

# COSBAS Functional Requirements Documentation

Git: <https://github.com/undecidables/Requirements-Documentation>

**<undecidables>**

Elzahn Botha *13033922*  
Jason Richard Evans *13032608*  
Renette Ros *13007557*  
Szymon Ziolkowski *12007367*  
Tienie Pritchard *12056741*  
Vivian Venter *13238435*

**March 2015**

# Contents

|          |  |          |
|----------|--|----------|
| <b>1</b> | <b>Introduction</b>                      | <b>2</b> |
| <b>2</b> | <b>Domain Model</b>                      | <b>2</b> |
| <b>3</b> | <b>Web Authentication</b>                | <b>2</b> |
| <b>4</b> | <b>Appointments</b>                      | <b>2</b> |
| <b>5</b> | <b>Availability</b>                      | <b>2</b> |
| 5.1      | Choose Calendar . . . . .                | 2        |
| 5.1.1    | Description . . . . .                    | 2        |
| 5.1.2    | Pre/Post conditions . . . . .            | 3        |
| 5.1.3    | Functionality . . . . .                  | 3        |
| 5.1.4    | Process Specification . . . . .          | 3        |
| 5.2      | View Availability . . . . .              | 3        |
| 5.2.1    | Description . . . . .                    | 3        |
| 5.2.2    | Pre/Post conditions . . . . .            | 3        |
| 5.2.3    | Functionality . . . . .                  | 3        |
| 5.2.4    | Process Specification . . . . .          | 3        |
| 5.3      | Set Availability . . . . .               | 3        |
| 5.3.1    | Description . . . . .                    | 3        |
| 5.3.2    | Pre/Post conditions . . . . .            | 3        |
| 5.3.3    | Functionality . . . . .                  | 4        |
| 5.3.4    | Process Specification . . . . .          | 4        |
| <b>6</b> | <b>Biometric Access</b>                  | <b>4</b> |
| 6.1      | Request Access . . . . .                 | 4        |
| 6.1.1    | Use Case diagram . . . . .               | 5        |
| 6.1.2    | Pre-/Post-Conditions . . . . .           | 5        |
| 6.1.3    | Process Specification . . . . .          | 6        |
| <b>7</b> | <b>Use Cases / Services Contracts</b>    | <b>6</b> |
| 7.1      | Use Cases / Services Contracts . . . . . | 6        |
| 7.2      | Authentication . . . . .                 | 7        |
| 7.2.1    | Alter Appointment . . . . .              | 8        |
| 7.3      | Required Functionality . . . . .         | 10       |
| 7.3.1    | Authentication . . . . .                 | 10       |
| 7.3.2    | Alter Appointment . . . . .              | 11       |
| 7.4      | Process Specification . . . . .          | 11       |
| 7.4.1    | Authentication . . . . .                 | 11       |
| 7.4.2    | Alter Appointment . . . . .              | 12       |

# 1 Introduction

In this section of the document we will identify and address a high level overview of the COSBAS (Computer Science Biometric Access System) system.

