

Software Requirements Specification

iVolunteer

Version 1.0

February 05, 2014

Submitted by:

Rakesh Patni(M120364CA)

Muthukumar(M120355CA)

Rajender Reddy(M120523CA)

Nachiappan V.(B100780CS)

Submitted in partial fulfillment
Of the requirements of
CS 3004 Software Engineering

Table of Contents

Table of Content.....	1
1.0. Introduction	4
1.1.Purpose	4
1.2. Scope of Project	4
1.3. Glossary	5
1.4. References	5
1.5. Overview of Document	6
2. REQUIREMENT SPECIFICATIONS	
2.1 System Environment	7
2.2. Use Cases	8
Seeker Use Cases:	
2.2.1 Seeker Registers	8
2.2.2 Seeker posts requests	9
2.2.3 Seeker checks status of an event's post.....	10
Admin Use cases:	
2.2.4 Admin approves or disapproves of registration of seeker....	11
Volunteers use cases:	
2.2.5 Volunteer registers	12
2.2.6 Volunteer views posts and takes action	13
2.2.7 Volunteer withdraws acceptance to volunteer	14
2.3 User characteristics	15
2.4 Detailed non functional requirements	
2.4.1 System specification and dependencies	15
2.4.2 Security	16
2.4.3 Assumption and dependencies	16

3.0. Requirements Specification	17
3.1 External Interface Requirements	17
3.2 Functional Requirements	17
3.2.1 Seeker registers	17
3.2.2 Seeker posts request	19
3.2.3 Seeker checks status of post	20
3.2.4 Volunteer registers	21
3.2.5 Volunteer takes action to accept/ignore	22
3.2.6 Volunteer withdraws commitment.....	23
3.2.7 Admin approves or disapproves seeker's registration	24
 3.3 Index	 26

1 INTRODUCTION

1.1 Purpose

The purpose of this document is to present a detailed description of the iVolunteer. iVolunteer is a volunteer-ship management system. This document will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system.

1.2 Scope

This website will serve as an online platform for people who seek for volunteers for certain events or activities which requires volunteers. The website along with the back-end software system aims at facilitating communication between the volunteer seekers and the volunteers. The objective is to establish a central communication medium for all the seekers and volunteers who are regularly involved in such events and activities. By providing better communication, it reduces the amount of time wasted in the process if done otherwise and also reduces redundancy of information. The system will also contain a relational database of all its users, i.e., seekers and volunteers.

1.3 Glossary

Term	Definition
Seeker	Seeker is a user who wants volunteers for an event.
Volunteer	Volunteer is user who could possibly volunteer for an event.
Admin	Admin is the administrator of the data base.
Data base	Collection of all information, which is stored data base and monitoring by system.
Software requirements Specification	A document that completely describes all of the functions of a proposed system and the constraints under which it Must operate. For example, this document.
Event	An event posted by the seeker, for which he/she seeking for volunteers
List	The list of the all the volunteers who have accepted for the event.
Registrant	Users who have registered with the website.

1.4 References

- IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.
- <http://techwhirl.com/writing-software-requirements-specifications/>
- <http://www.microtoolsinc.com/Howsrs.php>

