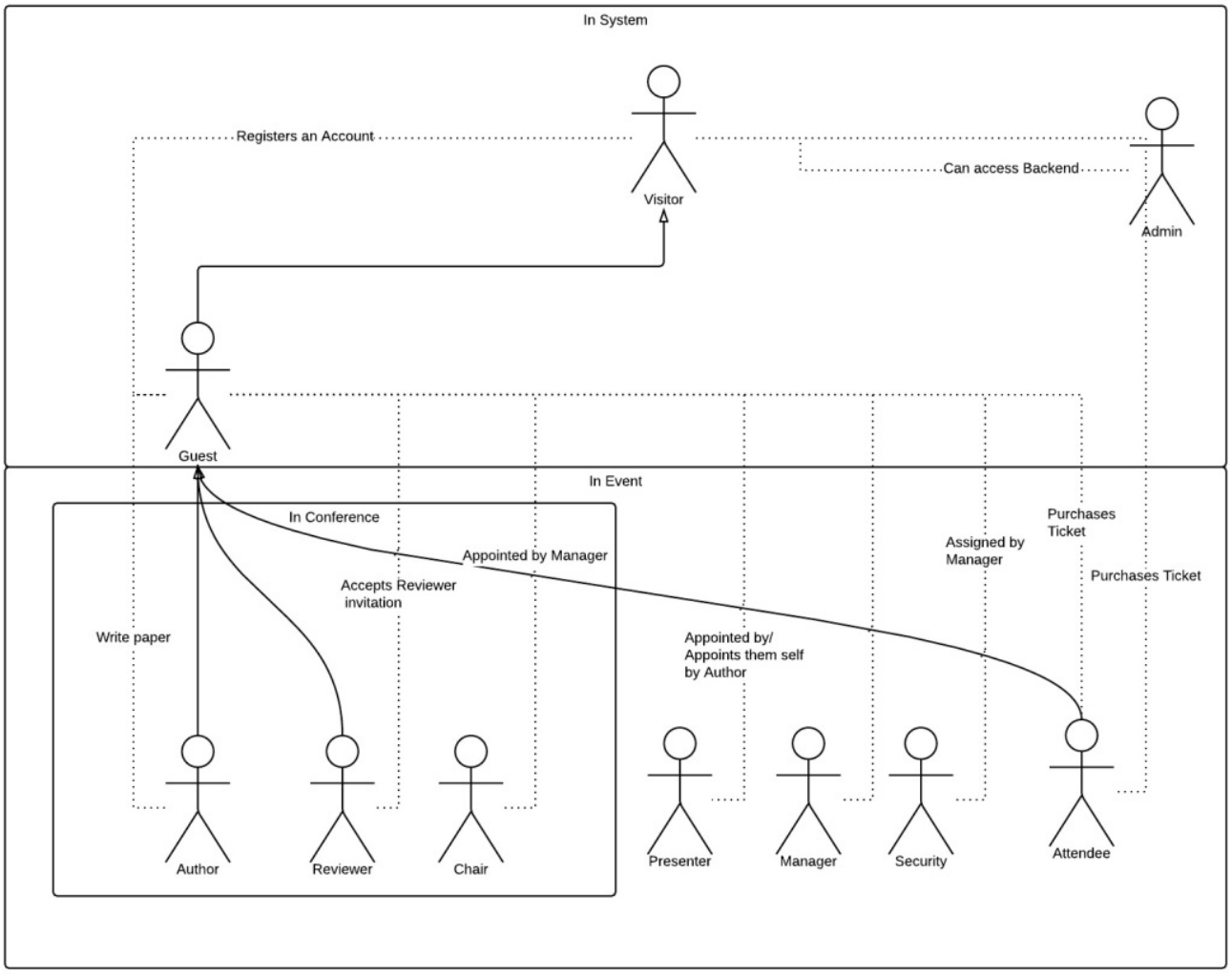
F_TP7	Attendees can select a ticket type for purchase	IOS, Web	Functional	Main	F_TP6
F_TP8	Email notification upon ticket purchase	IOS, Web	Functional	Main	F_TP1, F_TP2
F_TP9	Book ticket for multiple people if public event	IOS, Web	Functional	Main	F_TP1
F_TP10	Purchase ticket for multiple events	IOS, Web	Functional	Core	F_TP1
F_TP11	User makes payment	IOS, Web	Functional	Core	F_TP1, F_TP2
F_TP12	Different payment option	IOS, Web	Functional	Core	F_TP2
F_TP13	Option to limit payment method	Web	Functional	Main	
F_TP14	Payment log to keep track of transaction record	IOS, Web	Functional	Core	F_TP2
F_TP15	Ability to view payment log	IOS, Web	Functional	Core	F_TP14
F_TP16	Waiting list	Web	Functional	Stretch	
F_TP17	Ability to generate QR codes for tickets	IOS	Functional	Stretch	F_TP1, F_TP2
F_TP18	Ability to save tickets in iOS Passport	IOS	Functional	Stretch	F_TP1, F_TP2
F_A1	Generate financial statements	Web	Functional	Stretch	
F_A2	Generate statistical reports	Web	Functional	Dream	F_A3, F_A4, F_A5
F_A3	Number of attendees per event type	Web	Functional	Stretch	
F_A4	Satisfaction of Attendees per event type	Web	Functional	Stretch	F_A2
F_A5	Number of purchased tickets for each category of ticket	Web	Functional	Stretch	F_A2
F_A6	Generate budget reports	Web	Functional	Stretch	
F_A7	Ability to have sponsors	Web	Functional	Main	
F_A8	Ability to upload advertisement data	Web	Functional	Main	

F_A9	Viewing purchased and available seats with Statistics (birds-eye)	Web	Functional	Stretch	F_V20
F_A10	Viewing Birds-Eye-View of the event	Web	Functional	Dream	F_V20
F_A11	Displaying Financial Statements	Web	Functional	Stretch	F_A1
F_A12	Displaying Statistical Reports	Web	Functional	Dream	F_A2
F_IH1	Optional Surveys	Web	Functional	Main	F_E1
F_IH2	Security Help	IOS	Functional	Main	
F_IH3	Comment on Event	IOS, Web	Functional	Core	F_E1
F_IH4	Submit Questions for a talk	IOS, Web	Functional	Core	F_L3
F_IH5	Speaker can ask question and attendees can answer	IOS, Web	Functional	Stretch	F_L3
F_IH6	Generation of name badges	Web	Functional	Stretch	
F_IH7	Download list of Attendee Names	Web	Functional	Core	F_IH6
F_IH8	Social-Media communication with attendees	IOS	Functional	Dream	

* – a requirement that is slightly above its classification but not high enough to create a new classification

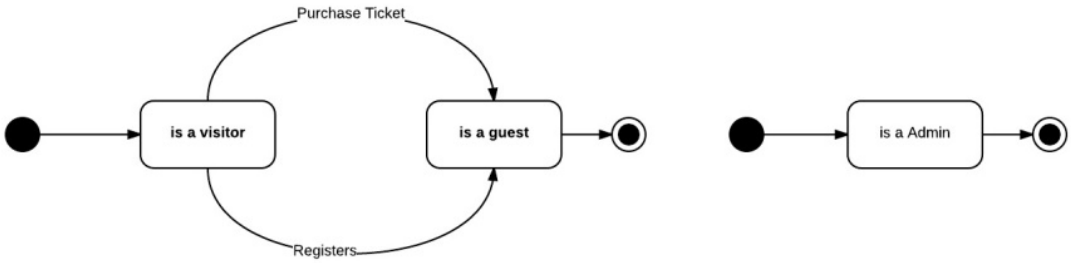
~ – a weak dependency. The dependency is not needed to fulfill the requirement, but having the dependency will improve the functionality of the requirement.

This diagram shows the inheritance between the types of users in the system. The dotted lines represent how the the user is upgraded in the system or in the event



These are the states in the system as user is either a Visitor, a Guest or an Admin

Authors, Reviewers, Chair, Presenters, Managers, Security and Attendees are all still guests in the system however their state changes for the event



USER MANAGEMENT SUBSYSTEM

Name: Create profiles		
Requirement #: F_U1	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow users to create a profile in the Conference Organiser system.		
Rationale: Users require a profile to use the Event Organiser.		
Source: Client		
Fit Criterion: Users can successfully create a profile to access the system.		
Dependencies: Refer to F_U3 for the details that a profile will contain.		
Classification: Core		
History: Created by Michael Kong on 30/Jul/2015		

Name: Remove Profile		
Requirement #: F_U2	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow users to remove an existing profile. Users will be logged out immediately following the removal of their profile.		
Rationale: Users may want to remove their profile from the system.		
Source: Client –		
Fit Criterion: Users successfully remove their profile from the system.		
Dependencies: • User must have an existing profile. Refer to F_U1. • User must be logged into the system. Refer to F_U5.		
Classification: Core		
History: Created by Michael Kong on 31/Jul/2015		

Name: Profile Details		
Requirement #: F_U3	**Requirement Type:** Functional	**Use Case ID**:
Description: All profile categories shall have the following common details: **Required details:** ● Title (Mr/Ms/Dr, associate professor, professor) - we will give a large list ● Name ● Date of Birth ● Email Address (unique identifier) ● Password ● Address (street, state, city, country) **Optional details:** Author and delegate profiles have the following additional details: ● Credit Card (issuer, number) Author profiles have the following optional details: ● Biographies ● Photos ● Link to web page		
Rationale: System should store profile details.		
Source: Client –		
Fit Criterion: Each profile of a particular profile category shall have its respective details.		
Dependencies: A single set of profile details can only exist for one profile. Refer to F_U1		
Classification: Core		
History: Created by Michael Kong on 30/Jul/2015		

Name: Update Profile		
Requirement #: F_U4	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow users to modify their profiles.		
Rationale: Users may want to update or change their details.		
Source: Client –		
Fit Criterion: Users can successfully modify their profile.		
Dependencies: User must have an existing profile. Refer to F_U1.		
Classification: Core		
History: Created by Michael Kong on 31/Jul/2015		

Name: Unique ID & Password		
Requirement #: F_U5	**Requirement Type:** Functional	**Use Case ID**:
Description: The system should allow all users to login using a unique identifier and password.		
Rationale: Users need to login to use the Event Organiser.		
Source: Client –		
Fit Criterion: Users successfully login to the system after entering the a unique identifier and the correct password associated with the unique identifier.		
Dependencies: User must have an existing profile. Refer to F_U1.		
Classification: Core		
History: Created by Michael Kong on 31/Jul/2015		

Name: Change and reset Password		
Requirement #: F_U6	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow all users to change their password.		
Rationale: Users may want to update or change their password.		
Source: Client – Hoa Dam		
Fit Criterion: Users can successfully change their password.		
Dependencies: ● User must have an existing profile. Refer to F_U1 ● User must be logged into the system. Refer to F_U5		
Classification: Core		
History: Created by Michael Kong on 31/Jul/2015		

