Service Project

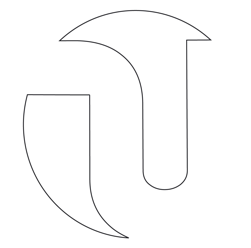
Create Service Request Use-Case

Abstract

|  |
| --- |
| ***This document provides an overview of the solution context, scope and high-level functional capabilities for the Service 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** |
| 30/10/2016 | 0.1 | Gavin Wolf | First pass through template. |
| 30/10/2016 | 0.2 | Gavin Wolf | Adding/revising text content. |
| 01/11/2016 | 0.2 | Gavin Wolf | Working through Cockburn template. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |



An Essential Unified Process Document

*Table of Contents*

[1 Introduction 4](#_Toc465781040)

[1.1 Document Purpose 4](#_Toc465781041)

[1.2 Document Scope 4](#_Toc465781042)

[1.3 Document Overview 4](#_Toc465781043)

[2 Use-Case Diagrams 5](#_Toc465781044)

[2.1 <Diagram Name> 5](#_Toc465781045)

[3 Actor Catalog 5](#_Toc465781046)

[4 Use-Case Catalog 6](#_Toc465781047)

[5 Use case Specification 7](#_Toc465781048)

[5.1 Introduction 7](#_Toc465781049)

[5.1.1 Document Purpose 7](#_Toc465781050)

[5.1.2 Document Scope 7](#_Toc465781051)

[5.1.3 Document Overview 7](#_Toc465781052)

[5.2 Brief Description 9](#_Toc465781053)

[5.3 Preconditions 9](#_Toc465781054)

[5.4 Basic Flow 9](#_Toc465781055)

[5.5 Alternative Flows 9](#_Toc465781056)

[5.5.1 Handle Database Error 9](#_Toc465781057)

[5.5.2 Handle Email System Error 10](#_Toc465781058)

[5.5.3 [Issues like photo size is too big?] 10](#_Toc465781059)

[5.6 Post-Conditions 10](#_Toc465781060)

[5.7 Special Requirements 11](#_Toc465781061)

[5.8 Extensibility 11](#_Toc465781062)

[5.9 Scenarios 11](#_Toc465781063)

[6 Example of use case specification 12](#_Toc465781064)

# 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
* **Create Service Request Use Case Overview** – overview of the Create Service Request use-case in the form of a use-case diagram 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.

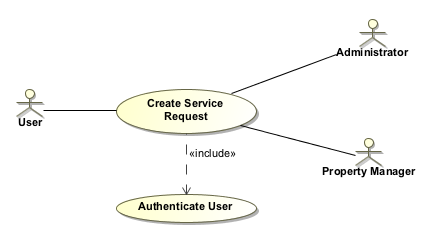
Use-Case Diagrams

This use case describes how a user uses the Service application to create a service request..

## Use-Case Diagram

[<Diagram description>]

<Diagram>



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 |
| --- | --- |
| User | A person who has created an account with Service. |
| Database[?] | Stores service requests and user information. |
| Email system[?] | Sends an email to the user confirming that their service request was received. |
| Administrator[?] | Is alerted to any issues in the process of a new service request being created. |
| Property manager[?] | Is alerted when a new Service Request is created for the property they manage. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

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 |
| --- | --- | --- | --- |
| [xx] | Create service request | Base | A user creates a service request, which can then be viewed by other users, including property managers, who can then work on fulfilling the service request. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Use case Specification

|  |
| --- |
| ***The Create Service Request Use-Case 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

Users will be able to create service requests that will be saved and visible to other users.

## Preconditions

* The user must be logged into the system
* The website must be connected to the system’s database
* The email system must be functioning normally

## Basic Flow

**{Log In}**

1. The use case begins when the actor **User** navigates to the website and satisfactorily inputs their credentials.

**{Select Create Service Request}**

1. The user clicks the option in the navigation bar to create a service request.

**{Populate Service Request Form Fields}**

1. The system displays a service request form, with fields including: “Title”, “Description”, “Location”, “Facility Name,” and “Upload Photo”. [CONFORM THIS TO THE PROJECT PROPOSAL.
2. The user clicks the submit button.

**{Add Service Request to Database}**

1. The information submitted by the user is saved to the **system database**.

**{Send Emails}**

1. The **email system** sends an email to the user confirming that their service request was received.
2. The **email system** sends an email to the **property manager** (if one is on record for the particular property) confirming that their service request was received.

