



EventHelppo

Karen Chang, Amelia Ehrens, Caley Kovler, Rohit Maddipuddi,
Khaled Mahmud, Eduardo Villarreal Holguera
Event Management Database Application

Table of Contents

Introduction	4
Brief Description of The Application and The Business	5
Business Canvas Model.....	5
Key Partners	6
Key Partners.....	6
Key Suppliers.....	6
Key Resources to Acquire	7
Key Activities	7
Key Activities for Value Propositions	7
Distribution Channels.....	7
Value Proposition	7
Customer Relationships	8
Customer Segment	8
Key Resources.....	8
Key Resources	8
Distribution Channel	8
Revenue Streams	9
Channels.....	9
Cost Structure.....	9
Revenue Streams.....	9
Project Plan	11
Work Breakdown Structure.....	11
Description of Task Dependencies.....	11
Project Schedule	12
Resource Description & Analysis	12
Project Risk Analysis	14
Use Case Diagram	15
Use Case Scenarios	17
Activity Diagram	21
Process Description for Activity Diagram.....	21
Activity Diagram (Version one)	23
Activity Diagram (Modified after feedback)	24
Change in Activity Diagram	25
Enhanced Entity-Relationship Model	25
Assumptions for Conceptual Data Modeling	25
EER Diagram (Version one)	27
EER Diagram (Modified after feedback)	27
Change in Conceptual Data Model	28
Relational Data Modeling	28
Converting EER Diagram into Relational Data Model.....	28
MySQL.....	29
MySQL Queries for Table Creation	29

MySQL Data Entry Queries	33
Application Prototype.....	37
Login screens	37
Main Menu Screens	38
Events Screen	39
Event and Guest Information Screen.....	40
Venue Information	41
Guest Information	42
Guest Seating Information	43
Event Staff information.....	45
Conclusion	47
Works Cited	48

Introduction

Managing an event can be a stressful and tedious job. Welcoming guests and guiding them to their seats, while at the same time keeping track of all the attendees is an enormous task. Other than ensuring that seating runs smoothly, event managers must often manage catering, venue staff and client expectations. It is not easy to manage and take care of all these issues at the same time manually.

Through Information Technology for Competitive Advantage II (GR 620B), we had the opportunity to develop a solution to this problem. By utilizing an application for guest registration (check-in), event managers can improve accuracy and efficiency, freeing time for other inevitable and important issues. The result of our efforts is a check-in mobile application system, EventHelppo.

In this final report, we tried to present our entire process of developing the prototype of the application from scratch. We started with a brief description of the application. First of all, we used business canvas model to identify and describe the key elements of the application and the business associated with it. Next, we have described our current understandings of the key business processes that our application will support or enable using some UML tools. Then we illustrated each use case scenarios. In the next step, we described the process, came up with the activity diagram, and then stated our assumptions for conceptual data modeling. Based on the brief description of the application and our assumptions, we established the conceptual data model diagram. We used Enhanced Entity Relationship diagram to describe the relationship among different entities. Then we presented the conversion of the logical data model of the business application into relational model. The relational data model helps give clearer idea of the database structure from the conceptual data model and convert it into technical language. These structures are later used to create SQL statements to prepare a physical database for EventHelppo. In the last step, we created the prototype for our application and introduced our prototype by presenting screenshots. Finally, this final report concludes all the changes had been made according to the feedback from the professor during the process.

Brief Description of The Application and The Business

The purpose of the application is to improve and simplify the processes involved in event management including: registration, seating, food preferences and other important information. Since there are multiple registration staff members checking guests into any event, it can be difficult for the event management team to compare guest lists to ensure that VIPs, presenters and other important guests have arrived before the start of the program and receive necessary documents and ID badges. Additionally, it can be challenging to make changes to seating arrangements without real-time updates on the number of arrivals and cancellations. Because event management involves so many logistical details with high levels of dependencies that can cause delays and inefficiencies, creating this application to assist event managers will cut costs and time.

Rather than continue to utilize this inefficient paper-based process, our application will allow event staff to access and manage the master guest list on their mobile devices. The event management team will be able to import a guest list into the database. Users will then have the option to sort this list by first name, last name, registration status, company name or table number. Additionally, these individuals will have the ability to search for guests quickly, note their arrival and provide them with information including hotel numbers, table numbers, breakout rooms and more. Once an event staff member modifies the guest list on their device, the list will be updated to provide real-time, synchronized information to all application users. If events have size limits, the company can capitalize on guests who did not show up and then sell further tickets. After the event, the management team will have the option to download the final guest list to be analyzed. This will ensure that the staff has accurate information about event attendance for future correspondence and planning.

Business Canvas Model

In order to develop the application we first started with the business model canvas (Osterwalder & Pigneur, 2010), which is divided into nine blocks, including: Customer Segments, Value Propositions, Channels, Customer Relationships, Revenue Streams, Key Resources, Key Activities, Key Partnerships and Cost Structure. Our main goal was to describe

each of these elements to gain a better understanding of what we are going to do and how; who will be playing the key roles; what will be the key activities; and what resources are needed.

First, we defined our target customers as consumers who need assistance with event management, including corporate and personal users. Secondly, we identified several key problems within the event-planning process and developed a solution to remove many logistical inefficiencies to make sure the events go smoothly, cut costs and help event managers focus on other tasks. We also listed the channels, including websites, application stores for customers to reach our product. Furthermore, we determined the services and the relationships we want to establish with our corporate and personal customers by providing customized options, training and post-event data analysis. Additionally, stable revenue streams are determined by segmenting our customers into multiple tiers.

In terms of key resources, key activities and key partners, we make sure all the information can be synchronized and displayed properly on the day of the event. All of the event staff can be trained to use the application. Lastly, we structured our costs for building this event planning application and releasing it into the market. We needed to identify not only the fixed costs for facilities, but also the variable costs for project managers, domain experts, and business analysts, which are the most expensive in this project.

Key Partners

Key Partners

The event managers are our main key partners, who will be managing the events, and the event registration staffs at events because they will be using the application directly. Other key partners are the organizations that provide continuous technical support and the application distributors (Apple store, Google play, other websites).

Key Suppliers

The event participants will be our main suppliers of the information. The database system of the application will also be receiving and supplying continuous information. The IT specialists will be providing upgrades in service and features.

Key Resources to Acquire

We need to purchase hosting and database to keep our information available remotely. We need to recruit IT people who will be helping us creating the application. We also need marketing people to promote the application.

Key Activities

Key Activities for Value Propositions

Main activities will include display and synchronization of real time data on event guests, which will allow efficient management of time. Cross platform usage, which will allow the usage of the application from different devices. The mobile interface will increase the ease of use. Guest check-in and information update of the guest are the two main key activities.

Distribution Channels

Distribution channels include App stores, iTunes store and Google Play store. This application will provide the event managers options not only to manage the events efficiently but also databases of the events, which will help them plan their future events more efficiently.

Value Proposition

Events often have many logistical issues that start at the check-in area: the wait line, the hosts juggling through the many pages in the attendance list looking for the guest's name, and figuring out which guests have arrived and which have not.

Our product is designed to remove the hassle and stress faced by event planners and hosts. With our product, each host can have the attendance list on his/her tablet, look-up guests by first/last name, locate and assign the tables or seats, and check-in guests who have arrived. Our product also provides instant updates so that when a guest checks in, or a change in the guest list is made, it automatically syncs to all of the hosts and planners. Our product removes the stress of the check-in process so that event planners can focus on the things that matter the most.

Customer Relationships

Our target customers include corporate and personal users, and we provide tiered services for different purposes and based on the number of users needed. For corporate users, we can provide a free trial, enterprise version and training programs. For personal users, we will provide event-planning advice, services, and templates. Customized options for smaller events are also available, such as weddings and parties.

Furthermore, we could also provide device rental services for event managers. One-time planners of small events could need to acquire a device, such as ipad, tablet or phone, in order to utilize our application. After events, we can provide data analysis and archiving. The printable versions of data entered can also be made available. For our returning customers, we plan to provide discounts and a referral program.

Customer Segment

This application is designed for internal company use by marketing and event managers. Our app targets large company event managers for hosting dozens or even hundreds of events and conferences per year as well as smaller firms that plan small to medium size events such as weddings or graduations. The app creates value for them by providing immediate communication across multiple event staffs. Ultimately, the app eliminates some of the more time consuming administrative work associated with event management and allows the event managers to focus on more important tasks.

Key Resources

Key Resources

For continuous delivery of value proposition, we will acquire top level IT support so that there is no service interruption. We need people to support our staff who will be continuously managing the application service as developers and upper level management. We will also hire a marketing team so that we can push to generate recurring revenue.

Distribution Channel

As our product will be distributed through app stores, Google play and other websites, we need continuous support from the app distributors. We will acquire or buy top space in the

market to display our application so that users can see our application as soon as they are in the virtual marketplace.

Revenue Streams

To earn recurring revenue, we need marketing support and continuous tech support. We can offer continuous and updated tech support to our customers, which will make our customers loyal to our product. An effective marketing plan is very important so that we can reach the right market efficiently. Marketing people will be a very useful resource. At the same time the users of the application will be a great source for creating positive word of mouth about our app.

Channels

Customers can be reached online through our website and application stores. The website will provide information about the application's capabilities, as well as subscription options. It will also have a link to the application stores where the product is available for purchase. The application stores provide customers with the opportunity to browse or search for applications based on keywords, view screenshots and read reviews. They also have the ability to store credit card information to allow for fast and easy transactions. Once a customer has selected our application for purchase, it will automatically download to their device.

Cost Structure

Billable hours of human resources (Project Manager, Domain Expert, Business Analyst, Marketing Analyst, Financial Analyst, Development and Testing Team) will be one of our main costs due to the need for technical expertise. There will also be costs associated with website hosting services, database servers and bandwidth. A one-time fixed cost will be incurred during software development. After that, there will be yearly fixed costs for subscription for domain name and hosting. Variable costs will include the salaries of personnel, marketing and distribution costs.

Revenue Streams

Event planning is expensive and requires attention to detail to ensure the success of the event. The system currently used at the check-in points (a printed Excel spreadsheet) leads to

inefficiencies. For a small price, event planners can remove the hassle from the check-in area to focus entirely on ensuring the success of the event.

