|  |
| --- |
| BSHC4, BSHCE4, BSHBIS4, BSHBISE4 |
| Requirements Specification (RS) |
| SaFDesk |

|  |
| --- |
| Keith Kelly  10/18/2012 |

Requirements Specification (RS)

Document Control

Revision History

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Date** | **Version** | **Scope of Activity** | **Prepared** | **Reviewed** | **Approved** |
| 18/10/2012 | 1 | Create | AB |  |  |

Distribution List

|  |  |  |
| --- | --- | --- |
| **Name** | **Title** | **Version** |
| Stephan Weibelzahl | Lecturer | 1 |
| Adrian Skehill | Lecturer | 1 |
|  |  |  |
|  |  |  |
|  |  |  |

Related Documents

|  |  |  |
| --- | --- | --- |
| **Title** | **Comments** | **Version** |
| SafDesk\_Usecase\_Rev1.pdf | Use Case Diagram | 1 |
| SaFDESK\_Architecture\_Rev1.pdf | Architecture Diagram | 1 |
| SaFDesk\_Mockup\_Rev1.png | GUI Mockups Created in Balsamic | 1 |

**Table of Contents**

[1 Introduction 4](#_Toc316977392)

[1.1 Purpose 4](#_Toc316977393)

[1.2 Project Scope 4](#_Toc316977394)

[1.3 Definitions, Acronyms, and Abbreviations 4](#_Toc316977395)

[2 User Requirements Definition 4](#_Toc316977396)

[3 Requirements Specification 4](#_Toc316977397)

[3.1 Functional requirements 4](#_Toc316977398)

[3.1.1 Use Case Diagram 5](#_Toc316977399)

[3.1.2 Requirement 1 <name of requirement in a few words> 5](#_Toc316977400)

[3.1.3 Requirement 2 <name of requirement in a few words> 6](#_Toc316977401)

[3.2 Non-Functional Requirements 7](#_Toc316977402)

[3.2.1 Performance/Response time requirement 8](#_Toc316977403)

[3.2.2 Availability requirement 8](#_Toc316977404)

[3.2.3 Recover requirement 8](#_Toc316977405)

[3.2.4 Robustness requirement 8](#_Toc316977406)

[3.2.5 Security requirement 8](#_Toc316977407)

[3.2.6 Reliability requirement 8](#_Toc316977408)

[3.2.7 Maintainability requirement 8](#_Toc316977409)

[3.2.8 Portability requirement 8](#_Toc316977410)

[3.2.9 Extendibility requirement 8](#_Toc316977411)

[3.2.10 Reusability requirement 8](#_Toc316977412)

[3.2.11 Resource utilization requirement 8](#_Toc316977413)

[4 GUI 8](#_Toc316977414)

[5 System Architecture 8](#_Toc316977415)

[6 System evolution 8](#_Toc316977416)

# Introduction

## Purpose

The purpose of this document is to set out the requirements for the development of SaFDesk, a Helpdesk management and Renewal notification platform.

The intended customers are companies who specialise in managing I.T infrastructure for a number of clients who currently do not use any Helpdesk software an currently do not have a formal renewal process.

## Project Scope

The scope of the project is to develop a simple and functional ticket logging application and a renewal tracking and notification system that has no need to be hosted on a client site. The project is to be designed as an in house application for my current company, ICT Project Management, and will allow us to streamline a number of daily, weekly and monthly tasks that we currently carry out.

.

## Definitions, Acronyms, and Abbreviations

JB Jboss

OS OpenShift

# User Requirements Definition

* The Ability to Record Support calls History via a ticketing mechanic.
* The ability to review existing tickets.
* The ability to edit existing tickets.
* The ability to close open tickets.
* The ability to track renewals.
* To be alerted when a renewal is approaching (SMS or Email).

