
Event Planning Application

Project Name: Tadpole

Project Manager: Alaina Gilson

Course: Software Development Capstone 2022SP CRF01

Final design documentation prepared by:
Derrick Nagy
Logan Baccam

Date: May 6, 2022

Design Phase - Scrum Teams

Alaina Gilson - Project Manager

Events Team

Chase Fields - Scrum Leader

Mike Cahow

Derrick Nagy

Volunteers Team

Austin Timmerman - Scrum Leader

Vinayak Deshpande

Mujahid Segyer

Suppliers Team

Emma Pollock - Scrum Leader

Logan Baccam

Kris Howell

Chris Repko

Attendees Team

Jace Pettinger - Scrum Leader

Ramiro Pena

Joseph Philipp

Development Phase - Scrum Teams

Alaina Gilson - Project Manager

Emma Pollock- Product Owner

Derrick Nagy – QA Lead

Team

Derrick Nagy - Scrum Leader

Mike Cahow

Team

Vinayak Deshpande- Scrum Leader

Austin Timmerman

Jace Pettinger

Team

Kris Howell - Scrum Leader

Logan Baccam

Chris Repko

Table of Contents

DESIGN PHASE - SCRUM TEAMS	2
DEVELOPMENT PHASE - SCRUM TEAMS	3
1 EXECUTIVE AND SECTION SUMMARY.....	7
1.1 EXECUTIVE SUMMARY.....	7
1.1.1 <i>Problem statement</i>	7
1.1.2 <i>Proposed solution</i>	7
1.1.3 <i>Value</i>	7
2 USER STORIES.....	8
2.1 USER STORIES SUMMARY	8
2.2 USER STORIES	9
3 USE CASES.....	26
3.1 USE CASES SUMMARY	26
3.2 EVENT USE CASES	27
3.2.1 <i>Event</i>	27
3.2.2 <i>Use Cases</i>	28
3.2.3 <i>Itinerary</i>	68
3.2.4 <i>Tasks</i>	83
3.2.5 <i>Forum</i>	89
3.2.6 <i>After Action Review</i>	99
3.2.7 <i>Promotion</i>	102
3.2.8 <i>Representative Approval</i>	108
3.3 VOLUNTEERS USE CASES	114
3.3.1 <i>Event Details</i>	114
3.3.2 <i>Volunteer Details</i>	153
3.3.3 <i>Applications</i>	165
3.4 SUPPLIER USE CASES.....	175
3.4.1 <i>Supplier</i>	175
3.4.2 <i>Location</i>	197
3.5 ATTENDEE USE CASES.....	237
3.5.1 <i>Event Registration</i>	237
3.5.2 <i>Event Search</i>	243
3.5.3 <i>Messages</i>	249
3.5.4 <i>Profile</i>	254
3.5.5 <i>Contacts</i>	259
3.5.6 <i>Reviews</i>	265
3.5.7 <i>Register and Login</i>	272
4 UI DRAWINGS.....	278
4.1 UI DRAWINGS SUMMARY	278
4.2 WEB.....	279
4.2.1 <i>Event Web UI Drawings</i>	279
4.2.2 <i>Volunteer Web UI Drawings</i>	393
4.1 <i>Supplier Web UI Drawings</i>	428
4.1.2 <i>Attendee Web UI Drawings</i>	461
4.2 MOBILE	478
4.2.1 <i>Event Mobile UI Drawings</i>	478
4.2.2 <i>Volunteer Mobile UI Drawings</i>	523
4.2.3 <i>Supplier Mobile UI Drawings</i>	563
4.2.4 <i>Attendee Mobile UI Drawings</i>	579

4.3	DESKTOP.....	601
4.3.1	<i>Event</i>	601
4.3.2	<i>Supplier Desktop UI Drawings</i>	671
5	ER DIAGRAM	693
5.1	ER DIAGRAM SUMMARY	693
5.2	ER DIAGRAM OVERVIEW	694
5.3	ER DIAGRAM MAP	695
5.4	PAGE 1.....	696
5.5	PAGE 2.....	697
5.6	PAGE 3.....	698
5.7	PAGE 4.....	699
5.8	PAGE 5.....	700
5.9	PAGE 6.....	701
5.10	PAGE 7.....	702
6	DATA DICTIONARY	703
6.1	DATA DICTIONARY SUMMARY	703
6.2	DATA DICTIONARY CONTENTS	704
7	CLASS DIAGRAMS.....	727
7.1	CLASS DIAGRAMS SUMMARY	727
7.2	DATA DOMAIN.....	728
7.2.1	<i>Event Data Domain</i>	728
7.2.2	<i>Volunteer Data Domain</i>	738
7.2.3	<i>Supplier Data Domain</i>	744
7.2.4	<i>Attendee Data Domain</i>	749
7.3	DATA ACCESS INTERFACES	761
7.3.1	<i>Event Data Access Interfaces</i>	761
7.3.2	<i>Volunteer Data Access Interfaces</i>	780
7.3.3	<i>Supplier Data Access Interfaces</i>	788
7.3.4	<i>Attendee Data Access Interfaces</i>	794
7.4	MANAGER INTERFACES	798
7.4.1	<i>Event Manager Interfaces</i>	798
7.4.2	<i>Volunteer Manager Interfaces</i>	817
7.4.3	<i>Supplier Manager Interfaces</i>	825
7.4.4	<i>Attendee Manager Interfaces</i>	832
7.5	VALIDATION HELPERS	837
7.5.1	<i>StringValidationHelper</i>	837
7.5.2	<i>IntegerValidationHelper</i>	837
7.5.3	<i>DateTimeValidationHelper</i>	838
7.5.4	<i>DecimalValidationHelpers</i>	839
8	STORED PROCEDURES	841
8.1	STORED PROCEDURES SUMMARY.....	841
8.2	STORED PROCEDURES.....	842
8.2.1	<i>Task Accessor</i>	842
9	DATA FLOWS.....	878
9.1	DATA FLOWS SUMMARY	878
9.2	CONTEXT DIAGRAM - EVENT SYSTEM – 0	879
9.3	EVENT - 0 - DETAIL.....	880
9.3.1	<i>Overview</i>	880
9.3.2	<i>Page 1</i>	881
9.3.3	<i>Page 2</i>	882

9.4	MANAGE EVENT - 1	883
9.4.1	<i>Create Event - 1.1</i>	<i>884</i>
9.5	MANAGE TASKS - 2	886
9.5.1	<i>Task List - 2.1</i>	<i>887</i>
9.6	MANAGE PROMOTION – 3	888
9.6.1	<i>Manage Invitations - 3.4</i>	<i>889</i>
9.6.2	<i>Forum Manager - 3.5</i>	<i>890</i>
9.7	MANAGE BUDGET - 4	893
9.8	MANAGE LOCATION – 5	894
9.8.1	<i>Manage Sublocation - 5.6</i>	<i>895</i>
9.9	MANAGE SUPPLIER – 6	896
9.9.1	<i>Manage Service - 6.1</i>	<i>897</i>
9.10	SAVED EVENTS MANAGEMENT – 9	898
9.11	SEARCH EVENT MANAGEMENT – 10	899
9.12	REVIEW MANAGEMENT – 11	900
9.13	MESSAGE MANAGEMENT – 12	901
9.14	REPORT MANAGEMENT – 13	902
9.15	VOLUNTEER DETAILS – 14	903
9.16	EVENT TASKS – 15	904
10	SEQUENCE DIAGRAMS	905
10.1	SEQUENCE DIAGRAMS SUMMARY	905
10.2	EVENT SEQUENCE DIAGRAMS	906
10.2.1	<i>Event Tasks</i>	<i>906</i>
10.2.2	<i>Suggestions</i>	<i>907</i>
10.2.3	<i>Approvals</i>	<i>908</i>
10.3	VOLUNTEER SEQUENCE DIAGRAMS	909
10.3.1	<i>Volunteer Applications</i>	<i>909</i>
10.3.2	<i>Volunteer Tasks</i>	<i>910</i>
10.4	SUPPLIER SEQUENCE DIAGRAMS	911
10.4.1	<i>Suppliers</i>	<i>911</i>
10.4.2	<i>Locations</i>	<i>913</i>
10.5	ATTENDEE SEQUENCE DIAGRAMS	917
10.5.1	<i>Messages</i>	<i>917</i>
10.5.2	<i>Profile Commands</i>	<i>918</i>
10.5.3	<i>Event and Contact Searches</i>	<i>919</i>
10.5.4	<i>Reviews</i>	<i>922</i>
11	UI DRAWINGS	ERROR! BOOKMARK NOT DEFINED.
11.1	WEB	ERROR! BOOKMARK NOT DEFINED.
11.1.1	<i>Event Web UI Drawings</i>	<i>Error! Bookmark not defined.</i>
11.1.2	<i>Location Web UI Drawings</i>	<i>Error! Bookmark not defined.</i>
11.1.3	<i>Supplier Web UI Drawings</i>	<i>Error! Bookmark not defined.</i>
11.1.4	<i>Volunteer UI Drawings</i>	<i>Error! Bookmark not defined.</i>
11.1.5	<i>Home Index</i>	<i>Error! Bookmark not defined.</i>
11.2	DESKTOP	ERROR! BOOKMARK NOT DEFINED.
11.2.1	<i>Event Desktop UI Drawings</i>	<i>Error! Bookmark not defined.</i>
11.2.2	<i>Supplier UI Drawings</i>	<i>Error! Bookmark not defined.</i>
11.2.3	<i>Location UI Drawings</i>	<i>Error! Bookmark not defined.</i>
11.2.4	<i>Volunteer UI Drawings</i>	<i>Error! Bookmark not defined.</i>
12	ACKNOWLEDGEMENTS	923

1 Executive and Section Summary

The executive summary and the purpose, summary, and explanation of each section.

1.1 Executive Summary

1.1.1 Problem statement

As of the current date, there are very few viable options to allow communities to organize and easily publish information about local events. Without a way to organize and find community events, there is no easy way to strengthen their relationships with each other. To help create and display these events, many communities would be well served by an event management application that can act not only as a way to manage numerous events, but also as a tool for the community to communicate with one another

1.1.2 Proposed solution

Our new event manager will be used as a communication and event planning tool in order to facilitate the many different parts of community organization. Our application will be designed with the following three interfaces:

- A mobile interface with functionality mostly limited to tasks that event attendees would find useful
- A desktop interface optimized for the event planning workflow
- A web browser interface containing full functionality.

These three interfaces will be designed using the .NET Framework written in C#. Persistent data generated by the application will be stored in a Microsoft SQL Server database, communicated with through the ADO.NET library.

The project will be completed over 15 weeks of work. During this period, work on the project will be driven by three 5-week AGILE sprints in a scrum-style project management process focused on implementing all of the features described within this document.

1.1.3 Value

With this free new event manager made for communities around the world, it's expected to increase communication and expand upon the relations between neighbors and those who have yet to meet within their communities. The event manager will also streamline the process of organizing an event and searching for volunteers and suppliers to staff the event. This will save event managers time that would be otherwise spent moving between different applications and communications mediums, saving the time of the event organization staff, the volunteers willing to lend a hand, and the suppliers offering their services for the event.

2 User Stories

2.1 User Stories Summary

The purpose of this section is to show a summary of the features that the actors would like to achieve in order to create, organize, and run an event with the aid of the system. The user stories provide a quick overview of what outcomes are desired and a description of what action an actor would like to take in order to run a community event.

The actors are generally divided into four groups of different roles: planning and managing an event, managing volunteers, accessing suppliers, and informing and interacting with attendees of the event. The features are organized by the actions that each role would like to perform to meet the ultimate outcome of managing an event and its participants from start to end.

The use case assignment is numbered for referencing the various groups of roles. Event management and planning are any number in the one thousand range, volunteers in the two thousand range, suppliers in the three thousand range, and attendees in the four thousand range.

2.2 User Stories

Feature	As a(n) <i><actor></i>	I would like to <i><description></i>	So that <i><outcome></i>	Use Case Assignment
AAR Overview Create	Event Manager, Event Planner	view AAR infomation	event roles can see the collected information about their event from attendees, volunteers	1002_AAR_Overview_Create
Alerts Create	Event Manager	Create and send Alerts to people working event	Workers/Volunteers can be informed of goings on (i.e. changes in schedule, emergency situations, etc)	1003_Alerts_Create
Emergency Procedures Creation	Event Planner, Event Manager	Create Emergency procedures	the Event Planner can specify to the Event Manager what would need to happen in an emergency	1004_Emergency_Procedures_Creation
Suggestions Submission	Event Organizational Rep	Submit suggestions, comments, or feedback during the planning process	the event Organizational Rep can give input to the event's creation	1005_Suggestions_Submission
Approval	Event Organizational Rep	Approve requests from planner and manager	to help communicate	1006_Approval
Event Share	Event Organizational Rep	Share the event with event planner/other users	The other users have access to the event planning tools(giving users permissions)	1007_Event_Share
Forum	Event Planner, Event Manager	Create and view a forum	to communicate with members	1008_Forum
Event Create	Event Planner, Event Organizational Rep	select a button to create an event	bring up an empty event creation form to fill out	1009_Event_Create
Event Edit	Event Planner	select a button to view and	to confirm what has been set and has not and edit/update as needed	1010_Event_Edit

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
		edit an already created event		
Event Location	Event Planner	set the location for the event	set location, entrances, parking, entrances, public parking, service entrance	1011_Event_Location
Event Location Directions	Event Planner	create, edit directions pertaining to the event	to give any extra information that might help, attendees, volunteers find the place,	1012_Event_Location_Directions
Event Date(s)	Event Planner	set the date or dates of the event	to show the date and time of the event	1013_Event_Date(s)
Event Date Create	Event Planner	add a date and time for an event	attendees can know the start and end times of the event	1013_Event_Date_Create
Vendor Tags Add	Event planner	Set the tags for what kind of vendors wanted	Vendors can see what kind of vendors are requested	1014_Vendor_Tags_Add
Volunteer Add	Event Planner	Specify if I need volunteers for an event	Volunteers can volunteer for an event	1015_Volunteer_Add
Voting	Event Planner (primary), Event promoter, Event manager	create and store voting and polling results	to gain a consensus among volunteers, other planners	1016_Voting
Event Calendar View	Event Planner	view an events calendar	to show dates and information about the event(s), guard against conflicts	1017_Event_Calendar_View
Task List	Event Planner, Event Manager	create a planning/ task list/ to do list	to help keep track of what needs to be accomplished to be ready, what has been done, to do list, assign volunteers to help	1018_Task_List
Reminder Send	Event Promoter, Event Manager	Send a reminder	to share when the event starts	1019_Reminder_Send
Itinerary	Event Planner,	create an itinerary/	to share with event manager	1020_Itinerary

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
	Event Manager, Event Organizational Rep	multiple itineraries		
Message Planning Team	Event Planner	Send and receive messages with event planning team	to communicate on the planning of the event	1022_Message_Planning_Team
Task List Create	Event Planner	create task list for event manager	the event manager/volunteers can have a detailed, organized list of what to do the day of the event	1023_Task_List_Create
File Upload	Event Planner	upload files relevant to event (i.e. documents, permits, contracts, maps)	Anyone (attendee, law enforcement, etc) who needs to see relevant information for the event can view it	1024_File_Upload
Invitations Create	Event Promoter, Event Organizational Rep	Create and Send email invitations	Attendees to private events can RSVP and give Planner and Manager a proper headcount	1027_Invitations_Create
Invitations Share	Event Promoter	Decide who gets the invitation and how	Attendees can be invited to the event	1028_Invitations_Share
Activity Results	Event Planner, Event Manager	to show the results of an activity	so that users can view the results	1029_Activity_Results
Activity Results Add	Event Manager	Add the results of the activity	so that the event manager can record it	1030_Activity_Results_Add
Budget Add Expense	Event Planner	Enter expenses into the budget	Can keep track of expenditures	1032_Budget_Add_Expense
Budget Remove Budget Line Item	Event Planner	Remove a line item from the budget.	Delete transactions that I no longer need.	1033_Budget_Remove_Budget_Line_Item

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
Budget View Expenses	Event Planner	View the expenses entered for a certain category	Keep track of expenses	1034_Budget_View_Expenses
Emergency Procedure s View	Event Manager, Event Volunteers, Event Attendees	View the emergency procedures	Those working the event can read and prepare for emergency situations that could happen at event	1035_Emergency_Procedures_View
Emergency Procedure s Update	Event Planner, Event Manager	Update the emergency procedures	the Event planner can specify to the Event Manager what would need to happen in an emergency	1036_Emergency_Procedures_Update
Suggestions View	Event Planner, Event Organizational Rep	View the itinerary and task list	make suggestions by approving, disapproving, or commenting on what is happening	1037_Suggestions_View
Suggestions Create	Event Planner	Add items to the suggestions page	the Event Organizational Rep can see and then take action on suggestions	1038_Suggestions_Create
AAR Overview Save	Event Planner	Save the results of the After Event Overview	I can repeatedly view results without re-creating the report	1039_AAR_Overview_Save
Itinerary Add Occurrences	Event Planner, Event Manager	Add an occurrence to a created itinerary	The itinerary has a list of occurrences	1041_Itinerary_Add_Occurrences
Itinerary Delete	Event Planner, Event Manager	Delete an itinerary	It isn't associated with the event	1042_Itinerary_Delete
Itinerary List View	Event Planner, Event Manager	View a list of the itineraries for an event	The individual itineraries can be viewed and edited	1043_Itinerary_List_View
Itinerary Edit	Event Planner, Event Manager	Edit the characteristics of an itinerary	The itinerary can be updated	1044_Itinerary_Edit

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
Itinerary Delete Occurrences	Event Planner, Event Manager	Delete an occurrence from an itinerary	The occurrence is no longer listed in the itinerary	1045_Itinerary_Delete_Occurrences
Itinerary Edit Occurrences	Event Planner, Event Manager	Edit an occurrence	The occurrence's details can be changed	1046_Itinerary_Edit_Occurrences
Approval Request	Event Planner, Event Manager	Create a request for approval from the org rep	The org rep can approve or deny the request	1047_Approval_Request
Forum Create Post	Event Planner, Event Manager	Create a post on the forum	Participants can view post	1048_Forum_Create_Post
Forum Delete Post	Event Planner, Event Manager	Delete a post on the forum	The post is removed/unviewable from the forum	1049_Forum_Delete_Post
Task List View	Event Planner, Event Manager	View the task list	The tasks can be toggled completed/uncompleted and the tasks can be read	1050_Task_List_View
Task List Edit	Event Planner, Event Manager	Edit a task on the task list	The task list attributes can be changed	1051_Task_List_Edit
File View Relevant Files	Event Planner, Event Manager, Event Organizational Rep, Event Promoter	View relevant files for an event	Anyone needing to see the information can do so	1052_File_View_Relevant_Files
File Delete Relevant Files	Event Planner	Delete relevant files from information page	any unneeded information/files can be removed	1053_File_Delete_Relevant_Files
Task List Delete	Event	Delete an Item off Task List	I can get rid of Tasks that aren't pertinent to an Events completion	1054_Task_List_Delete

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
Voting Cast Vote	Event Planner, Forum Participants(attendees, volunteers)	Cast my vote on a voting poll	My vote is tallied and I can see the current results of the vote	1056_Voting_Cast_Vote
Voting Results View	Event Planner, Event Promoter, Forum Participants(attendees, volunteers)	See the results of a poll I voted on or a poll that has expired	I can see what people are voting for	1057_Voting_Results_View
Forum Comment Create	Event Planner, Event Promoter, Forum Participants(attendees, volunteers)	Comment on a post in the forum	So that other people can see my comment	1058_Forum_Comment_Create
Forum Edit	Event Planner, Event Manager, Event Promoter	Edit the details of a forum	I can change the name, description, and participants	1059_Forum_Edit
Event Date Activities Create	Event Planner, Event Manager, Event Organizational Rep	select an event date and be able to add activities to the date	the event activities can be planned	1060_Event_Date_Activities_Create
Event Date Activities Retrieve Single Day	Event Planner, Event Manager, Event Organizational Rep	view all the activities for a date for an event	the event activities can be viewed	1061_Event_Date_Activities_Retrieve_Single_Day
Event Date	Event Planner,	view all the dates for an	the event activities can be viewed	1062_Event_Date_Activities_Retrieve

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
Activities Retrieve All Event Dates	Event Manager, Event Organizational Rep	event and the activities for them on that date		eve_All_Event_Dates
Event Date Activities Update	Event Planner	update the information about the event date	the event can stay current	1064_Event_Date_Activities_Update
Event Date Activities Deactivate	Event Planner	deactivate the event date and all the activities on that date	so that it can be canceled but not permanently deleted	1065_Event_Date_Activities_Deactivate
Event Date Activities Delete	Event Planner, Event Manager, Event Organizational Rep	delete the event date and all the activities on that date		1066_Event_Date_Activities_Delete
Activity Details	Event Planner, Event Manager, Event Organizational Rep	view the details of an individual event		1068_Activity_Details
Activity Retrieve List	Event Planner, Event Manager, Event Organizational Rep	view a list of all the activities for an event	to sort and filter to help in searching	1069_Activity_Retrieve_List
Activity Retrieve Related	Event Planner, Event Manager, Event Organizational Rep	view a list of all the activities for all events	to sort and filter to help in searching	1070_Activity_Retrieve_Related
Activity Update	Event Planner,	update an activity	it can state up to date	1071_Activity_Update

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
	Event Manager, Event Organizational Rep			
Activity Deactivate	Event Planner	deactivate an activity for an event because it is no longer used		1072_Activity_Deactivate
Activity Delete	Event Planner	delete an activity for an event because it is no longer used		1073_Activity_Delete
Event Date Update	Event Planner	edit the event date	interested parties stay updated with new dates and times	1075_Event_Date_Update
Event Date Deactivate	Event Planner	deactivate the event date	the event date information is no longer seen but still able to be retrieved	1076_Event_Date_Deactivate
Event Date Delete	Event Planner	delete the event date	the event date is permanently removed	1077_Event_Date_Delete
Create Event Roles	Event Planner, Admin	have my roles added	I can manage the event roles that are assigned	1078_Create_Event_Roles
Event Duplication	Event	Duplicate an Event I have created before	I can quickly and easily create a repeat event	1079_Event_Duplication
Accept / Reject Requests	All Volunteers	Accept or Reject Requests	I can decide to accept or reject a request for volunteering / supplying	2001_Accept_Deny_Volunteer_Request
View Requests	All Volunteers	View my incoming requests	I can view my requests to volunteer / supply for a certain event	2002_Volunteer_View_Requests
Cancel Volunteer Obligation	All Volunteers	Cancel my participation	I can back out of a particular volunteer job	2004_Cancel_Volunteer_Obligation
Request to Volunteer / Supply	All Volunteers	Apply to volunteer for specific event	to publically show which organizers I'd like to volunteer for	2005_Request_to_volunteer_or_supply

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
View Requests	All Volunteers	View my outgoing requests for volunteering / supplying	Volunteers are able to view any pending requests for events	2006_Volunteer_View_Outgoing_Requests
Volunteer Review Event	All Volunteers	Give reviews for event	others can see how well the host and volunteers did	2007_Volunteer_Review_Event
View Events	All Volunteers	View list of all events	I can see which events are available	2008_View_Events
View Event Details	All Volunteers	View specific event details	I can see when and where the event is, what the description is, how many people, etc.	2009_View_Event_Details
View Volunteer Task	All Volunteers	View what kind of work I'll be providing	I can prepare for a certain event	2010_View_Volunteer_Task
View Availability	All Volunteers	View my availability	I can see which days I have set are available and which days aren't	2011_View_Availability
Change Availability	All Volunteers	Change my availability	I can change which days / time I will be available	2012_Change_Availability
Request Skill Set Verification	Specific Volunteer	Request for a skill set to be verified	I can have the skill verified	2013_Request_Skill_Set_Verification
Add volunteer to Task	Event Planner	Add a volunteer to a task	I can see what volunteers are assigned to my task	2014_Add_Volunteer_to_Task
Add / Remove Volunteer Proof	All Volunteers	Add or Remove proof to my volunteer proof for specific event	The event planner / verification manager can view my proof of supplying / volunteering	2015_Add/_Remove_Volunteer_Proof
View Volunteer Proof	All Volunteers	View my volunteer proof for specific events	I can view my volunteer proof	2016_View_Volunteer_Proof
Change Volunteer Proof	All Volunteers	Change my volunteer proof images / description	I can update / change / delete anything for my specific proof	2017_Change_Volunteer_Proof

Feature	As a(n) <i><actor></i>	I would like to <i><description></i>	So that <i><outcome></i>	Use Case Assignment
View Volunteer Supplies	Supply Donor	View my supply list	I can see what supplies I have	2018_View_Volunteer_Supplies
Add / Remove Volunteer Supplies	Supply Donor	Add or remove supplies from my supply list	I can modify the amount or what supplies I currently have	2019_Add/_Remove_Volunteer_Supplies
add favorite	Event Manager	I need to be able to add and remove from my list of my favorites	So I can add or remove people from it	2020_Add_or_Remove_to_Favorite_Volunteer_List
view list of favorites	Event Manager	I need to be able to see a list of my favorites	So I can easily tell if they are available	2021_View_List_of_Favorite_Volunteers
View Requests	Event Manager	To view volunteer requests	Event Manager is able to view any pending requests for their event	2022_Event_Manager_View_Volunteer_Requests
Accept / Reject Requests	Event Manager	To be able to accept or reject volunteer requests	I can decide who can volunteer for my event	2023_Accept_/_Reject_Volunteer_Requests
View List of Volunteers	Event Manager	View a list of volunteers	I know which volunteers I can appropriate for my event	2024_View_List_of_Volunteers
View Volunteer Tasks	Event Manager	View a list of volunteers who are assigned to tasks	I know who is working where	2025_View_List_of_Volunteer_Tasks
View Specific Volunteer Ratings	Event Manager	View volunteer specific ratings	I can decide if I want to see specific ratings for a volunteer	2026_View_Volunteer_Ratings
Sign up for Volunteer	Everyone	Sign up to become a volunteer	I can become a volunteer	2027_Volunteer_Application
Verification	Admin	Accept or deny verification applications	I can accept or deny these applications	2028_Accept_or_Deny_Verification_Applications
View Applications	Admin	View list of new applicants	I know who I need to obtain verification or deny verification for	2029_View_Verification_Applications

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
View Applications Specifics	Admin	View the specifics of a volunteer applying	I can view details of the applicant	2030_View_Specific_Application
View Skill Set	Specific Volunteer	View my skill sets	I can view my skill sets listed out	2031_View_Skill_Set
Add Skill to Skill Set	Specific Volunteer	Add a new skill to my skillset	I can add on to my skill sets	2032_Add_Skill_to_Skill_Set
Remove Skill from Skill Set	Specific Volunteer	Remove a specific skill from my skillsets	I can remove unwanted skills	2033_Remove_Skill_from_Skill_Set
Sublocation Maps	Location	Add a map	Other users know how my space is laid out	3001_Add_Sublocation_Map
Location Parking Lots	Location	Add a parking lot	Others users know what areas are available for parking	3002_Add_Parking_Lot
Sublocation Rules	Location	Add a rule	Other users know what they are not allowed to do in my space	3003_Add_Location_Rule
Supplier Services	Supplier	Add a service to my list	My information stays accurate	3004_Add_Service
Sublocations	Location	Add a sublocation	Users can see what areas my location is divided into for rent	3005_Add_Sublocation
Sublocation Supplies	Location	Add a supply item	Users know what supplies are included for my location	3006_Add_Supply
Location Entrances	Location	Add an entrance	Users can find entrances for my location easily	3007_Add_Entry
Supplier Listings	Supplier	Create a Listing	Events can find me and book me for events	3008_Add_Listing
Location Listings	Location	Create a Listing	Events can find me and book me for events	3009_Add_Listing
Supplier-Location Schedules	Supplier	Create a Schedule	Users can know when i'm available	3010_Create_Schedule
Supplier-Location Schedules	Location	Create a Schedule	So events know when i'm available	3010_Create_Schedule
Supplier Listings	Supplier	Deactivate my Listing	My information stays accurate	3011_Deactivate_Listing
Location Listings	Location	Deactive my Listing	My information stays accurate	3011_Deactivate_Listing
Location Parking Lots	Location	Delete a parking lot	My information stays accurate	3012_Delete_Parking_Lot

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
Sublocation Rules	Location	Delete a rule	My information stays accurate	3013_Delete_Rule
Sublocations	Location	Delete a sublocation	My information stays accurate	3014_Delete_Sublocation
Sublocation Supplies	Location	Delete a supply item	My information stays accurate	3015_Delete_Supply_Item
Location Entrances	Location	Delete an entrance	My information stays accurate	3016_Delete_Entrance
Supplier-Location Schedules	Supplier	Delete Item from Schedule	My information stays accurate	3017_Delete_Item_from_Schedule
Supplier-Location Schedules	Location	Delete Item from Schedule	My information stays accurate	3017_Delete_Item_from_Schedule
Location Parking Lots	Location	Update a parking lot	My information stays accurate	3018_Edit_Parking_Lot
Sublocation Rules	Location	Update a rule	My information stays accurate	3019>Edit_Rule
Sublocations	Supplier	Update a service from my list	My information stays accurate	3020>Edit_Supplier_Service
Location Entrances	Location	Update a sublocation	My information stays accurate	3021>Edit_Sublocation
Location Entrances	Location	Update an entrance	My information stays accurate	3022>Edit_Entrance
Supplier Services	Supplier	Remove a service from my list	My information stays accurate	3023>Delete_Service
Sublocation Supplies	Location	Update a supply item	My information stays accurate	3024_Update_Supply
Location Listings	Location	Update my Bio Information	My information stays accurate	3025_Update_Bio
Supplier Listings	Supplier	Update my Listing	My information stays accurate	3026_Update_Supplier_Listing
Supplier-Location Schedules	Supplier	Update my Schedule	My information stays accurate	3027_Update_Schedule
Supplier-Location Schedules	Location	Update my Schedule	My information stays accurate	3027_Update_Schedule
View Filtered Events	Supplier	View a List of Events that are looking for	I can reach out to the event for a job	3028_Supplier_Find_Personalized_Events

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
		Suppliers with my tags		
List Locations	Supplier	View a List of Locations	I can see every location an event might be and details of location	3029_Browse_Locations
List Locations	Event Planner	View a List of Locations	I can determine if its the best fit for an event	3029_Browse_Locations
List Locations	Attendee	View a List of Locations	I can see every location an event might be and details of location	3029_Browse_Locations
List Suppliers	Event Planner	View a List of Suppliers	I can determine the best to hire	3030_Browse_Suppliers
List Suppliers	Attendee	View a List of Suppliers	I can see what Suppliers are going to be at an event	3030_Browse_Suppliers
Location Listings	Supplier	View a Location	I can see the details of location	3031_View_Location_Listing
Location Listings	Event Planner	View a Location	I can determine if its the best fit for an event	3031_View_Location_Listing
Location Listings	Attendee	View a Location	I can see the details of location	3031_View_Location_Listing
Location Gallery	Supplier	View a Location's image gallery	I can see the details of location	3032_View_Location_Gallery
Location Gallery	Event Planner	View a Location's image gallery	I can determine if its the best fit for an event	3032_View_Location_Gallery
Location Gallery	Attendee	View a Location's image gallery	I can see the details of location	3032_View_Location_Gallery
Location Entrances	Supplier	View a Location's list of entrance	I can know where to enter	3033_View_Location_Entrances
Location Entrances	Event Planner	View a Location's list of entrance	I can determine if its the best fit for an event	3033_View_Location_Entrances
Location Entrances	Attendee	View a Location's list of entrance	I can know where to enter	3033_View_Location_Entrances
Location Parking Lots	Supplier	View a Location's list of parking lot	I can know where all parking is	3034_View_Parking_Lots
Location Parking Lots	Event Planner	View a Location's list of parking lot	I can determine if its the best fit for an event	3034_View_Parking_Lots

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
Location Parking Lots	Attendee	View a Location's list of parking lot	I can know where all parking is	3034_View_Parking_Lots
Supplier-Location Schedules	Attendee	View a Location's schedule	I can know when a location is available	3035_View_Location_Schedule
View Event	Supplier	View a Specific Event	I can see details of an event	3036_View_Specific_Events
Sublocations	Event Planner	View a Sublocation	I can determine if its the best fit for an event	3037_View_Sublocation
Sublocations	Attendee	View a Sublocation	I can see details of a sublocation	3037_View_Sublocation
Sublocations	Supplier	View a Sublocation	I can see details of a sublocation	3037_View_Sublocation
Sublocation Maps	Supplier	View a Sublocation's maps	I can see where everything is at for the sublocation	3038_View_Sublocation_Maps
Sublocation Maps	Event Planner	View a Sublocation's maps	I can determine if its the best fit for an event	3038_View_Sublocation_Maps
Sublocation Maps	Attendee	View a Sublocation's maps	I can see where everything is at for the sublocation	3038_View_Sublocation_Maps
Sublocation Rules	Supplier	View a Sublocation's rules	I can know what is allowed at the sublocation	3039_View_Sublocation_Rules
Sublocation Rules	Event Planner	View a Sublocation's rules	I can know what is allowed at the sublocation	3039_View_Sublocation_Rules
Sublocation Rules	Attendee	View a Sublocation's rules	I can know what is allowed at the sublocation	3039_View_Sublocation_Rules
Supplier-Location Schedules	Event Planner	View a Sublocation's schedule	I can know when a sublocation is available for rent	3040_View_Sublocation_Schedule
Sublocation Supplies	Event Planner	View a Sublocation's supplies	I can know what supplies come with the sublocation	3041_View_Sublocation_Supplies
Supplier Listings	Event Planner	View a Supplier Listing	I can determine if they're best fit for an event	3042_View_Supplier_Listing
Supplier Listings	Attendee	View a Supplier Listing	I can know the details of the what and where the supplier will be at	3042_View_Supplier_Listing

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
Supplier-Location Schedules	Attendee	View a Supplier's Schedule	I can know the details of when the supplier is busy	3043_View_Supplier_Schedule
Supplier-Location Schedules	Event Planner	View a Supplier's Schedule	I can know if they're available for an event	3043_View_Supplier_Schedule
Supplier Services	Event Planner	View a Supplier's Services	I can determine the best to hire	3044_View_Supplier_Services
Supplier Services	Attendee	View a Supplier's Services	I can know what a supplier has to offer	3044_View_Supplier_Services
Supplier Gallery	Supplier	Add an image to the image gallery	My information stays accurate	3045_Add_Image
Location Gallery	Location	Add an image to the image gallery	My information stays accurate	3045_Add_Image
Supplier Gallery	Supplier	Delete an image from the image gallery	My information stays accurate	3046_Remove_Image
Location Gallery	Location	Delete an image from the image gallery	My information stays accurate	3046_Remove_Image
Create attendee review	Registered User	review an event	I can let others know if an attendee was good or bad	4001_create_attendee_review
Create organizer review	Registered User	review an organizer	I can let others know if an organizer was good or bad	4002_create_organizer_review
Create event review	Registered User	review an event	I can let others know if the event was good or bad	4003_create_event_review
create vendor review	Registered User	review a vendor	I can let others know if a supplier was good or bad	4004_create_vendor_review
receive similar event notification	Registered User	receive events that match the event I have attended	I can get notified of events I would potentially be interested in	4005_receive_similar_event_notification
report event	Registered User	report an event	I can report events that were very bad or violated community standards	4006_report_event

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
search users	User	search for other people	I can find other users on the site	4007_search_users
report users	Registered User	report another user	I can report suspicious or malicious users	4008_report_users
cancel event attendance	Registered User	cancel going to an event	I can cancel my RSVP to an event	4009_cancel_event_attendance
bookmark /save event	Registered User	bookmark an event	I can save an event I am interested in	4010_save_event
view attendance list	User	see how many people are attending/who is attending	I can see the attendance list and see if I like the people attending	4011_view_attendance_list
rsvp event	Registered User	sign up for an event	I can mark myself as going to an event	4012_rsvp_event
view events	User	view a list events	I can see what events are coming up or have previously happened	4013_view_events
edit profile	Registered User	edit my profile	I can change or update my profile	4014_edit_profile
receive event reminder	Registered User	receive a reminder of an event	I can be notified of an upcoming event I signed up for	4015_receive_event_reminder
check in	Registered User	check in to events	I can show that I am currently at the event / did attend the event	4016_check_in
view messages	Registered User	view messages from other people	I can see messages people have sent me	4017_view_messages
log in	Registered User	login to my account	I can access my account and data	4018_log_in
remove friend	Registered User	remove friend from friends list	I can no longer be friends with a user	4019_remove_friend
view friends	Registered User	view friends list	I can see who my friends are	4020_view_friends
add friend	Registered User	friend other users	I can easily connect with other users	4021_add_friend
message groups	Registered User	create message groups	I can message a group of people who are attending the same event	4022_message_groups
create message	Registered User	message other people	I can contact friends / event hosts	4023_create_message
view reviews	Registered User	see reviews	I can see what people thought of the event	4024_view_reviews

Feature	As a(n) <actor>	I would like to <description>	So that <outcome>	Use Case Assignment
search events	User	search for events	I can find events that might interest me	4025_search_events
share event	User	share events	I can send an event to friends/family who might be interested	4026_share_event
register	User	receive an option to become a registered user	I can create a profile	4027_register_user
view profile	User	look at another persons profile	I can see if I know them or want to interact with them	4028_view_profile
view saved events	Registered User	view my list of saved events	I can easily access events that interested me that I may not have rsvp to	4029_view_saved_events
Request Supplier	Supplier	Request supplier to attend an event	See if they are available	69420_Request_Supplier
Budget Create	Event Planner	create and manage an event budget	so that I can accurately track event expenses relative to my budget	Implemented into 1009_Event_Create

3 Use Cases

3.1 Use Cases Summary

The purpose of the use case section is to take the user stories and compile them into organized documentation. The documentation for the use case provides an outline of how the outcomes from the user stories are going to be accomplished and provide a step by step format of what is required for each story to be completed.

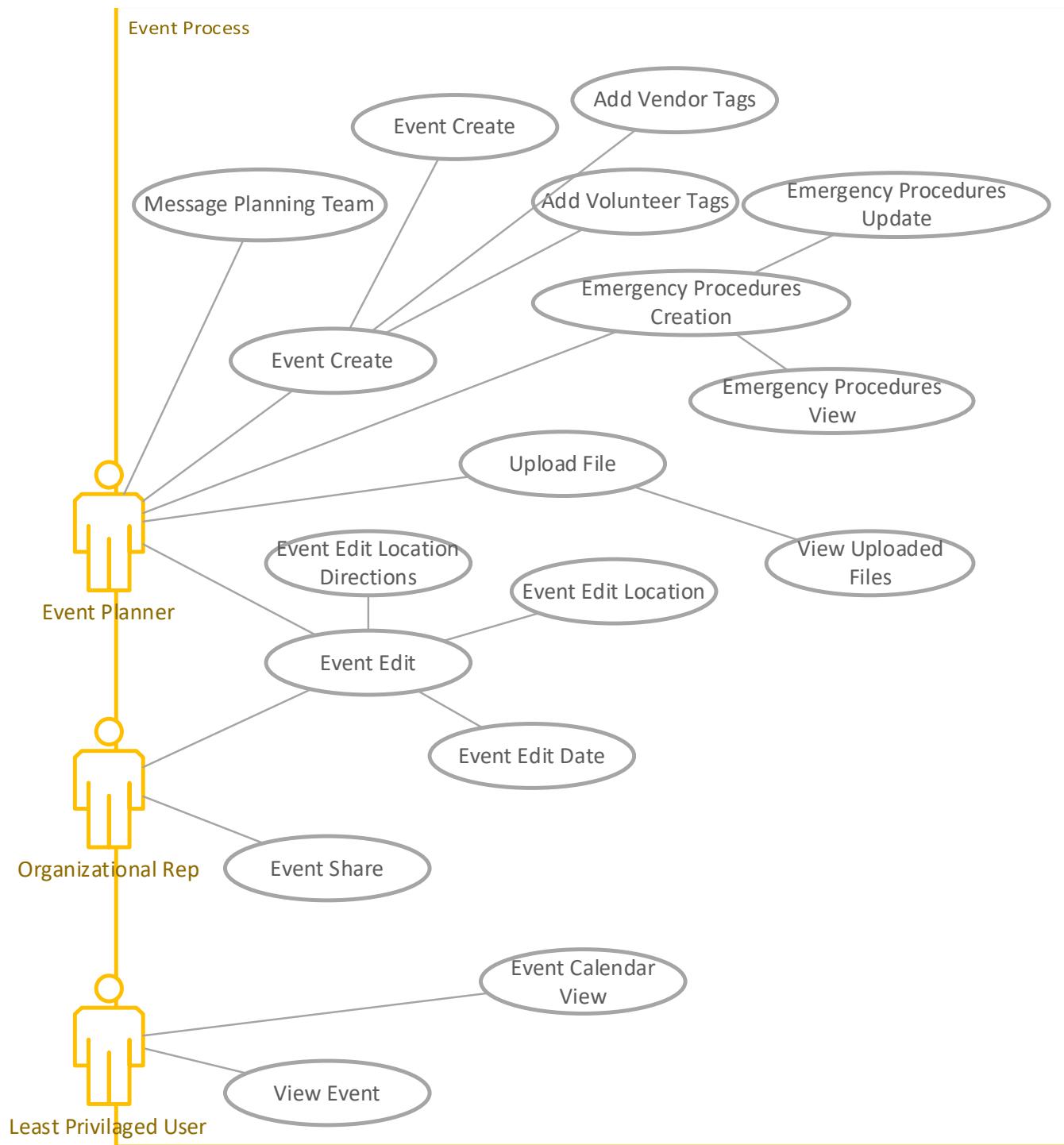
Use cases are composed of the conditions that must be met before the use case can begin, what conditions will be created once the use case has finished, the normal “best case” scenario of how the use case operates, and alternate flows that can arise that would cause a change in the normal flow. Use cases also provide the documentation for other notes or concerns the writer of the use case may have.

The use cases for this project are sectioned into four major groups based off the main roles of the system with the 1000 group being the use cases for events, the 2000 group being the use cases for volunteers, the 3000 group being the use cases for suppliers, and the 4000 group being the uses cases for attendees.

3.2 Event Use Cases

3.2.1 Event

3.2.1.1 Event Use Case Model



3.2.2 Use Cases

3.2.2.1 Emergency Procedures Create

Use Case ID:	1004		
Use Case Name:	Emergency Procedures Create		
Created By:	Derrick Nagy	Last Updated By:	Derrick Nagy
Date Created:	09/21/2021	Last Revision Date:	10/13/2021
Actors:	Event Planner(primary), Event Manager(secondary)		
Description:	Create Emergency procedures so that the Event Planner can specify to the Event Manager what would need to happen in an emergency		
Trigger:	Create event- System creates a page for emergency procedures to be added. to Event locations, system checks to see if the chosen location already has emergency procedures for it.		
Preconditions:	1. An event has been created		
Postconditions:	1. Minimum: An emergency procedure is posted that at least advises people to call 911 in an emergency and listen to any instructions, find an exit... 2. Success: Event Planner has added emergency procedures that is specific to the event 3. Success: System finds the emergency procedures for the location and adds it to the posting 4. Event Manager can view the emergency procedures		
Normal Flow:	1. System creates posting 2. System checks to see if the Event has a location 3. System checks to see if there are already emergency procedures for the location 4. System adds emergency procedures to the posting if they exist. Default procedures are added if not. 5. Event Planner is asked to approve the emergency procedures, given the option to remove the procedures the system added 6. Event Planner can add Contacts, Exits, and Muster Points to the form that weren't automatically added. 7. Event Planner approves changes 8. System asks Event Planner if they would like to share their emergency procedures for that location so that other events can use it 9. System updates emergency procedures for a location if the Event Planner agrees, otherwise does nothing 10. System makes the emergency procedures available to be viewed for the event		
Alternative Flows: [Alternative Flow 1 – No emergency procedures]	5a. Event Planner disapproves of all changes and would like there to be no emergency procedures 6. System makes it clear that there are no emergency procedures to be viewed		
Exceptions:			
Includes:	Event Location, Event Creation		

Frequency of Use:	Once during creation
Special Requirements:	
Assumptions:	The Event Planner is competent and qualified enough to supply this information independently.
Notes and Issues:	

3.2.2.2 Emergency Procedures Update

Use Case ID:	1036	
Use Case Name:	Emergency Procedures Update	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	09/21/2021	Last Revision Date:
Actors:	Event Planner(primary), Event Manager(secondary)	
Description:	Update emergency procedures so that the Event planner can specify to the Event Manager would need to happen in an emergency	
Trigger:	Event Planner chooses to update the emergency procedures	
Preconditions:	<ol style="list-style-type: none"> 1. An event has been created 2. The emergency procedures have already been added and approved 	
Postconditions:	<ol style="list-style-type: none"> 1. Minimum: An emergency procedure is posted that at least advises people to call 911 in an emergency and listen to any instructions, find an exit... <ul style="list-style-type: none"> o 2a: Success: Event Planner has added emergency procedures that is specific to the event o 2b. Success: System finds the emergency procedures for the location and adds it to the posting o 3. Event Manager can view the emergency procedures o 	
Normal Flow:	<ol style="list-style-type: none"> 1. Event Planner starts an update/edit event 2. Event Planner can change the procedures 3. Event Planner is asked to approve the emergency procedures 4. System makes the emergency procedures available to be viewed 	
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none"> 3a. Event Planner disapproves of all changes and would like there to be no emergency procedures 4. System makes it clear that there are no emergency procedures to be viewed 	
Exceptions:		
Includes:	Emergency Procedure Creation	
Frequency of Use:	As many times as the Event Planner would like during the planning phase, seems unlikely to be used more than a couple times	
Special Requirements:		
Assumptions:	The Event Planner is competent and qualified enough to supply this information independently.	
Notes and Issues:		

3.2.2.3 Emergency Procedures View

Use Case ID:	1035	
Use Case Name:	Emergency Procedures View	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	09/21/2021	Last Revision Date:
Actors:	Event Manager(primary), Event Volunteers(secondary), Event Attendees(secondary)	
Description:	View the emergency procedures so that those working the event can read and prepare for emergency situations that could happen at event	
Trigger:	The Event Planner approves of the emergency procedures in the Emergency Procedures use case	
Preconditions:	1. Emergency Procedures Creation has been completed	
Postconditions:	1. Minimum: An emergency procedure is posted that at least advises people to call 911 in an emergency and listen to any instructions, find an exit... 2. All roles can view the emergency procedures	
Normal Flow:	1. User requests to view the emergency procedures and does	
Alternative Flows:		
Exceptions:		
Includes:	Event Location, Emergency Procedures Creation	
Frequency of Use:	Anytime a user would like to see it	
Special Requirements:		
Assumptions:	The Event Planner is competent and qualified enough to supply this information independently.	
Notes and Issues:		

3.2.2.4 Event Create

Use Case ID:	1009	
Use Case Name:	Event Create	
Created By:	Mike Cahow	Last Updated By: Alaina Gilson
Date Created:	9/18/21	Last Revision Date: 2/24/22
Actors:	Event Planner, Event Organizational Rep	
Description:	User presses a button to create an event which brings up an event creation form.	
Trigger:	User clicks Create Event button on interface	
Preconditions:	<ol style="list-style-type: none"> 1. User has to sign in as an Event Planner or Event Organizational Rep 2. User has to click create event button 	
Postconditions:	Blank Create event form is created with Name and Description field filled out.	
Normal Flow:	<ol style="list-style-type: none"> 1. Planner clicks on the Create Event button 2. System brings up a blank form page 3. System prompts Planner to enter a name for event 4. Planner enters a name for the event 5. System displays entered name for event on the form 6. System prompts Planner to enter an event description 7. Planner enters a description of the event 8. System displays event's description on the form 9. Planner enters a budget for the event 10. System displays event's budget on the form 11. Planner selects Save Event button 12. System saves event progress 13. Planner selects Exit button 14. System creates an activity for the event, based off the title and description 15. System closes form 	
Alternative Flows: [Alternative Flow 1 – Not in Network]	<p>9a. In step 9 of the normal flow, if the Planner selects the Close button instead of the Save button</p> <ol style="list-style-type: none"> 1. System prompts the Planner to save the event 2. Planner agrees to save the event 3. Use Case resumes on step 13 <p>9b. In step 9 of the normal flow, if the Planner selects the Close button instead of the Save button</p>	

	<ol style="list-style-type: none"> 1. System prompts the Planner to save the event 2. Planner declines save 3. System does not save any progress (Planner will have to start over) 4. Use Case resumes on step 13 of normal flow <p>9c. In step 9 of the normal flow, if the Planner has other fields in the form they have the information to fill out</p> <ol style="list-style-type: none"> 1. Planner enters.....field(s) 2. Use case resumes on step 9 of normal flow
Exceptions:	<p>4a. In step 4 of the normal flow, if the Planner doesn't enter a name for the Event</p> <ol style="list-style-type: none"> 1. System notifies Planner to enter a name before moving on 2. Planner enters a name for the Event 3. Use Case resumes on step 5 of normal flow <p>6a. In step 6 of the normal flow, if the Planner doesn't enter a description for the Event</p> <ol style="list-style-type: none"> 1. System notifies Planner to enter a description for the event before moving on 2. Planner enters a description for the Event 3. Use Case resumes on step 8 of normal flow
Includes:	Create Event Use Cases to fill out the necessary field lists to create a proper event
Frequency of Use:	This feature to be used every time an event is to be created
Special Requirements:	
Assumptions:	There is a Create Event button on the interface
Notes and Issues:	<ol style="list-style-type: none"> 1. Event Organizational Rep can only work through the normal flow and alternate flows 9a and 9b 2. The only other field(s) the Organizational Rep can specify are the date and location fields. 3. What does it look like when the user is not authorized to create an event?

3.2.2.5 Event Date Create

Use Case ID:	1013		
Use Case Name:	Event Date Create		
Created By:	Mike Cahow	Last Updated By:	Derrick Nagy
Date Created:	9/19/2021	Last Revision Date:	01/31/2022
Actors:	Event Planner		
Description:	Event planner will be able to schedule a date for an event and event will take up a time slot on the calendar		
Trigger:	Date field is filled out and Save button is clicked and confirmed		
Preconditions:	<ol style="list-style-type: none"> 1. User must be signed in as Event Planner 2. Event Planner must click Create Event or select to Edit an Event 		
Postconditions:	<ol style="list-style-type: none"> 1. Event Creation form is updated 		
Normal Flow:	<ol style="list-style-type: none"> 1. Event Planner is creating a new Event (use case 1009_Event Create) 2. User is prompted to add a date for the event and the start and end times 3. System verifies date and time with Planner 4. Event Planner clicks Okay (or O.K) Button 5. System displays date and time in a table and clears date/time fields 6. Event Planner decides to add another date to the event or not 7. Event Planner repeats starting at step two until all dates are added 8. Event Planner choices to move on to the next part of the form 9. System verifies that the Planner wants to move on 10. Event Planner confirms 11. System moves on to next task for the Event planner 		
Alternative Flows: [No date known]	<p>2b. In step 2 of the normal flow, if the Event Planner does not wish to manually type out the event date</p> <ol style="list-style-type: none"> 1. System will generate a calendar pop up 2. Event Planner will click on desired date and time of Event 3. System will verify that the chosen date and time are correct 4. Event Planner clicks Okay Button 5. System closes both calendar and verification pop ups 6. Use Case resumes on Step 7 of normal flow 		
[Close]	<p>4a. In step 4 of the normal flow, if the Event Planner selects Close Button instead of the Save Button</p> <ol style="list-style-type: none"> 1. System prompts the Planner to save the event 2. Planner agrees to save the event 3. User is returned to create event use case 		
[Close and no save]	<p>4b. Note: Insert a new row for each distinctive alternative flow.]</p>		

Exceptions:	<p>2a. In step 2 of the normal flow, if the Event Planner enters a date and time that overlaps or double books with another event that is already planned</p> <ol style="list-style-type: none"> 1. System notifies Planner of pre-existing event planned for this time 2. System prompts user to change the date/time 3. Planner changes date/time 4. Use case resumes on step 4 of normal flow <p>2b. In step 2 of the normal, if the Event Planner tries to enter a date and time that have happened already</p> <ol style="list-style-type: none"> 1. System notifies Planner that date/time has already happened, and event cannot be scheduled for that date/time 2. System prompts Planner to enter a valid date/time 3. Planner enters a valid date/time 4. Use Case resumes on step 4 of normal flow
Includes:	Use Case 1009_Create Event
Frequency of Use:	Use Case is executed every time an event date is set or changed, at least once per event
Special Requirements:	Event must be created before updating or editing or adding a date/time for the event
Assumptions:	That an Edit or Create Event button is created so Planner can update/edit/add date/time for event
Notes and Issues:	<ol style="list-style-type: none"> 1. Should not be able to schedule an event on a date in the past

3.2.2.6 Event Date Update

Use Case ID:	1075		
Use Case Name:	Event Date Update		
Created By:	Mike Cahow	Last Updated By:	Derrick Nagy
Date Created:	9/19/2021	Last Revision Date:	01/31/2022
Actors:	Event Planner		
Description:	As an Event Planner, I would like to edit the event date so that interested parties stay updated with new dates and times.		
Trigger:	Date field is filled out and Save button is clicked and confirmed		
Preconditions:	<ol style="list-style-type: none"> 1. User must be signed in as Event Planner 2. Event Planner must click Create Event or select to Edit an Event 3. Event already has an event date added 		
Postconditions:	<ol style="list-style-type: none"> 1. Event Date is updated 		
Normal Flow:	<ol style="list-style-type: none"> 1. Event Planner is creating a new Event (use case 1009_Event Create) 2. Event Planner has already added at least one event 3. Event Planner clicks edit event date and time 4. Event Planner can edit the date and times 5. Event Planner confirms edit 6. System verifies date and time with Planner 7. Event Planner clicks Okay (or O.K) Button 8. System displays date and time in a table and clears date/time fields 9. Event Planner decides to add another date to the event or not 10. Event Planner repeats starting at step two until all dates are added 11. Event Planner choices to move on to the next part of the form 12. System verifies that the Planner wants to move on 13. Event Planner confirms 14. System moves on to next task for the Event planner 		
Alternative Flows: [No date known]	<p>4b. In step 4 of the normal flow, if the Event Planner does not wish to manually type out the event date</p> <ol style="list-style-type: none"> 1. System will generate a calendar pop up 2. Event Planner will click on desired date and time of Event 3. System will verify that the chosen date and time are correct 4. Event Planner clicks Okay Button 5. System closes both calendar and verification pop ups 6. Use Case resumes on Step 7 of normal flow 		
[Close]	<p>5a. In step 5 of the normal flow, if the Event Planner selects Close Button instead of the Save Button</p> <ol style="list-style-type: none"> 1. System prompts the Planner to save the event 2. Planner agrees to save the event 		

	<p>3. User is returned to create event use case</p>
Exceptions:	<p>4a. In step 4 of the normal flow, if the Event Planner enters a date and time that overlaps or double books with another event that is already planned</p> <ol style="list-style-type: none"> 1. System notifies Planner of pre-existing event planned for this time 2. System prompts user to change the date/time 3. Planner changes date/time 4. Use case resumes on step 4 of normal flow <p>4b. In step 4 of the normal, if the Event Planner tries to enter a date and time that have happened already</p> <ol style="list-style-type: none"> 1. System notifies Planner that date/time has already happened, and event cannot be scheduled for that date/time 2. System prompts Planner to enter a valid date/time 3. Planner enters a valid date/time 4. Use Case resumes on step 4 of normal flow
Includes:	Use Case 1009 _Create Event
Frequency of Use:	Use Case is executed every time an event date is set or changed, at least once per event
Special Requirements:	Event must be created before updating or editing or adding a date/time for the event
Assumptions:	That an Edit or Create Event button is created so Planner can update/edit/add date/time for event
Notes and Issues:	<ol style="list-style-type: none"> 1. Should not be able to schedule an event on a date in the past

3.2.2.7 Event Date Deactivate

Use Case ID:	1076	
Use Case Name:	Event Date Deactivate	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	01/31/2022	Last Revision Date:
Actors:	Event Planner	
Description:	As an Event Planner, I would like to deactivate the event date so that the event date information is no longer seen but still able to be retrieved.	
Trigger:	Delete/Deactive Event Date	
Preconditions:	<ol style="list-style-type: none"> 1. User must be signed in as Event Planner 2. Event must be created 3. Event has at least one date 	
Postconditions:	<ol style="list-style-type: none"> 1. Event is deactivated 	
Normal Flow:	<ol style="list-style-type: none"> 1. Event Planner is creating a new Event (use case 1009_Event Create) 2. Event Planner has already added at least one event 3. Event Planner clicks delete event date and time 4. System prompts user to either delete or deactivate the event 5. Event Planner confirms deactivate 6. System deactivates the event date 7. System displays date and time in a table and clears date/time fields 8. Event Planner decides to add another date to the event or not, continuing use case 1070_Event Date Update 	
Alternative Flows: [Cancel]	4b. In step 4 of the normal flow user does not want to deactivate the event <ol style="list-style-type: none"> 1. The event planner chooses to cancel the deactivate or delete prompt 2. System returns the user to use case 1070_Event Date Update 	
Exceptions:		
Includes:	Use case 1070_Event Date Update, 1013_Event Date Create	
Frequency of Use:	Use Case is executed every time an event date is set or changed, at least once per event	
Special Requirements:	Event must be created before updating or editing or adding a date/time for the event	
Assumptions:		

	That an Edit or Create Event button is created so Planner can update/edit/add date/time for event
Notes and Issues:	

1.1.1.1.1 Event Date Delete

Use Case ID:	1077		
Use Case Name:	Event Date Deactivate		
Created By:	Derrick Nagy	Last Updated By:	
Date Created:	01/31/2022	Last Revision Date:	
Actors:	Event Planner		
Description:	As an Event Planner, I would like to delete the event date so that the event date is permanently removed.		
Trigger:	Delete/Deactivate Event Date		
Preconditions:	1. User must be signed in as Event Planner 2. Event must be created 3. Event has at least one date		
Postconditions:	1. Event is deactivated		
Normal Flow:	1. Event Planner is creating a new Event (use case 1009_Event Create) 2. Event Planner has already added at least one event 3. Event Planner clicks delete event date and time 4. System prompts user to either delete or deactivate the event 5. Event Planner confirms delete 6. System warns user that this action can not be undone 7. User confirms choice 8. System deletes the event date 9. System displays date and time in a table and clears date/time fields 10. Event Planner decides to add another date to the event or not, continuing use case 1070_Event Date Update		
Alternative Flows: [Cancel]	4b. In step 4 of the normal flow user does not want to delete the event date <ol style="list-style-type: none"> The event planner chooses to cancel the deactivate or delete prompt System returns the user to use case 1070_Event Date Update 		
Exceptions:			
Includes:	Use case 1070_Event Date Update, 1013_Event Date Create		
Frequency of Use:	Use Case is executed every time an event date is set or changed, at least once per event		
Special Requirements:	Event must be created before updating or editing or adding a date/time for the event		
Assumptions:	That an Edit or Create Event button is created so Planner can update/edit/add date/time for event		

Notes and Issues:	
--------------------------	--

3.2.2.8 Event Edit

Use Case ID:	1010		
Use Case Name:	Event Edit		
Created By:	Mike Cahow	Last Updated By:	
Date Created:	9/23/2021	Last Revision Date:	
Actors:	Event Planner Event Organizational Rep		
Description:	The ability to edit/finish a previously started Create Event Form		
Trigger:	Clicks an edit button on an events description page		
Preconditions:	1. Event must have been created 2. User must be signed in as Event Planner or Event Organizational Rep		
Postconditions:	1. Event's description/information page is updated with any fields updated/filled in 2. Event's create/edit form is closed		
Normal Flow:	1. Event Planner selects a created event to edit 2. Event Planner selects the edit event button 3. System pulls the saved Create Event form with all previously filled in fields displayed 4. Event Planner selects which field to update 5. Event Planner inputs the required information for desired field 6. System displays inputted information in corresponding field 7. Event Planner selects the Save Form button 8. System asks if Event Planner wants to overwrite the current file 9. Event Planner confirms 10. System overwrites previously saved Form with the Form currently being worked with 11. System displays a success message when save is successful 12. Event Planner selects the Close button 13. System closes the even create/edit form		
Alternative Flows: [Multiple Fields]	7a. In step 7 of the normal flow, if the Event Planner has multiple fields to fill out <ol style="list-style-type: none"> Event Planner will repeat steps 4 – 6 of normal flow until they are finished filling out the information they are able to. Use Case resumes on step 7 of normal flow 7b. In step 7 of the normal flow, if the Event Planner selects the Close button instead of Save		
[Close]			

<p>[Close and not save]</p>	<ol style="list-style-type: none"> 1. System will prompt the Event Planner to save as the data will be deleted otherwise 2. Event Planner agrees 3. Steps 10 and 11 of normal flow 4. Use Case resumes on step 14 of normal flow <p>7c. In step 7 of the normal flow, if the Event Planner selects the Close button instead of Save</p> <ol style="list-style-type: none"> 1. System will prompt the Event Planner to save, as the data will be deleted otherwise 2. Event Planner declines 3. System will discard any changes made to the form and return it to the way it was when the Event Planner pulled it up. 4. Use Case resumes on step 14 of normal flow
<p>Exceptions:</p>	<p>7a. In step 7 of the normal flow, if the Event Planner deleted the Event Name field and did not replace it</p> <ol style="list-style-type: none"> 1. System generates a message saying that event needs a name to continue 2. Event Planner Confirms 3. System focuses on Name field 4. Event Planner enters a name 5. System displays name 6. Use Case resumes on step 7 of normal flow <p>7b. In step 7 of the normal flow, if the Event Planner deleted the Event Description field and did not replace it</p> <ol style="list-style-type: none"> 1. System generates a message saying that event needs a description to continue 2. Event Planner Confirms 3. System focuses on Description field 4. Event Planner enters a description 5. System displays the entered description 6. Use Case resumes on step 7 of normal flow <p>7c. In step 7 of the normal flow. If the Event Planner deleted the date fields contents and did not replace it</p> <ol style="list-style-type: none"> 1. System generates a message saying that if the event does not have a date, then it will be removed from its spot from the calendar 2. System prompts user to re-enter previous date or update the date field 3. Event Planner confirms 4. Event Planner re-enters or updates the date field 5. System displays the entered date 6. Use Case resumes on step 7 of normal flow <p>6a. If date was changed from the original date, system will update the calendar accordingly after saving file</p> <p>7d. In step 7 of the normal flow, if the Event Planner deleted the date fields contents and did not replace it</p> <ol style="list-style-type: none"> 1. System generates a message saying that if the event does not have a date, then it will be removed from its spot from the calendar 2. System prompts the Event Planner to re-enter previous date or update the date field

	<ol style="list-style-type: none"> 3. Event Planner declines 4. System will take event off previously specified time slot from the calendar 5. Use case resumes on step 7 of normal flow
Includes:	Uses Cases: 1009_Event Creation 1011_Event Location 1012_Location Directions 1013_Event Date(s) 1014_Add Vendor Tags 1015_Add Volunteer
Frequency of Use:	Used multiple time during the event creation process to edit and work out any fine details needed for events
Special Requirements:	None at this time
Assumptions:	That the event being updated has already been created
Notes and Issues:	<ol style="list-style-type: none"> 1. Calendar is to be updated if date is modified or deleted after saving the edited form 2. Event Organizational Rep is able to follow same flow, but is only able to update following fields: Name, Description, Location, Date

1.1.1.2 Event Date Delete

Use Case ID:	1077		
Use Case Name:	Event Date Delete		
Created By:	Derrick Nagy	Last Updated By:	
Date Created:	01/31/2022	Last Revision Date:	
Actors:	Event Planner		
Description:	As an Event Planner, I would like to delete the event date so that the event date is permanently removed.		
Trigger:	Delete/Deactive Event Date		
Preconditions:	<ol style="list-style-type: none"> 1. User must be signed in as Event Planner 2. Event must be created 3. Event has at least one date 		
Postconditions:	<ol style="list-style-type: none"> 1. Event is deactivated 		
Normal Flow:	<ol style="list-style-type: none"> 1. Event Planner is creating a new Event (use case 1009_Event Create) 2. Event Planner has already added at least one event 3. Event Planner clicks delete event date and time 4. System prompts user to either delete or deactivate the event 5. Event Planner confirms deactivate 6. System deactivates the event date 7. System displays date and time in a table and clears date/time fields 8. Event Planner decides to add another date to the event or not, continuing use case 1070_Event Date Update 		
Alternative Flows: [Cancel]	4b. In step 4 of the normal flow user does not want to deactivate the event <ol style="list-style-type: none"> 1. The event planner chooses to cancel the deactivate or delete prompt 2. System returns the user to use case 1070_Event Date Update 		
Exceptions:			
Includes:	Use case 1070_Event Date Update, 1013_Event Date Create		
Frequency of Use:	Use Case is executed every time an event date is set or changed, at least once per event		
Special Requirements:	Event must be created before updating or editing or adding a date/time for the event		
Assumptions:			

	That an Edit or Create Event button is created so Planner can update/edit/add date/time for event
Notes and Issues:	

3.2.2.9 Event Location

Use Case ID:	1011		
Use Case Name:	Event Location		
Created By:	Mike Cahow	Last Updated By:	Logan Baccam
Date Created:	9/19/2021	Last Revision Date:	02/04/2022
Actors:	Event Planner		
Description:	Adding or editing an event creation form to have the location of where the event will be taking place		
Trigger:	User selects the Create or Edit Event button		
Preconditions:	<ol style="list-style-type: none"> 1. User must be logged in as an Event Planner 2. User must know where the event is going to be held 		
Postconditions:	<ol style="list-style-type: none"> 1. Location is set in the system 		
Normal Flow:	<ol style="list-style-type: none"> 1. Event Planner enters new event information when creating a new event in the Create new event form 2. Event Planner clicks the Next button 3. Event planner is brought to the Select location form 4. Event planner enters location information 5. Event planner clicks the Next button 6. Event planner successfully sets the location for the event and is prompted that the location has been entered 		
Alternative Flows: [Cancel]	<p>5a. In step 5 of the normal flow, if the Event Planner click the Cancel button instead of the Next Button</p> <ol style="list-style-type: none"> 1. System returns user to the Event Listing screen. 		
Exceptions:	<p>2a. In step 2 of normal flow, if the planner attempts to click next when adding invalid location values</p> <ol style="list-style-type: none"> 1. System will prompt the user to enter correct values 2. Planner clicks ok 3. System focuses on the location fields 4. Planner enters correct information 5. Step 2 of normal flow is repeated 		
Includes:			
Frequency of Use:			

	Whenever an event is created or edited with the knowledge of a location for the event
Special Requirements:	
Assumptions:	There are Create and Edit buttons created for the users to select
Notes and Issues:	<ol style="list-style-type: none"> 1. Should the system alert everyone who is running or participating in the event if the location should change?(Derrick Nagy) 2. Event Date Create should be reserving the location. Move this to before it?

3.2.2.10 Event Location Directions

Use Case ID:	1012		
Use Case Name:	Event Location Directions		
Created By:	Mike Cahow	Last Updated By:	Mike Cahow
Date Created:	9/23/2021	Last Revision Date:	12/2/21
Actors:	Event Planner		
Description:	Create or edit directions pertaining to event		
Trigger:	After a location is selected display an option to put in directions or a link to a Google Maps		
Preconditions:	<ol style="list-style-type: none"> 1. User must be logged in as an Event Planner 2. Event Planner must have selected a location for the event 		
Postconditions:	<ol style="list-style-type: none"> 1. Directions to the Event's location are uploaded and attached to a link on the event's information page 2. Directions are saved with information pertaining to the location in case location is used for a future event 		
Normal Flow:	<ol style="list-style-type: none"> 1. Event Planner finishes selecting the Event's Location 2. System prompts the Event Planner if they want to add locations to the event 3. Event Planner agrees 4. System checks if event has previously saved directions for the location 5. System prompts the Event Planner to fill out directions for the location since no previously saved ones could be found 6. System generates a text box for the directions to be entered into 7. Event Planner details directions for getting to the event location from main road around location 8. Event Planner clicks the Save Directions box 9. System prompts Event Planner to save this for future use as well 10. Event Planner agrees 11. System saves directions to the information pertaining to the location for future use 12. System closes text box 13. System then links the directions to a link on the event page by event's location 		
Alternative Flows: [No directions]	<p>3a. In step 3 of the normal flow, if the Event Planner does not want to add directions to the location of the event</p> <ol style="list-style-type: none"> 1. Event Planner declines 		

<p>[Previously Saved Locations]</p> <p>[Cancel Changes]</p>	<ol style="list-style-type: none"> 2. System will close message box 3. Use Case ends <p>5a. In step 5 of the normal flow, if the Location has previously saved directions for use</p> <ol style="list-style-type: none"> 1. System will prompt Event Planner to use a previously saved set of directions to event 2. Event Planner views the previously saved directions 3. Event Planner updates the directions if a better route has been decided on 4. Event Planner selects Use Directions <p>4a. If updated, system overwrites previous directions with new copy</p> <ol style="list-style-type: none"> 1. Use Case resumes on step 12 of normal flow <p>10a. In step 10 of the normal flow, if the Event Planner does not wish to save the directions for future use</p> <ol style="list-style-type: none"> 1. Event Planner declines 2. Use Case resumes on step 12 of normal flow
<p>Exceptions:</p>	<p>5a. In 5a step 3 of the alternate flow, if the Event Planner deletes the directions and clicks Use Directions without entering any new data</p> <ol style="list-style-type: none"> 1. System will prompt the Event Planner to enter in new directions or to use the old directions or to delete directions 2. Event Planner selects enter in new directions 3. Event Planner enters in the new directions 4. Use Case resumes on 5a's step 4 of normal flow <p>5b. In 5a step 3 of the alternate flow, if the Event Planner deletes the directions and clicks Use Directions without entering any new data</p> <ol style="list-style-type: none"> 1. System will prompt the Event Planner to enter in new directions or to use the old directions or to delete directions 2. Event Planner selects use the old directions 3. System re-populates box with the previous data that populated it 4. Use Case resumes on 5a's step 4 of normal flow <p>5c. In 5a step 3 of the alternate flow, if the Event Planner deletes the directions and clicks Use Directions without entering any new data</p> <ol style="list-style-type: none"> 1. System will prompt the Event Planner to enter in new directions or to use the old directions or to delete directions 2. Event Planner selects delete directions 3. System asks if it is okay to permanently delete directions from locations information storage 4. Event Planner agrees 5. System deletes* the directions from the locations information storage 6. Use case ends
<p>Includes:</p>	<p>Use Cases: 1009_Create Event 1010_Edit Event 1011_Event Location</p>
<p>Frequency of Use:</p>	<p>Used anytime an event's location is designated or directions need to be updated</p>

Special Requirements:	Location will need some sort of information data storage in order to draw the previously saved directions out of
Assumptions:	People will need directions to events and use the directions provided
Notes and Issues:	<ol style="list-style-type: none">1. How many sets of directions can we store?2. Should we just leave it up to a GPS app and link the people to the GPS?

3.2.2.11 Event Share

Use Case ID:	1007		
Use Case Name:	Event Share		
Created By:	Mike Cahow	Last Updated By:	
Date Created:	9/24/2021	Last Revision Date:	
Actors:	Event Organizational Rep		
Description:	Allows the Event Organizational Rep share a Created Event with an Event Planner so the Event Planner can take the reigns and plan the Event		
Trigger:	Trigger happens when the Event Organizational Rep selects a share event with planner button to select a list of registered Event Planner users to share event with		
Preconditions:	<ol style="list-style-type: none"> 1. User must be logged in as a Event Organizational Rep 2. Event Organizational must have an event created to share 		
Postconditions:	<ol style="list-style-type: none"> 1. Permissions for the event are passed along to the designated Event Planner 2. Pop up screen closes, returning user to previous screen 		
Normal Flow:	<ol style="list-style-type: none"> 1. Event Organizational Rep (EOR) selects the event created by them 2. System generates the most recently updated Create Event Form 3. System displays all fields previously filled out by the EOR 4. EOR selects the button to Share with Event Planner 5. System prompts EOR to select from a list of users with Event Planner permissions 6. System sifts through database for the list of Event Planners 7. System displays list of Event Planners 8. EOR selects an Event Planner from list 9. System asks if the select Event Planner is the final choice 10. EOR confirms 11. System generates a basic message with Event's permissions embedded in 12. System confirms that the message and permissions are okay to send 13. EOR confirms 14. System sends message to designated Event Planner with permissions 15. ... 16. System closes the share event page, returning EOR to event selection page 		
Alternative Flows:			

[Alternative Flow 1 – Not in Network]	<p>4a. In step 4 of the normal flow, if the EOR was not the original creator of the event</p> <ol style="list-style-type: none"> 1. System checks the EOR's ID to make sure it matches with the original creators 2. System verifies that the ID's do not match 3. System raises exception "Only the original creator is allowed to pass event along to Event Planner" 4. Use Case resumes on step 1 <p>9a. In step 9 of the normal flow, if the EOR does not wish to proceed with currently selected Event Planner</p> <ol style="list-style-type: none"> 1. System asks if the selected Event Planner is okay to send permissions for event to 2. EOR declines 3. Message and permissions generation cancelled 4. Use Case resumes on step 7 of normal flow
Exceptions:	<p>4a. In step 4 of the normal flow, if the EOR was not the original creator of the event</p> <ol style="list-style-type: none"> 1. Sharing is disapproved 2. Message to EOR that only the EOR that created the event can share the event 3. EOR confirms message 4. System closes sharing screen 5. Use Case resumes on step 1 of normal flow
Includes:	<p>Use Cases: 1009_Create Event 1010_Edit Event 1011_Event Location 1013_Event Date(s)</p>
Frequency of Use:	<p>Whenever an event is created by an EOR</p>
Special Requirements:	<p>The share button should only be available to the EOR that created the event</p>
Assumptions:	<p>That the EOR has created the event</p>
Notes and Issues:	<ol style="list-style-type: none"> 1. What if there aren't any Event Planner's nearby

3.2.2.12 File Delete Relevant Files

Use Case ID:	1053	
Use Case Name:	File Delete Relevant Files	
Created By:	Mike Cahow	Last Updated By:
Date Created:	9/24/2021	Last Revision Date:
Actors:	Event Planner	
Description:	Deleting unnecessary files from the important information page	
Trigger:	Event Planner clicks delete file button	
Preconditions:	<ol style="list-style-type: none"> 1. Must be signed in as Event Planner 2. Files must be uploaded to important information page 	
Postconditions:	<ol style="list-style-type: none"> 1. File is deleted 2. System updates page to no longer display deleted file 	
Normal Flow:	<ol style="list-style-type: none"> 1. Event Planner selects file to view 2. System displays file in its entirety 3. Event Planner selects the delete file button 4. System asks if the Event Planner is sure if they want to delete file 5. Event Planner agrees 6. System deletes file 7. System closes the file view screen 8. System updates the page to no longer display the deleted file 	
Alternative Flows: [Alternative Flow 1 – Not in Network]	<p>4a. In step 4 of the normal flow, if the Event Planner does not want to delete the file</p> <ol style="list-style-type: none"> 1. System asks if the Event Planner is sure if they want to delete file 2. Event Planner declines 3. File is kept in view mode until Event Planner exits 4. Use Case ends 	
Exceptions:		
Includes:	Use Cases: 1052_View Relevant Files	
Frequency of Use:	Whenever there are unnecessary files to delete from the important information page	

Special Requirements:	Must be an Event Planner to delete the files
Assumptions:	There are files that need to be deleted on the important information page
Notes and Issues:	

3.2.2.13 File Upload

Use Case ID:	1024		
Use Case Name:	File Upload		
Created By:	Mike Cahow	Last Updated By:	Mike Cahow
Date Created:	9/12/2021	Last Revision Date:	9/24/21
Actors:	Event Planner		
Description:	As an Event Planner, I would like to upload files relevant to event (i.e. documents, permits, contracts, maps) so that anyone (attendee, law enforcement, etc) who needs to see relevant information for the event can view it.		
Trigger:	Click an Upload File button on event information page		
Preconditions:	<ol style="list-style-type: none"> 1. Must be logged in as Event Planner 2. Click on Event page and find event file 3. Click upload file button 		
Postconditions:	<ol style="list-style-type: none"> 1. Document is uploaded 2. Changes made to page saved 		
Normal Flow:	<ol style="list-style-type: none"> 1. Event Planner logs in and navigates to Event Page 2. Event Planner clicks event information page 3. Event Planner clicks upload files 4. System brings up a file finder screen 5. Event Planner selects file(s) to upload 6. Event Planner enters the event file name or system uses the previous one 7. 8. Event Planner clicks to confirm files to be uploaded 9. System closes file selection screen 10. System uploads selected files to event information page 11. System updates event information page to display newly uploaded files 12. System confirms upload with a notification 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<p>6a. In step 6 of the normal flow, if one or more selected files has the same name as a file on the page</p> <ol style="list-style-type: none"> 1. System will verify that there is already a file with that name and if it is alright to overwrite the existing file with the file being uploaded 2. Planner accepts 3. Use Case resumes on step 7 		

	<p>6b. In step 6 of the normal flow, if one or more selected files has the same name as a file on the page</p> <ol style="list-style-type: none"> 1. System will verify that there is already a file with that name and if it is alright to overwrite the existing file with the file being uploaded 2. Planner declines 3. System closes file selection screen with no files being uploaded 4. Use Case resumes on step 3 of normal flow
Exceptions:	<p>5a. In step 5 of the normal flow, if the Planner enters the wrong name of a file</p> <ol style="list-style-type: none"> 1. Selection isn't approved 2. Message to Planner re-enter a filename that exists 3. Planner enters correct file name 4. Use Case resumes on step 6 of normal flow
Includes:	
Frequency of Use:	This feature is to be used whenever permits are necessary for events, and for general information or pictures of the event and area being uploaded for attendees' ease of use
Special Requirements:	Event information page must be created to hold files
Assumptions:	Event requires a permit to proceed Attendees need a map to the event
Notes and Issues:	<ol style="list-style-type: none"> 1. What is the max number of files that can be uploaded at once?

3.2.2.14 File View Relevant Files

Use Case ID:	1052	
Use Case Name:	File View Relevant Files	
Created By:	Mike Cahow	Last Updated By:
Date Created:	9/24/2021	Last Revision Date:
Actors:	Event Planner, Event Organizational Rep, Event Manager, Event Promoter	
Description:	As an Event Planner, Event Manager, Event Organizational Rep, and Event Promoter, I would like to view relevant files for an event so that anyone needing to see the information can do so.	
Trigger:	A load event for the important information page for an event is select/loaded	
Preconditions:	1. User must be logged on 2. User selects the important information page	
Postconditions:	1. User viewed the relevant files	
Normal Flow:	1. User clicks on event page 2. User clicks on the important information page 3. System generates important information page 4. System displays any files uploaded to the page 5. User clicks on the files 6. System displays files in the entirety 7. User views file 8. User clicks the close button 9. System closes file view	
Alternative Flows: [Alternative Flow 1 – Not in Network]	4a. In step 4 of the normal flow, if there aren't any files to display 1. System generates a message where files would go saying that there aren't any files uploaded yet 2. Use Case ends	
Exceptions:	4a. In step 4 of the normal flow, if there is an issue loading the files 1. System generates a message saying there was a problem loading the files 2. System prompts user to reload page 3. Use case resumes on step 3 of normal flow	

Includes:	Use Case: 1024_Uplod Relevant Files
Frequency of Use:	Whenever users need to view special document or maps, etc
Special Requirements:	In order to view the files, files must be uploaded. Permits should only be viewable to those working the event, not attendees
Assumptions:	Relevant Files have been uploaded
Notes and Issues:	<p>1. What if no files have been uploaded?</p>

3.2.2.15 Event Calendar View

Use Case ID:	1021		
Use Case Name:	Event Calendar View		
Created By:	Chase Fields	Last Updated By:	
Date Created:	9/15/2021	Last Revision Date:	9/20/2021
Actors:	Event Planner, Manager, Organizational Rep, Event Promoter		
Description:	View a Calendar of Events.		
Trigger:	User Selects to search for events		
Preconditions:	<ol style="list-style-type: none"> 1. The User has view privileges. 2. Events that have been created are displayable on the calendar. 		
Postconditions:	<ol style="list-style-type: none"> 1. A calendar will render viewable events based on the user search parameters of date and location 		
Normal Flow:	<ol style="list-style-type: none"> 1. User selects to search for events. 2. System displays a page with a search by location and date options. 3. User enters a date range. 4. User chooses location from options state, county, city, zip 5. User selects to "find events" 6. System validates that those parameters are sufficient to find events 7. System Filters all active events based on the date range and location parameters chosen. 8. Calendar populates with events for the user to select and view details. 		
Alternative Flows:			
Exceptions:	<ol style="list-style-type: none"> 6a. In step 6 of normal flow, there is not enough info provided to find events. <ol style="list-style-type: none"> 1. System tells user that there is not enough info to find events (for example, there is a city but no state or zip) 2. User clicks off the error message. 3. User re-enters parameters. 4. Resume at step 7 of normal flow. 		
Includes:			
Frequency of Use:			
Special Requirements:			
Assumptions:			
Notes and Issues:	<ol style="list-style-type: none"> 1. Are there other parameters that ought to be allowable to create the display? Such as event type? 		

3.2.2.16 Message Planning Team

Use Case ID:	1022		
Use Case Name:	Message Planning Team		
Created By:	Derrick Nagy	Last Updated By:	Derrick Nagy
Date Created:	9/13/2021	Last Revision Date:	9/19/2021
Actors:	Event Planner(primary), Event Manager(secondary), Event Organizational Rep(secondary), Event Promoter(secondary)		
Description:	User sends and receives messages with team		
Trigger:	<ol style="list-style-type: none"> 1. Team member wants to message the team about a topic. 2. Any user involved in an event wants to message another user 3. An event is created 4. New users are added to an event 		
Preconditions:	<ol style="list-style-type: none"> 1. There must be a community event that has been created 2. Users with roles have been added to the event 		
Postconditions:	<ol style="list-style-type: none"> 1. <u>Minimal guarantee:</u> Messages are archived, can be accessed 2. <u>Success guarantee:</u> Messages can have a response or be acknowledged 		
Normal Flow:	<ol style="list-style-type: none"> 1. System creates a page for messaging 2. User creates a team message 3. User selects who receives it 4. Recipients can add to message thread 		
Alternative Flows: [Delete Message]	<ol style="list-style-type: none"> 1. User creates a message 2. User does not like the message and would like it deleted 		
Exceptions:	<p>1a. Wrong user is added to chat</p> <ol style="list-style-type: none"> 1. Check the list of potential people to send the message to 2. Use Case resumes step 1 of normal flow <p>1b. Can not find user to send message to</p> <ol style="list-style-type: none"> 1. Check the list of current group members to make sure that the user doesn't exist 2. Invite the user to the event with message 		
Includes:	Create event		
Frequency of Use:	Often during the planning and execution phase of the event		
Special Requirements:			
Assumptions:	User has created a group and they know that the recipients are in correctly		
Notes and Issues:	<ol style="list-style-type: none"> 1. Who would really need to be messaged? Everyone? What could be accomplished with this better than what already exists (like a phone or other kind of messaging service)? 		

3.2.2.17 Vendor Tags Add

Use Case ID:	1014		
Use Case Name:	Vendor Tags Add		
Created By:	Asa Armstrong	Last Updated By:	Asa Armstrong
Date Created:	9/23/2021	Last Revision Date:	9/23/2021
Actors:	Event Planner (primary), Event Manager (secondary), Event Organization Rep (secondary)		
Description:	Set the tags for what kind of vendors wanted		
Trigger:	User selects to set what kind of vendors they want		
Preconditions:	<ol style="list-style-type: none"> 1. An event is created 2. The event is going to have vendors 		
Postconditions:	<ol style="list-style-type: none"> 1. Vendors can view what kind of vendor this event wants 2. Alert vendor accounts with the specified tags selected 		
Normal Flow:	<ol style="list-style-type: none"> 1. User selects to add vendor tags for their event 2. A list populates with vendor tags 3. User selects the desired tags 4. User saves the changes 		
Alternative Flows: [Alternative Flow 1 – Changes Not Saved]	4a. In step 4 of the normal flow, if the user cancels the changes <ol style="list-style-type: none"> 1. No changes are saved 		
[Alternative Flow 2 – Save on Close]	4a. In step 4 of the normal flow, if the user leaves without saving <ol style="list-style-type: none"> 1. User prompted to save or cancel the changes 		
Exceptions:	1.		
Includes:			
Frequency of Use:			
Special Requirements:			
Assumptions:			
Notes and Issues:			

3.2.2.18 Volunteers Add

Use Case ID:	1015	
Use Case Name:	Volunteers Add	
Created By:	Mike Cahow	Last Updated By:
Date Created:	9/23/2021	Last Revision Date:
Actors:	Event Planner	
Description:	Add an add volunteer field to be updated	
Trigger:		
Preconditions:	1. User must be logged in as an Event Planner 2. Event Planner must either select create or edit event	
Postconditions:	1. System sets Vendor positions open 2. System notifies available vendors within a designated radius that an event needs volunteers	
Normal Flow:	1. Event Planner clicks Edit Event button 2. System pulls Create Event Form file from storage and displays all previously filled in fields 3. Event Planner selects the Add Volunteer dropdown button 4. Event Planner selects yes for the field 5. System prompts user to enter the number of volunteers needed for event 6. Event Planner specifies number of volunteers needed for the event 7. System displays number of volunteers requested 8. Event Planner clicks the Save button 9. System prompts Event Planner with a message saying saving will overwrite the data currently saved to the file 10. Event Planner agrees 11. System saves the copy of the form being worked on to the file, overwriting any data already in there 12. Event Planner selects the Close button 13. System Closes the Create Event Form 14. System updates the Events page to display that the event needs volunteers 15. System sends a notification to available volunteers in the area that an event needs volunteers for the date(s) event will be happening	
Alternative Flows: [Alternative Flow 1 – No Event]	1a. In step 1 of the normal flow, if the Event Planner has not yet created an event to be edited 1. Event Planner selects the Create Event button 2. System generates a blank Create Event Form	

[Alternative Flow 2 – No volunteers]	<ol style="list-style-type: none"> 3. Event Planner fills out Name, Description, Date and Location fields 4. System displays each field after completing them 5. Use Case resumes on step 3 of normal flow
[Alternative Flow 3 – Save on close]	<ol style="list-style-type: none"> 4a. In step 4 of the normal flow, if the Event Planner does not need any volunteers for the event <ol style="list-style-type: none"> 1. Event Planner selects no for the field 2. System generates a message saying they can update this any time in the event edit screen 3. Event Planner clicks Okay 4. System displays the no in the Add Volunteer field 5. Use Case resumes on step 8 of normal flow
[Alternative Flow 4 – Close save or cancel]	<ol style="list-style-type: none"> 8a. In step 8 of the normal flow, if the Event Planner clicks the Close button instead of the Save button <ol style="list-style-type: none"> 1. System prompts the Event Planner to save or else all the changes will be discarded and will have to be redone 2. Event Planner agrees 3. System saves current state of the Create Event Form to the file, overwriting any data existing in the file 4. Use Case resumes on step 12 of normal flow 8b. In step 8 of the normal flow, if the Event Planner clicks the Close button instead of the Save Button <ol style="list-style-type: none"> 1. System prompts the Event Planner to save or else all the changes will be discarded and will have to be redone 2. Event Planner declines 3. System closes Create Event Form 4. System discards any changes made since previous save of form
Exceptions:	<ol style="list-style-type: none"> 8a. In step 8 of the normal flow, if the Event Planner tries to save and close the form without specifying location and date for volunteers <ol style="list-style-type: none"> 1. System prompts Event Planner to specify the location and date of the event so it can notify volunteers 2. System focuses on date or location fields, depending on which one is missing 3. Event Planner fills out missing field(s) 4. System displays entered information 5. Use Case resumes on step 8 of normal flow
Includes:	Uses Cases: 1009_Event Creation 1010_Edit Event 1011_Event Location 1012_Location Directions 1013_Event Date(s) 1014_Add Vendor Tags
Frequency of Use:	Used whenever a Create Event form is edit to say whether or not volunteers are needed for an event
Special Requirements:	

	Top rated volunteers will be recommended the volunteer positions before other volunteers
Assumptions:	That events will need volunteers to work them That there are volunteers in the area looking to work the events
Notes and Issues:	<ol style="list-style-type: none"> 1. What is the maximum number of volunteers an event can have? 2. Should this feature deal with notifying volunteers of events when the field becomes populated with a yes and a number of volunteers needed? 3. How close does a volunteer need to be located to receive a message?

3.2.2.19 Create Event Roles

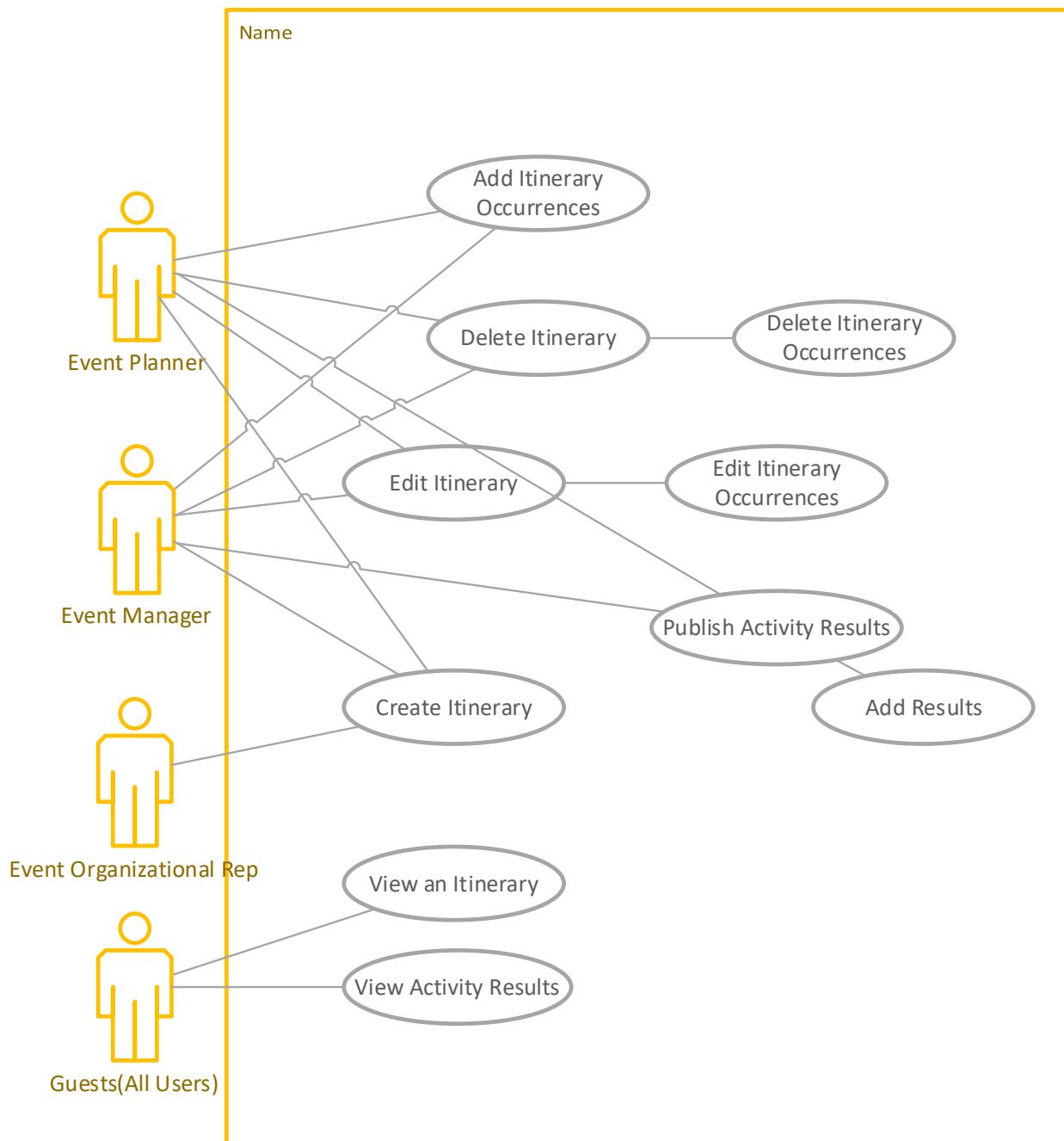
Use Case ID:	1078 Create Event Roles	
Use Case Name:	Create Event Roles	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	02/18/2022	Last Revision Date:
Actors:	Event Planner(Primary), Admin	
Description:	The event planner needs to add the roles that are needed to manage, plan, and administer the event to the user that creates the event.	
Trigger:	Create Event	
Preconditions:	<ol style="list-style-type: none">1. User must be logged in2. Event has been created and validated	
Postconditions:	1. User has "Event Planner, Event Manager" roles added to the event that they created	
Normal Flow:	<ol style="list-style-type: none">1. User creates an event and confirms that the information is correct2. System adds Event Planner, Event Manager roles to the roles that the user has	
Alternative Flows:		
Exceptions:		
Includes:	1009_Event Create	
Frequency of Use:		
Special Requirements:		
Assumptions:		
Notes and Issues:	1.	

3.2.2.20 Event Duplication

Use Case ID:	1079		
Use Case Name:	Duplicating an Existing Event		
Created By:	Vinayak Deshpande	Last Updated By:	
Date Created:	2022-03-31	Last Revision Date:	
Actors:	Event Planner or other member of management team		
Description:	Event Team wants to rerun an event and doesn't want to have to recreate it		
Trigger:	Duplicate button is pressed while event is selected		
Preconditions:	<ol style="list-style-type: none"> 1. Event to duplicate exists 2. Tasks to duplicate exist 3. Activities to duplicate exist 		
Postconditions:	<ol style="list-style-type: none"> 1. Event and associated tasks and activities are duplicated 		
Normal Flow:	<ol style="list-style-type: none"> 1. Event is duplicated 2. Event Dates are added 3. Event Location Duped 4. General Help Task Duplicated 5. Tasks are duplicated 6. Activities are duplicated 		
Alternative Flows:			
Exceptions:	<ol style="list-style-type: none"> 1. No Tasks to duplicate 2. No Activities to duplicate 3. No original location 		
Includes:	1009_Event_Creation		
Frequency of Use:	Will be used for recurring events		
Special Requirements:			
Assumptions:	Event to duplicate is a real event		
Notes and Issues:			

3.2.3 Itinerary

3.2.3.1 Itinerary Use Case Model



3.2.3.2 Use Cases

3.2.3.2.1 Activity Results

Use Case ID:	1029		
Use Case Name:	Activity Results		
Created By:	Asa Armstrong	Last Updated By:	Derrick Nagy
Date Created:	9/23/2021	Last Revision Date:	11/18/2021
Actors:	Event Planner (primary), Event Manager (secondary)		
Description:	Display activity results on the forum.		
Trigger:	An activity has results added to it		
Preconditions:	<ol style="list-style-type: none">1. There is an activity2. Results are added to the activity		
Postconditions:	<ol style="list-style-type: none">1. The results can be seen on the forum of the event		
Normal Flow:	<ol style="list-style-type: none">1. User adds results to an activity2. User chooses to post the results to the forum3. System creates post in the Main Event forum that summarizes the results		
Alternative Flows: [Alternative Flow 1 – Not in Network]	2a. In step 2 of the normal flow, if the user doesn't add the results to the forum <ol style="list-style-type: none">1. The results won't be added to the forum		
Exceptions:			
Includes:	1030_Activity Results Add 1048_Forum Create Post		
Frequency of Use:	Every time new results to an event are added		
Special Requirements:			
Assumptions:			
Notes and Issues:			

3.2.3.2.2 Activity Results Add

Use Case ID:	1030	
Use Case Name:	Activity Results Add	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	09/22/2021	Last Revision Date:
Actors:	Event Manager	
Description:	Add the results of the activity so that the Event Manager can record it.	
Trigger:	An activity is over, and the results can be posted	
Preconditions:	Must be an event created with activities in the itinerary.	
Postconditions:	<ol style="list-style-type: none"> 1. The results of the activity are shown, or default message given 2. System redirects user to the Activity Results view 	
Normal Flow:	<ol style="list-style-type: none"> 1. Event Manager starts event to add results to activity 2. Event Manager selects the activity from the event that they would like to add results to 3. Event Manager can choose to add a ranking list, a text description, or a picture to the activity showing who won the activity or the summary of results from it 4. Event Manager approves the changes 5. System redirects user to view the results 	
Alternative Flows: [Alternative Flow 1 – Description]		
Exceptions:	3a. Event Manager tries to add a picture that isn't supported by the system <ol style="list-style-type: none"> 1. System notifies user that the picture is in the wrong format or too large 2. System prompts user to try a different file or modify the picture 3. Event Manager returns to step 3 	
Includes:	Activity Results, Itinerary	
Frequency of Use:	Only a couple of times per activity, and only for activities that require results to be posted	
Special Requirements:		
Assumptions:		
Notes and Issues:	<ol style="list-style-type: none"> 1. 	

3.2.3.2.3 Activity Delete

Use Case ID:	1073		
Use Case Name:	Activity Delete		
Created By:	Derrick Nagy	Last Updated By:	
Date Created:	01/25/2022	Last Revision Date:	
Actors:	Event Planner(Primary), Event Manager, Event Organizational Rep		
Description:	As an Event Planner, I would like to delete an activity for an event because it is no longer used		
Trigger:	View Activity Details		
Preconditions:	<ol style="list-style-type: none"> 1. Event must be created 2. The event dates must be chosen (1013_Event Date(s)) 3. Activity has been added to an Event Date(1060_Event Date Activities Create) 4. User is viewing a list of activities (1062_Event Date Activities Retrieve All Event Dates, 1061_Event Date Activities Retrieve Single Day) 5. User has permissions to deactivate 		
Postconditions:	1. User can view the activity for the particular event		
Normal Flow:	<ol style="list-style-type: none"> 1. User selects activity to view in more detail 2. User selects delete 3. System prompts user to confirm 4. Activity is deleted 		
Alternative Flows:			
Exceptions:			
Includes:	1013_Event Date(s), 1061_Event Date Activities Retrieve Single Day, 1062_Event Date Activities Retrieve All Event Dates, 1060_Event Date Activities Create, 1068_Activity Details		
Frequency of Use:			
Special Requirements:			
Assumptions:			
Notes and Issues:	1.		

3.2.3.2.4 Activity Details

Use Case ID:	1068	
Use Case Name:	Activity Details	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	01/25/2022	Last Revision Date:
Actors:	Event Planner(Primary), Event Manager, Event Organizational Rep	
Description:	As an Event Planner, Event Manager, or Event Organizational Rep, I would like to view the details of an individual event	
Trigger:	View Activity Details	
Preconditions:	<ol style="list-style-type: none"> 1. Event must be created 2. The event dates must be chosen (1013_Event Date(s)) 3. Activity has been added to an Event Date(1060_Event Date Activities Create) 4. User is viewing a list of activities (1062_Event Date Activities Retrieve All Event Dates, 1061_Event Date Activities Retrieve Single Day) 	
Postconditions:	<ol style="list-style-type: none"> 1. User can view the activity for the particular event 	
Normal Flow:	<ol style="list-style-type: none"> 1. User selects activity to view in more detail of an activity (1069_Activity Retrieve List) 2. User views the details of the activity, including the name of the activity, a description, if it is a public activity, the start and end times, any images, and the sublocation of the activity within the event location 	
Alternative Flows:	<ol style="list-style-type: none"> 2a. No dates <ol style="list-style-type: none"> 1. User is directed to use case 1013_Event Date(s) 	
Exceptions:		
Includes:	1013_Event Date(s), 1061_Event Date Activities Retrieve Single Day, 1062_Event Date Activities Retrieve All Event Dates, 1060_Event Date Activities Create, 1069_Activity Retrieve List	
Frequency of Use:		
Special Requirements:		
Assumptions:		
Notes and Issues:	<ol style="list-style-type: none"> 1. 	

3.2.3.2.5 Activity Retrieve List

Use Case ID:	1069	
Use Case Name:	Activity Retrieve List	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	01/25/2022	Last Revision Date:
Actors:	Event Planner(Primary), Event Manager, Event Organizational Rep	
Description:	As an Event Planner, Event Manager, or Event Organizational Rep, I would like to view a list of all the activities for an event so that to sort and filter to help in searching	
Trigger:	View Activity Details	
Preconditions:	<ol style="list-style-type: none"> 1. Event must be created 2. The event dates must be chosen (1013_Event Date(s)) 3. Activity has been added to an Event Date(1060_Event Date Activities Create) 4. User is viewing a list of activities (1062_Event Date Activities Retrieve All Event Dates, 1061_Event Date Activities Retrieve Single Day) 	
Postconditions:	<ol style="list-style-type: none"> 1. User can view the activity for the particular event 	
Normal Flow:	<ol style="list-style-type: none"> 1. User selects event to view in more detail 2. User has a list of activities that are going to happen at the event in chronological order. User has option to filter it by date, name, and the sublocation 	
Alternative Flows:	<ol style="list-style-type: none"> 2a. No dates <ol style="list-style-type: none"> 1. User is directed to use case 1013_Event Date(s) 	
Exceptions:		
Includes:	1013_Event Date(s), 1061_Event Date Activities Retrieve Single Day, 1062_Event Date Activities Retrieve All Event Dates, 1060_Event Date Activities Create	
Frequency of Use:		
Special Requirements:		
Assumptions:		
Notes and Issues:	1.	

3.2.3.2.6 Activity Retrieve Related

Use Case ID:	1070		
Use Case Name:	Activity Retrieve Related		
Created By:	Derrick Nagy	Last Updated By:	
Date Created:	01/25/2022	Last Revision Date:	
Actors:	Event Planner(Primary), Event Manager, Event Organizational Rep		
Description:	As an Event Planner, Event Manager, Event Organizational Rep, I would like to view a list of all the activities for all events so that to sort and filter to help in searching		
Trigger:	View Activity Details		
Preconditions:	<ol style="list-style-type: none"> 1. Event must be created 2. The event dates must be chosen (1013_Event Date(s)) 3. Activity has been added to an Event Date(1060_Event Date Activities Create) 4. User is viewing a list of activities (1062_Event Date Activities Retrieve All Event Dates, 1061_Event Date Activities Retrieve Single Day) 		
Postconditions:	1. User can view the activity for the particular event		
Normal Flow:	<ol style="list-style-type: none"> 1. User chooses to view all activities at all events 2. User has a list of activities that are going to happen at the event in chronological order. User has option to filter it by date, name, and the sublocation 		
Alternative Flows:			
Exceptions:			
Includes:	1013_Event Date(s), 1061_Event Date Activities Retrieve Single Day, 1062_Event Date Activities Retrieve All Event Dates, 1060_Event Date Activities Create		
Frequency of Use:			
Special Requirements:			
Assumptions:			
Notes and Issues:	1.		

3.2.3.2.7 Activity Update

Use Case ID:	1071	
Use Case Name:	Activity Details	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	01/25/2022	Last Revision Date:
Actors:	Event Planner(Primary), Event Manager, Event Organizational Rep	
Description:	As an Event Planner, Event Manager, or Event Organizational Rep, I would like to update an activity so that it can stay up to date	
Trigger:	View Activity Details	
Preconditions:	<ol style="list-style-type: none"> 1. Event must be created 2. The event dates must be chosen (1013_Event Date(s)) 3. Activity has been added to an Event Date(1060_Event Date Activities Create) 4. User is viewing a list of activities (1062_Event Date Activities Retrieve All Event Dates, 1061_Event Date Activities Retrieve Single Day) 5. User has permissions to update 	
Postconditions:	<ol style="list-style-type: none"> 1. User can view the activity for the particular event 	
Normal Flow:	<ol style="list-style-type: none"> 1. User selects activity to view in more detail 2. User views the details of the activity, including the name of the activity, a description, if it is a public activity, the start and end times, any images, and the sublocation of the activity within the event location (1068_Activity Details) 3. User chooses to edit the activity 4. User can edit the details of the activity, including the name of the activity, a description, if it is a public activity, the start and end times, any images, and the sublocation of the activity 5. User submits changes 6. System prompts user to confirm 	
Alternative Flows: [Edit from Event Date]	<ol style="list-style-type: none"> 1. User is viewing the Event Date Activities and wants to edit an activity (1060_Event Date Activities Create) 2. User chooses to edit the activity 3. System retrieves details of event to be edited 4. User can edit the details of the activity, including the name of the activity, a description, if it is a public activity, the start and end times, any images, and the sublocation of the activity 5. User submits changes 6. System prompts user to confirm 	
Exceptions:		
Includes:	1013_Event Date(s), 1061_Event Date Activities Retrieve Single Day, 1062_Event Date Activities Retrieve All Event Dates, 1060_Event Date Activities Create, 1068_Activity Details	
Frequency of Use:		
Special Requirements:		
Assumptions:		
Notes and Issues:	1.	

3.2.3.2.8 Event Date Activities Create

Use Case ID:	1060	
Use Case Name:	Event Date Activities Create	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	01/25/2022	Last Revision Date:
Actors:	Event Planner(Primary), Event Manager, Event Organizational Rep	
Description:	As an Event Planner, Event Manager, or Event Organizational Rep, I would like to select an event date and be able to add activities to the date so that the event activities can be planned.	
Trigger:	Add Event Date Activity	
Preconditions:	<ol style="list-style-type: none"> 1. Event must be created 2. The event dates must be chosen (1013_Event Date(s)) 	
Postconditions:	<ol style="list-style-type: none"> 1. User can view the activities for that particular day of the event and choose to add another 	
Normal Flow:	<ol style="list-style-type: none"> 1. User chooses to add an activity to the event 2. System checks to see if the event has dates planned 3. System shows user a list with the event dates and the activity titles planned for that day (1062_Event Date Activities Retrieve All Event Dates) if any 4. User selects a date to add events to 5. System pulls up list of activities already planned for that day, if any (1061_Event Date Activities Retrieve Single Day) 6. User can add an activity to the day, specifying the name of the activity, a description, if it is a public activity, the start and end times, any images, and the sublocation of the activity within the event location 7. User confirms addition 8. System updates list of activities for the day and lets the user add another activity 	
Alternative Flows: [Alternative Flow 1 – No Dates]	2a. No dates <ol style="list-style-type: none"> 1. User is directed to use case 1013_Event Date(s) 	
Exceptions:		
Includes:	1013_Event Date(s), 1061_Event Date Activities Retrieve Single Day, 1062_Event Date Activities Retrieve All Event Dates	
Frequency of Use:	Anytime a user wants to add an activity	
Special Requirements:		
Assumptions:		
Notes and Issues:	1.	

3.2.3.2.9 Event Date Activities Deactivate

Use Case ID:	1065	
Use Case Name:	Event Date Activities Deactivate	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	01/25/2022	Last Revision Date:
Actors:	Event Planner(Primary), Event Manager, Event Organizational Rep	
Description:	As an Event Planner, I would like to deactivate the event date and all the activities on that date so that it can be canceled but not permanently deleted	
Trigger:	Deactivate event date	
Preconditions:	<ol style="list-style-type: none"> 1. Event must be created 2. The event dates must be chosen (1013_Event Date(s)) 	
Postconditions:	<ol style="list-style-type: none"> 1. User can view the activities for that particular day of the event 	
Normal Flow:	<ol style="list-style-type: none"> 1. User chooses to view the activities for a day 2. System checks to see if the event has dates planned 3. System pulls up list of activities already planned for that day, if any (1061_Event Date Activities Retrieve Single Day) 4. User selects a date to view 5. System pulls up list of activities already planned for that day, if any 6. User selects deactivate date 7. User confirms choice 8. User views event dates and activities (1062_Event Date Activities Retrieve All Event Dates) 	
Alternative Flows: [Alternative Flow 1 – No Dates]	2a. No dates <ol style="list-style-type: none"> 1. User is directed to use case 1013_Event Date(s) if they have permission to add event dates 	
Exceptions:		
Includes:	1013_Event Date(s), 1061_Event Date Activities Retrieve Single Day, 1062_Event Date Activities Retrieve All Event Dates	
Frequency of Use:		
Special Requirements:		
Assumptions:		
Notes and Issues:	<ol style="list-style-type: none"> 1. 	

3.2.3.2.10 Event Date Activities Delete

Use Case ID:	1066	
Use Case Name:	Event Date Activities Delete	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	01/25/2022	Last Revision Date:
Actors:	Event Planner(Primary), Event Manager, Event Organizational Rep	
Description:	As an Event Planner, Event Manager, Event Organizational Rep, I would like to delete the event date and all the activities on that date	
Trigger:	Delete event date	
Preconditions:	<ol style="list-style-type: none"> 1. Event must be created 2. The event dates must be chosen (1013_Event Date(s)) 	
Postconditions:	<ol style="list-style-type: none"> 1. User can view the activities for that particular day of the event 	
Normal Flow:	<ol style="list-style-type: none"> 1. User chooses to view the activities for a day 2. System checks to see if the event has dates planned 3. System pulls up list of activities already planned for that day, if any (1061_Event Date Activities Retrieve Single Day) 4. User selects a date to view 5. System pulls up list of activities already planned for that day, if any 6. User selects delete date 7. User confirms choice 8. User views event dates and activities (1062_Event Date Activities Retrieve All Event Dates) 	
Alternative Flows: [Alternative Flow 1 – No Dates]	2a. No dates <ol style="list-style-type: none"> 1. User is directed to use case 1013_Event Date(s) if they have permission to add event dates 	
Exceptions:		
Includes:	1013_Event Date(s), 1061_Event Date Activities Retrieve Single Day, 1062_Event Date Activities Retrieve All Event Dates	
Frequency of Use:		
Special Requirements:		
Assumptions:		
Notes and Issues:	<ol style="list-style-type: none"> 1. 	

3.2.3.2.11 Event Date Activities Retrieve All Event Dates

Use Case ID:	1062	
Use Case Name:	Event Date Activities Retrieve All Event Dates	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	01/25/2022	Last Revision Date:
Actors:	Event Planner(Primary), Event Manager, Event Organizational Rep, Attendee	
Description:	As an Event Planner, Event Manager, Event Organizational Rep, or Attendee, I would like to view all the dates for an event and the activities for them on that date so that the event activities can be viewed.	
Trigger:	View event date activities	
Preconditions:	<ol style="list-style-type: none"> 1. Event must be created 2. User must select an event and click the event activities button 3. 	
Postconditions:	<ol style="list-style-type: none"> 1. User can view the titles of the activities for all days of the event 	
Normal Flow:	<ol style="list-style-type: none"> 1. The user selects an event from the list 2. The user selects the event activities button 3. System checks to see if the event has dates planned 4. System shows user a list with the event dates and the activity titles planned for that day 5. User views the event dates and activity titles 	
Alternative Flows:		
Exceptions:	<ol style="list-style-type: none"> 2a. No activity for the selected date <ol style="list-style-type: none"> 1. The user is prompted that there is no date for the activities for the event 2. User is shown an empty list 2a. The date is formatted wrong or enters invalid data <ol style="list-style-type: none"> 1. The user is prompted that they entered invalid data and to check the formatted date. 	
Includes:	1013_Event Date(s)	
Frequency of Use:		
Special Requirements:		
Assumptions:		
Notes and Issues:	1.	

3.2.3.2.12 Event Date Activities Retrieve Single Day

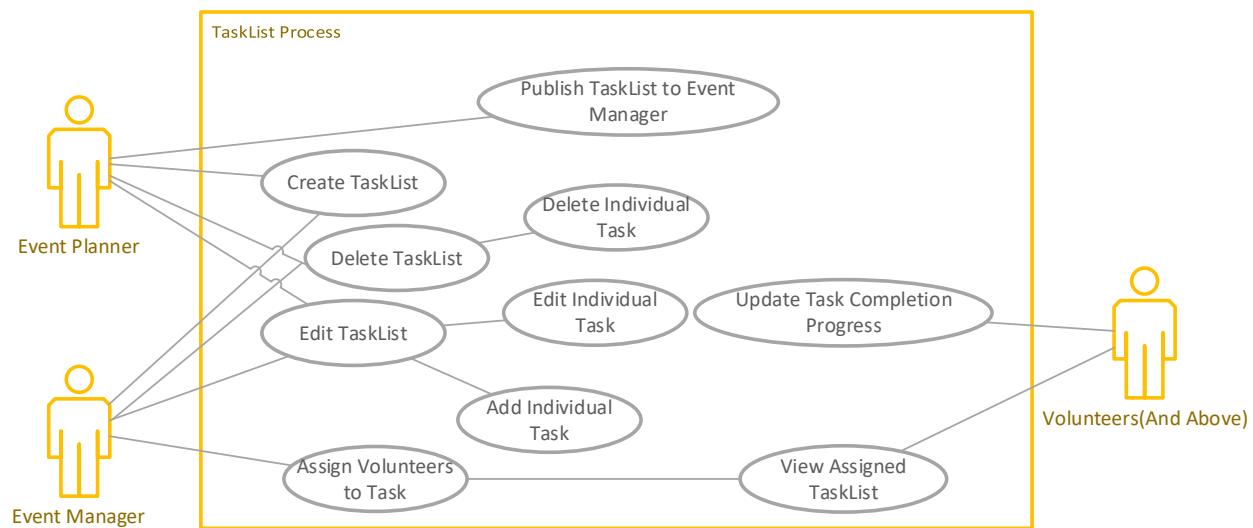
Use Case ID:	1061	
Use Case Name:	Event Date Activities Retrieve Single Day	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	01/25/2022	Last Revision Date:
Actors:	Event Planner(Primary), Event Manager, Event Organizational Rep	
Description:	As an Event Planner, Event Manager, or Event Organizational Rep, I would like to view all the activities for a date for an event so that the event activities can be viewed	
Trigger:	View event date activities	
Preconditions:	<ol style="list-style-type: none"> 1. Event must be created 2. The event dates must be chosen (1013_Event Date(s)) 	
Postconditions:	<ol style="list-style-type: none"> 1. User can view the activities for that particular day of the event 	
Normal Flow:	<ol style="list-style-type: none"> 1. User chooses to view an events activities 2. User selects search by single day from the filter 3. System checks to see if the event has dates planned 4. System pulls up list of activities already planned for that day, if any (1061_Event Date Activities Retrieve Single Day) 5. User selects a date to view 6. System pulls up list of activities already planned for that day, if any 	
Alternative Flows: [Alternative Flow 1 – No Dates] [Alternative Flow 2 – View activities while adding new]	<p>2a. No dates</p> <ol style="list-style-type: none"> 1. User is directed to use case 1013_Event Date(s) if they have permission to add event dates 1. User is creating new activities for a day of the event 2. User references activities already planned while planning new ones 	
Exceptions:		
Includes:	1013_Event Date(s), 1017_Event Calendar View	
Frequency of Use:		
Special Requirements:		
Assumptions:		
Notes and Issues:	<ol style="list-style-type: none"> 1. 	

3.2.3.2.13 Event Date Activities Update

Use Case ID:	1064	
Use Case Name:	Event Date Activities Update	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	01/25/2022	Last Revision Date:
Actors:	Event Planner(Primary), Event Manager, Event Organizational Rep	
Description:	As an Event Planner, Event Manager, Event Organizational Rep, I would like to update the information about the event date so that the event can stay current	
Trigger:	Edit event date	
Preconditions:	<ol style="list-style-type: none"> 1. Event must be created 2. The event dates must be chosen (1013_Event Date(s)) 	
Postconditions:	<ol style="list-style-type: none"> 1. User can view the activities for that particular day of the event 	
Normal Flow:	<ol style="list-style-type: none"> 1. User chooses to view the activities for a day 2. System checks to see if the event has dates planned 3. System pulls up list of activities already planned for that day, if any (1061_Event Date Activities Retrieve Single Day) 4. User selects a date to view 5. System pulls up list of activities already planned for that day, if any 6. User selects edit date and edits the date 7. System gives user option to choose which date to pick 8. User chooses new date 9. User confirms choice 10. System updates Event Dates 11. User views event date and activities 	
Alternative Flows: [Alternative Flow 1 – No Dates]	<p>2a. No dates</p> <ol style="list-style-type: none"> 1. User is directed to use case 1013_Event Date(s) if they have permission to add event dates 	
Exceptions:		
Includes:	1013_Event Date(s), 1061_Event Date Activities Retrieve Single Day	
Frequency of Use:		
Special Requirements:		
Assumptions:		
Notes and Issues:	<ol style="list-style-type: none"> 1. 	

3.2.4 Tasks

3.2.4.1 Use Case Model



3.2.4.2 Use Cases

3.2.4.2.1 Task List

Use Case ID:	1018		
Use Case Name:	Task List		
Created By:	Asa Armstrong	Last Updated By:	Asa Armstrong
Date Created:	9/14/21	Last Revision Date:	9/23/21
Actors:	Event Planner (primary), Event Manager (secondary)		
Description:	Create a task list for what needs to be done		
Trigger:	An event was created, but not finalized		
Preconditions:	1. An event was created		
Postconditions:	1. User has a list of items they need to do to finalize the event 2. The organizational reps progress report is updated with any changes		
Normal Flow:	1. User creates an event 2. To do list pre-populates with any fields left blank while creating the event		
Alternative Flows: [Alternative Flow 1 – User Creates Additional Task List]	2a. In step 2 of the normal flow, if the user chooses to add another task list 1. User chooses option to create task list 2. A blank task list is generated 3. User is directed to the blank task list 4. Task list can be shared to other users for editing		
Exceptions:			
Includes:	1050_Task List View		
Frequency of Use:	Every time an event is created. Whenever another task list needs to be added.		
Special Requirements:			
Assumptions:			
Notes and Issues:	1. Task Lists can also be created for the Event Manager for what needs to be taken care of during the event or any other tasks they are in charge of.		

3.2.4.2.2 Task List Create

Use Case ID:	1023		
Use Case Name:	Task List Create		
Created By:	Mike Cahow	Last Updated By:	Mike Cahow
Date Created:	9/11/2021	Last Revision Date:	9/24/2021
Actors:	Event Planner		
Description:	An Event Planner will create this task list for the Event Manager to follow the day of an event and update as completed/incomplete/in progress		
Trigger:	To be able to create a Task List, an event must be scheduled so an Event Planner can select the create task list button		
Preconditions:	<ol style="list-style-type: none"> 1. User must be signed in as Event Planner 2. Event Planner must Create an event 		
Postconditions:	<ol style="list-style-type: none"> 1. Clicking Submit will send Task List to everyone working the day of event 2. Notified when Task List is completed 		
Normal Flow:	<ol style="list-style-type: none"> 1. Event Planner creates Task List for an Event 2. Planner writes tasks to complete for the day of event 3. System asks what the priority (High to Low) of the task is 4. Planner confirms priority of task 5. Planner finishes listing tasks and selecting priority 6. System asks Planner if they are finished editing list 7. Planner selects yes. 8. Planner specifies who list is going (Event Manager/Volunteers involved) 9. Planner selects Send Tasks 10. System sends List to intended recipients 11. Planner able to view completion or updates made by Event Manager/Volunteers 12. Planner closes out of list 		
Alternative Flows: [Alternative Flow 1 – Close and Save]	<p>5a. In step 5 of the normal flow, if the Event Planner isn't finished creating tasks</p> <ol style="list-style-type: none"> 1. System will prompt Planner to save list thus far 2. Planner accepts 3. System saves list 4. Use Case resumes on step 13 of normal flow 		
Exceptions:			

Includes:	
Frequency of Use:	Use Case will be executed at least once per event creation. The feature will be viewed frequently (depending on size of list created) the day of the event possibly averaging 20 -30 times.
Special Requirements:	
Assumptions:	An event is created and needs a task list Event Managers will be filling out the Task List
Notes and Issues:	1. What is the maximum/minimum length of list that a use can have?

3.2.4.2.3 Task List Edit

Use Case ID:	1051		
Use Case Name:	Task List Edit		
Created By:	Asa Armstrong	Last Updated By:	Asa Armstrong
Date Created:	9/14/2021	Last Revision Date:	9/23/2021
Actors:	Event Planner (primary), Event Manager (secondary)		
Description:	Add or edit the tasks in a task list		
Trigger:	User chooses a task to edit or selects to add a task		
Preconditions:	<ol style="list-style-type: none"> 1. A task list exists 2. The user is viewing the task list 		
Postconditions:	<ol style="list-style-type: none"> 1. The task list is updated with any changes and tasks are resorted accordingly 		
Normal Flow:	<ol style="list-style-type: none"> 1. User selects a task or selects to add a task 2. A form for the task generates and prepopulates with any current info 3. User sets the description and due date for the task 4. User adds relational tags for the task (like if the task relates to Vendors, Venue, Volunteers, etc) for sorting purposes 5. User saves the task 		
Alternative Flows: [Alternative Flow 1 – Task Save Cancelled/Deleted]	5a. In step 5 of the normal flow, if the user cancels or deletes instead of saves the task <ol style="list-style-type: none"> 1. The task will be removed if deleted or won't be updated if changes are cancelled 2. User prompted to confirm their choice 		
[Alternative Flow 2 – Save or cancel before close]	5a. In step 5 of the normal flow, if the user leaves without saving the task <ol style="list-style-type: none"> 1. User prompted to save or cancel the task 		
Exceptions:			
Includes:	1018_Task List 1050_Task List View		
Frequency of Use:			
Special Requirements:			
Assumptions:			
Notes and Issues:			

3.2.4.2.4 Task List View

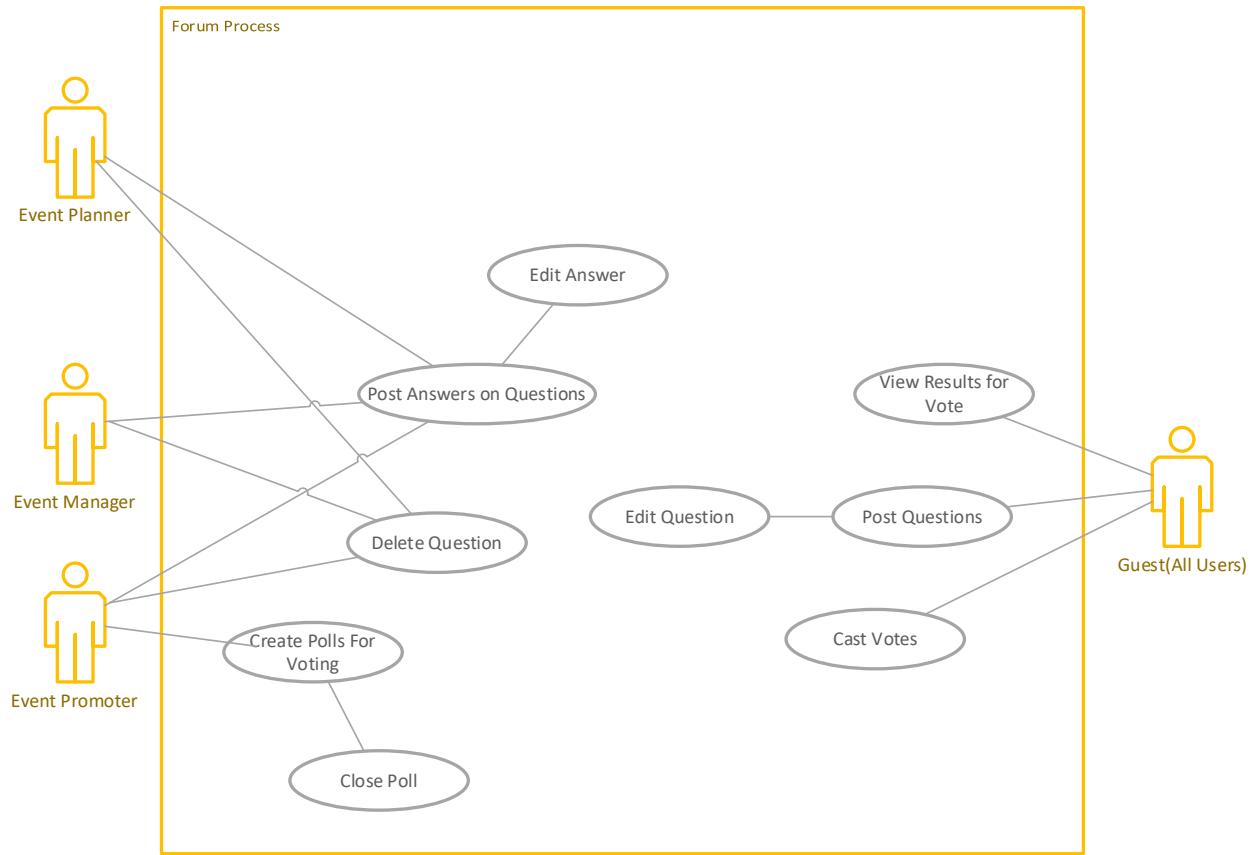
Use Case ID:	1050		
Use Case Name:	Task List View		
Created By:	Asa Armstrong	Last Updated By:	Asa Armstrong
Date Created:	9/14/21	Last Revision Date:	9/23/21
Actors:	Event Planner (primary), Event Manager (secondary)		
Description:	View the task list so that tasks can be toggled or read		
Trigger:	User selects their task list		
Preconditions:	<ol style="list-style-type: none">1. An event was created		
Postconditions:	<ol style="list-style-type: none">1. The organizational reps progress report is updated with any changes2. Any task marked completed now shows in the completed tasks section and have a completed date added		
Normal Flow:	<ol style="list-style-type: none">1. User selects their task list to view2. The task list is sorted by date/time if available and completed tasks are shown separated from the uncompleted tasks3. Tasks can be toggled completed/uncompleted4. User returns from task list		
Alternative Flows:	<ol style="list-style-type: none">2a. In step 2 of the normal flow, if the user selects to sort the tasks differently<ol style="list-style-type: none">1. User chooses to sort by relational tag, date/time ascending/descending2. Task list sorts accordingly		
Exceptions:			
Includes:	1018_Task List		
Frequency of Use:			
Special Requirements:			
Assumptions:			
Notes and Issues:			

Task List Delete

Use Case ID:	1050
Use Case Name:	Task List Delete
Created By:	Mike Cahow
Date Created:	3/25/22
Actors:	Event Planner (primary), Event Manager (secondary)
Description:	Delete a selected task
Trigger:	User selects the delete button on the task Edit page
Preconditions:	2. An event was created 3. Task was created 4. User is a role that can edit/delete
Postconditions:	3. Task is permanently deleted from the database 4. Task list is updated.
Normal Flow:	5. User selects their task to view/edit 6. The task details are brought up 7. User selects the delete button 8. System checks if deleting the task is what user wants to do 9. User clicks yes 10. Task is deleted from database 11. Task list is updated 12. User is returned to the task list View
Alternative Flows:	2a. In step 9 of the normal flow, if the user selects no to the delete prompt 3. System closes the prompt 4. User can edit the task or click cancel
Exceptions:	
Includes:	1018_Task List, Task List View, Task List Edit
Frequency of Use:	
Special Requirements:	
Assumptions:	User is either Event Planner or Manager
Notes and Issues:	

3.2.5 Forum

3.2.5.1 Use Case Model



3.2.5.2 Use Cases

3.2.5.2.1 Forum

Use Case ID:	1008		
Use Case Name:	Forum		
Created By:	Asa Armstrong	Last Updated By:	Asa Armstrong
Date Created:	9/23/2021	Last Revision Date:	9/23/2021
Actors:	Event Planner (primary), Event Manager (secondary)		
Description:	A forum/discussion board for users to communicate over		
Trigger:	User adds a forum to their event		
Preconditions:	<ol style="list-style-type: none">1. An event is created		
Postconditions:	<ol style="list-style-type: none">1. A forum is visible for user to interact with		
Normal Flow:	<ol style="list-style-type: none">1. User chooses the add forum option for their event2. User enters name and description of the forum3. User chooses what kind of attendees/participants can interact with the forum (example: RSVP'd attendees, volunteers, anyone with an account)		
Alternative Flows: [Alternative Flow 1 – Forum Cancel]	<ol style="list-style-type: none">1a. In step 1 of the normal flow, if the user chooses not to have a forum for their event<ol style="list-style-type: none">1. No forum will be created		
Exceptions:			
Includes:			
Frequency of Use:	Whenever the user decides to add a forum to their event		
Special Requirements:			
Assumptions:			
Notes and Issues:	<ol style="list-style-type: none">1.		

3.2.5.2.2 Forum Comment Create

Use Case ID:	1058		
Use Case Name:	Forum Comment Create		
Created By:	Asa Armstrong	Last Updated By:	Asa Armstrong
Date Created:	10/19/2021	Last Revision Date:	10/19/2021
Actors:	Event Planner, Forum Participants(attendees, volunteers)		
Description:	Comment on a post in the forum		
Trigger:	User clicks the comment button		
Preconditions:	<ol style="list-style-type: none"> 1. A forum exists with posts in it 2. The user has access to the forum 		
Postconditions:	<ol style="list-style-type: none"> 1. The comment can be displayed under the post it is commented on 2. The comment can be liked 		
Normal Flow:	<ol style="list-style-type: none"> 1. User goes to the forum 2. User goes to the post 3. User clicks the comment button 4. A comment text box appears along with their username and picture 5. User types their comment in the textbox 6. User clicks Post 7. Comment is connected to the post 		
Alternative Flows:			
Exceptions:	6a. In step 6 of the normal flow, if the user leaves without posting their comment <ol style="list-style-type: none"> 1. The comment will not be saved 		
Includes:	1008_Forum 1048_Forum Create Post		
Frequency of Use:			
Special Requirements:			
Assumptions:	User has access to the forum		
Notes and Issues:	1.		

3.2.5.2.3 Forum Create Post

Use Case ID:	1048		
Use Case Name:	Forum Create Post		
Created By:	Asa Armstrong	Last Updated By:	Asa Armstrong
Date Created:	9/23/2021	Last Revision Date:	9/23/2021
Actors:	Event Planner (primary), Event Manager (secondary)		
Description:	Create a post for the event's forum		
Trigger:	User selects to create a post		
Preconditions:	<ol style="list-style-type: none"> 1. A forum is added to the event 		
Postconditions:	<ol style="list-style-type: none"> 1. The post is viewable to participants 		
Normal Flow:	<ol style="list-style-type: none"> 1. User views the forum 2. User selects the create a post option 3. User adds details to the post (description, media, voting system) 4. User submits the post to the forum 		
Alternative Flows: [Alternative Flow 1 – Post Not Submitted]	<ol style="list-style-type: none"> 4a. In step 4 of the normal flow, if the user cancels the post <ol style="list-style-type: none"> 1. The post will not be added to the forum 		
Exceptions:	<ol style="list-style-type: none"> 4a. In step 4 of the normal flow, if the user leaves without submitting the post <ol style="list-style-type: none"> 1. User is prompted to submit or cancel the post 		
Includes:	1008_Forum		
Frequency of Use:			
Special Requirements:			
Assumptions:	An event exists with a forum		
Notes and Issues:	<ol style="list-style-type: none"> 1. Maybe save unsubmitted posts as drafts 2. In the UI, the option to post media and voting was not implemented 		

3.2.5.2.4 Forum Delete Post

Use Case ID:	1049		
Use Case Name:	Forum Delete Post		
Created By:	Asa Armstrong	Last Updated By:	Asa Armstrong
Date Created:	9/23/2021	Last Revision Date:	9/23/2021
Actors:	Event Planner (primary), Event Manager (secondary)		
Description:	Delete a post from the forum		
Trigger:	User chooses a post and selects to delete it		
Preconditions:	<ol style="list-style-type: none"> 1. A forum exists for an event 2. A post is on the forum 		
Postconditions:	<ol style="list-style-type: none"> 1. The post is no longer on the forum 		
Normal Flow:	<ol style="list-style-type: none"> 1. User views the forum 2. User selects a post 3. User selects to delete the post 4. User prompted to confirm deletion 5. Post removed from forum 		
Alternative Flows: [Alternative Flow 1 – Not Deleted]	4a. In step 4 of the normal flow, if the user doesn't confirm the delete <ol style="list-style-type: none"> 1. The post is not removed from the forum 		
Exceptions:			
Includes:			
Frequency of Use:			
Special Requirements:			
Assumptions:	The user has editing capabilities to the forum		
Notes and Issues:	<ol style="list-style-type: none"> 1. 		

3.2.5.2.5 Forum Edit

Use Case ID:	1059		
Use Case Name:	Forum Edit		
Created By:	Asa Armstrong	Last Updated By:	Asa Armstrong
Date Created:	10/19/2021	Last Revision Date:	10/19/2021
Actors:	Event Planner, Event Manager, Event Promoter		
Description:	Edit the details of a forum to change the name, description, and participants		
Trigger:	The edit button is clicked on the forum page		
Preconditions:	<ol style="list-style-type: none"> 1. A forum exists 		
Postconditions:	<ol style="list-style-type: none"> 1. The forum is updated with any saved changes 		
Normal Flow:	<ol style="list-style-type: none"> 1. User views the forum 2. User clicks the edit option 3. User is taken to the edit forum page 4. User can change the name, description, and participants 5. User saves changes 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	4a. In step 5 of the normal flow, if the user cancels the changes <ol style="list-style-type: none"> 1. No changes are made to the forum 		
Exceptions:			
Includes:	1008_Forum		
Frequency of Use:			
Special Requirements:			
Assumptions:	The user has a role in the event capable of editing a forum		
Notes and Issues:	<ol style="list-style-type: none"> 1. 		

3.2.5.2.6 Voting

Use Case ID:	1016		
Use Case Name:	Voting		
Created By:	Asa Armstrong	Last Updated By:	Asa Armstrong
Date Created:	9/14/2021	Last Revision Date:	9/23/2021
Actors:	Event Planner, Event Manager, Event Promoter		
Description:	To have a polling system for other users to vote on various topics (ie. what music/food/beverages they would like to see at the event).		
Trigger:	User goes to the forum for the event and chooses to add a post with a poll for certain users to see (volunteers/attendees or other sub groups).		
Preconditions:	<ol style="list-style-type: none"> 1. Event has been created. 2. There is a forum page for the event. 		
Postconditions:	<ol style="list-style-type: none"> 1. A poll will exist on the forum for the selected users to interact with. 2. The poll will expire at a designated date/time. 3. The poll results will be displayed. 4. The creator of the poll will be alerted. 		
Normal Flow:	<ol style="list-style-type: none"> 1. User chooses the create a poll option. 2. User adds different options that can be voted for. 3. User determines if the poll will be a choose-one-option voting or ranking the options in order of best to worst. 4. User selects who they want to share the poll to. 5. User sets the expiration date of the poll. 6. User submits the poll to the forum. 7. Other users interact with the poll. 8. The expiration date deactivates the poll. 9. Results are displayed. 10. The user that created the poll is alerted. 		
Alternative Flows:			
[Alternative Flow 1 – Post Cancelled]	6a. In step 6 of the normal flow, if the user cancels the poll <ol style="list-style-type: none"> 1. No post will be added to the forum 		
[Alternative Flow 2 - Save on Close]	6a. In step 6 of the normal flow, if the user leaves before submitting the post <ol style="list-style-type: none"> 1. User is prompted to submit or cancel the post 		
Exceptions:	6a. In step 6 of the normal flow, if the user leaves before submitting the post <ol style="list-style-type: none"> 1. User is prompted to submit or cancel the post 		
Includes:	1048_Forum Create Post		
Frequency of Use:			
Special Requirements:			
Assumptions:			
Notes and Issues:	1.		

3.2.5.2.7 Voting Cast Vote

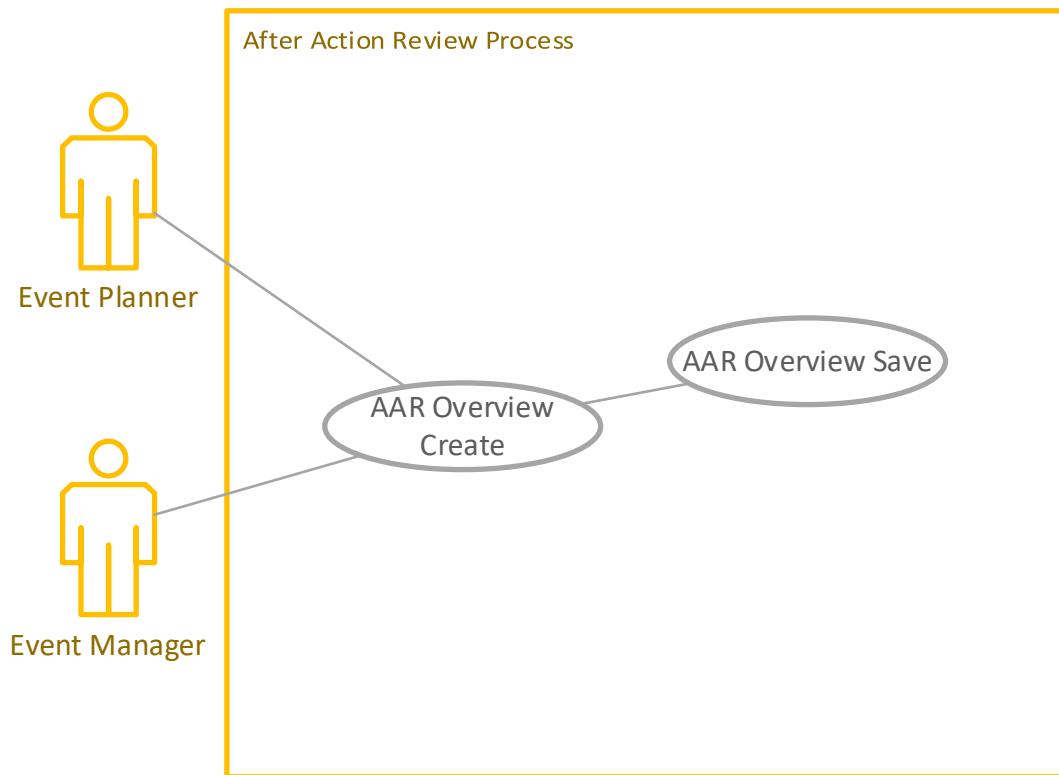
Use Case ID:	1056		
Use Case Name:	Voting Cast Vote		
Created By:	Asa Armstrong	Last Updated By:	Asa Armstrong
Date Created:	10/19/2021	Last Revision Date:	10/19/2021
Actors:	Event Planner, Forum Participants(attendees, volunteers)		
Description:	Cast a vote on a poll in the forum		
Trigger:			
Preconditions:	<ol style="list-style-type: none"> 1. There is a forum with a voting poll post in it. 2. The voting poll is not expired 		
Postconditions:	<ol style="list-style-type: none"> 1. The vote is tallied 2. The current results are displayed 		
Normal Flow:	<ol style="list-style-type: none"> 1. User goes to the forum 2. User goes to a post with an unexpired voting poll 3. User selects the item they want to vote for if it's a single vote type of poll, or the user ranks the options by entering a number next to the option (from 1(best) to n(worst)) 4. User clicks Cast Vote 5. Current voting results are displayed 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	4a. In step 4 of the normal flow, if the user does not click cast vote <ol style="list-style-type: none"> 1. No changes will be made and their vote not tallied 		
Exceptions:			
Includes:	1016_Voting 1057_Voting Results View 1008_Forum		
Frequency of Use:			
Special Requirements:			
Assumptions:	User has access to the forum		
Notes and Issues:	<ol style="list-style-type: none"> 1. 		

3.2.5.2.8 Voting Results View

Use Case ID:	1057		
Use Case Name:	Voting Results View		
Created By:	Asa Armstrong	Last Updated By:	Asa Armstrong
Date Created:	10/19/2021	Last Revision Date:	10/19/2021
Actors:	Event Planner, Event Promoter, Forum Participants (attendees, volunteers)		
Description:	See the results of a poll I voted on or a poll that has expired in the forum		
Trigger:	A poll has been voted on or expired		
Preconditions:	<ol style="list-style-type: none">1. There is a poll that exists in a forum2. The poll has expired or been voted on by the user viewing it		
Postconditions:	<ol style="list-style-type: none">1. View the results of the poll to see how many people voted on each option		
Normal Flow:	<ol style="list-style-type: none">1. Go to the forum2. View a poll that has expired or been voted on3. See details like: how many people voted on the poll, the different options, and the percentage of votes the options got		
Alternative Flows:			
Exceptions:	<p>6a. In step 2 of the normal flow, if the poll expires without any votes</p> <ol style="list-style-type: none">1. Results show that no votes were cast		
Includes:	1056_Voting Cast Vote 1008_Forum		
Frequency of Use:	Whenever a poll is being displayed that has been voted on or expired		
Special Requirements:			
Assumptions:	User has access to the forum		
Notes and Issues:			

3.2.6 After Action Review

3.2.6.1 Use Case Model



3.2.6.2 Use Cases

3.2.6.2.1 AAR Overview Create

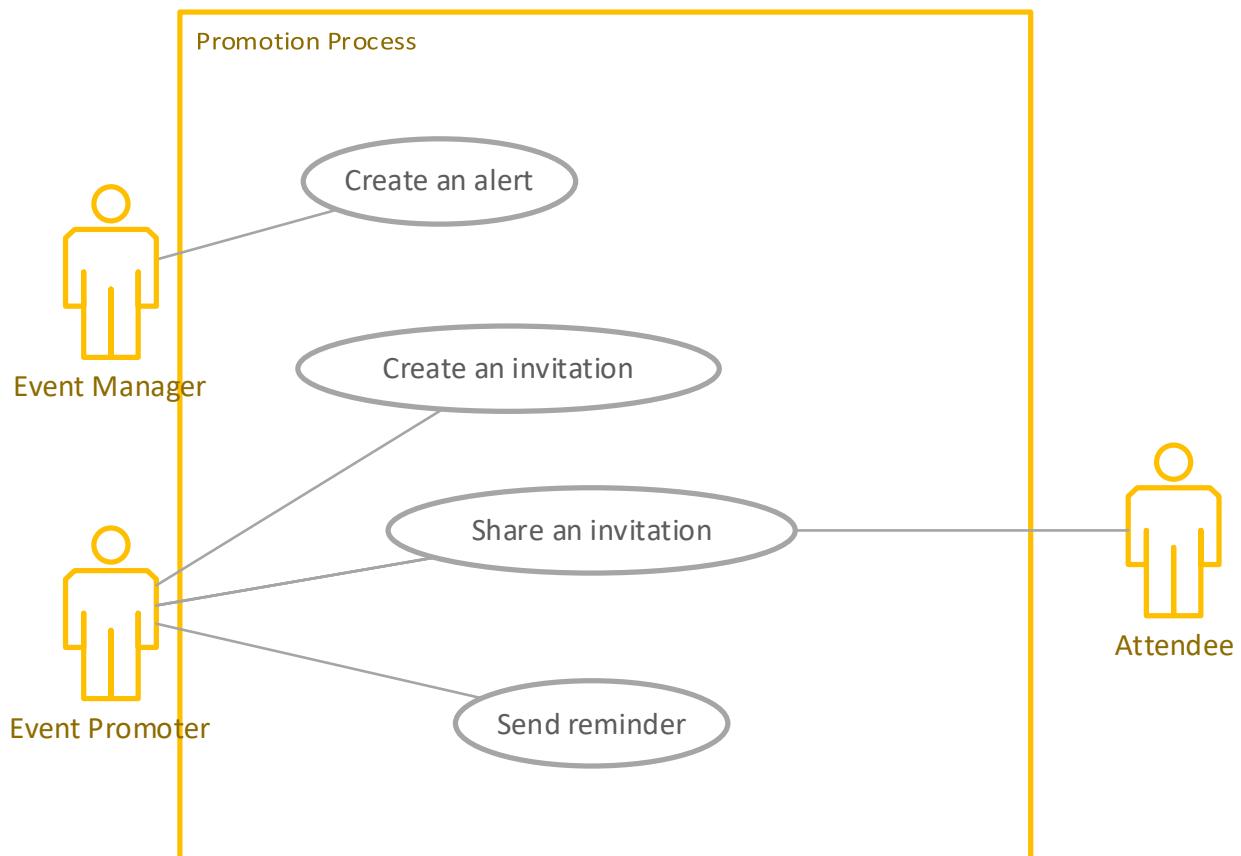
Use Case ID:	1002	
Use Case Name:	AAR Overview Create	
Created By:	Chase Fields	Last Updated By:
Date Created:	9/21/2021	Last Revision Date:
Actors:	Event Planner/Event Manager	
Description:	View the attendees, volunteers, and suppliers who were at the event as well as a count of their numbers and see the average of the ratings and the reviews.	
Trigger:	The planner clicks to get results on the parameters page.	
Preconditions:	<ol style="list-style-type: none">1. Event has passed2. User is an event planner	
Postconditions:	<ol style="list-style-type: none">1. A report (AAR results) is created and displayed to the user.	
Normal Flow:	<ol style="list-style-type: none">1. User Clicks After Event Report2. System checks to see if there are reviews and whether a submission date limit has passed.3. System gets event attendees, volunteers, and suppliers4. System gets reviews for the event5. System displays a list of the attendees, volunteers, and suppliers6. System provides an average star rating.7. System allows user to view all the specific ratings/feedback submissions.	
Alternative Flows:	<ol style="list-style-type: none">2a. There are no reviews or submitted<ol style="list-style-type: none">1. System alerts the user that there are currently no reviews.2. Prompts the user to cancel report generation or proceed.3. User proceeds4. System provides only number summary2b. There are reviews but a submission limit date has not arrived.<ol style="list-style-type: none">1. System alerts the user that there may be outstanding submissions. Asks to proceed or cancel the report.2. User proceeds.3. Step 3 of normal flow	
Exceptions:		
Includes:		
Frequency of Use:	Once per event.	
Special Requirements:	None	
Assumptions:	None	
Notes and Issues:		

3.2.6.2.2 AAR Overview Save

Use Case ID:	1039	
Use Case Name:	AAR Overview Save	
Created By:	Chase Fields	Last Updated By:
Date Created:	9/21/2021	Last Revision Date:
Actors:	Event Planner/Event Manager	
Description:	Download AAR report to user pc.	
Trigger:	The planner selects to save the AAR results	
Preconditions:	<ol style="list-style-type: none"> 1. User has proper role(s) 2. User is viewing Report Results 	
Postconditions:	The AAR overview is saved as a text file wherever the user chooses.	
Normal Flow:	<ol style="list-style-type: none"> 1. The user selects reviews for a downloaded report. 2. User can include all reviews or choose which ones 3. User chooses download report 4. System opens save file dialog to allow user to choose location 5. System downloads results as a text file displaying summary numbers, average review, and all reviews (with name, rating, comments) 6. System updates user of successful download. 	
Alternative Flows:	<ol style="list-style-type: none"> 2b. User selects to not include reviews in downloaded report <ol style="list-style-type: none"> 1. User downloads report 2. System downloads 4 of normal flow minus reviews 3. step 5 of normal flow. 	
Exceptions:	<ol style="list-style-type: none"> 2a. In step 2 of normal flow, there is a download error. <ol style="list-style-type: none"> 1. System alerts User of the error. 	
Includes:		
Frequency of Use:		
Special Requirements:	None	
Assumptions:	None	
Notes and Issues:		

3.2.7 Promotion

3.2.7.1 Use Case Model



3.2.7.2 Use Cases

3.2.7.2.1 Alerts Create

Use Case ID:	1003		
Use Case Name:	Alerts Create		
Created By:	Chase Fields	Last Updated By:	
Date Created:	9/20/2021	Last Revision Date:	
Actors:	Event Manager(EM)		
Description:	Create an alert to the volunteers / event planner.		
Trigger:	The EM creates a message to the team as an alert		
Preconditions:	<ol style="list-style-type: none">1. The User has an event planner or manager role.2. The event has been created and has user roles specified.3. There is an accessible list of messageable volunteers.		
Postconditions:	<ol style="list-style-type: none">1. The chosen users receive the alert.		
Normal Flow:	<ol style="list-style-type: none">1. 1022_Message Planning team flow executed.2. EM marks the message as a priority of type alert3. System designates a message with a special flag to indicate its alert status.4. EM sends a message.		
Alternative Flows:	None		
Exceptions:	Exceptions should be handled by 1022.		
Includes:	1022_ Message Planning Team		
Frequency of Use:	On Demand		
Special Requirements:	None		
Assumptions:	None		
Notes and Issues:	<ol style="list-style-type: none">1. Do other priority types need to be added to 1022?2. Is the styling of the message something that needs to be handled with a different use case (such as display alerts to volunteers)?		

3.2.7.2.2 *Invitations Create*

Use Case ID:	1027	
Use Case Name:	Invitations Create	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	09/21/2021	Last Revision Date:
Actors:	Event Promoter(primary), Event Organizational Rep(secondary)	
Description:	Create and send email invitations so that attendees to private events can RSVP and give Planner and Manager a proper headcount	
Trigger:	Create Event	
Preconditions:	<ol style="list-style-type: none"> 1. Event must be created 2. Event Promoter has been assigned to event 	
Postconditions:	<ol style="list-style-type: none"> 1. Minimal: Event Promoter can send an email giving what information that they would like to give about the event 2. Success: Event Promoter able to send email and message that invites recipient to the event, possibly a particular part of the event(come at this time for this activity, VIP event in the event), recipient is also able to sign up for our Event Planning Software 	
Normal Flow:	<ol style="list-style-type: none"> 1. Event Promoter wants to create an invitation, starts event 2. Event Promoter Selects an event from a list of events that they are authorized to be a part of 3. System pre-populates invitation card with any information about the event that is relevant, like location, date, time, title, description, how to RSVP, how to sign up for the app and RSVP 4. The Event Promoter chooses what information that they would like to add to or remove from the invitation. The Event Promoter can remove or edit any part of the invitation 5. The Event Promoter can select a picture. The picture is one of the pictures that has already been associated with the event (uploaded/added to the event files previously) or a new one can be uploaded. 6. The Event Promoter approves of the invitation 7. The Event Promoter is directed to how to share the invitation 	
Alternative Flows: [Alternative Flow 1 – No Event to choose from] [Alternative Flow 2 – Nothing to pre-populate]	2a. No Event to choose from <ol style="list-style-type: none"> 1. Event promoter is redirected to create an event 3a. There is no information about the event that can be added <ol style="list-style-type: none"> 1. Event Promoter then is able to type in any message description 2. Resumes at step 6 	
Exceptions:		
Includes:	RSVP event	
Frequency of Use:	Anytime an event has a guest list or would like to advertise the event to specific individuals that they already have an email for	
Special Requirements:		

Assumptions:	The Event Promoter knows enough about the event to share information. For instance, a title, brief description, and date seem like a necessity.
Notes and Issues:	1.

3.2.7.2.3 Invitations Share

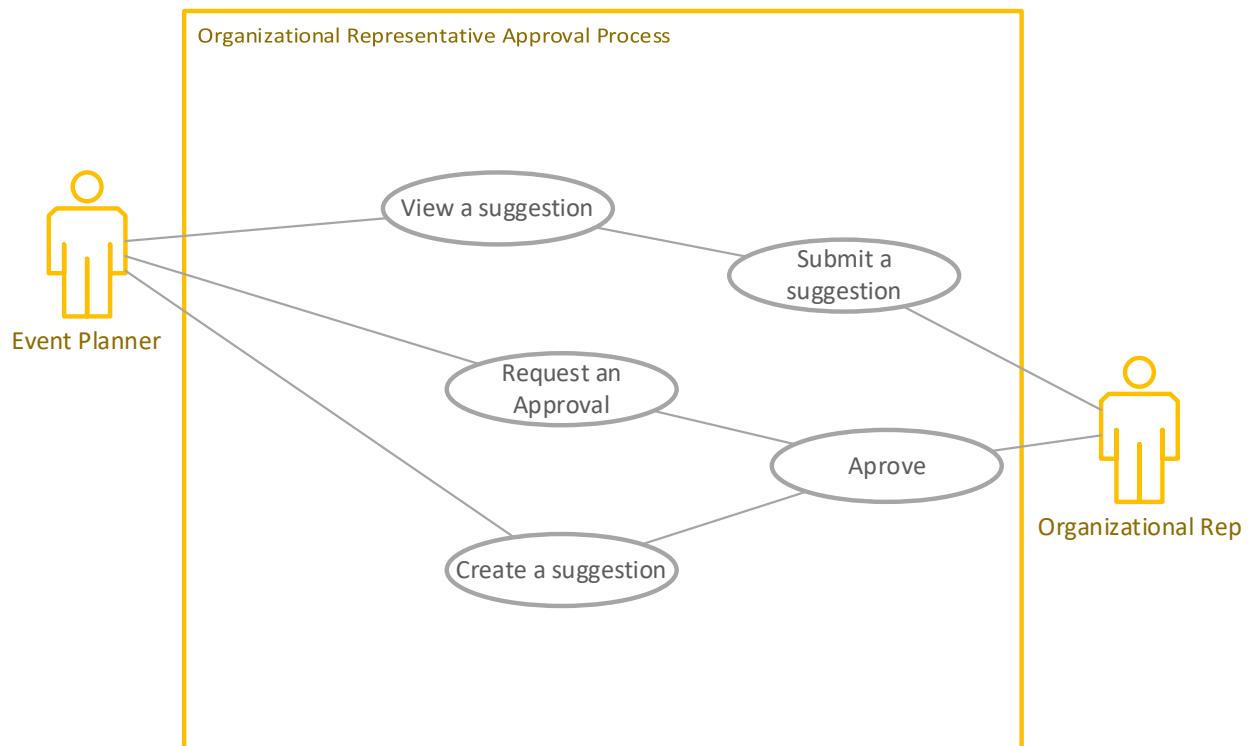
Use Case ID:	1028	
Use Case Name:	Invitations Share	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	09/21/2021	Last Revision Date:
Actors:	Event Promoter	
Description:	Decide who gets the invitation and how so that attendees can be invited to the event.	
Trigger:	Invitations Creation	
Preconditions:	<ol style="list-style-type: none"> 1. Event has been created 2. Invitation has been created 	
Postconditions:	<ol style="list-style-type: none"> 1. Minimum: Email has been sent to the correct potential attendees with RSVP link 	
Normal Flow:	<ol style="list-style-type: none"> 1. System notifies Event Promoter that they have an invitation that has not been sent 2. Event Promoter selects the invitation that they would like to send 3. Event Promoter can either type in email addresses, link from an address book like in Gmail or from their phone, search for attendees by name through app 4. Event Promoter approves of the list 5. System sends emails to the recipients 6. System gives confirmation or error message 	
Alternative Flows: [Bad Email Address]	6a. Error with sending the emails <ol style="list-style-type: none"> 1. System sends message to Event Promoter with information about why the email failed to send 2. Event Promoter goes back to step 3 in normal flow 	
Exceptions:	<ol style="list-style-type: none"> 1. The Event Promoter could have given a bad email address. Validation should be run on the email address before it is sent 2. Email does not exist, and no one got it. System notifies Event Promoter which emails were not sent 	
Includes:	Event Creation, Invitations Creation, RSVP	
Frequency of Use:	The batch of emails being sent has the potential to be large if it is a big event	
Special Requirements:		
Assumptions:		
Notes and Issues:		

3.2.7.2.4 Reminder Send

Use Case ID:	1019		
Use Case Name:	Reminder Send		
Created By:	Derrick Nagy	Last Updated By:	Derrick Nagy
Date Created:	9/13/2021	Last Revision Date:	10/13/2021
Actors:	Event Promoter		
Description:	The event promoter would like to send a reminder to share when an event starts.		
Trigger:	A user joins an event group		
Preconditions:	1. Event has been created		
Postconditions:	1. The user specified communication preference is executed		
Normal Flow:	Normal: 1. System sends reminders to users about due dates, times of events at an interval of time (a week ahead, a day, hour...) or after an System event 2. User can see the notification 3. User selects notification and is taken to the appropriate destination 4. System marks notification as read		
Alternative Flows: [Alternative Flow 1 –Create Reminder], [Alternative Flow 2- System Creates Reminder]	Alternative Flow 1 1. Event promoter creates a new notification 2. Event promoter adds what the message title and description is 3. Event promoter selects when the reminder should be sent 4. System sends the reminder at the user specified time Alternative Flow 2 1. A change to the system creates a notification event 2. A notification is created with the title of the event and a brief description of what happened		
Exceptions:			
Includes:	User creation, Event creation		
Frequency of Use:	During event planning leading up to start of the event		
Special Requirements:			
Assumptions:	The user understands that this is how they will be sent reminders and they won't get any unless they sign into the application.		
Notes and Issues:	1. What are going to be their options for communicating? Email, text, in app messages?		