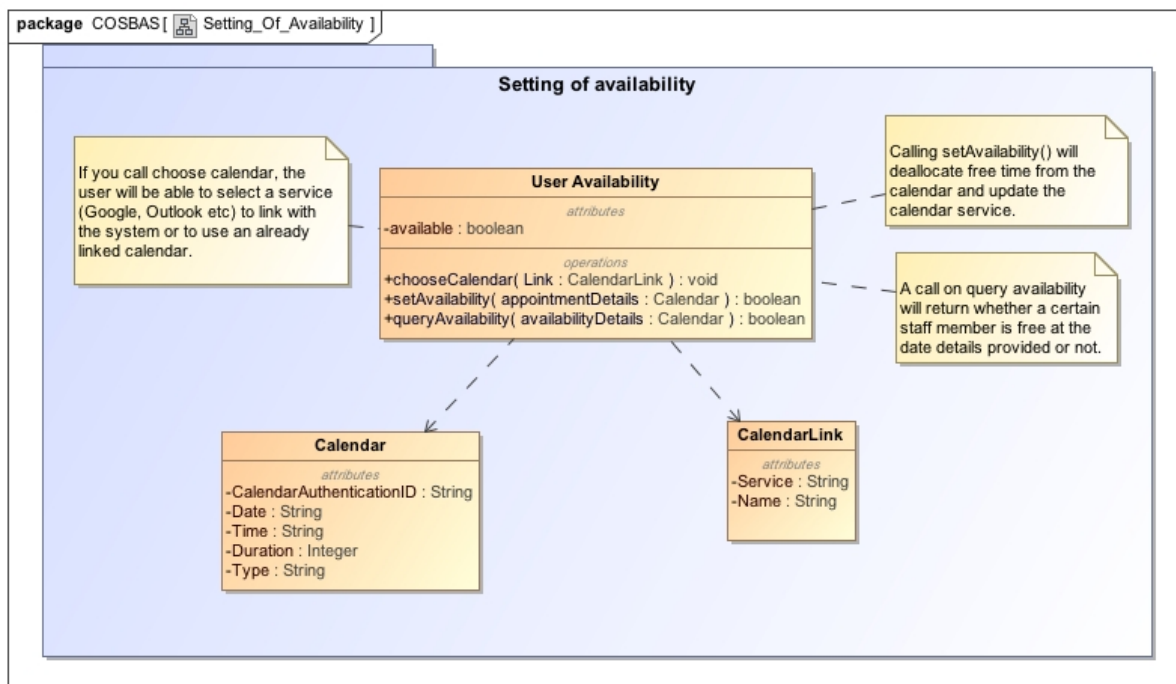
## 2 Domain Model

## 3 Web Authentication

## 4 Appointments

## 5 Availability

The following functionality of creating and setting will be available to only signed in users. Users who are not authorised or signed into the system will only be granted read-only privileges.



### 5.1 Choose Calendar

*Priority: Critical*

#### 5.1.1 Description

An authorised user of the system should be able to select which calendar account to associate with the system. If the user chose a linked service, then that service needs to be authenticated.

### 5.1.2 Pre/Post conditions

**Pre-conditions:** The user should be logged into the system and already have a calendar account from the linked services (Google & Outlook).

**Post-conditions:** Once the user has chosen the calendar to associate with the system, all updates in availability will be amended in that specific calendar.

### 5.1.3 Functionality

### 5.1.4 Process Specification

## 5.2 View Availability

*Priority: Critical*

### 5.2.1 Description

Both the authorised user and a non-authorised user (or guest) should be able to have a read-only functionality of viewing the availability of the associated staff member on their calendar.

### 5.2.2 Pre/Post conditions

**Pre-conditions:** An associated calendar for the staff member should exist.

**Post-conditions:** Be able to see whether the associated staff member has an appointment at a certain date and time or if they are free for appointments.

### 5.2.3 Functionality

### 5.2.4 Process Specification

## 5.3 Set Availability

*Priority: Critical*

### 5.3.1 Description

The authorised user should be able to set whether s/he is available at a certain date or time or if they have an appointment at a specific date or time. This can be set by the staff member himself/herself or by the approval of an appointment.

### 5.3.2 Pre/Post conditions

**Pre-conditions:** The user should be authorised and logged in and a calendar should exist.

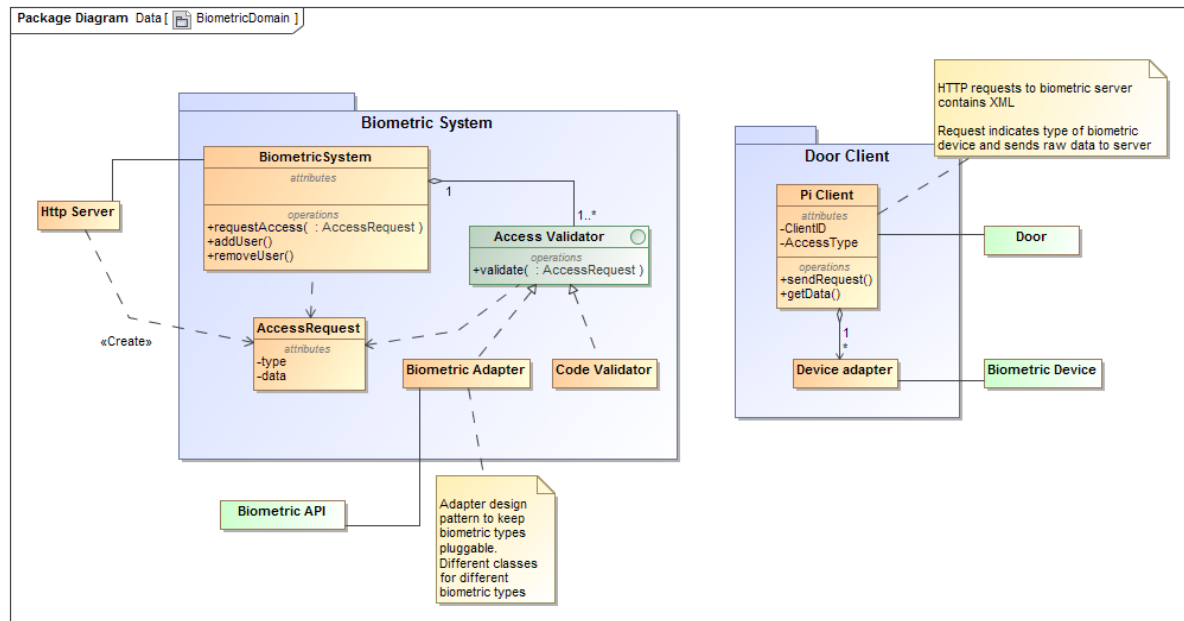
**Post-conditions:** The linked calendar should be updated with the particulars specified by the authorised user. (CRUD of appointments).