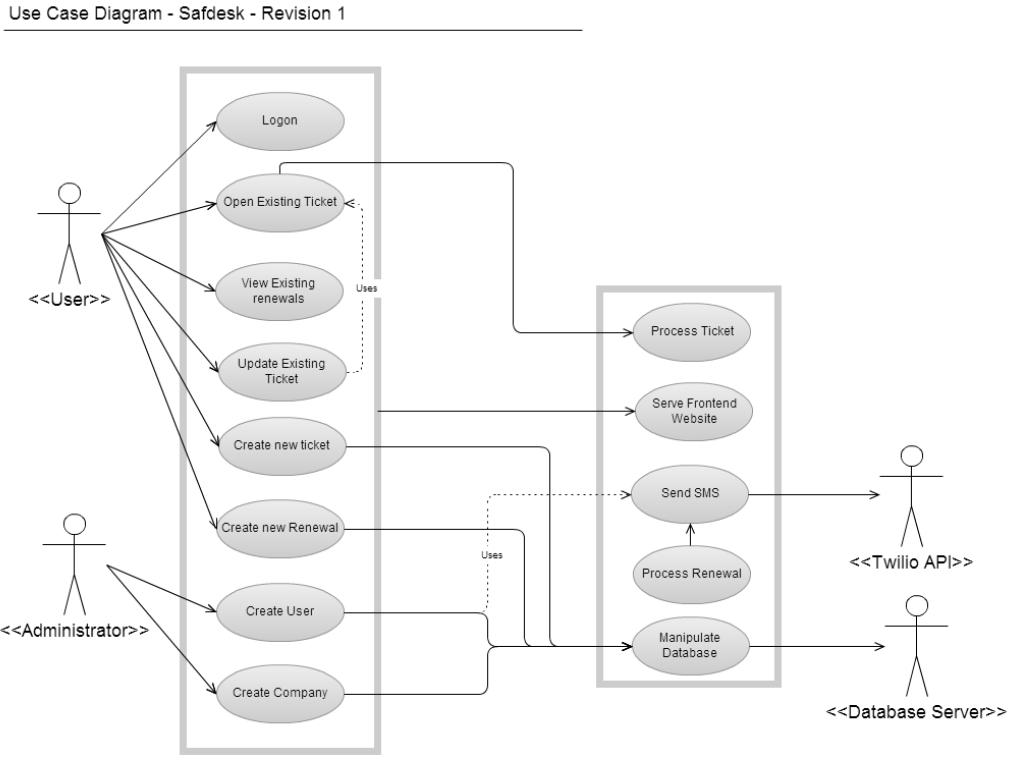
# Requirements Specification

## Functional requirements

The requirements below are based on the concept of the project and as such, are subject to change. This document will be updated as these requirements are updated or changed.

### Use Case Diagram

Below is low Resolution. Please see Attachment 1.



### 

### Requirement 1 – Logon

#### Description & Priority

The user logs on and is authenticated to the system. Their privilege level is obtained and they are presented with their available options. This is a top priority as it allows use of the system

#### Use Case

**Scope**

The scope of this use case is to allow the user to logon and access the system.

**Description**

This use case describes the process by which the user logs on to the system.

**Flow Description**

**Precondition**

The logon page is open in the user’s browser.

**Activation**

This use case starts when a user enters their username/password and hits the logon button.

**Main flow**

1. The System checks to see if the user exists (See E1)
2. The User is assigned the User Privilege (See A1)
3. The system presents the user with the options available to them

**Alternate flow**

A1 : Administrator Logon

1. The User is assigned the Administrator Privilege.
2. The use case continues at position 3 of the main flow.

**Exceptional flow**

E1 : Failed Logon

1. The system Cannot find the username/password pair entered.
2. The user is denied access.
3. The system returns to the logon page.

**Termination**

The system presents the options available to the user.

**Post condition**

System awaits user input.

### Requirement 2 – Open existing ticket

#### Description & Priority

The user opens an existing ticket. This is an elected process and as such, is a low priority.

#### Use Case

**Scope**

The scope of this use case is to allow the user to view an existing ticket.

