

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

# Call Calendar



## Requirements Definition Document

Version 2.0

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

**Title:**

Call Calendar

**Course:**

Distributed Software Development

**Document:**

Requirements Definition

**Participants:**

Biljana Stanić

Md. Abdur Razzaque

Hrvoje Pavlović

Marko Vuglec

Marko Veličković

Damian Marušić

**Supervisors:**

Federico Ciccozzi

Ivana Bosnić

**Date:**

**6<sup>th</sup> January 2015**

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

## Revision History

<b>Date</b>	<b>Version</b>	<b>Description</b>	<b>Author</b>
2014-11-12	0.1	Initial Draft	Biljana Stanić
2014-11-12	0.2	Added use cases	Hrvoje Pavlović
2014-11-12	0.3	Added list of requirements	Marko Vuglec
2014-11-12	0.4	Created backlog table	Marko Veličković
2014-11-12	0.5	Division of work	Abdur Razzaque
2014-11-12	0.6	Modified user stories	Marko Veličković
2014-11-12	0.7	Introduction part	Biljana Stanić
2014-11-12	0.8	Short background, related documents	Biljana Stanić
2014-11-12	0.9	High-level design introduction, requirements	Biljana Stanić
2014-11-12	0.10	Extended the content of the requirements table (id, motivation, source)	Biljana Stanić
2014-11-13	0.11	Assigned priority for requirements	Damian Marušić
2014-11-13	0.12	Extended use case diagrams	Damian Marušić
2014-11-14	0.13	Wrote detailed explanation for the use case diagrams	Abdur Razzaque, Damian Marušić
2014-11-14	1.0	Transferred everything to word/pdf and publish	Biljana Stanić
2014-12-22	1.1	Modified User story statuses	Hrvoje Pavlovic
2015-01-04	1.2	Added new functional requirements	Biljana Stanić
2015-01-05	1.3	Added new user stories	Biljana Stanić and Marko Vuglec
2015-01-05	1.4	Modifies use case diagrams	Biljana Stanić
2015-01-05	1.5	Added description for new use cases	Biljana Stanić
2015-01-06	2.0	Transferred everything to word/pdf and publish	Biljana Stanić

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

## Table of Contents

<b>1. Introduction .....</b>	<b>5</b>
<b>Purpose of this document.....</b>	<b>5</b>
<b>Document organization .....</b>	<b>5</b>
<b>Intended Audience.....</b>	<b>5</b>
<b>Scope.....</b>	<b>5</b>
<b>Definitions and acronyms .....</b>	<b>5</b>
<b>2. Background and objectives.....</b>	<b>6</b>
<b>Customer .....</b>	<b>6</b>
<b>Supervisors.....</b>	<b>6</b>
<b>Related documents .....</b>	<b>6</b>
<b>3. High level description of the domain .....</b>	<b>7</b>
<b>Prerequisites for development process .....</b>	<b>7</b>
<b>4. Requirements .....</b>	<b>7</b>
<b>5. Use case diagrams .....</b>	<b>12</b>
<b>6. Division of the work.....</b>	<b>25</b>

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

## 1. Introduction

### Purpose of this document

The purpose of this document is to describe requirements of the system that need to be developed. This will be used initially to ensure that we have common view of the tasks (within the group, and between the group and the customer). It will later be used to guide the development, and it is also one criterion that can be used to measure the quality of the delivered system.

### Document organization

The document is organized as follows:

- Section 1, *Introduction*, describes purpose and audience of this document, scope, definitions and acronyms.
- Section 2, *Short background*, introduces project, customers, supervisors and related documents.
- Section 3, *High level description of the domain*, includes prerequisites for development process
- Section 4, *Requirements*.
- Section 5, *Use case diagrams*.
- Section 6, *Division of work*.

### Intended Audience

The intended audience is:

- Team members;
- Local and remote supervisors;
- Customers.

### Scope

This document provides a detailed list and description of the requirements of the system that will be developed. Requirements are defined with the customers, priorities have been assigned, motivation for their use, etc. Here are presented use case diagrams where are listed functional requirements. Furthermore, non-functional requirements have been identified and explained.

### Definitions and acronyms

In the following table will be present and explained abbreviations that will be used in the document.

Acronym or abbreviation	Definitions
<b>MDH</b>	Mälardalen University, Västerås, Sweden
<b>FER</b>	Faculty of Electrical Engineering and Computing, Zagreb, Croatia
<b>DSD</b>	Distributed Software Development
<b>ES</b>	Embedded Systems
<b>RECO</b>	Research Coordination

**Table 1. Abbreviations with their explanations**

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

## 2. Background and objectives

The division for Research Coordination (RECO) constantly looks out for funding possibilities, helps the researchers with updated information about conditions, budget, consortia, and, also, helps putting the applications together. It is crucial to keep calls and deadlines in mind. RECO and the ES researchers would all benefit from a "Call calendar" containing information about funding bodies, different calls and their respective deadlines.

The goal of this project is to extend current functionalities of existing system at ES. Currently, there are researchers profiles/areas of interest, funding bodies and funding programs. System is divided into back and front office. Back office is used to manage researchers, projects and funding agencies. In front office researchers profiles/areas of interest and funding bodies/programs are shown.