### 5.3.3 Functionality

### 5.3.4 Process Specification

## 6 Biometric Access

This module encapsulates functionality regarding validating biometric data or temporary access code to give staff members or visitors access to the Computer Science Department



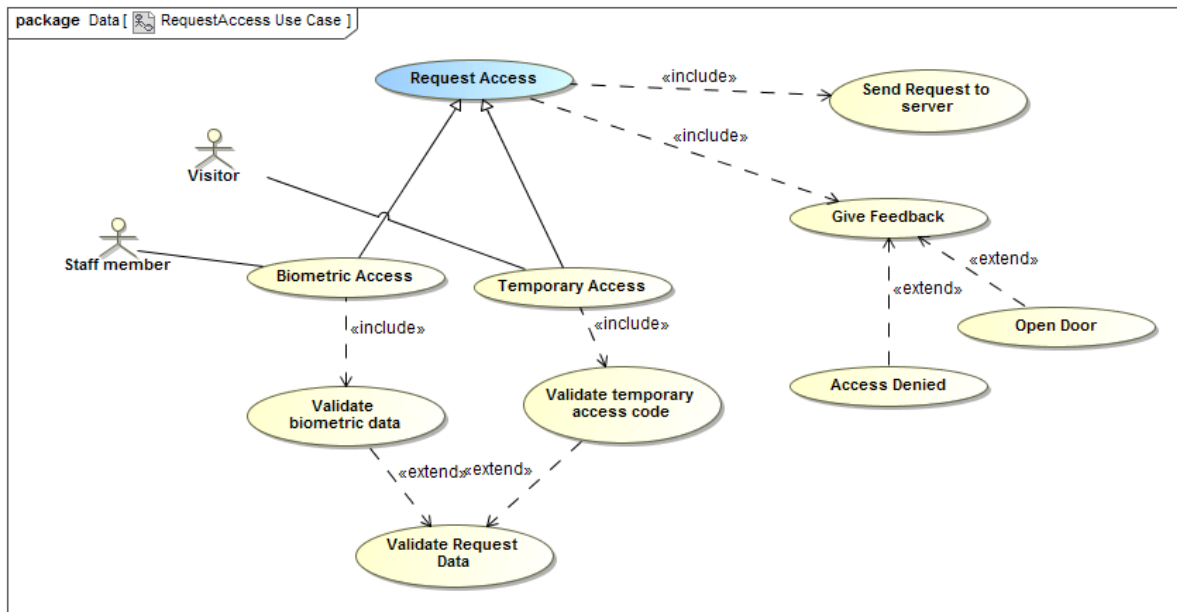
### 6.1 Request Access

*Priority: Important*

**Staff Access** A staff member can gain access at a door using biometrics.

**Visitor Access** A visitor can enter the department by entering their temporary access code at the door.

### 6.1.1 Use Case diagram



### 6.1.2 Pre-/Post-Conditions

#### Staff Access Pre-conditions:

- Staff member must be registered on system.
- Biometric data must validate correctly

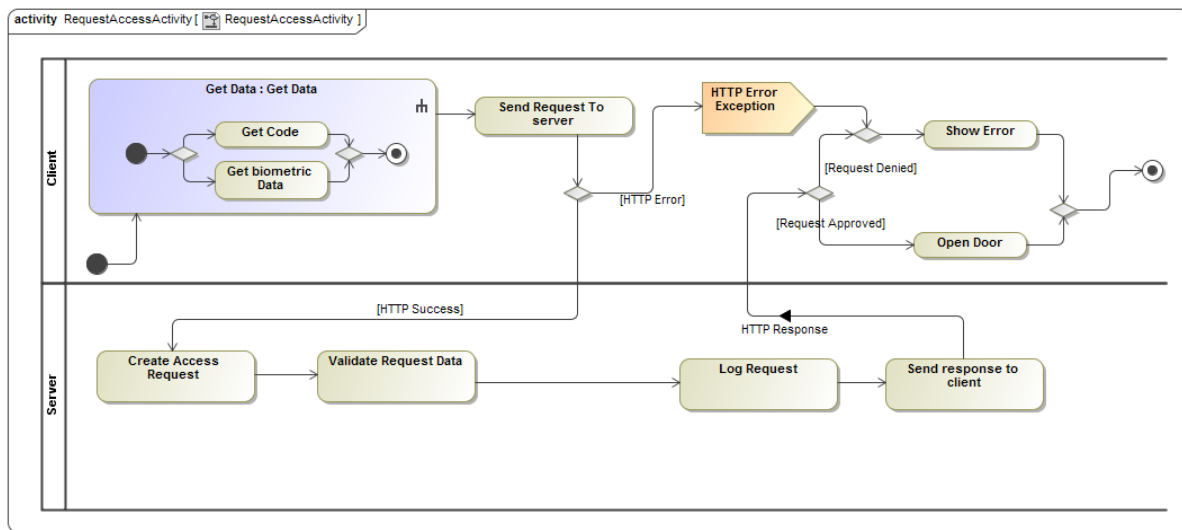
#### Visitor Access Pre-conditions:

- Visitor needs to have an appointment
- Temporary access code should be valid

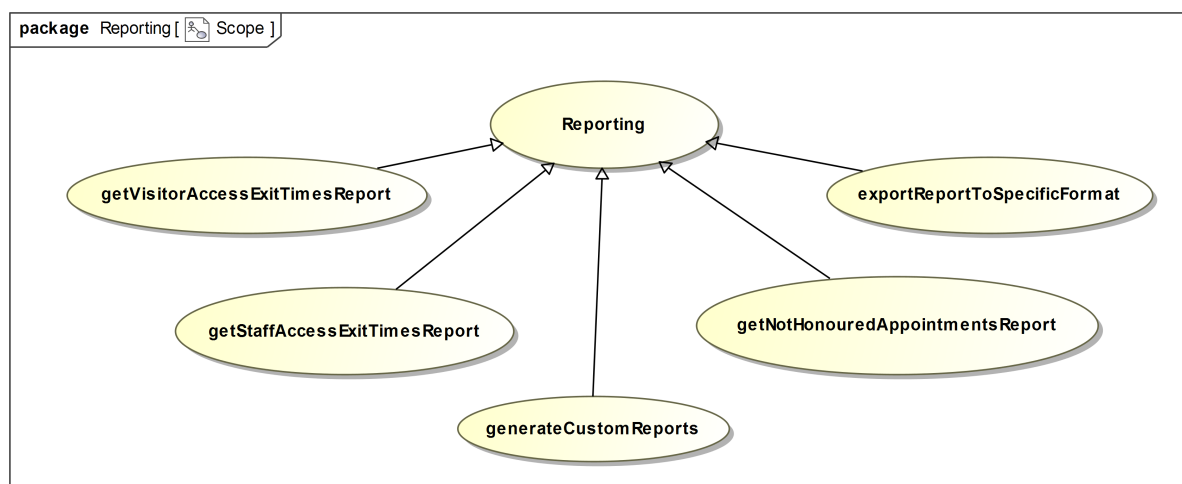
#### Post-conditions:

- User gains access
- Access logged

### 6.1.3 Process Specification



## 7 Reporting



### 7.1 getVisitorAccessExitTimes

*Priority: Important*

The getVisitorAccessExitTimes function allows a user to query the access and exit times of a specific visitor.

**Pre-conditions:**

- Visitor must exist.
- Date must be valid.

## 7.2 getStaffAccessExitTimesReport

*Priority: Important*

The getStaffAccessExitTimes function allows a privileged user(e.g: Head of Department) to query the access and exit times of a specific staff member.

### Pre-conditions:

- User must have the correct authorization to make use of this function.
- Date must be valid, the day must have occurred. System can not tell the future yet!
- Staff member must exist.

## 7.3 getNotHonouredAppointmentsReport

*Priority: Important*

The getNotHonouredAppointments function provides the user with a means of querying a visitor's or staff member's not honoured appointments.

### Pre-conditions:

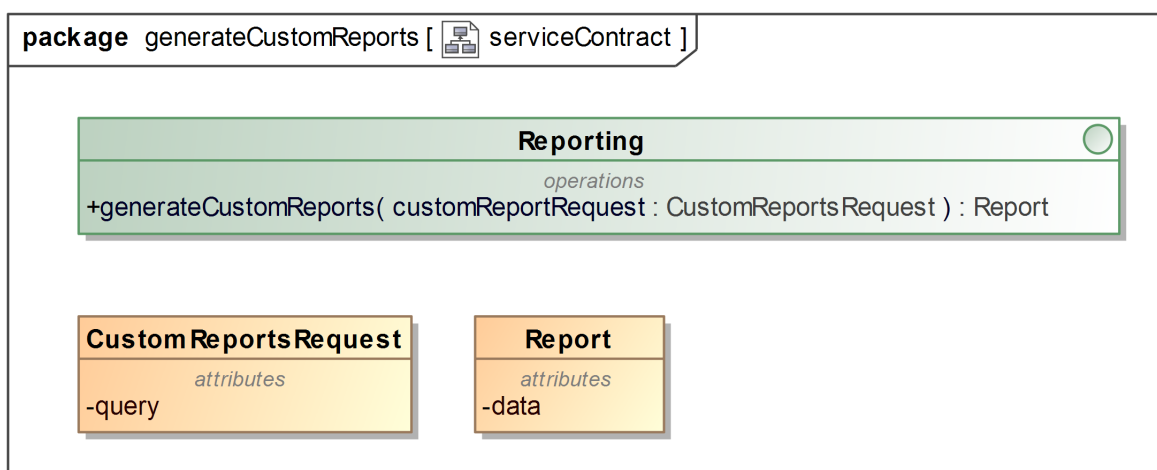
- Visitor or Staff must exist.

