UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Denkleiers • Leading Minds • Dikgopolo tša Dihlalefi

Software Requirements Specification and Technology Neutral Process Design

Members:
Matthew Botha 14214742
Gershom Maluleke 13229908
Nathan Dunkley 14145759
Aiden Malan
Sello Thosago
Nsovo Baloyi 12163262
Matheu Botha 14284104

Team India
March 4, 2016
v0.1

1   Introduction

The purpose of this document is to outline the core functionality of the new computer science publication system in a technology neutral design specification. This design is not necessarily a depiction of what the final system will look like. During the development of the system it is possible that new functionalities are added, or old ones removed. This is merely the first iteration of the development cycle. Which can be revised in the future. As well as iterativly enhanced and detailed as the need arises.

2   Vision

This projects aim is to provide a system to facilitate the administration of postgraduate publications, including but not limited to journal articles and conference papers. It should be able to store information on Users, Authors and Publications as well as employ a authentication system to ensure only specified parties within the system have access to operations on the stored information. The purpose of keeping track of the publications is, in part, to keep track of the DoE and UP weighted units earned by Users and/or authors. The reason for this is that these users/authors are required to publish a certain amount of units each year in order to determine pay, research funding, and maintaining their research position.

3   Background

Currently the publications are kept track of with a complicated excel spreadsheet. The purpose of the spreadsheet is to keep track of the status of publications in order to generate a report detailing the units earned by a researcher. There are many drawbacks to this system. It is tedious to maintain the spreedsheet. Data is spread over multiple sheets. The larger the database gets the slower it preforms. The information is displayed in a raw textual format which makes it a daunting task to sit through and analyse. It is difficult to add new functionality and change or enhance existing functionality. When pulling in data from multiple sheets it becomes difficult to debug the Lookups.

4   Architecture

4.1   Access Channels

This service should be accessable via web based clients. i.e. web based clients,mobile browsers. As well as an Android application.

4.2   Quality Requirements

- Reliability - The reports generated from the data provided to the system should be

- Scalability - It is estimated that there will be approximatly one hundred users. The system should be able to handle each user being active at the same time even though it is unlikly.

- Cost - Generating a report for all data in the system will be the worst case scenario in this system and thus have a high cost. The system should however be able to produce subsets of the report at a relativly low cost. For example a report for all researchers under a specific research group.

- Security - There should be strict access right applied to each of the subsystems and their components. This means that if a user, by definition of the buisness rules, is not allowed to access a certain functionality, then they will be denied access.

- Auditability - Actions performed on the system should be logged in a manner that would allow one to trace back the history of actions performed on the system.

4.3   Intergration Requirements

The system will operate in a standalone fashion. In the future, however, it may require intergration with google calender.

4.4   Architecture Constraints

Client has not provided any architecture constraints.

5   Functional Requirements and Application Design

¡¡¡¡¡¡ HEAD

5.1   Use case Prioritisation

5.1.1   User Subsystem

Register User: Critical
Update User: Important
Login User: Critical
Logout User: Critical
Add to Publication: Important

5.1.2 Publication Subsystem

Create Publication: Important
View Publication: Nice-to-Have
Update Publication: Important
Remove Publication: Important
Add Author: Important
Remove Author: Important

5.1.3 Conference Subsystem

Add Conference: important
Update Conference: important
Delete Conference: important

5.1.4 Journal Subsystem

Add Journal: important
Update Journal: important
Delete Journal: important

5.1.5 Thesis Subsystem

Add Thesis: important
Update Thesis: important
Delete Thesis: important

5.1.6 Report Subsystem

Log In to System: critical
View Available Reports: important
Select Filters to Apply to Reports: important
Select Output Type of Report: nice-to-have
Show Report: important

5.2 Use case/Services contracts

5.2.1 User Subsystem

Pre-conditions:

Register user: The user can not be registered to the system already and the user registering a new user has to be admin.

Update user: The user has to be logged in first before they can update their profile and they can only update their own profile.

Login user: The user has to be registered to the system first.

Logout user: The user has to be logged in first.

Add to publication: The user has to be logged in first and the publication has to exist in the system

Post-conditions:

Register user: The user is now registered and added to the database.

Update user: The user's profile is updated and the changes are saved to the database.

Login user: The user is logged into the system.

Logout user: The user is logged out of the system.

Add to publication: The user's name is now added to the author's of the selected publication.

5.2.2   Publication Subsystem

Use Cases:
    3.1 Create Publication:
        Pre-conditions: The user must be logged into the system
        Post-conditions: The publication is created
    3.2 View Publication:
        Pre-conditions: The user must be logged onto the system and the publication must be created before hand
        Post-conditions: The user will be able to view and edit(Add authors etc.) to the publication.
    3.3 Update Publication:
        Pre-conditions: The user must be logged onto the system and the publication must be created before hand
        Post-conditions: The publication is updated
    3.4 Remove Publication:
        Pre-conditions: The user must be logged onto the system, the publication must be created before hand and the user must confirm the removal of the publication
        Post-conditions: The publication will be removed
    3.5 Add Author:
        Pre-conditions: The user must be logged onto the system and the publication must be created before hand
        Post-conditions:A new author is added to the publication

3.6 Remove Author:

   Pre-conditions: The user must be logged onto the system, the publication must be created before hand and the user must confirm the removal of the selected author

   Post-conditions: The author is removed from the publication

5.2.3   Conference Subsystem

Add Conference:

   Pre-conditions: The user must be logged in.

   Post-conditions: The conference is added.

Update Conference:

   Pre-conditions: The user must be logged in and a the conference must be added prior to this use case.

   Post-conditions: The conference is updated.