**Description**

This use case describes the process by which a user views an existing ticket

**Flow Description**

**Precondition**

The user is logged into the system

**Activation**

This use case starts when the user selects an existing ticket

**Main flow**

1. The system queries the database for the ticket information
2. The ticket information is displayed.

**Alternate flow**

A1: No Alternate Flow

**Exceptional flow**

E1: No Exceptional Flow

**Termination**

The ticket information is displayed

**Post condition**

Ticket is viewable

### Requirement 3 – View existing Renewals

#### Description & Priority

The user views an existing renewal. This is an elected process and as such, is a low priority.

#### Use Case

**Scope**

The scope of this use case is to allow the user to view the details of an existing renewal.

**Description**

This use case describes the process by which the user views an existing renewal.

**Flow Description**

**Precondition**

The user is logged into the system.

**Activation**

This use case starts when the user selects an existing renewal.

**Main flow**

1. The system queries the database for the renewal details.
2. If the renewal is open the renewal is displayed (See A1).

**Alternate flow**

A1 : Expired Renewal

1. The System displays an expired renewal warning.

**Exceptional flow**

E1 : No exceptional flow.

**Termination**

The system displays the renewal information.

**Post condition**

Renewal information is viewable.

### Requirement 4 – Update existing ticket

#### Description & Priority

The user manipulated and changes data contained within a currently existing ticket. This is a main requirement of the application so is medium priority.

#### Use Case

**Scope**

The scope of this use case is to allow the user to edit an existing ticket.

**Description**

This use case describes the process by which the user makes changes to an existing ticket.

**Flow Description**

**Precondition**

Requirement 2 – Open existing ticket, has been successful.

**Activation**

This use case starts when a user selects to edit an open existing ticket.

**Main flow**

1. The System checks to see if the ticket is open (See A1).
2. Ticket is now editable.
3. Ticket changes are made.

**Alternate flow**

A1 : Ticket is closed

1. The ticket is currently marked as closed.
2. Ticket may be marked as open (Continues at Step 2 in main flow).
3. System returns to main menu.

**Exceptional flow**

E1 : No exceptional flow.

**Termination**

The user saves the changes made.

**Post condition**

System is at main menu.

### Requirement 5 – Create new ticket

#### Description & Priority

The user creates a new ticket and enters the required data. This is the primary action for the system and as such, is high priority.

#### Use Case

**Scope**

The scope of this use case is to allow the user to create a new ticket.

**Description**

This use case describes the process by which the user creates and populated a new ticket.

**Flow Description**

**Precondition**

User is logged in.

**Activation**

This use case starts when the user selects the Create new ticket action.

**Main flow**

1. The system gets the next available ticket ID.
2. The system displays the relevant fields.
3. The user selects the company from a drop down.
4. The user populates the fields required.
5. The user submits the ticket (See E1).

**Alternate flow**

A1 : No Alternate Flow.

**Exceptional flow**

E1 : Missing Data

1. The user neglected to enter a required piece of data.
2. The ticket entry is displayed, highlighting the missing field.
3. Continues at step 4 in main flow.

**Termination**

The user receives notification that the ticket was submitted successfully.

**Post condition**

The system is at the main menu.

### Requirement 6 – Create new renewal

#### Description & Priority

The user creates a new renewal and enters the required data. This is the primary action for the system and as such, is high priority.

#### Use Case

**Scope**

The scope of this use case is to allow the user to create a new renewal.

**Description**

This use case describes the process by which the user creates and populated a new renewal.

**Flow Description**

**Precondition**

User is logged in.

**Activation**

This use case starts when the user selects the Create new renewal action.

**Main flow**

1. The system gets the next available renewal ID.
2. The system displays the relevant fields.
3. The user selects the company from a drop down.
4. The user populates the fields required.
5. The user submits the renewal (See E1).

**Alternate flow**

A1 : No Alternate Flow.