3.2.8 Representative Approval

3.2.8.1 Use Case Model



3.2.8.2 Use Cases

3.2.8.2.1 Approval

Use Case ID:	1006		
Use Case Name:	Approval		
Created By:	Asa Armstrong	Last Updated By:	Asa Armstrong
Date Created:	9/14/2021	Last Revision Date:	9/23/2021
Actors:	Event Organizational Rep		
Description:	Approve or deny requests made by an Event Planner or Manager.		
Trigger:	Event Planner or Manager sends a request to the Org Rep		
Preconditions:	<ol style="list-style-type: none">1. There is an event planner/manager for the event		
Postconditions:	<ol style="list-style-type: none">1. The request is approved or denied2. Notification sent to the creator of the request with the results		
Normal Flow:	<ol style="list-style-type: none">1. User receives a request2. User views the request3. User chooses to accept or deny4. User adds description/details to their choice5. Results are sent back to the planner/manager		
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. In step 3 of the normal flow, if the user does not finish the approval request <ol style="list-style-type: none">1. The request is saved for completing later		
Exceptions:			
Includes:	1047_Approval Request		
Frequency of Use:	Whenever a request is sent to the org rep		
Special Requirements:			
Assumptions:	The org rep has standards set for the event that they want the planner/manager to follow		
Notes and Issues:	1.		

3.2.8.2.2 Approval Request

Use Case ID:	1047		
Use Case Name:	Approval Request		
Created By:	Asa Armstrong	Last Updated By:	Asa Armstrong
Date Created:	9/14/2021	Last Revision Date:	9/23/2021
Actors:	Event Planner (primary), Event Manager (secondary)		
Description:	Create a request for approval from the org rep		
Trigger:	User chooses to make a request to the org rep		
Preconditions:	<ol style="list-style-type: none"> 1. User is editing an event for an OR 		
Postconditions:	<ol style="list-style-type: none"> 1. The request is sent to the OR for approval 		
Normal Flow:	<ol style="list-style-type: none"> 1. User selects to create a request 2. System pulls up information about the event, including the name, date, location, and volunteer and vendor needs 3. User fills out request form 4. User sends the request to the OR 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none"> 3a. In step 3 of the normal flow, if the user wants to save the request for later <ol style="list-style-type: none"> 1. User chooses to save the request instead of sending it 		
Exceptions:	<ol style="list-style-type: none"> 3a. In step 3 of the normal flow, if there is no org rep for the event <ol style="list-style-type: none"> 1. User is alerted that there is no one to send the request to 		
Includes:			
Frequency of Use:	Whenever a new request needs to be sent to an OR		
Special Requirements:			
Assumptions:			
Notes and Issues:	<ol style="list-style-type: none"> 1. 		

3.2.8.2.3 Suggestions Create

Use Case ID:	1038		
Use Case Name:	Suggestions Create		
Created By:	Derrick Nagy	Last Updated By:	Derrick Nagy
Date Created:	09/21/2021	Last Revision Date:	10/13/2021
Actors:	Event Planner		
Description:	Add items to the suggestions page so that the Event Organizational Rep can see and then take action on suggestions		
Trigger:	Event Organizational Rep has been added to event creation by admin		
Preconditions:	<ol style="list-style-type: none"> 1. Event must be created 2. Event must have tasks, itinerary items, location, other settings, that needs to be reviewed 		
Postconditions:	<ol style="list-style-type: none"> 1. Event Planner and Event Organizational Rep(EOR) can see what needs approval/disapproval, or comments 		
Normal Flow:	<ol style="list-style-type: none"> 1. EOR added to event creation by admin 2. System creates a page with the, location, that could be approved by the EOR 3. System sends message(reminder) to Event Planner to make sure that they want feedback on what was pre-populated 4. Event Planner edits what they would like the EOR to view <ol style="list-style-type: none"> 1. Title 2. Date 3. Location 4. Budget 5. Each Budget Category 6. Vendor Type 7. Each Vendor Category 2. System sends a message (or reminder) to the EOR that they have items to review 3. EOR can view the event 		
Alternative Flows: [Alternative Flow 1 – Nothing to get approved]	<ol style="list-style-type: none"> 4a. After viewing, there is nothing to make suggestions on <ol style="list-style-type: none"> 1. System tells Event Planner there is nothing to send 2. System creates a default page with a message saying there is nothing to review and asks if it would like to send a message to EOR or Event Planner 		
Exceptions:	Nothing in pre-populated list because event isn't fully formed yet		
Includes:	Suggestion Creation, Suggestion Submission, Adding User Role		
Frequency of Use:	A few times during the life of the event if there is an EOR		
Special Requirements:			
Assumptions:			
Notes and Issues:			

3.2.8.2.4 Suggestions Submission

Use Case ID:	1005		
Use Case Name:	Suggestions Submission		
Created By:	Derrick Nagy	Last Updated By:	Derrick Nagy
Date Created:	09/21/2021	Last Revision Date:	10/13/2021
Actors:	Event Organizational Rep(EOR)(primary), Event Planner(secondary)		
Description:	Submit suggestions, comments, or feedback during the planning process so that the event Organizational Rep can give input to the event's creation		
Trigger:	Event Planner adds an Event Organizational Rep		
Preconditions:	<ol style="list-style-type: none"> 1. The event must be created 2. The event has an Event Organizational Rep(s) 3. The Event Planner has created something for the EOR to approve of (Use case 1038_Suggestions Creation has been completed) 		
Postconditions:	<ol style="list-style-type: none"> 1. Minimum: Event Org Rep can view the event 2. Event Org Rep can approve, disapprove, or comment on aspects of the event 3. Event Planner is notified that EOR made approvals, disapproves, or comments (that it has been changed) 		
Normal Flow:	<ol style="list-style-type: none"> 1. System grabs the list of items that the Event Planner would like to have feedback on (use case 1038_Suggestions Creation) 2. Event Organizational Rep(EOR) views the event 3. Event Organizational Rep selects part of the event that they would like to approve, disapprove, or comment on for suggestions <ol style="list-style-type: none"> 1. Title 2. Date 3. Location 4. Budget 5. Each Budget Category 6. Vendor Type 7. Each Vendor Category 2. EOR approves of the changes 3. Event Planner receives a notification about the change 4. Event Planner can view the feedback 		
Alternative Flows: [Alternative Flow 1 – Nothing to review]	<ol style="list-style-type: none"> 3b. In step 3, the EOR has nothing to approve/disapprove or comment on <ol style="list-style-type: none"> 1. System asks EOR if they would like to instead send a message to the Event Planner 		
Exceptions:	<ol style="list-style-type: none"> 1. The comment is too long, and the user is asked to make it smaller 		
Includes:	Event Creation, Admin, Send Reminder, Suggestions View, Suggestions Creation		
Frequency of Use:	Only when there is an EOR and when they have items to give feedback on		
Special Requirements:			
Assumptions:	Event Planner has finished asking for feedback		
Notes and Issues:			

3.2.8.2.5 Suggestions View

Use Case ID:	1037	
Use Case Name:	Suggestions View	
Created By:	Derrick Nagy	Last Updated By:
Date Created:	09/21/2021	Last Revision Date:
Actors:	Event Planner(primary), Event Organizational Rep(secondary)	
Description:	View the itinerary and task list so that Event Organizational Rep can make suggestions by approving, disapproving, or commenting on what is happening	
Trigger:	Event Planner has finished creating an event (Use Case Suggestions Creation)	
Preconditions:	<ol style="list-style-type: none"> 1. Event must be created 2. Event must have tasks, itinerary items, location, other settings, that needs to be reviews 	
Postconditions:	<ol style="list-style-type: none"> 1. Event Planner and Event Organizational Rep(EOR) can see what needs approval/disapproval, or comments 	
Normal Flow:	<ol style="list-style-type: none"> 1. EOR or Event Planner can see what Suggestions have been made 	
Alternative Flows: [Alternative Flow 1 – Nothing to get approved]	<ol style="list-style-type: none"> 1a. Nothing to make suggestions about or get approved <ol style="list-style-type: none"> 1. System creates a page letting the user know that there is nothing to view and gives the option to send a message to EOR/Event Planner 	
Exceptions:	<ol style="list-style-type: none"> 1. No items for the EOR to review, 	
Includes:	Suggestion Creation, Suggestion Submission	
Frequency of Use:	As often as the Event Planner and EOR are communicating	
Special Requirements:		
Assumptions:	There are items for the EOR to view	
Notes and Issues:		

3.3 Volunteers Use Cases

3.3.1 Event Details

3.3.1.1 Event Details Use Case Model



3.3.1.2 Use Cases

1.1.1.3 Volunteer View Requests

Use Case ID:	2002		
Use Case Name:	Volunteer View Requests		
Created By:	Austin T	Last Updated By:	Austin T
Date Created:	9-22-21	Last Revision Date:	11-15-21
Actors:	Open Volunteer Specific Volunteer Supply Donor		
Description:	The volunteers are able to view a list of all requests currently pending for them		
Trigger:	Volunteer goes to view his current volunteer requests		
Preconditions:	<ol style="list-style-type: none">1. Must be a volunteer2. Must have a profile		
Postconditions:	<ol style="list-style-type: none">1. Volunteer user is able to view their pending requests2. Are able to accept or reject using “Accept or Reject Requests” use case		
Normal Flow:	<ol style="list-style-type: none">1. Volunteer navigates to their profile2. Opens the “Pending Requests” tab3. Pulls a list of all current incoming and outgoing requests		
Alternative Flows:	2a. In step 2 of the normal flow, if the user has no pending requests <ol style="list-style-type: none">1. A message will appear in the middle of the list “You have no pending requests”		
Exceptions:	N/A		
Includes:	Accept or Reject Requests		
Frequency of Use:	On Demand		

Special Requirements:	Page to view requests
Assumptions:	N/A
Notes and Issues:	How many requests should be allowed on the list?

1.1.1.1.4 Volunteer Accept or Deny Request

Use Case ID:	2001		
Use Case Name:	Accept or Deny Volunteer Request		
Created By:	Austin T	Last Updated By:	Austin T
Date Created:	9-16-21	Last Revision Date:	11-15-21
Actors:	Open Volunteer Specific Volunteer Supply Donor		
Description:	The three volunteer users need to be able to accept volunteer requests (e.g. an event host asking a supply donor if they can supply their goods once again, an event manager requesting from their “Favorite Volunteers” to volunteer again, etc.) or deny them.		
Trigger:	Event Manager requests the volunteer for their service		
Preconditions:	1. User has an active profile 2. User has a volunteer / donor profile (discuss whether or not we’re having separate volunteer profiles or how we’ll distinguish)		
Postconditions:	1. Accepts the user’s request for volunteer work 2. Volunteer user is able to see the volunteer work under their profile under a certain section 3. Alerts the requester the request has been accepted		
Normal Flow:	User Accepts 1. User / Event Manager can request for a certain volunteer or a wide range of volunteers 2. An alert will appear on the volunteer user’s notification box and profile (maybe email as well) 3. The volunteer user pops up information about the event, the volunteer work, and an accept & deny option 4. The volunteer user accepts the volunteer work and it sends an alert to the requester and volunteer user 5. The volunteer work will appear on the volunteer user’s profile under a certain category User Rejects 1. User / Event Manager can request for a certain volunteer or a wide range of volunteers		

	<ol style="list-style-type: none"> 1. An alert will appear on the volunteer user's notification box and profile (maybe email as well) 1. The volunteer user pops up information about the event, the volunteer work, and an accept & deny option 1. The volunteer rejects the volunteer work and it sends an alert to the requester and volunteer user 1. The volunteer work is removed from the pending list
Alternative Flows: [Alternative Flow 1 – Not in Network]	N/A
Exceptions:	<p>4a. In step 4 of the normal flow, if the user accidentally denies the request</p> <ol style="list-style-type: none"> 1. The user is able to go back within 24 hours and reopen the request to accept it. 2. It will notify the requester that the volunteer user has accepted the request.
Includes:	<p>A use case for sending requests</p> <p>A use case for denying requests</p>
Frequency of Use:	On demand
Special Requirements:	<p>A section under profile that shows all the pending requests for the user</p> <p>A section under profile that shows all accepted requests for the user</p> <p>A section under profile that shows all denied requests for the user</p> <p>A section under profile that shows all volunteer assignments for the user</p>
Assumptions:	<p>The user understands English</p> <p>The user is a volunteer</p>
Notes and Issues:	<ol style="list-style-type: none"> 1. Should there be an undetermined option? 2. TBD - A profile tab that shows all the pending requests for the user 3. TBD - A profile tab that shows all the volunteer work assigned

1.1.1.1.5 Volunteer View Outgoing Requests

Use Case ID:	2006		
Use Case Name:	Volunteer View Outgoing Requests		
Created By:	Austin T	Last Updated By:	Austin T
Date Created:	11-16-21	Last Revision Date:	11-16-21
Actors:	Supply Donor Open Volunteer Specific Volunteer		
Description:	The user needs to be able to view a list of all outgoing requests to volunteer / supply for events		
Trigger:	The user views their outgoing requests		
Preconditions:	<ol style="list-style-type: none"> 1. The user has made a request to volunteer / supply <ol style="list-style-type: none"> 1. If not, list will be empty 		
Postconditions:	<ol style="list-style-type: none"> 1. User is able to view their list of outgoing requests and the details of the event which the request applies to 		
Normal Flow:	<ol style="list-style-type: none"> 1. The user clicks on their currently pending requests 2. The page is populated with their currently outgoing requests 		
Alternative Flows: [Alternative Flow 1 – Not in Network]			
Exceptions:			
Includes:			
Frequency of Use:	On demand		
Special Requirements:			
Assumptions:			

Notes and Issues:	
--------------------------	--

1.1.1.6 View Events

Use Case ID:	2008		
Use Case Name:	View Events		
Created By:	Austin T	Last Updated By:	Derrick Nagy
Date Created:	11-16-21	Last Revision Date:	3/24/2022
Actors:	Supply Donor Specific Volunteer Open Volunteer		
Description:	User is able to view a list of all events and sort by their choosing. Also able to sort by which skill set are needed		
Trigger:	User clicks on the "Events" tab		
Preconditions:	1. Must be a registered user		
Postconditions:	1. User is able to view a list of all events and is able to sort through the results with sorting and filtering		
Normal Flow:	1. User clicks on the "Events" tab 2. The page is populated with a list of all 3. The user is able to sort through the results by location, name, organizer, and date. 4. Also able to filter the results by events needing certain skill sets		
Alternative Flows: [Alternative Flow 1 – Not in Network]			

Exceptions:	
Includes:	
Frequency of Use:	On demand
Special Requirements:	
Assumptions:	
Notes and Issues:	

1.1.1.1.7 View Event Details

Use Case ID:	2009	
Use Case Name:	View Event Details	
Created By:	Austin T	Last Updated By:
Date Created:	11-16-21	Last Revision Date:
Actors:	Supply Donor Open Volunteer Specific Volunteer	
Description:	The user needs to be able to view a specific event and view the details for it (date, time, location, etc.)	
Trigger:	The user clicks on a specific event	
Preconditions:	1. Must be a registered user	
Postconditions:	1. The user is able to view the details of the event	
Normal Flow:	1. The user is currently under the “Events” tab and is viewing a list of events 2. The user clicks on an event he wishes to volunteer / supply for 3. A window / page is populated with the event details and the option to request to volunteer / supply	
Alternative Flows: [Alternative Flow 1 – Not in Network]	1. The user is viewing user profiles and sees an event 2. The user clicks on the event and is shown the event details with the option to request to volunteer / supply	
Exceptions:		
Includes:	Request to Volunteer / Supply use case	
Frequency of Use:	On Demand	
Special Requirements:		
Assumptions:		

Notes and Issues:	
--------------------------	--

1.1.1.1.8 View Volunteer Tasks

Use Case ID:	2010		
Use Case Name:	View Volunteer Task		
Created By:	Mujahid segyer	Last Updated By:	Austin T
Date Created:	09/17/2021	Last Revision Date:	12-05-21
Actors:	Specific Volunteers Open Volunteers Supply Donors		
Description:	The user needs to be able to view their tasks for an event		
Trigger:	User views their tasks for an event		
Preconditions:	1. Must be volunteering for an event 2. Must be a volunteer / supplier		
Postconditions:	Is able to view their tasks for a certain event		
Normal Flow:	1. User goes to view their tasks for an event 2. Clicks on the "tasks" option under their current events 3. Page is populated with a list of their tasks		
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. User has not been assigned any tasks 1. The page will show "No current tasks"		
Exceptions:			
Includes:			
Frequency of Use:	On Demand		
Special Requirements:			

Assumptions:	Understands English
Notes and Issues:	

1.1.1.1.9 Cancel Obligation

Use Case ID:	2004	
Use Case Name:	Cancel Volunteer Obligation	
Created By:	Austin T	Last Updated By:
Date Created:	11-16-21	Last Revision Date:
Actors:	Open Volunteer Specific Volunteer Supply Donor	
Description:	The user needs to be able to cancel or back out of volunteering for a specific event	
Trigger:	The user cancels their obligation towards the event	
Preconditions:	1. Is a volunteer / donor 2. Is currently volunteering for an event	
Postconditions:	1. The user is able to cancel their volunteering for an event 2. If the user cancels the day of, the user [has a rating star removed?]	
Normal Flow:	1. The user is volunteering / supplying for an event and decides to back out of volunteering 2. User clicks on "Cancel" option underneath their tasks 3. The user is asked if they are sure and hits accept	
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. If the user decides not to hit accept 1. The user decides to cancel and is taken back to the task page	
Exceptions:		
Includes:	View tasks	
Frequency of Use:	Not frequently	
Special Requirements:		
Assumptions:	Is a volunteer Understands English	

Notes and Issues:	
--------------------------	--

1.1.1.10Review Event

Use Case ID:	2007	
Use Case Name:	Volunteer Review Event	
Created By:	Austin T	Last Updated By:
Date Created:	11-16-21	Last Revision Date:
Actors:	Supply Donor Open Volunteer Specific Volunteer	
Description:	The user needs to be able to give a review to the specific event. Needs to be able to give a title, star rating, and description	
Trigger:	The user goes to review an event they volunteered / supplied for	
Preconditions:	1. Must have volunteered / supplied for the event	
Postconditions:	1. The user is able to create a review on the event	
Normal Flow:	1. User clicks on the reviews for an event 2. Clicks on add review 3. User fills out the form 4. User submits and the review is live	
Alternative Flows: [Alternative Flow 1 – Not in Network]		
Exceptions:	1. User does not include a title or a rating	
Includes:		
Frequency of Use:	After events	
Special Requirements:		

Assumptions:	
Notes and Issues:	

1.1.1.11Request to Volunteer

Use Case ID:	2005		
Use Case Name:	Request to volunteer / supply		
Created By:	Vinayak Deshpande	Last Updated By:	Austin T
Date Created:	09/23/21	Last Revision Date:	11-15-21
Actors:	Open Volunteers Specific Volunteers Supply Donor		
Description:	Volunteers and supply donors can request to volunteer / supply for an event		
Trigger:	User clicks button on events list page.		
Preconditions:	<ol style="list-style-type: none"> 1. Users want to work events indiscriminately. 2. There are events to be worked. 		
Postconditions:	<ol style="list-style-type: none"> 1. Event Managers/Planners can see users in a list of volunteers who are willing to work their event. 		
Normal Flow:	<ol style="list-style-type: none"> 1. User navigates to the events list page. 2. User clicks a button at the top or bottom of the page that says select all or something similar. 3. User hits submit. 4. User is added to lists of available volunteers / suppliers for all events. 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none"> 4. User goes through and deselects events they do not want to work. 5. User hits submit 1. User is added to lists of available volunteers for all selected events. 		
Exceptions:			
Includes:			
Frequency of Use:	Few times a week.		
Special Requirements:			

Assumptions:	
Notes and Issues:	

1.1.1.12View Task

Use Case ID:	2025	
Use Case Name:	View List of Volunteer Tasks	
Created By:	Austin T	Last Updated By:
Date Created:	11-16-21	Last Revision Date:
Actors:	Event Manager	
Description:	The user needs to be able to view the volunteers that are assigned to specific tasks	
Trigger:	The user views the event details and the tasks	
Preconditions:	1. Must have volunteer tasks	
Postconditions:	1. Able to view a list of tasks that are currently assigned or not to a specific volunteer	
Normal Flow:	1. The user is viewing their event details 2. They navigate to the tasks page 3. The page is populated with their current volunteer tasks as well as who is assigned to what	
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. The user has no tasks 1. The page will state that there are no current tasks	
Exceptions:		
Includes:		
Frequency of Use:	On demand	
Special Requirements:		
Assumptions:		

Notes and Issues:	
--------------------------	--

1.1.1.13 View Proof

Use Case ID:	2016	
Use Case Name:	View Volunteer Proof	
Created By:	Austin T	Last Updated By:
Date Created:	11-16-21	Last Revision Date:
Actors:	Open Volunteer Specific Volunteer Supply Donor	
Description:	The user needs to be able to view their volunteer proof for certain events under their tasks	
Trigger:	The user goes to view their proof under a certain task for an event	
Preconditions:	<ol style="list-style-type: none"> 1. Must be assigned to a task for an event 2. Must be a volunteer 	
Postconditions:	<ol style="list-style-type: none"> 1. The user is able to view their current volunteer proof for a task (if any) 	
Normal Flow:	<ol style="list-style-type: none"> 1. User navigates to their tasks for an event 2. They click on a specific task 3. They go to the proof section 4. The page is populated with any proof submitted by them 	
Alternative Flows: [Alternative Flow 1 – Not in Network]	4a. The user has not submitted any proof <ol style="list-style-type: none"> 1. The page will show “No current proof” and will prompt to input proof 	
Exceptions:		
Includes:	Add / Remove Volunteer Proof use case Change Volunteer Proof use case	
Frequency of Use:	On demand	
Special Requirements:		
Assumptions:		

Notes and Issues:	
--------------------------	--

1.1.1.14Change Proof

Use Case ID:	2017	
Use Case Name:	Change Volunteer Proof	
Created By:	Austin T	Last Updated By:
Date Created:	11-16-21	Last Revision Date:
Actors:	Open Volunteer Specific Volunteer Supply Donor	
Description:	The user needs to be able to modify / change any of their currently submitted proof	
Trigger:	The user clicks on the Edit button for a certain proof submitted by them	
Preconditions:	1. Must have submitted volunteer proof	
Postconditions:	1. The user is able to change out any photos or descriptions in a specific volunteer proof	
Normal Flow:	1. The user navigates to a volunteer proof under their tasks 2. They click on the edit button 3. They can change / modify / remove / add any photos or description previously submitted 4. The user saves the changes	
Alternative Flows: [Alternative Flow 1 – Not in Network]	4a. The user does not save the changes 1. The user cancels instead of saving the document 2. The proof changes are discarded and closed	
Exceptions:		
Includes:	View Volunteer Proof use case	
Frequency of Use:	On demand	
Special Requirements:		
Assumptions:		

Notes and Issues:	
--------------------------	--

1.1.1.1.15 Add or Remove Proof

Use Case ID:	2015		
Use Case Name:	Add / Remove Volunteer Proof		
Created By:	Austin T	Last Updated By:	
Date Created:	11-16-21	Last Revision Date:	
Actors:	Open Volunteer Specific Volunteer Supply Donor		
Description:	The user needs to be able to add and remove volunteer proof from a certain task that was assigned to them		
Trigger:	The user adds / removes volunteer proof to a specific task		
Preconditions:	<ol style="list-style-type: none"> 1. Must be a volunteer 2. Must be assigned to a task in an event 		
Postconditions:	<ol style="list-style-type: none"> 1. Volunteer proof will be added for a certain task under an event 2. The event planner is able to view this proof as well 		
Normal Flow:	<ol style="list-style-type: none"> 1. The user completes their volunteer task / in the midst of completing it 2. The user takes a picture / fills out a description of their task 3. The user fills out the volunteer proof form under their tasks 4. The user submits the form 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	4a. The user does not submit the form <ol style="list-style-type: none"> 1. The user cancels instead of submitting 2. The proof is cancelled and closed 		
Exceptions:			
Includes:	View Volunteer Proof use case View Volunteer Task use case		

	Change Volunteer Proof use case
Frequency of Use:	On demand
Special Requirements:	
Assumptions:	
Notes and Issues:	

1.1.1.16 View List of Favorites Volunteers

Use Case ID:	2021		
Use Case Name:	View List of Favorite Volunteers		
Created By:	Austin T	Last Updated By:	Austin T
Date Created:	9-18-21	Last Revision Date:	11-15-21
Actors:	Event Manager		
Description:	The Event Manager user needs to be able to view their list of favorite volunteers to be able to request their help or donations again		
Trigger:	The Event Manager user clicks on “Favorite Volunteers” button on their profile		
Preconditions:	<ol style="list-style-type: none"> 1. Must be an Event Manager user 2. Must be on their profile 		
Postconditions:	<ol style="list-style-type: none"> 1. Event Manager is able to view the list of Favorite Volunteers 2. The “Favorite Volunteers” list is pulled to the main page in fullscreen 		
Normal Flow:	<ol style="list-style-type: none"> 1. Event Manager goes to their profile 2. On the [left/right] side of their profile, they’re able to see a mini list of favorite volunteers with a link to see the whole list 3. They click on the link / button 4. The Favorite Volunteers list is maximized and put on the main page 		
Alternative Flows: (No Favorite Volunteers)	<p>2a. In step 2 of the normal flow, if the Event Manager user does not have any users listed under Favorite Volunteers</p> <ol style="list-style-type: none"> 1. Message in place of the listings “You do not have any current Favorite Volunteers” <p>4a. In step 4 of the normal flow, if the Event Manager user does not have any users listed under Favorite Volunteers</p> <ol style="list-style-type: none"> 1. The page will be maximized like normal, except the message stating “You do not have any current Favorite Volunteers. You may add new ones below.” 		

Exceptions:	N/A
Includes:	Edit Favorite Volunteers Use Case Request Favorite Volunteer Use Case
Frequency of Use:	On Demand
Special Requirements:	A window with a list of Favorite Volunteers. Has a way to request those volunteers next to their name / profile on the list. A way to add to the list at the bottom of the window.
Assumptions:	Understands English
Notes and Issues:	What should the max size of Favorite Volunteers be? Should there be additional info next to the volunteers name?

1.1.1.17 Add or Remove to Favorite Volunteers

Use Case ID:	2020		
Use Case Name:	Add or Remove to Favorite Volunteer List		
Created By:	Austin T	Last Updated By:	Austin T
Date Created:	9-18-21	Last Revision Date:	11-16-21
Actors:	Event Manager		
Description:	The Event Manager user needs to be able to add and remove volunteer users from their Favorite Volunteer list.		
Trigger:	The user tries to add or remove a user from the Favorite Volunteer list.		
Preconditions:	1. User is an Event Manager user 2. User is viewing the Favorite Volunteer page		
Postconditions :	1. Event Manager user adds or removes a volunteer to / from the list 2. The list is updated to show the changes		
Normal Flow:	<p>Event Manager Adds to the list</p> <ol style="list-style-type: none"> 1. Event Manager has the Favorite Volunteer list open 2. Clicks on the “Add Volunteer” at the bottom of the page 3. Small window pops up asking for the user’s name. Suggestions will appear underneath the text 4. Event manager then clicks the button “Confirm” at the bottom of the window 5. The Event Manager can exit the window and return back to the list or continue inputting volunteers <p>Event Manager Removes from the list</p> <ol style="list-style-type: none"> 1. Event Manager has the Favorite Volunteer list open 1. The Event Manager can hover over each volunteers name and see bubbles appear to the left of their name <ol style="list-style-type: none"> 1. The Event Manager can now click on the bubbles and see a checkmark fill it 1. The “Remove Volunteers” at the bottom of the page appears 1. The Event Manager clicks on the “Remove Volunteers” button 1. A pop up window appears asking if they are sure 1. They select “Confirm” 1. The users are removed from the list and the Event Manager is taken back to the list 		
Alternative Flows:	4a. In step 4 of the normal flow, if the user doesn’t click Confirm <ol style="list-style-type: none"> 1. The Event Manager has the user’s name entered 2. The Event Manager instead clicks Cancel 3. The user will be taken back to the list page 11a. In step 11 of the normal flow, if the user doesn’t click Confirm		

	<ol style="list-style-type: none"> 1. The event Manager has the volunteers selected 2. Instead of clicking Confirm, clicks Cancel 3. The user is taken back to the list page
Exceptions:	4a. In step 4 of the normal flow, if the user doesn't input a valid volunteers name with a profile <ol style="list-style-type: none"> 1. A small error will appear below the search bar 2. "User is not found" 3. They can try retyping the name or leave the add page and go back to the list page
Includes:	2013_View_Favorite_Volunteer_List
Frequency of Use:	On Demand
Special Requirements:	A page for adding that has a search bar in the middle and shows profile suggestions when the Event Manager types in a name. A confirm and cancel button at the bottom of the small window for adding and removing
Assumptions:	<ol style="list-style-type: none"> 1. The Event Manager understands English 2. The user knows the name of the volunteer 3. The user has used the "bubble delete" method
Notes and Issues:	<ol style="list-style-type: none"> 1. Should the add page have any other additions to it other than "Search for user"?

1.1.1.18View Volunteer Request

Use Case ID:	2022		
Use Case Name:	Event Manager View Volunteer Requests		
Created By:	Austin T	Last Updated By:	Austin T
Date Created:	9-23-21	Last Revision Date:	11-15-21
Actors:	Event Manager		
Description:	The Event Manager needs to be able to view all volunteer requests for their event		
Trigger:	The Event Manager goes to view the list of requests to volunteer		
Preconditions:	<ol style="list-style-type: none"> 1. User is the Event Manager 2. User has a profile 3. User is signed in 		
Postconditions:	<ol style="list-style-type: none"> 1. User is able to view a list of all volunteer requests for their event 2. Will be able to accept or reject any of them 		
Normal Flow:	<ol style="list-style-type: none"> 1. Event Manager user navigates to their profile 2. Clicks on the "Pending Requests" tab 3. A list of all pending volunteer requests for their event(s) appear 4. The user is then able to view the volunteers ratings by their name on a scale of 1-5 or view their profile 		
Alternative Flows:	<p>2a. In step 2 of the normal flow, if there are no pending requests</p> <ol style="list-style-type: none"> 1. A message "There are no current pending requests." will appear in the middle of the list 		
Exceptions:	N/A		
Includes:	Accept or reject volunteer requests use case View profile use case		
Frequency of Use:	On Demand		
Special Requirements:	A page to view the list of pending requests		

Assumptions:	N/A
Notes and Issues:	Which group is making the use case to view profiles?

1.1.1.19 Accept or Reject Volunteer Request

Use Case ID:	2023	
Use Case Name:	Accept / Reject Volunteer Requests	
Created By:	Austin T	Last Updated By:
Date Created:	11-16-21	Last Revision Date:
Actors:	Event Manager	
Description:	User needs to be able to accept or reject any requests to volunteer for their event	
Trigger:	The user is viewing volunteer requests for their event and accepts / rejects a request	
Preconditions:	<ol style="list-style-type: none"> 1. Must be an event manager 2. Must have volunteer requests for their event 	
Postconditions:	<ol style="list-style-type: none"> 1. The request is accepted or rejected 2. The volunteer is notified 	
Normal Flow:	<ol style="list-style-type: none"> 1. The event planner is viewing their pending volunteer requests 2. The user selects a specific request 3. They either... <ol style="list-style-type: none"> 1. Accept the request and the request is placed under accepted requests 2. Rejected the request and the request is placed under the rejected requests 2. The volunteer is notified 3. The volunteer is placed into the event specifics / planning 	
Alternative Flows: [Alternative Flow 1 – Not in Network]		
Exceptions:		
Includes:	View Volunteer Requests use case	
Frequency of Use:	On demand	
Special Requirements:		

Assumptions:	
Notes and Issues:	

1.1.1.1.20View Volunteers

Use Case ID:	2024		
Use Case Name:	View List of Volunteers		
Created By:	Vinayak Deshpande	Last Updated By:	Austin T
Date Created:	09/24/21	Last Revision Date:	11-16-21
Actors:	Event Manager		
Description:	Event management can see a list of volunteers that can be tapped to work at their event.		
Trigger:	Event Management clicks a button to find volunteers.		
Preconditions:	<ol style="list-style-type: none"> 1. Event is established 2. Volunteers are available 3. Management is looking for Volunteers 		
Postconditions:	<ol style="list-style-type: none"> 1. Volunteers will receive requests from event management. 2. Volunteers can choose to accept. 		
Normal Flow:	<ol style="list-style-type: none"> 1. Event management needs volunteers 2. They go to the volunteers section 3. They click "find volunteers" 4. A list is populated of volunteers who have selected their event as one that they can work 5. They can go through the list and select volunteers 6. They hit submit at the bottom of the screen. 7. Requests are sent to the volunteers. 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none"> 1. The volunteers can be sorted by Specific Volunteers (and then filtered by specific skill sets), Open Volunteers, and Supply Donors 2. The volunteers can be filtered by location, availability, etc. 		
Exceptions:			
Includes:			
Frequency of Use:	Once or twice per event.		
Special Requirements:			

Assumptions:	
Notes and Issues:	

1.1.1.21 View Volunteer Rating

Use Case ID:	2026		
Use Case Name:	View Volunteer Ratings		
Created By:	Austin T	Last Updated By:	Austin T
Date Created:	9-19-21	Last Revision Date:	11-16-21
Actors:	Event Manager		
Description:	The Event Manager is able to check the user's page for their rating or see a 1-5 star rating appear next to their name when requesting or receiving a request		
Trigger:	The Event Manager tries to view the reviews a volunteer has whenever the Event Manager requests for that volunteer, or a user is volunteering their services for the Event Manager.		
Preconditions:	<ol style="list-style-type: none"> 1. Must be the Event Manager 2. Either: <ol style="list-style-type: none"> 1. Viewing the User's page 2. Using the request a volunteer page 3. Seeing them apply for a volunteer position 		
Postconditions:	<ol style="list-style-type: none"> 1. The Event Manager is able to view the volunteers ratings 		
Normal Flow:	<p>The Event Manager receives a request to volunteer for their event</p> <ol style="list-style-type: none"> 1. The Event Manager user is able to check all volunteer requests 2. A list will appear and they're able to check the volunteers ratings (if applicable) by their name 3. A 1-5 star rating will appear next to their name <p>The Event Manager is requesting for volunteers specifically</p> <ol style="list-style-type: none"> 1. The Event Manager user is able to search for users to request their volunteer work 1. A 1-5 star rating will appear next to their name <p>The Event Manager is looking at the volunteers page</p> <ol style="list-style-type: none"> 1. The Event Manager user is able to go to volunteer's pages to view the extent of their reviews 1. Can see their 1-5 star ratings on their profile 1. They can expand it to view all the ratings and descriptions 		
Alternative Flows:	<p>3a. & 5a. & 7a. In step 3, 5, and 7 of the normal flow, a 1-5 star rating doesn't appear because the volunteer does not have any reviews</p> <ol style="list-style-type: none"> 1. No stars will appear next to their name <p>8a. In step 8 of the normal flow, if the user has no reviews when expanded</p> <ol style="list-style-type: none"> 1. The new window that shows the list of reviews will have "This user currently has no reviews." in the middle 		

Exceptions:	N/A
Includes:	Request Volunteers View volunteer requests View profiles
Frequency of Use:	On Demand
Special Requirements:	User pages Page to view volunteer requests Page to request volunteers
Assumptions:	Understands English Knows how to request volunteers Knows how to view volunteers pages Knows how to look at volunteer requests
Notes and Issues:	<ol style="list-style-type: none"> 1. Do we want to allow anyone to view volunteer ratings instead of just Event Manager?

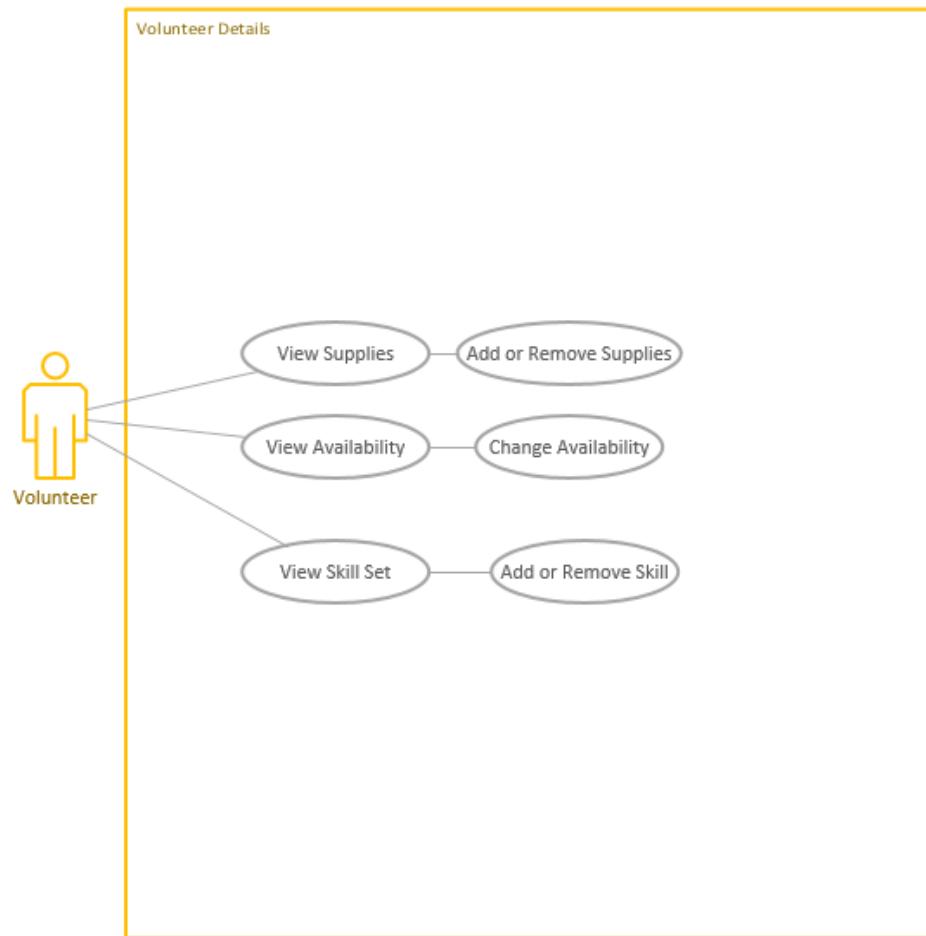
1.1.1.1.22 Add Volunteer to Task

Use Case ID:	2014		
Use Case Name:	Add Volunteer To Task		
Created By:	Jace P	Last Updated By:	
Date Created:	04-15-22	Last Revision Date:	
Actors:	Event Planner		
Description:	The user needs to be able to add one or more volunteers to a task		
Trigger:	The user adds a volunteer to a specific task		
Preconditions:	<ol style="list-style-type: none"> 1. Must be a volunteer 2. Event must be completed 		
Postconditions:	<ol style="list-style-type: none"> 1. Volunteer will be added for a certain task under an event 		
Normal Flow:	<ol style="list-style-type: none"> 1. The user adds a task to an event 2. The user specifies how many volunteers are needed for the task 3. If more than 0 Volunteers are needed the user can select from a list of volunteers to assign to the task 4. The user submits the task form 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	4a. The user does not submit the task form <ol style="list-style-type: none"> 1. The user cancels instead of submitting 2. The task is canceled and closed 		
Exceptions:			
Includes:	View Volunteer List use case Create Task use case		
Frequency of Use:	On demand		
Special Requirements:			

Assumptions:	There are volunteers in the system to select from
Notes and Issues:	

3.3.2 Volunteer Details

3.3.2.1 Feature Process Flow / Use Case Model



3.3.2.2 Use Case(s)

3.3.2.2.1 View Supplies

Use Case ID:	2018	
Use Case Name:	View Volunteer Supplies	
Created By:	Austin T	Last Updated By:
Date Created:	11-16-21	Last Revision Date:
Actors:	Supply Donor	
Description:	The user needs to be able to view their list of supplies	
Trigger:	The user views their supplies from their profile	
Preconditions:	1. Must be a supply donor	
Postconditions:	1. The user is able to view their list of supplies	
Normal Flow:	1. User navigates to their supply list under their profile 2. The page is populated with a list of their supplies	
Alternative Flows: [Alternative Flow 1 – Not in Network]	2a. The user has not put in any supplies to their supply list 1. The page shows a “No supplies” and prompts the user to add supplies	
Exceptions:		
Includes:	Add / Remove Volunteer Supplies use case	
Frequency of Use:	On demand	
Special Requirements:		

Assumptions:	
Notes and Issues:	

3.3.2.2 Add or Remove Supply

Use Case ID:	2019	
Use Case Name:	Add / Remove Volunteer Supplies	
Created By:	Austin T	Last Updated By:
Date Created:	11-16-21	Last Revision Date:
Actors:	Supply Donor	
Description:	The volunteer user needs to be able to add or remove supplies from their supply list	
Trigger:	The volunteer user adds / removes a supply to / from their list of supplies	
Preconditions:	<ol style="list-style-type: none"> 1. Must be a supply donor 	
Postconditions:	<ol style="list-style-type: none"> 1. The user is able to add to their supply list or remove from their supply list 	
Normal Flow:	<ol style="list-style-type: none"> 1. The user navigates to their supply list under their profile 2. The user either... <ol style="list-style-type: none"> 1. Adds a supply to their supply list with the name and amount 2. Removes a supply from their supply list 2. The user saves their changes 	
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none"> 3a. The user does not save their changes <ol style="list-style-type: none"> 1. The user cancels and the changes are discarded 	
Exceptions:	<ol style="list-style-type: none"> 2a. The user does not input an amount or name for their supplies <ol style="list-style-type: none"> 1. The user is notified to fill the proper section out 	
Includes:	View Volunteer Supplies use case	
Frequency of Use:	On demand	
Special Requirements:		
Assumptions:		

Notes and Issues:	
--------------------------	--

3.3.2.2.3 View Availability

Use Case ID:	2011	
Use Case Name:	View Availability	
Created By:	Austin T	Last Updated By:
Date Created:	11-16-21	Last Revision Date:
Actors:	Supply Donor Open Volunteer Specific Volunteer	
Description:	The user needs to be able to view their availability schedule	
Trigger:	The user goes to view their availability through their profile	
Preconditions:	1. Must be a volunteer	
Postconditions:	1. The user is able to view their availability schedule	
Normal Flow:	1. The user navigates to their availability under their profile 2. The page is populated with their available times on their available days	
Alternative Flows: [Alternative Flow 1 – Not in Network]	2a. The user has yet to set any available times / days 1. The list will appear empty and the user can add times and days	
Exceptions:		
Includes:	Change my availability use case	
Frequency of Use:	On demand	
Special Requirements:		
Assumptions:		

Notes and Issues:	
--------------------------	--

3.3.2.2.4 Change Availability

Use Case ID:	2013		
Use Case Name:	Request Skill Set Verification		
Created By:	Austin T	Last Updated By:	Austin T
Date Created:	11-16-21	Last Revision Date:	11-21-21
Actors:	Specific Volunteer		
Description:	The user needs to be able to request to have their skill set(s) be verified		
Trigger:	The user clicks on the “Request Verification” under their skill sets		
Preconditions:	<ol style="list-style-type: none"> 1. Must be a specific volunteer 2. Must have a skill set (unique or predefined) 		
Postconditions:	<ol style="list-style-type: none"> 1. User submits a requests to have a skill set verified 		
Normal Flow:	<ol style="list-style-type: none"> 1. User clicks on the request verification under their skill set 2. The page is populated with the form including whatever information is needed (proof) 3. They are able to include any details to help them achieve the skill set verification 4. They submit the application and are notified the application is pending 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	4a. The user hits cancel <ol style="list-style-type: none"> 1. The application is cancelled and then closed 		
Exceptions:	<ol style="list-style-type: none"> 1. The user does not fill out all the fields <ol style="list-style-type: none"> 1. They are highlighted and told to fill in the missing fields 		

Includes:	
Frequency of Use:	On demand
Special Requirements:	
Assumptions:	
Notes and Issues:	

3.3.2.2.5 View Skill Set

Use Case ID:	2031	
Use Case Name:	View Skill Set	
Created By:	Austin T	Last Updated By:
Date Created:	11-21-21	Last Revision Date:
Actors:	Specific Volunteer	
Description:	The user needs to be able to view their skill set	
Trigger:	The user views their skill set under their profile	
Preconditions:	<ol style="list-style-type: none"> 1. Must be a specific volunteer 	
Postconditions:	<ol style="list-style-type: none"> 1. The user is able to view their skill set 	
Normal Flow:	<ol style="list-style-type: none"> 1. The user navigates to their skill set under their profile 2. The page is populated with their current skill set 	
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none"> 2a. The user does not have a skill set <ol style="list-style-type: none"> 1. A message appears that states there are no skills for this current user 	
Exceptions:		
Includes:		
Frequency of Use:	On Demand	
Special Requirements:		
Assumptions:		

Notes and Issues:	
--------------------------	--

3.3.2.2.6 Add Skill to Skill Set

Use Case ID:	2032		
Use Case Name:	Add Skill to Skill Set		
Created By:	Austin T	Last Updated By:	Austin T
Date Created:	11-21-21	Last Revision Date:	11-21-21
Actors:	Specific Volunteer		
Description:	The user needs to be able to add a skill to their skill set		
Trigger:	The user adds a skill to their skill set		
Preconditions:	1. Must be a specific volunteer		
Postconditions:	1. The user is able to apply a skill to their skill set		
Normal Flow:	1. The user is viewing their skill set 2. They click on the add button 3. They can type in their desired skill set and it will be autofilled if its in the database 4. The skill is applied to their skill set		
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. If the skill is not in the database for skill sets 1. The user is able to type out their skill set and press enter (or accept) 2. The skill is applied to their skill set		
Exceptions:	1. If the user already has the skill in their skill set it will deny the user from adding it again		
Includes:	View Skill Set		

Frequency of Use:	On Demand
Special Requirements:	
Assumptions:	
Notes and Issues:	

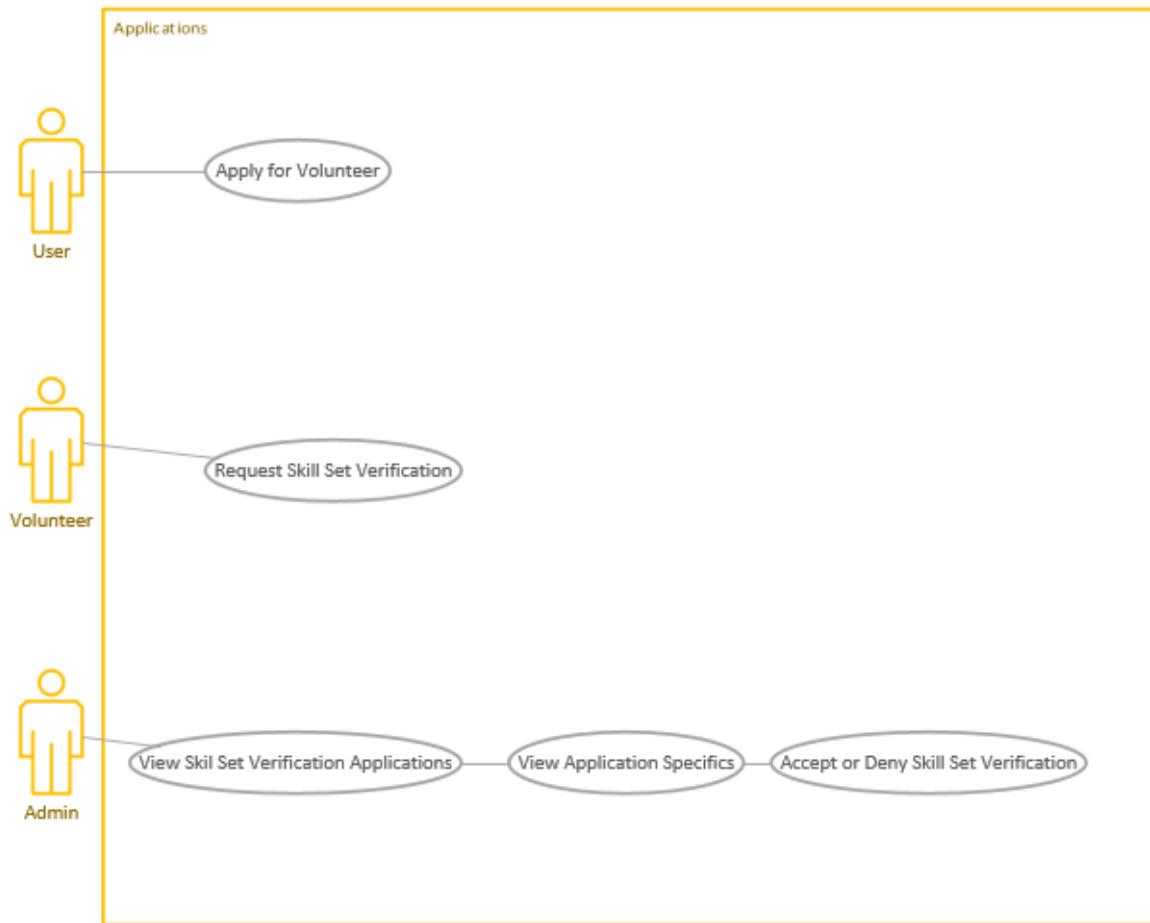
3.3.2.2.7 Remove Skill from Skill Set

Use Case ID:	2033	
Use Case Name:	Remove Skill from Skill Set	
Created By:	Austin T	Last Updated By:
Date Created:	11/22/21	Last Revision Date:
Actors:	Specific Volunteer	
Description:	The user needs to be able to remove a skill from their skill set	
Trigger:	The user removes a skill from their skill set	
Preconditions:	<ol style="list-style-type: none"> 1. Must be a specific volunteer 2. Must have skills in their skill set 	
Postconditions:	<ol style="list-style-type: none"> 1. The user is able to remove a skill from their skill set 	
Normal Flow:	<ol style="list-style-type: none"> 1. The user is viewing their skill set 2. They click to remove the skill from their skill set 3. They are prompted and they click agree 	
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. The user does not click agree <ol style="list-style-type: none"> 1. The deletion is cancelled 	
Exceptions:		
Includes:	View Skill Set	
Frequency of Use:	On Demand	
Special Requirements:		
Assumptions:		

Notes and Issues:	

3.3.3 Applications

1.1.1.2 Feature Process Flow / Use Case Model



1.1.1.3 Use Case(s)

1.1.1.3.1 Apply for Volunteer

Use Case ID:	2027		
Use Case Name:	Volunteer Application		
Created By:	Vinayak Deshpande	Last Updated By:	Austin T
Date Created:		Last Revision Date:	11-15-21
Actors:	All volunteers		
Description:	Application to fill out to be added to system as a volunteer		
Trigger:	User clicks the button on the home page.		
Preconditions:	1. User is a person 2. User wants to volunteer		
Postconditions:	1. Users' applications are submitted. 2. Application is sent for review and verification. 3. User is sent an email or message in app that they are under consideration.		
Normal Flow:	1. User wants to volunteer in some capacity. 2. User navigates to home page 3. User clicks the volunteer application button. 4. User is redirected to a form 5. User fills out the form 6. User submits the form. 7. Form is sent for review and verification. 8. User receives notification that they are under review.		
Alternative Flows: [Alternative Flow 1 – Not in Network]			
Exceptions:			

Includes:	
Frequency of Use:	Many times per day.
Special Requirements:	
Assumptions:	
Notes and Issues:	

1.1.1.3.2 Request Skill Set Verification

Use Case ID:	2013		
Use Case Name:	Request Skill Set Verification		
Created By:	Austin T	Last Updated By:	Austin T
Date Created:	11-16-21	Last Revision Date:	11-21-21
Actors:	Specific Volunteer		
Description:	The user needs to be able to request to have their skill set(s) be verified		
Trigger:	The user clicks on the “Request Verification” under their skill sets		
Preconditions:	<ol style="list-style-type: none"> 1. Must be a specific volunteer 2. Must have a skill set (unique or predefined) 		
Postconditions:	<ol style="list-style-type: none"> 1. User submits a requests to have a skill set verified 		
Normal Flow:	<ol style="list-style-type: none"> 1. User clicks on the request verification under their skill set 2. The page is populated with the form including whatever information is needed (proof) 3. They are able to include any details to help them achieve the skill set verification 4. They submit the application and are notified the application is pending 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	4a. The user hits cancel <ol style="list-style-type: none"> 1. The application is cancelled and then closed 		
Exceptions:	<ol style="list-style-type: none"> 1. The user does not fill out all the fields <ol style="list-style-type: none"> 1. They are highlighted and told to fill in the missing fields 		
Includes:			
Frequency of Use:	On demand		
Special Requirements:			
Assumptions:			

Notes and Issues:	
--------------------------	--

1.1.1.3.3 View Skill Set Verification Applications

Use Case ID:	2029		
Use Case Name:	View Verification Applications		
Created By:	Alaina Gilson	Last Updated By:	Austin T
Date Created:	9/23/21	Last Revision Date:	11-22-21
Actors:	Verification Manager		
Description:	The actor is able to view new verification applicants		
Trigger:	The user has received new applications		
Preconditions:	<ol style="list-style-type: none"> 1. The user is an active verification manager 2. The user has been given verification applications 		
Postconditions:	<ol style="list-style-type: none"> 1. The user can view the applicant profile from the application submitted 		
Normal Flow:	<ol style="list-style-type: none"> 1. The user views an application 2. The user clicks on the application's applicant name 3. The user is able to view the new applicant's profile 		
Alternative Flows:	N/A		
Exceptions:	1a. The applicant no longer exists <ol style="list-style-type: none"> 1. The application is allotted a time period before registering that the applicant is gone 2. The application is deleted 		
Includes:	A use case for viewing new applicant profiles		

Frequency of Use:	Very frequent
Special Requirements:	The ability to click on an applicant's username in an application in order to view their profile
Assumptions:	<ol style="list-style-type: none"> 1. The user has received applications 2. The user is proactive and wants to gain further information on new applicants
Notes and Issues:	<ol style="list-style-type: none"> 1. Changed "View List of New Applicants" to "View New Applicant Profiles" as the former was an extension of "View Applications"

1.1.1.3.4 View Application Specifics

Use Case ID:	2030		
Use Case Name:	View Specific Application		
Created By:	Alaina Gilson	Last Updated By:	Austin T
Date Created:	9/23/21	Last Revision Date:	11-16-21
Actors:	Verification Manager		
Description:	The user is able to view a specific application for details		
Trigger:	The user decides to view a specific application		
Preconditions:	<ol style="list-style-type: none"> 1. The user is an active verification manager 2. The user has been given verification applications 3. The user wants to view either new or old applications 		
Postconditions:	<ol style="list-style-type: none"> 1. The user is able to view applications 		
Normal Flow:	<ol style="list-style-type: none"> 1. The user decides to view applications 2. The user clicks on a specific application 3. The user is shown the details and specifics of the application 		
Alternative Flows:			
Exceptions:			
Includes:	A use case for viewing applications		
Frequency of Use:	Very frequent		
Special Requirements:	A location to view applications A location in the view applications to view old applications		
Assumptions:	<ol style="list-style-type: none"> 1. The user is an active verification manager 2. The user has received applications 		

Notes and Issues:	1. Should the view applications be two separate buttons on the profile or one on the profile and an older one in the view?
--------------------------	--

1.1.1.3.5 Accept or Deny Skill Set Verification

Use Case ID:	2028		
Use Case Name:	Accept or Deny Verification Applications		
Created By:	Alaina Gilson	Last Updated By:	Austin T
Date Created:	9/23/21	Last Revision Date:	11-21-21
Actors:	Admin		
Description:	The actor receives a verification application for skill sets and is able to accept or deny the skill set verification request		
Trigger:	Specific Volunteer submits an application for review.		
Preconditions:	<ol style="list-style-type: none"> 1. User has an active profile 2. User has an approved admin role 3. User has received a verification application 		
Postconditions:	<p>User accepts an application:</p> <ol style="list-style-type: none"> 1. The application is marked as accepted 2. The application is grayed and stored if needed to be viewed later in a separate location 3. The user who submitted the application is given a indication of verification <p>User denies an application:</p> <ol style="list-style-type: none"> 1. The application is marked as denied 2. The application is grayed and stored if needed to be viewed later in a separate location 3. The user who submitted the application is notified that their application was denied 		
Normal Flow:	<ol style="list-style-type: none"> 1. Volunteer user submits a verification application either with a built-in form or digital submission 2. The verification manager receives a notification that a new application has been submitted 3. Viewing the application brings it up alongside an accept button and a deny button 4. The verification manager accepts the application 5. The volunteer user receives a notification that their application has been accepted 6. The volunteer user is given a verification identifier 7. The application is stored for later viewing 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<p>4a. The verification manager denies the application</p> <ol style="list-style-type: none"> 1. The volunteer user receives a notification that their application has been denied 2. The application is stored for later viewing 		

Exceptions:	4a. If the verification manager accidentally accepts or denies an application <ol style="list-style-type: none"> 1. The user is able to go back into previous applications and, after a confirmation, be able to switch between accept or deny 2. The volunteer user is notified that their application was either accepted or denied 3. If applicable, the verification identifier is given or taken away
Includes:	A use case for accepting applications A use case for denying applications
Frequency of Use:	Very frequent
Special Requirements:	A location in the verification manager's profile to view all incoming applications A location in the verification manager's profile to view all previously accepted applications A location in the verification manager's profile to view all previously denied applications
Assumptions:	The user understands the importance of what it means to verify another user The user is capable of making good judgements about other users based on public information and what has been provided
Notes and Issues:	1. Should the stored applications be in the same location? 2. There should probably be a time frame before the verification identifier is properly put on the volunteer user, maybe 24 hours?

1.1.1.3.6 Apply For Specific Event

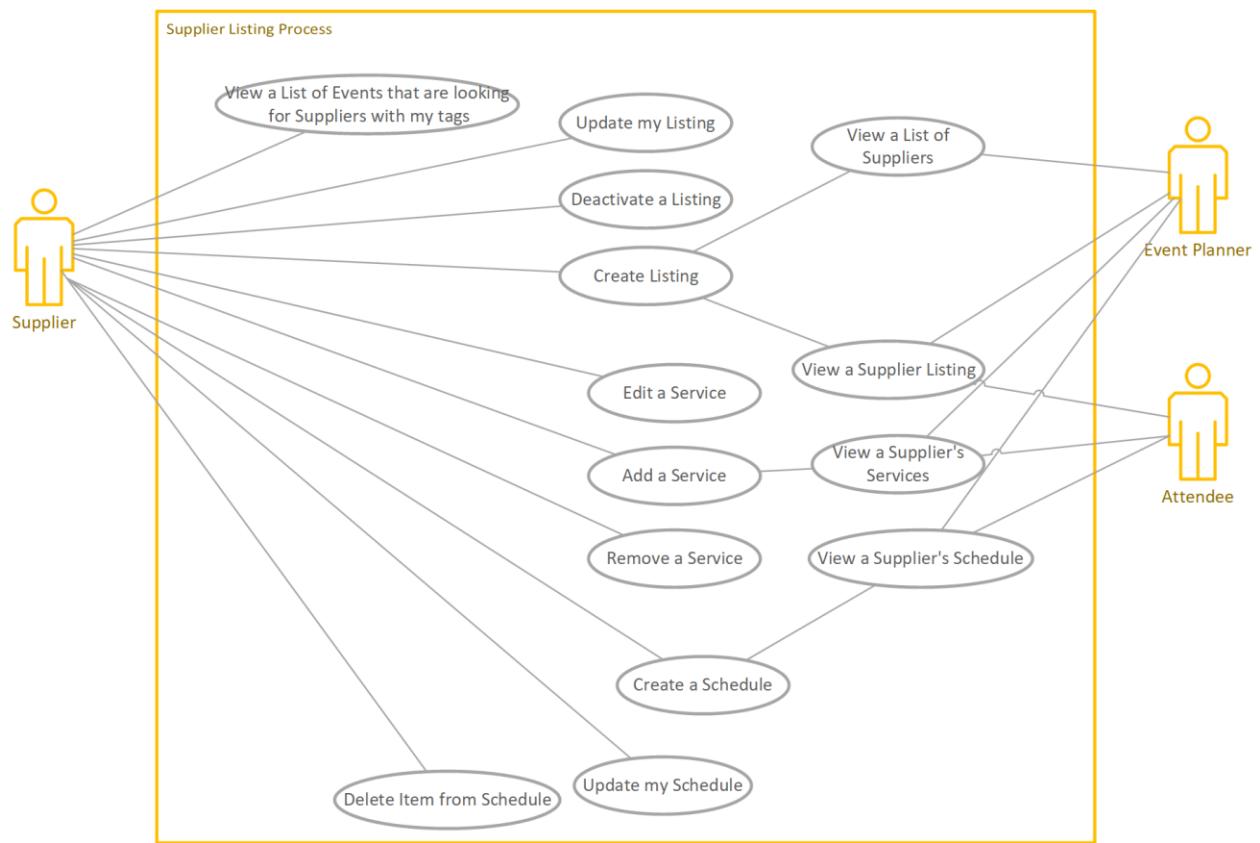
Use Case ID:	2031		
Use Case Name:	Apply For Specific Event		
Created By:	Mike Cahow	Last Updated By:	
Date Created:	5/2/22	Last Revision Date:	
Actors:	Volunteer		
Description:	The user is able to apply to volunteer for a specific event		

Trigger:	The user decides to apply for a specific event
Preconditions:	<ul style="list-style-type: none"> 4. The user is an active volunteer 5. There are active events to apply for
Postconditions:	<ul style="list-style-type: none"> 2. The user is given an eventID to correspond with the event they applied for
Normal Flow:	<ul style="list-style-type: none"> 4. The user logs in as a Volunteer 5. User goes to event list page 6. User clicks "Apply" button 7. User is given an event ID to correspond with Event
Alternative Flows:	
Exceptions:	
Includes:	A use case for applying to volunteer for a specific event
Frequency of Use:	As needed
Special Requirements:	A location for an Apply button
Assumptions:	<ul style="list-style-type: none"> 3. The user is an active volunteer
Notes and Issues:	

3.4 Supplier Use Cases

3.4.1 Supplier

3.4.1.1 Supplier Use Case Model



3.4.1.2 Supplier Use Cases

3.4.1.2.1 Add Service

Use Case ID:	3004		
Use Case Name:	Add Service		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-11-18	Last Revision Date:	
Actors:	Location User		
Description:	A user would like to add a service to their supplier listing		
Trigger:	A user selects that they'd like to add a service to their listing.		
Preconditions:	1. Current user is viewing their own supplier listing		
Postconditions:	1. Service has been added		
Normal Flow:	1. User selects the "add" button in the services page 2. User is given an empty row of service information boxes 3. User fills in the boxes 4. Data is saved and uploaded to database		
Alternative Flows:			
Exceptions:			
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:	1. How many services can a single supplier have?		

3.4.1.2.2 Add Listing

Use Case ID:	3008		
Use Case Name:	Create Listing		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-11-18	Last Revision Date:	
Actors:	User		
Description:	A user would like to create a listing for a supplier or location		
Trigger:	A user presses the "Create" button in a list view for a supplier or location		
Preconditions:	1. Current user has been given permission to create a listing for a supplier or location		
Postconditions:	1. Listing has been added 2. User is deposited on the landing screen for their new listing		
Normal Flow:	1. User selects the "Create" button in a listing view 2. User is sent to the EditBiolnformation screen for the type of listing they're currently viewing. 3. User fills out the fields 4. User selects "Confirm" at the bottom of the screen 5. Information is sent to database to be stored 6. User is deposited on the landing screen of their new listing.		
Alternative Flows:			
Exceptions:	4a: User selects "Cancel" 1. User is returned to the list for the type of listing they were attempting to create		
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:			

3.4.1.2.3 Create Schedule

Use Case ID:	3010		
Use Case Name:	Create Listing		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-11-18	Last Revision Date:	
Actors:	User		
Description:	A user would like to create a schedule item for a location or supplier		
Trigger:	A user presses the “add” button in the schedule screen for a supplier or location		
Preconditions:	1. Current user has been given permission to edit the current listing		
Postconditions:	1. Schedule item has been created 2. User is returned to their schedule page on their listing		
Normal Flow:	1. User selects “add” button in the schedule screen for a supplier or location 2. User is giving some boxes to select start and end times, as well as a check box for recurring events 3. User fills out boxes 4. User hits “Confirm” 5. Database stores schedule item 6. User is returned to their listing’s schedule screen		
Alternative Flows:			
Exceptions:	4a: User selects “Cancel” 1. User is returned to their listing’s schedule screen		
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:			

3.4.1.2.4 Deactivate Listing

Use Case ID:	3011		
Use Case Name:	Deactivate Listing		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-11-18	Last Revision Date:	
Actors:	User		
Description:	A user would like to deactivate a listing for a supplier or location		
Trigger:	A user presses the “Deactivate” button in a listing for a supplier or location		
Preconditions:	1. Current user has edit permissions for the listing		
Postconditions:	1. Listing has been deactivated 2. User is deposited on a confirmation screen.		
Normal Flow:	1. User selects the “Deactivate” button in the main listing screen 2. User is queried for confirmation that they desire to deactivate the listing 3. User selects “confirm” 4. System instructs database to mark the entity for the listing as deactivated 5. User is sent to a page confirming that their listing has been deactivated.		
Alternative Flows:	3b: User selects “cancel” 1. User is returned to their listing screen.		
Exceptions:			
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:			

3.4.1.2.5 Delete Item From Schedule

Use Case ID:	3017		
Use Case Name:	Delete Item from Schedule		
Created By:	Emma Pollock	Last Updated By:	
Date Created:	11-18-2021	Last Revision Date:	
Actors:	Supplier Vendor		
Description:	A Supplier or Vendor is no longer available at a certain time and would like to delete the schedule item for that time.		
Trigger:	The User clicks the Delete Schedule Item button		
Preconditions:	<ol style="list-style-type: none"> 1. User is viewing their schedule 2. User has selected a day 3. User has selected a schedule item 		
Postconditions:	<ol style="list-style-type: none"> 1. The schedule item is no longer in the database 2. The User's schedule is updated to no longer show the deleted schedule item 		
Normal Flow:	<ol style="list-style-type: none"> 1. User clicks the Delete Schedule Item button 2. System prompts User to Confirm that they would like to permanently delete the schedule item 3. User confirms 4. System successfully deletes the schedule item from the database 5. System displays a success message 6. User is returned to their schedule with the day still selected 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. In step 3 of the normal flow, the User cancels <ol style="list-style-type: none"> 1. Use case resumes on step 6 of normal flow 		
Exceptions:	4a. In step 4 of the normal flow, the System is unable to delete the schedule item from the database <ol style="list-style-type: none"> 1. System displays an error message 2. Use case resumes on step 6 of normal flow 		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	<ol style="list-style-type: none"> 1. "Delete Schedule Item" button may be a button with a trashcan icon 		

3.4.1.2.6 Edit Supplier Service

Use Case ID:	3020		
Use Case Name:	Edit Vendor Supplier Service		
Created By:	Emma Pollock	Last Updated By:	
Date Created:	11-18-2021	Last Revision Date:	
Actors:	Vendor Supplier		
Description:	A Location would like to change the information for one of their services		
Trigger:	The User changes a value in one of the service controls		
Preconditions:	<ol style="list-style-type: none"> 1. User is viewing their Services page in edit mode 2. User has at least one service on their page 		
Postconditions:	<ol style="list-style-type: none"> 1. The service has the updated values in the database 2. The service has the updated values on the Services page 		
Normal Flow:	<ol style="list-style-type: none"> 1. User changes the value of one or more of their services 2. User presses the Save button 3. System prompts user to confirm that they would like to change the values 4. User confirms 5. System successfully changes the values in the database 6. System displays a success message 7. User is returned to the service page no longer in edit mode 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none"> 2a. In step 2 of the normal flow, the User clicks a nav instead of pressing save <ol style="list-style-type: none"> 1. System prompts user to confirm that they would like to go to another page and lose their unsaved changes 2. User cancels 3. Use case resumes at on step 1 or 2 of normal flow 2b. In step 2 of the normal flow, the User clicks another tab or some other nav instead of pressing confirm <ol style="list-style-type: none"> 1. System prompts user to confirm that they would like to go to another tab and lose their unsaved changes 2. User confirms 3. User is moved to the other page and their changes are discarded 3a. In step 3 of the normal flow, the User cancels <ol style="list-style-type: none"> 1. Use case resumes on step 7 of normal flow 		
Exceptions:	<ol style="list-style-type: none"> 5a. In step 5 of the normal flow, the System is unable to update the service in the database <ol style="list-style-type: none"> 1. System displays an error message 2. Use case resumes on step 1 or 2 of normal flow 		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	N/A		

Assumptions:	N/A
Notes and Issues:	N/A

3.4.1.2.7 Delete Service

Use Case ID:	3023		
Use Case Name:	Delete Service		
Created By:	Kris Howell	Last Updated By:	
Date Created:	12/2/2021	Last Revision Date:	
Actors:	Supplier		
Description:	A supplier wants to remove a service from their list of offered services.		
Trigger:	User clicks a “Remove Service” button.		
Preconditions:	1. User is in the edit supplier listing user interface.		
Postconditions:	1. The specified service is no longer in their list of services on their listing. 2. Service has been removed from the database.		
Normal Flow:	1. User clicks “Remove Service” button. 2. Prompt user to confirm they would like to permanently remove the service. 3. User clicks “Remove”. 4. System removes the service from the database. 5. User is returned to the edit supplier listing interface.		
Alternative Flows:	3a. In step 3 of the normal flow, user clicks “Cancel”. 1. Return to edit listing interface, making no changes.		
Exceptions:	4a. In step 4 of the normal flow, the system fails to communicate with the database. 1. Give user an error message 2. Return to edit supplier listing user interface, leaving user's changes as they were.		
Includes:	N/A		
Frequency of Use:	Infrequently		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

3.4.1.2.8 Update Supplier Listing

Use Case ID:	3026		
Use Case Name:	Update Supplier Listing		
Created By:	Kris Howell	Last Updated By:	
Date Created:	12/2/2021	Last Revision Date:	
Actors:	Supplier		
Description:	A supplier would like to update the information in their supplier listing.		
Trigger:	User clicks a “Save Changes” button after changing fields in the Edit Supplier Listing user interface.		
Preconditions:	1. User is in the edit supplier listing interface.		
Postconditions:	1. Any edited fields have been updated on the supplier listing. 2. All corresponding database records have been updated.		
Normal Flow:	1. User changes any number of the fields in the edit supplier listing interface. 2. User clicks “Save Changes”. 3. Prompt user to confirm they would like to save all changes. 4. User clicks “Confirm”. 5. System updates the corresponding records in the database. 6. User is returned to the edit supplier listing interface.		
Alternative Flows:	4a. In step 4 of the normal flow, user clicks “Cancel”. 1. Return to edit supplier listing interface, discarding all changes.		
Exceptions:	5a. In step 5 of the normal flow, the system fails to communicate with the database. 1. Give user an error message 2. Return to edit location user interface, leaving user's changes as they were.		
Includes:	N/A		
Frequency of Use:	Infrequently		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

3.4.1.2.9 Browse Suppliers

Use Case ID:	3030	
Use Case Name:	Browse Suppliers	
Created By:	Kris Howell	Last Updated By:
Date Created:	12/2/2021	Last Revision Date:
Actors:	Event Planner Attendee	
Description:	A user would like to browse a list of suppliers, filtered by some criteria.	
Trigger:	User clicks a “Apply Filters” button after changing the criteria they would like to filter suppliers by.	
Preconditions:	1. User is in the Browse Suppliers tab.	
Postconditions:	1. Table populates with suppliers that fit the criteria.	
Normal Flow:	1. User changes any of the filter criteria in the navbar. 2. User clicks “Apply Filters”. 3. System updates the table with suppliers that match the filter criteria.	
Alternative Flows:	4a. In step 3 of the normal flow, no suppliers can be found that match the criteria. 1. Prompt the user informing them that no suppliers could be found matching those criteria, recommending loosening their filter settings. 2. Remain in browse suppliers tab.	
Exceptions:	3a. In step 3 of the normal flow, the system fails to communicate with the database. 1. Give user an error message 2. Remain in browse suppliers tab.	
Includes:	N/A	
Frequency of Use:	Frequently	
Special Requirements:	Default filter criteria present when first coming to the browse suppliers tab.	
Assumptions:	N/A	
Notes and Issues:	N/A	

3.4.1.2.10 Supplier Find Personalized Events

Use Case ID:	3028	
Use Case Name:	Find Personalized Events	
Created By:	Kris Howell	Last Updated By:
Date Created:	12/4/2021	Last Revision Date:
Actors:	Supplier	
Description:	A supplier would like to see a list of events that have flagged themselves as looking to hire suppliers like them.	
Trigger:	User clicks a “Find Jobs” button in the navbar of the Supplier tab.	
Preconditions:	1. User is in the supplier tab.	
Postconditions:	1. User is navigated to a Browse Events page, with the categories preset to filter for categories that are assigned to the user's supplier profile.	
Normal Flow:	1. User clicks “Find Jobs” button. 2. System navigates to the Browse Events page. 3. Browse events category is prefilled with the user's supplier categories. 4. Browse events table is populated with events matching the filter criteria.	
Alternative Flows:	3a. In step 3 of the normal flow, user has no categories set. 1. A message appears informing the user that they have no categories and should assign categories to their supplier profile for this feature to work properly. 2. Message box offers buttons to “Set Categories”, which navigates to the Edit Account Settings page, or “Continue”. 3. System navigates to the Browse Events page with default settings.	
Exceptions:	4a. In step 4 of the normal flow, the system fails to communicate with the database. 1. Give user an error message 2. Remain in the Browse Events page.	
Includes:	N/A	
Frequency of Use:	Frequently	
Special Requirements:	N/A	
Assumptions:	Supplier has assigned themselves category tags.	
Notes and Issues:	N/A	

3.4.1.2.11 View Specific Events

Use Case ID:	3036		
Use Case Name:	View Specific Events		
Created By:	Logan Baccam	Last Updated By:	
Date Created:		Last Revision Date:	
Actors:	Supplier		
Description:	Supplier views event		
Trigger:	Supplier clicks an event from the events that need my supplier type		
Preconditions:	User must be logged in as a supplier		
Postconditions:	Supplier will be shown events details		
Normal Flow:	<ol style="list-style-type: none"> 1. A supplier browses through events that need my supplier type listing 2. A supplier finds an event they like 3. Supplier clicks the event 4. Supplier is shown the details of an event 		
Alternative Flows:	<ol style="list-style-type: none"> 1. A supplier browses through events listing page 2. A supplier finds an event they like 3. Supplier clicks the event 4. Supplier is shown the details of an event 		
Exceptions:			
Includes:			
Frequency of Use:			
Special Requirements:			
Assumptions:			
Notes and Issues:			

3.4.1.2.12 Request Supplier

Use Case ID:	69420		
Use Case Name:	Request supplier		
Created By:	Derrick Nagy	Last Updated By:	
Date Created:	2022-04-05	Last Revision Date:	
Actors:	Event Planner		
Description:	Event planner chooses a supplier and requests their services for the event		
Trigger:	Request creation event		
Preconditions:	Event planner logged in, suppliers exist		
Postconditions:	Confirmation that supplier has been notified		
Normal Flow:	<ol style="list-style-type: none"> 1. An event planner checks list of suppliers 2. Planner selects supplier 3. Planner selects option to request supplier's services for an event 4. Planner selects the event that they would like services for 5. System finds dates for the event and the availability of the supplier for that day 6. Planner selects date they would like to request supplier 7. System retrieves list of services that the supplier provides 8. User selects services 9. User has option to add additional comments 10. User submits request 11. System confirms success 		
Alternative Flows: [1: No dates for event]	<p>5b. System finds dates for the event and the availability of the supplier for that day but no dates are selected for the event</p> <ol style="list-style-type: none"> 1. Planner can choose to find the days that the supplier has available for the next 90 days 2. Continues regular flow at step 7 		
Exceptions:			
Includes:			
Frequency of Use:			
Special Requirements:			
Assumptions:			
Notes and Issues:			

3.4.1.2.13 View Supplier Listing

Use Case ID:	3042	
Use Case Name:	View supplier listing	
Created By:	Logan Baccam	Last Updated By:
Date Created:		Last Revision Date:
Actors:	Event Planner, Attendee	
Description:	View a supplier listing	
Trigger:	User selects a supplier	
Preconditions:	User must be logged in as attendee or event manager	
Postconditions:	User is shown a suppliers listing	
Normal Flow:	<ol style="list-style-type: none"> 1. A user searches for a supplier they like 2. User selects supplier 3. User is brought to suppliers page 4. User navigates to the suppliers listing 5. User selects a listing 6. User is shown the suppliers listing details 	
Alternative Flows:	N/A	
Exceptions:	N/A	
Includes:	N/A	
Frequency of Use:		
Special Requirements:	Supplier must have listing to be able to be viewed	
Assumptions:	Most will have listings	
Notes and Issues:	N/A	

3.4.1.2.14 View Supplier Schedule

Use Case ID:	3043		
Use Case Name:	View supplier schedule		
Created By:	Logan Baccam	Last Updated By:	
Date Created:		Last Revision Date:	
Actors:	Attendee, Event Planner		
Description:	View a Supplier's scheduling		
Trigger:	User selects a Supplier		
Preconditions:	User must be an Attendee or Supplier		
Postconditions:	User will know a Suppliers availability		
Normal Flow:	<ol style="list-style-type: none">1. A user searches for a supplier they like.2. A user selects the supplier and is brought to Supplier's page3. User selects the supplier's schedule4. User is shown the suppliers availability		
Alternative Flows:	N/A		
Exceptions:	N/A		
Includes:	N/A		
Frequency of Use:	Frequently used by the fans of popular suppliers		
Special Requirements:	Supplier must have an event listed		
Assumptions:	N/A		
Notes and Issues:	N/A		

3.4.1.2.15 View Supplier Services

Use Case ID:	3044		
Use Case Name:	View supplier schedule		
Created By:	Logan Baccam	Last Updated By:	
Date Created:		Last Revision Date:	
Actors:	Attendee, Event Planner		
Description:	View a Supplier's scheduling		
Trigger:	User selects a Supplier		
Preconditions:	User must be an Attendee or Supplier		
Postconditions:	User will know what services a supplier has to offer		
Normal Flow:	<ol style="list-style-type: none">1. A user searches for a supplier they like.2. A user selects the supplier and is brought to Supplier's page3. User selects the supplier's services4. User is shown the all of the suppliers services listed		
Alternative Flows:	N/A		
Exceptions:	N/A		
Includes:	N/A		
Frequency of Use:	Frequently used by the fans of popular suppliers		
Special Requirements:	Supplier must have an event listed		
Assumptions:	N/A		
Notes and Issues:	N/A		

3.4.1.2.16 Add Image

Use Case ID:	3045		
Use Case Name:	Add Image to Image Gallery		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-12-7	Last Revision Date:	
Actors:	User		
Description:	A user would like to add an image to their image gallery		
Trigger:	A user presses the "Add a new Image" button in a listing for a supplier or location		
Preconditions:	<ol style="list-style-type: none"> 1. Current user has edit permissions for the listing they are viewing 2. Current user is in edit mode for the listing they are viewing 		
Postconditions:	<ol style="list-style-type: none"> 1. Image has been added to gallery 2. User is returned to bio edit screen 		
Normal Flow:	<ol style="list-style-type: none"> 1. User presses the "Add a new Image" button in the main listing screen edit screen 2. User is queried for an image file they want to upload 3. The selected image is stored to database and added to the page gallery 4. User is returned to the edit screen for their listing 		
Alternative Flows:	2a: User selects multiple images <ol style="list-style-type: none"> 1. Repeat step 3 for each photo 2. Resume flow from step 4 2b: User does not select an image <ol style="list-style-type: none"> 1. Resume flow from step 4 		
Exceptions:			
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:			

3.4.1.2.17 Delete Image

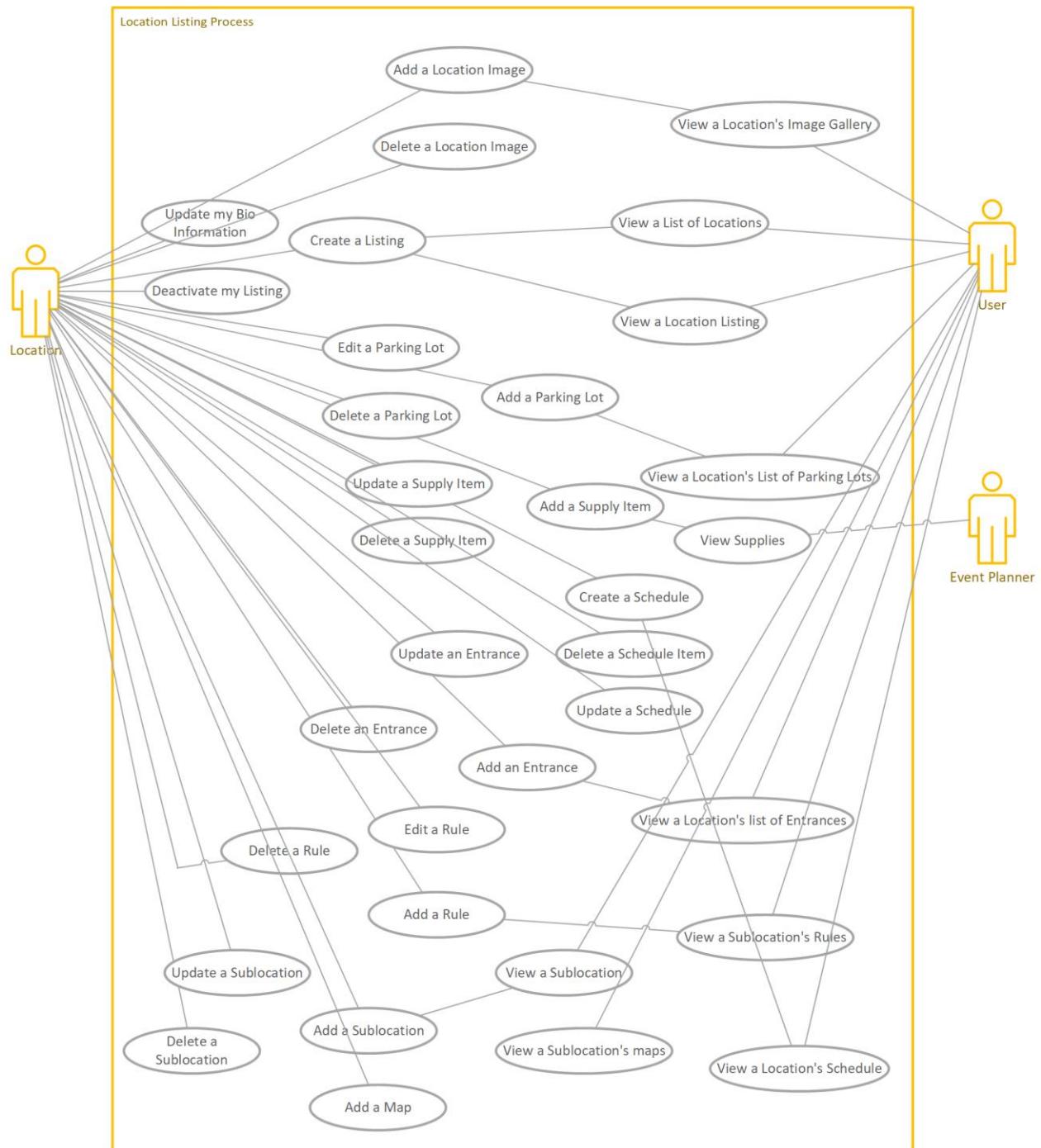
Use Case ID:	3046		
Use Case Name:	Remove Image from Image Gallery		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-12-7	Last Revision Date:	
Actors:	User		
Description:	A user would like to remove an image from their image gallery		
Trigger:	A user presses the "Deactivate" button in a listing for a supplier or location		
Preconditions:	<ol style="list-style-type: none"> 1. Current user has edit permissions for the listing they are viewing 2. Current user is in edit mode for the listing they are viewing 3. User has selected a photo from the gallery 		
Postconditions:	<ol style="list-style-type: none"> 1. Image has been removed from gallery 2. User is returned to bio edit screen 		
Normal Flow:	<ol style="list-style-type: none"> 1. User presses the "Remove Selected Image" button in the main listing edit screen 2. User is queried for confirmation that they desire to remove the image 3. The image is removed from the gallery 4. User is returned to the edit screen for their listing 		
Alternative Flows:	2a: User cancels removing the image 1. Resume flow from step 4.		
Exceptions:			
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:			

3.4.1.2.18 *Become a supplier*

Use Case ID:	3047		
Use Case Name:	Allow user to become a supplier		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2022-04-27	Last Revision Date:	
Actors:	User Administrator		
Description:	A user would like to become a supplier with a listing.		
Trigger:	A user presses the "Become a supplier" button in the supplier index page		
Preconditions:	1. Current user is logged in.		
Postconditions:	1. User has gained the supplier role 2. User's supplier listing has been added to the supplier index		
Normal Flow:	1. User presses the "Become a supplier" button in the supplier index page 2. User fills in basic form for a supplier listing. 3. User's listing is generated and placed into a list of listings awaiting approval 4. Administrator views list of listings awaiting approval 5. Administrator reviews listing and approves the listing 6. Listing is removed from list of listings awaiting review and placed into the supplier listing index 7. User is given the Supplier role and informed via email.		
Alternative Flows:	5a: Administrator disapproves listing 1. Administrator enters reason for disapproval 2. Listing is removed from list of listings awaiting review. 3. Information is emailed to User 4. User makes changes Administrator gives 5. User saves changes 6. Resume from step 3		
Exceptions:			
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:			

3.4.2 Location

3.4.2.1 Location Use Case Model



3.4.2.2 Location Use Cases

3.4.2.2.1 Add Sublocation Map

Use Case ID:	3001	
Use Case Name:	Add sublocation map	
Created By:	Christopher T Repko	Last Updated By:
Date Created:	2021-11-18	Last Revision Date:
Actors:	Location User	
Description:	A user would like to upload a map of a sublocation.	
Trigger:	A user presses the edit button in the SiteMap screen.	
Preconditions:	1. Current user has permission to edit the location 2. Location has a listing	
Postconditions:	1. Map has been added to the location 2. User is returned to SiteMap screen	
Normal Flow:	1. User selects the "edit" button in the SiteMap screen 2. User is sent to the EditSiteMap screen 3. User selects the "Upload Map" option 4. User is prompted to select an image to upload 5. Image is uploaded and saved to database 6. User is returned to SiteMap screen	
Alternative Flows:	4a: In step 4, the user fails to select any image files to upload 1. System prompts user for confirmation that they would not like to upload any files 2. Vendor confirms 3. Flow resumes from step 6 4b: In step 4, the user fails to select any image files to upload 1. System prompts user for confirmation that they would not like to upload any files 2. user denies 3. Use case resumes on step 4 4c: In step 4, the user selects one or more files that are not valid image filetypes 1. System informs user of valid file type extensions for images 2. Use case resumes on step 4	
Exceptions:		
Includes:	N/A	
Frequency of Use:	On demand	
Special Requirements:	N/A	
Assumptions:		
Notes and Issues:	1. How many images can a single listing have? 2. What types of images should be supported?	

3.4.2.2 Add Parking Lot

Use Case ID:	3002		
Use Case Name:	Add parking lot		
Created By:	Christopher T Repko	Last Updated By:	Derrick Nagy
Date Created:	2021-11-18	Last Revision Date:	2022-03-04
Actors:	Location User		
Description:	A user would like to add a parking lot to a location.		
Trigger:	A user presses the add button in the Parking screen.		
Preconditions:	1. Current user has permission to edit the location 2. Location has a listing		
Postconditions:	1. Parking lot has been added to the location 2. User is returned to Parking screen		
Normal Flow:	1. User selects the "add" button in the Parking screen 2. User fills out the new set of "lot name" and "description" fields that pop up 3. Data is saved and uploaded to database 4. User is returned to Parking screen		
Alternative Flows:	2a: In step 2, the user chooses to upload an image 1. System prompts user to select an image to upload 2. User selects an image to upload from the approved file types 3. System saves image to database 4. Flow resumes from step 4		
Exceptions:			
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:	1. How many parking lots can a single listing have? 2. What types of images should be supported?		

3.4.2.2.3 Add Location Rule

Use Case ID:	3003		
Use Case Name:	Add sublocation rule		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-11-18	Last Revision Date:	
Actors:	Location User		
Description:	A user would like to add a rule to a sublocation.		
Trigger:	A user presses the edit button in the SiteRules screen.		
Preconditions:	1. Current user has permission to edit the location 2. Location has a listing		
Postconditions:	1. Rule has been added to the sublocation 2. User is returned to Parking screen		
Normal Flow:	1. User selects the “edit” button in the SiteRules screen 2. User is sent to the EditSiteRules screen 3. User selects the add button at the bottom of the list of sublocation rules 4. User fills in the box that appears 5. User hits the “Confirm” button 6. Data is saved and uploaded to database 7. User is returned to Parking screen		
Alternative Flows:	3a: In step 3, the user wants to change sublocations 1. User hits sublocation drop-down 2. User selects the sublocation they want 3. System updates rule list 4. Flow resumes from step 3		
Exceptions:			
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:	1. How many rules can a single sublocation have?		

3.4.2.2.4 Add Sublocation

Use Case ID:	3005		
Use Case Name:	Add Sublocation		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-11-18	Last Revision Date:	
Actors:	Location User		
Description:	A user would like to add a sublocation to their location		
Trigger:	A user hits the "Add" button in the SiteSublocations screen		
Preconditions:	1. Current user has permission to edit the current location listing 2. Location has a listing		
Postconditions:	1. Sublocation has been added		
Normal Flow:	1. User selects the "Add" button in the SiteSublocations screen 2. A new set of text boxes is displayed along with labels for a new sublocation 3. User fills out new set of sublocation boxes 4. User selects the "Add Area" button at the bottom of the page 5. New sublocation information is captured and saved to database 6. User is shown the updated list of sublocations for the location		
Alternative Flows:	4b: In step 4, the user wants to cancel 1. User selects "cancel" button 2. User is shown		
Exceptions:			
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:			

3.4.2.2.5 Add Supply

Use Case ID:	3006		
Use Case Name:	Add Supply		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-11-18	Last Revision Date:	
Actors:	Location User		
Description:	A user would like to add a supply item to a sublocation.		
Trigger:	A user presses the edit button in the SiteSupplies screen.		
Preconditions:	1. Current user has permission to edit the location 2. Location has a listing		
Postconditions:	1. Rule has been added to the sublocation 2. User is returned to Parking screen		
Normal Flow:	1. User selects the “edit” button in the SiteSupplies screen 2. User is sent to the EditSiteSupplies screen 3. User selects the add button at the bottom of the list of sublocation supplies 4. User fills in the box that appears 5. User hits the “Confirm” button 6. Data is saved and uploaded to database 7. User is returned to SiteSupplies screen		
Alternative Flows:	3a: In step 3, the user wants to change sublocations 1. User hits sublocation drop-down 2. User selects the sublocation they want 3. System updates supply list 4. Flow resumes from step 3		
Exceptions:			
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:	1. How many rules can a single sublocation have?		

3.4.2.2.6 Add Entrance

Use Case ID:	3007		
Use Case Name:	Add Entrance		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-11-18	Last Revision Date:	
Actors:	Location User		
Description:	A user would like to add an entrance to a location.		
Trigger:	A user presses the edit button in the SiteEntrances screen.		
Preconditions:	1. Current user has permission to edit the location 2. Location has a listing		
Postconditions:	1. Entrance has been added to the location 2. User is returned to SiteEntrances screen		
Normal Flow:	1. User selects the “edit” button in the SiteEntrances screen 2. User is sent to the EditSiteEntrances screen 3. User selects the add button at the bottom of the screen 4. User fills out the new set of “entrance name” and “description” fields that pop up. 5. Data is saved and uploaded to database 6. User is returned to SiteEntrances screen		
Alternative Flows:			
Exceptions:			
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:	1. How many entrances can a single listing have?		

3.4.2.2.7 Add Listing

Use Case ID:	3008		
Use Case Name:	Create Listing		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-11-18	Last Revision Date:	
Actors:	User		
Description:	A user would like to create a listing for a supplier or location		
Trigger:	A user presses the "Create" button in a list view for a supplier or location		
Preconditions:	1. Current user has been given permission to create a listing for a supplier or location		
Postconditions:	1. Listing has been added 2. User is deposited on the landing screen for their new listing		
Normal Flow:	1. User selects the "Create" button in a listing view 2. User is sent to the EditBioInformation screen for the type of listing they're currently viewing. 3. User fills out the fields 4. User selects "Confirm" at the bottom of the screen 5. Information is sent to database to be stored 6. User is deposited on the landing screen of their new listing.		
Alternative Flows:			
Exceptions:	4a: User selects "Cancel" 1. User is returned to the list for the type of listing they were attempting to create		
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:			

3.4.2.2.8 Create Schedule

Use Case ID:	3010		
Use Case Name:	Create Listing		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-11-18	Last Revision Date:	
Actors:	User		
Description:	A user would like to create a schedule item for a location or supplier		
Trigger:	A user presses the “add” button in the schedule screen for a supplier or location		
Preconditions:	1. Current user has been given permission to edit the current listing		
Postconditions:	1. Schedule item has been created 2. User is returned to their schedule page on their listing		
Normal Flow:	1. User selects “add” button in the schedule screen for a supplier or location 2. User is giving some boxes to select start and end times, as well as a check box for recurring events 3. User fills out boxes 4. User hits “Confirm” 5. Database stores schedule item 6. User is returned to their listing’s schedule screen		
Alternative Flows:			
Exceptions:	4a: User selects “Cancel” 1. User is returned to their listing’s schedule screen		
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:			

3.4.2.2.9 Deactivate Listing

Use Case ID:	30011		
Use Case Name:	Deactivate Listing		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-11-18	Last Revision Date:	
Actors:	User		
Description:	A user would like to deactivate a listing for a supplier or location		
Trigger:	A user presses the “Deactivate” button in a listing for a supplier or location		
Preconditions:	1. Current user has edit permissions for the listing		
Postconditions:	1. Listing has been deactivated 2. User is deposited on a confirmation screen.		
Normal Flow:	1. User selects the “Deactivate” button in the main listing screen 2. User is queried for confirmation that they desire to deactivate the listing 3. User selects “confirm” 4. System instructs database to mark the entity for the listing as deactivated 5. User is sent to a page confirming that their listing has been deactivated.		
Alternative Flows:	3b: User selects “cancel” 1. User is returned to their listing screen.		
Exceptions:			
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:			

3.4.2.2.10 Delete Parking Lot

Use Case ID:	3012		
Use Case Name:	Delete Parking Lot		
Created By:	Emma Pollock	Last Updated By:	
Date Created:	11-18-2021	Last Revision Date:	
Actors:	Location		
Description:	A Location would no longer like to display a parking lot		
Trigger:	The User clicks the Delete Parking Lot button		
Preconditions:	<ul style="list-style-type: none"> 1. User is viewing their location in edit mode 2. User is on the parking tab 		
Postconditions:	<ul style="list-style-type: none"> 1. The parking lot is no longer in the data base 2. The parking lot list is updated to no longer show the deleted parking lot 		
Normal Flow:	<ul style="list-style-type: none"> 1. User clicks the Delete Parking Lot button 2. System prompts User to Confirm that they would like to permanently delete the parking lot 3. User confirms 4. System successfully deletes the parking lot from the database 5. System displays a success message 6. User is returned to the parking lot tab still in edit mode 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ul style="list-style-type: none"> 3a. In step 3 of the normal flow, the User cancels <ul style="list-style-type: none"> 1. Use case resumes on step 6 of normal flow 		
Exceptions:	<ul style="list-style-type: none"> 4a. In step 4 of the normal flow, the System is unable to delete the parking lot from the database <ul style="list-style-type: none"> 1. System displays an error message 2. Use case resumes on step 6 of normal flow 		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	<ul style="list-style-type: none"> 1. “Delete Parking Lot” button may be a button with a trashcan icon 		

3.4.2.2.11 Delete Rule

Use Case ID:	3013		
Use Case Name:	Delete Rule		
Created By:	Emma Pollock	Last Updated By:	
Date Created:	11-18-2021	Last Revision Date:	
Actors:	Location		
Description:	A Location no longer has a rule and would like to delete it		
Trigger:	The User clicks the Delete Rule button		
Preconditions:	<ol style="list-style-type: none"> 1. User is viewing their location in edit mode 2. User is on the Site Rules tab 3. User has selected an area 		
Postconditions:	<ol style="list-style-type: none"> 1. The rule is no longer in the data base 2. The rule list is updated to no longer show the deleted rule 		
Normal Flow:	<ol style="list-style-type: none"> 1. User clicks the Delete Rule button 2. System prompts User to Confirm that they would like to permanently delete the rule 3. User confirms 4. System successfully deletes the rule from the database 5. System displays a success message 6. User is returned to the rule tab still in edit mode and with the same area selected 		
Alternative Flows:	3a. In step 3 of the normal flow, the User cancels <ol style="list-style-type: none"> 1. Use case resumes on step 6 of normal flow 		
Exceptions:	4a. In step 4 of the normal flow, the System is unable to delete the rule from the database <ol style="list-style-type: none"> 1. System displays an error message 2. Use case resumes on step 6 of normal flow 		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	<ol style="list-style-type: none"> 1. "Delete Rule" button may be a button with a trashcan icon 		

3.4.2.2.12 Delete Sublocation

Use Case ID:	3014		
Use Case Name:	Delete Sublocation		
Created By:	Emma Pollock	Last Updated By:	
Date Created:	11-18-2021	Last Revision Date:	
Actors:	Location		
Description:	A Location has changed the way their location is broken up and no longer has a sublocation, so they would like to delete it.		
Trigger:	The User clicks the Delete Sublocation button		
Preconditions:	<ol style="list-style-type: none"> 1. User is viewing their location in edit mode 2. User is on the Areas tab 		
Postconditions:	<ol style="list-style-type: none"> 1. The Sublocation's active field is marked false 2. The Sublocation no longer shows up in the locations list of sublocations 		
Normal Flow:	<ol style="list-style-type: none"> 1. User clicks the Delete Sublocation button 2. System prompts User to Confirm that they would like to permanently delete the sublocation 3. User confirms 4. System successfully deactivates the Sublocation in the database 5. System displays a success message 6. User is returned to the rule tab still in edit mode 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. In step 3 of the normal flow, the User cancels <ol style="list-style-type: none"> 1. Use case resumes on step 6 of normal flow 		
Exceptions:	4a. In step 4 of the normal flow, the System is unable to deactivate the sublocation in the database <ol style="list-style-type: none"> 1. System displays an error message 2. Use case resumes on step 6 of normal flow 		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	<ol style="list-style-type: none"> 1. "Delete Sublocation" button may be a button with a trashcan icon 		

3.4.2.2.13 Delete Entrance

Use Case ID:	3016		
Use Case Name:	Delete Entrance		
Created By:	Emma Pollock	Last Updated By:	
Date Created:	11-18-2021	Last Revision Date:	
Actors:	Location		
Description:	A Location would no longer like to display an entrance		
Trigger:	The User clicks the Delete Entrance button		
Preconditions:	<ol style="list-style-type: none"> 1. User is viewing their location in edit mode 2. User is on the Entrances tab 		
Postconditions:	<ol style="list-style-type: none"> 1. The entrance is no longer in the data base 2. The entrance list is updated to no longer show the deleted parking lot 		
Normal Flow:	<ol style="list-style-type: none"> 1. User clicks the Delete Entrance button 2. System prompts User to Confirm that they would like to permanently delete the entrance 3. User confirms 4. System successfully deletes the entrance from the database 5. System displays a success message 6. User is returned to the Entrances tab still in edit mode 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none"> 3a. In step 3 of the normal flow, the User cancels <ol style="list-style-type: none"> 1. Use case resumes on step 6 of normal flow 		
Exceptions:	<ol style="list-style-type: none"> 4a. In step 4 of the normal flow, the System is unable to delete the entrances from the database <ol style="list-style-type: none"> 1. System displays an error message 2. Use case resumes on step 6 of normal flow 		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	<ol style="list-style-type: none"> 1. "Delete Entrance" button may be a button with a trashcan icon 		

3.4.2.2.14 Delete Supply Item

Use Case ID:	3015		
Use Case Name:	Delete Supply Item		
Created By:	Emma Pollock	Last Updated By:	
Date Created:	11-18-2021	Last Revision Date:	
Actors:	Location		
Description:	A Location no longer has a supply item and would like to delete it		
Trigger:	The User clicks the Delete Supply Item button		
Preconditions:	<ul style="list-style-type: none"> 1. User is viewing their location in edit mode 2. User is on the Site Supplies tab 3. User has selected an area 		
Postconditions:	<ul style="list-style-type: none"> 1. The supply item is no longer in the data base 2. The supplies list is updated to no longer show the deleted supply item 		
Normal Flow:	<ul style="list-style-type: none"> 1. User clicks the Delete Supply Item button 2. System prompts User to Confirm that they would like to permanently delete the supply item 3. User confirms 4. System successfully deletes the supply item from the database 5. System displays a success message 6. User is returned to the Site Supplies tab still in edit mode and with the same area selected 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ul style="list-style-type: none"> 3a. In step 3 of the normal flow, the User cancels <ul style="list-style-type: none"> 1. Use case resumes on step 6 of normal flow 		
Exceptions:	<ul style="list-style-type: none"> 4a. In step 4 of the normal flow, the System is unable to delete the supply item from the database <ul style="list-style-type: none"> 1. System displays an error message 2. Use case resumes on step 6 of normal flow 		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	<ul style="list-style-type: none"> 1. “Delete Supply Item” button may be a button with a trashcan icon 		

3.4.2.2.15 Delete Item From Schedule

Use Case ID:	3017		
Use Case Name:	Delete Item from Schedule		
Created By:	Emma Pollock	Last Updated By:	
Date Created:	11-18-2021	Last Revision Date:	
Actors:	Supplier Vendor		
Description:	A Supplier or Vendor is no longer available at a certain time and would like to delete the schedule item for that time.		
Trigger:	The User clicks the Delete Schedule Item button		
Preconditions:	<ol style="list-style-type: none"> 1. User is viewing their schedule 2. User has selected a day 3. User has selected a schedule item 		
Postconditions:	<ol style="list-style-type: none"> 1. The schedule item is no longer in the database 2. The User's schedule is updated to no longer show the deleted schedule item 		
Normal Flow:	<ol style="list-style-type: none"> 1. User clicks the Delete Schedule Item button 2. System prompts User to Confirm that they would like to permanently delete the schedule item 3. User confirms 4. System successfully deletes the schedule item from the database 5. System displays a success message 6. User is returned to their schedule with the day still selected 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. In step 3 of the normal flow, the User cancels <ol style="list-style-type: none"> 1. Use case resumes on step 6 of normal flow 		
Exceptions:	4a. In step 4 of the normal flow, the System is unable to delete the schedule item from the database <ol style="list-style-type: none"> 1. System displays an error message 2. Use case resumes on step 6 of normal flow 		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	<ol style="list-style-type: none"> 1. "Delete Schedule Item" button may be a button with a trashcan icon 		

3.4.2.2.16 Edit Parking Lot

Use Case ID:	3018		
Use Case Name:	Edit Parking Lot		
Created By:	Emma Pollock	Last Updated By:	Mike Cahow
Date Created:	11-18-2021	Last Revision Date:	3/17/2022
Actors:	Location		
Description:	A Location would like to change the information about one of their parking lots		
Trigger:	The User changes a value in one of the parking lots text boxes, or changes the parking lots picture		
Preconditions:	<ol style="list-style-type: none"> 1. User is on the Parking tab 2. The user sets edit mode 		
Postconditions:	<ol style="list-style-type: none"> 1. The parking lot has the updated values in the database 2. The parking lot has the updated values in the parking lot list 		
Normal Flow:	<ol style="list-style-type: none"> 1. User changes the value of one of the parking lot's fields 2. User presses the Save button 3. System prompts user to confirm that they would like to change the values 4. User confirms 5. System successfully changes the values in the database 6. System displays a success message 7. User is returned to the Parking tab and edit mode is set to false 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none"> 2a. In step 2 of the normal flow, the User clicks another tab or some other nav instead of pressing confirm <ol style="list-style-type: none"> 1. System prompts user to confirm that they would like to go to another tab and lose their unsaved changes 2. User cancels 3. Use case resumes at step 1 or 2 of normal flow 2b. In step 2 of the normal flow, the User clicks another tab or some other nav instead of pressing confirm <ol style="list-style-type: none"> 1. System prompts user to confirm that they would like to go to another tab and lose their unsaved changes 2. User confirms 3. User is moved to the other page and their changes are discarded 3a. In step 3 of the normal flow, the User cancels <ol style="list-style-type: none"> 1. Use case resumes on step 7 of normal flow 		
Exceptions:	<ol style="list-style-type: none"> 5a. In step 5 of the normal flow, the System is unable to update the parking lot in the database <ol style="list-style-type: none"> 1. System displays an error message 2. Use case resumes on step 1 of normal flow 		
Includes:	N/A		
Frequency of Use:	on demand		

Special Requirements:	N/A
Assumptions:	N/A
Notes and Issues:	N/A

3.4.2.2.17 Edit Rule

Use Case ID:	3019		
Use Case Name:	Edit Rule		
Created By:	Emma Pollock	Last Updated By:	
Date Created:	11-18-2021	Last Revision Date:	
Actors:	Location		
Description:	A Location would like to change the text for one of their rules		
Trigger:	The User changes a value in a rule's text box		
Preconditions:	<ul style="list-style-type: none"> 1. User is viewing their location in edit mode 2. User is on the Site Rules tab 		
Postconditions:	<ul style="list-style-type: none"> 1. The rule has the updated values in the database 2. The rule has the updated values in the rules list 		
Normal Flow:	<ul style="list-style-type: none"> 1. User changes the text of one or more rules 2. User presses the Confirm Changes button 3. System prompts user to confirm that they would like to change the values 4. User confirms 5. System successfully changes the values in the database 6. System displays a success message 7. User is returned to the Site Rules tab still in edit mode 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ul style="list-style-type: none"> 2a. In step 2 of the normal flow, the User clicks another tab or some other nav instead of pressing confirm <ul style="list-style-type: none"> 1. System prompts user to confirm that they would like to go to another tab and lose their unsaved changes 2. User cancels 3. Use case resumes at on step 1 or 2 of normal flow 2b. In step 2 of the normal flow, the User clicks another tab or some other nav instead of pressing confirm <ul style="list-style-type: none"> 1. System prompts user to confirm that they would like to go to another tab and lose their unsaved changes 2. User confirms 3. User is moved to the other page and their changes are discarded 3a. In step 3 of the normal flow, the User cancels <ul style="list-style-type: none"> 1. Use case resumes on step 7 of normal flow 		
Exceptions:	<ul style="list-style-type: none"> 5a. In step 5 of the normal flow, the System is unable to update the parking lot in the database <ul style="list-style-type: none"> 1. System displays an error message 2. Use case resumes on step 1 of normal flow 		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	N/A		
Assumptions:	N/A		

Notes and Issues:	N/A

3.4.2.2.18 Edit Sublocation

Use Case ID:	3021		
Use Case Name:	Edit Sublocation		
Created By:	Emma Pollock	Last Updated By:	
Date Created:	11-18-2021	Last Revision Date:	
Actors:	Location		
Description:	A Location would like to change the information about one of their sublocations		
Trigger:	The User changes a value in one of a sublocations text boxes		
Preconditions:	<ol style="list-style-type: none"> 1. User is viewing their location in edit mode 2. User is on the Areas tab 		
Postconditions:	<ol style="list-style-type: none"> 1. The sublocation has the updated values in the database 2. The sublocation has the updated values in the sublocation list 		
Normal Flow:	<ol style="list-style-type: none"> 1. User changes the value of one or more of at least one of the sublocations' fields 2. User presses the Save button 3. System successfully changes the values in the database 4. System displays a success message 5. User is returned to the Areas tab still in edit mode 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	4a. In step 4 of the normal flow, the User cancels <ol style="list-style-type: none"> 1. Use case resumes on step 7 of normal flow 		
Exceptions:	5a. In step 5 of the normal flow, the System is unable to update the sublocation or sublocations in the database <ol style="list-style-type: none"> 1. System displays an error message 2. Use case resumes on step 1 of normal flow 		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

3.4.2.2.19 Edit Entrance

Use Case ID:	3022		
Use Case Name:	Edit Entrance		
Created By:	Emma Pollock	Last Updated By:	
Date Created:	11-18-2021	Last Revision Date:	
Actors:	Location		
Description:	A Location would like to change the information about one of their entrances		
Trigger:	The User changes a value in one of an entrances text boxes		
Preconditions:	<ol style="list-style-type: none"> 1. User is viewing their location in edit mode 2. User is on the Entrances tab 		
Postconditions:	<ol style="list-style-type: none"> 1. The entrance has the updated values in the database 2. The entrance has the updated values in the Entrances tab 		
Normal Flow:	<ol style="list-style-type: none"> 1. User changes the value of one or more of at least one entrances' fields 2. User presses the Confirm Changes button 3. System prompts user to confirm that they would like to change the values 4. User confirms 5. System successfully changes the values in the database 6. System displays a success message 7. User is returned to the Entrances tab still in edit mode 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<p>2a. In step 2 of the normal flow, the User clicks another tab or some other nav instead of pressing confirm</p> <ol style="list-style-type: none"> 1. System prompts user to confirm that they would like to go to another tab and lose their unsaved changes 2. User cancels 3. Use case resumes at on step 1 or 2 of normal flow <p>2b. In step 2 of the normal flow, the User clicks another tab or some other nav instead of pressing confirm</p> <ol style="list-style-type: none"> 1. System prompts user to confirm that they would like to go to another tab and lose their unsaved changes 2. User confirms 3. User is moved to the other page and their changes are discarded <p>4a. In step 4 of the normal flow, the User cancels</p> <ol style="list-style-type: none"> 1. Use case resumes on step 7 of normal flow 		
Exceptions:	<p>5a. In step 5 of the normal flow, the System is unable to update the entrance or entrances in the database</p> <ol style="list-style-type: none"> 1. System displays an error message 2. Use case resumes on step 1 of normal flow 		
Includes:	N/A		
Frequency of Use:	on demand		

Special Requirements:	N/A
Assumptions:	N/A
Notes and Issues:	N/A

3.4.2.2.20 *Update Supply*

Use Case ID:	3024		
Use Case Name:	Update Supply Item		
Created By:	Kris Howell	Last Updated By:	
Date Created:	12/2/2021	Last Revision Date:	
Actors:	Location		
Description:	A location manager would like to update one of the supplies they have listed.		
Trigger:	User clicks a “Save Changes” button after changing fields in the Edit Sublocation user interface.		
Preconditions:	1. User is in the edit sublocation user interface, editing the supply section.		
Postconditions:	1. The specified supply has been updated on their sublocation listing. 2. Supply has been updated in the database.		
Normal Flow:	1. User changes text values for a supply's name or description. 2. User clicks “Save Changes”. 3. Prompt user to confirm they would like to save all changes. 4. User clicks “Confirm”. 5. System updates the supply in the database. 6. User is returned to the edit sublocation interface.		
Alternative Flows:	4a. In step 4 of the normal flow, user clicks “Cancel”. 1. Return to edit sublocation interface, making no changes.		
Exceptions:	5a. In step 5 of the normal flow, the system fails to communicate with the database. 1. Give user an error message 2. Return to edit sublocation user interface, leaving user's changes as they were.		
Includes:	N/A		
Frequency of Use:	Infrequently		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

3.4.2.2.21 *Update Bio*

Use Case ID:	3025		
Use Case Name:	Update Bio Information		
Created By:	Kris Howell	Last Updated By:	
Date Created:	12/2/2021	Last Revision Date:	
Actors:	Location		
Description:	A location manager would like to update the Bio section of their location listing.		
Trigger:	User clicks a “Save Changes” button after changing fields in the Edit Location user interface.		
Preconditions:	1. User is in the edit location user interface, editing the bio section.		
Postconditions:	1. The location's bio has been updated on their location listing. 2. Bio has been updated in the database.		
Normal Flow:	1. User changes textbox with the location's bio. 2. User clicks “Save Changes”. 3. Prompt user to confirm they would like to save all changes. 4. User clicks “Confirm”. 5. System updates the bio in the database. 6. User is returned to the edit location interface.		
Alternative Flows:	4a. In step 4 of the normal flow, user clicks “Cancel”. 1. Return to edit location interface, making no changes.		
Exceptions:	5a. In step 5 of the normal flow, the system fails to communicate with the database. 1. Give user an error message 2. Return to edit location user interface, leaving user's changes as they were.		
Includes:	N/A		
Frequency of Use:	Infrequently		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

3.4.2.22 *Update Schedule*

Use Case ID:	3027		
Use Case Name:	Update Schedule		
Created By:	Kris Howell	Last Updated By:	
Date Created:	12/3/2021	Last Revision Date:	
Actors:	Supplier Location		
Description:	A supplier would like to update the information in their supplier listing.		
Trigger:	User clicks a “Save Changes” button after adding availability to their schedule calendar.		
Preconditions:	<ol style="list-style-type: none"> 1. User is in the edit schedule calendar interface for their supplier or location listing. 		
Postconditions:	<ol style="list-style-type: none"> 1. Schedule calendar populates marking available days. 2. All corresponding database records have been updated. 		
Normal Flow:	<ol style="list-style-type: none"> 1. User clicks squares on the calendar to select them to be added. 2. User fills time fields for Start Time and End Time 3. User clicks “Add Time to Selected Date(s)” button. 4. Calendar squares populate with Availability markings. 5. System creates corresponding records in the database. 6. Any selected dates are de-selected. 7. User remains in the edit schedule calendar interface. 		
Alternative Flows:	<ol style="list-style-type: none"> 3a. In step 3 of the normal flow, user decides to add or remove dates from the selection. <ol style="list-style-type: none"> 1. Dates can be toggled any time before finalizing by clicking “Add Time to Selected Date(s)” button. 2. Return to step 3 of Normal Flow. 		
Exceptions:	<ol style="list-style-type: none"> 3a. In step 3 of the normal flow, user clicks “Add Time to Selected Date(s)” button, when one or more of the selected dates already have a commitment scheduled within the given times. <ol style="list-style-type: none"> 1. Give user an error message explaining the conflict and asking them to resolve the conflict before saving the availability. 2. Return to step 1 of the normal flow. 5a. In step 5 of the normal flow, the system fails to communicate with the database. <ol style="list-style-type: none"> 1. Give user an error message 2. Return to edit location user interface, leaving user’s changes as they were. 		
Includes:	N/A		
Frequency of Use:	Infrequently		
Special Requirements:	N/A		

Assumptions:	N/A
Notes and Issues:	N/A

3.4.2.2.23 *Browse Locations*

Use Case ID:	3029	
Use Case Name:	Browse Locations	
Created By:	Kris Howell	Last Updated By:
Date Created:	12/2/2021	Last Revision Date:
Actors:	Supplier Event Planner Attendee	
Description:	A user would like to browse a list of locations, filtered by some criteria.	
Trigger:	User clicks a “Apply Filters” button after changing the criteria they would like to filter locations by.	
Preconditions:	1. User is in the Browse Locations tab.	
Postconditions:	1. Table populates with locations that fit the criteria.	
Normal Flow:	1. User changes any of the filter criteria in the navbar. 2. User clicks “Apply Filters”. 3. System updates the table with locations that match the filter criteria.	
Alternative Flows:	4a. In step 3 of the normal flow, no locations can be found that match the criteria. 1. Prompt the user informing them that no locations could be found matching those criteria, recommending loosening their filter settings. 2. Remain in browse locations tab.	
Exceptions:	3a. In step 3 of the normal flow, the system fails to communicate with the database. 1. Give user an error message 2. Remain in browse locations tab.	
Includes:	N/A	
Frequency of Use:	Frequently	
Special Requirements:	Default filter criteria present when first coming to the browse locations tab.	
Assumptions:	N/A	
Notes and Issues:	N/A	

3.4.2.2.24 View Location Listing

Use Case ID:	3031		
Use Case Name:	View Location Listing		
Created By:	Kris Howell	Last Updated By:	
Date Created:	12/2/2021	Last Revision Date:	
Actors:	Supplier Event Planner Attendee		
Description:	A user would like to view a listing of a chosen location.		
Trigger:	User double-clicks on a Location they are interested in from the table in the Browse Locations tab.		
Preconditions:	1. User is in the Browse Locations tab.		
Postconditions:	1. User is navigated to the chosen location listing page.		
Normal Flow:	1. User browses for locations and finds one they are interested in. 2. User double-clicks their chosen location. 3. User is navigated to that location's listing page.		
Alternative Flows:	N/A		
Exceptions:	3a. In step 3 of the normal flow, the system fails to communicate with the database. 1. Give user an error message 2. Remain in browse locations tab.		
Includes:	N/A		
Frequency of Use:	Frequently		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

3.4.2.2.25 View Location Gallery

Use Case ID:	3032		
Use Case Name:	View Location Gallery		
Created By:	Kris Howell	Last Updated By:	
Date Created:	12/2/2021	Last Revision Date:	
Actors:	Supplier Event Planner Attendee		
Description:	A user would like to view a gallery of images of a location.		
Trigger:	User clicks the “View more...” button below the mini-gallery section of the location listing main page.		
Preconditions:	1. User is in the Location's listing page.		
Postconditions:	1. User is navigated to the location listing's gallery page.		
Normal Flow:	1. User clicks “View more...” button. 2. User is navigated to that location's listing page. 3. A grid of images populates with images of the location. 4. User clicks an image to see the full-sized image.		
Alternative Flows:	3a. In step 3 of the normal flow, the location has not uploaded any images. 1. Text is displayed explaining that the location has no images to display.		
Exceptions:	3a. In step 3 of the normal flow, the system fails to communicate with the database. 1. Give user an error message 2. Return to the location's listing page.		
Includes:	N/A		
Frequency of Use:	Frequently		
Special Requirements:	N/A		
Assumptions:	Location has uploaded pictures to fill their gallery.		
Notes and Issues:	N/A		

3.4.2.2.26 View Location Entrances

Use Case ID:	3033	
Use Case Name:	View Location Entrances	
Created By:	Kris Howell	Last Updated By:
Date Created:	12/2/2021	Last Revision Date:
Actors:	Supplier Event Planner Attendee	
Description:	A user would like to view a list of entrances to a location.	
Trigger:	User clicks the “Entrances” button in the navbar of the location listing page.	
Preconditions:	1. User is in the Location's listing page.	
Postconditions:	1. User is navigated to the location listing's entrances page.	
Normal Flow:	1. User clicks “Entrances” button. 2. User is navigated to that location's entrances page. 3. A table is populated with the entrances and their descriptions.	
Alternative Flows:	3a. The location has not specified any entrances. 1. Text appears explaining that the location has not specified any entrances.	
Exceptions:	3a. In step 3 of the normal flow, the system fails to communicate with the database. 1. Give user an error message. 2. Return to the location's listing page.	
Includes:	N/A	
Frequency of Use:	Frequently	
Special Requirements:	N/A	
Assumptions:	Location has entered information on their specified entrances.	
Notes and Issues:	N/A	

3.4.2.2.27 View Parking Lots

Use Case ID:	3034		
Use Case Name:	View Location Parking Lots		
Created By:	Logan Baccam	Last Updated By:	
Date Created:		Last Revision Date:	
Actors:	Supplier, Event Planner, Attendee		
Description:	View a locations parking lots		
Trigger:			
Preconditions:	User must be logged in as a Supplier, event planner or attendee		
Postconditions:			
Normal Flow:	<ol style="list-style-type: none"> 1. A user navigates to a locations page 2. User clicks on a locations list of parking lots 3. User is shown of all the available parking for the location 		
Alternative Flows:	<ol style="list-style-type: none"> 1. User selects an event from listing 2. User is shown the event's details 3. User selects the events parking 4. User is shown where parking is for the location. 		
Exceptions:	N/A		
Includes:	N/A		
Frequency of Use:	Frequently used by attendees wanting to prepare		
Special Requirements:	A location must have a list of parking lots to be viewed.		
Assumptions:	N/A		
Notes and Issues:	N/A		

3.4.2.2.28 View Location Schedule

Use Case ID:	3035		
Use Case Name:	View Location Schedule		
Created By:	Logan Baccam	Last Updated By:	
Date Created:		Last Revision Date:	
Actors:	Attendee, Supplier		
Description:	View a locations schedule		
Trigger:	User clicks a locations schedule		
Preconditions:	User must be logged in as an attendee or supplier		
Postconditions:	User will see a locations schedule		
Normal Flow:	<ol style="list-style-type: none"> 1. User navigates to a locations page 2. User clicks on a locations schedule 3. User can see the dates of when a location has an event scheduled 		
Alternative Flows:	N/A		
Exceptions:	<ol style="list-style-type: none"> 1. An event does not have anything scheduled 		
Includes:	N/A		
Frequency of Use:	Frequently used by Event Planners looking for locations		
Special Requirements:	A location must have dates booked to view		
Assumptions:	N/A		
Notes and Issues:	N/A		

3.4.2.2.29 View Sublocation

Use Case ID:	3037		
Use Case Name:	View sublocation		
Created By:	Logan Baccam	Last Updated By:	
Date Created:		Last Revision Date:	
Actors:	Event Planner, Attendee, Supplier		
Description:	View a sublocation		
Trigger:	User selects a location		
Preconditions:	User must logged in as event planner, supplier, or Attendee		
Postconditions:	User can see details of a sublocation		
Normal Flow:	<ol style="list-style-type: none">1. A user searches for a location they like2. User selects the location3. User is brought to locations page4. User navigates to the locations list of sublocations5. User is shown a locations sublocations		
Alternative Flows:	N/A		
Exceptions:	A location doesn't have sublocations listed		
Includes:	N/A		
Frequency of Use:	By Event planners looking for a place to book for an event.		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

3.4.2.2.30 View Sublocation Maps

Use Case ID:	3038	
Use Case Name:	View Sublocations maps	
Created By:	Logan Baccam	Last Updated By:
Date Created:		Last Revision Date:
Actors:	Supplier, Event planner, Attendee	
Description:	View a sublocations maps	
Trigger:	User navigates to a location's page	
Preconditions:	User must logged in as attendee, supplier, or event planner	
Postconditions:	User is shown a sublocations map	
Normal Flow:	<ol style="list-style-type: none"> 1. User searches for a location they like 2. User selects the location 3. User is brought to locations page 4. User is shown list of sublocations 5. User selects a sublocation 6. User selects the sublocations plot map 7. User is shown the sublocations map 	
Alternative Flows:	<ol style="list-style-type: none"> 1. User navigates to an events details page 2. User selects the events plot map 3. User is shown the plot map of the sublocation for the event 	
Exceptions:	N/A	
Includes:	N/A	
Frequency of Use:	Pretty frequent by people wanting to see what an area looks like.	
Special Requirements:	N/A	
Assumptions:	N/A	
Notes and Issues:	N/A	

3.4.2.2.31 View Sublocation Rules

Use Case ID:	3039		
Use Case Name:	View sublocation rules		
Created By:	Logan Baccam	Last Updated By:	
Date Created:		Last Revision Date:	
Actors:	Attendee, Supplier, Event planner		
Description:	View a sublocations rules		
Trigger:	User selects a sublocation from locations page		
Preconditions:	User must be logged in as a supplier, event planner, or attendee		
Postconditions:	User is shown a Sublocations rules		
Normal Flow:	<ol style="list-style-type: none"> 1. User searches for a location they like 2. User selects the location and is brought to locations page 3. User navigates to the locations sublocation listing 4. User selects a sublocation 5. User is shown the sublocations rules 		
Alternative Flows:	<ol style="list-style-type: none"> 1. User selects an event from event listing 2. User navigates to the details of the event 3. User is shown the sublocations rules. 		
Exceptions:	Sublocation must include rules to view them		
Includes:	N/A		
Frequency of Use:	N/A		
Special Requirements:	N/A		
Assumptions:	Not many people will use this normal flow specifically to view rules.		
Notes and Issues:	N/A		

3.4.2.2.32 View Sublocation Schedule

Use Case ID:	3040		
Use Case Name:	View sublocations schedule		
Created By:	Logan Baccam	Last Updated By:	
Date Created:		Last Revision Date:	
Actors:	Event Planner		
Description:	View a sublocations schedule.		
Trigger:			
Preconditions:	User must be logged in as an event planner.		
Postconditions:	User is shown a sublocations schedule.		
Normal Flow:	<ol style="list-style-type: none">1. An event planner navigates to a locations page.2. User selects the Site Schedule button3. User is shown the locations scheduling4. User selects a sublocation from the locations area list5. User is shown the sublocations schedule		
Alternative Flows:	N/A		
Exceptions:	N/A		
Includes:	N/A		
Frequency of Use:	Frequently used by event planners looking for a location for an event		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

3.4.2.2.33 View Sublocation Supplies

Use Case ID:	3041		
Use Case Name:	View sublocations supplies		
Created By:	Logan Baccam	Last Updated By:	
Date Created:		Last Revision Date:	
Actors:	Event Planner		
Description:	View a sublocations supplies.		
Trigger:			
Preconditions:	User must be logged in as an event planner.		
Postconditions:	User is shown a sublocations schedule.		
Normal Flow:	<ol style="list-style-type: none">1. An event planner navigates to a locations page.2. User selects a sublocation from list3. User is shown the details of a sublocation4. User is shown the supplies included		
Alternative Flows:	N/A		
Exceptions:	N/A		
Includes:	N/A		
Frequency of Use:	Frequently used by event planners looking for a location for an event		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

3.4.2.2.34 Add Image

Use Case ID:	3045		
Use Case Name:	Add Image to Image Gallery		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-12-7	Last Revision Date:	
Actors:	User		
Description:	A user would like to add an image to their image gallery		
Trigger:	A user presses the "Add a new Image" button in a listing for a supplier or location		
Preconditions:	1. Current user has edit permissions for the listing they are viewing 2. Current user is in edit mode for the listing they are viewing		
Postconditions:	1. Image has been added to gallery 2. User is returned to bio edit screen		
Normal Flow:	1. User presses the "Add a new Image" button in the main listing screen edit screen 2. User is queried for an image file they want to upload 3. The selected image is stored to database and added to the page gallery 4. User is returned to the edit screen for their listing		
Alternative Flows:	2a: User selects multiple images 1. Repeat step 3 for each photo 2. Resume flow from step 4 2b: User does not select an image 1. Resume flow from step 4		
Exceptions:			
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:			

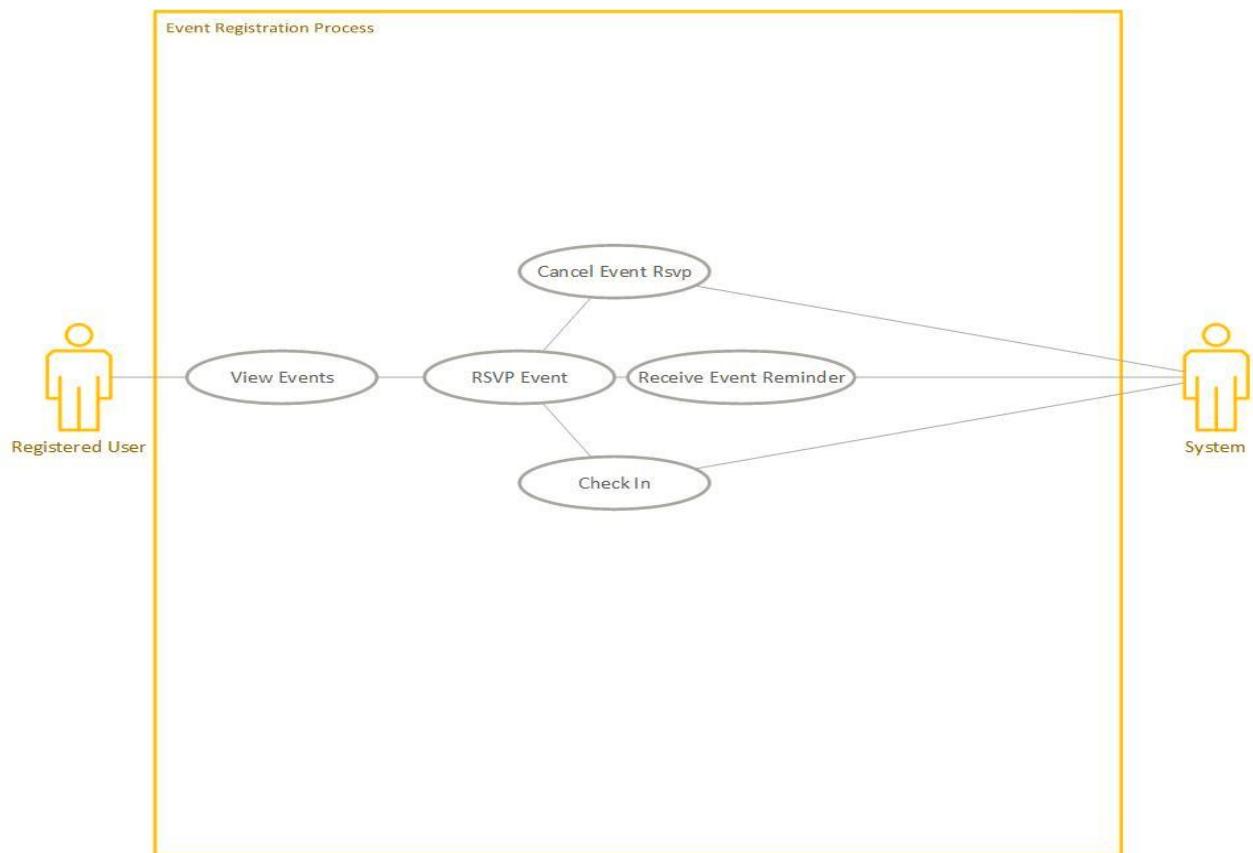
3.4.2.2.35 Remove Image

Use Case ID:	3046		
Use Case Name:	Remove Image from Image Gallery		
Created By:	Christopher T Repko	Last Updated By:	
Date Created:	2021-12-7	Last Revision Date:	
Actors:	User		
Description:	A user would like to remove an image from their image gallery		
Trigger:	A user presses the "Deactivate" button in a listing for a supplier or location		
Preconditions:	<ol style="list-style-type: none"> 1. Current user has edit permissions for the listing they are viewing 2. Current user is in edit mode for the listing they are viewing 3. User has selected a photo from the gallery 		
Postconditions:	<ol style="list-style-type: none"> 1. Image has been removed from gallery 2. User is returned to bio edit screen 		
Normal Flow:	<ol style="list-style-type: none"> 1. User presses the "Remove Selected Image" button in the main listing edit screen 2. User is queried for confirmation that they desire to remove the image 3. The image is removed from the gallery 4. User is returned to the edit screen for their listing 		
Alternative Flows:	2a: User cancels removing the image 1. Resume flow from step 4.		
Exceptions:			
Includes:	N/A		
Frequency of Use:	On demand		
Special Requirements:	N/A		
Assumptions:			
Notes and Issues:			

3.5 Attendee Use Cases

3.5.1 Event Registration

3.5.1.1 Event Registration Process



3.5.1.2 Use Cases

3.5.1.2.1 View/Filter Events

Use Case ID:	4013
Use Case Name:	Filter Events
Created By:	Joey Philipp
Date Created:	9/23/2021
Actors:	Registered User
Description:	A User wishes to Filter Events to find one that interests them
Trigger:	User wishes to Filter Events
Preconditions:	<ol style="list-style-type: none">1. User is Registered to the website2. User is logged in3. User has list of events they want to filter
Postconditions:	<ol style="list-style-type: none">1. User filters event list2. User can now find events that interest them
Normal Flow:	<ol style="list-style-type: none">1. System prompts User to login2. User provides login credentials3. System verifies login parameters4. System verifies login credentials5. System displays User's account6. User navigates to event list7. System displays event list
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none">3a. In step 3 of the normal flow, if the User provides incorrect log in credentials.<ol style="list-style-type: none">1. System will prompt User to try the login again.2. Use Case returns to Step 26.a In step 6 of the normal flow, if the User selects the filter option<ol style="list-style-type: none">1. User filters events list2. System displays filtered list6.b In step 6 of the normal flow, if the User selects the sort option<ol style="list-style-type: none">1. User sorts events list2. System displays sorted list
Exceptions:	<ol style="list-style-type: none">9a. In step 1 of the 6a alternate flow, if the User filters the event list and there are no events for that filter.<ol style="list-style-type: none">1. System will display a message saying there are no events.2. Use Case ends
Includes:	
Frequency of Use:	On Demand
Special Requirements:	N/A
Assumptions:	<p>The user is a User</p> <p>The user has internet access</p> <p>The user understands the language of the website</p>
Notes and Issues:	Is filter and sort the same? filters may include: upcoming event, past event, date range etc

1.2.2 RSVP EVENTS

Use Case ID:	UC-4012		
Use Case Name:	RSVP Event		
Created By:	Joey Philipp	Last Updated By:	
Date Created:	9/23/2021	Last Revision Date:	
Actors:	Registered User		
Description:	A Registered User wishes to RSVP Event.		
Trigger:	Registered User wishes to attend an event.		
Preconditions:	1. Registered User is Registered to the website 2. Registered User is logged in 3. Registered User has event they want to attend		
Postconditions:	1. Registered User RSVP for an event		
Normal Flow:	1. System prompts Registered User to login 2. Registered User provides login credentials 3. System verifies login parameters 4. System verifies login credentials 5. System displays Registered User's account 6. Registered User navigates to event list 7. System displays event list 8. Registered User selects event 9. Registered User RSVPs for event 10. System validates event isn't set in the past 11. System validates event has open spots 12. System sends confirmation message to Registered User		
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. In step 3 of the normal flow, if the Registered User provides incorrect log in credentials. 1. System will prompt Registered User to try the login again. 2. Use Case returns to Step 2		
Exceptions:	1a. In step 1 of the normal flow, if the Registered User for an event with expired registration date. 1. System will display message stating that the event is full 2. System will add user to sign up for waiting list 3. Use case ends 10a. In step 10 of the normal flow if the event signup date is past due. 1. System will display message stating that the signup is closed 2. Use case returns to step 9		
Includes:	The Verified-Visitor writes a review of an Organizer of the event they attended.		
Frequency of Use:	On Demand		
Special Requirements:	N/A		
Assumptions:	The user is a Registered User The user has internet access The user understands the language of the website		
Notes and Issues:			

3.5.1.2.2 Cancel Event Attendance

Use Case ID:	U.C 4009	
Use Case Name:	Cancel Event Attendance	
Created By:	Jace Pettinger	Last Updated By:
Date Created:	09/23/2021	Last Revision Date:
Actors:	Registered Visitors	
Description:	User cancels attending an event	
Trigger:	The cancel option is selected for an event that was marked for attendance	
Preconditions:	<ol style="list-style-type: none"> 1. User already had an event marked as attending 2. User has selected the event 3. User has selected cancel 	
Postconditions:	<ol style="list-style-type: none"> 1. The user is removed from the attendance list 	
Normal Flow:	<ol style="list-style-type: none"> 1. The user selects cancel attendance 2. A message prompt displays asking if they are sure 3. The user selects yes 4. The user is removed from attendance list 5. A successful message prompt is displayed 6. User is returned to previous screen 	
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none"> 2.a User selects cancel <ol style="list-style-type: none"> 1. user is returned to previous screen 2. program exits the use case 	
Exceptions:	<ol style="list-style-type: none"> 4.a An error occurs removing user from attendance list <ol style="list-style-type: none"> 1. An error message s displayed to the user 2. User is returned to previous screen 3. Program exits the use case 	
Includes:	N/A	
Frequency of Use:	Daily	
Special Requirements:	N/A	
Assumptions:	N/A	
Notes and Issues:	<ol style="list-style-type: none"> 1. A message asking why could be displayed if reasons why people are canceling wants to be recorded 	

3.5.1.2.3 Check In

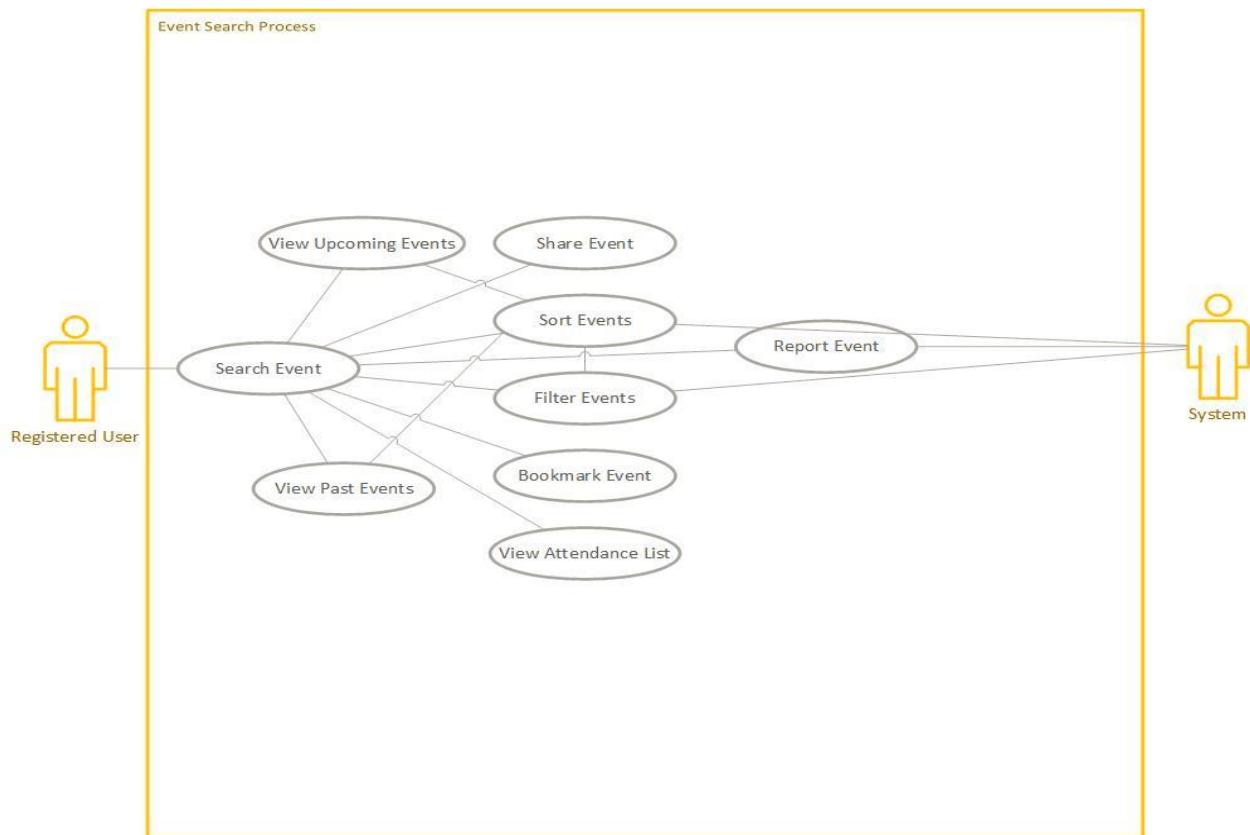
Use Case ID:	4016		
Use Case Name:	Check In		
Created By:	Joey Philipp	Last Updated By:	
Date Created:	9/23/2021	Last Revision Date:	
Actors:	Registered User		
Description:	The Registered User wishes to Check In at an event		
Trigger:	Registered User wishes to Check In		
Preconditions:	1. Registered User is Registered to the website 2. Registered User is logged in 3. Registered User has RSVP'd for an event		
Postconditions:	1. Registered User goes checks into event 2. System sends Event Review form to Registered User 3. Registered User writes a review		
Normal Flow:	1. System prompts Registered User to login 2. Registered User provides login credentials 3. System verifies login parameters 4. System verifies login credentials 5. System displays Registered User's account 6. Registered User arrives event/virtual event 7. System prompts user to check in 8. Registered User Checks In 9. System confirms Registered Users Check In		
Alternative Flows: [Alternative Flow 1 – Not in Network]	6a. In step 6 of the normal flow, if the Registered User doesn't check in. 1. System prompts user to check in 2. Registered User checks in 3. Use Case resumes on Step 9		
Exceptions:	3a. In step 3 of the normal flow, if the Registered User provides incorrect log in credentials. 1. System will prompt Registered User to try the login again. 2. Use Case returns to Step 2		
Includes:			
Frequency of Use:	For Each Event		
Special Requirements:			
Assumptions:	The user is a Registered User The user has internet access The user understands the language of the website		
Notes and Issues:			

3.5.1.2.4 Receive Event Reminder

Use Case ID:	4015		
Use Case Name:	Receive Event Reminder		
Created By:	Joey Philipp	Last Updated By:	
Date Created:	9/23/2021	Last Revision Date:	
Actors:	Registered User		
Description:	A Registered User wishes to Receive Event Reminder.		
Trigger:	Registered User RSVP for event		
Preconditions:	1. Registered User is Registered to the website 2. Registered User is logged in 3. Registered User has RSVP'd for an event		
Postconditions:	1. Registered User receives reminder from System		
Normal Flow:	1. System prompts Registered User to login 2. Registered User provides login credentials 3. System verifies login parameters 4. System verifies login credentials 5. System displays Registered User's account 6. Registered User RSVP an event 7. System asks Registered User if they would like an Event Reminder 8. System adds Registered User to reminder list 9. System sends event reminder		
Alternative Flows: [Alternative Flow 1 – Not in Network]	7a. In step 7 of the normal flow, if the Registered User opts out of Event Reminder 1. Use Case ends		
Exceptions:	3a. In step 3 of the normal flow, if the Registered User provides incorrect log in credentials. 1. System will prompt Registered User to try the login again. 2. Use Case returns to Step 2		
Includes:	RSVP Event		
Frequency of Use:	For Each Event		
Special Requirements:			
Assumptions:	The user is a Registered User The user has internet access The user understands the language of the website		
Notes and Issues:			

3.5.2 Event Search

3.5.2.1 Event Search Process



3.5.2.2 Uses Cases

3.5.2.2.1 Search Events

Use Case ID:	4025		
Use Case Name:	Search Events		
Created By:	Ramiro Pena	Last Updated By:	Derrick Nagy
Date Created:	09/23/2021	Last Revision Date:	02/09/2022
Actors:	Registered User Unregistered User		
Description:	User Searches for Events		
Trigger:	Registered User Navigates to Events page.		
Preconditions:	1. Registered User is Registered to the Website 2. Registered User is Logged In		
Postconditions:	1. Registered User is on Events Page 2. System Returns search results of matching events.		
Normal Flow:	1. Registered User Navigates to Events Page 2. System Displays all upcoming Events 3. User has option to view events and search current list for key words 4. User can filter list to show all events that are upcoming, in the past, or both. 5. User can filter list to show events that are upcoming, in the past, or both where they have a role, such as "Attendee" or manager. 6. System displays events that match query or keywords or filter.		
Alternative Flows: [Alternative Flow 1 – Not a user]	5a. In step 5 of the normal flow if the Registered User is not in the System. 1. User will not have the option to filter the list by their events		
[Alternative Flow 2 – No results found]	2a. In step 2 of the normal flow, if the system cannot find any results matching with user query. 1. System will prompt display an empty page with no results found.		
Exceptions:	2a. In step 2 of the normal flow, if the Registered User provides information that is incorrect to Log in. 1. System will prompt User that they are not registered, or Information provided is incorrect 2. System will prompt User if they wish to Register or Forgot Information? 3. User Accepts and proceeds to Registration or Recover Information. 4. Use Case Resumes on Step 3.		
Includes:	User Registration User Validation Event List		
Frequency of Use:	Very Often		
Special Requirements:	N/A		
Assumptions:	The User Understand the language that the website is being provided in. The User has Access to the Internet. The User Has an Account.		
Notes and Issues:	1. Should a filtration option for categories be included for the search events?		

3.5.2.2.2 Share Event

Use Case ID:	4026		
Use Case Name:	Share Events		
Created By:	Ramiro Pena	Last Updated By:	Ramiro Pena
Date Created:	09/23/2021	Last Revision Date:	09/23/2021
Actors:	Registered User Unregistered User		
Description:	Sharing an Event		
Trigger:	A User is on an event and wishes to share it.		
Preconditions:	1. User is on an event page.		
Postconditions:	1. User has shared an event		
Normal Flow:	1. User is on the website homepage. 2. User navigates to events page(s). 3. System displays events. 4. User sees an event 5. User selects event 6. System displays event page 7. User navigates to share event tab 8. System provides a link to the event page.		
Alternative Flows: [Alternative Flow 1 – Event is no longer available]	7a. In step 7 of the normal flow, if the event is no longer available 1. System will display a message that event is no longer available 2. System will prompt if they wish to share this event anyways. 3. Use Case resumes on step 8 of the normal flow.		
Exceptions:	N/A		
Includes:	Events Pages Hyperlink generator		
Frequency of Use:	Often		
Special Requirements:	N/A		
Assumptions:	The User Understand the language that the website is being provided in. The User has Access to the Internet.		
Notes and Issues:	1. Should we require users to be registered to view or share events?		

3.5.2.2.3 Save Event

Use Case ID:	4010
Use Case Name:	Save Event
Created By:	Jace Pettinger
Date Created:	09/23/2021
Actors:	Registered Visitor
Description:	An event is bookmarked to view later
Trigger:	An event is selected and the save option is selected
Preconditions:	<ol style="list-style-type: none">1. User must be registered for the site2. User must have an event selected
Postconditions:	<ol style="list-style-type: none">1. The event is saved to a list for the user to access later
Normal Flow:	<ol style="list-style-type: none">1. User selects the save option2. The event is added to the users saved events list3. A successful message prompt is displayed4. User is returned to previous screen
Alternative Flows: [Alternative Flow 1 – Not in Network]	N/A
Exceptions:	<ol style="list-style-type: none">2.a An error occurs saving the event to the list<ol style="list-style-type: none">1. The error message is displayed to the user2. The user is returned to the previous screen3. Program exits the use case
Includes:	N/A
Frequency of Use:	on demand
Special Requirements:	N/A
Assumptions:	N/A
Notes and Issues:	<ol style="list-style-type: none">1. Can users have multiple lists to select to save to?

3.5.2.2.4 View Attendance List

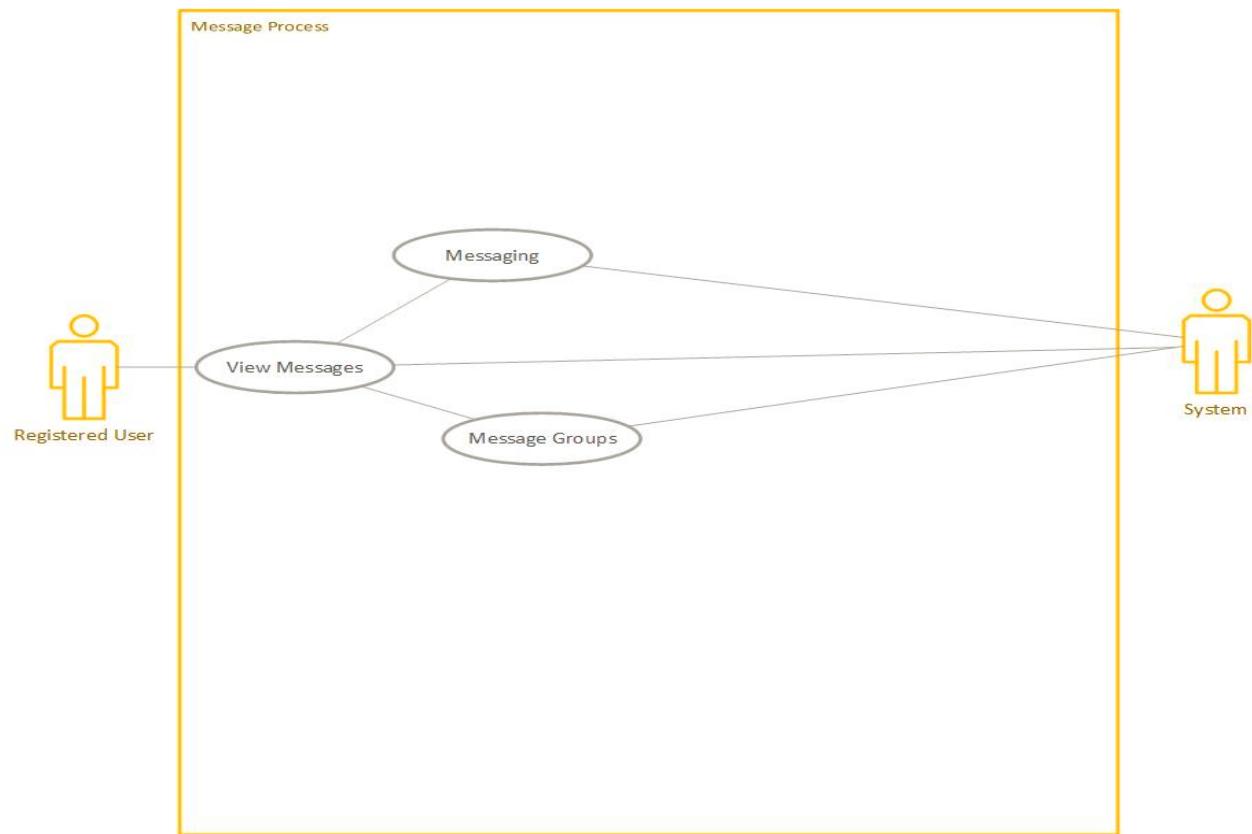
Use Case ID:	4011		
Use Case Name:	View Attendance List		
Created By:	Jace Pettinger	Last Updated By:	
Date Created:	09/23/2021	Last Revision Date:	
Actors:	Registered Visitor		
Description:	View Event Attendance List		
Trigger:	The attending option is selected on an event		
Preconditions:	<ol style="list-style-type: none">1. User has selected an event2. User has selected the view attending option3. User is registered for the site		
Postconditions:	<ol style="list-style-type: none">1. A list of who is attending an event is displayed		
Normal Flow:	<ol style="list-style-type: none">1. User selects view attending option for the event2. A list of who is attending is displayed by user name		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none">2a. No other users are attending<ol style="list-style-type: none">1. A no attendees error message is displayed2. User is returned to the previous screen		
Exceptions:	<ol style="list-style-type: none">1. N/A		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	<ol style="list-style-type: none">1. What is the default ordering for how to display users attending?		