**{Use Case Ends}**

1. The use case ends.

## Alternative Flows

### Handle Database Error

At **{Add Service Request to Database}** the save operation is unsuccessful.

* + - 1. The user is informed that an error has occurred and they should try again at a later time.
      2. [The system creates an **event log** entry to record the fact that a database error occurred. The **event log** entry includes the user and information about the service request if it is available.]
      3. The use case resumes the basic flow at **{Use case ends}**

### Handle Email System Error

At **{Send Email}** the email operation is unsuccessful.

1. [The system creates an event log entry to record the fact that an email system error occurred. The event log entry includes the user and information about the service request if it is available. The administrator will be able to view the event log.]
2. The use case resumes the basic flow at **{Use case ends}**

### [Issues like photo size is too big?]

## Post-Conditions

* The service request has been saved to the database and is available for other users to see.
* The user has received an email confirming that their request was received.
* The property manager (if one is on record) has received an email informing them that a new service request about their property has been submitted.
* Information about any system issues has been added to the event log.

**[SUBFLOWS REQUIRED???]**

## Special Requirements

The system shall correctly register the creation of a service request in at least 99% of instances where a user attempts to create a service request.

## Extensibility

[This section lists any public extension points that the use case makes available for other use cases to extend.]

[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 |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# 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** | **Examples** |
| --- | --- | --- |
| Use Case Name | Create Service Request |  |
| Use Case ID | [xx] |  |
| Revision and Reference | Revision = [0.2]  [Reference = URL of the use case (you get the URL by right-clicking on the entry in the index column)] | V02, http://SDI.server.de/servlet/is/4900/ |
| Use Case Diagram | [Description of the UML use case diagram for the actual use case. The diagram should include extend and include relationships if there is any.  The actual UML diagram figure may be added at the bottom of the template by uploading a bitmap generated from a UML editor.] |  |
| Status | In progress | One of the following:  Planned  in progress |
| Priority of accomplishment (optional) | Must have. Service requests are the lifeblood of the Service product. Users must be able to submit service requests so that property managers and other users can see them, and property managers can respond to them. | One of the following:  Must have: The system must implement this goal/ assumption to be accepted.  Should have: The system should implement this goal/ assumption: some deviation from the goal/assumption as stated may be acceptable.  Could have: The system should implement this goal/assumption, but may be accepted without it. |
| Goal | Service request is posted to website. The property manager associated with the property listed in the service request receives an email alert. |  |
| Summary | The user fills out the Service Request form, including the property, the issue to be fixed, and a picture of the issue. |  |
| Category | [Categorisation of use cases according to overall reference architecture.] [???] | *Context dependent* |
| Actor | The actors involved are users, the database system, the email system, the administrator, and property owners. |  |
| Primary Actor (initiates) | User. |  |
| Requested Information Resources  (optional) | When a service request is created, it must be created in the database and the property manager must be alerted. |  |
| Preconditions | The user must be logged into the system.  The website must be connected to the system’s database  The email system must be functioning normally |  |
| Triggers (optional) | The user clicks on “Create Service Request”. |  |
| Main success scenario | 1. User clicks “Create Service Request”.  2. User enters property name.  3. User enters property location.  4. User enters description of service issue.  5. User uploads picture of issue.  6. User clicks submit.  7. Database entry is created for the service request.  8. The property manager is sent an email alert that a service request has been submitted for one of their properties.  9. The service request is viewable to the public on the main page. |  |
| Extensions | [Extension of an action of the main success scenario. The action to be extended shall be referred to by its number (e.g. 1) appended by a letter (e.g. 1a).] | [1a. The user defines the temporal extent b. The user defines an unavailable temporal extent. A new dialogue window opens and requires a new temporal extent.] |
| Alternative paths (optional) | [Alternate path through the main success scenario w.r.t. an identified action.] | [4a. User can select to view report in different formats, e.g. tabular or graphical map] |
| Post conditions | The service request is viewable on the main page.  The user has been alerted via email that their submission was received.  The property manager has been alerted via email of the new request. | Report is displayed on the screen. |
| Non-functional requirements | [Description of non-functional requirements for this use case with respect to performance, security, quality of service or reliability.] | [Display of report expected after 20 seconds at the latest. ] |
| Validation statement | [List of statements that indicate how to validate the successful realization of the use case.] |  |
| Notes | [Additional notes or comments (also by other users).] |  |
| Author and date | Gavin Wolf. Updated 11/1/16. |  |