System will be upgraded so it could support adding funding agencies calls, as well as researches' applications to specific calls. It will also match calls with researchers profiles/research interests in order to ensure maximum user satisfaction. Moreover, it will contain information about different calls and their respective topics, deadlines and links to more information. Alongside viewing available calls list, in front office, functionality of sorting and filtering calls will be added. Users will be able to get various generated reports about calls and applications as well.

### Customer

Customers for the "Call Calendar" project are from research center Embedded Systems:

- Malin Rosqvist, Research Coordinator, Web project manager at ES  
*Email:* [malin.rosqvist@mdh.se](mailto:malin.rosqvist@mdh.se)  
*Web page:* [http://www.es.mdh.se/staff/215-Malin\\_Rosqvist](http://www.es.mdh.se/staff/215-Malin_Rosqvist)

- Irfan Šljivo, Web master & Presta shop developer at ES  
*Email:* [irfan.sljivo@mdh.se](mailto:irfan.sljivo@mdh.se)  
*Web page:* [http://www.es.mdh.se/staff/380-Irfan\\_Sljivo](http://www.es.mdh.se/staff/380-Irfan_Sljivo)

### Supervisors

The team has two supervisors, one local and one remote (based on the location of the team) situated in Sweden and Croatia. They are:

- Federico Ciccozzi  
*Email:* [federico.ciccozzi@mdh.se](mailto:federico.ciccozzi@mdh.se)

- Ivana Bosnić  
*Email:* [ivana.bosnic@fer.hr](mailto:ivana.bosnic@fer.hr)

### Related documents

In the document "Project Plan Document.pdf" is presented a detailed description of the project.

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

### 3. High level description of the domain

“Call Calendar” will be an upgrade of the existing system at ES that is powered by a Prestashop database. Researchers’ profiles/areas of interest can be found here and also funding bodies and funding programs. For this project we have to use model-view-controller (MVC) architecture pattern, because Prestashop is based on it. MVC splits application on three logical components and defines interactions between them.

In our “Call Calendar” project, every object that will be created should have a corresponding class where will be specified its properties. To use object, we will create controllers for front- and back-office. In these controllers will be specified information that will be presented in the specific pages in the front/end and forward this information to the smarty templates. Smarty templates are designed to display the information in the desired way. The back-office comes with predefined templates. When adding a page to the front, corresponding redirect page should be added in the root directory Prestashop-skeleton. Notice the naming convention between this file, the corresponding front controller and the included template file.

#### Prerequisites for development process

Preconditions that have to be fulfilled during development process of the project are listed below.

Activities
Finish configuration of mirror server
Secure mirror server
Add appropriate tables and fields (after conversation with customers) in the existing database
Install PHP editor - Notepad++
Install Github support
Install and configure XAMPP

**Table 2. List of activities**

Deliverables
Project Plan v.1
Requirements Definition v.1
Requirements Definition v.1

**Table 3. List of deliverables**

### 4. Requirements

In this section will be presented all requirements that are captured from customers. For each of them, functional and nonfunctional, will be assigned certain priorities. Priorities can be high, medium and low. This means that requirements with that are labeled as “high” priority will be implemented first because they are crucial for those that have “medium” and “low” priority. Moreover, “medium” priority requirements are those that are listed as desirable to be implemented. While, “low” priority requirements can be implemented but they are not so important for the normal functioning of the system (some additional features).

In the following table are listed all requirements, with corresponding ids denoting whether requirement is functional (“FR” prefix) or non-functional (“NF” prefix) or a certain document (“D” prefix), priorities, names, description, motivation and source.

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

ID	Functional	Name	Description	Motivation	Priority	Source
FR-01	Yes	Add new call	Add the call	1	High	Malin and Irfan
FR-02	Yes	View all calls	View all past/active calls	2, 34	High	Malin and Irfan
FR-03	Yes	Filtering calls by agencies	Filter calls by funding agency's name	3, 35	Medium	Malin and Irfan
FR-04	Yes	Filtering calls by status	Filtering calls by status (past/active)	3, 35	Medium	Malin and Irfan
FR-05	Yes	View specific call	Detail view of specific call	4, 36	High	Malin and Irfan
FR-06	Yes	Send notification to researchers	Send email notification to all researchers who are interested in call	5, 37	Medium	Malin and Irfan
FR-07	Yes	Adding application form to the call	Researcher's application is added to the specific call	6	High	Malin and Irfan
FR-08	Yes	Storing application's forms	All created applications are saved and stored	6	High	Malin and Irfan
FR-09	Yes	Viewing and generating reports about the call	Generate report with all important information about the specific call	7	Low	Malin and Irfan
FR-10	Yes	Generating reports about all applications for the call	Generate report about all applications for the specific call	8	Low	Malin and Irfan
FR-11	Yes	Custom filters and possibility to export results to excel	Filter calls and export retrieved data to excel file	9	Low	Malin and Irfan
FR-12	Yes	Custom filter - option to choose which fields to return	Filter calls using custom fields	10	Low	Malin and Irfan
FR-13	Yes	Create a project	The project should be created once the application is labeled as "granted"	11	High	Malin and Irfan
FR-14	Yes	Create a news feed	For every granted application should be created a news feed on the web page	12	Medium	Malin and Irfan
FR-15	Yes	Add new call type	Add the call type that will be attached to the call	13	High	Malin and Irfan
FR-16	Yes	Add new call status	Add the call status that will be attached to the call	14	High	Malin and Irfan
FR-17	Yes	Add attachment to the call	Add the attachment (documentation) to the call	15	High	Malin and Irfan
FR-18	Yes	Add deadline for the call	Add the deadline (internal/external) for the specific call	16	High	Malin and Irfan
FR-19	Yes	View all attachments for the call	View all added attachments for the specific call	17, 39	Medium	Malin and Irfan
FR-20	Yes	Download attachment from the call	Download a specific attachment from the chosen call	18, 38	Medium	Malin and Irfan



Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

FR-21	Yes	Add new application status	Add the application status that will be attached to the application	19	High	Malin and Irfan
FR-22	Yes	Add attachment to the application	Add the attachment (documentation) to the application	20	High	Malin and Irfan
FR-23	Yes	Add attachment of the application to the deadline	Add the attachment (documentation) to the deadline	21	High	Malin and Irfan
FR-24	Yes	Add funding program	Add funding program	22	Medium	Malin and Irfan
FR-25	Yes	Remove existing call	Remove existing call	23	High	Malin and Irfan
FR-26	Yes	Edit existing call	Edit data about the existing call	23	High	Malin and Irfan
FR-27	Yes	Deny a certain application	Deny a certain application that is not valuable	24	Medium	Malin and Irfan
FR-28	Yes	Edit existing application	Edit data about the existing application	25	High	Malin and Irfan
FR-29	Yes	Remove existing application	Remove existing application	25	High	Malin and Irfan
FR-30	Yes	Edit existing call type	Edit data about the existing call type	26	High	Malin and Irfan
FR-31	Yes	Remove existing call type	Remove existing call type	26	High	Malin and Irfan
FR-32	Yes	Edit existing call status	Edit data about the existing call status	27	High	Malin and Irfan
FR-33	Yes	Remove existing call status	Remove existing call status	27	High	Malin and Irfan
FR-34	Yes	Edit existing application status	Edit data about the existing application status	28	High	Malin and Irfan
FR-35	Yes	Remove existing application status	Remove existing application status	28	High	Malin and Irfan
FR-36	Yes	Edit existing attachment of the call	Edit data about the existing attachment of the call	29	High	Malin and Irfan
FR-37	Yes	Remove existing attachment of the call	Remove existing attachment of the call	29	High	Malin and Irfan
FR-38	Yes	Edit existing deadline for the call	Edit data about the existing deadline for the call	30	High	Malin and Irfan
FR-39	Yes	Remove existing deadline for the call	Remove existing deadline for the call	30	High	Malin and Irfan
FR-40	Yes	Edit existing attachment of the application	Edit data about the existing attachment of the application	31	High	Malin and Irfan
FR-41	Yes	Remove existing attachment of the application	Remove existing attachment of the application	31	High	Malin and Irfan
FR-42	Yes	Edit existing attachment of the application to the deadline	Edit data about the existing attachment of the application to the deadline	32	Medium	Malin and Irfan
FR-	Yes	Remove existing	Remove existing	32	Medium	Malin and

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

43		attachment of the application to the deadline	attachment of the application to the deadline			Irfan
FR-44	Yes	Remove existing funding program	Remove existing funding program	33	High	Malin and Irfan
FR-45	Yes	Edit existing funding program	Edit data about the existing funding program	33	Medium	Malin and Irfan
NF-01	No	Performance (SQL optimization)	Faster query at runtime	40	High	Project group
NF-02	No	User experience	Quick response of the given function	40	Medium	Project group
D-01	No	MoM	Minutes of meeting, document that contains information about the meetings that were held in the past week	41	High	Supervisors
D-02	No	Weekly reports	Document that contains information about team activities of the past week	41	High	Supervisors
D-03	No	Project plan	Document that contains details about the team, project vision and goals to be achieved	41	High	Supervisors
D-04	No	Requirements definition	Document that contains what is supposed to develop	41	High	Supervisors
D-05	No	Test report	The test report captures and summarizes the test results	41	High	Supervisors

**Table 4. Requirements definition**

Our product backlog consists of functional requirements, non-functional requirements and documents. Each user story has its own ID. The motivation for the specific requirement is shown in the following table. Admin (RECO) uses back-office of the application and researcher front-office.

ID	User story	Priority	Status
1	As an admin (RECO), I want to add call for a specific topic, so that researchers can apply for funding	High	Finished
2	As the admin (RECO), I want to see all past/active calls in the back-office, so could have a complete insight for all calls	High	Finished
3	As the admin (RECO), I want to filter all calls in the back-office by funding agency or status, so I could see filtered list of calls	Medium	Finished
4	As the admin (RECO), I want to see a specific call, so I could see all its fields	High	Finished
5	As the admin (RECO), I want to send notification for calls offered by funding agencies, so researchers can choose the interesting ones	Medium	Not started
6	As the admin (RECO), I want to add and store application form to the call for a specific topic, so that I have completed process of applying for the call	High	Finished
7	As the admin (RECO), I want to generate reports about the call, so that I can have an archive about them	Low	In process
8	As the admin (RECO), I want to generate a report of all applications for a	Low	Not started

