Project CM

UC: Manage appointments

Version 1.1

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Description | Author |
| 17/Oct/2013 | 1.0 | Initial UC description | Storm Cloud Development |
| 19/Jun/2014 | 1.1 | Inserted function points | Storm Cloud Development |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Use-Case book meeting rooms 4

1.1 Brief Description 4

1.2 Stored data 4

1.2.1 Overview 4

1.2.2 Description 4

2. Flow of Events 5

2.1 Screen Flow 5

2.2 Use Case Flows 5

2.2.1 Show all bookings (List View) 5

2.2.2 Show single booking (View Single) 6

2.2.3 Create new booking (Create) 7

2.2.4 Edit booking (Edit) 8

2.2.5 Delete booking (Delete) 9

3. Special Requirements 10

4. Preconditions 10

4.1 Rights 10

5. Postconditions 10

5.1 Appointment 10

6. Extension Points 10

7. Selenium test scenarios 10

8. Mockups 10

9. Function Points 11

UC: Manage appointments

# Use-Case book meeting rooms

## Brief Description

This use case describes how a user can manage his single appointments. The user can add, read, update and delete appointments (CRUD).

## Stored data

### Overview

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Auto-assigned | Unique | Mandatory | Modifiable |
| Id | Yes | Yes | Yes | No |
| Calendar | No | No | Yes | Yes |
| Title | No | No | Yes | Yes |
| Description | No | No | No | Yes |
| Start date | No | No | Yes | Yes |
| End date | No | No | Yes | Yes |

### Description

Each appointment has a start and an endtime. It also has a title to identify it and a description for further information, which is optional. Each appointment belongs to one calendar and is identified by its id.

# Flow of Events

## Screen Flow



## Use Case Flows

### Show all bookings (List View)



### Show single booking (View Single)



### Create new booking (Create)



### Edit booking (Edit)



### Delete booking (Delete)



# Special Requirements

n/a

# Preconditions

## Rights

View/edit/delete/create: The user must have the rights to do the operation.

# Postconditions

## Appointment

Edit: The appointment is updated

Delete: The appointment is deleted

Create: The appointment is created

# Extension Points

n/a

# Selenium test scenarios

Scenario 1: add appointment

Given I am logged in

And I am on the "calendar"

And I clicked "new"

When I type in the correct calendar data

And I click "create"

Then the appointment should be created

Scenario 2: delete appointment

Given I am logged in

And I am on the "calendar"

And I clicked "delete"

When I click “confirm”

Then the appointment should be removed

# Mockups

n/a

# Function Points

Below you can see the function point calculation for this use case.