Delete Conference:

   Pre-conditions: The user must be logged in and a the conference must be added prior to this use case.

   Post-conditions: The conference is deleted.

5.2.4   Journal Subsystem

Add Journal:

   Pre-conditions: The user must be logged in.

   Post-conditions: The journal is added.

Update Journal:

   Pre-conditions: The user must be logged in and a the journal must be added prior to this use case.

   Post-conditions: The journal is updated.

Delete Journal:

   Pre-conditions: The user must be logged in and a the journal must be added prior to this use case.

   Post-conditions: The journal is deleted.

5.2.5   Thesis Subsystem

Add Thesis:

   Pre-conditions: The user must be logged in.

   Post-conditions: The thesis is added.

Update Thesis:

   Pre-conditions: The user must be logged in and a the thesis must be added prior to this use case.

   Post-conditions: The thesis is updated.

Delete Thesis:

    Pre-conditions: The user must be logged in and a the thesis must be added prior to this use case.

    Post-conditions: The thesis is deleted.

5.2.6 Conference, Journal and Thesis Subsystem Request and Results Data Structures

5.2.7 Report Subsystem

Pre-Conditions: User must be logged in to be able to generate a report.
Post-Conditions:

- A simple but comprehensive report must be generated in the format chosen by the user.

- The report must only include papers that the current user is associated with.
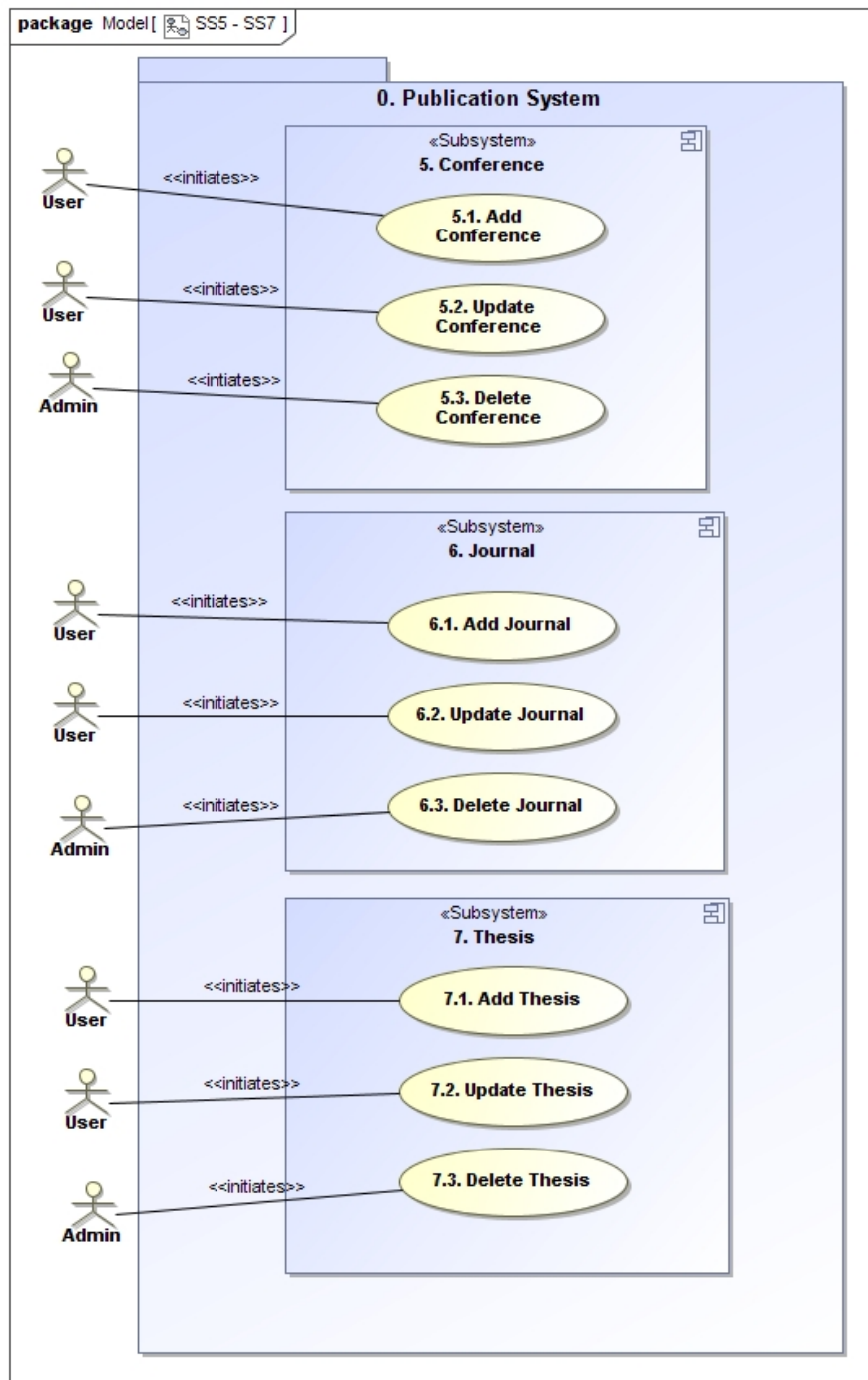
Figure 1: Use case diagram for the Conference, Journal and Thesis Subsystems

5.3 Required Functionality

5.3.1 User Subsystem Use Case

package Model[Update User]

User
+registerUser() : void
+updateUser() : void
+loginUser() : void
+logoutUser() : void
+addToPublication() : void

The user has to be registered to the system then the changes to the users profile are saved after the use case.

