|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Software Requirement Specifications**  Volunteer Organizer with Management Entity  (VOWME)  Version: [1.4]   |  |  | | --- | --- | | Project Code | VOWME | | Supervisor | Mr. Muhammad Shahzad | | Co Supervisor |  | |  |  | | Project Team | Jibran Tariq (14K- 3408) | | Submission Date |  | |

Document History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Name of Person** | **Date** | **Description of change** |
| 1.0 | Jibran Tariq | 04th March 2017 | Official First Draft |
| 1.1 | Jibran Tariq | 25th March 2017 | Document Revisions Fixed formatting issues and usage inconsistencies |
| 1.2 | Jibran Tariq | 14th April 2017 | Review Changes Made corrections based on |
| 1.3 | Jibran Tariq | 21st April 2017 | SRS Baseline Final review before baseline |
| 1.4 | Jibran Tariq | 10th May 2017 | Updated user interface mockups and use cases |

Distribution List

|  |  |  |
| --- | --- | --- |
| **Name** | **Role** | |
| Mr. Muhammad Shahzad | | Supervisor |
|  | | Co- Supervisor |

Document Sign-Off

|  |  |  |
| --- | --- | --- |
| **Version** | **Sign-off Authority** | **Sign-off Date** |
| 1.4 | Mr. Muhammad Shahzad |  |
|  |  |  |

**Table of Contents**

[1. Introduction 10](#_Toc481680319)

[1.1. Purpose of Document 10](#_Toc481680320)

[1.2. Intended Audience 10](#_Toc481680321)

[1.4 Document Convention 10](#_Toc481680322)

[2. Overall System Description 11](#_Toc481680323)

[2.1. Project Background 11](#_Toc481680324)

[2.2. Project Scope 11](#_Toc481680325)

[2.2.1. Features 11](#_Toc481680326)

[2.3. Not in Scope 12](#_Toc481680327)

[2.4. Project Objectives 12](#_Toc481680328)

[2.5. Stakeholders 12](#_Toc481680329)

[2.6. Operating Environment 12](#_Toc481680330)

[2.7. System Constraints 12](#_Toc481680331)

[2.8. Assumptions & Dependencies 13](#_Toc481680332)

[3. External Interface Requirements 14](#_Toc481680333)

[3.1. Hardware Interfaces 15](#_Toc481680335)

[3.2. Software Interfaces 16](#_Toc481680336)

[3.3. Communications Interfaces 16](#_Toc481680337)

[4. Functional Requirements 17](#_Toc481680338)

[4.1. Functional Hierarchy 17](#_Toc481680339)

[4.1.1. Login 17](#_Toc481680340)

[4.1.2. Logout 17](#_Toc481680341)

[4.1.3. Add Volunteer Events 17](#_Toc481680342)

[4.1.4. Delete Volunteer Opportunities 18](#_Toc481680343)

[4.1.5. Sign Up for Volunteer Opportunities 18](#_Toc481680344)

[4.1.6. Cancel Commitment 18](#_Toc481680345)

[4.1.7. Notify Volunteer 18](#_Toc481680346)

[4.1.8. Track Progress 18](#_Toc481680347)

[4.1.9. Customize Preferences 19](#_Toc481680348)

[4.1.10. Validate Members 19](#_Toc481680349)

[4.1.11. Register Volunteers 19](#_Toc481680350)

[4.1.12. Volunteer Feedback 19](#_Toc481680351)

[4.1.13. Android Application 19](#_Toc481680352)

[4.1.14. Ease of Use 20](#_Toc481680353)

[4.1.15. iOS Mobile Application 20](#_Toc481680354)

[4.2. Use Cases 20](#_Toc481680355)

[4.2.1. Events Management 20](#_Toc481680356)

[4.2.2. Time Management 21](#_Toc481680357)

[4.2.3. Members Management 23](#_Toc481680358)

[4.2.4. Reports Management 24](#_Toc481680359)

[4.2.5. Reports Management 25](#_Toc481680360)

[5. Non-functional Requirements 26](#_Toc481680361)

[5.1. Performance Requirements 26](#_Toc481680362)

[5.1.1. Application Response Time 26](#_Toc481680363)

[5.1.2. Dynamic Page Update 27](#_Toc481680364)

[5.1.3. File Compression 27](#_Toc481680365)

[5.1.4. Third-Party Code Libraries and Frameworks 27](#_Toc481680366)

[5.1.5. Serve Scaled Images 27](#_Toc481680367)

[5.2. Safety Requirements 27](#_Toc481680368)

[5.2.1. Source Code Documentation/Source Code Availability 27](#_Toc481680369)

[5.2.2. Android Version Support 28](#_Toc481680370)

[5.3. Security Requirements 28](#_Toc481680371)

[5.3.1. Website Cache Process 28](#_Toc481680372)

[5.3.2. Password Encryption 28](#_Toc481680373)

[5.3.3. Malicious Input Protection 28](#_Toc481680374)

[5.4. User Documentation 29](#_Toc481680375)

[6. References 30](#_Toc481680376)

[7. Appendices 31](#_Toc481680377)

1. Introduction

* 1. Purpose of Document

The purpose of this document is to describe the functionality of the Volunteer Organizer with Management Entity for Final Year Project of Master in Software Project Management.

