Rendezvous

Use Case 3: Create Event

Abstract

|  |
| --- |
| ***This document provides an overview of the solution context, scope and high-level functional capabilities for the <Rendezvous> project. It catalogs the actors that interact with the system and the use cases that together describe all the ways in which they interact with the system. It also documents the use-case diagrams that show the relationships that exist between the actors and the use cases.*** |

Version History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Author** | **Description** |
| 07/16/2015 | V01 | Avinash Rao |  |
|  |  | Nicholas Petty |  |
|  |  | Ciao Farias |  |
|  |  | Scott Hinden |  |
| 07/17/2015 | V02 | Nicholas Petty | Minor typos fixed. |

*Table of Contents*

[1 Introduction 3](#_Toc335776012)

[1.1 Document Purpose 3](#_Toc335776013)

[1.2 Document Scope 3](#_Toc335776014)

[1.3 Document Overview 3](#_Toc335776015)

[1.4 User Story 3](#_Toc335776015)

[2 Use-Case Diagrams 4](#_Toc335776016)

[2.1 <Diagram Name> 4](#_Toc335776017)

[3 Actor Catalog 4](#_Toc335776018)

[4 Use-Case Catalog 5](#_Toc335776019)

[5 Use case Specification x 6](#_Toc335776020)

[5.1 Document Overview 6](#_Toc335776021)

[5.2 Brief Description 7](#_Toc335776025)

[5.3 Preconditions 7](#_Toc335776026)

[5.4 Basic Flow 7](#_Toc335776027)

[5.5 Alternative Flows 7](#_Toc335776028)

[5.6 Post-Conditions 8](#_Toc335776031)

[5.7 Extensibility 8](#_Toc335776033)

[5.8 Additional Inforamation 8](#_Toc335776034)

[5.9 Scenarios 8](#_Toc335776034)

6 Example of a Use Case Specification ……………………………………….....9

# Introduction

## Document Purpose

The purpose of this document is to provide an overview of the system use-case model in order to provide a high-level understanding of:

* Context -An existing user, web administrator, website system, web moderator
* Scope – A new user joins the service, an existing user creates an event, and an existing user joins an event

## Document Scope

The scope of this document is limited to:

* Diagramming and cataloging the system actors and use cases for the <Rendezvous> project.

The scope of this document does *not* include consideration of:

* Detailed specification of each use case – this is provided separately in a Use-Case Specification document for each use case.

## Document Overview

This document contains the following sections:

* **Brief Description** – reminder of why the use case is needed
* **Use-Case Model Overview** – overview of the use-case model in the form of one or more use-case diagrams with supporting explanatory text
* **Actor Catalog** – catalog of all system actors
* **Use Case Catalog** – catalog of all system use-cases

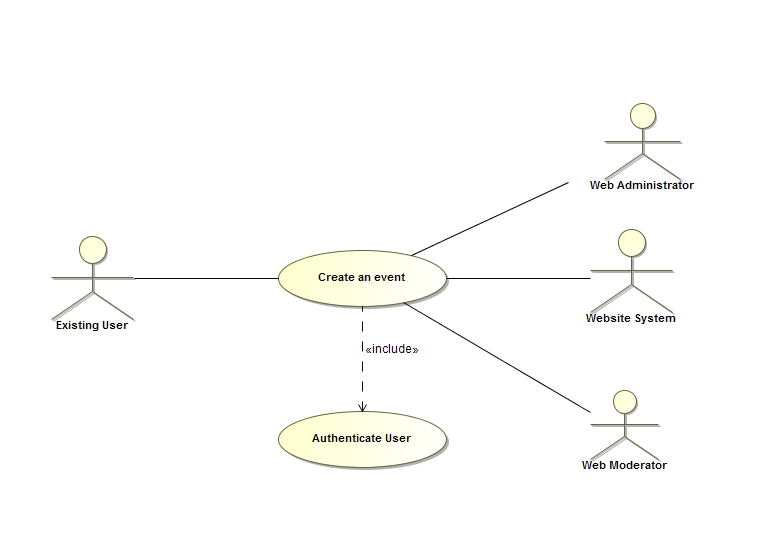
## User Story

* I as the Existing User need a functional log in system to achieve the goal of creating events.

Use-Case Diagrams

This section provides an overview of the use-case model in the form of one or more use-case diagrams with supporting explanatory text.

## <Figure1-3 Create Event>



Actor Catalog

The table below catalogs the system actors, specifying for each actor:

* Name – unique and meaningful name for the actor
* Brief Description – summarizing the role that the actor plays with respect to the system.

| Name | Brief Description |
| --- | --- |
| Existing User | Users with an account that can use all of the website’s functionality. |
| Web Administrator | Makes sure that the functionality of the website Is performing as desired. |
| Website System | Updates website’s database consistently as new users and existing users create and join event. |
| Web Moderator | Tasked with approving or disapproving created events |

Use-Case Catalog

The table below catalogs the system use cases, specifying for each use case:

* ID – unique identifier for the use case
* Name – unique and meaningful name for the use case
* Type:
  + *Base – end-to end interaction between an actor and the system*
  + *Inclusion – abstracted common part of many Base Use Cases that is explicitly referenced by these use cases*
  + *Extension – extension of one or more referenced Base Use-Cases*
  + *Abstract – use case describing generic aspects of many Base Use Cases which in turn specialize this general case in some way.*
* Brief Description – summarizes the use-case purpose in terms of the value produced for its actors and other stakeholders.