1.2 updateUser

«include»

«include»

getUser

saveChanges

---

package Model[Login User]

User
+registerUser() : void
+updateUser() : void
+loginUser() : void
+logoutUser() : void
+addToPublication() : void

The user has to be registered to the system first and if the login details are correct the user is logged into the system.

«include»

getUserPublications

1.3 loginUser

«include»

checkUserExists

«include»

startUserSession

«include»

validateLoginCredentials

9

package Model [🔲 Logout User]

User
+registerUser() : void
+updateUser() : void
+loginUser() : void
+logoutUser() : void
+addToPublication() : void

The user has to be logged in first then only will they be logged out of the system.

1.4 logoutUser

«include»

«include»

isUserLoggedIn

stopSession

---

package Model [🔲 Add to Publication]

User
+registerUser() : void
+updateUser() : void
+loginUser() : void
+logoutUser() : void
+addToPublication() : void

The user has to be logged into the system first. Then after the paper has been retrieved, the user is added to the publication

1.5 addToPublication

«include»

«include»

«include»

checkLoginStatus

getPaper

addUserToPaper

Figure 2: Create Publication

5.3.2 Publication Subsystem Use Case

11
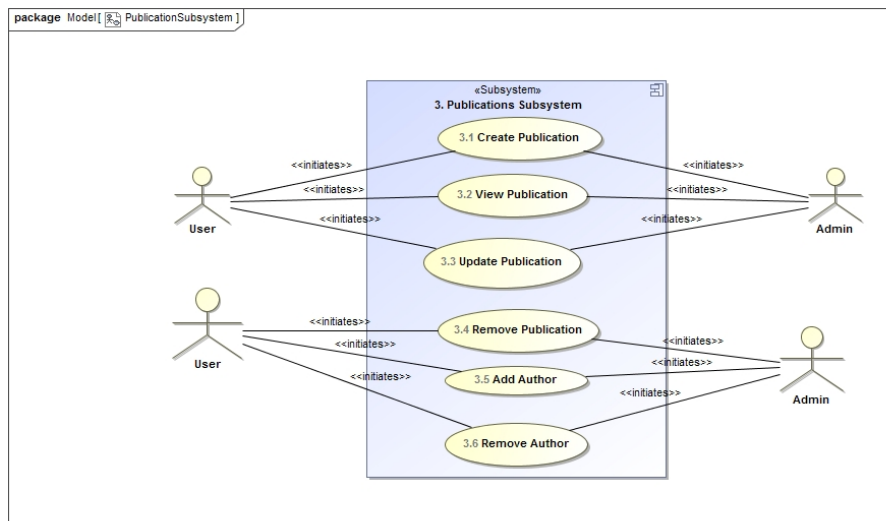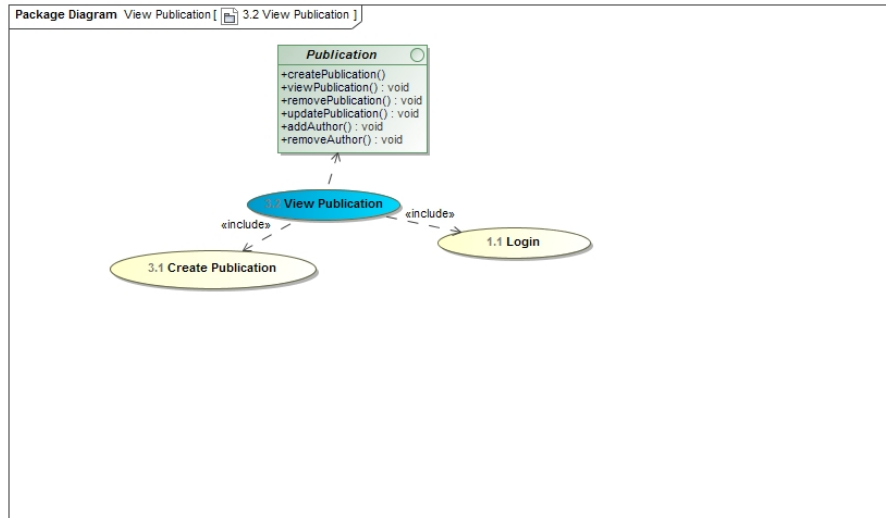
Figure 3: View Publication
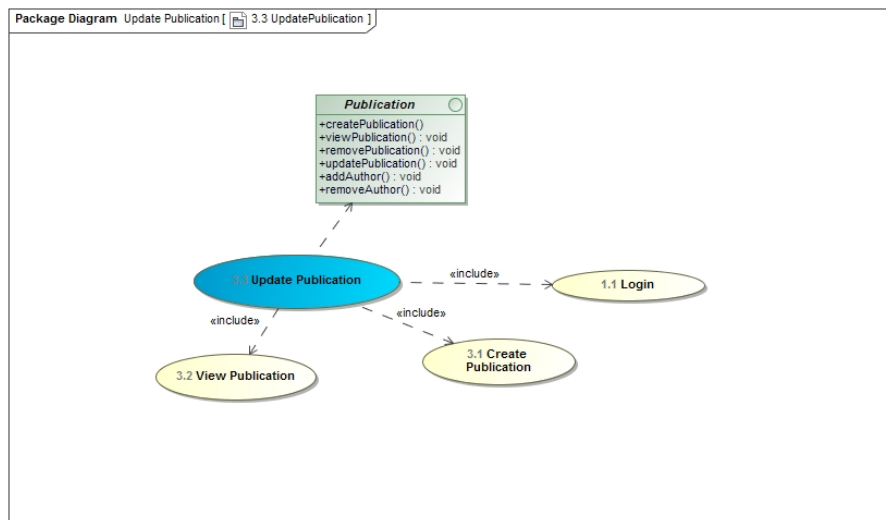
Figure 4: Update Publication
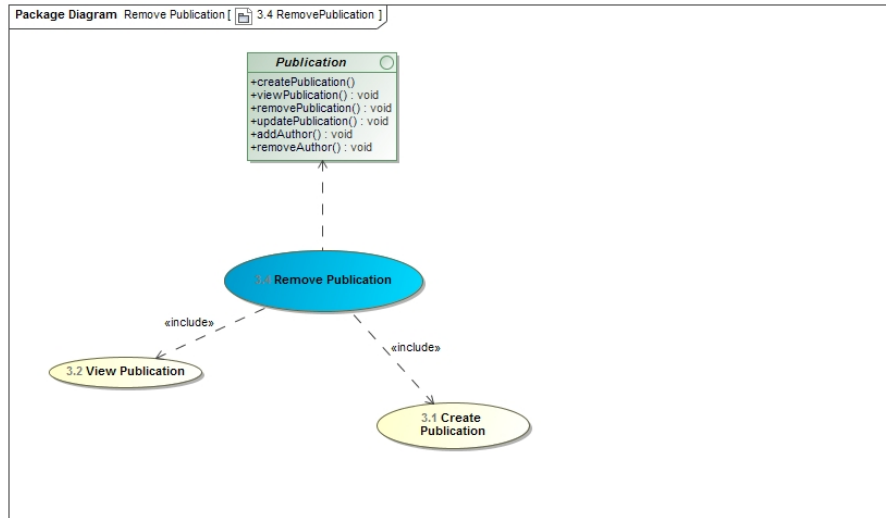
Figure 5: Remove Publication
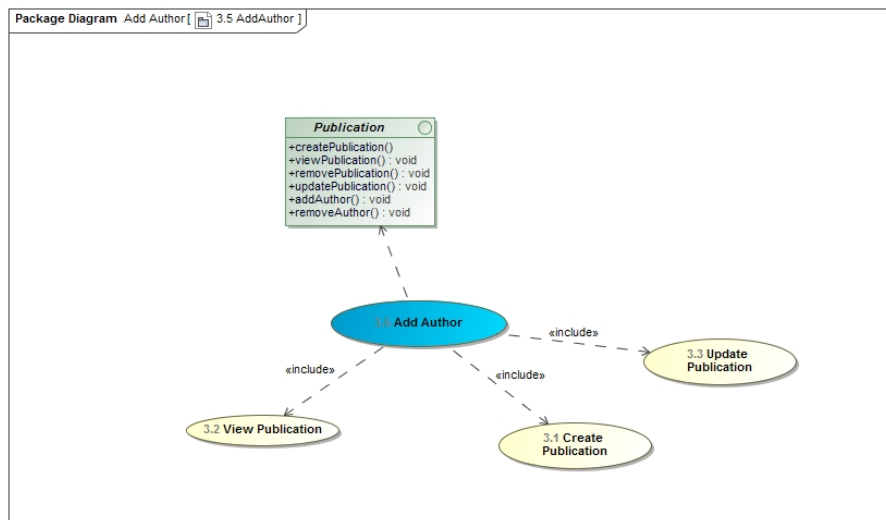
Figure 6: Add Author