* 1. Intended Audience

The intended audience for this SRS includes all the stakeholders in the Volunteer Organizer with Management Entity project. The document will be used by TCF Rahbar Team as the specification from which to implement the working program code. The document will be used by Mr. Muhammad Shahzad of the Supervisor as a statement of what functionality will be delivered during the project. The document will be used as a deliverable for the academic portion of the Final Year Project of Master program of Software Project Management and graded for its quality.

Team will be working with the society for TCF Rahbar Team, Cup of Kindness Team on the various campaigns organized by them. The campaigns are designed to "facilitate current issues around city”. The campaign consists of three sections containing lists of local organizations, area volunteers and other information that would be helpful to those in running campaigns profession. Team will be developing and modifying the way of registration online campaigns in accordance with the requirements outlined in this document.

[Note that this document assumes general knowledge about the purpose of the Volunteer Organizer with Management Entity project. The following text describes the overall purpose and long term goals of the project.]

**1.3 Abbreviations**

|  |  |
| --- | --- |
| Abbreviations | Description |
| AC1 | Based on the information passed through the Client  Transport Subsystem, Android Controller Subsystem formats the data so that it can be presented to the user via the Android GUI Subsystem. |
| AC2 | The Android Controller Subsystem converts or formats the  data so it can be easily transferred to the Service Layer via the Client Transport Subsystem |
| AC3 | According to the user interactions and the response from the  GCM API, Android Controller Subsystem passes a Database  Query to its internal Android Database Subsystem. |
| AD1 | RAW Data coming from the Internal Database in the  Android App that are not ready to pass it to the Android  GUI Subsystem |
| AF1 | Admin/Facilitator interacts with the Web App via mouse  clicks and selections |
| AG1 | Based on the Volunteer input or the server response,  Android App changes its view or display a visual indicator to reflect the respective changes |
| AG2 | Based on the user interaction, Android GUI passes RAW  information such as tapped button ids, selection indices etc. to the Android Controller |
| CT1 | Client Transport Subsystem transfers the data coming from  the Service Layer to the Android Controller without processing |
| CT2 | Client Transport Subsystem transfers the data coming from  the Service Layer to the Web Controller without processing |
| CT3 | Android Controller and Web Controller access the Remote  Database via the API; this request is made via HTTP GET or HTTP POST |
| GCM1 | Google Cloud Messaging servers notifies of the updated content in the Remote Database Subsystem via a JASON response to the registered Android devices. The response includes unique device IDs and information regarding the updated content, date and time. |
| RD1 | Remote Database provides RAW Data to the Server  Transport Subsystem based on the received database queries |
| ST1 | Server Transport sends the unformatted/unprocessed data  into the API for data processing |
| ST2 | As per Android Controller or the Web Controller's requests that are coming from the Application Layer, Server Transport Subsystem makes a Database Query to the Remote Database Subsystem |
| ST3 | The response coming from the API is passed to the Application Layer via the Server Transport via HTTP |
| ST4 | Server Transport sends the unformatted data such as username and hashed passwords to the subsystem |
| ST5 | Whenever a change has been made in the Remote Database,  Server Transport Subsystem notify of that change via the Google Cloud Messaging Service; this request is made via HTTP |
| VA1 | API Subsystem exports the data received from the  Remote Database Subsystem allowing services and apps external to the server such as the Android App, Web App or GCM can access the system data |
| VL1 | Volunteer interacts with the Android App via taps,  selections and swipes |
| VL2 | Volunteer interacts with the Web App via mouse clicks and  Selections |
| VO1 | Subsystem checks the user information provided by the Server Transport Subsystem for its validity and passes the response which can be either "valid" or "invalid" to the API |

* 1. Document Convention

The abbreviation TBD has been used to represent the phrase To Be Determined.

1. Overall System Description
   1. Project Background

People like to bring a positive change to the society, and they do that by volunteering for various projects and causes. For some, the act offers a chance to give something back to the community or make a difference to the lives of people around them – for others, it provides an opportunity to develop new skills or gain experience or knowledge that they didn’t have before.

It is important for us to encourage, support and engage people into voluntary activities. The support shouldn’t just be there because it’s for good causes but it is another step towards a better society. People who have a purpose outside of their everyday working role are more motivated, healthier and happier. So, volunteering for good causes is a great act whichever way you look at it.

* 1. Project Scope

The project is based on mobile as well as web application, which will develop to work on the Android platform and Server-side language.