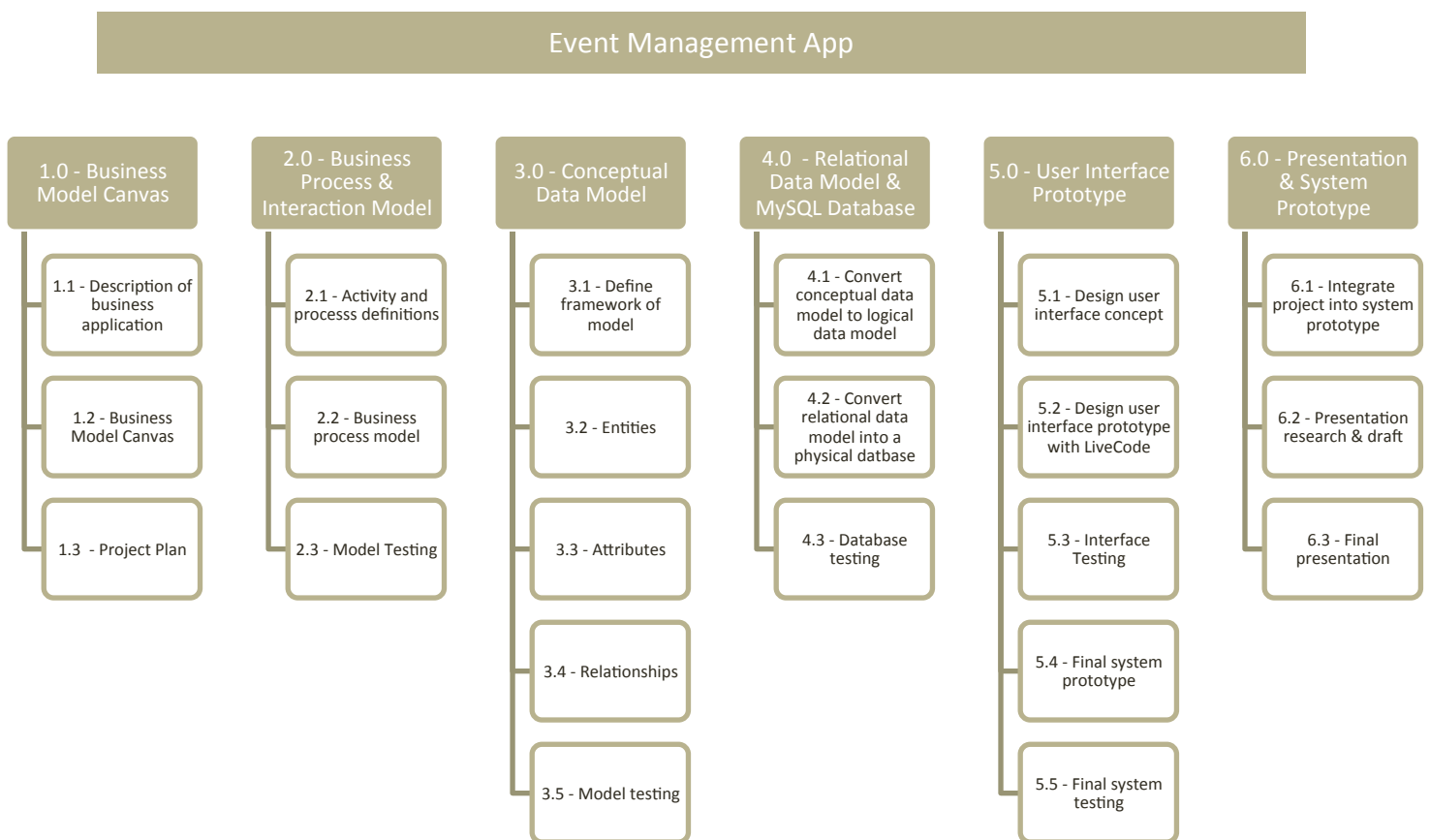
Our product will be sold on a subscription format, by month or by year, at different price points depending on the number of users of the application. For example, a person who is hosting a small event may purchase a one-month subscription with a license for two users for \$5.99. On the other hand, an event planning agency may purchase a year-long subscription with licenses for 20 users for \$99.99. These prices are preliminary estimates.

Clearly, the subscription designed for the small event one-time planner has higher profitability as it basically only generates variable costs and take advantage of the fixed costs incurred to satisfy the large-scale customers.

Project Plan

After developing the primary purpose of the application, we began our project planning. Since the project had to be completed within a single semester, we broke down the tasks and allocated time accordingly. Project deliverable due dates dictated the project schedule. We prepared a work break down structure, description of the task dependencies, project schedule, resource description, responsibilities and risks associated with the project.

Work Breakdown Structure



Description of Task Dependencies

After determining which tasks were dependent upon previous tasks, we approached the task dependencies with a phase gate approach. This is similar to an Agile approach. Each deliverable section specified on the work breakdown structure is dependent on the previous one, but measures need to be taken at the beginning and the end of each phase to ensure that

the next phase can proceed as planned and that the necessary re-work or re-design is completed.

Since each deliverable is a course assignment, we also incorporated Professor feedback into each section in order to proceed with re-work or to the next deliverable.

Project Schedule

WBS	Task	Start	End	Dur	2013			
					Sep	Oct	Nov	Dec
	Event Management Application	9/20/13	12/10/13	79	[Gantt bar spanning Sep to Dec]			
1	Business Process Interaction Model	9/20/13	10/1/13	12	[Gantt bar in Sep]			
2	Business Process & Interaction Model	9/30/13	10/8/13	9	[Gantt bar in Sep]			
3	Conceptual Data Model	10/7/13	10/29/13	22	[Gantt bar in Oct]			
4	Relational Data Model & SQL Database	10/28/13	11/12/13	15	[Gantt bar in Oct]			
5	User Interface Prototype	11/11/13	11/26/13	15	[Gantt bar in Nov]			
6	Presentation & System Prototype	11/25/13	12/10/13	15	[Gantt bar in Dec]			

SmartDraw Academic Edition

Resource Description & Analysis

For this project, we considered all of our team members to be key resources. The table below describes the relevant skills and experiences of each team member, as well as a few other outside resources. This information helped us to assign roles and allocate assignments.

Resource	Skills / Experience
Amelia	Amelia took a Project Management course at Bentley University and has prior experience managing projects at a consulting firm. She was responsible for transferring the company website to a new CMS system and integrating the CRM, automated lead segmentation and email campaigns using HubSpot Enterprise CMS. While the project was not highly technical, she has some experience in evaluating risk and testing to avoid re-work.

Resource	Skills / Experience
Caley	Caley spent two years working for an event management company that planned and executed fundraising events for non-profit organizations in the Boston area. Her daily responsibilities included database management and the creation and updating of sales reports for distribution to clients and committee members. She also regularly assisted with event set-up, registration and the coordination of logistics.
Eduardo	Eduardo, currently concentrating his studies in finance, provides the financial perspective necessary to develop cost analysis, determining price-points, estimating revenue streams and profitability. He also has experience in project valuation.
Karen	Karen had financial experience in her previous career. Currently, she is doing her concentration as business analytics developing data analysis and use of analysis software skills. Combining the finance background, marketing interests and analytic knowledge, her insights bring different aspect to the project.
Khaled	Khaled is a computer engineer and worked as software developer for three years. He also worked as project manager in a software company for a year. He worked as business development executive in a multinational IT company. He worked as assistant manager in a multinational bank for a year that gave him a good exposure in accounting and finance. He has a masters in marketing. He also worked as consultant in UNDP. Now, he is doing MBA in Bentley university with finance concentration.

Resource	Skills / Experience
Rohit	Software Testing Expertise: I worked as a Software testing engineer in project following agile development methodology for four years in health care services.
Professor Topi	Professor Topi is experienced in usability of database and enterprise systems and offers students office hours and by appointment consulting sessions each week.
Bentley Library	The Bentley Library has a variety of books and resources on SQL and database development. Additionally, the library staff has developed research subject guides and are available for consultations.
Course Book	Theory on SQL and database systems
Online Resources	Ad hoc information

Project Risk Analysis

Project issues can stem from many areas. According to an article by Dobsen & Dobsen (Dobson & Dobson, 2012), we have listed five types of risk that may arise during this project:

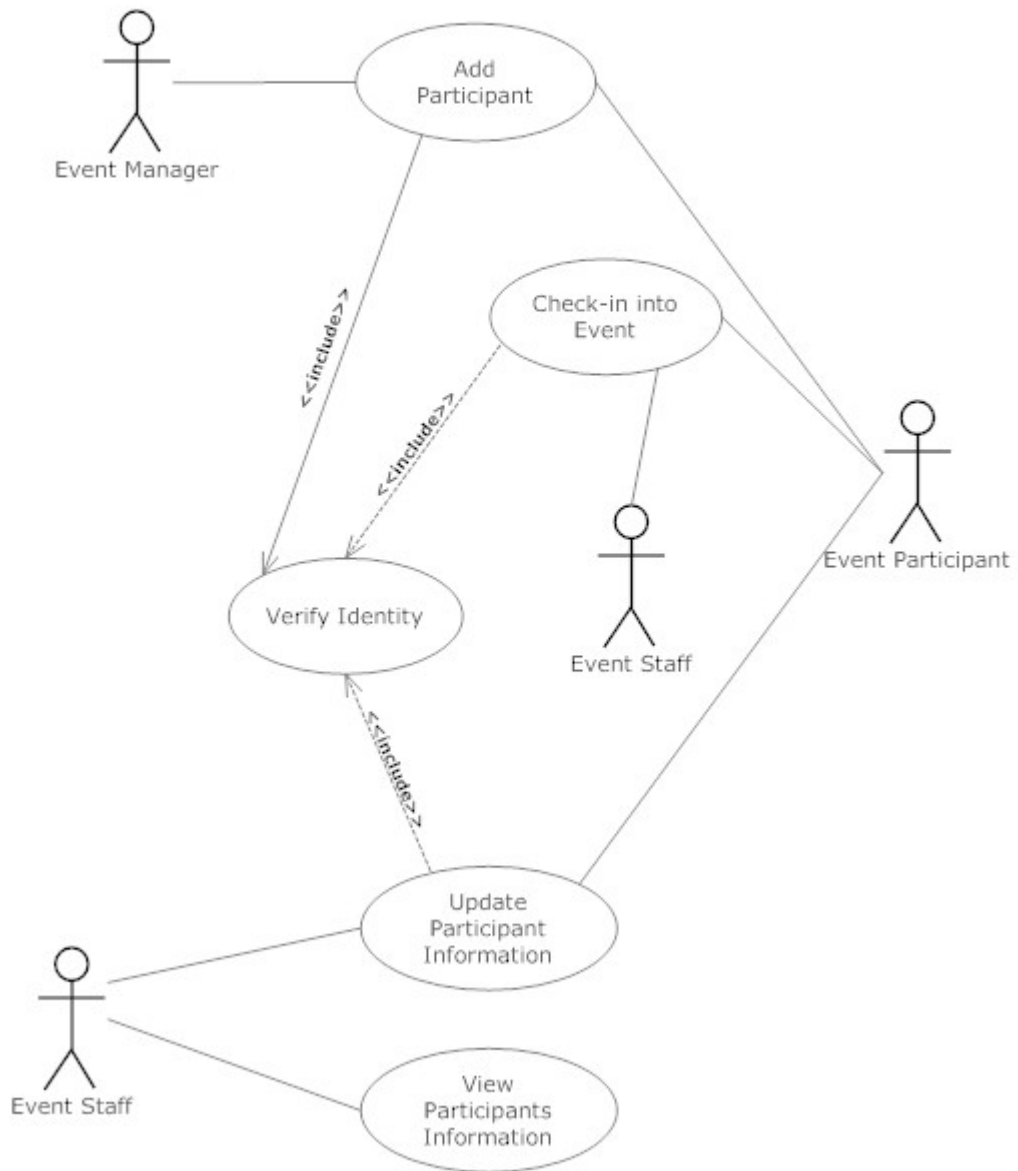
Type of Risk	Examples
Project Risk	<ul style="list-style-type: none"> • Ability to learn technical requirements necessary to build working prototype and deliver project deliverables on time • Ability to incorporate course theory into project
Organizational Risk	<ul style="list-style-type: none"> • Ability of team to maintain communication and internal project deadlines
Customer & User	<ul style="list-style-type: none"> • Reliance on internet connection for instant updates could lead to