**Exceptional flow**

E1 : Missing Data

1. The user neglected to enter a required piece of data.
2. The renewal entry is displayed, highlighting the missing field.
3. Continues at step 4 in main flow.

**Termination**

The user receives notification that the renewal was submitted successfully.

**Post condition**

The system is at the main menu.

### Requirement 7 – Create new user

#### Description & Priority

This allows an administrator to create new users and grant them access to the system.

#### Use Case

**Scope**

The scope of this use case is to allow an administrator to add a new user.

**Description**

This use case describes the process by which an administrator creates a new user.

**Flow Description**

**Precondition**

A user is logged in with administrator privileges.

**Activation**

The administrator selects the add new user action.

**Main flow**

1. The new user entry is displayed.
2. The administrator populates the required fields.
3. The system checks to see if the user exists. (See E1)
4. The administrator saves the new user.
5. The new user is notified (SMS/Email) of their username/password.

**Alternate flow**

A1 : No alternate flow

**Exceptional flow**

E1 : User already exists.

1. The defined user already exists.
2. The administrator is prompted to change the username.
3. Continues at step 3 in main flow.

**Termination**

The administrator saves the new user.

**Post condition**

System is at main menu.

### Requirement 8 – Create new company

#### Description & Priority

This allows an administrator to create new company.

#### Use Case

**Scope**

The scope of this use case is to allow an administrator to add a new company.

**Description**

This use case describes the process by which an administrator creates a new company.

**Flow Description**

**Precondition**

A user is logged in with administrator privileges.

**Activation**

The administrator selects the add new company action.

**Main flow**

1. The new company entry is displayed.
2. The administrator populates the required fields.
3. The system checks to see if the company exists. (See E1)
4. The administrator saves the new company.

**Alternate flow**

A1 : No alternate flow

**Exceptional flow**

E1 : Company already exists.

1. The defined company already exists.
2. The administrator is prompted to change the company.
3. Continues at step 3 in main flow.

**Termination**

The administrator saves the new company.

**Post condition**

System is at main menu.

## Non-Functional Requirements

These requirements will be specified during development when the full scope and capabilities of the platform technologies are tested.

# Interface requirements

## GUI

Please see attached PDF’s including the GUI mockups.

## Application Programming Interfaces (API)

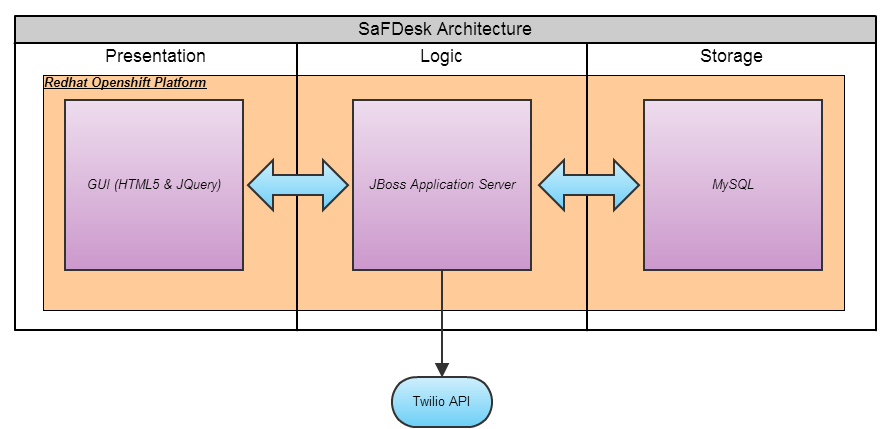
The end goal of this project is to expose an API to allow Logging and viewing of tickets. This API will then be able to be used for application development, possibly on an android handset or other device.

API’s being used include:

Twilio: <http://www.twilio.com/docs/api/rest>

# System Architecture

Below is a diagram outlining the 3-Tier architecture the application will be designed around and at this conceptual stage a class diagram is not possible but will be added as the development process happens.



# System Evolution

This section is subject to development and will be updated in further revisions.