The mobile interface will start from the register page where volunteer must provide basic details or can allow fetch information from 3rd party application like Facebook. After registration, system track volunteer’s location and based on location, system redirects to screen where list of registered campaign will be shown with characteristics like durations, location and short description and title of campaign. Complete details with register button should display when volunteer touches on the specific campaign. Volunteer can register into campaign by providing express interest on the specific campaign. System must generate notification to organizer. System will notify registered volunteer when new volunteer will get enroll in same campaign. Volunteer can send joining invitation request to peer volunteer with 120 characters note. Volunteer can accept or reject invitation request. System should provide details of peer volunteer to another volunteer when invitation request is accepted. Volunteer can provide feedback about campaign and volunteer for future actions.

The web application must utilize for creating campaigns, approval volunteer into campaign and monitor volunteer’s activities like invitation request to another volunteer. Organizer can change details of campaigns and invites peoples for campaigns.

* + 1. Features

Following are important features of the VowMe.

* **Provide location based Campaigns.**

System will provide campaigns based on volunteer’s location within range of 100 miles.

* **Notification of new registered volunteers.**

System will notify registered volunteers about new comers in campaigns.

* **Link peer volunteer within campaigns.**

Volunteer will able to send notification to another volunteer after registration in campaigns.

* **Follow up on campaigns.**

Organizer will monitor and send notification to volunteers about the activities and ask feedback on activities

* **Feedback about volunteers and campaigns.**

Volunteers will allow to provide feedback on activity of campaigns as well as about co-volunteers.

* **Send invitations to peoples.**

Organizer will able to send invitations to people via emails or Facebook.

* **Track Volunteer Hour**

Volunteers will able to track/manage volunteer hours to show their company.

* 1. Not in Scope

The stakeholders request some improvements which are completely out of the scope of what this team can do in the time frame allotted. These features are captured in this document as a reference to future development teams, as some of them are large enough to be a project themselves.

* Power failure backup for the dedicated server
* Verification of Volunteer Information.
* List current campaigns which are being used or being executed within city or area
  1. Project Objectives

This project will provide a way for volunteers to find registered volunteer campaign within their area, communicate more easily with other volunteers who are registered within same volunteer campaign. It will record/track volunteer hours of volunteers and will allow volunteers to provide feedback about campaigns and peers. Also, the application will provide a way for organizers to create the volunteer campaign, maintains volunteer’s basic information and send notifications related to campaign.

* 1. Stakeholders

The Project Supervisor, Mr. Muhammad Shahzad will be a key stakeholder in this project. Mr. Muhammad Shahzad may introduce new changes to the deadlines, structure or project deliverables to his discretion. If these changes have an impact on the project, the project owner will consult with him.

* 1. Operating Environment

[Describe the environment in which the software will operate, including the hardware platform, operating system, network environment and other software components or applications with which it must coexist.]

* 1. System Constraints

The two major constraints has been identified. The first constraint is the time conflict. The office timing and meeting with supervisor is between morning and evening that have resulted in meeting time conflicts. Other than that, personal laptop is not allowed in office, so it is really difficult to archive task on time that work for me.

The second constraint we have identified is lack of AngularJS development experience. Although the lack of knowledge in Android is also a major concern able.

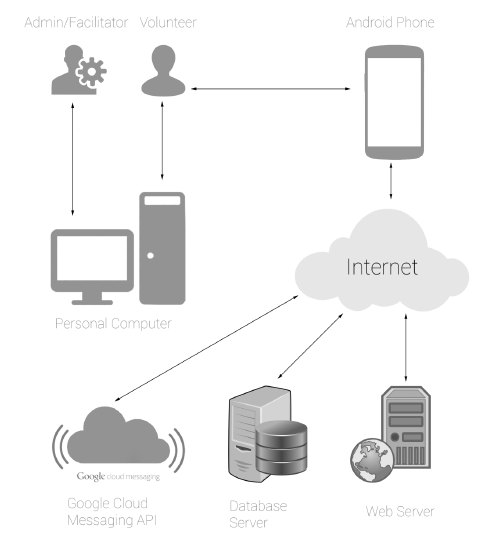
* 1. Assumptions & Dependencies

The project made two assumptions. The first assumption is that all meetings will proceed as planned and we will not postpone the meetings. The second assumption is the System Requirements Specification must be completed and pass the Requirements Review, no changes will be made to the requirements.

Project will be developing in weekends as well after office timings. The office and masters are in parallel, the time will be given lesser than normal project.