| ID | Name | Type | Brief Description |
| --- | --- | --- | --- |
| 0003 | Create Event | Abstract | Existing Users can create new events that can be joined by other users. |

# Use case Specification

|  |
| --- |
| ***The <Create Event> Use-Case Specification specifies all requirements relating to this use case, including the flows through the use case and any non-functional requirements that relate specifically to this use case.*** |

## Document Overview <Use case 0003>

This document contains the following sections:

* **Brief Description** – focusing on the purpose of the use case
* **Preconditions** – the state the system must be in before the use case can performed
* **Basic Flow** – what normally happens when the use case is performed
* **Alternative Flows** – unusual, optional or exceptional use case behavior
* **Post-Conditions** –valid system states after the use case has finished
* **Special Requirements** – non-functional requirements that relate specifically to this use case
* **Extensibility** – where the use case can be extended by other use cases
* **Scenarios** – lists the key representative scenarios for this use-case
* **Additional Information** – any other useful supporting information
* **References** – provides full reference details for all documents, white papers and books that are referenced by this document.

## Brief Description

Existing users can create new events.

## Preconditions

* Users must have a working internet connection.
* Website must be connected to system’s database.
* User must be logged in to an existing account.

## Basic Flow

1. Homepage

* System will allow users to log in while on the homepage.

1. Authenticate User

* Existing users will have the same functionality as visitors; they will be able to view all current events for a specific activity.

1. Creating events

* System will allow users to create new events for desired activities.
* These new events will automatically notify creators of any message or status changes.

## Alternative Flows

1. Homepage: Website Maintenance Error

* If website is unable to load and is currently offline, system apologizes to visitors for inconvenience.
* Visitors have the option to contact Web administrators in order to resolve issue.

1. Homepage: Sign In Error

* Visitor is unable to sign in due to account not existing. System will prompt user to the sign up page.
* Visitor’s newly made account is not stored in system database properly. System will prompt user to sign up page, or to contact a web administrator for further assistance.
* Visitor is unable to sign in due to incorrect password, or username. System prompts user to re-enter credentials.

1. Creating Event: Creation Error

* Existing User is not allowed to create events. System will prompt user to contact web administrator.
* User does not create an event due to empty fields during creation page. System will prompt user to fill in blank fields.
* Event is not stored in database properly.

## Post-Conditions

* Users will able to see a list of their own created events.
* Users will be able to edit their event’s information regarding time and location.
* Users will be notified of any messages posted to their event, as well as any changes to the status of their event.
* Upon creation, users will be notified by Web Moderators if their event is inappropriate and violates website’s terms and conditions.

## Extensibility

No public extension points defined.

## Additional Information

“Authenticate User” is an undocumented use case that is required for multiple use cases in our project.

## 5.9 Scenarios

A scenario is an instance or specific occurrence of a use case. This section lists key representative scenarios for this use-case, listing for each scenario:

* Name – a unique and meaningful name for the scenario
* Flows Exercised – List of the use case flows exercised by the scenario
* Additional Notes – any supporting notes about the scenario or its purpose.

| Name | Flows Exercised | Additional Notes |
| --- | --- | --- |
| Duplicate Event | 5.5.3 Alternative Flow | If a user attempts to create an event that already exists (i.e: similar name/timing), system will notify the user of the reason for error. They will be required to re-enter event information in the appropriate fields. |
| Multiple Events | 5.4.3 Basic Flow | A logged in user is able to create multiple events and will receive notifications for messages posted to each of their events. This is to show that users are able to have their events be organized through the system’s database. |
| Visitor unable to create | 5.4.2 Basic Flow | Visitor will be redirected to sign in page if they attempt to create event. |

# Example of use case specification

This section provides any additional information required to make the use case more accessible to the stakeholders and the development team.

| **Use Case Template** | **Description** |
| --- | --- |
| Use Case Name | Create Event |
| Use Case ID | 0003 |
| Revision and Reference | Revision = V01  Revision = V01 |
| Use Case Diagram | A signed in user should be able to retain all the functionality of the previous use cases. With this case a user can also create a new event which will have the exact properties and features as joining an event with the exception of being able to edit information about the event. |
| Status | In progress/development |
| Priority of accomplishment (optional) | Must have: The system must implement this goal/ assumption to be accepted.  This is one of the key features of the website. |
| Goal | User is able to create events as well as being able to edit them. |
| Summary | User signs into their account, and under list of events will have an option to create a new event. After the event is created they can edit the event as they wish. |
| Actor | Existing User, Web Administrator, Website System, Web Moderator |
| Primary Actor (initiates) | Existing User |
| Preconditions | * User must have internet connection. * System must be connected to database. * User must be signed into a valid account. |
| Main success scenario | 1. User signs in with valid account 2. User views list of events 3. User creates a new event |
| Extensions | 1a. User is now logged in with full functionality  2a. User has options to join and create new event  3a. After creating an event they can edit the information as they please. |
| Post conditions | * Users will able to see a list of their own created events. * Users will be able to edit their event’s information regarding time and location. * Users will be notified of any messages posted to their event, as well as any changes to the status of their event. * Upon creation, users will be notified by Web Moderators if their event is inappropriate and violates website’s terms and conditions. |
| Non-functional requirements | Display of report expected after 20 seconds at the latest. |
| Validation statement | user is able to create an event, find their event under the list as well as edit the information after creation, this user case is successful |
| Additional Info | “Authenticate User” is an undocumented use case that is required for multiple use cases in our project. |
| Author and date | Authors: Avinash Rao, Scott Hinden, Ciao 07/17/2015 |