UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

FUNCTIONAL REQUIREMENTS

Project: Figbook
CLIENT: FIGTORY ANIMATION

Team: Creativate

Armand Pieterse *u12167844*
Kgomotso Sito *u12243273*
Jimmy Peleha *u12230830*
Sphelele Malo *u12247040*
Ndivhuwo Nthambeleni *u10001183*

DEPARTMENT OF COMPUTER SCIENCE, UNIVERSITY OF PRETORIA

Here's a link to GitHub.
https://github.com/SpheMalo/COS-301-Main-Project.git

July 21, 2015

Contents

List of Figures

1 Vision and scope

1.1 Project Background

In the world of publishing; how books, magazines, graphic novels and other academic text are produced predominantly using traditional means of publishing. There has not been much drive to leverage the latest developments in technology.

The traditional manner of publishing allows limited room for collaboration and makes the production life-cycle for a book long and tedious. The process of producing the book is still limited to the editor and sub-editor working in collaboration, quality assurance remains the sole responsibility of the publisher.

This has constrained the amount of books publishing houses produce over a set period of time. The inability to seamlessly collaborate on text and quality assurance extends the production life cycle of a book.

1.2 Project vision

Figbook aims to address the issue mentioned above. We intend to provide an application that will allow authors, agents, editors, copy-editors and designers to work on the book in an environment that encourages collaborative efforts and simplifies the life-cycle.

The system will allow multiple authors to collaboratively work on a project together so as to reduce the time spent on multi-authored works.

1.3 Project scope

The core of this project is to allow multiple authors to work on a book collayboratively. The system should allow smooth communication between different user roles at the different phases of the books life cycle.

The application will allow the authors to work on and see their project go from its infancy to the design and finally publishing phase.

2 Application requirements and design

This section discusses for each module of the Figbook system the functional requirements as well as the process designs for the use cases and the domain models which must be maintained within persistent storage by that module.

2.1 Authentication

2.1.1 scope

This module takes care of everything involving the users account. This includes functionality like logging in and out of the system, editing of user accounts etc. Further description is provided in the use cases below
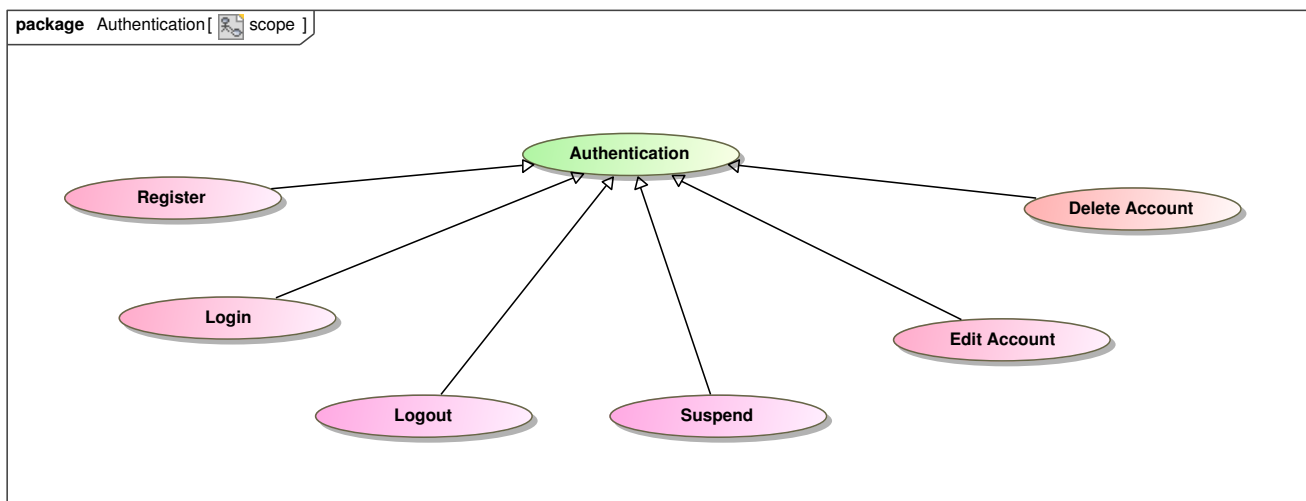