Name: Pay to upgrade account		
Requirement #: F_U7	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow users to upgrade to a paid account. This allows them to become a manager of an event.		
Rationale: Users want to create their own events. Businesses would like to generate money from the system.		
Source: Client		
Fit Criterion: Users can successfully pay to upgrade their account.		
Dependencies: User must have an existing profile. Refer to F_U1		
Classification: Core		
History: Created by Sandon Joubert on 11/09/2015		

Name: Password Encryption		
Requirement #: F_U8	**Requirement Type:** Functional	**Use Case ID**:
Description: User password shall be securely stored in the system using salting and hashing techniques.		
Rationale: Passwords and Credit Card information are very sensitive and should be encrypted to thwart hacking attempts.		
Source: Matt		
Fit Criterion: User's password and Credit Card Number is hashed and salted before it is stored in the database.		
Dependencies: None		
Classification: Main		
History: Created by Matthew Boroczky on 24/8/15		

Name: Create Conference		
Requirement #: F_U9	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow a user with Event Organiser privileges to create conferences		
Rationale: User wants to create a conference.		
Source: Client –		
Fit Criterion: User successfully creates a new conference in the system.		
Dependencies: • User must have an existing profile. Refer to F_U3. • User must have an upgraded account. Refer to F_U7.		
Classification: Core		
History: Created by Liew Loo Yee 11/9/2015		

Name: Enable paper submission for event		
Requirement #: F_U10	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow a user to submit paper for the event.		
Rationale: User wants to submit paper for presentation, as reference, handouts etc.		
Source: Client –		
Fit Criterion: User successfully submit paper for the event.		
Dependencies: • User must have an existing profile. Refer to F_U3. • User must have an upgraded account. Refer to F_U7.		
Classification: Core		
History: Created by Liew Loo Yee 11/9/2015		

Name: Create email		
Requirement #: F_U11	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall automatically create an email address using the domain name of our website for users with Event Organiser privileges.		
Rationale: Managers and organisers can protect their personal email accounts by using emails that are specific to our domain. The system should allow only one email per profile having Event Organiser privileges.		
Source: Matt		
Fit Criterion: Profiles with Event Organiser privileges are able to use an email address that is specific to the system's website's domain name.		
Dependencies: • User must have an existing profile. Refer to F_U3. • User must have an upgraded account. Refer to F_U7.		
Classification: Main		
History: Created by Matthew Boroczky on 24/8/15		

Name: Event Reminders		
Requirement #: F_U12	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall send an email or notification to remind attendees/presenters of upcoming events.		
Rationale: Users want to avoid missing events.		
Source: Matt		
Fit Criterion: Event attendees and presenters receive an email or notification at a set time before the event.		
Dependencies: • User must have an existing profile. Refer to F_U3.		
Classification: Main		
History: Created by Matthew Boroczky on 24/8/15		

Name: View Presenters and Presentation Times		
Requirement #: F_U13	**Requirement Type:** Functional	**Use Case ID**:
Description: Guests shall be able to view lists of presenters and timetables of presentations for events.		
Rationale: Event attendees may wish to check times of presentations and presenters that interest them. They may also wish to check for clashes between different presentations.		
Source: Matt		
Fit Criterion: A timetable of presentations and a list of presenters is generated by the system for event attendees that display scheduled presenters, topics, presentation abstracts and presentation times.		
Dependencies: • User must have an existing profile. Refer to F_U3.		
Classification: Stretch		
History: Created by Matthew Boroczky on 24/8/15		

Name: Multilingual support		
Requirement #: F_U14	**Requirement Type:** Functional	**Use Case ID**:
Description: Event Organisers shall have the option to provide direct translations for English words in the system. They specify the language from a list of available languages and the translations are saved into the database. If another translation is provided for a word or clause, the Event Organiser adding the new translation will be asked to select the better translation.		
Rationale: Providing multilingual support for Event Organisers helps users that are not proficient in English to use the system easily. Shifting the responsibility of creating and correcting translations to the Event Organisers relieve the need for professional translations on the developers' side.		
Source: All		
Fit Criterion: Event Organisers are able to input translations and compare translations if there are existing translations in the system.		
Dependencies: • User must have an existing profile. Refer to F_U3. • User must have an upgraded account. Refer to F_U7. • Database must be able to store Unicode characters. Refer to F_U16.		
Classification: Dream		
History: Created by Matthew Boroczky on 24/8/15		

Name: Default Mandarin Language Support		
Requirement #: F_U15	**Requirement Type:** Functional	**Use Case ID**:
Description: Admin and guests shall be able to view all functionality in both English and Mandarin		
Rationale: Two common languages in the Southern Hemisphere are English and Mandarin. Supporting these languages by default broadens our target market to Mandarin speaking countries such as China, Malaysia and Singapore.		
Source: All		
Fit Criterion: Default Mandarin translation for the system is provided.		
Dependencies: • User must have an existing profile. Refer to F_U3.		
Classification: Stretch		
History: Created by Matthew Boroczky on 24/8/15		

Name: Database type		
Requirement #: F_U16	**Requirement Type:** Functional	**Use Case ID**:
Description: Database shall have the ability to store numerous Unicode characters allowing all language characters to be stored into the database for names, addresses, event name as well as direct and default translation.		
Rationale: The database should have the ability to store Unicode characters to cater to the multilingual support of the system.		
Source: All		
Fit Criterion: The database can store Unicode characters and retrieve them correctly.		
Dependencies: None		
Classification: Dream		
History: Created by Matthew Boroczky on 24/8/15		

Name: Facebook integration		
Requirement #: F_U17	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow users to integrate their profile with their Facebook account.		
Rationale: Users want the added benefits of the system having access to their Facebook profile.		
Source: Client		
Fit Criterion: Users can successfully integrate their profile with Facebook.		
Dependencies: • User must have an existing profile. Refer to F_U3		
Classification: Stretch		
History: Created by Sandon Joubert on 11/09/2015		

Name: Login via Facebook		
Requirement #: F_U18	**Requirement Type:** Functional	**Use Case ID:**
Description: Users can log into the site using their Facebook account.		
Rationale: Flexibility in sign up forms is a crucial part of web design. Users who like to keep all their accounts bound together now have the option to do so.		
Source: CY		
Fit Criterion: A user may use their Facebook credentials to log in to the site		
Dependencies: None		
Classification: Stretch		
History: Created by Sandon Joubert 5/9/2015		

Name: Invite Friends to Event via Facebook		
Requirement #: F_U19	**Requirement Type:** Functional	**Use Case ID:**
Description: Users may be able to invite friends to the event semi-automatically via Facebook API		
Rationale: The social side of events are very important, by allowing an easy way for users to invite people whom they know it encourages users to do so.		
Source: CY		
Fit Criterion: A user can select friends retrieved from their friend list on Facebook and send an invitation for events.		
Dependencies: Users must have connected their profile to their Facebook account. Refer to F_U17		
Classification: Stretch		
History: Created by Sandon Joubert 5/9/2015		

Name: LinkedIn integration		
Requirement #: F_U20	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow users to integrate their profile with their LinkedIn account.		
Rationale: Users want the added benefits of the system having access to their LinkedIn profile.		
Source: Client		
Fit Criterion: Users can successfully integrate their profile with LinkedIn.		
Dependencies: • User must have an existing profile. Refer to F_U3		
Classification: Stretch		
History: Created by Sandon Joubert on 11/09/2015		

Name: Invite Friends to Event via LinkedIn		
Requirement #: F_U21	**Requirement Type:** Functional	**Use Case ID:**
Description: Users may be able to invite friends to the event semi-automatically via LinkedIn API		
Rationale: The social side of events are very important, by allowing an easy way for users to invite people whom they know it encourages users to do so.		
Source: CY		
Fit Criterion: A user may select friends automatically generated from LinkedIn to send an invitation to for the event		
Dependencies: Users must have connected their profile to their LinkedIn account. Refer to F_U20		
Classification: Stretch		
History: Created by Sandon Joubert 5/9/2015		

Name: Invite Friends to Event via Contacts (phone, email)		
Requirement #: F_U22	**Requirement Type:** Functional	**Use Case ID:**
Description: Users may be able to invite friends to the event semi-automatically via their phone or email contacts		
Rationale: Users want to be able to invite their phone contacts to events.		
Source: Matt		
Fit Criterion: A user may select friends retrieved from their contacts to send an invitation to for the event.		
Dependencies: • User must have an existing profile. Refer to F_U3		
Classification: Stretch		
History: Created by Sandon Joubert 5/9/2015		

Name: Match Accounts to Paper Contribution		
Requirement #: F_PM7	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall be able to match existing authors who have contributed to submitted papers automatically. The system shall match the respective user accounts to the submitted papers by their email address.		
Rationale: Authors of submitted papers want their profiles associated with their paper. Delegates reading papers in the system want to view the profile of the author(s).		
Source: Client		
Fit Criterion: System successfully matches authors with their papers. Delegates can access the profiles of the author(s) through papers.		
Dependencies: A paper must be submitted to the system. Refer to F_PM5		
Classification: Main		
History: Created by Michael Kong on 02/Aug/2015		

Name: Submit Paper		
Requirement #: F_PM5	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow users using the Author profile to submit papers to the system. During submission of papers, users shall fill in a form pertaining to the details of the paper and the author(s). Information of author(s) includes: a. First Name b. Last Name c. Email d. University Information of paper includes:		

a. Title b. Abstract c. 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: Authors want to submit papers to the system.
Source: Client
Fit Criterion: Users successfully submit papers to the system.
Dependencies: The system has to allow the uploading of documents. Refer to F_PM1, F_PM2, F_PM3, F_PM4
Classification: Core
History: Created by Michael Kong on 02/Aug/2015

Name: Modify Submission		
Requirement #: F_PM6	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow users using the Author profile to modify their submissions. Modifications include changing author(s) details and the paper submission. Paper submission must be done before the submission deadline set by the conference organiser.		
Rationale: Authors want to modify their submissions.		
Source: Client		
Fit Criterion: Users successfully modify their submissions.		
Dependencies: A paper must be submitted to the system. Refer to F_PM5		
Classification: Core		
History: Created by Michael Kong on 02/Aug/2015		

Name: Setup Categories and Subcategories for Profession		
Requirement #: F_PM10	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow Event Organisers/Managers to set up categories and subcategories for presentations and papers to allow sorting by categories and subcategories.		
Rationale: By implementing categories and subcategories, the system will provide a level of distinction which is needed for conference management statistics, conference management with multiple presentations on different topics and will aid in the paper award process. I.e. category Computer Science and Exercise Science should not be compared, Subcategories of Computer Security and Software Engineering should not be compared for subcategory awards but may be compared for Computer Science Awards		
Source: Michael		
Fit Criterion: Conference organisers can successfully add categories to the conference and subcategories. Based off this a paper can be assigned a category or subcategory and thus a presentation		
Dependencies: None		
Classification: Main		
History: Created by Matthew Boroczky 4/9/2015		