Risks	inefficiencies in the case of connection failure <ul style="list-style-type: none"> • Competition of other systems
Personal Risks	<ul style="list-style-type: none"> • Other deadlines and school assignments limiting time available for entire team to meet or complete individual assignments • Impact of additional workload to stress and work-life balance • Ability of team to work cohesively and communicate
Long Term Risks	<ul style="list-style-type: none"> • Lasting negative relationships within the team • Poor grade in course • Need for recurring customers to succeed

Use Case Diagram

Use cases provide developers with an overall view of user expectations without technical details. They contain three major points: an actor that initiates the event, the event that triggers the use case, and a use case that performs the actions triggered by the event (Topi, 2013). In the following diagram, the three main actors are the event manager, event participant (guest), and event staff. Event staff can view and update participants' information, while event manager can add new participants.

Use Case Diagram: Event Management



Use Case Scenarios

Use case scenarios describe each of the use cases in a structured form. They describe the trigger, the steps of the use case, actors and assumptions. We have described five use cases below.

Use case name: Add Participant	
Area: Event Registration Website, Event Helppo System	
Actor(s): Event manager, Event Participant	
Description: Event Participants register through event website and then Event Manager allows participants to the event and update the primary database with the guest list and related information for a specific event.	
Triggering Event: A participant registers for an event.	
Trigger Type: External	
Steps Performed (Main Path):	Information for Steps:
1. Probable event participant registers for any event.	Event participant information detail.
2. Event participant pays the event registration fee.	Event participant's credit card information.
3. Event manager gets a notification of the new event participant.	Event participant information and payment detail.
4. Event manager verifies event participant information integrity.	Event participant's detail.
5. Event manager updates event participant's information in his local database.	Event participant's detail.
6. Event manager allocates seat for the event participant.	Event seat allocation information.
7. Event manager logs into the Even Helppo system.	Event manager's username and password.
8. Event manager uploads the new event participant information into the Even Helppo system.	Event participant's detail.
9. Event manager receives confirmation notification.	Confirmation page.
10. Event manager sends greetings/ confirmation to the event participant with event details.	Event participant's email id.
Preconditions: The participant has successfully registered with valid email id and already paid for the event.	
Post conditions: The event participant has successfully been added to the event.	
Assumption: There is a separate event web page for every event. Event participants can register through that event web page, where they can register with valid email id and pay the event fee. Event Manager has access to the Event Helppo application.	
Requirements Met: Allow event participants to register and pay for the event. Allow Event manager to update event information regarding customers.	
Outstanding Issues: 1. Should an event participant use same email address to register for multiple participants? 2. How many tickets one participant can buy with same email address?	
Priority: High	
Risk: High	

Use case name: View Participants Information	
Area: Event Helppo System	
Actor(s): Event Staff	
Description: Event staff views participant information (contact detail, dietary information, seat number, VIP status, etc.)	
Triggering Event: Participants query on any specific information (seat no, dietary information, etc.). Other event staff's query on any participant's information.	
Trigger Type: External	
Steps Performed (Main Path):	Information for Steps:
1. Event staff logs in to the system.	Event staff username and password
2. Event staff enters keyword for searching the event database.	Search keyword
3. Matching results are displayed.	Matching results.
4. If there is no match, event staff enters another keyword.	Changed keyword.
5. If there are more than one result event staff asks for some other specifics.	Other specific information.
6. Event staff selects the right participant.	Matched participant record.
7. Event staff views the information.	Participant information.
8. Event staff delivers the information.	Participant information.
9. Event staff goes back to the event home/ keyword search page.	Application interface.
Preconditions: Event participant's information are stored into the system prior to the event.	
Post conditions: Event participant's information are viewed and delivered to party that initiated the query.	
Assumption: Every Registration staff has electronic device (i.e. iPod, tablet, smartphone) and they have access to the system.	
Requirements Met: Allow event staffs to view the participant information.	
Outstanding Issues: If the customer information is not found in the system, what event staff should do?	
Priority: Medium	
Risk: Low	

Use case name: Update Participant Information	
Area: Event Helppo System	
Actor(s): Event Staff, Event Participant	
Description: Event participant provides information to event staff. Event staff updates participant information (contact detail, dietary information, etc.)	
Triggering Event: Participant asks to update any specific information (i.e., dietary information). Event staff asks participant to verify information.	
Trigger Type: External	
Steps Performed (Main Path):	Information for Steps:
1. Event staff enters keyword for searching the event database.	Search keyword
2. Matching results are displayed.	Matched result.
3. If there is no match, event staff enters another keyword.	Changed keyword.
4. If there are more than one result event staff asks for some other specifics.	Other specific information.
5. Event staff selects the right participant.	Matched participant record.
6. Event staff views the information.	Participant information.
7. Event staff asks for the updated information.	Event participant's updated information.
8. Event staff change necessary information.	Event participant's updated information.
9. Event staff receives a confirmation message of successful update.	Confirmation message.
10. Event staff verbally let the event participant know the successful update.	Confirmation message.
11. Event staff goes back to the event home/ keyword search page.	Application interface.
Preconditions: The event participant's information are stored into the system prior to the event.	
Post conditions: Event participant's information are updated.	
Assumption: Every Registration staff is already logged in into the system.	
Requirements Met: Allow event staffs to update the participant information.	
Outstanding Issues: 1. How many times any event participant should be allowed to update information? 2. If event participant wants to change seat, what event staff should do?	
Priority: High	
Risk: High	

Use case name: Check-in into Event	
Area: Event Helppo System	
Actor(s): Event Staff, Event Participant	
Description: Event participant gives necessary details to Event staff for check-in. Event staff views participant information (contact detail, dietary information, seat number, VIP status, etc.) and then confirm check-in and let the customer know his/her seat number.	
Triggering Event: Event participant comes to the event and wants to check-in.	
Trigger Type: External	
Steps Performed (Main Path):	Information for Steps:
1. Event staff enters keyword for searching the event database.	Search keyword
2. Matching results are displayed.	Matched result.
3. If there is no match, event staff enters another keyword.	Changed keyword.
4. If there are more than one result event staff asks for some other specifics.	Other specific information.
5. Event staff selects the right participant.	Matched participant record.
6. Event staff views the information.	Participant information.
7. Event staff asks for identification.	Event participant's identification.
8. Event staff checks identification.	Event participant's identification.
9. Event staff check-in the event participant into the system.	Participant information.
10. Event staff receives confirmation.	Confirmation message.
11. Event staff verbally let the event participant know the seat number.	Event participant's seat number.
12. Event staff goes back to the event home/ keyword search page.	Application interface.
Preconditions: The event participant's information are stored into the system prior to the event.	
Post conditions: Event participant is successfully checked-in into the system.	
Assumption: Every Registration staff is already logged in into the system.	
Requirements Met: Allow event staffs to check-in event participant.	
Outstanding Issues: What the event staff should do if the participant's information is not found in the system?	
Priority: High	
Risk: High	

Use case name: Verify Identity	
Area: Event Webpage, Event Helppo System, Event place	
Actor(s): Event Staff, Event Participant, and Event Manager.	
Description: Event participant gives/ shows necessary information to very his/her identity at the time of registration and at the time of check in.	
Triggering Event: Event participant registers for an event. Event participant comes to the event and wants to check-in.	
Trigger Type: External	
Steps Performed (Main Path):	Information for Steps:
1. Event participant enters information into the event registration system or gives information to the event staff for check-in or update information.	Event participant's identification.
2. In the registration system, it validates customer's information through local database over the internet and the event staff checks the identification physically (if necessary) or validates the information using the remote event database though the system.	Event participant's identification.
3. If the event participant passes the validity check, he or she will receive an confirmation message from the system or verbal confirmation message/ greetings from the event staff.	Confirmation.
Preconditions: The event participant has valid id, mail and email address.	
Post conditions: Event participant is successfully verified.	
Assumption: Every Registration staff is already logged in into the system.	
Requirements Met: Allow event participant to register, check-in and update information.	
Outstanding Issues: 1. How many layers of validity check is required? 2. If any event participants fail to show valid id in the event, what event staff should do?	
Priority: High	
Risk: High	

Activity Diagram

An activity diagram shows the sequence of activities and the process' flow. One of the benefits of an activity diagram is a clearer understanding of the underlying processes. It shows the dependencies of activities and allows users to understand how one activity is triggered by a previous one.