1. External Interface Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **Description** | **Use** |
| **Input** | User Interface | Contains the different navigation links and customization forms that the user can select from | User Interface allows user to navigate through the various links on the main page. Admin/Facilitator can also input the upcoming volunteer opportunities through this Interface. |
| **Input** | Database | Database will contain the list of  volunteers to validate login, it will also contain other data used by the application | The application will retrieve and display data received from the database. |
| **Input** | Internet | The Internet will be used to access the website. It will also be used to connect to the database server. | The application will be using the internet, as it is web-based. |
| **Output** | Device Screen | Application will display the information requested to the user using this screen. | The user sees different navigation links, forms, and customization options on device screen. |



* 1. Hardware Interfaces



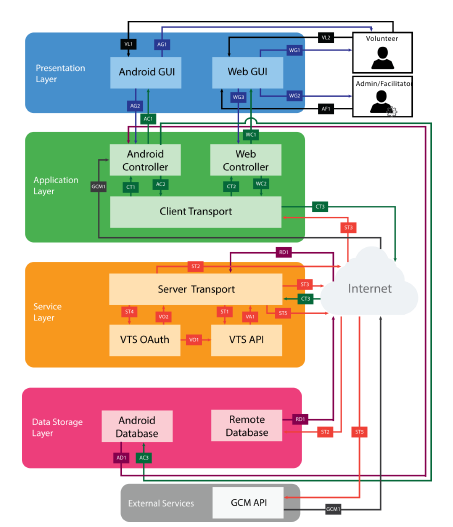
* 1. Software Interfaces

The Android graphical interface of Presentation Layer will depend on the Android operating system, minimum API level of 16, which corresponds to version 4.1.2 (Jelly Bean) for a mobile application. The layer will have additional Android-specific libraries that help transfer data between system components. The graphical interface of the presentation layer will use the Web components as the building blocks of the Web application. The creation of the web page will depend on various Web technologies such as HTML5, CSS3, JavaScript and Angular. The Web GUI will be compatible with all major versions of Web browsers such as Mozilla Firefox, Google Chrome, Internet Explorer and Safari published over the past two years.

* 1. Communications Interfaces

The Application Layer consists of Android Controller, Web Controller and Client Transport. The Android API will be used to synchronize the remote database and the Android database in Android Controller. Android Controller also uses the SQLite database and other APIs to establish a direct connection to the Android database and process the database query when the system is not connected to the Internet. Web Controller will make AJAX requests by HTTP GET and HTTP POST methods to transfer data to the client transport. Android and Web applications will use JSON analysis libraries to analyze data before passing it on to customer transport. Client Transport will make an HTTP request to transfer data, including attachments to the service layer.

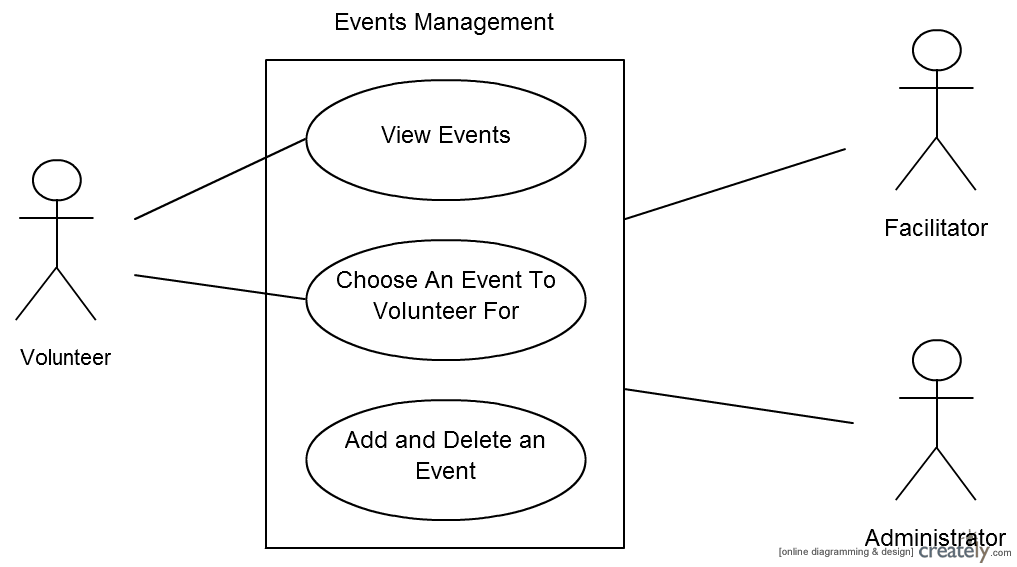
Another layer which contains Data Storage Layer is mainly responsible for storing data. The Layer has Remote Database and Android Database. The Server Transport subsystem will update and retrieve data only from Remote Database whereas Android Database is used by Android Database when the system is offline. Android Database will be synced with Remote Database via Google Cloud Messaging (GCM) when there is internet connection. Any update on Remote Database will be notified to Android Database for syncing process. The Android Database will be accessed directly from Android Controller in Application Layer by relational SQLite queries.