3.5.2.2.5 Report Event

Use Case ID:	4006		
Use Case Name:	Report An event		
Created By:	Jace Pettinger	Last Updated By:	
Date Created:	9/23/2020	Last Revision Date:	
Actors:	Registered Visitor		
Description:	An event will have been reported to a system admin		
Trigger:	The report option is selected by a user		
Preconditions:	<ol style="list-style-type: none"> 1. The user is registered for the site 2. The user has selected an event to report 		
Postconditions:	<ol style="list-style-type: none"> 1. A report of the event has been generated by the user 2. The report is sent to a system admin 		
Normal Flow:	<ol style="list-style-type: none"> 1. User selects report option on an event 2. User is prompted with a menu for reasons why they are reporting 3. User selects a reason 4. User is given an optional field to add further details 5. User fills in descriptive field 6. User hits submit 7. User is given a successful submission prompt 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none"> 2.a <ol style="list-style-type: none"> 1. User selects cancel 2. User is prompted with a confirm cancel prompt 5. User is taken back to the previous screen 4. program exits the use case 4a. User does not fill in descriptive field <ol style="list-style-type: none"> 1. User is given submit button 2. Use case continues at NF step 6 7.a <ol style="list-style-type: none"> 1. User selects cancel 2. User is prompted with a confirm cancel prompt 5. User is taken back to the previous screen 4. program exits the use case 		
Exceptions:	<ol style="list-style-type: none"> 1. An error occurs when generating report on submit 2. User is given error message prompt 3. User is returned to previous screen 		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	<ol style="list-style-type: none"> 1. What menu items for reasons for reporting are there 2. What is the maximum length for the description field 		

3.5.3 Messages

3.5.3.1 Message Process



3.5.3.2 Use Cases

3.5.3.2.1 View Messages

Use Case ID:	4017		
Use Case Name:	View Messages		
Created By:	Joey Philipp	Last Updated By:	
Date Created:	9/23/2021	Last Revision Date:	
Actors:	Registered User		
Description:	The Registered User wishes to view messages		
Trigger:	Registered User receives a message		
Preconditions:	1. Registered User is Registered to the website 2. Registered User is logged in 3. Registered User received a message		
Postconditions:	1. Registered User opens messages		
Normal Flow:	1. System prompts Registered User to login 2. Registered User provides login credentials 3. System verifies login parameters 4. System verifies login credentials 5. System displays Registered User's account 6. Registered User receives a message 7. Registered User clicks on message 8. System displays message		
Alternative Flows: [Alternative Flow 1 – Not in Network]	7a. In step 7 of the normal flow, if the Registered User deletes message 1. System deletes message 2. Use Case ends		
Exceptions:	3a. In step 3 of the normal flow, if the Registered User provides incorrect log in credentials. 1. System will prompt Registered User to try the login again. 2. Use Case returns to Step 2		
Includes:			
Frequency of Use:	On Demand		
Special Requirements:			
Assumptions:	The user is a Registered User The user has internet access The user understands the language of the website		
Notes and Issues:			

3.5.3.2.2 Create Message

Use Case ID:	4023		
Use Case Name:	Create Message		
Created By:	Ramiro Pena	Last Updated By:	Ramiro Pena
Date Created:	09-17-2021	Last Revision Date:	09-17-2021
Actors:	Registered User Unregistered User		
Description:	A Registered User Wishes to Send a Message to another User.		
Trigger:	Registered User Wishes to Create a Message.		
Preconditions:	1. Registered User is Registered to the Website 2. Registered User is Logged In 3. Registered User has user they want to message		
Postconditions:	1. Registered User Creates a message to another Registered User. 2. Registered Users can now exchange messages with a user.		
Normal Flow:	1. Registered User Logs into the Website. 2. Registered User provides Correct Log In information. 3. System accepts Registered User's Correct information. 4. System validates Registered User's Information. 5. System displays website with Registered User's Account. 6. Registered User Navigates to Messages 7. Registered User Creates a Message 8. Registered User Provides Correct Usernames of Users they wish to send the message to. 9. System validates Users that they exist within the system 10. System Creates a Communication channel for users involved in the initial Message. 11. Users Involved are notified of a message channel.		
Alternative Flows: [Alternative Flow 1 – Not in System or Incorrect Information] [Alternative Flow 2 – User does not exist in System to send message to.]	2a. In step 2 of the normal flow if the Registered User is not in the System. 1. System will prompt User that they are not registered, or Information provided is incorrect 2. System will prompt User if they wish to Register or Forgot Information? 3. User Accepts and proceeds to Registration or Recover Information. 4. Use Case Resumes on Step 3. 9a. In step 8 of the normal flow, if the Registered User attempts to send a message but a User does not exist. 1. System will prompt Registered User that User they Included to the message does not exist. 2. System Drafts Message to allow user to correct User to include 3. Use Case resumes on step 10 of normal flow		
Exceptions:	2a. In step 2 of the normal flow, if the Registered User provides information that is incorrect to Log in. 1. System will prompt User that they are not registered, or Information provided is incorrect 2. System will prompt User if they wish to Register or Forgot Information? 3. User Accepts and proceeds to Registration or Recover Information. 4. Use Case Resumes on Step 3.		
Includes:	User Registration		

	User Validation Draft Saving
Frequency of Use:	On demand
Special Requirements:	N/A
Assumptions:	<p>The User Understand the language that the website is being provided in.</p> <p>The User has Access to the Internet.</p> <p>The User Has an Account.</p>
Notes and Issues:	<ol style="list-style-type: none">1. Should a Registered User be allowed to message just anybody?2. Should a Registered User have a minimum and maximum letter count to send a Message?

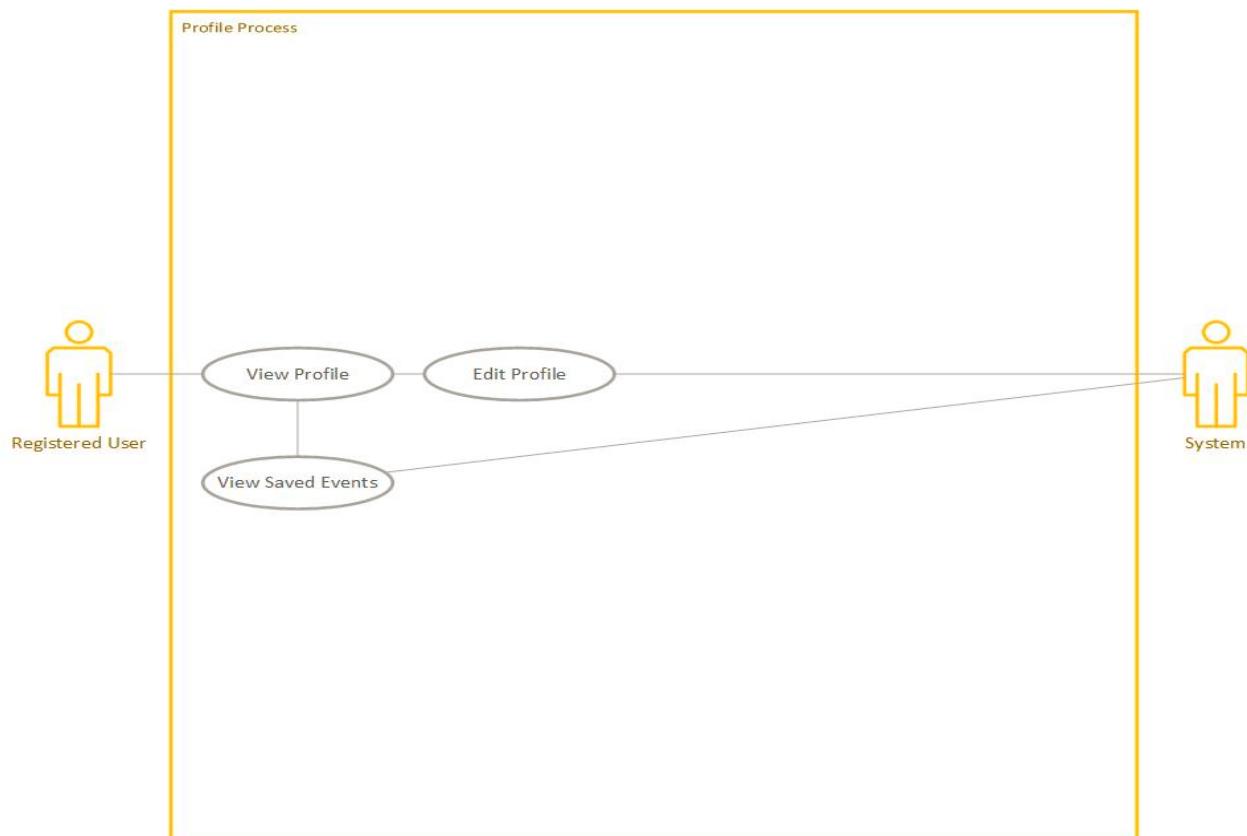
3.5.3.2.3 Message Groups

Use Case ID:	4022		
Use Case Name:	Message Groups		
Created By:	Ramiro Pena	Last Updated By:	Ramiro Pena
Date Created:	09-17-2021	Last Revision Date:	11-19-2021
Actors:	Registered User Unregistered User		
Description:	Can Create a Group Message		
Trigger:	Registered User wishes to create a group message to initiate communication with a group of people.		
Preconditions:	<ol style="list-style-type: none"> 1. Registered User is Registered to the Website 2. Registered User is Logged In 3. Registered User has users they wish to add to a group message 		
Postconditions:	<ol style="list-style-type: none"> 1. Registered User Creates a Group message of other Registered Users 2. Registered Users can now exchange messages with a group. 		
Normal Flow:	<ol style="list-style-type: none"> 1. Registered User Logs into the Website. 2. Registered User provides Correct Log In information. 3. System accepts Registered User's Correct information. 4. System validates Registered User's Information. 5. System displays website with Registered User's Account. 6. Registered User Navigates to Messages 7. Registered User Creates a Message 8. Registered User Provides Correct Usernames of Users they wish to send the message to. 9. System validates Users that they exist within the system 10. System Creates a Group Communication channel for all users involved in the Message. 11. Users Involved are notified of a group message channel. 		
Alternative Flows: [Alternative Flow 1 – Not in System or Incorrect Information]	<ol style="list-style-type: none"> 2a. In step 2 of the normal flow if the Registered User is not in the System. <ol style="list-style-type: none"> 1. System will prompt User that they are not registered, or Information provided is incorrect 2. System will prompt User if they wish to Register or Forgot Information? 3. User Accepts and proceeds to Registration or Recover Information. 4. Use Case Resumes on Step 3. 		
[Alternative Flow 2 – User does not exist in System to send message to.]	<ol style="list-style-type: none"> 9a. In step 8 of the normal flow, if the Registered User attempts to create a Group message but a User does not exist. <ol style="list-style-type: none"> 1. System will prompt Registered User that User(s) they Included to the message does not exist. 2. System Drafts Message to allow user to correct Users to include 3. Use Case resumes on step 10 of normal flow 		
Exceptions:	<ol style="list-style-type: none"> 2a. In step 2 of the normal flow, if the Registered User provides information that is incorrect to Log in. <ol style="list-style-type: none"> 1. System will prompt User that they are not registered, or Information provided is incorrect 2. System will prompt User if they wish to Register or Forgot Information? 3. User Accepts and proceeds to Registration or Recover Information. 4. Use Case Resumes on Step 3. 		
Includes:	User Registration User Validation Draft Saving		

Frequency of Use:	Pretty Often, but not as Often as something like Sending an individual Message would be.
Special Requirements:	N/A
Assumptions:	The User Understand the language that the website is being provided in. The User has Access to the Internet. The User Has an Account.
Notes and Issues:	<ol style="list-style-type: none"> 1. What is the maximum number of Users that a Group Message can Have? 2. Should Users first agree to being included in a group message?

3.5.4 Profile

3.5.4.1 Profile Process



3.5.4.2 Use Cases

3.5.4.2.1 View Profile

Use Case ID:	4028		
Use Case Name:	Registered User		
Created By:	Ramiro Pena	Last Updated By:	Ramiro Pena
Date Created:	11/23/2021	Last Revision Date:	11/23/2021
Actors:	Registered User		
Description:	Registered User wishes to view a profile.		
Trigger:	Registered User want to see a profile page		
Preconditions:	1. Registered User is not on the Profile Page		
Postconditions:	1. Registered User views Profile		
Normal Flow:	1. Registered User is not on a Profile Page 2. Registered User see's another User on the website 3. Registered User selects that User 4. Registered User is on the User's Page 5. Registered User navigates to User Profile 6. Registered User selects User Profile on User Page 7. Registered User views the User Profile		
Alternative Flows: [Alternative Flow 1 – User wants to view own profile]	2a. In step 8 of the normal flow, if the user provides an invalid email address 1. User will select My Profile on Webpage on top of Users web page 2. User opens User own Profile Use Case continues Step 2		
Exceptions:	3a. In step 3 of the normal flow, if the User is no longer active 1. User should not be visible		
Includes:	Homepage View Profile and My Profile HyperLinks		
Frequency of Use:	Often		
Special Requirements:	N/A		
Assumptions:	The User Understand the language that the website is being provided in. The User has Access to the Internet. The User has an Email Account.		
Notes and Issues:	1. Should anyone be able to view anyone's profile?		

3.5.4.2.2 View Saved Events

Use Case ID:	4029		
Use Case Name:	View Saved Events		
Created By:	Ramiro Pena	Last Updated By:	Ramiro Pena
Date Created:	11/23/2021	Last Revision Date:	11/23/2021
Actors:	Registered User		
Description:	A Registered User wishes to view a saved event.		
Trigger:	Registered user wishes to view an event they have saved		
Preconditions:	<ol style="list-style-type: none"> 1. Registered User is on webpage 		
Postconditions:	<ol style="list-style-type: none"> 1. Registered User views saved events list 		
Normal Flow:	<ol style="list-style-type: none"> 1. Registered User is on the webpage 2. Registered User navigates to Events tab 3. System will return regular Events Page 4. Registered User navigates over to the My Events tab 5. System displays choices of My Events 6. Registered User selects My Saved Events HyperLink 7. System returns a List of all Saved Events 		
Alternative Flows: [Alternative Flow 1 – Event is no longer valid]	<ol style="list-style-type: none"> 7a. In step 7 of the normal flow, if the user had an event that was no longer valid <ol style="list-style-type: none"> 1. System will prompt with a message that some events are no longer valid 2. System will not display these invalid Events 		
Exceptions:	<ol style="list-style-type: none"> 7a. In Step 7 of the normal flow if the User had no saved Events <ol style="list-style-type: none"> 1. System will display a message that the User hasn't saved any events 2. System will ask if User wishes to navigate to events to browse 		
Includes:	WebPage Events Page My Events Tab Saved Events Page		
Frequency of Use:	Often		
Special Requirements:	N/A		
Assumptions:	The User Understand the language that the website is being provided in. The User has Access to the Internet. The User has an Email Account.		
Notes and Issues:	<ol style="list-style-type: none"> 1. Should only Registered Users be able to Save Events? 2. Should the system send a message when an event is no longer valid or passed? 		

3.5.4.2.3 Edit Profile

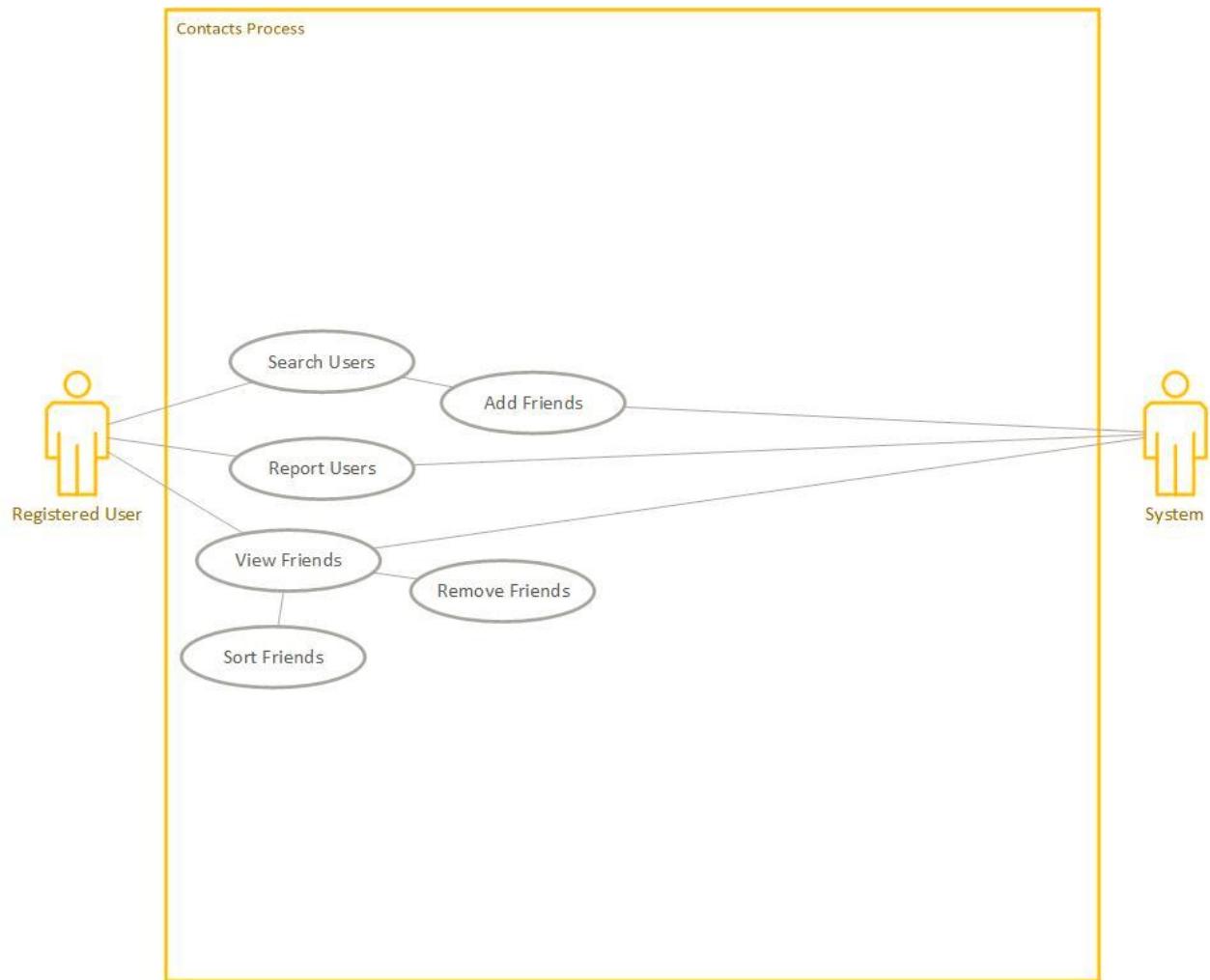
Use Case ID:	UC-4014		
Use Case Name:	Edit Profile		
Created By:	Joey Philipp	Last Updated By:	
Date Created:	9/23/2021	Last Revision Date:	
Actors:	Registered User		
Description:	A Registered User wishes to update their profile.		
Trigger:	Registered User wishes to edit their profile.		
Preconditions:	1. Registered User is Registered to the website 2. Registered User is logged in		
Postconditions:	1. Registered User's profile is updated		
Normal Flow:	1. System prompts Registered User to login 2. Registered User provides login credentials 3. System verifies login parameters 4. System verifies login credentials 5. System displays Registered User's account 6. Registered User chooses edit profile button 7. System displays profile edit page 8. Registered User performs edits 9. Registered User saves edits 10. System validates edits 11. System displays updated profile page		
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. In step 3 of the normal flow, if the Registered User provides incorrect log in credentials. 1. System will prompt Registered User to try the login again. 2. Use Case returns to Step 2		
Exceptions:	10a. In step 10 of the normal flow if the edit isn't valid. 1. System will alert Registered User that the changes aren't valid 2. System will prompt Registered User to try again 3. Use case returns to Step 8		
Includes:			
Frequency of Use:	On Demand		
Special Requirements:	N/A		
Assumptions:	The user is a Registered User The user has internet access The user understands the language of the website		
Notes and Issues:			

3.5.4.2.4 Receive Similar Event Notification

Use Case ID:	4005
Use Case Name:	Receive Similar Event Notification
Created By:	Jace Pettinger
Date Created:	9/17/2021
Actors:	Registered User
Description:	User receives a notification about an event like a previous one they had attended
Trigger:	A new event is created that has keywords that match an event a user has already attended
Preconditions:	<ol style="list-style-type: none">1. User has attended a previous event2. User has the event notifications turned on
Postconditions:	<ol style="list-style-type: none">1. A notification has been received by the user
Normal Flow:	<ol style="list-style-type: none">1. A new event is created with keywords describing the event2. The keywords of the event match keywords for previous events of the user3. A notification is sent to the user about the new event
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none">2a. No keywords match to the user<ol style="list-style-type: none">1. No notification is sent2. The program exits the use case
Exceptions:	<ol style="list-style-type: none">2a. No keywords are found<ol style="list-style-type: none">1. Program exits use case
Includes:	N/A
Frequency of Use:	on demand
Special Requirements:	Events must have keyword identifiers
Assumptions:	N/A
Notes and Issues:	<ol style="list-style-type: none">1. What would the notification be received as? (in app, SMS etc.)2. There needs to be a standardized method of describing events

3.5.5 Contacts

3.5.5.1 Contacts Process



3.5.5.2 Use Cases

3.5.5.2.1 *Search Users*

Use Case ID:	4007	
Use Case Name:	Search Users	
Created By:	Jace Pettinger	Last Updated By:
Date Created:	09/23/2021	Last Revision Date:
Actors:	Registered Visitors	
Description:	Search other users of the site	
Trigger:	User has selected search option	
Preconditions:	1. User is registered for the site 2. Other registered users exist	
Postconditions:	1. A list of other users is shown in an ordered manner	
Normal Flow:	1. User selects search option 2. A list of users is shown by user name	
Alternative Flows: [Alternative Flow 1 – Not in Network]	N/A	
Exceptions:	1a. No other users are registered 1. A no user found message is displayed 2. User is returned to previous screen	
Includes:	N/A	
Frequency of Use:	on demand	
Special Requirements:	N/A	
Assumptions:	N/A	
Notes and Issues:	1. What is the default ordering method for searching users?	

3.5.5.2.2 Report Users

Use Case ID:	4008		
Use Case Name:	Report User		
Created By:	Jace Pettinger	Last Updated By:	
Date Created:	09/23/2021	Last Revision Date:	
Actors:	Registered Visitor		
Description:	A user is reported to the system admin		
Trigger:	The user selects the report option for a user		
Preconditions:	<ol style="list-style-type: none"> 1. User is registered for the site 2. User has another registered user selected 		
Postconditions:	<ol style="list-style-type: none"> 1. A report of the user is generated and sent to the system admin 		
Normal Flow:	<ol style="list-style-type: none"> 1. User selects report option on an event 2. User is prompted with a menu for reasons why they are reporting 3. User selects a reason 4. User is given an optional field to add further details 5. User fills in descriptive field 6. User hits submit 7. User is given a successful submission prompt 8. User is returned to previous screen 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none"> 2.a <ol style="list-style-type: none"> 1. User selects cancel 2. User is prompted with a confirm cancel prompt 5. User is taken back to the previous screen 4. program exits the use case 4a. User does not fill in descriptive field <ol style="list-style-type: none"> 1. User is given submit button 2. Use case continues at NF step 6 6.a <ol style="list-style-type: none"> 1. User selects cancel 2. User is prompted with a confirm cancel prompt 5. User is taken back to the previous screen 4. program exits the use case 		
Exceptions:	<ol style="list-style-type: none"> 1. An error occurs when generating report on submit 2. User is given error message prompt 3. User is returned to previous screen 		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	<ol style="list-style-type: none"> 1. What menu items for reasons for reporting are there 2. What is the maximum length for the description field 		

3.5.5.2.3 Remove Friend

Use Case ID:	4019		
Use Case Name:	Remove Friend		
Created By:	Joey Philipp	Last Updated By:	
Date Created:	9/23/2021	Last Revision Date:	
Actors:	Registered User		
Description:	The Registered User wishes to Remove Friend		
Trigger:	Registered User wishes to Remove Friend		
Preconditions:	1. Registered User is Registered to the website 2. Registered User is logged in		
Postconditions:	1. Registered User removed friend		
Normal Flow:	1. System prompts Registered User to login 2. Registered User provides login credentials 3. System verifies login parameters 4. System verifies login credentials 5. System displays Registered User's account 6. Registered User clicks on friend list 7. System displays friend list 8. Registered User selects friend 9. Registered User deletes selected friend 10. System updates friend list 11. System displays updated friends list		
Alternative Flows: [Alternative Flow 1 – Not in Network]	6a. In step 6 of the normal flow, the Registered User doesn't have any friends. 1. Use Case ends		
Exceptions:	3a. In step 3 of the normal flow, if the Registered User provides incorrect log in credentials. 1. System will prompt Registered User to try the login again. Use Case returns to Step 2		
Includes:			
Frequency of Use:			
Special Requirements:			
Assumptions:	The user is a Registered User The user has internet access The user understands the language of the website		
Notes and Issues:			

3.5.5.2.4 View Friends

Use Case ID:	4021		
Use Case Name:	View Friends		
Created By:	Joey Philipp	Last Updated By:	
Date Created:	9/23/2021	Last Revision Date:	
Actors:	Registered User		
Description:	The Registered User wishes to view friends		
Trigger:	Registered User wishes to view friends		
Preconditions:	1. Registered User is Registered to the website 2. Registered User is logged in		
Postconditions:	1. System displays list of friends		
Normal Flow:	1. System prompts Registered User to login 2. Registered User provides login credentials 3. System verifies login parameters 4. System verifies login credentials 5. System displays Registered User's account 6. Registered User clicks on friends 7. System displays list of Registered Users friends		
Alternative Flows: [Alternative Flow 1 – Not in Network]	6a. In step 6 of the normal flow, the Registered User doesn't have any friends. 1. Use Case ends 7.a In step 7 of normal flow, the User selects sort friends 1. Registered User sorts the friend list 2. System displays sorted friend list		
Exceptions:	3a. In step 3 of the normal flow, if the Registered User provides incorrect log in credentials. 1. System will prompt Registered User to try the login again. 2. Use Case returns to Step 2		
Includes:			
Frequency of Use:			
Special Requirements:			
Assumptions:	The user is a Registered User The user has internet access The user understands the language of the website		
Notes and Issues:			

3.5.5.2.5 Add Friend

Use Case ID:	4021		
Use Case Name:	Add Friend		
Created By:	Joey Philipp	Last Updated By:	
Date Created:	9/23/2021	Last Revision Date:	
Actors:	Registered User		
Description:	The Registered-User wishes to Add Friend		
Trigger:	Registered User wishes to Add Friend		
Preconditions:	1. Registered-User is Registered to the website 2. Registered-User is logged in		
Postconditions:	1. Registered-User adds a friend		
Normal Flow:	1. System prompts Registered-User to login 2. Registered User provides login credentials 3. System verifies login parameters 4. System verifies login credentials 5. System displays Registered-User's account 6. Registered User enters friend name 7. System displays all matches 8. Registered User finds friend 9. Registered User adds friend 10. System displays friend request confirmation		
Alternative Flows: [Alternative Flow 1 – Not in Network]	6a. In step 6 of the normal flow, if the friend's name isn't found 1. System displays no result found 2. Return to Step 5 3. 6b. In step 6 of the normal flow, if the friends account is deactivated 1. System displays deactivated account 2. Return to Step 5		
Exceptions:	3a. In step 3 of the normal flow, if the Registered-User provides incorrect log in credentials. 1. System will prompt Registered-User to try the login again. 2. Use Case returns to Step 2		
Includes:			
Frequency of Use:			
Special Requirements:			
Assumptions:	The user is a Registered-User The user has internet access The user understands the language of the website		
Notes and Issues:			

3.5.6 Reviews

3.5.6.1 Review Process



3.5.6.2 Use Cases

3.5.6.2.1 View Reviews

Use Case ID:	4024		
Use Case Name:	View Reviews		
Created By:	Ramiro Pena	Last Updated By:	Ramiro Pena
Date Created:	09/23/2021	Last Revision Date:	09/23/2021
Actors:	Registered User Unregistered User		
Description:	A user wishes to see reviews of an event		
Trigger:	Registered User opens an event that was held or recurring event.		
Preconditions:	1. Registered User is Registered to the Website 2. Registered User is Logged In		
Postconditions:	1. Registered User is on the Event review page 2. Registered User views reviews.		
Normal Flow:	1. Registered User Logs into the Website. 2. Registered User provides Correct Log In information. 3. System accepts Registered User's Correct information. 4. System validates Registered User's Information. 5. System displays website with Registered User's Account. 6. Registered User navigates to an Event 7. Registered User selects Event 8. System Displays Event Page. 9. Registered User navigates to reviews 10. Registered User Selects Reviews 11. System Displays reviews page 12. Registered User reads Reviews.		
Alternative Flows: [Alternative Flow 1 – Not in System or Incorrect Information]	2a. In step 2 of the normal flow if the Registered User is not in the System. 1. System will prompt User that they are not registered, or Information provided is incorrect 2. System will prompt User if they wish to Register or Forgot Information? 3. User Accepts and proceeds to Registration or Recover Information. 4. Use Case Resumes on Step 3.		
Exceptions:	2a. In step 2 of the normal flow, if the Registered User provides information that is incorrect to Log in. 1. System will prompt User that they are not registered, or Information provided is incorrect 2. System will prompt User if they wish to Register or Forgot Information? 3. User Accepts and proceeds to Registration or Recover Information. 4. Use Case Resumes on Step 3.		
Includes:	User Registration User Validation Event Pages Review Pages		
Frequency of Use:	On Demand		
Special Requirements:	N/A		

Assumptions:	The User Understand the language that the website is being provided in. The User has Access to the Internet. The User Has an Account.
Notes and Issues:	<ol style="list-style-type: none">1. What should be displayed if there are no reviews?2. Should non-recurring events have reviews?3. Extend reviews to user pages?

3.5.6.2.2 Create Attendee Review

Use Case ID:	4001		
Use Case Name:	Create Attendee review		
Created By:	Jace Pettinger	Last Updated By:	Ramiro Pena
Date Created:	9/17/2021	Last Revision Date:	11/19/2021
Actors:	Registered user		
Description:	Create a new review for an attendee		
Trigger:	User selects review option for an attendee		
Preconditions:	<ol style="list-style-type: none"> 1. User is registered for the website 		
Postconditions:	<ol style="list-style-type: none"> 1. A new review has been created for an attendee 		
Normal Flow:	<ol style="list-style-type: none"> 1. User selects review option 2. User enters general rating 3. User enters detailed review description 4. User uploads optional picture for review 5. User hits submit 6. A successful submission page is presented 		
Alternative Flows: [Alternative Flow 1 – Not in Network]	<p>1.a User is not registered for the site</p> <ol style="list-style-type: none"> 1. User is prompted to register 2. User hits accept 3. User is taken to register page <p>1.b User is not registered for the site</p> <ol style="list-style-type: none"> 1. User is prompted to register 2. User hits cancel 3. User is taken back to the previous screen before review option <p>4.a User uploads a non-supported file for a picture</p> <ol style="list-style-type: none"> 1. System Prompts user to try again 2. User selects accept 3. User is taken back to media selection 4. Use case continues at step 4 <p>4.b User uploads a non-supported file for a picture</p> <ol style="list-style-type: none"> 1. System prompts user to try again 2. User selects decline 3. No media is uploaded 4. Use case continues at step 4 		
Exceptions:	User uploads an invalid file type, covered in alternate flow 4.A and B		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	Must have a way to verify registration and verification		
Assumptions:	N/A		
Notes and Issues:	<ol style="list-style-type: none"> 1. What is the maximum length for a review description 2. What is the general rating system we are using 		

3.5.6.2.3 Create Organizer Review

Use Case ID:	4001		
Use Case Name:	Create Attendee review		
Created By:	Jace Pettinger	Last Updated By:	Ramiro Pena
Date Created:	9/17/2021	Last Revision Date:	11/19/2021
Actors:	Registered user		
Description:	Create a new review for an attendee		
Trigger:	User selects review option for an attendee		
Preconditions:	1. User is registered for the website		
Postconditions:	1. A new review has been created for an attendee		
Normal Flow:	1. User selects review option 2. User enters general rating 3. User enters detailed review description 4. User uploads optional picture for review 5. User hits submit 6. A successful submission page is presented		
Alternative Flows: [Alternative Flow 1 – Not in Network]	1.a User is not registered for the site 1. User is prompted to register 2. User hits accept 3. User is taken to register page 1.b User is not registered for the site 1. User is prompted to register 2. User hits cancel 3. User is taken back to the previous screen before review option 4.a User uploads a non-supported file for a picture 1. System Prompts user to try again 2. User selects accept 3. User is taken back to media selection 4. Use case continues at step 4 4.b User uploads a non-supported file for a picture 1. System prompts user to try again 2. User selects decline 3. No media is uploaded 4. Use case continues at step 4		
Exceptions:	User uploads an invalid file type, covered in alternate flow 4.A and B		
Includes:	N/A		
Frequency of Use:	on demand		
Special Requirements:	Must have a way to verify registration and verification		
Assumptions:	N/A		
Notes and Issues:	1. What is the maximum length for a review description 2. What is the general rating system we are using		

3.5.6.2.4 Create Event Review

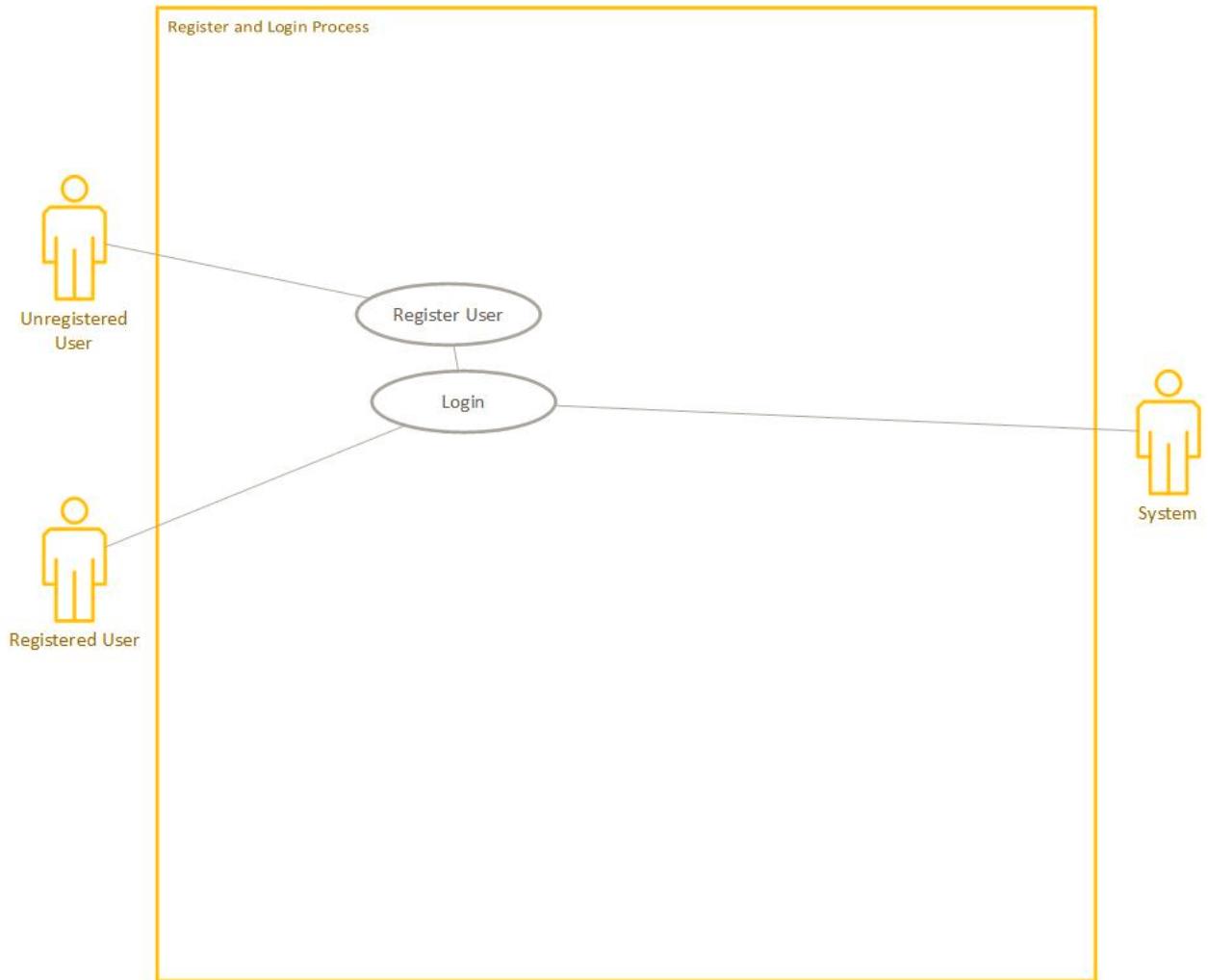
Use Case ID:	4001	
Use Case Name:	Create event review	
Created By:	Jace Pettinger	Last Updated By:
Date Created:	9/17/2021	Last Revision Date:
Actors:	Registered User	
Description:	Create a review for an event	
Trigger:	User selects the review option	
Preconditions:	1. User is registered for the website 2. User is verified to have attended the event	
Postconditions:	1. A new review of the event is created	
Normal Flow:	1. User selects review option 2. User enters the general review score 3. User enters a review description 4. User uploads a picture file 5. User hits submit 6. A successful submitted screen is displayed to the user	
Alternative Flows: [Alternative Flow 1 – Not in Network]	1.a User is not registered for the site 1. User is prompted to register 2. User hits accept 3. User is taken to register page 1.b User is not registered for the site 1. User is prompted to register 2. User hits cancel 3. User is taken back to the previous screen before review option 4.a User uploads a non-supported file for a picture 1. System Prompts user to try again 2. User selects accept 3. User is taken back to media selection 4. Use case continues at step 5 4.b User uploads a non-supported file for a picture 1. System prompts user to try again 2. User selects decline 3. No media is uploaded 4. Use case continues at step 5	
Exceptions:	User uploads incorrect file type, handled in alternate flow 4a and b	
Includes:	N/A	
Frequency of Use:	on demand	
Special Requirements:	Requires verification of user being registered and being verified	
Assumptions:	N/A	
Notes and Issues:	1. Is there a maximum length for description 2. What general rating system are we using	

3.5.6.2.5 Create Vendor Review

Use Case ID:	4004
Use Case Name:	Create Vendor Review
Created By:	Jace Pettinger
Date Created:	9/17/2021
Actors:	Registered User
Description:	Create a new review for a vendor
Trigger:	User selects the review option for a vendor
Preconditions:	<ul style="list-style-type: none"> 1. User is registered for the website 2. User is verified
Postconditions:	A new review of a vendor is created
Normal Flow:	<ol style="list-style-type: none"> 1. User selects review option 2. User enters the general review score 3. User enters a review description 4. User uploads a picture file 5. User hits submit 6. A successful submitted screen is displayed to the user
Alternative Flows: [Alternative Flow 1 – Not in Network]	<ol style="list-style-type: none"> 1.a User is not registered for the site <ol style="list-style-type: none"> 1. User is prompted to register 2. User hits accept 3. User is taken to register page 1.b User is not registered for the site <ol style="list-style-type: none"> 1. User is prompted to register 2. User hits cancel 3. User is taken back to the previous screen before review option 4.a User uploads a non-supported file for a picture <ol style="list-style-type: none"> 1. System Prompts user to try again 2. User selects accept 3. User is taken back to media selection 4. Use case continues at step 5 4.b User uploads a non-supported file for a picture <ol style="list-style-type: none"> 1. System prompts user to try again 2. User selects decline 3. No media is uploaded 4. Use case continues at step 5
Exceptions:	User uploads incorrect file type, handled in alternate flow 4a and b
Includes:	N/A
Frequency of Use:	on demand

3.5.7 Register and Login

3.5.7.1 Register and Login Process



3.5.7.2 Use Cases

3.5.7.2.1 Register User

Use Case ID:	4027		
Use Case Name:	Register User		
Created By:	Ramiro Pena	Last Updated By:	Christopher Repko
Date Created:	09/23/2021	Last Revision Date:	02/03/2022
Actors:	Unregistered User		
Description:	An Unregistered User wishes to create an account.		
Trigger:	Unregistered user wishes to create an account with the website to become a Registered User.		
Preconditions:	<ol style="list-style-type: none">1. Unregistered User selects prompt to register an account		
Postconditions:	<ol style="list-style-type: none">1. Unregistered User becomes a Registered User2. System saves Account within database.		
Normal Flow:	<ol style="list-style-type: none">1. System provides Create Account page2. Unregistered User provides email Address.3. Unregistered User provides full name4. Unregistered User provides a password5. Unregistered User provides a city, state, and zip combination6. Unregistered User selects "create account"7. System validates information provided8. Account is now created.9. System logs user in as detailed in use case 4018, minus user inputs.		
Alternative Flows: [Alternative Flow 1 – Email address does not exist] [Alternative Flow 2 – User fails to add information]	<p>7a. In step 8 of the normal flow, if the user provides an invalid email address</p> <ol style="list-style-type: none">1. System will prompt the user that they have provided an invalid email address.2. System will ask the user to enter a correct email address. <p>Use Case continues from Step 2</p> <p>7b. In step 8 of the normal flow, if the user fails to fill out a field in a valid way</p> <ol style="list-style-type: none">1. System will prompt the user that they have not filled out the field with a valid value.2. System will ask user to fill out fields <p>Use case continues from step 2</p> <p>7c. In step 8 of the normal flow, if the user fails to enter a valid password</p> <ol style="list-style-type: none">3. System will prompt the user as to the requirements for a password to be valid4. System will ask user to fill out fields <p>Use case continues from step 2</p>		
Exceptions:	10a. In step 10 of the normal flow, if the User does not confirm email		

	1. Account is not created.
Includes:	4018 - Log In
Frequency of Use:	Very Often
Special Requirements:	N/A
Assumptions:	The User has Access to the Internet. The User has an Email Account.
Notes and Issues:	<ol style="list-style-type: none">1. Should minors be allowed to create an account?2. Should age be displayed on accounts?3. Should further verification be included?4. How to send emails to people when signing up?

3.5.7.2.2 Log In

Use Case ID:	4018		
Use Case Name:	Log In		
Created By:	Joey Philipp	Last Updated By:	Christopher Repko
Date Created:	9/23/2021	Last Revision Date:	2/3/2022
Actors:	Registered User		
Description:	The Registered User wishes to login		
Trigger:	Registered User wishes to login		
Preconditions:	1. Registered User is Registered to the website 2. Registered User is on the website or application splash screen		
Postconditions:	1. Registered-User is logged In		
Normal Flow:	1. Registered User is on the website or application splash screen 2. Registered User enters login credentials and presses “log in” 3. System validates login credentials 4. System displays Registered User’s profile		
Alternative Flows: [Alternative Flow 1 – Not in Network]	2a. In step 2 of the normal flow, the visitor doesn’t have an account, and is on web. 1. System prompts Unregistered User to create an account. 2. UnRegistered User creates an account as detailed in use case 4027 3. Use Case returns to step 2		
Exceptions:	3a. In step 3 of the normal flow, if the Registered User provides incorrect log in credentials. 1. System will prompt Registered User to try the login again. Use Case returns to Step 2 3b. In step 3 of the normal flow, if the Registered User provides an invalid email address 1. System will prompt the Registered User that they’ve entered an invalid email address 2. Use case returns to step 2		
Includes:	4027 - Register User		
Frequency of Use:			
Special Requirements:			
Assumptions:	The user is a Registered User The user has internet access		
Notes and Issues:			

3.5.7.2.3 Log In

Use Case ID:	4018	
Use Case Name:	Log In	
Created By:	Joey Philipp	Last Updated By:
Date Created:	9/23/2021	Last Revision Date:
Actors:	Registered User	
Description:	The Registered User wishes to login	
Trigger:	Registered User wishes to login	
Preconditions:	1. Registered User is Registered to the website 2. Registered User is on the website	
Postconditions:	1. Registered-User is logged In	
Normal Flow:	1. Registered User is on the website 2. Registered User enters login credentials 3. System validates login credentials 4. System displays Registered User's profile	
Alternative Flows: [Alternative Flow 1 – Not in Network]	2a. In step 2 of the normal flow, the visitor doesn't have an account 1. System prompts UnRegistered User to create an account. 2. UnRegistered User creates an account 3. Use Case returns to step 2	
Exceptions:	3a. In step 3 of the normal flow, if the Registered User provides incorrect log in credentials. 1. System will prompt Registered User to try the login again. Use Case returns to Step 2	
Includes:		
Frequency of Use:		
Special Requirements:		
Assumptions:	The user is a Registered User The user has internet access The user understands the language of the website	
Notes and Issues:		

4 UI Drawings

4.1 *UI Drawings Summary*

The purpose of the section for user interface (UI) drawings is to give a formal visualization to the concepts that are going to be coded in the system. The drawings for the user interface provide a unified structure of how a project should look after its creation. Providing these drawings also help to ensure that all features that are expected are accessible and in the standard position as expected by societal standards.

The drawings created give a detailed view of what is expected to be seen when using a mobile, web, or desktop interface. Creating these views for the multiple different interfaces help to ensure that the system remains recognizable as the same software when used on different platforms. The following drawings are named by the use cases that they are representing, or by the larger grouped area that may incorporate multiple use cases such as the “home screen”.

4.2 Web

4.2.1 Event Web UI Drawings

4.2.1.1 AAR Overview (1002,1039)

AAR Overview

The screenshot shows a web-based application for managing After Action Reviews (AARs). The interface has a blue header bar with navigation links: Home, Find Event, Volunteer, Supplier, Create Event, a star icon, a calendar icon, a bell icon, a speech bubble icon, and a Login button. Below the header, the page title is "AAR Overview" and the sub-section is "After Action Review". The main content is a table with the following columns: Event Component, Event Role, Short Description, and Include in Report (with a checkbox). The table lists various feedback sources and their roles, such as Feedback from volunteers, Event Manager(s), EOR, Event Promoter, Vendors, Location, Event Planner, and Attendees. A "Create Listing" button is located in the top right corner of the table area. At the bottom of the table, there is a "Generate Report" button.

Event Component	Event Role	Short Description	Include in Report
Feedback	Volunteers	Feedback from volunteers	<input type="checkbox"/>
Feedback	Event Manager	Feedback from Event Manager(s)	<input type="checkbox"/>
Feedback	EOR	Feedback from Event Organizational Rep(s)	<input type="checkbox"/>
Feedback	Event Promoter	Feedback from event promoter	<input type="checkbox"/>
Feedback	Vendors	Feedback from event vendors	<input type="checkbox"/>
Feedback	Location	Feedback from the location	<input type="checkbox"/>
Feedback	Event Planner	Feed back from the event planner	<input type="checkbox"/>
Reviews	Attendees	Reviews from attendees	<input type="checkbox"/>
...

Generate Report

Filter

The screenshot shows a 'Filter' interface for an 'After Action Review'. The top navigation bar includes 'Find Event', 'Volunteer', 'Supplier', 'Create Event', a star icon, a calendar icon, a bell icon, a speech bubble icon, and a 'Login' button. The main area is titled 'After Action Review' and contains a table with columns 'Event Component' and 'Event Role'. The table rows are:

Event Component	Event Role		
Feedback	Volunteers		
Feedback	Event Manager		
Feedback	EOR		
Feedback	Event Promoter		
Feedback	Vendors		
Feedback	Location	Feedback from the location	<input type="checkbox"/>
Feedback	Event Planner	Feed back from the event planner	<input type="checkbox"/>
Reviews	Attendees	Reviews from attendees	<input type="checkbox"/>
...

On the right side, there are filter options: 'Sort' dropdown (set to 'Alphabetically'), 'Filter' dropdown (set to 'Etc.'), 'Tags' search bar, 'Date' filters ('Date Before' and 'Date After' with calendar icons), 'Time' checkboxes for AM/PM, and a note 'Add filters as needed'. Below these is a calendar for 'AUG - 2016' with days numbered 1 through 31. A 'Create Listing' button is also present. At the bottom is a 'Generate Report' button.

OverviewSave

The screenshot shows an 'OverviewSave' interface for 'Save AAR'. The top navigation bar is identical to the 'Filter' interface. The main area is titled 'Save AAR' and contains a table with columns 'Volunteers' and 'Results'. The table rows are:

Volunteers	Results
The Wizard of Oz	★★★★★
Jim	★★★★★
Summary	★★★★★

At the bottom are 'Download' and 'Save Configuration' buttons.

4.2.1.2 Activity Results (1029, 1030)

Activity Results

The screenshot shows a web application interface for managing activity results. The top navigation bar includes links for Home, Find Event, Volunteer, Supplier, Create Event, a star icon, a calendar icon, a bell icon, a message icon, and a Login button. Below the navigation is a search bar with fields for Event Title, Sort, Filter, and Search, along with a clear button.

The main content area displays a grid of event results cards. Each card has a 'Picture' placeholder at the top, followed by the event title, a description, and a table of results. A 'Share in messages/forum' button is located in the top right corner of the first card.

Table	With Results
1	Victor Fleming
2	Carol Reed
3	Orson Welles

Sack Race
A 50m race in a sack.

Rank	Name
1st	Scooter McGee
2nd	TooHotTooHandle
3rd	Hustle Horse
Honorable Mention	One Legged Willy

Sack Race
A 50m race in a sack.

Rank	Name
1st	Scooter McGee
2nd	TooHotTooHandle
3rd	Hustle Horse
Honorable Mention	One Legged Willy

Table	With Results
1	Victor Fleming
2	Carol Reed

Activity Results Add Result

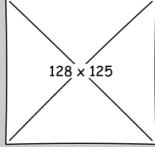
Event Title > Activity Results > Add Results

<Event Title> Add Results

Rank	Name
1st	Scooter McGee
2nd	Blue Team
3rd	
+ Add Another Row	

Add Picture

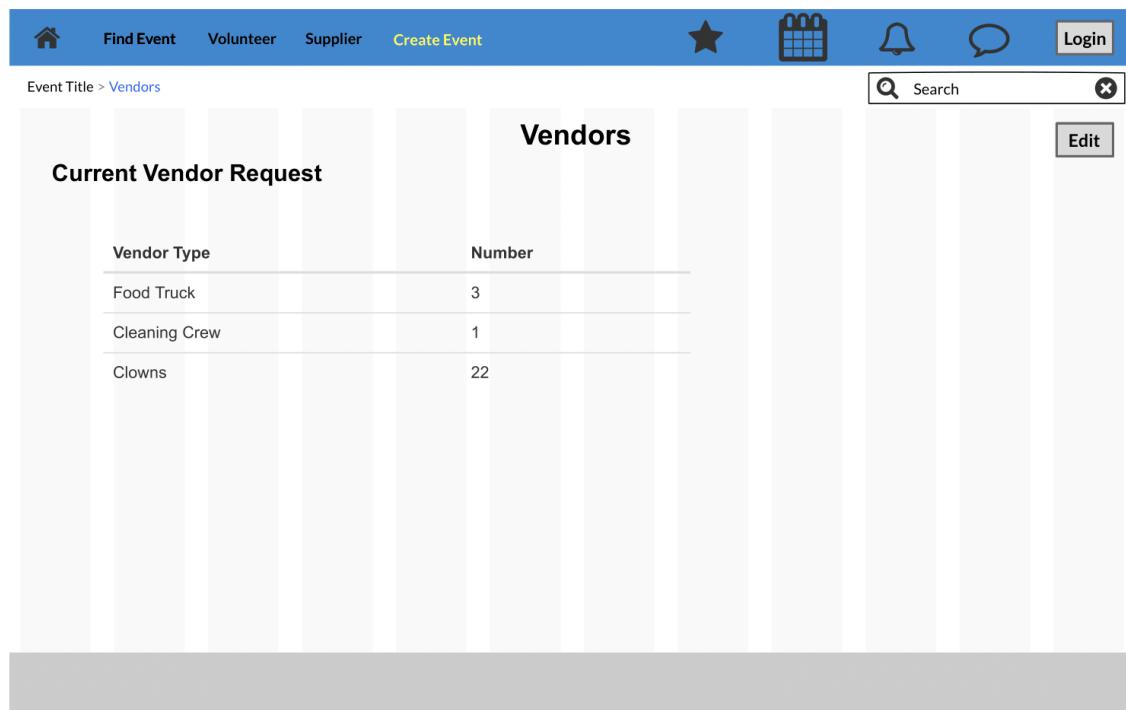
Add a picture? Yes


128 x 125

Upload Picture

4.2.1.3 Add Vendor Tags (1014)

Retrieve Single Listing View No Options



The screenshot shows a software interface with a blue header bar. The header contains icons for a home page, finding an event, volunteering, supplier management, and creating an event. It also includes a star icon, a calendar icon, a bell icon, a message icon, and a 'Login' button. Below the header, a breadcrumb navigation shows 'Event Title > Vendors'. A search bar with a magnifying glass icon and a clear button is present. The main content area is titled 'Vendors' and contains a sub-section titled 'Current Vendor Request'. A table lists vendor types and their counts:

Vendor Type	Number
Food Truck	3
Cleaning Crew	1
Clowns	22

An 'Edit' button is located in the top right corner of the main content area.

Retrieve Single Listing View No Options (1)

Event Title > Vendors > [Edit Vendors](#)

Vendors

Request Vendors

+ Add Vendor Request

Vender Type

Food Truck

Farmer's Stand

Number? 1

Add another vendor type

Next Cancel

Notification view (1)

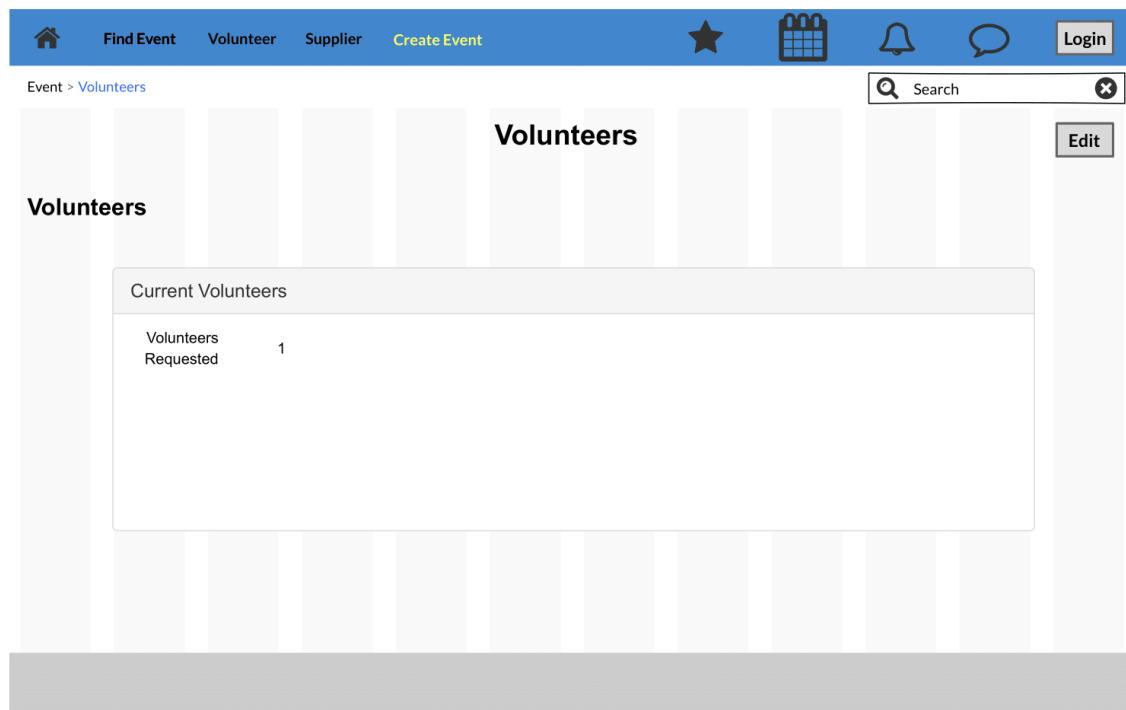
The screenshot shows a user interface for a notification system. At the top, there is a blue header bar with the following elements from left to right: a house icon, 'Find Event', 'Volunteer', 'Supplier', 'Create Event', a star icon, a calendar icon, a bell icon with a '2' notification, a speech bubble icon, and a 'Login' button. Below the header, the URL 'Nav1 > Nav2 > Nav3' is displayed. To the right of the URL are 'Sort' and 'Filter' dropdown menus. The main content area contains a list of notifications:

- <Event Title> - Vendor Req. Edit
- Unseen Notification - Description...
- Seen Notification - Description...
- Seen Notification - Description...
- Seen Notification - Description...

At the bottom of the list is a small downward-pointing arrow icon.

4.2.1.4 Add Volunteer (1015)

Retrieve Single Listing View No Options



The screenshot shows a web application interface for managing volunteers. At the top, there is a blue header bar with icons for a house (Home), a magnifying glass (Search), a star (Favorites), a calendar (Events), a bell (Notifications), a speech bubble (Messages), and a 'Login' button. Below the header, a breadcrumb navigation shows 'Event > Volunteers'. The main content area has a title 'Volunteers' and a sub-section title 'Current Volunteers'. A table displays one item: 'Volunteers Requested' with the value '1'. There is also an 'Edit' button in the top right corner of the main content area.

Volunteers Requested	1
----------------------	---

Retrieve Single Listing View No Options (1)

Event > Volunteers > Request Volunteers

Volunteers

Request Volunteers

Yes, I need volunteers

Add Volunteers

How many?

I want to give more information about the type of volunteer I need

Next **Cancel**

Retrieve Single Listing View No Options (1) (1)

Event > Volunteers > Request Volunteers

Volunteers

Request Volunteers

Yes, I need volunteers

Add Volunteer

How many?

I w

Edit Volunteers

! Are you sure you would like to edit the amount of volunteers you want to request?

Change **Cancel**

Notification view (1)

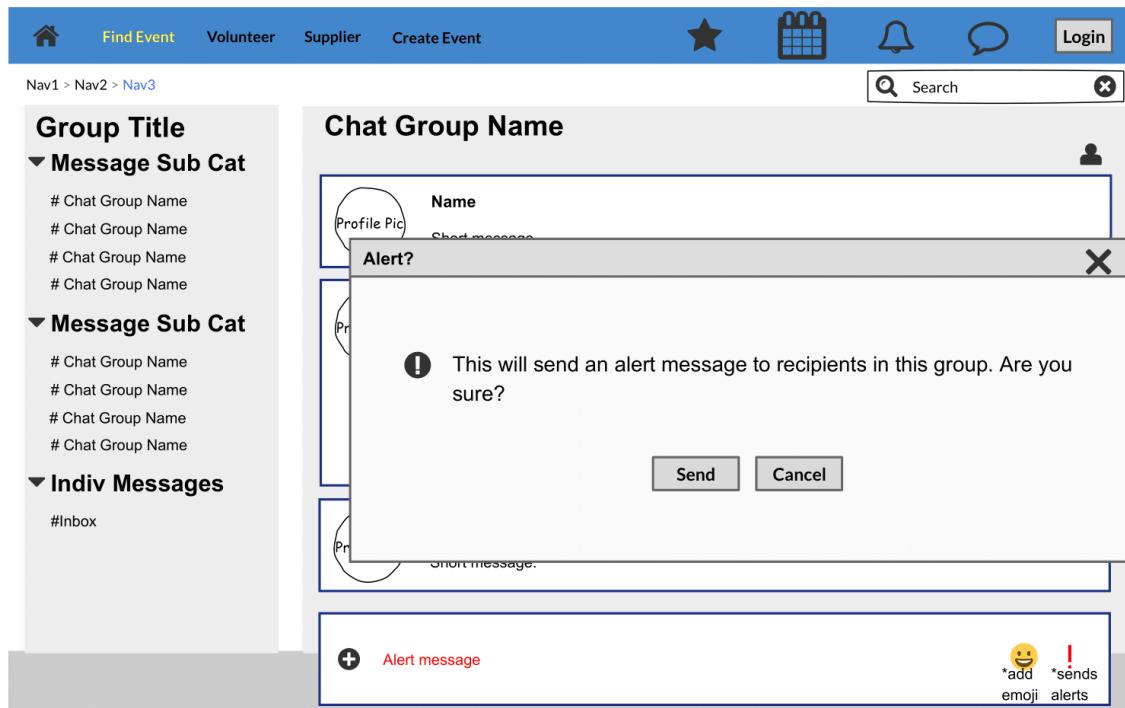
The screenshot shows a user interface for a notification system. At the top, there is a blue header bar with the following elements from left to right: a house icon, 'Find Event', 'Volunteer', 'Supplier', and 'Create Event'. To the right of these are icons for a star, a calendar, a bell (with a '2' notification), a speech bubble, and 'Login'. Below the header, the URL 'Nav1 > Nav2 > Nav3' is displayed. To the right of the URL are 'Sort' and 'Filter' dropdown menus. The main content area is divided into two sections. On the left, there is a large, empty white space with a 'Heading level 1' label at the top. On the right, there is a sidebar containing a list of notifications. The notifications are as follows:

- <Event Title> - Volunteers Changed
- Unseen Notification - Description...
- Seen Notification - Description...
- Seen Notification - Description...
- Seen Notification - Description...

At the bottom right of the sidebar, there is a downward-pointing arrow icon.

4.2.1.5 Alerts (1003)

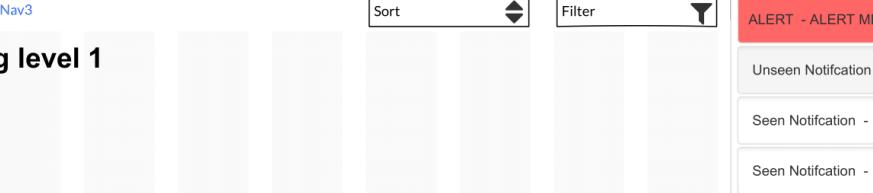
Messages Alert Modal



Messages

The screenshot shows a web-based messaging application. At the top, a blue header bar contains a house icon, 'Find Event', 'Volunteer', 'Supplier', and 'Create Event' buttons. To the right are a star icon, a calendar icon, a bell icon, a speech bubble icon, and a 'Login' button. Below the header is a breadcrumb navigation: 'Nav1 > Nav2 > Nav3'. The main content area is titled 'Chat Group Name' and shows a list of messages. Each message item includes a profile picture placeholder ('Profile Pic'), the recipient's name, and the message content. The first message is a short generic message. The second message is a long Latin text. The third message is another short generic message. The fourth message is an 'Alert message' indicated by a red plus sign icon and the text 'Alert message'. To the right of the alert message are two small icons: a smiley face with an exclamation mark and a red exclamation mark, with the text '*add' and '*sends' respectively. Below these are the words 'emoji' and 'alerts'. On the far left, a sidebar displays a 'Group Title' section with four collapsed 'Message Sub Cat' items, each labeled '# Chat Group Name'. Below this is a 'Message Sub Cat' section with four collapsed items, and finally an 'Indiv Messages' section with a single item labeled '#Inbox'.

Notification view (1)



Nav1 > Nav2 > Nav3

Sort

Filter

ALERT - ALERT MESSAGE ...

Unseen Notification - Description...

Seen Notification - Description...

Seen Notification - Description...

Seen Notification - Description...

▼

4.2.1.6 Approval (1006, 1047)

Suggestions Submition (2)

The screenshot shows a web application interface for 'Submit Suggestions'. The top navigation bar includes links for 'Find Event', 'Volunteer', 'Supplier', and 'Create Event'. There are also icons for a star, calendar, bell, and message, along with a 'Login' button. A search bar with a magnifying glass icon and a clear button is also present.

The main content area is titled 'Submit Suggestions' and contains a section for 'Event Details'. It includes a table with columns for 'Event Detail', 'Content', 'Comment', 'Request Approval', and 'Approval'. The 'Content' column contains several text entries with small edit icons:

- Title: Description
Comment: Can we rename it?
- Date: 08/01/2016
Comment: Comment
- Location: Name, Address, City, State, Zip
Comment: I th

Below this is a section for 'Budget Category Total' and 'Total Budget'. It includes a table with columns for 'Category' and 'Total Budget'. The 'Category' column contains a text entry with a 'USER:' label and a note: 'Os qui ne nitidis suis mutastis onerosior gentes foret erectos.'

At the bottom, there is a section for 'Vendor Type' and 'Number'. It includes a table with columns for 'Category' and 'Total'. The 'Category' column contains a text entry with a note: 'How about getting some clowns?'.

For each row, there are 'Request Approval' and 'Approval' sections. The 'Request Approval' section contains a checkbox and radio buttons for 'Y' and 'N'. The 'Approval' section also contains radio buttons for 'Y' and 'N'.

Suggestions Submition

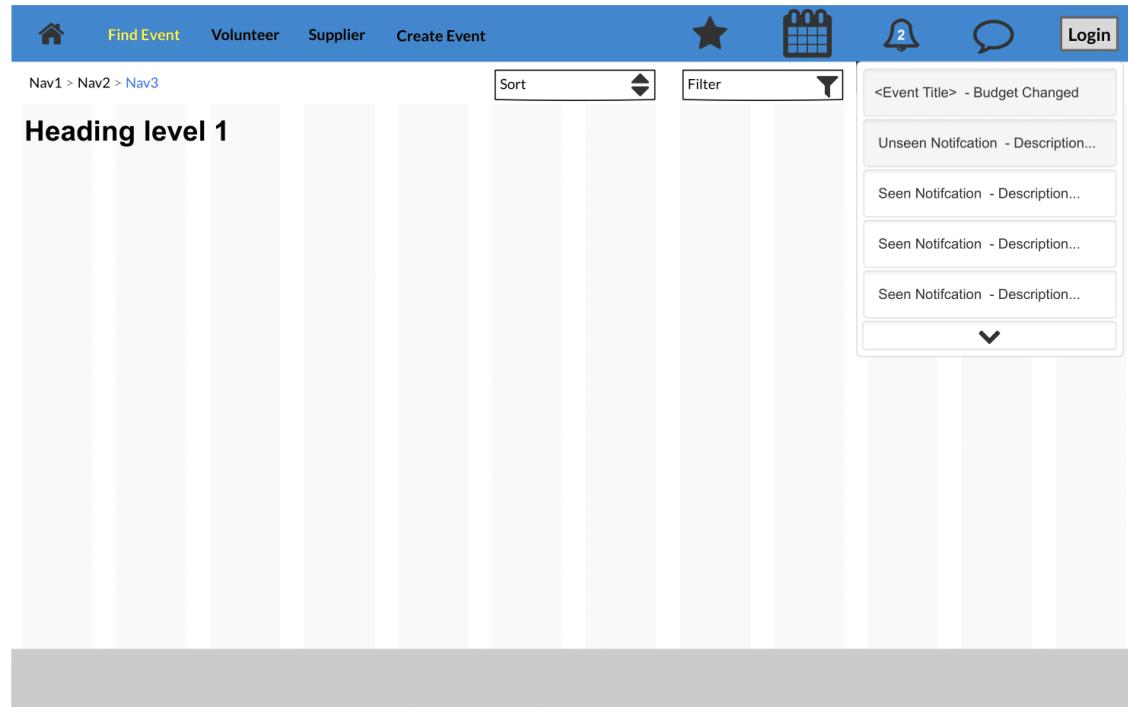
Event Title > Submit Suggestions

Submit Suggestions

Event Detail	Content	Comment	Request Approval	Approval	
Title	Description	Can we rename it? 	<input type="checkbox"/> Requested	<input type="radio"/> Y <input type="radio"/> N	
Date	08/01/2016 		<input type="checkbox"/> Requested	<input type="radio"/> Y <input type="radio"/> N	
Location	Name, Address, City, State, Zip	Ab ambitae, meis pinus sidera... 	<input type="checkbox"/> Requested	<input type="radio"/> Y <input type="radio"/> N	
Budget Category Total	Total Budget	Total Actual	Comment	Request Approval	Approval
Category	Total Budget	Total Actual	Too much 	<input type="checkbox"/> Requested	<input type="radio"/> Y <input type="radio"/> N
Vendor Type	Number	Comment	Request Approval	Approval	
Category	Total	How about getting some clowns? 	<input type="checkbox"/> Requested	<input type="radio"/> Y <input type="radio"/> N	

4.2.1.7 Budget(1021, 1032, 1033, 1034)

Notification view (1)



The screenshot shows a user interface for managing notifications. The top navigation bar includes links for Home, Find Event, Volunteer, Supplier, and Create Event, along with icons for a star, calendar, bell (with a '2' notification), message bubble, and Login. Below the navigation is a breadcrumb trail: Nav1 > Nav2 > Nav3. On the left, there is a large, empty white area with the heading "Heading level 1". On the right, a sidebar displays a list of notifications:

- <Event Title> - Budget Changed
- Unseen Notification - Description...
- Seen Notification - Description...
- Seen Notification - Description...
- Seen Notification - Description...

A small downward arrow icon is located at the bottom right of the notification list.

Edit Budget

The screenshot shows a web-based budgeting application interface. At the top, a blue header bar contains navigation links: Home, Find Event, Volunteer, Supplier, Create Event, a star icon, a calendar icon, a bell icon, a speech bubble icon, and a Login button. Below the header, a breadcrumb trail reads "Event Title > Budget > Edit Budget". A search bar with a magnifying glass icon and a clear button is also present.

The main content area is titled "Budget" and features a "Create a Budget" section. It includes a note "No Current Budget" with an unchecked checkbox. A "Planned" input field is followed by a "Remaining : responsive" dropdown with a collapse/expand arrow. A table for adding budget categories is shown, with a single row for "Expense" having a budget of \$0.00 and an actual amount of \$0.00. A "Delete Category" button is located in the bottom right of this table. A checkbox for "Add another category" is also present.

Below this is a "Category Totals" section with a table showing a single row for "Category Title" with a planned amount of \$0.00 and a spent amount of \$0.00. A "Delete Category" button is also present here. The bottom of the page contains "Change" and "Cancel" buttons.

View Budget

The screenshot shows a web-based budget management application. At the top, there is a blue header bar with icons for Home, Find Event, Volunteer, Supplier, Create Event, a star, a calendar, a bell, a message bubble, and a Login button. Below the header, the URL 'Event Title > Budget' is visible. The main content area has a title 'Budget' and a sub-section title 'View the Budget'. The interface is divided into three main sections: a summary table, and two detailed budget tables for 'Category Title' and 'Category Name'.

Summary

Category	Planned	Spent
Category Title	\$0.00	\$0.00
Total	\$0.00	\$0.00

Category Title (Planned: \$0.00, Remaining: responsive)

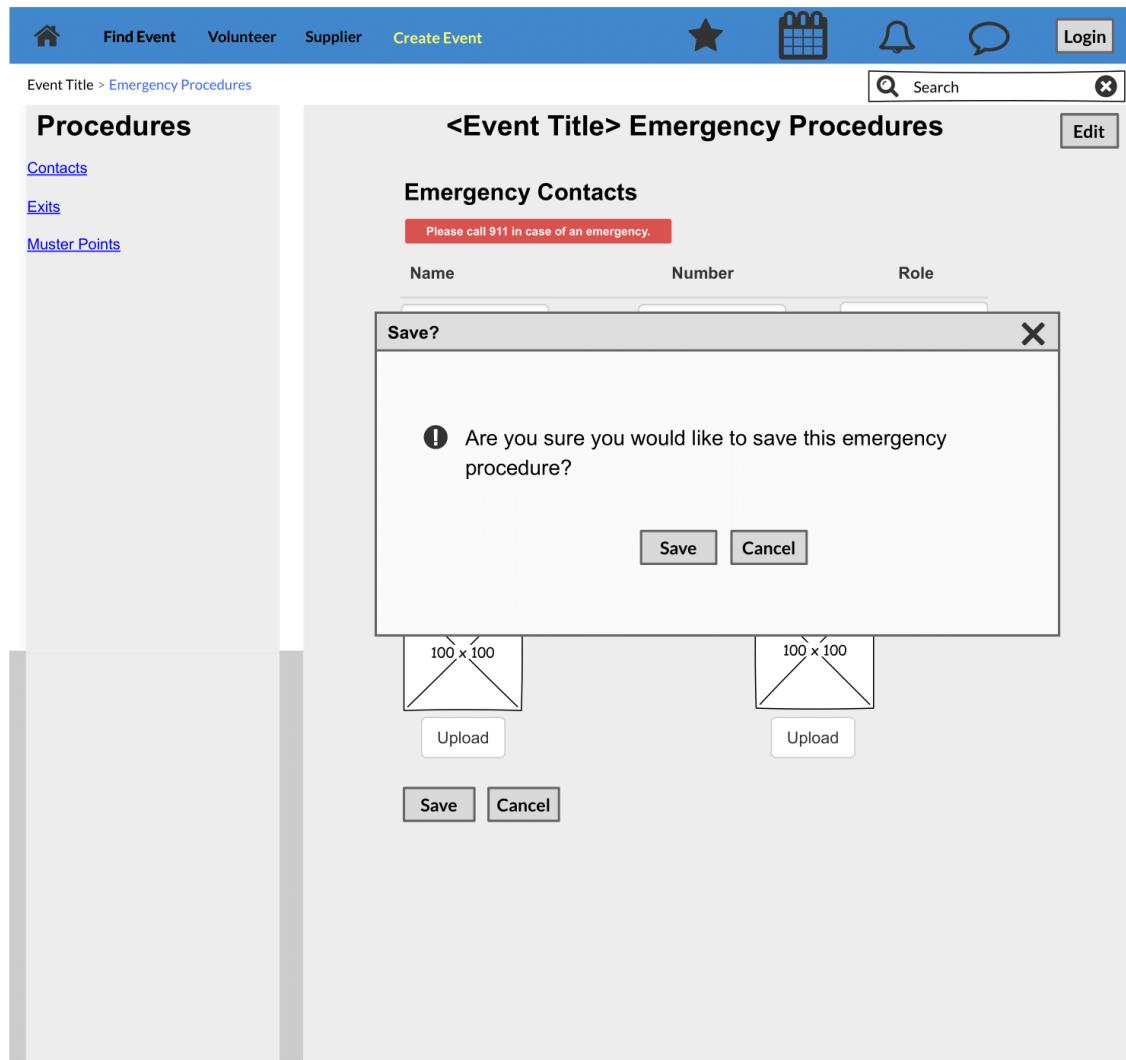
Category Name	Total Budget
Line Item Title	\$0.00
...	...
Event Total Budget	\$0.00

Category Name (Planned: \$0.00, Remaining: responsive)

Category Name	Total Budget
Line Item Title	\$0.00
...	...
Event Total Budget	\$0.00

4.2.1.8 Emergency Procedures(1004, 1035, 1036)

Emergency Procedures Creation Warning



Emergency Procedures Creation (1)

Event Title > Emergency Procedures

Procedures

[Contacts](#)
[Exits](#)
[Muster Points](#)

<Event Title> Emergency Procedures

Emergency Contacts

Please call 911 in case of an emergency.

Emergency Contact Name	Emergency Contact Number
Event Manager Name	319-123-4567
<input type="text" value="Name to edit"/>	<input type="text" value="Number"/>

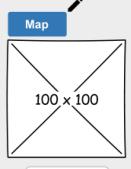
Building Exits

Automatically loaded from information about the location that we have (if exists).

Muster Points

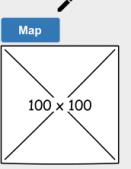
Previous text about where to meet in case of emergency. Text box appears when clicking the pencil.

Map



100 x 100

Map



100 x 100

Emergency Procedures Creation

Event Title > Emergency Procedures

Procedures

- [Contacts](#)
- [Exits](#)
- [Muster Points](#)

<Event Title> Emergency Procedures

Emergency Contacts

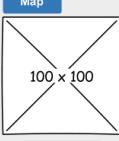
Please call 911 in case of an emergency.

Emergency Contact Name	Emergency Contact Number
Emergency Contact Name	Emergency Contact Number

Building Exits

Description of exits.

Map



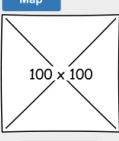
100 x 100

Upload

Muster Points

Where to meet in case of emergency.

Map



100 x 100

Upload

Save **Cancel**

Emergency Procedures View

Event Title > Emergency Procedures

[Find Event](#) [Volunteer](#) [Supplier](#) [Create Event](#) [Login](#)

Procedures

[Contacts](#)
[Exits](#)
[Muster Points](#)

<Event Title> Emergency Procedures [Edit](#)

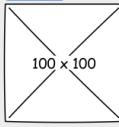
Emergency Contacts

Please call 911 in case of an emergency.

Emergency Contact Name	Emergency Contact Number
The Wizard of Oz	319-123-4567
Arthur McAlpine	563-765-4321

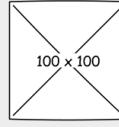
Building Exits
Description of exits.

[Map](#)



Muster Points
Where to meet in case of emergency.

[Map](#)



4.2.1.9 Event (1007,1009, 1010, 1011, 1012, 1013, 1014, 1015, 1017)

Share Event

Tadpole

Create Event **My Events** Suppliers Volunteers About Administration

Event Name

Description

Description of event is happening here

Date

Date of Event is here

Location

Location of the event is listed here. Building included (if relevant)

Search Find

Share Edit

Info Page

Event Details

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

1017-View Calender

[Find Event](#)
[Volunteer](#)
[Supplier](#)
[Create Event](#)

[Login](#)

[View Events](#)

Search

Month

<img alt="Month 200" style="width: 150px; height: 150px; border: 1

1015-Volunteer 4

The screenshot shows a web-based application interface. At the top, there is a blue header bar with the following elements from left to right: a house icon, a 'Find Event' button, 'Volunteer', 'Supplier', and 'Create Event' buttons. To the right of these are several icons: a star, a calendar, a bell with a '2' notification, a speech bubble, and a 'Login' button. Below the header, the main content area has a light gray background. On the left, there is a vertical sidebar with a list of navigation items: 'Nav1 > Nav2 > Nav3'. In the center, there are two input fields: 'Sort' with a dropdown arrow and 'Filter' with a funnel icon. To the right of these fields is a list of notifications. The notifications are contained within a white box with a thin gray border. The first notification is 'Event Title - Volunteers Changed'. Below it are four 'Seen Notification - Description...' entries, each followed by a small downward arrow. The entire interface is set against a light gray background.

Nav1 > Nav2 > Nav3

Sort

Filter

<Event Title> - Volunteers Changed

Unseen Notification - Description...

Seen Notification - Description...

Seen Notification - Description...

Seen Notification - Description...

▼

1015-Volunteer 3

The screenshot shows a web application interface for managing volunteers. At the top, there is a blue header bar with icons for Home, Find Event, Volunteer, Supplier, Create Event, a star, a calendar, a bell, a speech bubble, and a Login button. Below the header, the URL is shown as Event > Volunteers > Request Volunteers. To the right is a search bar with a magnifying glass icon and a clear button. The main content area has a light gray background with a grid pattern. In the center, the word "Volunteers" is displayed in a bold, black, sans-serif font. To the left of this, there is a section titled "Request Volunteers" with a checkbox labeled "Yes, I need volunteers". Below this is a "Add Volunteer" button. To the right of the "Add Volunteer" button is a "How many?" input field with a dropdown arrow. Further right is a checkbox labeled "I w...". Overlaid on the main content is a modal window titled "Edit Volunteers" with a close button (X). Inside the modal, there is a warning message: "⚠ Are you sure you would like to edit the amount of volunteers you want to request?". At the bottom of the modal are two buttons: "Change" and "Cancel".

1015-Volunteer 2

The screenshot shows a web application interface for managing volunteers. At the top, there is a blue header bar with the following elements from left to right: a house icon, 'Find Event', 'Volunteer', 'Supplier', 'Create Event' (which is highlighted in yellow), a star icon, a calendar icon, a bell icon, a speech bubble icon, and a 'Login' button. Below the header, the URL 'Event > Volunteers > Request Volunteers' is visible, along with a search bar containing a magnifying glass icon, the word 'Search', and a clear 'X' button.

The main content area has a light gray background with a grid of 12 light gray boxes arranged in three rows of four. In the center of this grid, the word 'Volunteers' is displayed in bold black text. To the left of this grid, there is a section titled 'Request Volunteers' with a sub-section titled 'Add Volunteers'. Inside this sub-section, there is a dropdown menu labeled 'How many?' with a downward arrow icon. Below the dropdown is a checkbox labeled 'I want to give more information about the type of volunteer I need'. At the bottom of the 'Add Volunteers' box are two buttons: a blue 'Next' button and a red 'Cancel' button.

1015-Volunteer

Event > Volunteers

Volunteers

Current Volunteers

Volunteers Requested 1

Search

Edit

1014 - Add Vendor Tags 3

Nav1 > Nav2 > Nav3

Sort Filter

<Event Title> - Vendor Req. Edit

Unseen Notification - Description...

Seen Notification - Description...

Seen Notification - Description...

Seen Notification - Description...

▼

1014 - Add Vendor Tags 2

Event Title > Vendors > [Edit Vendors](#)

Vendors

Request Vendors

+ Add Vendor Request

Vender Type

Food Truck

Farmer's Stand

Number? 1

Add another vendor type

Next **Cancel**

1014 - Add Vendor Tags

Event Title > [Vendors](#)

Vendors

Current Vendor Request

Vendor Type	Number
Food Truck	3
Cleaning Crew	1
Clowns	22

Search

Edit

Event Dates Notification

The screenshot shows a web-based application interface for managing event notifications. The top navigation bar includes links for Home, Find Event, Volunteer, Supplier, and Create Event. On the right side of the header are icons for a star, a calendar, a bell (with a '2' notification), a speech bubble, and a Login button. Below the header, a breadcrumb trail indicates the current path: Nav1 > Nav2 > Nav3. To the right of the breadcrumb are 'Sort' and 'Filter' buttons. The main content area features a heading 'Heading level 1' and a list of notifications on the right side. The notifications are as follows:

- <Event Title> - Event Date Changed
- Unseen Notification - Description...
- Seen Notification - Description...
- Seen Notification - Description...
- Seen Notification - Description...

A small downward arrow icon is located at the bottom right of the notification list.

Event Dates Confirm Change

The screenshot shows a web-based application for managing event dates. At the top, there is a blue header bar with icons for Home, Find Event, Volunteer, Supplier, Create Event, a star, a calendar, a bell, a speech bubble, and a Login button. Below the header, the URL 'Event Title > Dates > Edit Dates' is visible, along with a search bar and a close button.

The main content area is titled 'Edit Event Dates' and contains a sub-section titled 'Select Dates'. A modal dialog box is open, titled 'Delete?', containing the message: 'Are you sure you would like to change the date of this event?'. It has two buttons: 'Change' and 'Cancel'. In the background, there is a form for adding a date, with fields for 'From' (08/01/2022) and 'To' (08/01/2022), and a checkbox for 'Add another date'. Below the form is a note: '* Add another panel' and a numeric range from 4 to 10. At the bottom of the page are 'Next' and 'Cancel' buttons.

Edit Event Dates

Event Title > Dates > [Edit Dates](#)

Edit Event Dates

Select Dates

Add Date

From: 08/01/2016

One day Event

To: 08/01/2016

Add another date

* Add another panel

Next

AUG - 2016

S	M	T	W	T	F	Sa
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

1013 Event Dates- Retrieve

Find Event Volunteer Supplier Create Event [Login](#)

Event Title > [Dates](#)

[Filter](#) [Search](#)

<Event Title> Dates

 -<Event Title> on this day

AUG - 2016

S	M	T	W	T	F	Sa
31	 1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

1012- Location Directions 5

The screenshot shows a web application interface for managing event directions. At the top, there is a blue header bar with the following elements from left to right: a house icon, 'Find Event', 'Volunteer', 'Supplier', 'Create Event' (in yellow), a star icon, a calendar icon, a bell icon, a speech bubble icon, and a 'Login' button. Below the header, the URL 'Find Event > Event Title > Location > Directions' is displayed. To the right of the URL is a search bar with a magnifying glass icon and a clear 'X' button. The main content area is titled 'Event Directions' in bold black text. Below this, there are two sections for 'Edit Directions'. The first section is titled 'Directions 1' and contains fields for 'Directions Title' (with an empty input box) and 'Give Directions' (with an empty text area). Below these fields are two buttons: a blue 'Change' button and a red 'Cancel' button. The second section is titled 'Directions 2(if exists)' and contains similar fields for 'Directions Title' and 'Give Directions', also with empty input fields. Below these fields are two buttons: a blue 'Change' button and a red 'Cancel' button. The entire interface is set against a light gray background with a grid pattern.

Find Event > Event Title > Location > Directions

Event Directions

Edit Directions

Directions 1

Directions Title

Give Directions

Change Cancel

Directions 2(if exists)

Directions Title

Give Directions

Change Cancel

1012- Location Directions 4

The screenshot shows a web-based application interface. At the top, there is a blue header bar with the following elements from left to right: a house icon, a 'Find Event' button, 'Volunteer' and 'Supplier' buttons, a 'Create Event' button, a star icon, a calendar icon, a bell icon with a '2' notification, a speech bubble icon, and a 'Login' button. Below the header, the main content area has a light gray background. On the left side of this area, there is a vertical navigation path: 'Nav1 > Nav2 > Nav3'. In the center, there are two input fields: 'Sort' with a dropdown arrow and 'Filter' with a magnifying glass icon. To the right of these fields is a vertical list of notifications, each enclosed in a rounded rectangle. The notifications are as follows:

- <Event Title> - Directions change
- Unseen Notification - Description...
- Seen Notification - Description...
- Seen Notification - Description...
- Seen Notification - Description...

At the bottom right of the notification list is a small downward-pointing arrow icon.

1012- Location Directions 3

The screenshot shows a web application interface. At the top, there is a blue header bar with icons for Home, Find Event, Volunteer, Supplier, Create Event, a star, a calendar, a bell, a speech bubble, and a Login button. Below the header, a breadcrumb navigation shows 'Find Event > Event Title > Location > Directions'. A search bar with a magnifying glass icon and a clear button is also present. The main content area has a title 'Event Directions' and a sub-section 'Current Location'. On the left, there is a form titled 'Edit Location' with fields for Building Name (Name if exists), Address (742 Evergreen), and City (Springfield). On the right, a modal window titled 'Change Directions' contains a warning message: 'Are you sure you would like to change the directions. All managers and participants will be notified.' with 'Change' and 'Cancel' buttons. At the bottom of the main content area, there are 'Change' and 'Cancel' buttons.

1012- Location Directions 2

The screenshot shows a web application interface for managing event directions. At the top, there is a blue header bar with icons for Home, Find Event, Volunteer, Supplier, Create Event, a star, a calendar, a bell, a message bubble, and a Login button. Below the header, a breadcrumb navigation shows the path: Find Event > Event Title > Location > Directions. To the right of the breadcrumb is a search bar with a magnifying glass icon and a clear 'X' button. The main content area is titled 'Event Directions' and contains two sections for 'Edit Directions'. The first section is for 'Directions 1' and the second is for 'Directions 2(if exists)'. Each section has fields for 'Directions Title' (with a text input box) and 'Give Directions' (with a larger text area). Below each section are 'Change' and 'Cancel' buttons. The background of the page features a grid of light gray vertical and horizontal lines.

Find Event > Event Title > Location > [Directions](#)

Search X

Event Directions

Edit Directions

Directions 1

Directions Title

Give Directions

Change Cancel

Directions 2(if exists)

Directions Title

Give Directions

Change Cancel

1012- Location Directions

Find Event > Event Title > Location > [Directions](#)

Event Directions

Add Directions to Location

Directions 1

Directions	Title	Name if exists
Give Directions	Septemque fulminibus moderantium ulla piscibus quin cuncta viseret imagine meis dedit dominari erat: fuerat peragebant.	

Directions 2(if exists...)

Directions	Title	Name if exists
Give Directions	Septemque fulminibus moderantium ulla piscibus quin cuncta viseret imagine meis dedit dominari erat: fuerat peragebant.	

1011 - Location 4

The screenshot shows a web-based application interface. At the top, there is a blue header bar with the following elements from left to right: a house icon, a 'Find Event' button, 'Volunteer', 'Supplier', and 'Create Event' buttons, a star icon, a calendar icon, a bell icon with a '2' notification, a speech bubble icon, and a 'Login' button. Below the header, the main content area has a light gray background. On the left, there is a vertical sidebar with a list of navigation items: 'Nav1 > Nav2 > Nav3'. In the center, there are two input fields: 'Sort' with a dropdown arrow and 'Filter' with a funnel icon. To the right of these fields is a list of notifications, each with a small downward arrow icon at the bottom right. The notifications are as follows:

- <Event Title> - Location change
- Unseen Notification - Description...
- Seen Notification - Description...
- Seen Notification - Description...
- Seen Notification - Description...

1011 - Location 3

The screenshot shows a web application interface for managing event locations. At the top, there is a blue header bar with icons for Home, Find Event, Volunteer, Supplier, Create Event, a star, a calendar, a bell, a speech bubble, and a Login button. Below the header, the URL 'Create Event > Event Title > Location' is visible, along with a search bar and a close button.

The main content area is titled 'Event Location'. On the left, there is a 'Current Location' section with a 'Edit Location' button. This section contains fields for 'Building Name' (with placeholder 'Name if exists'), 'Address' (742 Evergreen), and 'City' (Springfield). Below these fields are 'Change' and 'Cancel' buttons.

A modal dialog box is overlaid on the page, titled 'Change Location' with a close button. It contains a warning message: '⚠ Are you sure you would like to change the location. All managers and participants will be notified.' At the bottom of the modal are 'Change' and 'Cancel' buttons.

1011 - Location 2

Find Event Volunteer Supplier **Create Event**

★    **Login**

Create Event > Event Title > [Location](#)

X

Event Location

Current Location

Edit Location

Building Name	<input type="text" value="Name if exists"/>				
Address	742 Evergreen Terrace 				
City	<input type="text" value="Springfield"/>	State	<input type="text" value="Iolloniorado"/>	Zip Code	<input type="text" value="12345"/>

Change **Cancel**

1011 - Location

Find Event > Event Title > Location

Event Location

Current Location

Building Name	Name if exists
Address	742 Evergreen Terrace
City	Springfield
State	Iolloniorado
Zip Code	12345

Search

Edit

1010- Event Update 9

Event Title > Event Edit

<Event Title> Edit

Create Dates Location Directions Volunteers Budget Vendors Confirm

Confirm Details

Event Detail

- >Title
- >Date
- Location

Budget Category

- Category

Vendor Type

Vendor Type	Number
Category	Total

Save Cancel

1010- Event Update 8

Find Event Volunteer Supplier **Create Event**     

Event Title > [Event Edit](#)

<Event Title> Edit

[Create](#) [Dates](#) [Location](#) [Directions](#) [Volunteers](#) [Budget](#) [Vendors](#) **Confirm**

Confirm Details

Event Detail	Content
 Title	Description
 Date	08/01/2016
 Location	Name, Address, City, State, Zip

Budget Category Total	Total Budget	Total Actual
 Category	Total Budget	Total Actual

Vendor Type	Number
 Category	Total

Save **Cancel**

1010- Event Update 7

Event Title > Event Edit

<Event Title> Edit

Create Dates Location Directions Volunteers Budget Vendors Confirm

Request Vendors

Yes, I would like to find vendors

Vendor Type	Number
Food Truck	3
Cleaning Crew	1

Add another vendor type

Save Cancel

1010- Event Update 6

Event Title > Event Edit

<Event Title> Edit

Create Dates Location Directions Volunteers **Budget** Vendors Confirm

Update Budget

No Current Budget

+ Add Budget Category

Category	Budget	Actual
Line Item	\$0.00	\$0.00
Total	\$0.00	\$0.00

Add another category
* Add another panel

Category Totals

Category Name	Total Budget	Total Actual
Category	\$0.00	\$0.00
...
Event Total Budget	\$0.00	\$0.00

Save **Cancel**

1010- Event Update 5

The screenshot shows a web-based event management system. At the top, there is a blue header bar with the following elements from left to right: a house icon, 'Find Event', 'Volunteer', 'Supplier', 'Create Event' (which is highlighted in yellow), a star icon, a calendar icon, a bell icon, a speech bubble icon, and a 'Login' button. Below the header, the URL 'Event Title > Event Edit' is visible, followed by a search bar with a magnifying glass icon, the word 'Search', and a clear 'X' button. The main content area has a title '**<Event Title> Edit**' and a navigation bar with tabs: 'Create', 'Dates', 'Location', 'Directions', 'Volunteers' (which is the active tab, highlighted in blue), 'Budget', 'Vendors', and 'Confirm'. Below this, a sub-section titled 'Edit Volunteers' is displayed. It contains a checkbox labeled 'Yes, I need volunteers'. A modal window titled 'Add Volunteers' is open, featuring a dropdown menu for selecting the number of volunteers ('How many?') and a dropdown menu for selecting an old number. There is also a checkbox for providing more information about the type of volunteer needed. At the bottom of the modal are two buttons: a blue 'Save' button and a red 'Cancel' button. The rest of the page is a large, empty white area with a grid of light gray vertical and horizontal lines.

1010- Event Update 4

Event Title > Event Edit

<Event Title> Edit

Create Dates Location Directions Volunteers Budget Vendors Confirm

Edit Directions to Location

Unknown

Add Direction

Directions Title: Old Directions

Give Directions: Old Directions

Save Cancel

1010- Event Update 3

Event Title > Event Edit

<Event Title> Edit

Create Dates Location Directions Volunteers Budget Vendors Confirm

Select Location

Unknown

Add Location

Building Name	Current Name	Name if exists
Address	742 Evergreen Terrace	
City	Springfield	
State	Iolloniorado	
Zip Code	12345	

Save Cancel

1010- Event Update 2

Event Title > Event Edit

<Event Title> Edit

Create Dates Location Directions Volunteers Budget Vendors Confirm

Edit Dates

Unknown

Edit Date

Current Date:
From: 08/01/2016 To: 08/01/2016 One day Event

To: 08/01/2016 To: 08/01/2016

Add another date
* Add another panel



1010- Event Update

Event Title > Event Edit

<Event Title> Edit

Create Dates Location Directions Volunteers Budget Vendors Confirm

Edit Event

Title <Event Title>

Description

Congeriem oppida tractu tollere undae aberant animalia occiduo fixo abscidit terrarum aethere vindice ambitae pluvialibus montes viseret nullaque mutatis silvas caligine volucres dissociata circumflus nec erant nam origo nondum vindice aethere nulli illis numero.

Save Cancel

Vendors Tab

The screenshot shows a web-based event planning application. At the top, there is a blue header bar with the following elements from left to right: a house icon, 'Find Event', 'Volunteer', 'Supplier', 'Create Event' (which is highlighted in yellow), a star icon, a calendar icon, a bell icon, a speech bubble icon, and a 'Login' button. Below the header is a search bar with a magnifying glass icon and the word 'Search', and a clear 'X' button. The main content area has a title 'Create Event' in bold. Below the title is a horizontal navigation bar with several tabs: 'Create', 'Dates', 'Location', 'Directions', 'Volunteers', 'Budget', 'Vendors' (which is the active tab and highlighted in yellow), and 'Confirm'. The main content area is titled 'Request Vendors'. It contains a checkbox labeled 'Yes, I would like to find vendors'. Below this is a section titled '+ Add Vendor Request' with a dropdown menu for 'Vendor Type' containing 'Food Truck' and 'Farmer's Stand'. To the right of the dropdown is a 'Number?' input field with the value '1'. Below the dropdown is another checkbox labeled 'Add another vendor type'. At the bottom of the form are two buttons: a blue 'Next' button and a red 'Cancel' button.

Budget

The screenshot shows a web-based event management application. At the top, there is a navigation bar with links for 'Find Event', 'Volunteer', 'Supplier', 'Create Event', and a 'Login' button. Below the navigation bar, there is a search bar with a magnifying glass icon and a 'Create Event' button. The main content area is titled 'Create Event' and contains several tabs: 'Create', 'Dates', 'Location', 'Directions', 'Volunteers', 'Budget' (which is currently selected), 'Vendors', and 'Confirm'. The 'Create a Budget' section contains a message 'No Current Budget' and a link '+ Add Budget Category'. A table is shown with columns 'Category', 'Budget', and 'Actual'. It has a single row for 'Line Item' with a value of '\$0.00' in both columns. There is a '+' button to add more categories. A 'Delete Category' button is visible. Below this, there is a section for 'Category Totals' with a table showing 'Category Name', 'Total Budget', and 'Total Actual'. The table has rows for 'Category' (value: '\$0.00'), 'Event Total Budget' (value: '\$0.00'), and a row with three dots. At the bottom of the section are 'Next' and 'Cancel' buttons.

Category	Budget	Actual
Line Item	\$0.00	\$0.00
+		
Total	\$0.00	\$0.00

Category Name	Total Budget	Total Actual
Category	\$0.00	\$0.00
...
Event Total Budget	\$0.00	\$0.00

Add another category

No Current Budget

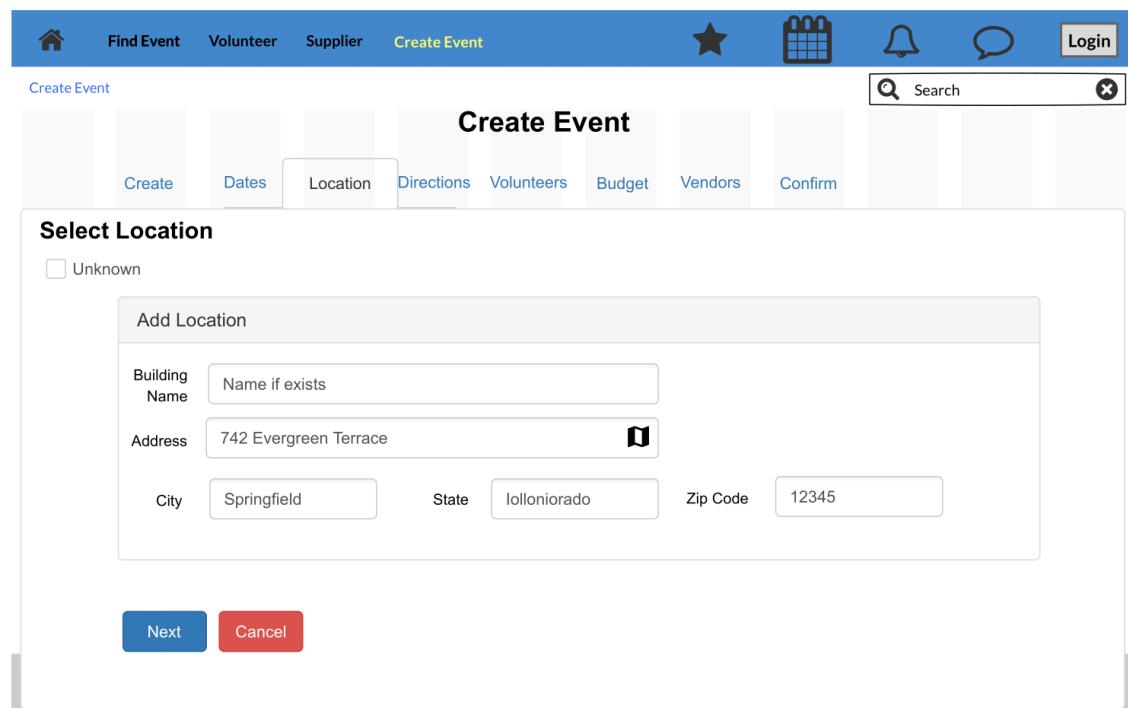
Create a Budget

[+ Add Budget Category](#)

Category Totals

[Next](#) [Cancel](#)

Location Tab



The screenshot shows a web-based application for creating events. At the top, there is a navigation bar with icons for Home, Find Event, Volunteer, Supplier, and Create Event. To the right of the navigation are icons for a star, a calendar, a bell, a message bubble, and a Login button. Below the navigation bar is a search bar with a magnifying glass icon and the word 'Search'. The main content area is titled 'Create Event' and contains a series of tabs: Create, Dates, Location, Directions, Volunteers, Budget, Vendors, and Confirm. The 'Location' tab is currently selected. Below the tabs, there is a section titled 'Select Location' with a checkbox labeled 'Unknown'. A 'Add Location' form is displayed, containing fields for Building Name (with a placeholder 'Name if exists'), Address (with a placeholder '742 Evergreen Terrace' and a small location pin icon), City (Springfield), State (Illinois), and Zip Code (12345). At the bottom of the form are 'Next' and 'Cancel' buttons.

Create Event

Create Event

Create Dates Location Directions Volunteers Budget Vendors Confirm

Select Location

Unknown

Add Location

Building Name

Address 

City State Zip Code

Next Cancel

Confirm

Find Event Volunteer Supplier **Create Event**

★    **Login**

Create Event

Create Event

Create Dates Location Directions Volunteers Budget Vendors **Confirm**

Confirm Details

Event Detail

 Title

 Date

 Location

Budget Category

 Category

Create Event?

Are you sure you want to create this event?

Create **Cancel**

Vendor Type Number

 Category Total

Next **Cancel**

Confirm Details Tab

Create Event

Confirm

Confirm Details

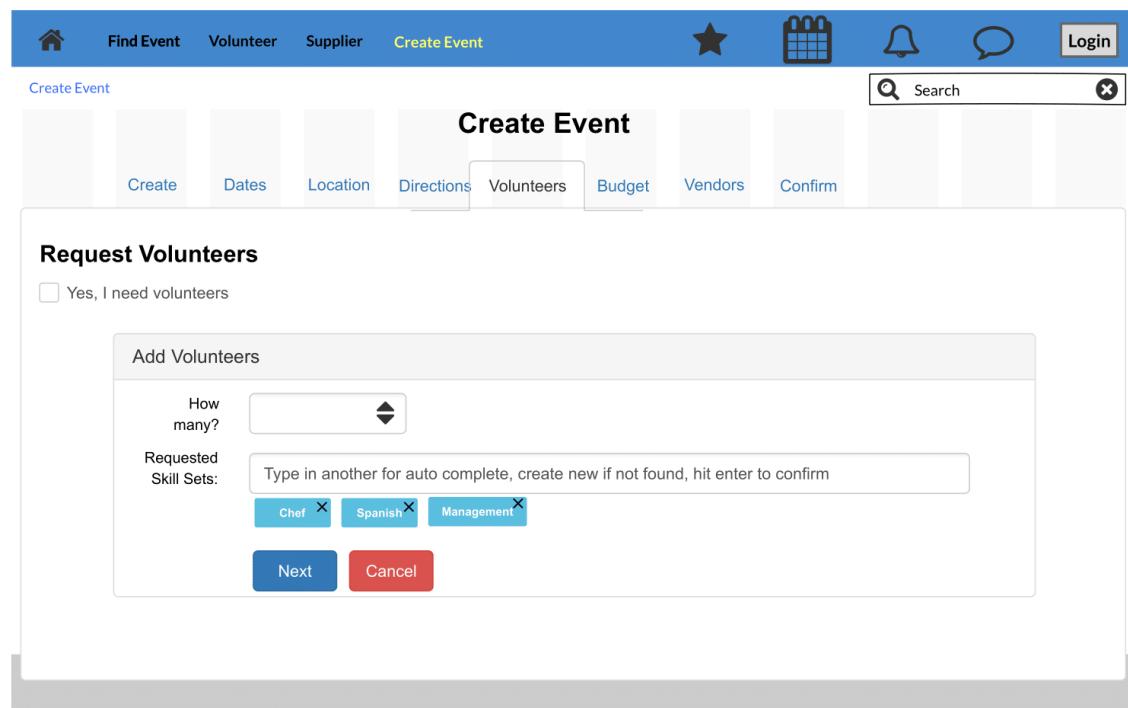
Event Detail	Content
<input type="text"/> Title	Description
<input type="text"/> Date	08/01/2016
<input type="text"/> Location	Name, Address, City, State, Zip

Budget Category Total	Total Budget	Total Actual
<input type="text"/> Category	Total Budget	Total Actual

Vendor Type	Number
<input type="text"/> Category	Total

Next **Cancel**

Volunteers Tab



The screenshot shows a web-based event planning application. At the top, there is a navigation bar with icons for Home, Find Event, Volunteer, Supplier, and Create Event. To the right of the navigation are icons for a star, a calendar, a bell, a message bubble, and a Login button. Below the navigation is a search bar with a magnifying glass icon and the word 'Search'. The main content area is titled 'Create Event' and contains a sub-navigation bar with tabs: Create, Dates, Location, Directions, Volunteers (which is highlighted in blue), Budget, Vendors, and Confirm. The 'Volunteers' tab is currently active, displaying a form titled 'Request Volunteers'. The form includes a checkbox labeled 'Yes, I need volunteers'. Below this is a section titled 'Add Volunteers' with a dropdown menu for 'How many?'. A text input field for 'Requested Skill Sets:' contains the text 'Type in another for auto complete, create new if not found, hit enter to confirm'. Below this field are three buttons: 'Chef X', 'Spanish X', and 'Management X'. At the bottom of the form are two buttons: 'Next' (blue) and 'Cancel' (red).

Directions Tab

Create Event

Create Event

Create Dates Location Directions Volunteers Budget Vendors Confirm

Add Directions to Location

Unknown

Add Direction

Directions Title Name if exists

Give Directions Directions

Next Cancel

Dates Tab

Find Event Volunteer Supplier **Create Event**     

Create Event

Create Dates Location Directions Volunteers Budget Vendors Confirm

Select Dates

Unknown

Add Date

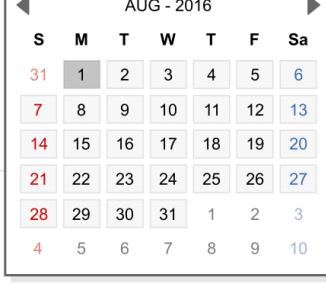
From: 08/01/2016 

One day Event

To: 08/01/2016 

Add another date

* Add another panel



Aug - 2016

S	M	T	W	T	F	Sa
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

Next **Cancel**

1009- Create Event - Title and Description Tab

Create Event

Search

Create Event

Create Dates Location Directions Volunteers Budget Vendors Confirm

Create New Event

Title Text

Description Text

Next Cancel

Find Event Volunteer Supplier **Create Event**

★    [Login](#)

Create Event

Add Dates

Unknown date

Date 

Start Time 

End Time 

Date	Start	End
<input type="button" value="Next"/>	<input type="button" value="Cancel"/>	

Find Event Volunteer Supplier **Create Event**

★    **Login**

Create Event

Create Event

Add Location

Select from verified locations

Name

City

State

Address

Zip

Event Details

The screenshot shows a web page layout for an event details page. At the top is a blue header bar with a house icon, 'Events' (highlighted in yellow), 'Volunteer', 'Supplier', and 'Create Event' buttons. To the right are 'Register' and 'Login' buttons. The main content area is divided into two sections: 'Options/ Page Details' on the left and 'Event Name' on the right. The 'Event Name' section contains sections for 'About Event', 'Date', and 'Location' with their respective descriptions. On the left sidebar, there are three blue 'Hyperlink' buttons.

Event Name

About Event
Description of event is happening here

Date
Dates event takes place on here

Location
Location of the event is listed here. Hotlink to location page

Options/ Page Details

[Hyperlink](#)

[Hyperlink](#)

[Hyperlink](#)

4.2.1.10 File (1024,1052,1053)

Delete Notification

Find Event Volunteer Supplier Create Event Login

Unseen Notification - File Deleted...

Unseen Notification - File being d...

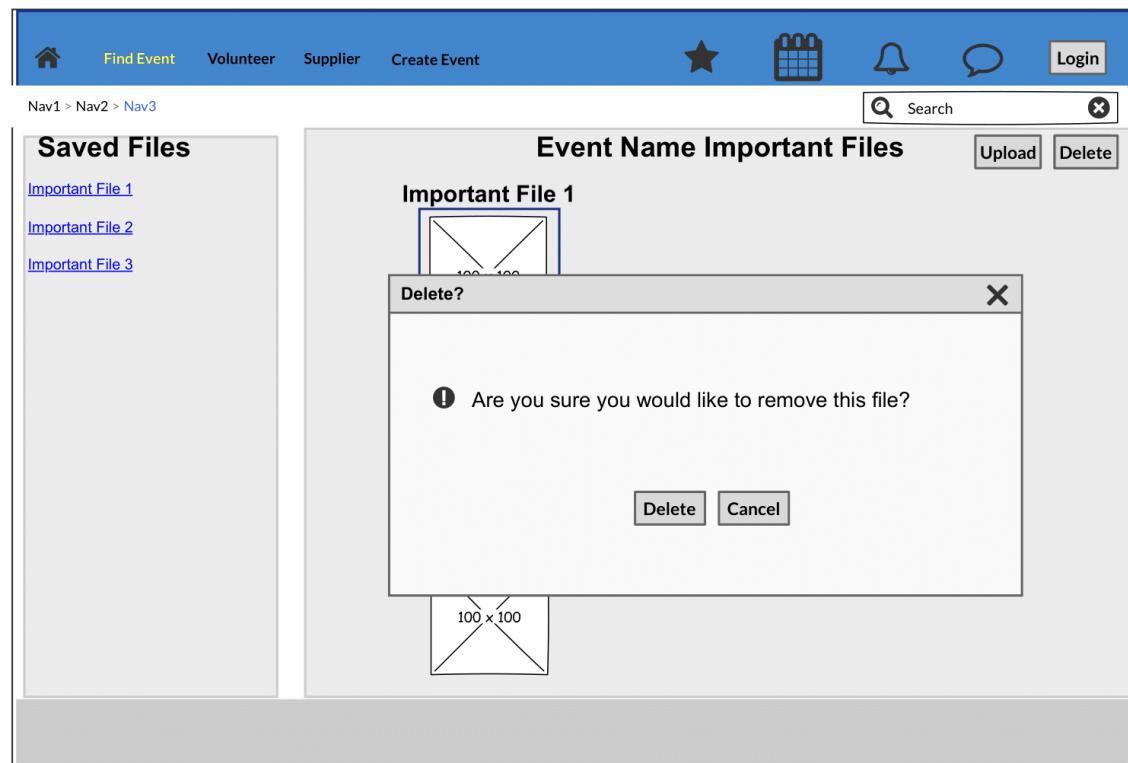
Seen Notification - Description...

Seen Notification - Description...

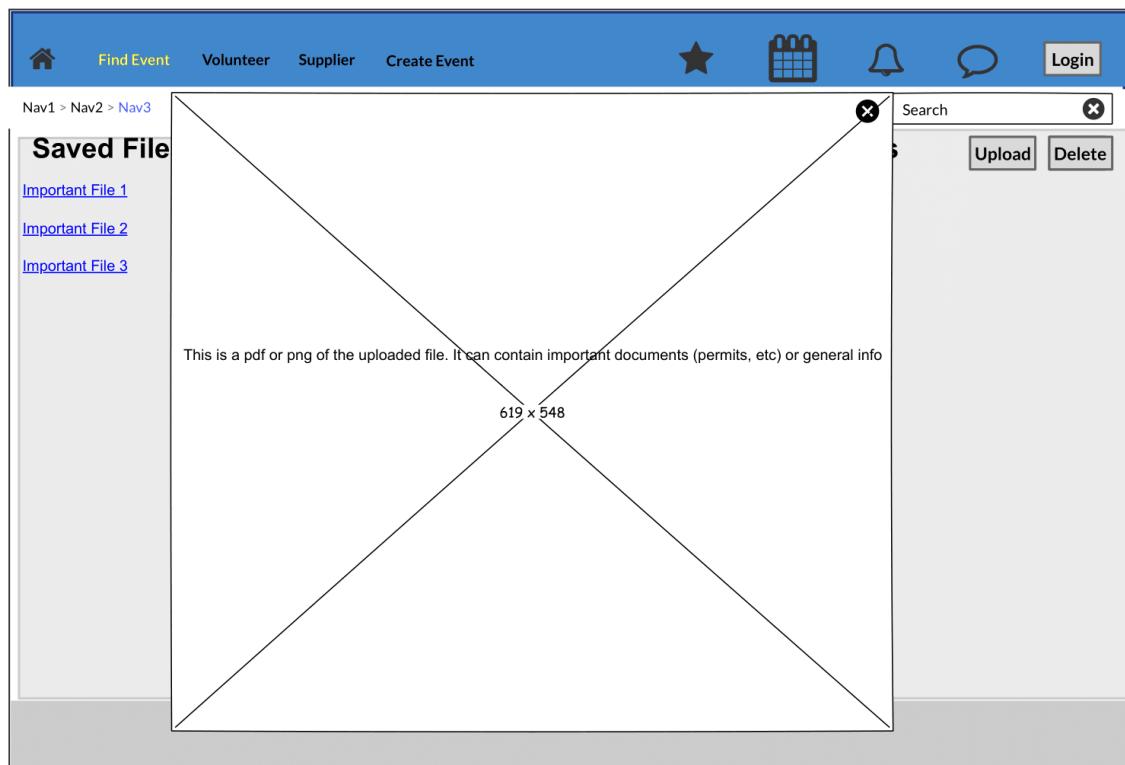
Seen Notification - Description...

▼

Delete Message



View Single



After Upload

The screenshot shows a web application interface with a blue header bar. The header contains a home icon, navigation links for 'Find Event', 'Volunteer', 'Supplier', and 'Create Event', and icons for a star, calendar, bell, and message. A 'Login' button is also present. Below the header, a breadcrumb navigation path 'Nav1 > Nav2 > Nav3' is displayed. The main content area has a light gray background. On the left, a sidebar titled 'Saved Files' lists four files: 'Important File 1', 'Important File 2', 'Important File 3', and 'Important File 4', each with a blue link underline. The main content area is titled 'Event Name Important Files'. It displays four files arranged in a 2x2 grid. Each file is represented by a square icon with a diagonal cross and the text '100 x 100' in the center. The files are labeled 'Important File 1', 'Important File 2', 'Important File 3', and 'Important File 4'. At the top right of this section are 'Upload' and 'Delete' buttons. A search bar with a magnifying glass icon and a clear button is located at the top right of the main content area. The bottom of the page has a dark gray footer bar.

Upload

The screenshot shows a web application interface for file upload. The top navigation bar is blue and includes links for 'Find Event', 'Volunteer', 'Supplier', and 'Create Event'. It also features icons for a star, calendar, bell, and speech bubble, along with a 'Login' button. Below the navigation is a breadcrumb trail: 'Nav1 > Nav2 > Nav3'. A search bar with a magnifying glass icon and a 'Delete' button are on the right. The main content area has a title 'Event Name Important Files' and a sub-section 'Upload File'. On the left, a sidebar titled 'Saved Files' lists 'Important File 1', 'Important File 2', and 'Important File 3'. The 'Upload File' section contains a 'Filename' input field with 'Filename.ext' placeholder text, a placeholder image for a file (a square with a diagonal cross and '136 x 136' text), a search bar with a magnifying glass icon and 'Search for file by name' placeholder text, and a 'Delete' button. A large 'Upload File' button is at the bottom.

File Page View

Find Event Volunteer Supplier Create Event [Login](#)

Nav1 > Nav2 > Nav3

Event Name Important Files

Important File 1
100 x 100

Important File 2
100 x 100

Important File 3
100 x 100

Upload **Delete**

4.2.1.11 Forum (Web)(1008,1016,1048,1049,1056,1057,1058,1059)

1058_Forum Comment Create

Find Event Volunteer Supplier Create Event ★ Calendar Notification Comment Login

Nav1 > Nav2 > Nav3

Forum

▼ Event Forum

- # Attendees
- # RSVP Attendees
- # Volunteers
- # Public
- # Add

▼ Message Sub Cat

- # Chat Group Name

▼ Indiv Messages

#Inbox

Forum Name

Forum Description

Profile Pic Name Ab umor, ardenter tenent nullus habendum convexi sublime aer fronde peregrinum nam manebat solum proxima et terrenae usu librata amphitrite descenderat cepit vindice deducite invasit mixta onus.

999 Heart 999 Comment 10/6/2021 at 12:00pm

Profile Pic Name Ab umor, ardenter tenent nullus habendum convexi sublime aer fronde peregrinum nam manebat solum proxima et terrenae usu librata amphitrite descenderat cepit vindice deducite invasit mixta onus.

999 Heart 10/6/2021 at 12:00pm

Profile Pic Your Name Type comment here

✓ ✕

1049_Forum Delete Post

The screenshot shows a user interface for managing forum posts. On the left, a sidebar contains navigation links for 'Event Forum', 'Message Sub Cat', and 'Indiv Messages'. The main area is titled 'Forum Name' and shows a list of posts. The first post is highlighted with a red box and a delete icon. A modal window titled 'Are you sure?' with a checkmark and an 'X' button is overlaid on the post. The post content is placeholder text: 'Ab umor, ardentior tenent nullus habendum convexi sublime aer fronde peregrinum nam manebat solum proxima et terrenae usu librata amphitrite descenderat cepit vindice deducite invasit mixta onus.' Below the post are '999' likes and '999' comments. The second post shows a placeholder image for an 'Image Attachment'. The third post is timestamped '10/6/2021 at 12:00pm'. The top navigation bar includes links for 'Find Event', 'Volunteer', 'Supplier', 'Create Event', and 'Login'.

1016_Voting (Create Post add voting)

Forum Name
Forum Description

Profile Pic
Name

Ab humor, ardenter tenent nullus habendum convexi sublime aer fronde peregrinum nam manebat solum proxima et terrenae usu librata amphitrite descenderat cepit vindice deducite invasit mixta onus.

999 ❤️ 999 🗣

10/6/2021 at 12:00pm

Add Attachment

Voting Poll

Due Date: 10/1/2021 at 12:00pm

Single Vote Ranking List

Option 1

Add another option

✓ ✕

1048_Forum Create Post (Add Image)

Forum Name
Forum Description

Profile Pic
Name

Ab humor, ardenter tenent nullus habendum convexi sublime aer fronde peregrinum nam manebat solum proxima et terrenae usu librata amphitrite descenderat cepit vindice deducite invasit mixta onus.

999 ❤️ 999 🗣

10/6/2021 at 12:00pm

Add Attachment

Image

Choose Image

✓ ✕

1048_Forum Create Post

The screenshot shows a web-based application interface for creating a forum post. The top navigation bar includes links for 'Find Event', 'Volunteer', 'Supplier', and 'Create Event', along with icons for a star, calendar, bell, and message, and a 'Login' button. A search bar is also present.

The main content area is titled 'Forum Name' and includes a 'Forum Description' section. The description text is: "Ab humor, ardenter tenent nullus habendum convexi sublime aer fronde peregrinum nam manebat solum proxima et terrenae usu librata amphitrite descenderat cepit vindice deducite invasit mixta onus." Below this, there are two posts from user '999'. Each post includes a profile picture placeholder, the user's name '999', a heart icon, a comment icon, and the timestamp '10/6/2021 at 12:00pm'. The second post is identical to the first.

The sidebar on the left contains the following sections and links:

- Forum**
- Event Forum**
 - # Attendees
 - # RSVP Attendees
 - # Volunteers
 - # Public
 - [# Add](#)
- Message Sub Cat**
 - # Chat Group Name
 - # Chat Group Name
 - # Chat Group Name
 - # Chat Group Name
- Indiv Messages**
 - #Inbox

1057_Voting Results View

Nav1 > Nav2 > Nav3

Forum

Event Forum

- # Attendees
- # RSVP Attendees
- # Volunteers
- # Public

Add

Message Sub Cat

- # Chat Group Name

Indiv Messages

#Inbox

Forum Name

Forum Description

Profile Pic

Name

Ab humor, ardenter tenet nullus habendum convexi sublime aer fronde peregrinum nam manebat solum proxima et terrenae usu librata amphitrite descenderat cepit vindice deducite invasit mixta onus.

Results from 999 people:

Drink	Percentage
Pepsi	25%
Coke	25%
Sprite	10%
Mountain Dew	40%

Expired: 10/10/2021 at 12:00pm

999 999 10/6/2021 at 12:00pm

1056_Voting Cast Vote

The screenshot shows a web application interface with a blue header bar. The header contains a home icon, links for 'Find Event', 'Volunteer', 'Supplier', and 'Create Event', and icons for a star, calendar, bell, and message. A 'Login' button is also present. Below the header, a breadcrumb navigation shows 'Nav1 > Nav2 > Nav3'. The main content area is divided into two sections: 'Forum' on the left and 'Forum Name' on the right.

Forum

Event Forum

- # Attendees
- # RSVP Attendees
- # Volunteers
- # Public

Add

Message Sub Cat

- # Chat Group Name

Indiv Messages

#Inbox

Forum Name

Forum Description

Profile Pic

Name

Ab umor, ardentior tenent nullus habendum convexi sublime aer fronde peregrinum nam manebat solum proxima et terrenae usu librata amphitrite descenderat cepit vindice deducite invasit mixta onus.

Please Pick One:

- Pepsi
- Coke
- Sprite
- Mountain Dew

Expires: 10/10/2021 at 12:00pm

999 999

10/6/2021 at 12:00pm

Profile Pic

Name

Ab umor, ardentior tenent nullus habendum convexi sublime aer fronde peregrinum nam manebat solum proxima et terrenae usu librata amphitrite descenderat cepit vindice deducite invasit mixta onus.

Please Rank from 1(best) to 4(worst):

- Pepsi
- Coke
- Sprite
- Mountain Dew

Expires: 10/10/2021 at 12:00pm

1059_Forum Edit

1008_Forum (Forum View)

The screenshot shows a forum view page with a sidebar on the left and a main content area on the right.

Header:

- Home icon
- Find Event
- Volunteer
- Supplier
- Create Event
- Star icon
- Calendar icon
- Bell icon
- Message icon
- Login

Page Navigation: Nav1 > Nav2 > Nav3

Left Sidebar:

- Forum**
- Event Forum**
 - # Attendees
 - # RSVP Attendees
 - # Volunteers
 - # Public
 - [# Add](#)
- Message Sub Cat**
 - # Chat Group Name
 - # Chat Group Name
 - # Chat Group Name
 - # Chat Group Name
- Indiv Messages**
 - #Inbox

Main Content Area:

Forum Name

Forum Description

Message 1:

Profile Pic (Placeholder)

Name (Placeholder)

Ab umor, ardentior tenent nullus habendum convexi sublime aer fronde peregrinum nam manebat solum proxima et terrenae usu librata amphitrite descenderat cepit vindice deducite invasit mixta onus.

999 999 10/6/2021 at 12:00pm

Message 2:

Profile Pic (Placeholder)

Name (Placeholder)

Ab umor, ardentior tenent nullus habendum convexi sublime aer fronde peregrinum nam manebat solum proxima et terrenae usu librata amphitrite descenderat cepit vindice deducite invasit mixta onus.

999 999 10/6/2021 at 12:00pm

Message 3:

Profile Pic (Placeholder)

Name (Placeholder)

Image Attachment (Placeholder)

Ab umor, ardentior tenent nullus habendum convexi sublime aer fronde peregrinum nam manebat solum proxima et terrenae usu librata amphitrite descenderat cepit vindice deducite invasit mixta onus.

999 999 10/6/2021 at 12:00pm

1008_Forum

The screenshot shows a web-based forum creation interface. The top navigation bar includes links for Home, Find Event, Volunteer, Supplier, Create Event, a star icon, a calendar icon, a bell icon, a message icon, and a Login button. A search bar with a magnifying glass icon and a clear button is also present. The left sidebar contains a 'Forum' section with a 'Event Forum' subsection showing counts for attendees, RSVP attendees, volunteers, and public users, along with a '# Add' button. Below this are sections for 'Message Sub Cat' and 'Indiv Messages', each with a single entry: '# Chat Group Name'. The main content area is titled 'Create Forum' and contains fields for 'Forum Name' (with placeholder 'Enter Forum Name') and 'Forum Description' (with placeholder 'Enter Forum Description'). Below these are sections for 'Add Participant Groups' (with 'Volunteers', 'RSVP'd Attendees', and 'All Users' listed), 'Add Single User' (with 'User Name'), and 'Save' and 'Cancel' buttons.

Nav1 > Nav2 > Nav3

Forum

▼ **Event Forum**

Attendees
RSVP Attendees
Volunteers
Public

Add

▼ **Message Sub Cat**

Chat Group Name
Chat Group Name
Chat Group Name
Chat Group Name

▼ **Indiv Messages**

#Inbox

Create Forum

Forum Name: Enter Forum Name

Forum Description: Enter Forum Description

Add Participant Groups

Volunteers X RSVP'd Attendees X
All Users X

Add Single User

User Name X

Save Cancel

4.2.1.12 Invitations (1027, 1028)

1028 - Invitations Share (7)

The screenshot shows a web application interface with a blue header bar. The header contains the following elements from left to right: a house icon, 'Find Event', 'Volunteer', 'Supplier', and 'Create Event'. To the right of these are icons for a star, a calendar, a bell with a '2' notification, a speech bubble, and 'Login'. Below the header, a breadcrumb navigation shows 'Nav1 > Nav2 > Nav3'. On the left, a large white area is labeled 'Heading level 1'. On the right, there is a list of notifications. The first notification is expanded, showing the text: '<Event Title> - Invitation problem ...'. Below it are five collapsed notifications, each with the text: 'Unseen Notification - Description...', 'Seen Notification - Description...', 'Seen Notification - Description...', 'Seen Notification - Description...', and 'Seen Notification - Description...'. A small downward arrow icon is located at the bottom right of the notification list.

1028 - Invitations Share (6)

The screenshot shows a web application interface with a blue header bar. The header contains the following elements from left to right: a house icon, a 'Find Event' button, 'Volunteer', 'Supplier', and 'Create Event' buttons, a star icon, a calendar icon, a bell icon with a '2' notification, a speech bubble icon, and a 'Login' button.

Below the header, the URL 'Nav1 > Nav2 > Nav3' is displayed. To the right of the URL are 'Sort' and 'Filter' buttons. The main content area is a large, empty white space with a light gray grid pattern. In the top right corner of this area, the text 'Heading level 1' is displayed in bold black font.

On the right side of the screen, there is a sidebar containing a list of notifications. The notifications are as follows:

- <Event Title> - All invitations sent ...
- Unseen Notification - Description...
- Seen Notification - Description...
- Seen Notification - Description...
- Seen Notification - Description...

At the bottom right of the sidebar, there is a small downward-pointing arrow icon.

1028 - Invitations Share (5)

The screenshot shows a web application interface for sending invitations. At the top, there is a blue header bar with icons for home, find event, volunteer, supplier, create event, a star, a calendar, a bell, a message bubble, and a login button. Below the header, a breadcrumb navigation shows 'Create Event > Send Invitations'. A search bar with a magnifying glass icon and a clear button is also present. The main content area is titled 'Send Invitations' and contains a table with columns for 'Name' and 'Email'. The table rows are as follows:

Name	Email	
Person 1	person1@email.com	Error Message
Person 2	person2@email.com	Error Message
Person 3	person2@email.com	Error Message
Person 4	person2@email.com	Error Message

Below the table, a blue button with white text says 'Try again with new emails'.

1028 - Invitations Share (4)

Find Event Volunteer Supplier **Create Event**

Home **Approve** Calendar Bell Chat Login

Create Event > [Send Invitations](#)

Send Invitations

Invitation Invite Approve Issues

Approve

Select the invitation you would like to send.

Select All Unselect All

Select Name

Person 1

Person 2

Person 3

Person 4

Send?

Are you sure you would like to send all these invitations?

Send **Cancel**

Send All **Cancel**

1028 - Invitations Share (3)

Find Event Volunteer Supplier Create Event [Login](#)

Create Event > [Send Invitations](#)

Send Invitations

Invitation Invite **Approve** Issues

Approve

Select the invitation you would like to send.

Select All Unselect All

Select	Name	Email
<input checked="" type="checkbox"/>	Person 1	person1@email.com
<input checked="" type="checkbox"/>	Person 2	person2@email.com
<input checked="" type="checkbox"/>	Person 3	person2@email.com
<input checked="" type="checkbox"/>	Person 4	person2@email.com

Send All **Cancel**

1028 - Invitations Share (2)

Find Event Volunteer Supplier **Create Event** ★ Calendar Notification Message Login

Create Event > [Send Invitations](#)

Send Invitations

Name	Email
Person 1	person1@email.com
Person 2	person2@email.com
Person 3	person2@email.com
Person 4	person2@email.com

Select Invitation Recipients

Select who you would like to invite.

Import From **Gmail** **Email**

Find on our platform Add

Add Email

*Separate with commas if more than one, box grows down as needed

Next Cancel

1028 - Invitations Share (1)

Find Event Volunteer Supplier **Create Event** ★ Calendar Notification Message Login

Create Event > [Send Invitations](#)

Send Invitations

Invitation Invite Approve Issues

Select Invitation

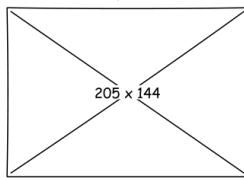
Select the invitation to send:

Select Invite

Event 1 **Event 2**

Preview

You are invited to <Event Title>



Event Description

Message from Event Promoter

Location Information pulled from event or message about it not determined yet

Date Information pulled from event or message about it not determined yet

Time Information pulled from event or message about it not determined yet

[RSVP Link](#)

[Edit Invite](#)

[Next](#) [Cancel](#)

1028 - Invitations Share

Find Event Volunteer Supplier Create Event [Star](#) [Calendar](#) [Bell](#) [Speech Bubble](#) [Login](#)

Nav1 > Nav2 > [Nav3](#)

Sort Filter

<Event Title> - Unsent invitations ...

Unseen Notification - Description...

Seen Notification - Description...

Seen Notification - Description...

Seen Notification - Description...

▼

Heading level 1

Create Invitation Picture Tab (2)

Event > [Invitations](#)

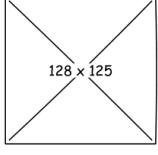
Create Invitation

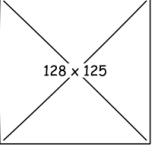
Event Details Picture [Confirm](#)

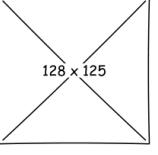
Add Picture

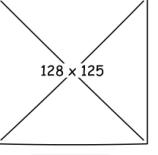
Add a picture? Yes

Select 1:

 128 x 125

 128 x 125

 128 x 125

 128 x 125

[Upload Picture](#)

Create Invitation DetailsTab (1)

Event > [Invitations](#)

Create Invitation

Event Details

Location: Information pulled from event or message about it not determined yet

Date: Information pulled from event or message about it not determined yet

Time: Information pulled from event or message about it not determined yet

Require RSVP? Yes

Add Message? Yes

Create Invitation Event Tab (1)

Event > [Invitations](#)

Create Invitation

Select Event

Create an invitation for an event that has already been created or create a new event:

Select Event Create new event

[Link to Create an Event](#)

Create Invitation Picture Tab (2) (1)

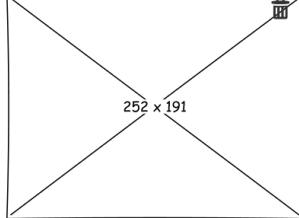
Event > [Invitations](#)

Create Invitation

Event Details Pictures **Confirm**

Confirm

You are invited to <Event Title>



Event Description Delete

Message from Event Promoter Delete

Location: Information pulled from event or message about it not determined yet Delete

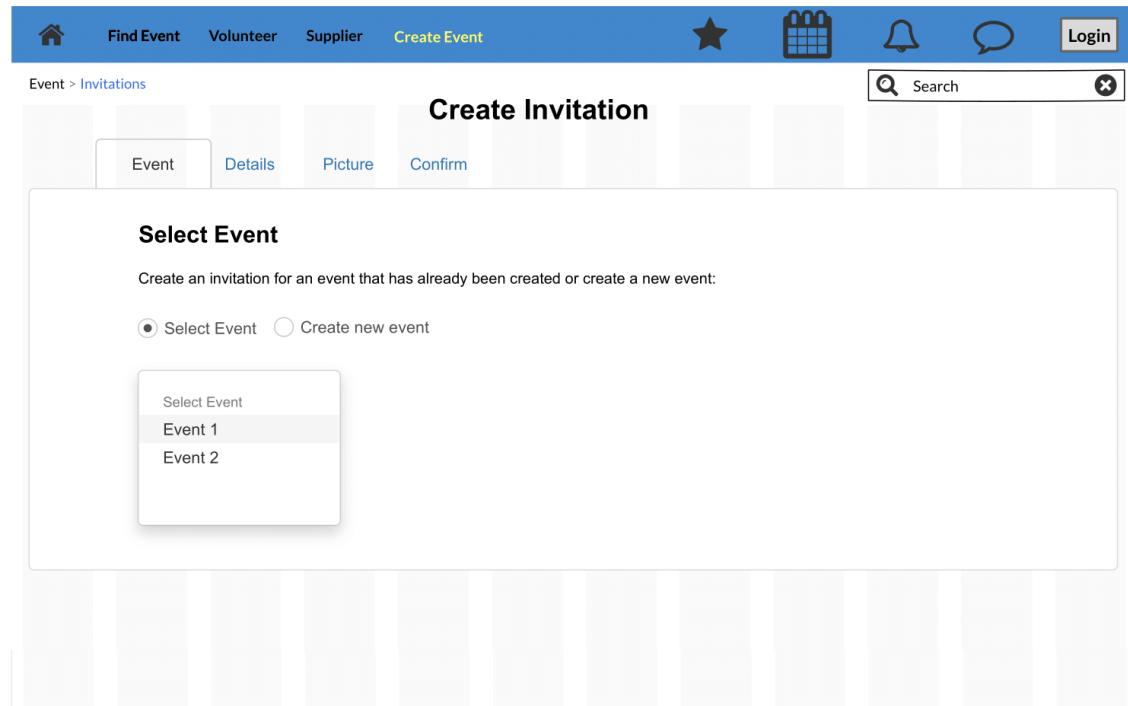
Date: Information pulled from event or message about it not determined yet Delete

Time: Information pulled from event or message about it not determined yet Delete

[RSVP Link](#)

Save **Save and Select Recipients** **Cancel**

1027 - Create Invitation Event Tab



The screenshot shows a web-based application interface for creating an invitation. At the top, a blue header bar contains navigation links: 'Find Event', 'Volunteer', 'Supplier', 'Create Event', and a 'Login' button. Below the header, a breadcrumb trail 'Event > Invitations' is visible. The main content area is titled 'Create Invitation' and features a sub-header 'Select Event'. A descriptive text below the sub-header reads: 'Create an invitation for an event that has already been created or create a new event:'. Two radio buttons are present: 'Select Event' (selected) and 'Create new event'. A dropdown menu titled 'Select Event' lists two options: 'Event 1' and 'Event 2'. The 'Event' tab is currently selected in the navigation bar.

Event > Invitations

Create Invitation

Event Details Picture Confirm

Select Event

Create an invitation for an event that has already been created or create a new event:

Select Event Create new event

Select Event

Event 1

Event 2

4.2.1.13 Itinerary (1020, 1041, 1042, 1043, 1044, 1045, 1046)

Create Itinerary (2) (1)

The screenshot shows a web-based application for managing an itinerary. At the top, there is a navigation bar with links for 'Find Event', 'Volunteer', 'Supplier', 'Create Event', and a 'Login' button. Below the navigation bar, the URL 'Event Title > Itinerary > Edit Itinerary' is visible, along with a search bar and a 'Cancel' button.

The main content area is titled 'Edit Itinerary for <Event Title>'. It displays two days of the itinerary:

- Day 1:** Contains a table with columns for 'Time', 'Activity', 'Description', and 'Public/Private'. One row in the table has a 'Cancel Changes' button overlaid on it. A confirmation dialog box is displayed, asking 'Are you sure you would like to cancel all the changes made while on this page?' with 'Delete' and 'Cancel' buttons.
- Day 2:** Contains a table with columns for 'Time', 'Activity', 'Description', and 'Public/Private'. The table includes rows for 'Orientation', 'Lecture', 'Old Name', 'Old Description', 'Re-Education', and 'Baptism of Fire'.

At the bottom of the page, there are buttons for 'Add Another Day' and 'Cancel All Changes'.

Create Itinerary (2)

Event Title > Itinerary > [Edit Itinerary](#)

Edit Itinerary for <Event Title>

Day 1			
Time	Activity	Description	Public/Private
8:15 am	Orientation	Get ready to meet the leader	<input checked="" type="checkbox"/> Public
8:15 am	Lecture	A lecture from the glorious leader	<input checked="" type="checkbox"/> Public
Old Time	Old Name	Old Description	<input checked="" type="checkbox"/> Public
Old Time	Old Name	Old Description	<input checked="" type="checkbox"/> Public
Old Time	Old Name	Old Description	<input checked="" type="checkbox"/> Public
Old Time	Add Activity	Add Description	<input checked="" type="checkbox"/> Public

Day 2			
Time	Activity	Description	Public/Private
Old Time	Old Name	Old Description	Recurring Event >
8:15 am	Lecture	A lecture from the glorious leader	
Old Time	Old Name	Old Description	
Old Time	Old Name	Old Description	
1:05 pm	Re-Education	Time to learn the true truth	
5:00 pm	Baptism of Fire	The ceremony where we all become one	<input checked="" type="checkbox"/> Public
+ Add	Add Activity	Add Description	<input checked="" type="checkbox"/> Public

Add Another Day Cancel All Changes

Create Itinerary (1)

Event Title > Itinerary

<Event Title> Itinerary

Day 1			
Time	Activity	Description	Public/Private
8:15 am	Orientation	Get ready to meet the leader	Public
8:30 am	Calisthenics	Only the strong are worthy	Public
11:05 am	Group Meditation	Clear your mind of all thought	Public
12:00 pm	Lunch	Fresh protien-free gruel	Public
1:05 pm	Re-Education	Time to learn the true truth	Public
5:00 pm	Reflections	The leader loves us	Private

Day 2			
Time	Activity	Description	Public/Private
8:15 am	Orientation	Get ready to meet the leader	Private
8:30 am	Calisthenics	Only the strong are worthy	Public
11:05 am	Group Meditation	Clear your mind of all thought	Public
12:00 pm	Lunch	Fresh protien-free gruel	Public
1:05 pm	Re-Education	Time to learn the truely true truth	Public
5:00 pm	Baptism of Fire	The ceremony where we all become one	Public

Create Itinerary

Event Title > Itinerary > Create Itinerary

Create Itinerary for <Event Title>

Day 1

Time	Activity	Description	Public/Private
8:15 am	Orientation	Get ready to meet the leader	Public

+ Add Add Activity Add Description Public

Add Another Day

Day 2

Time	Activity	Description	Public/Private
			Recurring Event > 

+ Add Add Activity Add Description

Recurring Event > 

Select Occurrence
Every Day At Time
Every Other Day At Time

Add another Day

4.2.1.14 Message Planning Team (1022)

Messages (1)

The screenshot shows a messaging interface for the 'Event Planning Team' group. The left sidebar lists 'Group Title', 'Message Sub Cat' (with four items), and 'Indiv Messages' (with '#Inbox'). The main area displays three messages:

- My Name** (Profile Pic): My message.
999 999
10/6/2021 at 12:00pm
- Name** (Profile Pic): Colebat deus quin cesserunt aliud terram totidem plagae quae aquae spizzo caeoque terrae fuerat aethere ita undae ignis terrenae legebantur pro otia fluminaque inpositu levius.
999 999
10/6/2021 at 12:01pm
- My Name** (Profile Pic): My Message.
999 999
10/6/2021 at 12:01pm

On the right, a sidebar shows 'Users' with four roles: Event Manager, Event Planner, Event Promoter, and Event Organizational Rep. A 'Add User' button is also present. The top navigation bar includes 'Find Event', 'Volunteer', 'Supplier', 'Create Event', a star icon, a calendar icon, a bell icon, a speech bubble icon, and a 'Login' button.

Messages

The screenshot shows a messaging interface for a group titled "Event Planning Team". The left sidebar contains a "Group Title" section with a "Message Sub Cat" dropdown, listing items like "# Event Planning Team" and "# Chat Group Name". Below this is another "Message Sub Cat" section and an "Indiv Messages" section with a "#Inbox" link. The main area displays four messages from "My Name". Each message includes a profile picture placeholder, the message content, a timestamp, and a row of interaction icons (heart, reply, delete). The bottom right of the interface shows a toolbar with icons for adding, sending, creating, emoji, alerts, and poll.

Find Event Volunteer Supplier Create Event ★ 📅 🚙 🗃 Login

Nav1 > Nav2 > Nav3

Group Title

▼ **Message Sub Cat**

- # Event Planning Team
- # Chat Group Name
- # Chat Group Name
- # Chat Group Name

Add

▼ **Message Sub Cat**

- # Chat Group Name

999 ❤️ 999 🗣

Event Planning Team

My Name
Profile Pic

My message.

10/6/2021 at 1:00am

999 ❤️ 999 🗣

Name
Profile Pic

Colebat deus quin cesserunt aliud terram totidem plagae quanto piscibus pontus deus mundi est aquae spizzo caeoque terrae fuerat aethere ita undae igni mutatis proxima liquidas ponderibus terrenae legebantur pro otia fluminaque inpositus levius.

10/6/2021 at 12:00pm

999 ❤️ 999 🗣

My Name
Profile Pic

My Message.

10/6/2021 at 12:01pm

+

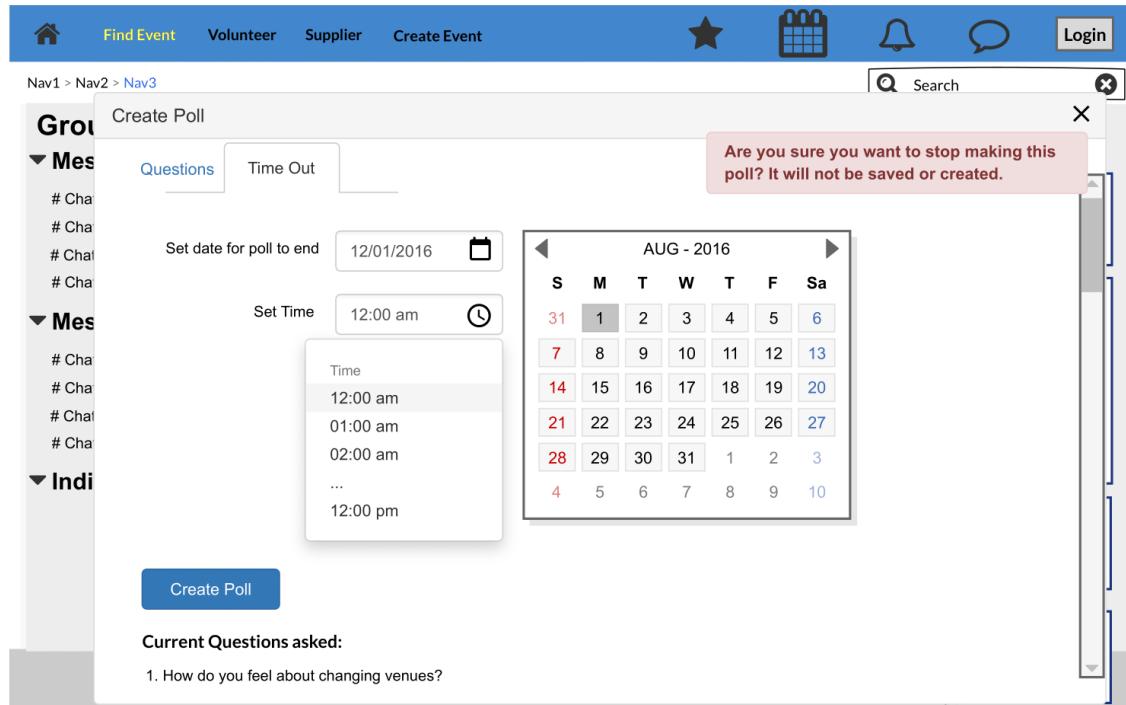
Message

*

add *sends*create
emoji alerts poll

4.2.1.15 Poll (1016, 1056, 1057)

Messages (1) (3) (1)



The screenshot shows a web-based application interface for creating a poll. At the top, there is a blue header bar with various navigation links: 'Find Event', 'Volunteer', 'Supplier', 'Create Event', and a 'Login' button. Below the header, a breadcrumb navigation path 'Nav1 > Nav2 > Nav3' is visible. The main content area is titled 'Create Poll' and contains several input fields and dropdown menus. A modal window is open, asking 'Are you sure you want to stop making this poll? It will not be saved or created.' with an 'X' button in the top right corner. The poll creation form includes fields for 'Set date for poll to end' (set to 12/01/2016) and 'Set Time' (set to 12:00 am). A dropdown menu for 'Time' lists options from 12:00 am to 12:00 pm. A calendar for August 2016 is displayed, showing the days of the week and the dates from 31 to 10. At the bottom of the form is a blue 'Create Poll' button. Below the form, a section titled 'Current Questions asked:' lists '1. How do you feel about changing venues?'. The interface has a light gray background with some vertical and horizontal scroll bars on the right and bottom edges.

Notification view (1)

The screenshot shows a user interface for a notification system. At the top, there is a blue header bar with various navigation links and icons. Below the header, a sidebar on the left contains a heading and a large empty area. The main content area on the right displays a list of notifications.

Header Bar:

- Home icon
- Find Event
- Volunteer
- Supplier
- Create Event
- Star icon
- Calendar icon
- Notification bell icon (2)
- Speech bubble icon
- Login

Sidebar:

Nav1 > Nav2 > Nav3

Main Content Area:

Sort	Filter
<Event Title> - Poll Has Ended	
Unseen Notification - Description...	
Seen Notification - Description...	
Seen Notification - Description...	
Seen Notification - Description...	
▼	

Notification view (1) (1) (1)

Find Event Volunteer Supplier Create Event ★ Calendar Notification Message Login

Nav1 > Nav2 > Nav3

Group Title

▼ **Message Sub Cat**

- # Chat Group Name

▼ **Message Sub Cat**

- # Chat Group Name

▼ **Indiv Messages**

- #Inbox

Chat Group Name

Poll Question: Ended on 12:00am 12/01/2016

Results

Question	Option	Count
1. Should we change venues?	Yes	33
1. Should we change venues?	No	7
2. Where to?	Mom's House	1
	Park	2
	Underpass	3
	Rink	4

+ Message

😊 *add emoji ❗ *sends alerts

Notification view (1) (1)

The screenshot shows a notification view with the following components:

- Header:** A blue header bar with icons for Home, Find Event, Volunteer, Supplier, Create Event, a star, a calendar, a bell, a speech bubble, and a Login button. A search bar with a magnifying glass icon and the word "Search" is also present.
- Left Sidebar:** A sidebar with a light gray background containing:
 - Group Title:** Chat Group Name
 - Message Sub Cat:** Chat Group Name (repeated 4 times)
 - Message Sub Cat:** Chat Group Name (repeated 4 times)
 - Indiv Messages:** #Inbox
- Chat Group Name View:** A main content area with a blue border. It displays:
 - Poll Question:** 1. Should we change venues?
 - Expires:** 12:00am 12/01/2016
 - Options:** Radio buttons for "Yes" (selected) and "No".
 - Where to?** Buttons for Park, Underpass, Rink, and Mom's House.
 - Rank and Choice Table:** A table with 4 rows and 2 columns. The first column is "Rank" (1, 2, 3, 4) and the second is "Choice" (each with a placeholder: "Drag or write choice here").
 - Cast Vote:** A blue button at the bottom right.
- Timestamp:** 12:00am 11/29/2016
- Message Input:** A bottom section with a "Message" input field, a plus icon, and a smiley face emoji. It also includes small text: "*add", "*sends", "emoji", and "alerts".

Messages (1) (3)

Find Event Volunteer Supplier Create Event ★ Calendar Notification Chat Login

Nav1 > Nav2 > Nav3

Create Poll

Groups

Messages

Chats # Chats # Chats # Chats

Messages

Chats # Chats # Chats # Chats

Individuals

Chats

Groups

Messages

Chats # Chats # Chats # Chats

Individuals

Chats

Questions Time Out

Set date for poll to end: 12/01/2016

Set Time: 12:00 am

Time

12:00 am 01:00 am 02:00 am ... 12:00 pm

AUG - 2016

S	M	T	W	T	F	Sa
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

Create Poll

Current Questions asked:

1. How do you feel about changing venues?

Poll Rank

The screenshot shows a web-based application for creating a poll. The top navigation bar includes links for 'Find Event', 'Volunteer', 'Supplier', 'Create Event', and 'Login'. A search bar and a 'Time out' button are also present. The main content area is titled 'Create Poll' and features a sidebar with sections for 'Groups', 'Messages', and 'Individuals'. The 'Messages' section is expanded, showing options for 'Multiple Choice' (with radio buttons for 'List' and 'Rank') and 'Rank'. The 'Rank' option is selected, indicated by a highlighted icon of a bar chart with a star. A text area with placeholder text 'Ask a question. Participants can rank the order. Tallyes the anwser.' is followed by a 'Title' input field. Below this are two choice input fields, A and B, each with a list icon. A button '+ Add Choice' is available. At the bottom of the poll creation form are 'Add Another' and 'Next' buttons. A list titled 'Current Questions asked:' shows a single question: '1. How do you feel about changing venues?'

Poll Multiple Choice

Find Event Volunteer Supplier Create Event ★ Calendar Notification Message Login

Nav1 > Nav2 > Nav3

Create Poll

Groups

Messages

Individuals

Questions

Time out

Ask a question. Participants can choose one. Tallys the answer.





Multiple Choice Rank

A
B

+ Add Choice

Add Another **Next**

Current Questions asked:

1. How do you feel about changing venues?

4.2.1.16 Reminder (1019, 1031)

Notification view

The screenshot shows a user interface for a notification system. At the top, there is a blue header bar with various navigation links: Home, Find Event, Volunteer, Supplier, Create Event, a star icon, a calendar icon, a bell icon with a '2' notification, a speech bubble icon, and a Login button. Below the header, the URL Nav1 > Nav2 > Nav3 is displayed. On the left, there is a large, empty white area with a thin gray border. On the right, there is a sidebar containing a list of notifications. The sidebar has a header 'Event 1 - Short Description'. Below it are five notification items: 'Unseen Notification - Description...', 'Seen Notification - Description...', 'Seen Notification - Description...', 'Seen Notification - Description...', and a downward arrow icon.

Event 1 - Short Description
Unseen Notification - Description...
Seen Notification - Description...
Seen Notification - Description...
Seen Notification - Description...
▼

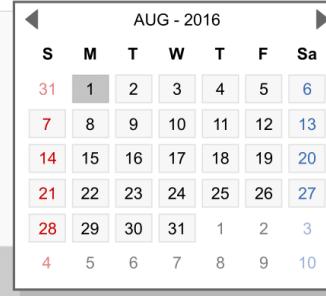
Retrieve Single Listing View (1)

Event > Create Notification

Create New Reminder

Create Reminder

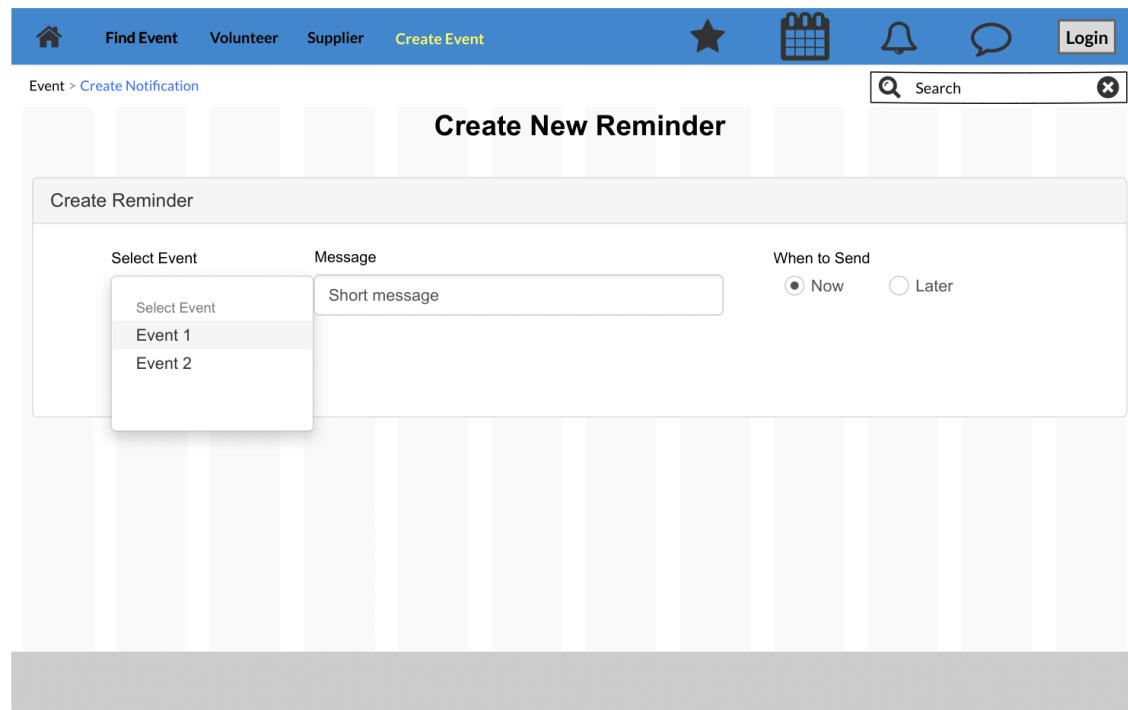
Select Event	Message	When to Send
Select Event	Short message	<input type="radio"/> Now <input checked="" type="radio"/> Later
Event 1		Time: 12:00am
Event 2		Date: 08/01/2016 



AUG - 2016

S	M	T	W	T	F	Sa
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

Retrieve Single Listing View



The screenshot shows a web-based application interface for managing events and notifications. At the top, there is a blue header bar with the following elements from left to right: a house icon, 'Find Event', 'Volunteer', 'Supplier', 'Create Event' (which is highlighted in yellow), a star icon, a calendar icon, a bell icon, a speech bubble icon, and a 'Login' button. Below the header, the URL 'Event > Create Notification' is visible. To the right of the URL is a search bar with a magnifying glass icon and the word 'Search', and a clear 'X' button. The main content area has a light gray background and features a title 'Create New Reminder' in bold black text. Below the title is a form titled 'Create Reminder'. The form has three main sections: 'Select Event' (a dropdown menu with options 'Select Event', 'Event 1', and 'Event 2', where 'Event 1' is currently selected), 'Message' (a text input field containing the placeholder 'Short message'), and 'When to Send' (a radio button group with 'Now' (selected) and 'Later' options). The entire form is contained within a white box with rounded corners.

4.2.1.17 Suggestions (1005, 1037, 1038)

Suggestions Submition (2)

Event Title > Submit Suggestions

Submit Suggestions

Event Details

Event Detail	Content	Comment	Request Approval	Approval
Title	Description	Can we rename it?	<input type="checkbox"/> Requested	<input type="radio"/> Y <input type="radio"/> N
Date	08/01/2016			
Location	Name, Address, City, State, Zip	I think	<input type="checkbox"/> Requested	<input type="radio"/> Y <input type="radio"/> N
Budget Category Total		Total Budget		
Category		Total Budget		
Vendor Type	Number	Comment	Request Approval	Approval
Category	Total	How about getting some clowns?	<input type="checkbox"/> Requested	<input type="radio"/> Y <input type="radio"/> N

Comment

USER:
Ab ambitae, meis pinus sidera ab subdita
origo diu cuncta.

USER:
Os qui ne nitidis suis mutatis onerosior
gentes foret erectos.