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

	specific call, so I can have a file that can be stored		
9	As the admin (RECO), I want to get a certain information about the call and export it, so I can have it in a form of a file	Low	In process
10	As the admin (RECO), I want to specify which information about calls I want to get as a result, so I do not get unnecessary information	Low	Not started
11	As the admin (RECO), when I approve the application, I want a project to be created automatically, so I can have the project related to the application	High	Finished
12	As the admin (RECO), when I approve the application, I want a news feed to be created automatically, so I can have that information displayed publicly on the web page	Medium	Not started
13	As the admin (RECO), I want to add call type, so that it can be linked with the call	High	Finished
14	As the admin (RECO), I want to add call status, so that it can be linked with the call	High	Finished
15	As the admin (RECO), I want to add attachment to the call, so I can have documents that will support description of the call	High	Finished
16	As the admin (RECO), I want to add deadline for the call, so that researchers can have insight until when they are able to submit application for the call	High	Finished
17	As the admin (RECO), I want to view attachments for the call, so I can see which files are stored	Medium	Finished
18	As the admin (RECO), I want to download attachment for the call, so I can view its content	Medium	Not started
19	As the admin (RECO), I want to add application status, so that it can be linked with the application	High	Finished
20	As the admin (RECO), I want to add attachment to the application, so that application process can be complete	High	Not started
21	As the admin (RECO), I want to add attachment of the application to the deadline, so that application process can be complete	High	Not started
22	As the admin (RECO), I want to add funding program, so that it can be linked with the call	Medium	Not started
23	As the admin (RECO), I want to manage (edit/remove) existing call(s), so I could change/remove their content when it is needed	High	Finished
24	As the admin (RECO), I can deny certain application, when I determine that it is not valuable	Medium	Finished
25	As the admin (RECO), I want to manage (edit/remove) existing application(s), so I could change/remove their content when it is needed	High	Finished
26	As the admin (RECO), I want to manage (edit/remove) existing call type(s), so I could change/remove their content when it is needed	High	Finished
27	As the admin (RECO), I want to manage (edit/remove) existing call status(s), so I could change/remove their content when it is needed	High	Finished
28	As the admin (RECO), I want to manage (edit/remove) existing application status(s), so I could change/remove their content when it is needed	High	Finished
29	As the admin (RECO), I want to manage (edit/remove) existing attachment of the call, so I could change/remove their content when it is needed	High	Finished
30	As the admin (RECO), I want to manage (edit/remove) existing deadline for the call, so I could change/remove their content when it is needed	High	Finished
31	As the admin (RECO), I want to manage (edit/remove) existing attachment of the application, so I could change/remove their content when it is needed	High	Finished
32	As the admin (RECO), I want to manage (edit/remove) existing	High	Finished

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

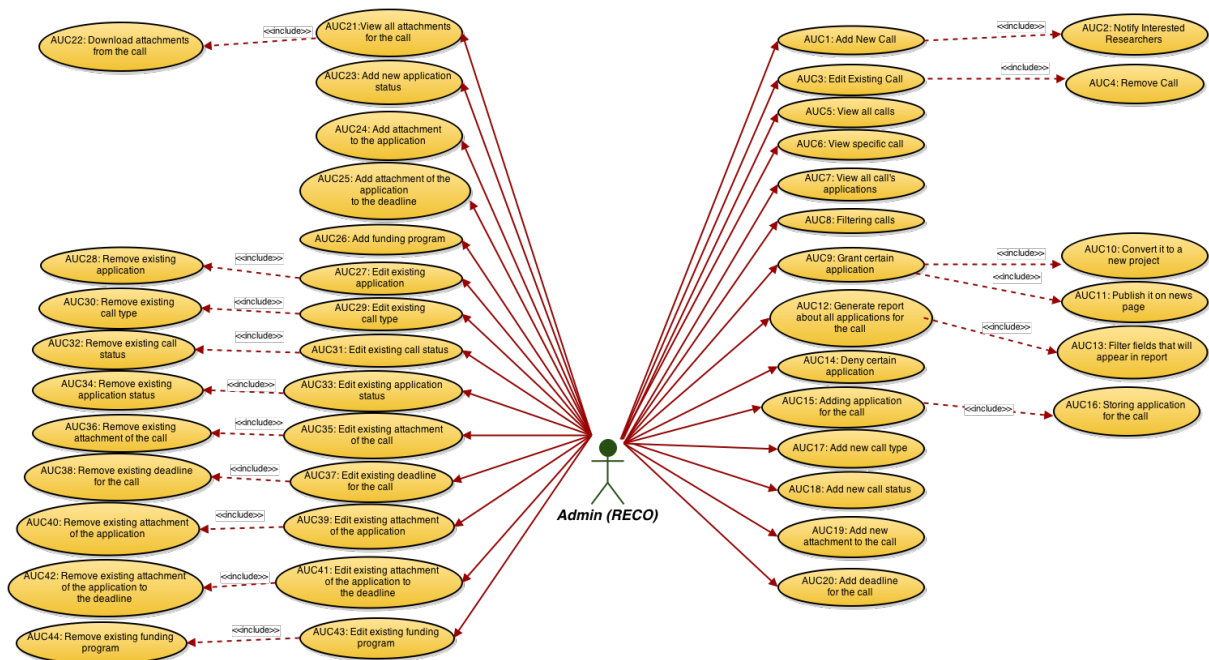
	attachment of the application to the deadline, so I could change/remove their content when it is needed		
33	As the admin (RECO), I want to manage (edit/remove) existing funding program, so I could change/remove their content when it is needed	High	Finished
34	As a researcher, I want to see all active calls, so that i can decide if I am interested in it	High	Finished
35	As the researcher, I want to filter all calls by funding agency or status, so I could see filtered list of calls	Medium	Finished
36	As the researcher, I want to see details about each call, so that I can apply	High	Finished
37	As the researcher, I want to get notification for calls offered by funding agencies, so I can choose the interesting ones	Medium	Not started
38	As the researcher, I want to download attachment for the call, so I can view its content	Medium	Finished
39	As the researcher, I want to view attachments for the call, so I can see which files are stored	Medium	Finished
40	As the user, I want a fast page load, so I can have a lower response time	Medium	Finished
41	As a supervisor, I want to have insight into the progress and development of the project, so I can advise and improve possible difficulties	High	In process