13
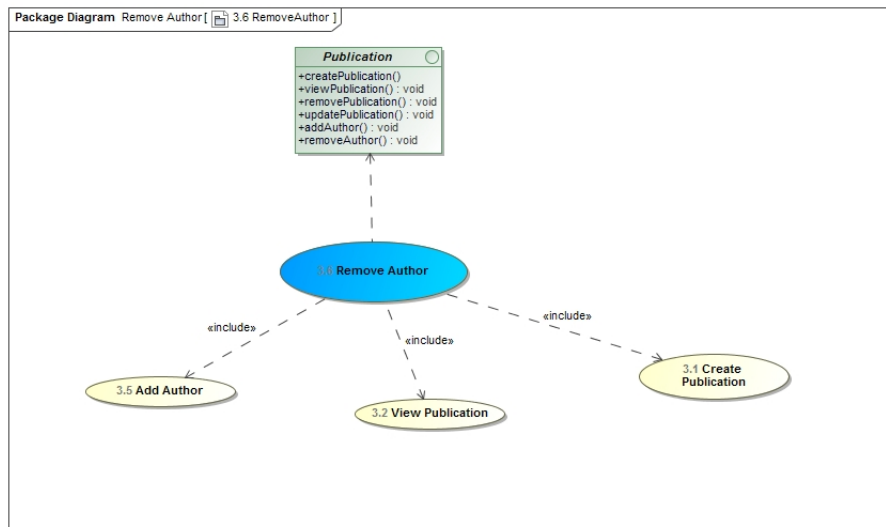
Figure 7: Remove Author

5.3.3 Research Group Subsystem Use Case
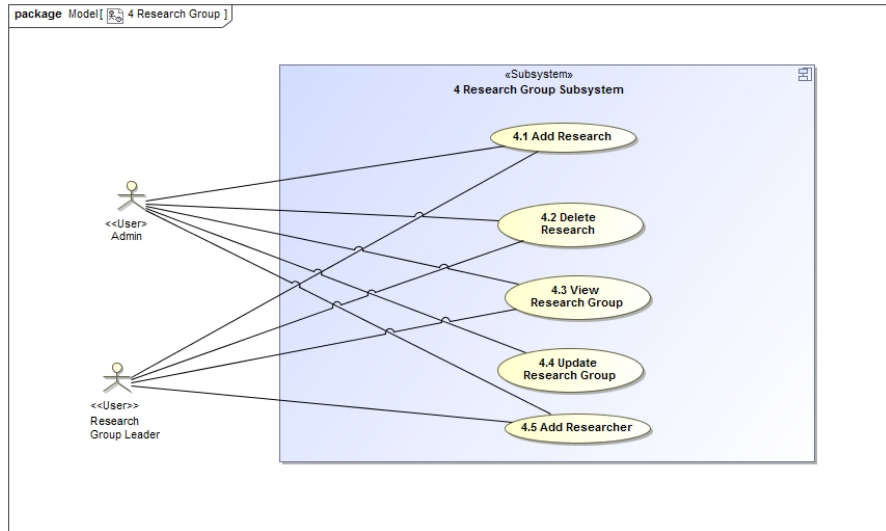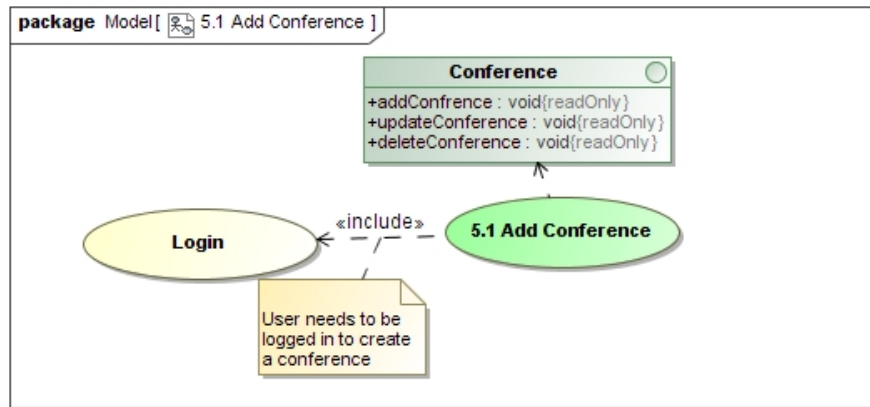
Figure 8: Add Conference