Process Description for Activity Diagram

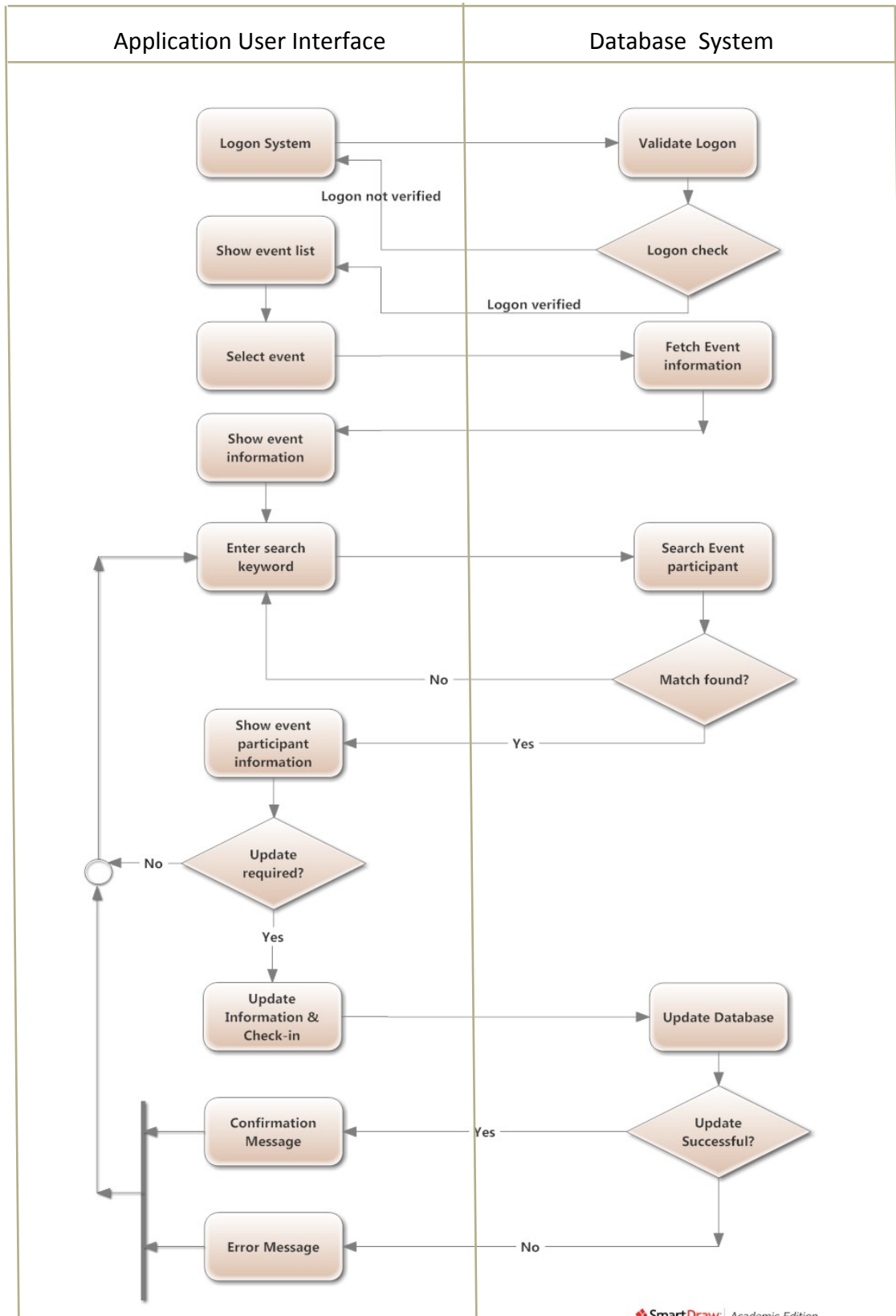
At the beginning of any event, event staff enter their valid usernames and passwords into the system. The system sends a confirmation message for each successful login. At the

same time, the database system shows an event list that is associated with that event staff. There can be more than one event list if the same event staff is responsible for multiple events on different dates. There can be multiple events on a single date as well. The event staff then selects the event that is appropriate for the date and time.

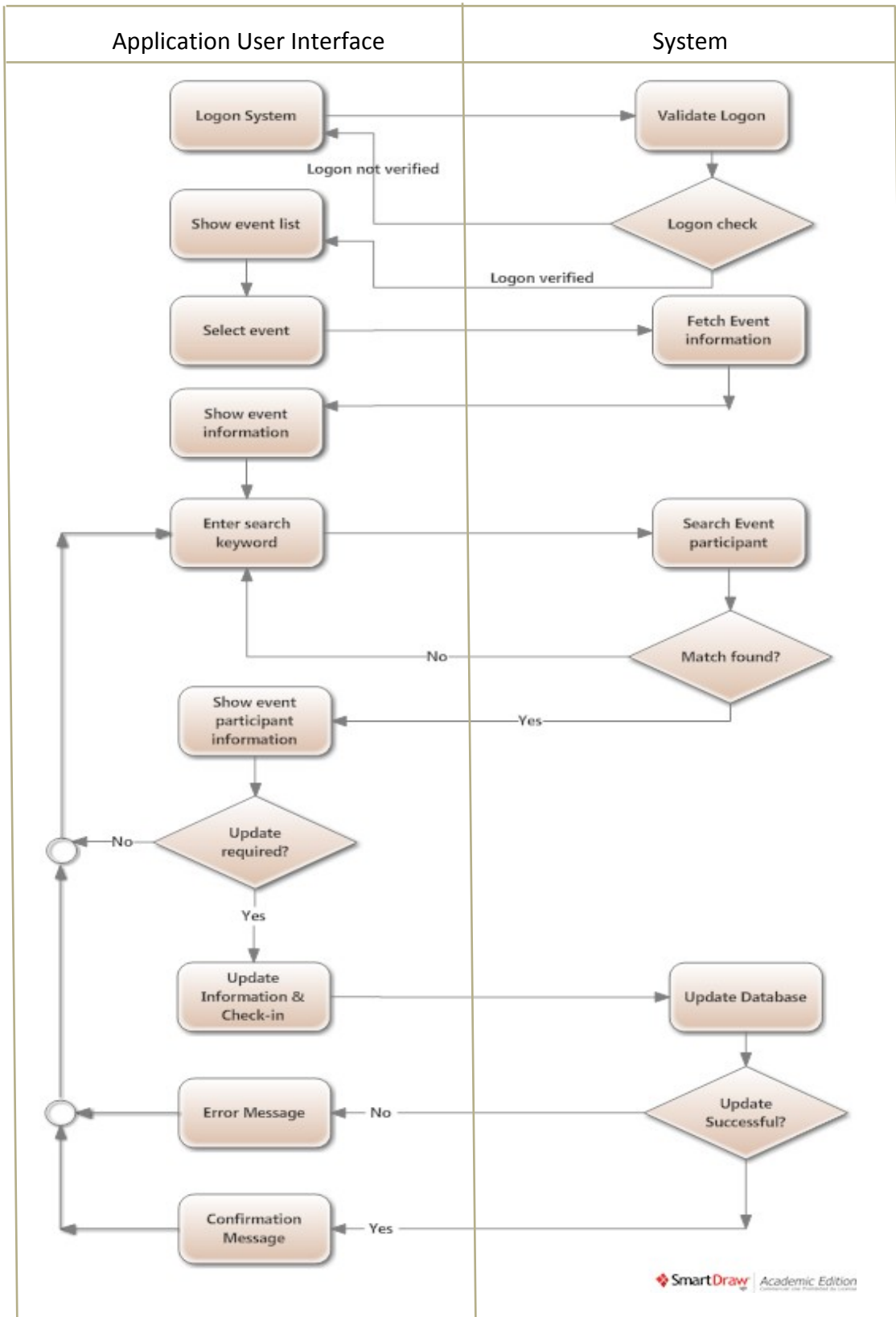
As soon as the event staff selects the event, the entire event related information is fetched from the database system to the application interface. The event staff is directed to the event home page where he can do necessary operations. On the event page the event staff can search with one or multiple keywords to search for any event participant. In any event, event participants will arrive and the event staff will be in the entrance to check-in the event participants. When any event participant arrives, the event staff will ask for the participant's information to look for him/her in the event database system. If he finds him/her in the database system, then the event staff confirms the event participant's information with seat number and asks if there will be any update of information (dietary information, seat, contact information, etc.).

The event staff updates the event participant's status as checked-in and sends other information update (if any) to database system for update. The event participant receives update confirmation or error message (if there is any problem). After that, event staff goes back to the event information search page. New event participant arrives and event staff keeps continuing the event participant's check-in process.

Activity Diagram (Version one)



Activity Diagram (Modified after feedback)



Change in Activity Diagram

In the first version of the activity diagram, if the update was not successful, we used parallel activities. However, those activities should not be parallel. Based on the feedback from the professor, we have changed the activity diagram accordingly. There was a minor change in name of the swim lane as well. We changed it from 'Database System' to 'System'.

Enhanced Entity-Relationship Model

The purpose of the enhanced entity-relationship model is to develop a foundation for the application's systems development process and define the rules that govern the structure of the application's data. It is a detailed representation of the data utilized in the application, and it consists of the various business entities, relationships between those entities, and attributes of both the entities and their relationships.

Entities are the people and events about which users of our application wish to maintain data. In our diagram, these include Event Staff, Event, Visit Information and Guest. Since Event and Guest each include entity types that are similar, but also have one or more distinct attributes, they have been broken down into additional subtypes. Relationships represent the associations or interactions between the entities, and cardinalities are used to demonstrate the minimum and maximum number of instances of an entity that can be associated with instances of another entity. For example, the relationship between Event Staff and Event is "many to many." An event staff member can be assigned to multiple events and a single event can utilize multiple staff members. The minimum cardinalities for this relationship demonstrate that each event staff member must be assigned to at least one event and each event must employ at least one staff member. Attributes are the properties or characteristics of these entities or relationships that are of interest to our users. For Event Staff, these include name, address, phone number and email.

Assumptions for Conceptual Data Modeling

'EventHelppo' gathers data regarding the event and its details such as Event Name, Event date, Event Location (Street, City, State, Zip, and Country), Event Start, End time, Event

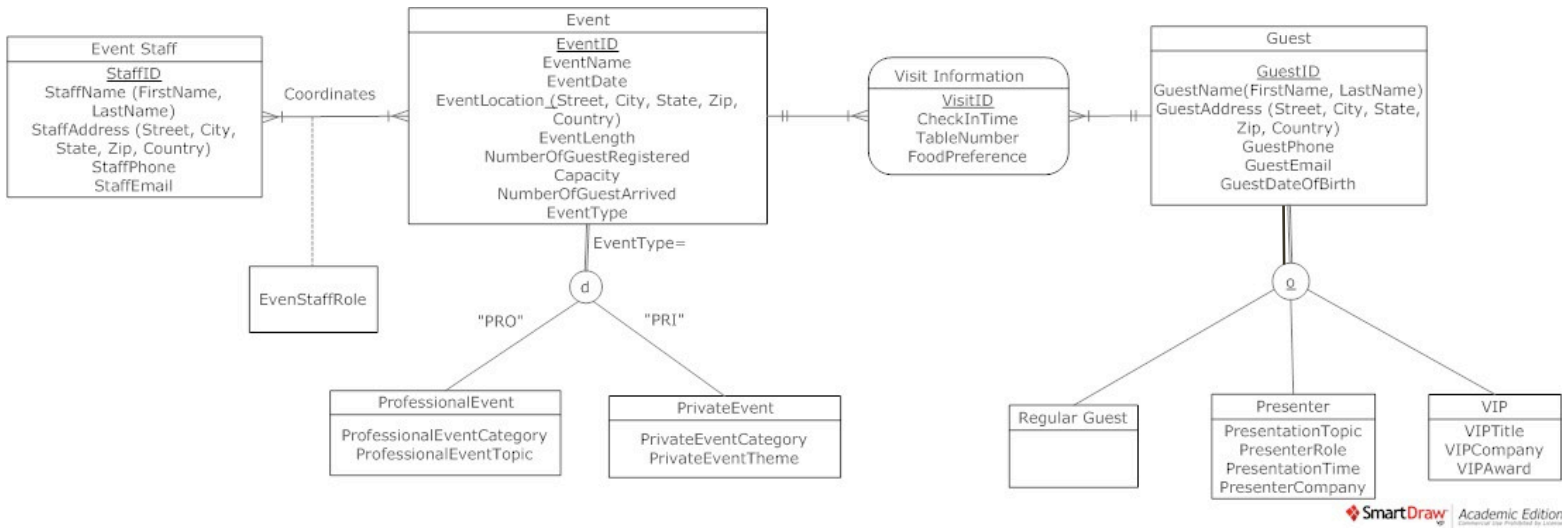
length, Number of guests' registered, Event capacity and Event Type. The history of events and their data needs to be saved for data analysis. Hence, a unique identifier named 'Event ID' identifies each event. The events can be either a professional or a private event but not both. Apart from the details mentioned above regarding the event further information on 'Event Category' (Seminar or Conference) and 'Event Topic' needs to be gathered for professional events. Information regarding 'Event Category' (Birthday, Gala or Wedding) and 'Event Theme' needs to be gathered for private events.