Name: Allow ZIP files to be uploaded		
Requirement #: F_PM4	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow ZIP files containing papers and items related to papers, to be uploaded		
Rationale: Certain papers such as Software Engineering papers includes references to source files that should be included and may be needed in order to review a paper		
Source: Client		
Fit Criterion: A author can upload a zip file which is successfully stored on the server and can be downloaded		
Dependencies: None		
Classification: Main		
History: Created by Matthew Boroczky 4/9/2015		

Name: Allow Word Documents to be Uploaded		
Requirement #: F_PM3	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow authors to upload their papers in .doc and .docx formats		
Rationale: Many papers are in Word Document format. The system should support the ability to upload Word Documents for this reason.		
Source: Client		
Fit Criterion: A author can upload one or more .doc or .docx files which will be successfully stored on the server and can be downloaded		
Dependencies: None		
Classification: Main		
History: Created by Matthew Boroczky 4/9/2015		

Name: Allow LaTeX Documents to be Uploaded		
Requirement #: F_PM2	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow authors to upload their papers in LaTeX format		
Rationale: Academic papers are generally written using Latex (e.g. Computer Science papers). As the system is also a conference management system which supports the reviewing of papers it should have the ability to store LaTeX files		
Source: Client		
Fit Criterion: A author can upload one or more LaTeX files which will be successfully stored on the server and can be downloaded		
Dependencies: None		
Classification: Main		
History: Created by Matthew Boroczky 4/9/2015		

Name: Allow PDF Documents to be Uploaded		
Requirement #: F_PM1	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow authors to upload their papers in PDF format		
Rationale: Academic papers and general papers are usually published in PDF format. Therefore it is crucial that the system at the minimum has the ability to store PDF papers.		
Source: Client		
Fit Criterion: A author can upload one or more PDF files which will be successfully stored on the server and can be downloaded		
Dependencies: None		
Classification: Core		
History: Created by Matthew Boroczky 4/9/2015		

Name: Allow manager to Restrict Upload Types		
Requirement #: F_PM9	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow Event Organisers to choose what types of files that users are allowed to upload.		
Rationale: Certain Conferences or Events may not support Latex for example and may only wish to allows users to upload PDF documents		
Source: Matt		
Fit Criterion: A Administrator selects the files that are allowed to be uploaded and when a user goes to upload a file they allowed files are uploaded and stored on the server and a error is given if the user attempts to upload a invalid file format		
Dependencies: The system must allow file uploading. Refer to F_PM1, F_PM2, F_PM3, F_PM4		
Classification: Main		
History: Created by Matthew Boroczky 4/9/2015		

Name: Allow Automatic Detection of Paper Contributors		
Requirement #: F_PM8	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall be able to identify the contributing authors of a paper on LaTeX documents.		
Rationale: By automating this process it makes it easier on the user however if the user types in the wrong name, or the reading of the latex file is not correct, this will cause difficulties and for this reason the user should be able to manually override the value		
Source: Matt		
Fit Criterion: User uploads a LaTeX file and the system displays the authors in a textfield that allows them to edit the authors names if they are incorrect		
Dependencies: The system must allow users to upload LaTeX documents. Refer to F_PM2		
Classification: Stretch		
History: Created by Matthew Boroczky 4/9/2015		

Name: Allow Automatic Detection of Paper Abstract		
Requirement #: F_PM11	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall be able to automatically retrieve the abstract from the paper. Abstracts will be taken from the Abstract tab on papers in LaTeX format, and. the first 300 words of Word documents or PDFs. Paragraphs with more than 300 words will be appended with ellipsis.		
Rationale: To simplify the paper uploading process of the user.		
Source: Matt		
Fit Criterion: User uploads a Document and the system displays the abstract in a textbox that allows them to edit the abstract if it is incorrect		
Dependencies: The system must allow file uploading. Refer to F_PM1, F_PM2, F_PM3, F_PM4		
Classification: Stretch		
History: Created by Matthew Boroczky 4/9/2015		

Name: Set limit on file sizes		
Requirement #: F_PM12	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the user to set the size limit of uploaded files.		
Rationale: Large files contributes to longer processing time and takes up space, may lead to congestion and inconvenience to organizers while dealing with files.		
Source: Client		
Fit Criterion: Display limit of file size when user wants to upload file, or display error message when a file exceeding size limit is being uploaded.		
Dependencies: The system must allow file uploading. Refer to F_PM1, F_PM2, F_PM3, F_PM4		
Classification: Stretch		
History: Created by Liew Loo Yee 11/9/2015		

PAPER REVIEW SUBSYSTEM

Name: View the reviewer list		
Requirement #: F_PR14	**Requirement Type:** Functional	
Description: The system shall provide a GUI to list all the reviewers and their expertise.		
Rationale: A chair wants to view the reviewer list.		
Source: Chair		
Fit Criterion: A GUI catalogue with different expertise skill should be displayed.		
Dependencies: User must have an existing profile. Refer to F_U3.		
Classification: Core		
History: Created by Ruixi He 02/9/2015		

Name: Assign Paper to Reviewer		
Requirement #: F_PR1	**Requirement Type:** Functional	
Description: The system shall be able to assign a paper to a reviewer.		
Rationale: A reviewer needs to be notified and be able to access the paper		
Source: Chair		
Fit Criterion: Reviewer receives notification and can access paper		
Dependencies: User must have an existing profile. Refer to F_U3.		
Classification: Core		
History: Created by Matthew Boroczky 11/09/2015		

Name: Add reviewer to the reviewer list		
Requirement #: F_PR15	**Requirement Type:** Functional	**Dependant:**
Description: The system shall provide the Chair a GUI to add new reviewers to the list and also classify their expertise.		
Rationale: A chair wants to add a new reviewer to the list by classify his/her expertise.		
Source: Chair		
Fit Criterion: A new reviewer can be added successful into the particular category.		
Dependencies: The Chair should be able to view a list of reviewers. Refer to F_PR14.		
Classification: Core		
History: Created by Ruixi He 02/9/2015		

Name: Suggest reviewer based on LinkedIn's skill		
Requirement #: F_PR19	**Requirement Type:** Functional	**Dependant:**
Description: The system shall provide an option to automatically suggest reviewers based on the reviewers' skill descripted in LinkedIn.		
Rationale: A chair wants add reviewer to the reviewer list, and the system can provide an option to suggest one by the skill described in LinkedIn.		
Source: Chair		
Fit Criterion: An GUI option about whether add this people to the list by the skill of the LinkedIn.		
Dependencies: Reviewer should have a LinkedIn profile. Refer to F_U20		
Classification: Dream		
History: Created by Ruixi He 02/9/2015		

Name: Delete reviewer from the reviewer list		
Requirement #: F_PR3	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide the Chair a GUI to delete an existing reviewer from the list.		
Rationale: A chair wants to delete an exist reviewer from the list.		
Source: Chair		
Fit Criterion: An existing reviewer can be deleted successfully from the list.		
Dependencies: Reviewer must be added to the list before being deleted. Refer to F_PR15		
Classification: Core		
History: Created by Ruixi He 02/9/2015		

Name: Assign Multiple Papers to a Reviewer		
Requirement #: F_PR2	**Requirement Type:** Functional	**Dependant:**
Description: The system shall allow multiple papers to be assigned to a single reviewer.		
Rationale: A Reviewer can review more than one paper		
Source: Client		
Fit Criterion: A reviewer should be able to review more than one paper		
Dependencies: User must have an existing profile. Refer to F_U3		
Classification: Core		
History: Created by Matthew Boroczky 11/09/2015		

Name: Select reviewer preferences		
Requirement #: F_PR11	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide reviewers a GUI to select and change he/her preferences (Chemistry, Math, Philosophy, etc) to categorise the expert.		
Rationale: A reviewer wants to select or change he/ her preferences.		
Source: Reviewer		
Fit Criterion: The preference can be selected successfully.		
Dependencies: The reviewer must have an upgraded account. Refer to F_U7		
Classification: Core		
History: Created by Ruixi He 07/9/2015		

Name: Send request to assign papers to reviewers		
Requirement #: F_PR12	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide the Chair with a GUI to send request to the reviewer to assign paper to him/her on the reviewer list by expertise.		
Rationale: A chair wants to assign a paper to a corresponding expertise reviewer from the reviewer list.		
Source: Chair		
Fit Criterion: A assigning request can be sent successfully to a corresponding reviewer.		
Dependencies: Users must have an upgraded account. Refer to F_U7		
Classification: Core		
History: ● Created by Ruixi He 02/9/2015 ● Edited by Ruixi He 07/9/2015		

Name: Accept/Reject the reviewing of a paper		
Requirement #: F_PR8	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow the reviewer to accept or reject a paper that was assigned to them for review.		
Rationale: The reviewer may get a paper that they do not wish to review for any reason.		
Source: Client		
Fit Criterion: Reviewer can accept or reject reviewing papers assigned to them		
Dependencies: Papers must be assigned to reviewers. Refer to F_PR1		
Classification: Main		
History: Created by Sandon Joubert 11/09/2015		

Name: Check COI manually		
Requirement #: F_PR16	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide reviewers a GUI to check the conflict of interest in a manual way.		
Rationale: The system display a GUI to let the reviewer to confirm whether has COI of the paper.		
Source: Reviewer		
Fit Criterion: A paper can be confirmed by the reviewer by using the GUI of the system.		
Dependencies: Papers must be assigned to reviewers. Refer to F_PR1		
Classification: Core		
History: Created by Ruixi He 07/9/2015		

Name: Check COI through LinkedIn and Facebook		
Requirement #: F_PR17	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide a function to check the COI through LinkedIn and/or Facebook when the chair send an assign paper request to a reviewer.		
Rationale: The system can check the COI automatically through LinkedIn and/or Facebook when a chair send a request to a reviewer to check whether the paper author is a friend of the reviewer.		
Source:		
Fit Criterion: A alert message can be displayed when there is a conflict by the checking of LinkedIn and/ or Facebook.		
Dependencies: • Paper must be assigned to reviewers. Refer to F_PR1 • User must be logged on to Facebook. Refer to F_U17 • User must be logged on to LinkedIn. Refer to F_U20		
Classification: Stretch		
History: Created by Ruixi He 07/9/2015		

Name: Reviewer accept/reject the assign paper request		
Requirement #: F_PR13	**Requirement Type:** Functional	**Use Case ID:**
Description: The system should provide a reviewer with a GUI to accept/reject request from the chair.		
Rationale: A reviewer decide to accept or reject the assign paper request send by the chair.		
Source: Reviewer		
Fit Criterion: A request can be accept or reject successfully by reviewer.		
Dependencies: The reviewer must be sent a request to review paper. Refer to F_PR12		
Classification: Core		
History: ● Created by Ruixi He 02/9/2015 ● Edited by Ruixi He 07/9/2015		