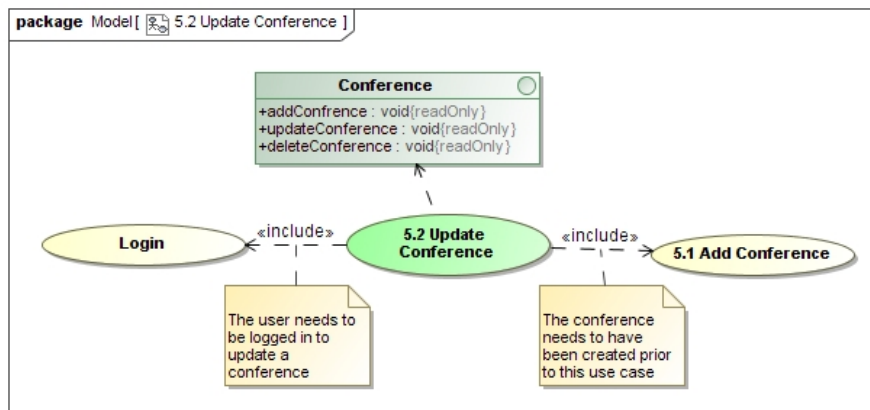
Figure 9: Update Conference
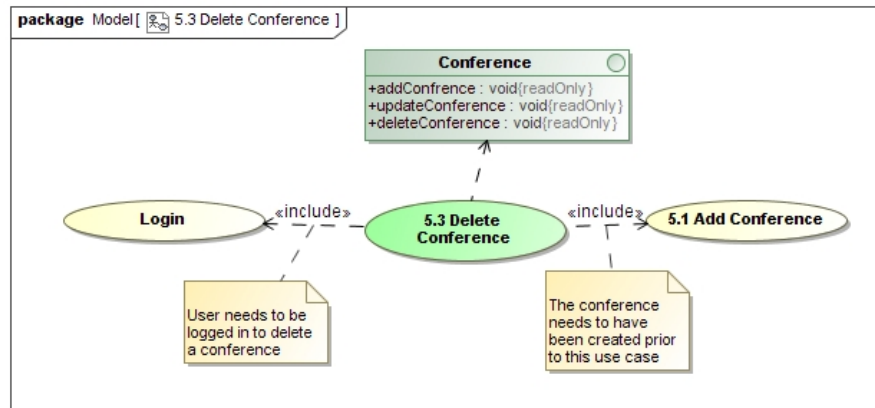
Figure 10: Delete Conference
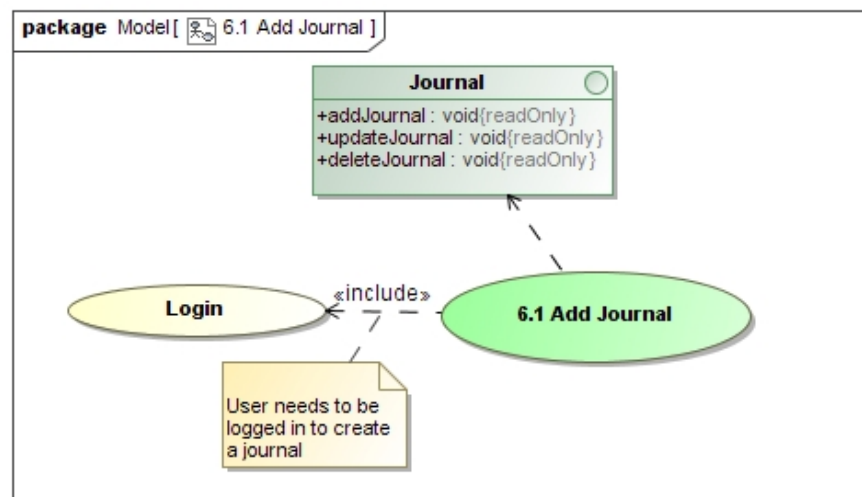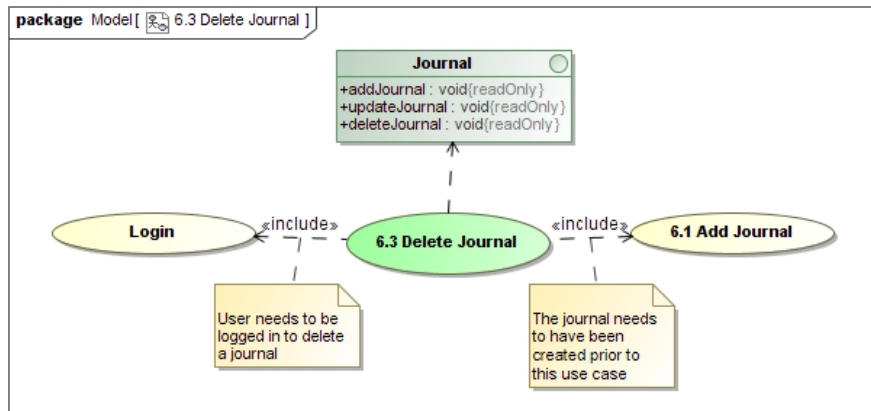