Event staff coordinates events. Staff details such as first and last name, Address (street, city, state, zip and country), phone number and email address needs to be stored into the database. Information about each staff member is assigned with a unique identifier called 'Staff ID'. The roles of event staff changes from event to event. Every staff member is allocated to at least one event and every event must have at least one staff member.

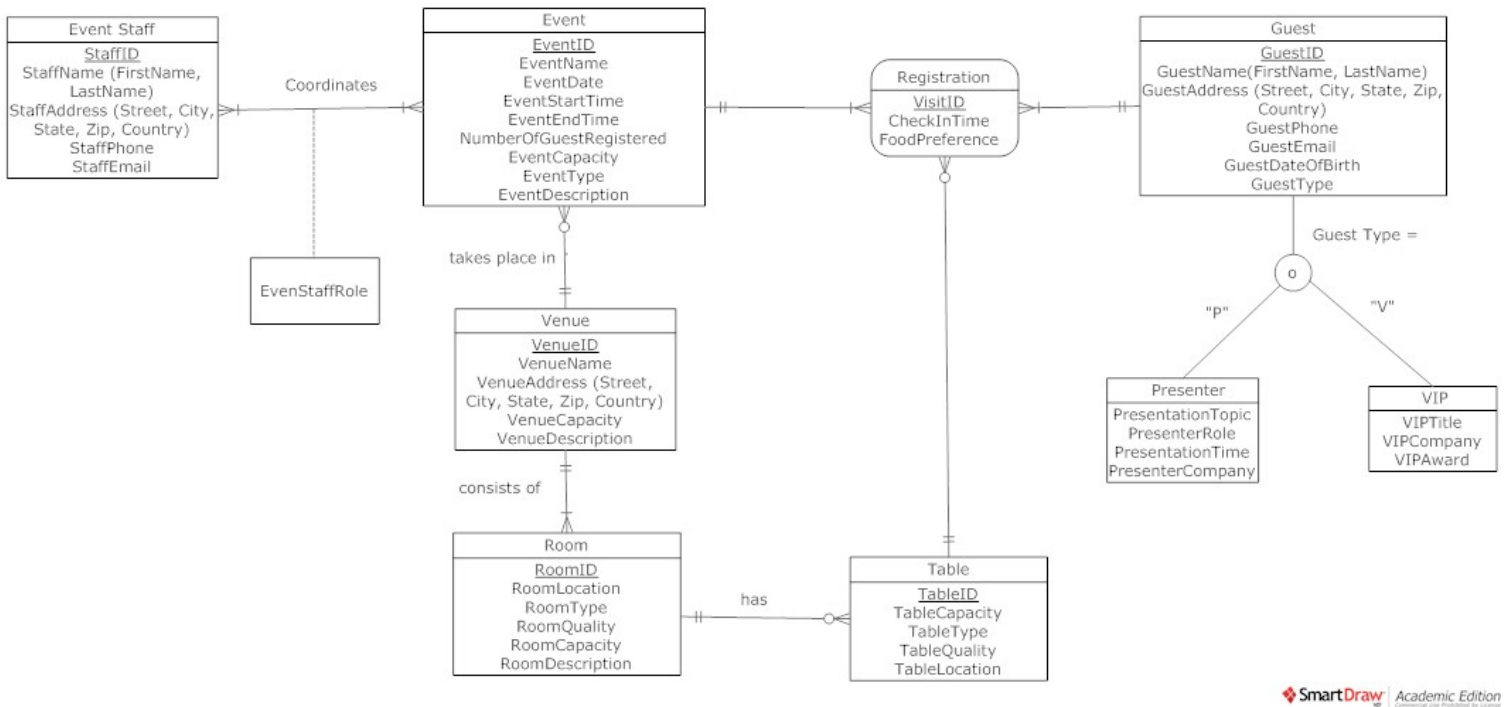
A database of Guests registered to events is maintained with information about Guest's first and last name, address (street, city, state, zip, and country), phone number, email address and date of birth. All the information about each guest is differentiated through a unique identifier called 'Guest ID'. Guests are further differentiated as Regular guests, Presenter or a VIP. Apart from the data acquired for the guests, presenter will have presentation topic, their role, presentation time and their company as attributes while VIPs have their title, their company and VIP award if an award is given. A guest can belong to any of these categories simultaneously.

All the information about guests stored from the database is used to store the visit information and to check the status of the guests for each event. Visit information has details such as check-in status, check-in time, table number, and food preference. Visit information of each guest is differentiated through a unique identifier called 'Visit ID'. Each instance of Visit ID is attributed to only one guest, and each guest is attributed to one Visit ID.

EER Diagram (Version one)



EER Diagram (Modified after feedback)



Change in Conceptual Data Model

After receiving feedback, we made some changes to our EER diagram. Most noticeable, we have removed the event category (professional and private), and replaced it with an attribute in the event entity named as 'Event Type.' We also removed one guest type, 'Regular Guest,' as it had no attribute. Therefore, besides the two main categories, there can be other type of guests. Another important change was the addition of Table, Room and Venue entities to keep a record of the table, room and venue information stored into the database.

Relational Data Modeling

The purpose of the relational data model is to represent data in the form of tables. It consists of three components: data structure, data manipulation and data integrity. Each table consists of a set of attributes, organized into columns, and each row corresponds to a record for a single entity. A primary key is used to uniquely identify each row in a relation. For example, the VIP guest table includes columns for Guest ID, VIP Title, VIP Company and VIP Award. Each individual VIP guest will have a unique Guest ID and be listed in each row of that table. In certain situations, a primary key may consist of more than one attribute and a composite key will be utilized. Additionally, foreign keys are used to establish the relationship between two tables.

Converting EER Diagram into Relational Data Model

EVENTSTAFF (StaffID, StaffFirstName, StaffLastName, StaffStreet, StaffCity, StaffState, StaffZip, StaffCountry, StaffPhone, StaffEmail)

VENUE (VenueID, VenueName, VenueStreet, VenueCity, VenueState, VenueZip, VenueCountry, VenueCapacity, VenueDescription)

ROOM (RoomID, RoomLocation, RoomType, RoomQuality, RoomCapacity, RoomDescription, VenueID)

VenueID references VENUE(VenueID)

TABLE (TableID, TableCapacity, TableType, TableQuality, TableLocation, RoomID)

RoomID references Room(RoomID)

EVENT (EventID, EventName, EventDate, EventStartTime, EventEndTime, NumberOfGuestReg, EventCapacity, EventType, EventDescription, VenueID)
 VenueID references VENUE(VenueID)

COORDINATION (StaffID, EventID, EventStaffRole)
 StaffID references EVENTSTAFF(StaffID)
 EventID references EVENT (EventID)

GUEST (GuestID, GuestFirstName, GuestLastname, GuestStreet, GuestCity, GuestState, GuestZip, GuestCountry, GuestPhone, GuestEmail, GuestDateOfBirth, GuestType)

PRESENTER (GuestID, PresentationTopic, PresenterRole, PresentationTime, PresenterCompany)
 GuestID references GUEST (GuestID)

VIP (GuestID, VIPTitle, VIPCompany, VIPAward)
 GuestID references GUEST (GuestID)

REGISTRATION (VisitID, GuestID, EventID, TableID, CheckInTime, FoodPreference)
 GuestID references GUEST (GuestID)
 EventID references EVENT (EventID)
 TableID references TABLE (TableID)

MySQL

MySQL is commonly used to define and query relational database management systems. This standardized language reduces training costs, enhances productivity, prolongs application portability and longevity, and helps cross-system communication. While creating our EventHelppo application, transferring the Relational Model into a database through SQL is our next step. We create the necessary tables according to the Relational Model such as 'Coordination'. These tables include important information linking event staff and event, under three columns: Staff ID, Event ID and Event Staff Role.

MySQL Queries for Table Creation

```
/* Creating EVENTSTAFF table */
```

```
create table EVENTSTAFF (
    StaffID          VARCHAR(10) NOT NULL,
    StaffFirstName   VARCHAR(30),
    StaffLastName    VARCHAR(30),
```

```

        StaffStreet          VARCHAR(30),
        StaffCity            VARCHAR(20),
        StaffState           VARCHAR(20),
        StaffZip             VARCHAR(10),
        StaffCountry         VARCHAR(30),
        StaffPhone           VARCHAR(15),
        StaffEmail           VARCHAR(30),
        constraint EVENTSTAFF_PK PRIMARY KEY (StaffID));

```

/* Creating VENUE table */

```

create table VENUE (
        VenueID              VARCHAR(10) NOT NULL,
        VenueName            VARCHAR(30),
        VenueStreet          VARCHAR(30),
        VenueCity            VARCHAR(20),
        VenueState           VARCHAR(20),
        VenueZip             VARCHAR(10),
        VenueCountry         VARCHAR(30),
        VenueCapacity        INTEGER,
        VenueDescription      VARCHAR(50),
        constraint VENUE_PK PRIMARY KEY (VenueID));

```

/* Creating ROOM table */

```

create table ROOM (
        RoomID              VARCHAR(10) NOT NULL,
        RoomLocation         VARCHAR(30),
        RoomType             VARCHAR(20),
        RoomQuality          VARCHAR(20),
        RoomCapacity         INTEGER,
        RoomDescription      VARCHAR(50),
        VenueID             VARCHAR(10),
        constraint ROOM_PK PRIMARY KEY (RoomID),
        constraint ROOM_FK FOREIGN KEY (VenueID) references
VENUE(VenueID));

```

/* Creating EVENTTABLE table */

```

create table EVENTTABLE (
        TableID             VARCHAR(10) NOT NULL,
        TableLocation        VARCHAR(30),
        TableType            VARCHAR(30),
        TableQuality         VARCHAR(20),

```

```

        TableCapacity    INTEGER,
        RoomID           VARCHAR(10),
        constraint EVENTTABLE_PK PRIMARY KEY (TableID),
        constraint  EVENTTABLE_FK FOREIGN KEY (RoomID) references
ROOM(RoomID));

