Rendezvous

Use Case 2: Join Event

Abstract

|  |
| --- |
| ***This document provides an overview of the solution context, scope and high-level functional capabilities for the 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
* 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 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

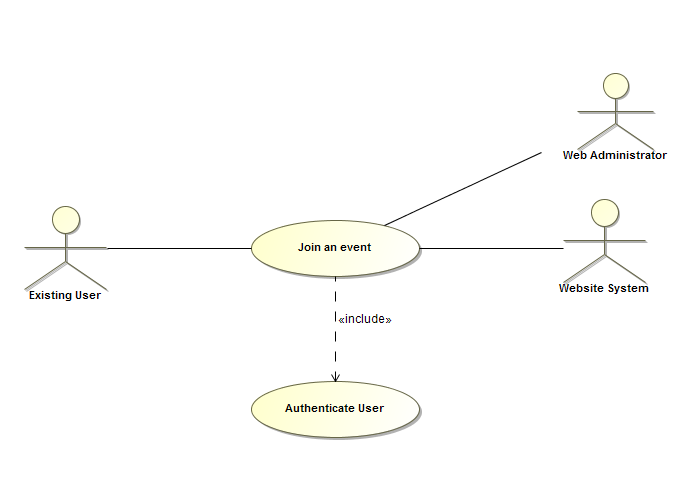
## User Story

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

## <Figure1-2 Join 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 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 |
| --- | --- | --- | --- |
| 0002 | Join Event | Abstract | Existing Users can join created events. |

# Use case Specification

|  |
| --- |
| ***The 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 0002>

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 join created 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

* Include use case *Authenticate Customer* to authenticate the user’s account credentials before allowing the user to access the site.

1. Viewing Events

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

1. Joining Events

* Existing users will have the added functionality of being able to join an event.
* Once an event is joined, the system will continue to notify user of any changes or messages that are posted that are related to the specific event.

## Alternative Flows

1. Homepage: 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. Viewing Events: Event Error

* Events that have already occurred still display to the user under current events.

1. Joining Events: Event Error

* Existing User is unable to join a current event. System will prompt user to notify web administrator or try again later.
* Existing user is not notified properly of changes to events they have already joined.

## Post-Conditions

* User can now see the list of joined events.
* User will now be notified by system of any event status changes.

## 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.

## 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 |
| --- | --- | --- |
| Past Due Event | 5.5 Alternative Flow | User is attempting to join an event that took place already. They will run into errors. |
| Event Deletion | 5.4 Basic Flow | After joining an event, the event is now deleted/cancelled. The user will receive notifications properly of the event’s new status. |
| Event Joining | 5.4 Basic Flow | If a visitor is viewing events, and decides to log in they will be redirected to the event listings and be able to join events accordingly. |

# 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 | Join Event |
| Use Case ID | 0002 |
| Revision | Revision = V01  Revision = V02 |
| Use Case Diagram | A logged in user will attempt to join an event. At this stage the system will check to authenticate the user’s account. If the user is indeed logged in this will be processed by system and add them to the desired event. |
| Status | In progress/development |
| Priority of accomplishment (optional) | Must have: The system must implement this goal/ assumption to be accepted. |
| Goal | System allows users that are logged in join an event of their choosing. |
| Summary | The user is able to log in, view events as well as join events and receive notifications from changes to an event. |
| Actor | Existing User, Web Administrator, Website System |
| Primary Actor (initiates) | Existing User |
| Preconditions | User must have internet connection. System must be connected to database to allow users to access their accounts and posts. |
| Main success scenario | 1. User logs into account 2. User views various events 3. User joins events |
| Extensions | 1a. User enters account info and gains full access  2a. User can view various events that are taking place from several different activities.  3a. User joins an event which allows the system to update the user on changes to an event. |
| Post conditions | User can now see a list of joined events  User will now be notified of any status changes on events |
| Non-functional requirements | Update of status change on notifications expected to send immediately. |
| Validation statement | If a user can join an event successfully, the user case was 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 |