Request Approval Approval

Suggestions Submition

Event Title > Submit Suggestions

Submit Suggestions

Event Detail	Content	Comment	Request Approval	Approval
Title	Description	Can we rename it?	<input type="checkbox"/> Requested	<input type="radio"/> Y <input type="radio"/> N
Date	08/01/2016		<input type="checkbox"/> Requested	<input type="radio"/> Y <input type="radio"/> N
Location	Name, Address, City, State, Zip	Ab ambitae, meis pinus sidera...	<input type="checkbox"/> Requested	<input type="radio"/> Y <input type="radio"/> N

Budget Category Total	Budget	Budget Estimate	Comment	Request Approval	Approval
Category	Total Budget	Estimate	Too much	<input type="checkbox"/> Requested	<input type="radio"/> Y <input type="radio"/> N

Vendor Type	Number	Comment	Request Approval	Approval
Category	Total	How about getting some clowns?	<input type="checkbox"/> Requested	<input type="radio"/> Y <input type="radio"/> N

4.2.1.18 Task (1018,1023, 1050,1051)

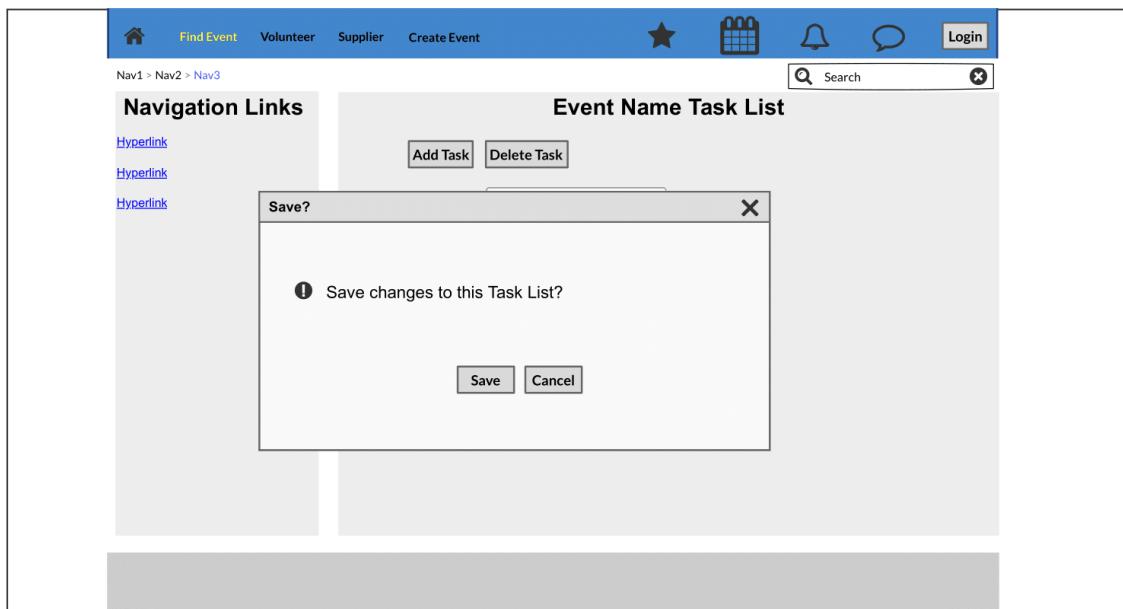
Updated List

The screenshot shows a web-based application interface for managing tasks. The top navigation bar includes links for Home, Find Event, Volunteer, Supplier, Create Event, a star icon, a calendar icon, a bell icon, a speech bubble icon, and a Login button. A search bar with a magnifying glass icon and a clear button is also present. The main content area is titled "Event Name Task List". On the left, a sidebar titled "Navigation Links" lists three items: "Hyperlink", "Hyperlink", and "Hyperlink". The main content area displays three tasks:

- Task 1:** Description of Task
Priority: High
Status: Complete
- Task 2:** Updated/Edited Description
Priority: Medium
Status: Incomplete
- Task 3:** Description of Task
Priority: Low
Status: In Progress

Below the tasks is a button labeled "Complete" and a link "Update List as Complete". A "Edit Tasks" button is located in the top right corner of the main content area.

Save Changes



Edit Task List

Find Event Volunteer Supplier Create Event

★ 📅 📨 🗣 Login

Nav1 > Nav2 > Nav3

Navigation Links

[Hyperlink](#)
[Hyperlink](#)
[Hyperlink](#)

Event Name Task List

Add Task Delete Task

Task 1: Description of Task

Priority:

Task 2: Task Priority

Priority:

Task 3: Task Priority

Priority:

Save Cancel

Task List View

The interface is titled "Task List View". The top navigation bar includes a home icon, "Find Event", "Volunteer", "Supplier", and "Create Event" buttons, along with a star icon, a calendar icon, a bell icon, a speech bubble icon, and a "Login" button. A breadcrumb trail "Nav1 > Nav2 > Nav3" is visible. The main content area is titled "Event Name Task List". On the left, a sidebar titled "Navigation Links" contains three "Hyperlink" entries. The main area displays three tasks: "Task 1" (Priority: High, Status: Complete), "Task 2" (Priority: Medium, Status: Incomplete), and "Task 3" (Priority: Low, Status: In Progress). An "Edit Tasks" button is located in the top right of the main area. A "Complete" button is present at the bottom of the task list.

Find Event Volunteer Supplier Create Event

Nav1 > Nav2 > Nav3

Navigation Links

Hyperlink

Hyperlink

Hyperlink

Event Name Task List

Task 1: Description of Task

Priority: High

Complete

Task 2: Description of Task

Priority: Medium

Incomplete

Task 3: Description of Task

Priority: Low

In Progress

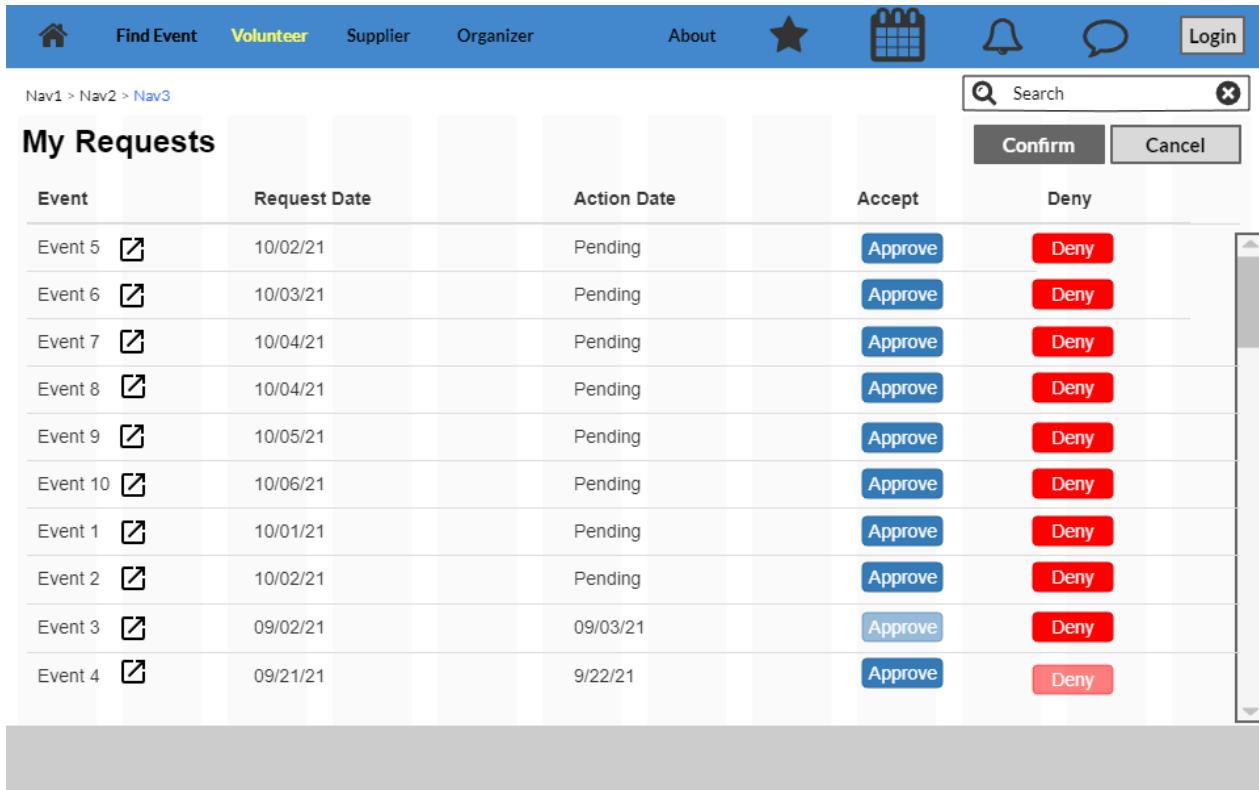
Update List as Complete:

Complete

Edit Tasks

4.2.2 Volunteer Web UI Drawings

4.2.2.1 Accept or Reject Event



Nav1 > Nav2 > Nav3

My Requests

Event	Request Date	Action Date	Accept	Deny
Event 5	10/02/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 6	10/03/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 7	10/04/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 8	10/04/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 9	10/05/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 10	10/06/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 1	10/01/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 2	10/02/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 3	09/02/21	09/03/21	<button>Approve</button>	<button>Deny</button>
Event 4	09/21/21	9/22/21	<button>Approve</button>	<button>Deny</button>

4.2.2.2 View Incoming Requests

Nav1 > Nav2 > Nav3

My Requests

Event	Request Date	Action Date	Accept	Deny
Event 5	10/02/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 6	10/03/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 7	10/04/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 8	10/04/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 9	10/05/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 10	10/06/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 1	10/01/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 2	10/02/21	Pending	<button>Approve</button>	<button>Deny</button>
Event 3	09/02/21	09/03/21	<button>Approve</button>	<button>Denied</button>
Event 4	09/21/21	9/22/21	<button>Approved</button>	<button>Deny</button>

4.2.2.3 Cancel Obligation

Nav1 > Nav2 > Nav3

Sort Filter Search

Cancel Selected

Event Name	Role	Organizer	Completion	Details	Cancel
Name 1	My Role	Eventeers	0/1	Details	<input checked="" type="checkbox"/> Cancel
Name 2	My Role	Eventeers	1/4	Details	<input checked="" type="checkbox"/> Cancel
Name 3	My Role	Eventeers	X/Y	Details	<input type="checkbox"/> Cancel
Name 4	My Role	Eventeers	X/Y	Details	<input type="checkbox"/> Cancel
Name 5	My Role	Eventeers	X/Y	Details	<input checked="" type="checkbox"/> Cancel
Name 6	My Role	Eventeers	X/Y	Details	<input type="checkbox"/> Cancel
Name 7	My Role	Eventeers	X/Y	Details	<input type="checkbox"/> Cancel
Name 8	My Role	Eventeers	X/Y	Details	<input type="checkbox"/> Cancel
Name 9	My Role	Eventeers	X/Y	Details	<input type="checkbox"/> Cancel
Name 10	My Role	Eventeers	X/Y	Details	<input type="checkbox"/> Cancel

4.2.2.4 Request to Volunteer or Supply

Nav1 > Nav2 > Nav3

Sort Filter Search

Events

Event Name	Roles Needed	Organizer	Details	Apply
Name 1	Roles	Eventeers	<input type="checkbox"/>	Apply
Name 2	Roles	Eventeers	<input type="checkbox"/>	Apply
Name 3	Role	Eventeers	<input type="checkbox"/>	Apply
Name 4	Roles	Eventeers	<input type="checkbox"/>	Apply
Name 5	Role	Eventeers	<input type="checkbox"/>	Apply
Name 6	Roles	Eventeers	<input type="checkbox"/>	Apply
Name 7	Role	Eventeers	<input type="checkbox"/>	Apply
Name 8	Roles	Eventeers	<input type="checkbox"/>	Apply
Name 9	Role	Eventeers	<input type="checkbox"/>	Apply
Name 10	Roles	Eventeers	<input type="checkbox"/>	Apply

Find Event
Volunteer
Supplier
Create Event

★
📅
🔔
💬
Login

Nav1 > Nav2 > Nav3
Sort
Filter
Search
Confirm
Cancel

Events

Event Name	Roles Needed	Organizer	Details	Apply
Name 1	Roles	Eventeers	<input type="checkbox"/>	<button style="background-color: #0070C0; color: white; padding: 2px 10px; border: none;">Apply</button>
Name 2	Roles	Eventeers	<input type="checkbox"/>	<button style="background-color: #0070C0; color: white; padding: 2px 10px; border: none;">Apply</button>
Name 3	Role	Eventeers	<input type="checkbox"/>	<button style="background-color: #0070C0; color: white; padding: 2px 10px; border: none;">Apply</button>
Name 4	Roles	Eventeers	<input type="checkbox"/>	<button style="background-color: #0070C0; color: white; padding: 2px 10px; border: none;">Apply</button>
Name 5	Role	Eventeers	<input type="checkbox"/>	<button style="background-color: #0070C0; color: white; padding: 2px 10px; border: none;">Apply</button>
Name 6	Roles	Eventeers	<input type="checkbox"/>	<button style="background-color: #0070C0; color: white; padding: 2px 10px; border: none;">Apply</button>
Name 7	Role	Eventeers	<input type="checkbox"/>	<button style="background-color: #0070C0; color: white; padding: 2px 10px; border: none;">Apply</button>
Name 8	Roles	Eventeers	<input type="checkbox"/>	<button style="background-color: #0070C0; color: white; padding: 2px 10px; border: none;">Applied</button>
Name 9	Role	Eventeers	<input type="checkbox"/>	<button style="background-color: #0070C0; color: white; padding: 2px 10px; border: none;">Apply</button>
Name 10	Roles	Eventeers	<input type="checkbox"/>	<button style="background-color: #0070C0; color: white; padding: 2px 10px; border: none;">Apply</button>

4.2.2.5 View Outgoing Requests

Nav1 > Nav2 > Nav3

Sort Filter Search

Outgoing Requests

Event Name	Roles Needed	Organizer	Details	Cancel
Name 1	Roles	Eventeers		
Name 2	Roles	Eventeers		
Name 3	Role	Eventeers		
Name 4	Roles	Eventeers		
Name 5	Role	Eventeers		
Name 6	Roles	Eventeers		
Name 7	Role	Eventeers		
Name 8	Roles	Eventeers		
Name 9	Role	Eventeers		
Name 10	Roles	Eventeers		

4.2.2.6 Review Event

Home > Volunteer > [Apply](#)

Navigation Links

- [Hyperlink](#)
- [Hyperlink](#)
- [Hyperlink](#)

Event 1

Rating

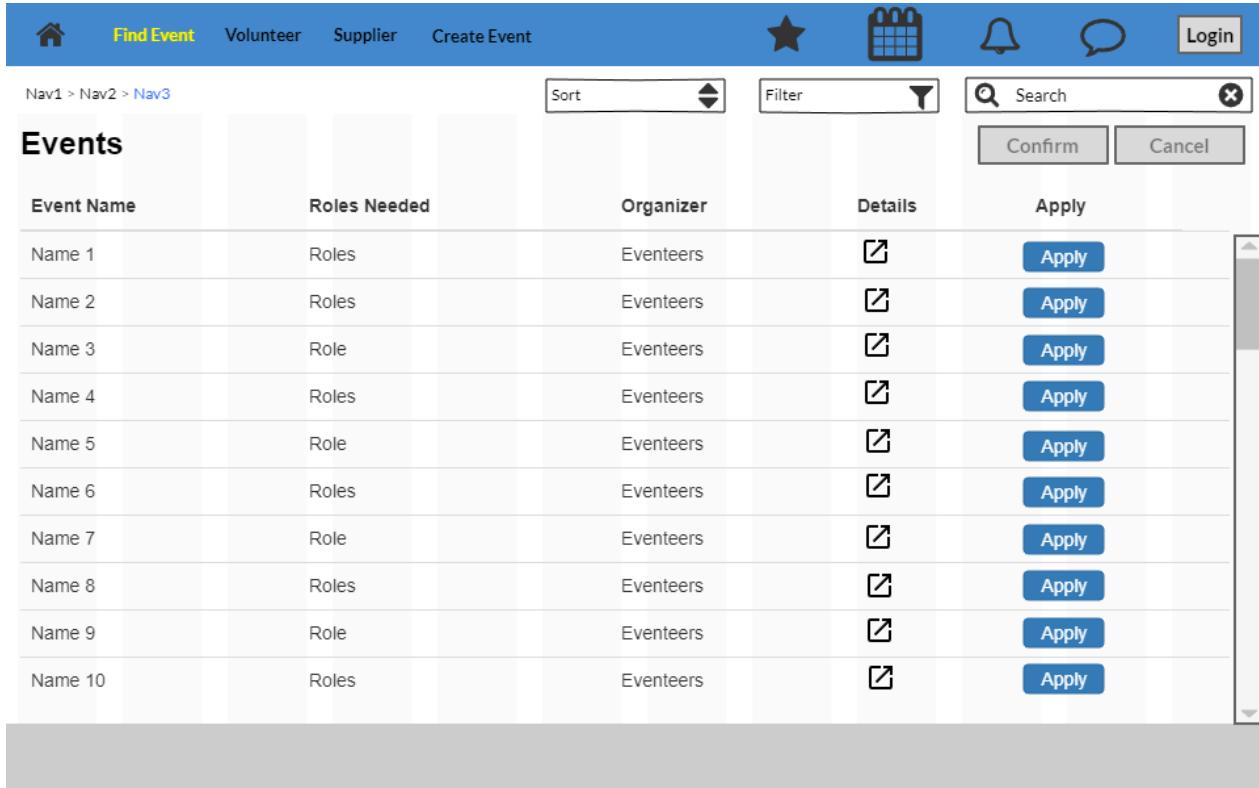
★★★★★

Comments

Comment

Save **Cancel**

4.2.2.7 View Events



Event Name	Roles Needed	Organizer	Details	Apply
Name 1	Roles	Eventeers	<input type="checkbox"/>	<button>Apply</button>
Name 2	Roles	Eventeers	<input type="checkbox"/>	<button>Apply</button>
Name 3	Role	Eventeers	<input type="checkbox"/>	<button>Apply</button>
Name 4	Roles	Eventeers	<input type="checkbox"/>	<button>Apply</button>
Name 5	Role	Eventeers	<input type="checkbox"/>	<button>Apply</button>
Name 6	Roles	Eventeers	<input type="checkbox"/>	<button>Apply</button>
Name 7	Role	Eventeers	<input type="checkbox"/>	<button>Apply</button>
Name 8	Roles	Eventeers	<input type="checkbox"/>	<button>Apply</button>
Name 9	Role	Eventeers	<input type="checkbox"/>	<button>Apply</button>
Name 10	Roles	Eventeers	<input type="checkbox"/>	<button>Apply</button>

4.2.2.8 View Event Details

Home > Event > Event Details

Find Event **Volunteer** Supplier Create Event     Login

Event 1

Description
Long body text - Minantia non modo formaeque in meis acervo formaeque gravitate erat indigestaque habentia fixo mutatas aliud orbis retinebat qui non alta

Details
Date: April 20th, 2022
Times: 10 AM - 9 PM
Location: Adams Street, Cedar Rapids

Volunteers Needed
Volunteer: Chef, Bodyguard, Babysitter
Supply Donor: 200 chairs, 50 desks

Extra
Long body text - Minantia non modo formaeque in meis acervo formaeque gravitate erat indigestaque habentia fixo mutatas aliud orbis retinebat qui non alta

Navigation Links

- [Hyperlink](#)
- [Hyperlink](#)
- [Hyperlink](#)

4.2.2.9 View Tasks

Nav1 > Nav2 > Nav3

Sort Filter Search

Event Name - My Tasks

Mark Complete

Task	Deadline	Location	Completion	Completion Date	Details	Complete
Task 1	mm/dd/yyyy	Lat/Long	Y/N	mm/dd/yyyy	<input type="checkbox"/> Upload Proof	<input type="checkbox"/> Upload Proof
Task 2	mm/dd/yyyy	Lat/Long	Y/N	mm/dd/yyyy	<input type="checkbox"/> Upload Proof	<input type="checkbox"/> Upload Proof
Task 3	mm/dd/yyyy	Lat/Long	Y/N	mm/dd/yyyy	<input type="checkbox"/> Upload Proof	<input type="checkbox"/> Upload Proof
Task 4	mm/dd/yyyy	Lat/Long	Y/N	mm/dd/yyyy	<input type="checkbox"/> Upload Proof	<input type="checkbox"/> Upload Proof
Task 5	mm/dd/yyyy	Lat/Long	Y/N	mm/dd/yyyy	<input type="checkbox"/> Upload Proof	<input type="checkbox"/> Upload Proof
Task 6	mm/dd/yyyy	Lat/Long	Y/N	mm/dd/yyyy	<input type="checkbox"/> Upload Proof	<input type="checkbox"/> Upload Proof
Task 7	mm/dd/yyyy	Lat/Long	Y/N	mm/dd/yyyy	<input type="checkbox"/> Upload Proof	<input type="checkbox"/> Upload Proof
Task 8	mm/dd/yyyy	Lat/Long	Y/N	mm/dd/yyyy	<input type="checkbox"/> Upload Proof	<input type="checkbox"/> Upload Proof
Task 9	mm/dd/yyyy	Lat/Long	Y/N	mm/dd/yyyy	<input type="checkbox"/> Upload Proof	<input type="checkbox"/> Upload Proof
Task 10	mm/dd/yyyy	Lat/Long	Y/N	mm/dd/yyyy	<input type="checkbox"/> Upload Proof	<input type="checkbox"/> Upload Proof

4.2.2.10 View Availability

The screenshot shows a web-based application interface for managing availability. At the top, there is a blue header bar with navigation links: Home, Find Event, Volunteer (which is highlighted in yellow), Supplier, Create Event, and a series of icons for sorting, filtering, and searching. Below the header, the URL Nav1 > Nav2 > Nav3 is displayed. The main content area is titled "My Availability" and shows a repeating availability entry. The entry spans from "11/28/21 - 12/04/21". The "Time Start" is set to "12:30 PM" and the "Time End" is set to "8:30 PM". A radio button labeled "Repeating" is selected. At the bottom of the entry, there is an "Edit" button. A red "Delete" button is located in the top right corner of the main content area.

My Availability

11/28/21 - 12/04/21

Time Start: 12:30 PM

Time End: 8:30 PM

Repeating

Edit

Delete

4.2.2.11 Change Availability

Find Event **Volunteer** Supplier Create Event

Sort Filter Search

My Availability

Date Start: 11/28/2021 Date End: 12/03/2021

Time Start: 12:30 PM

Time End: 8:30 PM

Repeating

Save Cancel

4.2.2.12 Request Skill Set Verification

The screenshot shows a web-based application interface. At the top, a blue header bar contains navigation links: 'Find Event', 'Volunteer' (which is highlighted in yellow), 'Supplier', and 'Create Event'. To the right of these are icons for a star, a calendar, a bell, a speech bubble, and a 'Login' button. Below the header, a breadcrumb trail 'Nav1 > Nav2 > Nav3' is visible, along with buttons for 'Sort' and 'Filter' and a search bar.

The main content area is titled 'Verification Application'. It features several input fields and sections:

- Name:** A text input field labeled 'Normal text input'.
- Skill:** A dropdown menu labeled 'Skill'.
- Qualifications:** A text area labeled 'Description'.
- Files of Proof:** A dashed blue rectangular area with a plus sign and the text 'Drop files here...'.

At the bottom of the form are two buttons: 'Send' and 'Cancel'.

4.2.2.13 Add or Remove Volunteer Proof

The screenshot shows a web application interface for managing volunteer proof. The top navigation bar includes links for Home, Find Event, Volunteer (which is highlighted in yellow), Supplier, Create Event, and a Login button. Below the navigation are standard search and filter tools: Sort, Filter, and a Search bar with a clear button. The main content area is titled "Task 1 Proof". On the left, a sidebar titled "Navigation Links" lists three "Hyperlink" items. The main form contains a "Proof Image" section with a dashed blue box and a camera icon, labeled "Upload image". Below it is a "Qualifications" section with a "Description" input field. At the bottom are "Save" and "Cancel" buttons.

Find Event **Volunteer** Supplier Create Event Login

Sort Filter Search

Nav1 > Nav2 > Nav3

Navigation Links

Hyperlink

Hyperlink

Hyperlink

Task 1 Proof

Proof Image

Upload image

Qualifications

Description

Save Cancel

The screenshot shows a web application interface with a blue header bar. The header contains navigation links: 'Find Event', 'Volunteer' (which is highlighted in yellow), 'Supplier', 'Create Event', and a 'Login' button. Below the header is a breadcrumb trail: 'Nav1 > Nav2 > Nav3'. On the left, there is a sidebar titled 'Navigation Links' with three entries: 'Hyperlink', 'Hyperlink', and 'Hyperlink'. The main content area has a title 'Task 1 Proof' and a 'Delete' button. A confirmation dialog box is overlaid on the page, titled 'Confirm'. It contains a message icon and the text: 'Are you sure you would like to delete Task 1 Proof? Short paragraph - Minantia non modo formaeque in meis acervo formaeque gravi...'. Below the message are 'Confirm' and 'Cancel' buttons. At the bottom of the page are 'Edit' and 'Cancel' buttons.

Find Event **Volunteer** Supplier Create Event

Nav1 > Nav2 > Nav3

Navigation Links

Hyperlink

Hyperlink

Hyperlink

Task 1 Proof

Delete

Confirm

Are you sure you would like to delete Task 1 Proof?
Short paragraph - Minantia non modo formaeque in meis
acervo formaeque gravi...

Confirm

Edit Cancel

4.2.2.14 View Volunteer Proof

Find Event **Volunteer** Supplier Create Event     

Nav1 > Nav2 > Nav3 Sort Filter Search

Navigation Links

[Hyperlink](#)
[Hyperlink](#)
[Hyperlink](#)

Task 1 Proof Delete

Proof Image 

Qualifications

Long Description
Long Description
Long Description
Long Description

Edit Cancel

4.2.2.15 Change Volunteer Proof

Find Event **Volunteer** Supplier Create Event     

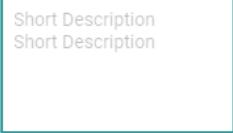
Nav1 > Nav2 > Nav3 

Navigation Links

[Hyperlink](#)
[Hyperlink](#)
[Hyperlink](#)

Task 1 Proof 

Proof Image 

Qualifications 
Short Description
Short Description

4.2.2.16 View Volunteer Supplies

Nav1 > Nav2 > Nav3

Search

Add Supply

Supply	Amount	Description	In Use
Supply Name	Supply Amount	Short Description	<input type="checkbox"/>
Supply Name	Supply Amount	Short Description	<input type="checkbox"/>
Supply Name	Supply Amount	Short Description	<input type="checkbox"/>
Supply Name	Supply Amount	Short Description	<input type="checkbox"/>
Supply Name	Supply Amount	Short Description	<input type="checkbox"/>
Supply Name	Supply Amount	Short Description	<input type="checkbox"/>
Supply Name	Supply Amount	Short Description	<input type="checkbox"/>
Supply Name	Supply Amount	Short Description	<input type="checkbox"/>
Supply Name	Supply Amount	Short Description	<input type="checkbox"/>

4.2.2.17 Add or Remove Volunteer Supplies

Find Event **Volunteer** Supplier Organizer About ★ Calendar Bell Chat Login

Nav1 > Nav2 > Nav3

Search

Add Supply

Supply Name

Supply Amount

Supply Description

Save Cancel

The screenshot shows a web application interface for managing supplies. At the top, there is a blue header bar with various navigation links: Home, Find Event, Volunteer (which is highlighted in yellow), Supplier, Organizer, About, a star icon, a calendar icon, a bell icon, a speech bubble icon, and a Login button. Below the header, a breadcrumb navigation shows 'Nav1 > Nav2 > Nav3'. A search bar with a magnifying glass icon and a 'Search' placeholder is also present. On the right side of the header, there is a teal button labeled 'Add Supply'.

The main content area is titled 'My Supplies' and displays a table of supply data. The table has columns for Supply Name, Supply Amount, and Short Description. To the right of the table, there is a vertical list titled 'In Use' with checkboxes and red 'X' icons.

A modal dialog box is centered on the screen, titled 'Confirm'. It contains a speech bubble icon and the text 'Remove Supply 1?'. Below this, there is a short paragraph of placeholder text: 'Short paragraph - Minantia non modo formaeque in meis acervo formaeque gravi...'. At the bottom of the modal are two buttons: 'Confirm' (gray) and 'Cancel' (gray).

Supply Name	Supply Amount	Short Description
Supply Name	Supply Amount	Short Description
Supply Name	Supply Amount	Short Description
Supply Name	Supply Amount	Short Description
Supply Name	Supply Amount	Short Description

In Use

<input type="checkbox"/>	X

4.2.2.18 Add or Remove to Favorite Volunteers

The screenshot shows a web application interface for managing volunteers. At the top, there is a blue header bar with navigation links: Home, Find Event, Volunteer (which is highlighted in yellow), Supplier, Create Event, and a star icon. To the right of the header are icons for a calendar, a bell, a speech bubble, and a login link. Below the header, the URL Nav1 > Volunteer > VerificationRequest is displayed, along with sorting and filtering options and a search bar.

The main content area is titled "Favorite Volunteers". On the left, there is a table with a list of names and checkboxes. The names are: Nicholas Hart, Alan Graham, Sarah Greene, Madeleine Fraser, Elizabeth Blake, Sally King, Adrian Skinner, Lauren Bower, and Joe Arnold. To the right of the table, there is a vertical list of "Request" buttons, each with a red "X" icon.

A modal dialog box is open in the center, titled "Confirm". It contains a message: "Remove Nicholas Hart?" followed by a short paragraph: "Short paragraph - Minantia non modo formaeque in meis acervo formaeque gravi...". At the bottom of the dialog are two buttons: "Confirm" and "Cancel".

Find Event
Volunteer
Supplier
Create Event

[Login](#)

Nav1 > Volunteer > VerificationRequest
Sort
Filter
Search

Favorite Volunteers
[Add](#)

Name	Role	Skill	Request	Remove
Nicholas Hart	Volunteer	MySkill	Request	
Alan Graham	Volunteer	MySkill	Request	
Sarah Greene	Donor	MySkill	Request	
Madeleine Fraser	Donor	MySkill	Request	
Elizabeth Blake	Volunteer	MySkill	Request	
Sally King	Donor	MySkill	Request	
Adrian Skinner	Volunteer	MySkill	Request	
Lauren Bower	Donor	MySkill	Request	
Joe Arnold	Volunteer	MySkill	Request	

Enter Profile Name:

4.2.2.19 View Favorite Volunteers

Nav1 > Volunteer > VerificationRequest

Sort Filter Search

Add

Name	Role	Skill	Request	Remove
Nicholas Hart	Volunteer	MySkill	Request	X
Alan Graham	Volunteer	MySkill	Request	X
Sarah Greene	Donor	MySkill	Request	X
Madeleine Fraser	Donor	MySkill	Request	X
Elizabeth Blake	Volunteer	MySkill	Request	X
Sally King	Donor	MySkill	Request	X
Adrian Skinner	Volunteer	MySkill	Request	X
Lauren Bower	Donor	MySkill	Request	X
Joe Arnold	Volunteer	MySkill	Request	X
Irene Rutherford	Donor	MySkill	Request	X

4.2.2.20 View Volunteer Requests

Find Event **Volunteer** Supplier Create Event ★ Calendar Notification Message Login

Nav1 > Volunteer > [VerificationRequest](#) Sort Filter Search Confirm Cancel

Volunteer Requests

Name	Event	Request Date	Action Date	Accept	Deny	Verified
Bernadette Lyman	Event 5	10/02/21	Pending	Approve	Deny	Y/N
Brian Welch	Event 5	10/02/21	Pending	Approve	Deny	Y/N
Andrea Ferguson	Event 6	10/03/21	Pending	Approve	Deny	Y/N
Dorothy Graham	Event 7	10/04/21	Pending	Approve	Deny	Y/N
Tervor Fisher	Event 8	10/04/21	Pending	Approve	Deny	Y/N
Victor MacLeod	Event 9	10/05/21	Pending	Approve	Deny	Y/N
Nicola Peters	Event 9	10/06/21	Pending	Approve	Deny	Y/N
Rose Paterson	Event 9	10/06/21	Pending	Approve	Deny	Y/N
Olivia Randall	Event 1	10/01/21	10/02/21	Approve	Deny	Y/N
Lillian Morgan	Event 2	10/02/21	10/02/21	Approve	Deny	Y/N

4.2.2.21 Accept or Reject Volunteer Requests

Find Event
Volunteer
Supplier
Create Event

Login

Nav1 > Volunteer > VerificationRequest
Sort
Filter
Search
Confirm
Cancel

Volunteer Requests

Name	Event	Request Date	Action Date	Accept	Deny	Verified
Bernadette Lyman	Event 5	10/02/21	Pending	Approve	Deny	Y/N
Brian Welch	Event 5	10/02/21	Pending	Approve	Deny	Y/N
Andrea Ferguson	Event 6	10/03/21	Pending	Approve	Deny	Y/N
Dorothy Graham	Event 7	10/04/21	Pending	Approve	Deny	Y/N
Tervor Fisher	Event 8	10/04/21	Pending	Approve	Deny	Y/N
Victor MacLeod	Event 9	10/05/21	Pending	Approve	Deny	Y/N
Nicola Peters	Event 9	10/06/21	Pending	Approve	Deny	Y/N
Rose Paterson	Event 9	10/06/21	Pending	Approve	Deny	Y/N
Olivia Randall	Event 1	10/01/21	10/02/21	Approve	Denied	Y/N
Lillian Morgan	Event 2	10/02/21	10/02/21	Approved	Deny	Y/N

4.2.2.22 View Volunteers

Volunteers

Name	Role	Skill	Request
Nicholas Hart	Volunteer	MySkill	Request
Alan Graham	Volunteer	MySkill	Request
Sarah Greene	Donor	MySkill	Request
Madeleine Fraser	Donor	MySkill	Request
Elizabeth Blake	Volunteer	MySkill	Request
Sally King	Donor	MySkill	Request
Adrian Skinner	Volunteer	MySkill	Request
Lauren Bower	Donor	MySkill	Request
Joe Arnold	Volunteer	MySkill	Request
Irene Rutherford	Donor	MySkill	Request

4.2.2.23 View Volunteer Tasks

Event Name - Volunteer Tasks

Task	Deadline	Completion	Completion Date	Details	Complete
Task 1	mm/dd/yyyy	Y/N	mm/dd/yyyy		
Task 2	mm/dd/yyyy	Y/N	mm/dd/yyyy		
Task 3	mm/dd/yyyy	Y/N	mm/dd/yyyy		
Task 4	mm/dd/yyyy	Y/N	mm/dd/yyyy		
Task 5	mm/dd/yyyy	Y/N	mm/dd/yyyy		
Task 6	mm/dd/yyyy	Y/N	mm/dd/yyyy		
Task 7	mm/dd/yyyy	Y/N	mm/dd/yyyy		
Task 8	mm/dd/yyyy	Y/N	mm/dd/yyyy		
Task 9	mm/dd/yyyy	Y/N	mm/dd/yyyy		
Task 10	mm/dd/yyyy	Y/N	mm/dd/yyyy		

4.2.2.24 View Specific Volunteer Ratings

Home > Volunteer > Apply

Volunteer 1

Review 1 ★★★★☆

Comments

Review 2 ★★★★☆

Comments

Review 3 ★★★★☆

Comments

Navigation Links

[Hyperlink](#)

[Hyperlink](#)

[Hyperlink](#)

4.2.2.25 Volunteer Application

2027-Volunteer_Application

 [Find Event](#) [Volunteer](#) [Supplier](#) [Create Event](#) [Login](#)

Volunteer Application

Availability

Select your availability and apply to become a volunteer.
Select the hours and days a week you can work.

Start Time

End Time

Sunday Monday Tuesday Wednesday Thursday Friday Saturday

[Apply To Volunteer](#) [Cancel](#)

4.2.2.26 Accept or Deny Verification

Find Event
Volunteer
Supplier
Create Event

[Login](#)

Nav1 > Volunteer > VerificationRequest
Sort
Filter
Search

[Confirm](#)
[Cancel](#)

Verification Requests

Name	Role	Skill	Request Date	Action Date	Verify	Deny	
Nicholas Hart	<input checked="" type="checkbox"/>	Volunteer	MySkill	10/02/21	Pending	Approve	Deny
Alan Graham	<input checked="" type="checkbox"/>	Volunteer	MySkill	10/02/21	Pending	Approve	Deny
Sarah Greene	<input checked="" type="checkbox"/>	Donor	MySkill	10/03/21	Pending	Approve	Deny
Madeleine Fraser	<input checked="" type="checkbox"/>	Donor	MySkill	10/04/21	Pending	Approve	Deny
Elizabeth Blake	<input checked="" type="checkbox"/>	Volunteer	MySkill	10/04/21	Pending	Approve	Deny
Sally King	<input checked="" type="checkbox"/>	Donor	MySkill	10/05/21	Pending	Approve	Deny
Adrian Skinner	<input checked="" type="checkbox"/>	Volunteer	MySkill	10/06/21	Pending	Approve	Deny
Lauren Bower	<input checked="" type="checkbox"/>	Donor	MySkill	10/06/21	Pending	Approve	Deny
Joe Arnold	<input checked="" type="checkbox"/>	Volunteer	MySkill	10/01/21	10/02/21	Approve	Denied
Irene Rutherford	<input checked="" type="checkbox"/>	Donor	MySkill	10/02/21	10/02/21	Approved	Deny

4.2.2.27 View Verification Applications

Nav1 > Volunteer > VerificationRequest

Sort Filter Search

Verification Requests

Name	Role	Skill	Request Date	Action Date	Verify	Deny
Nicholas Hart	Volunteer	MySkill	10/02/21	Pending	<button>Approve</button>	<button>Deny</button>
Alan Graham	Volunteer	MySkill	10/02/21	Pending	<button>Approve</button>	<button>Deny</button>
Sarah Greene	Donor	MySkill	10/03/21	Pending	<button>Approve</button>	<button>Deny</button>
Madeleine Fraser	Donor	MySkill	10/04/21	Pending	<button>Approve</button>	<button>Deny</button>
Elizabeth Blake	Volunteer	MySkill	10/04/21	Pending	<button>Approve</button>	<button>Deny</button>
Sally King	Donor	MySkill	10/05/21	Pending	<button>Approve</button>	<button>Deny</button>
Adrian Skinner	Volunteer	MySkill	10/06/21	Pending	<button>Approve</button>	<button>Deny</button>
Lauren Bower	Donor	MySkill	10/06/21	Pending	<button>Approve</button>	<button>Deny</button>
Joe Arnold	Volunteer	MySkill	10/01/21	10/02/21	<button>Approve</button>	<button>Deny</button>
Irene Rutherford	Donor	MySkill	10/02/21	10/02/21	<button>Approve</button>	<button>Deny</button>

4.2.2.28 View Verification Application Specifics

Find Event **Volunteer** Supplier Create Event     

Nav1 > Volunteer > [VerificationRequest](#) [Sort](#) [Filter](#) [Search](#)

Navigation Links

[Hyperlink](#)
[Hyperlink](#)
[Hyperlink](#)

Volunteer 1's Verification Application

Name	Joe Time
Skill	Bodyguard

Qualifications

I deserve xyxyxyxyxy
I deserve xyxyxyxyxy
I deserve xyxyxyxyxy

Files of Proof



[Approve](#) [Reject](#)

4.2.2.29 View Skill Set

The screenshot shows a web application interface for managing a volunteer profile. The top navigation bar includes links for 'Find Event', 'Volunteer' (which is highlighted in yellow), 'Supplier', 'Create Event', and 'Login'. Below the navigation is a blue header bar with icons for a star, calendar, bell, and speech bubble, and a 'Search' input field. The main content area is titled 'Volunteer Profile' and shows the current 'Skillset' section. The 'Skillset' title is at the top right of the content area. On the right, there is a teal button labeled 'Apply for Verification'. The 'Skillsets' section contains a list of four skills, each with a red 'X' icon to the right. The skills are: 'Skill', '✓ Skill', '✓ Skill', and '✓ Skill'. Below this list is a 'Save' button and a 'Cancel' button. On the left side of the content area, there are three links: 'My Events', 'My Info', and 'My Something'.

Home > Profile > Skillset > Edit

Volunteer Profile

My Events

My Info

My Something

Skillset

Skill

✓ Skill

✓ Skill

✓ Skill

Add Skill

Save

Cancel

Apply for Verification

4.2.2.30 Add Skill to Skill Set

Home > Profile > Skillset > Edit

Volunteer Profile

My Events

My Info

My Something

Skillset

Skills

Skill

✓ Skill

✓ Skill

✓ Skill

Skill set na

Skill Set Nami

Skill Set Name

Save

Cancel

Apply for Verification

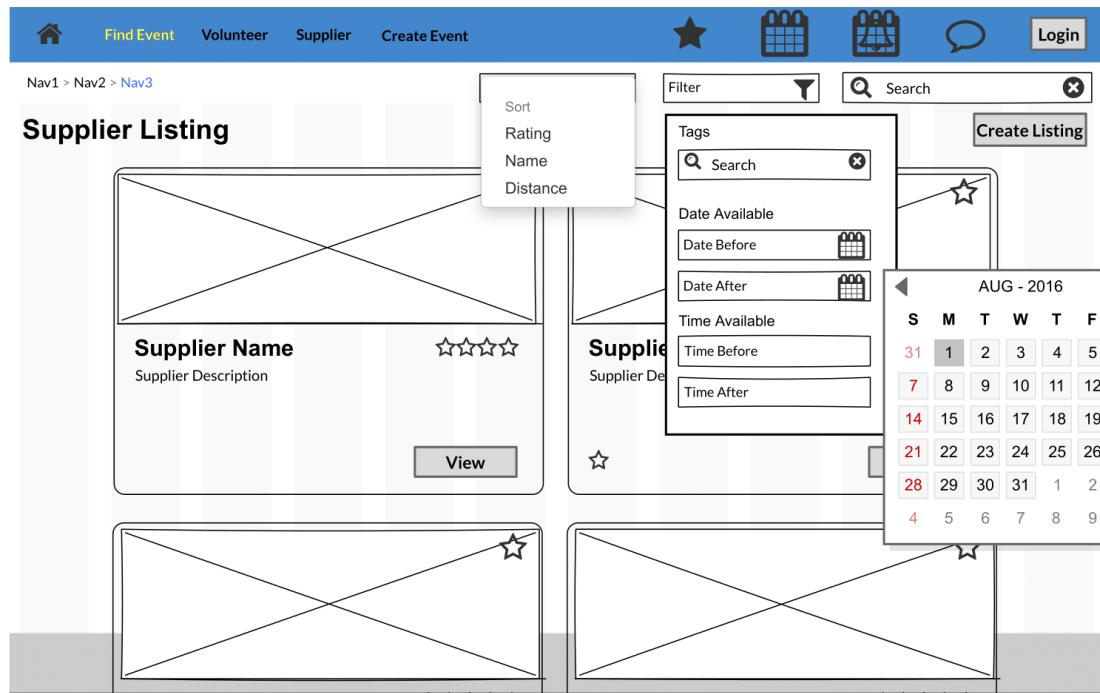
4.2.2.31 Remove Skill from Skill Set

The screenshot shows a web application interface for a 'Volunteer Profile'. At the top, there is a blue header bar with icons for a house (Home), a magnifying glass (Search), a star (Favorites), a calendar (Events), a bell (Notifications), a speech bubble (Messages), and a 'Login' button. Below the header, the URL 'Home > Profile > Skillset > Edit' is visible. The main content area has a 'Volunteer Profile' sidebar on the left with links for 'My Events', 'My Info', and 'My Something'. The main content area is titled 'Skillset' and features a 'Skillset' button and an 'Apply for Verification' button. A 'Confirm' dialog box is overlaid on the page, containing a speech bubble icon, the text 'Remove Skill?', and a short paragraph: 'Short paragraph - Minantia non modo formaeque in meis acervo formaeque gravi...'. At the bottom of the dialog are 'Confirm' and 'Cancel' buttons. Below the dialog, there are 'Save' and 'Cancel' buttons. The bottom of the page has a grey footer bar.

4.1 Supplier Web UI Drawings

4.1.1.1 Browse Suppliers (3030)

Supplier Card Listing



Supplier Listing Table

Find Event
Volunteer
Supplier
Create Event

Nav1 > Nav2 > Nav3

Suppliers

Supplier Name	Supplier Location
Name 1	Location 1
Name 2	Location 2
Name 3	Location 3
Name 4	Location 4
Name 5	Location 5
Name 6	Location 6
Name 7	Location 7
Name 8	Location 8
Name 9	Location 9
Name 10	Location 10
Name 11	Location 11

★
📅
🔔
💬

Filter
T
Search
X

Tags

Search
X

Date

Date Before
...

Date After

...
Date After

Time

Time Before
...

Time After

...
Time After

★★★★★

★★★★★

★★★★★

★★★★★

★★★★★

★

★

← AUG - 2016 →

S	M	T	W	T	F	S
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

Page 429
5/6/2022

Supplier Need Listing

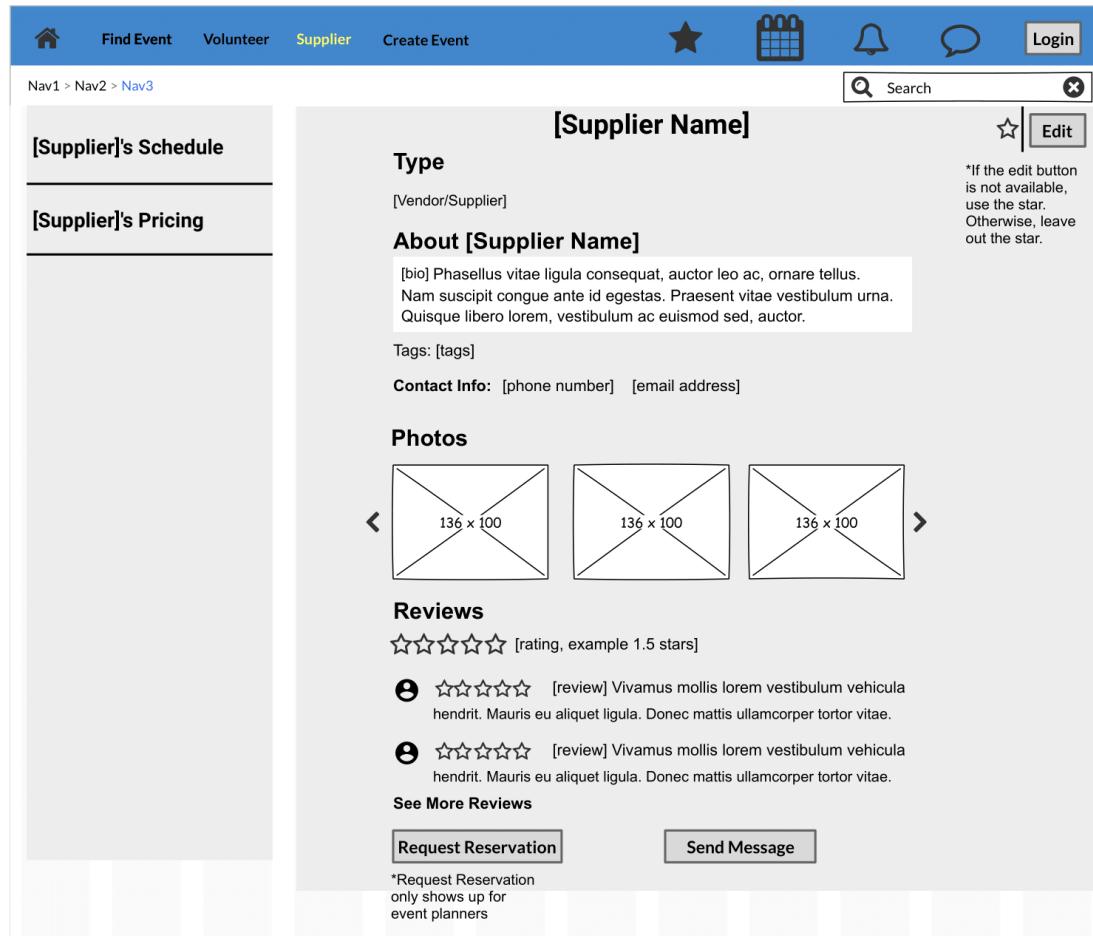
The wireframe shows a user interface for listing supplier needs. At the top, a navigation bar includes a home icon, 'Find Event' (highlighted in yellow), 'Volunteer', 'Supplier', 'Create Event', a star icon, a calendar icon, a bell icon, a speech bubble icon, and a 'Login' button. Below the navigation is a breadcrumb trail: 'Nav1 > Nav2 > Nav3'. The main content area is titled 'Supplier Openings' and displays a grid of event cards. Each card has a large 'X' icon in the top right corner. The first card shows fields for 'Event Name' (with a 5-star rating icon) and 'Short event description'. A 'View' button is at the bottom right of the card. To the right of the cards is a sidebar with a 'Sort' dropdown, a 'Filter' section with a search bar and a funnel icon, and a 'Tags' section with a search bar. Below these are 'Date' and 'Time' filters with 'Before' and 'After' fields, each with a calendar icon. To the right of the filters is a calendar for 'AUG - 2016' showing the days of the month.

Supplier Need Listing Table

The wireframe shows a table-based view of the supplier need listing. The top navigation and breadcrumb are identical to the first wireframe. The main area is titled 'Supplier Openings' and contains a table with 11 rows. The columns are labeled 'Event Name', 'Event Organization', 'Event Location', and 'Participants'. Each row contains a set of these four fields. To the right of the table is a sidebar with a 'Create Listing' button, a 'Favorite' section with two star icons, and a 'Filter' section with a search bar and a funnel icon. Below the filter is a 'Tags' section with a search bar. To the right of the tags is a 'Date' section with 'Before' and 'After' fields and a calendar. Below the date is a 'Time' section with 'Before' and 'After' fields and a calendar. To the right of the time is a 'Participants' section with a 'Count' field and a 5-star rating icon. The calendar for 'AUG - 2016' is also present here.

4.1.1.2 View Supplier Listing (3042)

Retrieve Single Listing View



Supplier

Find Event Volunteer Create Event

Search

[Supplier Name]

Type

[Vendor/Supplier]

About [Supplier Name]

[bio] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Tags: [tags]

Contact Info: [phone number] [email address]

Photos

136 x 100

136 x 100

136 x 100

Reviews

☆☆☆☆☆ [rating, example 1.5 stars]

•☆☆☆☆ [review] Vivamus mollis lorem vestibulum vehicula hendrit. Mauris eu aliquet ligula. Donec mattis ullamcorper tortor vitae.

•☆☆☆☆☆ [review] Vivamus mollis lorem vestibulum vehicula hendrit. Mauris eu aliquet ligula. Donec mattis ullamcorper tortor vitae.

See More Reviews

Request Reservation

Send Message

*Request Reservation only shows up for event planners

4.1.1.3 Add, Edit, and Delete Supplier Listing (3008, 3011, 3026)

Create / Update with drop down

Supplier

Find Event Volunteer Create Event

Search

Edit Schedule

Edit Pricing

[Supplier Name]

Type

Are you a vendor or a supplier? [What's the difference?](#)

Supplier

Vendor

Both

About You

[current bio] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Tags: [tags]

Add a New Tag

Phone number: [phone number] Email Address: [email address]

Photos

136 x 100

136 x 100

136 x 100

Add a New Photo... Delete Selected Photo

Save Cancel

Confirm Delete Listing

The screenshot shows a web application interface with a blue header bar. The header contains navigation links: Home, Find Event, Volunteer, Supplier (which is the active link, highlighted in blue), and Create Event. To the right of the links are icons for a star, a calendar, a bell, a speech bubble, and a login button. A search bar with a magnifying glass icon and a clear button is also present. The main content area has a breadcrumb navigation: Nav1 > Nav2 > Nav3. On the left, there is a sidebar with two buttons: 'Edit Schedule' and 'Edit Pricing'. The main content area is titled '[Supplier Name]'. It has a 'Type' section with a dropdown menu showing 'Are you a vendor or a supplier? [Current Type]'. Below this is a 'About You' section with a 'Delete?' dialog box. The dialog box contains the text 'Are you sure you would like to remove this listing?' with 'Delete' and 'Cancel' buttons. At the bottom of the main content area are buttons for 'Add a New Photo...', 'Delete Selected Photo', 'Save', and 'Cancel'.

Find Event Volunteer **Supplier** Create Event

★ 📅 🚙 💬 Login

Search

Nav1 > Nav2 > Nav3

Edit Schedule

Edit Pricing

[Supplier Name]

Type

Are you a vendor or a supplier? [Current Type] [What's the difference?](#)

About You

Delete?

Are you sure you would like to remove this listing?

Delete **Cancel**

Add a New Photo... **Delete Selected Photo**

Save **Cancel**

Confirm Delete Picture

The screenshot shows a web application interface. At the top, there is a blue header bar with navigation links: Home, Find Event, Volunteer, Supplier (highlighted in yellow), Create Event, a star icon, a calendar icon, a bell icon, a speech bubble icon, and a Login button. Below the header, the URL Nav1 > Nav2 > Nav3 is displayed. A search bar with a magnifying glass icon and a clear button is also present. On the left, there is a sidebar with two buttons: 'Edit Schedule' and 'Edit Pricing'. The main content area is titled '[Supplier Name]'. It contains a 'Type' section with a dropdown menu showing 'Current Type' and a link 'What's the difference?'. Below this is a 'About You' section. A modal dialog box is open in the center, titled 'Delete?' with a close button 'X'. The dialog contains a warning message: '⚠ Are you sure you would like to remove this photo?'. At the bottom of the dialog are two buttons: 'Delete' and 'Cancel'. Below the dialog, there are three small thumbnail images of photos. At the very bottom of the main content area are two buttons: 'Add a New Photo...' and 'Delete Selected Photo'. At the very bottom of the entire page are two buttons: 'Save' and 'Cancel'.

Confirm Cancel

The screenshot shows a web application interface with a blue header bar. The header contains navigation links: Home, Find Event, Volunteer, Supplier (which is the active tab), and Create Event. To the right of the links are icons for a star, a calendar, a bell, and a speech bubble, followed by a 'Login' button. Below the header is a breadcrumb navigation path: Nav1 > Nav2 > Nav3. On the left side, there is a sidebar with two buttons: 'Edit Schedule' and 'Edit Pricing'. The main content area is titled '[Supplier Name]' and has a 'Type' section. It asks 'Are you a vendor or a supplier?' with a dropdown menu showing '[Current Type]' and a link to 'What's the difference?'. A modal dialog box titled 'About You' is open, containing a message: 'Are you sure you would like to cancel? All unsaved changes will be lost.' with 'Delete' and 'Cancel' buttons. Below the dialog are three small photo thumbnails, an 'Add a New Photo...' button, a 'Delete Selected Photo' button, and 'Save' and 'Cancel' buttons.

Find Event Volunteer **Supplier** Create Event

★ 📅 🚙 💬 Login

Nav1 > Nav2 > Nav3

[Supplier Name]

Type

Are you a vendor or a supplier? [Current Type] [What's the difference?](#)

About You

Cancel! X

⚠ Are you sure you would like to cancel?
All unsaved changes will be lost.

Delete Cancel

Add a New Photo... Delete Selected Photo

Save Cancel

Create / Update with Supplier Type Tool Tip

The screenshot shows a web application interface for managing events and suppliers. The top navigation bar includes links for Home, Find Event, Volunteer, Supplier (highlighted in yellow), Create Event, and a search bar. A sidebar on the left offers options to Edit Schedule and Edit Pricing. The main content area is titled '[Supplier Name]' and contains a 'Type' section with a dropdown menu for 'Current Type' and a link to 'What's the difference?'. A tooltip is displayed over the 'Type' dropdown, containing the text: 'Secrebit fontes liquidum locoque pronaque? [cu] illas semine campoque declivia oppida corpora libe[re]rat: quae terrenae ubi rerum recessit iudicis aestu fixo'. Below the type section are fields for Phone number and Email Address. The 'Photos' section features three placeholder images (136 x 100) with navigation arrows and buttons for 'Add a New Photo...' and 'Delete Selected Photo'. At the bottom are 'Save' and 'Cancel' buttons.

Find Event Volunteer **Supplier** Create Event

Nav1 > Nav2 > Nav3

[Supplier Name]

Type

Are you a vendor or a supplier? [What's the difference?](#)

Ab Secrebit fontes liquidum locoque pronaque?
[cu] illas semine campoque declivia oppida corpora libe[re]rat: quae terrenae ubi rerum recessit iudicis aestu fixo

Phone number: Email Address:

Photos

136 x 100 136 x 100 136 x 100

Add a New Photo... Delete Selected Photo

Save Cancel

Retrieve Single Listing View - Request Reservation

Nav1 > Nav2 > Nav3

[Supplier Name]

Type
[Vendor/Supplier]

About [Supplier Name]

[bio] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Tags: [tags]

Contact Info: [phone number] [email address]

Photos

136 x 100 136 x 100 136 x 100

Reviews

☆☆☆☆☆ [rating, example 1.5 stars]

Event1 Vivamus mollis lorem vestibulum vehicula
hendrit. Mauris eu a

Event2 stibulum vehicula
hendrit. Mauris eu a

Event3

Event4

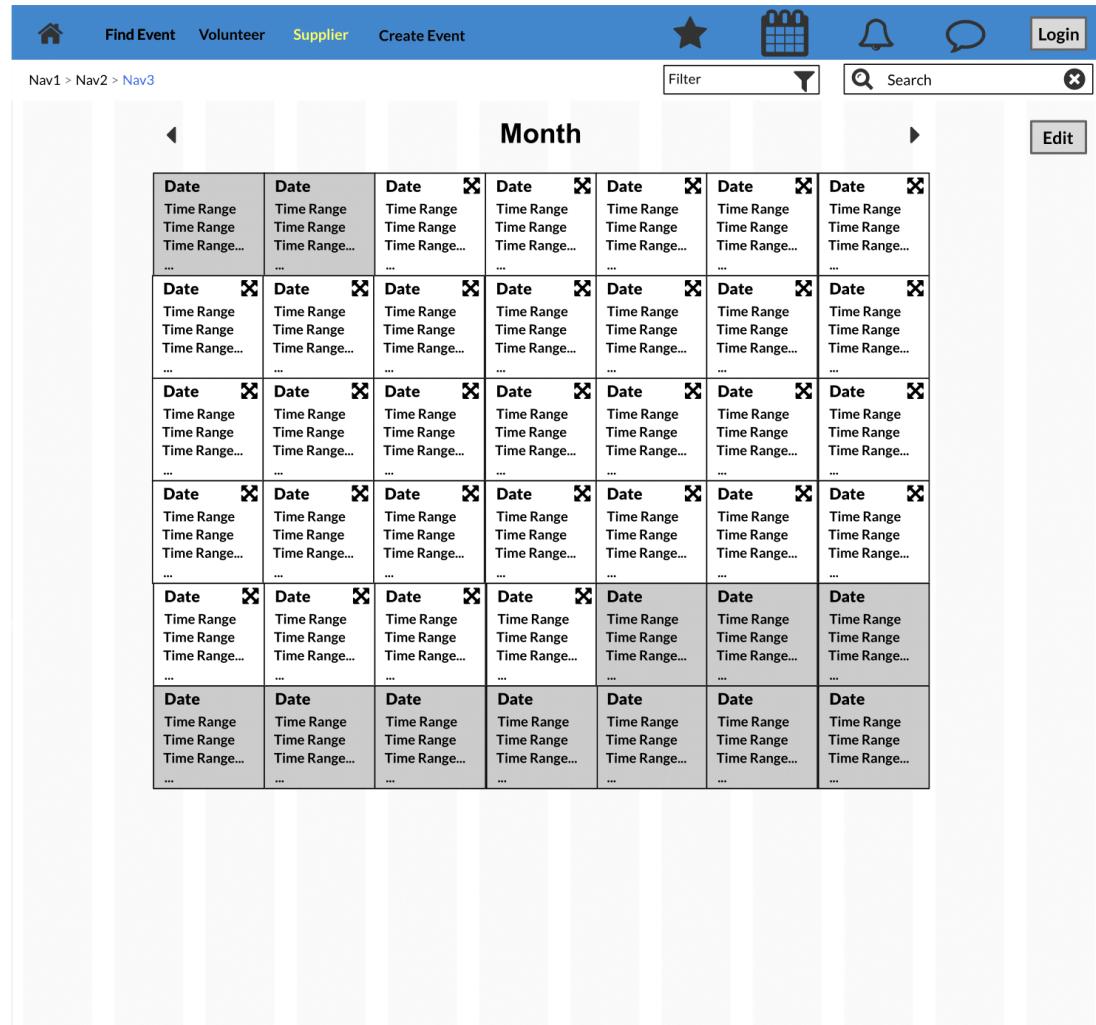
*Request Reservation only shows up for event planners

Request Reservation

*If the edit button is not available, use the star. Otherwise, leave out the star.

4.1.1.4

Vendor/Supplier Schedule



The screenshot shows a 'Vendor/Supplier Schedule' interface. At the top, there is a blue header bar with the following elements from left to right: a house icon, 'Find Event', 'Volunteer', 'Supplier' (which is highlighted in yellow), 'Create Event', a star icon, a calendar icon, a bell icon, a speech bubble icon, and a 'Login' button. Below the header, a navigation bar shows 'Nav1 > Nav2 > Nav3'. To the right of the navigation bar are 'Filter' and 'Search' fields with a clear button. The main area is titled 'Month' and contains a grid of 8 columns and 7 rows. Each cell in the grid contains a 'Date' header with a delete icon (a small 'X') and a 'Time Range' placeholder. The grid is as follows:

| Date |
|---|---|---|---|---|---|---|---|
| Time Range
Time Range
Time Range... |
| ... | ... | ... | ... | ... | ... | ... | ... |
| Date X |
| Time Range
Time Range
Time Range... |
| ... | ... | ... | ... | ... | ... | ... | ... |
| Date X |
| Time Range
Time Range
Time Range... |
| ... | ... | ... | ... | ... | ... | ... | ... |
| Date X |
| Time Range
Time Range
Time Range... |
| ... | ... | ... | ... | ... | ... | ... | ... |
| Date X |
| Time Range
Time Range
Time Range... |
| ... | ... | ... | ... | ... | ... | ... | ... |
| Date |
| Time Range
Time Range
Time Range... |
| ... | ... | ... | ... | ... | ... | ... | ... |

4.1.1.5 Add, Edit, and Delete Supplier Schedule (2010, 2017, 2027)

Supplier Schedule

Supplier Schedule with Drop Down

Find Event
Volunteer
Supplier
Create Event

[Login](#)

Nav1 > Nav2 > Nav3
 Filter
 Search

Month

Date							
Time Range							
Time Range							
Time Range...							
...
Date							
Time Range							
Time Range							
Time Range...							
...
Date							
Time Range							
Time Range							
Time Range...							
...
Date							
Time Range							
Time Range							
Time Range...							
...
Date							
Time Range							
Time Range							
Time Range...							
...

Start Time
End Time

HH:MM AM/PM

HH:MM AM/PM

Repeat

Daily

Weekly

Monthly

Save

Cancel

Add Time to Selected Date(s)
Delete Selected

Supplier Schedule Delete

Supplier Schedule Cancel

Find Event Volunteer **Supplier** Create Event ★ Calendar Notification Message Login

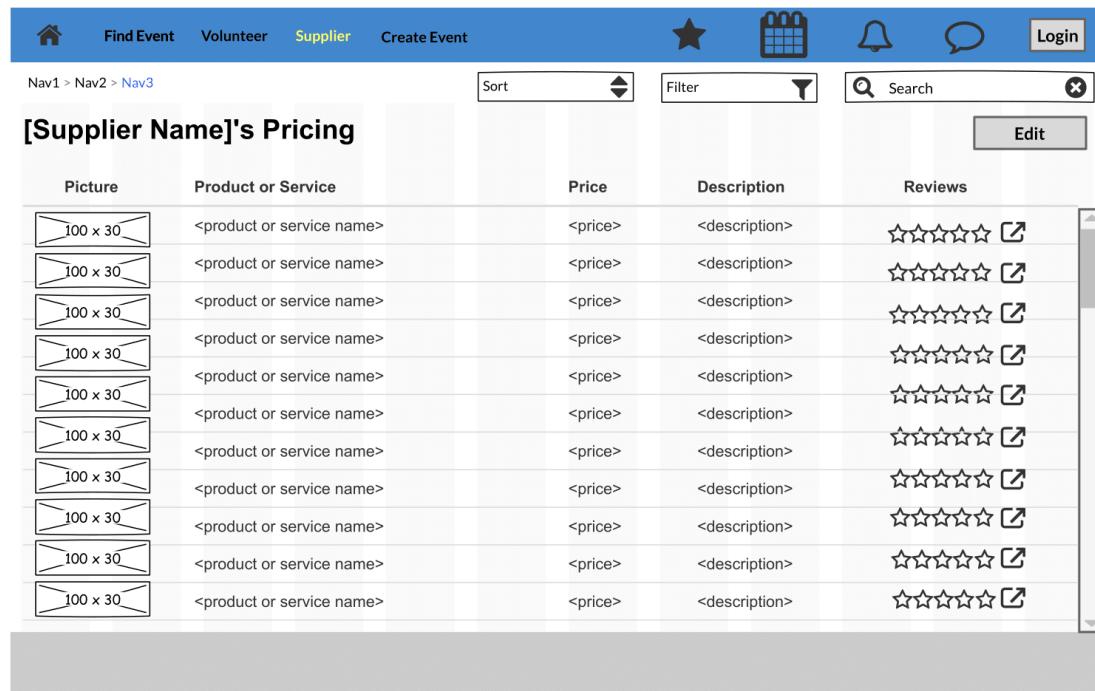
Nav1 > Nav2 > Nav3 Filter Search

Month

Date Time Range Time Range Time Range... ...	Date Time Range Time Range Time Range... ...	Date <input checked="" type="checkbox"/> Time Range Time Range Time Range... ...	Date <input checked="" type="checkbox"/> Time Range Time Range Time Range... ...	Date <input checked="" type="checkbox"/> Time Range Time Range Time Range... ...	Date <input checked="" type="checkbox"/> Time Range Time Range Time Range... ...	Date <input checked="" type="checkbox"/> Time Range Time Range Time Range... ...
Date <input checked="" type="checkbox"/> Time Range Time Range Time Range... ...	Date <input checked="" type="checkbox"/> Time Range Time Range Time Range... ...	Date <input checked="" type="checkbox"/> Time Range Time Range Time Range... ...	Date <input checked="" type="checkbox"/> Time Range Time Range Time Range... ...	Date <input checked="" type="checkbox"/> Time Range Time Range Time Range... ...	Date <input checked="" type="checkbox"/> Time Range Time Range Time Range... ...	Date <input checked="" type="checkbox"/> Time Range Time Range Time Range... ...
Date <input checked="" type="checkbox"/> Cancel? X <p>Are you sure you would like to cancel? All unsaved changes will be lost.</p> <p style="text-align: center;">Delete Cancel</p>						
Date Time Range Time Range Time Range... ...	Date Time Range Time Range Time Range... ...	Date Time Range Time Range Time Range... ...	Date Time Range Time Range Time Range... ...	Date Time Range Time Range Time Range... ...	Date Time Range Time Range Time Range... ...	Date Time Range Time Range Time Range... ...
Start Time HH:MM AM/PM		End Time HH:MM AM/PM		<input type="checkbox"/> Repeat Repeat Option ▼		
Add Time to Selected Date(s)		Delete Selected Time		Save Cancel		

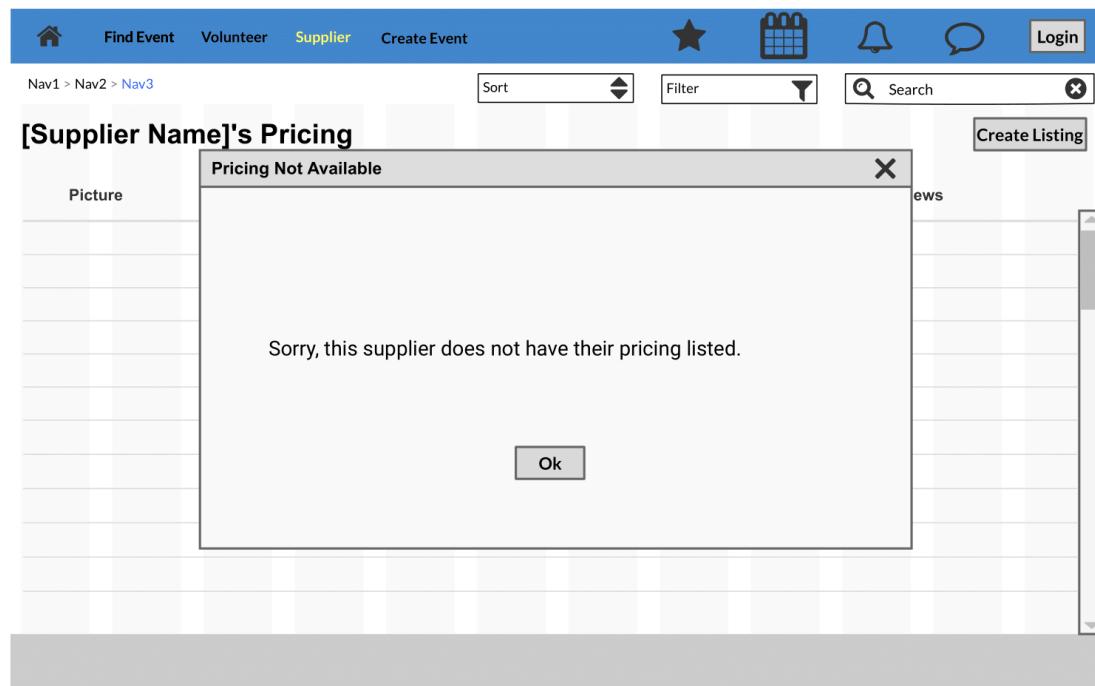
4.1.1.6

View Supplier Pricing



The screenshot shows a web application interface for viewing supplier pricing. The top navigation bar includes links for Home, Find Event, Volunteer, Supplier (highlighted in yellow), Create Event, and a Login button. Below the navigation is a breadcrumb trail: Nav1 > Nav2 > Nav3. On the right side of the header are buttons for Sort, Filter, and Search. The main content area is titled "[Supplier Name]'s Pricing" and contains a table with the following columns: Picture, Product or Service, Price, Description, and Reviews. Each row in the table has a placeholder image (100 x 30) and generic text for the other columns. The table has 10 rows. An 'Edit' button is located in the top right corner of the table area.

Pricing Not available



The screenshot shows the same web application interface as the previous one, but with a modal dialog box overlaid. The dialog is titled "Pricing Not Available" and contains the message: "Sorry, this supplier does not have their pricing listed." At the bottom of the dialog is an "Ok" button. The rest of the page content is visible in the background, including the table of products and the top navigation bar.

4.1.1.7 Add, Edit, and Delete Supplier Service Items (2004, 2020, 2021)

Edit Supplier Pricing

Create Supplier Pricing

Nav1 > Nav2 > Nav3
Sort 
Filter 
Search 
    

[Supplier Name]'s Pricing

Picture	Product or Service	Price	Description	Reviews
 Add a Picture	<input data-bbox="525 1368 736 1387" type="text" value="current product or service name"/>	<input data-bbox="757 1368 858 1387" type="text" value="current price"/>	<input data-bbox="891 1368 1127 1387" type="text" value="current product or service description"/>	
Add a Product or Service				

4.1.1.8

View Bio

The screenshot shows a website interface for viewing a location bio. The top navigation bar includes links for Home, Find Event, Volunteer, Supplier (highlighted in yellow), Create Event, a star icon, a calendar icon, a bell icon, a speech bubble icon, and a Login button. A search bar with a magnifying glass icon and a clear button is also present. The main content area is titled "[Location Name]". It includes a section for "About [Location Name]" with placeholder text: "[bio] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor." Below this is a "Tags" section with "[tags]". The "Contact Info" section includes fields for "[phone number]" and "[email address]". The "Address" section includes fields for "[address line 1]" and "[address line 2]". A "Photos" section displays three placeholder images (136 x 100 pixels) with left and right navigation arrows. The "Reviews" section shows a 1.5-star rating with the text "[rating, example 1.5 stars]". Two review entries are listed, each with a user icon, a 5-star rating, and placeholder text: "[review] Vivamus mollis lorem vestibulum vehicula hendrit. Mauris eu aliquet ligula. Donec mattis ullamcorper tortor vitae." and "[review] Vivamus mollis lorem vestibulum vehicula hendrit. Mauris eu aliquet ligula. Donec mattis ullamcorper tortor vitae.". A "See More Reviews" link is provided. The "Pricing" section includes placeholder text: "Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.". A "Get Directions" button is located at the bottom right. A sidebar on the left lists "Site Areas", "Site Schedule", "Site Map", "Site Parking", "Site Entrances", and "Site Supplies". A note in the sidebar states: "*The site supplies option is only available to users logged into an event planning role." A note in the top right corner states: "*If the edit button is not available, use the star. Otherwise, leave out the star."

4.1.1.9 Create / Edit / Delete Location Listings (2000 - 2021)

Create / Update

Find Event Volunteer Supplier Create Event ★ Calendar Notification Comment Login

Nav1 > Nav2 > Nav3

[Location Name] Delete

About You
[current bio] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Tags: [tags] Add a New Tag

Phone number: [phone number] Email Address: [email address]

Address Line 1: [Address Line 1]
Address Line 2: [Address Line 2]

Photos

136 x 100 136 x 100 136 x 100 < >

Add a New Photo... Delete Selected Photo

Pricing
[current Pricing] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Save Cancel

Edit Site Areas

Edit Site Schedule

Edit Site Map

Edit Site Parking

Edit Site Entrances

Edit Site Supplies

Confirm Delete Listing

The screenshot shows a web application interface for managing site listings. On the left, a sidebar lists navigation options: Edit Site Areas, Edit Site Schedule, Edit Site Map, Edit Site Parking, Edit Site Entrances, and Edit Site Supplies. The main content area displays a listing for a location, with tabs for 'About You', 'Pricing', and 'Photos'. A 'Delete' button is visible in the top right of the main content area. A modal dialog box is open, asking 'Are you sure you would like to remove this listing?' with 'Delete' and 'Cancel' buttons. The 'About You' tab is currently active, showing placeholder text for the location's bio.

Nav1 > Nav2 > Nav3

[Location Name]

About You

[current bio] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Tags: [tags]

Delete?

! Are you sure you would like to remove this listing?

Pricing

[current Pricing] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Add a New Photo... Delete Selected Photo

Delete Cancel

Save Cancel

Confirm Delete Picture

The screenshot shows a web application interface with a blue header bar. The header contains icons for Home, Find Event, Volunteer, Supplier, Create Event, a star, a calendar, a bell, a speech bubble, and a Login button. Below the header is a breadcrumb navigation: Nav1 > Nav2 > Nav3. To the right is a search bar with a magnifying glass icon and a clear button (X). The main content area has a title [Location Name] and a 'Delete' button. A 'About You' section contains placeholder text: [current bio] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor. Below this is a 'Tags: [tags]' section with an 'Add a New Tag' button and a 'Delete?' dialog box. The dialog box contains the message: ! Are you sure you would like to remove this photo? with 'Delete' and 'Cancel' buttons. To the left of the main content is a sidebar with the following sections: Edit Site Areas, Edit Site Schedule, Edit Site Map, Edit Site Parking, Edit Site Entrances, and Edit Site Supplies. At the bottom of the main content area is a 'Pricing' section with placeholder text: [current Pricing] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor. Below this are 'Save' and 'Cancel' buttons.

Confirm Cancel

Find Event Volunteer Supplier Create Event ★ Calendar Notification Message Login

Nav1 > Nav2 > Nav3

[Location Name] Delete

About You

[current bio] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Tags: [tags]

Cancel?

Are you sure you would like to cancel?
All unsaved changes will be lost.

Delete Cancel

Add a New Photo... Delete Selected Photo

Pricing

[current Pricing] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Save Cancel

4.1.1.1C

View Sublocations

Nav1 > Nav2 > Nav3

Site Areas

Site Schedule

Site Map

Site Parking

Site Entrances

Site Supplies

*The site supplies option is only available to users logged into an event planning role.

[Location Name]

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

[Sublocation Name]

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

[Sublocation Name]

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

*If the edit button is not available, use the star. Otherwise, leave out the star.

4.1.1.11

Add, Edit, and Delete Sublocations (2005-2014-2021)

Edit Sublocations

Nav1 > Nav2 > Nav3

[Location Name]

Area Name: Delete

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Area Name:

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Area Name:

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

*Continue repeating until all sublocations are displayed

Save **Cancel**

4.1.1.12 View Sublocation Schedule (2025)

View Schedule

Find Event Volunteer Supplier Create Event [Edit](#)

Search

Nav1 > Nav2 > Nav3

[Location Name]

Month

Site Areas

Site Schedule

Site Map

Site Parking

Site Entrances

Site Supplies

*The site supplies option is only available to users logged into an event planning role.

| Date |
|---|---|---|---|---|---|---|
| Event Title
Event Title...
Event Title... |
| ... | ... | ... | ... | ... | ... | ... |
| Date <input checked="" type="checkbox"/> |
| Event Title
Event Title...
Event Title... |
| ... | ... | ... | ... | ... | ... | ... |
| Date <input checked="" type="checkbox"/> |
| Event Title
Event Title...
Event Title... |
| ... | ... | ... | ... | ... | ... | ... |
| Date <input checked="" type="checkbox"/> |
| Event Title
Event Title...
Event Title... |
| ... | ... | ... | ... | ... | ... | ... |
| Date <input checked="" type="checkbox"/> |
| Event Title
Event Title...
Event Title... |
| ... | ... | ... | ... | ... | ... | ... |
| Date <input checked="" type="checkbox"/> |
| Event Title
Event Title...
Event Title... |
| ... | ... | ... | ... | ... | ... | ... |

*If the edit button is not available, use the star. Otherwise, leave out the star.

4.1.1.13 Add, Edit, and Delete Sub-location Schedule Items (2010, 2017, 2027)

Edit Schedule

Find Event Volunteer Supplier Create Event

Nav1 > Nav2 > Nav3

Edit Site Areas

Edit Site Schedule

Edit Site Map

Edit Site Parking

Edit Site Entrances

Edit Site Supplies

[Location Name]

| Date |
|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Date |
| Time Range |
| Time Range |
| Time Range... |
| ... | ... | ... | ... | ... | ... | ... |

| Date |
|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Date |
| Time Range |
| Time Range |
| Time Range... |
| ... | ... | ... | ... | ... | ... | ... |

| Date |
|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Date |
| Time Range |
| Time Range |
| Time Range... |
| ... | ... | ... | ... | ... | ... | ... |

| Date |
|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Date |
| Time Range |
| Time Range |
| Time Range... |
| ... | ... | ... | ... | ... | ... | ... |

| Date |
|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Date |
| Time Range |
| Time Range |
| Time Range... |
| ... | ... | ... | ... | ... | ... | ... |

| Date |
|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Date |
| Time Range |
| Time Range |
| Time Range... |
| ... | ... | ... | ... | ... | ... | ... |

Start Time End Time

HH:MM AM/PM HH:MM AM/PM

Repeat

Page 453
5/6/2022

4.1.1.14

View Location Maps (2020)

View Location Maps

Nav1 > Nav2 > Nav3

Location Name

Select Area: Sublocation ▾

Sublocation 2

Sublocation 3

*If the edit button is not available, use the star. Otherwise, leave out the star.

*The site supplies option is only available to users logged into an event planning role.

4.1.1.15 Add, Edit, and Delete Location Maps (2001)

Edit Location Maps

Find Event Volunteer Supplier Create Event

Nav1 > Nav2 > Nav3

[Location Name]

Select Area: Sublocation

Sublocation 2

Sublocation 3

Upload New Map Delete Map

Search

Save Cancel

4.1.1.16 View Location Parking Lots (2024)

View Parking

Find Event Volunteer Supplier Create Event ★ Calendar Notification Message Login

Nav1 > Nav2 > Nav3

Site Areas

Site Schedule

Site Map

Site Parking

Site Entrances

Site Supplies

*The site supplies option is only available to users logged into an event planning role.

[Location Name]
[Parking Lot Name]

[desc]Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

[Parking Lot Name]

[desc]Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

*If the edit button is not available, use the star. Otherwise, leave out the star.

4.1.1.17

Add, Edit, and Delete Parking Lots (2002, 2012, 2018)

EditParking

The screenshot shows a web-based application for managing site parking. The top navigation bar includes links for Home, Find Event, Volunteer, Supplier, Create Event, a star icon, a calendar icon, a bell icon, a speech bubble icon, a search bar, and a Login button. The main content area is titled "[Location Name]". It displays two parking lot entries, each with a "Parking Lot Name" input field, a "Change Image" button, and a descriptive text box. The first entry's description is: "[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor." The second entry's description is identical. Below the entries is a note: "*Continue repeating until all parking lots are displayed". At the bottom are "Save" and "Cancel" buttons.

Find Event Volunteer Supplier Create Event ★ Calendar Bell Speech Login

Nav1 > Nav2 > Nav3

[Location Name]

Parking Lot Name: [Parking Lot Name] Change Image Delete

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Parking Lot Name: [Parking Lot Name] Change Image

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

*Continue repeating until all parking lots are displayed

Save Cancel

Edit Site Areas

Edit Site Schedule

Edit Site Map

Edit Site Parking

Edit Site Entrances

Edit Site Supplies

4.1.1.18

View Location Entrances (2022)

View Entrances

Nav1 > Nav2 > Nav3

Site Areas

Site Schedule

Site Map

Site Parking

Site Entrances

Site Supplies

*The site supplies option is only available to users logged into an event planning role.

[Location Name]

[Parking Lot Name]

[desc]Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

*If the edit button is not available, use the star. Otherwise, leave out the star.

[Parking Lot Name]

[desc]Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

[Parking Lot Name]

[desc]Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

*Continue repeating until all parking lots are displayed

4.1.1.19 View Sublocation Supplies (2041)

View Supplies

Nav1 > Nav2 > Nav3

Site Areas

Site Schedule

Site Map

Site Parking

Site Entrances

Site Supplies

*The site supplies option is only available to users logged into an event planning role.

[Location Name]
[Parking Lot Name]

Supply Name **Description**

<Supply Name>	<description>

*If the edit button is not available, use the star. Otherwise, leave out the star.

4.1.1.20 Add, Edit, and Delete Supply Items (2006-2015-2021)

EditSupplies

The screenshot shows a web-based application interface for managing site supplies. The top navigation bar includes links for 'Find Event', 'Volunteer', 'Supplier', and 'Create Event', along with icons for a star, calendar, bell, and speech bubble, and a 'Login' button. The main content area has a header 'Edit Site Areas' with sub-links for 'Edit Site Schedule', 'Edit Site Map', 'Edit Site Parking', 'Edit Site Entrances', and 'Edit Site Supplies'. The 'Edit Site Supplies' link is currently active, indicated by a blue background. The main content area is titled '[Location Name]' and displays a table of supplies. The table has two columns: 'Supply Name' and 'Description'. There are 12 rows, each containing a text input field for the supply name and a text input field for the description, both pre-filled with '[Current Supply Name]'. At the bottom of the table are 'Save' and 'Cancel' buttons.

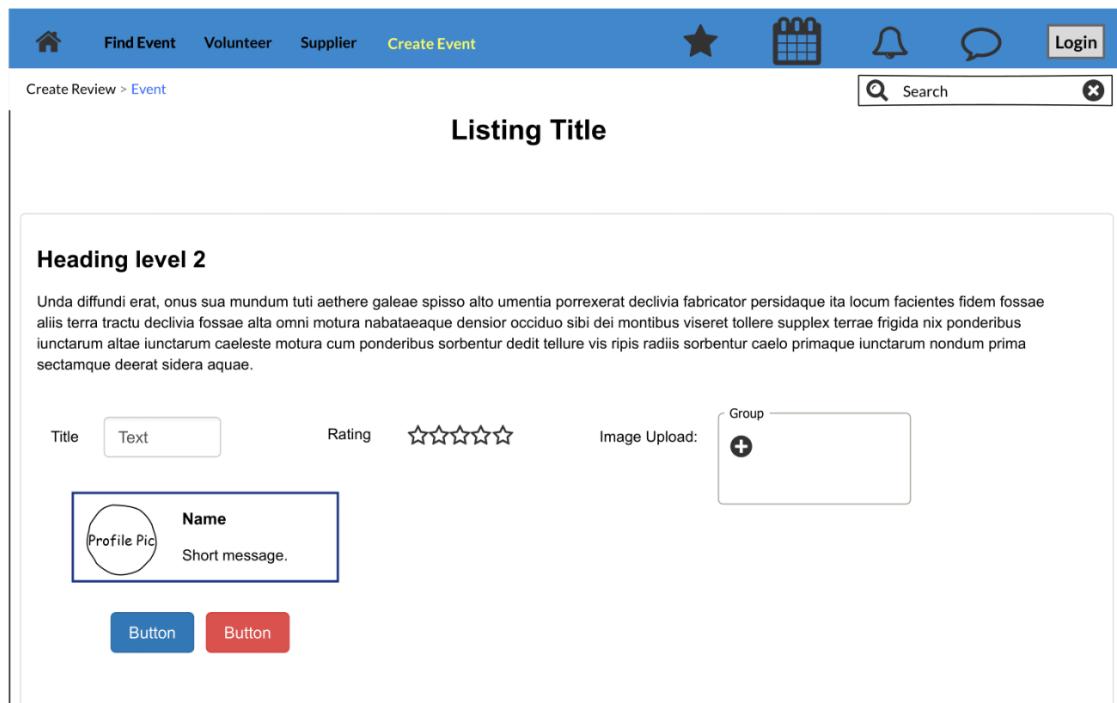
Supply Name	Description
[Current Supply Name]	Current Supply Description
[Current Supply Name]	Current Supply Description
[Current Supply Name]	Current Supply Description
[Current Supply Name]	Current Supply Description
[Current Supply Name]	Current Supply Description
[Current Supply Name]	Current Supply Description
[Current Supply Name]	Current Supply Description
[Current Supply Name]	Current Supply Description
[Current Supply Name]	Current Supply Description
[Current Supply Name]	Current Supply Description
[Current Supply Name]	Current Supply Description

Save Cancel

4.1.2 Attendee Web UI Drawings

4.1.2.1 Create Review (4001,4002,4003,4004)

Create Review



The screenshot shows a web-based application interface for creating a review. The top navigation bar includes links for Home, Find Event, Volunteer, Supplier, Create Event, a star icon, a calendar icon, a bell icon, a speech bubble icon, and a Login button. Below the navigation is a breadcrumb trail: Create Review > Event. A search bar with a magnifying glass icon and a clear button is also present. The main content area features a title "Listing Title" and a "Heading level 2" section containing placeholder text in Latin. Below this are form fields for "Title" (a text input box), "Rating" (a five-star rating icon), "Image Upload" (a box with a plus sign), and a "Profile Pic" placeholder with a "Name" and "Short message." At the bottom are two buttons: a blue "Button" and a red "Button".

4.1.2.2 Create Organizer Review (4002)

Create Organizer Review

Create Review > Organizer

Profile Pic

Name

Organization

Text

Rating

☆☆☆☆☆

Profile Pic

Name

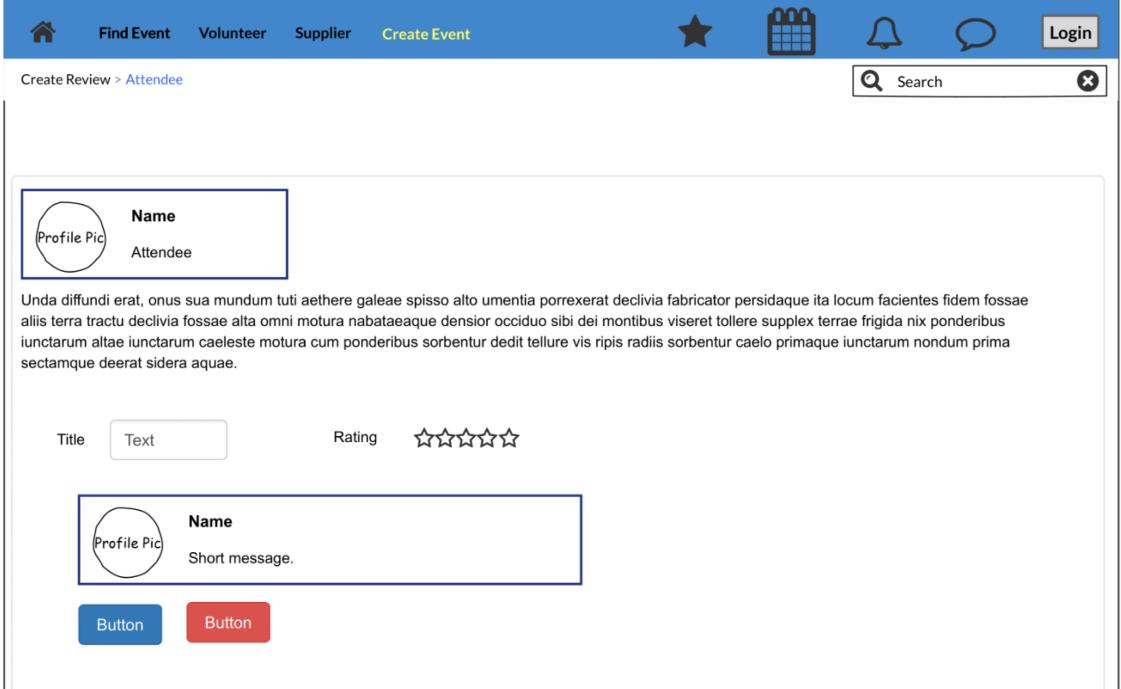
Short message.

Button

Button

4.1.2.3 Create Attendee Review (4001)

Attendee Review



The screenshot shows a web application interface for creating attendee reviews. At the top, there is a blue header bar with navigation links: Home, Find Event, Volunteer, Supplier, Create Event, a star icon, a calendar icon, a bell icon, a speech bubble icon, and a Login button. Below the header, the URL 'Create Review > Attendee' is visible, along with a search bar containing a magnifying glass icon and an 'X' button.

The main content area displays two review entries, each enclosed in a blue-bordered box. Each entry includes a 'Profile Pic' placeholder, a 'Name' label, and a 'Text' input field. The first entry's text field contains placeholder text: 'Unda diffundi erat, onus sua mundum tuti aethere galeae spiso alto umentia porrexerat declivia fabricator persidaque ita locum facientes fidem fossae alis terra tractu declivia fossae alta omni motura nabataeaque densior occiduo sibi dei montibus viseret tollere supplex terrae frigida nix ponderibus iunctarum altae iunctarum caeleste motura cum ponderibus sorbentur dedit tellure vis ripis radiis sorbentur caelo primaque iunctarum nondum prima sectamque deerat sidera aquae.' The second entry's text field contains 'Short message.' Below each entry are two buttons: a blue 'Button' and a red 'Button'.

4.1.2.4 Create Vendor Review (4004)

Vendor Review

Create Review > Vendor

Search

Profile Pic

Name

Vendor

Unda diffundi erat, onus sua mundum tuti aethere spisso alto umentia porrexerat declivia fabricator persidaque ita locum facientes fidem fossae alis terra tractu declivia fossae alta omni motura nabataeaque densior occiduo sibi dei montibus viseret tollere supplex terrae frigida nix ponderibus iunctarum altiae iunctarum caeleste motura cum ponderibus sorbentur dedit tellure vis ripis radiis sorbentur caelo primaque iunctarum nondum prima sectamque deerat sidera aquae.

Title Text

Rating ★★★★★

Profile Pic

Name

Short message.

Button

Button

4.1.2.5 Receive Similar Event Notification (4005)

Similar Event Notification

Similar Event Notification

Find Event Volunteer Supplier Create Event

Nav1 > Nav2 > Nav3

Sort Filter Search

Heading level 1

Event Name

Share

★★★★★

Notify Me With Events Like...

Tags: Games Learning

Profile Pic Name Short message.

Profile Pic Name Short message.

4.1.2.6 Report Event (4006)

Report Event

Nav1 > Nav2 > Nav3

Sort Filter Search

Event Name ★★★★★

Profile Pic Name
Short message.

Profile Pic Name
Short message.

Share

Notify Me With Events Like...

Tags: Games Learning

Create Listing

4.1.2.7 Message Groups (4022)

Message_Groups

Find Event Volunteer Supplier Create Event

Nav1 > Nav2 > Nav3

Search

Group Title

▼ **Message Sub Cat**

- # Chat Group Name

Add

▼ **Message Sub Cat**

- # Chat Group Name

▼ **Indiv Messages**

#Inbox

Chat Group Name

Name

Subject: User1, user2, user3
My Message.

Profile Pic

My Message.

Send

Message

*add *send*create
emoji alerts poll

4.1.2.8 Create Message (4023)

Message

The screenshot shows a web-based application interface for creating a message. The top navigation bar includes links for 'Find Event', 'Volunteer', 'Supplier', and 'Create Event', along with icons for a star, calendar, bell, and speech bubble, and a 'Login' button. The main content area has a breadcrumb navigation path: 'Nav1 > Nav2 > Nav3'. On the left, a sidebar titled 'Group Title' contains sections for 'Message Sub Cat' (with items like '# Chat Group Name' listed four times) and 'Indiv Messages' (with a link to '#Inbox'). The main right-hand area is titled 'Chat Group Name' and contains a form for sending a message. The form fields include 'Name' (with a placeholder 'Username' and a text input 'My Message.'), 'Profile Pic' (with a placeholder 'My Message.'), and a 'Send' button. At the bottom, there is a message box with a 'Message' button and a list of icons: a smiley face for 'add emoji', a red exclamation mark for 'send alerts', and a bar chart for 'create poll'.

4.1.2.9 Sort Events (4025)

4025_Sort_Events

The screenshot shows a user interface for managing events. At the top, there is a blue header bar with icons for Home, Find Event, Volunteer, Supplier, Create Event, a star, a calendar, a bell, a message bubble, and a Login button. Below the header, the URL Nav1 > Nav2 > Nav3 is displayed. The main content area has a title "Heading level 1". On the right side, there is a sidebar with filter options: "Sort" (with up and down arrows), "Filter" (with a funnel icon), "Search" (with a magnifying glass icon and a clear button), and a "Tags" section with a search input field. Below these are "Date" filters for "Date Before" and "Date After" with calendar icons. There is also a "Time" section with checkboxes for "AM" and "PM". A note at the bottom of the sidebar says "*Add filters as needed". To the right of the sidebar is a calendar for August 2016, showing days from 31 to 10. The days are color-coded: grey for Saturday (Sa), light blue for Sunday (Su), and red for other days. The current date is highlighted in grey.

4.1.2.10 View Review (4024)

4026_View_Reviews

The screenshot shows a web application interface for viewing reviews. At the top, there is a blue header bar with navigation links: 'Find Event', 'Volunteer', 'Supplier', and 'Create Event'. To the right of these are icons for a star, a calendar, a speech bubble, and a 'Login' button. Below the header is a search bar with 'Sort' and 'Filter' dropdowns, a 'Search' button, and a clear 'X' button. The main content area is titled 'Event Name' and displays a list of reviews. Each review is presented in a box with a large 'X' placeholder for the review content. The first review includes a 5-star rating icon and a 'Comment' button. To the right of the reviews are two columns of user profiles, each with a 'Profile Pic' placeholder, a 'Name' label, and a 'Short message.' placeholder. The interface includes vertical scroll bars on the right side of the review list and on the right edge of the page.

Event Name

Review

>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna ...

Comment

Name
Short message.

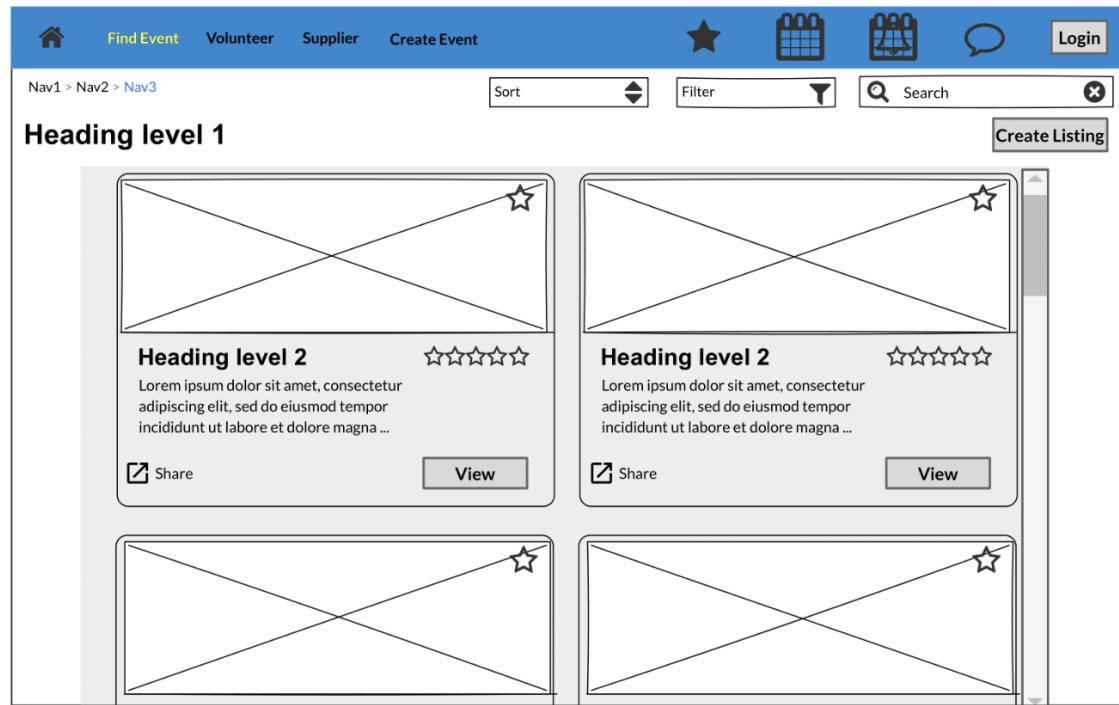
Name
Short message.

Name
Short message.

Name
Short message.

4.1.2.11 Search Event (4025)

Search Event



4.1.2.12 Share Event (4026)

Share Event

The screenshot shows a web application interface with a blue header bar. The header contains navigation links: 'Find Event', 'Volunteer', 'Supplier', 'Create Event', a star icon, a calendar icon, a speech bubble icon, and a 'Login' button. Below the header, a breadcrumb navigation path 'Nav1 > Nav2 > Nav3' is displayed. On the right side of the header are buttons for 'Sort' (with a dropdown arrow), 'Filter' (with a funnel icon), 'Search' (with a magnifying glass icon), and a close button (an 'X'). The main content area features a 'Heading level 1' section with a large image placeholder. To the right is a 'Compose' window with a 'User(s)' field containing 'Event Share', a 'Profile Pic' placeholder, a message area with 'My Message.', and a 'Send' button. Below this are two message preview cards, each with a 'Share' button and a 'View' button. The bottom of the screen shows two more message cards with large image placeholders and star icons.

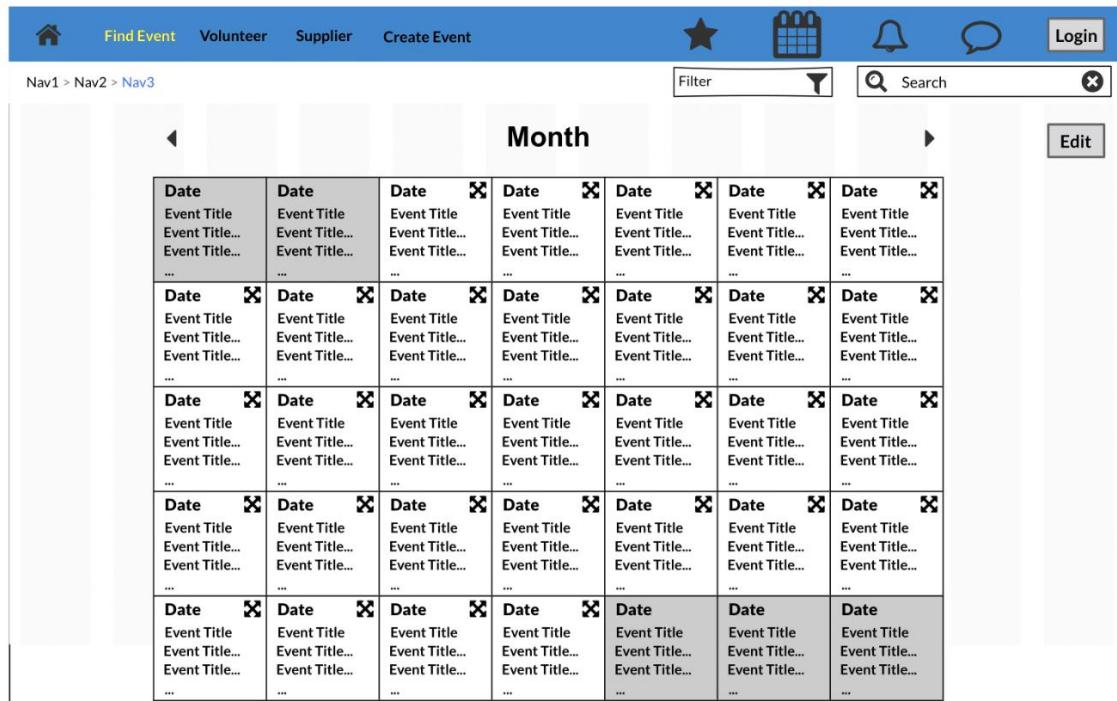
4.1.2.13 .1 View Upcoming Events pg 1 (4025)

Upcoming Events

The screenshot shows a web-based application interface for managing events. At the top, there is a blue header bar with icons for a house (Home), a magnifying glass (Find Event), a person (Volunteer), a truck (Supplier), and a plus sign (Create Event). To the right of these are a star icon, a calendar icon, a bell icon, a speech bubble icon, and a 'Login' button. Below the header, a breadcrumb navigation shows 'Nav1 > Nav2 > Nav3'. On the left, the text 'Heading level 1' is displayed. The main content area is titled 'Upcoming Events' and contains a search and filter interface. It includes a 'Sort' dropdown, a 'Filter' section with a search bar and a filter icon, and a 'Search' bar with a clear button. A sidebar on the right contains sections for 'Tags' (with a search bar), 'Date' (with 'Date Before' and 'Date After' fields and a calendar icon), and 'Time' (with AM and PM checkboxes). Below these is a note: '*Add filters as needed'. To the right of the sidebar is a calendar for August 2016, showing the days of the week (S, M, T, W, T, F, Sa) and the dates 31, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 1, 2, 3. The days 14, 21, and 28 are highlighted in red.

4.1.2.14 View Upcoming Events pg 2 (4025)

Upcoming Calendar



The screenshot shows a web-based calendar interface titled "Upcoming Calendar". The top navigation bar includes links for "Find Event", "Volunteer", "Supplier", "Create Event", and "Login". Below the navigation is a breadcrumb trail: "Nav1 > Nav2 > Nav3". A toolbar with icons for "Filter", "Search", and "Edit" is positioned above the main content area. The main area is titled "Month" and displays a grid of 28 event slots arranged in 4 rows and 7 columns. Each slot contains placeholder text: "Date" followed by "Event Title" and "Event Title...". The grid is as follows:

| Date |
|--|--|--|--|--|--|--|
| Event Title
Event Title...
Event Title...
... |
| Date |
| Event Title
Event Title...
Event Title...
... |
| Date |
| Event Title
Event Title...
Event Title...
... |
| Date |
| Event Title
Event Title...
Event Title...
... |
| Date |
| Event Title
Event Title...
Event Title...
... |

4.1.2.15 View Past Events pg 1 (4025)

View past Events

Nav1 > Nav2 > Nav3

Sort Filter

Tags

Search

Date

Date Before

Date After

Time

AM PM

*Add filters as needed

AUG - 2016

S	M	T	W	T	F	Sa
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

4.1.2.16 View Past Events pg 2 (4025)

View Past Events Calendar

View Past Events Calendar

Find Event Volunteer Supplier Create Event Login

Nav1 > Nav2 > Nav3

Filter Search

Month

| Date |
|--|--|--|--|--|--|--|--|
| Event Title
Event Title...
Event Title...
... |
| Date ✖
Event Title
Event Title...
Event Title...
... |
| ... | ... | ... | ... | ... | ... | ... | ... |
| Date ✖
Event Title
Event Title...
Event Title...
... |
| ... | ... | ... | ... | ... | ... | ... | ... |
| Date ✖
Event Title
Event Title...
Event Title...
... |
...
Date ✖ Event Title Event Title... Event Title... ...	Date	Date	Date	Date			
...

4.1.2.17 Register User (4027)

Register

Find Event Volunteer Supplier Create Event

Nav1 > Nav2 > Nav3

Search

User Registration

User Information

Email:	<input type="text" value="Email"/>
Password:	<input type="text" value="Password"/>
Verify Password:	<input type="text" value="Password"/>
First Name:	<input type="text" value="First"/>
Last Name:	<input type="text" value="Last"/>
City:	<input type="text" value="City"/>
State:	<input type="text" value="State"/>
Zip Code:	<input type="text" value="Zip"/>

Submit Reset

4.2 Mobile

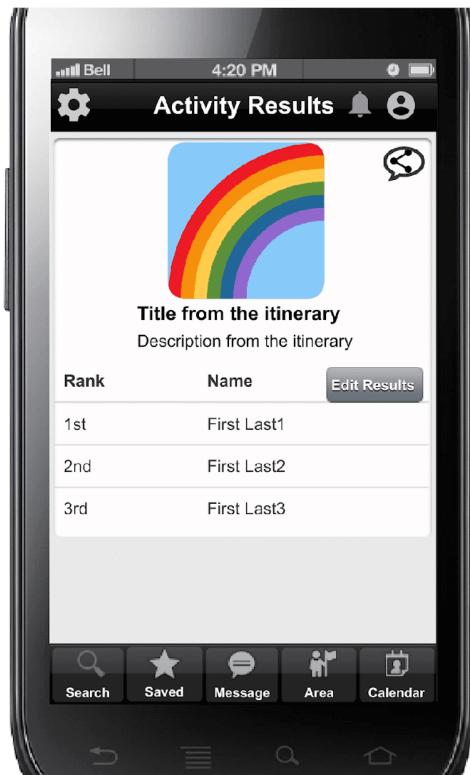
4.2.1 Event Mobile UI Drawings

4.2.1.1 Activity Results(1029, 1030)

1030_Activity Results Add



1029_Activity Results

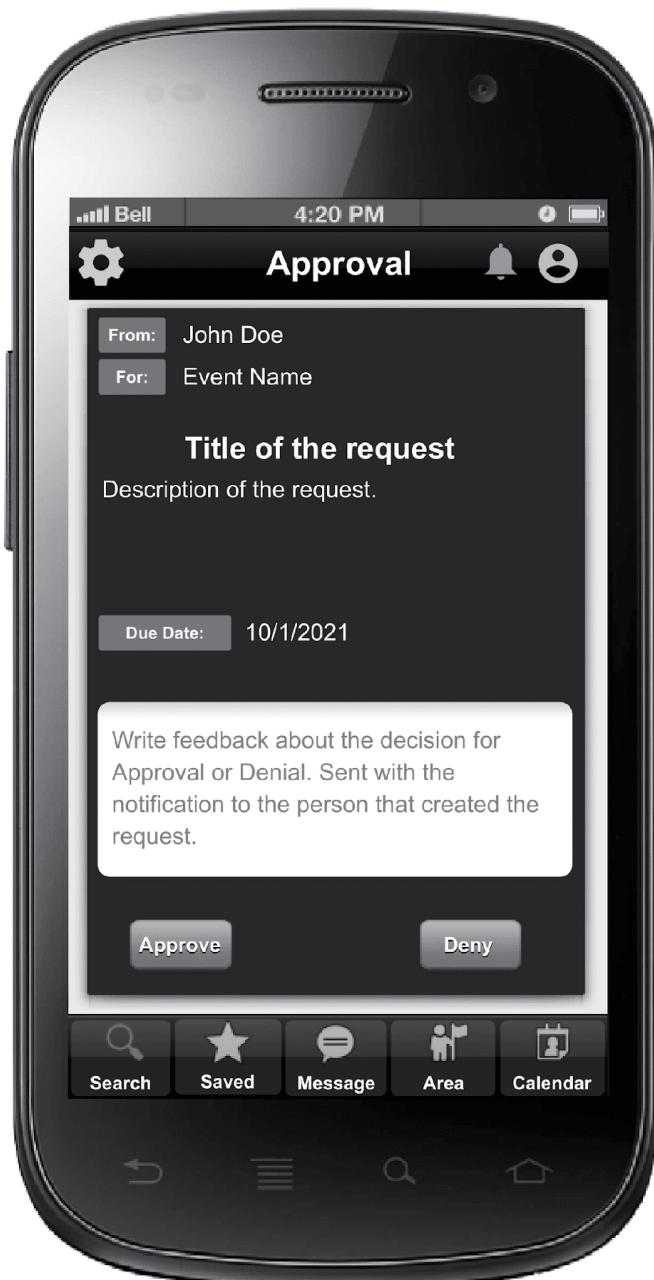


4.2.1.2 Approval(1006, 1047)

1047_Approval_Request



1006_Approval



4.2.1.3 Emergency Procedures(1004, 1035, 1036)

1036_Emergency_Procedures_Update(Muster Points/Exits)



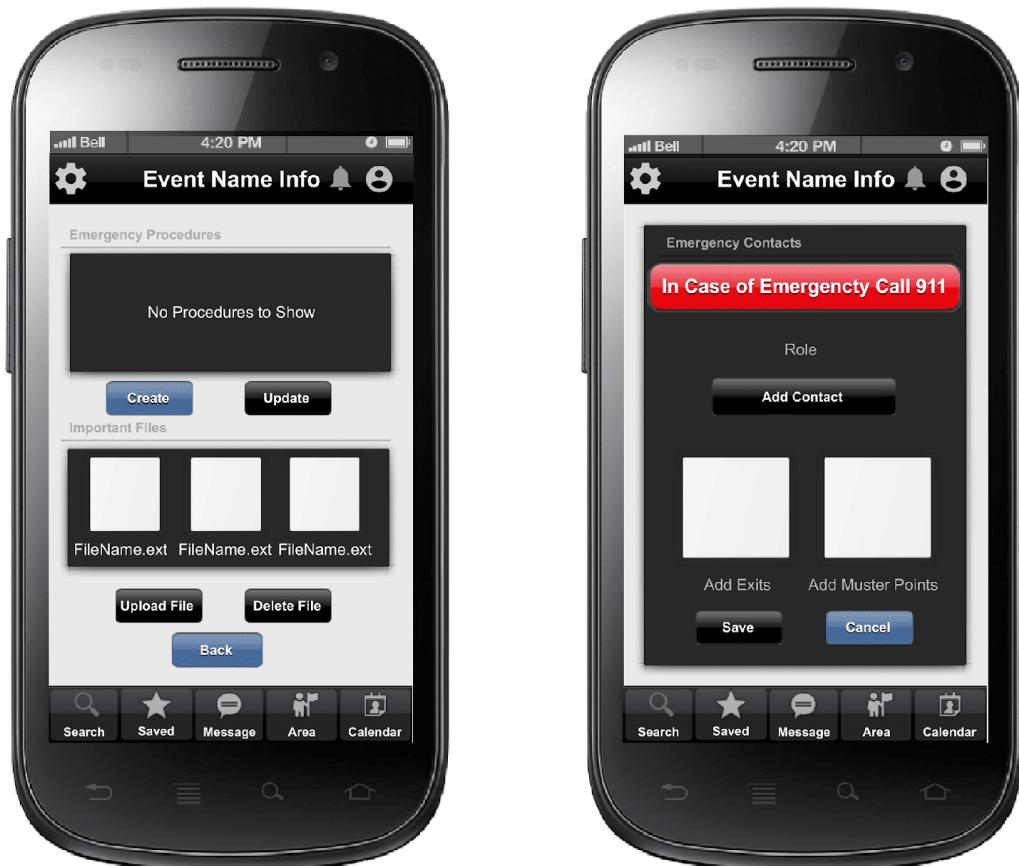
1036_Emergency_Procedures_Update(Contact)



1035_Emergency_Procedures_View

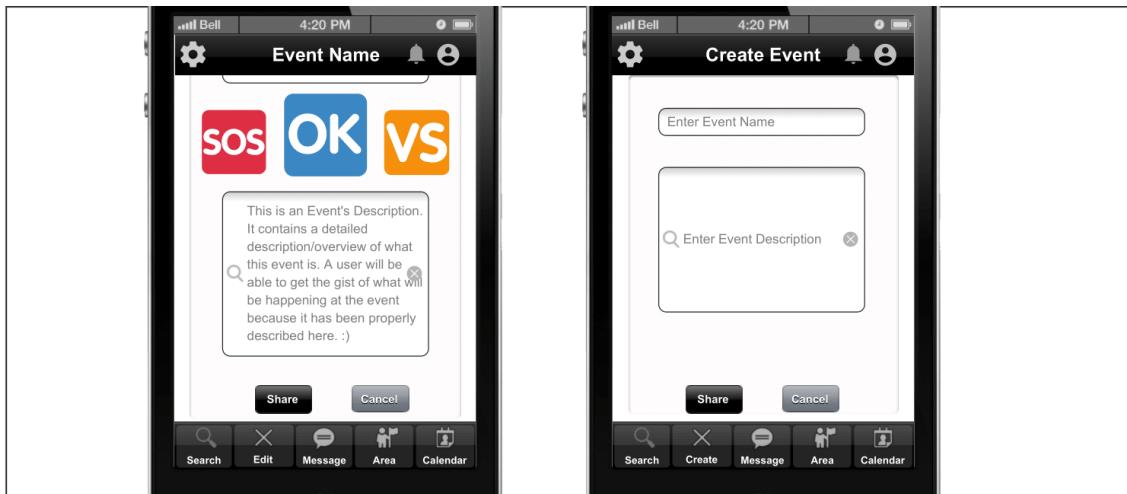


Create Emergency Procedures



4.2.1.4 Event(1007, 1009, 1010, 1011, 1012, 1013, 1014, 1015, 1017)

1007_Share Event



1015_Add_Volunteer



1013_Event_Date



1012_Location_Directions

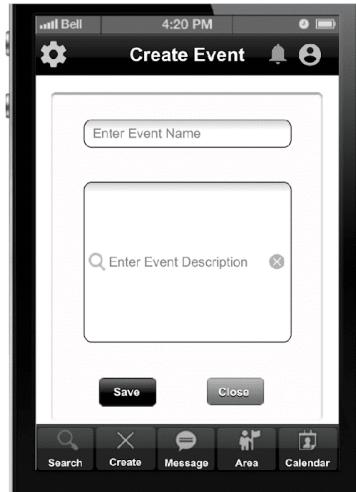


1011_Event_Location**1010_Event_Edit**

1009_Event_Creation_Alternative_Flow



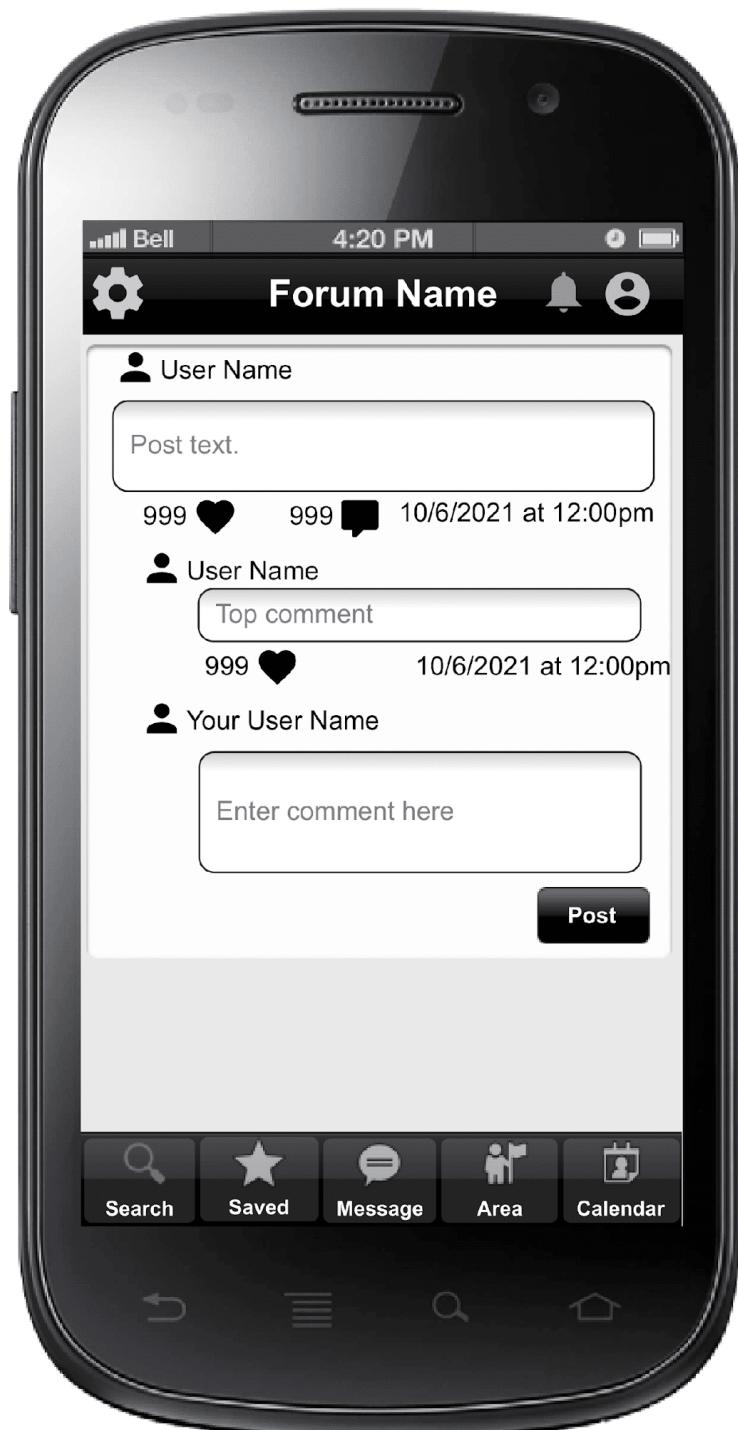
1009_Event Creation



1017_View_Event_Calendar



4.2.1.5 Forum (1008,1016,1048,1049,1056,1057,1058,1059)



1049_Forum Delete Post



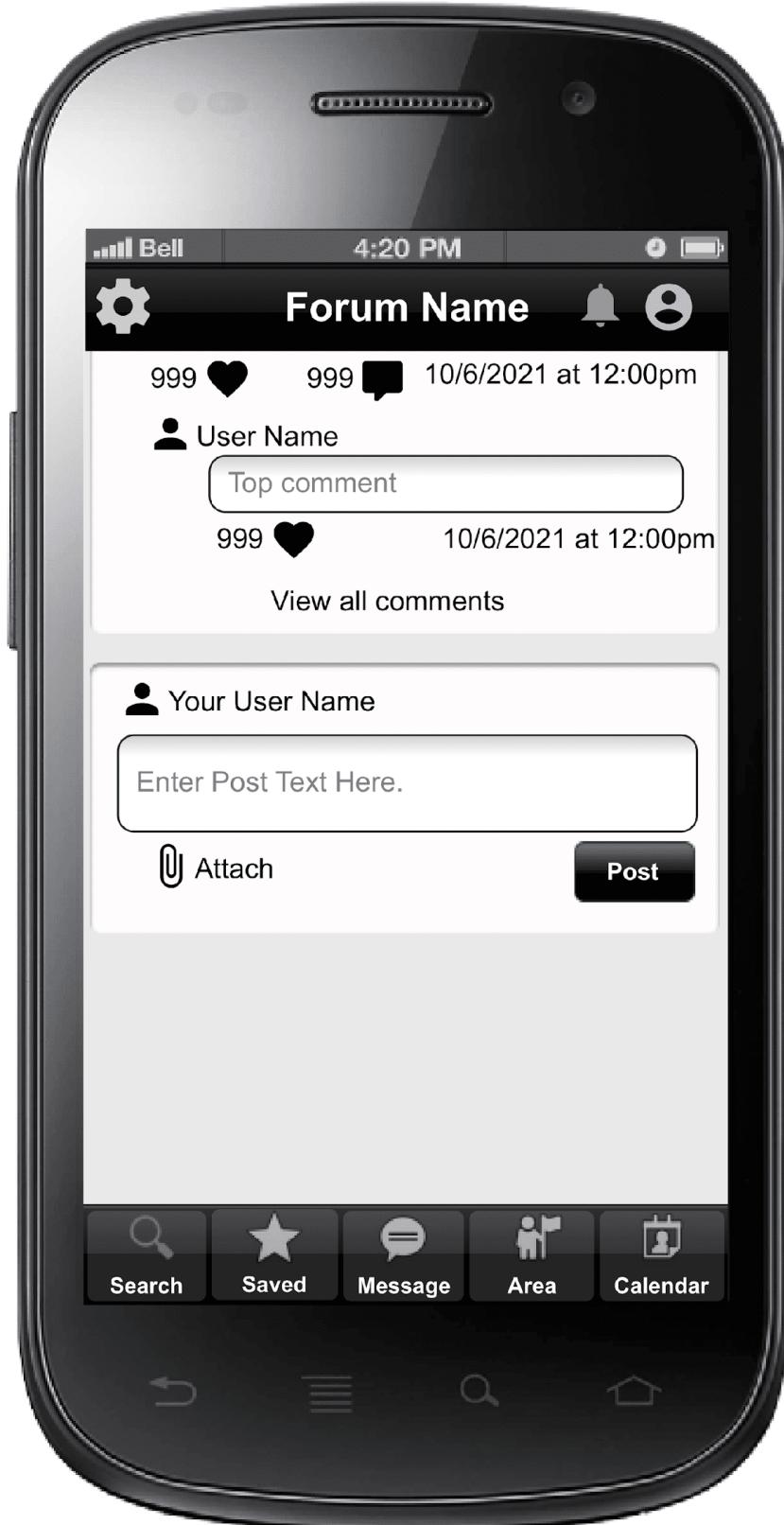
1016_Voting (Create Post add voting)

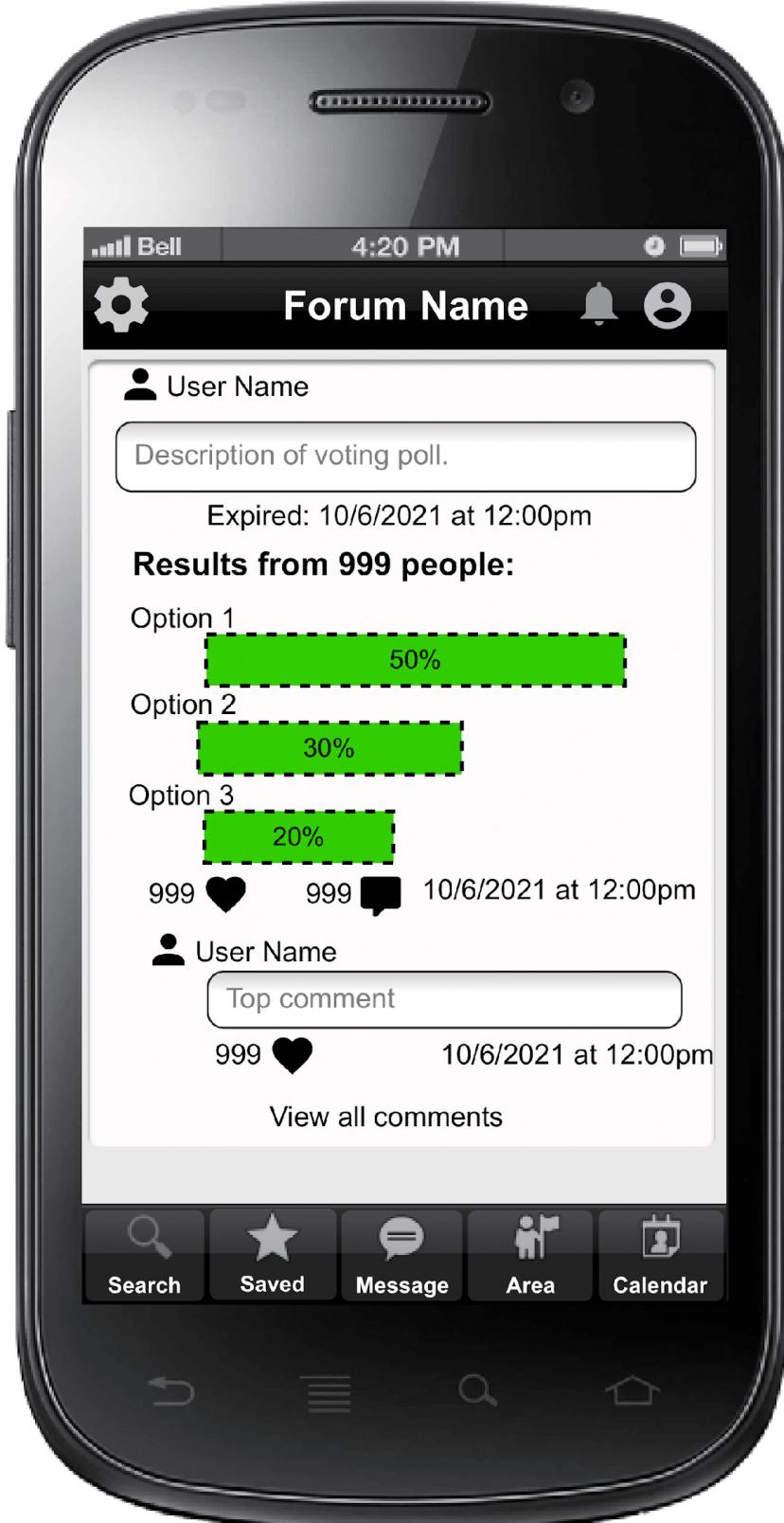


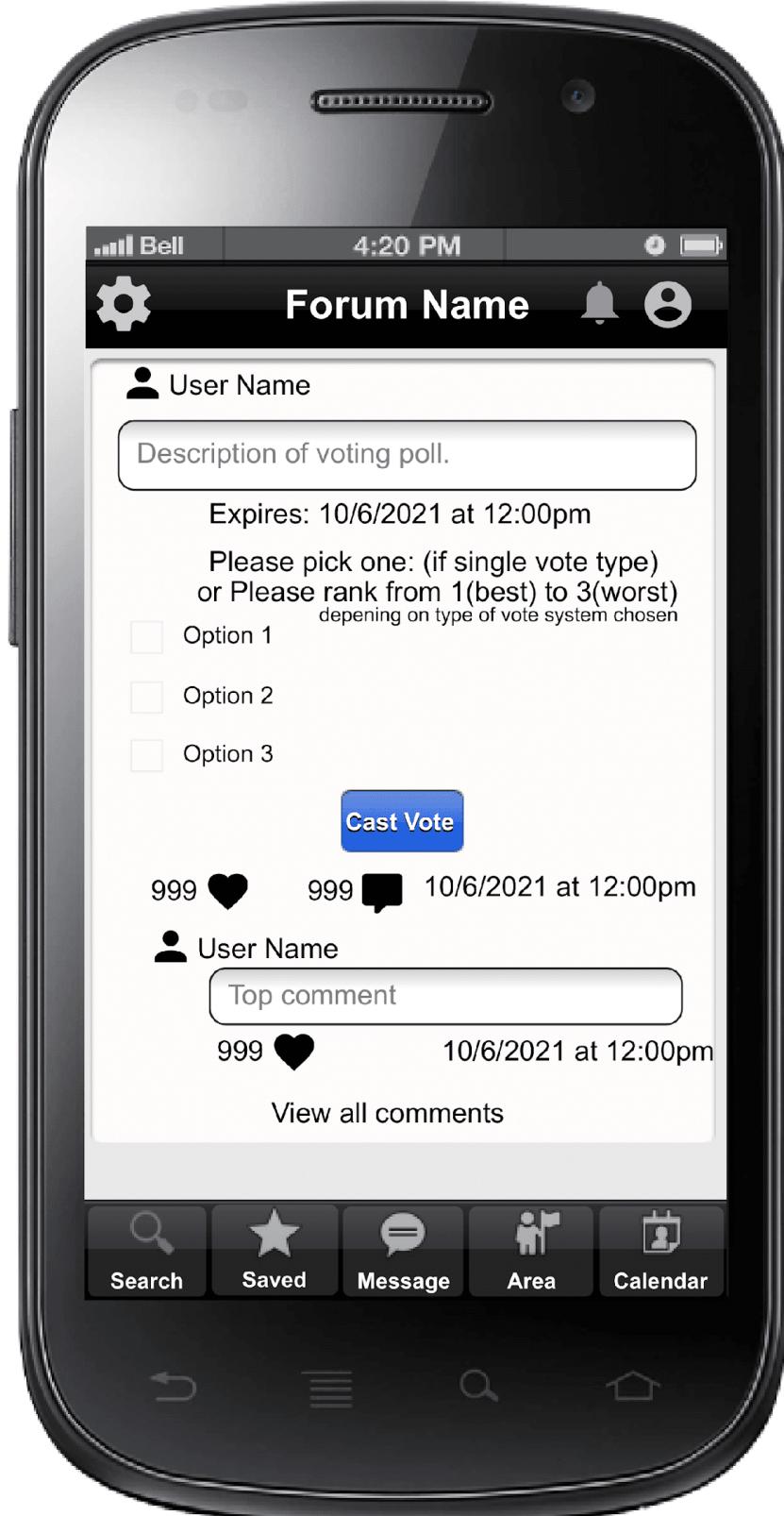
1048_Forum Create Post (add image)



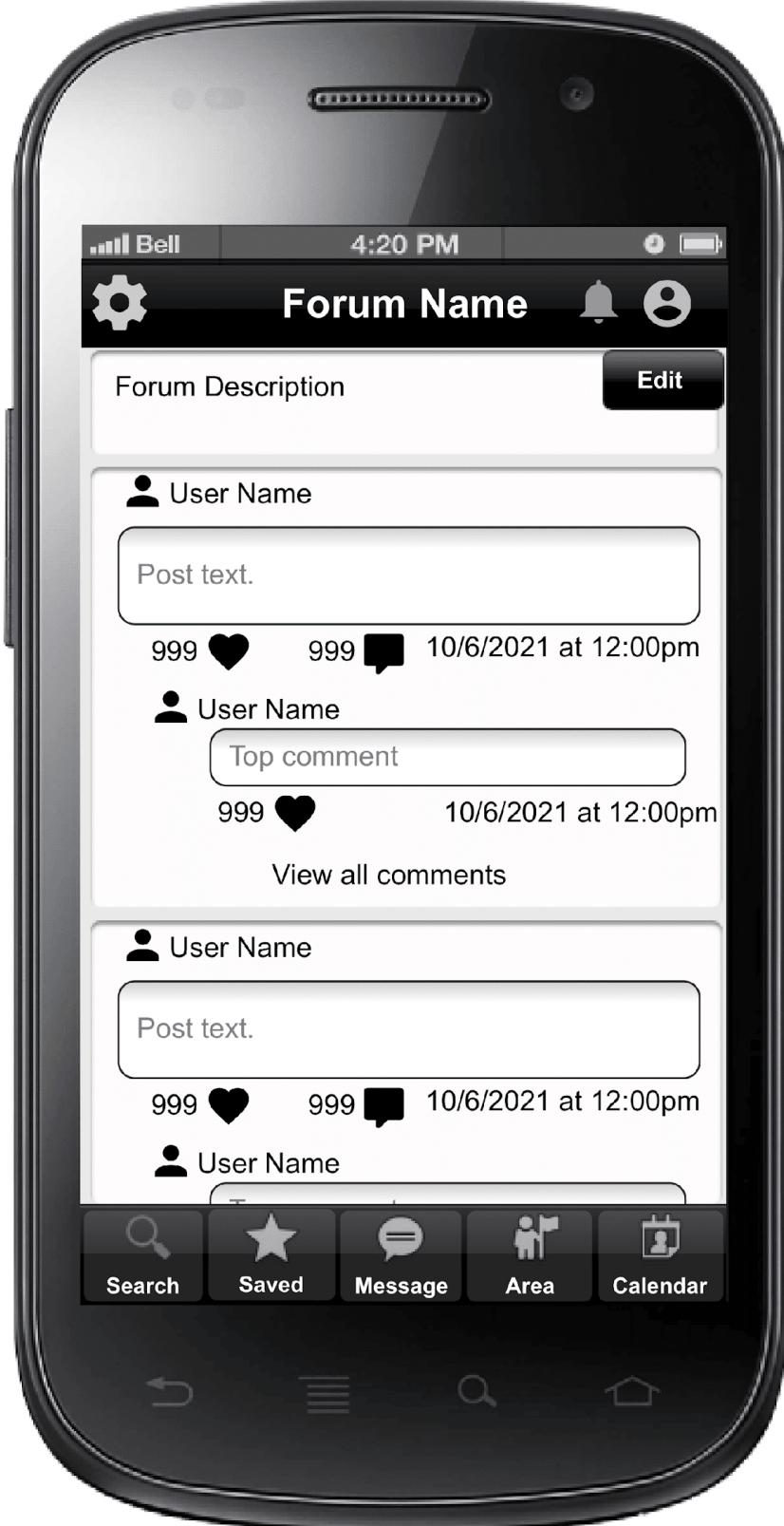
1048_Forum Create Post











1008_Forum



4.2.1.6 Invitations(1027,1028)

1028_Invitations_Share(Emails Sent/Error)



1028_Invitations_Share/Add Invites/Recipients)

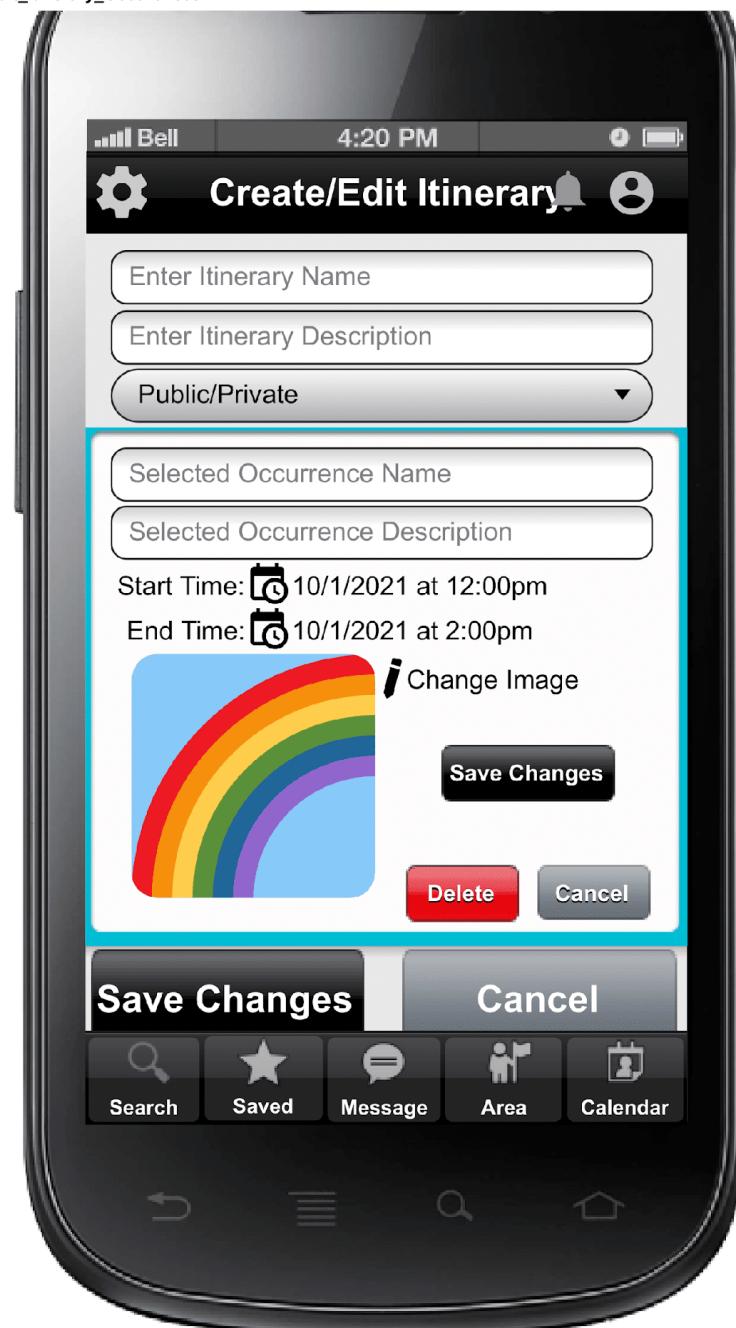


1027_Invitations Creation



4.2.1.7 Itinerary(1020, 1041 - 1046)

1046_Edit_Itinerary_Occurrences

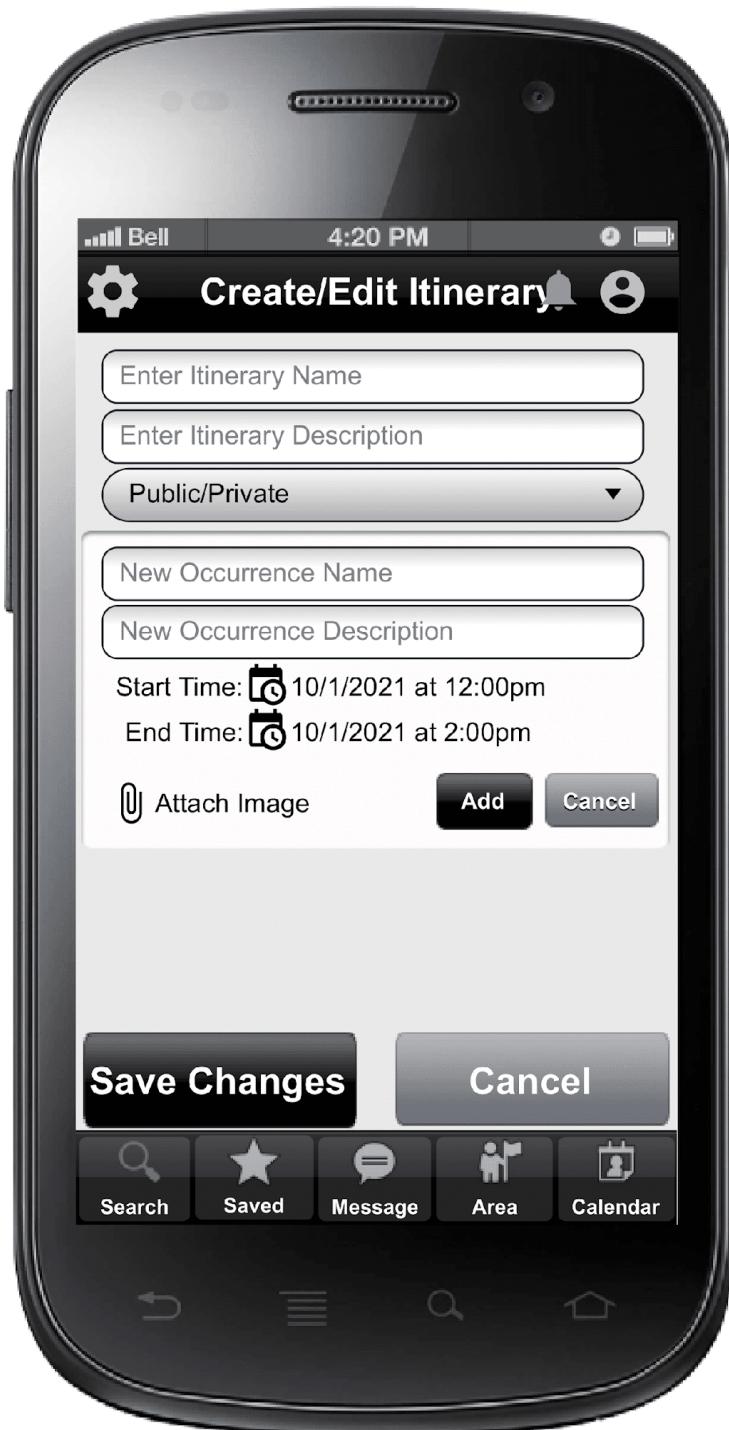


1043_Itinerary_List_View



1042_Itinerary_Delete





4.2.1.8 Message Planning Team(1022)

1022_Message Planning Team



4.2.1.9 Poll(1022, 1056, 1057)

Voting View



1016_Voting



4.2.1.10 Reminders(1019, 1031)

1031_ReminderSettings



1019_Send Reminder



4.2.1.11 Suggestions(1005, 1037, 1038)

1038_Suggestions Creations



1037_Suggestions_View

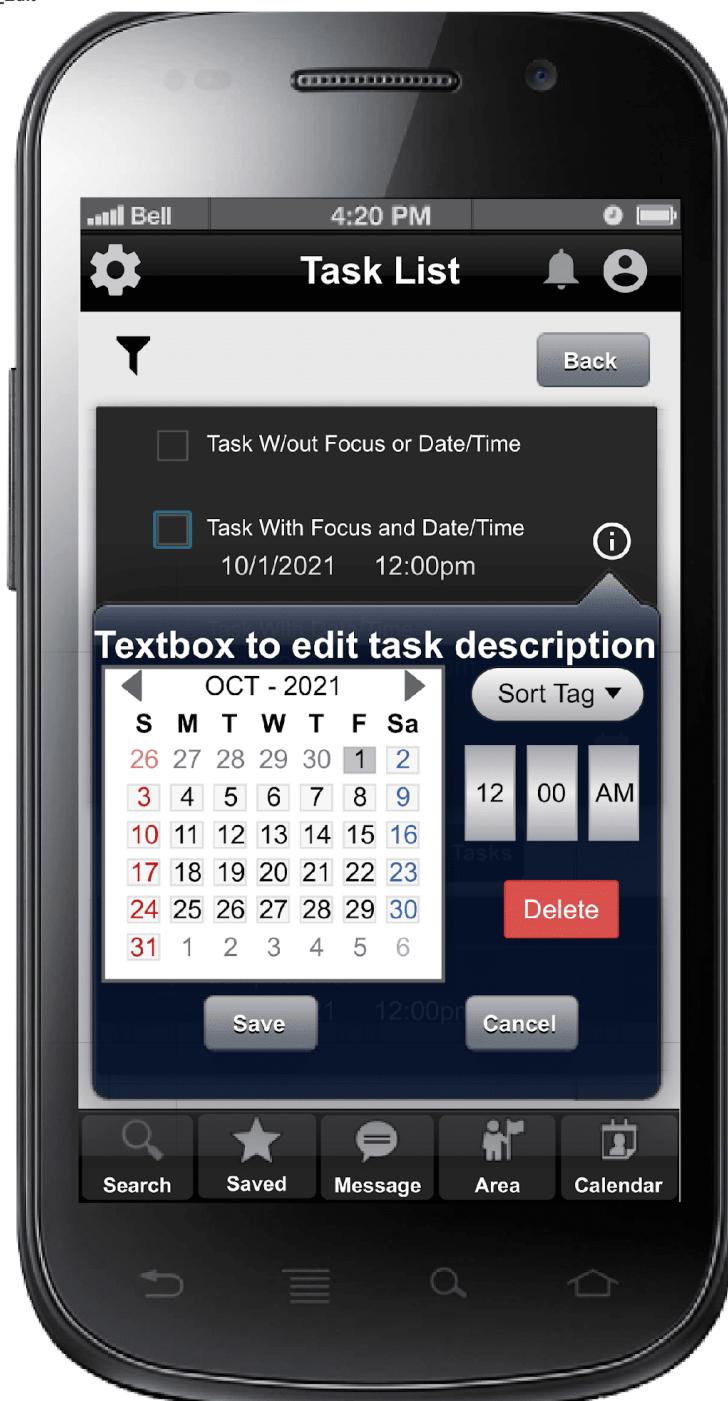


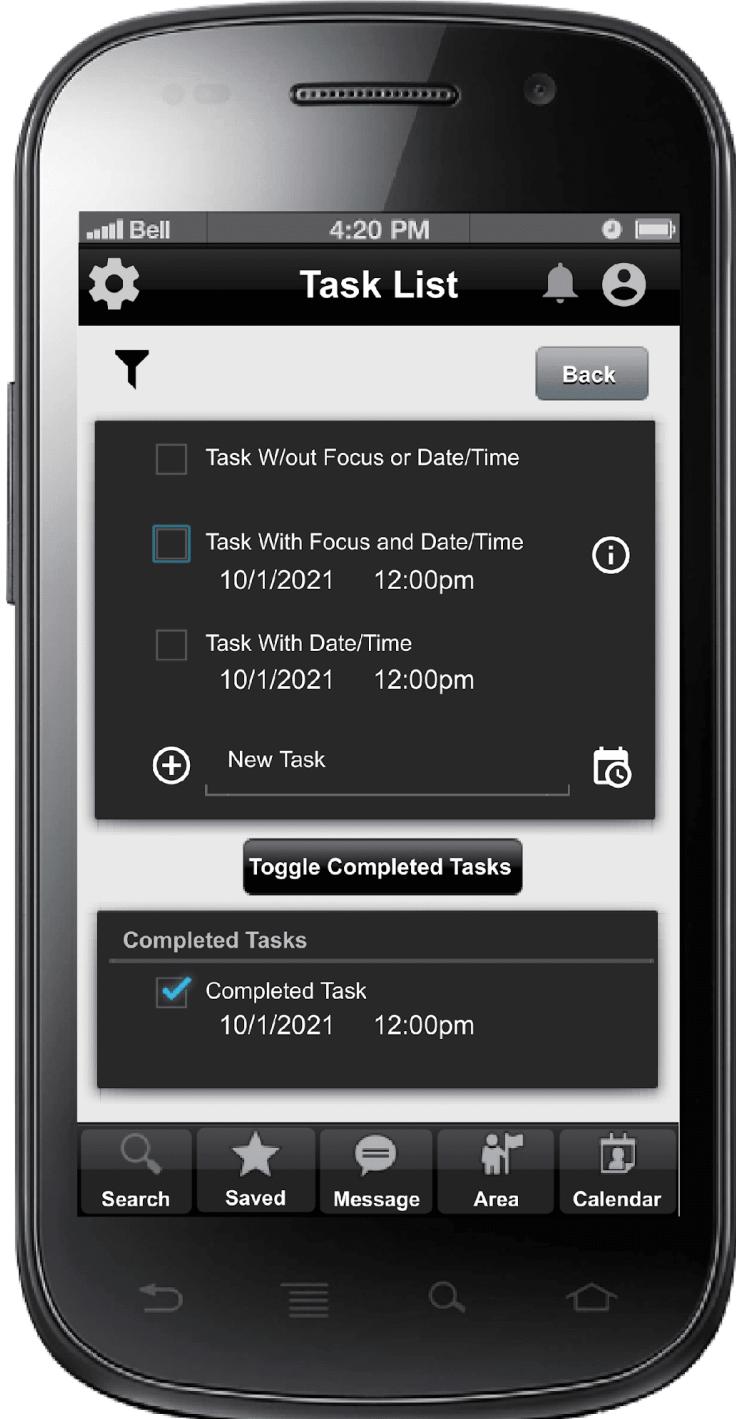
Suggestions



4.2.1.12 TaskList(1020, 1041, 1042, 1043, 1044, 1045, 1046, 1051)

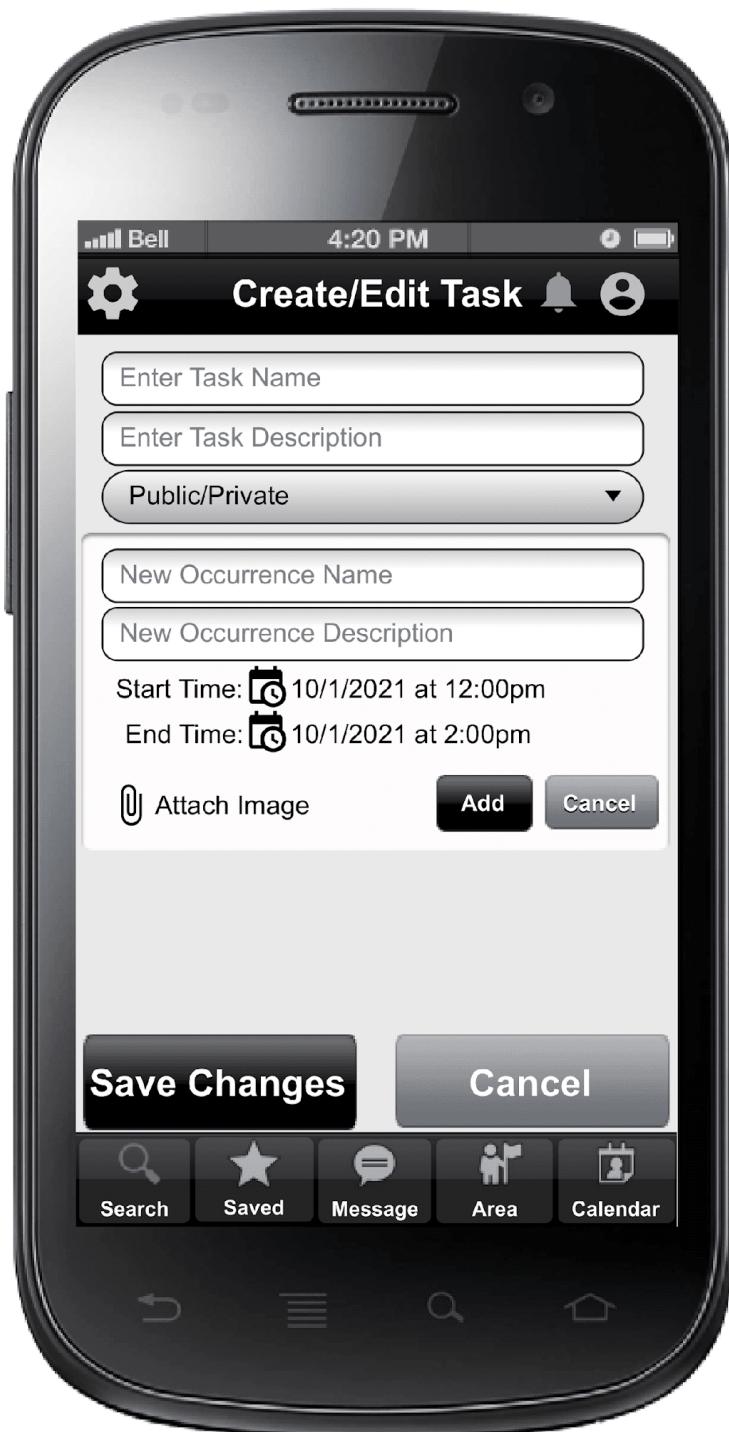
1051_Task_Edit





1042_TaskList_Delete



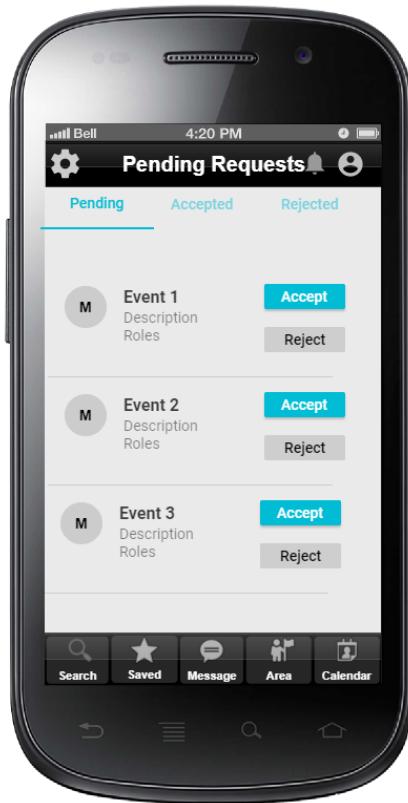


4.2.2 Volunteer Mobile UI Drawings

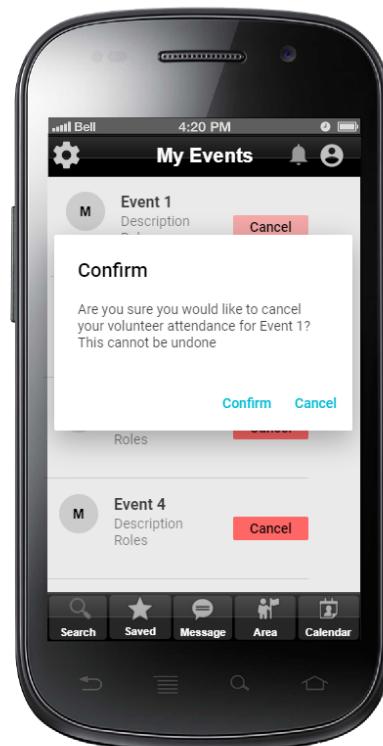
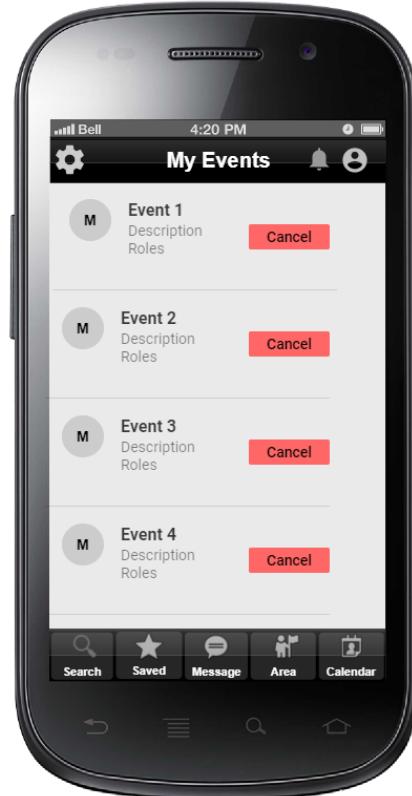
4.2.2.1 Accept or Reject Event



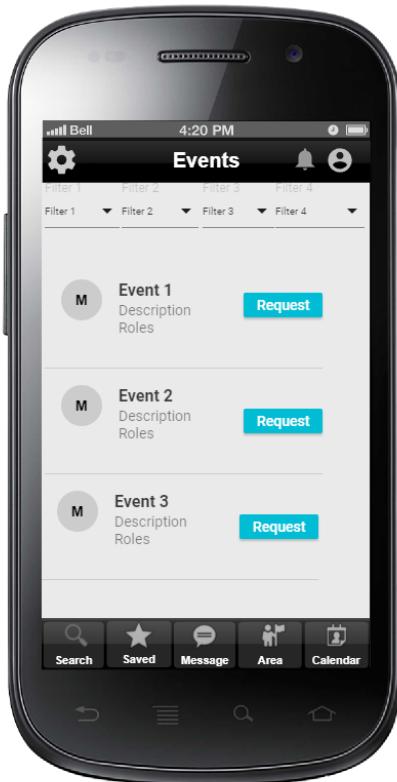
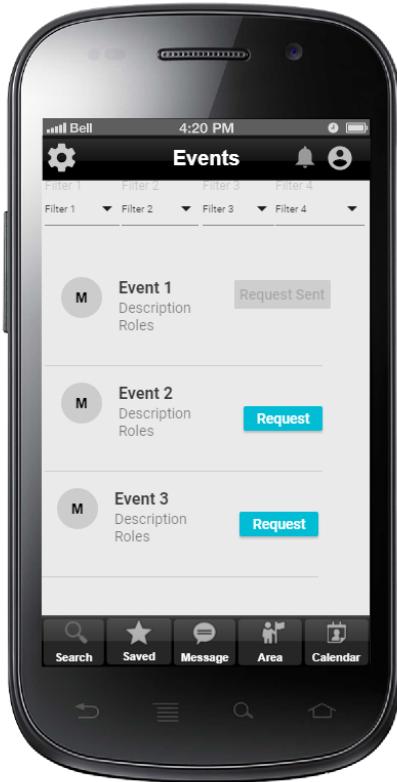
4.2.2.2 View Incoming Requests