1. Functional Requirements
   1. Functional Hierarchy

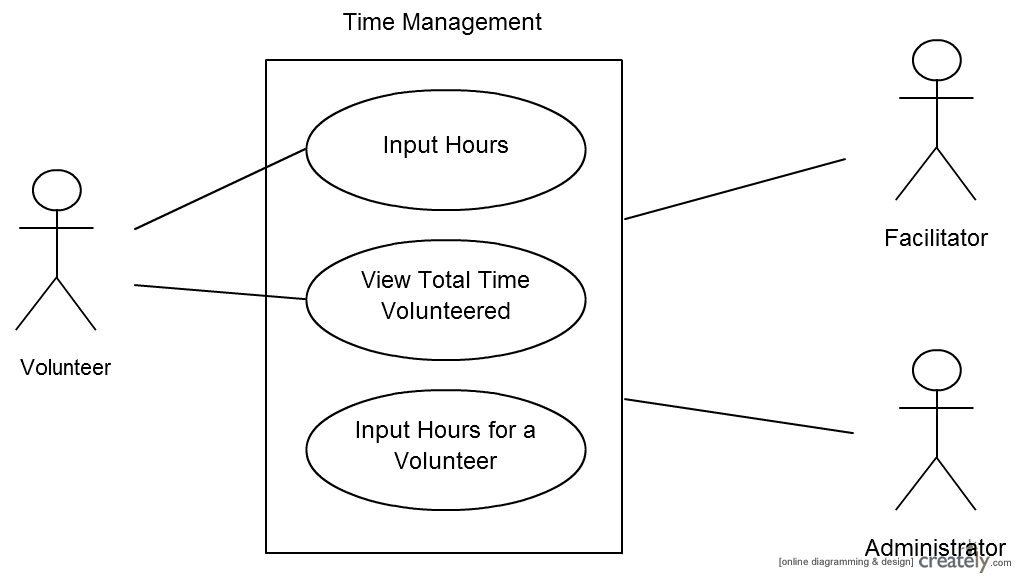
This section will cover requirements that are important to our customer. The application’s main function is to provide the ability to register volunteer activities and organizer activities for the achieving volunteer tasks. In several requirements below, there are references to different types of users such as Administrator, Organizer, and Volunteers. Administrator who manage the Volunteers/Organizer. Opportunities are divided by categories and each category will have a designated Organizer assigned by the Administrator.

* + 1. Login
       - **Description:** The System shall allow users to login with their Username and password. When a user logs in to the system for the first time, the system shall allow them to enter their Username for validation. When the Username is validated, the system shall ask the user to establish their password. When a user logs in to the system again, they will be required to enter their Username and Password for validation.
       - **Constraints:** User must be connected to the Internet to access the System and have a valid email account.
       - **Standards:** None
       - **Priority:** 1-Critical
    2. Logout
       - **Description**: The System shall allow volunteers to logout of the system. When the user is logged out, the system shall redirect to the login page.
       - **Constraints:** User must be connected to the Internet to access the System.
       - **Standards:** None
       - **Priority:** 1-Critical
    3. Add Volunteer Events
       - **Description:** The System shall allow organizers to input the new or upcoming volunteer events. An opportunity may include a title, description, date and time, location and images.
       - **Constraints:** User must be connected to the Internet and logged into the System.
       - **Standards:** None
       - **Priority:** 2-High
    4. Delete Volunteer Opportunities
       - **Description:** The System shall allow organizers to delete volunteer opportunities previously entered the System. If volunteers have committed to an opportunity and it is cancelled, the system will notify all volunteers through Email.
       - **Constraints:** User must be connected to the Internet, logged into the Volunteer Tracking System, and have a valid Email address to receive the Email notification.
       - **Standards:** None
       - **Priority:** 2-High
    5. Sign Up for Volunteer Opportunities
       - **Description:** The volunteers shall be able to see the details of an opportunity such as the date, time, and location and have an option to sign up for an opportunity to indicate they will be volunteering at that opportunity.
       - **Constraints:** User must be connected to the Internet and logged into the System.
       - **Standards:** None
       - **Priority:** 2-High
    6. Cancel Commitment
       - **Description:** The volunteers shall be able to cancel a commitment they previously made. If volunteers previously signed up for an opportunity, the system shall allow them to cancel their commitment to indicate they will no longer be volunteering at that opportunity.
       - **Constraints:** User must be connected to the Internet and logged into the System**.**
       - **Standards:** None
       - **Priority:** 2-High
    7. Notify Volunteer
       - **Description:** The System shall notify the volunteer and the opportunity facilitator upon the volunteer’s acceptance/commitment or cancellation of an opportunity. This notification will be system generated. The volunteer and the facilitator will be able to see this notification.
       - **Constraints:** User must be connected to the Internet and logged into the System to access the notifications.
       - **Standards:** None
       - **Priority:** 2-High
    8. Track Progress
       - **Description:** The System shall allow users to track progress of their volunteer activities and the status of different service levels. Service levels are different levels that volunteers can achieve based on the total number of hours.
       - **Constraints:** User must be connected to the Internet and logged into the System.
       - **Standards:** None
       - **Priority:** 1-Critical
    9. Customize Preferences
       - **Description:** The System shall allow volunteers to customize their preferences. Preferences include changes of date of availability, location and interest in different opportunity categories.
       - **Constraints:** User must be connected to the Internet and logged into the System.
       - **Standards:** None
       - **Priority:** 2-High
    10. Validate Members
        - **Description:** The System shall validate the members every year. Every year, the system shall make all members inactive who is not active in a year.
        - **Constraints:** User must be connected to the Internet to access the System and the administrator must have access to the list of members who is inactive in a year.
        - **Standards:** None
        - **Priority:** 4-Low
    11. Register Volunteers
        - **Description:** The System shall allow Admin to register volunteers and allow access into the system.
        - **Constraints:** User must be connected to the Internet to access the web-based system and have a valid email account.
        - **Standards:** None
        - **Priority:** 1-Critical
    12. Volunteer Feedback
        - **Description:** The Volunteer Tracking System shall provide a social aspect to the interface where the volunteers can input and share their stories.
        - **Constraints:** Text must be displayed in a readable format.
        - **Standards:** None
        - **Priority:** 2-High
    13. Android Application
        - **Description:** The shall be available in the form of an Android Application.
        - **Constraints:** The Android device shall support Android version 4.1.2 or higher.
        - **Standards:** None
        - **Priority:** 3-Moderate
    14. Ease of Use
        - **Description:** The System shall provide a user-friendly interface. The system shall also limit the number of clicks to allow a user to reach their desired page easily.
        - **Constraints:** The screen size along with any information must be displayed in a readable format.
        - **Standards:** None
        - **Priority: 3-Moderate**
    15. iOS Mobile Application
        - **Description:** The System shall be available in the form of an iOS Application.
        - **Constraints:** The iOS device shall support iOS version 7.1 or higher.
        - **Standards:** None
        - **Priority:** 5-Future
  1. Use Cases
     1. Events Management