Figure 1: Scope for authentication module

2.1.2 Use Cases

1. **registerAccount priority: critical** This use case creates a user account that gets persisted to database.
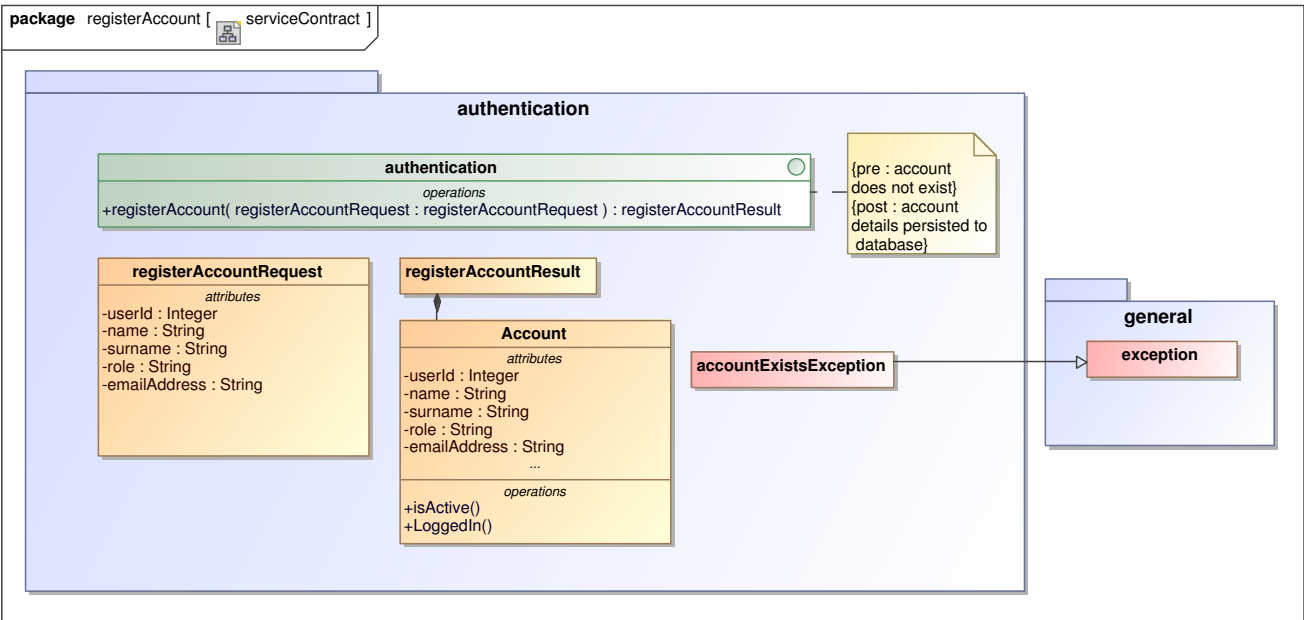
   Service Contract:

Figure 2: Service contract for registering new user account

2. **Login - priority: critical** This use case allows a user to log into an existing account.
Service Contract The service contract for the login service is shown in figure below.
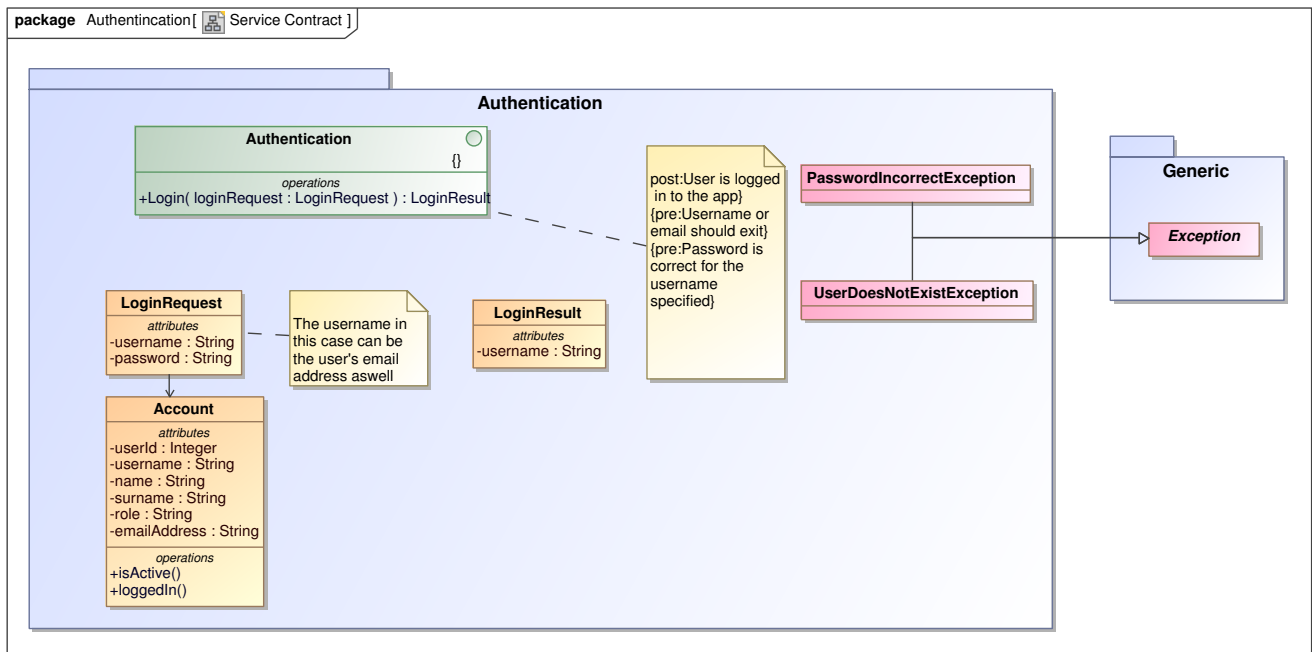

