Service Project

Delete user use-case with Individual use-case specifications

User story

I, as an administrator, need the ability to delete users so that I can ensure a positive community.

Abstract

|  |
| --- |
| ***Service is an application open to the public, and as such is vulnerable to abusive, toxic participants. Administrators will have the ability to remove users or property managers from Service if they are not contributing positively to the community. The delete user use-case specification describes all requirements relating to this use case, including the interaction flows, functional requirements, and non-functional requirements*** |

Version History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Author** | **Description** |
| 27/10/2016 | 1 | Nick Petty | Replaced template words, wrote abstract. |
| 29/10/2016 | 2 | Nick Petty | Changed use case to delete user. |
| 30/10/2016 | 2.1 | Nick Petty | Wrote basic and alternative flows. |
| 1/11/2016 | 2.2 | Nick Petty | Completed additional sections. |
| 1/11/2016 | 2.3 | Nick Petty | Completed use case diagram. |

An Essential Unified Process Document

*Table of Contents*

[1 Introduction 3](#_Toc466459645)

[1.1 Document Purpose 3](#_Toc466459646)

[1.2 Document Scope 3](#_Toc466459647)

[1.3 Document Overview 3](#_Toc466459648)

[2 Use-Case Diagrams 4](#_Toc466459649)

[2.1 Delete a user (or property manager) 4](#_Toc466459650)

[3 Actor Catalog 4](#_Toc466459651)

[4 Use-Case Catalog 5](#_Toc466459652)

[5 Use case Specification 6](#_Toc466459653)

[5.1 Introduction 6](#_Toc466459654)

[5.1.1 Document Purpose 6](#_Toc466459655)

[5.1.2 Document Scope 6](#_Toc466459656)

[5.1.3 Document Overview 6](#_Toc466459657)

[5.2 Brief Description 8](#_Toc466459658)

[5.3 Preconditions 8](#_Toc466459659)

[5.4 Basic Flow 8](#_Toc466459660)

[5.5 Alternative Flows 8](#_Toc466459661)

[5.5.1 Website system failure 8](#_Toc466459662)

[5.5.2 Find user by searching 9](#_Toc466459663)

[5.5.3 Cancel action 9](#_Toc466459664)

[5.6 Post-Conditions 9](#_Toc466459665)

[5.7 Special Requirements 10](#_Toc466459666)

[5.8 Extensibility 10](#_Toc466459667)

[5.9 Scenarios 10](#_Toc466459668)

[6 Example of use case specification 12](#_Toc466459669)

# 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 – the people or things that interact with the system (the *Actors*)
* Scope – the things of value that the system performs for its Actors (the *Use Cases*).

## Document Scope

The scope of this document is limited to:

* Diagramming and cataloging the system actors and use cases for the Service 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:

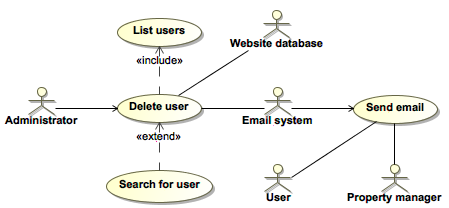
* **Brief Description** – reminder of why the use case is needed
* **Delete user use-case** – overview of the use-case model in the form of one or more use-case diagrams with supporting explanatory text
* **Actor Catalog** – catalog of all system actors
* **Use Case Catalog** – catalog of all system use-cases
* **References** – provides full reference details for all documents, white papers and books that are referenced by this document.

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.

## Delete a user (or property manager)

The administrator views a list of users, including property managers, and selects one to delete. Alternatively, the administrator can search for a specific user and delete them. After confirming deletion, the email system notifies the deleted user. In this use case, “user” can refer to basic Service system users or property managers.