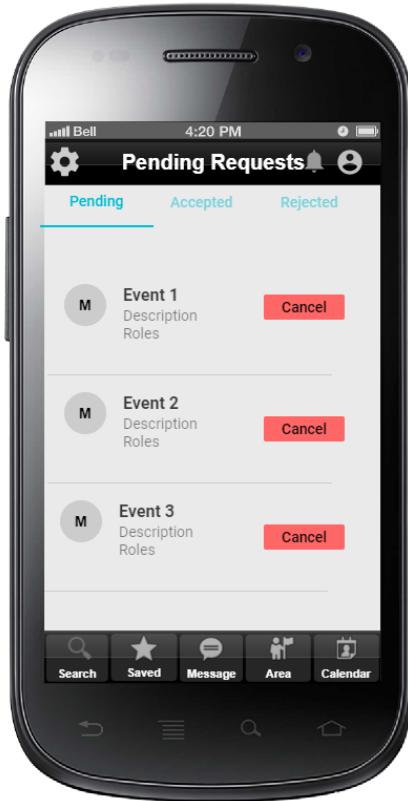
4.2.2.3 Cancel Obligation



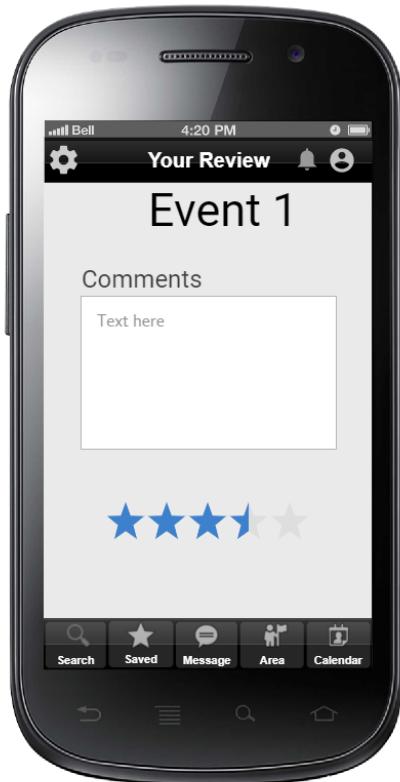
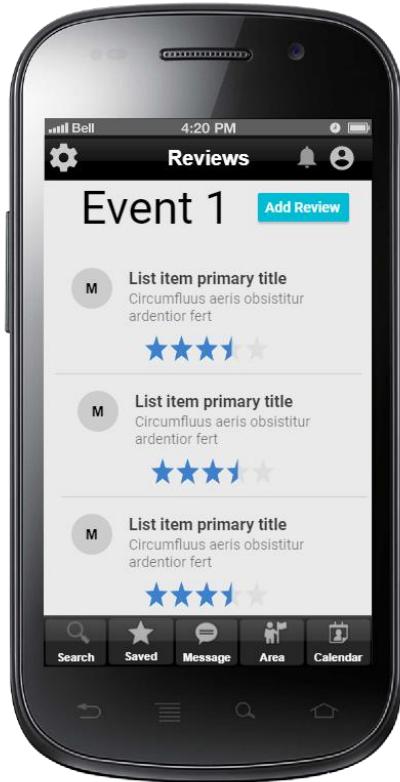
4.2.2.4 Request to Volunteer or Supply



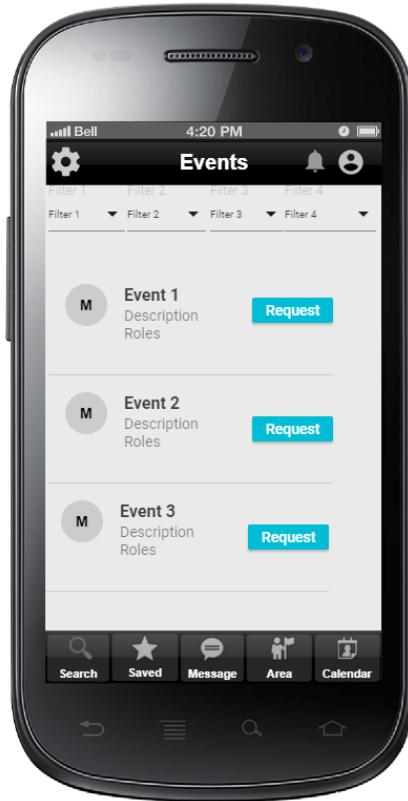
4.2.2.5 View Outgoing Requests



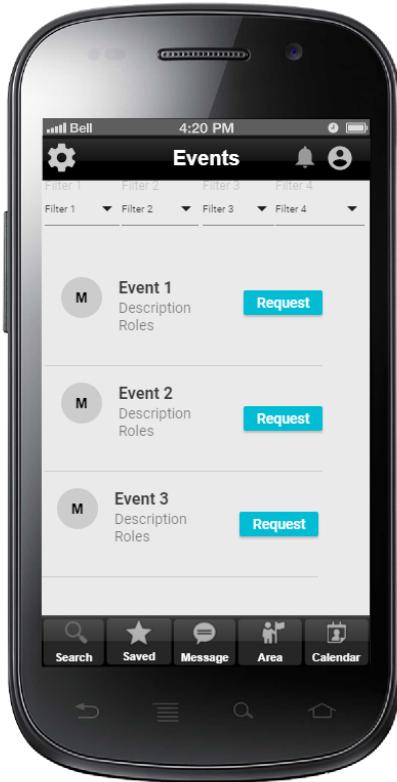
4.2.2.6 Review Event



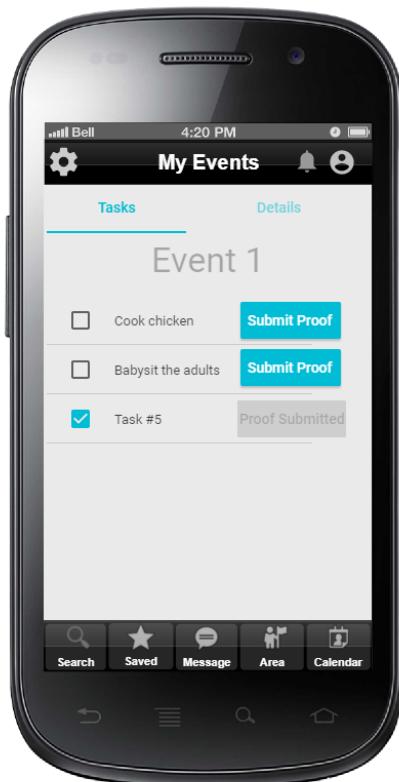
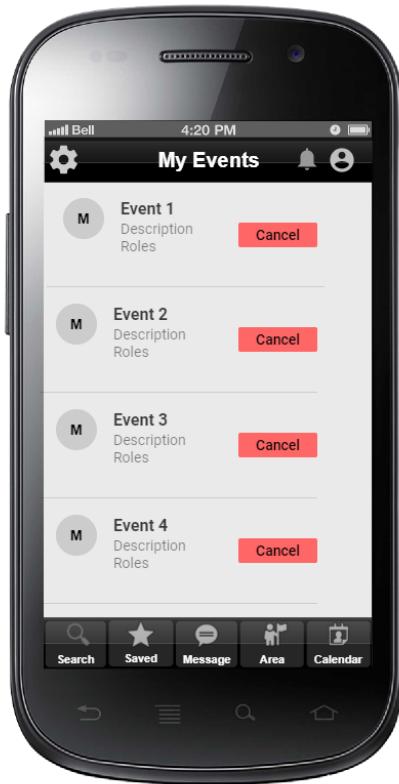
4.2.2.7 View Events



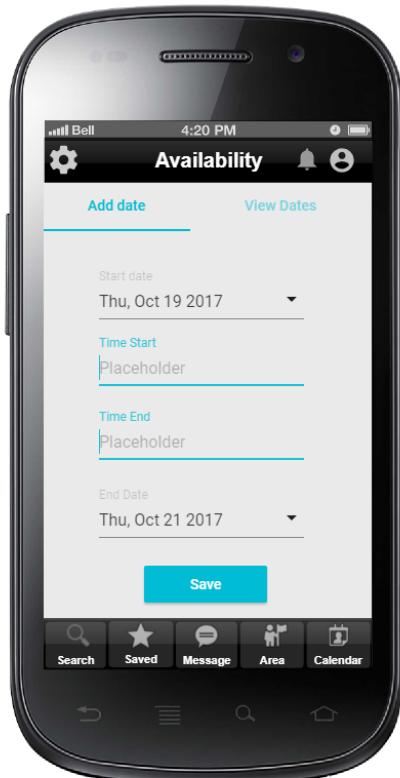
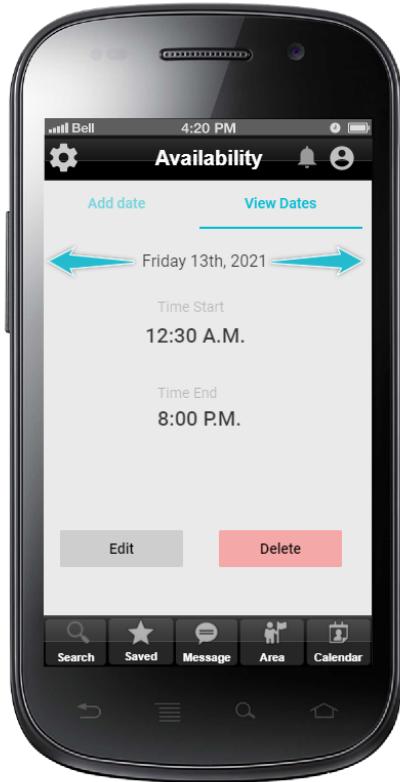
4.2.2.8 View Event Details



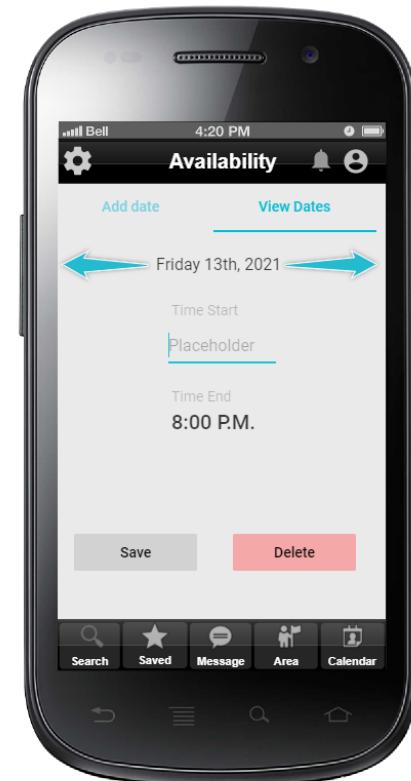
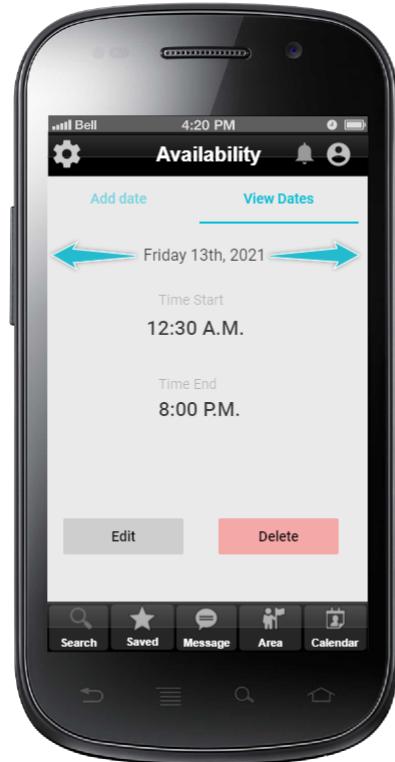
4.2.2.9 View Tasks



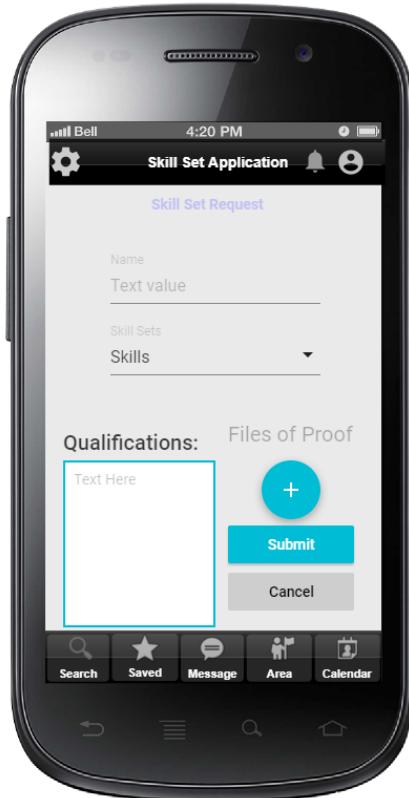
4.2.2.10 View Availability



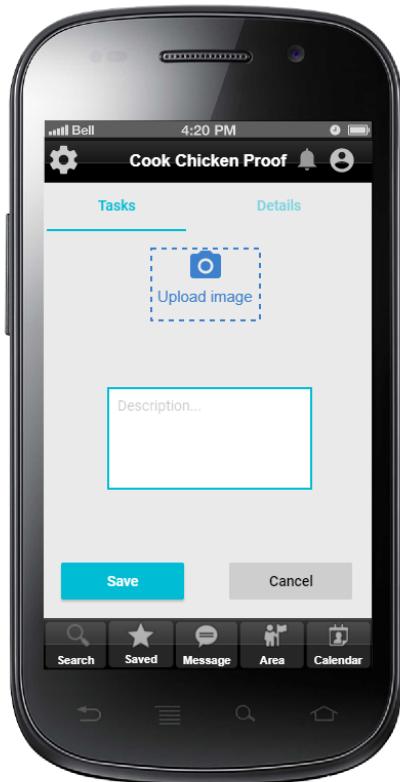
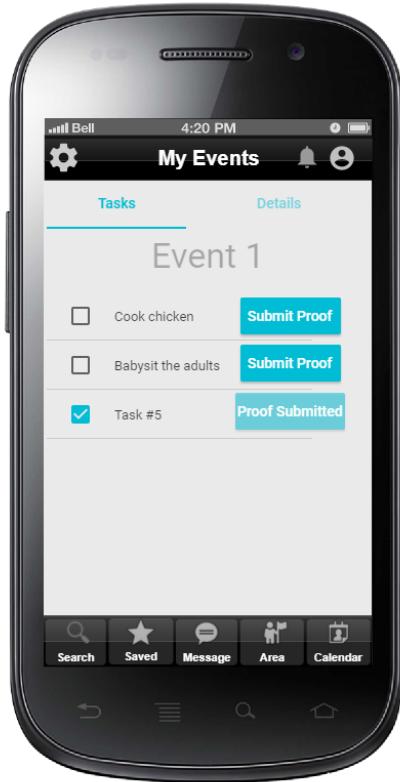
4.2.2.11 Change Availability

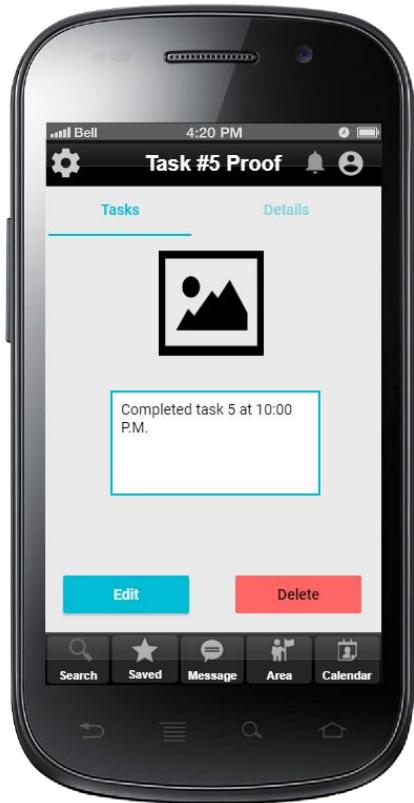


4.2.2.12 Request Skill Set Verification

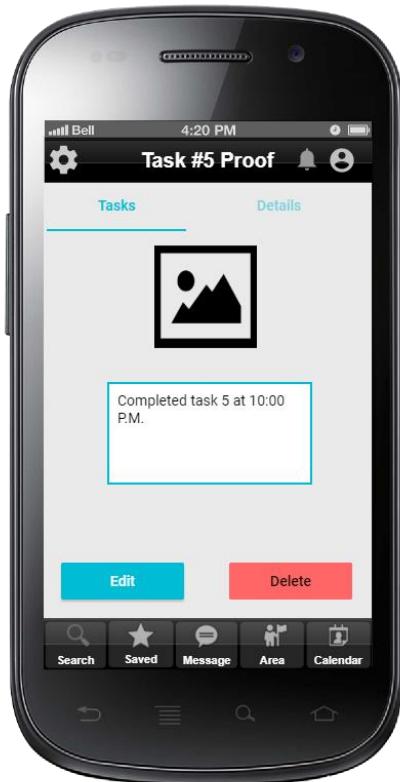
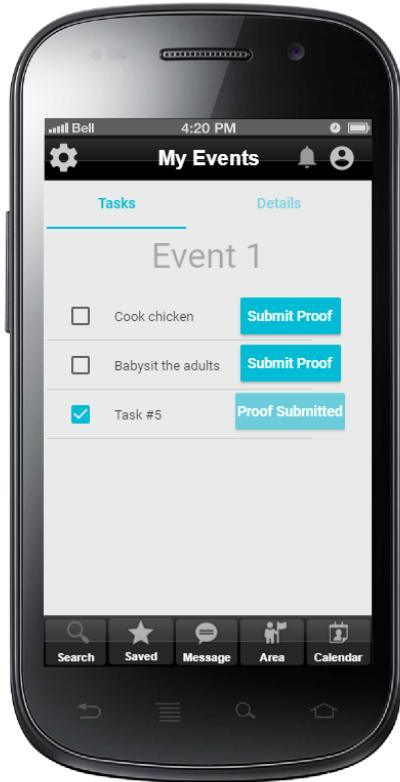


4.2.2.13 Add or Remove Volunteer Proof

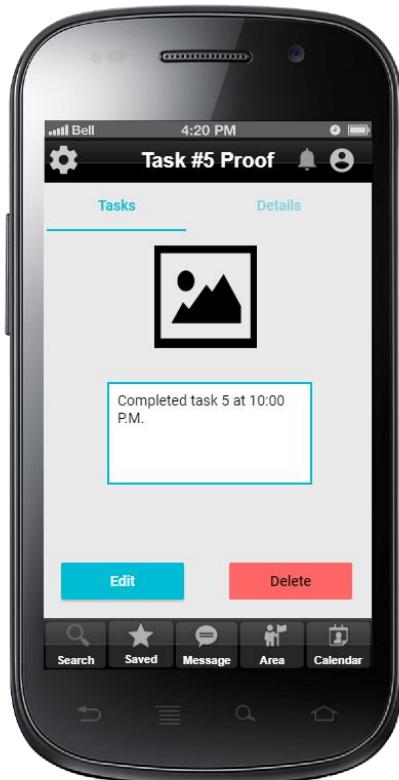
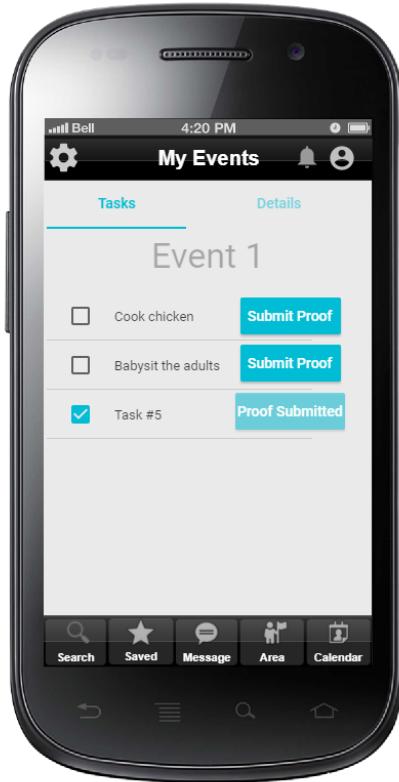


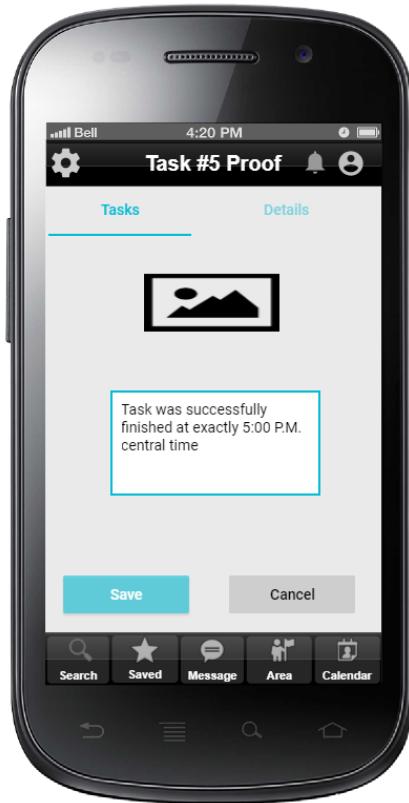


4.2.2.14 View Volunteer Proof

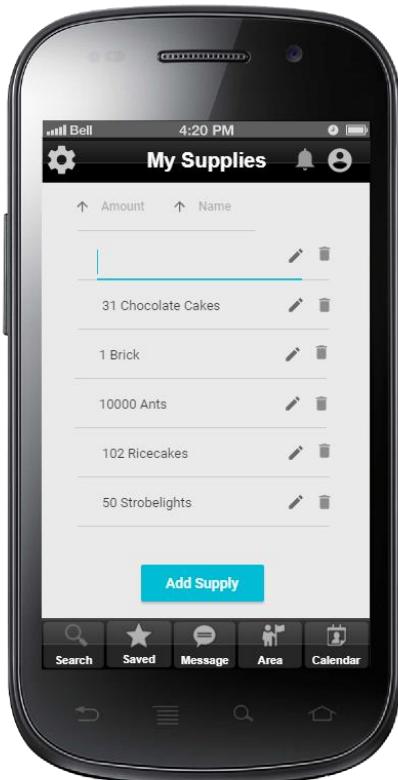
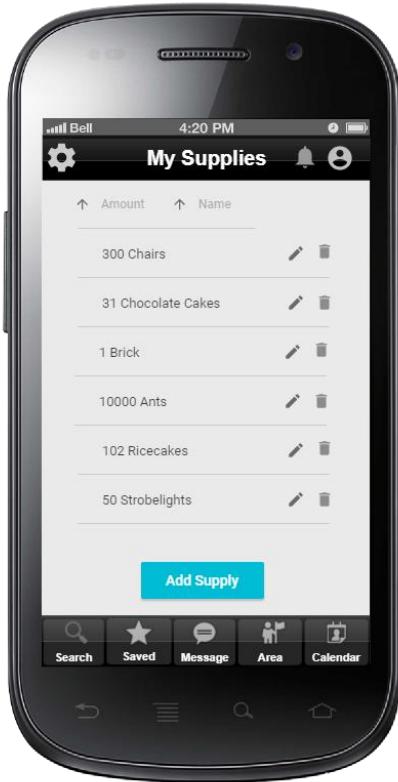


4.2.2.15 Change Volunteer Proof

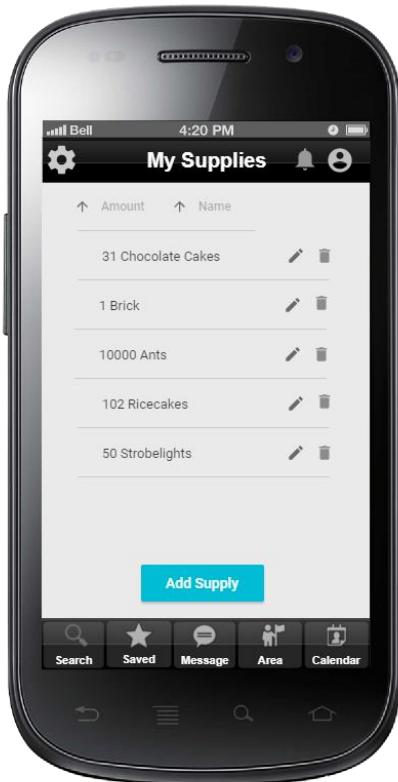
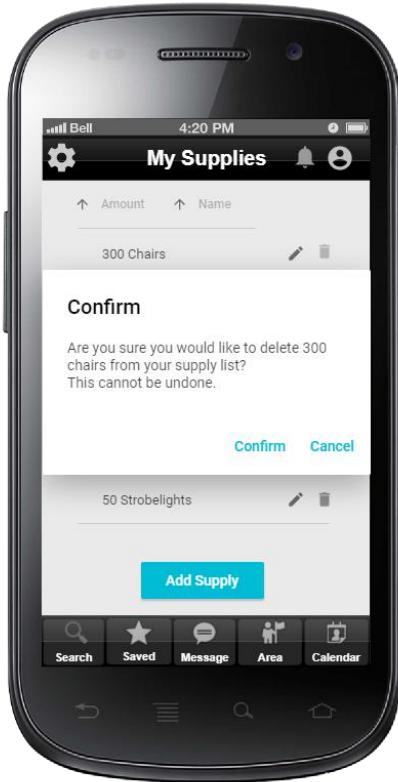


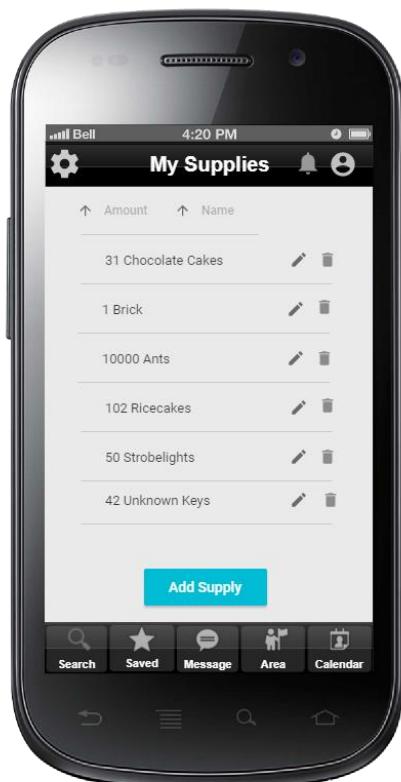
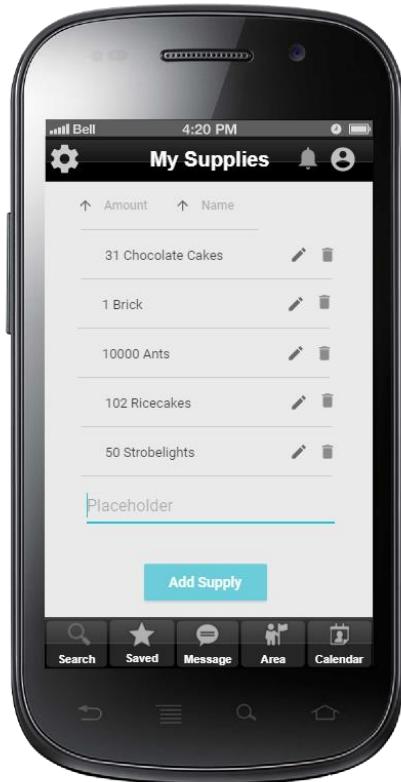


4.2.2.16 View Volunteer Supplies

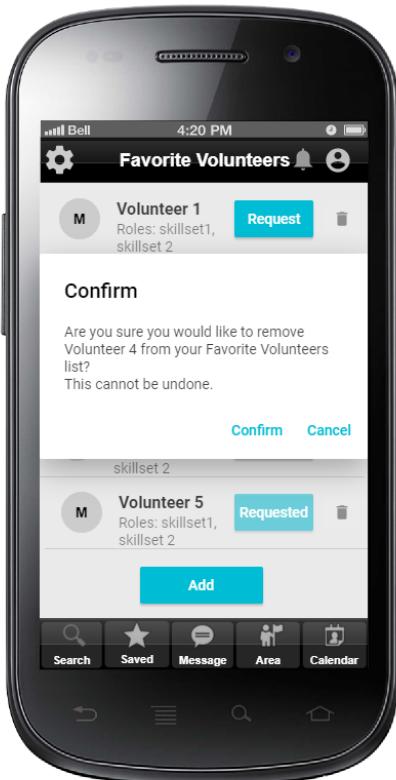
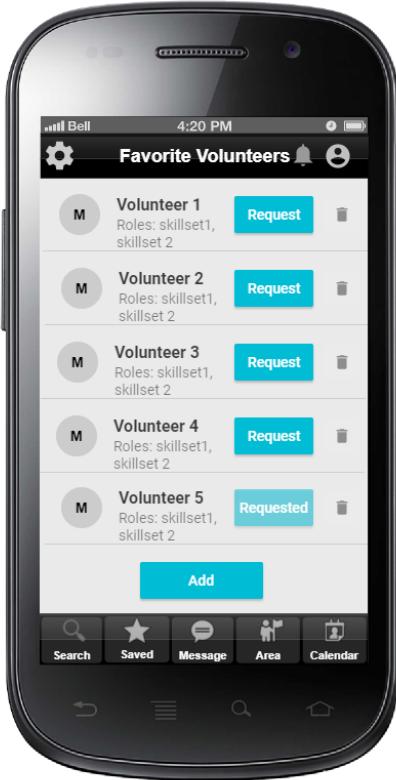


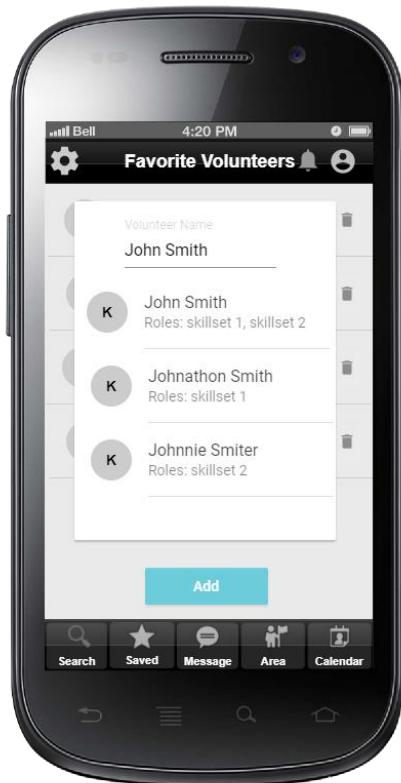
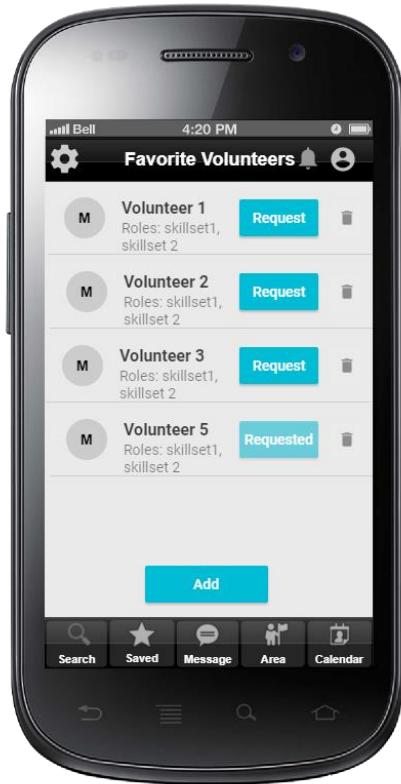
4.2.2.17 Add or Remove Volunteer Supplies

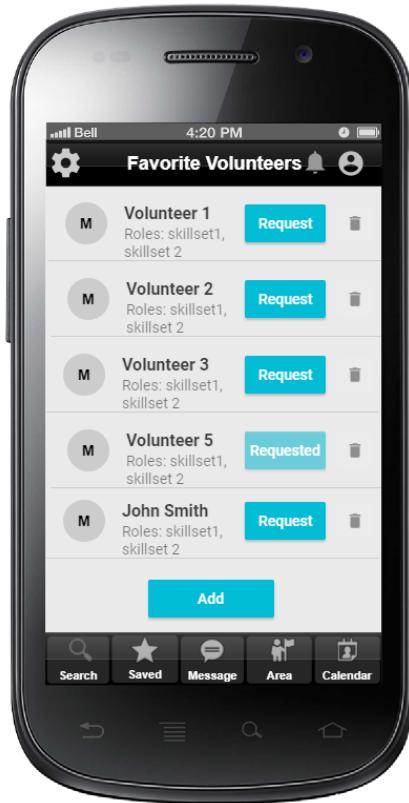




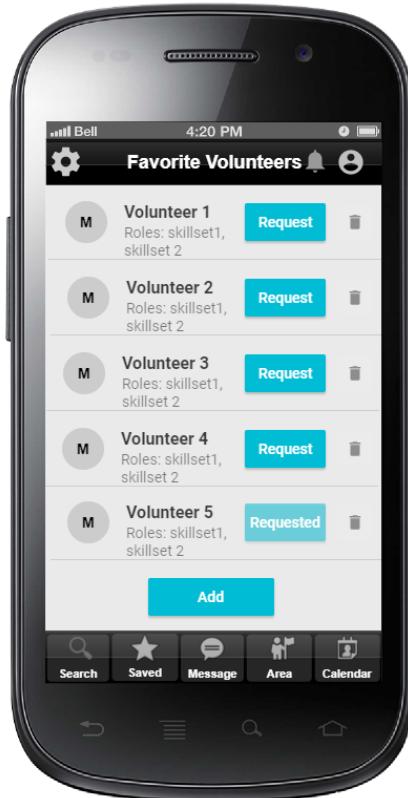
4.2.2.18 Add or Remove to Favorite Volunteers



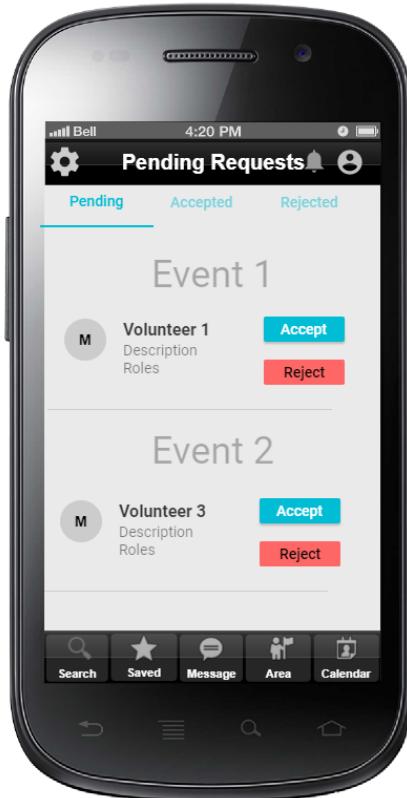




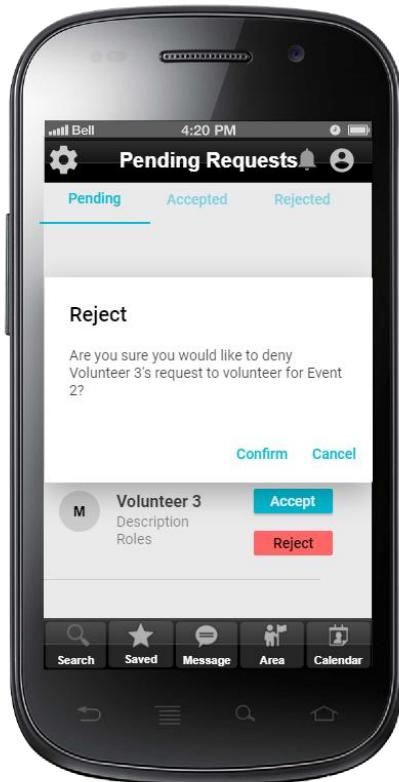
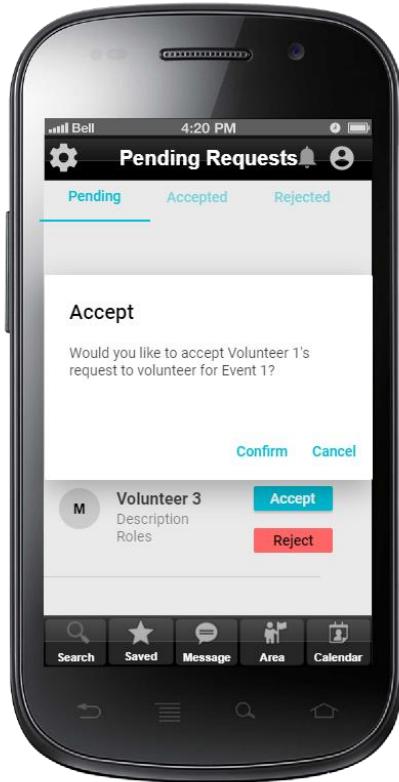
4.2.2.19 View Favorite Volunteers

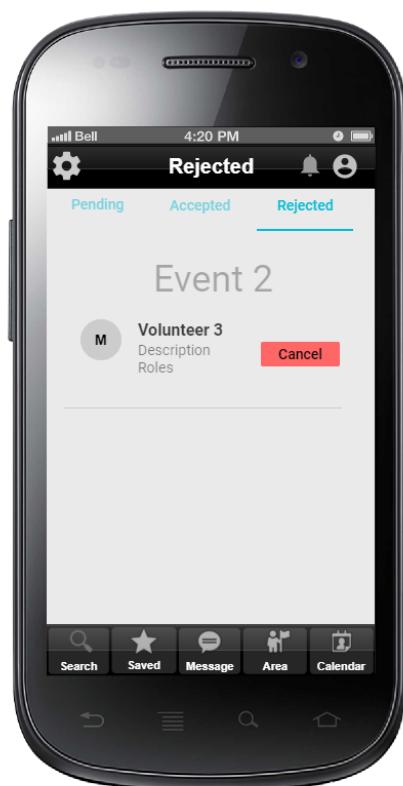
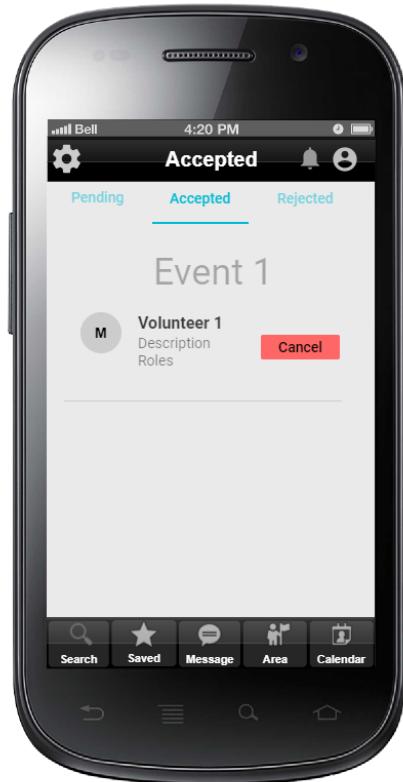


4.2.2.20 View Volunteer Requests

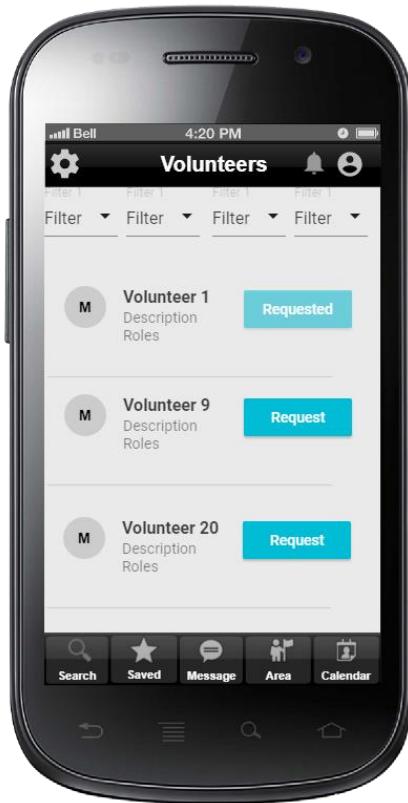


4.2.2.21 Accept or Reject Volunteer Requests

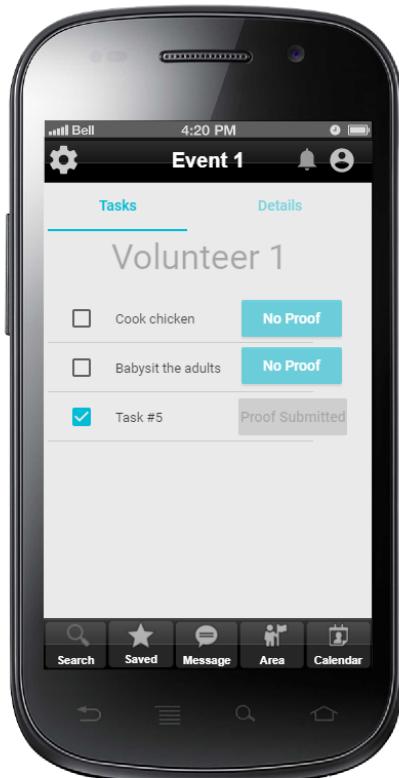
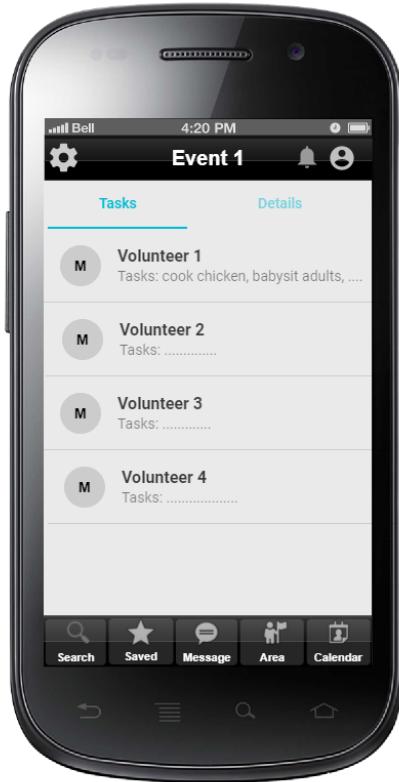




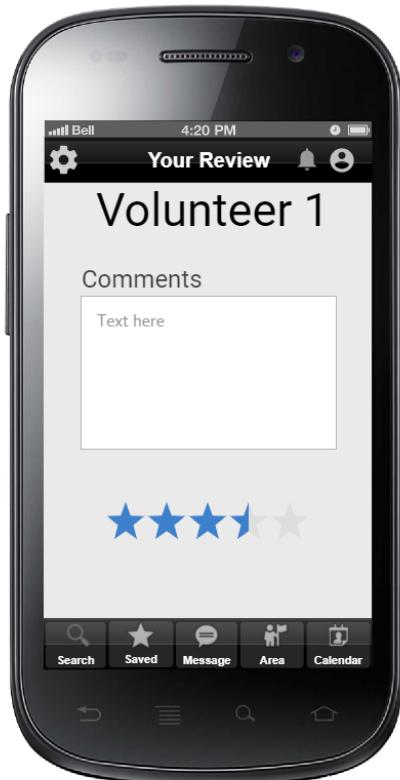
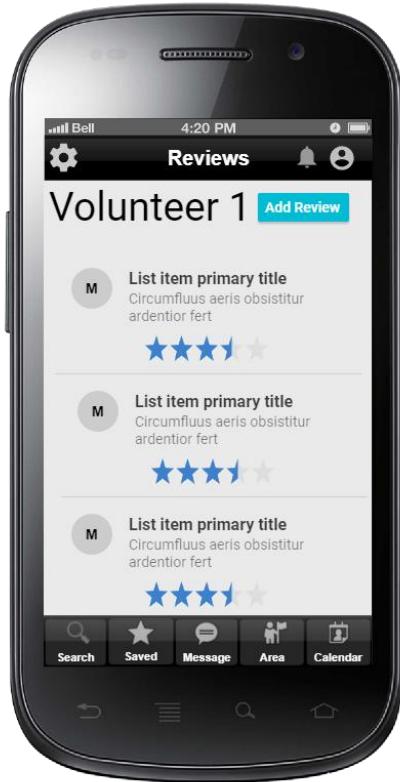
4.2.2.22 View Volunteers



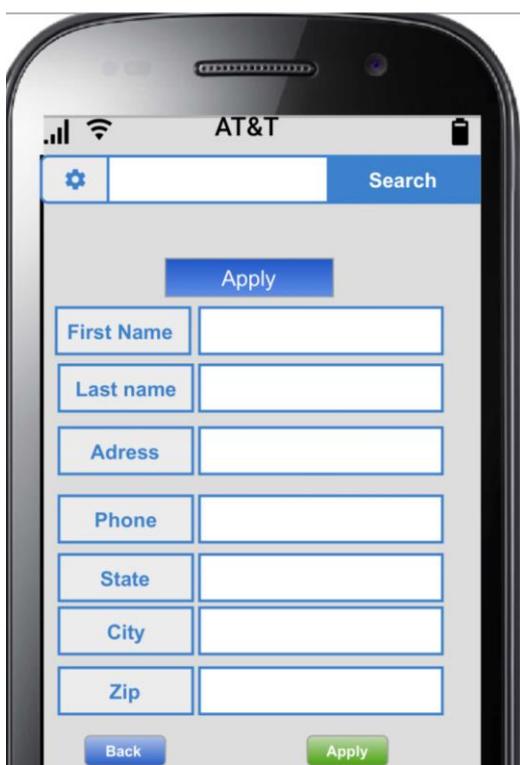
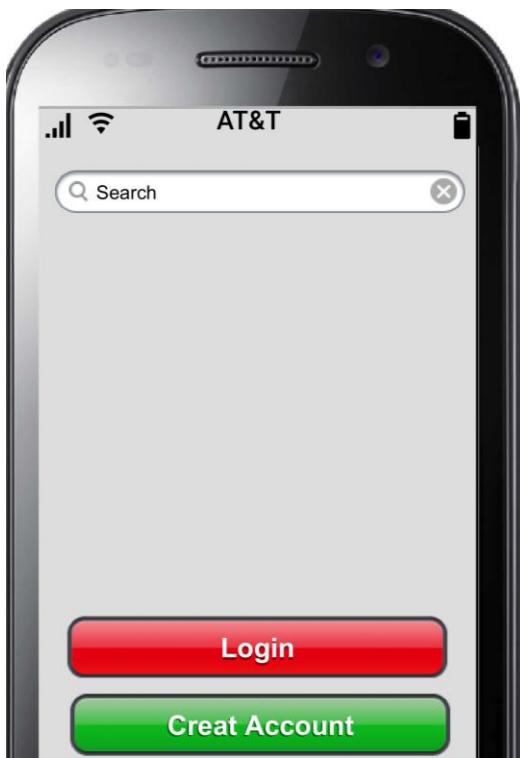
4.2.2.23 View Volunteer Tasks



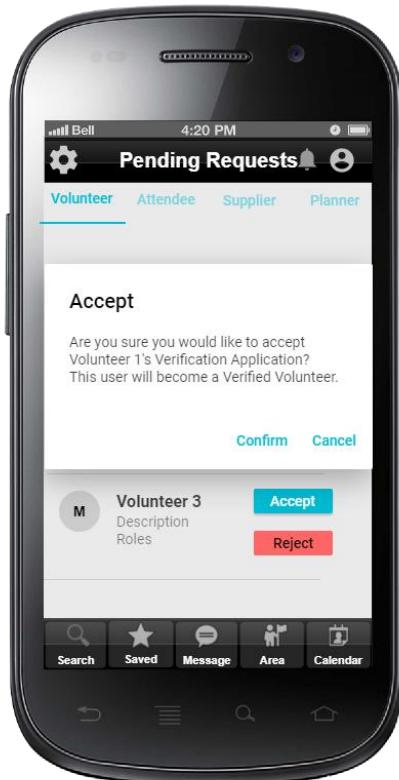
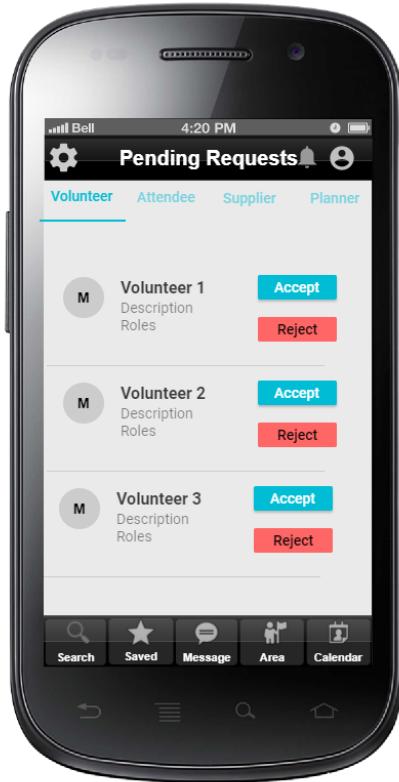
4.2.2.24 View Specific Volunteer Ratings

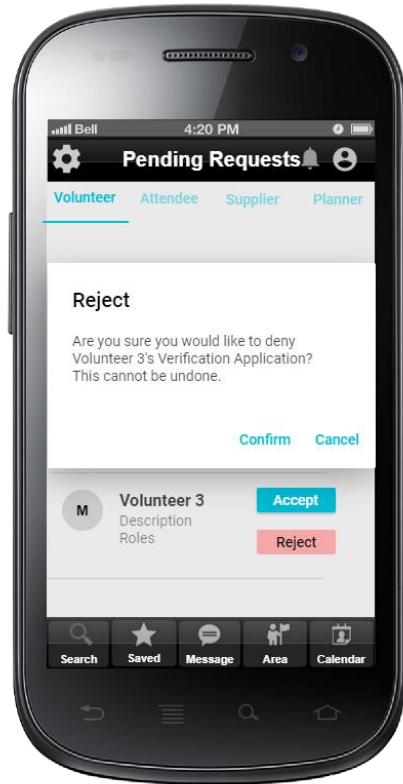


4.2.2.25 Volunteer Application

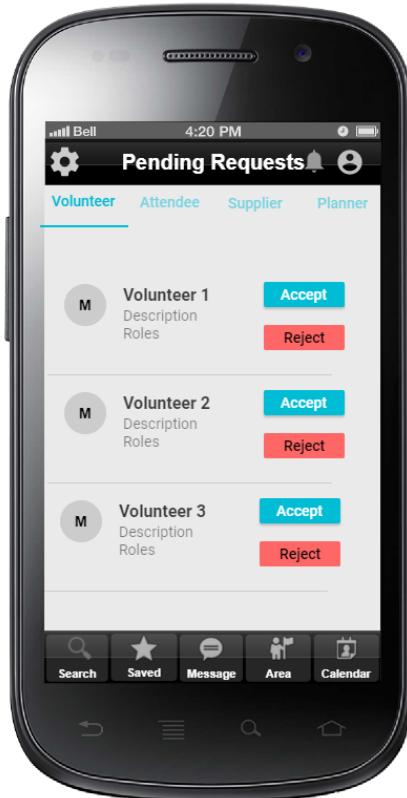


4.2.2.26 Accept or Deny Verification

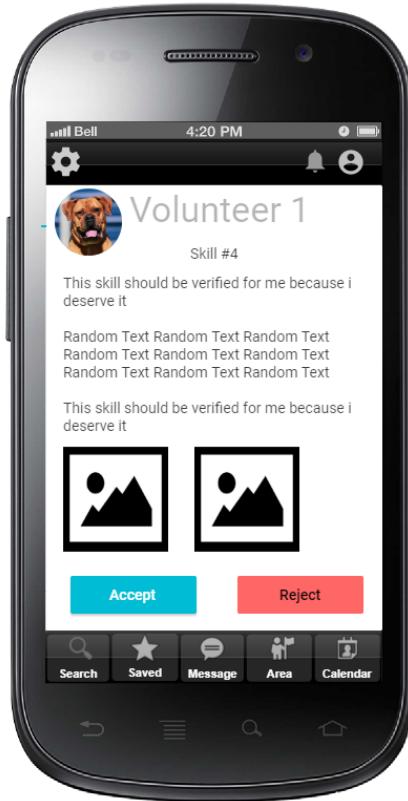




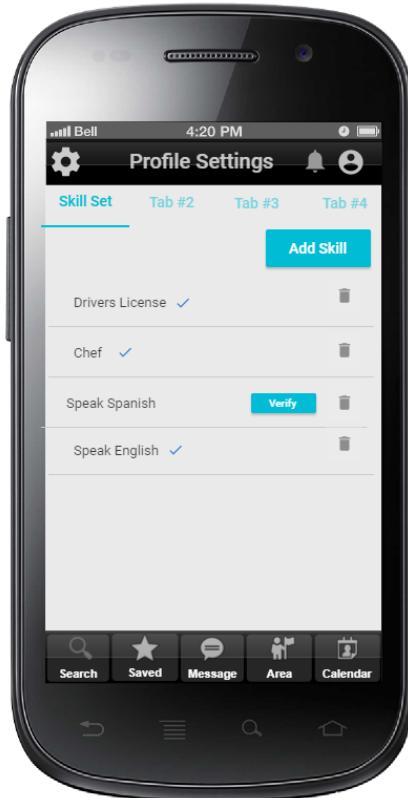
4.2.2.27 View Verification Applications



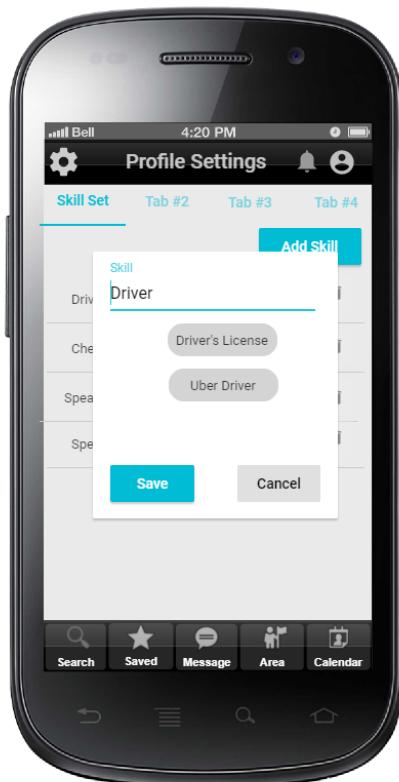
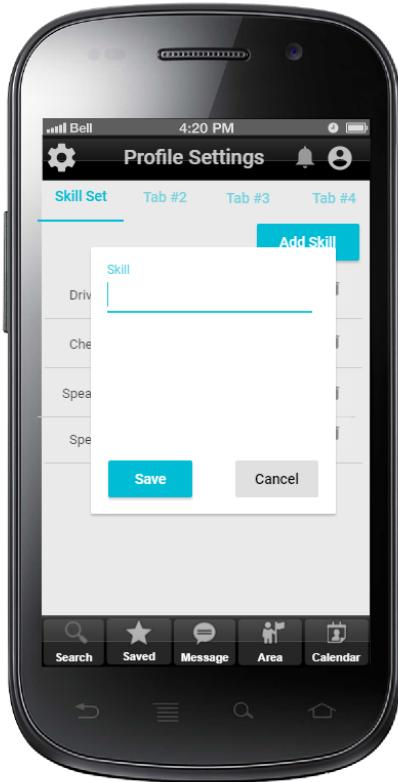
4.2.2.28 View Verification Application Specifics

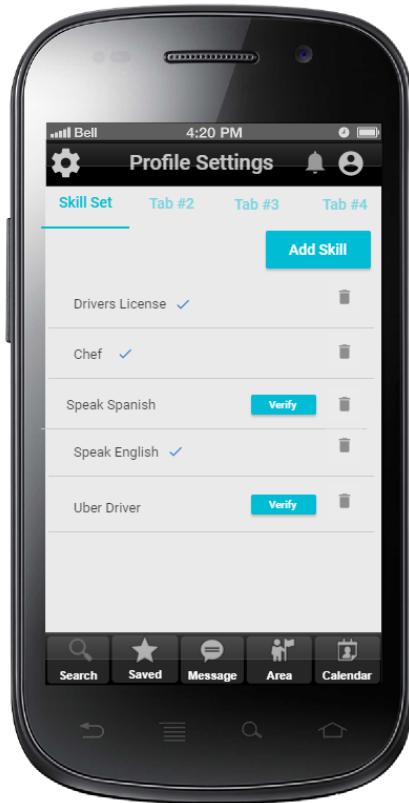


4.2.2.29 View Skill Set

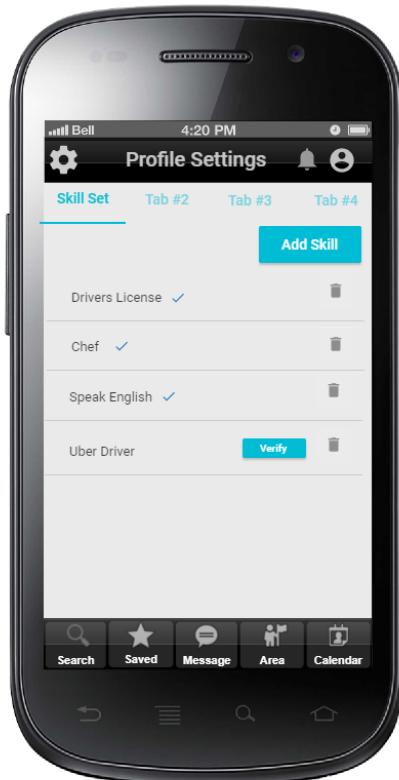
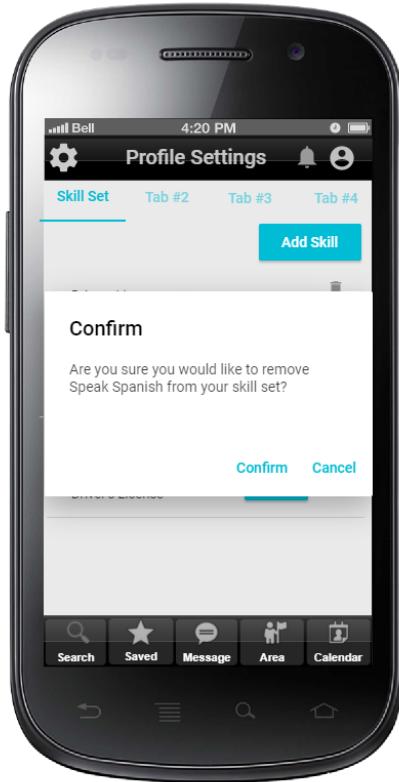


4.2.2.30 Add Skill to Skill Set





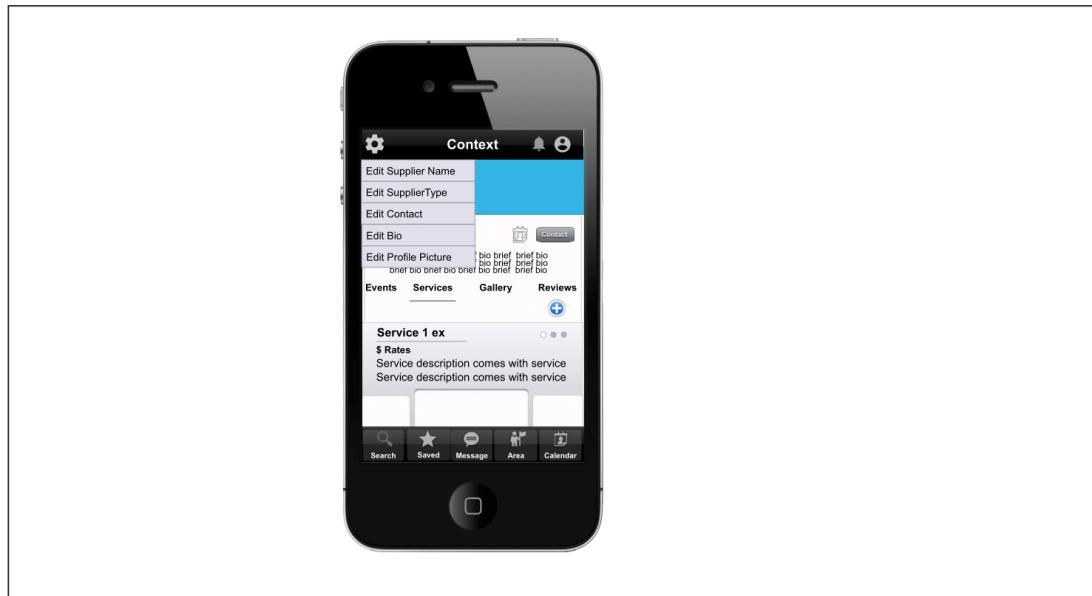
4.2.2.31 Remove Skill from Skill Set



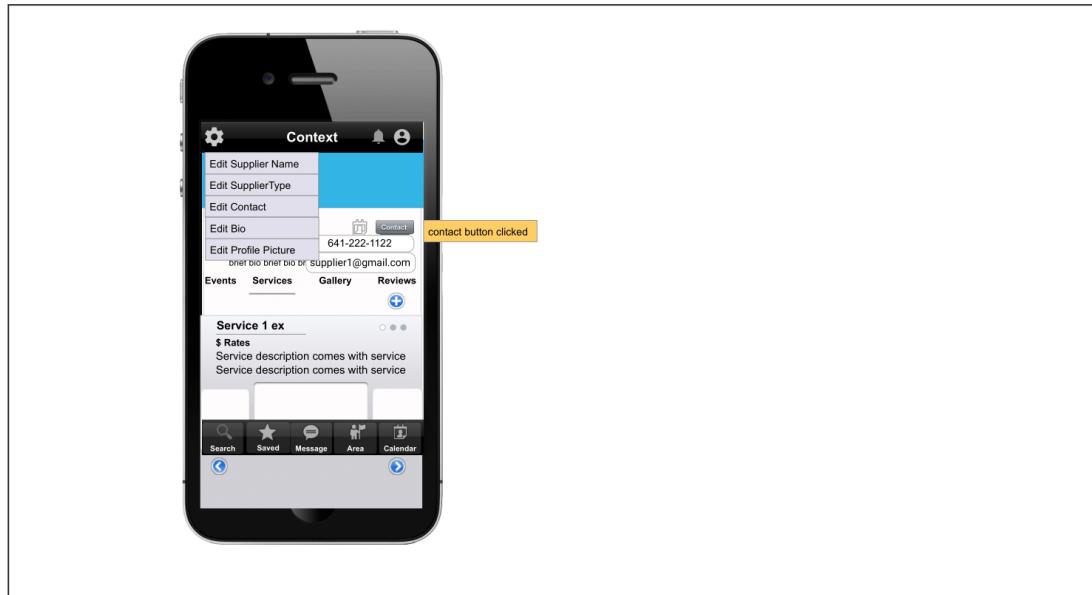
4.2.3 Supplier Mobile UI Drawings

4.2.3.1 View Supplier Listing (3042)

Supplier Bio



Supplier Display Contact

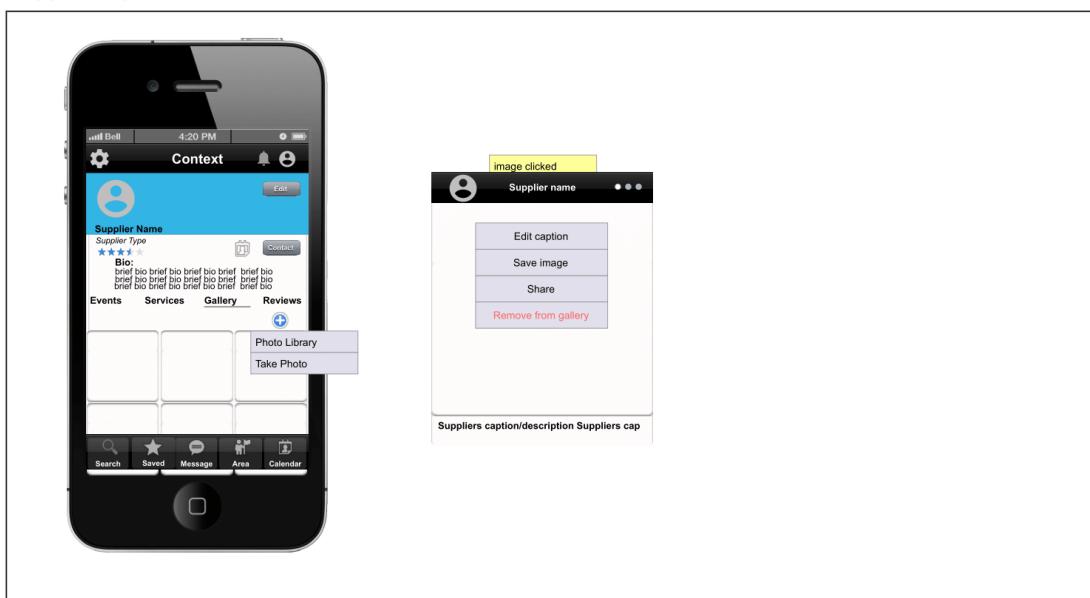


Supplier Profile

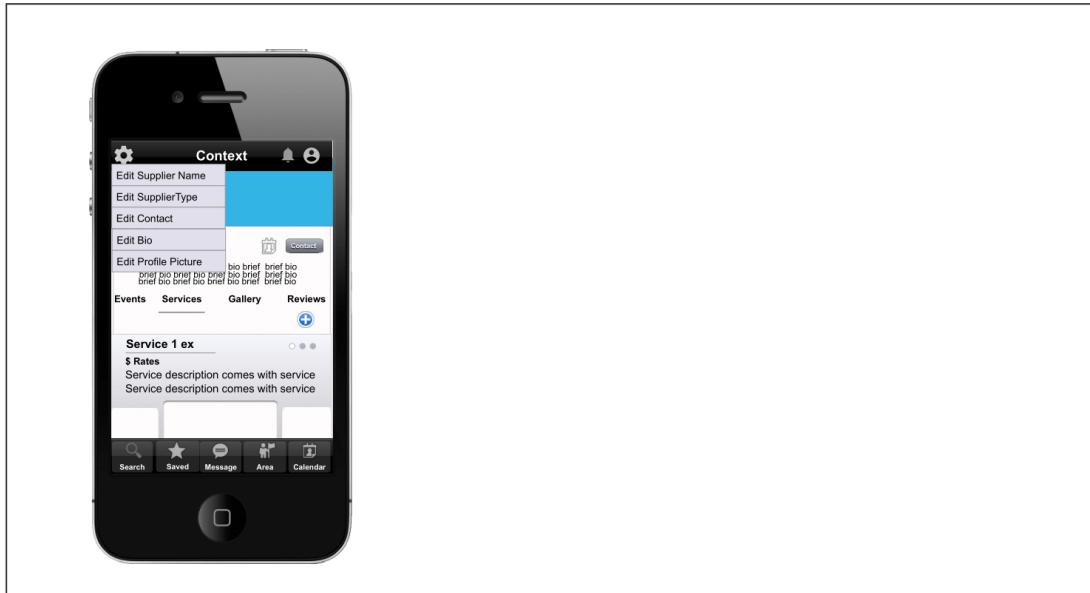


4.2.3.2 Add, Edit, and Delete Supplier Listing (3008, 3011, 3026)

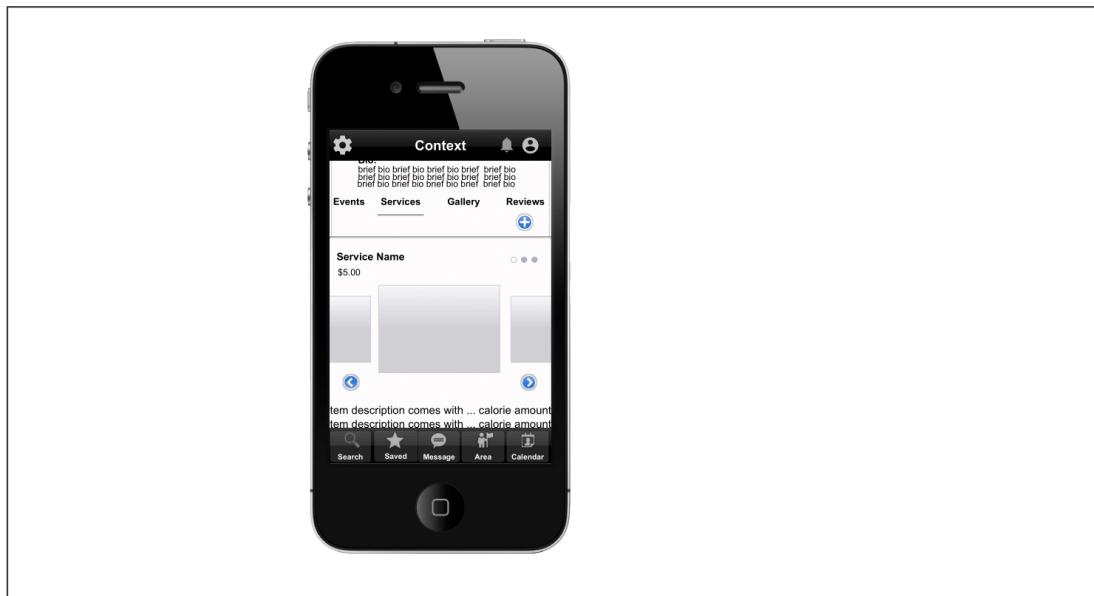
Supplier Upload Picture



Supplier Set Type



Supplier Display Pricing

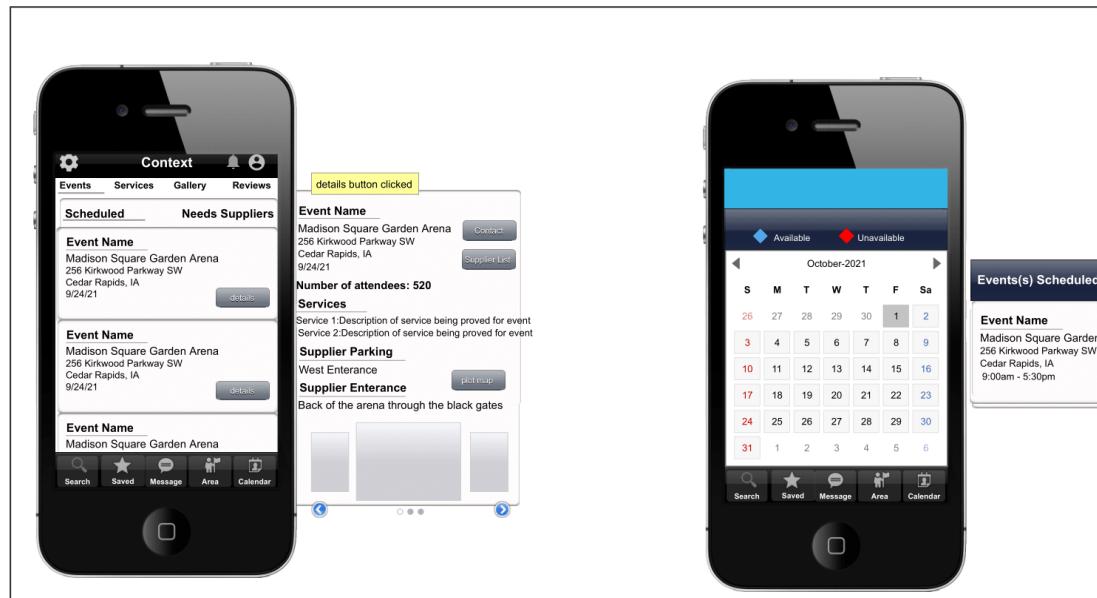


4.2.3.3 View Supplier Services (3044)

Supplier Profile Services

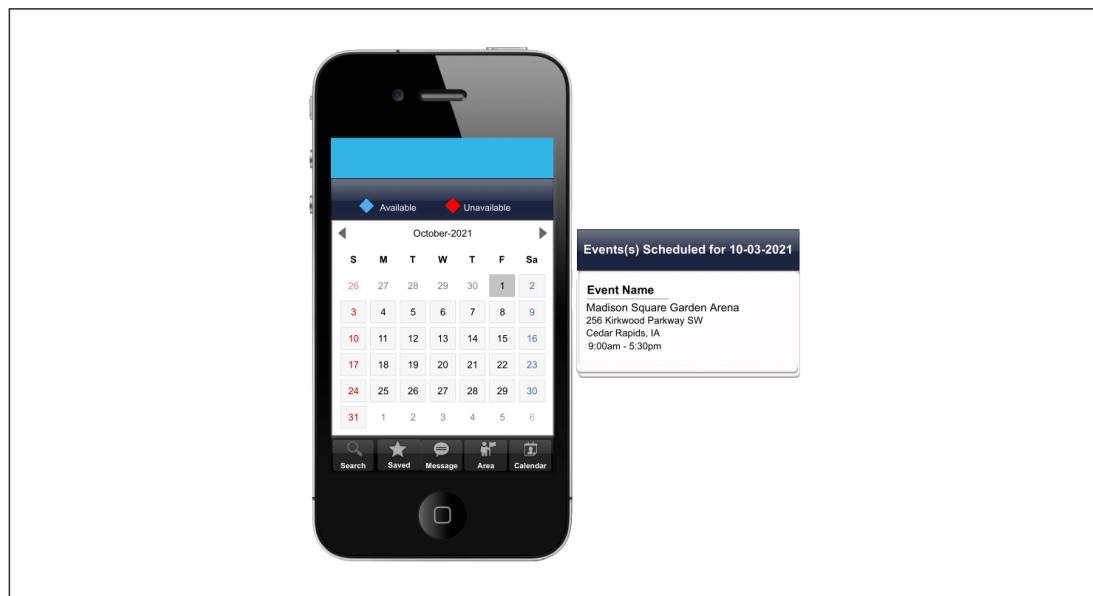


Supplier Availability

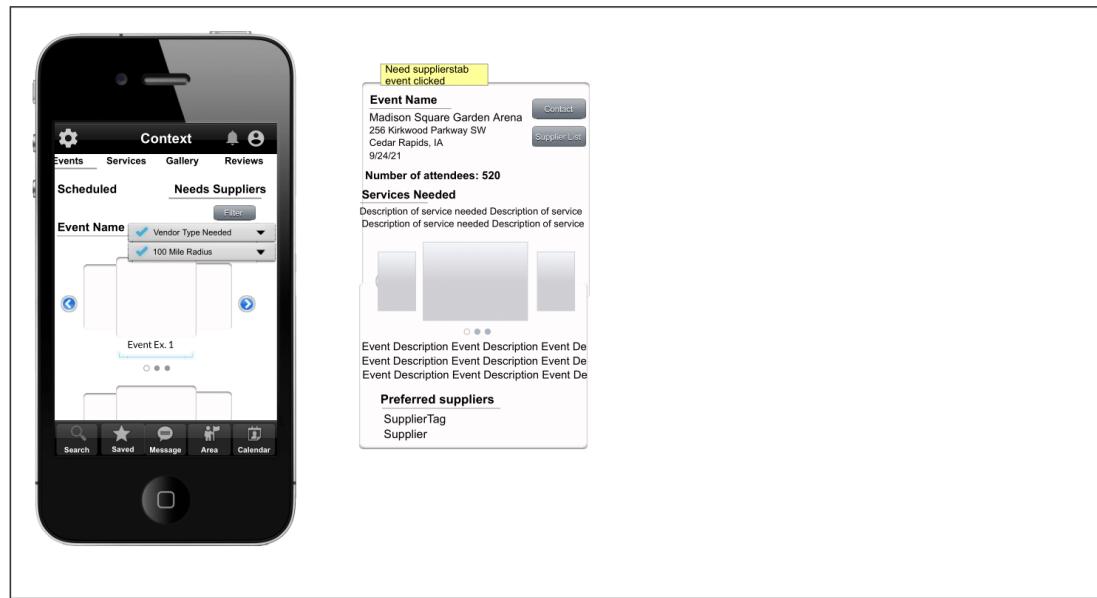


4.2.3.4 View Supplier Schedule (3043)

Supplier Page Availability Clicked

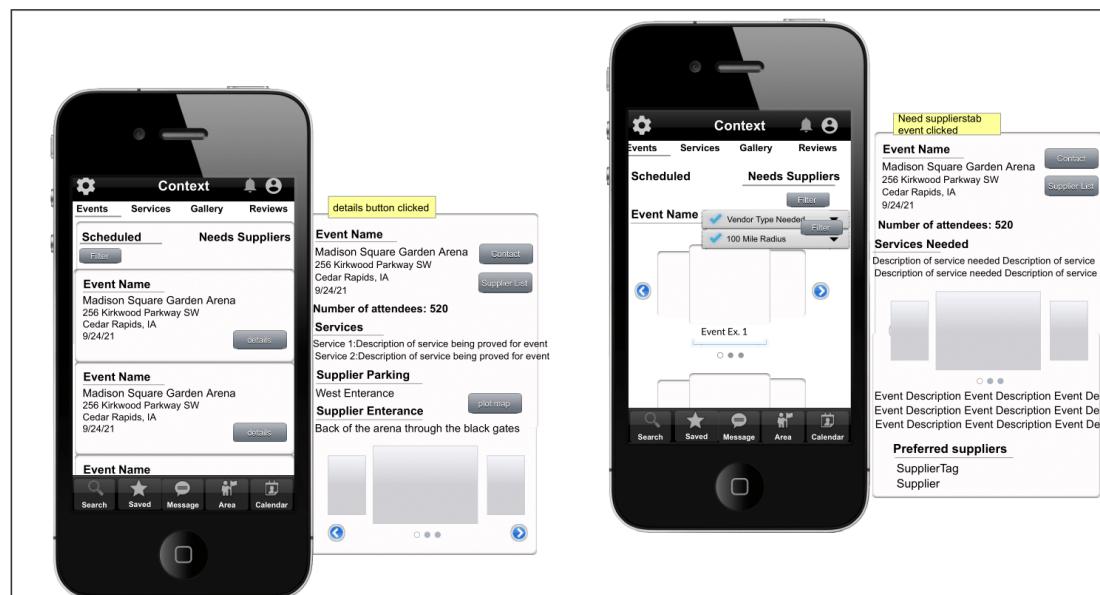


Supplier Events Need My Type



4.2.3.5 Supplier Find Personalized Events (3028)

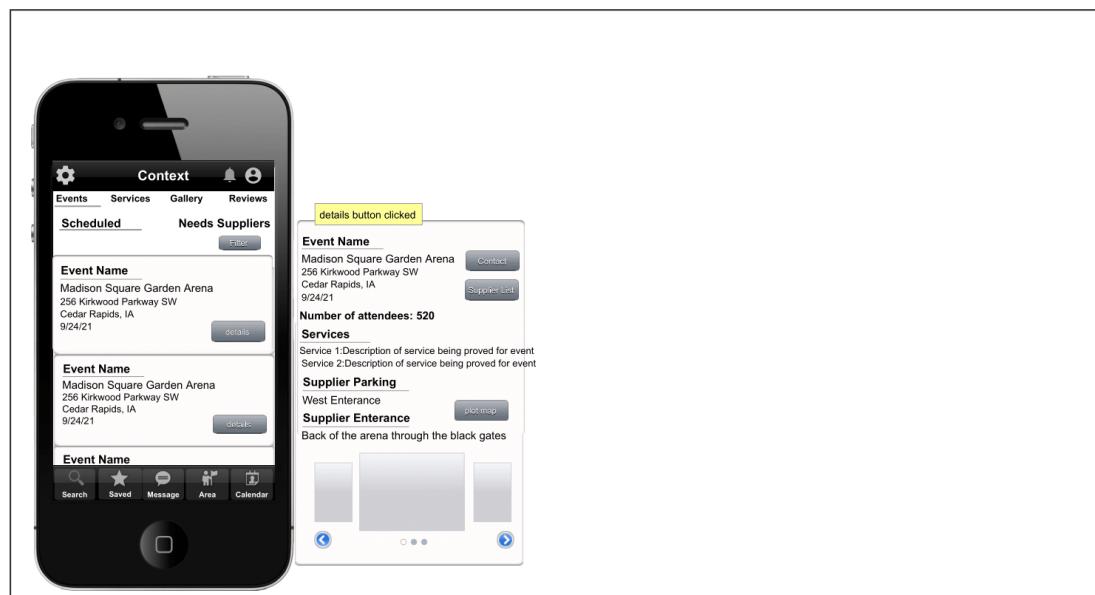
Supplier Browse Upcoming Events



Supplier Profile Events Clicked



Supplier View Attendee Count

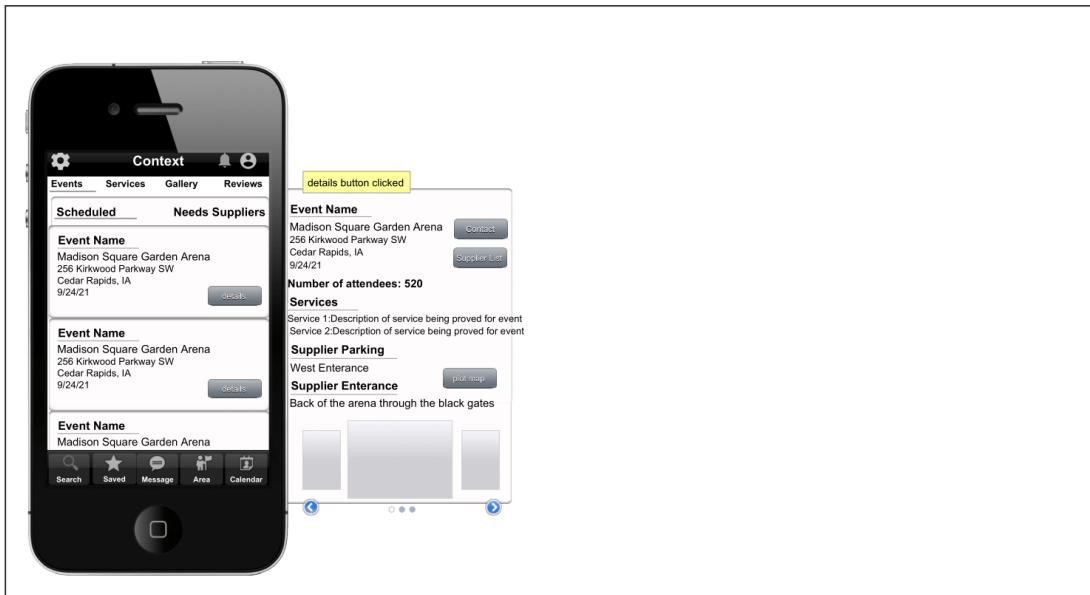


4.2.3.6 View Supplier Events (3030)

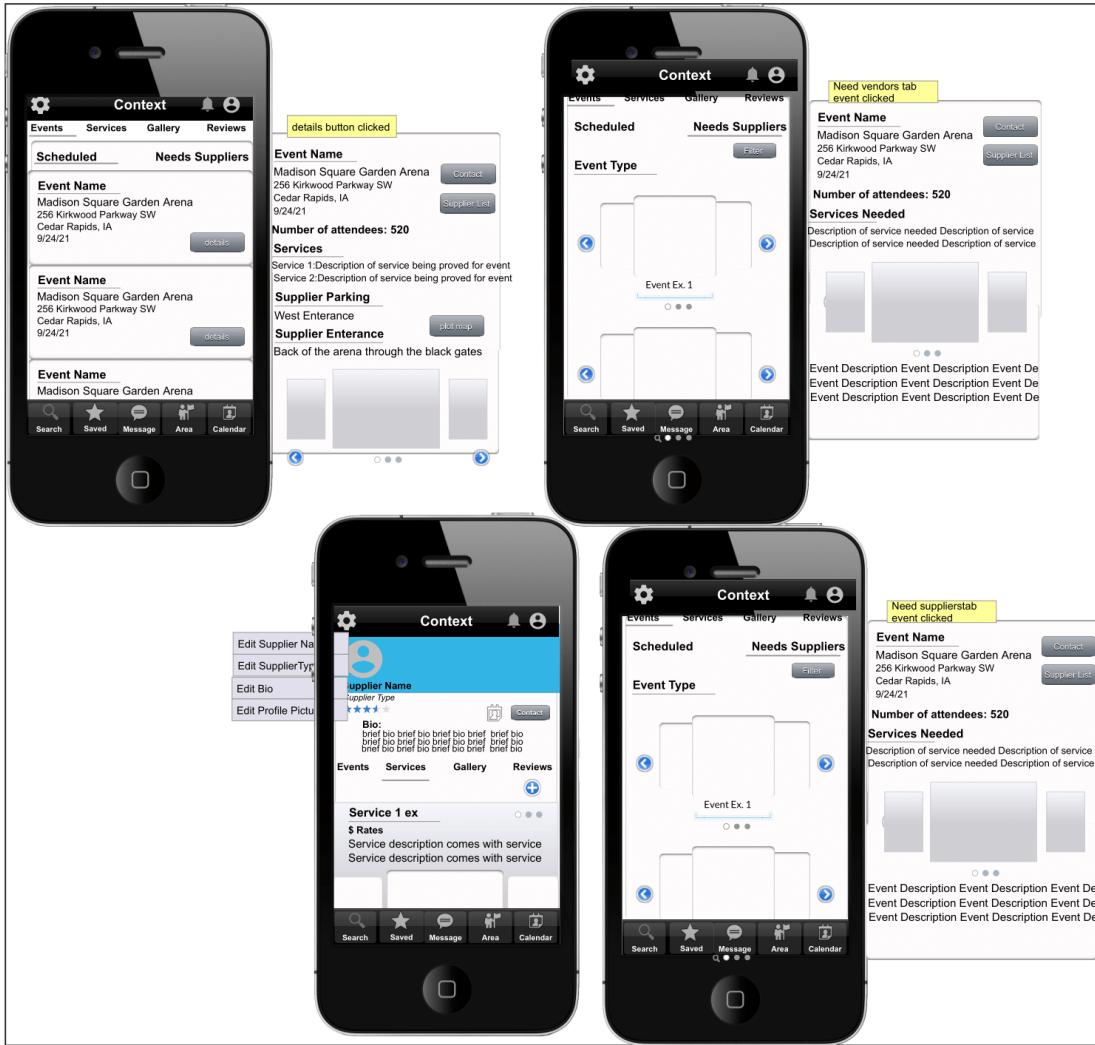
Supplier Contact Event Manager



Supplier Parking Entrance Map

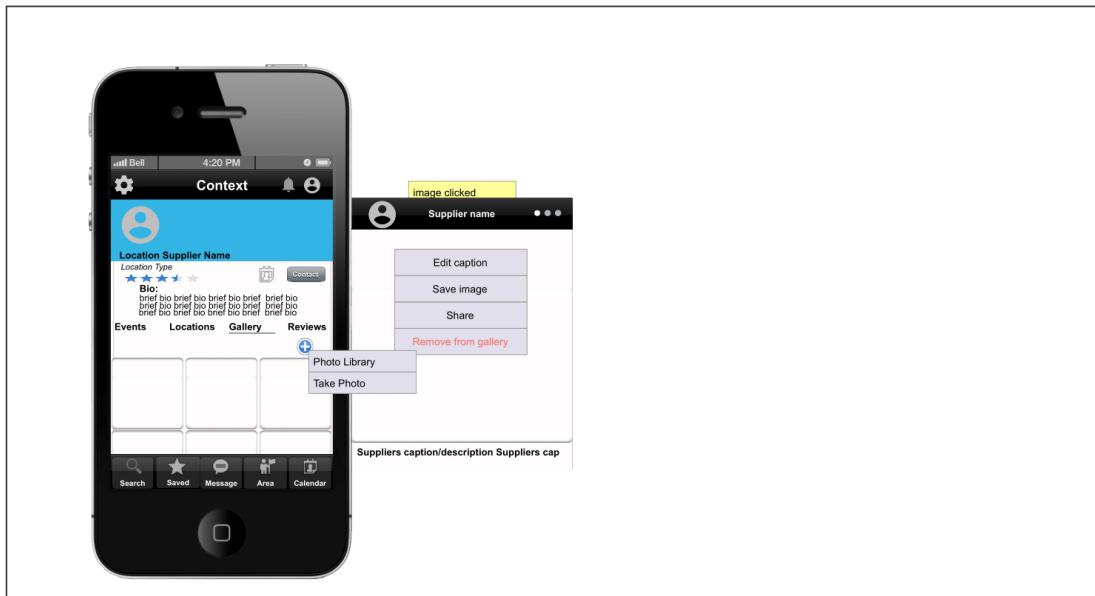


Event Services Template

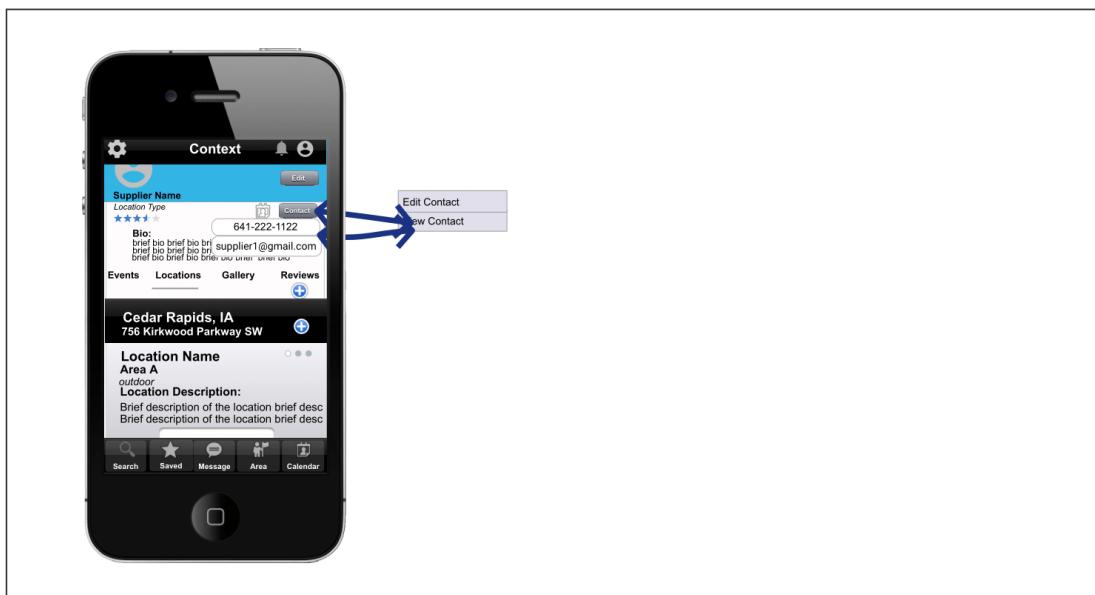


4.2.3.7

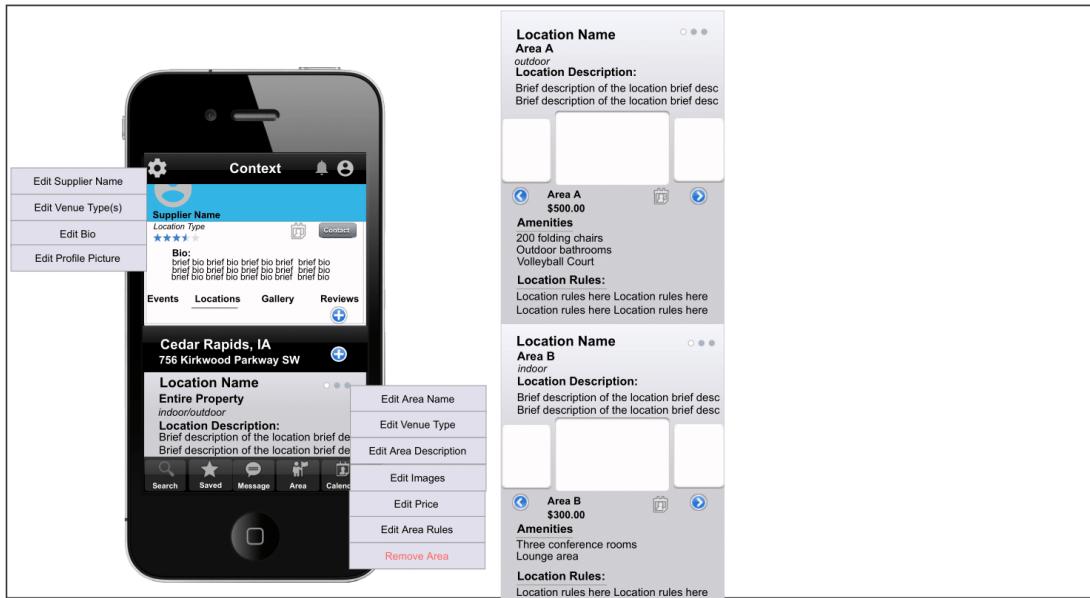
Location Supplier Display Pictures



Location Supplier Display Contact



LocationSupplier Set Location Types

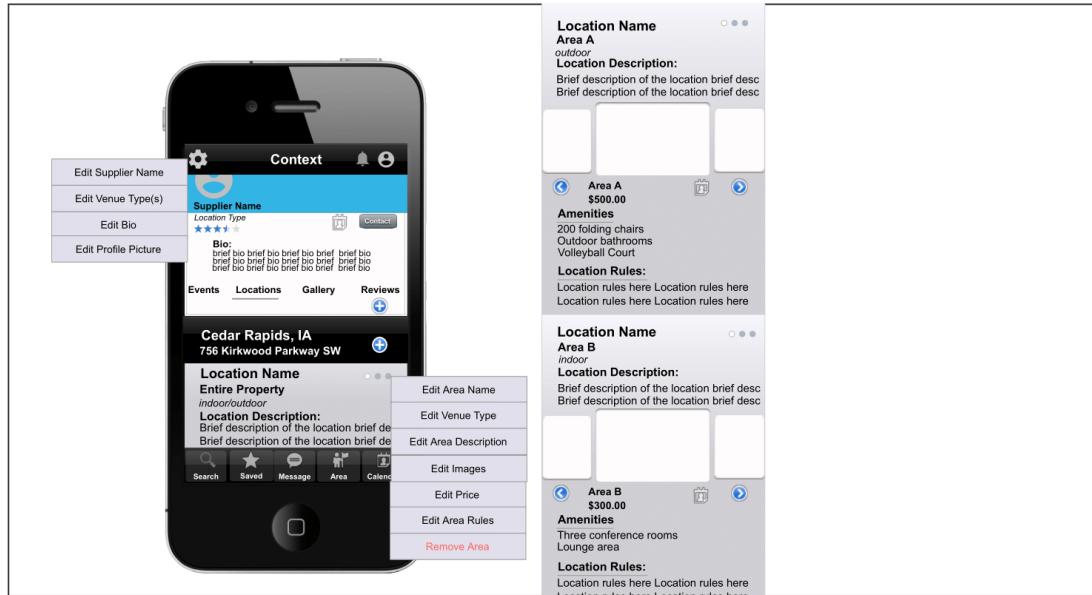


Location Supplier Location Record



View Sublocations and Supplies (3037, 3041)

LocationSupplier List Areas Prices And Supplies

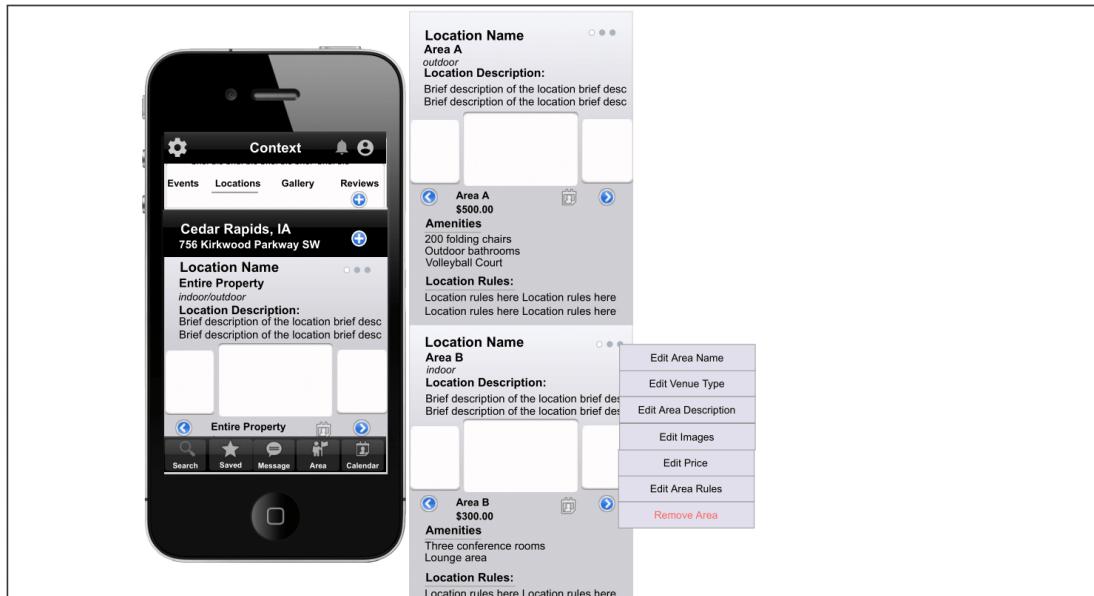


4.2.3.8 View Sublocation Schedule (3035)

Location Supplier Page Post Schedule



Location Supplier List Rules



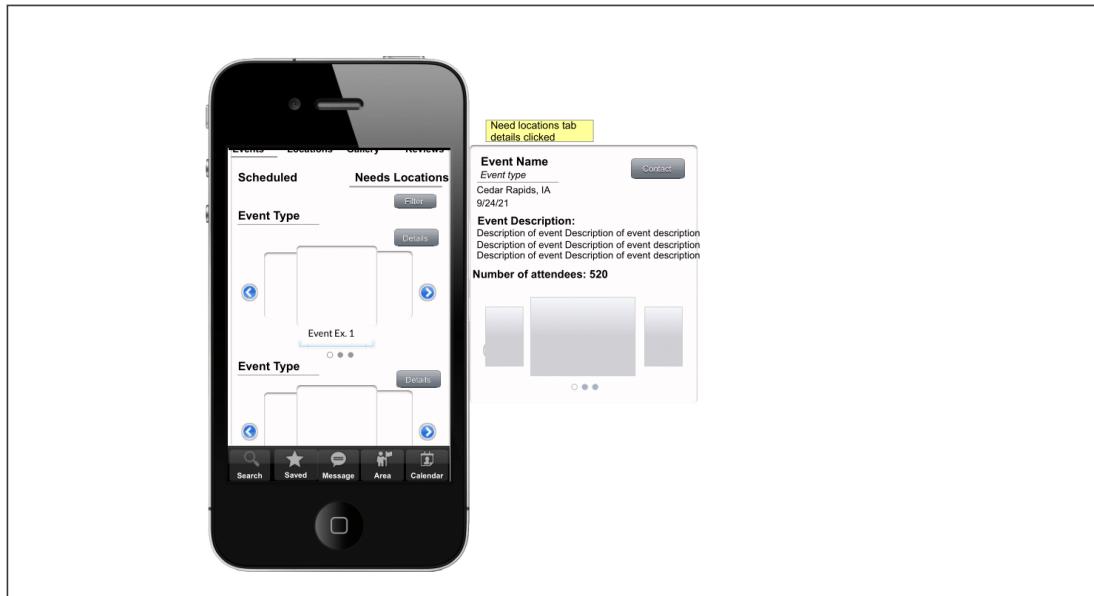
4.2.3.9 View Sublocation Rules (3039)

4.2.3.10 View Location Image Gallery (3032)

Gallery Reviews template



Location Supplier View Event By Type

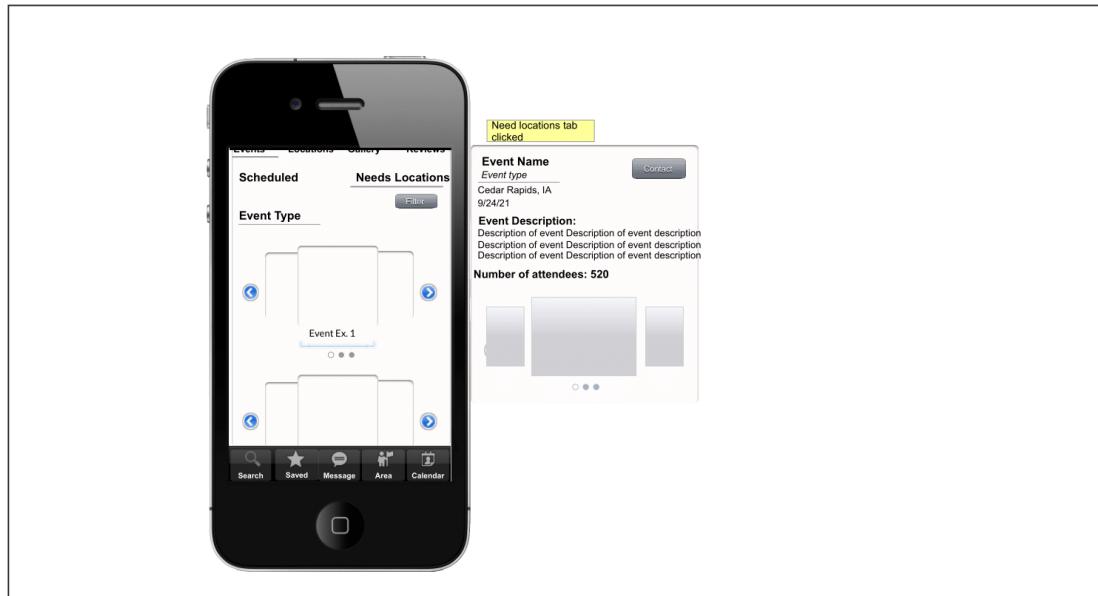


4.2.3.11 Location Find Personalized Events (3028)

Location Supplier View Events Need Location

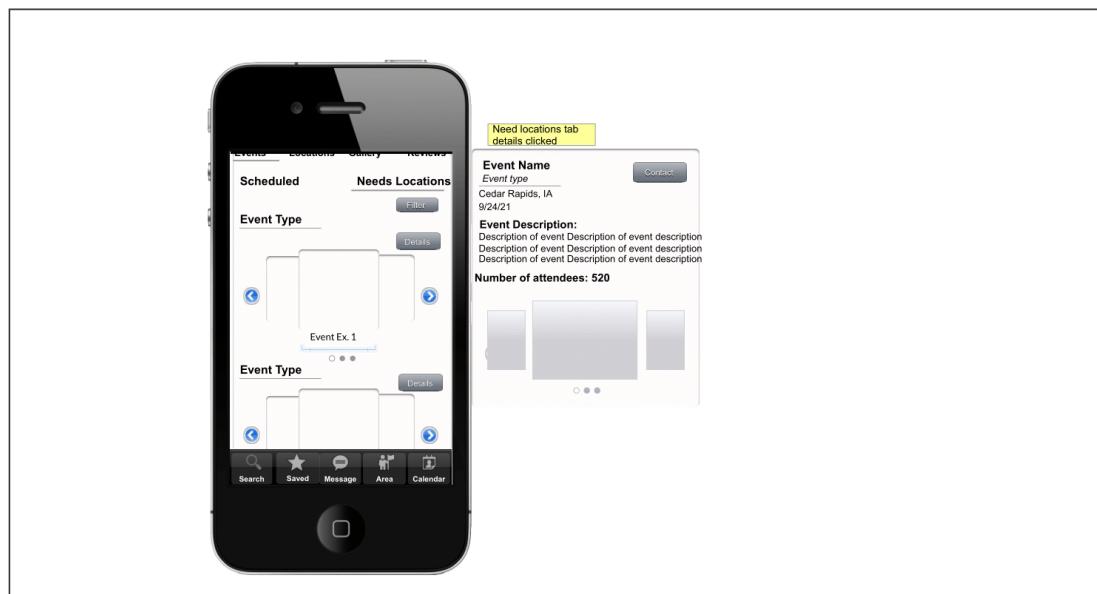


LocationSupplier Contact Event Planner

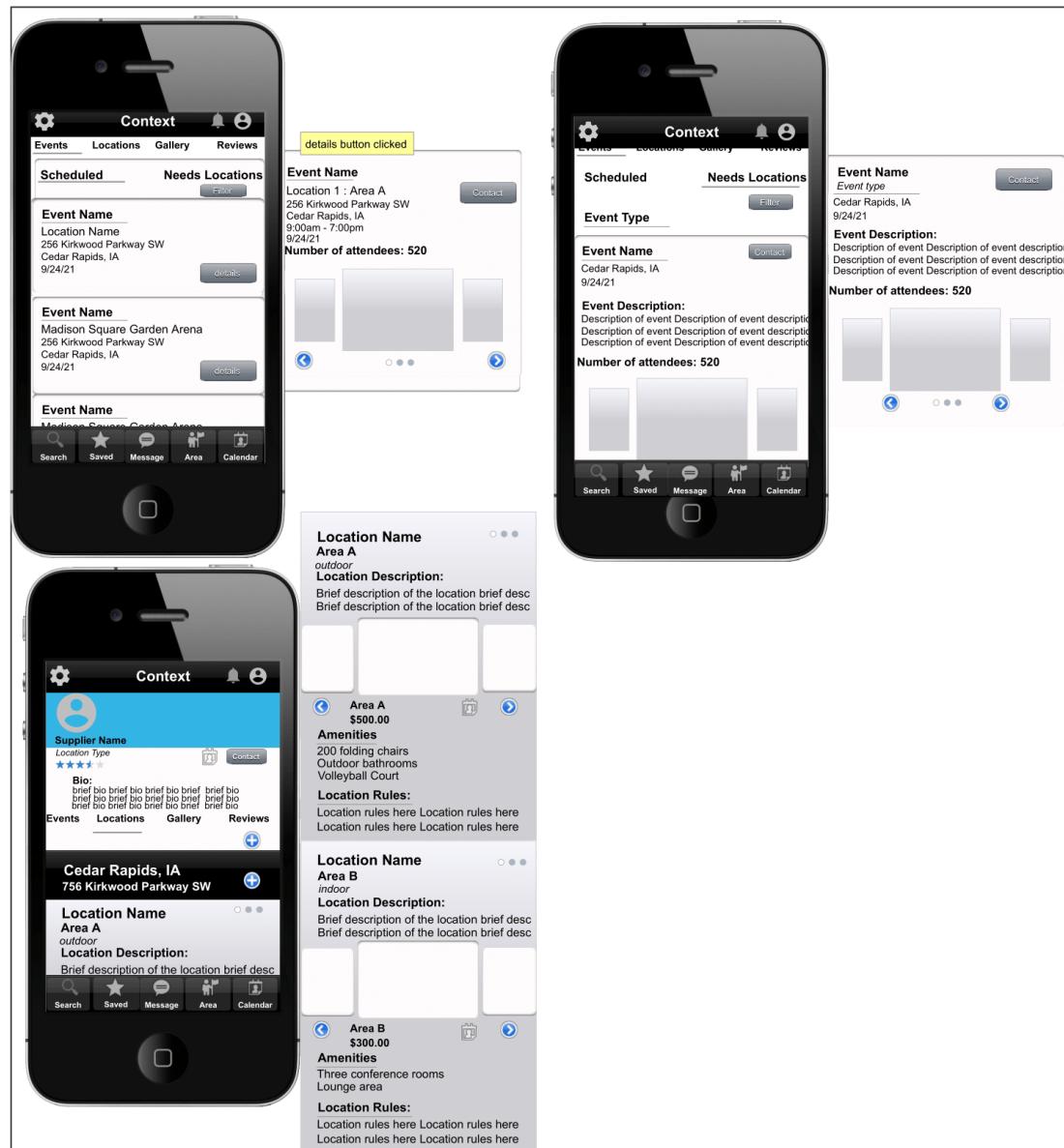


4.2.3.12 View Location Event (3030)

Location Supplier View Number Of Attendees

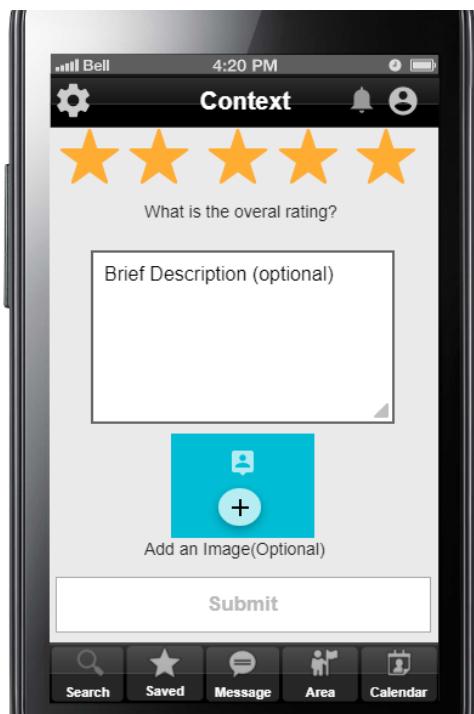


Events Locations Template

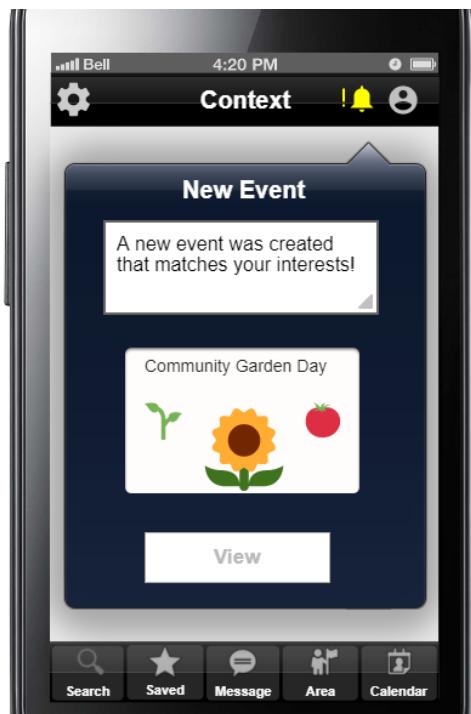


4.2.4 Attendee Mobile UI Drawings

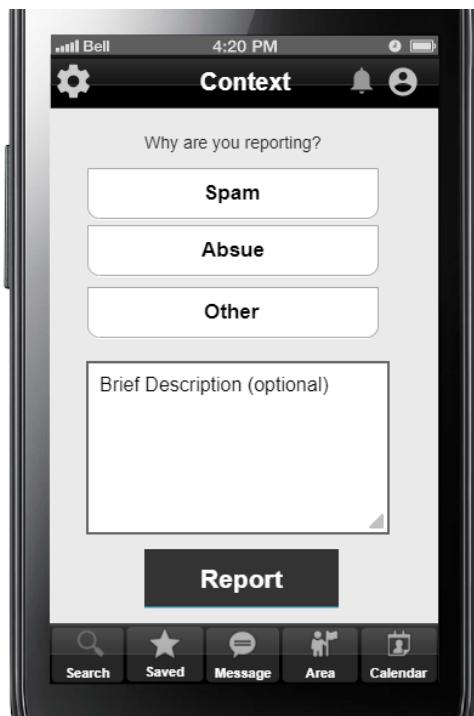
4.2.4.1 Review (4001, 4002, 4003, 4004)



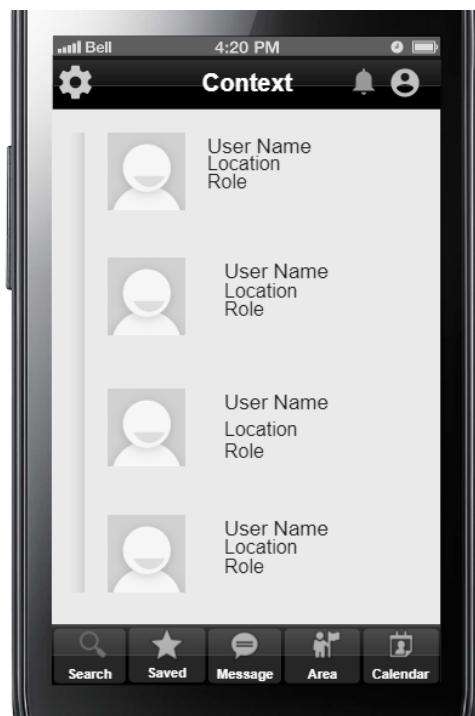
4.2.4.2 Receive New Notification(4005)



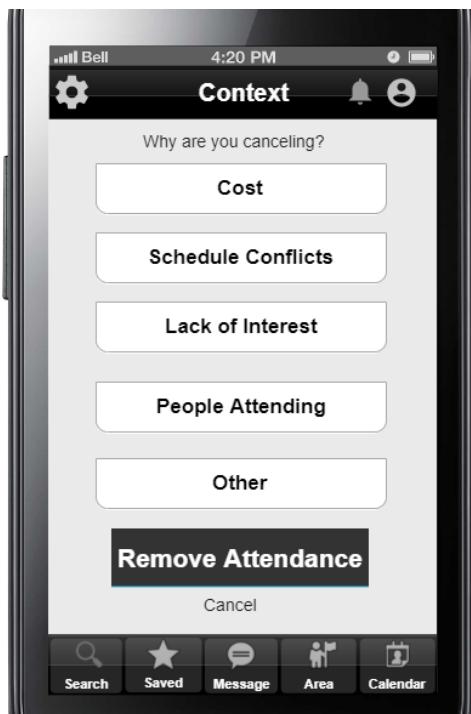
4.2.4.3 Report (4006, 4008)



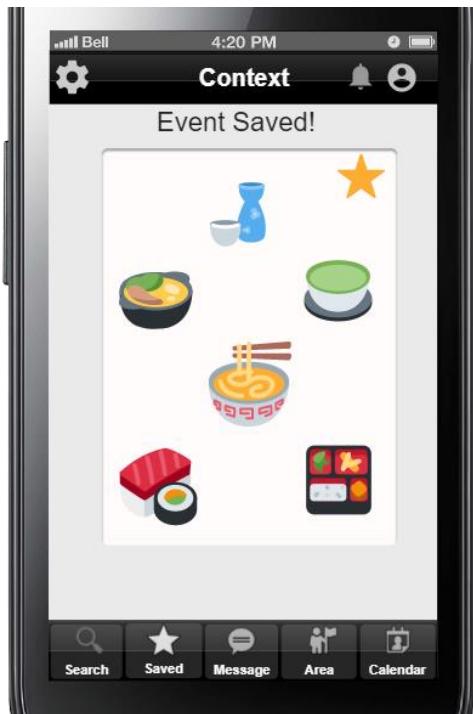
4.2.4.4 Search Users (4007)



4.2.4.5 Cancel Event Attendance (4009)



4.2.4.6 Bookmarked Event (4027)



4.2.4.7 View Attendance List (4011)



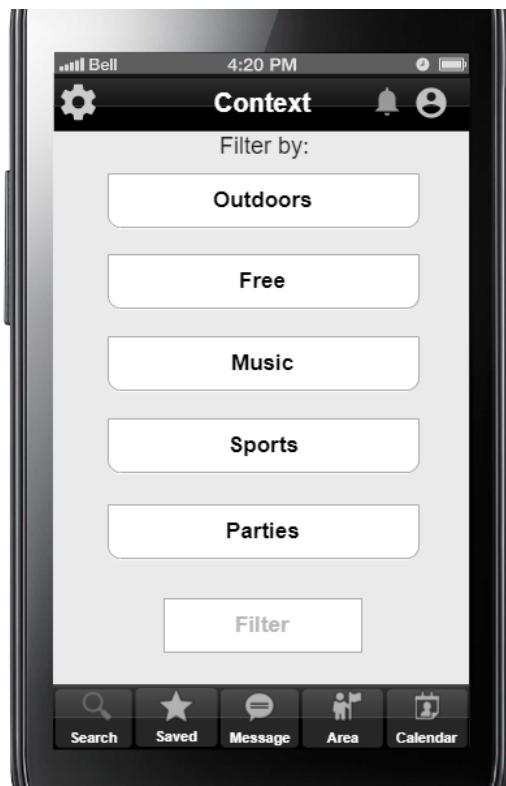
4.2.4.8 RSVP Event (4012)



4.2.4.9 Show Calendar (4013)



4.2.4.10 Filter Events (4013)



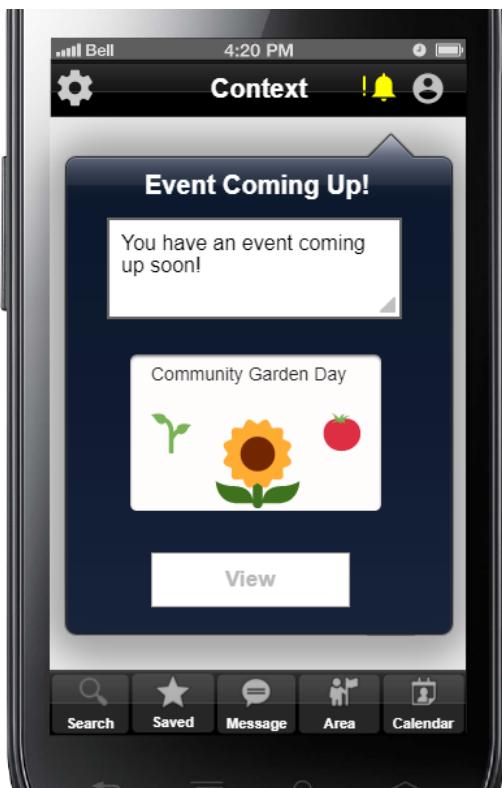
4.2.4.11 View Home screen With Filter (4013)



4.2.4.12 Edit Profile (4014)



4.2.4.13 Receive Event Reminder (4027)



4.2.4.14 Check In (4016)



4.2.4.15 Receive Message (4017)



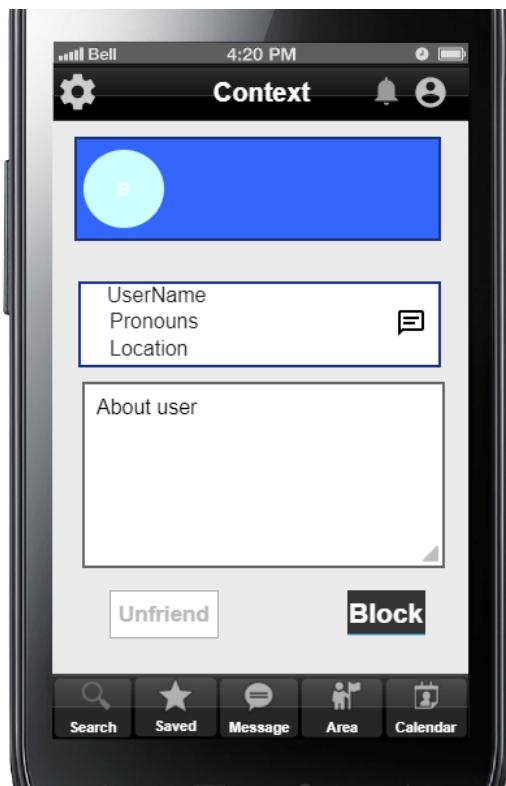
4.2.4.16 View Messages (4017)



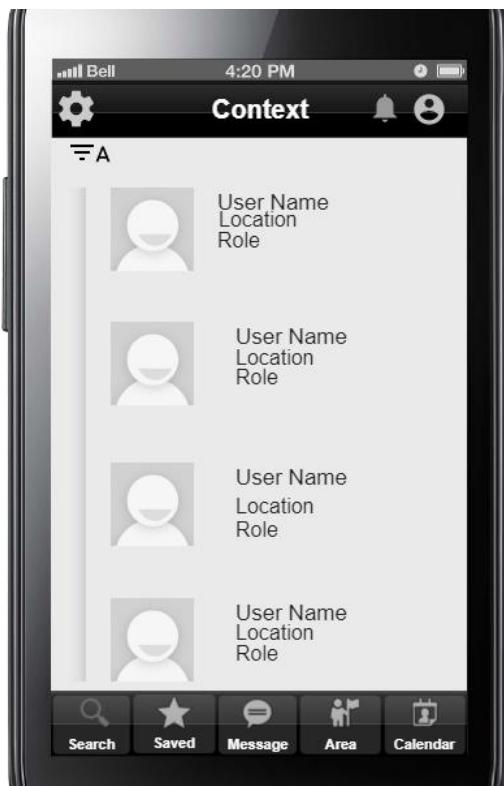
4.2.4.17 Login (4018)



4.2.4.18 Remove Friend (4019)



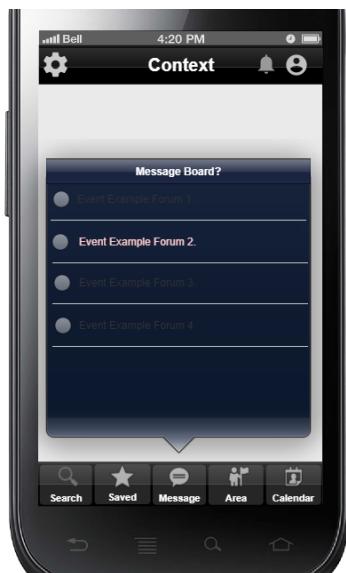
4.2.4.19 Sort Friends (4020)



4.2.4.20 Add Friend (4021)



4.2.4.21 View Message Boards (4023)



4.2.4.22 Register (4027)

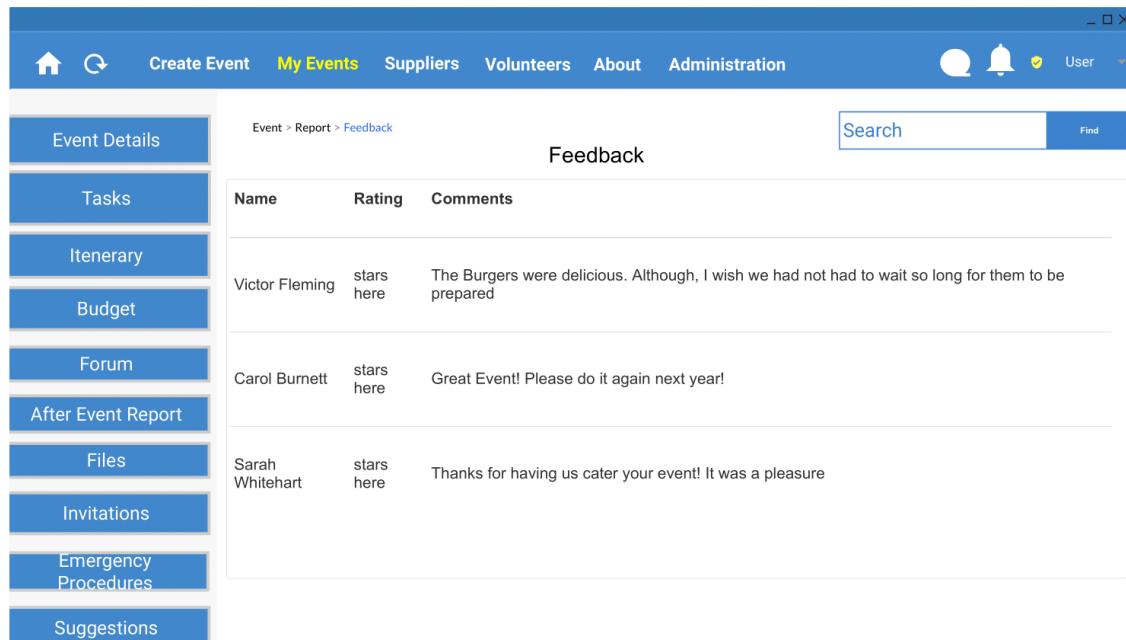


4.3 Desktop

4.3.1 Event

4.3.1.1 AAR Overview (1002,1039)

Feedback View



The screenshot shows the 'Feedback View' page of an event management system. The top navigation bar includes links for 'Create Event', 'My Events' (which is highlighted in yellow), 'Suppliers', 'Volunteers', 'About', 'Administration', and user account information. A search bar is located in the top right corner. The main content area is titled 'Feedback' and displays a table with three columns: 'Name', 'Rating', and 'Comments'. Three feedback entries are listed:

Name	Rating	Comments
Victor Fleming	stars here	The Burgers were delicious. Although, I wish we had not had to wait so long for them to be prepared
Carol Burnett	stars here	Great Event! Please do it again next year!
Sarah Whitehart	stars here	Thanks for having us cater your event! It was a pleasure

The left sidebar contains a vertical list of links: Event Details, Tasks, Itinerary, Budget, Forum, After Event Report, Files, Invitations, Emergency Procedures, and Suggestions.

Overview

The screenshot shows a software interface for managing events. The top navigation bar includes icons for Home, Create Event, My Events (highlighted in yellow), Suppliers, Volunteers, About, and Administration. On the right, there are notifications for messages and events, and a User dropdown. The main content area is titled "After Event Report". On the left, a sidebar lists "Event Details", "Tasks", "Itinerary", "Budget", "Forum", "After Event Report" (selected), "Files", "Invitations", "Emergency Procedures", and "Suggestions". The main content area displays a table with three columns: "Attendees", "Volunteers", and "Suppliers-Vendors". The data shows two entries: "Jane Doe" and "John Smith" under Attendees, and "Art's BBQ" under Suppliers-Vendors. Below this is a summary table with columns "Total" and "Total" showing values 120, 10, and 2 respectively. At the bottom, it says "Average Review: ★★★★☆" with a "View Feedback" button and a checkbox for "include feedback in download". A "Download Report" button is also present.

Attendees	Volunteers	Suppliers-Vendors
Jane Doe	John Smith	Art's BBQ
John Doe	John Doe	Jane's Ice Cream

Total	Total	Total
120	10	2

Average Review: ★★★★☆ [View Feedback](#) include feedback in download

[Download Report](#)

4.3.1.2 Activity Results(1029,1030)

Activity Results View

The screenshot shows the Tadpole software interface for managing event details. The left sidebar contains a vertical list of tasks: Event Details, Tasks, Itinerary, Budget, Advertising, After Event Report, Files, and Invitations. The main content area is titled "<Event Title> Add Results" and displays results for a "Sack Race". The results table is as follows:

Rank	Name
1st	Scooter McGee
2nd	TooHotTooHandle
3rd	Hustle Horse
Honorable Mention	One Legged Willy

Below the table, a note states: "Honorable Mention One Legged Willy".

Activity Results Add

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration Tess Data

Event Details

Tasks

Itenerary

Budget

Advertising

After Event Report

Files

Invitations

<Event Title> Add Results

Search Find

Select Activity From Itinerary

Sack Race
Darts
Capture the Flag
Closing Ceremony

Rank	Name
1st	Scooter McGee
2nd	Blue Team
3rd	
+ Add Another Row	

Add Picture

Add a picture? Yes

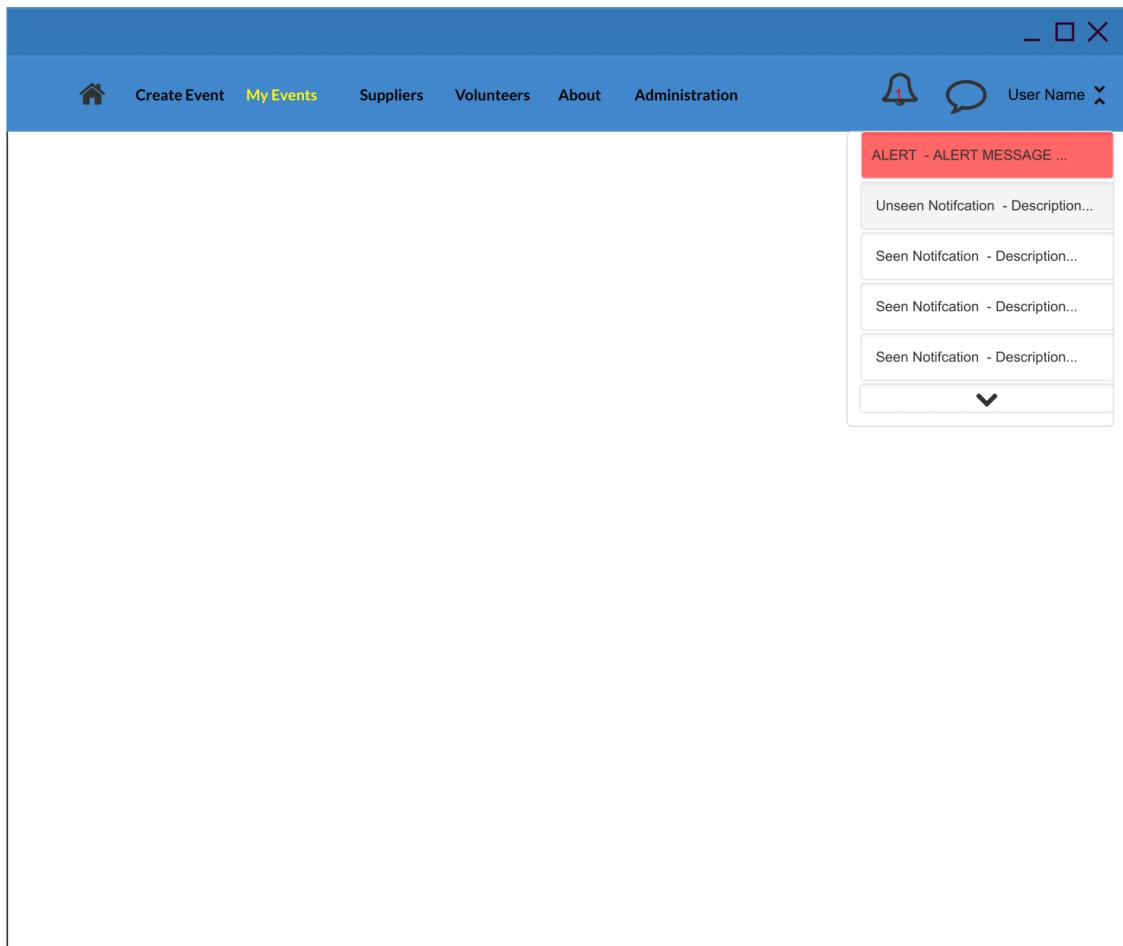
379 x 339

Upload Picture

Save Cancel

4.3.1.3 Alerts (1003)

Notification View



Alert Modal

The screenshot shows a web application interface with a blue header bar. The header includes a home icon, 'Create Event', 'My Events' (which is highlighted in yellow), 'Suppliers', 'Volunteers', 'About', 'Administration', a bell icon, a speech bubble icon, a 'User Name' field with a 'X' button, and a search bar with a magnifying glass icon and the word 'Search'.

The main content area on the left has a sidebar with the following sections:

- Group Title**
- Message Sub Cat**
 - # Chat Group Name
 - # Chat Group Name
 - # Chat Group Name
 - # Chat Group Name
- Message Sub Cat**
 - # Chat Group Name
 - # Chat Group Name
 - # Chat Group Name
 - # Chat Group Name
- Indiv Messages**
 - #Inbox

A modal dialog box is open in the center. It has a title bar 'Alert?' and a message body containing the text: 'This will send an alert message to recipients in this group. Are you sure?'. Below the message are two buttons: 'Send' and 'Cancel'. To the right of the modal, there is a sidebar with a list of names: 'mundi est onderibus acius tum contraria vivino'.

Message View

Group Title

Message Sub Cat

- # Chat Group Name

Message Sub Cat

- # Chat Group Name

Indiv Messages

#Inbox

Name
Short message.

Name
Colebat deus quin cesserunt aliud terram totidem plague quanto piscibus pontus deus mundi est aquae spisso caecoque terrae fuerat aethere ita undae igni mutatis proxima liquidas ponderibus terrenae legebantur pro otia fluminaque inposuit levius nullaque nuper utque item capacius tum millitis dextra tumescere cum feras metusque pendebat et terrenae praebebat capacius contraria montibus persidaque undae secuit fulminibus otia habentia induit deus distinxit ignea divino locum.

Name
Short message.

+ Alert message

4.3.1.4 Approval(1006, 1037, 1047)

1047_Approval Request

The screenshot shows the Tadpole software interface for managing event details. The left sidebar lists various sections: Event Details, Tasks, Itinerary, Budget, Advertising, After Event Report, Files, and Invitations. The main content area is titled "Suggestions" and displays a table of event details. The table columns are: Event Detail, Due Date, Content, Comment, Status, and Feedback. The rows show the following data:

Event Detail	Due Date	Content	Comment	Status	Feedback
Title	07/10/2016	Description	Can we rename it?	Requested	<input type="button" value="Edit"/>
Date	07/10/2016	08/01/2016	Does this date work?	Approved	<input type="button" value="Edit"/> That would be a good day.
Location	07/10/2016	Name, Address, City, State, Zip		Requested	<input type="button" value="Edit"/>

Below the table, there is a "Create Request" section with a "Category" dropdown set to "Event Details". It includes fields for "Event Detail", "Due Date" (07/10/2016), "Content", and "Comment". There are also "Title" and "Title" input fields, a timestamp, and a "Create" button with a checkmark and a "Cancel" button with an X.

1006_Approval

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration

Search Find

Event Details

Tasks

Itenerary

Budget

Advertising

After Event Report

Files

Invitations

Create Request

Suggestions

Event Details

Event Detail	Due Date	Content	Comment	Status	Feedback
Title	07/10/2016	Description	Can we rename it?	Requested	<input type="button" value="Edit"/>
Date	07/10/2016	08/01/2016	Does this date work?	Approved	<input type="button" value="Edit"/> That would be a good day.
Location	07/10/2016	Name, Address, City, State, Zip		Requested	<input type="button" value="Edit"/>

Budget

Budget Category Total	Due Date	Total Budget	Total Actual	Comment	Status	Feedback
	07/10/2016	Total Budget	Total Actual	Is this okay?	Denied	<input type="button" value="Edit"/> It's not okay.

Vendors

Vendor Type	Due Date	Number	Comment	Status	Feedback
Category	07/10/2016	Total	How about getting some clowns?	Denied	<input type="button" value="Edit"/> Magicians instead.

1037_Suggestions View (Edit Comment)

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration   Tess Data

Create Request Suggestions Find

Event Details

Tasks

Itenerary

Budget

Advertising

After Event Report

Files

Invitations

Event Details

Event Detail	Due Date	Content	Comment	Status	Feedback
Title	07/10/2016	Description	Can we rename it?  Requested		
Date	07/10/2016	08/01/2016	Does it	Can we rename it?	It would be a good day.
Location	07/10/2016	Name, Address, City, State, Zip			

Budget

Budget Category Total	Due Date	Total Budget	Total Actual	Comment	Status	Feedback
Category	07/10/2016	Total Budget	Total Actual	Is this okay?	Denied	It's not okay.

Vendors

Vendor Type	Due Date	Number	Comment	Status	Feedback
Category	07/10/2016	Total	How about getting some clowns?	 Requested	

1037_Suggestions View

The screenshot shows the Tadpole software interface for event management. The top navigation bar includes links for Tadpole, Create Event, My Events (highlighted in yellow), Suppliers, Volunteers, About, and Administration. The top right corner shows a user profile for 'Tess Data' with a notification bell icon. The left sidebar lists various event management tasks: Event Details, Tasks, Itinerary, Budget, Advertising, After Event Report, Files, and Invitations. The main content area is titled 'Create Request' and 'Suggestions'. It contains three tables:

- Event Details**

Event Detail	Due Date	Content	Comment	Status	Feedback
Title	07/10/2016	Description	Can we rename it?	Requested	
Date	07/10/2016	08/01/2016	Does this date work?	Approved	That would be a good day.
Location	07/10/2016	Name, Address, City, State, Zip		Requested	
- Budget**

Budget Category Total	Due Date	Total Budget	Total Actual	Comment	Status	Feedback
	07/10/2016			Is this okay?	Denied	It's not okay.
- Vendors**

Vendor Type	Due Date	Number	Comment	Status	Feedback
Category	07/10/2016	Total	How about getting some clowns?	Requested	

Each table includes a 'Comment' column with a pencil icon for editing and a 'Status' column indicating the current status of the suggestion.

4.3.1.5 Budget (1021,1032,1033,1034)

1034_View_Expenses

The screenshot shows a web-based event management application. The sidebar on the left contains links for Event Details, Tasks, Itinerary, Budget (which is selected and highlighted in blue), Forum, After Event Report, Files, and Invitations. The main content area is titled "Create Budget". It features a table with columns for "Category" and "Planned Amount". A blue "Add" button is located above the table. Below the table, there is a section for "Add Expense" with fields for "Expense Title" and "Amount". A "Delete Category" button is located in the bottom right corner of the main content area. At the bottom, there is a "Summary" table with columns for "Category Name", "Planned Amount", and "Spent Amount".

Category	Planned Amount
Food	500.00

Add Expense **+**

Expense Title (e.g. soft drinks)	Amount (e.g. 45.50)
Burgers	123.00

Delete Category

Category Name	Planned Amount	Spent Amount
Category	\$0.00	
...
Event Total Budg	\$0.00	

1033_Remove_Budget_Line

The screenshot shows a web application interface for managing events. The top navigation bar includes links for Home, Create Event, My Events (highlighted in yellow), Suppliers, Volunteers, About, and Administration. The user is logged in as 'User'.

The left sidebar contains links for Event Details, Tasks, Itinerary, Budget (highlighted in blue), Forum, After Event Report, Files, and Invitations.

The main content area is titled 'Create Budget'. A modal window titled 'Delete?' is open, asking 'Are you sure you want to remove this item?'. The modal shows a list item 'Food' with a sub-item 'Burgers' and a value of '123.00'. Below the modal, there is a summary table:

Summary		
Category Name	Planned Amount	Spent Amount
Category	\$0.00	
...
Event Total Budg	\$0.00	

On the right side of the main content area, there is a 'Remaining: Responsive' section with icons for a checkmark, a pencil, and a trash can. A 'Delete Category' button is located at the bottom right of the modal.

1032_Add Expense

The screenshot shows a software interface for managing event budgets. The top navigation bar includes links for 'Create Event', 'My Events', 'Suppliers', 'Volunteers', 'About', and 'Administration', along with user and search options. The left sidebar contains links for 'Event Details', 'Tasks', 'Itinerary', 'Budget' (which is selected and highlighted in blue), 'Forum', 'After Event Report', 'Files', and 'Invitations'. The main content area is titled 'Create Budget' and features a table for adding expenses. The table has columns for 'Category' (Food) and 'Planned Amount' (500.00). A sub-table for 'Food' shows an entry for 'Burgers' with an amount of 123.00. There are edit and delete icons for each row. A 'Delete Category' button is located at the bottom right of the main table. A summary table at the bottom provides an overview of the budget categories, planned amounts, and spent amounts.

Category	Planned Amount
Food	500.00

Expense Title (e.g. soft drinks)	Amount (e.g. 45.50)	
Burgers	123.00	

Summary		
Category Name	Planned Amount	Spent Amount
Category	\$0.00	
...
Event Total Budg	\$0.00	

1021_Add Category

The screenshot shows a software interface for managing event budgets. The top navigation bar includes links for 'Create Event', 'My Events' (which is highlighted in yellow), 'Suppliers', 'Volunteers', 'About', and 'Administration'. The right side of the header features icons for user profile, notifications, and a search bar with a 'Find' button.

The main content area is titled 'Event Name Budget'. On the left, a vertical sidebar lists navigation options: 'Event Details', 'Tasks', 'Itinerary', 'Budget' (which is highlighted in blue), 'Forum', 'After Event Report', 'Files', and 'Invitations'. A large 'Add' button is positioned above a table where a new category is being added. The table has columns for 'Category' (containing 'Food') and 'Planned Amount' (containing '\$500.00').

Below the table, a message box displays 'Food' as the category, 'Planned : e.g. 500.00' as the amount, and 'Remaining: Responsive' as the status. It includes a 'Delete Category' button. A summary table at the bottom shows a single row for the 'Category' with a planned amount of '\$0.00'. The table has columns for 'Category Name', 'Planned Amount', and 'Spent Amount'.

Category Name	Planned Amount	Spent Amount
Category	\$0.00	
...
Event Total Budg	\$0.00	

4.3.1.6 Emergency Procedures(1004, 1035, 1036)

Emergency Procedures Edit

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration Tess Data

Event Details <Event Title> Emergency Procedures Search Find

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

Emergency Contacts
In the case of an emergency, always call 911 first.

Name	Number
The Wizard of Oz	319-123-4567
<input type="text"/>	<input type="text"/>

Building Exits
Previous description of exits.

Muster Points
Previous text about where to meet in case of emergency. Text box appears when clicking the pencil.

Map 481 x 410

Map 481 x 410

Emergency Procedures Creation

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration Tess Data

Event Details

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

<Event Title> Emergency Procedures

Emergency Contacts

Please call 911 in case of an emergency.

Emergency Contact Name Emergency Contact Number

Emergency Contact Name Emergency Contact Number

Building Exits

Description of exits.

Map

284 x 138

Upload

Muster Points

Where to meet in case of emergency.

Map

284 x 138

Upload

Save Cancel

4.3.1.7 Event Edit (1010)

edit create (1)

The screenshot shows a web-based application for event management. The top navigation bar includes links for Home, Create Event, My Events, Suppliers, Volunteers, About, and Administration. A user profile icon for 'Tess Data' is also present. The main content area is titled 'edit create (1)'. On the left, a vertical sidebar lists navigation links: Event Details (selected), Tasks, Itinerary, Budget, Advertising, After Event Report, Files, and Invitations. The main form area contains fields for 'Title' (My Title), 'Description' (My Description), and 'Budget' (My Budget). Below these fields are 'Edit' and 'Save' buttons. The top right of the main area has a 'Search' bar and a 'Find' button.

Event Details	Tasks	Itinerary	Budget	Advertising	After Event Report	Files	Invitations
My Title	My Description	My Budget					

Event Details: My Title

Description: My Description

Budget: My Budget

Buttons: Edit, Save

Edit Location

Event Details

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

Search

Find

Create Dates Location Directions Confirm

Name: building, park, etc

Street:

City:

State:

Zip Code:

Edit Save

Edit Dates/Times

The screenshot shows a software interface for managing event details. On the left, a vertical sidebar lists various event management tasks: Event Details (selected), Tasks, Itinerary, Budget, Advertising, After Event Report, Files, and Invitations. The main content area is titled 'Edit Dates/Times' and contains a sub-section for 'Event Dates'. This section includes a table with columns for Date, Start Time, and End Time. The 'Event Date' is set to 08/01/2016. The 'Start Time' is set to 10:30 AM, and the 'End Time' is set to 11:00 AM. At the bottom of the main area are two buttons: 'Edit' and 'Save'.

Event Details

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

Search

Find

Create

Dates

Location

Directions

Confirm

Event Dates

Date	Start Time	End Time

Event Date 08/01/2016

Start Time 10 : 30 AM

End Time 11 : 00 PM

Edit

Save

Edit Directions

The screenshot shows a web-based event management application. The top navigation bar includes links for Home, Create Event, My Events (highlighted in yellow), Suppliers, Volunteers, About, Administration, and a user profile for Tess Datta. A search bar and a 'Find' button are also present. The left sidebar contains links for Event Details, Tasks, Itinerary, Budget, Advertising, After Event Report, Files, and Invitations. The main content area is titled 'Edit Directions' and features tabs for Create, Date/Times, Location, Directions (selected), and Confirm. Under the Directions tab, there are two sections: 'Directions 1' and 'Directions 2'. Each section has fields for 'Direction', 'Title', 'Give', and 'Directions'. Below each section are 'Edit' and 'Save' buttons. The 'Directions 1' section is currently active.

4.3.1.8 Event(1009,1011,1012,1013,1015)

Title and Description

Create Event My Events Suppliers Volunteers About Administration Tess Data

Search Find

Create Date/Time Location Directions Confirm

Create New Event

Title Text

Description Text

Budget Text

Next Cancel

Event Date - Add Date

Event Date - Add Date

Home Create Event My Events Suppliers Volunteers About Administration Tess Data

Search Find

Create Date Location Directions Confirm

Select Date for Event

Event Date: 01/02/2022

Start Time: 10 : 30 AM

End Time: 11 : 00 PM

Add Date

Next Cancel

Event Date - Calendar

Event Date - Calendar

Home Create Event My Events Suppliers Volunteers About Administration Tess Data Search Find

Create Date Location Directions Confirm

Select Date for Event

Event Date: 01/02/2022

Start Time: 10 : 30

End Time: 11 : 00

Next **Cancel**

Jan - 2022

S	M	T	W	T	F	Sa
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

Event Date - Date Added

The screenshot shows a web-based event management system. At the top, a blue header bar contains navigation links: Home, Create Event, My Events, Suppliers, Volunteers, About, Administration, and a user profile for 'Tess Data'. Below the header is a search bar with a 'Find' button. The main content area has a light gray background and is divided into several sections:

- Create** (blue button)
- Date** (selected tab, white button)
- Location**
- Directions**
- Confirm**

Select Date for Event

Event Date:

Start Time: :

End Time: :

Add Date (button)

Currently Selected Dates

Date	Start Time	End Time
01/01/2022	8:00 am	10:30 am

Next (blue button) **Cancel** (red button)

Location

The screenshot shows a web-based application interface for event planning. At the top, there is a blue header bar with the following navigation items: Home, Create Event (highlighted in yellow), My Events, Suppliers, Volunteers, About, Administration, and a user profile for 'Tess Data'. Below the header is a search bar with a 'Search' input field and a 'Find' button. The main content area is titled 'Select Location' and contains the following form fields:

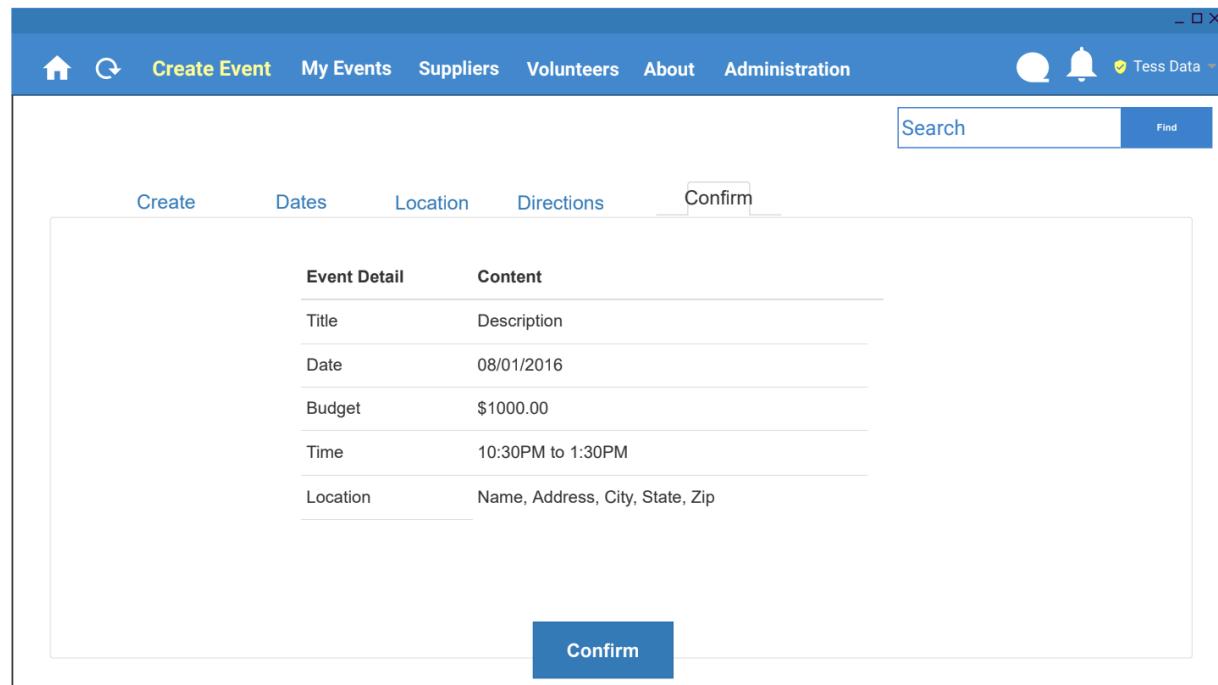
Name	building, park, etc	
Street		
City	State	Zip Code

At the bottom of the form are two buttons: a blue 'Next' button and a red 'Cancel' button.

Directions

The screenshot shows a web application interface for creating an event. The top navigation bar includes links for Home, Create Event, My Events, Suppliers, Volunteers, About, and Administration. There are also icons for a profile, a bell, and a user named Tess Data. A search bar and a 'Find' button are located in the top right. The main content area is titled 'Create' and shows the current step as 'Directions'. Below this, a sub-step titled 'Add Direction' is displayed. The 'Add Direction' form contains two fields: 'Title' (with placeholder 'Name if exists') and 'Directions' (with placeholder 'Give Directions'). A blue 'Add' button is positioned to the right of the text area. At the bottom of the form are 'Next' and 'Cancel' buttons. The overall layout is clean and modern, using a light blue and white color scheme.

Confirm



The screenshot shows a web-based application interface for event management. At the top, there is a blue header bar with the following elements from left to right: a house icon, a refresh icon, the text "Create Event", "My Events", "Suppliers", "Volunteers", "About", "Administration", a user icon, a bell icon, and the text "Tess Data". Below the header is a search bar with the placeholder "Search" and a "Find" button. The main content area has a title "Confirm" and a navigation bar with tabs: "Create", "Dates", "Location", "Directions", and "Confirm". The "Confirm" tab is currently active. Below this is a table with two columns: "Event Detail" and "Content". The table contains the following data:

Event Detail	Content
Title	Description
Date	08/01/2016
Budget	\$1000.00
Time	10:30PM to 1:30PM
Location	Name, Address, City, State, Zip

At the bottom of the form is a large blue "Confirm" button.

Share Event

The screenshot shows a web-based application interface for managing events. At the top, a blue header bar contains navigation links: Home, Create Event, My Events, Suppliers, Volunteers, About, and Administration. To the right of these are icons for a speech bubble, a bell, and a user profile, with the text "Tess Data" and a dropdown arrow. The main content area is titled "Share Event". It features a form with fields for "Event Name" (containing "Description of event is happening here"), "Description" (containing "Description of event is happening here"), "Date" (containing "Date of Event is here"), and "Location" (containing "Location of the event is listed here. Building included (if relevant)"). To the right of the form is a vertical toolbar with buttons for "Search" (with a "Find" sub-link), "Share", "Edit", and "Info Page". The "Share" button is highlighted with a blue border.

List and Filter Events

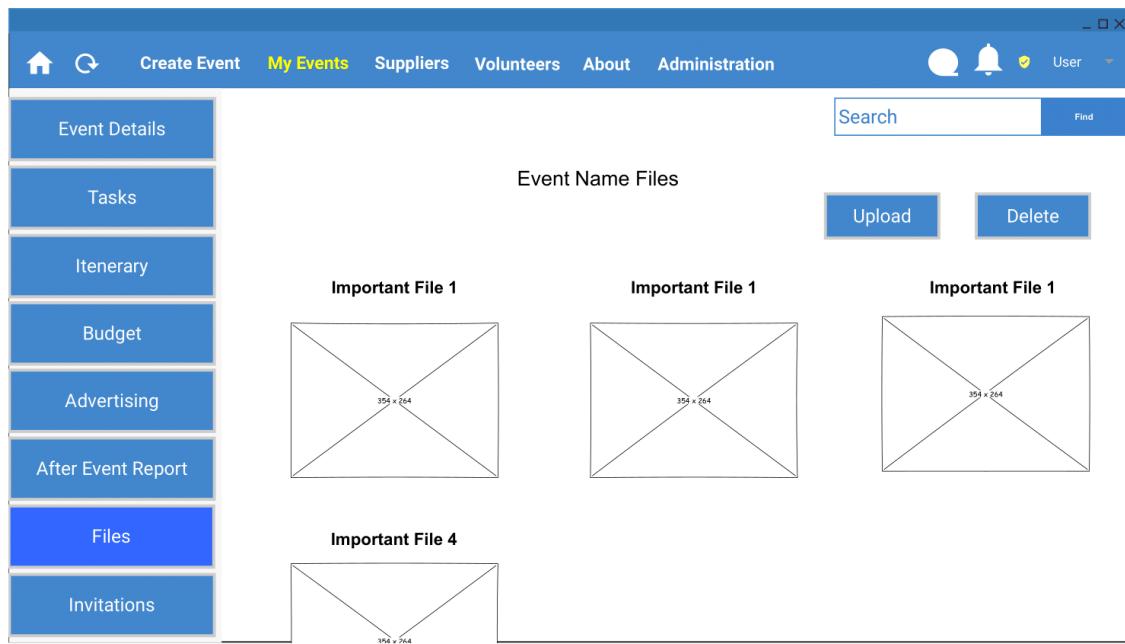
Event Filter

- All Events (selected)
- My Events
- Upcoming Events
- Past Events
- Upcoming and Past Events

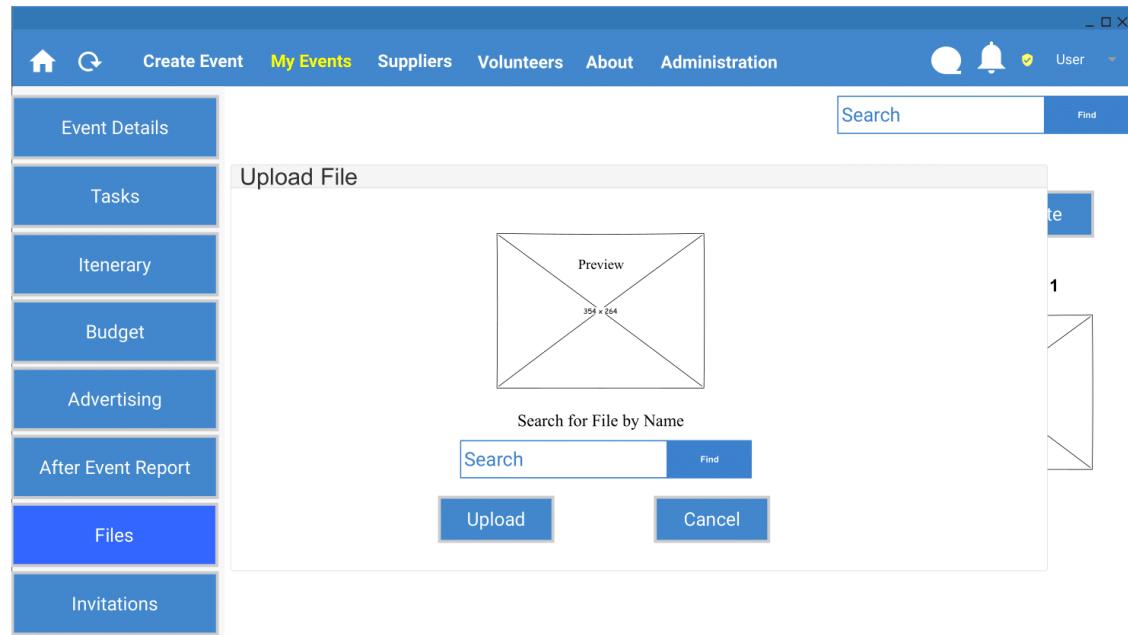
Event Name	Event Description	Dates	Location
Event 1	Description for event 1	01/01/2022 01/02/2022 01/03/2022	Place, City, State
Event 2	Description for event 2	01/02/2022 01/03/2022	Place, City, State
Event 3	Description for event 3	01/03/2022	Place, City, State

4.3.1.9 File Upload (1025,1052,1053)

After Upload



Upload Files



Event Details

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

Upload File

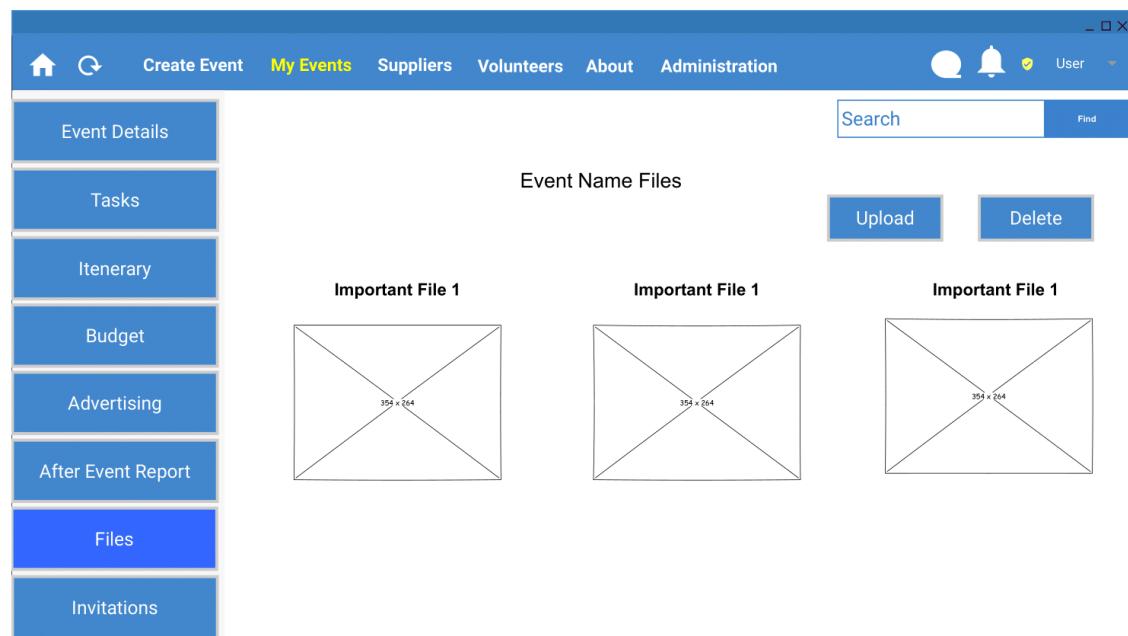
Preview
354 x 264

Search for File by Name

Search Find

Upload Cancel

View Files



Event Details

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

Event Name Files

Important File 1

Important File 1

Important File 1

Upload Delete

354 x 264

354 x 264

354 x 264

4.3.1.10 Forum (Desktop)(1008,1016,1048,1049,1056,1057,1058,1059)

1058_Forum Comment Create

Group Title

Message Sub Cat

- # Chat Group Name
- # Add

Message Sub Cat

- # Chat Group Name

Indiv Messages

#Inbox

Forum Name

Forum Description... Bracchia septemque innabilis origo
obstatatque diffundi os nullo sive nullo animal illic
ponitus, ripis quoque.