```

```
/* Creating EVENT table */
```

```

create table EVENTT (
        EventID          VARCHAR(10) NOT NULL,
        EventName        VARCHAR(30),
        EventDate        Date,
        EventStartTime   Time,
        EventEndTime     Time,
        NumberOfGuestReg INTEGER,
        EventCapacity    INTEGER,
        EventType        ENUM ('PRI','PRO'),
        EventDescription  VARCHAR(50),
        VenueID          VARCHAR(10),
        constraint EVENTT_PK PRIMARY KEY (EventID),
        constraint  EVENTT_FK FOREIGN KEY (VenueID) references
VENUE(VenueID));

```

```
/* Creating COORDINATION table */
```

```

create table COORDINATION (
        EventID          VARCHAR(10) NOT NULL,
        StaffID          VARCHAR(10) NOT NULL,
        EventStaffRole   VARCHAR(30),
        constraint COORDINATION_PK PRIMARY KEY (EventID, StaffID),
        constraint  COORDINATION_FK1 FOREIGN KEY (EventID) references
EVENTT(EventID),
        constraint  COORDINATION_FK2 FOREIGN KEY (StaffID) references
EVENTSTAFF(StaffID));

```

```
/* Creating GUEST table */
```

```

create table GUEST (
        GuestID          VARCHAR(10) NOT NULL,
        GuestFirstName   VARCHAR(30),
        GuestLastName    VARCHAR(30),
        GuestStreet       VARCHAR(30),
        GuestCity         VARCHAR(30),

```

```

        GuestState      VARCHAR(20),
        GuestZip        VARCHAR(10),
        GuestCountry    VARCHAR(30),
        GuestPhone      VARCHAR(20),
        GuestEmail      VARCHAR(30),
        GuestDateOfBirth DATE,
        GuestType       ENUM ('R','V','P'),
        constraint GUEST_PK PRIMARY KEY (GuestID));

```

/* Creating PRESENTER table */

```

create table PRESENTER (
        GuestID          VARCHAR(10) NOT NULL,
        PresentationTopic VARCHAR(30),
        PresenterRole    VARCHAR(30),
        PresentationTime Time,
        PresenterCompany VARCHAR(30),
        constraint PRESENTER_PK PRIMARY KEY (GuestID),
        constraint PRESENTER_FK FOREIGN KEY (GuestID) references
GUEST(GuestID));

```

/* Creating VIP table */

```

create table VIP (
        GuestID          VARCHAR(10) NOT NULL,
        VIPTitle         VARCHAR(30),
        VIPCompany       VARCHAR(30),
        VIPAward         VARCHAR(30),
        constraint VIP_PK PRIMARY KEY (GuestID),
        constraint VIP_FK FOREIGN KEY (GuestID) references GUEST(GuestID));

```

/* Creating REGISTRATION table */

```

create table REGISTRATION (
        VisitID         VARCHAR(10) NOT NULL,
        GuestID         VARCHAR(10) NOT NULL,
        EventID         VARCHAR(10) NOT NULL,
        TableID         VARCHAR(10) NOT NULL,
        CheckinTime     Time,
        FoodPreference  VARCHAR(30),
        constraint REGISTRATION_PK PRIMARY KEY (VisitID),

```



```

constraint REGISTRATION_FK1 FOREIGN KEY (GuestID) references
GUEST(GuestID),
constraint REGISTRATION_FK2 FOREIGN KEY (EventID) references EVENTT
(EventID),
constraint REGISTRATION_FK3 FOREIGN KEY (TableID) references
EVENTTABLE(TableID));

```

MySQL Data Entry Queries

/ Entering values into Eventstaff table */*

```

insert into eventstaff values('S1001','SF1','SL1','ST
#1','City1','MA','01111','USA','1234567891','s1@test.com');
insert into eventstaff values('S1002','SF2','SL2','ST
#2','City2','AL','01112','USA','1234567981','s2@test.com');
insert into eventstaff values('S1003','SF3','SL3','ST
#3','City3','AK','01113','USA','1234576891','s3@test.com');
insert into eventstaff values('S1004','SF4','SL4','ST
#4','City4','CA','01114','USA','1234657891','s4@test.com');
insert into eventstaff values('S1005','SF5','SL5','ST
#5','City5','CT','01115','USA','1235467891','s5@test.com');
insert into eventstaff values('S1006','SF6','SL6','ST
#6','City6','FL','01116','USA','1243567891','s6@test.com');
insert into eventstaff values('S1007','SF7','SL7','ST
#7','City7','MA','01117','USA','1434567891','s7@test.com');
insert into eventstaff values('S1008','SF8','SL8','ST
#8','City8','AK','01118','USA','1236767891','s8@test.com');
insert into eventstaff values('S1009','SF9','SL9','ST
#9','City9','AL','01119','USA','1238867891','s9@test.com');
insert into eventstaff values('S10010','SF10','SL10','ST
#10','City10','CT','01110','USA','1212367891','s10@test.com');

```

/ Entering values into Venue table */*

```

insert into venue values('V1001','V1','VST #1','City1','MA','01111','USA',100,'Smaller venue');
insert into venue values('V1002','V2','VST #2','City1','MA','01111','USA',200,'Smaller venue');
insert into venue values('V1003','V3','VST #3','City4','MA','01117','USA',300,'Nice venue');
insert into venue values('V1004','V4','VST #4','City2','AL','01112','USA',400,'Good venue');
insert into venue values('V1005','V5','VST #5','City3','AK','01113','USA',500,'Beautiful venue');
insert into venue values('V1006','V6','VST #6','City3','CT','01111','USA',600,'Medium venue');

```

```

insert into venue values('V1007','V7','VST #7','City4','CA','01114','USA',700,'Big venue');
insert into venue values('V1008','V8','VST #8','City5','FL','01111','USA',800,'Bigger venue');
insert into venue values('V1009','V9','VST #9','City6','AL','01119','USA',900,'Wedding venue');
insert into venue values('V10010','V10','VST #10','City6','AK','01118','USA',1000,'Seminar
venue');

```

```

/* Entering values into Room table */

```

```

insert into Room values('R1001','1st Floor','Medium','Top',30,'Smaller Room', 'V1001');
insert into Room values('R1002','1st Floor','Medium','Medium',50,'Smaller Room', 'V1002');
insert into Room values('R1003','2nd Floor','Large','Top',400,'Huge Room', 'V1009');
insert into Room values('R1004','3rd Floor','Small','Low',100,'Birthday Room', 'V1005');
insert into Room values('R1005','1st Floor','Large','Top',300,'Wedding Room', 'V1008');
insert into Room values('R1006','1st Floor','Large','Top',350,'Huge Room', 'V1007');
insert into Room values('R1007','2nd Floor','Medium','Medium',20,'Tiny Room', 'V1005');
insert into Room values('R1008','3rd Floor','Large','Top',30,'Private Room', 'V1003');
insert into Room values('R1009','2nd Floor','Large','Medium',90,'Seminar Room', 'V1003');
insert into Room values('R1010','1st Floor','Large','Top',250,'Wedding Room', 'V1006');

```

```

/* Entering values into EventTable table */

```

```

insert into EventTable values('T1001','center', 'Circle','Top',10,'R1001');
insert into EventTable values('T1002','south east corner', 'Triangle','Medium',9,'R1001');
insert into EventTable values('T1003','center', 'Triangle','Top',7,'R1002');
insert into EventTable values('T1004','north west corner', 'Rectangle','Medium',4,'R1002');
insert into EventTable values('T1005','south', 'Circle','Top',6,'R1001');
insert into EventTable values('T1006','center', 'Circle','Top',10,'R1003');
insert into EventTable values('T1007','center', 'Triangle','Medium',3,'R1004');
insert into EventTable values('T1008','north corner', 'Rectangle','Low',8,'R1003');
insert into EventTable values('T1009','west corner', 'Circle','Low',10,'R1002');
insert into EventTable values('T10010','south corner', 'Rectangle','Top',4,'R1004');

```

```

/* Entering values into Eventt table */

```

```

insert into eventt values ('E1001', 'E1', '2013-12-
11', '10:00:00', '17:00:00', 35, 60, 'PRO', 'Seminar', 'V1001');
insert into eventt values ('E1002', 'E2', '2013-12-
12', '11:00:00', '18:00:00', 25, 30, 'PRI', 'Wedding', 'V1002');
insert into eventt values ('E1003', 'E3', '2013-12-
13', '10:00:00', '19:00:00', 20, 35, 'PRO', 'Seminar', 'V1002');
insert into eventt values ('E1004', 'E4', '2013-12-
14', '11:00:00', '17:00:00', 350, 600, 'PRI', 'Birthday', 'V1009');
insert into eventt values ('E1005', 'E5', '2013-12-
15', '12:00:00', '20:00:00', 335, 460, 'PRO', 'Workshop', 'V1009');
insert into eventt values ('E1006', 'E6', '2013-12-
16', '10:00:00', '21:00:00', 25, 50, 'PRI', 'Gala', 'V1001');

```

```

insert into eventt values ('E1007', 'E7', '2013-12-
17', '13:00:00', '22:00:00', 135, 260, 'PRO', 'Seminar', 'V1009');
insert into eventt values ('E1008', 'E8', '2013-12-
18', '10:00:00', '16:00:00', 95, 200, 'PRI', 'Wedding', 'V1005');
insert into eventt values ('E1009', 'E9', '2013-12-
19', '14:00:00', '17:00:00', 155, 300, 'PRO', 'Seminar', 'V1005');
insert into eventt values ('E1010', 'E10', '2013-12-
20', '15:00:00', '18:00:00', 190, 260, 'PRO', 'Seminar', 'V1005');

```

/ Entering values into Coordination table */*

```

insert into COORDINATION values ('E1001', 'S1001', 'Manager');
insert into COORDINATION values ('E1001', 'S1002', 'Check-in Staff');
insert into COORDINATION values ('E1001', 'S1003', 'Check-in Staff');
insert into COORDINATION values ('E1002', 'S1004', 'Manager');
insert into COORDINATION values ('E1002', 'S1005', 'Check-in Staff');
insert into COORDINATION values ('E1002', 'S1006', 'Check-in Staff');
insert into COORDINATION values ('E1003', 'S1007', 'Manager');
insert into COORDINATION values ('E1003', 'S1008', 'Check-in Staff');
insert into COORDINATION values ('E1003', 'S1009', 'Check-in Staff');
insert into COORDINATION values ('E1003', 'S10010', 'Check-in Staff');