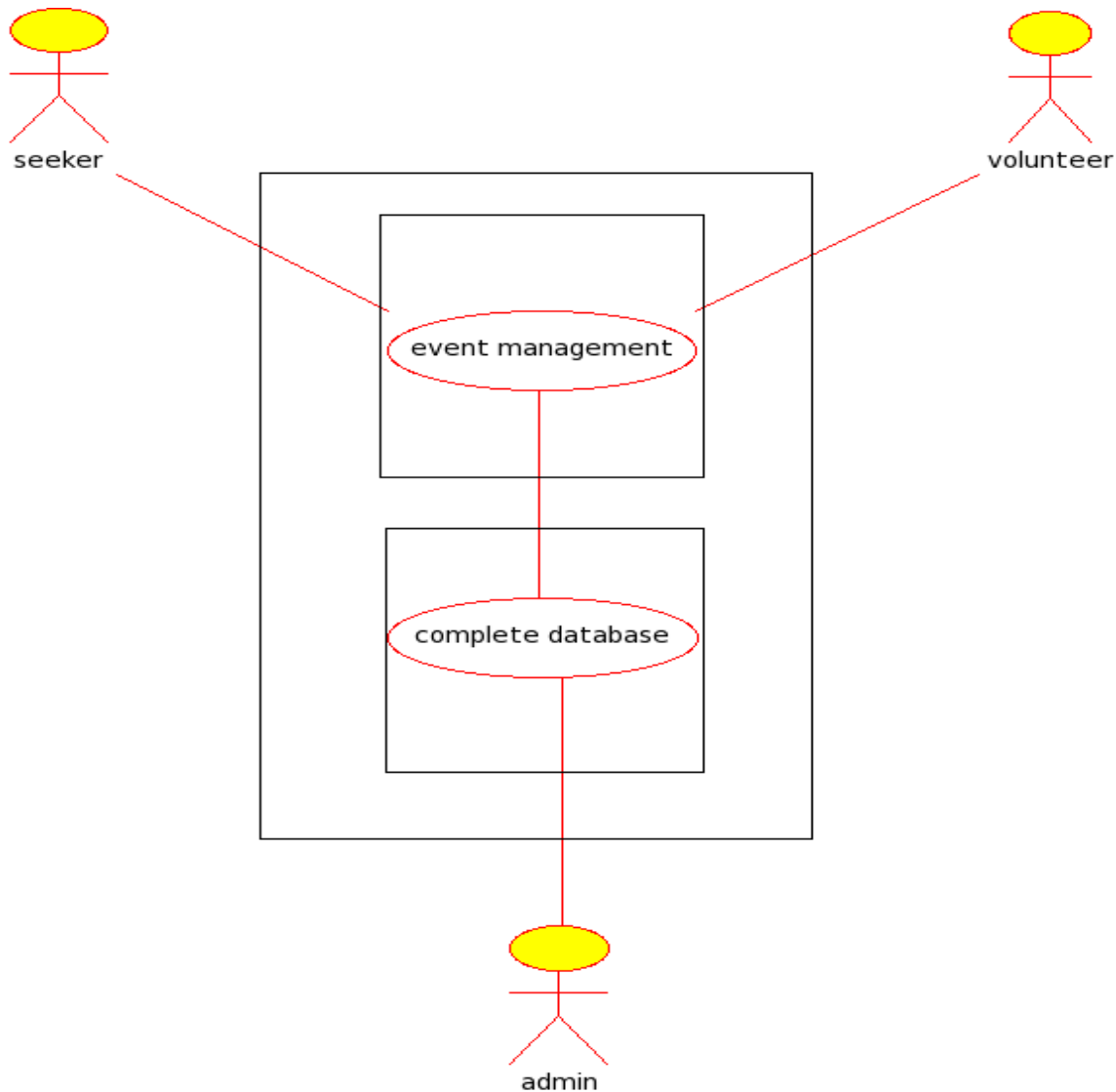
1.5 Overview of Document

The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.

The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product. Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

2 Overall description

2.1 System environment



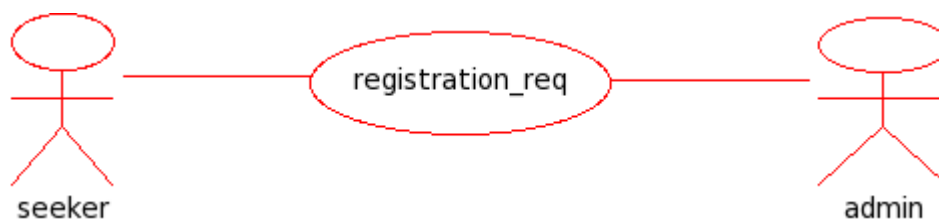
Description: The system will have two parts namely, Event management and the complete database. The event management component involves event description, volunteer details and the seeker's details for a particular event. Only this component of the system is accessible to the users of the system. The database will include all the records of all events, all seekers and volunteers. This component of the system will be accessible only to the admin.