Figure 11: Add Journal
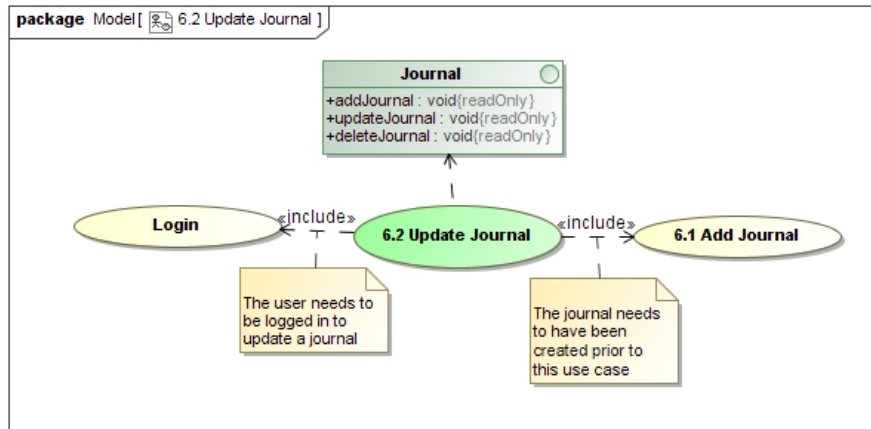
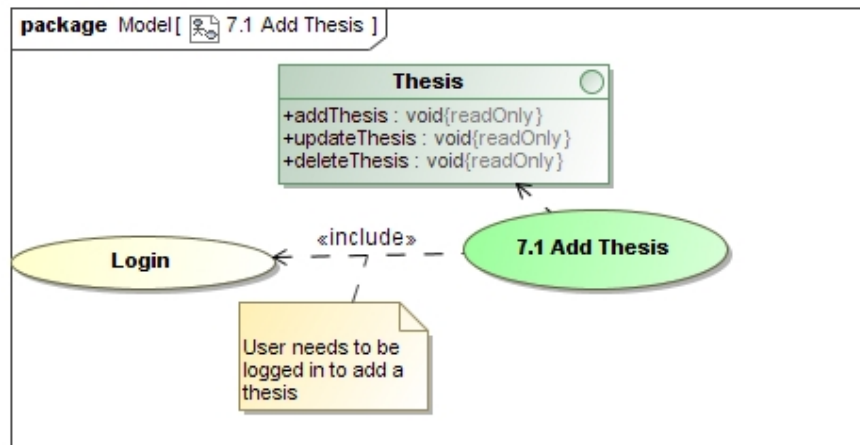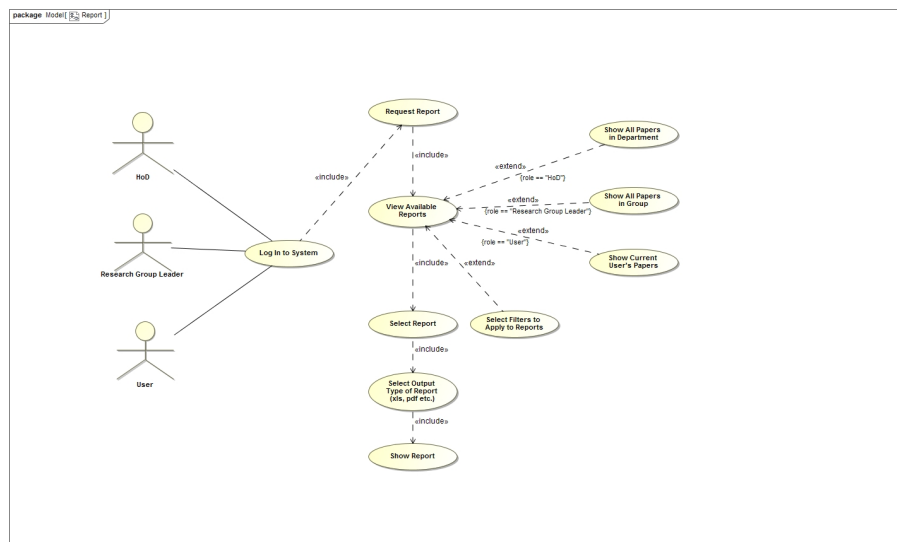Figure 12: Update Journal