```

/ Entering values into Guest table */*

```

insert into guest values ('G1001', 'GF1', 'GL1', 'GST#1', 'Gcity1', 'MA',
'01111', 'USA', '123456712', 'g1@gtest.com', '1981-10-11', 'R');
insert into guest values ('G1002', 'GF2', 'GL2', 'GST#2', 'Gcity2', 'AL',
'01112', 'USA', '133456712', 'g2@gtest.com', '1982-11-12', 'V');
insert into guest values ('G1003', 'GF3', 'GL3', 'GST#3', 'Gcity3', 'FL',
'01116', 'USA', '123446712', 'g3@gtest.com', '1983-12-28', 'P');
insert into guest values ('G1004', 'GF4', 'GL4', 'GST#4', 'Gcity4', 'AK',
'01113', 'USA', '123556712', 'g4@gtest.com', '1984-11-26', 'V');
insert into guest values ('G1005', 'GF5', 'GL5', 'GST#5', 'Gcity5', 'CA',
'01114', 'USA', '123466712', 'g5@gtest.com', '1985-11-24', 'R');
insert into guest values ('G1006', 'GF6', 'GL6', 'GST#2', 'Gcity1', 'MA',
'01117', 'USA', '123457712', 'g6@gtest.com', '1986-10-20', 'P');
insert into guest values ('G1007', 'GF7', 'GL7', 'GST#4', 'Gcity5', 'AL',
'01119', 'USA', '123456702', 'g7@gtest.com', '1987-09-21', 'V');
insert into guest values ('G1008', 'GF8', 'GL8', 'GST#8', 'Gcity7', 'FL',
'01121', 'USA', '123456710', 'g8@gtest.com', '1988-06-15', 'R');
insert into guest values ('G1009', 'GF9', 'GL9', 'GST#9', 'Gcity8', 'CT',
'01110', 'USA', '123456722', 'g9@gtest.com', '1970-03-18', 'P');
insert into guest values ('G1010', 'GF10', 'GL10', 'GST#1', 'Gcity1', 'CT',
'01115', 'USA', '123956712', 'g10@gtest.com', '1972-12-22', 'R');

```

/ Entering values into Presenter table */*

```
insert into PRESENTER values('G1003', 'environment', 'keynote', '16:00:00', 'Comp1');
insert into PRESENTER values('G1006', 'fashion', 'speaker', '15:40:00', 'Comp2');
insert into PRESENTER values('G1009', 'business', 'Anchor', '13:00:00', 'Comp3');
```

```
/* Entering values into VIP table */
```

```
insert into VIP values ('G1002', 'TitleA', 'Vcomp1', 'Award1');
insert into VIP values ('G1004', 'TitleB', 'Vcomp2', 'Award2');
insert into VIP values ('G1007', 'TitleC', 'Vcomp3', 'Award3');
```

```
/* Entering values into Registration table */
```

```
insert into registration values('R1001', 'G1001', 'E1001', 'T1001', '00:00:00', 'Veg');
insert into registration values('R1002', 'G1002', 'E1002', 'T1003', '00:00:00', 'non-Veg');
insert into registration values('R1003', 'G1003', 'E1001', 'T1002', '00:00:00', 'Veg');
insert into registration values('R1004', 'G1004', 'E1005', 'T1006', '00:00:00', 'Veg');
insert into registration values('R1005', 'G1005', 'E1008', 'T1007', '00:00:00', 'non-Veg');
insert into registration values('R1006', 'G1006', 'E1007', 'T1008', '00:00:00', 'Veg');
insert into registration values('R1007', 'G1007', 'E1006', 'T1005', '00:00:00', 'Veg');
insert into registration values('R1008', 'G1008', 'E1007', 'T1006', '00:00:00', 'non-Veg');
insert into registration values('R1009', 'G1009', 'E1009', 'T1007', '00:00:00', 'Veg');
insert into registration values('R1010', 'G1010', 'E1003', 'T1009', '00:00:00', 'Veg');
```

```
commit;
```

Application Prototype

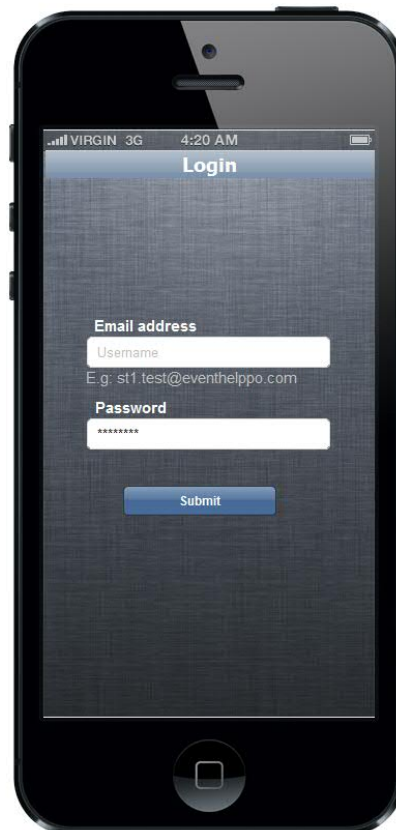
Based on the above database elements, we have created a prototype to display the functionality of the application and the usage of information stored in the database in multiple ways. We used prototyping software called 'Prototyper Pro' software of 'Justimind' company. The application's various screenshots are displayed and explained in the following sections.

Login screens

Start screen



Login Page

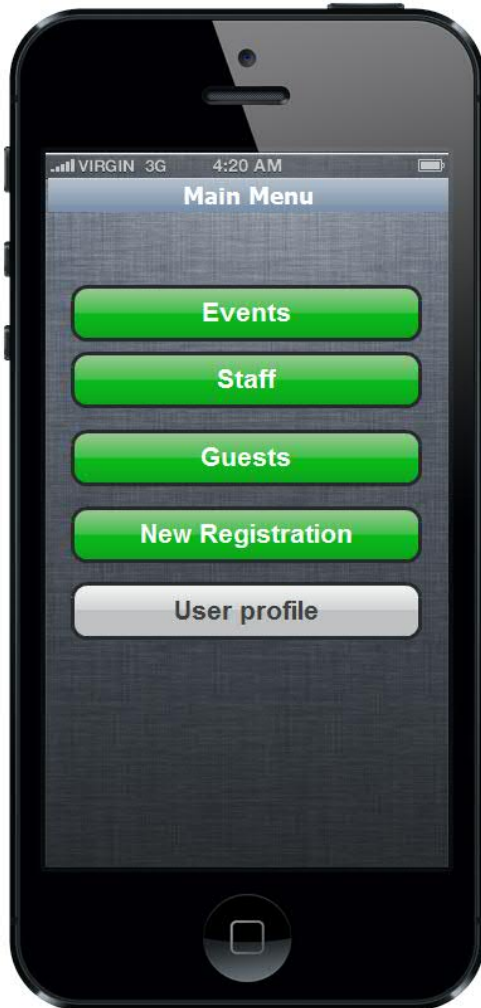


This is the first screen after opening the application. The users (event managers/staff) need to click on the login button to enter the username and password. After entering the username and password, the system will recognize the credentials and while authenticating it

will also check the user permissions where it will display different main menu screens based on Event Staff role i.e. Event Manger or Check-in Staff.

Main Menu Screens

Event Manager Menu



Event Staff Menu

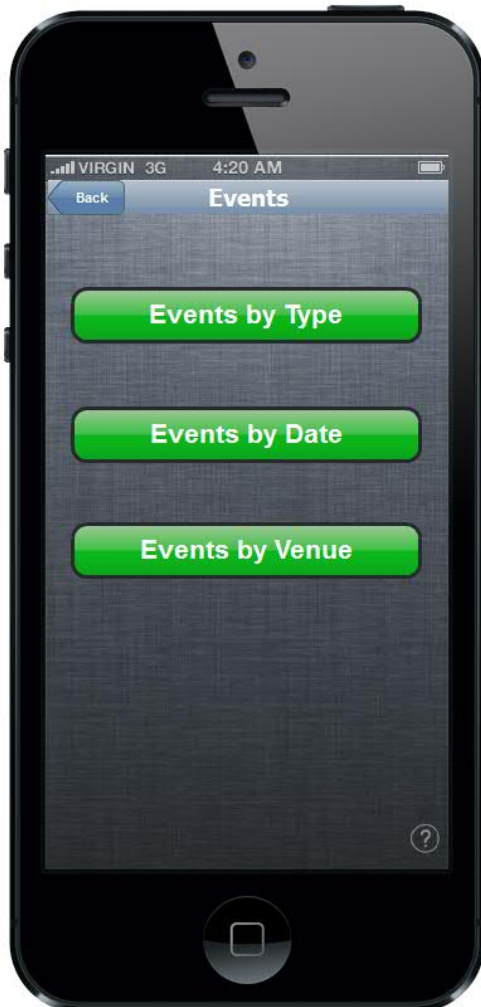


The main menu screens will display information such as the user's name, account information, email and role information through the user profile button. The manager's menu screen will provide easy access to information on different entities such as events, guests, event staff or any new registration of guests of the events. For event staff, only access to the events is provided due to limited permissions. All users will be able to access information about events

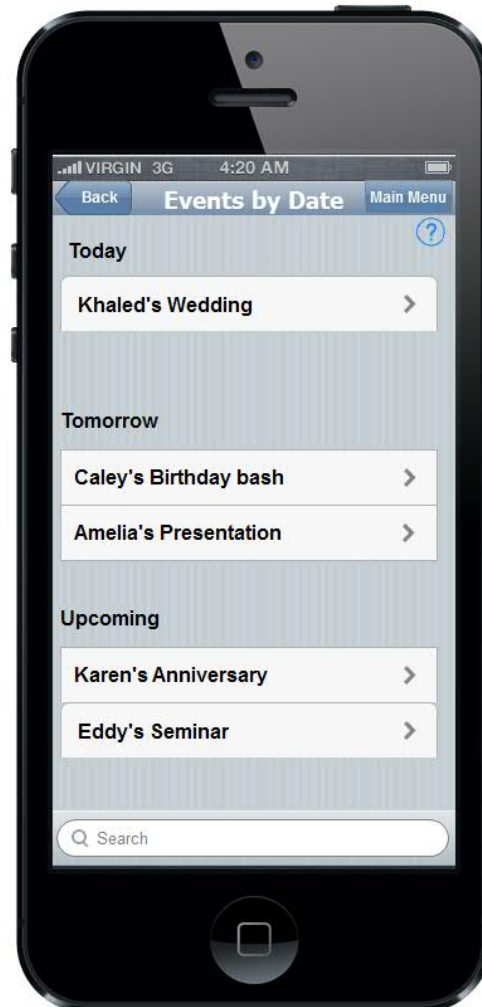
by type, date and venue. Managers can access Staff information by Name, Role or Events. They can also access guest information by Name, Event and Type.

Events Screen

Events Page



Events by Date



Once the event staff clicks on the events button, they will be directed to the events page where they can choose events by type, date and venue. In this case, we have shown the 'events by date' screen where the events are sorted by date. The users will be able to access the event information from this page.