2.2 Use Cases

This section outlines the use cases for each of the active users and the admin separately. The seeker, the volunteer and the admin have multiple use cases each in the system.

2.2.1 Seeker registers

Diagram:



Brief Description: The seeker accesses the iVolunteer website, navigates to the registration page and registers.

Precondition: The seeker has already accessed the website.

Step-By-Step:

1. The system displays the on-line form containing fields such as -name of organization, name of the representative. address. Phone, email, field of service. Strength of organization and its scope.
2. The seeker fills all the mandatory fields and also uploads a scanned copy of a document certified by a Govt. authority authenticating the

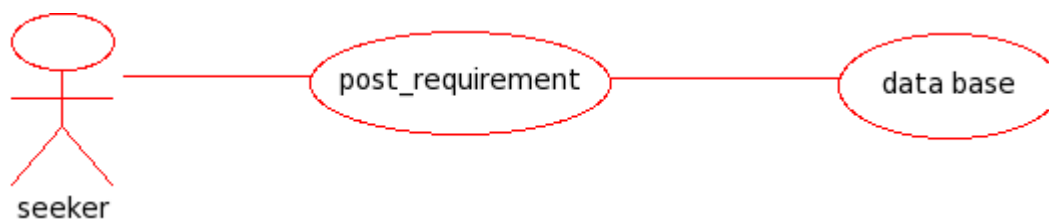
organization.

3. He clicks on the submit button, which the system acknowledges with a message of “*successful completion and approval pending*”.
4. The seeker exits from the website.

2.2.2 Seeker posts request

Actor involved: Seeker

Diagram:



Brief Description: The seeker submits his requirements of volunteers.

Precondition: The seeker has logged in to his profile page using his ID and password.

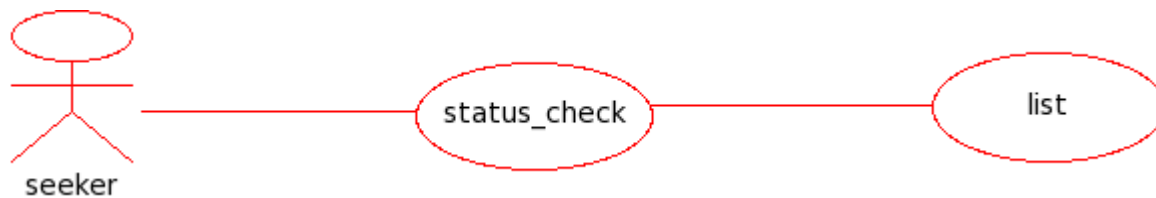
Step-By-Step:

1. The seeker chooses the *post* button (to make a post).

2. System generates a form consisting of fields such as - event description, date, time, duration, location of event. Also he is to specify the deadline before which he expects the list of volunteers to be sent to him.
3. Clicks the *submit* button, to which the system *acknowledges*.
4. Seeker logs out of the system.

2.2.3 Seeker checks status of an event's post

Diagram:



Brief description: The seeker checks for the list of volunteers who have volunteered for the event.

Precondition: The seeker must already be logged in and should have already posted about the event.

Step-By-Step:

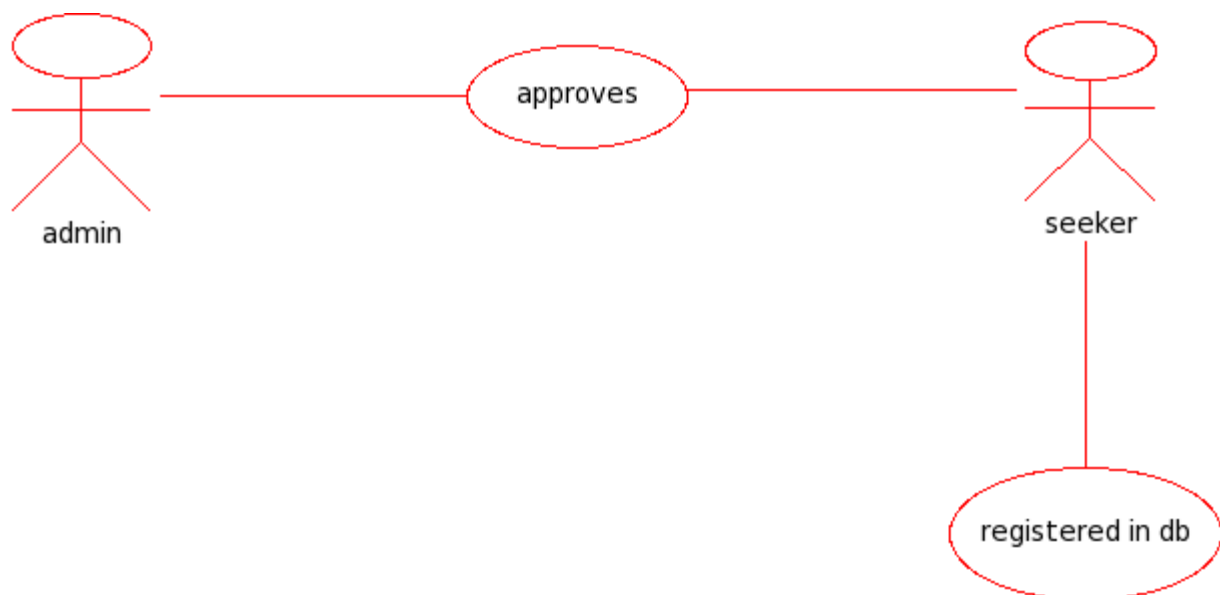
1. The seeker should be able to get the list of volunteers who have volunteered in one click of a button.
2. The document must contain the profile details of all the volunteers.
3. If the number of volunteers who have accepted to volunteer for the event

are lesser than that required by the seeker as specified in his post, then the document must indicate that.

4. Has an option to log out of his profile.

2.2.4 Admin approves or disapproves of registration of seekers

Diagram:



Brief description: The admin should be able to view the profiles of all registrants and choose to accept or reject his/her approval.

Precondition: Admin must be logged into his profile

Step-by-step:

1. Admin should be able to view all pending requests if any by clicking on the *new requests* link

2. By clicking on a specific request, admin should be able to view the details of the registrants.
3. Admin must have access to the scanned copy of legal identification submitted by the seeker along with the details of the registrant.
4. Should be able to approve or reject a registration request by clicking on respective buttons which appear along with the details of a request, following which an email must be sent to registrant reporting the admin's response. If accepted by admin, the email must contain the user-name and password allotted (system generated).
5. Has an option to log out of profile.

2.2.5 Volunteer registers

Diagram:



Brief Description: Volunteer registers himself/herself in the website to provide his/her services.

Precondition: The volunteer accesses to the iVolunteer website.

Step-By-Step:

1. The System generates a form containing fields such as -name,age,sex ,address,identity proof (number).occupation etc.
2. After the details of the form have been filled, the system should generate an user-name for the registrant, and request for a new password.
3. On successful completion of the above system generates a *“thank you for registering”* message
4. The volunteer exits form the system.

2.2.6 Volunteer views all posts relevant to his interests and take an accept/ignore action.

Diagram:



Brief description: The volunteer should be able to view all posts relevant to his interests selected during registration

Precondition: Volunteer must be logged in.

Step-by-step

1. Should be able to click on an available post to view the details of the post

2. Should be able to accept or ignore an event by clicking on “Accept” or “Ignore”.
3. Once he/she accepts to volunteer, he should be able to see the position in the list.
4. Should be able to commit to any number of events that don't clash in timings.
5. An option of logging out must be available

2.2.7 Volunteer withdraws his acceptance to volunteer

Diagram:



Brief description: The volunteer must be able to withdraw his acceptance to volunteer if the list has not already been submitted to the seeker.

Precondition: A volunteer must be logged in and must have accepted to volunteer to an event which appears on his home page.