Name: Distribute papers to reviewers		
Requirement #: F_PR4	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide a function to distribute paper to the reviewer has assigned by chair automatically.		
Rationale: The system can distribute paper to a reviewer automatically who has already been assigned the paper to by a chair.		
Source:		
Fit Criterion: A paper can be distributed successfully to the reviewer by the system.		
Dependencies: • User must have an existing profile. Refer to F_U3. • User must have an upgraded account. Refer to F_U7.		
Classification: Core		
History: Created by Ruixi He 07/9/2015		

Name: Comment on papers – Reviewer		
Requirement #: F_PR21	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide reviewers with a GUI to make comment on paper after reviewing the paper.		
Rationale: A reviewer wants to make comment on the paper.		
Source: Reviewer		
Fit Criterion: Comment can be successfully make be reviewer.		
Dependencies: • User must have an existing profile. Refer to F_U3. • User must have an upgraded account. Refer to F_U7.		
Classification: Core		
History: ● Created by Ruixi He 02/9/2015 ● Edited by Ruixi He 07/9/2015		

Name: Scaling the reviewed papers		
Requirement #: F_PR7	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide reviewers with a GUI to scale a rank to the paper been reviewed.		
Rationale: A reviewer wants to scale a rank to the reviewed paper.		
Source: Reviewer		
Fit Criterion: Comment can be successfully make by reviewer.		
Dependencies: Functionality for paper reviewing should exist. Refer to F_PR5		
Classification: Main		
History: Created by Ruixi He 07/9/2015		

Name: Accept/reject the reviewed paper		
Requirement #: F_PR25	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide the chair with a GUI to accept or reject the paper after the chair read the comments of the paper provided by the reviewer.		
Rationale: A chair wants to accept or reject the paper after read the comments by the reviewer.		
Source: Chair		
Fit Criterion: The paper can be accept or reject successfully.		
Dependencies: The paper must be commented. Refer to F_PR21.		
Classification: Core		
History: ● Created by Ruixi He 02/9/2015 ● Edited by Ruixi He 07/9/2015		

Name: Communication between paper author and reviewer		
Requirement #: F_PR10	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide a GUI for paper author and reviewer to communicate. During the communication, the chair will act as an intermediary between reviewer and paper author.		
Rationale: A paper author and the reviewer of that paper wants to communicate to each other, and chair act as an intermediary.		
Source: Chair, Author, Reviewer		
Fit Criterion: The GUI for author and reviewer to communicate should be displayed.		
Dependencies: • User must have an existing profile. Refer to F_U3. • User must have an upgraded account. Refer to F_U7.		
Classification: Main		
History: ● Created by Ruixi He 02/9/2015 ● Edited by Ruixi He 07/9/2015		

Name: Create best paper award		
Requirement #: F_PR22	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide the chair with a GUI to nominate the best paper award by the scaling of the reviewed paper.		
Rationale: A chair wants to nominate the best paper award from the reviewed paper.		
Source: Chair		
Fit Criterion: A paper can be nominated the best paper award successfully.		
Dependencies: Categories and subcategories must be set up for paper categorization. Refer to F_PM10		
Classification: Main		
History: ● Created by Ruixi He 02/9/2015 ● Edited by Ruixi He 07/9/2015		

Name: Ability to send paper link to friend for peer-review prior to submitting to reviewer		
Requirement #: F_PR20	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide paper uploading functions and a unique link to authors. Authors can send this link to friends for peer review of the uploaded paper.		
Rationale:		
Source: Sandon		
Fit Criterion: On paper upload the author can receive a permalink to the paper.		
Dependencies: User must be able to upload papers. Refer to F_PM1, F_ PM2, F_ PM3, F_PM4		
Classification: Stretch		
History: Created by Sandon Joubert 5/9/2015		

Name: Multiple Reviewers for a paper		
Requirement #: F_PR26	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the PC Chair to allow more than one reviewer per paper.		
Rationale: Many conference management systems allow this functionality to create greater impartiality on the review.		
Source: Sandon		
Fit Criterion: The PC chair can select a natural number from an input field, in the conference settings corresponding to the amount of reviewers per paper.		
Dependencies: • User must have an existing profile. Refer to F_U3. • User must have an upgraded account. Refer to F_U7.		
Classification: Main		
History: Created by Sandon Joubert 5/9/2015		

Name: Reviewer Anonymous – Blind Review		
Requirement #: F_PR5	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the reviewer is kept anonymous to the author(s).		
Rationale: It is important to remove any possible factors resulting in non-impartial reviews.		
Source: Sandon		
Fit Criterion: An author(s) who uploads their paper for review will simply see the reviewer as 'reviewer' if there are multiple it would be represented with a letter or number appended as a suffix to distinguish between multiple.		
Dependencies: Papers must be distributed to reviewers first. Refer to F_PR4		
Classification: Core		
History: Created by Sandon Joubert 5/9/2015		

Name: Reviewer and Author Anonymous – Double-Blind Review		
Requirement #: F_PR6	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the reviewer is kept anonymous to the author(s) and the author(s) are kept anonymous to the reviewer.		
Rationale: It is important to remove any possible factors resulting in non-impartial reviews.		
Source: Sandon		
Fit Criterion: An author(s) who uploads their paper for review will simply see the reviewer as 'reviewer' if there are multiple it would be represented with a letter or number appended as a suffix to distinguish between multiple. The same is applied to author(s) for reviewers.		
Dependencies: Papers must be distributed to reviewers first. Refer to F_PR4		
Classification: Main		
History: Created by Sandon Joubert 5/9/2015		

Name: Automatically flag reviews with a difference that is too high		
Requirement #: F_PR9	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall flag reviews on a paper that are higher than a set value.		
Rationale: It is important to remove any possible factors resulting in non-impartial reviews.		
Source: Sandon		
Fit Criterion: Two reviewers submit scores for a paper differing by more than the number set by the PC Chair; the paper review is then flagged.		
Dependencies: A review on the paper must have been performed. Refer to F_PR5		
Classification: Main		
History: Created by Sandon Joubert 5/9/2015		

Name: Ability to customise the flag value of the review		
Requirement #: F_PR18	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the Chair to set a tolerance value for paper reviews. The default would be three (3).		
Rationale: It is important to remove any possible factors resulting in non-impartial reviews.		
Source: Sandon		
Fit Criterion: The PC Chair is presented with an option for how much scores may differ.		
Dependencies: None		
Classification: Main		
History: Created by Sandon Joubert 5/9/2015		

Name: Ability for chair to define maximum and minimum reviewers		
Requirement #: F_PR23	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the manager of events to set the number of reviewers for the event.		
Rationale: Manager would like to control how many reviewers are associated with the event		
Source: Sandon		
Fit Criterion: The PC Chair is presented with an option for a minimum and maximum number of reviewers which they can set.		
Dependencies: None		
Classification: Main		
History: Created by Sandon Joubert 5/9/2015		

Name: Ability to set a maximum amount of papers a reviewer may review		
Requirement #: P_PR24	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the event manager to set the maximum amount of papers each reviewer can review.		
Rationale: The event manager would like to not overload any particular reviewers.		
Source: Sandon		
Fit Criterion: The event manager is presented with an option to set maximum papers a reviewer can review.		
Dependencies: None		
Classification: Main		
History: Created by Sandon Joubert 5/9/2015		

EVENT CREATION SUBSYSTEM

Name: Create an Event		
Requirement #: F_E1	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow a user using a Conference Chair profile to create events.		
Rationale: User wants to create an event.		
Source: Client –		
Fit Criterion: User successfully creates a new event in the system.		
Dependencies: • User must have an existing profile. Refer to F_U3. • User must have an upgraded account. Refer to F_U7.		
Classification: Core		
History: Created by Michael Kong on 30/Jul/2015		

Name: Upload Poster		
Requirement #: F_E2	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow event organisers to upload posters for the event.		
Rationale: Allows the event to me more publicised and provides and marketing aspect to the actual event/conference. This image should be included as a footer to any invitation to the event.		
Source: Matt		
Fit Criterion: Picture is successfully uploaded, viewed, saved in the database and embedded into email footers.		
Dependencies: None		
Classification: Stretch		
History: Created by Matthew Boroczky on 24/8/15		

VENUE CREATION SUBSYSTEM

Name: Build a Conference Room		
Requirement #: F_V1	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall have the ability for managers to graphically build a conference room/event room that consists of seats and utilities such as a screen or a stage in a Birds-Eye-View representation.		
Rationale: This is an important part of an event booking system and will tie into the ticketing system. This will reduce/eliminate the cost an organisation may pay to development companies to develop a seat selection tool or reduce cost from purchasing other third party application. Having this information centralised in the one system will also reduce errors and data duplication.		
Source: Matt		
Fit Criterion: A conference room builder has been built and a manager has access to some graphical representation of the room structure which can be displayed on the drawing board.		
Dependencies: Weak dependency on F_V15. User interface will be improved with the availability of F_V15.		
Classification: Dream		
History: Created by Matthew Boroczky 28/8/2015		

Name: Drag-and-drop for Utilities Location		
Requirement #: F_V2	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the manager to drag and drop utilities that may be provided within a room onto the graphical drawing board to allows design around utilities. Utilities include items such as Screens, Stage, Speakers, Microphones, Disabled hearing areas, Wheelchair ramps, Whiteboards, Smartboards, Bathrooms. Food areas, Couches		
Rationale: This will provide some form of map and aid booking. None of these functionalities are mandatory to account for companies who hire graphic designers to design they're birds-eye-view map		
Source: Matt		
Fit Criterion: There are numerous graphical utilities which can be dragged onto the drawing panel		
Dependencies: System must have the functionality to build a room. Refer to F_V1		
Classification: Dream		