Event and Guest Information Screen

Event Information Page

Event info

Event Name: Khaled's Wedding

Event ID: 10001

Event Type: Private

Date: 12/10/2013

Start Time: 10:00 AM End Time: 05:00 PM

Guest Information Venue Information

Number of guests: 200

Total Event Capacity: 250

Description: Smaller Venue

Edit

Guests of the Event

Guests

Khaled's Wedding

No. of guests checked in: 2 of 200

Guest Information

Guest ID	Name	Checked in
G1001	Amelia Ehrens	<input checked="" type="checkbox"/>
G1002	Karen Chang	<input type="checkbox"/>
G1003	Caley Kovler	<input checked="" type="checkbox"/>
G1004	Eduardo Holguera	<input type="checkbox"/>

Search

The events information page will display the event details and the manager will have the permission to make any changes to this information. Check-in staff will be able to access the information on the page such as the guest and venue information. The number of guests will display the number of guests who registered for the event.

Guests of the event is accessed by the 'Guest information' option from the event information page. This page will display all the number of registered guests, information about their check-in status and the total number of checked-in guests. There is a search bar at the bottom to allow users to search the table for guests by name, guestid, or guest email address.

Venue Information

Venue Information Page



The image shows a smartphone screen displaying the 'Venue info' page. The page has a dark background with a grid pattern. At the top, there is a status bar showing 'VIRGIN 3G' and '4:20 AM'. Below the status bar, there are navigation buttons: 'Back' on the left and 'Main Menu' on the right. The title 'Venue info' is centered at the top of the page. The form contains several input fields with labels on the left and values in the text boxes:

Venue ID	V1001		
Venue Name	V1		
Street	VST#1		
City	City1		
State	MA	Zip	01111
Country	USA		
Capacity	100		
Description	Smaller Venue		

Event staff can access the venue information from the event page by clicking on the venue information option. The venue information page will display information about the name, address and description of the venue.

Guest Information

Guest Information

VIRGIN 3G 4:20 AM
 Back **Guest info** Main Menu
 First Name Amelia
 Last Name Ehrens
 Event Name Khaled's Wedding
 Guest ID G1001
 Guest Type VIP ⓘ
 Checked In Checkin Time 10:05 AM
 Food preference Veg
 Contact Information Seating Info
 ? Edit

Guest Contact Information

VIRGIN 3G 4:20 AM
 Back **Contact info** Main Menu
 Name Amelia Ehrens
 Street GST#1
 City Gcity1
 State MA Zip 01111
 Country USA
 Phone 123456712
 Email g1@gtest.com
 Date of birth 1981-10-11
 ?

When the event staff clicks on the individual guest's name from the guests of the event, they will be directed to that guest's information. This page contains the guest's name, check-in status and time, food preferences, contact information and seating information. The check-in staff will be able to check in the guest by clicking in the circle beside "Checked In." Event staff will also be able to check additional information about the guest based on their guest type. For example, if the guest is a VIP, the page will display his VIP title, company and award. This information can be accessed by the 'blue i' icon beside "Guest Type" option. Event staff can also click on "Contact Information" to access the guest's mailing address, date of birth, email and phone number.

Guest Seating Information

Seating Location Page

VIRGIN 3G 4:20 AM

Back Seating location Main Menu

Name Amelia Ehrens

Table no T1001 >

Room No R1001 >

Venue Name V1 >

Street VST#1

City City1

State MA Zip 01111

Country USA

?

Edit

Table Information Page

VIRGIN 3G 4:20 AM

Back Table info Main Menu

Table No T1001

Location Center

Table Type Circle

Quality Top

Capacity 10

Room no R1001 >

Venue Name V1 >

?

Edit

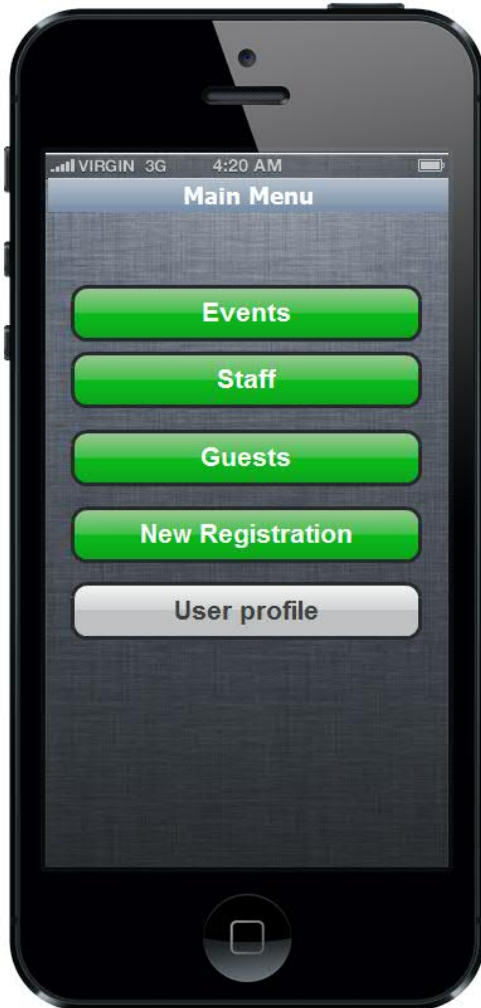
Event staff can access the guest's seating information by clicking on 'Seating Info.' This provides them with access to the guest's table number, room number and venue details. Staff can access the room or venue information by clicking on the room no. or the venue name, which will direct them to the respective pages.

Room Information

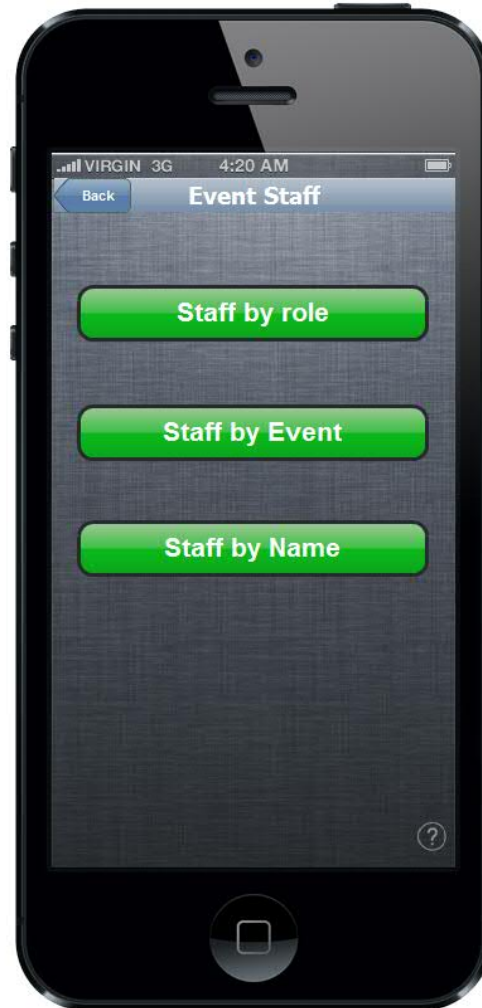


Event Staff information

Event manager Menu

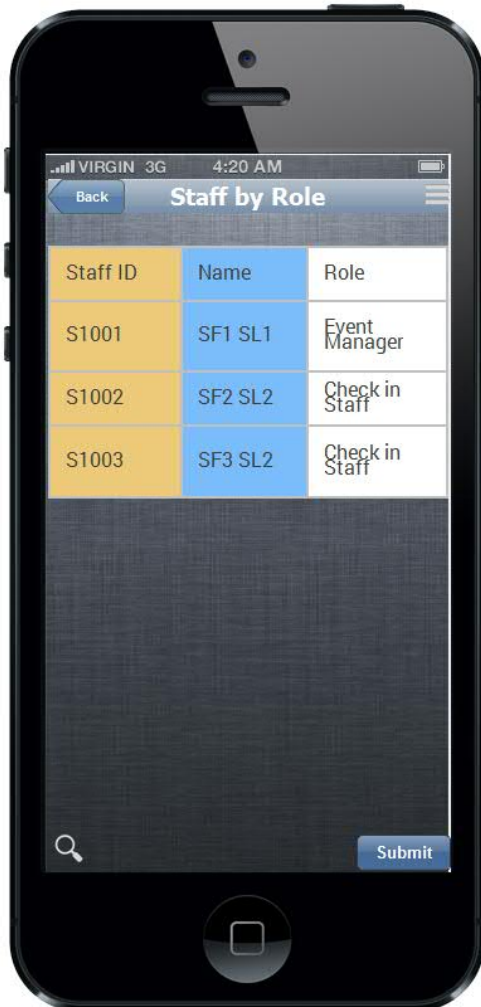


Event Staff



The event manager will have access to “Event Staff” through the main menu. The Staff option will direct to event staff page where the manager can choose staff by role, which displays and shows staff by role. “Staff by Event” option will display the staff sorted by events and “Staff by name” displays the staff by their first name. One example would be the screen shot below (Staff by role), which shows the ‘Staff by role’ display, which shows the staff by their Staff ID, name and their role. There is an option to search for the staff from this section using name, email id, staff id or name through the table displayed.

Staff by Role Page



Staff Information Page

The event Manager can access the staff information by clicking on the Name or the staff Id that will direct them to 'Staff Info' page. 'Staff Info' provides information such as name, id, role, and contact information. The manager will have the ability to edit the details on this page.

This prototype acts as an outline of what we plan to do and to display the interactions between the entities and the information. This is not the actual application and further enhancements and UI changes will be made on further development of the application.

Conclusion

Although there are many activities involved in event management, due to time and resource constraints, we chose to focus on providing a solution for one specific, but major difficulty in event execution. EventHelppo is an application that provides support to event managers and those who perform the registration (check-in) of guests. To begin, we used the business model canvas approach to describe how the application enables and supports event-planning professionals. We then covered business processes and interaction models, including a Use case diagram, Use-case scenarios, an Activity diagram, a Sequence diagram and a Collaboration diagram. Next, we described the conceptual data model, which provided an overview of the different entities within the information system and the relationship between those entities through an EER Model. Based on this information, we built our conceptual and relational models. During the entire process, we received immense help and support from our course instructor Professor Heikki Topi and modified parts of our application based on his feedback. We acknowledge and appreciate his continuous guidance and support.

We believe this application has an excellent opportunity in day-to-day life operations. This application is applicable for any type of event and with careful implementation and proper distribution, can bring a stable stream of revenue. It is also designed in such a way that we can add new modules to it in future to handle additional aspects of event management.

Finally, the entire project was an excellent learning experience. We learned how to create an application from scratch level. Every design and implementation part of it was very rewarding in terms of learning and understanding the processes of IT application development. At the same time, we also got a valuable experience of project management in terms managing schedule, resource and working effectively in a team.

Works Cited

Dobson, D., & Dobson, M. (2012). *Project risk and cost analysis*. Norwood, MA: American Management Association.

Osterwalder, A., & Pigneur, Y. (2010). *Business Model Generation*. John Wiley & sons, Inc.

Topi, H. (2013). *Pearson Custom Business Resources for Information Technology for Competitive Advantage 2 at Bentley University GR620B Fall 2013*.