## 7.4 generateCustomReports

*Priority: Nice to have*

The generateCustomReports function allows the user to create a custom report based on a query provided by the user.

### 7.4.1 Service Contract



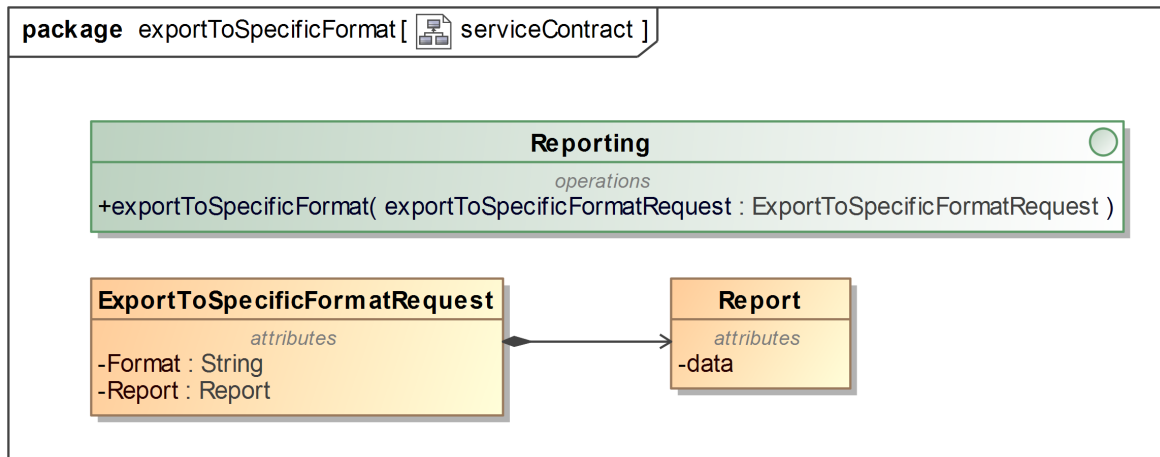
## 7.5 exportReportToSpecificFormat

*Priority: Important*



The `exportReportToSpecificFormat` function will allow the user to export a report to a specific format which will be specified by the user.

### 7.5.1 Service Contract



#### Pre-conditions:

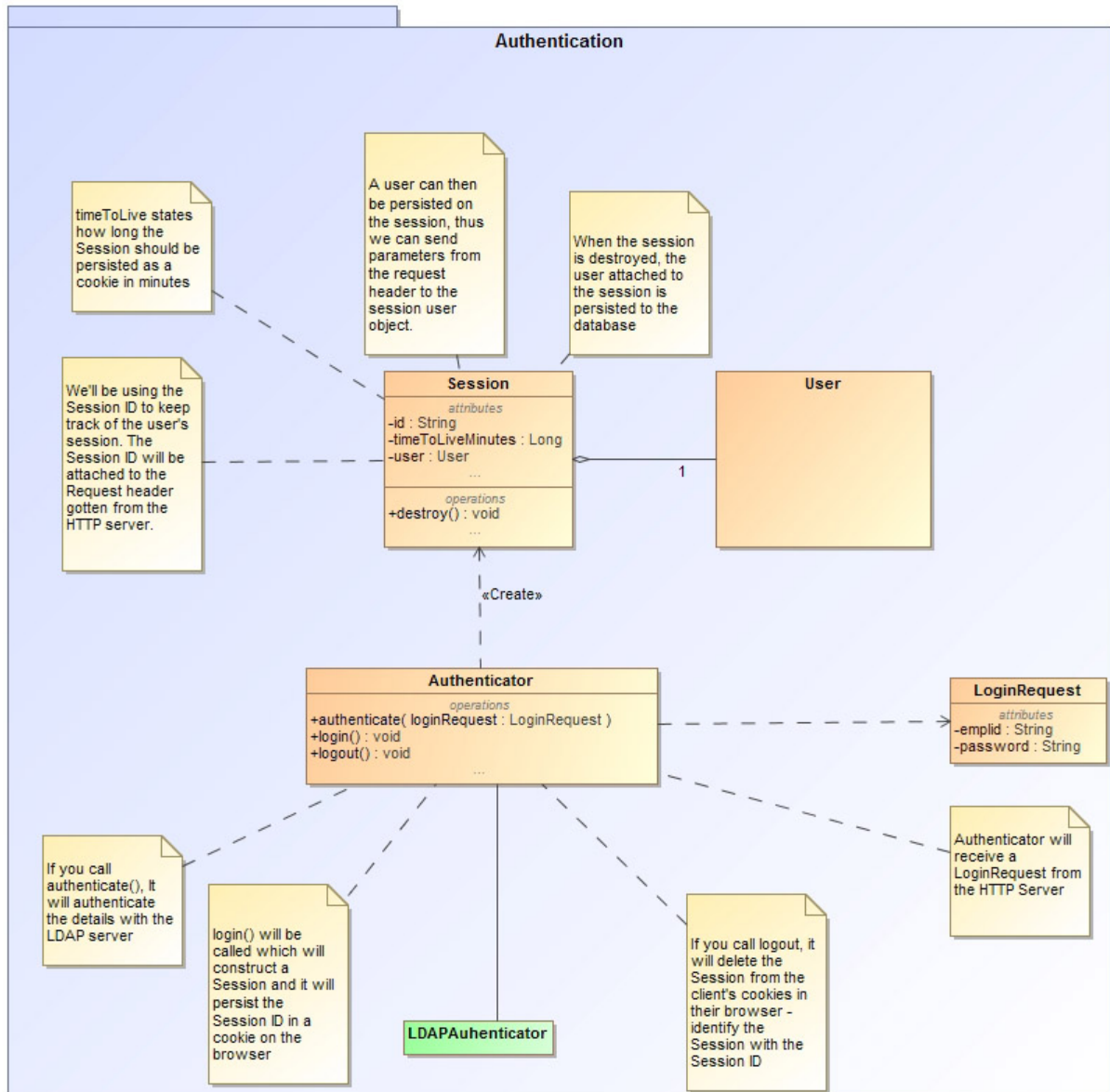
- The format specified by the user is supported by the system