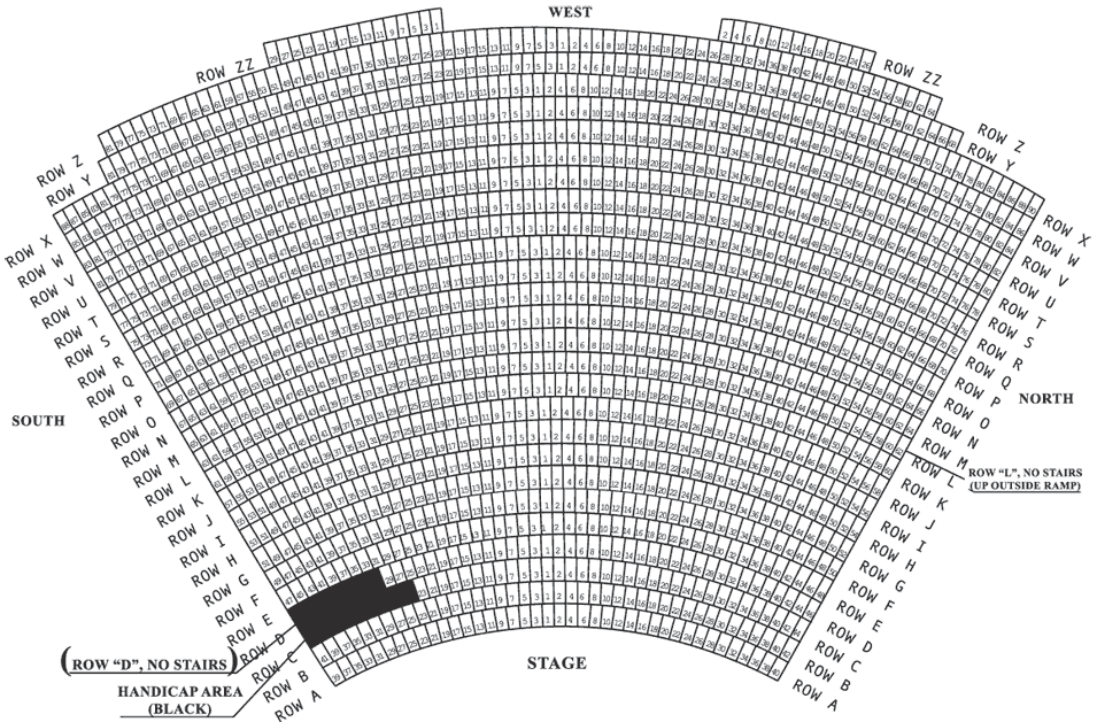
History: Created by Matthew Boroczky 28/8/2015

Name: Drag-and-drop for Seat Location		
Requirement #: F_V3	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the manager to drag and drop seats provided within a room. This ensures complete customizability to the manager to allow and accurate representation of the seating layout in all possible rooms		
Rationale: This is the software that will not only be used to display the seating locations to planners but also to customers in allowing them to book specified seats.		
Source: Matt		
Fit Criterion: Seats can be dragged onto the drawing panel		
Dependencies: System must have the functionality to build a room. Refer to F_V1		
Classification: Dream		
History: Created by Matthew Boroczky 28/8/2015		

Name: Selection of Seat Numbers by Axis		
Requirement #: F_V4	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide the functionality for managers to select the number of seats they want in both the X and the Y axis, generating a rectangular/cube block of seats which is displayed on the drawing board		
Rationale: This will improve efficiency of the room builder reducing the need to individually drag and drop hundreds or thousands of seats onto the drawing board.		
Source: Matt		
Fit Criterion: A tool has been developed to allow manual entry of the number of seats in the X and Y axis and one selected the seats should appear on the drawing board.		
Dependencies: System must have the functionality to build a room. Refer to F_V1		
Classification: Dream		
History: Created by Matthew Boroczky 28/8/2015		

Name: Fan Out		
Requirement #: F_V5	**Requirement Type:** Functional	**Use Case ID:**

Description: The system shall provide a fan out option which allows the user to select a the number of rows in the first row and the number of rows in the last row. The system should automatically fan out increasing/decreasing the number of seats by the appropriate amount (default one). An example of a fanned out venue is below (note that the seating in the diagram is also curved)



Rationale: This increases the applicability of the venue builder and moves it from a semi-useful tool to a tool that has the potential to be realistically used within the industry.

Source: Matt

Fit Criterion: The fan our tools when click will fan out the selected seats

Dependencies: System must have the functionality to build a room. Refer to F_V1

Classification: Dream

History: Created by Matthew Boroczky 28/8/2015

Name: Unavailable Seats		
Requirement #: F_V6	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall have the ability to display unavailable seats by greying out seats that have been booked when being displayed to both the public when booking a ticket and managers when viewing the selected seats sold out.		
Rationale: If the system was to not indicate the status of the seats it would have no functional purpose other than a map. By displaying unavailable seats it allows the view to be used as a seat selection tool in the ticketing process.		
Source: Matt		
Fit Criterion: When seats that are unavailable appear grey to both guest booking tickets and managers viewing report information.		
Dependencies: System must have the functionality to build a room. Refer to F_V1		
Classification: Dream		
History: Created by Matthew Boroczky 28/8/2015		

Name: Seat Types		
Requirement #: F_V7	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow organisers to set different types of seats. The types and number of number should be up to the organisers however there will be some default seat types in the system. The two default seats present in the system for organisers to employ is wheelchair seating, pram specific seating, children seating, hearing loop seating and couple seating (two seats that are joined together and require a minimum purchase of 2 tickets)		
Rationale: Many events have special forms of seating due to individual requirements or users wants. By providing the ability to have special seating types as well as having default seat types the practicality of the system is increased.		
Source: Matt		
Fit Criterion: When managers can create specific seat types, when managers can select built-in/default seat types, when seat types are displayed on the drawing board, when seat types are displayed during seat selection for ticket purchasing		
Dependencies: None		
Classification: Main		
History: Created by Matthew Boroczky 28/8/2015		

Name: Seat Classes		
Requirement #: F_V8	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow organisers to set different classes of seats. The classes and number of classes should be up to the organisers. To accurately describe the notion of seat classes the ideas is similar to plane seats where they have first class, business class, premium and economy seats. This is similar for events where there are different seat classes each holding their own price.		
Rationale: For the application to be used for not only conferences but events there must be some form of classing available. Therefore to allow the booking via seat selection to events there must be a way to graphically determine what seats belong to what class and a way to graphically represent this to guests purchasing tickets.		
Source: Matt		
Fit Criterion: When organisers can create numerous seat classes, specify what seats belong to which class and have the system indicate which seats belong to which class		
Dependencies: None		
Classification: Main		
History: Created by Matthew Boroczky 28/8/2015		

Name: Selecting Seats for Types		
Requirement #: F_V9	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow Managers to select seats and specify what type of seat they are		
Rationale: Having a GUI for selecting seats and classifying their type is more effective than simply specifying the rows and number of seats in each row and/or column		
Source: Matt		
Fit Criterion: When Managers can select one or more seats and specify their Type. Once their Type has been selected an icon on the seat will be added to indicate the type, and a legend will appear to indicate what seats are of what type.		
Dependencies: Seat types must be specified. Refer to F_V7		
Classification: Main		
History: Created by Matthew Boroczky 28/8/2015		

Name: Selecting Seats for Classes		
Requirement #: F_V10	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the Manager to select seats and specify which class they belong to.		
Rationale: Having a GUI for selecting seats and classifying their class is more effective than simply specifying the rows and number of seats in each row and/or column		
Source: Matt		
Fit Criterion: When Managers can select one or more seats and specify their class. Once their class has been selected the seats background will change colour and a legend will appear to indicate what seats are of what class.		
Dependencies: Seat classes must be specified. Refer to F_V8		
Classification: Main		
History: Created by Matthew Boroczky 10/9/2015		

Name: Automatic Legend Generation		
Requirement #: F_V11	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall automatically generate a legend of the utilities, seat types and seat classes present on the room builder.		
Rationale: Having graphical representations of different types is good however if customers and managers can't interpret the graphical meaning than the use of graphics to display meaning is useless - a legend is required.		
Source: Client		
Fit Criterion: When a manager has one or more objects on the drawing panel and the venue design is saved the legend is automatically generated.		
Dependencies: Users shall be able to select seat types and seat classes. Refer to F_V9, F_V10		
Classification: Dream		
History: Created by Matthew Boroczky 28/8/2015		

Name: Seating Release Conditions		
Requirement #: F_V12	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide the ability for conditions to be applied to different seats, seat types, or seat classes.		
Rationale: For management purposes organisers need to apply constraints from the release of seats such as when 1/4 of the first class seats at the front row are booked release the first class seats on the balcony on both the left and right side. Constraints such as this ensure optimal ticket purchasing and reduce the gaps (unbooked seats) between booked seats, which has the potential to substantially minimise cost by not releasing certain sections.		
Source: Matt		
Fit Criterion: When managers can specify release conditions and the release conditions can be simulation and the effect are correct based off the conditions		
Dependencies: Seat classes must be specified. Refer to F_V8		
Classification: Dream		
History: Created by Matthew Boroczky 28/8/2015		

Name: Ability to draw venue with line tools		
Requirement #: F_V13	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide a line tool that allows the user to draw the map of a room.		
Rationale: This removes the need of graphic designers to draw maps which can be quite costly and not necessarily worthwhile for cheaper events such as small local events with 50-100 people.		
Source: Client		
Fit Criterion: A straight-line tool has been implemented in the system allowing the drawing of the room		
Dependencies: System must have the functionality to build a room. Refer to F_V1		
Classification: Dream		
History: Created by Matthew Boroczky 28/8/2015		

Name: Ability to add Labels		
Requirement #: F_V14	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall have a label element which can be dragged onto the drawing board		
Rationale: This is necessary for communication between designer of the system and customers (potentially even other managers of the system need labels to understand)		
Source: Client		
Fit Criterion: A Label element has been implemented which can be dragged into the drawing panel and can be doubled clicked in order to override the current default text		
Dependencies: System must have the functionality to build a room. Refer to F_V1		
Classification: Dream		
History: Created by Matthew Boroczky 28/8/2015		

Name: 3D Venue Builder		
Requirement #: F_V15	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow the event manager to use a GUI interface to construct a 3D version of the venue.		
Rationale: Event managers may wish to show off the venue in more detail.		
Source: Matthew		
Fit Criterion: Event managers can create a 3D environment		
Dependencies: None		
Classification: Dream		
History: Created by Sandon Joubert 11/09/2015		

Name: Virtual Walkthrough		
Requirement #: F_V16	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall users may experience a virtual walkthrough of the venue.		
Rationale: Users may wish to check out the special layout of a venue.		
Source: Matthew		
Fit Criterion: Users can successfully walkthrough the venue in a virtual environment		
Dependencies: Requires a 3D venue builder. Refer to F_V15		
Classification: Dream		
History: Created by Sandon Joubert 11/09/2015		

Name: Selecting, moving, removing and modifying Objects		
Requirement #: F_V17	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide individual selection as well as multiple selection on objects on the drawing board. Once selected, they can be moved, deleted or edited.		
Rationale: Designs will never be final and all individuals make mistakes. By implementing the ability to modify the objects in the view we are adding necessary functionality for a drag-and-drop view		
Source: Client		
Fit Criterion: Individual and multiple items can be selected, moved, removed and depending on the object becomes modified (seat class and type)		
Dependencies: Requires objects to be available on the drawing board first. Refer to F_V2, F_V3		
Classification: Dream		
History: Created by Matthew Boroczky 28/8/2015		

Name: Book Accommodation for Guests		
Requirement #: F_V18	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall have the ability to book many rooms, and different types of rooms for conference managers and event organisers. This allows conference managers to offer rooms for the conference and give the ability to add tickets to the event/conference that are inclusive of accommodation types.		
Rationale: Certain events/conferences have hotel already booked for their attendees. Providing some level of automation would aid the organisers job.		
Source: Client –		
Fit Criterion: Organisers can have specified number and types of rooms and a request has successfully been sent to the associated hotel.		
Dependencies: None		
Classification: Stretch		
History: Created by Matthew Boroczky 28/8/2015		