Figure 13: Delete Journal

Figure 14: Add Thesis

5.3.4 Conference Subsystem Use Case

5.3.5 Journal Subsystem Use Case

5.3.6 Thesis Subsystem Use Case

5.3.7 Report Subsystem Use Case
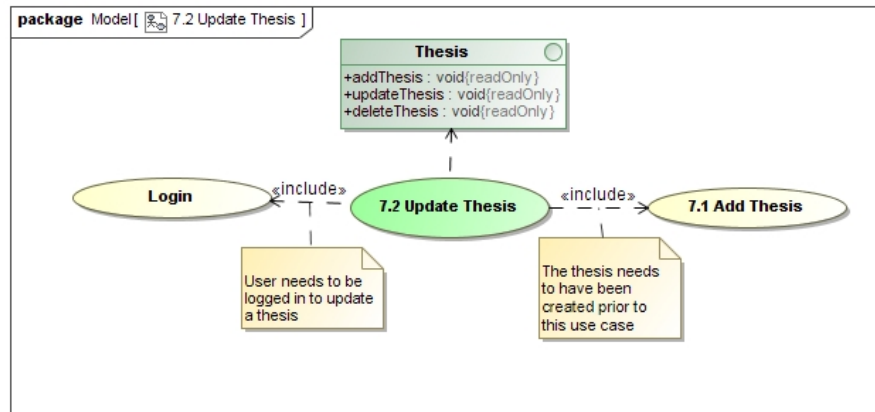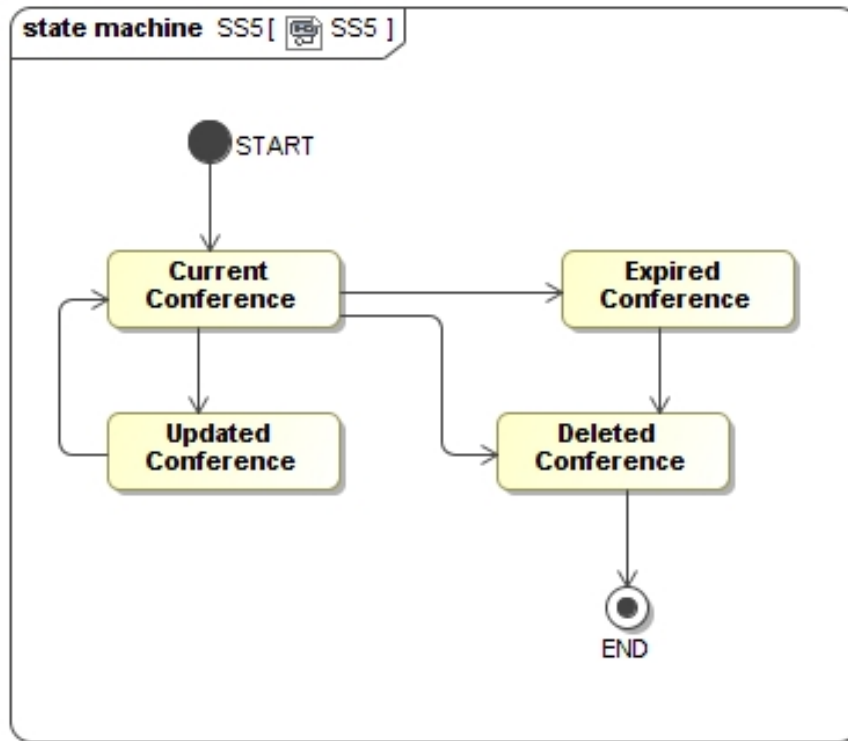
Figure 15: Update Thesis
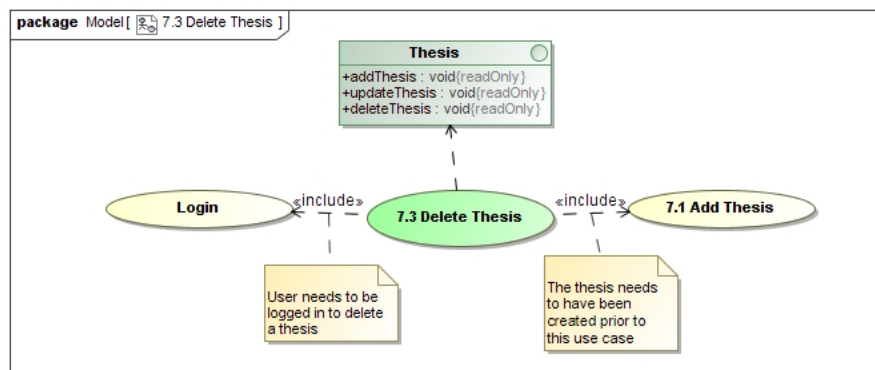
Figure 16: Delete Thesis