Step-by-step:

1. Should be able to withdraw in a click of a button.

2. Log out of his/her profile

2.3 User Characteristics:

1. The seeker is expected to be Internet literate and be able to post his requirements on his profile page of the iVolunteer Website.
2. The Admin is expected to be Internet literate and be able to maintain a database.
3. The Volunteer is expected to be Internet literate and be able to accept or deny the seeker's requirements.

2.4 Detailed non-functional requirements:

2.4.1 System specifications and dependencies

1. The back-end of the website should run on a server with less than 1gb memory.
2. The front-end of the website must be supported by IE9 (or later), Mozilla Firefox and Google chrome.
3. The website must be built to be hosted on a Linux server.
4. The event description, profile of a seeker, or profile of a volunteer must not take up more than 2mb each of the server's secondary storage space.
5. The website must be able to handle up to at least 5 simultaneous connections assuming it has the supporting server hardware.
6. The website need not comply or interface with any other software application.
7. Most of the information to be handled will be plain text
8. A user should not have to wait for more than thirty seconds for the

website to process any request.

2.4.2 Security

1. The server must be secured and accessed only through a password.
2. There must be an option for all users and admin to change their login password.
3. The entire system's data base must be visible to the admin. This includes the information of all seekers and volunteers which have provided by them during the time of registration. Also, it must contain the details of all events along with the event's description and list of volunteers.
4. Any edit to the database by the admin must be done only after a password prompt.
5. There need not be any encryption methodology used to store the data in the data base.
6. The communication from client to server need not be encrypted.
7. The server must keep a log of user's login activity in the server.

2.4.3 Assumptions and dependencies:

1. The users have a pdf reader installed in his/her client system. The list generated must be imported to pdf and provided to the seeker.
2. Every committed volunteer shows up in the event.

3.0 Requirements specification

3.1 Verification requirements

Since the posts made by a seeker are sent directly to the volunteer without the admin intervening, the volunteers maybe misused for an event. In order to avoid this, Every seeker at the time of registration must submit a scan copy of a legal identification document as an image to an upload link provided on the website. This copy of identification ensures that the seeker's account is not a fraudulent one.

The image uploaded must be available to the admin at the time of choosing to approve a seeker's request. Also, this image will be maintained in the database along with the seeker's profile for future reference.

3.2 Functional Requirements

3.2.1 Seeker registers

Use Case name	Seeker registers
Xref	—
Trigger	The seeker accesses the iVolunteer site
Precondition	System displays online form for registration
Basic Path	1. The system displays the on-line form containing fields such as -name of organization,name of the

	<p>representative. address. Phone, email, field of service. Strength of organization and its scope.</p> <p>2. The seeker fills all the mandatory fields and also uploads a scanned copy of a document certified by a Govt. authority authenticating the organization.</p> <p>3. He clicks on the submit button, which the system acknowledges with a message of <i>successful completion and approval pending.</i></p> <p>4. The seeker exits from the website.</p>
Alternative Path	None
Post condition	Request form successfully submitted
Exception Path	The seeker abandons the process at any time
Other	—

3.2.2 Seeker posts request

Use Case name	Seeker posts request
Xref	–
Trigger	The seeker clicks the post button on his profile page
Precondition	The seeker has logged into his profile
Basic Path	<ol style="list-style-type: none">1. The seeker chooses the <i>post</i> button (to make a post).2. System generates a form consisting of fields such as - event description, date, time, duration, location of event. Also he is to specify the deadline before which he expects the list of volunteers to be sent to him.3. Clicks the <i>submit</i> button , to which the system acknowledges .4. Seeker logs out of the system.
Alternative Path	None
Post condition	The request is saved in the database
Exception Path	The seeker can abandon the post any time

Others	The deadline mentioning is mandatory to post
--------	--

3.2.3 Seeker checks status of post

Use Case name	Checks status of post
Xref	—
Trigger	Seeker clicks on the check status button
Precondition	The seeker is logged in to his profile and has posted an event.
Basic Path	<ol style="list-style-type: none"> 1. The seeker should be able to get the list of volunteers who have volunteered in one click of a button. 2. The document must contain the profile details of all the volunteers. 3. If the number of volunteers who have accepted to volunteer for the event are lesser than that required by the seeker as specified in his post, then the document must indicate that. 4. Has an option to log out of his profile.