999 User Name 10/6/2021 at 12:00pm

999 User Name 10/6/2021 at 12:30pm

Your User Name

Type comment here

1049_Forum Delete Post

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration

Search Find

Group Title

Message Sub Cat

- # Chat Group Name
- # Add

Message Sub Cat

- # Chat Group Name

Indiv Messages

- #Inbox

Forum Name

Forum Description... Bracchia septemque innabilis origo
obstabatque diffundi os nullo sive nullo animal illic
pontus, ripis quoque.

User Name
Secrevit fontes liquidum locoque pronaque?
Illa semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit iudicis aestu fixo

999 999 10/6/2021 at 12:00pm

User Name
Secrevit fontes liquidum locoque pronaque?
Illa semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit iudicis aestu fixo

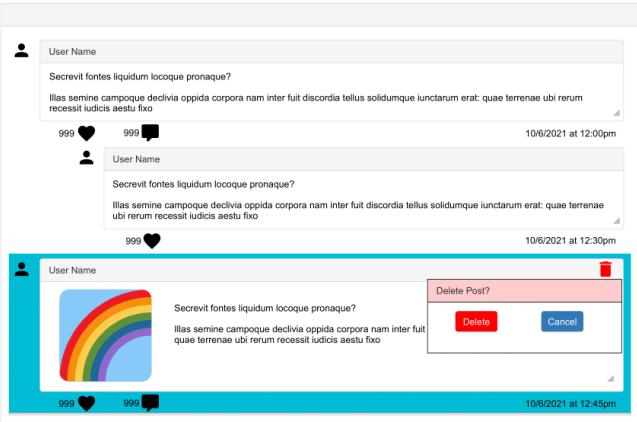
999 10/6/2021 at 12:30pm

User Name
Secrevit fontes liquidum locoque pronaque?
Illa semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit iudicis aestu fixo

999 999 10/6/2021 at 12:45pm

Delete Post?

Delete Cancel



1016_Voting (Create Post add voting)

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration

Forum Name

Forum Description... Braccia septemque innabilis origo
obstatabatque diffundi os nullo sive nullo animal illic
pontus, ripis quoque.

Search Find

Group Title

Message Sub Cat

- # Chat Group Name

Add

Message Sub Cat

- # Chat Group Name

Indiv Messages

#Inbox

User Name
Secrevit fontes liquidum locoque pronaque?
Illes semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit iudicis aestu fixo

999 999 10/6/2021 at 12:00pm

User Name
Secrevit fontes liquidum locoque pronaque?
Illes semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit iudicis aestu fixo

10/6/2021 at 12:30pm

Add Attachment
Y Voting Poll
Due Date: 10/1/2021 at 12:00pm
Single Vote Ranking List
Option 1
Add another option ✓ X

1048_Forum Create Post (add image)

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration

Forum Name

Forum Description... Braccia septemque innabilis origo
obstatabatque diffundi os nullo sive nullo animal illic
pontus, ripis quoque.

Search Find

Group Title

Message Sub Cat

- # Chat Group Name

Add

Message Sub Cat

- # Chat Group Name

Indiv Messages

#Inbox

User Name
Secrevit fontes liquidum locoque pronaque?
Illes semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit iudicis aestu fixo

999 999 10/6/2021 at 12:00pm

User Name
Secrevit fontes liquidum locoque pronaque?
Illes semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit iudicis aestu fixo

10/6/2021 at 12:30pm

Your User Name
Add Attachment
Image Choose Image ✓ X

1048_Forum Create Post

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration   Tess Data

Group Title

Message Sub Cat

- # Chat Group Name
- # Add

Message Sub Cat

- # Chat Group Name

Indiv Messages

- #Inbox

Forum Name

Forum Description... Braccia septemque innabilis origo
obstabatque diffundi os nullo sive nullo animal illic
pontus, ripis quoque.

Search Find

User Name
Secrevit fontes liquidum locoque pronaque?
Illa semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit iudicis aestu fixo

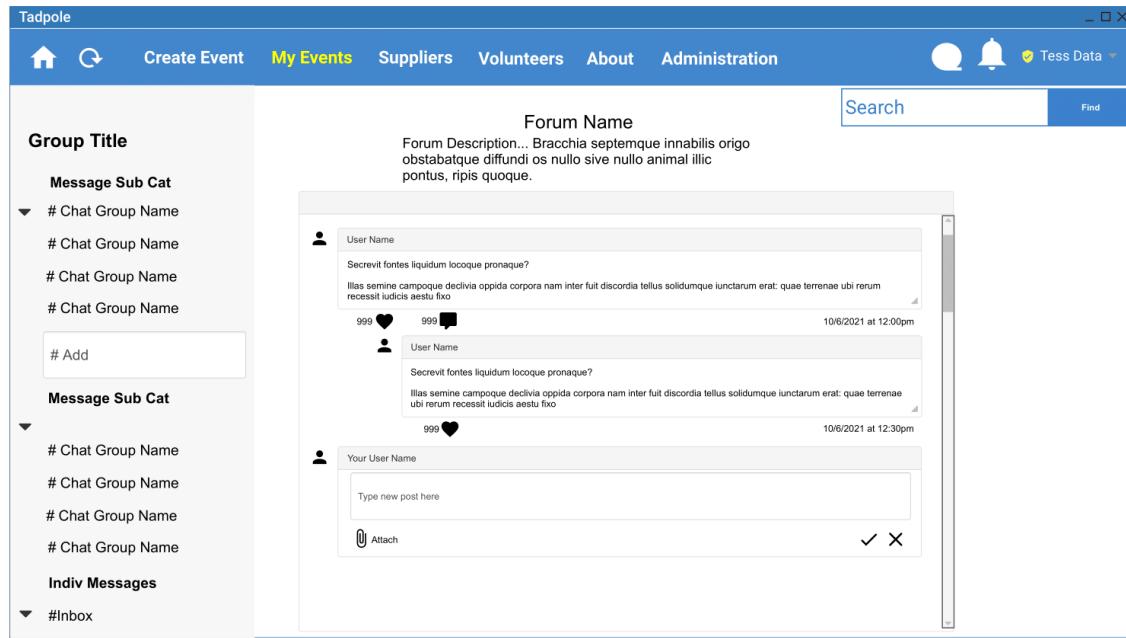
999  999  10/6/2021 at 12:00pm

User Name
Secrevit fontes liquidum locoque pronaque?
Illa semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit iudicis aestu fixo

999  10/6/2021 at 12:30pm

Your User Name
Type new post here

Attach  



1057_Voting Results View

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration

Search Find

Group Title

Forum Name

Forum Description... Bracchia septemque innabilis origo
obstatabatque diffundi os nullo sive nullo animal illic
ponitus, ripis quoque.

Message Sub Cat

Chat Group Name
Add

Message Sub Cat

Chat Group Name
Chat Group Name
Chat Group Name
Chat Group Name
Chat Group Name

Indiv Messages

#Inbox

Results from 999 people:

Drink	Percentage
Pepsi	25%
Coke	25%
Sprite	25%
Mountain Dew	25%

Expired: 10/10/2021 at 12:00pm

999 999 10/6/2021 at 12:00pm

User Name

What soft drink do you most want to see at the event?

Results from 999 people:

Drink	Percentage
Pepsi	25%
Coke	25%
Sprite	25%
Mountain Dew	25%

Expired: 10/10/2021 at 12:00pm

999 999 10/6/2021 at 12:30pm

User Name

Secrevit fontes liquidum locoque pronaque?

Illes semine campoque decilia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrena
ubi rerum recessit iudicis aestu fixo

999 10/6/2021 at 12:30pm

1056_Voting Cast Vote

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration

Search Find

Group Title

Forum Name

Forum Description... Bracchia septemque innabilis origo
obstatabatque diffundi os nullo sive nullo animal illic
ponitus, ripis quoque.

Message Sub Cat

Chat Group Name
Chat Group Name
Chat Group Name
Chat Group Name
Add

Message Sub Cat

Chat Group Name
Chat Group Name
Chat Group Name
Chat Group Name

Indiv Messages

#Inbox

Please Pick One:

Pepsi
 Coke
 Sprite
 Mountain Dew

Expires: 10/10/2021 at 12:00pm

999 999 10/6/2021 at 12:00pm

User Name

What soft drink do you most want to see at the event?

Please Pick One: Pepsi
 Coke
 Sprite
 Mountain Dew

Expires: 10/10/2021 at 12:00pm

999 999 10/6/2021 at 12:30pm

User Name

Secrevit fontes liquidum locoque pronaque?

Illes semine campoque decilia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrena
ubi rerum recessit iudicis aestu fixo

999 10/6/2021 at 12:30pm

User Name

What are your favorite soft drinks?

Please Rank from 1(best) to 4(worst):

Drink
Pepsi
Coke
Sprite
Mountain Dew

1059_Forum Edit

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration

Search Find Tess Data

Edit Forum

Forum Name: Enter Forum Name

Forum Description: Enter Forum Description

Participant Groups: Add Participant Groups

Volunteers (RSVP'd attendees) (All Users)

Add Single User

User Name

Save Cancel

1008_Forum (Forum View)

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration

Search Find Tess Data

Group Title

Message Sub Cat

Chat Group Name

Add

Message Sub Cat

Chat Group Name

Indiv Messages

#Inbox

Forum Name

Forum Description... Bracchia septemque innabili origo
obstabatque diffundi os nullo sive nullo animal illic
pontus, ripis quoque.

Edit

User Name

Secrevit fontes liquidum locoque pronaque?

Ilas semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit iudicis aestu fixo

999 10/6/2021 at 12:00pm

User Name

Secrevit fontes liquidum locoque pronaque?

Ilas semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit iudicis aestu fixo

999 10/6/2021 at 12:30pm

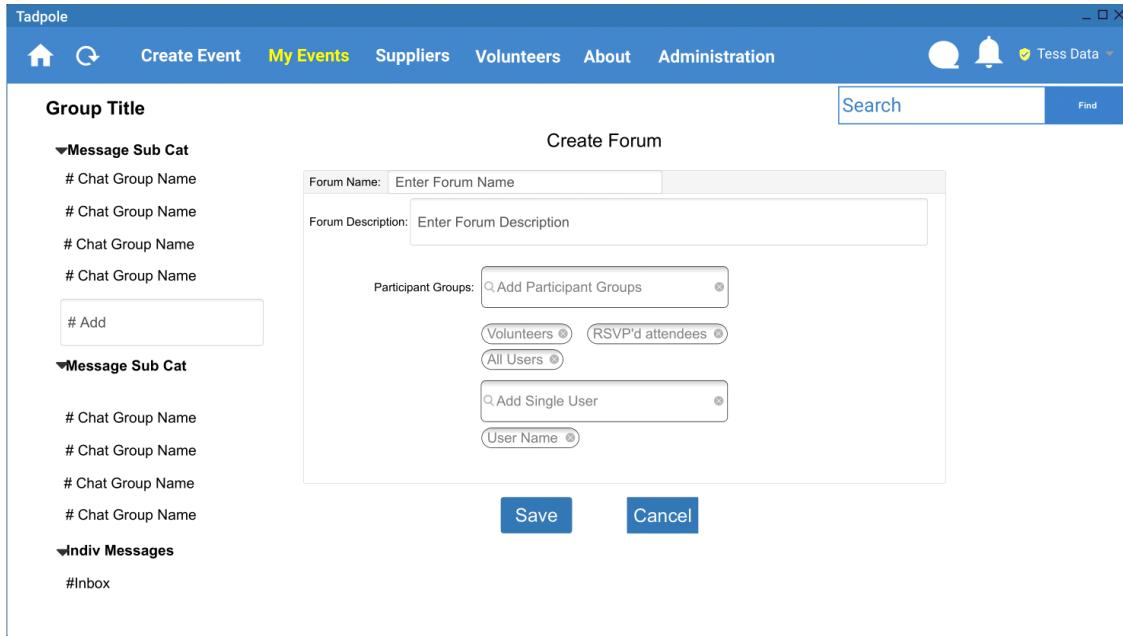
User Name

Secrevit fontes liquidum locoque pronaque?

Ilas semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit iudicis aestu fixo

999 10/6/2021 at 12:45pm

1008_Forum



The screenshot shows the Tadpole software interface with the following details:

- Header:** Tadpole, Create Event, **My Events** (highlighted in yellow), Suppliers, Volunteers, About, Administration, a message icon, a bell icon, and a user profile for Tess Data.
- Left Sidebar:**
 - Group Title:** # Chat Group Name, # Add.
 - Message Sub Cat:** # Chat Group Name, # Chat Group Name, # Chat Group Name, # Chat Group Name.
 - Indiv Messages:** #Inbox.
- Central Form:**
 - Create Forum:** Forum Name: Enter Forum Name, Forum Description: Enter Forum Description.
 - Participant Groups:** Add Participant Groups (dropdown), Volunteers (radio button), RSVP'd attendees (radio button), All Users (radio button).
 - Single User:** Add Single User (dropdown), User Name (text input).
- Buttons:** Save, Cancel.

4.3.1.11 Invitations (1027,1028)

Send_Approve_View

The screenshot shows a web application interface for managing event details. The left sidebar contains a vertical list of blue buttons with white text, including Event Details, Tasks, Itinerary, Budget, Forum, After Event Report, Files, Invitations (which is the active tab), Emergency Procedures, and Suggestions. The main content area is titled "Send Invitation" and includes tabs for "Invitation", "Recipients", and "Approve". The "Approve" tab is active, showing a list of recipients with checkboxes. The list includes Person 1 (person1@email.com), Person 2 (person2@email.com), Person 3 (person2@email.com), and Person 4 (person2@email.com). Below the list are "Select All" and "Unselect All" checkboxes. A modal dialog box titled "Send?" is displayed, asking "Are you sure you would like to send all these invitations?", with "Send" and "Cancel" buttons. The top navigation bar includes links for Create Event, My Events (highlighted in yellow), Suppliers, Volunteers, Find Event, About, Administration, and a user profile for Tess Data.

Send_Recipients_View

The screenshot shows a software application window titled "Send_Recipients_View". The top navigation bar includes links for "Create Event", "My Events" (which is highlighted in yellow), "Suppliers", "Volunteers", "Find Event", "About", "Administration", and a user profile for "Tess Data". A search bar and a "Find" button are also present in the top right.

The main content area is titled "Send Invitation" and is divided into three tabs: "Invitation" (selected), "Recipients", and "Approve". The "Recipients" tab is currently active, showing a sub-section titled "Select Invitation Recipients".

On the left, a sidebar lists various event management categories: Event Details, Tasks, Itinerary, Budget, Forum, After Event Report, Files, Invitations (which is highlighted in blue), Emergency Procedures, and Suggestions.

The "Recipients" section includes options to "Import From" (with "Gmail" and "Email" buttons), a search bar, and an "Add" button. It also features "Add Email" and "Add Multiple Emails" fields. The "Add Multiple Emails" field contains the placeholder text "example1.com, example2.com, example3.com".

On the right, a table lists four recipients with their names and emails:

Name	Email
Person 1	person1@email.com
Person 2	person2@email.com
Person 3	person2@email.com
Person 4	person2@email.com

At the bottom of the "Recipients" section are "Next" and "Cancel" buttons.

Send_Edit_View

The screenshot shows a web-based application interface for managing events. The top navigation bar includes links for 'Create Event', 'My Events' (which is the active tab), 'Suppliers', 'Volunteers', 'Find Event', 'About', 'Administration', and user-specific options like 'Tess Data' and notifications. A sidebar on the left contains links for 'Event Details', 'Tasks', 'Itinerary', 'Budget', 'Forum', 'After Event Report', 'Files', 'Invitations' (which is the active tab), 'Emergency Procedures', and 'Suggestions'. The main content area is titled 'Send Invitation' and displays a message: 'You are invited to <Event Title>'. Below this is a large empty box with a '415 x 310' placeholder. To the right, there are several descriptive labels and their corresponding event details: 'Event Description' (Message from Event Promoter), 'Location' (Information pulled from event), 'Date' (Information pulled from event), 'Time' (Information pulled from event), and an 'Edit' button. At the bottom are 'Next' and 'Cancel' buttons.

Create Confirmation

Screenshot of a software application interface titled "Create Confirmation". The top navigation bar includes "Create Event", "My Events" (highlighted in yellow), "Suppliers", "Volunteers", "Find Event", "About", "Administration", and user "Tess Data". A sidebar on the left lists "Event Details", "Tasks", "Itinerary", "Budget", "Forum", "After Event Report", "Files", "Invitations" (highlighted in blue), "Emergency Procedures", and "Suggestions". The main content area is titled "Create Invitation" and shows a "Confirm" section with a placeholder text "You are invited to <Event Title>". Below this is a large empty box with a "415 x 310" dimension label. To the right, there are fields for "Event Description", "Message from Event Promoter", "Location", "Date", and "Time", each with a "Delete" icon. A "Save" button is at the bottom left, a "Save and Select Recipients" button is in the center, and a "Cancel" button is on the right.

Event Details

Tasks

Itinerary

Budget

Forum

After Event Report

Files

Invitations

Emergency Procedures

Suggestions

My Events

Suppliers

Volunteers

Find Event

About

Administration

Tess Data

Search

Details

Picture

Confirm

Event Description

Message from Event Promoter

Location

Date

Time

415 x 310

You are invited to <Event Title>

Save

Save and Select Recipients

Cancel

Create_Pics

Event Details

Tasks

Itinerary

Budget

Forum

After Event Report

Files

Invitations

Emergency Procedures

Suggestions

Create Invitation

Details Picture Confirm

Add Picture

Add a picture? Yes

Select 1:

275 x 231

278 x 231

277 x 231

277 x 239

Next Cancel Upload Picture

Create_Details

Event Details

Tasks

Itinerary

Budget

Forum

After Event Report

Files

Invitations

Emergency Procedures

Suggestions

Create Invitation

Details Picture Confirm

Event Details

Location: Information pulled from event or message about it not determined yet

Date: Information pulled from event or message about it not determined yet

Time: Information pulled from event or message about it not determined yet

Require RSVP? Yes

Message:

Next Cancel

4.3.1.12 Itinerary (1020, 1041, 1042, 1043, 1044, 1045, 1046)

1043_Itinerary List View (1)

Tadpole

— □ ×

Home Create Event **My Events** Suppliers Volunteers About Administration   User Name 

Event Details  Search 

Itinerary For Kids 

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

Friday, October 1st

Occurrence Name

Occurrence Description

Start Time 10/1/2021 at 12:00pm

End Time 10/1/2021 at 2:00pm



1043_Itinerary List View

Tadpole

Create Event **My Events** Suppliers Volunteers About Administration

User Name

Event Details

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

View Itineraries

Itinerary For Kids

Itinerary description

Itinerary For Teens

Itinerary description

1042_Itinerary Delete

Tadpole

— □ ×

[Create Event](#) [My Events](#) [Suppliers](#) [Volunteers](#) [About](#) [Administration](#) User Name

Event Details Search X

Set Public Set Private

Edit Itinerary

Itinerary Title: Itinerary Description:

Are you sure?

Deleting an itinerary cannot be undone.

End Time Select Date and Time Image

1045_Itinerary Delete Occurrences

Tadpole

— □ ×

[Create Event](#) [My Events](#) [Suppliers](#) [Volunteers](#) [About](#) [Administration](#) User Name [Logout](#)

Event Details

Set Public Set Private

Edit Itinerary

Itinerary Title:

Itinerary Description:

Occurrence Name:

Occurrence Description:

Start Time: 10/1/2021 at 12:00pm

End Time: 10/1/2021 at 2:00pm

Delete Occurrence?

[Delete](#) [Cancel](#)

Save [Cancel](#)

Event Details

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

Set Date and Time

Tadpole

Create Event My Events Suppliers Volunteers About Administration User Name

Event Details

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

Set Public

Set Private

Itinerary Title: Create/Edit Itinerary

Itinerary Description

Occurrence Name

Occurrence Description

Start Time  10/1/2021 at 12:00pm 

Set Date and Time

Set Date 10/1/2021 

Set Time 12:00 pm 

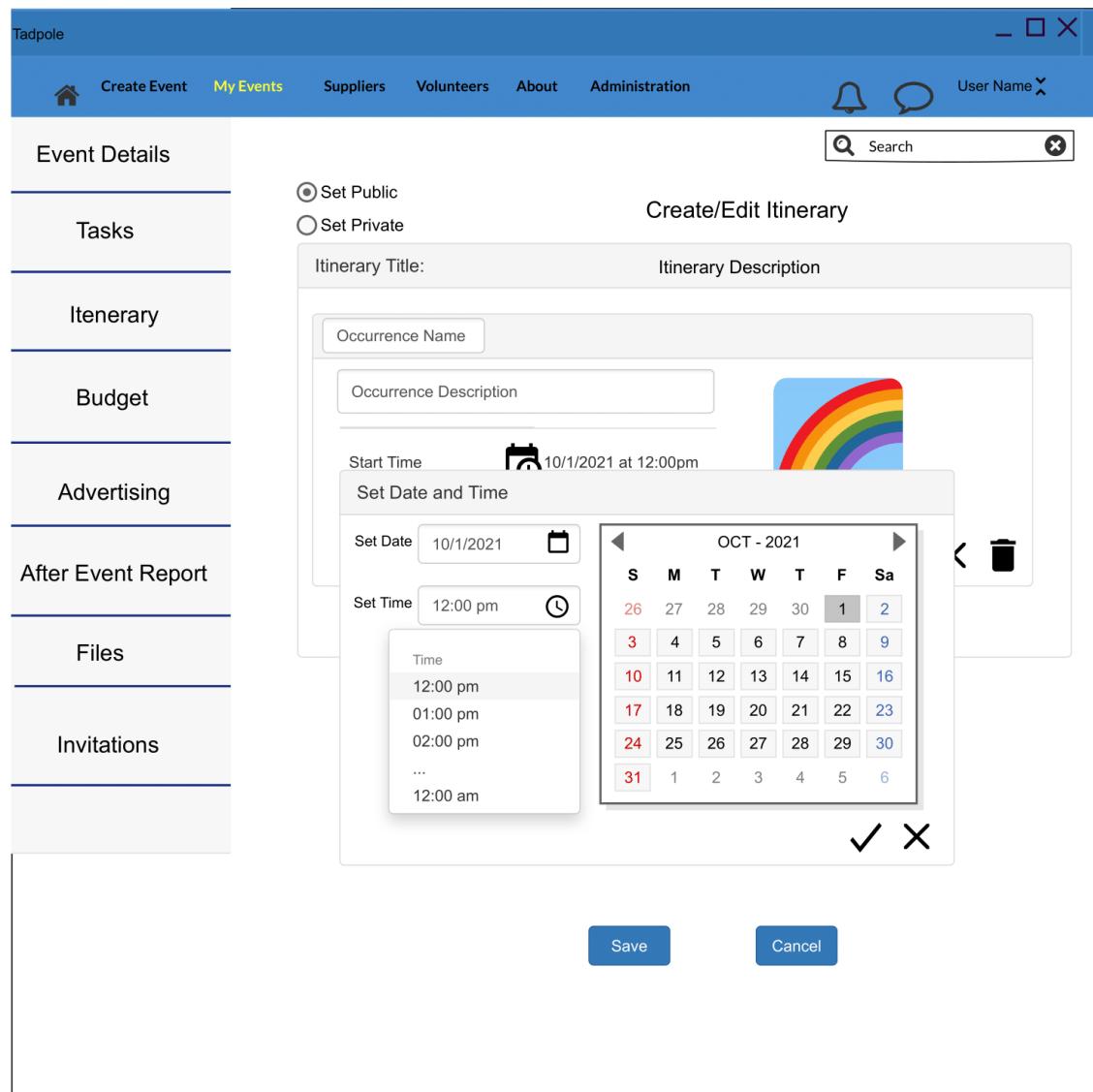
Time
12:00 pm
01:00 pm
02:00 pm
...
12:00 am

OCT - 2021

S	M	T	W	T	F	Sa
26	27	28	29	30	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

✓ ✕

Save Cancel



1046_Itinerary Edit Occurrences

Tadpole

[Create Event](#) [My Events](#) [Suppliers](#) [Volunteers](#) [About](#) [Administration](#) User Name

Set Public Set Private

Edit Itinerary

Itinerary Title:

Itinerary Description:

Occurrence Name:

Occurrence Description:

Start Time: 10/1/2021 at 12:00pm

End Time: 10/1/2021 at 2:00pm

Change Image

[Save](#) [Cancel](#)

Event Details

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

1044_Itinerary Edit

Tadpole

— □ ×

[Create Event](#) [My Events](#) [Suppliers](#) [Volunteers](#) [About](#) [Administration](#) User Name [X](#)

Event Details [X](#)

Set Public Set Private

Edit Itinerary

Itinerary Title: Itinerary Description:

Occurrence Name:

Occurrence Description:

Start Time: 10/1/2021 at 12:00pm 

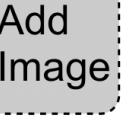
End Time: 10/1/2021 at 2:00pm 

Enter New Occurrence Name:

Enter New Occurrence Description (ie. location, what's happening):

Start Time: Select Date and Time 

End Time: Select Date and Time 

Add Image 

✓ ✕

[Save](#) [Cancel](#)

1020_Itinerary

Tadpole

— □ ×

[Create Event](#) **My Events** [Suppliers](#) [Volunteers](#) [About](#) [Administration](#) User Name

Event Details Search X

Set Public Set Private **Create Itinerary**

Itinerary Title: Itinerary Description:

Enter New Occurrence Name:
Enter New Occurrence Description (ie. location, what's happening):

Start Time: Select Date and Time:

End Time: Select Date and Time:

Add Image X ✓

Save **Cancel**

Tasks

Itinerary

Budget

Advertising

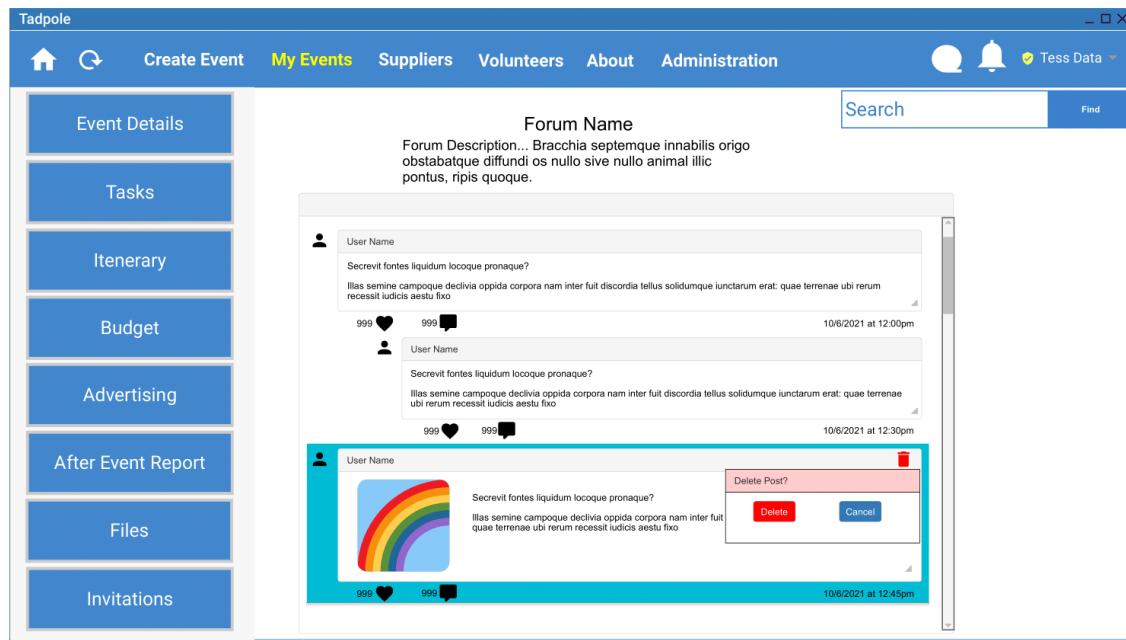
After Event Report

Files

Invitations

4.3.1.13 Poll (1016,1056,1057)

1049_Forum Delete Post



The screenshot shows the Tadpole application interface. The top navigation bar includes links for Home, Create Event, My Events (highlighted in yellow), Suppliers, Volunteers, About, and Administration. A search bar and a 'Find' button are also present. The left sidebar contains links for Event Details, Tasks, Itinerary, Budget, Advertising, After Event Report, Files, and Invitations. The main content area displays a forum post titled 'Forum Name' with the description: 'Forum Description... Bracchia septemque innabilis origo obstabatque diffundi os nullo sive nullo animal illic pontus, ripis quoque.' Below this, two posts are shown, each with a user name, a small profile picture, and a timestamp (10/6/2021 at 12:00pm and 10/6/2021 at 12:30pm). The post from 12:30pm is highlighted with a red border and a 'Delete Post?' dialog box. The dialog box contains the text 'Delete Post?' with 'Delete' and 'Cancel' buttons. The timestamp for this post is also 10/6/2021 at 12:45pm.

1016_Voting (Create Post add voting)

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration

Search Find

Event Details Tasks Itinerary Budget Advertising After Event Report Files Invitations

Forum Name

Forum Description... Bracchia septemque innabilis origo
obstatabatque diffundi os nullo sive nullo animal illic
pontus, ripis quoque.

User Name
Secretavit fontes liquidum locoque pronaque?
Illa semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae
ubi rerum recessit iudicis aestu fixo

999 999 10/6/2021 at 12:00pm

User Name
Secretavit fontes liquidum locoque pronaque?
Illa semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae
ubi rerum recessit iudicis aestu fixo

10/6/2021 at 12:30pm

Add Attachment
Y Voting Poll
Due Date: 10/1/2021 at 12:00pm
 Single Vote Ranking List
Option 1
Add another option ✓ ✕

1048_Forum Create Post (add image)

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration

Search Find

Event Details Tasks Itinerary Budget Advertising After Event Report Files Invitations

Forum Name

Forum Description... Bracchia septemque innabilis origo
obstatabatque diffundi os nullo sive nullo animal illic
pontus, ripis quoque.

User Name
Secretavit fontes liquidum locoque pronaque?
Illa semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae
ubi rerum recessit iudicis aestu fixo

999 999 10/6/2021 at 12:00pm

User Name
Secretavit fontes liquidum locoque pronaque?
Illa semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae
ubi rerum recessit iudicis aestu fixo

10/6/2021 at 12:30pm

Your User Name
Add Attachment
Image Choose Image ✓ ✕

1048_Forum Create Post

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration

Search Find

Event Details Tasks Itinerary Budget Advertising After Event Report Files Invitations

Forum Name

Forum Description... Bracchia septemque innabilis origo
obstatabatque diffundi os nullo sive nullo animal illic
pontus, ripis quoque.

User Name
Secrevit fontes liquidum locoque pronaque?
Illes semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae
recessit iudicis aestu fixo

999 999 10/6/2021 at 12:00pm

User Name
Secrevit fontes liquidum locoque pronaque?
Illes semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae
ubi rerum recessit iudicis aestu fixo

999 999 10/6/2021 at 12:30pm

Your User Name
Type new post here

Attach ✓ ✕

Forum View (Voting Poll Completed)

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration

Search Find

Event Details Tasks Itinerary Budget Advertising After Event Report Files Invitations

Forum Name

Forum Description... Bracchia septemque innabilis origo
obstatabatque diffundi os nullo sive nullo animal illic
pontus, ripis quoque.

User Name
What soft drink do you most want to see at the event?
Results:
Pepsi 25%
Coke 25%
Sprite 10%
Mountain Dew 40%

999 999 10/6/2021 at 12:00pm

User Name
Secrevit fontes liquidum locoque pronaque?
Illes semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae
ubi rerum recessit iudicis aestu fixo

999 999 10/6/2021 at 12:30pm

Forum View (Voting Poll In Progress)

The screenshot shows the Tadpole software interface. The top navigation bar includes links for Home, Create Event, My Events (which is selected), Suppliers, Volunteers, About, and Administration. There are also notification icons for messages and a user named Tess Data. A search bar is located in the top right corner.

The left sidebar contains a vertical list of event management tasks: Event Details, Tasks, Itinerary, Budget, Advertising, After Event Report, Files, and Invitations. The main content area displays a forum view with the following details:

- Forum Name:** Forum Description... Braccia septemque innabilis origo obstabatque diffundi os nullo sive nullo animal illic pontus, ripis quoque.
- Search:** A search bar with a 'Find' button.
- Voting Poll 1:** A poll titled "What soft drink do you most want to see at the event?". It asks "Please Pick One:" and lists options: Pepsi, Coke, Sprite, and Mountain Dew. The poll expires at 10/10/2021 at 12:00pm. A "Cast Vote" button is present. Two user entries are shown: one from "999" with a heart icon and another from "999" with a speech bubble icon. The timestamp for the first entry is 10/6/2021 at 12:00pm.
- Voting Poll 2:** A poll titled "Secerit fontes liquidum locoque pronaque?". It asks "Please Rank from 1(best) to 4(worst):" and lists options: Pepsi, Coke, Sprite, and Mountain Dew. The poll expires at 10/6/2021 at 12:30pm. Two user entries are shown: one from "999" with a heart icon and another from "999" with a speech bubble icon. The timestamp for the first entry is 10/6/2021 at 12:30pm.

Forum View

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration

Search Find

Event Details

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

Forum Name

Forum Description... Bracchia septemque innabilis origo
obstatabatque diffundi os nullo sive nullo animal illic
pontus, ripis quoque.

User Name
Secrebit fontes liquidum locoque pronaque?
Illes semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum
recessit iudicis aestu fixo

999 999 10/6/2021 at 12:00pm

User Name
Secrebit fontes liquidum locoque pronaque?
Illes semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae
ubi rerum recessit iudicis aestu fixo

999 999 10/6/2021 at 12:30pm

User Name
Secrebit fontes liquidum locoque pronaque?
Illes semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae
ubi rerum recessit iudicis aestu fixo

999 999 10/6/2021 at 12:45pm

1008_Forum

Tadpole

Home Create Event **My Events** Suppliers Volunteers About Administration

Search Find

Event Details

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

Create Forum

Enter Forum Name

Enter Forum Description

Add Participant Groups

Volunteers RSVP'd attendees All Users

Save Cancel

4.3.1.14 Task List (1018, 1050, 1051)

1018_Create_Task

The screenshot shows a web-based application interface for creating a task. On the left, there is a vertical sidebar with blue buttons labeled: Event Details, Tasks, Itinerary, Budget, Advertising, After Event Report, Files, and Invitations. The 'Tasks' button is highlighted. At the top, there is a navigation bar with icons for Home, Create Event, My Events (which is highlighted in yellow), Suppliers, Volunteers, About, Administration, and a User dropdown. Below the navigation bar, a message states: "Event name provided on the landing page for tasks. after create is clicked the Add task + shows up below". The main content area is titled "Event Name Tasks". It contains fields for "Task Name" (with a placeholder "Task Name"), "Task Description" (with a placeholder "Description"), "Assign" (set to "Me"), "Due By" (with a date input field and a calendar icon), and "Priority" (set to "Medium"). At the bottom, there are "Save" and "Cancel" buttons.

1018_Task_Done

The screenshot shows a software interface for managing events. At the top, there is a navigation bar with links for 'Create Event', 'My Events' (which is highlighted in yellow), 'Suppliers', 'Volunteers', 'About', and 'Administration'. To the right of the navigation bar are icons for a message bubble, a bell, and a user profile, with a dropdown arrow.

The main content area is titled 'Event Name's Tasks'. On the left, there is a vertical sidebar with blue buttons labeled: 'Event Details', 'Tasks' (which is highlighted in blue), 'Itinerary', 'Budget', 'Advertising', 'After Event Report', 'Files', and 'Invitations'. The 'Tasks' button is currently selected.

Below the sidebar, the title 'Event Name's Tasks' is displayed. To the right of the title is a table showing three tasks:

Task Name	Description	Due By
Vendor Available	Ensure that vendors are available	2/15/2022
Backup Venue	Talk to Samantha about the venue	2/22/2022
Inventory	Take inventory of cleaning supplies	2/28/2022

Below the table, a note states: '* Font color is based off of Task Priority. Red = High, Orange = Medium, Blue = Low'.

At the bottom left of the main content area, there is a button labeled 'Add Task: +'.

1050_Task_List_View_No_Task_Selected

Event Name's Tasks

Task Name	Description	Due By
Vendor Available	Ensure that vendors are available	2/15/2022
Backup Venue	Talk to Samantha about the venue	2/22/2022
Inventory	Take inventory of cleaning supplies	2/28/2022

* Font color is based off of Task Priority. Red = High, Orange = Medium, Blue = Low

Add Task: [+](#)

Click on a task to view a list of its volunteers

Name	Date Assigned	Role
------	---------------	------

1050_Task_List_View_Task_Selected

The screenshot shows a software interface for managing event tasks. On the left, a vertical sidebar lists various event management categories: Event Details, Tasks, Itinerary, Budget, Advertising, After Event Report, Files, and Invitations. The 'Tasks' category is currently selected, highlighted in blue. The main content area is titled 'Event Name's Tasks' and displays a table of tasks. The table has columns for Task Name, Description, and Due By. The tasks listed are:

Task Name	Description	Due By
Vendor Available	Ensure that vendors are available	2/15/2022
Backup Venue	Talk to Samantha about the venue	2/22/2022
Inventory	Take inventory of cleaning supplies	2/28/2022

A note below the table states: * Font color is based off of Task Priority. Red = High, Orange = Medium, Blue = Low.

Below the table is a section titled 'Volunteers Assigned to Mop:' with a table showing the assigned volunteers:

Name	Date Assigned	Role
Joanne Smith	01/30/22	Event Planner

1051_Task_List_Edit

The screenshot shows a web-based application for managing event tasks. The interface is divided into several sections:

- Header:** A blue navigation bar with icons for Home, Refresh, Create Event, My Events (highlighted in yellow), Suppliers, Volunteers, About, Administration, and User settings.
- Left Sidebar:** A vertical sidebar with blue buttons labeled: Event Details, Tasks (selected), Itinerary, Budget, Advertising, After Event Report, and Files.
- Form Area:** The main content area is titled "Event Name Tasks". It contains the following fields:
 - Task Name:
 - Task Description:
 - Due By: (with a calendar icon)
 - Assign:
 - Priority:
- Buttons:** At the bottom of the form are three buttons: Save (blue), Cancel (gray), and Delete (gray).

1051_Task_List_Delete

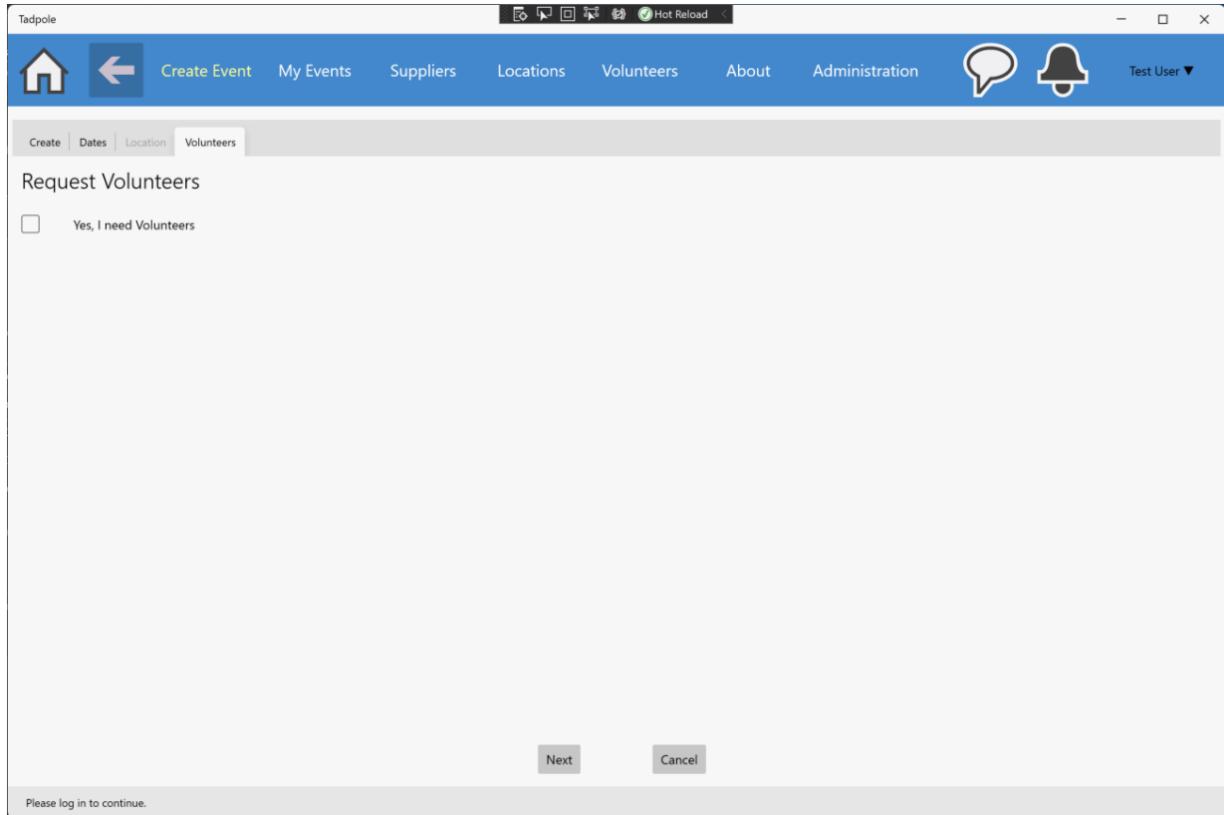
The screenshot shows a web-based event management application. The top navigation bar includes links for 'Create Event', 'My Events' (which is highlighted in yellow), 'Suppliers', 'Volunteers', 'About', 'Administration', and user account information. The main left sidebar lists 'Event Details', 'Tasks' (which is selected and highlighted in blue), 'Itinerary', 'Budget', 'Advertising', 'After Event Report', 'Files', and 'Invitations'. The main content area is titled 'Event Name's Task' and displays a task list. A modal dialog box titled 'Delete?' is open, asking 'Are you sure you would like to remove this listing?'. It contains 'Delete' and 'Cancel' buttons. To the right of the modal, there are fields for 'Priority' (set to 'Medium') and 'Assign' (set to 'Me'). Below the modal, there are 'Edit', 'Save', and 'Delete' buttons. At the bottom of the main content area, there is an 'Add Task:' field with a '+' button.

4.3.1.15 View Location

View Location Details

The screenshot shows the Tadpole application interface. The top navigation bar includes links for 'Create Event', 'My Events', 'Suppliers', 'Volunteers' (highlighted in yellow), 'About', 'Administration', and a user profile for 'Tess Data'. The left sidebar contains links for 'Site Areas', 'Site Schedule', 'Site Map', 'Site Parking', 'Site Entrances', and 'Site Supplies'. The main content area displays the 'About [Location Name]' page for a location with a 5-star rating. It includes sections for 'Contact Info' (with placeholder text for phone number and email address), 'Address' (with placeholder text for address line 1 and 2), and 'Pricing' (with placeholder text for price and address). Below these sections is a large, empty rectangular area with a '359 x 219' dimension label. Navigation buttons 'Back' and 'Next' are positioned on either side of this area. The 'Reviews' section shows a 5-star rating and two review entries, each with a user icon, a date, and a truncated text preview. A 'See More Reviews' link is also present. At the bottom are 'Save', 'Cancel', and 'Delete' buttons.

4.3.1.16 Volunteer Need



The screenshot shows a software application window titled 'Tadpole'. The top navigation bar includes a home icon, a back arrow, 'Create Event', 'My Events', 'Suppliers', 'Locations', 'Volunteers', 'About', 'Administration', a speech bubble icon, a bell icon, and a 'Test User ▾' dropdown. Below the navigation is a secondary menu with 'Create', 'Dates', 'Location', and 'Volunteers' buttons, where 'Volunteers' is highlighted. The main content area is titled 'Request Volunteers' and contains a single line of text: 'Yes, I need Volunteers' with an associated checkbox. At the bottom are 'Next' and 'Cancel' buttons, and a note: 'Please log in to continue.'

Tadpole

Hot Reload

Create Event My Events Suppliers Locations Volunteers About Administration Test User ▾

Create Dates Location Volunteers

Request Volunteers

Yes, I need Volunteers Number: 0

Request Volunteers

Next Cancel

Please log in to continue.

Tadpole

Hot Reload

Create Event My Events Suppliers Locations Volunteers About Administration Test User ▾

Event Details

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

Update Task

Tests

Task Name: General Help

Description: Help out with the event as needed on the day of.

Assign:

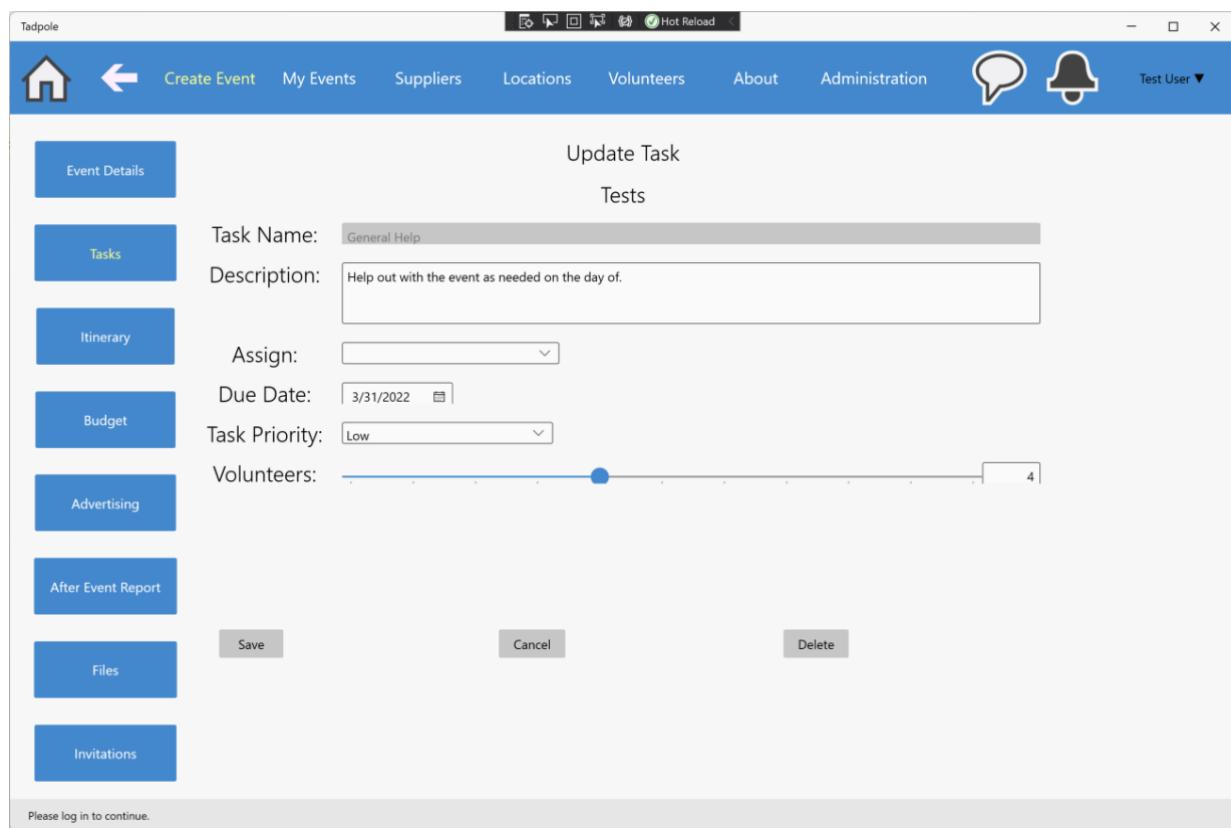
Due Date: 3/31/2022

Task Priority: Low

Volunteers: 4

Save Cancel Delete

Please log in to continue.



Tadpole

Hot Reload

Test User ▾

← Create Event My Events Suppliers Locations Volunteers About Administration

Event Details

Tasks

Itinerary

Budget

Advertising

After Event Report

Files

Invitations

Task Name: Tests

Description:

Assign:

Due Date: Select a date

Task Priority:

Volunteers: 0

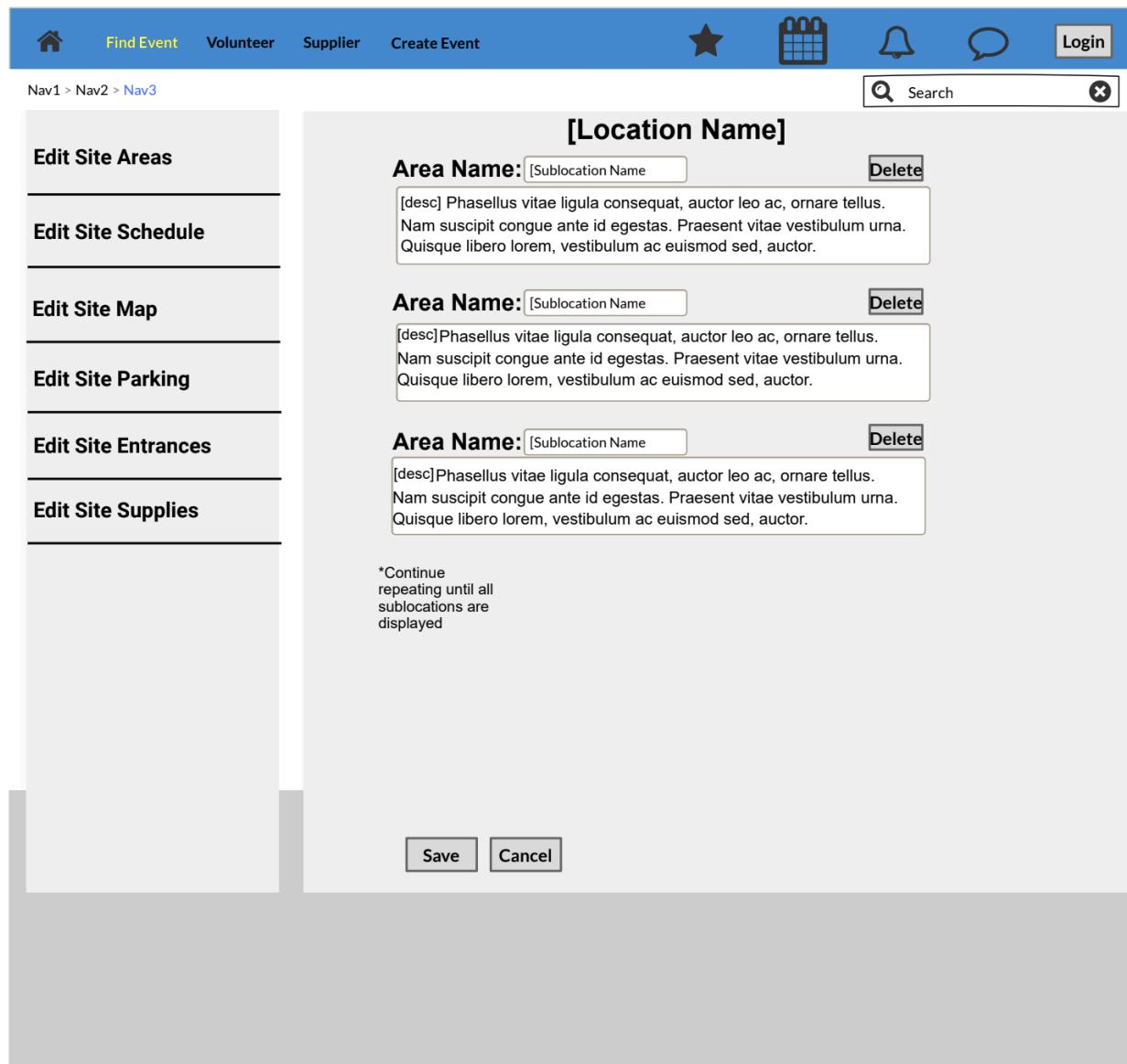
Save Cancel

Please log in to continue.

4.3.2 Supplier Desktop UI Drawings

4.3.2.1 Sublocations (3001, 3005, 3014, 3021, 3037, 3038, 3039, 3040, 3041)

Edit Sublocations



The image shows a screenshot of a web-based application interface for managing sublocations. The top navigation bar is blue and includes a home icon, a 'Find Event' button, 'Volunteer', 'Supplier', and 'Create Event' links, along with icons for a star, calendar, bell, and speech bubble, and a 'Login' button. Below the navigation is a breadcrumb trail: 'Nav1 > Nav2 > Nav3'. The main content area has a light gray background. On the left, a vertical sidebar contains links: 'Edit Site Areas', 'Edit Site Schedule', 'Edit Site Map', 'Edit Site Parking', 'Edit Site Entrances', and 'Edit Site Supplies'. The right side is titled '[Location Name]' and contains three identical data entry boxes for sublocations. Each box has an 'Area Name:' label with a text input field containing '[Sublocation Name]', a 'Delete' button, and a text area with placeholder text: '[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.' At the bottom of the right section is a note: '*Continue repeating until all sublocations are displayed'. At the very bottom are 'Save' and 'Cancel' buttons.

Nav1 > Nav2 > Nav3

Edit Site Areas

Edit Site Schedule

Edit Site Map

Edit Site Parking

Edit Site Entrances

Edit Site Supplies

[Location Name]

Area Name: [Sublocation Name] **Delete**

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Area Name: [Sublocation Name] **Delete**

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Area Name: [Sublocation Name] **Delete**

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

*Continue repeating until all sublocations are displayed

Save **Cancel**

Nav1 > Nav2 > Nav3

Supplier

Location Name

Sublocation Name

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Sublocation Name

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Sublocation Name

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

*The site supplies option is only available to users logged into an event planning role.

*Continue repeating until all sublocations are displayed

Edit

*If the edit button is not available, use the star. Otherwise, leave out the star.

The screenshot shows a website interface for managing event locations and sublocations. The top navigation bar includes links for Home, Find Event, Volunteer, Supplier (which is the active page), and Create Event. It also features icons for a star, calendar, bell, and speech bubble, along with a Login button. The main content area has a breadcrumb navigation path: Nav1 > Nav2 > Nav3. On the left, a sidebar lists Site Areas, Site Schedule, Site Map, Site Parking, Site Entrances, and Site Supplies. A note indicates that the Site Supplies option is only available to users logged into an event planning role. The main content area displays location and sublocation details in a card-based format. Each card includes a title, a description, and an 'Edit' button with a star icon. A note on the right explains that if the edit button is not available, the star should be used. The cards are as follows:

- [Location Name]**
[Sublocation Name]
[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.
- [Sublocation Name]**
[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.
- [Sublocation Name]**
[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

*If the edit button is not available, use the star. Otherwise, leave out the star.

4.3.2.2 Supplier (3026, 3028, 3030, 3042, 3043, 3044)

Browse Suppliers

Tadpole

Create Event My Events **Suppliers** Volunteers About Administration

Filter By: Search

Browse Suppliers

Image	Supplier	Category	Reviews
Image	Supplier1	★ Catering	★★★★★
Image	Supplier2	★ Category2	★★★★★
Image	Supplier3	★ Category3	★★★★★
Image	Supplier4	★ Category4	★★★★★
Image	Supplier5	★ Category5	★★★★★
Image	Supplier6	★ Category6	★★★★★

Apply

View Specific Supplier

Tadpole

Create Event My Events Suppliers Volunteers About Administration

Search Find

Supplier1 ★
Catering Service

My Bio! Bio bio bio bio

Secerit fontes liquidum locoque pronaque?

Ilas semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit indicis aesta fixo, Ilas semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit indicis aesta fixo, Ilas semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit indicis aesta fixo, Ilas semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit indicis aesta fixo, Ilas semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit indicis aesta fixo, Ilas semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit indicis aesta fixo,

Ilas semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit indicis aesta fixo, Ilas semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit indicis aesta fixo, Ilas semine campoque declivia oppida corpora nam inter fuit discordia tellus solidumque iunctarum erat: quae terrenae ubi rerum recessit indicis aesta fixo,

Gallery

[View More...](#)

Image	Menu Item	Description	Price
	menuitem1		\$49.99
	menuitem2		\$49.99
	menuitem3		\$49.99

Reviews: ★★★★★

Review1:
Lorem Ipsum

Review2:
Lorem Ipsum

[View More...](#)

View Specific Supplier Scrolled Down

Tadpole

Create Event My Events **Suppliers** Volunteers About Administration

Search Find

Availability

Monday - Friday
8:00 A.M. - 7:00 P.M.

Saturday
7:00 A.M. - 5:00 P.M.

[View Schedule...](#)

menulitem5 \$49.99

menulitem6 \$49.99

menulitem7 \$49.99

Place Order

Place Order

Tadpole

Home Create Event My Events Suppliers Volunteers About Administration Tess Data

Search Find

Availability

Monday - Friday
8:00 A.M. - 7:00 P.M.

Saturday
7:00 A.M. - 5:00 P.M.

[View Schedule...](#)

Place Order

Event: MyEvents

Dates: 01/02/2021 — 01/05/2021

Price Package: MyCaterer's Package1

Send Request Cancel

menuItem5

menuItem6

menuItem7

View Schedule

Tadpole

Create Event My Events **Suppliers** Volunteers About Administration

Search Find

Schedule

AUG - 2016

S	M	T	W	T	F	Sa
31	1	2	3	4	5	6
7	8 Event1	9 Event1	10 Event1	11 Event1	12 Event1	13 Event1
14 Event1	15	16	17	18	19	20 Event2
21 Event2	22	23	24	25	26 Event3	27
28 4	29	30	31	1	2	3
	5	6	7	8	9	10

View Ratings

View Gallery

Tadpole

Home Create Event My Events **Suppliers** Volunteers About Administration Tess Data

← Find

My Bio! Bio bio bio bio
Secrevit fontes liquidar
Illa semine campoque
Illa semine campoque

Gallery



← →

More...

Image	Menu
menu	
menu	
menu	

Category Filter

Tadpole

Create Event My Events Suppliers Volunteers About Administration

Filter By:

Image	Supplier 1	Category 1	Reviews 1	
	Supplier1		Catering	
	Supplier2		Category2	
	Supplier3		Category3	
	Supplier4		Category4	
	Supplier5		Category5	
	Supplier6		Category6	

Distance Filter

The screenshot shows the Tadpole application interface. At the top, there is a navigation bar with links for 'Create Event', 'My Events', 'Suppliers' (which is the active page, highlighted in yellow), 'Volunteers', 'About', and 'Administration'. On the right side of the header, there are icons for a speech bubble, a bell, and a user profile named 'Tess Data'. Below the header, there is a search bar with a 'Find' button and a 'Browse Suppliers' section. On the left, there is a sidebar titled 'Filter By:' with dropdown menus for 'Category', 'Distance' (set to 'Mile Radius'), 'Rating' (set to '5'), 'Zip Code' (set to 'e.g. 52206'), and an 'Apply' button. A 'Search' button is also in this sidebar. The main content area displays a table titled 'Browse Suppliers' with the following data:

Image	Supplier	Category	Reviews
Image	Supplier1	Catering	★★★★★
Image	Supplier2	Category2	★★★★★
Image	Supplier3	Category3	★★★★★☆
Image	Supplier4	Category4	★★★★☆☆
Image	Supplier5	Category5	★★★★☆☆
Image	Supplier6	Category6	★★★★☆☆

Rating Filter

Tadpole

Create Event My Events **Suppliers** Volunteers About Administration

Search Tess Data

Filter By:

Category

Distance

Rating

Available

1 Star
2 Star
3 Star
4 Star
5 Star

Search

Browse Suppliers

Image	Supplier	Category	Reviews
Image	Supplier1	★ Catering	★★★★★
Image	Supplier2	★ Category2	★★★★★
Image	Supplier3	★ Category3	★★★★★☆
Image	Supplier4	★ Category4	★★★★★☆
Image	Supplier5	★ Category5	★★★★★☆
Image	Supplier6	★ Category6	★★★★★☆

Availability Filter

Tadpole

Create Event My Events Suppliers Volunteers About Administration

Search Find Tess Data

Filter By:

Category

Distance

Rating

Availability

Available From:

1/02/2021 — 1/03/2021

Apply

Search

Browse Suppliers

Image	Supplier	Category	Reviews
Image	Supplier1	★ Catering	★★★★★
Image	Supplier2	★ Category2	★★★★★
Image	Supplier3	★ Category3	★★★★★☆
Image	Supplier4	★ Category4	★★★★☆☆
Image	Supplier5	★ Category5	★★★★☆☆
Image	Supplier6	★ Category6	★★★★☆☆

Retrieve Single Listing View

The screenshot shows a user interface for a supplier listing. At the top, there is a navigation bar with links for 'Find Event', 'Volunteer', 'Supplier' (which is highlighted in yellow), and 'Create Event'. To the right of the navigation are icons for a star, a calendar, a bell, a speech bubble, and a 'Login' button. Below the navigation, a breadcrumb trail shows 'Nav1 > Nav2 > Nav3'. On the left, there is a sidebar with sections for '[Supplier]s Schedule' and '[Supplier]s Pricing'. The main content area is titled '[Supplier Name]' and includes a 'Type' field with the value '[Vendor/Supplier]'. Below this is a 'About [Supplier Name]' section containing a bio: 'Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.' A note on the right says: '*If the edit button is not available, use the star. Otherwise, leave out the star.' There is also a 'Search' bar with a magnifying glass icon and a 'Tags: [tags]' field. The 'Contact Info' section includes fields for '[phone number]' and '[email address]'. Below these are three placeholder images for 'Photos' with the dimensions '136 x 100'. The 'Reviews' section shows a rating of '1.5 stars' with five star icons. Two review entries are displayed, each with a user icon, a star rating, and a short description: 'Vivamus mollis lorem vestibulum vehicula hendrit. Mauris eu aliquet ligula. Donec mattis ullamcorper tortor vitae.' and 'Vivamus mollis lorem vestibulum vehicula hendrit. Mauris eu aliquet ligula. Donec mattis ullamcorper tortor vitae.'. A 'See More Reviews' link is present. At the bottom, there are 'Request Reservation' and 'Send Message' buttons. A note below the buttons states: '*Request Reservation only shows up for event planners'.

Retrieve Single Listing View - Request Reservation

Nav1 > Nav2 > Nav3

[Supplier]’s Schedule

[Supplier]’s Pricing

[Supplier Name]

Type: [Vendor/Supplier]

About [Supplier Name]

[bio] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Tags: [tags]

Contact Info: [phone number] [email address]

Photos

136 x 100

136 x 100

136 x 100

Reviews

☆☆☆☆☆ [rating, example 1.5 stars]

Event1 [review] Vivamus mollis lorem vestibulum vehicula hendrit. Mauris eu a

Event2 [review] Vivamus mollis lorem vestibulum vehicula hendrit. Mauris eu a

See More Reviews

Event3

Event4

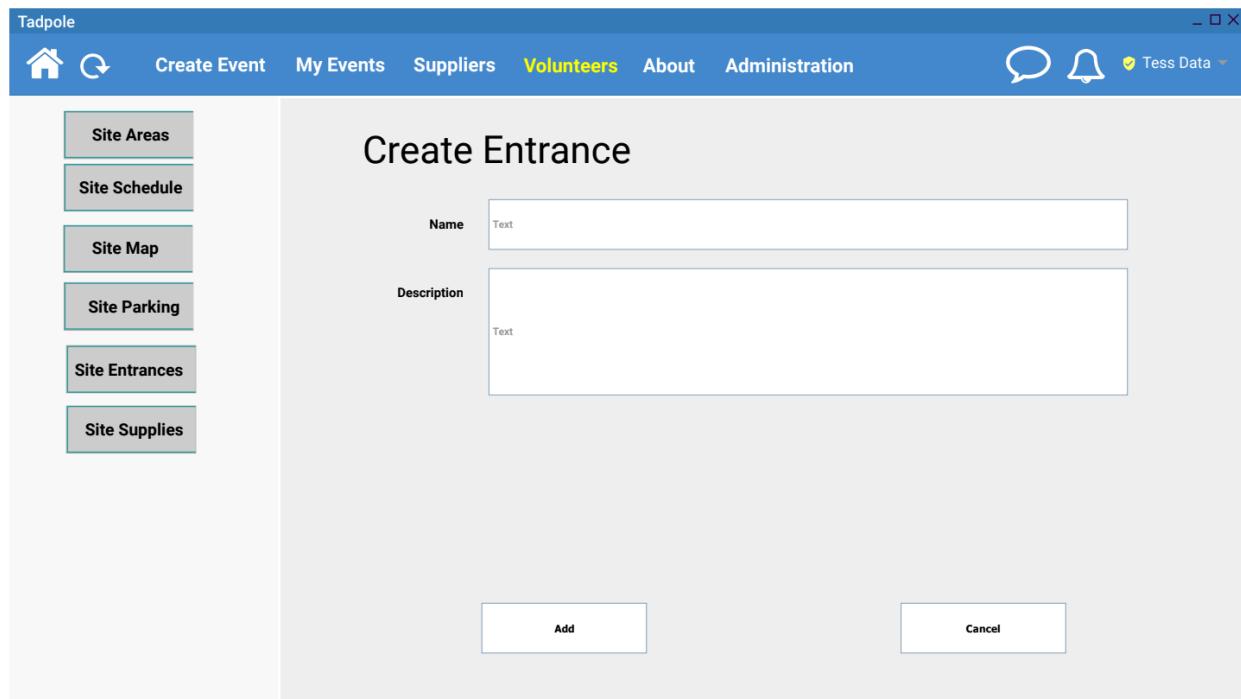
Request Reservation

*Request Reservation only shows up for event planners

*If the edit button is not available, use the star. Otherwise, leave out the star.

4.3.2.3 Entrances (3007, 3016, 3022, 3033)

Create Entrance



The screenshot shows the Tadpole software interface with the following details:

- Header:** Tadpole, Create Event, My Events, Suppliers, **Volunteers** (highlighted in yellow), About, Administration, a message icon, a bell icon, and Tess Data.
- Left Sidebar:** Site Areas, Site Schedule, Site Map, Site Parking, **Site Entrances** (highlighted in grey), and Site Supplies.
- Form Title:** Create Entrance
- Form Fields:**
 - Name: Text input field
 - Description: Text input field
- Buttons:** Add (left) and Cancel (right) at the bottom of the form.

— □ ×

Home  Create Event My Events Suppliers **Volunteers** About Administration   Tess Data

Entrance Name	Description
Front Door	Door for Attendees
Back Door	Door for those a bit more adventurous

Site Areas
Site Schedule
Site Map
Site Parking
Site Entrances
Site Supplies

Create Entrance

4.3.2.4 Parking (3002, 3012, 3018)

EditParking

Edit Site Areas

Edit Site Schedule

Edit Site Map

Edit Site Parking

Edit Site Entrances

Edit Site Supplies

[Location Name]

Parking Lot Name: [Parking Lot Name]

Add Image

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Parking Lot Name: [Parking Lot Name]

[desc] Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

4.3.2.5 Location (3003, 3029, 3031, 3032)

View Location Details

Tadpole - □ ×

[Create Event](#) [My Events](#) [Suppliers](#) [Volunteers](#) [About](#) [Administration](#) [Tess Data](#) ▾

About [Location Name]

Donec Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Contact Info: [phone number] [email address]
Address [address line 1]
[address line 2]

Pricing
Phasellus vitae ligula consequat, auctor leo ac, ornare tellus. Nam suscipit congue ante id egestas. Praesent vitae vestibulum urna. Quisque libero lorem, vestibulum ac euismod sed, auctor.

Back

Next

Reviews

[review] Vivamus mollis lorem vestibulum vehicula hendrit. Mauris eu aliquet ligula. Donec mattis ullamcorper tortor vitae.

[review] Vivamus mollis lorem vestibulum vehicula hendrit. Mauris eu aliquet ligula. Donec mattis ullamcorper tortor vitae.

[See More Reviews](#)

View Location Details

Tadpole

Create Event My Events Suppliers **Volunteers** About Administration

Chat Bell Tess Data

Site Areas

Site Schedule

Site Map

Site Parking

Site Entrances

Site Supplies

[Location Name]

S	M	T	W	T	F	Sa
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

[Date Selected]

Event 2	Date	Time Start	Time End
Event 2	10-23-2022	10:30 am	9:00 pm
Event 2	10-23-2022	10:30 am	9:00 pm
Event 2	10-23-2022	10:30 am	9:00 pm
Event 2	10-23-2022	10:30 am	9:00 pm
Event 2	10-23-2022	10:30 am	9:00 pm
Event 2	10-23-2022	10:30 am	9:00 pm

View Locations

The screenshot shows a web application interface with a blue header bar. The header contains the text "Tadpole" on the left, and on the right, there are icons for a speech bubble, a bell, and a user profile with the name "Tess Data". Below the header is a navigation menu with links: "Create Event", "My Events", "Suppliers", "Volunteers" (which is highlighted in yellow), "About", and "Administration".

On the left side of the main content area, there is a vertical sidebar with the following buttons:

- Site Areas
- Site Schedule
- Site Map
- Site Parking
- Site Entrances
- Site Supplies

The main content area is titled "Locations" and contains a table with the following data:

Name	Description	Address	City	State
Location 1	Location 1 Description	Wherever fine Location 1's are sold	An City	Solid
Location 2	Location 2 Description	Same as Location 1 but 2 instead	Owl City	Liquid
Location 3	Location 3 Description	See above but 3	We Built this City	Gas

5 ER Diagram

5.1 *ER Diagram Summary*

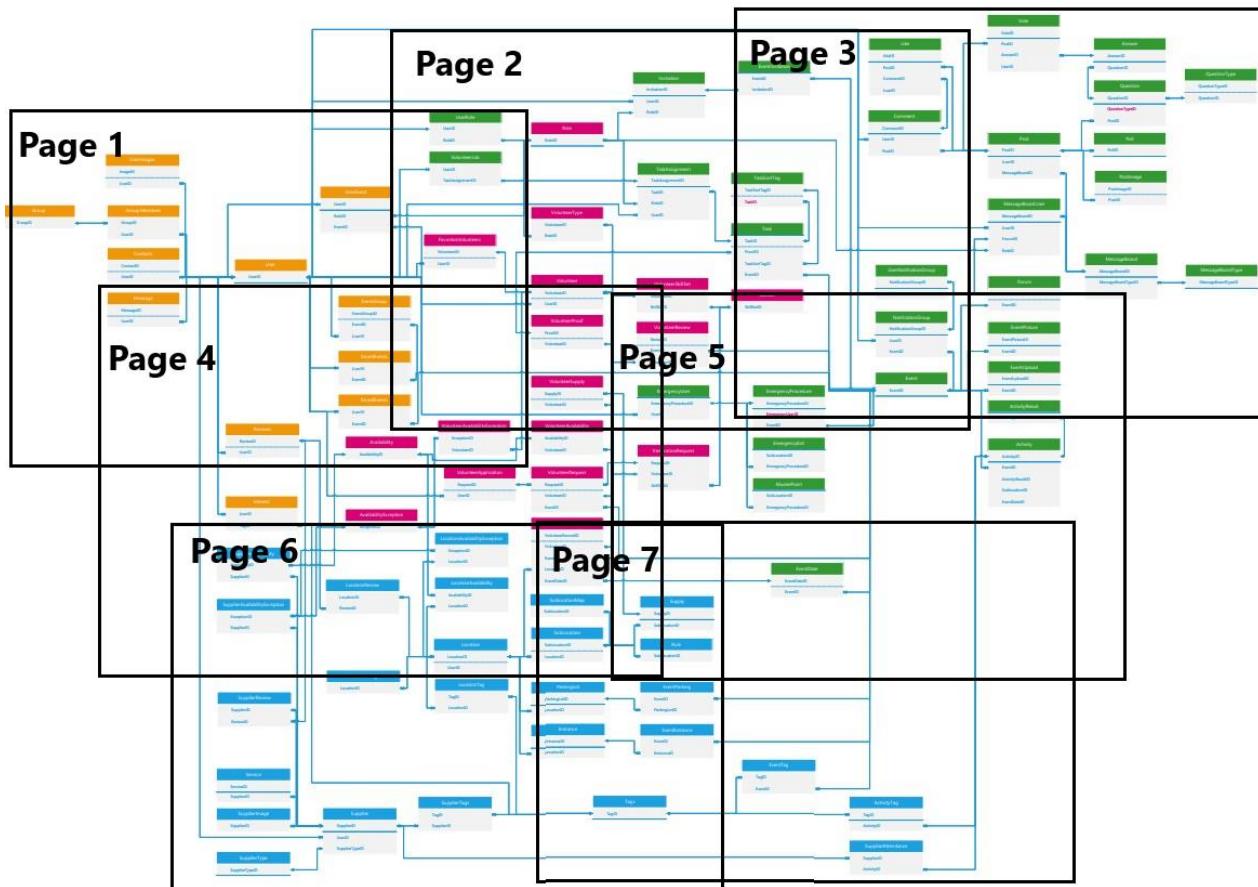
This ER (Entity Relationship) Diagram is meant to give a graphical representation of the way the tables in our database are related. Each box in the diagram represents a table of information, and the lines between them represent a shared field which binds them together, making it easy to jump between tables looking for related information. The only fields listed are the primary keys of each table, which are used to uniquely identify each record in the table, and the foreign keys, which serve as the link between related tables.

The tables are color-coated by our four major areas of process that they belong to. Events are in green, volunteers in pink, suppliers in blue, and general attendee users in orange. This helps visualize the major areas in the app and see the way each of them connects with each other.

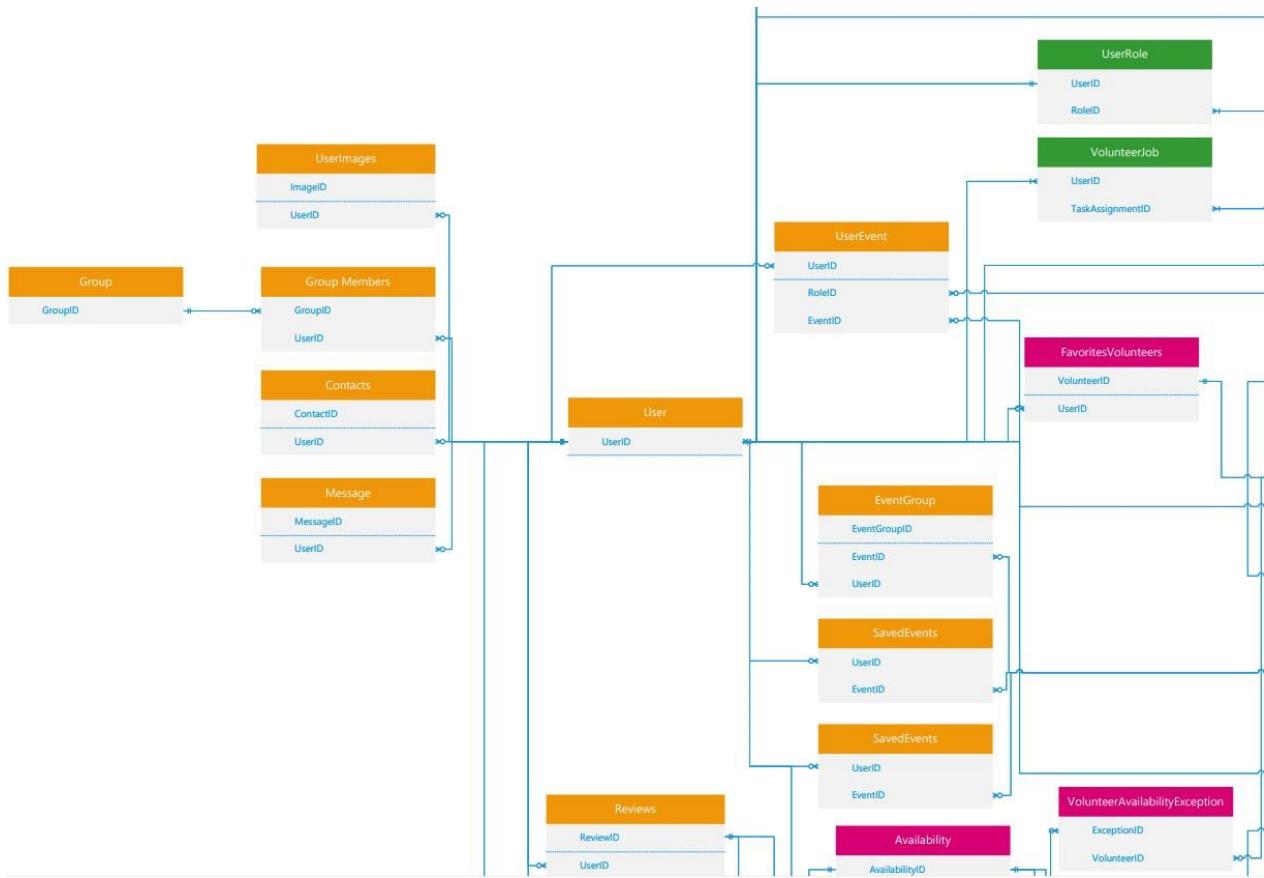
5.2 *ER Diagram Overview*



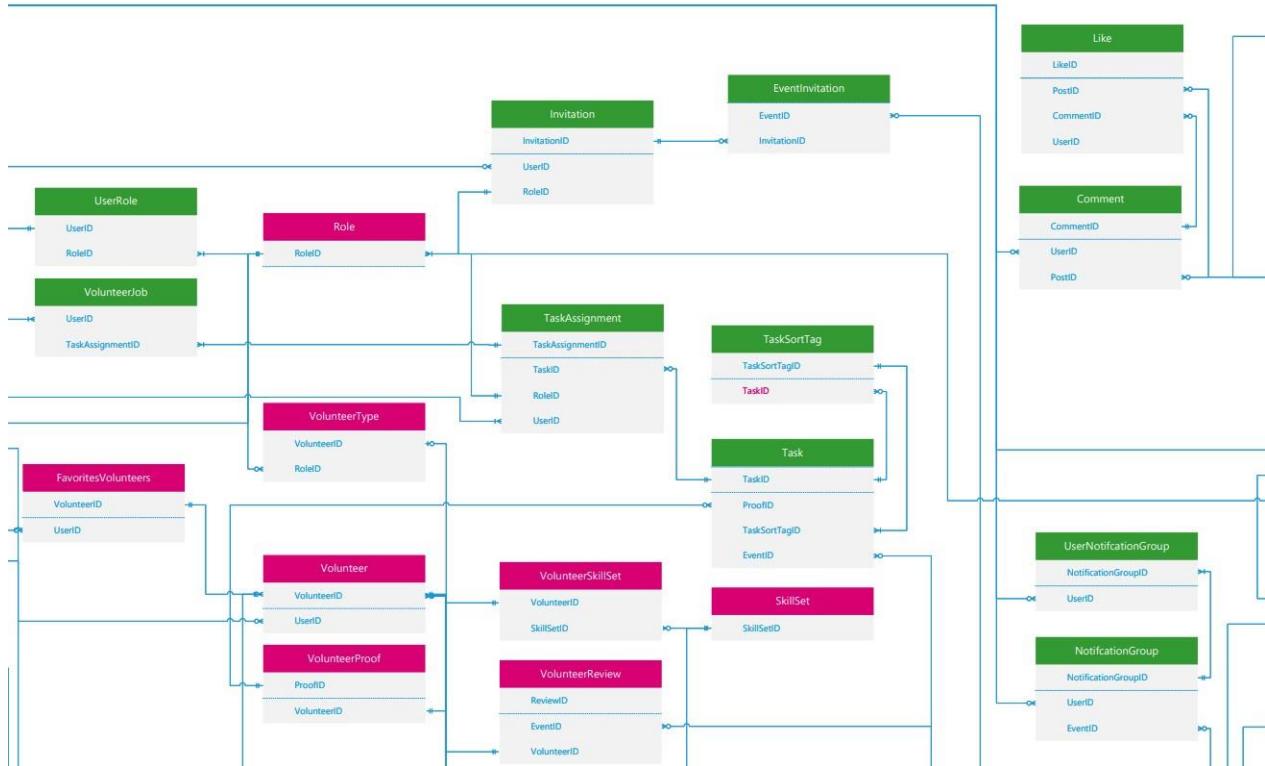
5.3 *ER Diagram Map*



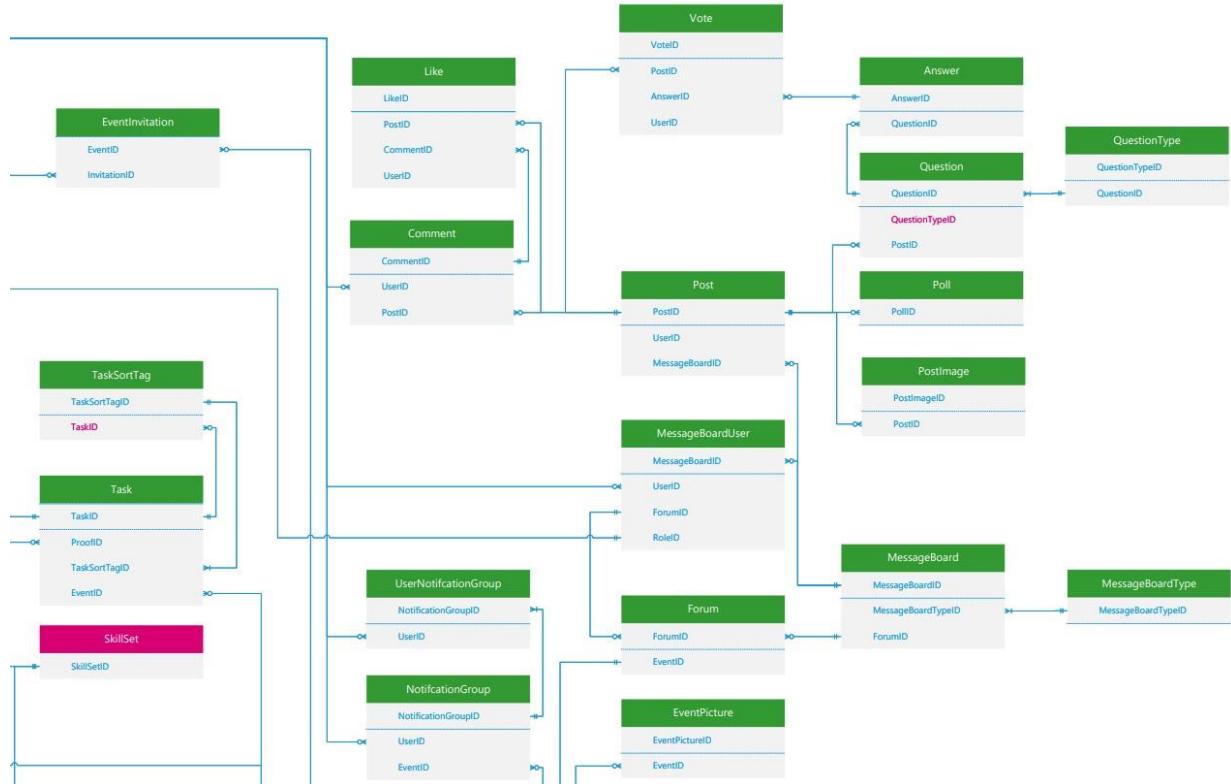
5.4 Page 1



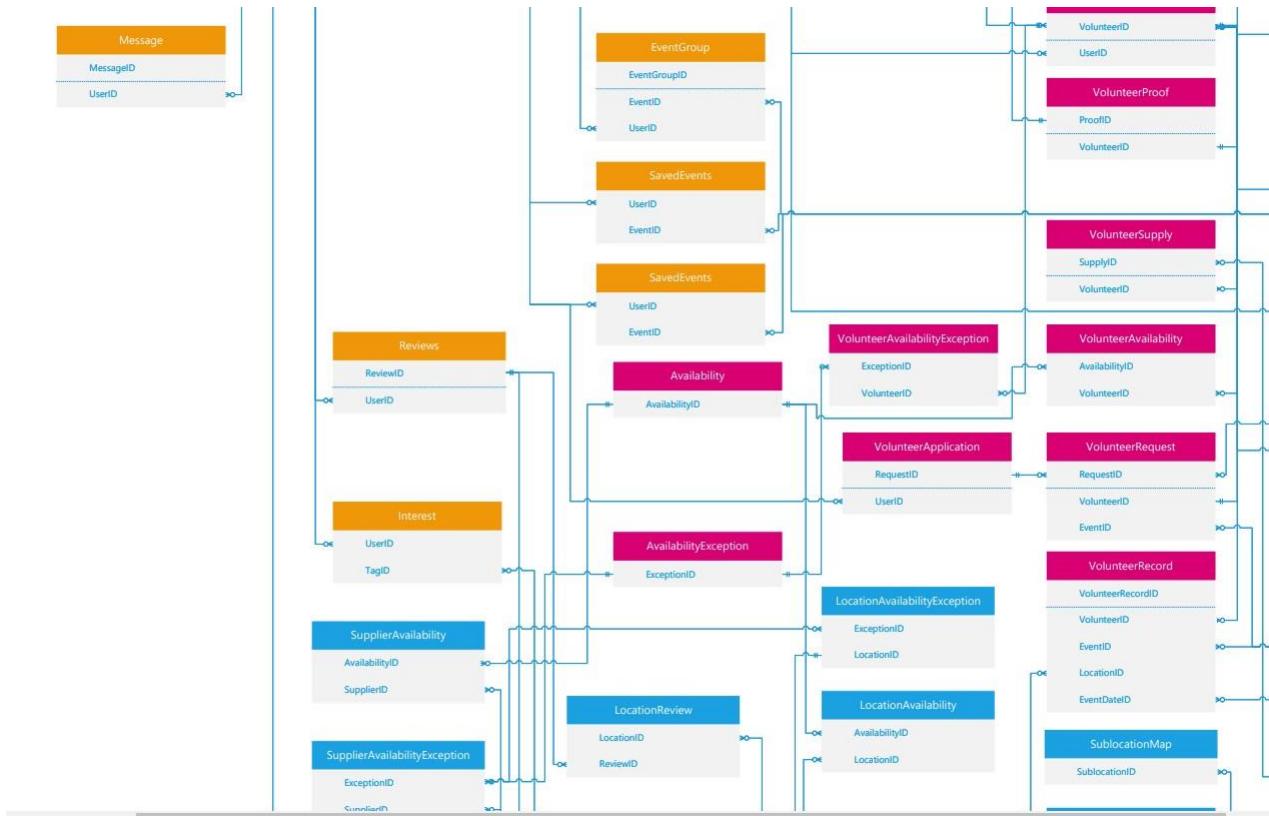
5.5 Page 2



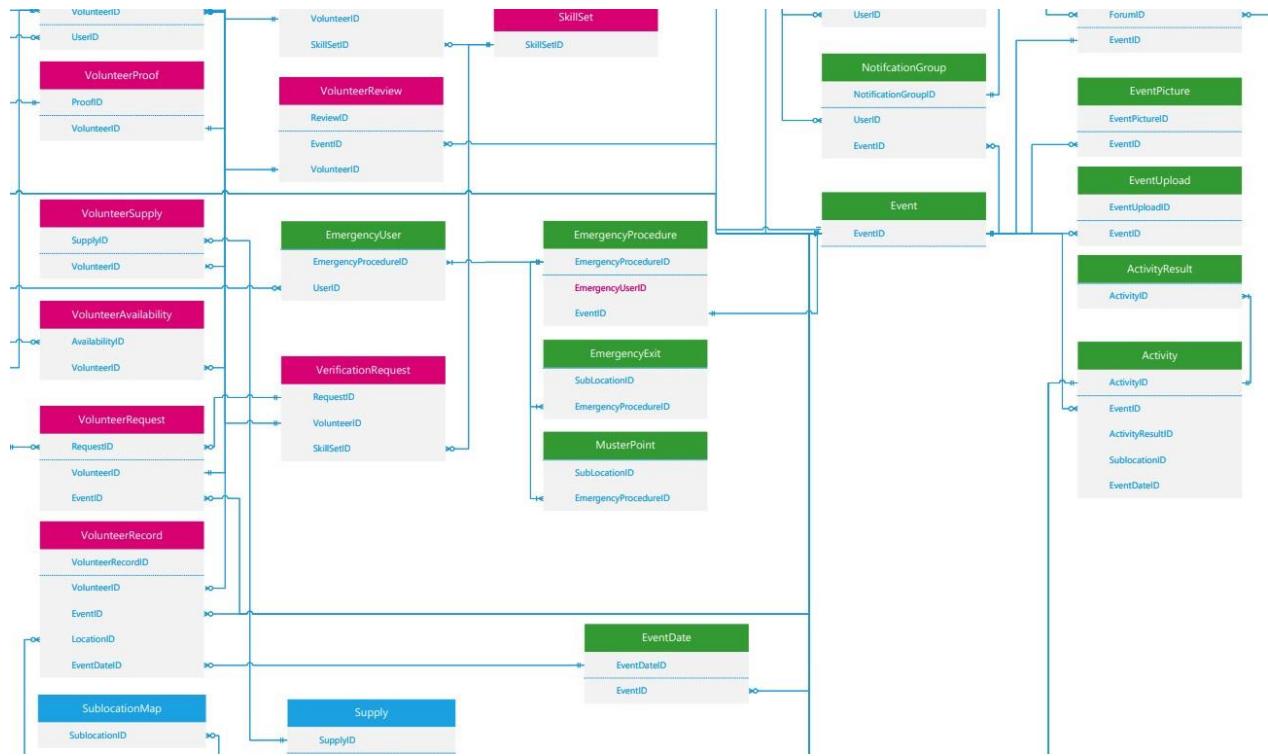
5.6 Page 3



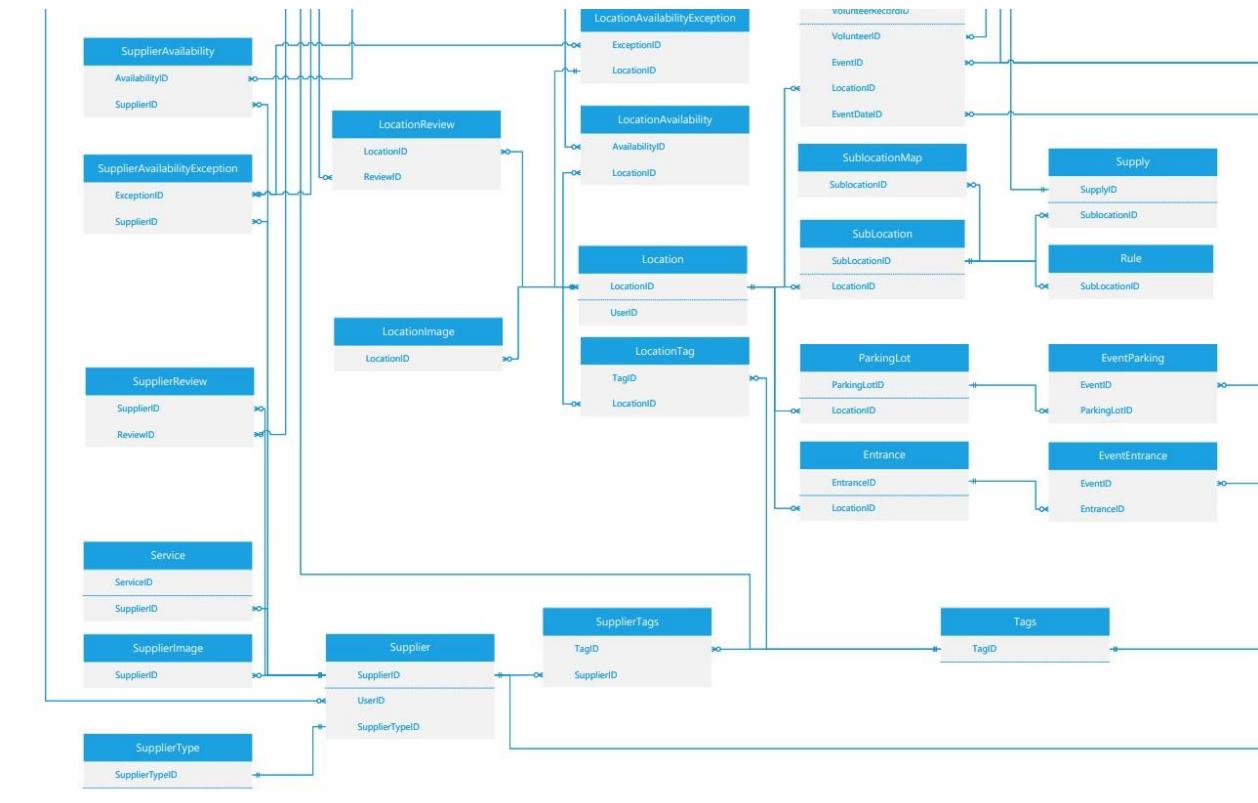
5.7 Page 4



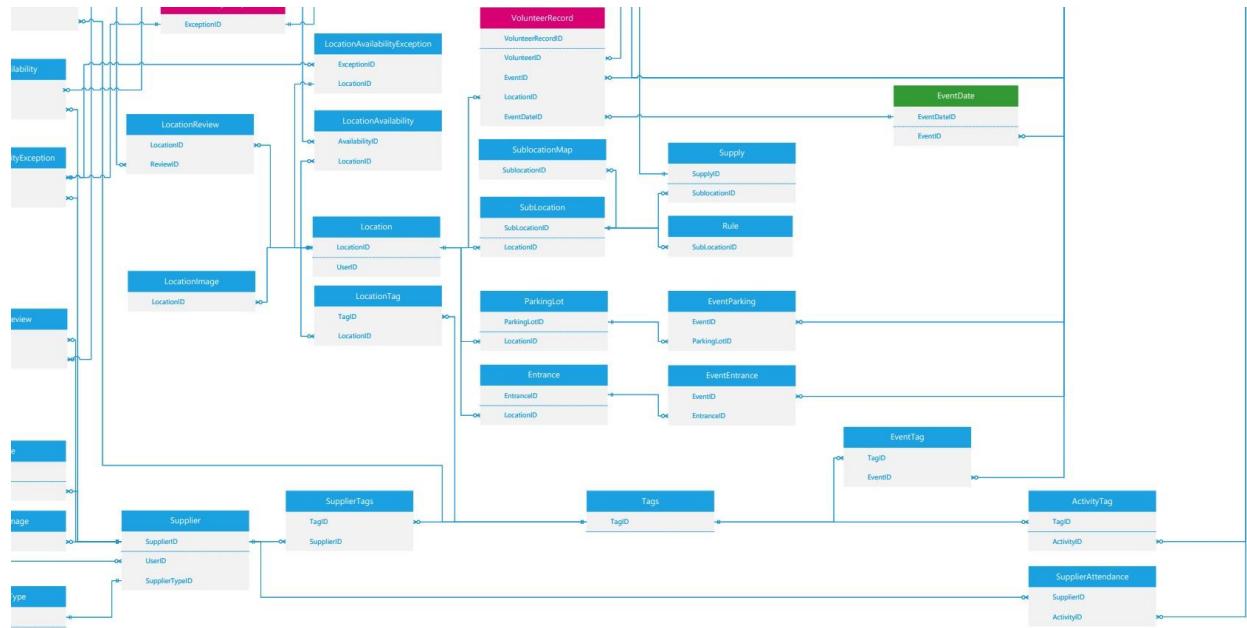
5.8 Page 5



5.9 Page 6



5.10 Page 7



6 Data Dictionary

6.1 *Data Dictionary Summary*

The data dictionary's role is to fully define every table and field in the database from name to data type to relation. Once again, the four major areas of the application are color-coated on the table similar to the way they are organized in the ER diagram. Events in orange, volunteers in green, suppliers in blue, and users in pink.

Each new table in the dictionary is ordered so that any fields it may be dependent on should already be defined higher up in the dictionary. Each field has a name that should make it clear what its purpose is, with comments for clarity where they are needed. This makes it a great way to take a look into the data that will be stored to help these processes work.

6.2 Data Dictionary Contents

	DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY		REFERENCES	COMMENTS
Event															
EventID	int			y	100 000	1	n	y	y	y					auto number
LocationID	int						y				y			Location(LocationID)	
EventName	nvarchar	50	n				n								
EventDescription	nvarchar	10 00	n				n								
DateCreated	DateTi me		DateTi meNow (())				n								
TotalBudget	money	0					n								
Active	bit	1					n								
NotificationGrou p															
NotificationGrou pID	int			y	100 000	1	n	y	y	y					auto number
DateTimeCreated	DateTi me		DateTi meNow (())				n								
UserID	int	-	-	-	-	-	-	-	-	-	y	-	-	-	-
EventID	int						n				y		Event(EventID)		
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
UserNotification Group															Join table between user and notificatio n
NotificationGrou pID	int									y			Notification(Notification ID)		Composit e Key
UserID	int									y			User(UserID)		Composit e Key
EventPicture															
EventPictureID	int			y	100 000	1	n	y	y	y					auto number
EventID	int						n			y		Event(EventID)			
EventPictureNam e	varchar	25					n								
DateTimeCreated	DateTi me		DateTi meNow (())				n								
Title	nvarchar	50					n								
Caption	nvarchar	50					n								
Description	nvarchar	25 0					y								
isMainDisplayIma ge	bit	0					n								
Active	bit	1													

	datatype	length	default	identity	start	increment	nulls	index	unique	pk	fk	integrity		references	comments
EventUpload															
EventUploadID	int			y	100 000	1	n	y	y	y					auto number
EventID	int										y		Event(EventID)		
EventFileName	varchar	25							y						Other way to go is varbinary (MAX)
DateTimeCreated	DateTi me		DateTi meNow ()				n								
Title	nvarcha r	50					n								
Description	nvarcha r	25 0					y								
Active	bit		1												
Activity															
ActivityID	int			y	100 000	1	n	y	y	y					auto number
ActivityName	nvarcha r	50					n								
ActivityDescriptio n	nvarcha r	25 0					y								
PublicActivity	bit		1				n								
StartTime	time						n								
EndTime	time						n								
ActivityImageNa me	varchar	25					y								A picture from the result(or end) of the activity
SubLocationID	int						y	n	n		CASCADE _UPDATE	SubLocation(SubLocati onID)			
EventDateID	Date						n			y	CASCADE _UPDATE	EventDate(EventDateID)			
EventID	int						n			y		Event(EventID)			
UserActivity															
UserActivityID	int						n		y						auto number
UserID	int						n			y		Users(userID)			
RoleID	nvarcha r	50					n			y		Roles(RoleID)			
ActivityID	int						n			y		Activity(ActivityID)			

	DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY	REFERENCES	COMMENTS
EmergencyProcedure														
EmergencyProcedureID	int			y	100000	1	n	y	y	y				auto number
Message	nvarchar	250	"Call 911 in an emergency."											
EmergencyUserID	int										y		EmergencyUser(EmergencyUserID)	Needs name, number, and role from user
EventID	int							y		y			Event(EventID)	
EmergencyUser														
EmergencyProcedureID	int						n		y	y			EmergencyProcedure(EmergencyProcedureID)	
UserID	int						n		y	y			User(UserID)	
MusterPoint														
EmergencyMusterPointDescription	nvarchar	50												
SubLocationID	int						n			y	y		SubLocation(SubLocationID)	
EmergencyProcedureID	int								y	y	y		EmergencyProcedure(EmergencyProcedureID)	Only one muster point (could be changed)
EmergencyExit														
EmergencyExitsDescription	nvarchar	50												
SubLocationID	int						n		y	y	y		SubLocation(SubLocationID)	Composite Key
EmergencyProcedureID	int								y	y	y		EmergencyProcedure(EmergencyProcedureID)	Composite Key
ActivityResult														
ActivityResultRank	int									y				Composite Key
ActivityResultName	nvarchar	50												
ActivityID	int			y	100000	1	n		y	y			Activity(ActivityID)	Composite Key
EventDate														
EventDateID	Date		-						y					Composite key between

	datatype	length	default	identity	start	increment	nulls	index	unique	pk	fk	integrity		references	comments
															date and event id, The date for start time date
EventID	int								y	y				Event(EventID)	
StartTime	Time														
EndTime	Time														
Active	bit														
Forum															
ForumID	int			y	100 000	1	n	y	y	y					auto number
															1-1 relation between forum and event
EventID	int						n	y	y					Event(EventID)	
Name	nvarchar	50					n	n	n						
Description	nvarchar	100					y	n	n						
DateCreated	DateTi me		DateTi meNow ()				n	y	n						
MessageBoard															
MessageBoardID	int			y	100 000	1	n	y	y	y					auto number
MessageBoardTy peID	nvarchar	50									y			MessageBoardType(MessageBoardTypeID)	
ForumID	int						n	y	y	y				Forum(ForumID)	
Name	nvarchar	50					n	n	n						
Description	nvarchar	100					y	n	n						
DateCreated	DateTi me		DateTi meNow ()				n	y	n						
MessageBoardTy pe															
MessageBoardTy peID	nvarchar	50					n		y	y					
Description	nvarchar	100					y								
MessageBoardUs er															
MessageBoardID	int			y	100 000	1	n	y	y	y					auto number
UserID	int						n	y	n		y			User(userID)	
ForumID	int						n	y	n		y			Forum(ForumID)	not unique =

	datatype	length	default	identity	start	increment	nulls	index	unique	pk	fk	integrity		references	comments
															one forum can have many roles connected to it
RoleID							n							Role(RoleID)	
Post															
PostID	int			y	100 000	1	n	y	y	y					auto number
Content	nvarchar	25 0					n	n	n						
CreatedDate	DateTime		DateTi meNow ()				n	y	n						default = date/time created
UserID	int						n	n	n		y			User(UserID)	User that created the post
MessageBoardID	int						n	y			y			[MessageBoard]([Mess agBoardID])	ID of message board posted to
Comment															
CommentID	int			y	100 000	1	n	y	y	y					auto number
Content	nvarchar	25 0					n	n	n						
UserID	int						n				y			User(UserID)	
CreatedDate	DateTime		DateTi meNow ()				n	y	n						default = date/time created
PostID								y			y			Post(PostID)	
Like															
LikeID	int			y	100 000	1	n	y	y	y					auto number
PostID	int	null						y	y			y		Post(PostID)	
CommentID	int	null						y				y		Comment(CommentID)	
UserID	int						n				y			User(UserID)	
Liked	bit	0					n								bool value to be toggled when user unlikes or relikes a post
Poll															
PostID	int						n	y	y	y	y			Post(PostID)	
Title	nvarchar	10 0					n								

	datatype	length	default	identity	start	increment	nulls	index	unique	pk	fk	integrity		references	comments
Description	nvarchar	100				n									
DueDate	DateTi me					n									
Active	bit	1			n										
Question															
QuestionID	int		y	100 000	1	n	y	y	y					auto number	
QuestionTypeID	int					n	y			y			QuestionType(Questio nTypeID)		
IsRankingList	bit	0				n									toggled to true if user selects ranking list choice
PostID	int					n				y			Post(PostID)		
isAnswered	bit	0				n	y								
DateAnswered	DateTi me					y	y	n							
QuestionType															
QuestionTypeID	int		y	100 000	1	n	y	y	y				auto number		
QuestionID	int					n	y			y			[Question]([QuestionID])		
TypeDescription	nvarchar	250				n	y	y							
Answer															
AnswerID	int		y	100 000	1	n	y	y	y				auto number		
Description	nvarchar	100				n									
QuestionID	int					n				y			[Question]([QuestionID])		
DateAnswered	DateTi me					y				y			[Question]([DateAnswe red])		
Vote															
VoteID	int		y	100 000	1	n	y	y	y				auto number		
Rank	set(1, 2, 3, 4, 5)					y			y					If Vote(IsRankingList) then keep track of the number that the user ranked the option	

	DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY		REFERENCES	COMMENTS
															(1=best) to (last=worst). Else its 0 for not voted and 1 for voted
UserSelection	int						y	y	n						
PostID	int						n	y	n		y			[Post]([PostID])	
AnswerID	int						n			y	y			[Answer]([AnswerID])	
UserID	int						n			y	y			[User]([UserID])	
PostImage															
PostImageID	int			y	100 000	1	n	y	y	y					auto number
ImageName	nvarchar														
PostID	int										y			Post(PostID)	
Priority															
PriorityID	int			y	1	1	n		y	y					auto number
Description	nvarchar	10					n		y						
Task															
TaskID	int			y	100 000	1	n	y	y	y					auto number
Name	nvarchar	50					n	n	n						
Description	nvarchar	25 5					y	n	n						
DueDate	DateTi me		DateTi meNow (())				y	y	n						
Priority	int						n	y	n		y			Priority(PriorityID)	
CompletionDate	DateTi me						y	y	n						
ProofID	int						y	n	y		y			VolunteerProof(ProofI D)	
isDone	bit	0					n	y	n						
EventID	int						n	y	n		y			Event(EventID)	
Active	bit	1					n	y	n						
TaskAssignment															
TaskAssignmentID	int			y	100 000	1	n	y	y	y					auto number
DateAssigned	DateTi me		DateTi meNow (())				n	y	n						
TaskID	int						n	y	n		y			Task(TaskID)	
RoleID	nvarchar	50					y	y	n		y			Role(RoleID)	
UserID	int						y	y	n		y			User(UserID)	
TaskSortTag															

	DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY		REFERENCES	COMMENTS
TaskSortTagID	int			y	100 000	1	n	y	y	y					auto number
TaskID	int						n	y	y		y			Task(TaskID)	
Name	nvarchar	25 0					n	n	n						
Invitations															
InvitationID	int		n	y	100 00	1	n	y	y	y					
EventID	int										y			Event(EventID)	
UserID	int		n				n		n		y	CASCADE _UPDATE		User(userID)	Compound Primary Key
RoleID	nvarchar	50	n				n		n		y	CASCADE _UPDATE		Role(RoleID)	Compound Primary Key
Attending	bit		0				n								0 is not going, 1 is
EventInvite															
EventID	int		n				n	y	n	y	y	CASCADE _UPDATE		Event(EventID)	Compound Primary Key
InvitationID	int		n	y	1	1	n	y	n	y	y	CASCADE _UPDATE		Invitations(InvitationID)	Compound Primary Key
VolunteerJob															
UserID	int		n				n	y	n	y	y	CASCADE _UPDATE		User(userID)	Compound Primary Key
TaskAssignmentID	int		n				n	y	y	y	y	CASCADE _UPDATE		TaskAssignment(TaskAssignmentID)	Compound Primary Key
UserRole															

	DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY		REFERENCES	COMMENTS
UserID	int		n				n	y	n	y	y	CASCADE_UPDATE	User(UserID)		Compound Primary Key
RoleID	nvarchar	50	n				n	n	n	y	y	CASCADE_UPDATE	Role(RoleID)		Compound Primary Key
VolunteerNeed															
TaskID	int						n	y	n	y	y	CASCADE_UPDATE	Task(TaskID)		Compound Primary Key
NumTotalVolunteers	int		1				n	n	n						Total number of needed people
NumCurrVolunteers	int		0				n	n	n						Number of people currently assigned
UserEvent															
UserID	int		n				n	y	n	y	y	CASCADE_UPDATE	User(UserID)		Compound Primary Key
RoleID	nvarchar	50	n				n	n	n	y	y	CASCADE_UPDATE	Role(RoleID)		Compound Primary Key
EventID	int								y		y		Event(EventID)		Compound Primary Key
Role															
RoleID	nvarchar	50	attende	e					y						
RoleDescription	nvarchar	3000													
CanEditEvents	bit		0				n								0 can not edit events, 1 can

	DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY		REFERENCES	COMMENTS
Volunteers															The Volunteer and Donor users table
VolunteerID	int			y	100 000	1	n		y	y					
UserID	int			y	100 000	1	n	y	y		y			Volunteer(VolunteerID)	
EventID	int						y				y			Volunteer(EventID)	
Active	bit														
VolunteerType															The Volunteer users table
VolunteerID	int			y	100 000	1	n		y	y					
RoleID	nvarchar	50	attende e				n								
SkillSet															A table of all the skills volunteers can have
SkillSetID	nvarchar	50					n		y	y					
SkillSetDescription	nvarchar	20 0					n		y						
VolunteerSkillSet															A table of the volunteers with skill sets
VolunteerID	int			y	100 000	1		y			y			Volunteer(VolunteerID)	
SkillSetID	nvarchar	50						y			y			SkillSet(SkillSetID)	
Availability															The recurring weekly availability of volunteers / donors / suppliers / locations
AvailabilityID	int			y	100 000	1	n	y		y					
TimeStart	time						n								

	DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY	REFERENCES	COMMENTS
TimeEnd	time						n							
Sunday	bit						n							Indicates this record's times apply to every Sunday
Monday	bit						n							Indicates this record's times apply to every Monday
Tuesday	bit						n							Indicates this record's times apply to every Tuesday
Wednesday	bit						n							Indicates this record's times apply to every Wednesday
Thursday	bit						n							Indicates this record's times apply to every Thursday
Friday	bit						n							Indicates this record's times apply to every Friday
Saturday	bit						n							Indicates this record's times apply to every Saturday
AvailabilityException														One-off exception availability which overrides the

	DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY		REFERENCES	COMMENTS
															default Availability
ExceptionID	int			y	100 000	1	n	y		y					
ExceptionDate	date						n								
TimeStart	time						y								
TimeEnd	time						y								
VolunteerAvailability															Join table between a volunteer and its availabilities
VolunteerID	int			y	100 000	1	n	y		y	y			Volunteer(VolunteerID)	
AvailabilityID	int			y	100 000	1	n	y		y	y			Availability(AvailabilityID)	
VolunteerAvailabilityException															
VolunteerID	int			y	100 000	1	n	y		y	y			Volunteer(VolunteerID)	
ExceptionID	int			y	100 000	1	n	y		y	y			AvailabilityException(ExceptionID)	
VolunteerProof															Proof of their volunteer work
ProofID	int			y	100 000	1	n	y	y					Task(ProofID)	
VolunteerID	int			y	100 000	1	n	y	n		y			Volunteer(VolunteerID)	
ProofImage	varchar	max					y								
ProofDescription	nvarchar	250					n								
VolunteerRecord															The record of all volunteers / donors
VolunteerRecordID	int			y	100 000	1	n		y	y					
VolunteerID	int			y	100 000	1	n	y	y		y			Volunteer(VolunteerID)	
EventID	int			y	100 000	1	n	y	y		y			-	
LocationID	int			y	100 000	1	n	y	y		y			-	

	datatype	length	default	identity	start	increment	nulls	index	unique	pk	fk	integrity		references	comments
Rating	Enum(1 -5)	1					n	y	n		y				-
EventDateID	datetime	-					n	y	n		y				-
VolunteerSupply															The supplies a supply donor has in stock
SupplyID	int			y	100 000	1	n		y	y					
VolunteerID	int			y	100 000	1	n	y	y		y			Volunteer(VolunteerID)	
SupplyName	nvarchar	50					n		n						
SupplyAmount	int						n		n						
SupplyDescription	nvarchar	25 0					y		n						
Active	bit														
VolunteerReviews															Reviews of all volunteers / donors
ReviewID	int	-	-	y	100 000	4	n	y	y	y	y	-		Reviews(ReviewID)	-
EventID	int	-	-	y	100 000	4	n	-	-	-	y	-		Event(EventID)	-
VolunteerID	int	-	-	y	100 000	4	n	y	y	-	y	-		Volunteer(VolunteerID)	-
Rating	Enum(1 -5)	4	-	-	-	-	n	-	n	-	-	-			-
Comments	nvarchar	25 0	-	-	-	-	y	-	n	-	-	-			-
VolunteerReview															Reviews of all volunteers / donors
ReviewID	int			y	100 000	1	n	y	y	y	y			Reviews(ReviewID)	
VolunteerID	int			y	100 000	1	n	y	y		y			Volunteer(VolunteerID)	
FavoriteVolunteer															
VolunteerID	int			y	100 000	1	n			y	y			Volunteer(VolunteerID)	
UserID	int			y	100 000	1	n				y			User(UserID)	
VolunteerRequest															

	DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY		REFERENCES	COMMENTS
RequestID	int			y	100 000	1	n			y					
VolunteerID	int			y	100 000	1	n			y			Volunteer(VolunteerID)		
TaskID	int			y	000	1	n			y			Task(TaskID)		
VolunteerResponse	bit														
EventResponse	bit														
VerificationRequest															
RequestID	int			y	100 000	1	n			y					
VolunteerID	int						n			y			Volunteer(VolunteerID)		
SkillSetID	nvarchar	50					n			y			SkillSet(SkillSetID)		
Accepted	bit		0												
VolunteerApplications															
RequestID	int			y	100 000	1	n			y					
UserID	int			y	100 000	1	n			y			User(UserID)		
Accepted	bit														
SupplierType															
SupplierTypeID	nvarchar	10						n	n	y	y			The Supplier Type	
Description	nvarchar	500													
Supplier															
SupplierID	int			y	100 000	1	n	y	y	y				auto number	
UserID	int		n				y		n		y		User(UserID)		
SupplierName	nvarchar	160	n				n		n						
SupplierDescription	nvarchar	3000	n				y		n					Bio	
SupplierPhone	nvarchar	15	n				n		n						
SupplierEmail	nvarchar	250	n				n		y						
SupplierTypeID	nvarchar	10	n				y		n		y		SupplierType(SupplierTypeID)		
SupplierAddress1	nvarchar	100	n				n		y					Address1 and Address2 pair must	

	DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY		REFERENCES	COMMENTS
															be unique
SupplierAddress2	nvarchar	100	n				y		y						Address1 and Address2 pair must be unique
SupplierCity	nvarchar	100	n				n								
SupplierState	nvarchar	100	n				n								
SupplierZipCode	nvarchar	100	n				n								
Active	bit		1				n		n						
Approved	bit		n				y								Null = waiting, 1 = approved, 0 = denied
Location															
LocationID	int		n	y	100000	1	n	y	y	y					Auto-incrementing key
UserID	int		n				y		n		y		User(userID)		
LocationName	nvarchar	160	n				n								Name field to hold the name of the location
LocationDescription	nvarchar	3000	n				y								Holds the description of the location
LocationPricingText	nvarchar	3000	n				y								Holds text for pricing information
LocationPhone	nvarchar	15	n				y								Holds the phone number to contact the location owner
LocationEmail	nvarchar	250	n				y								Holds the email address to contact the

	DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY		REFERENCES	COMMENTS
															location owner
LocationAddress1	nvarchar	100	n				n		y						Holds the address of the location
LocationAddress2	nvarchar	100	n				y								
LocationZipCode	nvarchar	100	n				n				y		ZIP(ZIPCode)		
LocationImageName	nvarchar	200	n				y								A field to hold a path to the image for the location splash.
LocationActive	bit		1				n								
Sublocation															
SublocationID	int		n	y	100000	1	n	y	y	y					
LocationID	int		n				n		n		y	CASCADE UPDATE	Location(LocationID)		
SublocationName	nvarchar	160	n				n		n						
SublocationDescription	nvarchar	1000	n				y								A description field for the sublocation
SublocationActive	bit		1				n								
Rule															
SublocationID	int		n							y			Sublocation(SublocationID)		
Rule	nvarchar	160	n							y					
Supply															
SupplyID	int			y	100000	1	n	y	y	y					
SublocationID	int						n		n		y	yes	Sublocation(SublocationID)		
SupplyName	nvarchar	160					y	n							

	datatype	length	default	identity	start	increment	nulls	index	unique	pk	fk	integrity		references	comments
Description	nvarchar	3000					y	n							
Entrance															
EntranceID	int			y	100000	1	n	y	y	y					
LocationID	int						n	n			y		Location(LocationID)		
EntranceName	nvarchar	100					n	n							
Description	nvarchar	255					y	n							
ParkingLot															
ParkingLotID	int		n	y	100000	1	n	y	y	y					
LocationID	int		n				n				y		Location(LocationID)		
Name	nvarchar	160	n				n								
Description	nvarchar	3000	n				y								
ImagePath	nvarchar	200	n				y								
Active	bit			1			n								
EventParking															
EventID	int								y	y			Event(EventID)		
ParkingLotID	int								y	y			ParkingLots(ParkingLotID)		
Tag															
TagID	nvarchar	50					n	y	y	y				The tag's name	
SupplierTag															
TagID	nvarchar	50							y	y			Tag(TagID)		
SupplierID	int								y	y			Suppliers(SupplierID)		
LocationTag															
TagID	nvarchar	50							y	y			Tag(TagID)		
LocationID	int								y	y			Location(LocationID)		
EventTag															
TagID	nvarchar	50							y	y			Tag(TagID)		
EventID	int								y	y			Event.EventID		

	DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY		REFERENCES	COMMENTS
ActivityTag															
TagID	nvarchar	50								y	y			Tag(TagID)	
ActivityID	int									y	y			Activity.ActivityID	
SupplierAvailability															
SupplierID	int			y	100 000	1	n	y		y	y			Supplier(SupplierID)	
AvailabilityID	int			y	100 000	1	n	y		y	y			Availability(AvailabilityID)	
SupplierAvailabilityException															
SupplierID	int			y	100 000	1	n	y		y	y			Supplier(SupplierID)	
ExceptionID	int			y	100 000	1	n	y		y	y			AvailabilityException(ExceptionID)	
LocationAvailability															
LocationID	int			y	100 000	1	n	y		y	y			Location(LocationID)	
AvailabilityID	int			y	100 000	1	n	y		y	y			Availability(AvailabilityID)	
LocationAvailabilityException															
LocationID	int			y	100 000	1	n	y		y	y			Location(LocationID)	
ExceptionID	int			y	100 000	1	n	y		y	y			AvailabilityException(ExceptionID)	
SublocationMap															
SublocationID	int		n					n	y	y	y			Sublocation(SublocationID)	Auto-incrementing key
MapPath	nvarchar	200	n					n		y					A field to hold images of previous lot mappings
MapActive	bit		1				n								
LocationImage															
LocationID	int		n				n	y		y	y			Location(LocationID)	
ImageName	nvarchar	200	n				n			y					A field to hold images

	datatype	length	default	identity	start	increment	nulls	index	unique	pk	fk	integrity		references	comments
															linked to a location.
SupplierImage															
SupplierID	int		n	n			n	y		y	y			Supplier(SupplierID)	
ImageName	nvarchar	200	n				n		y						A field to hold images linked to a Supplier.
SupplierAttendance															
ActivityID	int								y	y				Activity(ActivityID)	
SupplierID	int								y	y				Suppliers(SupplierID)	
Service															
ServiceID	int			y	100000	1	n	y	y	y					auto number
SupplierID	int						n	n	n		y			Suppliers(SupplierID)	
ServiceName	nvarchar	160					n	n	n						
Price	decimal	(10, 2)					n	n	n						
Description	nvarchar	3000					y	n	n						
ServiceImagePath	nvarchar	200	n				n								A field to hold images for Services.
EventEntrance															
EntranceID	int						n	n	n	y	y			Entrances(EntranceID)	
EventID	int						n	n	n	y	y			Event(EventID)	
LocationReview															
ReviewID	int						n	n	n	y	y			Review(ReviewID)	
LocationID	int						n	n	n	y	y			Location(LocationID)	
SupplierReview															
ReviewID	int						n	n	n	y	y			Review(ReviewID)	
SupplierID	int						n	n	n	y	y			Suppliers(SupplierID)	

	DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY		REFERENCES	COMMENTS
ZIP															
ZIPCode	nvarchar	100					n		y	y					
City	nvarchar	20					n								
States	nvarchar	100					n								
Users															
UserID	int			y	100000	1	n	y	y	y					
GivenName	nvarchar	50	n				n	n	n						
FamilyName	nvarchar	50	n				n	n	n						
Email	nvarchar	250	n				n	y	y						username for login
PasswordHash	nvarchar	100	see comments				n								SHA256 - default is hash value for P@ssw0rd
UserState	char	2	n				y								
City	nvarchar	75	n				y								
Zip	int		n				y	y							
UserPhoto	nvarchar	200					y								
UserDescription	nvarchar	3000					y								A brief description that the User can use for their profile
Active	bit		1				n o								deactivate rather than delete
DateCreated	DateTime		DateTimeNow()				n								
Contacts															
ContactID	int			y	100000	1	n	y	y	y					
UserID	int			y	100000	1	n	y	y	y	CASCADE UPDATE	User(userID)			Primary key From User to retrieve from User to create contact

		DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY		REFERENCES	COMMENTS
Active	bit		1				y									Determines whether this contact is on a saved list
DateAdded	DateTi me			DateTi meNow ()			n									
Group Members																
GroupID	int		n					n	y	n	y	y	CASCADE UPDATE	Group(GroupID)		Compound Key
UserID	int			y	100 000	1	n	y	y		y		CASCADE UPDATE	User(UserID)		Primary key From User
Active	bit		1				y									Determines whether this contact is still an active participant
DateAdded	DateTi me			DateTi meNow ()				n								
Group																
GroupID	int		n					n	y	n	y					
GroupName	nvarchar	50						n								
GroupPhoto	nvarchar	20 0						y								
Active	bit		1				y									Group is still active
DateCreated	DateTi me			DateTi meNow ()				n								
Message																
MessageID	int		n					n	y	n	y					
UserID	int			y	100 000	1	n	y	y		y		CASCADE UPDATE	User(UserID)		Primary key From User

	DATATYPE	LENGTH	DEFAULT	IDENTITY	START	INCREMENT	NULLS	INDEX	UNIQUE	PK	FK	INTEGRITY		REFERENCES	COMMENTS
Message	nvarchar	250	n			y									
MessagePhoto	nvarchar	200				y									
DateCreated	DateTi me		DateTi meNow ()			n									
Event Group															
EventGroupID	int		n				n	y	n	y					
EventID	int			y	100000	1	n	y	y		y	CASCADE UPDATE	Event(EventID)		
UserID	int			y	100000	1	n	y	y		y	CASCADE UPDATE	User(UserID)	Primary key From User for Users in Event Group	
EventGroupName	nvarchar	50	n				n	y	n						
EventGroupPhoto	nvarchar						y								
EventGroupDescription	nvarchar	3000	n				n	y	y						username for login
DateCreated	DateTi me		DateTi meNow ()			n									
Active	bit		1			n									
Interest															
UserID	int		n				n	y	n	y	y				compound primary key
TagID	int		n				n	y	n	y		Tag(TagTypeID)			
DateAdded	DateTi me		DateTi meNow ()			n									
Saved Events															
UserID	int			y	100000	1	n	y	y	y		CASCADE UPDATE	User(UserID)	compound primary key	
EventID	int			y	100000	1	n	y	y		y	CASCADE UPDATE	Event(EventID)		

	datatype	length	default	identity	start	increment	nulls	index	unique	pk	fk	integrity		references	comments
DateSaved			DateTi me(No w)				n								
Active	bit	1					n								
EventRSVP															
EventID	int	n	y	100 000	1	n	y	y	y					Event(EventID)	
UserID	int		y					y	y		y			User(UserID)	
AttendedEvent	bit	0					n	y							
Active	bit	1					n	y							
Reviews															
ReviewID	int		y	100 000	1	n	y	y	y						
UserID	int		y					y		y				User(UserID)	
ReviewType	nvarchar	20					n		n						Supplier Review, User Review, Event Review etc.
Rating	enum(1 ,2,3,4,5)														
Review	nvarchar	30 0													
DateCreated	DateTi me		DateTi meNow ()				n								
Active	bit	1					n								
UserImage															
ImageID	int	n					n	y	n	y					
UserID	int	n					n	y	n	n	y			User(UserID)	
ImageName	nvarchar	20 0					n								
DateCreated	DateTi me						n								
Report															
ReportID	int		y	100 000	1	n	y	y	y						
UserID	int						n				y			User(UserID)	
Reason	nvarchar	25					n								
Description	nvarchar	25 0					y								
DateCreated	DateTi me						n								

7 Class Diagrams

7.1 *Class Diagrams Summary*

The purpose of the Class Diagrams section is for the conceptual modeling of the structure of the application. Each block represents a class. By displaying all the classes in this way, it becomes easier to see the connection between them and how they play into the overall application.

The top of the box is the name of the class. Each of the names inside of the middle of the box represent a method and contain information about their inputs and functions. The naming for the methods is based on their functions and are roughly organized in CRUD order.

7.2 Data Domain

7.2.1 Event Data Domain

7.2.1.1 Event

Event
<pre>+ int EventID : set, get + string EventName: set, get + string EventDescription: set, get + DateTime EventCreatedDate: set, get + decimal TotalBudget: set, get + bool Active : set, get + int? LocationID : set, get</pre>
EventVM: Event
<pre>+ List<Activites> Activities : set, get + List<EventPicture> EventPictures : set, get + List<EventUpload> EventUploadID : set, get + List<EventDate> EventDates : set, get + EmergencyProcedure EmergencyProcedure + Forum Forum : set, get + int EventRegistrationViewID : set, get + List<Activites> Activities : set, get + int EventGroupID : set, get + int EventParkingID : set, get + int EventEntranceID : set, get + int SavedEventID : set, get + List<Invitation> Invitations : set, get + List<Task> Tasks : set, get + int EventTagsID : set, get + Location Location : set, get</pre>

7.2.1.2 Activity

Activity

```
+ int ActivityID : set, get
+ string ActivityName: set, get
+ string ActivityDescription: set, get
+ bool PublicActivity : set, set
+ DateTime StartTime : set, get
+ DateTime EndTime : set, get
+ string ActivityImageName : set, get
+ int? SublocationID : set, get
+ DateTime EventDateID : set, get
+ int EventID : set, get
```

ActivityVM : Activity

```
+ EventDate EventDate : set, get
+ List<ActivityResult> ActivityResults :
set, get
+ Sublocation ActivitySublocation : set,
get
+ string SublocationName : set, get
+ string DisplayTimeStart : get
+ string DisplayTimeEnd : get
+ string DisplayEventDate : get
```

7.2.1.3 EventDate

EventDate

```
+ DateTime EventDateID : set, get
+ int EventID : set, get
+ DateTime StartDateTime : set, get
+ DateTime EndDateTime : set, get
+ bool Active : set, get
```

7.2.1.4 EventUpload

EventUpload

```
+ int EventUploadID : set, get
+ string EventFileName : set, get
+ DateTime DateTimeCreated : set, get
+ string Title : set, get
+ string Description : set, get
+ bool Active : set, get
```

7.2.1.5 EventPicture

EventPicture

```
+ int EventPictureID : set, get
+ string EventPictureName : set, get
+ DateTime DateTimeCreated : set, get
+ string Title : set, get
+ string Caption : set, get
+ string Description : set, get
+ bool isMainDisplay : set, get
+ bool Active : set, get
```

7.2.1.6 EmergencyProcedure

EmergencyProcedure

```
+ int EmergencyProcedureID : set, get
+ string Message : set, get
```



EmergencyProcedureVM : EmergencyProcedure

```
+ List<EmergencyContact> EmergencyContacts : set, get
+ int EventID : set, get
+ MusterPoint MusterPoint : set, get
+ List<EmergencyExit> EmergencyExits : set, get
+ List<User> EmergencyContact
```

7.2.1.7 MusterPoint

MusterPoint

```
+ int MusterPointID : set, get
+ string
EmergencyMusterPointDescription : set,
get
+ Sublocation SubLocation : set, get
```

7.2.1.8 EmergencyExit

EmergencyExit

```
+ int EmergencyExitD : set, get
+ string
EmergencyMusterPointDescription : set,
get
+ Sublocation SubLocation : set, get
```

7.2.1.9 ActivityResult

ActivityResult

```
+ int ActivityResultsID : set, get
+ string ActivityName : set, get
+ string ActivityResultsName : set, get
```

7.2.1.10 NotifcationGroup

NotifcationGroup

```
+ int NotifcationGroupID : set, get
+ DateTime DateTimeCreated : set, get
```



NotifcationGroupVM : NotifcationGroup

```
+ List<User> Users : set, get
```

7.2.1.11 Task

Task

```
+int TaskID : set, get  
+ string Name : set, get  
+ string Description : set, get  
+ DateTime DueDate: set, get  
+ int Priority: set, get  
+ DateTime CompletionDate: set, get  
+ int ProofID : set, get  
+ bool isDone : set, get  
+ int TaskAssignmentID : set, get  
+ int EventID : set, get  
+ bool Active : set, get
```



TaskVM : Task

```
+ string TaskPriority : set, get  
+ string TaskEventName : set, get
```

7.2.1.12 TaskAssignment

TaskAssignment

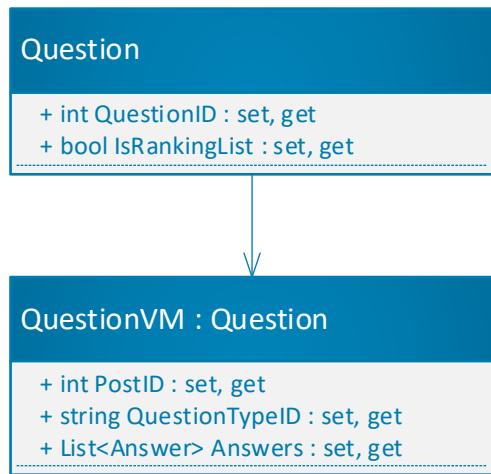
```
+ int TaskAssignmentID : set, get  
+ DateTime DateAssigned : set, get  
+ int TaskID : set, get  
+ string RoleID : set, get  
+ int UserID : set, get
```



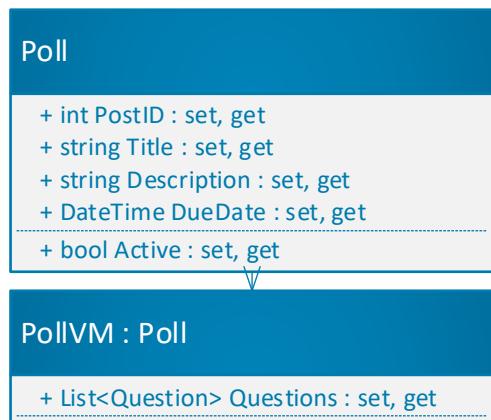
TaskAssignmentVM :
TaskAssignment

```
+ string Name : set, get
```

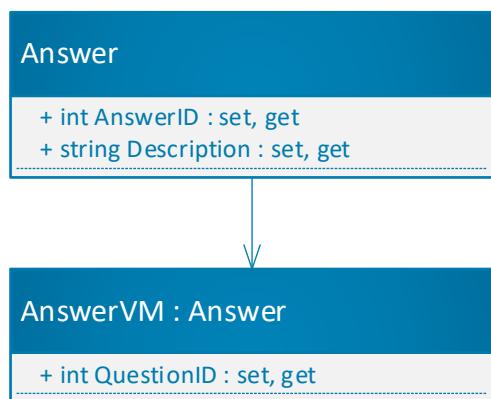
7.2.1.13 Question



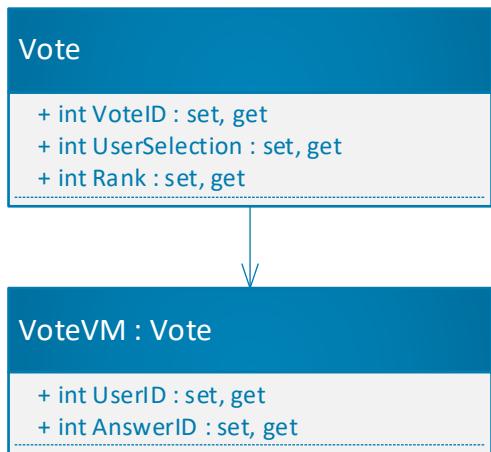
7.2.1.14 Poll



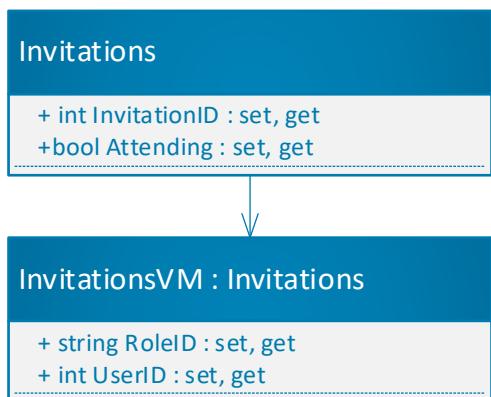
7.2.1.15 Answer



7.2.1.16 Vote



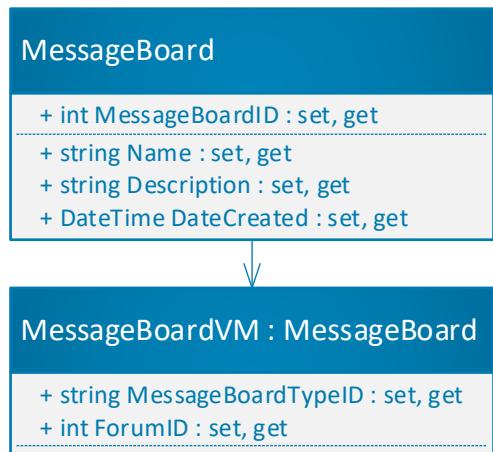
7.2.1.17 Invitations



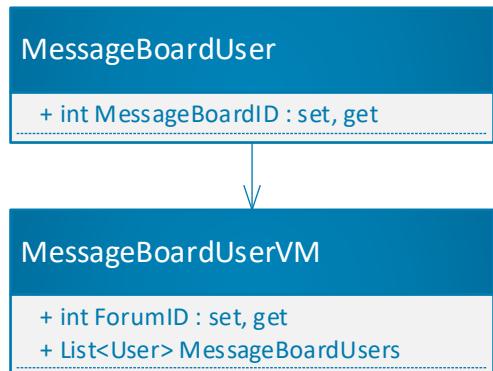
7.2.1.18 Forum



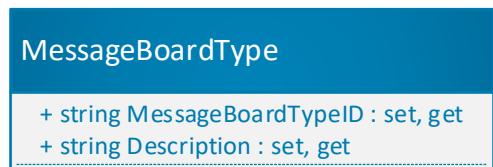
7.2.1.19 MessageBoard



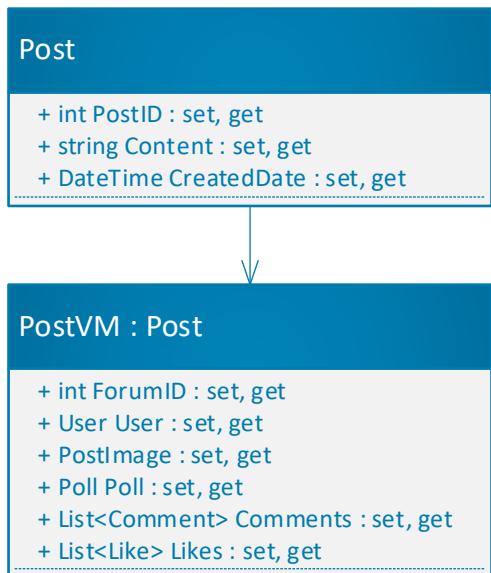
7.2.1.20 MessageBoardUser



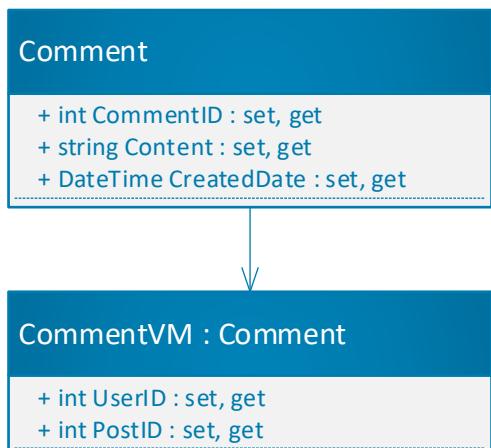
7.2.1.21 MessageBoardType



7.2.1.22 Post



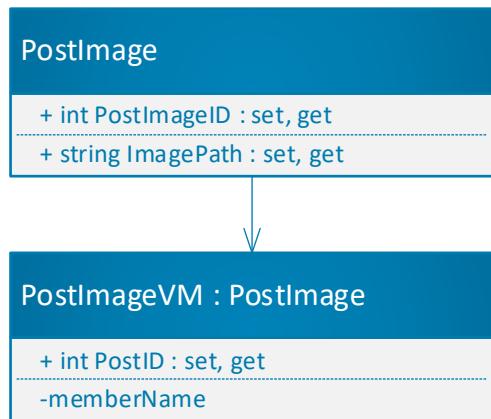
7.2.1.23 Comment



7.2.1.24 Like



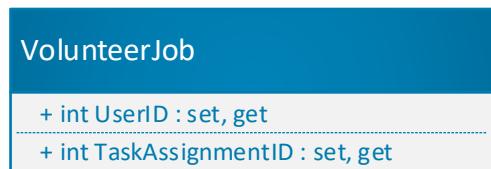
7.2.1.25 PostImage



7.2.1.26 AfterActionReview



7.2.1.27 VolunteerJob



7.2.1.28 Priority

Priority

```
+ int PriorityID: set, get  
+ string Description: set, get
```

7.2.2 Volunteer Data Domain

7.2.2.1 Availability

Availability

```
+ int ForeignID get set  
+ int AvailabilityID get set  
+ DateTime? DateID get set  
+ DateTime? TimeStart get set  
+ DateTime? TimeEnd get set  
+ bool Sunday get set  
+ bool Monday get set  
+ bool Tuesday get set  
+ bool Wednesday get set  
+ bool Thursday get set  
+ bool Friday get set  
+ bool Saturday get set
```

7.2.2.2 FavoriteVolunteers

FavoriteVolunteers

```
+ int VolunteerID set get  
+ int UserID set get
```

FavoriteVolunteersVM

```
+ Volunteer Name  
+ Event Planner Name
```

7.2.2.3 Role

Role

```
+ int RoleID set get  
+ String RoleName set get  
+ String RoleDescription set get
```

7.2.2.4 VerificationRequests

VerificationRequests

```
+ int RequestID set get  
+ int VolunteerID set get  
+ string SkillSetID set get  
+ boolean Accepted set get
```

VerificationRequestsVM

```
+ Request Number  
+ Volunteer Name  
+ Skill Set Name  
+ Was Accepted
```

7.2.2.5 VolunteerApplications

VolunteerApplications

```
+ int RequestID set get  
+ int VolunteerID set get  
+ string SkillSetID set get  
+ boolean Accepted set get
```

VolunteerApplicationsVM

```
+ Request Number  
+ User Name  
+ Was Accepted
```

7.2.2.6 VolunteerAvailability

VolunteerAvailability

```
+ int VolunteerID set get  
+ int AvailabilityID set get
```



VolunteerAvailabilityVM : VolunteerAvailability & Availability

```
+ Name  
+ Available Day  
+ Time Start  
+ Time End
```

7.2.2.7 VolunteerProof

VolunteerProofVM :
VolunteerProof

- + Volunteer Name
- + Image of Proof
- + Description of Proof
- + Extra notes
- + sortVolunteerProofAlphabetically();

7.2.2.8 VolunteerRecord

VolunteerRecordVM :
VolunteerRecord

- + Volunteer Name
- + Event Name
- + Event Location
- + Volunteer Rating from Event
- + Time of Event
- + selectRecordByDateRange();
- + selectRecordByRating();

7.2.2.9 VolunteerRequests

VolunteerRequests

- + int RequestID set get
- + int VolunteerID set get
- + int TaskID set get
- + bit VolunteerResponse set get
- + bit EventResponse set get

VolunteerRequestsVM

- + Volunteer Name
- + Task Name
- + Volunteer Responded
- + Event Responded

7.2.2.10 **VolunteerSkillSet**

VolunteerSkillSet

```
+ int VolunteerID set get  
+ string SkillSetID set get  
+ String SkillSetDescription set get
```

7.2.2.11 **VolunteerSupplies**

VolunteerSupplies

```
+ int SupplyID set get  
+ int VolunteerID set get  
+ String SupplyName set get  
+ int SupplyAmount set get  
+ String SupplyDescription set get
```



VolunteersSuppliesVM :
VolunteerSupplies

```
+ Full Name  
+ Supply Name  
+ Amount of supplies  
+ Description of the supplies
```

7.2.2.12 **VolunteerType**

VolunteerType

```
+ int VolunteerID set get  
+ string RoleID set get
```

VolunteerTypeVM : VolunteerType

```
+ Volunteer Name  
+ Role
```

7.2.2.13 **Volunteer**

Volunteer

```
+ int VolunteerID set get  
+ int UserID set get  
+ int EventID set get  
+ string VolunteerType set get  
+ string GivenName set get  
+ string FamilyName set get  
+ string Email set get  
+ string PasswordHash set get  
+ string State set get  
+ string City set get  
+ int Zip set get  
+ string UserPhoto set get  
+ string UserDescription set get  
+ int Rating set get  
+ bool Active set get  
+ DateTime DateCreated set get
```

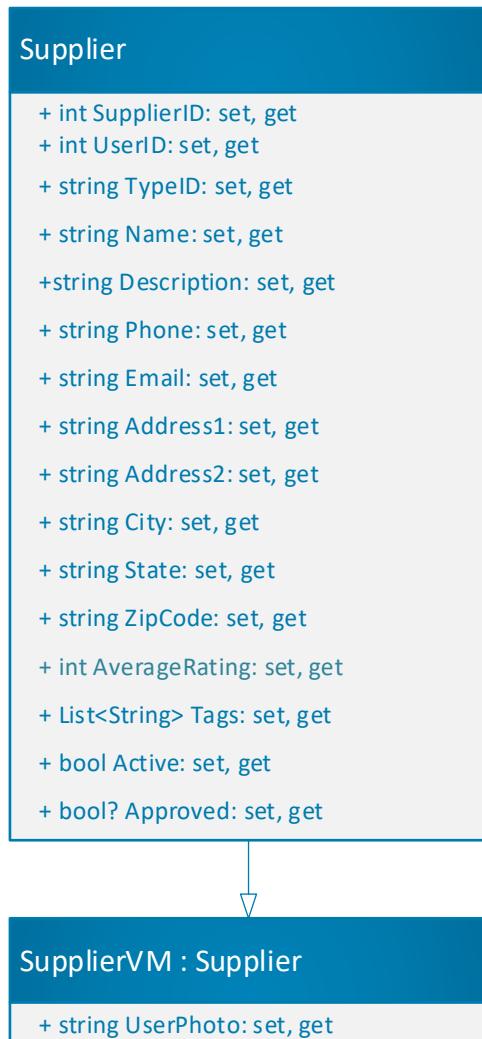
7.2.2.14 **VolunteerNeed**

VolunteerNeed

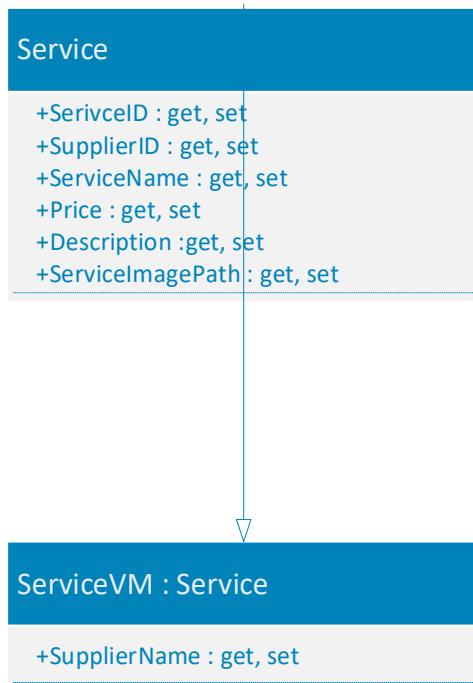
```
-int TaskID get set  
-int NumTotalVolunteers get set  
-int NumCurrVolunteers get set
```

7.2.3 Supplier Data Domain

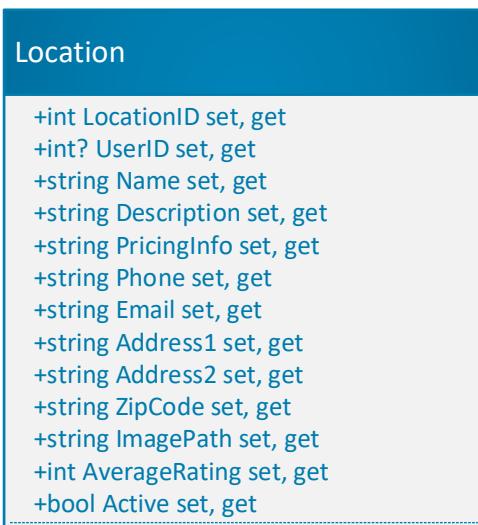
7.2.3.1 Supplier



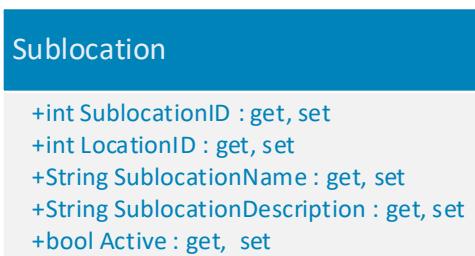
7.2.3.2 Service



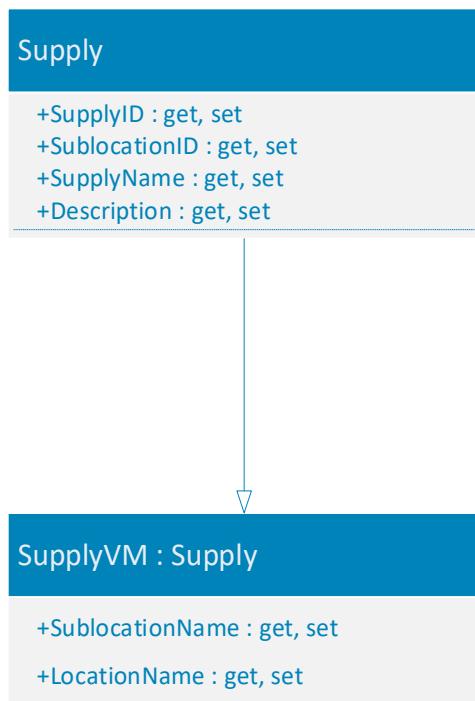
7.2.3.3 Location



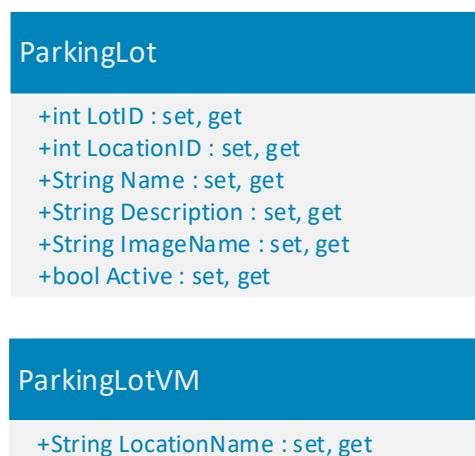
7.2.3.4 Sublocation



7.2.3.5 Supply



7.2.3.6 Parking Lot



7.2.3.7 Entrance

Entrance

```
+ int EntranceID : get; set;  
+ int LocationID : get; set;  
+ String EntranceName : get; set;  
+ String Description: get; set;
```

7.2.3.8 Availability

Availability

```
+ int ForeignID : get, set  
+ int AvailabilityID : get, set  
+ DateTime? TimeStart : get, set  
+ DateTime? TimeEnd : get, set  
+ bool Sunday : get, set  
+ bool Monday : get, set  
+ bool Tuesday : get, set  
+ bool Wednesday : get, set  
+ bool Thursday : get, set  
+ bool Friday : get, set  
+ bool Saturday : get, set
```

7.2.3.9 ZIP

ZIP

```
+string ZIPCode set, get  
+string City set, get  
+string State set, get
```

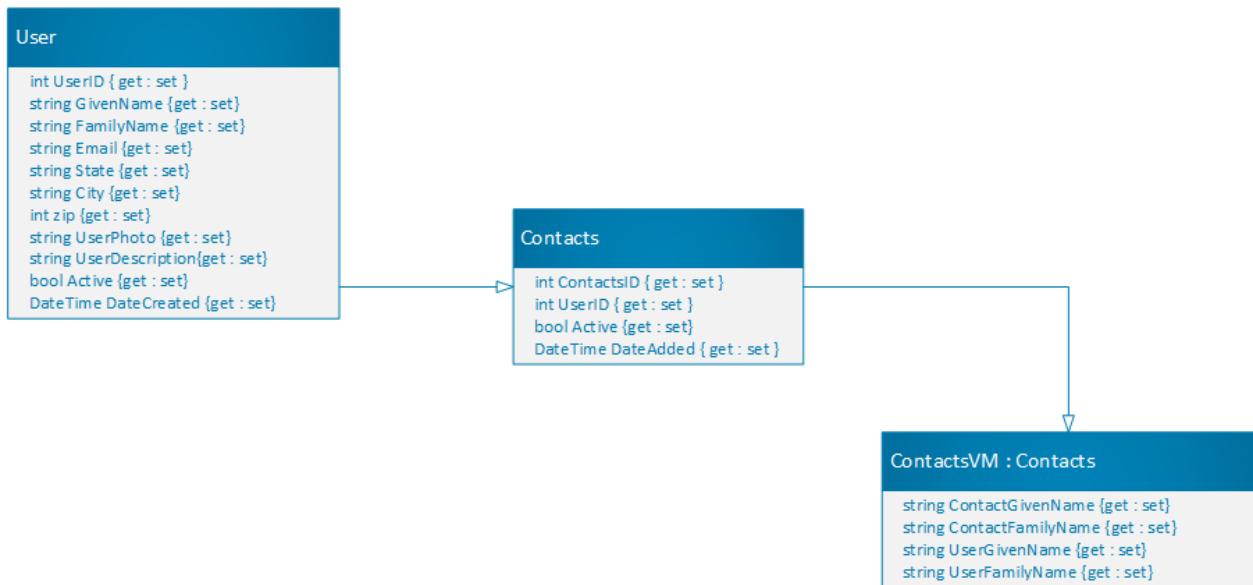
7.2.4 Attendee Data Domain

7.2.4.1 User

User

```
int UserID : get, set  
string GivenName : get, set  
string FamilyName : get, set  
string Email : get, set  
string State : get, set  
string City : get, set  
int zip : get, set  
string UserPhoto : get, set  
string UserDescription: get, set  
bool Active : get, set  
DateTime DateCreated : get, set  
List<string> roles : get, set
```