Figure 17: Conference State Diagram

5.4 Process Specifications

5.4.1 Conference Subsystem

5.4.2 Journal Subsystem

5.4.3 Thesis Subsystem

5.5 Domain Model

6 Open Issues

- What does DoE stand for?
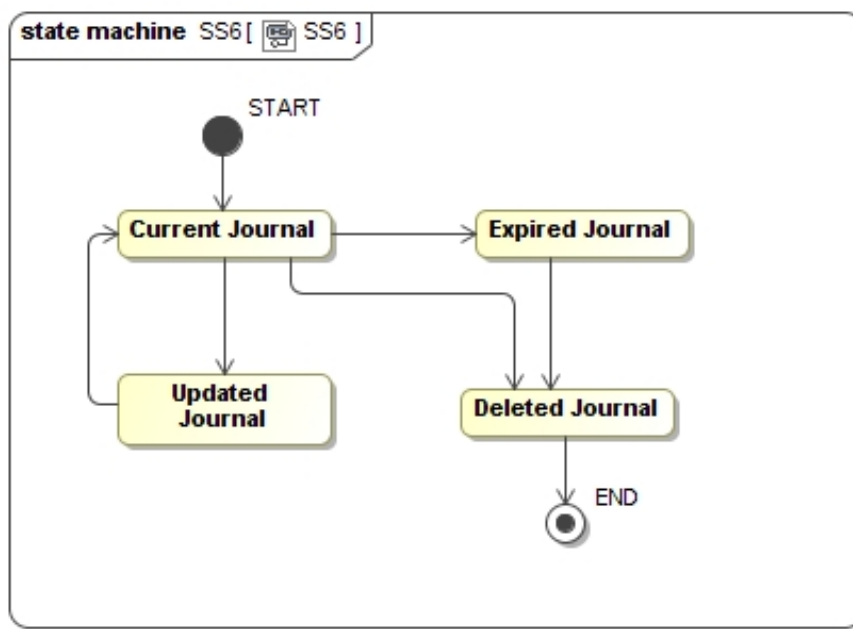- Specifics with regard to the DoE and UP weighted contributions
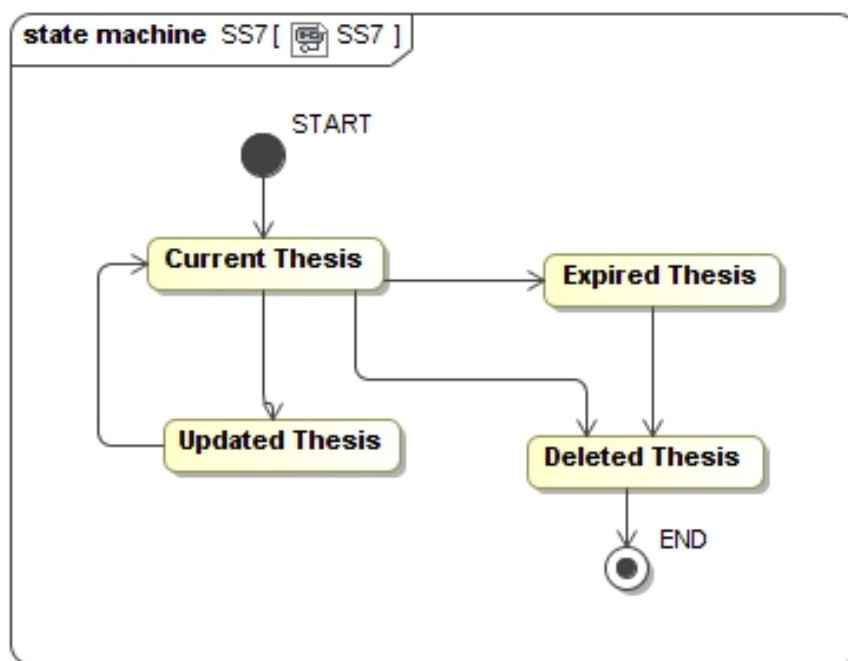- Researcher funding

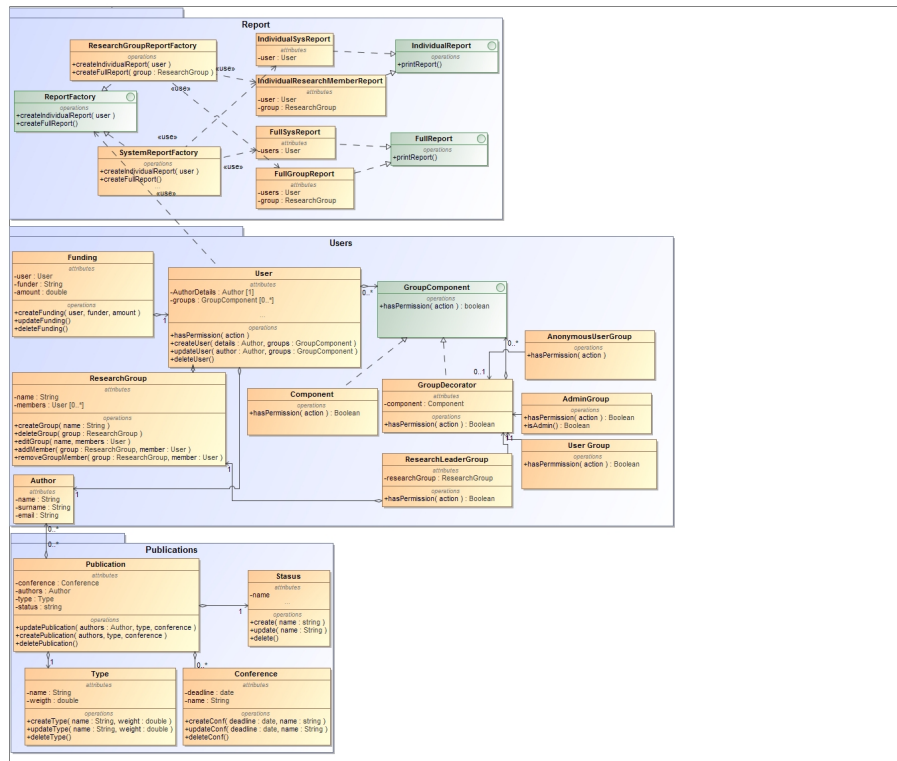Figure 18: Journal State Diagram

Figure 19: Thesis State Diagram

Figure 20: Domain Model