Name: Pricing options for Accommodation		
Requirement #: F_V19	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall have pricing options for accommodation.		
Rationale: For events/conferences that provide accommodation booking services, display pricing options for users who is keen to make a booking.		
Source: Client –		
Fit Criterion: Organizers can have specified pricing options to the associated hotel on the event page.		
Dependencies: None		
Classification: Stretch		
History: Created by Liew Loo Yee 11/9/2015		

Name: Upload Birds-Eye-View Image of Venue		
Requirement #: F_V20	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow organisers to upload a Birds-Eye-View image of the venue.		
Rationale: This allows the graphical design to be left to the designer whilst allowing the organiser to take advantage of the seat selection tool for ticketing and analytics.		
Source: Client		
Fit Criterion: A image can be uploaded and is presented on a low layer on the drawing board (i.e. drawing utilities and seat are on top of the image)		
Dependencies: None		
Classification: Stretch		
History: Created by Matthew Boroczky 28/8/2015		

CONFERENCE LOGISTICS SUBSYSTEM

Name: Book Accommodation for Guests		
Requirement #: F_L1	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall have the ability to Book numerous amounts of rooms, and different types of rooms for conference managers and event organisers. This allows conference managers to offer rooms for the conference and give the ability to add tickets to the event/conference that are inclusive of accommodation types.		
Rationale: Certain events/conferences have hotel already booked for their attendees. Providing some level of automation would aid the organisers job.		
Source: Client –		
Fit Criterion: Organisers can have specified number and types of rooms and a request has successfully been sent to the associated hotel.		
Dependencies: None		
Classification: Stretch		
History: Created by Matthew Boroczky 28/8/2015		

Name: Room Selection for Speakers		
Requirement #: F_L2	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the organiser to select rooms for presenters.		
Rationale: This will allow the organiser to manage that each track have how many presenter, present for how much time.		
Source: Michael		
Fit Criterion: Assign presenter into different track by topics, room criteria and other reason.		
Dependencies: May only occur within a created conference. Refer to F_U9		
Classification: Main		
History: Created by Chee Yeong Lim at September 4th, 2015		

Name: Create a presentation		
Requirement #: F_L3	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow the event manager to create presentations for the event. An event may have one or many presentations.		
Rationale: It's not much of an event without presentations.		
Source: Matt		
Fit Criterion: Event manager can successfully create a presentation for the event and add extras if they wish.		
Dependencies: May only occur within a created conference. Refer to F_U9		
Classification: Core		
History: Created by Sandon Joubert 11/09/2015		

Name: Determine Available Rooms for Presenters		
Requirement #: F_L4	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall suggest suitable rooms for presenters based on presentation topic and room preference.		
Rationale: A huge conference might have a few session on going at the same time, this features will save time on managing room.		
Source: Toby		
Fit Criterion: When organiser want to select room for presenters, the system will recommend the organiser to place presenters on certain room based on the topic, room criteria and time.		
Dependencies: None		
Classification: Main		
History: Created by Chee Yeong Lim at September 4th, 2015		

Name: Filter Criteria for Room Booking		
Requirement #: F_L5	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow presenters to select required equipment for their presentation.		
Rationale: Due to scarcity, the organiser can't place every equipment in every room.		
Source: Blithe		
Fit Criterion: When a presenter is assigned for a topic, the presenter can select the equipment needed.		
Dependencies: None		
Classification: Stretch		
History: Created by Chee Yeong Lim at September 4th, 2015		

Name: Timetable Generation		
Requirement #: F_L6	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall generate a personalized timetable for users based on their selected presentations.		
Rationale: Attendees and presenters can have a personalized timetable, so that they will know they should attend which session.		
Source: Sandon		
Fit Criterion: A conference usually have a few track on going for a few days, looking at the full conferences timetable will be a bit mess, participants can select the session they are interested before hand.		
Dependencies: Rooms for presentations must be created and speakers must be assigned to the rooms. Refer to F_L2, F_L3		
Classification: Main		
History: Created by Chee Yeong Lim at September 4th, 2015		

Name: User Specific Meal Preferences		
Requirement #: F_L7	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow participants to select their meal preferences (ie: normal, vegetarian, halal).		
Rationale: As an organiser, we should take care of everyone food preferences.		
Source: Loo Yee		
Fit Criterion: Attendees always have different food preferences, attendee select their preferences, which allow the organiser to prepare the foods according to the ratio in order to reduce food wastage.		
Dependencies: None		
Classification: Stretch		
History: Created by Chee Yeong Lim at September 4th, 2015		

Name: Video Conference Capabilities		
Requirement #: F_L8	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide functionality for live broadcasting to attendees in remote areas.		
Rationale: Conferences location always have limited space, live broadcast is one of the way to extend the information spread.		
Source: Matt		
Fit Criterion: Attendees don't have to travel to the conferences anymore, attendees have the right to access to the topic which is related to the attendees and don't have to worry that the participants limitation for certain session.		
Dependencies: None		
Classification: Dream		
History: Created by Chee Yeong Lim at September 4th, 2015		

Name: Ability to have Private or Public Events		
Requirement #: F_L9	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow users to search for public or invite-only events.		
Rationale: The event can be hidden from search or publicly available based on the event property.		
Source: Toby		
Fit Criterion: Organiser can organize Annual General Meeting or other private events for his own company or certain targeted group.		
Dependencies: Events must be created. Refer to F_E1		
Classification: Main		
History: Created by Chee Yeong Lim at September 4th, 2015		

Name: Ability to have Private or Public Presentations/Performances		
Requirement #: F_L10	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow presentations/performances to be marked as public or private.		
Rationale: There is some product or technique are not available in market and it is still under development, some attendees are invited to be the pilot program.		
Source: CY		
Fit Criterion: Some technology company do have secret track in a conferences while some other session on going.		
Dependencies: Events must be created. Refer to F_E1		
Classification: Stretch		
History: Created by Chee Yeong Lim at September 4th, 2015		

Name: Automatically generate Visa Letters		
Requirement #: F_L11	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall be able to generate letters to indicate proof of attendance for users from foreign countries. These letters may be used for visa application purposes.		
Rationale: Visa approval is one of the issues that blocked attendees from oversea, it is also an extra work for organiser to personalize each of the letter when the person request for it.		
Source: Matt		
Fit Criterion: Oversea Attendees can use the letter as a proof to apply for Visa for going to the conference.		
Dependencies: None		
Classification: Stretch		
History: Created by Chee Yeong Lim at September 4th, 2015		

Name: Ability to have a presenter that is not an author of a paper		
Requirement #: F_L12	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow an author to select a presenter, or presenters, to present their topic/research/finding.		
Rationale: Author of a paper might not good in presentation, in order to let attendees have better understanding, author is allow to choose presenter.		
Source: Loo Yee		
Fit Criterion: When the paper is elected, the actor can select the presenter to represent himself for present the paper.		
Dependencies: None		
Classification: Main		
History: Created by Chee Yeong Lim at September 4th, 2015		

Name: Ability to have backup presenters		
Requirement #: F_L13	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the author to appoint a backup presenter to present the topic.		
Rationale: In order to minimize the unknown circumstances affect the conference workflow.		
Source: Toby		
Fit Criterion: If the presenter doesn't show up in the event, the backup presenter will be the one who present for the topic.		
Dependencies: System shall allow authors to select a presenter. Refer to F_L12		
Classification: Main		
History: Created by Chee Yeong Lim at September 4th, 2015		

Name: Ability to transfer access to a event to new organisers		
Requirement #: F_L14	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the organiser to assign another person as an organiser and remove self from the role.		
Rationale: To make sure that the event can continue if the organiser changed.		
Source: Michael		
Fit Criterion: Organiser can promote any management role to the organiser position and revoke himself.		
Dependencies: • Event must be created. Refer to F_E1 • User must have an upgraded account. Refer to F_U7.		
Classification: Dream		
History: Created by Chee Yeong Lim at September 4th, 2015		

Name: Ability to copy event data as a template for others user		
Requirement #: F_L15	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall be able to use existing event data as a template for new events.		
Rationale: Some events are organized in a series, ability to use an old event as template saved a lot of time for the event organiser.		
Source: CY		
Fit Criterion: Event organiser can go to the old event, and select "Use as template" to create new event with most of the data from old event.		
Dependencies: Feature to create events must exist. Refer to F_E1		
Classification: Stretch		
History: Created by Chee Yeong Lim at September 4th, 2015		

Name: Ability to publish conference/event proceedings before and after the event		
Requirement #: F_L16	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall be able to publish conference/event proceedings for attendees.		
Rationale: The attendees doesn't have to ask each speaker for resources, as the organiser will publish it during or after the conference.		
Source: Sandon		
Fit Criterion: The organiser can use the resources that had been submitted during call for papers to publish proceedings.		
Dependencies: • User must have an account. Refer to F_U1 • Event must exist. Refer to F_E1 • Papers must be submitted to the conference. Refer to F_PM1, F_PM2, F_PM3, F_PM4		
Classification: Stretch		
History: Created by Chee Yeong Lim at September 4th, 2015		

TICKETING AND PAYMENT

TICKETING

Name: User purchase ticket		
Requirement #: F_TP1	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide users the ability to purchase tickets for an event.		
Rationale: Provide convenience to buy and book tickets via the conference management system so users can save time and secure a spot as early as they can		
Source: Client		
Fit Criterion: By clicking a buy ticket button navigates user to select ticket and select payment method		
Dependencies: Users must be able to book tickets. Refer to F_TP9		
Classification: Core		
History: Created by Liew Loo Yee 5/9/2015		

Name: Interface with external payment system		
Requirement #: F_TP2	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall be able to interact with external payment systems to handle payments. External systems will include Mastercard, Visa, PayPal.		
Rationale: Handling transactions involving money should always be handled by an outside agent for security.		
Source: Client		
Fit Criterion: The system successfully makes transactions to external payment systems.		
Dependencies: None		
Classification: Main		
History: Created by Sandon Joubert 11/09/2015		

Name: Select seat type and class of seat		
Requirement #: F_TP3	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow an attendee to select a type of seat and class of seat through a GUI 2D interface of the venue. Types are like wheelchair, hearing-aid, normal, etc. Classes are like first-class, economy, etc.		
Rationale: Different attendees have different needs as well as different willingness to get better seats.		
Source: Client		
Fit Criterion: Attendees can successfully select a seat type and class of seat.		
Dependencies: None		
Classification: Stretch		
History: Created by Sandon Joubert 11/09/2015		

Name: Viewing bird's eye view of the event system		
Requirement #: F_TP4	**Requirement Type:** Functional	**Use Case ID**:
Description: The system shall allow attendees to view the venue from a 2D bird's eye perspective.		
Rationale: Attendees would like an idea of the layout of the venue.		
Source: Client		
Fit Criterion: Users can successfully view the venue.		
Dependencies: None		
Classification: Main		
History: Created by Sandon Joubert 11/09/2015		