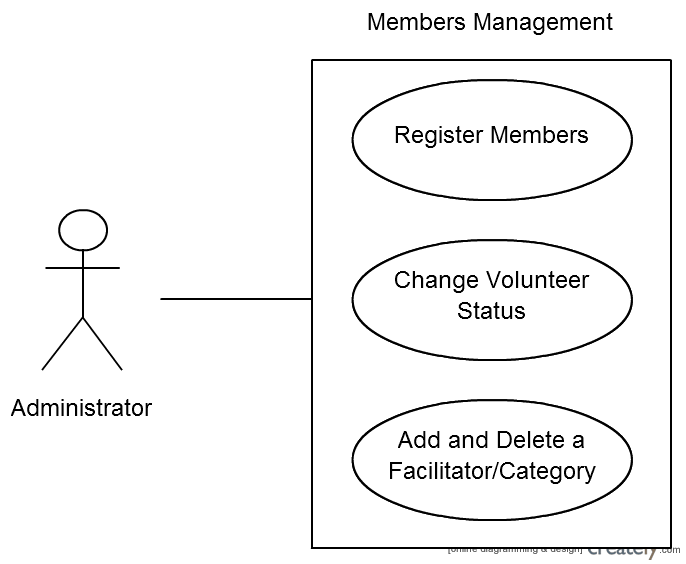
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Event Management** | | | | | |
| **Use case Id:** | | | 01 | | |
| **Actors:**  Volunteer, Facilitator, Administrator | | | | | |
| **Feature:** Creations of campaigns and approvals of campaigns | | | | | |
| **Pre-condition:** | | Volunteer and Facilitator must be login | | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | | **Software Reaction** |
| **1.** | Facilitator created campaign | | | | System will create campaign and ready for volunteer to see. |
| **2.** | Volunteer clicks on campaign | | | | System will show the details of selected campaign |
|  |  | | | |  |
| **Alternate Scenarios:** Volunteer clicks on campaign and Facilitator change the details of campaign | | | | | |
| **1a:**    **2a:** | | | | | |
| **Post Conditions** | | | | | |
| **Step#** | **Description** | | | | |
|  | Sequentially list conditions expected at the completion of the use case. | | | | |
|  |  | | | | |
|  |  | | | | |
| **Use Case Cross referenced** | | | | <Related use cases, which use or are used by this use case> | |