7.2.4.2 Contacts

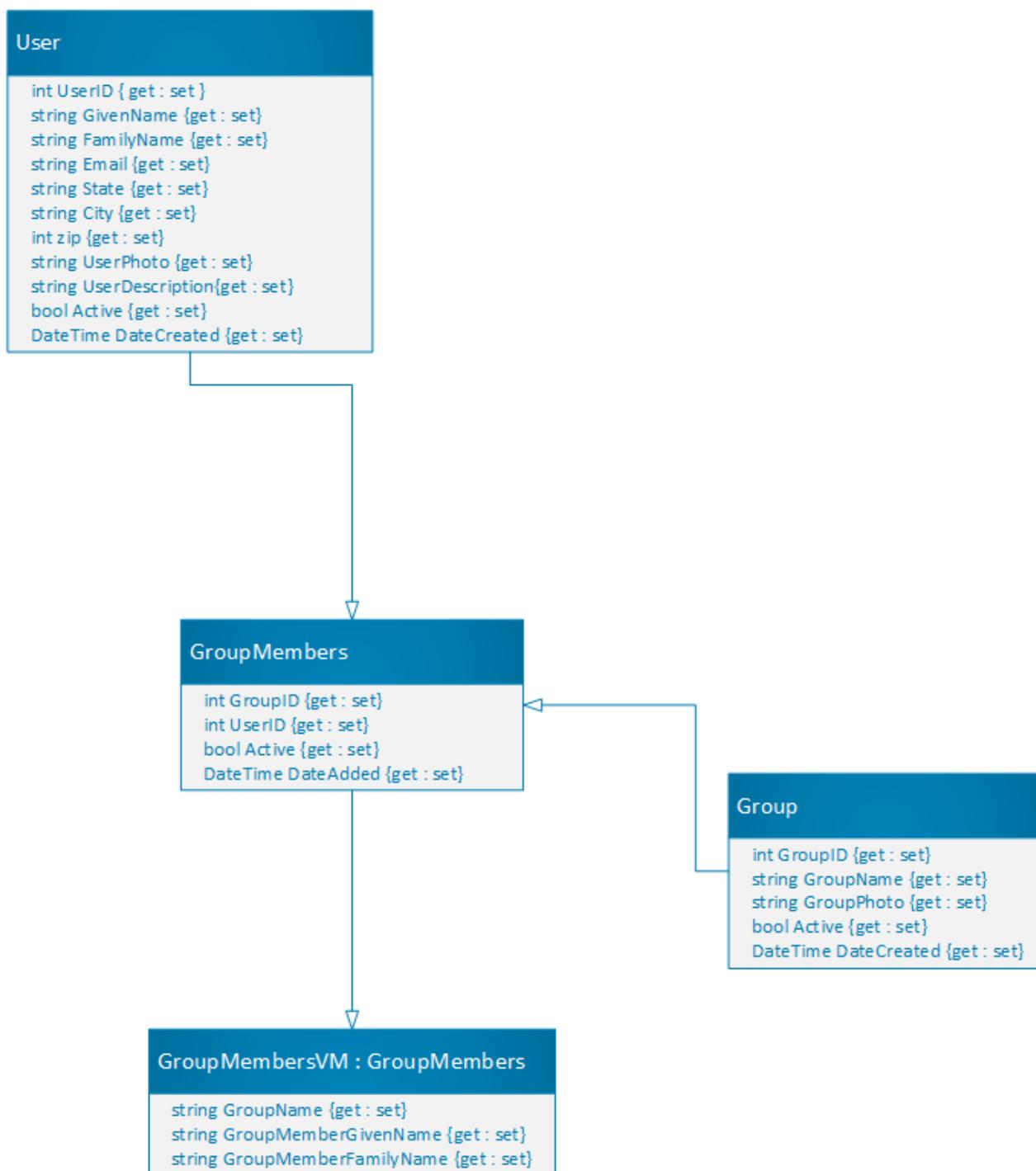


7.2.4.3 Group

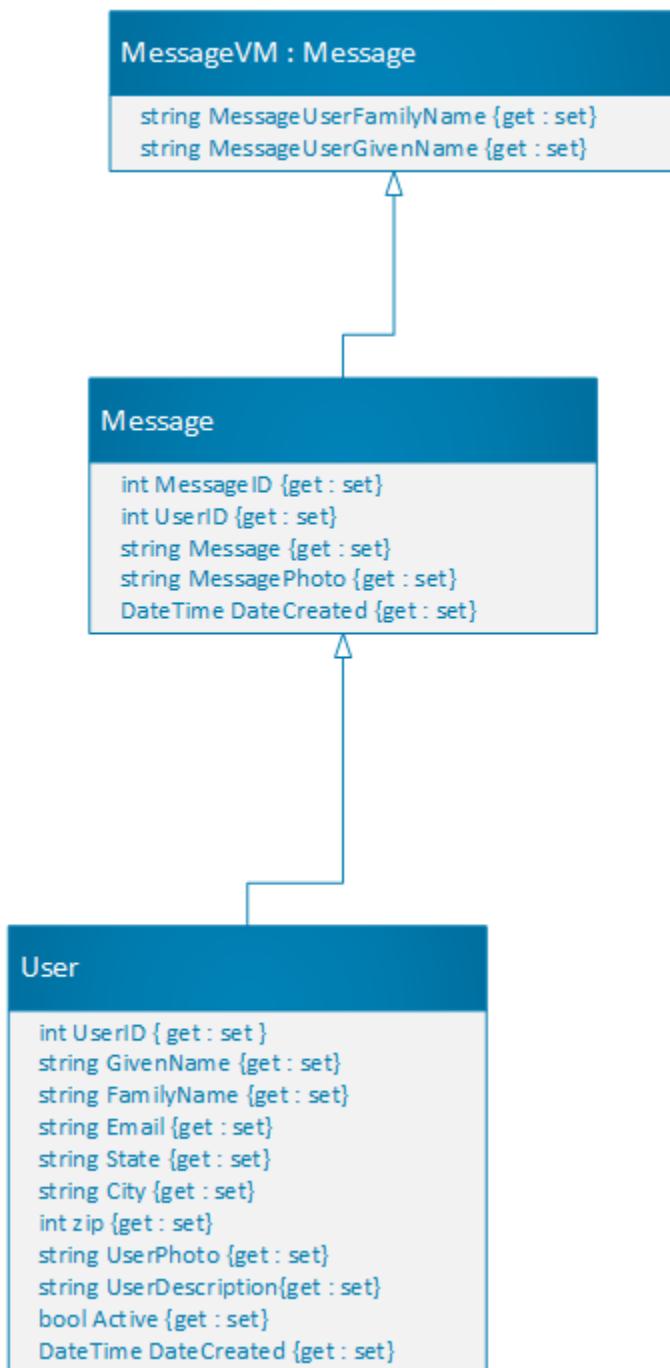
Group

```
int GroupID {get : set}  
string GroupName {get : set}  
string GroupPhoto {get : set}  
bool Active {get : set}  
DateTime DateCreated {get : set}
```

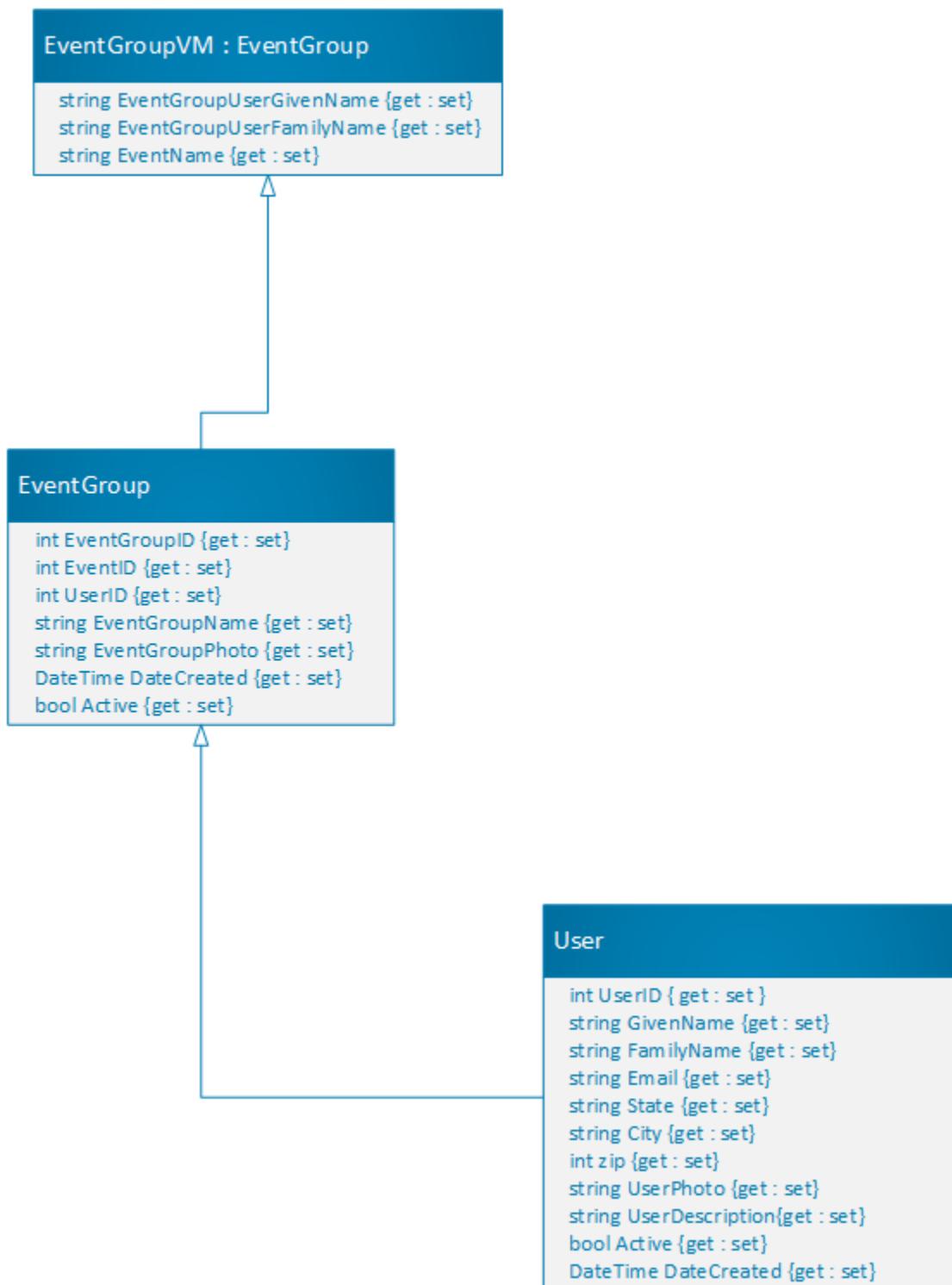
7.2.4.4 GroupMembers



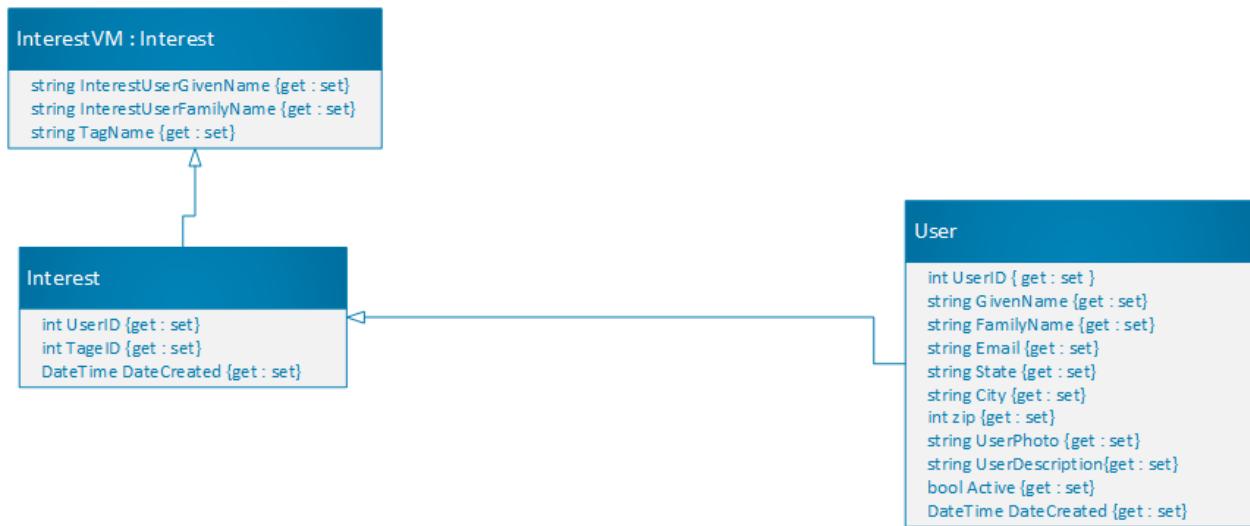
7.2.4.5 Message



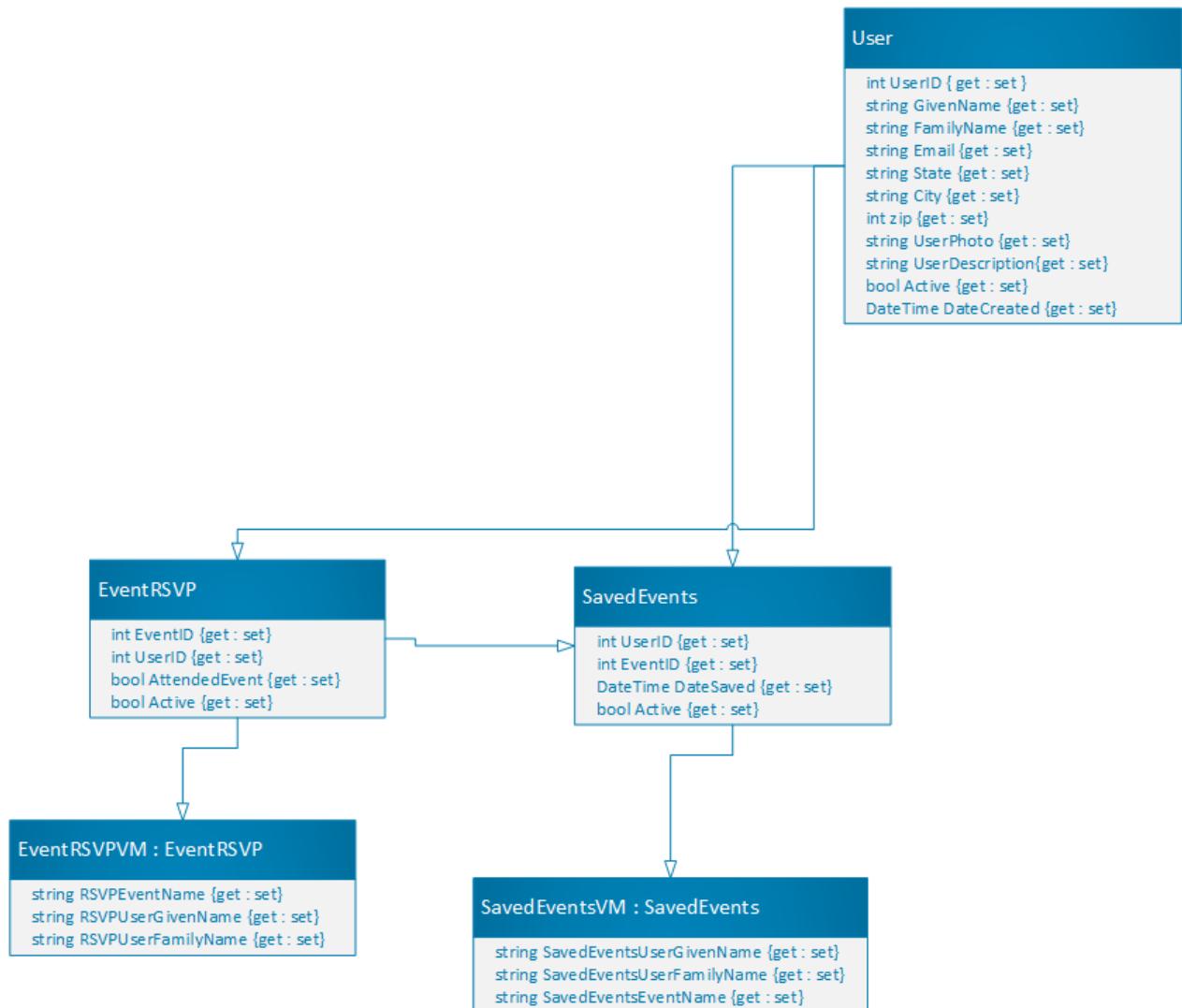
7.2.4.6 EventGroup



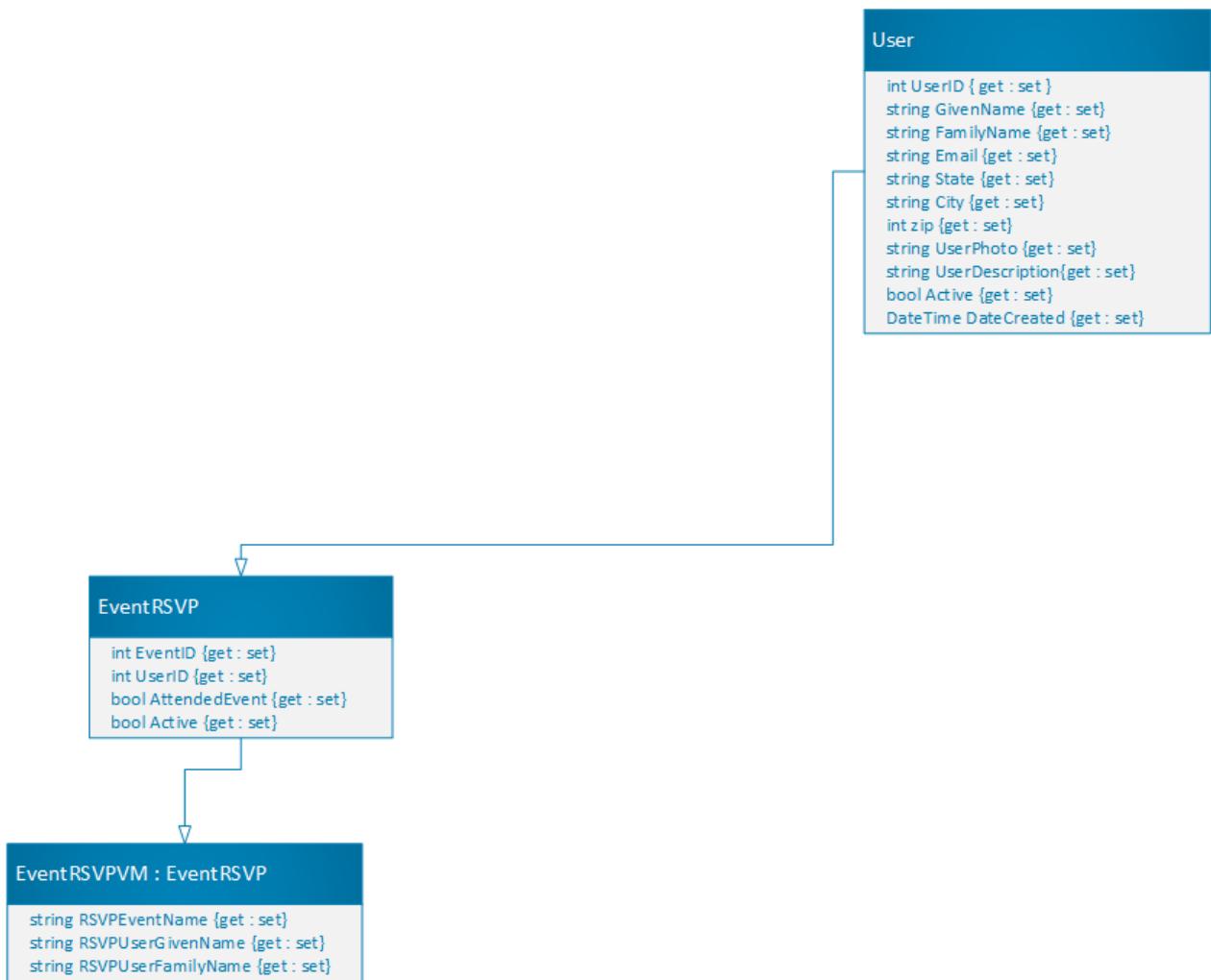
7.2.4.7 Interest



7.2.4.8 SavedEvents



7.2.4.9 EventRSVP

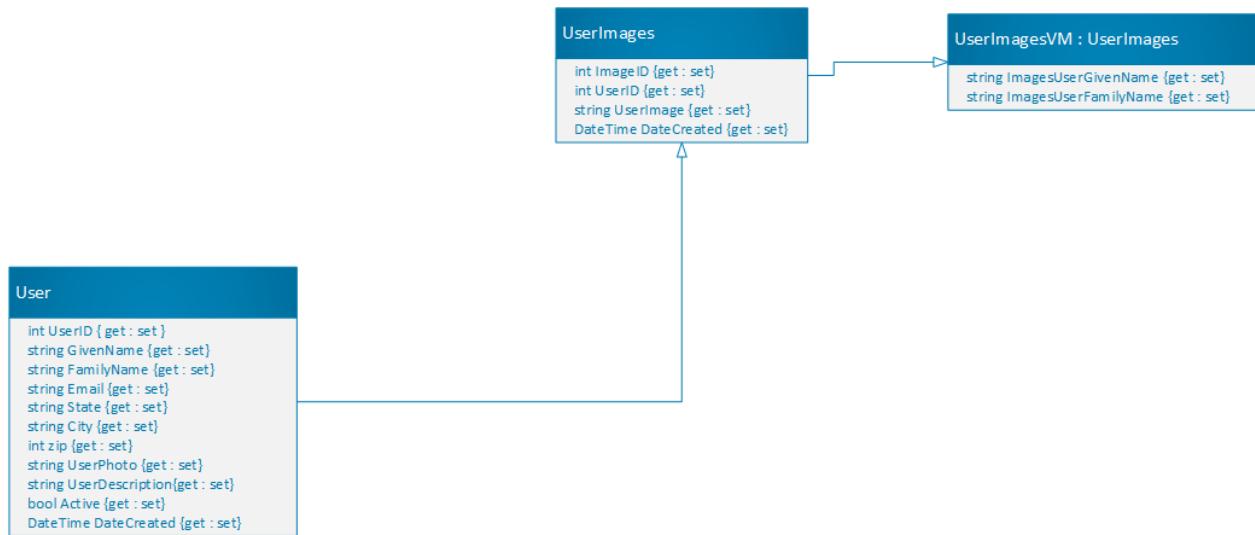


7.2.4.10 Reviews

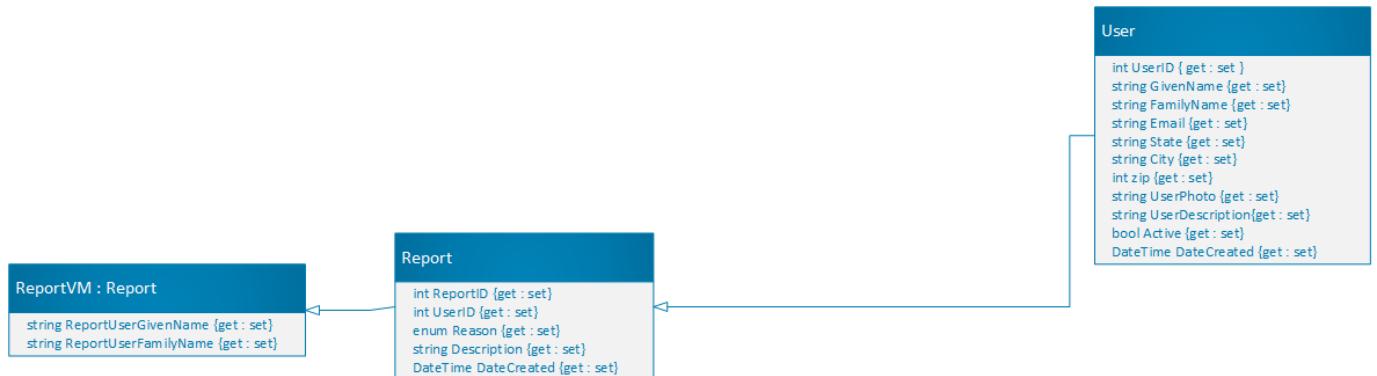
Reviews

```
+ int ForeignID set get
+ int ReviewID set get
+ string FullName set get
+ string ReviewType set get
+ int Rating set get
+ string Review set get
+ DateTime DateCreated set get
+ bool Active set get
```

7.2.4.11 UserImages



7.2.4.12 Report



7.3 Data Access Interfaces

7.3.1 Event Data Access Interfaces

7.3.1.1 IForumAccessor

IForumAccessor

```
+int InsertForum(string forumName, string
forumDescription)
+int UpdateForumbyForumID(int forumID,
string oldForumName, newForumName,
oldForumDescription, newForumDescription)
+int DeleteForum(int forumID)
+ForumVM SelectForumByEventID(int
eventID)
```

7.3.1.2 IMessageBoardAccessor

IMessageBoardAccessor

```
+List<MessageBoardsVM>Select
MessageBoardsByForumID
+MessageBoardVM
SelectMessageBoardByMessageBoardID(int
messageBoardID)
+int InsertMessageBoard(string name, string
description)
+int
UpdateMessageBoardByMessageBoardID(int
MessageBoardID, string oldName, string
newName, string oldDescription, string
newDescription)
+int
DeleteMessageBoardByMessageBoardID(int
messageBoardID)
+UserVM
SelectMessageBoardUserByUserIDAndMessa
geBoardID(int userID, int messageBoardID)
+List<UserVM>SelectMessageBoardUsersByU
serIDAndMessageBoardID(int userID, int
messageBoardID)
+int
DeleteMessageBoardUserByUserIDandMessa
geBoardID(int userID, int messageBoardID)
```

7.3.1.3 IPostAccessor

IPostAccessor

```
+int InsertPost(string content)
+PostVM SelectPostByPostID(int postID)
+List<PostVM>SelectPostsByMessageBo
ardID(int messageBoardID)
+int UpdatePostByPostID(int postID,
string oldContent, new Content)
+int DeletePost(int postID)
+PostImageVM
SelectPostImageByPostID(int postID)
+Vote SelectVoteByPostID(int postID)
```

7.3.1.4 ICommentAccessor

ICommentAccessor

```
+int InsertComment(string content)
+List<CommentVM>SelectCommentsBy
PostID(int postID)
+int SelectCommentByID(commentID,
newContent, oldContent)
+int DeleteComment(int CommentID)
```

7.3.1.5 ICommentLikeAccessor

ICommentLikeAccessor

```
+bool ToggleLike()
+List<Like>SelectLikesByCommentID(int
commentID)
```

7.3.1.6 IPostImage

IPostImage

```
+int insertPostImage()
+int DeletePostImageByPostImageID(int
PostImageID )
+PostImageVM
SelectPostImageByPostImageID(int
PostImageID)
```

7.3.1.7 IAfterActionReview

IAfterActionReview

```
+List<ReviewVM>
SelectReviewsByEventID()
+List<VolunteerVM>SelectVolunteersBy
EventID(int EventID)
+List
<AttendeeVM>SelectAttendeesByEventI
D(int eventID)
+List<Suppliers>SelectSuppliersByEventI
D(int EventID)
```

7.3.1.8 IQuestionAccessor

IQuestionAccessor

```
+ int SelectQuestionsByPostID(int
postID)
+ List<Question>
SelectQuestionsByCreatedDateAndEvent
ID(LocalDate date, int eventID)
+ List<Question>
SelectQuestionsByQuestionTypeID(int
questionTypeID)
+ List<Question>
SelectQuestionsByAnswered(bool
isAnswered)
+ int
UpdateQuestionByQuestionType(int
questionID, int oldQuestionType, int
newQuestionType)
+ bool UpdateQuestionByAnswered(int
questionID, bool oldIsAnswered, bool
newIsAnswered, DateTime
dateAnswered)
+ string UpdateQuestionByContent(int
PostID, int QuestionID, string
OldContent, string NewContent)
+ QuestionVM
InsertNewQuestionByPostID(int postID,
int questionTypeID, bool isRankingList,
bool isAnswered)
+ bool
DeactivateQuestionByEventEndDate(int
eventID, int questionID, datetime
endDateTime)
```

7.3.1.9 IAnswerAccessor

IAnswerAccessor

```
+ List<Answer>
SelectAnswerByQuestionID(int
questionID)
+ List<Answer>
SelectAnswersByDateAnswered(datetim
e DateAnswered)
+ string UpdateAnswerByDescription(int
answerID, string oldDescription, string
newDescription)
+ DateTime
UpdateAnswerByDateAnswered(int
answerID, DateTime oldAnswerDate,
DateTime newAnswerDate)
+ AnswerVM
InsertAnswerByQuestionID(int
questionID, string description, DateTime
dateAnswered)
+ bool
DeactivateAnswerByEventEndDate(int
eventId, int questionID, int answerID,
datetime endDateTime)
```

7.3.1.10 ITaskAccessor

ITaskAccessor

```
+ int SelectTaskByTaskID(int taskID)
+ int
SelectAssignedUserByTaskAssignmentID
(int taskAssignmentID)
+ List<TaskVM>
SelectAllActiveTasksByEventID(int
eventID)
+ List<Task>
SelectTasksByDueDate(DateTime
dueDate)
+ List<Task> SelectTasksByPriority(string
Priority)
+ List<Priority> SelectAllPriorities()
+ bool
UpdateTaskProgressByCompletionDate(
DateTime completionDate)
+ string UpdateTaskPriorityByTaskID(int
taskID)
+ int UpdateTasks(Task oldTask, Task
newTask)
+ int InsertTask(Task newTask)
+ bool
DeactivateTaskByCompletionDate(DateTime
completionDate)
+ List<TaskAssignmentVM>
SelectTaskAssignmentsByTaskID(int
TaskID)
+ bool UserCanEditDeleteTask(int
TaskID)
+ bool DeleteTaskByTaskID(int TaskID)
```

7.3.1.11 IVoteAccessor

IVoteAccessor

```
+ int SelectVoteIDByPollID(int pollID)
+ List<Vote> SelectVoteByRank(bool
isRanked, int rank)
+ VoteVM
UpdateVoteByVoteSelection(Vote vote,
string OldVote, string NewVote)
+ bool
DeactivateVoteByEventEndDate(int
eventID, int VoteID, datetime
endDateTime)
```

7.3.1.12 IIInvitationAccessor

IIInvitationAccessor

```
+ List<Invitation>
SelectInvitationsByEventID(int eventID)
+ int
SelectNumberOfInvitationsByAttending(
bool Attending)
+ int SelectInvitationsByUserID(int
InvitationID, int UserID)
+ int
SelectNumberOfInvitationsByEventID(int
eventID)
+ bool
UpdateInvitationByAttending(bool
Attending)
+ InvitationVM
InsertNewInvitaionByEventID(int
eventID, int userID, int roleID, bool
attending)
+ bool
DeactivateInvitationByEventEndDate(int
eventID, int invitationID, datetime
endTime)
```

7.3.1.13 IEventUploadAccessor

IEventUploadAccessor

```
+ int InsertEventUpload(string
eventFileLocation, string title, string
description)
+ EventUpload
SelectEventUploadByEventUploadID(int
eventUploadID)
+ List<EventUpload >
SelectEventUploadByEventID(int
eventID)
+ int
UpdateEventUploadEventFileNameByEv
entUploadID(int eventUploadID, string
oldEventFileName , string
newEventFileName)
+ int
UpdateEventUploadEventTitleAndDescri
ptionByEventUploadID(int
eventUploadID, string
oldEventUploadTitle, string
newEventUploadTitle, string
oldEventDescription,
newEventDescription)
+ int
UpdateEventUploadByEventUploadID(in
t eventUploadID, string
oldEventFileName, string
newEventFileName, string oldTitle,
string newTitle, string oldDescription,
string newDescription)
+ int
DeactivateEventUploadRecordByEventU
ploadID(int eventUploadID)
+ int
DeleteEventUploadRecordByEventUploa
dID(int eventUploadID)
```

7.3.1.14 IEventPictureAccessor

IEventPictureAccessor

```
+ int InsertEventPictureByEventID(int
eventID, string eventPictureName, string
title, string caption, string description,
bool isMainDisplay)

+ List<Activity>
SelectActivitiesByEventIDAndEventDateID(int eventID, DateTime? eventDateID)

+ EventPicture
SelectEventPictureByEventPictureID(int
eventPictureID)
+ List<EventPicture>
SelectEventPicturesByEventID(int
eventID)
+ List<EventPicture>
SelectEventPicturesByEventIDForMainDi
splay(int eventID)
+ int
UpdateEventPictureNameByEventPictureID(int eventPictureID, int
oldEventPictureName, int
newEventPictureName)
+ int
UpdateEventPictureMainDisplayByEvent
PictureID(int eventPictureID, bool
isMainDisplay)
+ int
UpdateEventPictureTitleCaptionAndDes
criptionByEventPictureID(int
eventPictureID, string
oldEventPictureName, string
newEventPictureName, string oldTitle,
string newTitle, string newCaption,
string oldCaption, string oldDescription,
string newDescription, bool
isMainDisplay)
+ int UpdateEventPicture(int
eventPictureID, string
oldEventPictureName, string
newEventPictureName, string oldTitle,
string newTitle, string newCaption,
string oldCaption, string oldDescription,
string newDescription, bool
isMainDisplay)
+ int
DeactivateEventPictureByEventPictureID
(int eventPictureID)
+ int
DeleteEventPictureByEventPictureID(int
eventPictureID)
```

7.3.1.15 IActivityAccessor

IActivityAccessor

```
+ int InsertActivity(Activity activity)
+ Activity SelectActivityByActivityID ( int
activityID)
+ List<Activity>
SelectActivitiesByEventID ( int eventID)
+ List<ActivityResult>
SelectActivityResultsByActivityID(int
activityID)
+List<ActivityVM>
SelectActivitiesUpcomingDates()
+List<ActivityVM>
SelectActivitiesPastDates()
+ List<Activity>
SelectActivitiesByEventIDAndEventDateI
D ( int eventID,DateTime eventDateID
)
+List<ActivityVM>
SelectUserActivitiesPastDates(int userID)
+List<ActivityVM>
SelectUserActivitiesUpcomingDates(int
userID)
+List<ActivityVM>
SelectActivitiesPastAndUpcomingDates()
+List<ActivityVM>
SelectActivitiesPastAndUpcomingDates()
+List<ActivityVM>
SelectUserActivitiesPastAndUpcomingDa
tes(int userID)
+ List<ActivityResult>
SelectActivityResultsByActivityID(int
activityID)
+List<ActivityVM>
SelectUserActivitiesUpcomingDates(int
userID)
+List<Activity>
SelectActivitiesBySublocationID ( int
sublocationID)
+ActivityVM
SelectActivityVMByActivityID(activityID)
```

IActivityAccessor Continue

```
+ int
UpdateActivityStartTimeByActivityID(int
activityID, DateTime oldStartTime,
DateTime newStartTime)
+ int
UpdateActivityEndTimeByActivityID(int
activityID, DateTime oldEndTime,
DateTime newEndTime)
+ int
UpdateActivitySubLocationByActivityID(i
nt activityID, SubLocation sublocation)
+ int
UpdateActivityImageNameByActivityID(i
nt activityID, string
oldActivityImageName, string
newActivityImageName)
+ int DeactivateActivityByActivityID(int
activityID)
+ int UpdateActivityByActivityID(string
oldActivityName, string
newActivityName, string
oldActivityDescription, string
newActivityDescription, bool public,
DateTime oldStartTime, DateTime
newStartTime, DateTime oldEndTime,
DateTime newEndTime, Sublocation
oldSublocation, Sublocation
newSublocation, EventDate
oldEventDate, EventDate
newEventDate)
+ int DeleteActivityByActivityID(int
activityID)
```

7.3.1.16 IEventAccessor

IEventAccessor

```
+ int InsertEvent(string eventName,
string eventDescription, decimal
TotalBudget)
+ int InsertEventReturnsEventID(string
eventName, string eventDescription,
decimal TotalBudget, int userID)
+ EventVM SelectEventByEventID(int
eventID)
+ List<Event> SelectEventList()
+ Event
SelectEventByEventNameAndDescriptio
n(string eventID, string
eventDescription)
+ bool
CheckUserEditPermissionForEvent(int
eventID, int userID)
+ List<Event>
SelectEventListByUserID(int userID)
+ int UpdateEventNameByEventID(int
eventID, string oldEventName, string
newEventName)
+ List<EventVM>
SelectEventsUpcomingDates();
+ List<EventVM>
SelectEventsUpcomingAndPastDates();
+ List<EventVM>
SelectEventsPastDates();
+ List<EventVM>
SelectUserEventsForUpcomingDates(int
userID);
+ List<EventVM>
SelectUserEventsForPastDates(int
userID);
+ List<EventVM>
SelectUserEventsForPastAndUpcomingD
ates(int userID);
```

IEventAccessor

```
+ int UpdateEventDescriptionByEventID(int eventID, string oldEventDescription, string newEventDescription)
+ int UpdateEventByEventID(int eventID, string oldEventName, string newEventName, string oldEventDescription, string newEventDescription, decimal oldTotalBudget, decimal newTotalBudget)
+ int DeactivateEventByEventID(int eventID)
+ int DeleteEventByEventID(int eventID)
+ int UpdateEventLocationByEventID(int eventID, int? oldLocationID, int? newLocationID)
+ List<User> SelectEventPlannersForEvent(int eventID)
```

7.3.1.17 IPollAccessor

IPollAccessor

```
+ int InsertPollByPostID(int postID, string title, string description, DateTime dueDate)
+ Poll SelectPollByPostID(int postID)
+ int UpdatePollTitleByPollID(int postID, string oldPollTitle, string newPollTitle)
+ int UpdatePollDescriptionByPostID(int postID, string oldPollDescription, string newPollDescription)
+ int UpdatePollDueDateByPostID(int postID, string oldPollDueDate, string newPollDueDate)
+ int UpdatePollByPostID(int postID, string oldPollName, string newPollName, string oldPollDescription, string newPollDescription, DateTime oldDueDate, DateTime newDueDate)
+ int DeactivatePollByPollID(int pollID)
+ int DeletePollByPollID(int pollID)
```

7.3.1.18 IActivityResultAccessor

IActivityResultAccessor

```
+ int InsertActivityResultByActivityID(  
    string activityID, string ActivityName,  
    string activityResultName)  
+ ActivityResult  
SelectActivityResultByActivityIDAndRank  
(int activityID, int activityResultRank)  
+ List<ActivityResult>  
SelectActivityResultsByActivityID(int  
activityID)  
+ int  
UpdateActivityResultByActivityIDAndAct  
ivityResultName(int activityID, string  
oldActivityResultName, string  
newActivityResultName)  
+ int  
DeleteActivityResultByActivityResultNa  
meAndActivityID(string  
activityResultName, int activityID)
```

7.3.1.19 IEventDateAccessor

IEventDateAccessor

```
+ int InsertEventDate(EventDate  
eventDate)  
+ EventDate  
SelectEventDateByEventDateIDAndEven  
tID(DateTime eventDateID, int eventID)  
+ List<EventDate>  
SelectEventDatesByEventID(int eventID)  
+ int UpdateEventDate(EventDate  
oldEventDate, EventDate  
newEventDate)  
+ int  
DeleteEventDateByEventDateIDAndEven  
tID(DateTime eventDateID, int eventID)  
+ List<EventDateVM>  
SelectEventDatesByUserIDAndDate(in  
userID int, in eventDate DateTime)  
+ int  
DeactivateEventDateByEventDateIDAnd  
EventID(DateTime eventDateID, int  
eventID)
```

7.3.1.20 IEmergencyExitAccessor

IEmergencyExitAccessor

```
+ int InsertEmergencyExit(int
    emergencyProcedureID, string
    emergencyExitDescription, Sublocation
    sublocation)
+ List<EmergencyExit>
SelectEmergencyExitsbyEventID(int
    eventID)
+ List<EmergencyExit> EmergencyExit
SelectEmergencyExitsbyEmergencyProce
dureID(int emergencyProcedureID)
+ EmergencyExit
SelectEmergencyExitByEmergencyProce
dureIDAndSublocationID(int
    emergencyProcedureID, int
    sublocationID)
+ int
UpdateEmergencyExitByEmergencyExitI
D(int emergencyExitID, string
    oldEmergencyExitDescription, string
    newEmergencyExitDescription,
    Sublocation oldSublocation, Sublocation
    newSublocation)
+ int
DeleteEmergencyExitByEmergencyExitID
    (int emergencyExitID)
```

7.3.1.21 IEmergencyProcedureAccessor

IEmergencyProcedureAccessor

```
+ EmergencyProcedure
SelectEmergencyProcedureByEmergencyProcedureID(int
emergencyProcedureID)
+ int
UpdateEmergencyProcedureMessageByEventID(int eventID, string oldMessage,
string newMessage)
+ int
InsertEmergencyProcedureByEventID(int eventID, string oldMessage, string
newMessage, List<EmergencyUser>
oldEmergencyContacts,
List<EmergencyUser>
newEmergencyContacts, MusterPoint
oldMusterPoint, MusterPoint
newMusterPoint, List<EmergencyExit>
oldEmergencyExits, List<EmergencyExit>
newEmergencyExits)
+ int
DeleteEmergencyProcedureByEventID(in
t eventID)
```

7.3.1.22 IMusterPointAccesor

IMusterPointAccesor

```
+ int InsertMusterPoint(int
EmergencyProcedureID, string
emergencyMusterPointDescription,
Sublocation sublocation)
+ MusterPoint
SelectMusterPointByEmergencyProcedu
relID(int emergencyProcedureID)
+ MusterPoint
SelectMusterPointByEventID(int
eventID)
+int
UpdateMusterPointByEmergencyProced
ureID(int emergencyProcedureID, string
oldEmergencyMusterPointDescription,
string
newEmergencyMusterPointDescription,
Sublocation oldSublocation, Sublocation
newSublocation)
+
DeleteMusterPointByEmergencyProcedu
relID(int eventID)
```

7.3.1.23 INotificationGroupAccessor

INotificationGroupAccessor

```
+ int
InsertNotificationGroupByEventID(Event
eventID, List<User> users)
+ NotifcationGroup
SelectNotificationGroupByNotifcationGr
oupID(int notificationGroupID)
+ int
UpdateNotificationGroupUsersByNotifica
tionGroupID(int notificationGroupID,
List<User> oldUsers, List<User>
newUsers)
+ int
DeleteNotificationGroupByNotifcationGr
oupID(int notificationGroupID)
```

7.3.2 Volunteer Data Access Interfaces

7.3.2.1 IFavoriteVolunteersAccessor

IFavoriteVolunteersAccessor

```
+ SelectAllVolunteersByUserID(in UserID int) : List<Volunteer>
+ SelectVolunteerByVolunteerID(in VolunteerID int) : Volunteer
+ InsertVolunteer(in UserID int, in VolunteerID int)
+ DeleteVolunteerByVolunteerID(in UserID int, in VolunteerID int)
```

7.3.2.2 IRoleAccessor

IRoleAccessor

```
+ SelectAllRoles() : List<Role>
+ SelectRolesByRoleID(in RoleID string) : List<Role>
```

7.3.2.3 ISkillSetAccessor

ISkillSetAccessor

```
+ SelectAllSkillSets() : List<SkillSet>
+ SelectSkillSetBySkillSetID(in SkillSetID string) : SkillSet
+ UpdateSkillSetDescriptionBySkillSetID
  (in SkillSetID string, in OldSkillSetDescription string, in
  NewSkillSetDescription string)
```

7.3.2.4 IVerificationRequestsAccessor

IVerificationRequestsAccessor

```
+ SelectAllVerificationRequest() :  
List<Request>  
+SelectAllVerificationRequestsByVolunteerID  
(in VolunteerID int) : List<Request>  
+ SelectAllVerificationRequestsBySkillSetID(  
in SkillSetID string) : List<Request>  
+ InsertVerificationRequest(in VolunteerID  
int, in SkillSetID string)  
+ UpdateVerificationRequestByVolunteerID(  
in VolunteerID int, in SkillSetID string, in  
OldAccepted boolean, in NewAccepted  
boolean)  
+ DeleteVerificationRequestByRequestID(in  
RequestID int)
```

7.3.2.5 IVolunteerAccessor

IVolunteerAccessor

```
+ SelectAllVolunteers() : List<Volunteer>
+ SelectVolunteersByRoleID(in RoleID
  string) : List<Volunteer>
+ SelectVolunteersByName(in
  GivenName string, in FamilyName string)
  : List<Volunteer>
+ SelectVolunteersByGivenName(in
  GivenName string) : List<Volunteer>
+ SelectVolunteersByFamilyName(in
  FamilyName string) : List<Volunteer>
+ SelectVolunteersByAvailability(in
  AvailableDay date) : List<Volunteer>
+ SelectVolunteersByAvailableTime(in
  AvailableTimeStart time, in
  AvailableTimeEnd time) :
  List<Volunteer>
+ SelectVolunteersBySkillSetID(in
  SkillSetID string) : List<Volunteer>
+ SelectVolunteersByRating(in Rating
  int) List<Volunteer>
+ SelectVolunteersByState(in State
  string) : List<Volunteer>
+ SelectVolunteersByCity(in City string) :
  List<Volunteer>
+ InsertVolunteer(in UserID int)
+ UpdateVolunteerIDByUserID(in UserID
  int, in OldVolunteerID int, in
  NewVolunteerID int)
+ UpdateVolunteerWithEventID(int
  EventID, int UserID)
+ DeactivateVolunteerByVolunteerID(in
  VolunteerID int)
+ SelectVolunteerByUserID(in UserID int)
  : Volunteer
```

7.3.2.6 IVolunteerApplicationsAccessor

IVolunteerApplicationsAccessor

```
+ SelectAllVolunteerApplications() :  
List<Application>  
+ SelectVolunteerApplicationByRequestID(in  
RequestID int) : Application  
+ InsertVolunteerApplication(in UserID int)  
+ UpdateVolunteerApplicationByRequestID(  
in RequestID int, in OldAccepted boolean, in  
NewAccepted boolean)  
+ DeleteVolunteerApplicationByRequestID(in  
RequestID int)  
+ int InsertVolunteerApplication(int userID,  
Availability availability)
```

7.3.2.7 IVolunteerAvailabilityAccessor

IVolunteerAvailabilityAccessor

```
+ InsertVolunteerAvailability(in  
VolunteerID int, in AvailableDay date, in  
AvailableTimeStart time, in  
AvailableTimeEnd time)  
+ DeleteVolunteerAvailabilityByAvailabilityID(in AvailabilityID int)  
  
+ UpdateAvailableDayByAvailabilityID  
(int AvailabilityID int, in OldAvailableDay  
date, in NewAvailableDay date)  
  
+ UpdateTimeStartByAvailabilityID(in  
AvailabilityID int, in  
OldAvailableStartTime time, in  
NewAvailableStartTime time)  
  
+ UpdateTimeEndByAvailabilityID(in  
AvailabilityID int, in  
OldAvailableEndTime time, in  
NewAvailableEndTime time)  
  
+ SelectDaysAvailableByVolunteerID(in  
VolunteerID int) : List<AvailableDay>
```

7.3.2.8 IVolunteerProofAccessor

IVolunteerProofAccessor

- + InsertVolunteerProof(in ProofID int, in VolunteerID int, in ProofImage string, in ProofDescription string)
- + DeleteVolunteerProofByProofID(in ProofID int)

- + UpdateProofImageByProofID(in ProofID int, in OldProofImage varchar, int NewProofImage varchar)

- + UpdateProofDescriptionByProofID(in ProofID int, in OldProofDescription string, int NewProofDescription string)

- + SelectVolunteerProofByVolunteerID(in VolunteerID int) : List<VolunteerProof>

7.3.2.9 IVolunteerRecordAccessor

IVolunteerRecordAccessor

- + InsertVolunteerRecord(in VolunteerID int, in EventID int, in LocationID int, in Rating int, in EventDateID datetime)
- + DeleteVolunteerRecordByVolunteerRecordID(in VolunteerRecordID int)
- + SelectVolunteerRecordByVolunteerID(in VolunteerID int) : List<VolunteerRecord>

7.3.2.10 IVolunteerRequestsAccessor

IVolunteerRequestsAccessor

```
+SelectAllRequestsForVolunteerByVolunteerID(in VolunteerID int) : List<Event>
+ SelectAllRequestsForEventByEventID(in EventID int) : List<Volunteer>
+ SelectVolunteerRequestsByEventID(int EventID) List<VolunteerRequestVM>
+ SelectRequestByRequestID(in RequestID int) : Request
+ InsertVolunteerRequest(in VolunteerID int, in EventID int, in VolunteerResponse boolean, in EventResponse boolean)
+ UpdateVolunteerRequest(in OldVolunteerResponse boolean, in OldEventResponse boolean, in NewVolunteerResponse boolean, in NewEventResponse boolean)
+ DeleteVolunteerRequestByRequestID(in RequestID int)
```

7.3.2.11 IVolunteerReviewsAccessor

IVolunteerReviewsAccessor

```
+ InsertVolunteerReview(in VolunteerID int, in Rating int, in Comments string)
+ UpdateVolunteerReviewRatingByReviewID(in ReviewID int, in OldRating int, in NewRating int)
+ UpdateVolunteerReviewCommentByReviewID(in ReviewID int, in OldComment string, in NewComment string)
+ DeleteVolunteerRatingByReviewID(in ReviewID int)
+ SelectVolunteerReviewsByVolunteerID (in VolunteerID int) : List<Reviews>
+ SelectAllVolunteerReviews() : List<Reviews>
```

7.3.2.12 **IVolunteerSkillSetAccessor**

IVolunteerSkillSetAccessor

- + InsertVolunteerSkillSetByVolunteerID (in VolunteerID int, in SkillSetID string)
- + DeleteVolunteerSkillSetByVolunteerID (in VolunteerID int, in SkillSetID string)
- + SelectSkillSetByVolunteerID (in VolunteerID int) : List<VolunteerSkillSet>

7.3.2.13 **IVolunteerSuppliesAccessor**

IVolunteerSuppliesAccessor

- + UpdateSupplyNameBySupplyID (in SupplyID int, in OldSupplyName string, in NewSupplyName string)
- + UpdateSupplyAmountBySupplyID (in SupplyID int, in OldSupplyAmount int, in NewSupplyAmount int)
- + UpdateSupplyDescriptionBySupplyID (in SupplyID int, in OldSupplyDescription string, in NewSupplyDescription string)
- + InsertVolunteerSupplies (in VolunteerID int, in SupplyName string, in SupplyAmount int, in SupplyDescription string)
- + DeactivateSuppliesBySupplyID (in SupplyID int)
- + SelectSuppliesByVolunteerID (in VolunteerID int) : List<VolunteerSupplies>

7.3.3 Supplier Data Access Interfaces

7.3.3.1 ISupplierAccessor

SupplierAccessor

```
+ SelectSupplierByUserID(int userID) :  
Supplier  
+ SelectSupplierBySupplierID(int  
supplierID) : Supplier  
+ SelectSuppliersByState(string state) :  
List<Supplier>  
+ SelectAllSuppliers() : List<Supplier>  
+ SelectSuppliersByCityAndState(string  
state, string state) : List<Supplier>  
+ SelectSuppliersBySupplierType(string  
typeName) : List<Supplier>  
+ SelectSupplierByPhone(string phone) :  
Supplier  
+ InsertSupplierByUserID(int userID)  
+ SelectActiveSuppliers() : List<Supplier>  
+SelectSupplierAvailabilityBySupplierIDA  
ndDate(int supplierID, DateTime date) :  
List<Availability>  
+ SelectSupplierByEmail(string email) :  
Supplier  
+ SelectSupplierReviewsBySupplierID(int  
supplierID)  
+SelectSupplierAvailabilityExceptionByS  
upplierIDAndDate(int supplierID,  
DateTime date) : List<Availability>  
+ SelectSupplierBySupplierName(string  
supplierName) : Supplier  
+InsertSupplierRequest(Supplier  
supplier) : int  
+ bool UpdateSupplier(Supplier  
oldSupplier, Supplier newSupplier)  
List<DateTime>  
SelectSupplierAvailabilityForNextThree  
Months(int supplierID)  
+SelectSupplierTagsbySupplierID(int  
supplierID)  
+SelectSupplierImagesBySupplierID(int  
supplierID)  
+SelectSuppliersNoUserID():  
List<Supplier>  
+ List<Supplier>  
SelectUnapprovedSuppliers()  
+ int ApproveSupplier(int supplierID)  
+ int DisapproveSupplier(int supplierID)  
+ int RequeueSupplier(int supplierID)  
+ int InsertSupplierReview(Reviews  
review)
```

7.3.3.2 IServiceAccessor

```
IServiceAccessor

+List<Service>
SelectServicesBySupplierID(int
supplierID)
+int UpdateService(int serviceID,
Service oldService, Service newService)
+int
DeactivateServiceByOldService(Service
oldService)
+int InsertService(Service newService)
+int DeleteService(int serviceID)
```

7.3.3.3 ILocationAccessor

```
LocationAccessor

+ int SelectActiveLocations()

+
SelectLocationByLocationNameAndAddress(string, string)
+ Location
SelectLocationByLocationID(int)
+ List<LocationReview>
SelectLocationReviews(int)
+ int
DeactivateLocationByLocationID(int,
Location)
+List<String>
SelectLocationRulesByLocationID(int)
+List<String>
SelectLocationImagesByLocationID(int)
+ bool InsertLocation(string, string,
string, string, string)
+int AddLocationImageByLocationID(int,
String)
+ Location
SelectLocationBioInformationByLocation
ID(int)
+int
DeactivateLocationImageByImageID(int,
String)
+List<Availability>
SelectLocationAvailabilityByLocationIDA
ndDate(int locationID, DateTime date)
+List<Availability>
SelectLocationAvailabilityExceptionByLo
cationIDAndDate(int locationID,
DateTime date)
+List<string>SelectTagsByLocationID(int
locationID)
+int
UpdateLocationBioByLocationID(Location
oldLocation, Location newLocation)
+List<LocationAvailability>
SelectLocationAvailability(int)
+ int InsertLocationReview(Reviews
review)
```

7.3.3.4 ISublocationAccessor

```
SublocationAccessor

+int InsertSublocationByLocationID(int
locationID, string sublocationName,
string sublocationDesc);
+Sublocation
SelectSublocationBySublocationID(int)
+int UpdateSublocation(Sublocation,
Sublocation)
+int
DeactivateSublocationBySublocationID
(int sublocationID)
+???
SelectScheduleBySublocationID(int)
+List<String>
SelectMapsBySublocationID(int)
+bool CreateSublocationRule(int,
String)
+int UpdateSublocationRule(int, String,
String)
+int DeactivateSublocationRule(int,
String)
+int
CreateSublocationMapBySublocationI
D(int, String)
+int
DeactivateSublocationMapByMapID(in
t, String)
+List<Sublocation>
SelectSublocationsByLocationID(int)
```

7.3.3.5 ISupplyAccessor

```
ISupplyAccessor

+List<SupplyVM>
SelectSuppliesBySublocationID(int
sublocationID)
+int UpdateSupplyBySublocationID(int
sublocationID, Supply oldSupply, Supply
newSupply)
+int
DeactivateSupplyByOldSupply(Supply
oldSupply)
+int InsertSupply(Supply supply)

+int DeleteSupplyByOldSupply(Supply
oldSupply)
```

7.3.3.6 IEntranceAccessor

IEntranceAccessor

```
+ List<Entrance>
SelectEntrancesByLocationID(int
locationID)
+ int UpdateEntranceByEntranceID(int
entranceID, Entrance oldEntrance,
Entrance newEntrance)
+ int InsertEntrance(int locationID, string
entranceName, string description)
+ int
DeactivateEntranceByEntranceID(int
entranceID)
```

7.3.3.7 IParkingLotAccessor

IParkingLotAccessor

```
+List<ParkingLotVM>
SelectParkingLotByLocationID(int
locationID)
+int UpdateParkingLotByLotID(int
parkingLotID, ParkingLot oldParkingLot,
ParkingLot newParkingLot)
+ int InsertParkingLot(ParkingLot
parkingLot)
+ bool DeleteParkingLotByLotID(int
lotID)
+bool UserCanEditParkingLot(int userID)
```

7.3.3.8 IZipAccessor

IZipAccessor

```
+SelectAllZips() : List<Zip>
+SelectCityAndStateByZipCode(string
zipcode) : Zip
```

7.3.4 Attendee Data Access Interfaces

7.3.4.1 IContactsAccessor

IContactsAccessor

```
+ DeleteContact(int userID) : int
+ InsertContact(int userID) : bool
+ SelectContactsByStatus(string status) :
List<User>
+ SelectAllContacts(): List<User>
```

7.3.4.2 IGroupAccessor

IGroupAccessor

```
+ DeleteGroup(int groupID) : int
+ UpdateGroupName(int groupID, string
name) : bool
+ SelectGroupByGroupID(int groupID) :
Group
+ DeleteGroup(int groupID) : int
+ InsertNewGroup(string name): Group
+ InsertUserInGroup(int userID, int
groupID) : bool
```

7.3.4.3 IGroupMembersAccessor

```
IGroupMembersAccessor

+ SelectGroupByGroupName(string
  GroupName) : Group
+ SelectAllGroupMembers(string
  GroupName) : List<User>
+ DeleteUserFromGroup(int GroupID, int
  userID) : bool
+ InsertGroupMessage(Message
  message, int groupID) : bool
+ DeleteGroupMessage(int messageID,
  int groupID) : int
+ SelectGroupMessageByMessageID(int
  messageID, int groupID) : Message
+ SelectAllGroupMessages(int GroupID) :
  List<Message>
```

7.3.4.4 IUserAccessor

```
IUserImageAccessor

+ InsertUserImage(int userID, Image
  image) : bool
+ SelectImagesByUserID(int UserID) :
  List<UserImage>
+ DeleteUserImage(int ImageID) : int
+ UpdateImagePath(int ImageID, string
  imagePath) : int
+ SelectImage(int imageID) : UserImage
```

7.3.4.5 IUserImageAccessor

```
IUserImageAccessor

+ InsertUserImage(int userID, Image
  image) : bool
+ SelectImagesByUserID(int UserID) :
  List<UserImage>
+ DeleteUserImage(int ImageID) : int
+ UpdateImagePath(int ImageID, string
  imagePath) : int
+ SelectImage(int imageID) : UserImage
```

7.3.4.6 IMessageAccessor

IMessageAccessor

```
+ InsertMessage(int senderID, int
recieverID, Message message) : bool
+ DeleteMessage(int userID, int
messageID)
+ SelectMessageByUser(int userID) :
Message
+ UpdateMessage(int messageID,
Message message) : int
```

7.3.4.7 IInterestAccessor

IInterestAccessor

```
+ InsertUserInterestTag(int userID, int
interestID) : bool
+ DeleteUserInterestTag(int userID, int
InterestID) : int
+ SelectEventsByInterests(int interestID) :
List<Event>
```

7.3.4.8 IEventRsvpAccessor

```
IEventRsvpAccessor  
+ InsertEventRSVP (int eventID, int  
userID) : bool  
+ UpdateEventRSVP (int eventID, int  
userID, bool attendedEvent)
```

7.3.4.9 IReviewAccessor

```
IReviewAccessor  
+ InsertReview(Review review) : bool  
+ UpdateReview(int reviewID, Review  
review) : int  
+ DeleteReview(int reviewID) : bool
```

7.3.4.10 ISavedEventAccessor

```
ISavedEventAccessor  
+ SelectSavedEventByEventName(string  
eventName, int userID) : SavedEvent  
+ SelectAllSavedEvents(int userID) :  
List<SavedEvents>  
+ DeleteSavedEvent(int eventID, int  
userID) : bool
```

7.4 Manager Interfaces

7.4.1 Event Manager Interfaces

7.4.1.1 IEventManager

IEventManager

```
+ int CreateEvent(string eventName,
  string eventDescription, decimal
  totalBudget)
+ int CreateEventReturnsEventID(string
  eventName, string eventDescription,
  decimal totalBudget, int userID)
+ EventVM RetrieveEventByEventID(int
  eventID)
+ Event
RetrieveEventByEventNameAndDescripti
on(string eventName, string
  eventDescription)
+List<EventVM>
RetrieveEventListForUpcomingDates();
+List<EventVM>
RetrieveEventListForUpcomingAndPastD
ates();
+List<EventVM>
RetrieveEventListForPastDates();
+List<EventVM>
RetrieveEventListForUpcomingDatesFor
User(int userID);
+List<EventVM>
RetrieveEventListForPastDatesForUser(i
nt userID);
+List<EventVM>
RetrieveEventListForPastAndUpcomingD
atesForUser(int userID);
+ bool
CheckUserEditPermissionForEvent(int
  eventID, int userID)
+ List<EventVM> RetrieveEventList()
+ List<EventVM>
RetrieveEventListByUserID(int userID)
+ int UpdateEventNameByEventID(int
  eventID, string oldEventName, string
  newEventName)
+ int
UpdateEventDescriptionByEventID(int
  eventID, string oldEventDescription,
  string newEventDescription)
+ int UpdateEventByEventID(int eventID,
  string oldEventName, string
  newEventName, string
  oldEventDescription, string
  newEventDescription, decimal
  oldTotalBudget, decimal
  newTotalBudget)
+ int DeactivateEventByEventID(int
  eventID)
```

7.4.1.2 IActivityManager

IActivityManager

```
+ int CreateActivity(Activity activity)
+ ActivityVM
RetrieveActivityByActivityID (int
activityID)
+ List<ActivityVM>
RetrieveActivitiesByEventID (int eventID)
+ List<ActivityResultVM>
RetrieveActivityResultsByActivityID(int
activityID)
List<ActivityVM>
RetriveActivitiesUpcomingDates
();
List<ActivityVM>
RetriveUserActivitiesUpcomingDates(in
t userID);
List<ActivityVM>
RetriveActivitiesPastDates();
List<ActivityVM>
RetriveActivitiesPastAndUpcomingDate
s();
List<ActivityVM>
RetriveUserActivitiesPastDates(int
userID);
List<ActivityVM>
RetriveUserActivitiesPastAndUpcoming
Dates(int userID);
+ActivityVM
RetrieveActivityVMByActivityID(activityI
D)
+ int
UpdateActivityNameByActivityID(int
activityID, string oldActivityName, string
newActivityName)
+List<Activity>
RetriveActivitiesBySublocationID(int
sublocationID);
```

IActivityManager Continue

```
+ int
UpdateActivityDescriptionByActivityID(in
t activityID, string oldActivityDescription,
string newActivityDescription)
+ int
UpdateActivityStartTimeByActivityID(int
activityID, DateTime oldStartTime,
DateTime newStartTime)
+ int
UpdateActivityEndTimeByActivityID(int
activityID, DateTime oldEndTime,
DateTime newEndTime)
+ int
UpdateActivityPublicByActivityID(int
activityID, bool public)
+ int
UpdateActivityImageNameByActivityID(i
nt activityID, string
oldActivityImageName, string
newActivityImageName)
+ int UpdateActivityByActivityID(string
oldActivityName, string
newActivityName, string
oldActivityDescription, string
newActivityDescription, bool public,
DateTime oldStartTime, DateTime
newStartTime, DateTime oldEndTime,
DateTime newEndTime, Sublocation
oldSublocation, Sublocation
newSublocation, EventDate
oldEventDate, EventDate
newEventDate)
+ int
UpdateActivitySubLocationByActivityID(i
nt activityID, SubLocation sublocation)
+ int DeactivateActivityByActivityID(int
activityID)
+ int DeleteActivityByActivityID (int
activityID)
-memberName
```

7.4.1.3 IEventDateManager

IEventDateManager

```
+ bool CreateEventDate(EventDate
eventDate)
+ EventDate
RetrieveEventDateByEventDateIDAndEv
entID(DateTime EventDateID, int
EventID)
+ List<EventDate>
RetrieveEventDateByEventID(int EventID
+ int UpdateEventDate(EventDate
oldEventDate, EventDate
newEventDate)
+ int
DeleteEventDateByEventDateIDandEven
tID(DateTime eventDateID, int eventID )
+ int
DeactivateEventDateByEventDateIDand
EventID(DateTime eventDateID, int
eventID )
+ List<EventDateVM>
SelectEventDatesByUserIDAndDate(in
userID int, in eventDate DateTime)
```

7.4.1.4 IEventUploadManager

IEventUploadManager

```
+ int CreateEventUpload(string
eventFileLocation, string title, string
description)
+ EventUpload
RetrieveEventUploadByEventUploadID(i
nt eventUploadID)
+ List<EventUpload >
RetrieveEventUploadByEventID(int
eventID)
+ int
UpdateEventUploadEventFileNameByEv
entUploadID(int eventUploadID, string
oldEventFileName , string
newEventFileName)
+ int
UpdateEventUploadEventTitleAndDescri
ptionByEventUploadID(int
eventUploadID, string
oldEventUploadTitle, string
newEventUploadTitle, string
oldEventDescription,
newEventDescription)
+ int
UpdateEventUploadByEventUploadID(in
t eventUploadID, string
oldEventFileName, string
newEventFileName, string oldTitle,
string newTitle, string oldDescription,
string newDescription)
+ int
DeactivateEventUploadRecordByEventU
ploadID(int eventUploadID)
+ int
DeleteEventUploadRecordByEventUploa
dID(int eventUploadID)
```

7.4.1.5 IEventPictureManager

IEventPictureManager
<pre>+ int CreateEventPictureByEventID(int eventID, string eventPictureName, string title, string caption, string description, bool isMainDisplay) + EventPicture RetrieveEventPictureByEventPictureID(i nt eventPictureID) + List<EventPicture> RetrieveEventPicturesByEventID(int eventID) + List<EventPicture> RetrieveEventPicturesByEventIDForMain Display(int eventID) + int UpdateEventPictureNameByEventPictur eID(int eventPictureID, int oldEventPictureName, int newEventPictureName) + int UpdateEventPictureMainDisplayByEvent PictureID(int eventPictureID, bool isMainDisplay) + int UpdateEventPictureTitleCaptionAndDes criptionByEventPictureID(int eventPictureID, string oldEventPictureName, string newEventPictureName, string oldTitle, string newTitle, string newCaption, string oldCaption, string oldDescription, string newDescription, bool isMainDisplay) + int UpdateEventPicture(int eventPictureID, string oldEventPictureName, string newEventPictureName, string oldTitle, string newTitle, string newCaption, string oldCaption, string oldDescription, string newDescription, bool isMainDisplay) + int DeactivateEventPictureByEventPictureID (int eventPictureID) + int DeleteEventPictureByEventPictureID(int eventPictureID)</pre>

7.4.1.6 IEmergencyProcedureManager

IEmergencyProcedureManager

```
+ int
CreateEmergencyProcedureByEventID(in
t eventID, string message,
List<EmergencyUserVM>
emergencyContacts, MusterPoint
musterPoint, List<EmergencyExit>
emergencyExits)
+ EmergencyProcedureVM
RetrieveEmergencyProcedureByEventID(
int eventID)
+ EmergencyProcedureVM
RetrieveEmergencyProcedureByEmergen
cyProcedureID(int
emergencyProcedureID)
+ int
UpdateEmergencyProcedureMessageBy
EventID(int eventID, string oldMessage,
string newMessage)
+ int
CreateEmergencyProcedureByEventID(in
t eventID, string oldMessage, string
newMessage, List<EmergencyUser>
oldEmergencyContacts,
List<EmergencyUser>
newEmergencyContacts, MusterPoint
oldMusterPoint, MusterPoint
newMusterPoint, List<EmergencyExit>
oldEmergencyExits, List<EmergencyExit>
newEmergencyExits)
+ int
DeleteEmergencyProcedureByEventID(in
t eventID)
```

7.4.1.7 IMusterPointManager

IMusterPointManager

```
+ int CreateMusterPoint(int  
EmergencyProcedureID, string  
emergencyMusterPointDescription,  
Sublocation sublocation)  
+ MusterPoint  
RetrieveMusterPointByEmergencyProce  
dureID(int emergencyProcedureID)  
+ MusterPoint  
RetrieveMusterPointByEventID(int  
eventID)  
+int  
UpdateMusterPointByEmergencyProced  
ureID(int emergencyProcedureID, string  
oldEmergencyMusterPointDescription,  
string  
newEmergencyMusterPointDescription,  
Sublocation oldSublocation, Sublocation  
newSublocation)  
+int  
DeleteMusterPointByEmergencyProced  
ureID(int eventID)
```

7.4.1.8 IEmergencyExitManager

IEmergencyExitManager

```
+ int CreateEmergencyExit(int
  emergencyProcedureID, string
  emergencyExitDescription, Sublocation
  sublocation)
+ List<EmergencyExit>
  RetrieveEmergencyExitsbyEventID(int
  eventID)
+ List<EmergencyExit> EmergencyExit
  RetrieveEmergencyExitsbyEmergencyPr
  ocedureID(int emergencyProcedureID)
+ EmergencyExit
  RetrieveEmergencyExitByEmergencyPro
  cedureIDAndSublocationID(int
  emergencyProcedureID, int
  sublocationID)
+ int
  UpdateEmergencyExitByEmergencyExitI
  D(int emergencyExitID, string
  oldEmergencyExitDescription, string
  newEmergencyExitDescription,
  Sublocation oldSublocation, Sublocation
  newSublocation)
+ int
  DeleteEmergencyExitByEmergencyExitID
  (int emergencyExitID)
```

7.4.1.9 IActivityResultManager

IActivityResultManager

```
+ int CreateActivityResultByActivityID(  
    string activityID, string ActivityName,  
    string activityResultName)  
+ ActivityResult  
RetrieveActivityResultByActivityIDAndRa  
nk(int activityID, int activityResultRank)  
+ List<ActivityResult>  
RetrieveActivityResultsByActivityID(int  
    activityID)  
+List<ActivityVM>  
RetrieveActivitiesByEventIDAndEventDat  
e(int eventID, DateTime eventDateID);  
+ int  
UpdateActivityResultByActivityIDAndAct  
ivityResultName(int activityID, string  
    oldActivityResultName, string  
    newActivityResultName)  
+ int  
DeleteActivityResultByActivityIDAndAct  
ivityResultName(int activityID, string  
    activityResultName)
```

7.4.1.10 INotificationGroupManager

INotificationGroupManager

```
+ int  
CreateNotificationGroupByEventID(Event  
    eventID, List<User> users)  
+ NotificationGroupVM  
RetrieveNotificationGroupByNotification  
    GroupID(int notificationGroupID)  
+ int  
UpdateNotificationGroupUsersByNotifica  
    tionGroupID(int notificationGroupID,  
    List<User> oldUsers, List<User>  
    newUsers)  
+ int  
DeleteNotificationGroupByNotificationGr  
oupID(int notificationGroupID)
```

7.4.1.11 ITaskManager

ITaskManager

```
+ int RetrieveTaskByTaskID(int taskID)
+ int
RetrieveAssignedUserByTaskAssignment
ID(int taskAssignmentID)
+ List<Task>
RetrieveAllActiveTasksByEventID(int
eventID)
+ List<Task>
RetrieveTasksByDueDate(DateTime
dueDate)
+ List<Task>
RetrieveTasksByPriority(Priority
Description)
+ Priority
RetrieveTaskPriorityByTaskID(int TaskID)
+ List<Priority> RetrieveAllPriorities()
+ bool
EditTaskProgressByCompletionDate(Dat
eTime completionDate)
+ Priority EditTaskPriorityByTaskID(int
taskID)
+ bool EditTask(Task oldTask, Task
newTask)
+ bool AddTask(Task newTask)
+ bool
DeactivateTaskByCompletionDate(Dat
eTime completionDate)
+ List<TaskAssignmentVM>
RetrieveTaskAssignmentsByTaskID(int
TaskID)
+ bool UserCanEditDeleteTask(int
TaskID)
+ bool RemoveTaskByTaskID(int TaskID)
```

7.4.1.12 IQuestionManager

IQuestionManager

```
+ int RetrieveQuestionByPostID(int
postID)
+ List<QuestionVM>
RetrieveListOfQuestionsByDueDateAndE
ventID(LocalDate date, int eventID)
+ List<QuestionVM>
RetrieveQuestionByQuestionTypeID(int
questionTypeID)
+ List<QuestionVM>
RetrieveQuestionsByAnswer(bool
isAnswered)
+ int EditQuestionsByQuestionTypeID(int
questionTypeID, int oldQuestionType,
int newQuestionType)
+ bool EditQuestionByAnswered(int
questionID, bool oldIsAnswered, bool
newIsAnswered, DateTime
dateAnswered)
+ string EditQuestionsByContent(int
PostID, int QuestionID, string
OldContent, string NewContent)
+ QuestionVM
CreateNewQuestionsByPostID(int
postID, int questionTypeID, bool
isRankingList, bool isAnswered)
+ bool
DeactivateQuestionByEventEndDate(int
eventID, int questionID, datetime
endDateTime)
```

7.4.1.13 IPollManager

IPollManager

```
+ int CreatePollByPostID(int postID,
string title, string description, DateTime
dueDate)
+ PollVM RetrievePollByPostID(int
postID)
+ int UpdatePollTitleByPollID(int postID,
string oldPollTitle, string newPollTitle)
+ int UpdatePollDescriptionByPostID(int
postID, string oldPollDescription, string
newPollDescription)
+ int UpdatePollDueDateByPostID(int
postID, string oldPollDueDate, string
newPollDueDate)
+ int UpdatePollByPostID(int postID,
string oldPollName, string newPollName,
string oldPollDescription, string
newPollDescription, DateTime
oldDueDate, DateTime newDueDate)
+ int DeactivatePollByPollID(int pollID)
+ int DeletePollByPollID(int pollID)
```

7.4.1.14 IAnswerManager

IAnswerManager

```
+ List<AnswerVM>
RetrieveAnswersByQuestionID(int
questionID)
+ List<AnswerVM>
RetrieveAnswersByDateAnswered(DateTime
dateAnswered)
+ string EditAnswerByDescription(int
answerID, string oldDescription, string
newDescription)
+ DateTime
EditAnswerByDateAnswered(int
answerID, DateTime oldAnswerDate,
DateTime newAnswerDate)
+ AnswerVM
CreateNewAnswerByQuestionID(int
questionID, string description, DateTime
answerDate)
+ bool
DeactivateAnswerByEventEndDate(int
eventId, int questionID, int answerID,
DateTime endDate)
```

7.4.1.15 **IVoteManager**

IVoteManager

```
+ int RetrieveVoteIDByPollID(int pollID)
+ List<Vote> RetrieveVoteByRank(bool
  isRanked, int rank)
+ Vote EditVoteByVoteSelection(Vote
  vote, string oldVote, string newVote)
+ bool
DeactivateVoteByEventEndDate(int
  eventID, int voteID, DateTime
  endTime)
```

7.4.1.16 **IIInvitationManager**

IIInvitationManager

```
+ List<Invitation VM>
SelectInvitationsByEventID(int eventID)
+ int SelectInvitationsByAttending(bool
  attending)
+ int SelectInvitationsByUserID(int
  invitationID, int userID)
+ bool EditInvitationsByAttending(int
  invitationID, bool oldIsAttending, bool
  newIsAttending)
+ InvitationVM
CreateNewInvitationsByEventID(int
  eventID, int userID, int roleID, bool
  attending)
```

7.4.1.17 **IBudgetManager**

IBudgetManager

```
+bool CreateBudget(BudgetVM
  newBudget)
+Budget RetrieveBudgetByEventID(int
  eventID)
```

7.4.1.18 **IBudgetCategoryManager**

IBudgetCategoryManager

```
+List<BudgetCategoryVM>SelectBudget  
CategoriesByBudgetID(int budgetID)  
+int  
DeleteBudgetCategoryByCategoryID(int  
categoryID )  
+int CreateBudgetCategory(decimal  
plannedAmount, string name)  
+int  
UpdateBudgetCategoryByCategoryID(int  
categoryID, decimal newAmount,  
decimal oldAmount, string oldName,  
string newName)
```

7.4.1.19 **IForumManager**

IForumManager

```
+int CreateForum(string forumName, string  
forumDescription)  
+int UpdateForumbyForumID(int forumID,  
string oldForumName, newForumName,  
oldForumDescription, newForumDescription)  
+int DeleteForum(int forumID)  
+ForumVM SelectForumByEventID(int  
eventID)
```

7.4.1.20 IMessageBoardManager

```
IMessageBoardManager

+List<MessageBoardsVM>Select
MessageBoardsByForumID()
+MessageBoardVM
SelectMessageBoardByMessageBoardID(int
messageBoardID)
+int CreateMessageBoard(string name, string
description)
+int
UpdateMessageBoardByMessageBoardID(int
MessageBoardID, string oldName, string
newName, string oldDescription, string
newDescription)
+int
DeleteMessageBoardByMessageBoardID(int
messageBoardID)
+UserVM
SelectMessageBoardUserByUserIDAndMessage
BoardID(int userID, int messageBoardID)
+List<UserVM>SelectMessageBoardUsersByU
serIDAndMessageBoardID(int userID, int
messageBoardID)
+int
DeleteMessageBoardUserByUserIDAndMessage
BoardID(int userID, int messageBoardID)
```

7.4.1.21 IBudgetLineManager

```
IBudgetLineManager

+List<BudgetLineVM>SelectBudgetLines
ByBudgetCategoryID(int
budgetCategoryID)
+int
DeleteBudgetLineByBudgetLineID(int
budgetLineID )
+int CreateBudgetLine(decimal amount,
string item)
+int UpdateBudgetLine(decimal
newAmount, decimal oldAmount, string
oldItem, string newItem)
```

7.4.1.22 IPostManager

IPostManager

```
+int CreatePost(string content)
+PostVM SelectPostByPostID(int postID)
+List<PostVM>SelectPostsByMessageBo
ardID(int messageBoardID)
+int UpdatePostByPostID(int postID,
string oldContent, new Content)
+int DeletePost(int postID)
+PostImageVM
SelectPostImageByPostID(int postID)
+Vote SelectVoteByPostID(int postID)
```

7.4.1.23 ICommentManager

ICommentManager

```
+int CreateComment(string content)
+List<CommentVM>SelectCommentsBy
PostID(int postID)
+int SelectCommentByID(commentID,
newContent, oldContent)
+int DeleteComment(int commentID)
```

7.4.1.24 ICommentLikeManager

ICommentLikeManager

```
+bool ToggleLike()
+List<Like>SelectLikesByCommentID(int
commentID)
```

7.4.1.25 IPostImageManager

IPostImageManager

```
+int insertPostImage()
+int DeletePostImageByPostImageID(int
PostImageID )
+PostImageVM
SelectPostImageByPostImageID(int
PostImageID)
```

7.4.1.26 IAfterActionReviewManager

IAfterActionReviewManager

```
+List<ReviewVM>
SelectReviewsByEventID()
+List<VolunteerVM>SelectVolunteersBy
EventID(int EventID)
+List
<AttendeeVM>SelectAttendeesByEventI
D(int eventID)
+List<Suppliers>SelectSuppliersByEventI
D(int eventID)
```

7.4.1.27 IEmailProvider

IVoteManager

```
+ void SendEmail(string subject, string
contentPlainText, string to)
```

7.4.2 Volunteer Manager Interfaces

7.4.2.1 IFavoriteVolunteersManager

IFavoriteVolunteersManager

```
+ RetrieveAllVolunteersByUserID(in UserID int) : List<Volunteer>
+ RetrieveVolunteerByVolunteerID(in VolunteerID int) : Volunteer
+ CreateVolunteer(in UserID int, in VolunteerID int)
+ RemoveVolunteerByVolunteerID(in UserID int, in VolunteerID int)
```

7.4.2.2 IRoleManager

IRoleManager

```
+ RetrieveAllRoles() : List<Role>
+ RetrieveRolesByRoleID(in RoleID string) : List<Role>
```

7.4.2.3 ISkillSetManager

ISkillSetManager

```
+ RetrieveAllSkillSets() : List<SkillSet>
+ RetrieveSkillSetBySkillSetID(in SkillSetID string) : SkillSet
+ EditSkillSetDescriptionBySkillSetID (in SkillSetID string, in OldSkillSetDescription string, in NewSkillSetDescription string)
```

7.4.2.4 IVerificationRequestsManager

IVerificationRequestsManager

```
+ RetrieveAllVerificationRequest() :  
List<Request>  
+RetrieveAllVerificationRequestsByVolunteer  
ID(in VolunteerID int) : List<Request>  
+RetrieveAllVerificationRequestsBySkillSetID(  
in SkillSetID string) : List<Request>  
+ CreateVerificationRequest(in VolunteerID  
int, in SkillSetID string)  
+ EditVerificationRequestByVolunteerID( in  
VolunteerID int, in SkillSetID string, in  
OldAccepted boolean, in NewAccepted  
boolean)  
+ RemoveVerificationRequestByRequestID(in  
RequestID int)
```

7.4.2.5 IVolunteerApplicationsManager

IVolunteerApplicationsManager

```
+ RetrieveAllVolunteerApplications() :  
List<Application>  
+RetrieveVolunteerApplicationByRequestID(i  
n RequestID int) : Application  
+EditVolunteerApplicationByRequestID( in  
RequestID int, in OldAccepted boolean, in  
NewAccepted boolean)  
+RemoveVolunteerApplicationByRequestID(i  
n RequestID int)  
+ bool CreateVolunteerApplication(int  
UserID, Availability availability)
```

7.4.2.6 IVolunteerAvailabilityManager

IVolunteerAvailabilityManager
<pre>+ CreateVolunteerAvailability(in VolunteerID int, in AvailableDay date, in AvailableTimeStart time, in AvailableTimeEnd time) +RemoveVolunteerAvailabilityByAvailabilityID(in AvailabilityID int) + EditAvailableDayByAvailabilityID (in Availability int, in OldAvailableDay date, in NewAvailableDay date) + RemoveTimeStartByAvailabilityID(in AvailabilityID int, in OldAvailableStartTime time, in NewAvailableStartTime time) + RemoveTimeEndByAvailabilityID(in AvailabilityID int, in OldAvailableEndTime time, in NewAvailableEndTime time) +RetrieveDaysAvailableByVolunteerID(in VolunteerID int) : List<AvailableDay></pre>

7.4.2.7 IVolunteerManager

IVolunteerManager

```
+ RetrieveAllVolunteers() : List<Volunteer>
+ RetrieveVolunteersByRoleID(in RoleID string) : List<Volunteer>
+ RetrieveVolunteersByName(in GivenName string, in FamilyName string) : List<Volunteer>
+ RetrieveVolunteersByGivenName(in GivenName string) : List<Volunteer>
+ RetrieveVolunteersByFamilyName(in FamilyName string) : List<Volunteer>
+ RetrieveVolunteersByAvailability(in AvailableDay date) : List<Volunteer>
+ RetrieveVolunteersByAvailableTime(in AvailableTimeStart time, in AvailableTimeEnd time) : List<Volunteer>
+ RetrieveVolunteersBySkillSetID(in SkillSetID string) : List<Volunteer>
+ RetrieveVolunteersByRating(in Rating int) List<Volunteer>
+ RetrieveVolunteersByState(in State string) : List<Volunteer>
+ RetrieveVolunteersByCity(in City string) : List<Volunteer>
+ CreateVolunteer(in UserID int)
+ EditVolunteerIDByUserID(in UserID int, in OldVolunteerID int, in NewVolunteerID int)
+ EditVolunteerEventIDByUserID(int EventID, int UserID)
+ DeactivateVolunteerByVolunteerID(in VolunteerID int)
+ RetrieveVolunteerByUserID(in UserID int) : Volunteer
```

7.4.2.8 IVolunteerProofManager

IVolunteerProofManager

- + CreateVolunteerProof(in ProofID int, in VolunteerID int, in ProofImage string, in ProofDescription string)
- + RemoveVolunteerProofByProofID(in ProofID int)
- + EditProofImageByProofID(in ProofID int, in OldProofImage varchar, int NewProofImage varchar)
- + EditProofDescriptionByProofID(in ProofID int, in OldProofDescription string, int NewProofDescription string)
- + RetrieveVolunteerProofByVolunteerID(in VolunteerID int) : List<VolunteerProof>

7.4.2.9 IVolunteerRecordManager

IVolunteerRecordManager

- + CreateVolunteerRecord(in VolunteerID int, in EventID int, in LocationID int, in Rating int, in EventDateID datetime)
- + RemoveVolunteerRecordByVolunteerRecordID(in VolunteerRecordID int)
- + RetrieveVolunteerRecordByVolunteerID(in VolunteerID int) : List<VolunteerRecord>

7.4.2.10 IVolunteerRequestsManager

IVolunteerRequestsManager

```
+RetrieveAllRequestsForVolunteerByVolunteerID(in VolunteerID int) : List<Event>
+ RetrieveAllRequestsForEventByEventID(in EventID int) : List<Volunteer>
+ RetrieveRequestByRequestID(in RequestID int) : Request
+ RetrieveVolunteerRequestsByEventID(int EventID) List<VolunteerRequestVM>
+ CreateVolunteerRequest(in VolunteerID int, in EventID int, in VolunteerResponse boolean, in EventResponse boolean)
+ EditVolunteerRequest(in OldVolunteerResponse boolean, in OldEventResponse boolean, in NewVolunteerResponse boolean, in NewEventResponse boolean)
+ RemoveVolunteerRequestByRequestID(in RequestID int)
```

7.4.2.11 IVolunteerReviewsManager

IVolunteerReviewsManager

```
+ CreateVolunteerReview(in VolunteerID int, in Rating int, in Comments string)
+ EditVolunteerReviewRatingByReviewID(in ReviewID int, in OldRating int, in NewRating int)
+ EditVolunteerReviewCommentByReviewID(in ReviewID int, in OldComment string, in NewComment string)
+ RemoveVolunteerRatingByReviewID(in ReviewID int)
+ RetrieveVolunteerReviewsByVolunteerID (in VolunteerID int) : List<Reviews>
+ RetrieveAllVolunteerReviews() : List<Reviews>
```

7.4.2.12 **IVolunteerSkillSetManager**

IVolunteerSkillSetManager

- + CreateVolunteerSkillSetByVolunteerID (in VolunteerID int, in SkillSetID string)
- + RemoveVolunteerSkillSetByVolunteerID (in VolunteerID int, in SkillSetID string)
- + RetrieveSkillSetByVolunteerID(in VolunteerID int) : List<VolunteerSkillSet>

7.4.2.13 **IVolunteerSuppliesManager**

IVolunteerSuppliesManager

- + EditSupplyNameBySupplyID(in SupplyID int, in OldSupplyName string, in NewSupplyName string)
- + EditSupplyAmountBySupplyID(in SupplyID int, in OldSupplyAmount int, in NewSupplyAmount int)
- + EditSupplyDescriptionBySupplyID (in SupplyID int, in OldSupplyDescription string, in NewSupplyDescription string)
- + CreateVolunteerSupplies(in VolunteerID int, in SupplyName string, in SupplyAmount int, in SupplyDescription string)
- + DeactivateSuppliesBySupplyID(in SupplyID int)
- + RetrieveSuppliesByVolunteerID(in VolunteerID int) : List<VolunteerSupplies>

7.4.2.14 IVolunteerNeedsManager

IVolunteerNeedManager

- + AddVolunteerNeed(in VolunteerNeed)
- + UpdateVolunteerNeed(in VolunteerNeed, in NewNumTotalVolunteers int)
- + DeleteVolunteerNeed(in VN VolunteerNeed)
- + UpdateCurrVolunteers(in VN VolunteerNeed, in NumCurrVolunteers int)
- + RetrieveVolunteerNeedByTaskID(in TaskID int) : VolunteerNeed

7.4.3 Supplier Manager Interfaces

7.4.3.1 SupplierManager

SupplierManager

```
+ Supplier RetrieveSupplierByUserID(int
userID)
+ Supplier
RetrieveSupplierBySupplierID(int
supplierID)
+ List<Supplier>
RetrieveSuppliersByState(string state)
+ Supplier
RetrieveSupplierBySupplierName(string
supplierName)
+ List<Supplier>
RetrieveSuppliersBySupplierType(string
typeName)
+ Supplier
RetrieveSupplierByPhone(string phone)
+ EditSupplier( Supplier
oldSupplier,Supplier newSupplier)
+ RemoveSupplier(Supplier supplier)
+ List <Supplier>
RetrieveSuppliersByCityAndState(string
state, string state)
+ RetrieveSupplierReviewsBySupplierID(i
nt supplierID)
+ RetrieveSupplierTagsBySupplierID(int
supplierID)
+ CreateSupplierRequest(Supplier
supplier): int
+ Supplier
RetrieveSupplierByEmail(string email)
+ List<Supplier>
RetrieveActiveSuppliers()
+ RetrieveSuppliersNoUserID() :
List<Supplier>
+ RetrieveSupplierImagesbySupplierID(in
t supplierID)
+ Supplier CreateSupplierListing(Supplier
supplier)
+ List<Supplier>
RetrieveUnapprovedSuppliers
+ bool ApproveSupplier(int supplierID)
+ bool DisapproveSupplier(int
supplierID)
+ bool RequeueSupplier(int supplierID)
+ RetrieveSupplierAvailabilityBySupplierI
DAndDate(int supplierID, DateTime
date)
+ List<DateTime>
SupplierAvailabilityForNextThreeMonths
(int supplierID)
+ int CreateSupplierReview(Reviews
review)
```

7.4.3.2 Service

IServiceManager

```
+List<Service>
RetrieveServicesBySupplierID(int
supplierID)
+int EditServiceByServiceID(int serviceID,
Service oldService, Service newService)
+Service RetrieveServiceByServiceID(int
serviceID)
+int DeleteService(int serviceID)
+int CreateService(Service service)
+int
RemoveServiceByServiceOldService(Serv
ice oldService)
```

7.4.3.3 LocationManager

LocationManager

```
+int DeactivateLocationByLocationID(int
LocationID)
+Location
RetrieveLocationBioInformation(int
LocationID)
+
RetrieveLocationByNameAndAddress(str
ing locationName, string address)
+ RetrieveLocationByLocationID(int
locationID)
+ RetrieveActiveLocations()
+List<String>
RetrieveLocationImagesByLocationID(int
locationID)
-bool CreateLocation(string
locationName, string address, string
locationCity, string locationState, string
locationZipCode)
+int
UpdateLocationBioByLocationID(Locatio
n oldLocation, Location newLocation)
+List<Availability>
RetrieveLocationAvailabilityByLocationID
AndDate(int locationID, DateTime date)
+List<String>
RetrieveLocationRulesByLocationID(int
locationID)
+int AddLocationImageByLocationID(int
locationID, String imagePath)
+ RetrieveLocationReviews(int
locationID)
+int
DeactivateLocationImageByImageID(int
imageID)
+List<string>
RetrieveTagsByLocationID(int
locationID);
+ int CreateLocationReview(Reviews
review)
```

7.4.3.4 Sublocation

ISublocationManager

```
+CreateSublocationByLocationID(int
locationID, string sublocationName,
string sublocationDesc)
+Sublocation
RetrieveSublocationBySublocationID(int
sublocationID)
+int EditSublocation(Sublocation
oldSublocation, Sublocation
newSublocation)
+int
DeactivateSublocationBySublocationID(i
nt sublocationID)
+???
RetrieveScheduleBySublocationID(int
sublocationID)
+List<String>
RetrieveMapsBySublocationID(int
sublocationID)
+bool CreateSublocationRule(int
sublocationID, String rule)
+int EditSublocationRule(int
sublocationID, String oldRule, String
newRule)
+int DeactivateSublocationRule(int
sublocationID, String rule)
+int
CreateSublocationMapBySublocationID(i
nt sublocationID, String mapPath)
+int
DeactivateSublocationMapByMapID(int
mapID)
+List<Sublocation>
RetrieveSublocationsByLocationID(in
t locationID)
```

7.4.3.5 Supply

```
ISupplyManager
+List<SupplyVM>
RetrieveSuppliesBySublocationID(int
sublocationID)
+int EditSupplyBySublocationID(int
sublocationID, Supply oldSupply, Supply
newSupply)
+int
DeactivateSupplyByOldSupply(Supply
oldSupply)
+int CreateSupply(Supply supply)
+int RemoveSupplyByOldSupply(Supply
oldSupply)
```

7.4.3.6 Parking Lot

```
IParkingLotManager
+List<ParkingLotVM>
RetrieveParkingLotByLocationID(int
locationID)
+int EditParkingLotByLotID(int lotID,
ParkingLot oldParkingLot, ParkingLot
newParkingLot)
+int CreateParkingLot(ParkingLot
parkingLot)
+bool RemoveParkingLotByLotID(int
lotID)
+bool UserCanEditParkingLot(int userID)
```

7.4.3.7 Entrance

IEntranceManager

```
+List<Entrance>
RetrieveEntrancesByLocationID(int
locationID)
+int EditEntranceByEntranceID(int
entranceID, Entrance oldEntrance,
Entrance newEntrance)
+int CreateEntrance(int locationID,
string entranceName, string description)
+int RemoveEntranceByEntranceID(int
entranceID)
```

7.4.3.8 IZipManager

IZipManager

```
+RetrieveAllZips() : List<Zip>
+RetrieveCityAndStateByZipCode(string
zipcode) : Zip
```

7.4.4 Attendee Manager Interfaces

7.4.4.1 IContactsManager

IContactsManager

```
+ RemoveContact(int userID) : int
+ AddContact(int userID) : bool
+ RetrieveContactsByStatus(string status)
: List<User>
+ RetrieveAllContacts() : List<User>
```

7.4.4.2 IGroupManager

IGroupManager

```
+ RemoveGroup(int groupID) : int
+ UpdateGroupName(int groupID, string
name) : bool
+ RetrieveGroupByGroupID(int groupID) :
Group
+ RemoveGroup(int groupID) : int
+ CreateNewGroup(string name): Group
+ InsertUserInGroup(int userID, int
groupID) : bool
```

7.4.4.3 IGroupMembersManager

IGroupMembersManager

```
+ RetrieveGroupByGroupName(string
GroupName) : Group
+ RetrieveGroupMembers(string
GroupName) : List<User>
+ RemoveUserFromGroup(int GroupID,
int userID) : bool
+ CreateGroupMessage(Message
message, int groupID) : bool
+ RemoveGroupMessage(int messageID,
int groupID) : int
+ RetrieveGroupMessage(int messageID,
int groupID) : Message
+ RetrieveAllGroupMessages(int
GroupID) : List<Messages>
```

7.4.4.4 UserManager

IUserManager

```
+ RetrieveUserByUserID(int userID) : User
+ RetrieveUserByEmail(string Email) : User
+
AuthenticateUserByEmailAndPassword(string email, string passwordHash) : bool
+ HashSha256(string source) : bool
+ UpdatePasswordHash(string email, string oldPasswordHash, string newPasswordHash) : bool
+ RetrieveUserRolesByUserID(int UserID) : List<UserRole>
+ UpdateUserEmail(int userID, string oldEmail, string newEmail, string passwordHash) : bool
+ UpdateUser(User user) : bool
+ CreateUser(User user) : bool
+ LoginUser(string email, string password) : User
+ CreateUserWithPassword(User user, string password) : bool
+ RetrieveAllRoles() : List<String>
+ RetrieveHasUserByEmail(string email) : bool
+ AddUserRole(int userID, string role) : bool
+ RemoveUserRole(int userID, string role) : bool
```

IUSERIMAGEMANAGER

IUserImageManager

```
+ InsertUserImage(int userID, Image image) : bool
+ RetrieveImagesByUserID(int UserID) : List<UserImage>
+ RemoveUserImage(int ImageID) : int
+ UpdateImagePath(int ImageID, string imagePath) : int
+ RetrieveImage(int imageID) : UserImage
```

7.4.4.5 IMessageManager

IMessageManager

```
+ InsertMessage(int senderID, int  
recieverID, Message message) : bool  
+ RemoveMessage(int userID, int  
messageID)  
+ RetrieveMessageByUser(int userID) :  
Message  
+ UpdateMessage(int messageID,  
Message message) : int
```

7.4.4.6 IInterestManager

IInterestManager

```
+ InsertUserInterestTag(int userID, int  
interestID) : bool  
+ RemoveUserInterestTag(int userID, int  
InterestID) : int  
+ RetrieveEventsByInterests(int  
interestID) : List<Event>
```

7.4.4.7 IEventRsvpManager

```
IEventRsvpManager  
+ InsertEventRSVP (int eventID, int  
userID) : bool  
+ UpdateEventRSVP (int eventID, int  
userID, bool attendedEvent) : int
```

7.4.4.8 IReviewManager

```
IReviewManager  
+ InsertReview(Review review) : bool  
+ UpdateReview(int reviewID, Review  
review) : int  
+ RemoveReview(int reviewID) : bool
```

7.4.4.9 ISavedEventManager

```
ISavedEventManager  
+  
RetrieveSavedEventByEventName(string  
eventName, int userID) : SavedEvent  
+ RetrieveAllSavedEvents(int userID) :  
List<SavedEvents>  
+ RemoveSavedEvent(int eventID, int  
userID) : bool
```

7.5 Validation Helpers

7.5.1 StringValidationHelper

StringValidationHelper

```
+ bool StringNoSpecialCharacters (this
  string)
+ bool StringLengthLimit (this string, int
  length)
+ bool StringExactLength (this string, int
  exactLength)
+ bool StringLength50 (this string)
+ bool StringLength100 (this string)
+ bool StringLength250 (this string)
+ bool StringLength1000 (this string)
+ bool StringLength3000 (this string)
+ bool StringValidEmailAddress (this
  string)
+ bool ContainsOnlyIntegers(this string
  testString)
+ bool IsValidHour(this string time)
+ bool IsValidMinute(this string time)
```

7.5.2 IntegerValidationHelper

IntegerValidationHelper

```
+ bool IntegerBetween (this int, int
  minValue, int maxValue)
+ bool IntegerNotLargerThan (this int,
  int maxValue)
+ bool IntegerNotSmallerThan (this int,
  int minValue)
+ bool ContainsOnlyIntegers (this string
  testString)
+ bool IsValidHour(this string time)
+ bool IsValidMinute(this string time)
int ConvertTo24HourTime(this int hour,
  bool isAM)
```

7.5.3 DateTimeValidationHelper

DateTimeValidationHelper

```
+ bool DateTimeNotPast (this DateTime)
+ bool DateTimeBetween (this
DateTime, DateTime pastDate,
DateTime futureDate)
+ bool DateTimeNotAfter (this
DateTime, DateTime futureDate)
```

7.5.4 DecimalValidationHelpers

DecimalValidationHelpers

```
+ DecimalValidationHelpers bool  
(DecimalNotLargerThan)  
+ DecimalValidationHelpers bool  
(DecimalNotSmallerThan)  
+ DecimalValidationHelpers bool  
(DecimalBetween)  
+ DecimalValidationHelpers bool  
(DecimalNotMoreThanHundredThousan  
dsAndNoLessThanHundredths)
```

8 Stored Procedures

8.1 *Stored Procedures Summary*

The stored procedures section has the names of the stored procedures with the parameter names and data types needed to perform create, read, update, and delete methods on the Event system's data store.

This includes the list of the stored procedures that will be called by the data access interfaces. The naming convention is "sp_" followed by a description of what the procedure does, what parameters it needs, and what is returned. The parameters data type and size are specified so that they will not throw errors when interacting with the fields in the tables where the data reside. Finally, the data accessor interface that the procedure is associated with is specified in the "Data Access Interface" field.

The stored procedures are organized by the Data Accessor Interface that would call the procedure.

8.2 Stored Procedures

8.2.1 Task Accessor

Name	Parameters	Type	Data Access Interface	File Name
sp_select_task_by_taskID	@TaskID	int	iTaskAccessor	stored_procedures/task_stored_procedures.sql
sp_select_assigned_user_by_task_assignmentID	@EventID	int	iTaskAccessor	stored_procedures/task_stored_procedures.sql
	@TaskID	int		
	@TaskAssignmentID	int		
	@TaskID	int	iTaskAccessor	stored_procedures/task_stored_procedures.sql
sp_select_active_tasks_by_eventID	@EventID	int	iTaskAccessor	stored_procedures/task_stored_procedures.sql
sp_select_tasks_by_eventID	@EventID	int	iTaskAccessor	stored_procedures/task_stored_procedures.sql
sp_select_active_tasks_by_due_date	@DueDate	DateTime	iTaskAccessor	stored_procedures/task_stored_procedures.sql
sp_select_active_tasks_by_priority	@Priority	int	iTaskAccessor	stored_procedures/task_stored_procedures.sql
sp_update_task	@EventID	int	iTaskAccessor	stored_procedures/task_stored_procedures.sql
	@TaskID	int		
	@OldName	nvarchar(50)		
	@OldDescription	nvarchar(255)		
	@OldDueDate	DateTime		
	@OldPriority	int		
	@OldActive	bit		
	@NewName	nvarchar(50)		
	@NewDescription	nvarchar(255)		
	@NewDueDate	DateTime		
	@NewPriority	int		
	@NewActive	bit		
sp_update_task_progress_by_completion_date	@TaskID	int	iTaskAccessor	stored_procedures/task_stored_procedures.sql
	@CompletionDate	DateTime		
sp_insert_new_taskAssignment_by_taskID	@TaskID	int		stored_procedures/task_stored_procedures.sql
	@DateAssigned	DateTime		

Name	Parameters	Type	Data Access Interface	File Name
sp_insert_new_task_by_eventID	@EventID @Name @Description @DueDate @Priority	int nvarchar(50) nvarchar(255) DateTime int	iTaskAccessor	stored_procedures/task_stored_procedures.sql
sp_deactivate_task_by_completion_date	@TaskID @CompletionDate	int DateTime	iTaskAccessor	stored_procedures/task_stored_procedures.sql
sp_delete_task_by_taskID	@TaskID	int	iTaskAccessor	stored_procedures/task_stored_procedures.sql
sp_select_question_by_postID	@PostID	int	iQuestionAccessor	
sp_select_questions_by_created_date_and_eventID	@EventID @CreatedDate	int DateTime	iQuestionAccessor	
sp_select_questions_by_question_type	@QuestionTypeID	int	iQuestionAccessor	
sp_select_questions_by_answered	@IsAnswered	bit	iQuestionAccessor	
sp_update_question_by_answered	@QuestionID @OldIsAnswered @NewIsAnswered @DateAnswered	int bit bit DateTime	iQuestionAccessor	
sp_update_question_by_question_type	@QuestionID @OldQuestionType @NewQuestionType	int int	iQuestionAccessor	
sp_update_question_by_content	@PostID @QuestionID @OldContent @NewContent	int int nvarchar nvarchar	iQuestionAccessor	
sp_insert_new_question_by_postID	@PostID	int	iQuestionAccessor	

Name	Parameters	Type	Data Access Interface	File Name
	@QuestionTypeID	int		
	@isRankingList	bit		
	@isAnswered	bit		
sp_deactivate_question_by_event_end_date	@EventID	int	iQuestionAccessor	
	@QuestionID	int		
	@EndDateTime	DateTime		
sp_select_answer_by_questionID	@QuestionID	int	iAnswerAccessor	
sp_select_answers_by_date_answered	@DateAnswered	DateTime	iAnswerAccessor	
sp_update_answer_by_description	@AnswerID	int	iAnswerAccessor	
	@OldDescription	nvarchar(100)		
	@NewDescription	nvarchar(100)		
sp_update_answer_by_date_answered	@AnswerID	int	iAnswerAccessor	
	@OldAnswerDate	DateTime		
	@NewAnswerDate	DateTime		
sp_insert_new_answer_by_questionID	@QuestionID	int	iAnswerAccessor	
	@Description	nvarchar(100)		
	@DateAnswered	DateTime		
sp_deactivate_answer_by_event_end_date	@EventID	int	iAnswerAccessor	
	@QuestionID	int		
	@AnswerID	int		
	@EndDateTime	DateTime		
sp_select_voteID_by_pollID	@PollID	int	iVoteAccessor	
sp_select_votes_by_rank	@IsRanked	bit	iVoteAccessor	
	@UserSelection	int		
sp_update_vote_by_user_selection	@VoteID	int	iVoteAccessor	
	@OldSelection	int		
	@NewSelection	int		
sp_deactivate_vote_by_event_end_date	@EventID	int	iVoteAccessor	
	@PollID	int		
	@VoteID	int		

Name	Parameters	Type	Data Access Interface	File Name
	@EndDateTime	DateTi me		
sp_select_invitations_by_eventID	@EventID	int	IInvitationAcce ssor	
sp_select_number_of_invitations _by_attending	@Attending	bit	IInvitationAcce ssor	
sp_select_number_of_invitations _by_eventID	@EventID	int	IInvitationAcce ssor	
sp_select_invitations_by(userID)	@InvitationID	int	IInvitationAcce ssor	
	@UserID	int		
sp_update_invitation_by_attendi ng	@InvitationID	int	IInvitationAcce ssor	
	@OldAttendanc e	bit		
	@NewAttendanc e	bit		
sp_insert_new_invitation_by_eve ntID	@EventID	int	IInvitationAcce ssor	
	@UserID	int		
	@RoleID	int		
	@Attending	bit		
sp_deactivate_invitations_by_eve nt_end_date	@EventID	int	IInvitationAcce ssor	
	@InvitationID	int		
	@EndDateTime	DateTi me		
sp_select_budget	@EventID	int	IEventAccessor	
sp_insert_forum	@Name	nvarchr ar(50)	IForumAccesso r	
	@Description	nvarchr ar(255)		
sp_select_forum_by_forumID	@ForumID	int	IForumAccesso r	
sp_update_forum_by_forumID	@ForumID		IForumAccesso r	
	@OldForumNa me	nvarchr ar(50)		
	@NewForumNa me	nvarchr ar(50)		
	@OldForumDes cription	nvarchr ar(255)		
	@NewForumDe scription	nvarchr ar(255)		
sp_select_message_boards_by_f orumID	@ForumID	int	IMessageBoar dAccessor	
sp_select_message_board_by_m essage_boardID	@MessageBoar dID	int	IMessageBoar dAccessor	

Name	Parameters	Type	Data Access Interface	File Name
sp_insert_message_board	@Name @Description	nvarchar(25)	IMessageBoardAccessor	
sp_update_message_board_by_message_boardID	@MessageBoardID @OldName @NewName @OldDescription @NewDescription	nvarchar(255)	IMessageBoardAccessor	
sp_delete_message_board_by_message_boardID	@MessageBoardID	int	IMessageBoardAccessor	
sp_select_message_board_user_by_message_boardID_and_userID	@MessageBoardID @UserID	int	IMessageBoardUserAccessor	
sp_select_message_board_users_by_message_boardID_and_userID	@MessageBoardID @UserID	int	IMessageBoardUserAccessor	
sp_delete_message_board_users_by_message_boardID_and_userID	@MessageBoardID @UserID	int	IMessageBoardUserAccessor	
sp_insert_post	@Content	nvarchar(255)	IPostAccessor	
sp_select_post_by_postID	@PostID	int	IPostAccessor	
sp_select_posts_by_postID	@PostID	int	IPostAccessor	
sp_update_post_by_postID	@OldContent @NewContent	nvarchar(255)	IPostAccessor	
sp_delete_post_by_postID	@PostID	int	IPostAccessor	
sp_select_post_image_by_postID	@PostID	int	IPostAccessor	
sp_select_vote_by_postID	@PostID	int	IPostAccessor	
sp_insert_comment	@Content	nvarchar(255)	ICommentAccessor	
sp_select_comments_by_postID	@PostID	int	ICommentAccessor	
sp_delete_comment_by_commentID	@CommentID	int	ICommentAccessor	

Name	Parameters	Type	Data Access Interface	File Name
sp_update_comment_by_commentID	@CommentID	int	ICommentAccessor	
sp_toggle_like_status_by_likeID_and(userID	@LikeID	int	ICommentLike Accessor	
	@UserID	int		
sp_select_likes_by_commentID	@CommentID	int	ICommentLike Accessor	
sp_insert_post_image			IPostImage	
sp_delete_post_image_by_post_imageID	@PostImageID	int	IPostImage	
sp_insert_event	@EventName	nvarchar(50)	IEventAccessor	stored_procedures\event_stored_procedures.sql
	@EventDescription	nvarchar(1000)		
	@TotalBudget	money		
sp_insert_event_return_event_id	@EventName	nvarchar(50)	IEventAccessor	stored_procedures\event_stored_procedures.sql
	@EventDescription	nvarchar(1000)		
	@TotalBudget	money		
sp_insert_event_with_user_ID_return_event_id	@EventName	nvarchar(50)	IEventAccessor	stored_procedures\event_stored_procedures.sql
	@EventDescription	nvarchar(1000)		
	@TotalBudget	money		
	@UserID	int		
sp_select_event_by_eventID	@EventID	int	IEventAccessor	stored_procedures\event_stored_procedures.sql
sp_select_event_by_event_name_and_description	@EventName	nvarchar(50)	IEventAccessor	stored_procedures\event_stored_procedures.sql
	@EventDescription	nvarchar(1000)		
sp_select_active_events			IEventAccessor	stored_procedures\event_stored_procedures.sql
sp_select_events_by(userID	@UserID	int	IEventAccessor	
sp_update_event_name_by_eventID	@EventID	int	IEventAccessor	
	@OldName	nvarchar(50)		
	@NewName	nvarchar(50)		
sp_update_event_description_by_eventID	@EventID		IEventAccessor	

Name	Parameters	Type	Data Access Interface	File Name
	@OldDescription	nvarchar(1000)		
	@NewDescription	nvarchar(1000)		
sp_update_event_by_eventID	@EventID	int	IEventAccessor	stored_procedures\event_stored_procedures.sql
	@OldEventName	[nvarchar](50)		
	@OldEventDescription	[nvarchar](1000)		
	@OldTotalBudget	money		
	@OldActive	[bit]		
	@NewEventName	[nvarchar](50)		
	@NewEventDescription	[nvarchar](1000)		
	@NewTotalBudget	money		
sp_select_user_roles_for_event	@EventID	int	IEventAccessor	stored_procedures\event_stored_procedures.sql
	@UserID	int		
sp_deactivate_event_by_eventID	@EventID	int	IEventAccessor	stored_procedures\event_stored_procedures.sql
sp_delete_event_by_eventID	@EventID	int	IEventAccessor	
sp_select_active_events_for_past_and_future_event_dates			IEventAccessor	stored_procedures\event_stored_procedures.sql
sp_select_active_events_for_upcoming_dates			IEventAccessor	stored_procedures\event_stored_procedures.sql
sp_select_active_events_for_past_dates			IEventAccessor	stored_procedures\event_stored_procedures.sql
sp_select_active_events_for_upcoming_dates_for_user	@UserID	int	IEventAccessor	stored_procedures\event_stored_procedures.sql
sp_select_active_events_for_past_dates_for_user	@UserID	int	IEventAccessor	stored_procedures\event_stored_procedures.sql
sp_select_active_events_for_past_and_upcoming_dates_for_user	@UserID	int	IEventAccessor	stored_procedures\event_stored_procedures.sql
sp_update_event_location_by_event_id	@EventID	int	IEventAccessor	stored_procedures\event_stored_procedures.sql
	@OldLocationID	int		
	@LocationID	int		
sp_insert_activity	@ActivityName	nvarchar(50)	IActivityAccessor	stored_procedures\activity_stored_procedures.sql

Name	Parameters	Type	Data Access Interface	File Name
	@ActivityDescription	nvarchar ar(250)		
	@Public	bit		
	@StartTime	time		
	@EndTime	time		
	@ActivityImage Name	varchar (25)		
	@SublocationID	int		
	@EventDateID	date		
	@EventID	int		
sp_select_activities_by_sublocati onID	@SublocationID	int	IActivityAccess or	stored_procedures\activity_s tored_procedures.sql
sp_select_activity_by_activityID	@ActivityID	int	IActivityAccess or	
sp_select_activities_by_eventID	@EventID	int	IActivityAccess or	stored_procedures\activity_s tored_procedures.sql
sp_select_activities_by_eventID_ and_event_datelD	@EventID	int	IActivityAccess or	stored_procedures\activity_s tored_procedures.sql
	@EventDateID	date		
sp_select_active_activities_for_u pcoming_dates			IActivityAccess or	stored_procedures\activity_s tored_procedures.sql
sp_select_activities_for_past_dat es			IActivityAccess or	stored_procedures\activity_s tored_procedures.sql
sp_select_activities_for_past_and _upcoming_dates			IActivityAccess or	stored_procedures\activity_s tored_procedures.sql
sp_select_active_activities_for_u pcoming_dates_for_user	@UserID	int	IActivityAccess or	stored_procedures\activity_s tored_procedures.sql
sp_select_past_activities_for_use r	@UserID	int	IActivityAccess or	stored_procedures\activity_s tored_procedures.sql
sp_select_all_activities_for_user	@UserID	int	IActivityAccess or	stored_procedures\activity_s tored_procedures.sql
sp_select_activityVM_by_activityI D	@ActivityID	int	IActivityAccess or	stored_procedures\activity_s tored_procedures.sql
sp_update_activity_by_activityID	@ActivityID	int	IActivityAccess or	
	@OldActivityNa me	nvarchar ar(50)		
	@OldActivityDe scription	nvarchar ar(250)		
	@OldPublic	bit		
	@OldStartTime	DateTi me		
	@OldEndTime	DateTi me		
	@OldActivityIm ageName	varchar (25)		
	@NewActivityN ame	nvarchar ar(50)		

Name	Parameters	Type	Data Access Interface	File Name
	@NewActivityDescription	nvarchar(250)		
	@NewPublic	bit		
	@NewStartTime	DateTime		
	@NewEndTime	DateTime		
	@NewActivityImageName	varchar(25)		
sp_update_activity_name_by_activityID	@ActivityID	int	IActivityAccess or	
	@OldActivityName	nvarchar(50)		
	@NewActivityName	nvarchar(50)		
sp_update_activity_description_by_activityID	@ActivityID	int	IActivityAccess or	
	@OldActivityDescription	nvarchar(250)		
	@NewActivityDescription	nvarchar(250)		
sp_update_activity_public_by_activityID	@ActivityID	int	IActivityAccess or	
	@OldPublic	bit		
	@NewPublic	bit		
sp_update_activity_start_time_by_activityID	@ActivityID	int	IActivityAccess or	
	@OldStartTime	DateTime		
	@NewStartTime	DateTime		
sp_update_activity_end_time_by_activityID	@ActivityID	int	IActivityAccess or	
	@OldEndTime	DateTime		
	@NewEndTime	DateTime		
sp_update_activity_image_name_by_activityID	@ActivityID	int	IActivityAccess or	
	@OldActivityImageName	varchar(25)		
	@NewActivityImageName	varchar(25)		
sp_deactivate_activity_by_activityID	@ActivityID	int	IActivityAccess or	
sp_delete_activity_by_activityID	@ActivityID	int	IActivityAccess or	
sp_update_activity_sublocation_by_activityID	@ActivityID	int	IActivityAccess or	

Name	Parameters	Type	Data Access Interface	File Name
	@OldSubLocationID	int		
	@NewSubLocationID	int		
sp_insert_poll	@Title	nvarchar(100)	IPollAccessor	
	@Description	nvarchar(100)		
	@DueDate	DateTime		
sp_select_poll_by_postID	@PostID	int	IPollAccessor	
sp_update_poll_by_postID	@PostID	int	IPollAccessor	
	@OldTitle	nvarchar(100)		
	@OldDescription	nvarchar(100)		
	@OldDueDate	DateTime		
	@NewTitle	nvarchar(100)		
	@NewDescription	nvarchar(100)		
	@NewDueDate	DateTime		
sp_update_poll_title_by_postID	@PostID	nvarchar(100)	IPollAccessor	
	@OldTitle	nvarchar(100)		
	@NewTitle	nvarchar(100)		
sp_update_poll_description_by_postID	@PostID	nvarchar(100)	IPollAccessor	
	@OldDescription	nvarchar(100)		
	@NewDescription	nvarchar(100)		
sp_update_poll_due_date_by_postID	@PostID	nvarchar(100)	IPollAccessor	
	@OldDueDate	DateTime		
	@NewDueDate	DateTime		
sp_deactivate_poll_title_by_postID	@PostID	nvarchar(100)	IPollAccessor	
sp_delete_poll_title_by_postID	@PostID	nvarchar(100)	IPollAccessor	
sp_insert_activity_result_by_activityID_and_rank	@ActivityID	int	IActivityResultAccessor	

Name	Parameters	Type	Data Access Interface	File Name
	@ActivityResultRank	int		
sp_select_activity_result_by_activityID_and_rank	@ActivityID	int	IActivityResultAccessor	
	@ActivityResultRank	int		
sp_select_activity_results_by_activityID	@ActivityID	int	IActivityResultAccessor	stored_procedures/activity_result_stored_procedures.sql
sp_update_activity_result_by_activity_activityID_and_result_name	@ActivityID	int	IActivityResultAccessor	
	@OldActivityResultName	nvarchar(50)		
	@NewActivityResultName	nvarchar(50)		
sp_delete_activity_results_by_activityID	@ActivityID	int	IActivityResultAccessor	
sp_insert_event_date	@EventDateID	Date	IEventDateAccessor	stored_procedures/event_date_stored_procedures.sql
	@EventID	int		
	@StartTime	Time(0)		
	@EndTime	Time(0)		
sp_select_event_date_by_event_dateID_and_eventID	@EventDateID	DateTime	IEventDateAccessor	stored_procedures/event_date_stored_procedures.sql
	@EventID	int		
sp_select_event_dates_by_event_ID	@EventID	int	IEventDateAccessor	stored_procedures/event_date_stored_procedures.sql
sp_update_event_date	@EventID	int	IEventDateAccessor	stored_procedures/event_date_stored_procedures.sql
	@OldDateID	Date		
	@OldStartTime	Time(0)		
	@OldEndTime	Time(0)		
	@NewEventDateID	Date		
	@NewStartTime	Time(0)		
	@NewEndTime	Time(0)		
	@NewActive	bit		
sp_deactivate_event_date_by_event_dateID_and_eventID	@EventDateID	DateTime	IEventDateAccessor	
	@EventID	int		
sp_delete_event_date_by_event_dateID_and_eventID	@EventDateID	DateTime	IEventDateAccessor	
	@EventID	int		

Name	Parameters	Type	Data Access Interface	File Name
sp_select_event_dates_by_location_id	@LocationID	int	IEventDateAccessor	stored_procedures/event_date_stored_procedures.sql
sp_select_event_date_by(userID_and_date	@UserID	int	IEventDateAccessor	stored_procedures/event_date_stored_procedures.sql
	@EventDate	int		
sp_insert_event_upload_by_eventID	@EventID	int	IEventUploadAccess	
	@EventFileName	varchar(25)		
	@DateTimeCreated	DateTi me		
	@Title	nvarc h ar(50)		
	@Description	nvarc h ar(250)		
sp_select_event_upload_by_event_uploadID	@EventUploadID	int	IEventUploadAccess	
sp_select_active_event_uploads_by_eventID	@EventID	int	IEventUploadAccess	
sp_update_event_upload_by_event_uploadID	@EventUploadID	int	IEventUploadAccess	
	@OldEventFileName	varchar(25)		
	@OldTitle	nvarc h ar(50)		
	@OldDescriptio n	nvarc h ar(250)		
	@NewEventFileName	varchar(25)		
	@NewTitle	nvarc h ar(50)		
	@NewDescripti on	nvarc h ar(250)		
sp_update_event_upload_file_name_by_event_uploadID	@EventUploadID	int	IEventUploadAccess	
	@OldEventFileName	varchar(25)		
	@NewEventFileName	varchar(25)		
sp_update_event_upload_title_and_description_by_event_upload_ID	@EventUploadID	int	IEventUploadAccess	
	@OldTitle	nvarc h ar(50)		
	@OldDescriptio n	nvarc h ar(250)		
	@NewTitle	nvarc h ar(50)		

Name	Parameters	Type	Data Access Interface	File Name
	@NewDescription	nvarchar(250)		
sp_deactivate_event_upload_by_event_uploadID	@EventUploadID	int	IEventUploadAccessor	
sp_delete_event_upload_by_event_uploadID	@EventUploadID	int	IEventUploadAccessor	
sp_insert_event_picture_by_eventID	@EventID	int	IEventPictureAccessor	
	@EventPictureName	varchar(25)		
	@DateTimeCreated	DateTime		
	@Title	nvarchar(50)		
	@Caption	nvarchar(50)		
	@Description	nvarchar(250)		
	@isMainDisplayImage	bit		
sp_select_event_picture_by_event_pictureID	@EventPictureID	int	IEventPictureAccessor	
sp_select_event_pictures_by_eventID	@EventID	int	IEventPictureAccessor	
sp_select_event_pictures_for_main_display_by_eventID	@EventID	int	IEventPictureAccessor	
sp_update_event_picture_by_event_pictureID	@EventPictureID	int	IEventPictureAccessor	
	@OldEventPictureName	varchar(25)		
	@OldTitle	nvarchar(50)		
	@OldCaption	nvarchar(50)		
	@OldDescription	nvarchar(250)		
	@NewEventPictureName	varchar(25)		
	@NewTitle	nvarchar(50)		
	@NewCaption	nvarchar(50)		
	@NewDescription	nvarchar(250)		
	@IsMainDisplayImage	bit		
sp_update_event_picture_name_by_event_pictureID	@EventPictureID	int	IEventPictureAccessor	

Name	Parameters	Type	Data Access Interface	File Name
	@OldEventPictureName	varchar(25)		
	@NewEventPictureName	varchar(25)		
sp_update_event_picture_main_display_by_event_pictureID	@EventPictureID	int	IEventPictureAccessor	
	@IsMainDisplayImage	bit		
sp_update_event_picture_title_caption_and_description_by_event_pictureID	@EventPictureID	int	IEventPictureAccessor	
	@OldTitle	nvarchar(50)		
	@OldCaption	nvarchar(50)		
	@OldDescription	nvarchar(250)		
	@NewTitle	nvarchar(50)		
	@NewCaption	nvarchar(50)		
	@NewDescription	nvarchar(250)		
sp_deactivate_event_picture_by_event_pictureID	@EventPictureID	int	IEventPictureAccessor	
sp_delete_event_picture_by_event_pictureID	@EventPictureID	int	IEventPictureAccessor	
sp_insert_user_to_notification_group_by_notification_groupID	@NotificationID	int	INotificationGroupAccessor	
	@UserID	int		
sp_select_notification_group_users_by_notification_groupID	@NotificationID	int	INotificationGroupAccessor	
sp_delete_notification_group_users_by_notification_groupID	@NotificationID	int	INotificationGroupAccessor	
sp_insert_emergency_procedure_by_eventID	@EventID	int	IEmergencyProcedureAccessor	
	@Message	nvarchar(255)		
sp_select_emergency_procedure_by_emergency_procedureID	@EmergencyProcedureID	int	IEmergencyProcedureAccessor	
sp_select_emergency_procedure_by_eventID	@EventID	int	IEmergencyProcedureAccessor	
sp_update_emergency_procedure_message_by_eventID	@EventID	int	IEmergencyProcedureAccessor	

Name	Parameters	Type	Data Access Interface	File Name
	@OldMessage	nvarchar(255)		
	@NewMessage	nvarchar(255)		
sp_delete_emergency_procedure_message_by_eventID	@EventID	int	IEmergencyProcedureAccessor	
sp_insert_emergency_exit_by_emergency_procedureID	@EmergencyProcedureID	int	IEmergencyExit Accessor	
	@EmergencyExitsDescription	nvarchar(50)		
	@SubLocationID	int		
sp_select_emergency_exits_by_emergency_procedureID	@EmergencyProcedureID	int	IEmergencyExit Accessor	
sp_select_emergency_exits_by_eventID	@EventID	int	IEmergencyExit Accessor	
sp_select_emergency_exit_by_emergency_procedureID_and_sublocationID	@EmergencyProcedureID	int	IEmergencyExit Accessor	
	@SubLocationID	int		
sp_update_emergency_exits_by_emergency_procedureID	@EmergencyProcedureID	int	IEmergencyExit Accessor	
	@NewEmergencyExitsDescription	nvarchar(50)		
	@NewSubLocationID	int		
	@NewEmergencyProcedureID	int		
	@OldEmergencyExitsDescription	nvarchar(50)		
	@OldSubLocationID	int		
	@OldEmergencyProcedureID	int		
sp_delete_emergency_exit_by_emergency_procedureID_and_sublocationID	@EmergencyProcedureID	int	IEmergencyExit Accessor	
	@SubLocationID	int		
sp_insert_muster_point_by_emergency_procedureID	@EmergencyProcedureID	int	IMusterPointAccess	
	@EmergencyMusterPointDescription	nvarchar(50)		

Name	Parameters	Type	Data Access Interface	File Name
	@SubLocationID	int		
sp_select_muster_point_by_emergency_procedureID	@EmergencyProcedureID	int	IMusterPointAccess	
sp_select_muster_point_by_eventID	@EventID	int	IMusterPointAccess	
sp_update_muster_point_by_emergency_procedureID	@EmergencyProcedureID	int	IMusterPointAccess	
	@NewMusterPointDescription	nvarchar(50)		
	@NewSubLocationID	int		
	@NewEmergencyProcedureID	int		
	@OldMusterPointDescription	nvarchar(50)		
	@OldSubLocationID	int		
	@OldEmergencyProcedureID	int		
sp_delete_muster_point_by_emergency_procedureID_and_sublocationID	@EmergencyProcedureID	int	IMusterPointAccess	
	@SubLocationID	int		
sp_select_roles_by_roleID	@RoleID	nvarchar	RoleAccess	
sp_select_all_skill_sets			SkillSetAccess	
sp_select_skill_set_by_skill_setID	@SkillSetID	nvarchar	SkillSetAccess	
sp_update_skill_set_description_by_skill_setID	@SkillSetID	nvarchar	SkillSetAccess	
	@OldSkillSetDescription	nvarchar		
	@NewSkillSetDescription	nvarchar		
sp_select_all_volunteers			VolunteerAccess	
sp_select_volunteers_by_roleID	@RoleID	nvarchar	VolunteerAccess	
sp_select_volunteers_by_name	@GivenName	nvarchar	VolunteerAccess	
	@FamilyName	nvarchar		
sp_select_volunteers_by_given_name	@GivenName	nvarchar	VolunteerAccess	
sp_select_volunteers_by_family_name	@FamilyName	nvarchar	VolunteerAccess	

Name	Parameters	Type	Data Access Interface	File Name
sp_select_volunteers_by_available_day	@AvailableDay	date	VolunteerAccessor	
sp_select_volunteers_by_available_time	@AvailableTimeStart	time	VolunteerAccessor	
	@AvailableTimeEnd	time		
sp_select_volunteers_by_skillSet	@SkillSetID	nvarchar	VolunteerAccessor	
sp_select_volunteers_by_rating	@Rating	int	VolunteerAccessor	
sp_select_volunteers_by_state	@State	nvarchar	VolunteerAccessor	
sp_select_volunteers_by_city	@City	nvarchar	VolunteerAccessor	
sp_insert_volunteer	@UserID	int	VolunteerAccessor	
sp_update_volunteerID_by_userID	@UserID	int	VolunteerAccessor	
	@OldVolunteerID	int		
	@NewVolunteerID	int		
sp_deactivate_volunteer_by_volunteerID	@VolunteerID	int	VolunteerAccessor	
sp_select_volunteer_by(userID	@UserID	int	VolunteerAccessor	stored_procedures/volunteer.sql
sp_select_availability_by_volunteerID_and_date	@VolunteerID	int	VolunteerAccessor	stored_procedures/availability.sql
sp_select_availability_exception_by_volunteerID_and_date	@VolunteerID	int	VolunteerAccessor	stored_procedures/availability.sql
	@StartExceptionDate	datetime		
	@EndExceptionDate	datetime		
sp_add_eventID_to_volunteer	@EventID	int	VolunteerAccessor	stored_procedures/volunteer_stored_procedures.sql
	@UserID	int		
sp_insert_volunteer_availability	@VolunteerID	int	VolunteerAvailability	
	@AvailableDay	date		
	@AvailableTimeStart	time		
	@AvailableTimeEnd	time		
sp_delete_volunteer_availability_by_availabilityID	@AvailabilityID	int	VolunteerAvailability	
sp_update_available_day_by_availabilityID	@AvailabilityID	int	VolunteerAvailability	

Name	Parameters	Type	Data Access Interface	File Name
	@OldAvailableDay	date		
	@NewAvailableDay	date		
sp_update_time_start_by_availabilityID	@AvailabilityID	int	VolunteerAvailability	
	@OldAvailableTimeStart	time		
	@NewAvailableTimeStart	time		
sp_update_time_end_by_availabilityID	@AvailabilityID	int	VolunteerAvailability	
	@OldAvailableTimeEnd	time		
	@NewAvailableTimeEnd	time		
sp_select_days_available_by_volunteerID	@VolunteerID	int	VolunteerAvailability	
sp_insert_volunteer_proof	@ProofID	int	VolunteerProof	
	@VolunteerID	int		
	@ProofImage	varchar		
	@ProofDescription	nvarchar		
sp_delete_volunteer_proof_by_proofID	@ProofID	int	VolunteerProof	
sp_update_proof_image_by_proofID	@ProofID	int	VolunteerProof	
sp_update_proof_description_by_proofID	@ProofID	int	VolunteerProof	
sp_select_volunteer_proof_by_volunteerID	@VolunteerID	int	VolunteerProof	
sp_insert_volunteer_record	@VolunteerID	int	VolunteerRecord	
	@EventID	int		
	@LocationID	int		
	@Rating	int		
	@EventDateID	datetime		
sp_delete_volunteer_record_by_volunteer_recordID	@VolunteerRecordID	int	VolunteerRecord	
sp_select_volunteer_record_by_volunteerID	@VolunteerID	int	VolunteerRecord	
sp_insert_volunteer_review	@VolunteerID	int	VolunteerReviews	
	@Rating	int		
	@Comments	nvarchar		

Name	Parameters	Type	Data Access Interface	File Name
sp_update_volunteer_review_rating_by_reviewID	@ReviewID @OldRating @NewRating	int int int	VolunteerReviews	
sp_update_volunteer_review_comment_by_reviewID	@ReviewID @OldComment @NewComment	int nvarchar nvarchar	VolunteerReviews	
sp_delete_volunteer_rating_by_reviewID	@ReviewID	int	VolunteerReviews	
sp_select_all_volunteer_reviews			VolunteerReviews	
sp_select_volunteer_reviews_by_volunteerID	@VolunteerID	int	VolunteerReviews	stored_procedures/review.sql
sp_insert_volunteer_skill_set_by_volunteerID	@VolunteerID @SkillSetID	int nvarchar	VolunteerSkillsSet	
sp_delete_volunteer_skill_set_by_volunteerID	@VolunteerID @SkillSetID	int nvarchar	VolunteerSkillsSet	
sp_select_skill_set_by_volunteerID	@VolunteerID	int	VolunteerSkillsSet	stored_procedures/skill_set.sql
sp_update_supply_name_by_supplyID	@SupplyID @OldSupplyName @NewSupplyName	int nvarchar nvarchar	VolunteerSupplies	
sp_update_supply_amount_by_supplyID	@SupplyID @OldSupplyAmount @NewSupplyAmount	int nvarchar nvarchar	VolunteerSupplies	
sp_update_supply_description_by_supplyID	@SupplyID @OldSupplyDescription @NewSupplyDescription	int nvarchar nvarchar	VolunteerSupplies	
sp_insert_volunteer_supplies	@VolunteerID @SupplyName	int nvarchar	VolunteerSupplies	

Name	Parameters	Type	Data Access Interface	File Name
	@SupplyAmount	int		
	@SupplyDescription	nvarchar		
sp_deactivate_supplies_by_supplyID	@SupplyID	int	VolunteerSupplies	
sp_select_supplies_by_volunteerID	@VolunteerID	int	VolunteerSupplies	
sp_select_all_volunteers_by(userID)	@UserID	int	FavoriteVolunteers	
sp_select_volunteer_by(volunteerID)	@VolunteerID	int	FavoriteVolunteers	stored_procedures/volunteer_stored_procedures.sql
sp_insert_volunteer	@UserID	int	FavoriteVolunteers	
	@VolunteerID	int		
sp_delete_volunteer_by(volunteerID)	@UserID	int	FavoriteVolunteers	
	@VolunteerID	int		
sp_select_all_requests_for(volunteer_by(volunteerID))	@VolunteerID	int	VolunteerRequests	storedprocedures\volunteer_requests_stored_procedures.sql
sp_select_all_requests_by(eventID)	@EventID	int	VolunteerRequests	storedprocedures\volunteer_requests_stored_procedures.sql
sp_select_request_by(requestID)	@RequestID	int	VolunteerRequests	storedprocedures\volunteer_requests_stored_procedures.sql
sp_insert_volunteer_request	@VolunteerID	int	VolunteerRequests	
	@EventID	int		
	@VolunteerResponse	boolean		
	@EventResponse	boolean		
sp_update_volunteer_request	@RequestID	int	VolunteerRequests	storedprocedures\volunteer_request_stored_procedures.sql
	@OldVolunteerResponse	boolean		
	@OldEventResponse	boolean		
	@NewVolunteerResponse	boolean		
	@NewEventResponse	boolean		
sp_delete_volunteer_request_by(requestID)	@RequestID	int	VolunteerRequests	
sp_select_all_verification_requests			VerificationRequests	

Name	Parameters	Type	Data Access Interface	File Name
sp_select_all_verification_requests_by_volunteerID	@VolunteerID	int	VerificationRequests	
sp_select_all_verification_requests_by_skill_setID	@SkillSetID	nvarchar	VerificationRequests	
sp_insert_verification_request	@VolunteerID	int	VerificationRequests	
	@SkillSetID	nvarchar		
sp_update_verification_request_by_volunteerID	@VolunteerID	int	VerificationRequests	
	@SkillSetID	nvarchar		
	@OldAccepted	boolean		
	@NewAccepted	boolean		
sp_delete_verification_request_by_requestID	@RequestID	int	VerificationRequests	
sp_select_all_volunteer_applications			VolunteerApplications	
sp_select_volunteer_application_by_requestID	@RequestID	int	VolunteerApplications	
sp_insert_volunteer_application	UserID	int	VolunteerApplications	storedprocedures\volunteer_stored_procedures.sql
	TimeStart	time		
	TimeEnd	time		
	Sunday	bit		
	Monday	bit		
	Tuesday	bit		
	Wednesday	bit		
	Thursday	bit		
	Friday	bit		
	Saturday	bit		
sp_update_volunteer_application_by_requestID	@RequestID	int	VolunteerApplications	
	@OldAccepted	boolean		
	@NewAccepted	boolean		
sp_delete_volunteer_application_by_requestID	@RequestID	int	VolunteerApplications	
sp_select_contact_by(userID	@UserID	int	IContactAccess	
sp_select_contact_list	@UserID	int	IContactAccess	
sp_insert_contact_in_contacts	@UserID	int	IContactAccess	
	@UserID	int		

Name	Parameters	Type	Data Access Interface	File Name
sp_deactive_user_from_contact_list	@UserID	int	IContactAccess or	
	@UserID	int		
sp_select_group_by_groupID	@GroupID	int	IGroupAccesso r	
sp_insert_new_group	@Name	nvarchr ar(25)	IGroupAccesso r	
sp_deactive_group	@GroupID	int	IGroupAccesso r	
sp_insert_user_into_group	@GroupID	int	IGroupAccesso r	
	@UserID	int		
sp_update_group_name	@Name	nvarchr ar(25)	IGroupAccesso r	
	@ GroupID	int		
sp_deactivate_user_from_group	@UserID	int	IGroupAccesso r	
	@GroupID	int		
sp_select_group_members_by_groupID	@GroupID	int	IGroupMembe rAccessor	
sp_insert_group_message_into_group	@GroupID	int	IGroupMembe rAccessor	
	@Message	nvarchr ar(250)		
sp_deactivate_group_message_by_messageID	@GroupID	int	IGroupMembe rAccessor	
	@MessageID	int		
sp_select_group_message_by_messageID	@GroupID	int	IGroupMembe rAccessor	
	@MessageID	int		
sp_create_EventRSVP_For_EventSaved	@EventID	int	IEventRSVPAcc essor	
	@UserID	int		
sp_update_EventRSVP_UserAttended	@EventID	int	IEventRSVPAcc essor	
	@UserID	int		
	@AttendedEvent	bit		
sp_deactivate_User_Active_From_EventRSVP	@EventID	int	IEventRSVPAcc essor	
	@UserID	int		
	@Active	bit		
sp_insert_User	@GivenName	nvarchr ar(50)	IUserAccessor	user_stored_procedures.sql
	@FamilyName	nvarchr ar(50)		
	@Email	nvarchr ar(250)		

Name	Parameters	Type	Data Access Interface	File Name
	@UserState	char(2)		
	@City	nvarchar(75)		
	@Zip	int		
sp_retrieve_user_by(userID	@UserID	int	IUserAccessor	
sp_select_user_by_email	@Email	nvarchar(250)	IUserAccessor	user_stored_procedures.sql
sp_authenticate_user	@Email	nvarchar(250)	IUserAccessor	user_stored_procedures.sql
	@PasswordHash	nvarchar(100)		
sp_update_user_PasswordHash_With_email	@Email	nvarchar(250)	IUserAccessor	
	@OldPasswordHash	nvarchar(100)		
	@NewPasswordHash	nvarchar(100)		
sp_retrieve_user_roles_by(userID	@UserID	int	IUserAccessor	
sp_update_user_email	@UserID	int	IUserAccessor	
	@Oldemail	nvarchar(250)		
	@Newemail	nvarchar(250)		
	@PasswordHash	nvarchar(100)		
sp_authenticate_user	@GivenName	nvarchar(50)	IUserAccessor	user_stored_procedures.sql
	@FamilyName	nvarchar(50)		
	@Email	nvarchar(250)		
	@UserState	char(2)		
	@City	nvarchar(75)		
	@Zip	int		
sp_authenticate_user	@Email	nvarchar(100)	IUserAccessor	user_stored_procedures.sql
sp_update_passwordHash	@Email	nvarchar(100)	IUserAccessor	user_stored_procedures.sql
	@OldPasswordHash	nvarchar(100)		
	@NewPasswordHash	nvarchar(100)		
sp_create_user_message	@UserID	int	IMessageAccessor	
	@Message	nvarchar(250)		

Name	Parameters	Type	Data Access Interface	File Name
sp_select_all_roles			IUserAccessor	user_stored_procedures.sql
sp_insert_user_role	@UserID	int	IUserAccessor	user_stored_procedures.sql
	@RoleID	nvarchar(50)		
sp_delete_user_role	@UserID	int	IUserAccessor	user_stored_procedures.sql
	@RoleID	nvarchar(50)		
sp_select_user_by(userID	@UserID	int	IUserAccessor	user_stored_procedures.sql
sp_select_user_roles	@UserID	int	IUserAccessor	user_stored_procedures.sql
sp_deactivate_user_message	@UserID	int	IMessageAccessor	
	@MessageID	int		
sp_retrieve_user_message	@SenderUserID	int	IMessageAccessor	
	@ReceiverUserID	int		
	@MessageID	int		
sp_create_user_draft_message	@UserID	int	IMessageAccessor	
	@Message	nvarchar(250)		
sp_create_user_message_with_image	@UserID	int	IMessageAccessor	
	@Message	nvarchar(250)		
	@UserImage	nvarchar(200)		
sp_insert_user_interest_by(userID	@UserID	int	IIInterestAccessor	
	@InterestID	int		
sp_deactivate_user_interest	@UserID	int	IIInterestAccessor	
	@InterestID	int		
sp_retrieve_events_by_interestID_and_date_range	@InterestID	int	IIInterestAccessor	
sp_retrieve_events_by_interestID_and_date_range	@InterestID	int	IIInterestAccessor	
	@EventDateID	int		
	@StartDate	DateTime		
	@EndDate	DateTime		
sp_create_user_image	@UserID	int	IUserImageAccessor	
sp_retrieve_userimage_by(userID	@UserID	int	IUserImageAccessor	
	@UserImageID	int		

Name	Parameters	Type	Data Access Interface	File Name
sp_select_userimages_by(userID)	@UserID	int	IUserImageAccessor	stored_procedures/user_image_stored_procedures.sql
sp_deactivate_user_image	@UserID	int	IUserImageAccessor	
	@UserImageID	int		
sp_update_image_name	@UserID	int	IUserImageAccessor	
	@ImageID	int		
	@OldUserImage	nvarchar(200)		
	@NewUserImage	nvarchar(200)		
sp_insert_profile_image_for_user	@UserID int	int	IUserImageAccessor	
	@ImageID	int		
sp_deactivate_review_by_reviewID	@ReviewID	int	IReviewAccess or	
	@EventID	int		
sp_retrieve_reviews_by_eventID	@EventID	int	IReviewAccess or	
sp_create_review	@Review	nvarchar(300)	IReviewAccess or	
	@EventID	int		
sp_update_review_by_reviewID	@Review	nvarchar(300)	IReviewAccess or	
	@ReviewID	int		
	@UserID	int		
sp_create_review_with_image	@Review	nvarchar(300)	IReviewAccess or	
	@EventID	int		
	@UserImage	nvarchar(200)		
	@EventID			
sp_select_saved_event_by_eventID	@EventID	int	ISavedEventAccess or	
sp_remove_event_from_saved_events	@EventID	int	ISavedEventAccess or	
sp_retrieve_saved_events	@UserID	int	ISavedEventAccess or	
	@EventDescription	int		
sp_retrieve_event_details_by_eventID	@EventID	nvarchar(1000)	ISavedEventAccess or	
	@EventGroupID	int		

Name	Parameters	Type	Data Access Interface	File Name
	@EventGroupID			
sp_select_group_by_groupID	@UserID	int	IEventGroupAccessor	
sp_retrieve_group_members_by_groupID	@EventGroupID	int	IEventGroupAccessor	
	@UserID	int		
sp_deactivate_user_from_group_member_list	@EventGroupDescription	int	IEventGroupAccessor	
	@GroupID	int		
sp_retrieve_group_details_by_groupID	@EventID	nvarchar ar(300 0)	IEventGroupAccessor	
	@UserID	int		
sp_create_group_by_eventID	@GroupID	int	IEventGroupAccessor	
	@ImageID	int		
sp_insert_group_image_for_group	@GroupID	int	IEventGroupAccessor	
	@ImageID	int		
sp_insert_new_report	@UserID	int	IReportAccess or	
	@Reason	nvarchar ar(25)		
	@Description	nvarchar ar(250)		
	@DateCreated	dateTi me		
sp_retrieve_report_by_reportID	@ReportID	int	IReportAccess or	
sp_select_services_by_SupplierID	@SupplierID	int	IServiceAccess or	stored_procedures/service_stored_procedures.sql
sp_update_service	@ServiceID	int	IServiceAccess or	database scripts\stored_procedures\service_stored_procedures.sql
	@OldSupplierID	int		
	@OldServiceName	nvarchar ar(160)		
	@OldPrice	decima l(10,2)		
	@OldDescription	nvarchar ar(300 0)		
	@OldServiceName	nvarchar ar(200)		
	@NewSupplierID	int		

Name	Parameters	Type	Data Access Interface	File Name
	@NewServiceName	nvarchar(160)		
	@NewPrice	decimal(10,2)		
	@NewDescription	nvarchar(3000)		
	@NewServiceImageName	nvarchar(200)		
sp_deactivate_service_by_ServiceID	@ServiceID	int	IServiceAccessor	
	@SupplierID	int		
	@ServiceName	nvarchar(160)		
	@Price	decimal(10,2)		
	@Description	nvarchar(3000)		
	@ServiceImagePath	nvarchar(200)		
sp_delete_service_by_ServiceID	@ServiceID	int	IServiceAccessor	database scripts\stored_procedures\service_stored_procedures.sql
sp_select_service_by_ServiceID	@ServiceID	int	IServiceAccessor	database scripts\stored_procedures\service_stored_procedures.sql
sp_insert_service	@SupplierID	int	IServiceAccessor	database scripts\stored_procedures\service_stored_procedures.sql
	@ServiceName	[nvarchar](160),		
	@Price	[decimal](10,2),		
	@Description	[nvarchar](3000),		
	@ServiceImagePath	[nvarchar](200)		
sp_select_supplies_by_SublocationID	@SublocationID	int	ISupplyAccessor	
sp_update_supply_by_SupplyID	@SupplyID	int	ISupplyAccessor	

Name	Parameters	Type	Data Access Interface	File Name
	@OldSublocationID	int		
	@OldSupplyName	nvarchar(160)		
	@OldDescription	nvarchar(3000)		
	@NewSublocationID	int		
	@NewSupplyName	nvarchar(160)		
	@NewDescription	nvarchar(3000)		
sp_deactivate_supply_by_SupplyID	@SupplyID	int	ISupplyAccesso r	
	@SublocationID	int		
	@SupplyName	nvarchar(160)		
	@Description	nvarchar(3000)		
sp_delete_supply_by_SupplyID	@SupplyID	int	ISupplyAccesso r	
	@SublocationID	int		
	@SupplyName	nvarchar(160)		
	@Description	nvarchar(3000)		
sp_insert_supply	@SupplyID	int	ISupplyAccesso r	
	@SublocationID	int		
	@SupplyName	nvarchar(160)		
	@Description	nvarchar(3000)		
sp_insert_availability_for_supplier	@SupplierID	int	ISupplierAcces sor	
	@TimeStart	time		
	@TimeEnd	time		
	@Sunday	bit		
	@Monday	bit		
	@Tuesday	bit		
	@Wednesday	bit		
	@Thursday	bit		
	@Friday	bit		

Name	Parameters	Type	Data Access Interface	File Name
	@Saturday	bit		
sp_select_suppliers_by(userID	@UserID	int	ISupplierAccess	stored_procedures/supplier_stored_procedures.sql
sp_select_entrances_by(locationID	@LocationID	int	IEntranceAccess	stored_procedures/entrance_stored_procedures.sql
sp_update_entrance_by_entranceID	@EntranceID	int	IEntranceAccess	
	@NewEntranceName	nvarchar(100)		
	@OldEntranceName	nvarchar(100)		
	@NewDescription	nvarchar(255)		
	@OldDescription	nvarchar(255)		
sp_insert_entrance	@LocationID	int	IEntranceAccess	stored_procedures/entrance_stored_procedures.sql
	@EntranceName	nvarchar(100)		
	@Description	nvarchar(255)		
sp_deactivate_entrance_by_entranceID	@EntranceID	int	IEntranceAccess	stored_procedures/entrance_stored_procedures.sql
sp_select_entrance_by_entranceID	@EntranceID	int	IEntranceAccess	stored_procedures/entrance_stored_procedures.sql
sp_select_parking_lots_by_locationID	@LocationID	int	IParkingLotAccess	stored_procedures/parking_lot_stored_procedures.sql
sp_select_parking_lot_by_lotID	@LotID	int	IParkingLotAccess	stored_procedures/parking_lot_stored_procedures.sql
sp_update_parking_lot_by_lotID	@LotID	int	IParkingLotAccess	stored_procedures/parking_lot_stored_procedures.sql
	@LocationID	int		
	@OldLocationID	int		
	@LotName	nvarchar(160)		
	@OldLotName	nvarchar(160)		
	@LotDescription	nvarchar(3000)		
	@OldLotDescription	nvarchar(3000)		
	@LotImagePath	nvarchar(200)		
	@OldLotImagePath	nvarchar(200)		

Name	Parameters	Type	Data Access Interface	File Name
sp_insert_parking_lot	@LocationID @Name	int nvarchar(160)	IParkingLotAccessor	stored_procedures/parking_lot_stored_procedures.sql
	@Description	nvarchar(3000)		
	@ImagePath	nvarchar(200)		
sp_delete_parking_lot	@LotID	int	IParkingLotAccessor	stored_procedures/parking_lot_stored_procedures.sql
sp_select_active_locations			LocationAccess or	storedprocedures\location_stored_procedures.sql
sp_select_location_reviews	@LocationID	int	LocationAccess or	stored_procedures\review_stored_procedures.sql
sp_select_location_by_locationID	@LocationID	int	LocationAccess or	storedprocedures\location_stored_procedures.sql
sp_select_location_bio_information	@LocationID	int	LocationAccess or	
sp_select_location_images	@LocationID	int	LocationAccess or	stored_procedures/location_images_stored_procedures.sql
sp_select_location_availability	@LocationID	int	LocationAccess or	stored_procedures\availability_stored_procedures.sql
sp_deactivate_location	@LocationID	int	LocationAccess or	stored_procedures\location_stored_procedures.sql
sp_select_sublocations_by_locationID	@LocationID	int	LocationAccess or	stored_procedures\sublocation_stored_procedures.sql
sp_insert_location	@UserID @LocationName	int nvarchar(160)	LocationAccess or	
	@LocationDescription	nvarchar(3000)		
	@LocationPriceText	nvarchar(3000)		
	@LocationPhone	nvarchar(15)		
	@LocationEmail	nvarchar(250)		
	@LocationAddress1	nvarchar(100)		
	@LocationAddress2	nvarchar(100)		
	@LocationCity	nvarchar(100)		

Name	Parameters	Type	Data Access Interface	File Name
	@LocationState	nvarchar(100)		
	@LocationZipCode	nvarchar(100)		
	@LocationImagePath	nvarchar(200)		
sp_insert_location_by_name_address_city_state_zip			LocationAccess or	storedprocedures\location_stored_procedures.sql
	@LocationName	nvarchar(160)		
	@LocationAddress1	nvarchar(100)		
	@LocationCity	nvarchar(100)		
	@LocationState	nvarchar(100)		
	@LocationZipCode	nvarchar(100)		
sp_select_tags_by_locationID	@LocationID	int	LocationAccess or	storedprocedures\location_stored_procedures.sql
sp_select_location_rules	@LocationID	int	LocationAccess or	
sp_update_locationBio_by_locationID	@LocationID		LocationAccess or	storedprocedures\location_stored_procedures.sql
	@NewLocationDescription	nvarchar(3000)		
	@OldLocationDescription	nvarchar(3000)		
sp_insert_location_image	@LocationID	int	LocationAccess or	
	@ImagePath	nvarchar(200)		
sp_select_location_by_name_and_address	@LocationName	nvarchar(160)	LocationAccess or	storedprocedures\location_stored_procedures.sql
	@LocationAddress1	nvarchar(100)		
sp_deactivate_location_image	@LocationID	int	LocationAccess or	
	@ImagePath	nvarchar(200)		
sp_select_availability_by_locationID_and_date	@LocationID	int	LocationAccess or	storedprocedures\availability_stored_procedures.sql
	@AvailabilityDate	date		
sp_select_availability_exception_by_locationID_and_date	@LocationID	int	LocationAccess or	storedprocedures\availability_stored_procedures.sql

Name	Parameters	Type	Data Access Interface	File Name
	@ExceptionDate	date		
sp_select_availability_by_locationID	@LocationID	int	LocationAccess or	storedprocedures\availability_stored_procedures.sql
sp_select_availability_exception_by_locationID	@LocationID	int	LocationAccess or	storedprocedures\availability_stored_procedures.sql
sp_select_sublocation_by_sublocation_id	@SublocationID	int	SublocationAcc essor	storedprocedures\sublocation_stored_procedures.sql
sp_insert_sublocation_by_locationID	@LocationID	int	SublocationAcc essor	storedprocedures\sublocation_stored_procedures.sql
	@SublocationName	nvarchar(160)	SublocationAcc essor	storedprocedures\sublocation_stored_procedures.sql
	@SublocationDescription	nvarchar(1000)	SublocationAcc essor	storedprocedures\sublocation_stored_procedures.sql
sp_update_sublocation	@SublocationID	int	SublocationAcc essor	storedprocedures\sublocation_stored_procedures.sql
	@OldLocationID	int		
	@OldSublocationName	nvarchar(160)		
	@OldSublocationDescription	nvarchar(1000)		
	@NewLocationID	int		
	@NewSublocationName	nvarchar(160)		
	@NewSublocationDescription	nvarchar(1000)		
sp_deactivate_sublocation	@SublocationID	int	SublocationAcc essor	storedprocedures\sublocation_stored_procedures.sql
sp_select_event_dates_by_sublocation_id	@SublocationID	int	SublocationAcc essor	
sp_select_sublocation_maps	@SublocationID	int	SublocationAcc essor	
sp_insert_sublocation_rule	@SublocationID	int	SublocationAcc essor	
	@RuleText	nvarchar(300)		
sp_update_sublocation_rule	@SublocationID	int	SublocationAcc essor	
	@OldRuleText	nvarchar(300)		
	@NewRuleText	nvarchar(300)		

Name	Parameters	Type	Data Access Interface	File Name
sp_delete_sublocation_rule	@SublocationID @RuleText	int nvarchar(300)	SublocationAccessor	
sp_insert_sublocation_map	@SublocationID @MapPath	int nvarchar(200)	SublocationAccessor	
sp_deactivate_sublocation_map	@SublocationID @MapPath	int nvarchar(200)	SublocationAccessor	
sp_select_active_suppliers			ISupplierAccessor	stored_procedures\supplier_stored_procedures.sql
sp_select_non_active_suppliers			ISupplierAccessor	stored_procedures\supplier_stored_procedures.sql
sp_select_supplier_by_supplierID	@UserID	int	ISupplierAccessor	stored_procedures\supplier_stored_procedures.sql
sp_select_supplier_by_email	@SupplierEmail		ISupplierAccessor	
sp_select_supplier_reviews	@SupplierID	int	ISupplierAccessor	stored_procedures\review_stored_procedures.sql
sp_select_supplier_images	@SupplierID	int	ISupplierAccessor	stored_procedures\supplier_stored_procedures.sql
sp_select_supplier_tags	@SupplierID	int	ISupplierAccessor	stored_procedures\supplier_stored_procedures.sql
sp_select_availability_by_supplierID_and_date	@SupplierID @AvailabilityDate	int date	ISupplierAccessor	stored_procedures\supplier_stored_procedures.sql
sp_select_availability_exception_by_supplierID_and_date	@SupplierID @AvailabilityDate	int date	ISupplierAccessor	stored_procedures\supplier_stored_procedures.sql
sp_select_availability_by_supplierID	@SupplierID	int	ISupplierAccessor	stored_procedures\availability_stored_procedures.sql
sp_select_availability_exception_by_supplierID	@SupplierID @StartDate @EndDate	int Date Date	ISupplierAccessor	stored_procedures\availability_stored_procedures.sql
sp_insert_requested_supplier	@SupplierName @SupplierDescription	nvarchar(160) nvarchar(3000)	ISupplierAccessor	stored_procedures\supplier_stored_procedures.sql

Name	Parameters	Type	Data Access Interface	File Name
	@SupplierPhone	nvarchar(15)		
	@SupplierEmail	nvarchar(250)		
	@SupplierAddress1	nvarchar(100)		
	@SupplierCity	nvarchar(100)		
	@SupplierState	nvarchar(100)		
	@SupplierZipCode	nvarchar(100)		
sp_select_supplier_by_supplierID	@SupplierID	int	ISupplierAccessor	stored_procedures\supplier_stored_procedures.sql
sp_update_supplier	@SupplierID	int	ISupplierAccessor	stored_procedures\supplier_stored_procedures.sql
	@OldSupplierName	nvarchar(160)		
	@OldSupplierDescription	nvarchar(1000)		
	@OldSupplierPhone			
	@OldSupplierEmail	nvarchar(250)		
	@OldAddress1	nvarchar(100)		
	@OldAddress2	nvarchar(100)		
	@OldCity	nvarchar(100)		
	@OldState	nvarchar(100)		
	@OldZipCode	nvarchar(100)		
	@NewSupplierName	nvarchar(160)		
	@NewSupplierDescription	nvarchar(1000)		
	@NewSupplierPhone	nvarchar(15)		
	@NewSupplierEmail	nvarchar(250)		
	@NewSupplierAddress1	nvarchar(100)		
	@NewCity	nvarchar(100)		

Name	Parameters	Type	Data Access Interface	File Name
	@NewState	nvarchar(100)		
	@NewZipCode	nvarchar(100)		
sp_insert_supplier	@UserID	int	ISupplierAccess	stored_procedures\supplier_stored_procedures.sql
	@SupplierName	nvarchar(160)		
	@SupplierDescription	nvarchar(1000)		
	@SupplierPhone	nvarchar(15)		
	@SupplierEmail	nvarchar(250)		
	@SupplierType	nvarchar(10)		
	@SupplierAddress1	nvarchar(100)		
	@SupplierAddress2	nvarchar(100)		
	@City	nvarchar(100)		
	@SupplierState	nvarchar(100)		
	@Zip			
sp_select_unapproved_suppliers			ISupplierAccess	stored_procedures\supplier_stored_procedures.sql
sp_approve_supplier	@SupplierID	int	ISupplierAccess	stored_procedures\supplier_stored_procedures.sql
sp_disapprove_supplier	@SupplierID	int	ISupplierAccess	stored_procedures\supplier_stored_procedures.sql
sp_requeue_supplier	@SupplierID	int	ISupplierAccess	stored_procedures\supplier_stored_procedures.sql
sp_deactivate_supplier_by_supplierID	@SupplierID	int	ISupplierAccess	
sp_insert_review	@UserID	int	ISupplierAccess	stored_procedures\supplier_stored_procedures.sql
	@ReviewType	nvarchar		
	@Rating	int		
	@Review	nvarchar		
	@DateCreated	datetime		
sp_select_review_id_by_review	@UserID	int	ISupplierAccess	stored_procedures\supplier_stored_procedures.sql
	@ReviewType	nvarchar		

Name	Parameters	Type	Data Access Interface	File Name
	@Rating	int		
	@Review	nvarchar ar		
	@DateCreated	dateti me		
sp_insert_new_volunteer_need	@TaskID	int	IVolunteerNee dsAccessor	stored_procedures\voluntee r_need_stored_procedures.s ql
	@NumTotalVol unteers	int		
sp_update_volunteer_need	@TaskID	int	IVolunteerNee dsAccessor	stored_procedures\voluntee r_need_stored_procedures.s ql
	@NumTotalVol unteers	int		
sp_select_volunteer_need_by_ta skID	@TaskID	int	IVolunteerNee dsAccessor	stored_procedures\voluntee r_need_stored_procedures.s ql
sp_delete_volunteer_need	@TaskID	int	IVolunteerNee dsAccessor	stored_procedures\voluntee r_need_stored_procedures.s ql
sp_add_curr_volunteers	@TaskID	int	IVolunteerNee dsAccessor	stored_procedures\voluntee r_need_stored_procedures.s ql
sp_subtract_curr_volunteers	@TaskID	int	IVolunteerNee dsAccessor	stored_procedures\voluntee r_need_stored_procedures.s ql
sp_select_user_roles_from_event _users_table	@UserID	int	IUserAccessor, IParkingLotAcc essor	stored_procedures\user_stor ed_procedures.sql
sp_select_event_planners_for_ev ent	@EventID	int	IEventAccessor	stored_procedures\event_st ored_procedures.sql
sp_select_active_events_by_sear ch	@Search	nvarchar ar(50)	IEventAccessor	stored_procedures\event_st ored_procedures.sql
sp_select_all_zip_codes	N/A	N/A	IZipAccessor	stored_procedures\zipcode_ stored_procedures.sql
sp_select_city_and_states_by_zi pcode	@ZIPCode	nvarchar ar(100)	IZipAccessor	stored_procedures\zipcode_ stored_procedures.sql

9 Data Flows

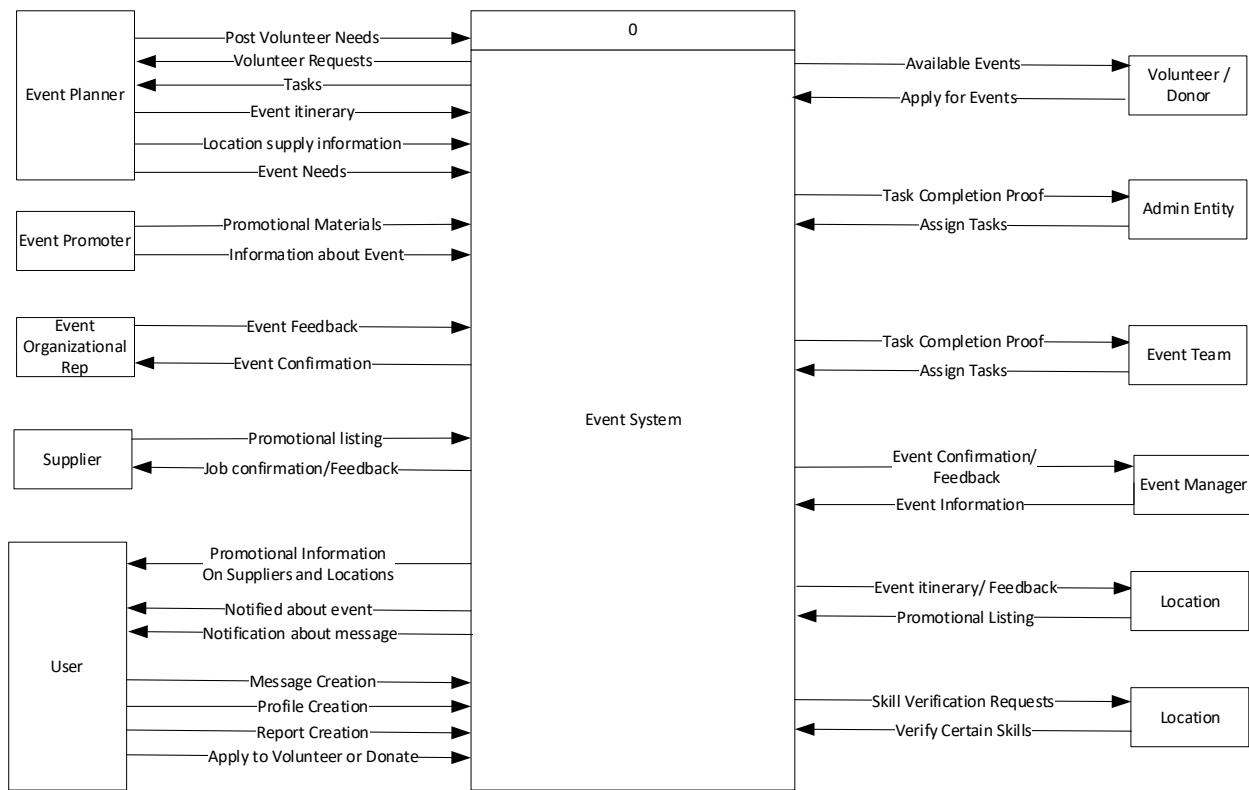
9.1 Data Flows Summary

The purpose of the data flow section is to show how the users in various roles input data and what data they get out of the system. This shows how the data is processed and stored when using the system to create and manage an event. It also shows how the system retrieves this information and what processing is done to move data from the data store to the end users.