## 8 Use Cases / Services Contracts

### 8.1 Use Cases / Services Contracts

## 8.2 Authentication

The following section will describe functionality around logging staff members in and out.



### Authenticate

*Priority: Critical*

A user's details need to be authenticated with LDAP.

#### Pre-conditions:

- A user must have an EMPLID from the University of Pretoria
- A user must have an associated password for their EMPLID from the University
- A successful connection to LDAP is important
- A user must be registered as a staff member on LDAP
- A successful validation response after an LDAP authentication is needed to authenticate a user

#### Post-conditions:

- A user is successfully authenticated on the server

## Log In

*Priority: Critical*

A user must be logged into the system once they have been authenticated. A user is logged in by creating a cookie containing the session ID.

### Pre-conditions:

- A user must have been successfully authenticated by the system to be logged in.

### Post-conditions:

- A user is successfully logged in and can thus access features which require authentication.
- A user is taken to the booking management page on the website

## Log Out

*Priority: Critical*

The system must be able to log a user out. A user is logged out by destroying the cookie containing their session ID.

### Pre-conditions:

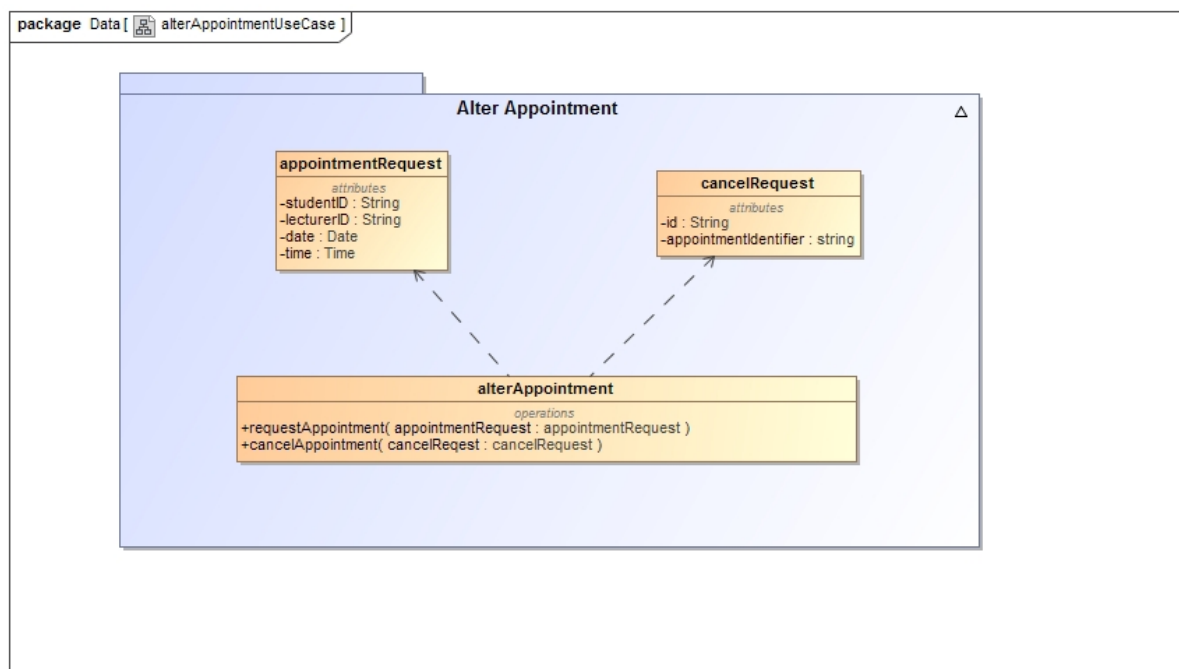
- A user must be logged in before they can be logged out

### Post-conditions:

- A user's session cookie is destroyed and they are logged out on the system and taken back to the home page.