* + 1. Time Management



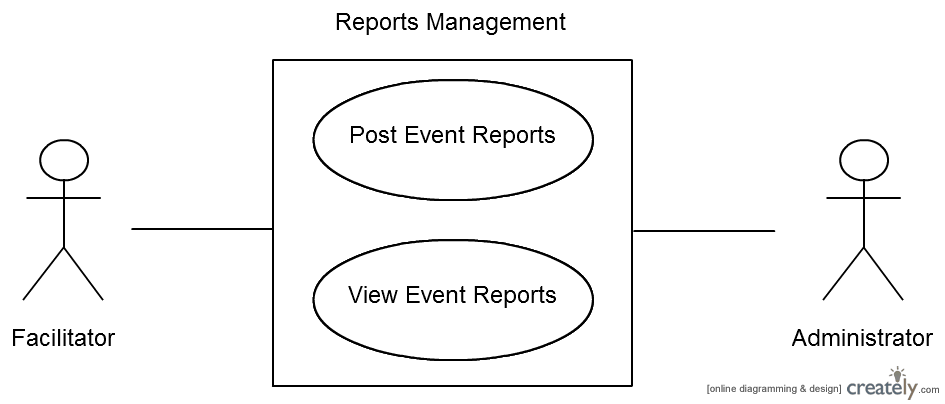
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<Use case Id: name>** | | | | | |
| **Use case Id:** | | | Write use case reference number. | | |
| **Actors:**  <List of actors (external agents), indicating who initiated the use case> | | | | | |
| **Feature:** <Feature from which the use case is driven> | | | | | |
| **Pre-condition:** | | <List the assumptions required before this Use Case can be executed. > | | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | | **Software Reaction** |
| **1.** | Numbered actions of the actors | | | | Numbered description of system responses |
| **2.** |  | | | |  |
|  |  | | | |  |
| **Alternate Scenarios:** Write additional, optional, branching or iterative steps. Refer to specific action number to ensure understandability. | | | | | |
| **1a:**    **2a:** | | | | | |
| **Post Conditions** | | | | | |
| **Step#** | **Description** | | | | |
|  | Sequentially list conditions expected at the completion of the use case. | | | | |
|  |  | | | | |
|  |  | | | | |
| **Use Case Cross referenced** | | | | <Related use cases, which use or are used by this use case> | |
|  | | | |  | |

* + 1. Members Management



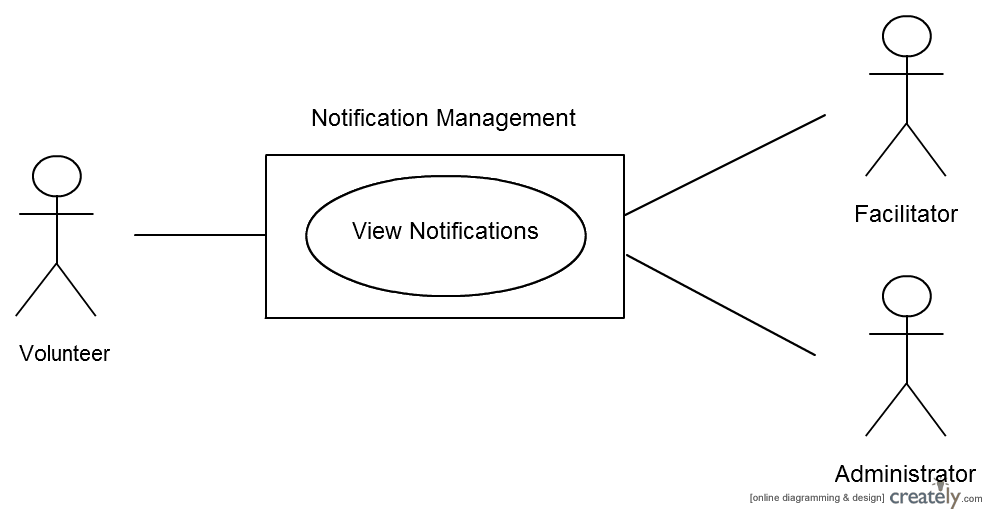
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<Use case Id: name>** | | | | | |
| **Use case Id:** | | | Write use case reference number. | | |
| **Actors:**  <List of actors (external agents), indicating who initiated the use case> | | | | | |
| **Feature:** <Feature from which the use case is driven> | | | | | |
| **Pre-condition:** | | <List the assumptions required before this Use Case can be executed. > | | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | | **Software Reaction** |
| **1.** | Numbered actions of the actors | | | | Numbered description of system responses |
| **2.** |  | | | |  |
|  |  | | | |  |
| **Alternate Scenarios:** Write additional, optional, branching or iterative steps. Refer to specific action number to ensure understandability. | | | | | |
| **1a:**    **2a:** | | | | | |
| **Post Conditions** | | | | | |
| **Step#** | **Description** | | | | |
|  | Sequentially list conditions expected at the completion of the use case. | | | | |
|  |  | | | | |
|  |  | | | | |
| **Use Case Cross referenced** | | | | <Related use cases, which use or are used by this use case> | |
|  | | | |  | |

* + 1. Reports Management



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<Use case Id: name>** | | | | | |
| **Use case Id:** | | | Write use case reference number. | | |
| **Actors:**  <List of actors (external agents), indicating who initiated the use case> | | | | | |
| **Feature:** <Feature from which the use case is driven> | | | | | |
| **Pre-condition:** | | <List the assumptions required before this Use Case can be executed. > | | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | | **Software Reaction** |
| **1.** | Numbered actions of the actors | | | | Numbered description of system responses |
| **2.** |  | | | |  |
|  |  | | | |  |
| **Alternate Scenarios:** Write additional, optional, branching or iterative steps. Refer to specific action number to ensure understandability. | | | | | |
| **1a:**    **2a:** | | | | | |
| **Post Conditions** | | | | | |
| **Step#** | **Description** | | | | |
|  | Sequentially list conditions expected at the completion of the use case. | | | | |
|  |  | | | | |
|  |  | | | | |
| **Use Case Cross referenced** | | | | <Related use cases, which use or are used by this use case> | |
|  | | | |  | |