Alternative Path	None
Post condition	The seeker becomes aware of the status of his request.
Exception Path	The seeker can abandon the process any time
Others	The seeker can also update details about post, excepting the deadline.

3.2.4 Volunteer registers

Use Case name	Volunteer registers
Xref	None
Trigger	He clicks on the register button
Pre condition	The volunteer has accessed into the iVolunteer website
Basic Path	<ol style="list-style-type: none"> 1. The System generates a form containing fields such as -name,age,sex ,address,identity proof (number).occupation etc. 2. <i>After the details of the form have been filled, the system should generate an username for the registrant, and request for a new password.</i> 3. On successful completion of the above system generates a

	<p><i>“thank you for registering”</i> message</p> <p>4. The volunteer exits form the system.</p>
Alternative Path	None
Post condition	The volunteer successfully registers himself
Exception Path	The seeker can abandon the process any time
Other	—

3.2.5 Volunteer takes an action to accept or ignore

Use Case name	Volunteer sees the available posts and takes accept/ignore action
Xref	None
Trigger	Volunteer clicks on the upcoming events button.
Precondition	Volunteer has logged in to his account
Basic Path	<ol style="list-style-type: none"> 1. Should be able to click on an available post to view the details of the post 2. Should be able to accept or ignore an event by clicking on “Accept” or “Ignore”.

	<ol style="list-style-type: none"> 3. Once he/she accepts to volunteer, he should be able to see the position in the list. 4. Should be able to commit to any number of events that don't clash in timings 5. An option of logging out must be available
Alternative Path	None
Post Condition	The system records the reply of the volunteer
Exception Path	The volunteer can abandon the process any time
Others	If the event has already received enough number of volunteers as requested by the seeker, then the volunteer must get a number denoting his position in the list i.e., waiting list number.

3.2.6. Volunteer withdraws his commitment

Use Case name	Volunteer withdraws his acceptance
Xref	None
Trigger	The volunteer clicks on the 'participation status' button

Precondition	The volunteer logs in to his account and clicks on the relevant post
Basic Path	<ol style="list-style-type: none"> 1. Should be able to withdraw in a click of a button if the list has not already been submitted to the seeker. 2. Log out of his/her profile
Alternative Path	None
Post condition	The system removes replaces his name with another volunteer's
Exception Path	The volunteer can abandon the process any time
Others	Once a volunteer withdraws, his entry is removed from the list if not already submitted to the seeker.

3.2.7 Admin approves or disapproves the seekers registration

Use case name	Admin approves or disapproves the seeker registration
Xref	None
Trigger	Admin clicks the 'new registrants' button

Precondition	He has logged into his account
Basic Path	<ol style="list-style-type: none"> 1. Admin should be able to view all pending requests if any by clicking on the “<i>new requests</i>” link 2. By clicking on a specific request, admin should be able to view the details of the registrants. 3. Admin must have access to the scanned copy of legal identification submitted by the seeker along with the details of the registrant. 4. Should be able to approve an registration request by clicking on respective buttons which appear along with the details of a request, following which an email must be sent to registrant reporting the admins response. The email must contain the user-name and password allotted(system generated). 5. Has an option to log out of profile.

Alternative Path	The admin may choose to reject the registration request if specifications are not satisfied
Exception Path	The admin choose to abandon the process any time
Others	None

3.3 Index

Admin:4,7,9,10,12,13,17,20

Approval:8,10,20

Database:10,11,14,16

Event:4,7,8,12,14

Interface:3

Post:12,14,16,17,18

Seeker:4,7,8,9,10,12,13,14,15,16,17,18

Stakeholders:3

Status:9,17,19

Volunteer:3,4,7,8,9,10,11,12,13,14,15,17,18,19,20