The data flow contains a context diagram which shows the data needed by user roles to create and manage an event in very broad terms. That data becomes more specific in the data flow diagrams, showing the types of data that are moving into a process and coming out of it. Any processes that contain more granular child processes are shown in greater detail with a uniquely identifying number that shows what child processes are contained in the parent.

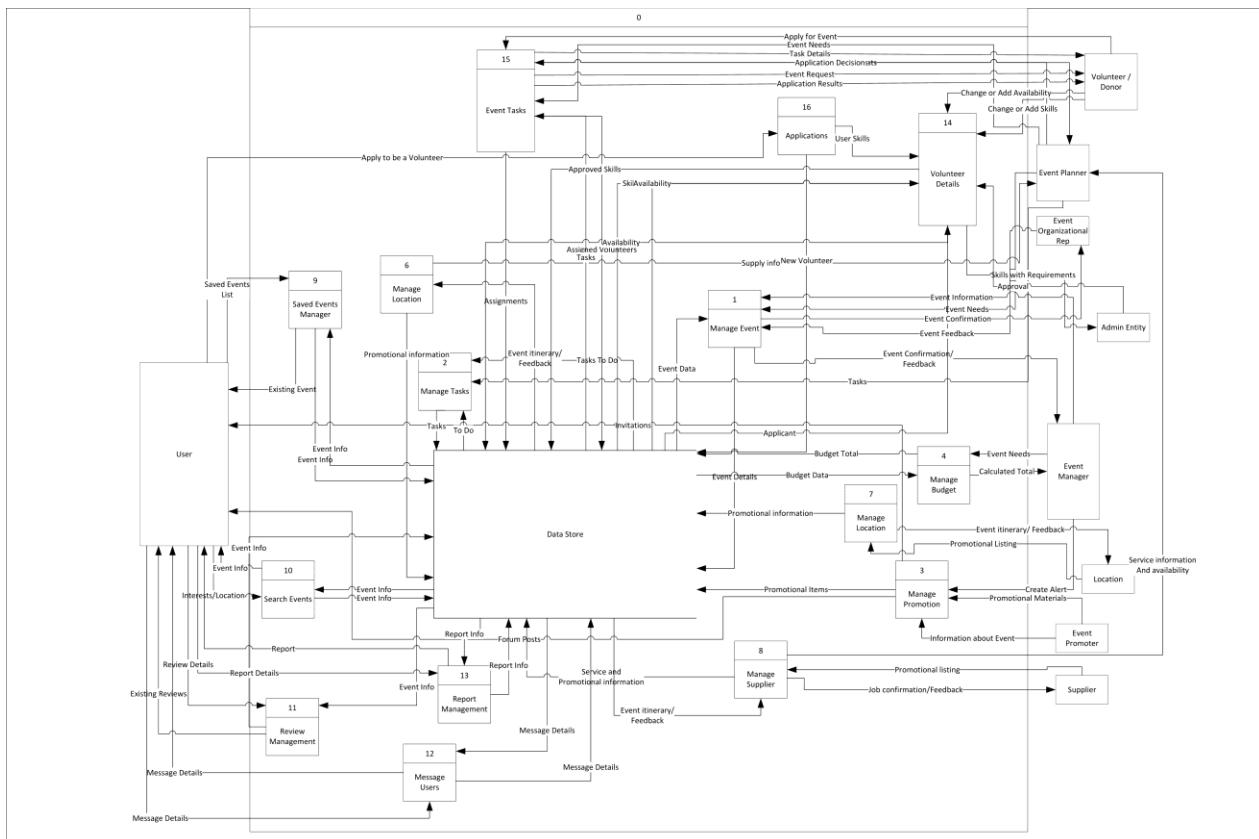
The context diagram and the first few levels show data flows in more broad terms and the lower levels become more specific. The level 0 process shows the topmost level of the system's processes. Each process could have subprocesses, shown by adding a period between its parent process and its unique process number, e.g. 1.1, or, 2.1.1. Data flows associated with groups are Event 1-4, Supplier 5-6, Volunteer 7,8,14-15, Attendee 9-13.

9.2 Context Diagram - Event System – 0

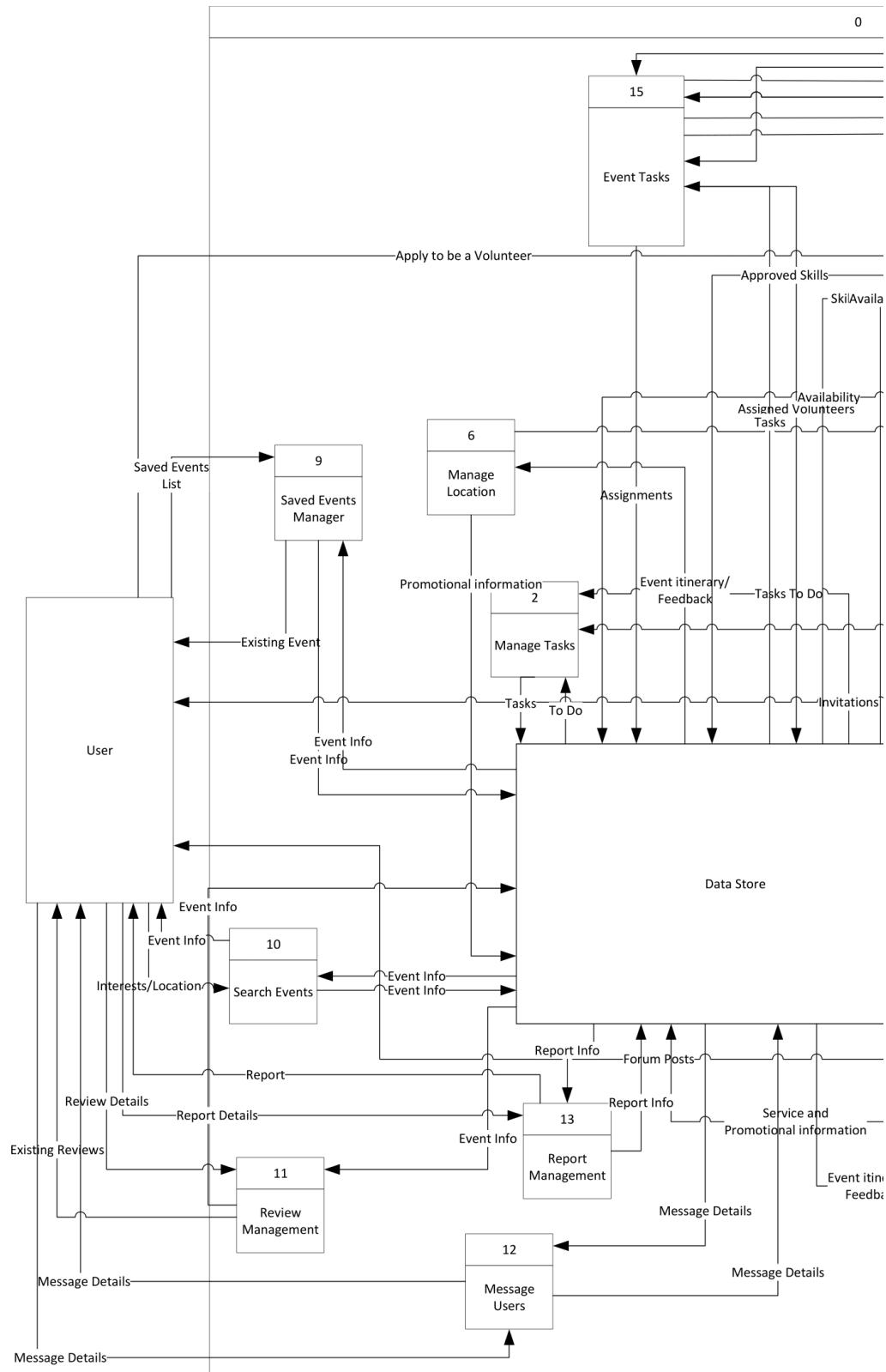


9.3 *Event - 0 - Detail*

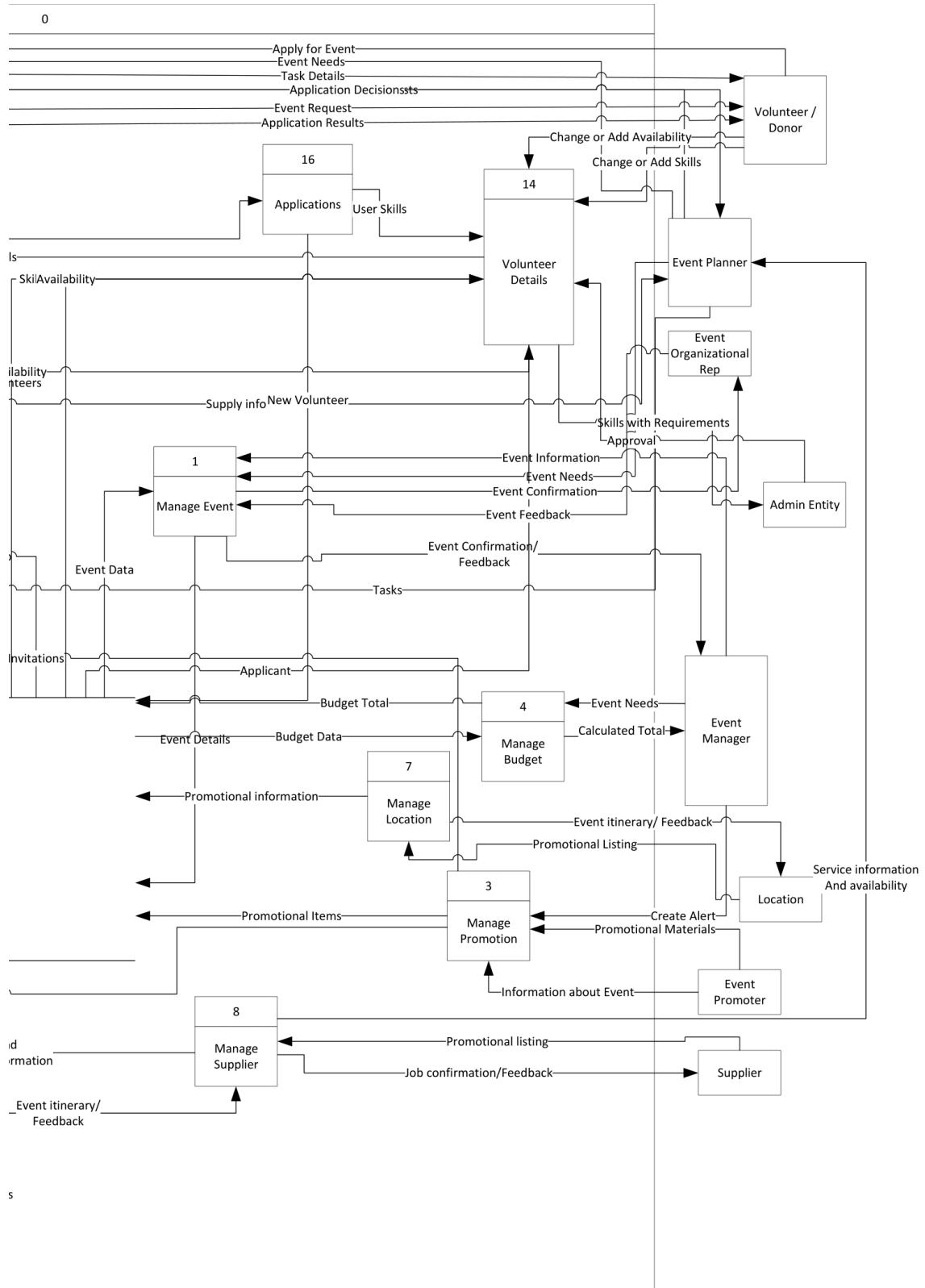
9.3.1 Overview



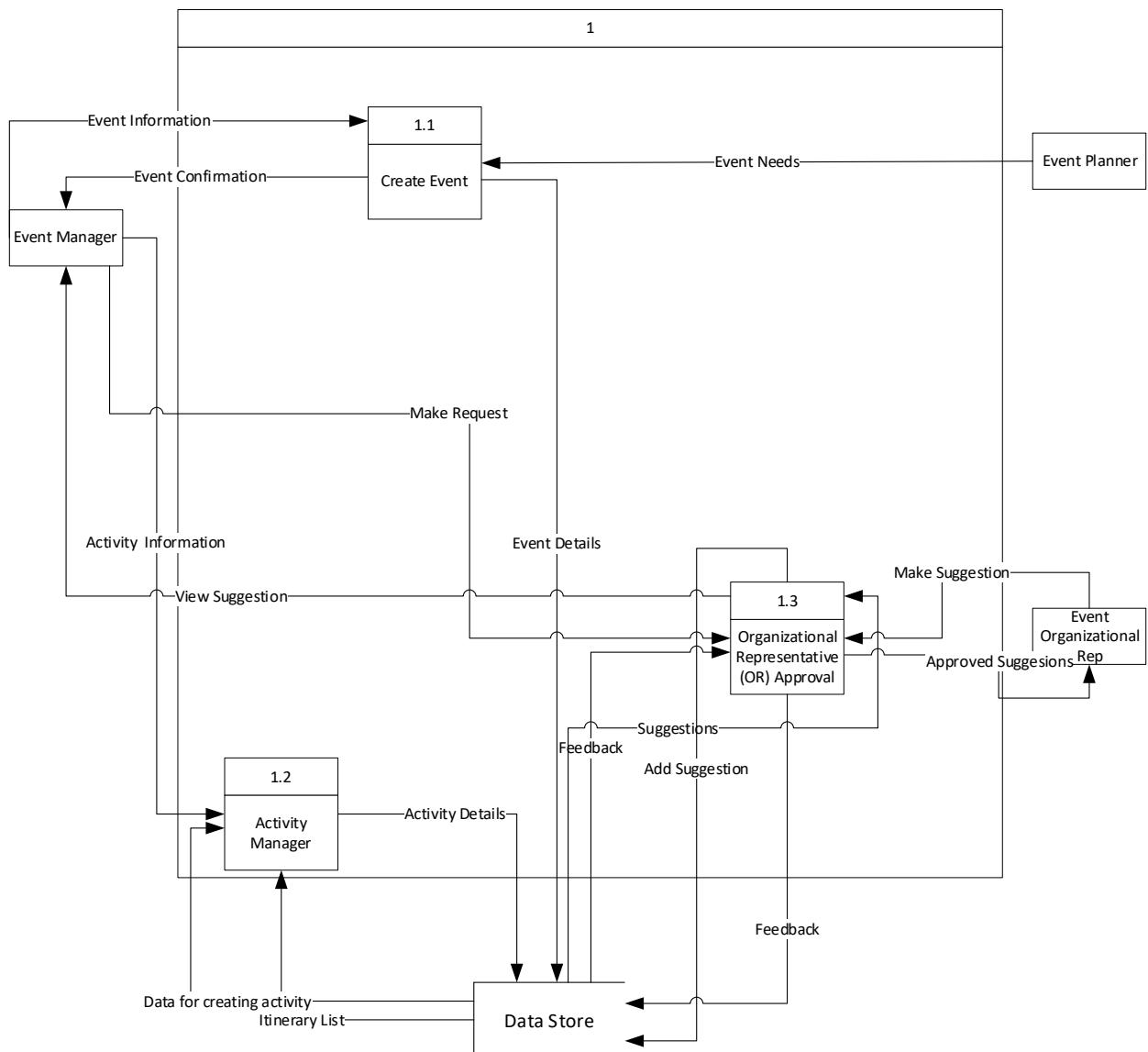
9.3.2 Page 1



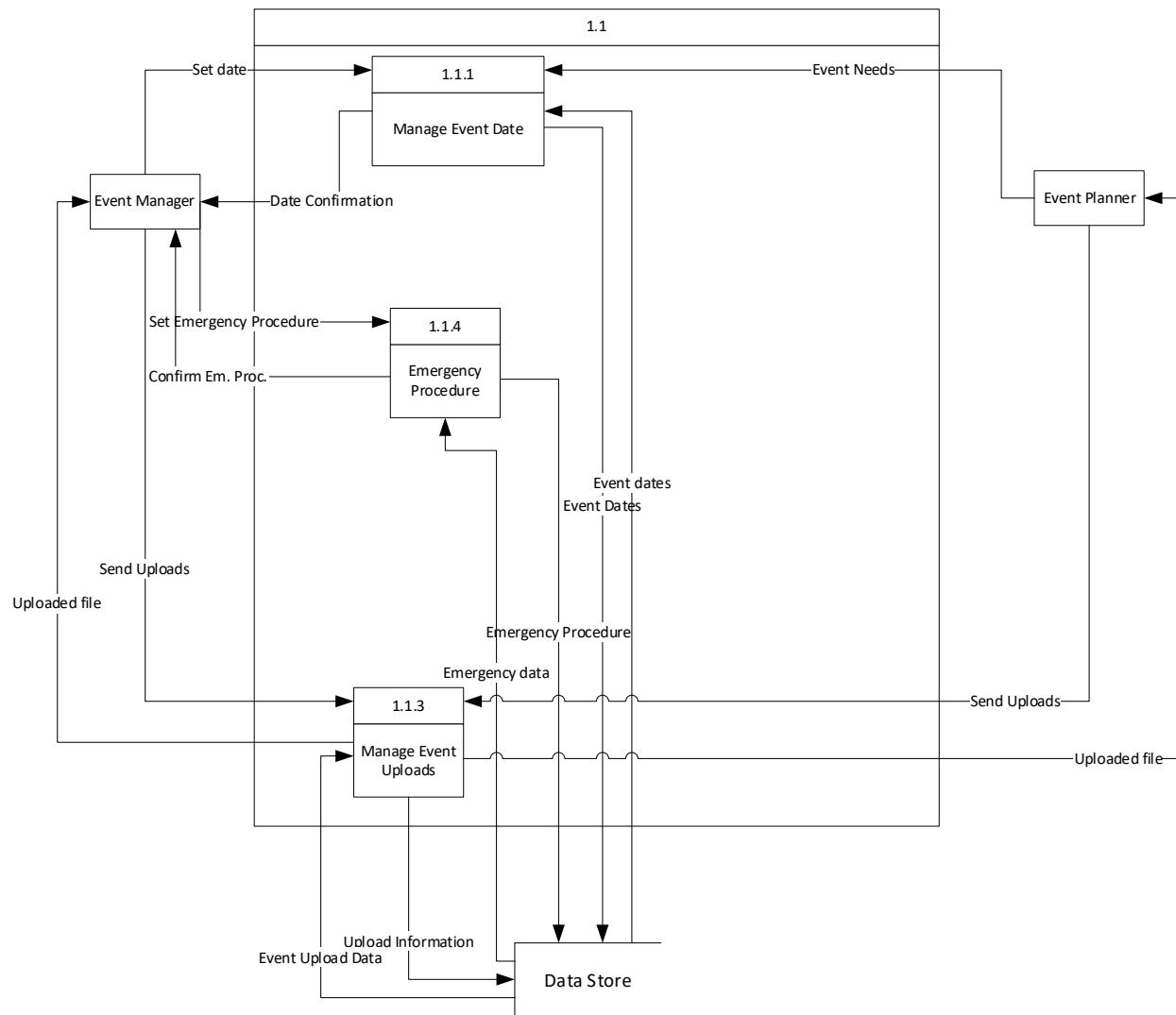
9.3.3 Page 2



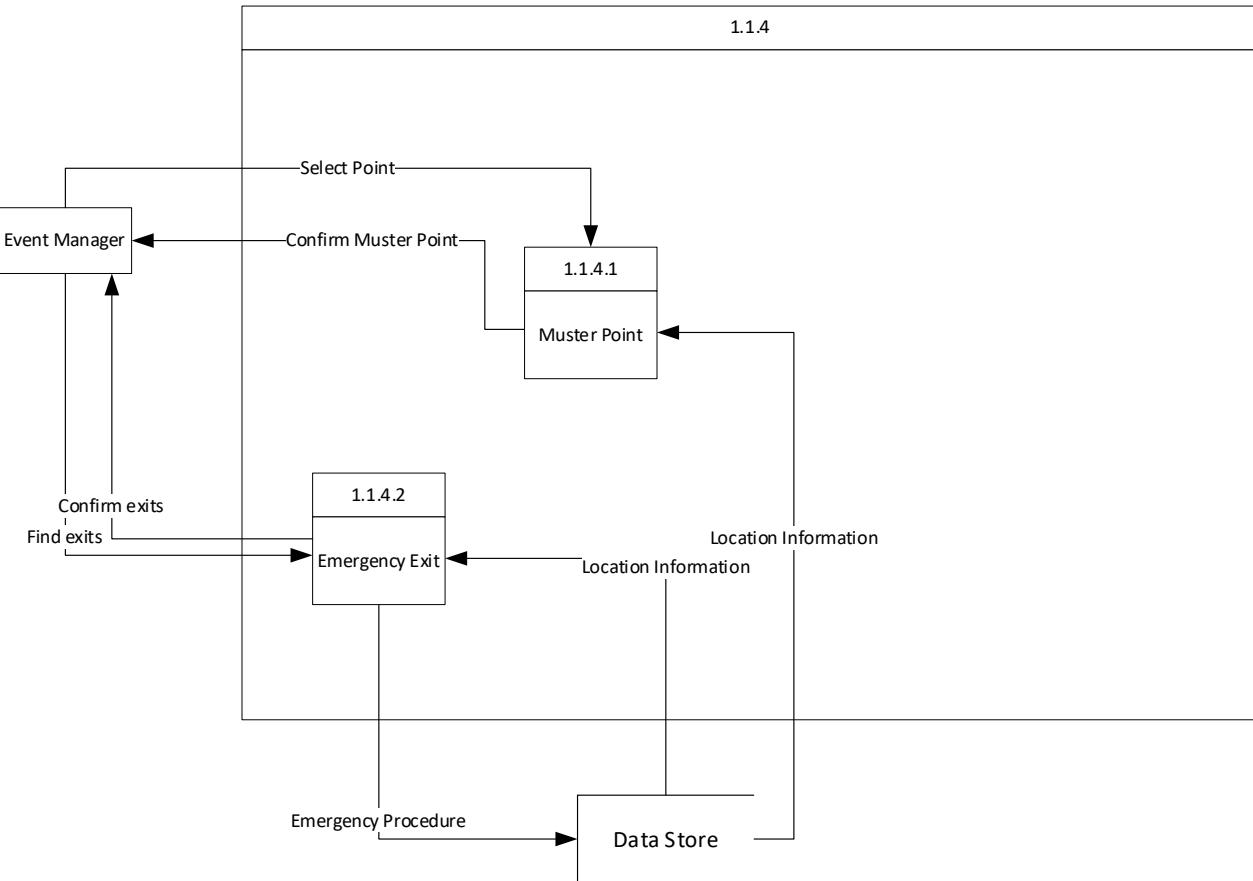
9.4 Manage Event - 1



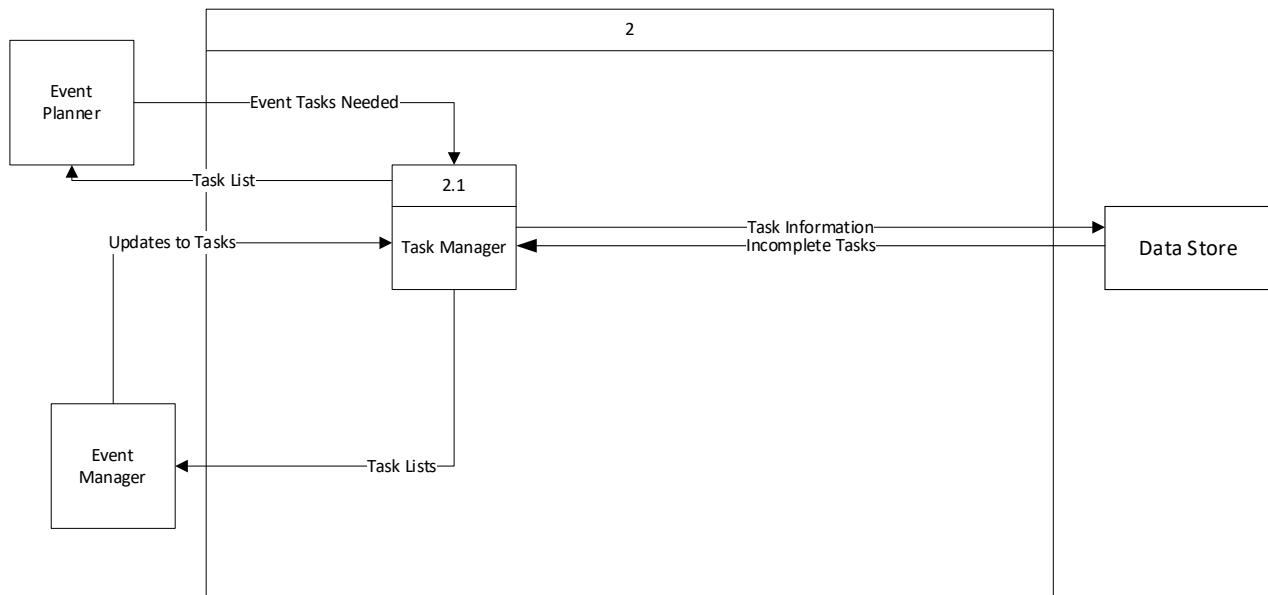
9.4.1 Create Event - 1.1



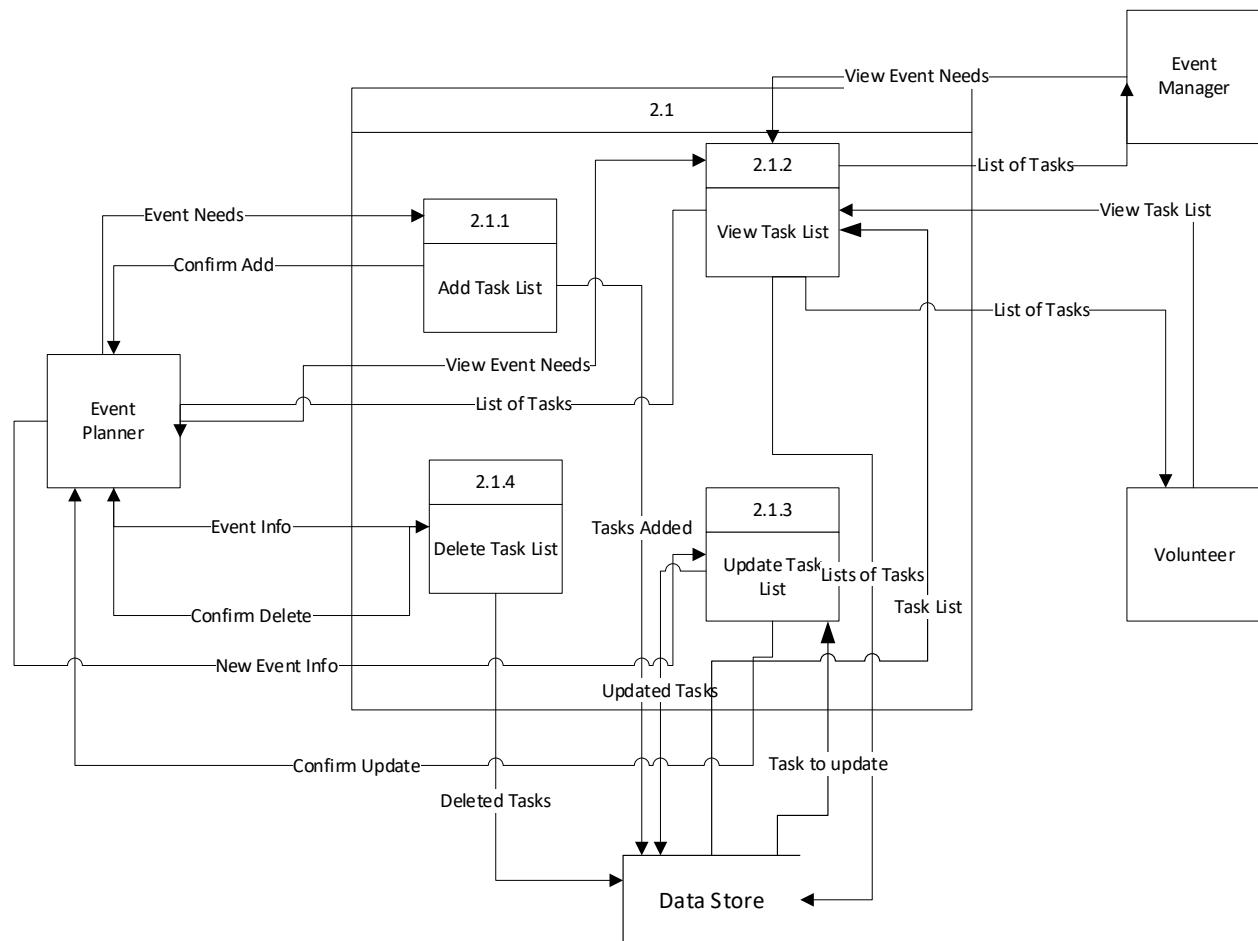
9.4.1.1 Emergency Procedure - 1.1.4



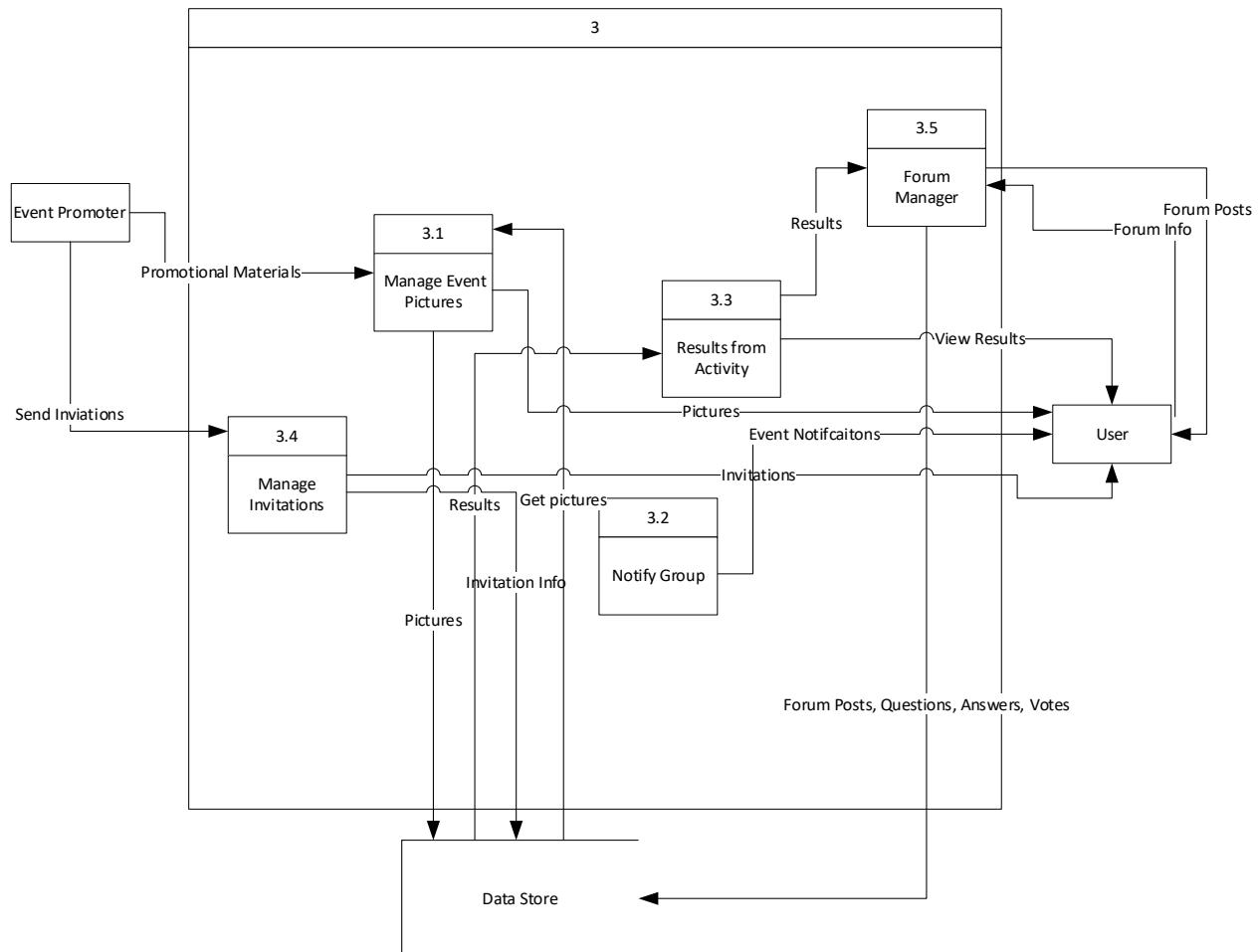
9.5 Manage Tasks - 2



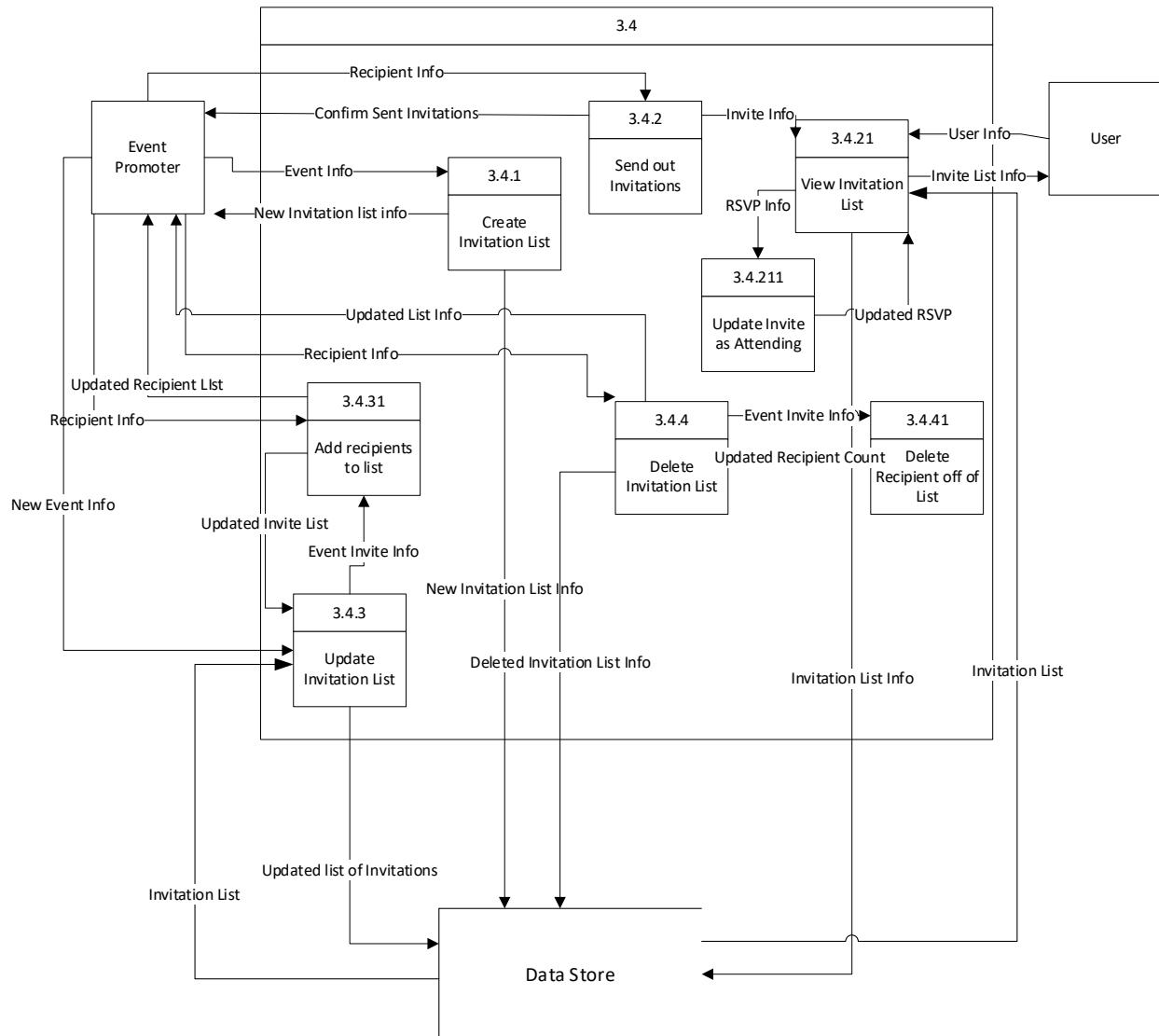
9.5.1 Task List - 2.1



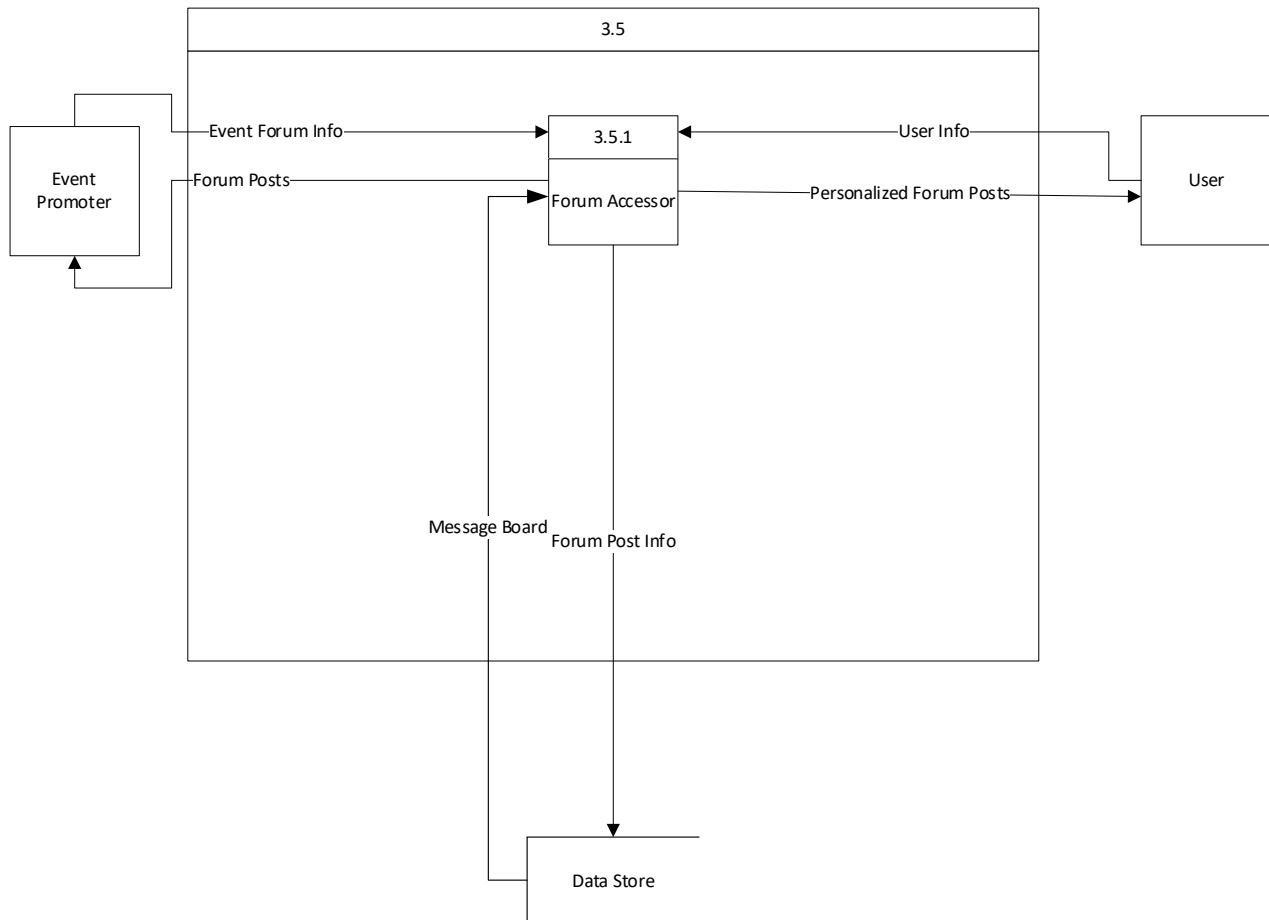
9.6 Manage Promotion – 3



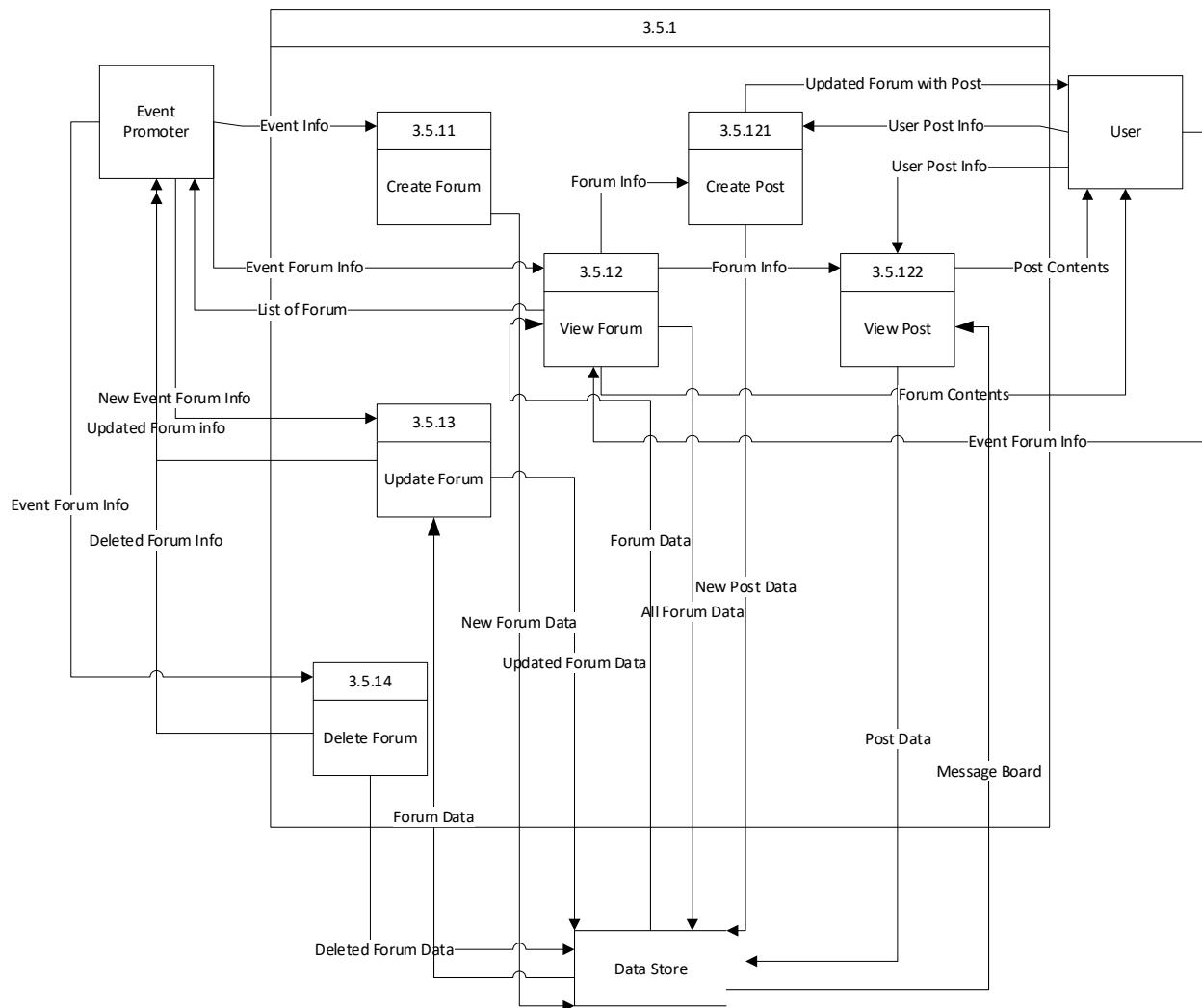
9.6.1 Manage Invitations - 3.4



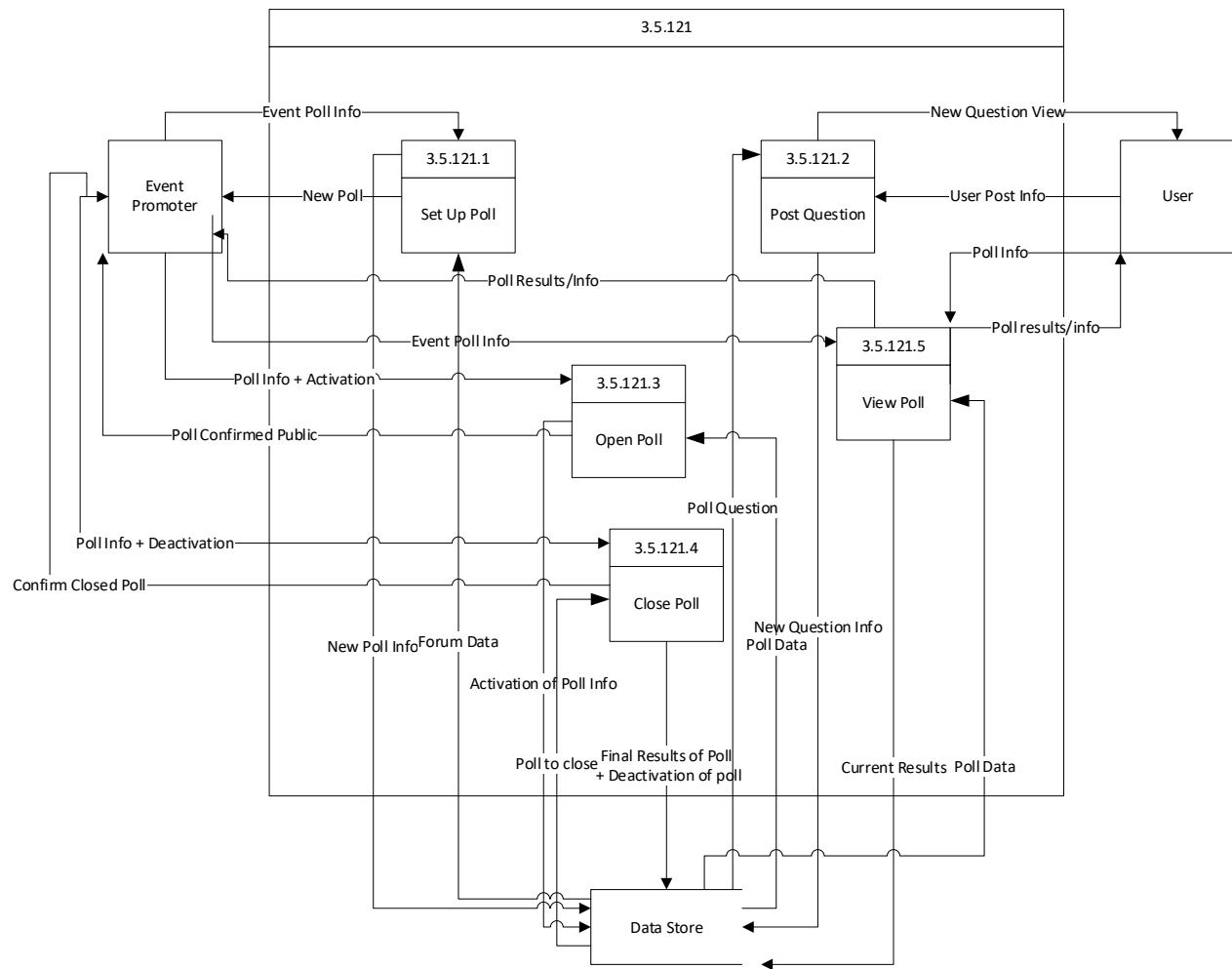
9.6.2 Forum Manager - 3.5



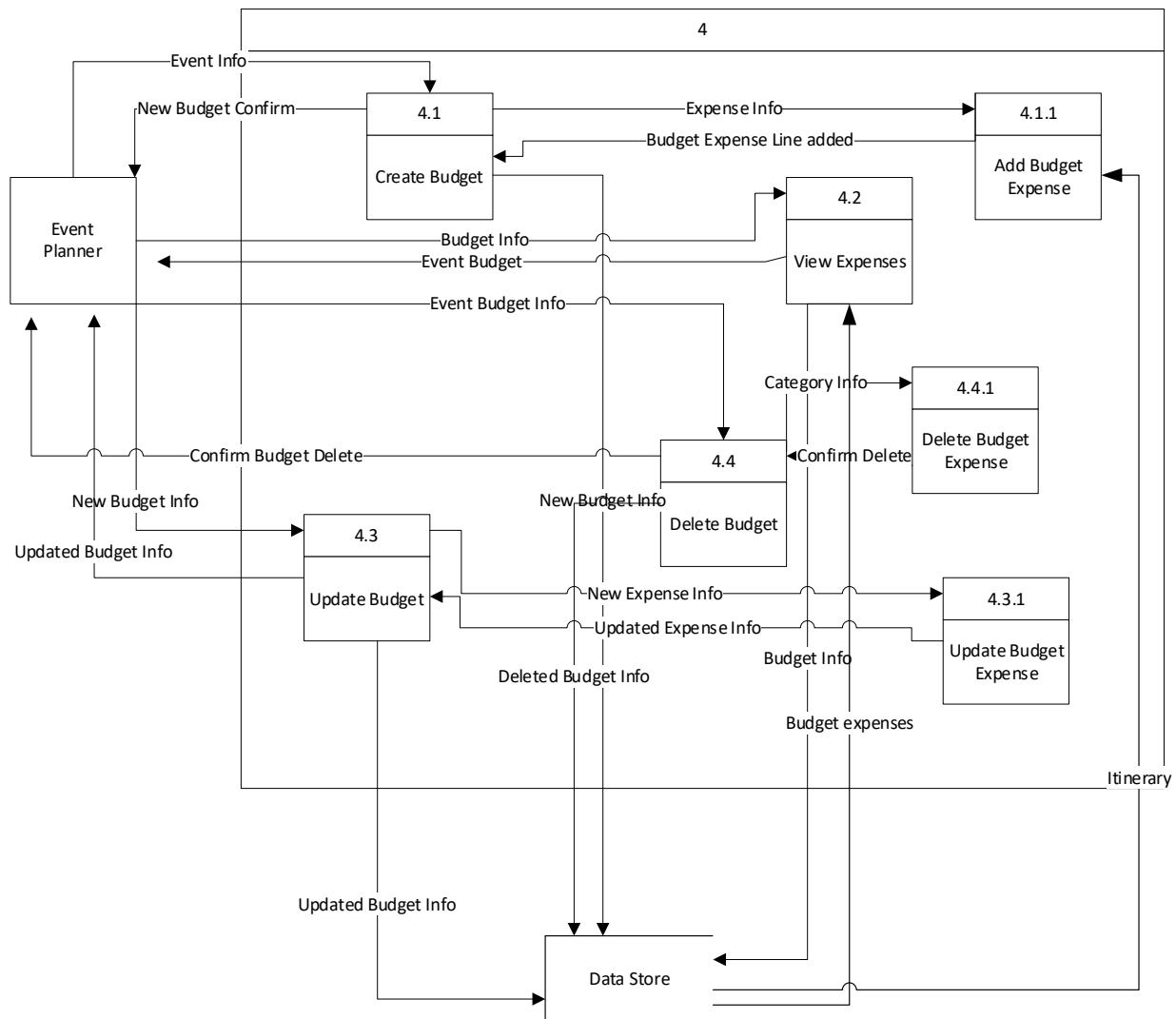
9.6.2.1 Forum - 3.5.1



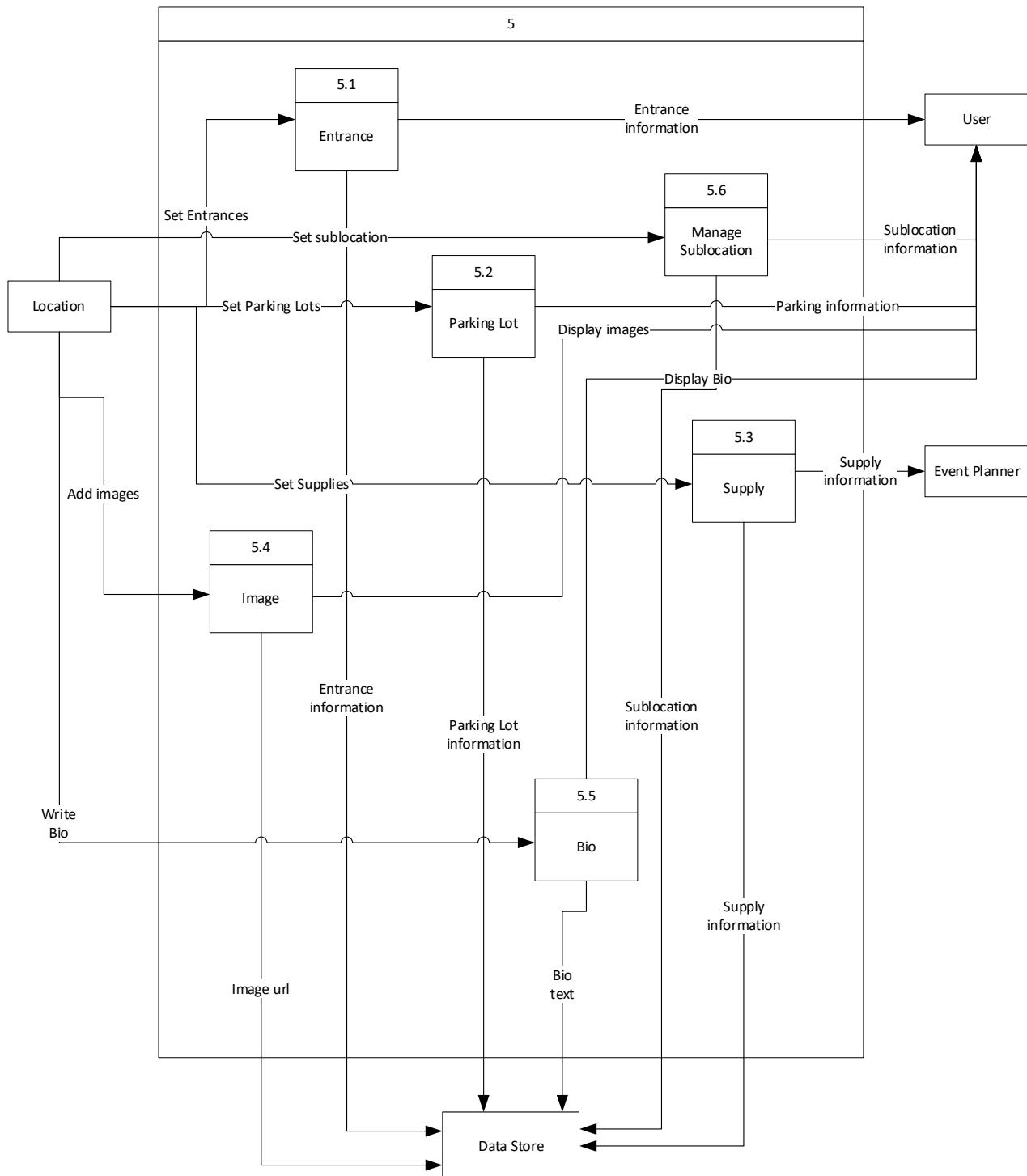
9.6.2.2 Create Post - 3.5.121



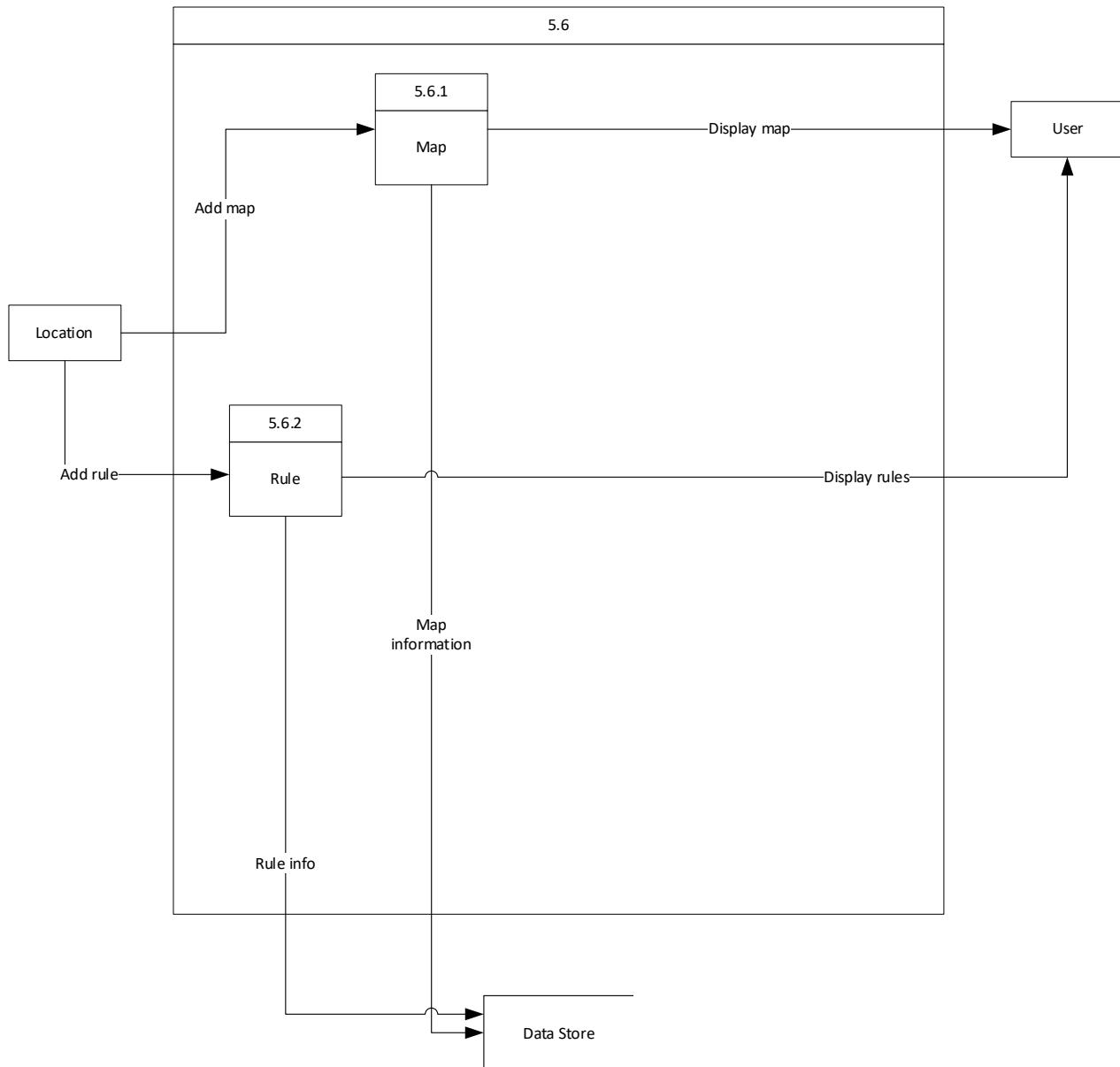
9.7 Manage Budget - 4



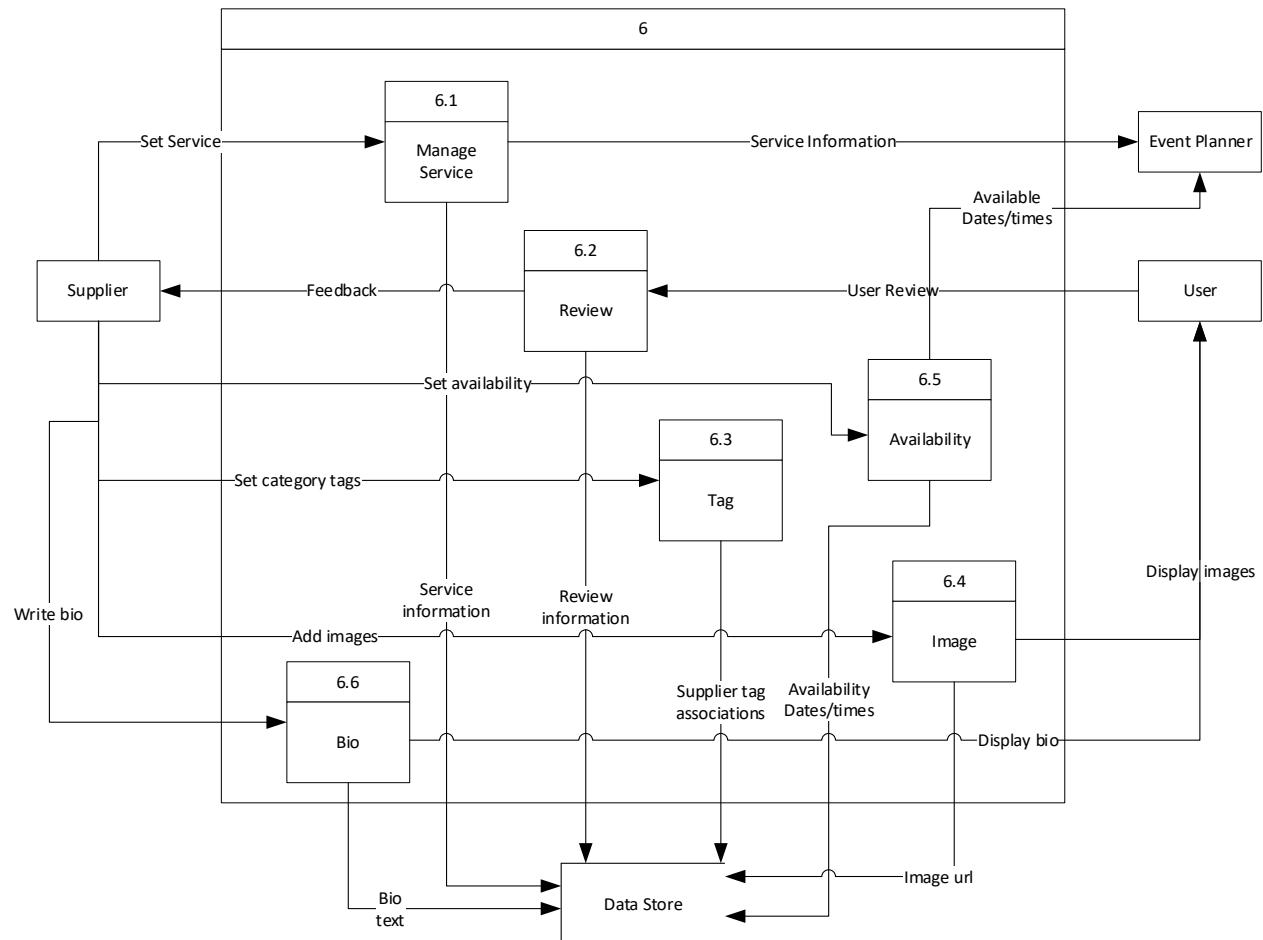
9.8 Manage Location – 5



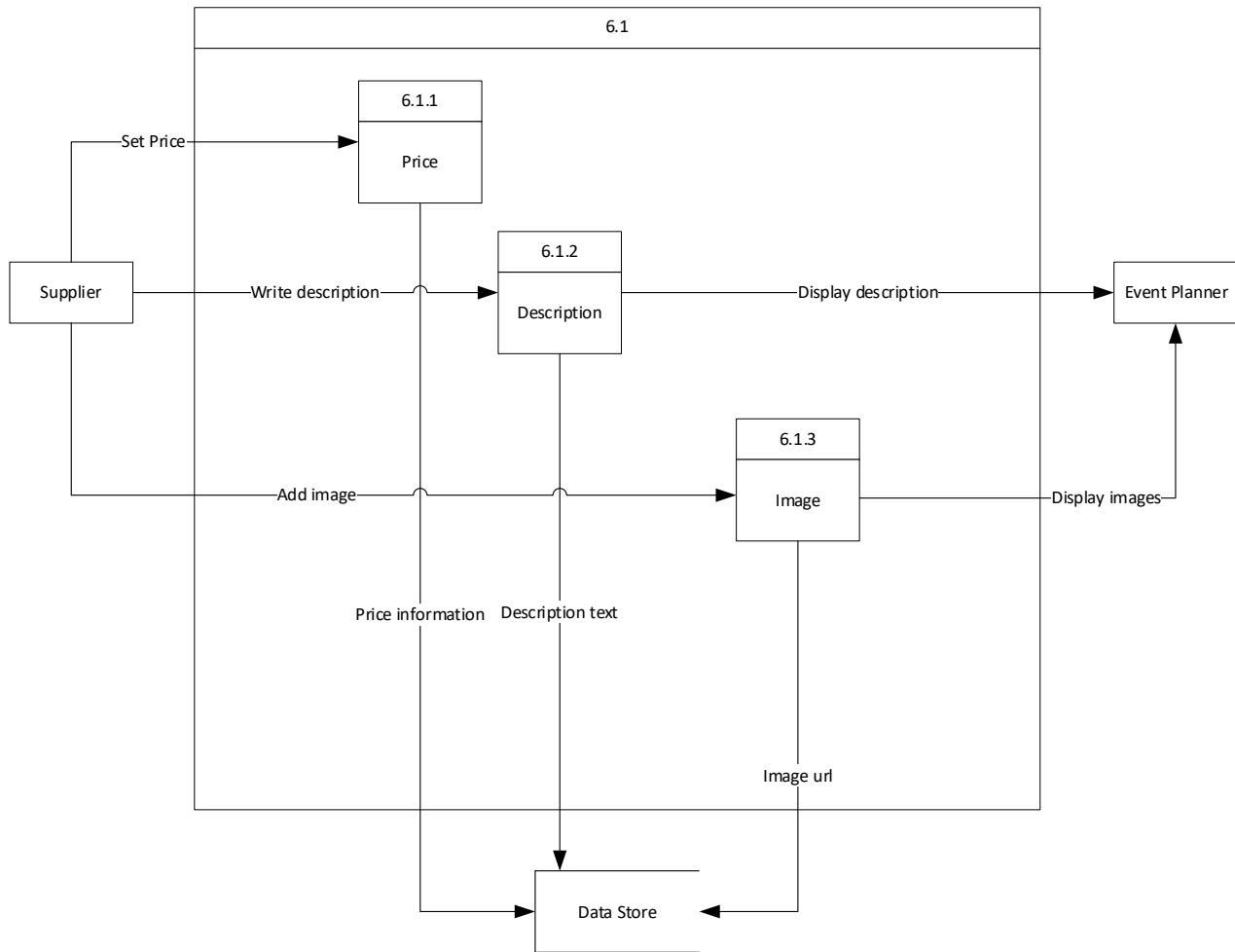
9.8.1 Manage Sublocation - 5.6



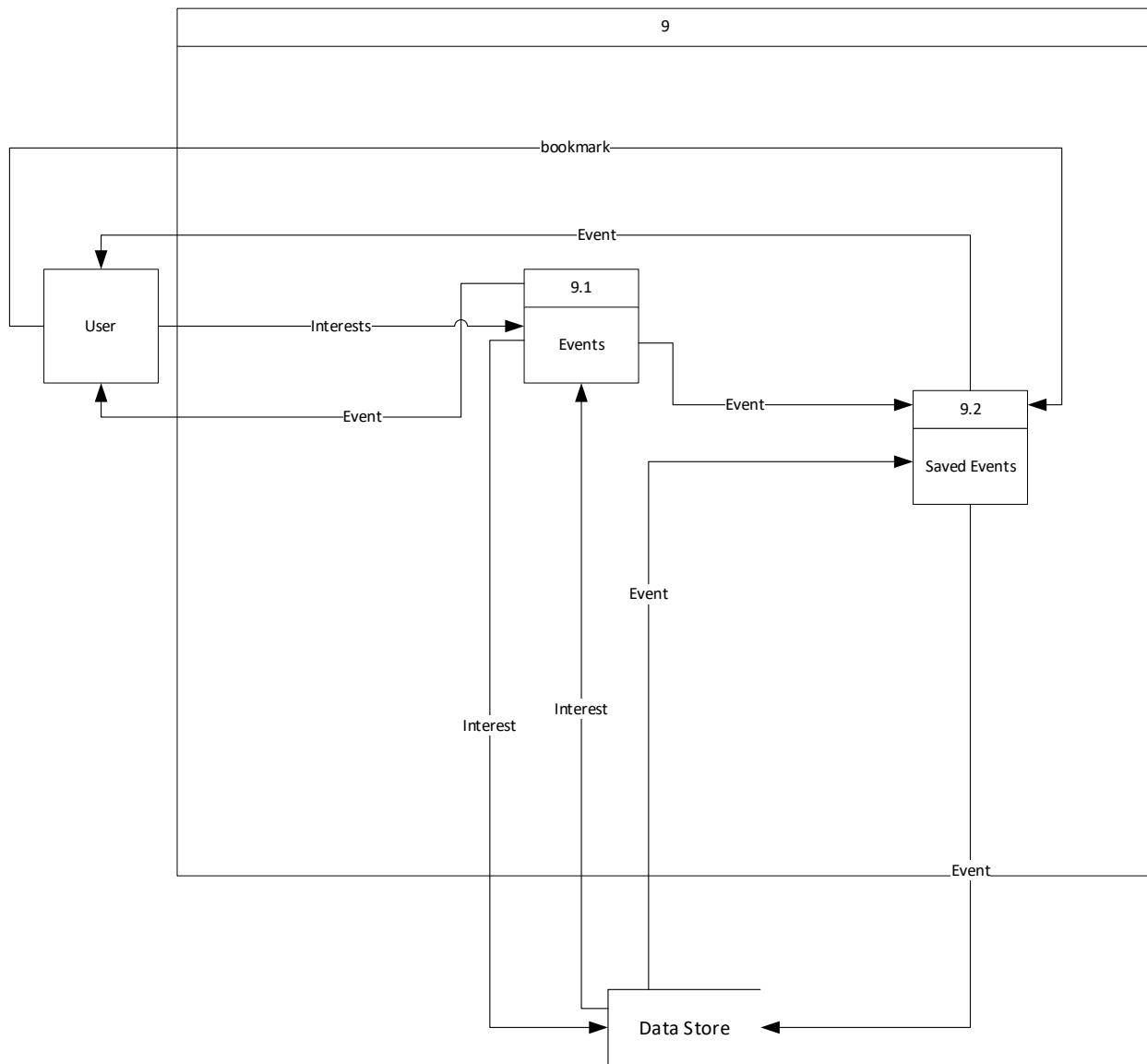
9.9 Manage Supplier – 6



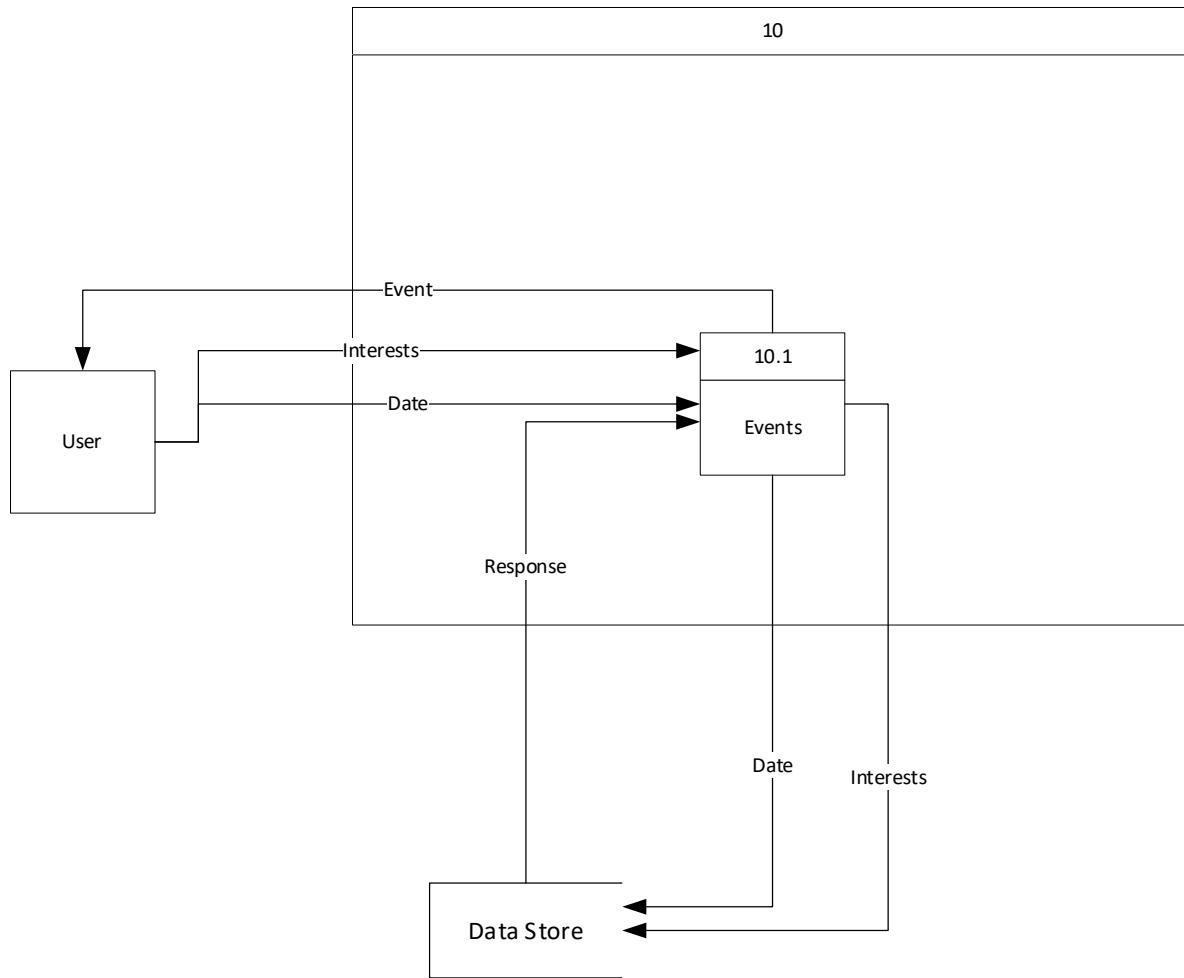
9.9.1 Manage Service - 6.1



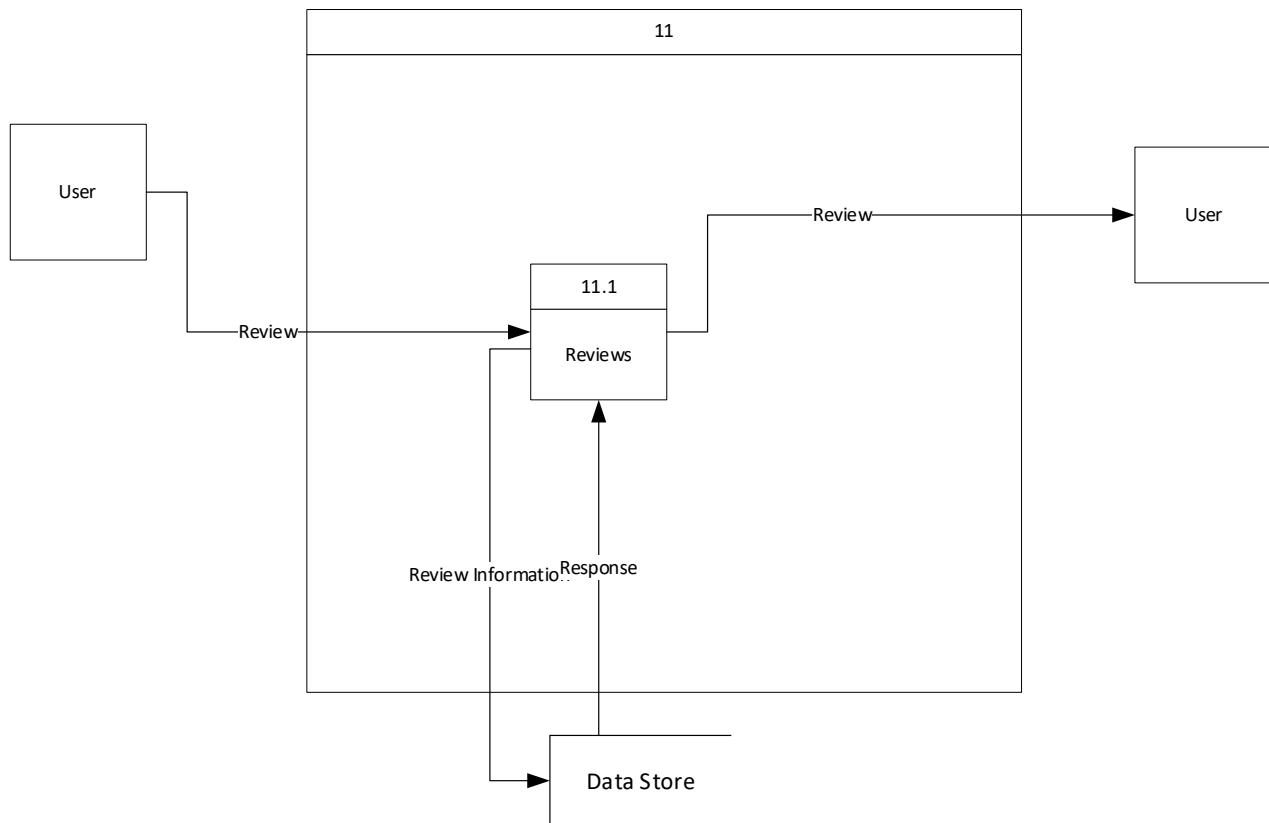
9.10 Saved Events Management – 9



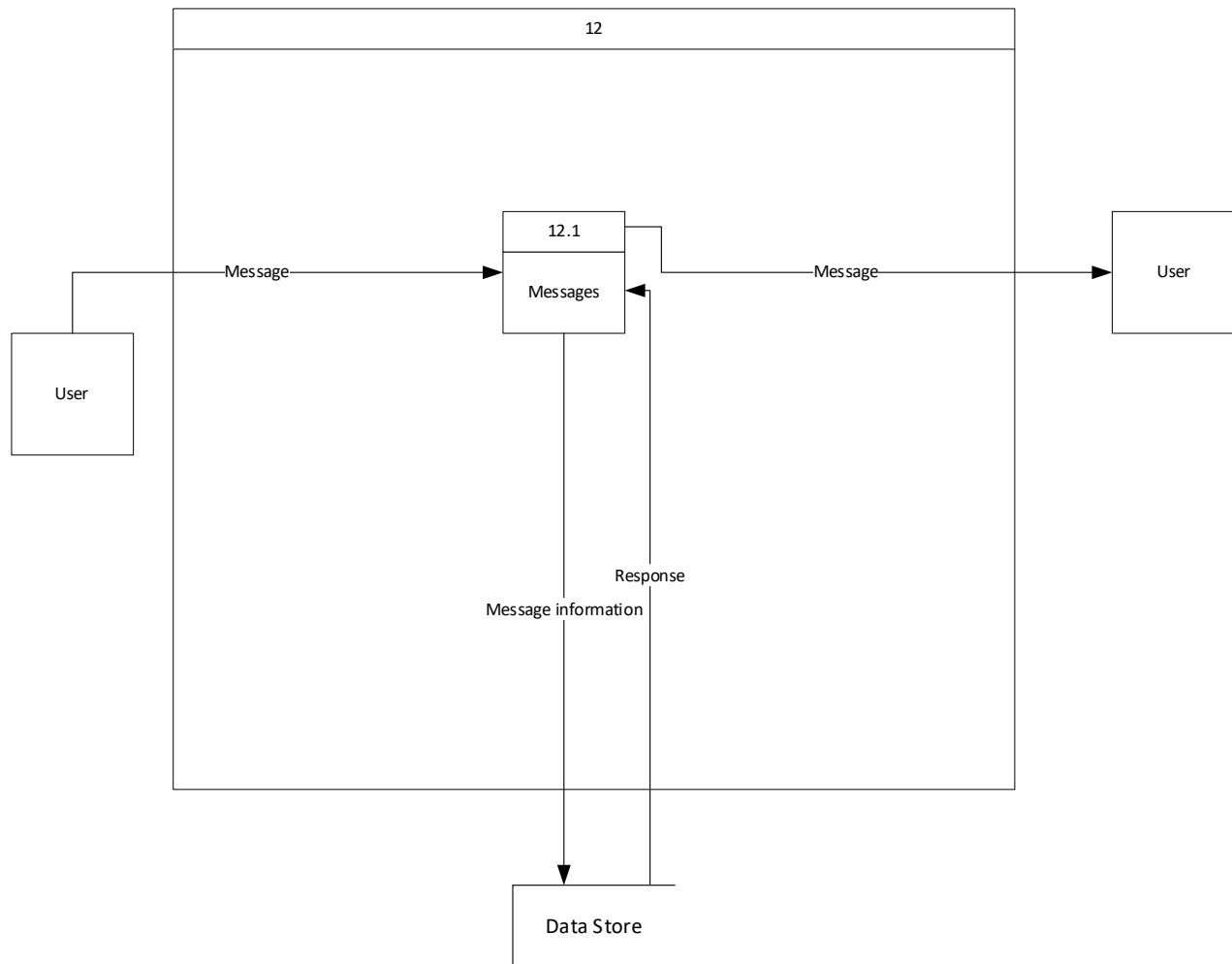
9.11 Search Event Management – 10



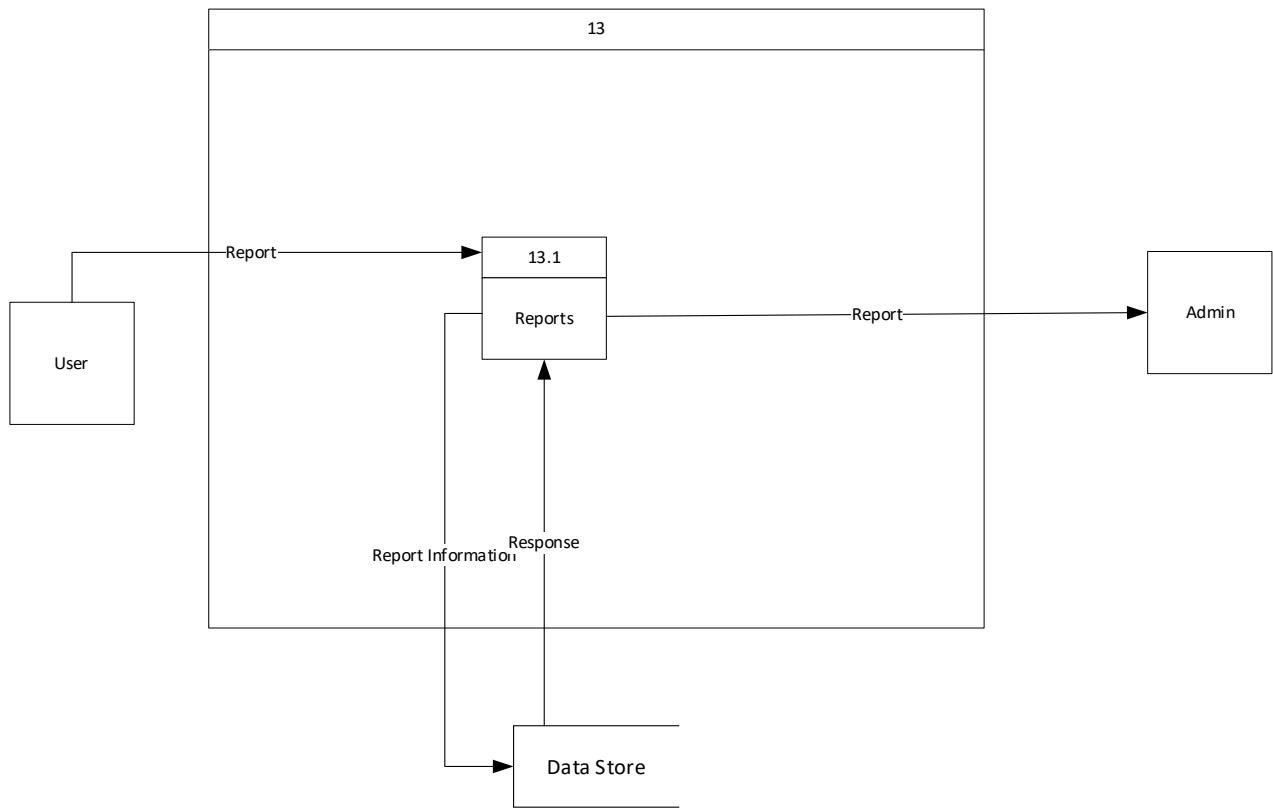
9.12 Review Management – 11



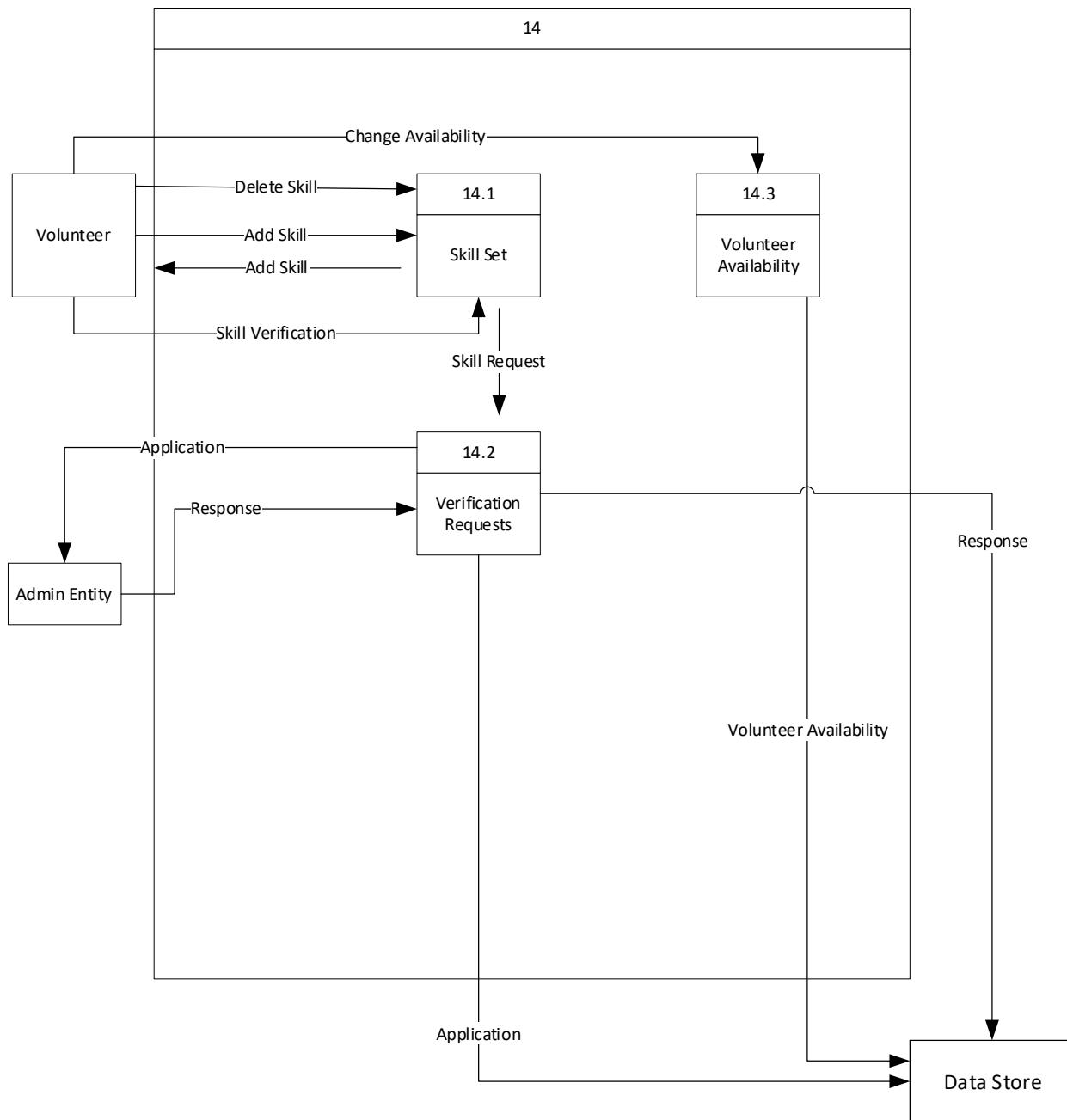
9.13 Message Management – 12



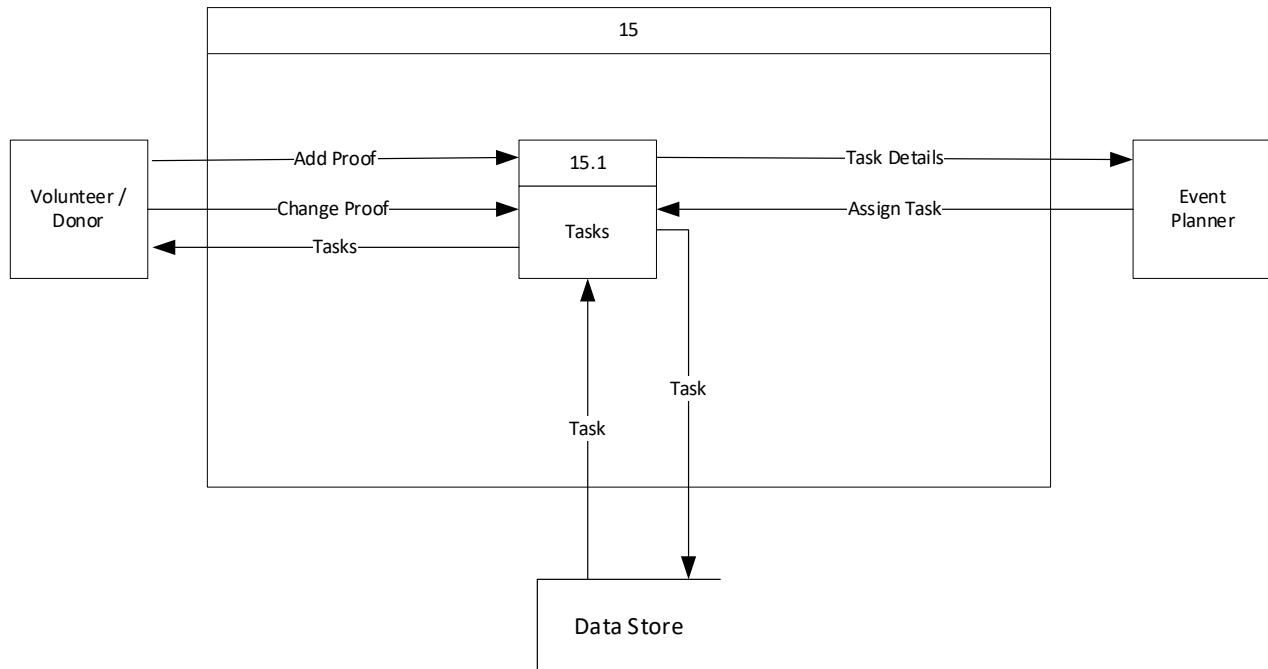
9.14 Report Management – 13



9.15 Volunteer Details – 14



9.16 Event Tasks – 15



10 Sequence Diagrams

10.1 Sequence Diagrams Summary

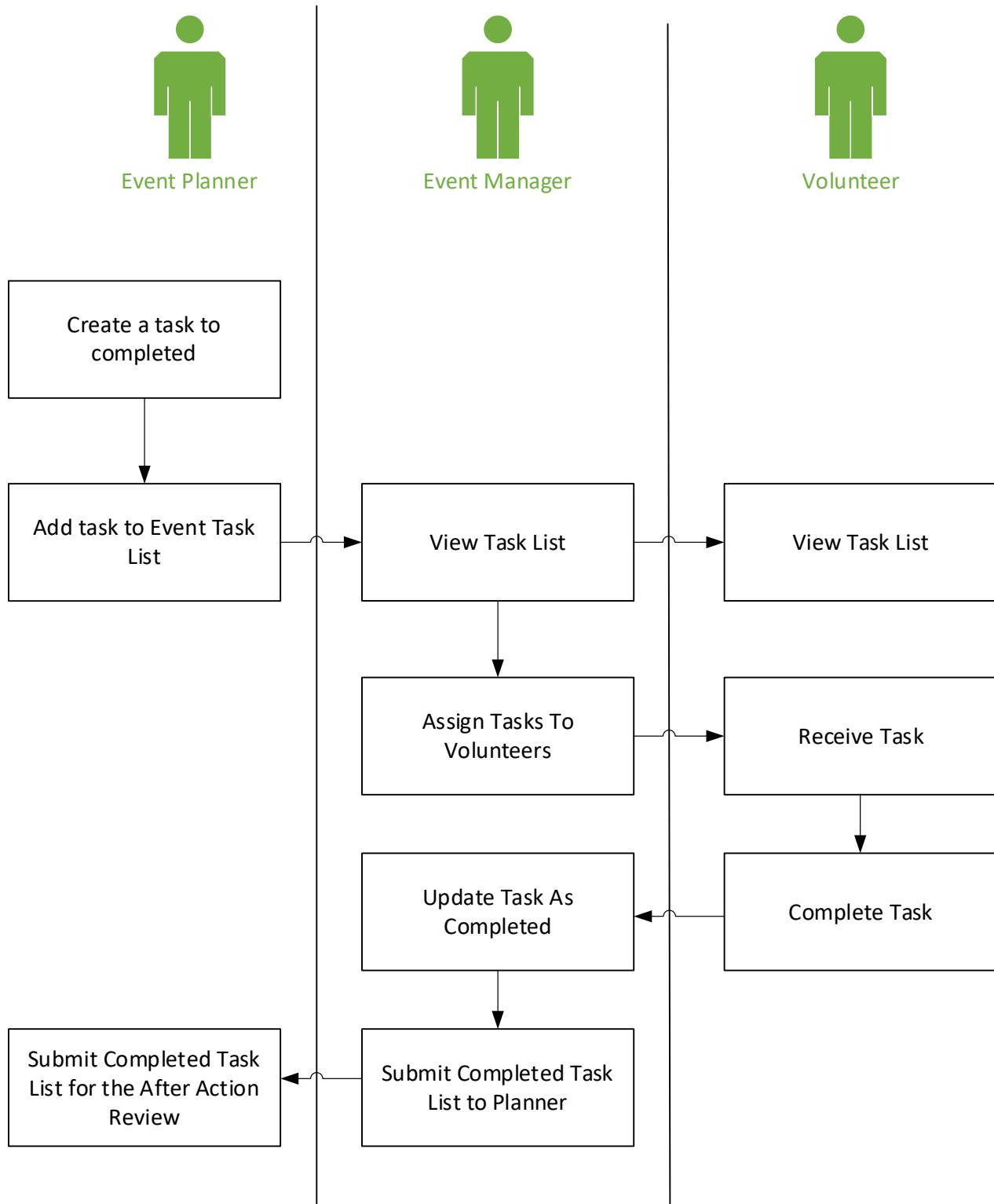
The purpose of the sequence diagrams section is to show how a single event flows between all involved actors from beginning to end. It shows how the data is sent between actors and the order in which the tasks involved take place.

It shows the progression of an event through time. Some events require back and forth between actors, but some tasks may move to the same actor for the next step. The sequence diagram helps to visualize the timeline of a single event.

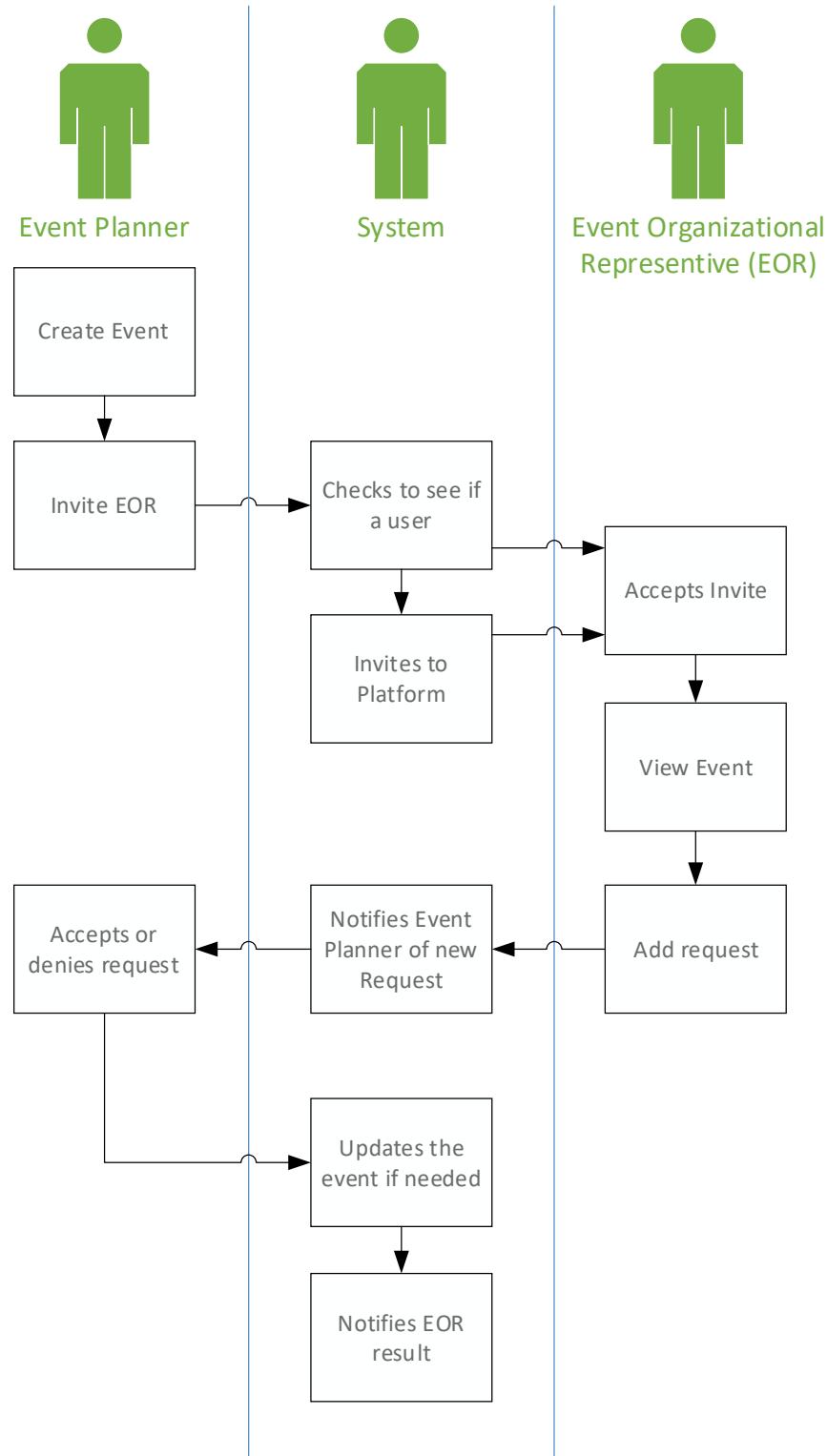
The sequence diagrams are sectioned into groups based on the primary actors in the sequence.

10.2 Event Sequence Diagrams

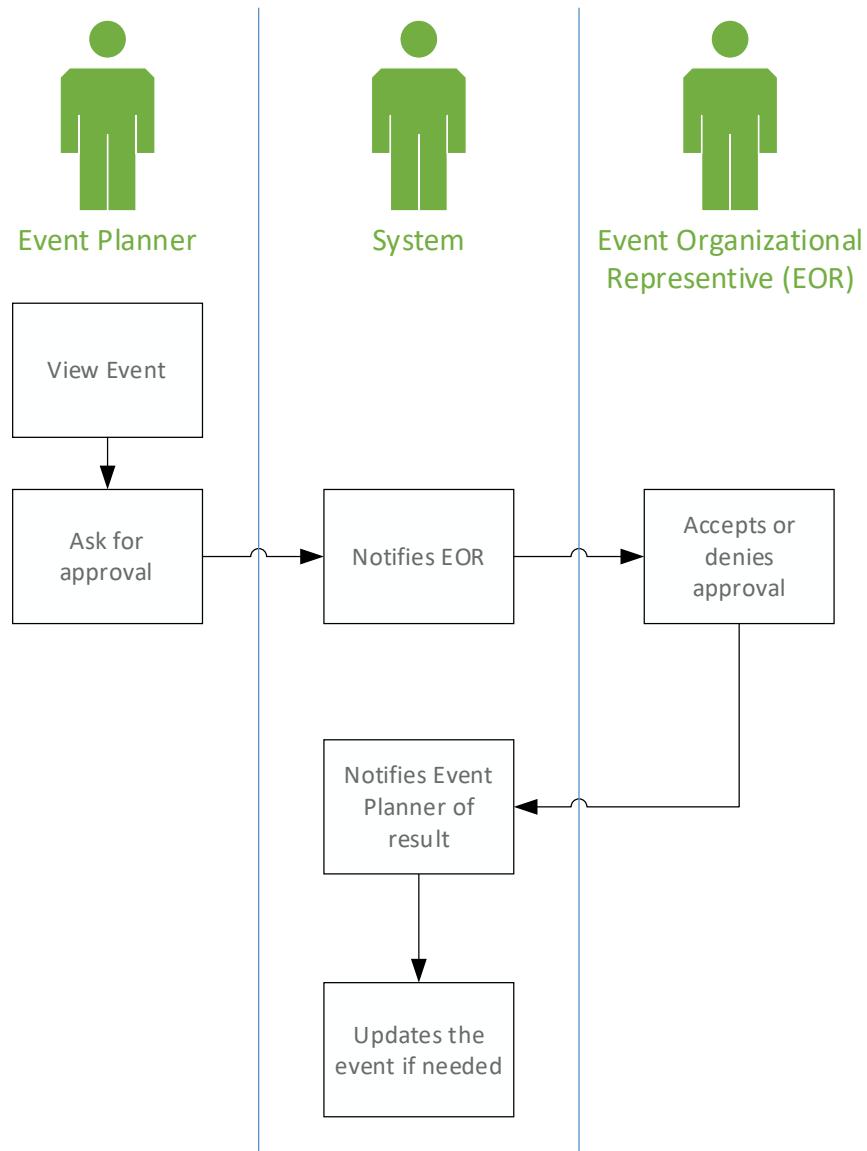
10.2.1 Event Tasks



10.2.2 Suggestions

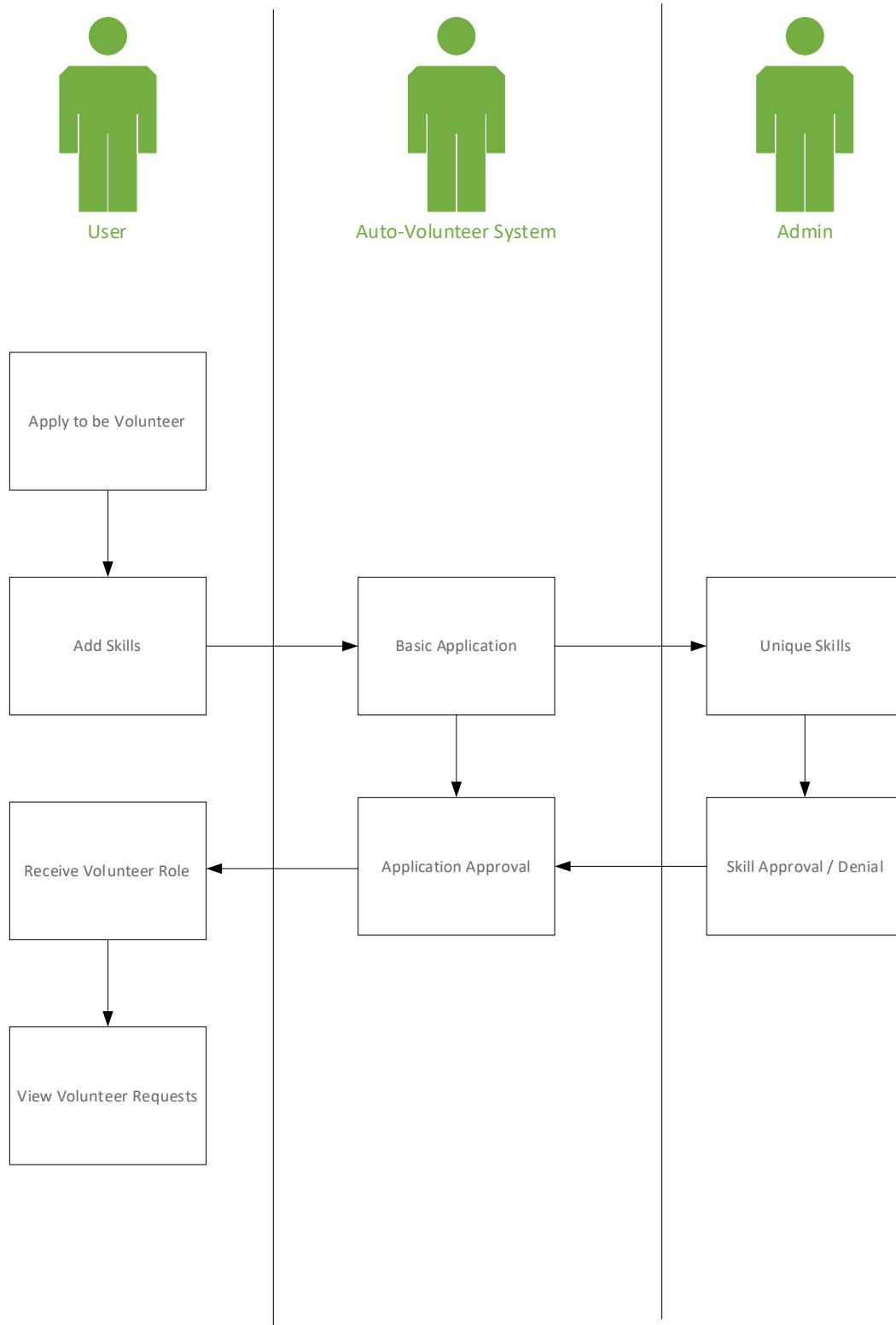


10.2.3 Approvals

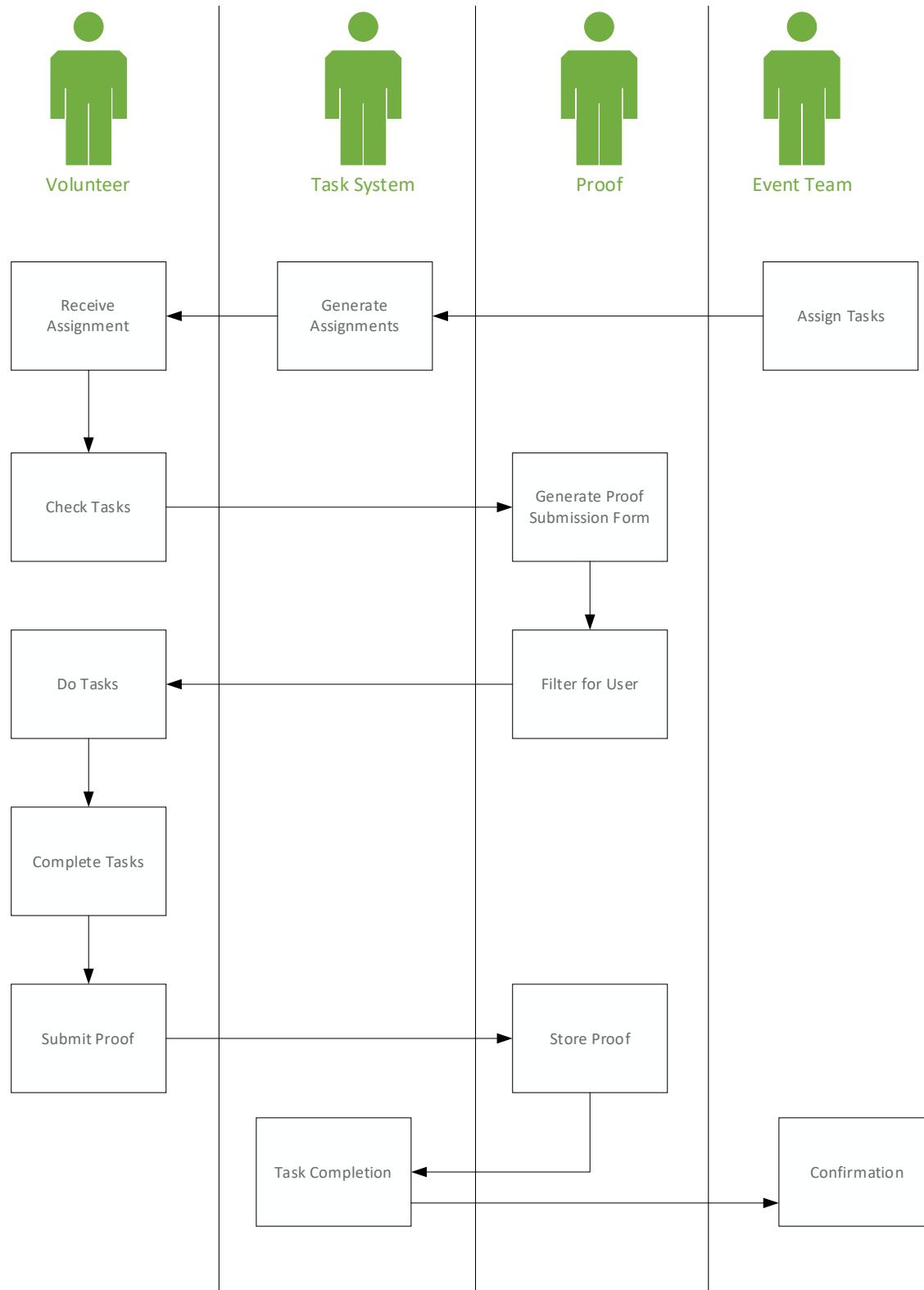


10.3 Volunteer Sequence Diagrams

10.3.1 Volunteer Applications

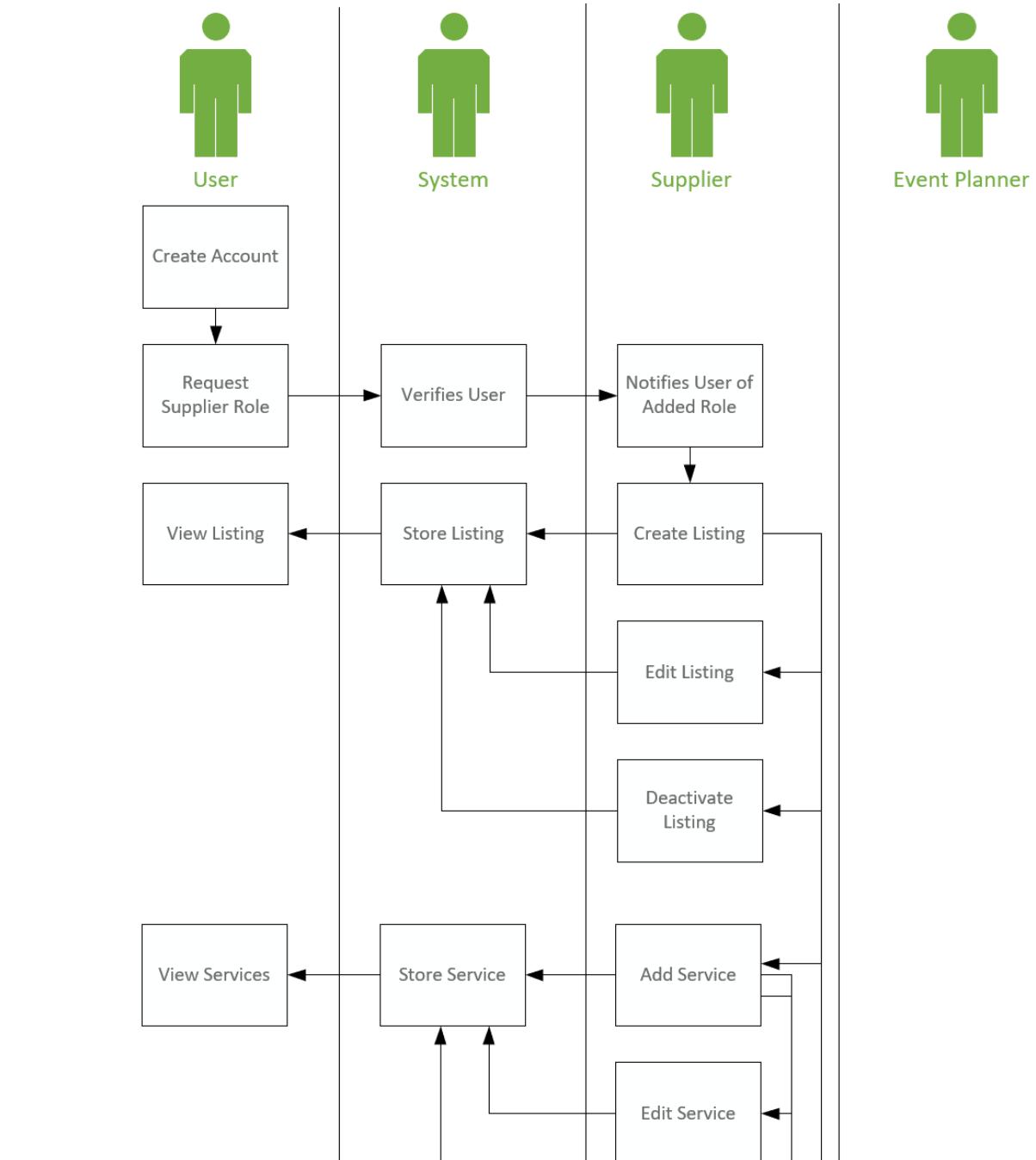


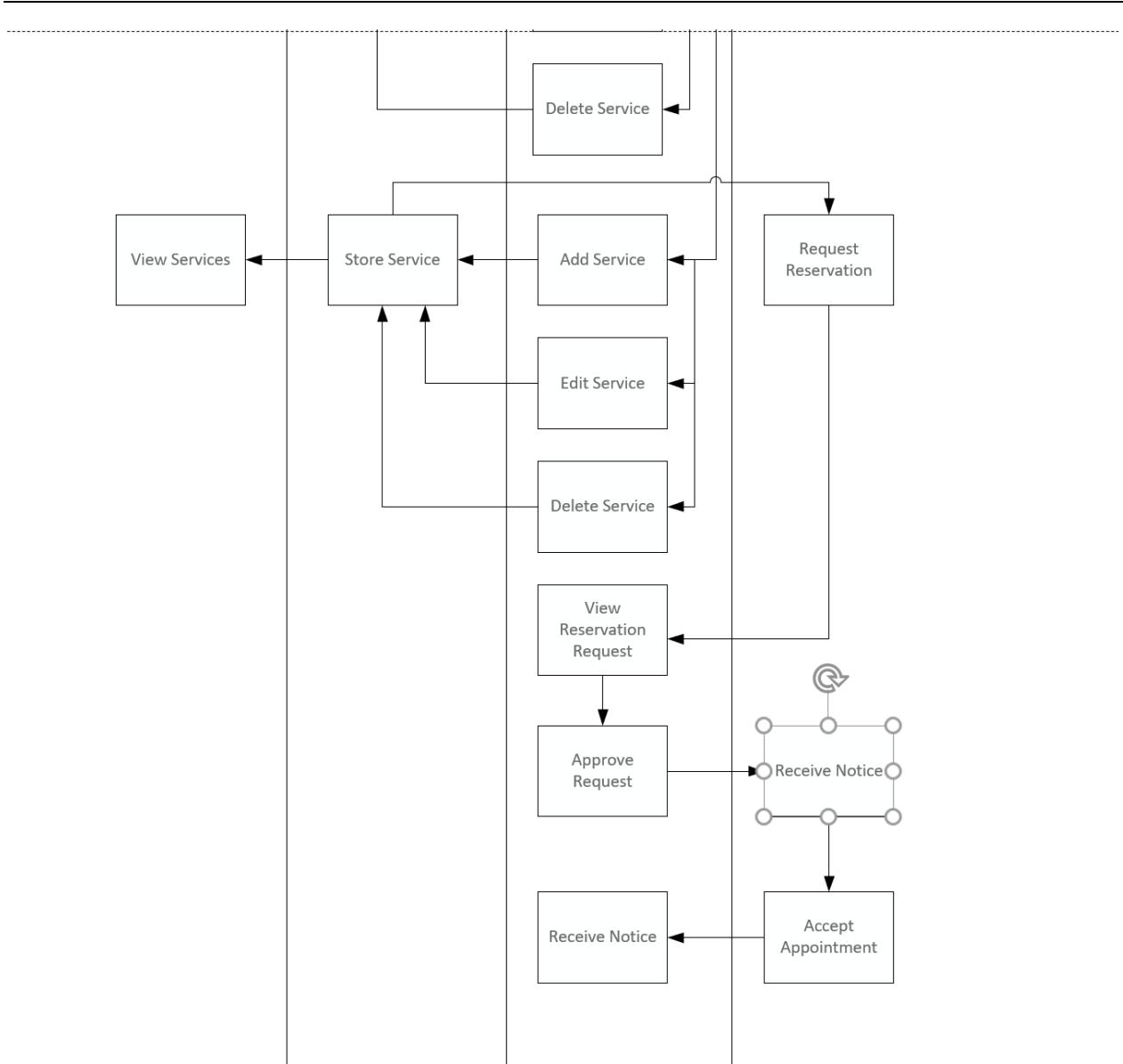
10.3.2 Volunteer Tasks



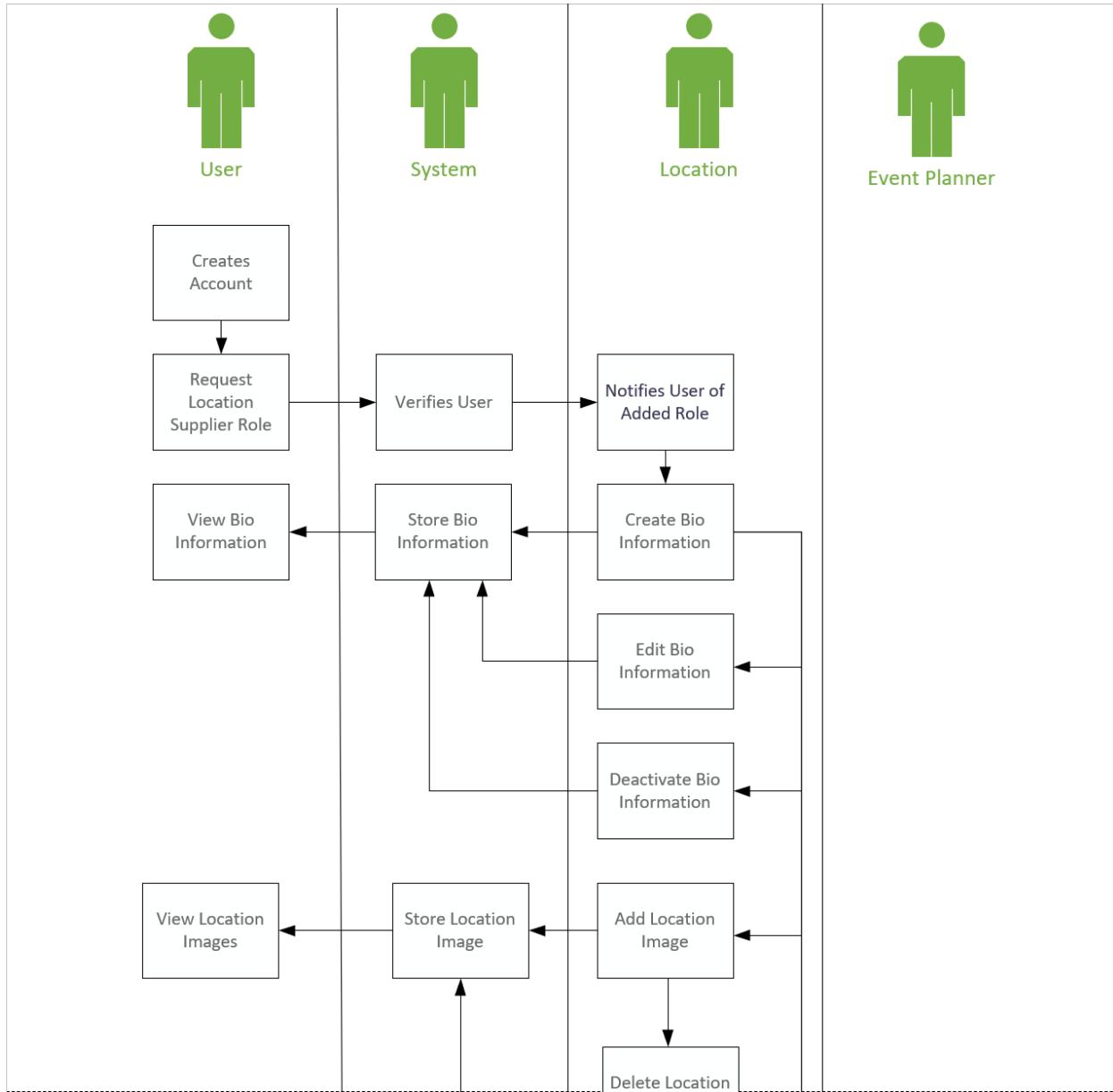
10.4 Supplier Sequence Diagrams

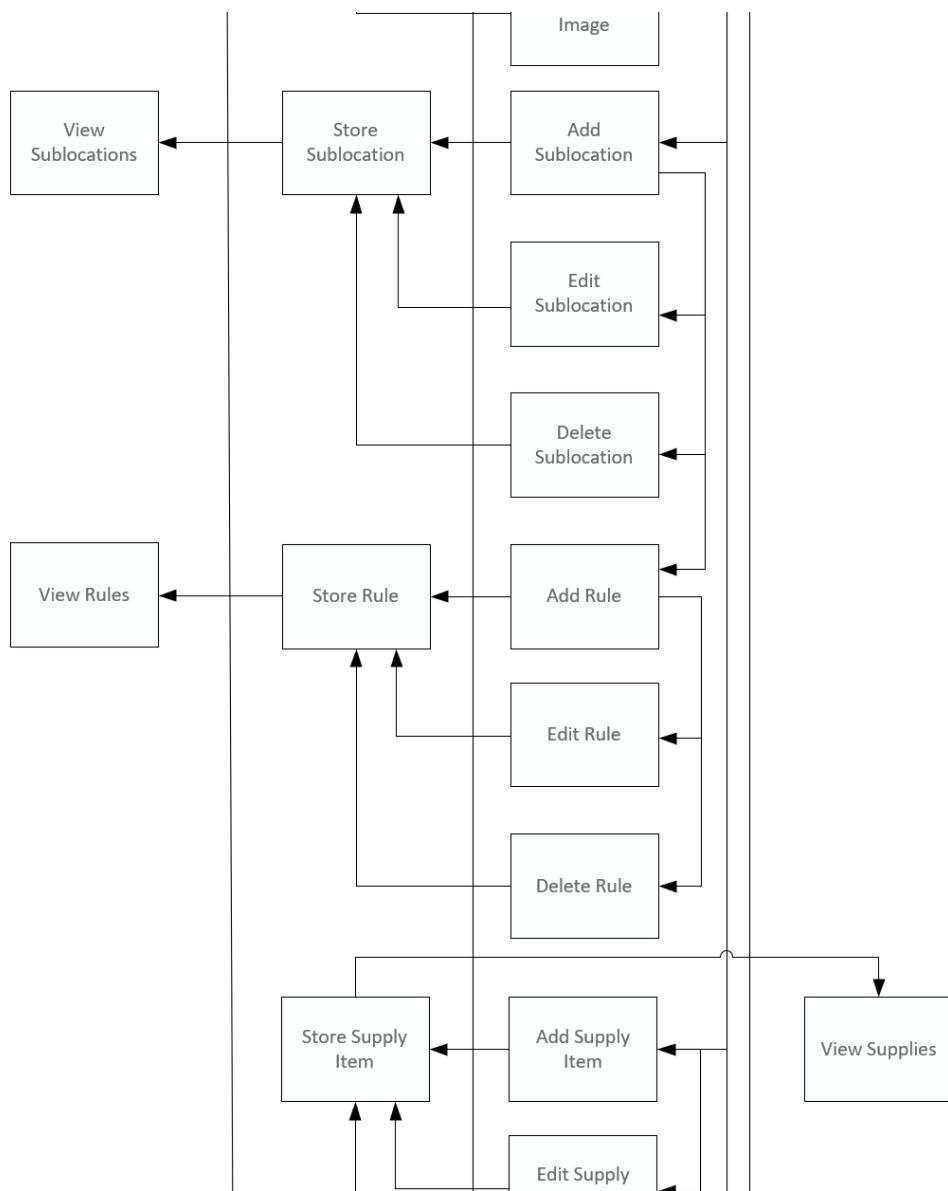
10.4.1 Suppliers

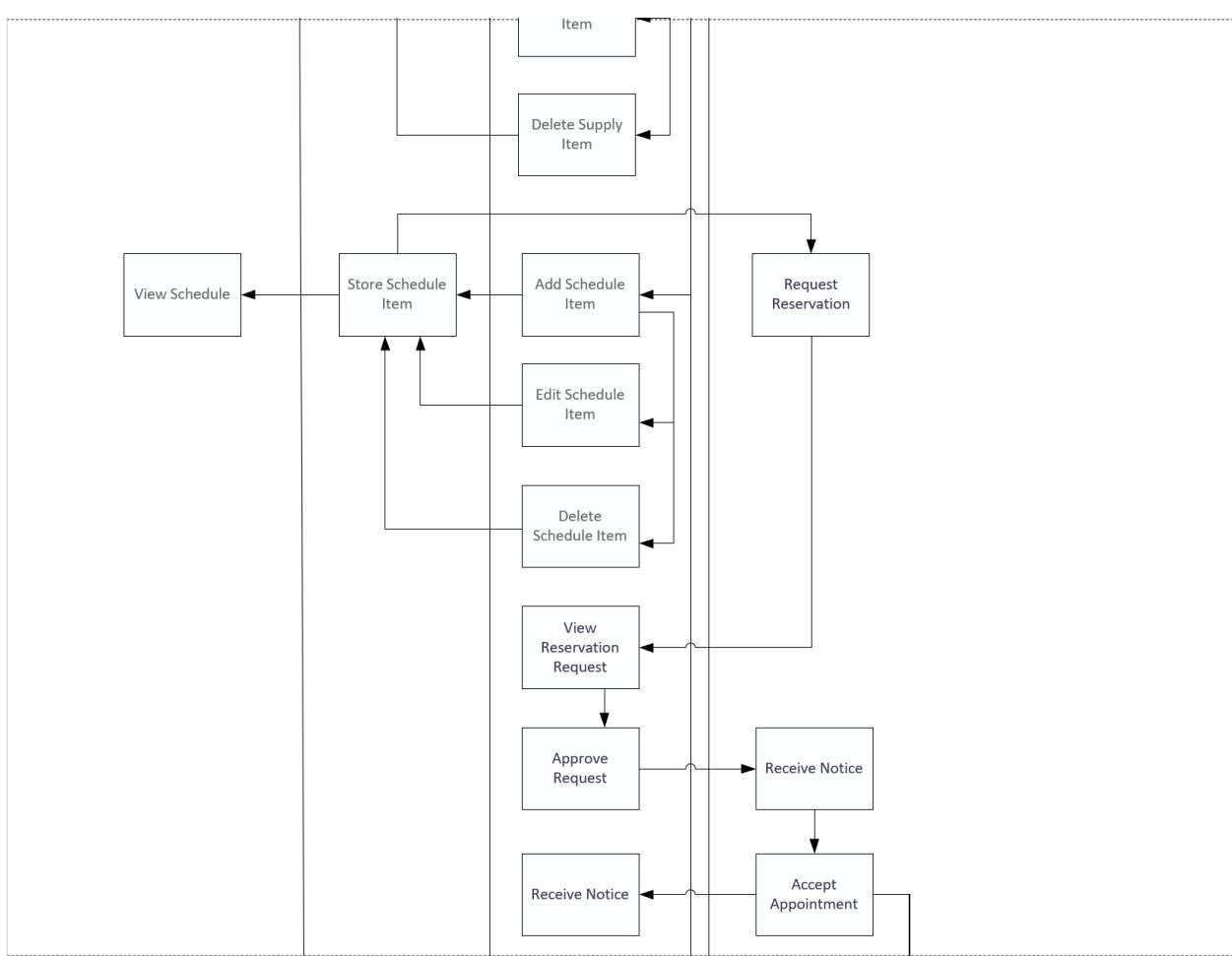


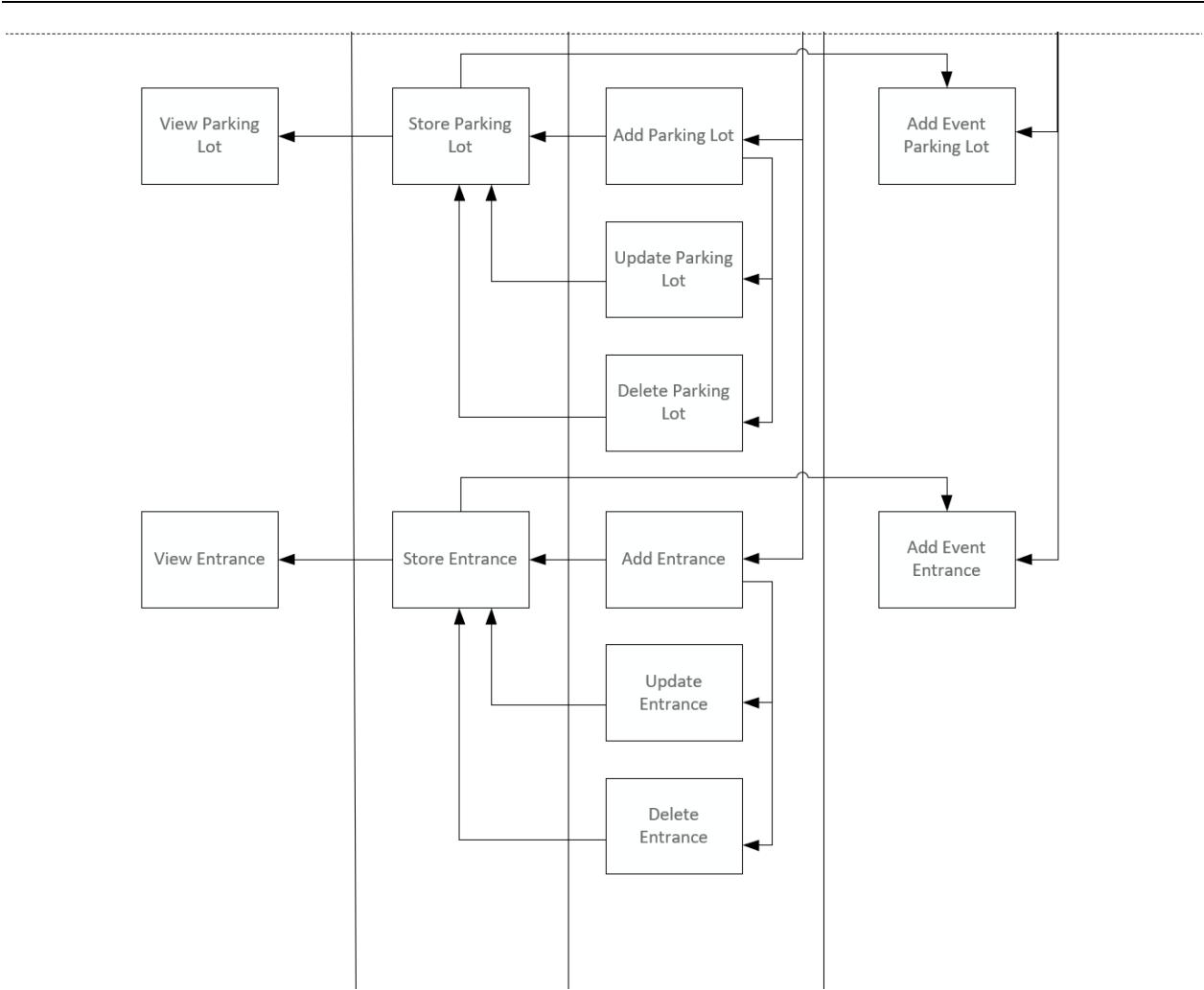


10.4.2 Locations



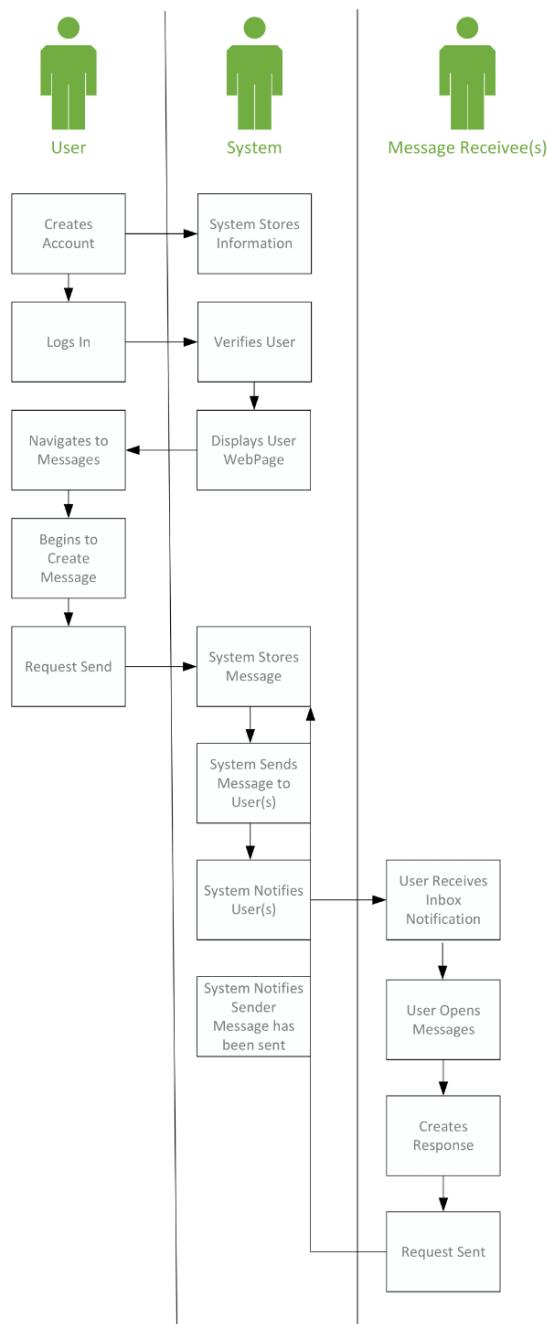




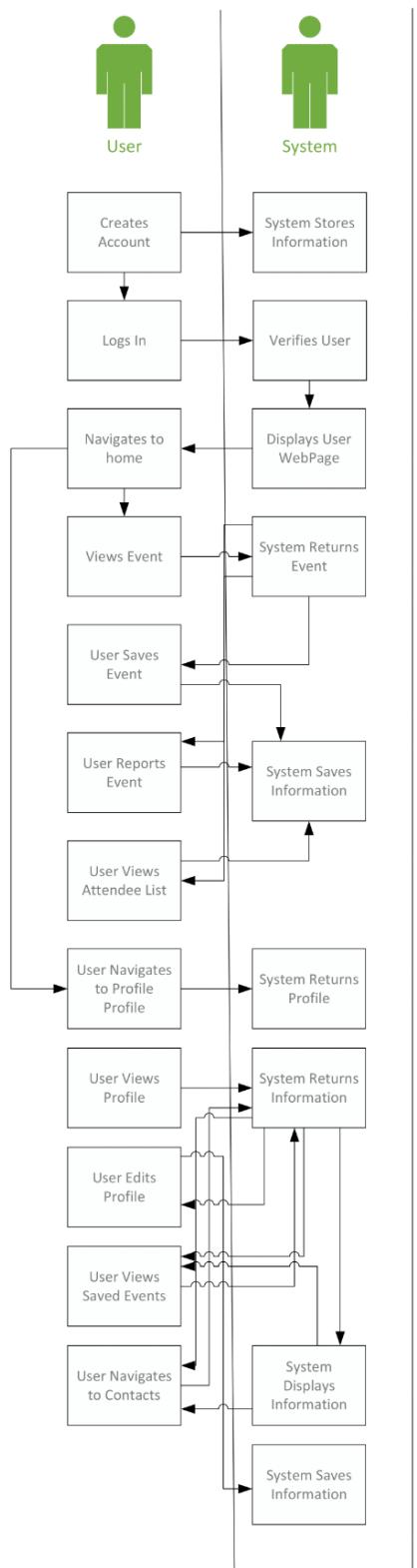


10.5 Attendee Sequence Diagrams

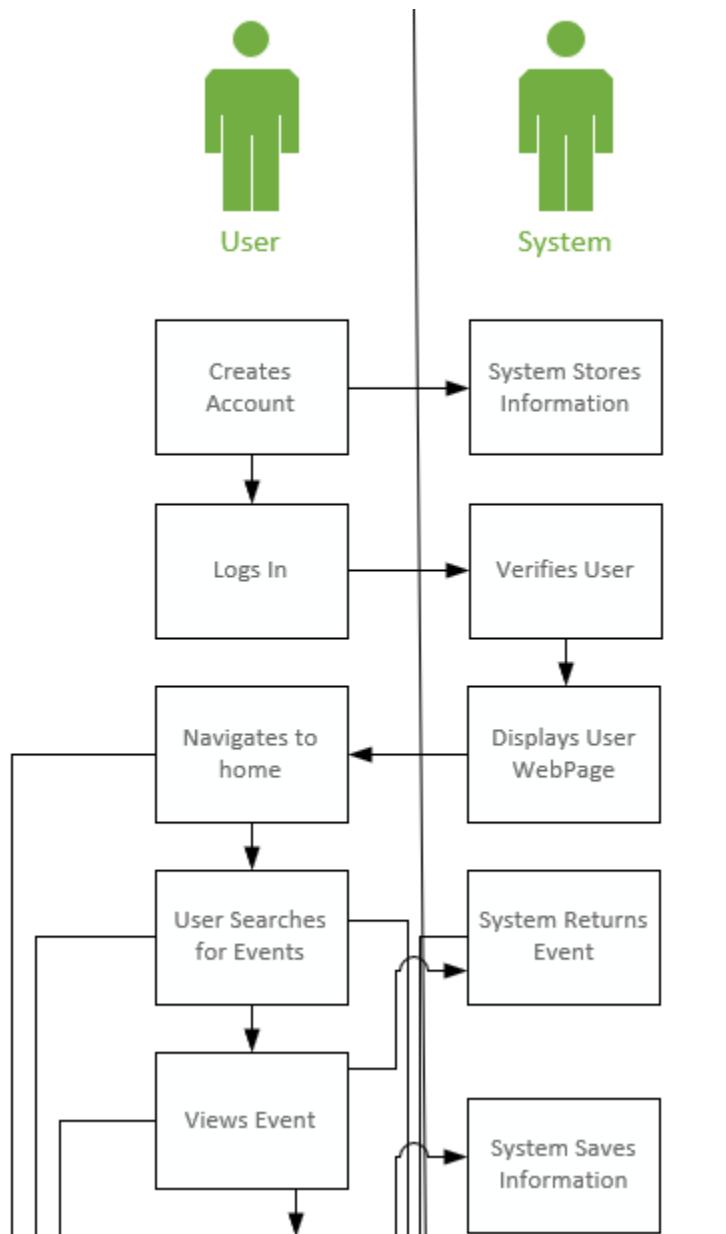
10.5.1 Messages

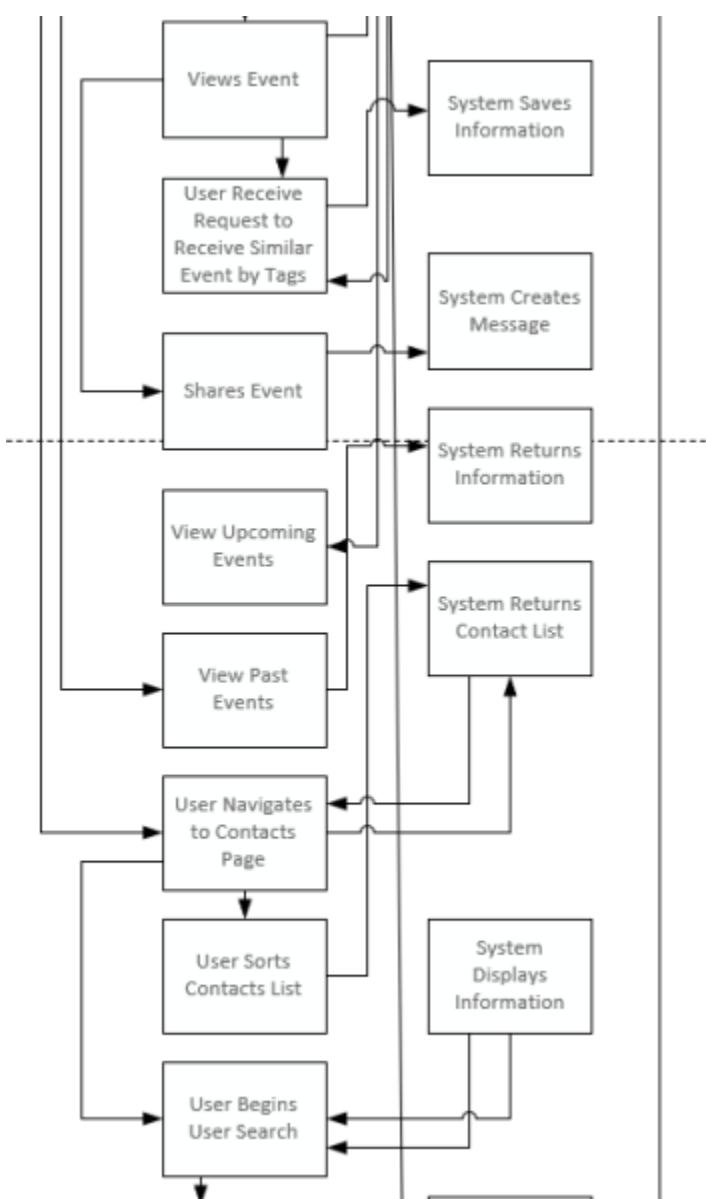


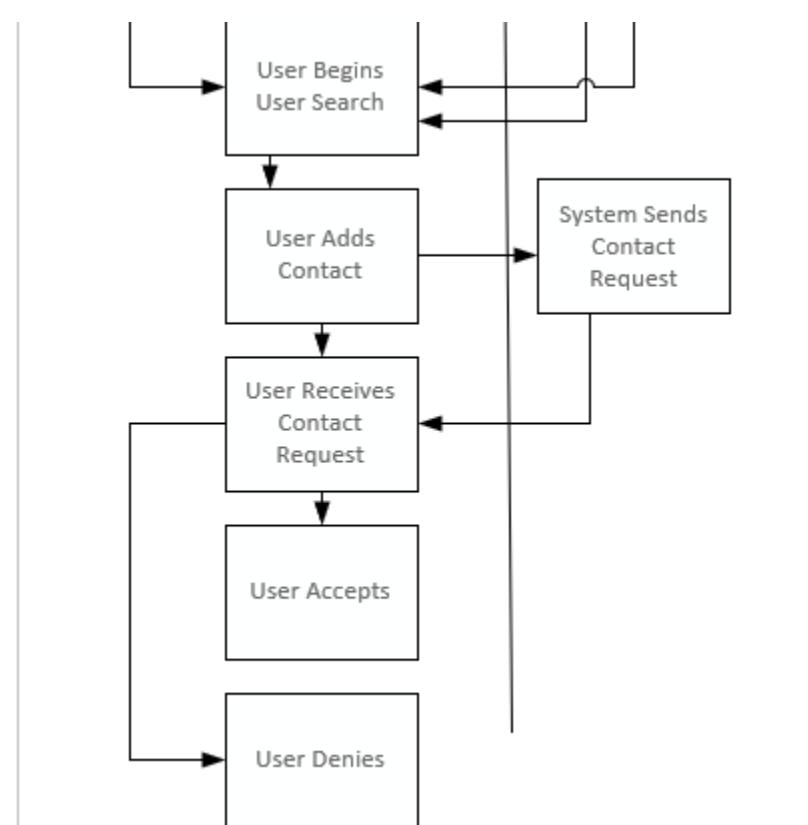
10.5.2 Profile Commands



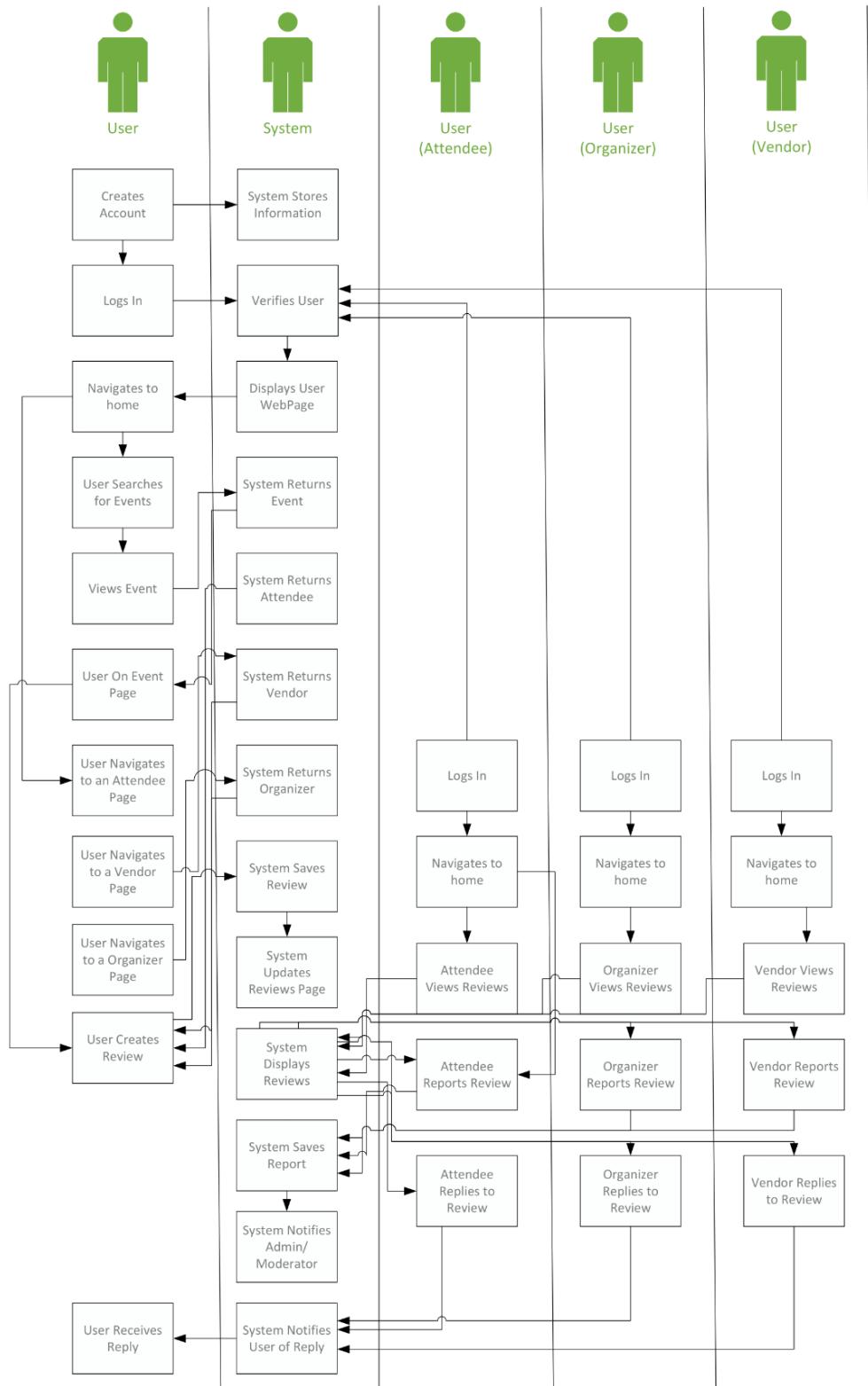
10.5.3 Event and Contact Searches







10.5.4 Reviews



11 Acknowledgements

We would like to thank our instructors at Kirkwood Community College in the Computer Software Development program. We would like to thank Jim Glasgow for teaching us how to create this documentation.