* + 1. Reports Management



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<Use case Id: name>** | | | | | |
| **Use case Id:** | | | Write use case reference number. | | |
| **Actors:**  <List of actors (external agents), indicating who initiated the use case> | | | | | |
| **Feature:** <Feature from which the use case is driven> | | | | | |
| **Pre-condition:** | | <List the assumptions required before this Use Case can be executed. > | | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | | **Software Reaction** |
| **1.** | Numbered actions of the actors | | | | Numbered description of system responses |
| **2.** |  | | | |  |
|  |  | | | |  |
| **Alternate Scenarios:** Write additional, optional, branching or iterative steps. Refer to specific action number to ensure understandability. | | | | | |
| **1a:**    **2a:** | | | | | |
| **Post Conditions** | | | | | |
| **Step#** | **Description** | | | | |
|  | Sequentially list conditions expected at the completion of the use case. | | | | |
|  |  | | | | |
|  |  | | | | |
| **Use Case Cross referenced** | | | | <Related use cases, which use or are used by this use case> | |

1. Non-functional Requirements
   1. Performance Requirements

This section discusses the performance requirements for this project. The main concerns regarding the performance of the product are based on primarily the availability and the responsiveness. The product is expected to offer a continuous and smooth experience to the user.

* + 1. Application Response Time
       - **Description**: The response time between the user interaction and the result must be less than 8 seconds on the site and on the Android application
       - **Constraints**: Internet connections and hardware capacity of devices.
       - **Standards**: None
       - **Priority**: 3-Moderate
    2. Dynamic Page Update
       - **Description**: Only the necessary parts of the web page will be updated when the user interacts instead of completely reloading the page.
       - **Constraints**: JavaScript may not be enabled by default on Web browsers
       - **Standards**: AJAX standards.
       - **Priority:** 3-Moderate
    3. File Compression
       - **Description**: The JavaScript and CSS files will be compressed to reduce the size of the file. The size of JPEG images should be less than 3 MB.
       - **Constraints**: None
       - **Standards**: None
       - **Priority**: 4-Low
    4. Third-Party Code Libraries and Frameworks
       - **Description:** JavaScript and CSS libraries will be directly accessible from the Caching Servers, thereby improving access time in remote locations.
       - **Constraints**: Not all third-party libraries offer Cache URLs
       - **Standards**: None
       - **Priority**: 4-Low
    5. Serve Scaled Images
       - **Description:** Images on different scales will be used in different scenarios. That is, thumbnails, full-screen images.
       - **Constraints**: None
       - **Standards**: None
       - **Priority**: 4-Low
  1. Safety Requirements

This section discusses the requirements safety requirements needed to maintain the security of the system and control the privacy of information.

* + 1. Source Code Documentation/Source Code Availability
       - **Description:** The source code must be well documented with comments and details about the feature. The code will help anyone who wants to develop this product in the future.
       - **Constraints**: All the documentations shall be in English.
       - **Standards**: None
       - **Priority**: 3-Moderate
    2. Android Version Support
       - **Description:** The mobile version of the system will be based on Android. The application supports a minimum API level of 16, which corresponds to version 4.1.2 (Jelly Bean).
       - **Constraints**: All the documentations shall be in English.
       - **Standards**: The Android device supports Android version 4.1.2 or higher.
       - **Priority**: 4-Low
  1. Security Requirements

This section discusses the requirements security and privacy requirements needed to maintain the security of the system and control the privacy of information.

* + 1. Website Cache Process
       - **Description:** The age of the web site cache will be limited to 7 days.
       - **Constraints:** None
       - **Standards:** HTTP cache control standards
       - **Priority:** 4-Low
    2. Password Encryption
       - **Description:** All user passwords must be encrypted in the MySQL database.
       - **Constraints:** None
       - **Standards:** MD5 hash function
       - **Priority:** 1-Critical
    3. Malicious Input Protection
       - **Description:** All user passwords must be encrypted in the MySQL database.
       - **Constraints:** None
       - **Standards:** MD5 hash function
       - **Priority:** 1-Critical
  1. User Documentation

[List the user documentation components that will be delivered along with the software, such as user manuals, online help, context-sensitive help and tutorials.]

1. References

[This section should provide a complete list of all documents referenced at specific point in time. Each document should be identified by title, report number (if applicable), date, and publishing organization. Specify the sources from which the references can be obtained. (This section is like the bibliography in a published book).]

1. Appendices