### 8.2.1 Alter Appointment

This section will describe the functionality regarding requesting and cancelling an appointment with a lecturer.



## Requesting Appointment

*Priority: Important*

The requestAppointment function allows a user of the system to request an appointment with a staff member that is also making use of the system.

**Pre-conditions:**

- Lecturer must exist.
- Date and time of the requested appointment must be valid entries.

**Post-conditions:**

- Appointment will be saved for the lecturer to approve or deny later on.
- User gets an appointment identifier back.
- Lecturer is notified of the requested appointment.

**Cancelling Appointment**

*Priority: Important*

The `cancelAppointment` function allows the user to cancel an appointment if the user is either the staff member who's appointment it is or the user who made the appointment.

**Pre-conditions:**

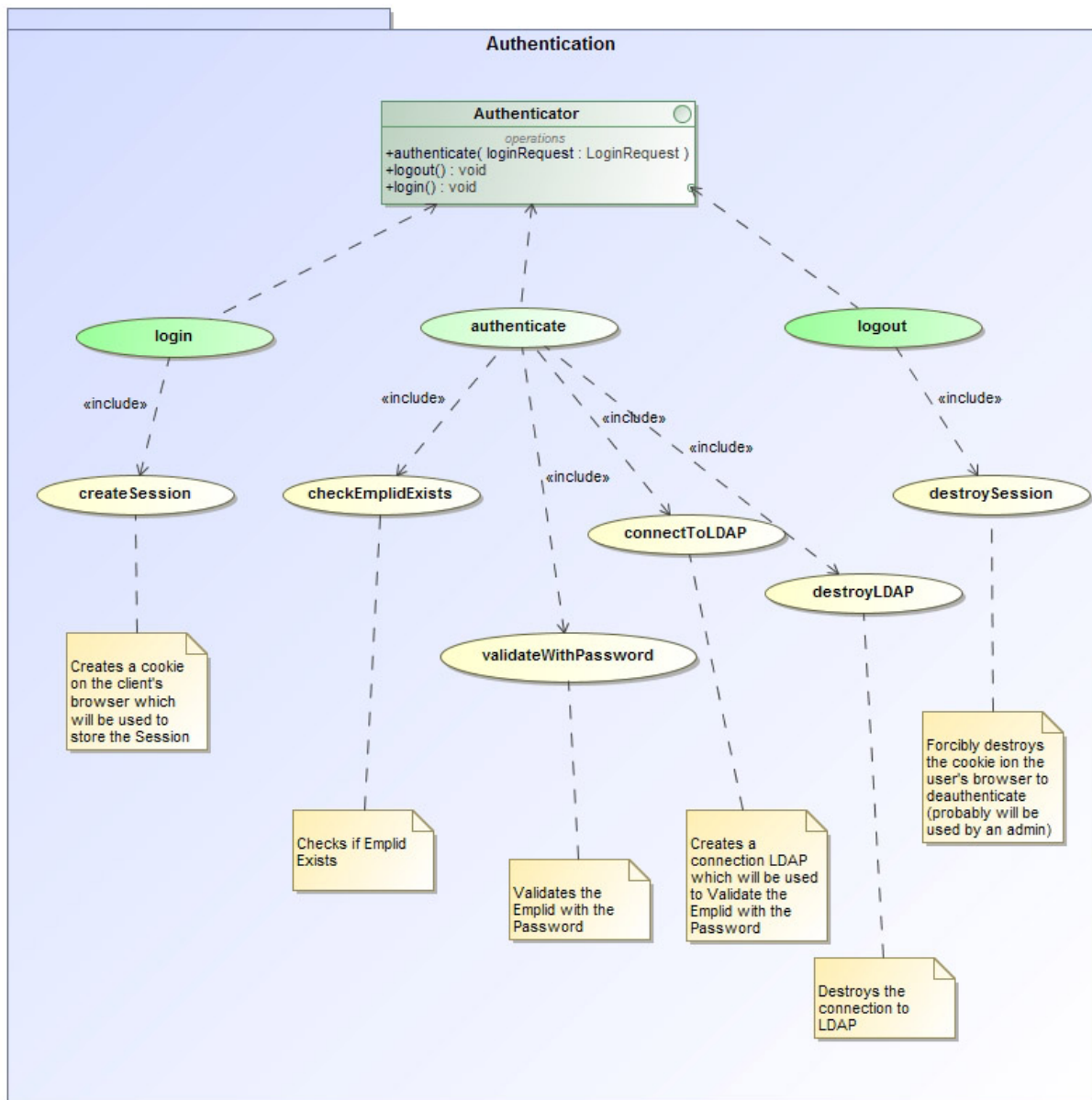
- The appointment must exist.
- The user cancelling the appointment has to be the person that the appointment is with or the person who made the appointment.