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 |
| --- | --- |
| Administrator | User that is responsible for monitoring the Service community. |
| Website system | Contains a database of users and their service requests. |
| Email system | Sends an email to a deleted user informing them that their account has been closed. |
| User | A person that has an account on Service. |
| Property manager | A person that has an account on Service and has proven ownership or responsibility for a property. |

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 |
| --- | --- | --- | --- |
| 1 | Create account | Base | Allows a person to register as a user, giving them access to use Service. |
| 2 | View service requests | Base | Allows users to view the service requests that have been created. |
| 3 | Delete user | Base | Allows an administrator to remove users and posts that do not conform to Service’s policies. |
| 4 | Respond to service request | Base | Allows a property manager to communicate with users about the status of a service request. |
| 5 | Create service request | Base | Allows a user to create a service request to communicate an issue that needs to be fixed. |
| 6 | Authenticate user | Inclusion | Allows a user to be identified and given access to Service. |
| 7 | List users | Inclusion | Shows an administrator all users of Service. |
| 8 | Search for user | Inclusion | Allows an administrator to find a specific user based on search criteria. |
| 9 | Send email | Base | The email system sends an email to a user. |

# Use case Specification

One section per use cases – according to the use case specification template below

|  |
| --- |
| ***The Delete user 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.*** |

## Introduction

### Document Purpose

The primary objectives of the Use-Case Specification are to:

* Provide a complete set of requirements relating to the use case, including the flows through the use case and all significant external behavior of the use case
* Provide additional supporting information to support the communication of the purpose, requirements, constraints and flows through the use case.

### Document Scope

The scope of this document is limited to consideration of:

* The specification this specific use case, including any non-functional requirements and constraints that relate specifically to this use case.

This scope of this document does not include consideration of:

* Requirements that relate specifically to any other use case – these are described in separate specifications for the other use cases
* Requirements that do not relate to any specific use case – these are described in a separate Supplementary Requirements document.

### Document Overview

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

Administrators will be able to remove users that are a problem in the Service community.

## Preconditions

* Administrator must be logged into the system.
* The website must be connected to the system’s database.
* The email system must be functioning normally.
* The administrator must be able to see the list of existing users.
* The administrator must be able to search for a specific user.

## Basic Flow

{Manage users}

1. The use case begins when the administrator selects to manage the Service system’s users.
2. The administrator is shown a list of all users and property managers, along with a delete button for each.
3. The administrator is shown a search bar for all users and property managers.

{Select to delete user}

1. The administrator clicks the delete button under a user’s name.

{Verify decision}

1. The administrator is asked to confirm the deletion.

{Delete user}

1. The website system verifies that the selected user is valid.
2. The website system deletes the user from the users or property managers table in the database.
3. The administrator is alerted of action completion.

{Send email}

1. The email system sends an email to the deleted user informing of their account being closed.

## Alternative Flows

### Website system failure

#### Unable to load list of users

* If the system is unable to retrieve the list of users, an error message is displayed and the option to retry in given.
* Normal website navigation is available.

#### Unable to {delete user}

* The user selected for deletion cannot be found in the database, or the connection to the database has been interrupted.
* The error is displayed for the administrator and no deletion occurs.

#### Unable to {send email}

* The email system is not responsive, or unable to find a valid address for the deleted user.
* The administrator is alerted of the email failure and shown the address and message that would have been sent.
* The administrator can save the message and address to attempt manually sending the email again later.

### Find user by searching

#### {Search for user}

* The administrator may use the search bar to find a specific user by name, email address, or other information in the user’s profile.
* After the system receives the search terms, it returns a filtered list of users with relevant profile information.
* The administrator can {delete user} at this point.

#### Unable to {search for user}

* If the search finds no matches in the user database, no results are returned.
* The administrator is alerted the no matches were found and is brought back to the {manage users} page.

### Cancel action

#### {Cancel deletion}

* After the administrator selects to {delete user}, they must {verify decision}.
* If they select to cancel the delete action, no deletion takes place and they are returned to the {manage users} page.