Name: Optional accommodation for guests		
Requirement #: F_TP5	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow guests to choose their accommodation booked through the organiser		
Rationale: Provide convenience for guests looking for accommodation for attending the conference, or guests who do not have time to go through accommodation booking process		
Source: Sandon		
Fit Criterion: After purchasing ticket, ask user if they want to have their accommodations booked by the system automatically, presented by a button or link		
Dependencies: None		
Classification: Stretch		
History: Created by Liew Loo Yee 31/8/2015		

Name: Managers create categories of tickets (other than general admission)		
Requirement #: F_TP6	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow different categories of tickets to be sold/generated. Examples of these categories are children tickets, special guests, seated or standing tickets, or tickets for various talks.		
Rationale: Cater for different types of events with a variety of ticket types and pricing, create more flexibility		
Source: Michael		
Fit Criterion: Conference manager can create multiple ticket types (with specified pricing) whilst creating/managing conference page at the “ticket” segment		
Dependencies: None		
Classification: Main		
History: Created by Liew Loo Yee 31/8/2015		

Name: Attendees can select a ticket type for purchase		
Requirement #: F_TP7	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow attendees to choose their preferred ticket type.		
Rationale: If conference managers offer choices for ticket types, users should be able to pick ticket types of their own preference.		
Source: Michael		
Fit Criterion: The ticketing segment displays different ticket types (if any) in form of checkbox or drop-down list		
Dependencies: Different ticket categories must be created. Refer to F_TP6		
Classification: Main		
History: Created by Liew Loo Yee 31/8/2015		

Name: Email notification upon ticket purchase		
Requirement #: F_TP8	**Requirement Type:** Functional	**Use Case ID:**
Description: Users get email notification of purchase summary upon transaction/booking		
Rationale: Confirmation on transaction, confirm buyer identity, also act as a record for backtrack purposes.		
Source: Michael		
Fit Criterion: Sends automatically generated email through a no-reply account, embed link in the email for purchase confirmation		
Dependencies: Users need to successfully purchase a ticket. Refer to F_TP1, F_TP2		
Classification: Main		
History: Created by Liew Loo Yee 31/8/2015		

Name: Book ticket for multiple people if public event		
Requirement #: F_TP9	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow users to book more than one ticket (group booking) for public event that requires no identification for each attendee		
Rationale: In case of a company or education institute, provide convenience of having one person to book ticket for a group of people going to conference, higher chances to have group seating arranged, avoid trouble and missing out on the booking		
Source: Michael		
Fit Criterion: A drop down list of ticket quantity in the ticket booking section, strictly for public events only. Submission of attendees' details depends on the demand of the conference manager.		
Dependencies: None		
Classification: Main		
History: Created by Liew Loo Yee 31/8/2015		

Name: Purchase ticket for multiple events		
Requirement #: F_TP10	**Requirement Type:** Functional	**Use Case ID:**
Description: User should be able to purchase tickets for multiple events with their account		
Rationale: It is not ideal to restrict user on ticket purchasing, limitation on number of tickets to be purchased by user reduces user experience.		
Source: Client		
Fit Criterion: Ability to purchase tickets even though user has already purchased tickets for other events.		
Dependencies: Users must be able to purchase tickets. Refer to F_TP1, F_TP2		
Classification: Core		
History: Created by Liew Loo Yee 6/9/2015		

Name: User makes payment		
Requirement #: F_TP11	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow users to make payment online, for purchasing tickets, through implemented payment methods.		
Rationale: Online transaction provide convenience, saves traveling time and provide 24/7 accessibility		
Source: Client		
Fit Criterion: For priced tickets, display payment options and ask user to fill in information regarding their credit card/paypal etc.		
Dependencies: • System should allow users to purchase tickets. Refer to F_TP1 • System should interface with external payment system. Refer to F_TP2		
Classification: Core		
History: Created by Liew Loo Yee 6/9/2015		

Name: Different payment option		
Requirement #: F_TP12	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall support different payment options such as paypal, credit card, visa etc.		
Rationale: To cater for different types of payment method, expand market size and enhance user experience		
Source: Michael		
Fit Criterion: Display several types of payment option for selection on the ticket payment page		
Dependencies: System should interface with external payment system. Refer to F_TP2		
Classification: Core		
History: Created by Liew Loo Yee 6/9/2015		

Name: Option to limit payment method		
Requirement #: F_TP13	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the organiser to limit payment method for the event's ticket purchasing procedures		
Rationale: Some organiser may want to limit to payment method for easy managing and analytic purposes		
Source: Matthew		
Fit Criterion: When creating an event, organiser can select the option to limit payment method before publish, or during page editing		
Dependencies: None		
Classification: Main		
History: Created by Liew Loo Yee 6/9/2015, edited by Liew Loo Yee 10/9/2015		

Name: Payment log to keep track of transaction record		
Requirement #: F_TP14	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow users to keep track of their personal transaction record through a payment log		
Rationale: To keep track of user's payment, for both user and system's record and reference purposes		
Source: Matthew		
Fit Criterion: When a transaction is made, add the information to the transaction log.		
Dependencies: Users need to make payments for tickets. Refer to F_TP1, F_TP2		
Classification: Core		
History: Created by Liew Loo Yee 6/9/2015, edited by Liew Loo Yee 10/9/2015		

Name: Ability to view payment log		
Requirement #: F_TP15	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow users to view their own payment log/transaction records		
Rationale: For user's own reference, know what they paid for and when the transaction is made, can act as useful evidence when error or problem occurs to be forwarded to organiser or system developer		
Source: Matthew		
Fit Criterion: On the user's personal page, display option to view transaction record.		
Dependencies: The system shall provide a payment log. Refer to F_TP14		
Classification: Core		
History: Created by Liew Loo Yee 10/9/2015		

Name: Waiting list		
Requirement #: F_TP16	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall create a waiting list for users who wish to attend a fully booked event. If more tickets are released those at the front of the queue are given priority.		
Rationale: Possible attendees are would like to be able to still go to the event.		
Source: Junyan Fan		
Fit Criterion: When clients press the button about waiting list, the system will show a list. In this list, custom can check future event and the time of start sale ticket.		
Dependencies: None		
Classification: Stretch		
History: Created by Junyan Fan 05/9/2015		

Name: Ability to generate QR codes for tickets		
Requirement #: F_TP17	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall generate QR codes for tickets which allow for quick ticket processing at the event.		
Rationale: QR codes make ticket processing process very easy and fast.		
Source: Loo Yee		
Fit Criterion: When client buys a ticket, the system will provide a pdf version of the ticket with a QR code. Can be scanned at event as admittance.		
Dependencies: Users need to purchase tickets. Refer to F_TP1, F_TP2		
Classification: Stretch		
History: Created by by Junyan Fan 05/9/2015		

Name: Ability to save tickets in iOS Passport		
Requirement #: F_TP18	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall save a user's tickets to iOS Passport app if they are using an iPhone. Event organisers can check the ticket in the 'passport' app of every apple mobile devices.		
Rationale: Lots of people use apple mobile devices, if we cans save the ticket to 'passport', user can check their e-ticket easily.		
Source: CY		
Fit Criterion: After user bought a ticket, client will find a "save to passport" button near the ticket, if client press it, system will ask the apple id, then save it to these id's passport.		
Dependencies: Users need to purchase tickets. Refer to F_TP1, F_TP2		
Classification: Stretch		
History: Created by Junyan Fan 05/9/2015		

Name: Generate financial statements		
Requirement #: F_A1	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide the chair a GUI to generate financial statements.		
Rationale: A chair wants to generate financial statement by using the system.		
Source: Chair		
Fit Criterion: The financial statement can be generated and displayed successfully.		
Dependencies: None		
Classification: Stretch		
History: Created by Ruixi He 07/9/2015		

Name: Generate statistical reports		
Requirement #: F_A2	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide the chair a GUI to generate statistical reports.		
Rationale: A chair wants to generate statistical reports by using the system.		
Source: Chair		
Fit Criterion: The statistical reports can be generated and displayed successfully.		
Dependencies: Requires data to generate reports. Refer to F_A3, F_A4, F_A5		
Classification: Dream		
History: Created by Ruixi He 07/9/2015		

Name: Number of attendees per event type		
Requirement #: F_A3	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall calculate the number of attendees per event type.		
Rationale: During the generating of the statistical report, the system can calculate the number of attendees per event type.		
Source: Chair		
Fit Criterion: The number of attendees per event type can be calculated and generate to the statistical report successfully.		
Dependencies: None		
Classification: Stretch		
History: Created by Ruixi He 07/9/2015		

Name: Satisfaction of Attendees per event type		
Requirement #: F_A4	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall record attendee satisfaction of events categorized by their type.		
Rationale: During the generating of the statistical report, the system can also generate the satisfaction of attendees per event type by rank from 1 to 5.		
Source: Chair		
Fit Criterion: The satisfaction rank can be generated successfully.		
Dependencies: None		
Classification: Stretch		
History: Created by Ruixi He 07/9/2015		

Name: Number of purchased tickets for each category of ticket		
Requirement #: F_A5	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall calculate the number of purchased tickets for each ticket category.		
Rationale: During the generating of the statistical report, the system can calculate the number of purchased tickets for each category of ticket.		
Source: Chair		
Fit Criterion: The number of purchased tickets for each category of ticket can be calculated and generate to the statistical report successfully..		
Dependencies: None		
Classification: Stretch		
History: Created by Ruixi He 07/9/2015		

Name: Generate budget reports		
Requirement #: F_A6	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall generate budget reports.		
Rationale: A chair wants to generate the budget report by using the system.		
Source: Chair		
Fit Criterion: The budget report can be successfully generated.		
Dependencies: None		
Classification: Stretch		
History: Created by Ruixi He 07/9/2015		

Name: Ability to have sponsors		
Requirement #: F_A7	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide the chair a GUI to add a sponsor into the event.		
Rationale: A chair wants to add a sponsor into the event.		
Source: Chair		
Fit Criterion: The sponsor can be added successfully.		
Dependencies: None		
Classification: Main		
History: Created by Ruixi He 07/9/2015		

Name: Ability to upload advertisement data		
Requirement #: F_A8	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide the chair a GUI to upload advertisement data to the system.		
Rationale: A chair wants to upload advertisement data to the system.		
Source: Chair		
Fit Criterion: The advertisement data can be uploaded successfully.		
Dependencies: None		
Classification: Main		
History: Created by Ruixi He 07/9/2015		

Name: Viewing purchased and available seats with Statistics (birds-eye)		
Requirement #: F_A10	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the chair to view the seating statistics in birds-eye view.		
Rationale: A chair wants to view the bird-eye side of the purchased and available seats.		
Source: Chair		
Fit Criterion: The different colour of the seats can be displayed successfully.		
Dependencies: A birds-eye view image of the venue must be available. Refer to F_V20		
Classification: Stretch		
History: Created by Ruixi He 07/9/2015		