**Table 5. Motivation for the specific requirement**

## 5. Use case diagrams

There are two types of users on the system : admin (RECO) and researchers.

In the following figure will be shown all use cases that are captured for the admin (RECO).



**Figure 1. Use case diagram for the admin (RECO)**

Other actor is researcher that has more limited functions on the system comparing to the admin (RECO). In the following tables are shown detailed explanations for each and every use case.

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

Use case name:	<b>AUC1: Add New Call</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to make a new call</li> <li>2. Admin (RECO) fills in form</li> <li>3. Admin (RECO) submits filled in form</li> <li>4. New call is made and stored</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	New call is stored
Special requirements:	

**Table 6. Add new call**

Use case name:	<b>AUC2: Notify Interested Researchers</b>
Participating actor(s):	Admin (RECO) System
Preconditions:	Call is made
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) makes a call</li> <li>2. Click broadcast</li> <li>3. System identifies all the researchers that may be interested in call</li> <li>4. System notifies all the researchers that may be interested in call</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2a. No interested researcher found <ol style="list-style-type: none"> <li>1. No notifications sent</li> </ol> </li> </ol>
Postcondition:	Potentially interested researchers are notified about certain call
Special requirements:	

**Table 7. Notify interested researchers**

Use case name:	<b>AUC3: Edit Existing Call</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) chooses to edit previously made call</li> <li>2. Admin (RECO) makes changes</li> <li>3. Admin (RECO) submits changes</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	A message about edited call is displayed
Special requirements:	

**Table 8. Edit existing call**

Use case name:	<b>AUC4: Remove Call</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to remove existing call</li> <li>2. Admin (RECO) selects a certain call and presses Remove button</li> </ol>

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

	3. The call has been deleted
Alternative Flows:	
Postcondition:	Researchers will be notified
Special requirements:	Call is made

**Table 9. Remove call**

Use case name:	<b>AUC5: View all calls</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call
Basic Flow:	1. Admin (RECO) views list with all call
Alternative Flows:	
Postcondition:	Admin (RECO) has insight in all calls
Special requirements:	

**Table 10. View all calls**

Use case name:	<b>AUC6: View specific call</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call
Basic Flow:	1. Admin (RECO) click on the specific call 2. Data about the call has been displayed
Alternative Flows:	
Postcondition:	Admin (RECO) has insight in specific call
Special requirements:	

**Table 11. View specific call**

Use case name:	<b>AUC7: View all call's applications</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call
Basic Flow:	1. Admin (RECO) views all applications for a specific call
Alternative Flows:	
Postcondition:	Admin (RECO) has insight in all applications made for certain call
Special requirements:	Researcher applied for a call

**Table 12. View all call's applications**

Use case name:	<b>AUC8: Filtering calls</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call
Basic Flow:	1. Funding Agency chooses filtering options 2. List with filtered calls has been displayed
Alternative Flows:	
Postcondition:	Admin (RECO) has insight in all filtered calls
Special requirements:	

**Table 13. Filtering calls**

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

Use case name:	<b>AUC9: Grant certain application</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created received application(s) to previously made call
Basic Flow:	1. Admin (RECO) picks application to grant 2. Admin (RECO) grants application for a certain call
Alternative Flows:	
Postcondition:	Application is granted and will become a new project
Special requirements:	

**Table 14. Grant certain application**

Use case name:	<b>AUC10: Convert granted application to a new project</b>
Participating actor(s):	Admin (RECO) System
Preconditions:	There is granted application
Basic Flow:	1. Granted application is automatically converted to new project and stored
Alternative Flows:	
Postcondition:	New project in database
Special requirements:	

**Table 15. Convert granted application to a new project**

Use case name:	<b>AUC11: Publish granted application on news page</b>
Participating actor(s):	Admin (RECO) System
Preconditions:	There is granted application
Basic Flow:	1. Granted application automatically generates news about it on the news page
Alternative Flows:	
Postcondition:	News on the news page
Special requirements:	

**Table 16. Publish granted application on news page**

Use case name:	<b>AUC12: Generate report about all applications for the call</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call
Basic Flow:	1. Admin (RECO) wants generated report about all calls from application 2. Report has been generated
Alternative Flows:	
Postcondition:	Generated report for Admin (RECO)
Special requirements:	

**Table 17. Generate report about all applications for the call**

Use case name:	<b>AUC13: Filter fields that will appear in report</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call
Basic Flow:	1. Admin (RECO) wants generated report about all calls from application