## Post-Conditions

The selected user is removed from the Service system. They are sent an email informing them of the action. Their service requests are not removed unless the administrator also takes additional steps to do so. After completing all actions, the administrator is returned to the updated user management page.

## Special Requirements

* Database with users and property managers.
* Reliable, responsive network that does not drop action events.
* Secure administrator account that cannot be accessed outside of trained Service personnel.
* Company policies regarding unacceptable user behaviour that will result in account termination.
* Users informed of account termination policies.
* Email creation system that details why a user’s account was deleted and what courses of action are available.

## Extensibility

No public extension points defined.

## 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 |
| --- | --- | --- |
| Delete Property Manager | 5.4 Basic Flow | Although property managers are a special type of Service user, they are still managed by the same user systems. |
| User does not exist | 5.5.2 Find user by search | If a user does not exist in the system, the search will not return anything. |
| User has already been deleted | 5.4 Basic Flow, 5.5.1.2 Unable to delete user | The user list may be cached on a local machine while another administrator deletes users from the database. The cached list is invalid but not refreshed until an action is taken, like searching or deleting. |
| Email failure | 5.5.1.3 Unable to send email | The deleted user may have an invalid email address, or the email system may be down. This ensures notification can be sent regardless of system functionality. |
| Wrong user selected | 5.5.3 Cancel action | If the search returns the wrong user, or the administrator clicks on the wrong user in the user list, the confirmation dialog will inform them of which user the action is being performed on. |
| No action need | 5.4 Basic flow | If the administrator finds that deleting a user is not necessary, the manage users page has navigation links to the rest of the website. |

# 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.

This template is an extended version of the original template defined by Cockburn [1], in particular extended with a possibility to describe Requested Information Resources often found useful when dealing with data oriented systems. [1] Cockburn, A. Writing Effective Use Cases. ISBN-13: 9780201702255. Addison-Wesley (2001).

| **Use Case Template** | **Description** |
| --- | --- |
| Use Case Name | Remove a user or property manager from the Service application |
| Use Case ID | 3 |
| Revision and Reference | Revision = V02  Reference = not available |
| Use Case Diagram | The administrator opens the user management page. They select a user to delete from the list of users presented or they use the search bar to find a specific user. After choosing the user to delete, the administrator is asked to confirm the action. Upon confirmation, the user is removed from the system and an information email is sent to the deleted user. |
| Status | Planned |
| Priority of accomplishment (optional) | Must have |
| Goal | Administrators are able to delete users from Service. |
| Summary | The administrator uses the user management interface to delete abusive users from Service. |
| Category | Community management |
| Actor | Website administrator, user, property manager, website system, email system |
| Primary Actor (initiates) | Website administrator |
| Stakeholder (optional) | Service community participants |
| Requested Information Resources  (optional) | * Administrator observation (read) * Alert (manage) * User-specific effect (update, manage) |
| Preconditions | * Administrator logged into Service website. * User list populated with system users. * Email server online. |
| Triggers (optional) | User or property manager violates terms of service and their account must be revoked. |
| Main success scenario | 1. Administrator views all users 2. A user is selected for deletion 3. The action is confirmed 4. The user is deleted from Service 5. An email is sent to the deleted user. |
| Extensions | 2a. The administrator uses the search bar  3a. The action is cancelled  5a. The email is saved to be resent later |
| Alternative paths (optional) | 5a. The email address is no longer valid and there is no way to contact the deleted user. |
| Post conditions | * Success message is shown to the administrator. * Copy of email is sent to the administrator. * Database no longer has the deleted user. * The deleted user cannot log in or create new service requests. * The deleted user’s posts are not removed. |
| Non-functional requirements | * Website actions complete in under 5 seconds. * Database and email system are available. * Administrator account is accessed securely. |
| Validation statement | The selected user is deleted from the Service application and cannot log in. |
| Notes | Users in this case refers both to normal users and property managers. |
| Author and date | Nick Petty, November 1, 2016 |