**Post-conditions:**

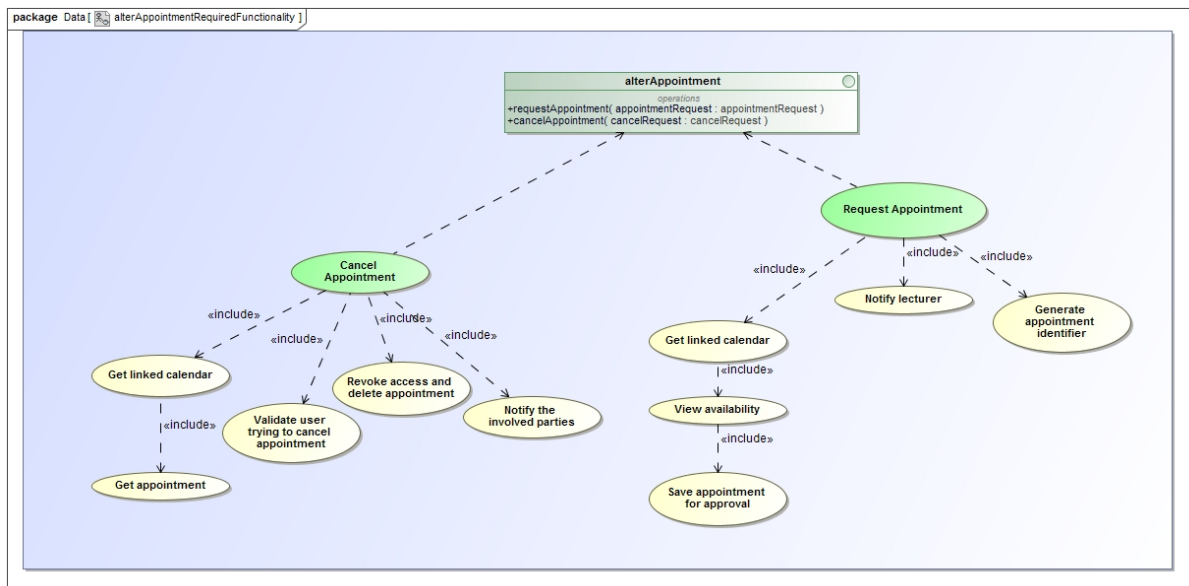
- The appointment will be cancelled.
- Both parties are notified.
- Access that was granted for the appointment is revoked.

## 8.3 Required Functionality

### 8.3.1 Authentication

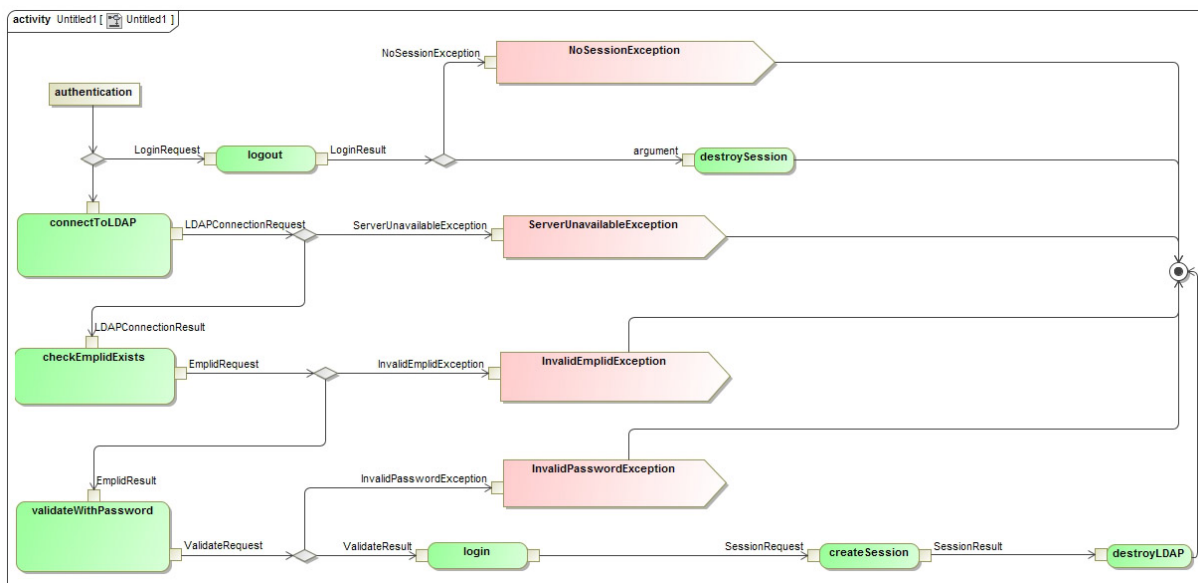


### 8.3.2 Alter Appointment



## 8.4 Process Specification

### 8.4.1 Authentication



## 8.4.2 Alter Appointment