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

	2. Admin (RECO) select which fields from application it wants 3. Report has been generated
Alternative Flows:	
Postcondition:	Generated report with filtered fields is created
Special requirements:	

**Table 18. Filter fields that will appear in report**

Use case name:	<b>AUC14: Deny certain application</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call
Basic Flow:	1. Admin (RECO) selects application that it wants to deny 2. Admin (RECO) denies selected application
Alternative Flows:	
Postcondition:	Application is marked as denied
Special requirements:	

**Table 19. Deny certain application**

Use case name:	<b>AUC15 and AUC16 : Adding application for the call and Storing application for the call</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call
Basic Flow:	1. Admin (RECO) decides to make a new application 2. Admin (RECO) fills in form 3. Admin (RECO) submits filled in form 4. New application is made and stored
Alternative Flows:	2. Alert about incorrectly filled fields is shown 3a. Requested fields are incorrectly filled in
Postcondition:	New application is stored
Special requirements:	

**Table 20. Adding application for the call and Storing application for the call**

Use case name:	<b>AUC17: Add new call type</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office
Basic Flow:	1. Admin (RECO) decides to make a new call type 2. Admin (RECO) fills in form 3. Admin (RECO) submits filled in form 4. New call type is made and stored
Alternative Flows:	2. Alert about incorrectly filled fields is shown 3a. Requested fields are incorrectly filled in
Postcondition:	New call type is stored
Special requirements:	

**Table 21. Add new call type**

Use case name:	<b>AUC18: Add new call status</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office
Basic Flow:	1. Admin (RECO) decides to make a new call status



Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

	<ol style="list-style-type: none"> <li>2. Admin (RECO) fills in form</li> <li>3. Admin (RECO) submits filled in form</li> <li>4. New call status is made and stored</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	New call status is stored
Special requirements:	

**Table 22. Add new call status**

Use case name:	<b>AUC19: Add new attachment to the call</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to add new attachment to the call</li> <li>2. Admin (RECO) fills in form</li> <li>3. Admin (RECO) submits filled in form</li> <li>4. New attachment to the call is added and stored</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	New attachment to the call is stored
Special requirements:	

**Table 23. Add new attachment to the call**

Use case name:	<b>AUC20: Add deadline for the call</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to add deadline for the call</li> <li>2. Admin (RECO) fills in form</li> <li>3. Admin (RECO) submits filled in form</li> <li>4. New deadline for the call is added and stored</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	New deadline for the call is stored
Special requirements:	

**Table 24. Add deadline for the call**

Use case name:	<b>AUC21: View all attachments for the call</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) selects and chooses a specific call</li> <li>2. Attachments have been listed</li> </ol>
Alternative Flows:	
Postcondition:	Admin (RECO) has insight in all attachments for the certain call
Special requirements:	

**Table 25. View all attachments for the call**

Use case name:	<b>AUC22: Download attachments from the call</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) selects and chooses a specific call</li> <li>2. Admin (RECO) downloads attachments for the selected call</li> <li>3. Attachments have been downloaded</li> </ol>
Alternative Flows:	
Postcondition:	Attachments have been downloaded
Special requirements:	

**Table 26. Download attachments from the call**

Use case name:	<b>AUC23: Add new application status</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to make a new application status</li> <li>2. Admin (RECO) fills in form</li> <li>3. Admin (RECO) submits filled in form</li> <li>4. New application status is made and stored</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	New application status is stored
Special requirements:	

**Table 27. Add new application status**

Use case name:	<b>AUC24: Add attachment to the application</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created application
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to add new attachment to the application</li> <li>2. Admin (RECO) fills in form</li> <li>3. Admin (RECO) submits filled in form</li> <li>4. New attachment to the application is added and stored</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	New attachment to the application is stored
Special requirements:	

**Table 28. Add attachment to the application**

Use case name:	<b>AUC25: Add attachment of the application to the deadline</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created application
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to add attachment of the application to the deadline</li> <li>2. Admin (RECO) fills in form</li> <li>3. Admin (RECO) submits filled in form</li> <li>4. New attachment of the application to the deadline is added and stored</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	New attachment of the application to the deadline is stored
Special requirements:	

**Table 29. Add attachment of the application to the deadline**

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

Use case name:	<b>AUC26: Add new funding program</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to add a new funding program</li> <li>2. Admin (RECO) fills in form</li> <li>3. Admin (RECO) submits filled in form</li> <li>4. New funding program is made and stored</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	New funding program is stored
Special requirements:	

**Table 30. Add new funding program**

Use case name:	<b>AUC27: Edit existing application</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created application
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) chooses to edit previously made application</li> <li>2. Admin (RECO) makes changes</li> <li>3. Admin (RECO) submits changes</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	A message about edited application is displayed
Special requirements:	

**Table 31. Edit existing application**

Use case name:	<b>AUC28: Remove existing application</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created application
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to remove existing application</li> <li>2. Admin (RECO) selects a certain application and presses Remove button</li> <li>3. The application has been deleted</li> </ol>
Alternative Flows:	
Postcondition:	
Special requirements:	Application is made