Name: Viewing Birds-Eye-View of the event		
Requirement #: F_A11	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the chair to view the event area as birds-eye view.		
Rationale: A chair wants view the whole event area in a birds-eye view.		
Source: Chair		
Fit Criterion: The birds-eye view of the whole event area can be showed successfully.		
Dependencies: A birds-eye view image of the venue must be available. Refer to F_V20		
Classification: Dream		
History: Created by Ruixi He 07/9/2015		

Name: Displaying Financial Statements		
Requirement #: F_A12	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall be able display generated financial reports and statements.		
Rationale: The Organisers or administrator want to have a report or diagram to tell them Financial Statements, it can let them know what happen about their business.		
Source: Client		
Fit Criterion: in the home page, the system have button about report, the report have a selection about financial report, and the organisers can see the diagram, table about the financial.		
Dependencies: The financial reports and statements must be generated. Refer to F_A1		
Classification: Stretch		
History: Created by by Junyan Fan 05/9/2015		

Name: Displaying Statistical Reports		
Requirement #: F_A12	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall be able to display generated statistical report and statements. This report include the sum of user, the increasing about user...		
Rationale: The Organisers or administrator want to have a statistical report to tell them statistical data about every part. It can let them know what happen about their conference.		
Source: Matt		
Fit Criterion: in the home page, the system have button about report, the report have a selection about Statistical Reports, and the organisers can see statistical data about every part.		
Dependencies: The statistical reports and statements must be generated. Refer to F_A2		
Classification: Dream		
History: Created by by Junyan Fan 05/9/2015		

IN-HOUSE SUBSYSTEM

Name: Optional Surveys		
Requirement #: F_IH1	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow event/conference organisers to create surveys in the system. They should be able to specify a question, answer type - multiple choice or textfield and multiple choice answers if a multiple choice question. They should be able to select when the survey is released/sent out and select individual users as-well as categories of users who will receive the survey e.g. presenters of computer science security paper.		
Rationale: Eliminates the need for the client to resort to the use of third party applications for surveys and generating links for users to be sent out via email - simplified for client.		
Source: Michael		
Fit Criterion: When a organiser can create a questionnaire/survey with questions and answers, and can select a users or categories of users who will receive the survey, and can specify the release time/date, and the correct users receive the survey at the correct time		
Dependencies: The event must exist. Refer to F_E1		
Classification: Main		
History: Created by Matthew Boroczky 3/9/2015		

Name: Security Help		
Requirement #: F_IH2	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide a help button on the IOS application. This will notify security that you are in danger/an emergency situation is occurring. Security will be notified and your location will be sent to them.		
Rationale: Not necessary but a useful feature when dealing with large events where locating injured or threatened individuals would be difficult. To prevent accidental requesting of security there should be a twostep process for verification.		
Source: Sandon		
Fit Criterion: After a individual clicks the security button and confirms they are in trouble security employees on both web platforms and IOS devices will receive a notification that someone is in danger		
Dependencies: None		
Classification: Main		

History: Created by Matthew Boroczky 3/9/2015

Name: Comment on Event		
Requirement #: F_IH3	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow general comments to be made by attendees and presenters after the conference.		
Rationale: Feedback is an important part in planning future events which is why feedback of previous events is a useful tool that will be provided to conference organisers		
Source:		
Fit Criterion: After the event date and time has finished attendees (including presenters - anyone with a ticket excluding managers and organisers) has the ability to comment on the event		
Dependencies: The event must exist. Refer to F_E1		
Classification: Core		
History: Created by Matthew Boroczky 3/9/2015		

Name: Submit Questions for a Presentation		
Requirement #: F_IH4	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow attendees to submit questions during a presentation for the presenter to answer		
Rationale: Having a question system not only opens the ability to have an online audience asking questions from at home but also allows the ability for physical attendees to ask question when they may not be able to ask them due to the large number of attendees at the conference i.e. if you're at the back of a conference of thousands of people it will be difficult to ask a question		
Source:		
Fit Criterion: Attendees can successfully post a question and presenters can successfully view posted questions		
Dependencies: The presentation must exist. Refer to F_L3		
Classification: Stretch		
History: Created by Matthew Boroczky 4/9/2015		

Name: Speaker can ask question and attendees can answer		
Requirement #: F_IH5	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow the speaker to submit questions during a presentation. Attendees then can respond to that question through the system.		
Rationale: Speakers may sometimes seek crowd involvement of participation, building it into the in-house system allows this to be done fast and easily especially when dealing with larger audiences.		
Source: Junyan Fan		
Fit Criterion: Speaker can use the ask a question button, specifying the question and answers.		
Dependencies: The presentation must exist. Refer to F_L3		
Classification: Stretch		
History: Created by Junyan Fan 05/9/2015		

Name: Generation of Name Badges		
Requirement #: F_IH6	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall be able to generate a list of attendees names.		
Rationale: Event managers want to get a pdf file of attendees names that can be printed as name badges.		
Source: Michael		
Fit Criterion: Event managers shall be able to see a list of attendee names.		
Dependencies: None		
Classification: Stretch		
History: Created by Junyan Fan 05/9/2015		

Name: Download list of Attendee Names		
Requirement #: F_IH7	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall allow a list of all attendees to be downloaded.		
Rationale: A list of attendees can be useful for manual attendance checking at the door or into presentations		
Source: Junyan Fan		
Fit Criterion: A list of attendee's is downloaded on button press		
Dependencies: A list of attendees must be generated. Refer to F_IH6		
Classification: Core		
History: Created by Junyan Fan 05/9/2015		

Name: Social-Media communication with attendees (find people, who's around you and their expertise)		
Requirement #: F_IH8	**Requirement Type:** Functional	**Use Case ID:**
Description: The system shall provide attendees a social-media platform for communication with all attendees. Attendees can search for other attendees, view attendees with similar interests and expertise.		
Rationale: A user wants to use the social-media platform to communicate with the other attendees.		
Source: System user.		
Fit Criterion: The social-media platform can be launched successfully.		
Dependencies: None		
Classification: Dream		
History: Created by Ruixi He 07/9/2015		

NON-FUNCTIONAL REQUIREMENTS

Name: Availability		
Requirement #: NFREQ_01	**Requirement Type:** Non-Functional	**Use Case ID:** None
Description: The system shall be able to work 24/7 except during maintenance.		
Rationale: Users who are located globally want to access the system 24/7.		
Source: All users who interact with the system.		
Fit Criterion: Users can use the system at any time (24/7).		
Dependencies: None		
Classification: Core		
History: Created by Michael Kong on 06/09/2015		

Name: Performance		
Requirement #: NFREQ_02	**Requirement Type:** Non-Functional	**Use Case ID:** None
Description: The system shall be responsive. The system must respond to actions by any users within 20 seconds.		
Rationale: Users want to receive a quick response from the system.		
Source: All users who interact with the system.		
Fit Criterion: The system responds to the user's action in 20 seconds or less.		
Dependencies: None		
Classification: Core		
History: Created by Michael Kong on 06/09/2015		

Name: Learnability		
Requirement #: NFREQ_03	**Requirement Type:** Non-Functional	**Use Case ID:** None
Description: The system shall not require any specific knowledge or technical skills to operate. Within 1 day or less of exploring the system, the user should understand how to use the system regardless of their role within the system.		
Rationale: Users want to understand how to use the system quickly.		
Source: All users who interact with the system.		
Fit Criterion: The user is able to use the Event Organiser System in 1 day or less.		
Dependencies: None		
Description: The system shall not require any specific knowledge or technical skills to operate. Within 1 day or less of exploring the system, the user should understand how to use the system regardless of their role within the system.		
Rationale: Users want to understand how to use the system quickly.		

Name: Reliability		
Requirement #: NFREQ_04	**Requirement Type:** Non-Functional	**Use Case ID:** None
Description: In 1 day or less, the system shall fully recover from any failure.		
Rationale: Users want to be able to access the system.		
Source: All users who interact with the system.		
Fit Criterion: The system takes 1 day or less to recover from any failure.		
Dependencies: None		
Classification: Core		
History: Created by Michael Kong on 06/09/2015		

Name: Scalability		
Requirement #: NFREQ_05	**Requirement Type:** Non-Functional	**Use Case ID:** None
Description: The system shall support at least 500 simultaneous users at any given time.		
Rationale: The system will be used by many people across different countries across the world.		
Source: All users who interact with the system.		
Fit Criterion: The system should support at least 500 users at any given time.		
Dependencies: None		
Classification: Core		
History: Created by Michael Kong on 06/09/2015		

Name: Security		
Requirement #: NFREQ_06	**Requirement Type:** Non-Functional	**Use Case ID:** None
Description: The system shall automatically log a user out after 10 minutes of idle time.		
Rationale: The system wants 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: Main		
History: Created by Michael Kong on 06/09/2015		

Name: Scalability		
Requirement #: NFREQ_10	**Requirement Type:** Non-Functional	**Use Case ID:** None
Description: The system shall be able to store at least 5000 user profiles.		
Rationale: The system must be able to store user profiles so that users can access the system and its functionalities.		
Source: All users who interact with the system.		
Fit Criterion: At least 5000 user profiles can be stored in the system.		
Dependencies: None		
Classification: Core		
History: Created by Michael Kong on 06/09/2015		

Name: Documentation		
Requirement #: NFREQ_11	**Requirement Type:** Non-Functional	**Use Case ID:** None
Description: The system shall have adequate documentation so that any purchasers of the system are able to understand the workings of the system.		
Rationale: Purchasers of the system want to be able to understand the system without needing to look at the code of the system.		
Source: Potential purchasers system.		
Fit Criterion: Sufficient documentation including planning, design, implementation and testing documents are created.		
Dependencies: None		
Classification: Core		
History: Created by Michael Kong on 06/09/2015		

Name: Extensibility		
Requirement #: NFREQ_12	**Requirement Type:** Non-Functional	**Use Case ID:** None
Description: The system shall be able to have additional functionality built on it.		
Rationale: The system may need to adopt new features to meet the various needs of users in the future.		
Source: All users who interact with the system.		
Fit Criterion: Additional functionality can be added to the system.		
Dependencies: NFREQ_11		
Classification: Stretch		
History: Created by Michael Kong on 06/09/2015		

Name: Maintainability		
Requirement #: NFREQ_13	**Requirement Type:** Non-Functional	**Use Case ID:** None
Description: Potential defects in the system shall be easily corrected or repaired.		
Rationale: Potential defects in the system may occur. So to quickly correct these defects, the system needs to be built in a way that these defects may be quickly isolated and corrected.		
Source: All users who interact with the system.		
Fit Criterion: Defects within the system are quickly isolated and corrected.		
Dependencies: NFREQ_11		
Classification: Stretch		
History: Created by Michael Kong on 06/09/2015		