**Table 32. Remove existing application**

Use case name:	<b>AUC29: Edit existing call type</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call type
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) chooses to edit previously made call type</li> <li>2. Admin (RECO) makes changes</li> <li>3. Admin (RECO) submits changes</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	A message about edited call type is displayed
Special requirements:	

**Table 33. Edit existing call type**

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

Use case name:	<b>AUC30: Remove existing call type</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call type
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to remove existing call type</li> <li>2. Admin (RECO) selects a certain call type and presses Remove button</li> <li>3. The call has been deleted</li> </ol>
Alternative Flows:	
Postcondition:	
Special requirements:	Call type is made

**Table 34. Remove existing call type**

Use case name:	<b>AUC31: Edit existing call status</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call status
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) chooses to edit previously made call status</li> <li>2. Admin (RECO) makes changes</li> <li>3. Admin (RECO) submits changes</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	A message about edited call status is displayed
Special requirements:	

**Table 35. Edit existing call status**

Use case name:	<b>AUC32: Remove existing call status</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created call status
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to remove existing call status</li> <li>2. Admin (RECO) selects a certain call status and presses Remove button</li> <li>3. The call status has been deleted</li> </ol>
Alternative Flows:	
Postcondition:	
Special requirements:	Call status is made

**Table 36. Remove existing call status**

Use case name:	<b>AUC33: Edit existing application status</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created application status
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) chooses to edit previously made application status</li> <li>2. Admin (RECO) makes changes</li> <li>3. Admin (RECO) submits changes</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	A message about edited application status is displayed
Special requirements:	

**Table 37. Edit existing application status**

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

Use case name:	<b>AUC34: Remove existing application status</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created application status
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to remove existing application status</li> <li>2. Admin (RECO) selects a certain application status and presses Remove button</li> <li>3. The application status has been deleted</li> </ol>
Alternative Flows:	
Postcondition:	
Special requirements:	Application status is made

**Table 38. Remove existing application status**

Use case name:	<b>AUC35: Edit existing attachment of the call</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created attachment of the call
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) chooses to edit previously added attachment of the call</li> <li>2. Admin (RECO) makes changes</li> <li>3. Admin (RECO) submits changes</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	A message about edited attachment of the call is displayed
Special requirements:	

**Table 39. Edit existing attachment of the call**

Use case name:	<b>AUC36: Remove existing attachment of the call</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created attachment of the call
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to remove existing attachment of the call</li> <li>2. Admin (RECO) selects a certain attachment of the call and presses Remove button</li> <li>3. The attachment of the call has been deleted</li> </ol>
Alternative Flows:	
Postcondition:	
Special requirements:	Attachment of the call is added

**Table 40. Remove existing attachment of the call**

Use case name:	<b>AUC37: Edit existing deadline for the call</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created deadline for the call
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) chooses to edit previously added deadline for the call</li> <li>2. Admin (RECO) makes changes</li> <li>3. Admin (RECO) submits changes</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	A message about edited deadline for the call is displayed
Special requirements:	

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

**Table 41. Edit existing deadline for the call**

Use case name:	<b>AUC38: Remove existing deadline for the call</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created deadline for the call
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to remove existing deadline for the call</li> <li>2. Admin (RECO) selects a certain deadline for the call and presses Remove button</li> <li>3. The deadline for the call has been deleted</li> </ol>
Alternative Flows:	
Postcondition:	Researchers will be notified
Special requirements:	Deadline for the call is added

**Table 42. Remove exiting deadline for the call**

Use case name:	<b>AUC39: Edit existing attachment of the application</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has added attachment of the application
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) chooses to edit previously added attachment of the application</li> <li>2. Admin (RECO) makes changes</li> <li>3. Admin (RECO) submits changes</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2. Alert about incorrectly filled fields is shown</li> <li>3a. Requested fields are incorrectly filled in</li> </ol>
Postcondition:	A message about edited attachment of the application is displayed
Special requirements:	

**Table 43. Edit existing attachment of the application**

Use case name:	<b>AUC40: Remove existing attachment of the application</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has added attachment of the application
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) decides to remove existing attachment of the application</li> <li>2. Admin (RECO) selects a certain attachment of the application and presses Remove button</li> <li>3. The attachment of the application has been deleted</li> </ol>
Alternative Flows:	
Postcondition:	
Special requirements:	Attachment of the application is added

**Table 44. Remove existing attachment of the application**

Use case name:	<b>AUC41: Edit existing attachment of the application to the deadline</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has added attachment of the application to the deadline
Basic Flow:	<ol style="list-style-type: none"> <li>1. Admin (RECO) chooses to edit previously added attachment of the application to the deadline</li> <li>2. Admin (RECO) makes changes</li> <li>3. Admin (RECO) submits changes</li> </ol>

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

Alternative Flows:	2. Alert about incorrectly filled fields is shown 3a. Requested fields are incorrectly filled in
Postcondition:	A message about edited attachment of the application to the deadline is displayed
Special requirements:	

**Table 45. Edit existing attachment of the application to the deadline**

Use case name:	<b>AUC42: Remove existing attachment of the application to the deadline</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has added attachment of the application to the deadline
Basic Flow:	1. Admin (RECO) decides to remove existing attachment of the application to the deadline 2. Admin (RECO) selects a certain attachment of the application to the deadline and presses Remove button 3. The attachment of the application to the deadline has been deleted
Alternative Flows:	
Postcondition:	
Special requirements:	Attachment of the application to the deadline is added

**Table 46. Remove existing attachment of the application to the deadline**

Use case name:	<b>AUC43: Edit existing funding program</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created funding program
Basic Flow:	1. Admin (RECO) chooses to edit previously made funding program 2. Admin (RECO) makes changes 3. Admin (RECO) submits changes
Alternative Flows:	2. Alert about incorrectly filled fields is shown 3a. Requested fields are incorrectly filled in
Postcondition:	A message about edited funding program is displayed
Special requirements:	

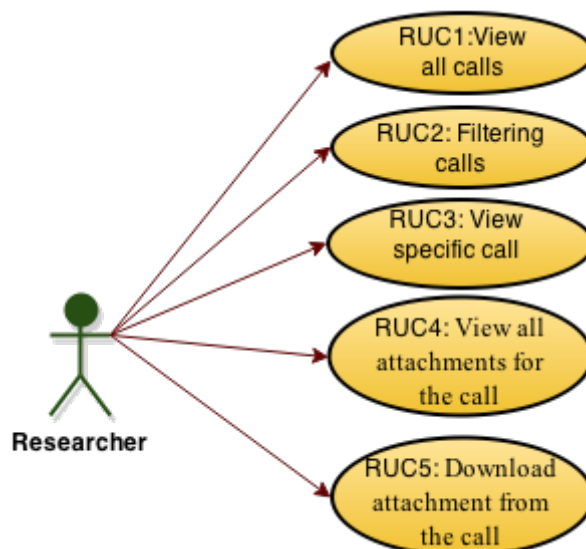
**Table 47. Edit existing funding program**

Use case name:	<b>AUC44: Remove existing funding program</b>
Participating actor(s):	Admin (RECO)
Preconditions:	Admin (RECO) has access to back office Admin (RECO) has created funding program
Basic Flow:	4. Admin (RECO) decides to remove existing funding program 5. Admin (RECO) selects a certain funding program and presses Remove button 6. The funding program has been deleted
Alternative Flows:	
Postcondition:	
Special requirements:	Funding program is made

**Table 48. Remove existing funding program**

In the following figure is shown use case diagram for the researcher as the user of the system. As it can be seen, researcher has limited number of functionalities that can use.

Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06



**Figure 2. Use case diagram for the researcher**

It is important to mention that both funding agency and researcher are extending the role of the user.

Use case name:	<b>RUC1: View all calls</b>
Participating actor(s):	Researcher
Preconditions:	
Basic Flow:	<ol style="list-style-type: none"> <li>1. Researcher opens all calls</li> <li>2. List of all calls is shown</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>2a. Database connection is not available</li> <li>2b. Appropriate message is generated</li> </ol>
Postcondition:	Listed all calls
Special requirements:	

**Table 49. View all calls - researcher**

Use case name:	<b>RUC2: Filtering calls</b>
Participating actor(s):	Researcher
Preconditions:	
Basic Flow:	<ol style="list-style-type: none"> <li>1. Researcher wants to get filtered calls</li> <li>2. Researcher chooses filtering parameters</li> <li>3. Researcher gets list of filtered calls</li> </ol>
Alternative Flows:	
Postcondition:	Filtered calls are shown
Special requirements:	

**Table 50. Filter calls - researcher**

Use case name:	<b>RUC3: View specific call</b>
Participating actor(s):	Researcher
Preconditions:	There is at least one call
Basic Flow:	<ol style="list-style-type: none"> <li>1. Researcher chooses a specific call</li> <li>2. Details about that call is displayed</li> </ol>
Alternative Flows:	



Call Calendar	Version: 2.0
Requirements Definition Document	Date: 2015-01-06

Postcondition:	Details about specific call are shown
Special requirements:	

**Table 51. View specific call - researcher**

Use case name:	<b>RUC4: View all attachments for the call</b>
Participating actor(s):	Researcher
Preconditions:	There is at least one call
Basic Flow:	<ol style="list-style-type: none"> <li>3. Researcher chooses a specific call</li> <li>4. Researcher chooses an attachment field</li> <li>5. List with all attachments for the call is displayed</li> </ol>
Alternative Flows:	
Postcondition:	List with all attachments for the call is displayed
Special requirements:	

**Table 52. View all attachments for the call - researcher**

Use case name:	<b>RUC5: Download attachment from the call</b>
Participating actor(s):	Researcher
Preconditions:	There is at least one call
Basic Flow:	<ol style="list-style-type: none"> <li>6. Researcher chooses a specific call</li> <li>7. Researcher chooses an attachment field</li> <li>8. Researcher downloads the attachment from the call</li> </ol>
Alternative Flows:	
Postcondition:	The attachment has been downloaded
Special requirements:	

**Table 53. Download attachment from the call - researcher**

## 6. Division of the work

In order to better handle the implementation process, we have agreed to split the work into several parts. Moreover, for every specific part is assigned a small group that consists of two-three team members. This division is based on the experience and interests of the members for the specific task.

Role	Team Member(s)
Database Manager	Marko Vuglec, Hrvoje Pavlović
Front-End Manager	Biljana Stanić, Marko Veličković
Back-End Manager	Damian Marušić, Abdur Razzaque
Testing Manager	Marko Vuglec, Hrvoje Pavlović

**Table 54. Division of work**