Figure 3: Service contract for logging into a user account

3. **Logout - priority: important**
This use case allows a user to log out of the system safely.
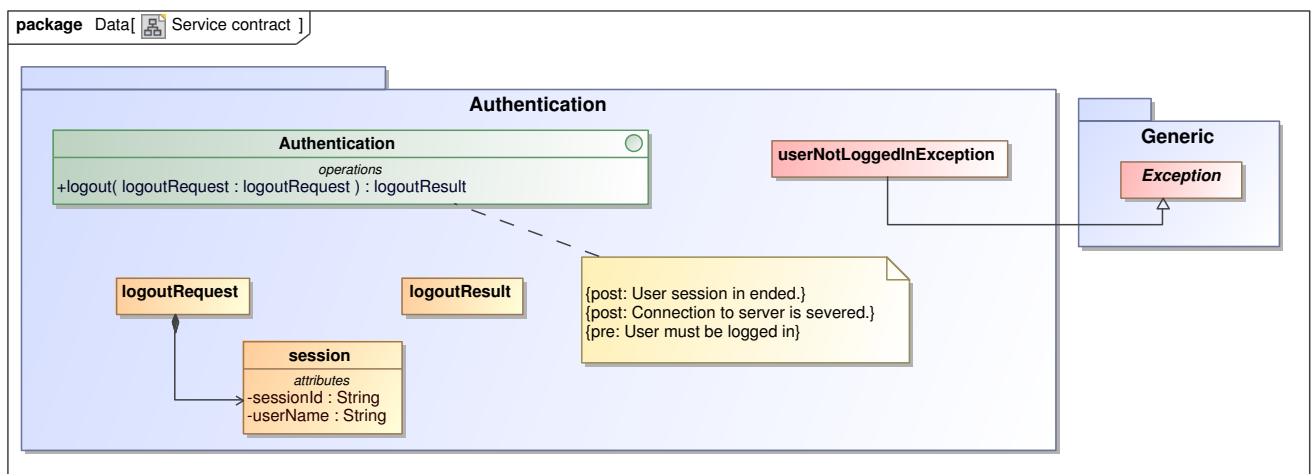
Service Contract:

Figure 4: Service contract for logging a user out of the system

4. **editAccount - priority: important**

   This allows a user to make changes to their profile details. They will be allowed to edit the details they added when creating the account.
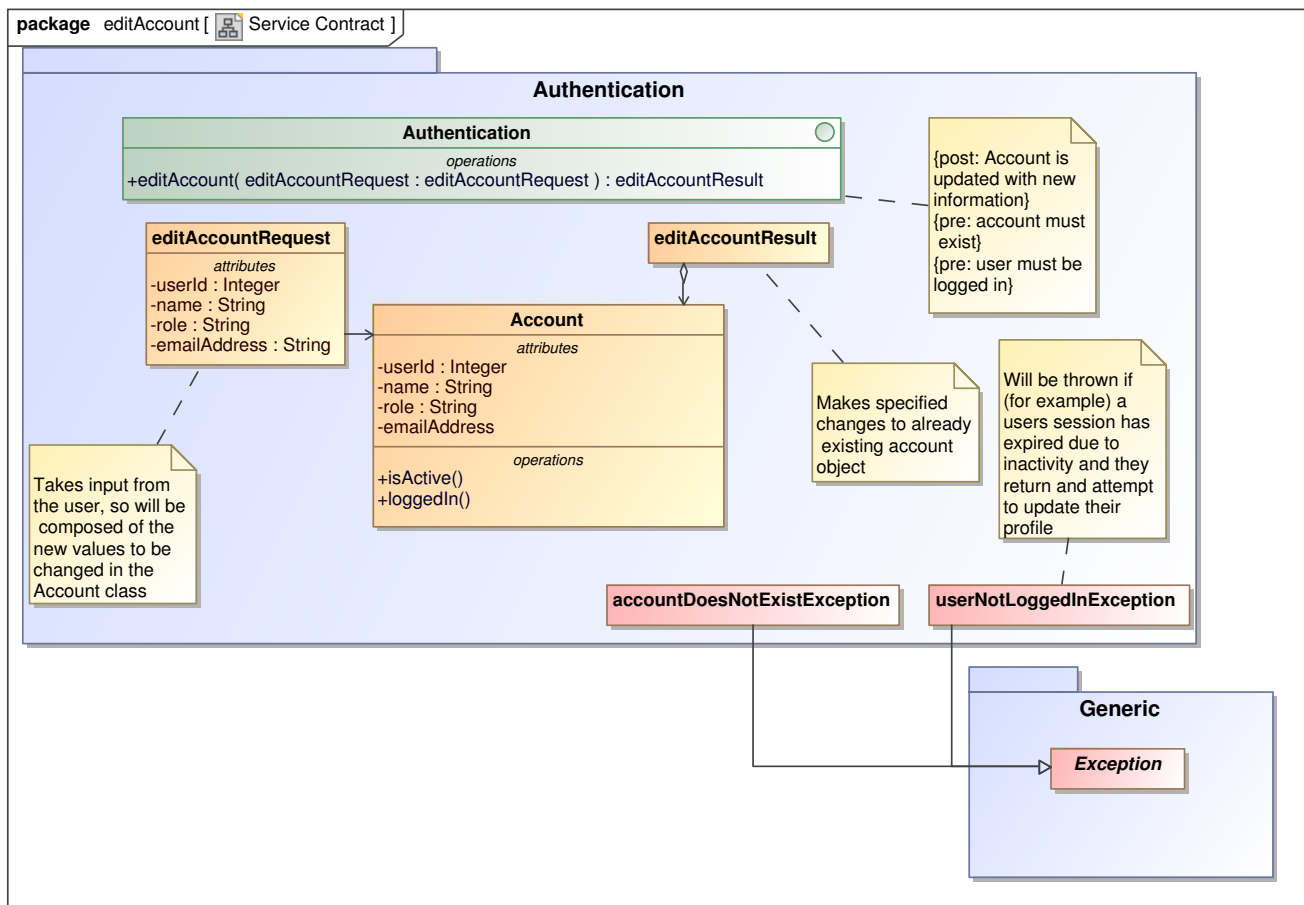
   Service Contract:

Figure 5: Service contract for editing a users account

5. **deleteAccount   priority: important** This use case creates a user account that gets persisted to database.
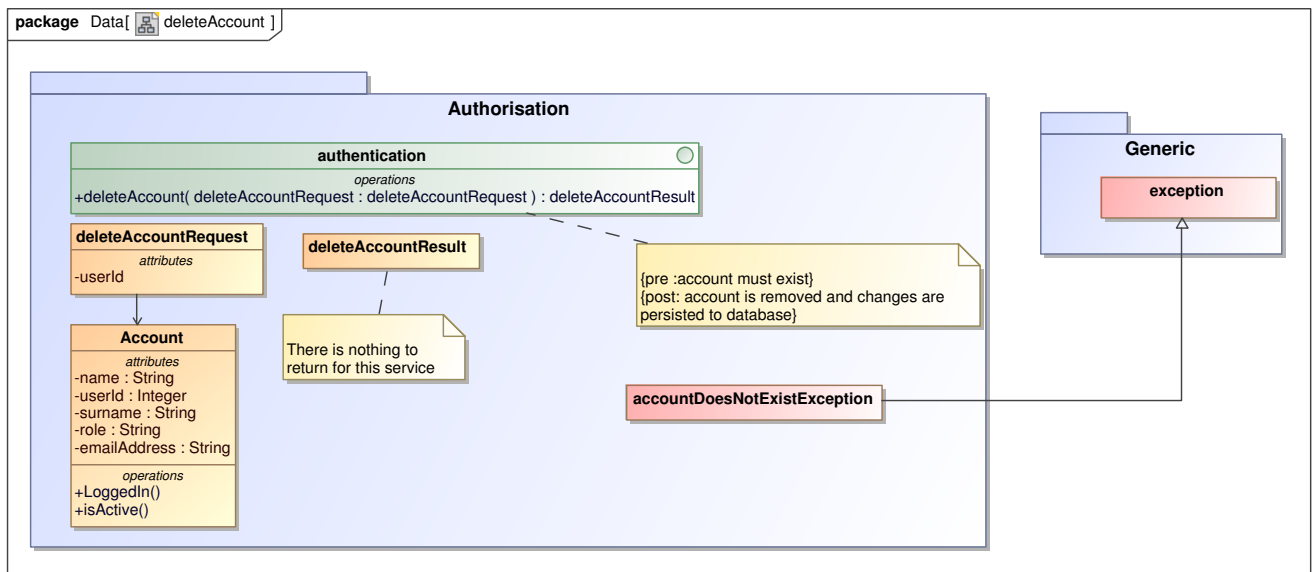
   Service Contract:

Figure 6: Service contract for deleting a users account

6. **SuspendAccount - priority: important**

This use case allows the system administrator to suspend a users account for any reason he/she may see as appropriate. A suspended account bars the user from logging in and/or using the account until the administrator lifts the suspension.
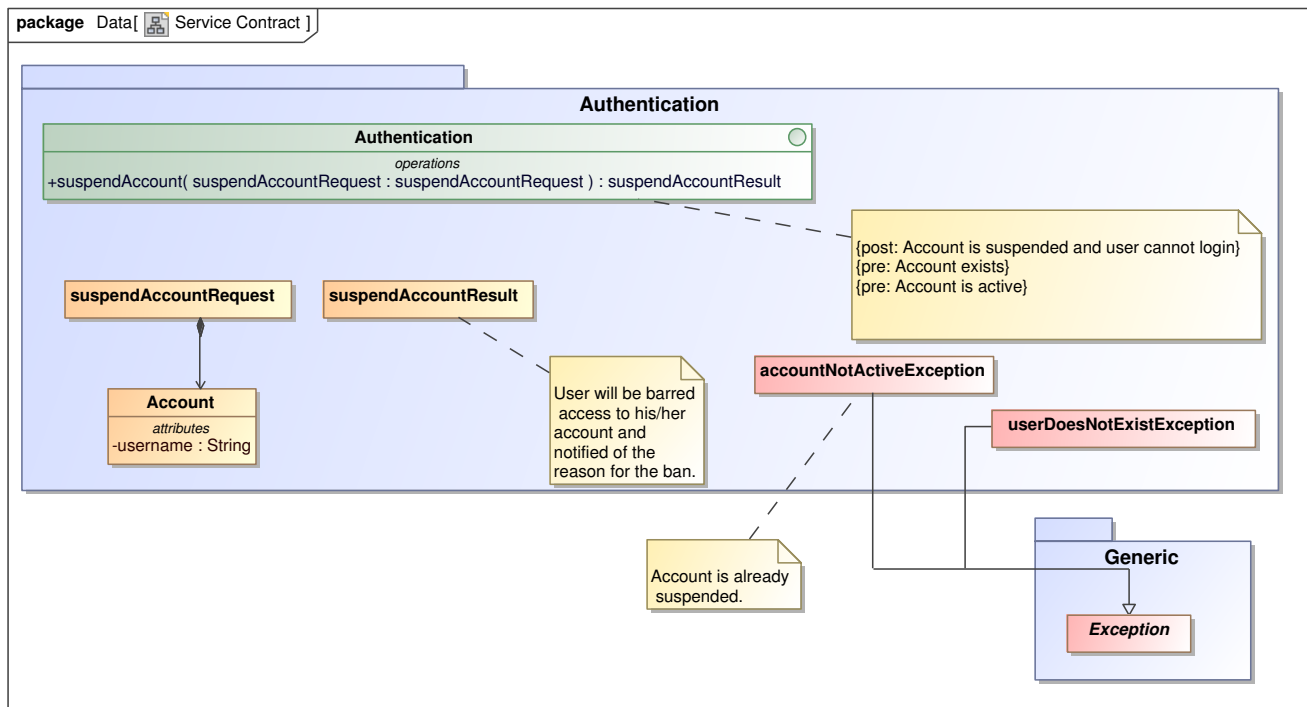
Service Contract:

Figure 7: Service contract for suspending a users account

2.1.3 Domain Model

The Authentication module will keep Account objects in the database. Any part of the system that checks for user authorisation will check against the account table to see if that particular user has authorisation to do whatever it is they are attempting to do.
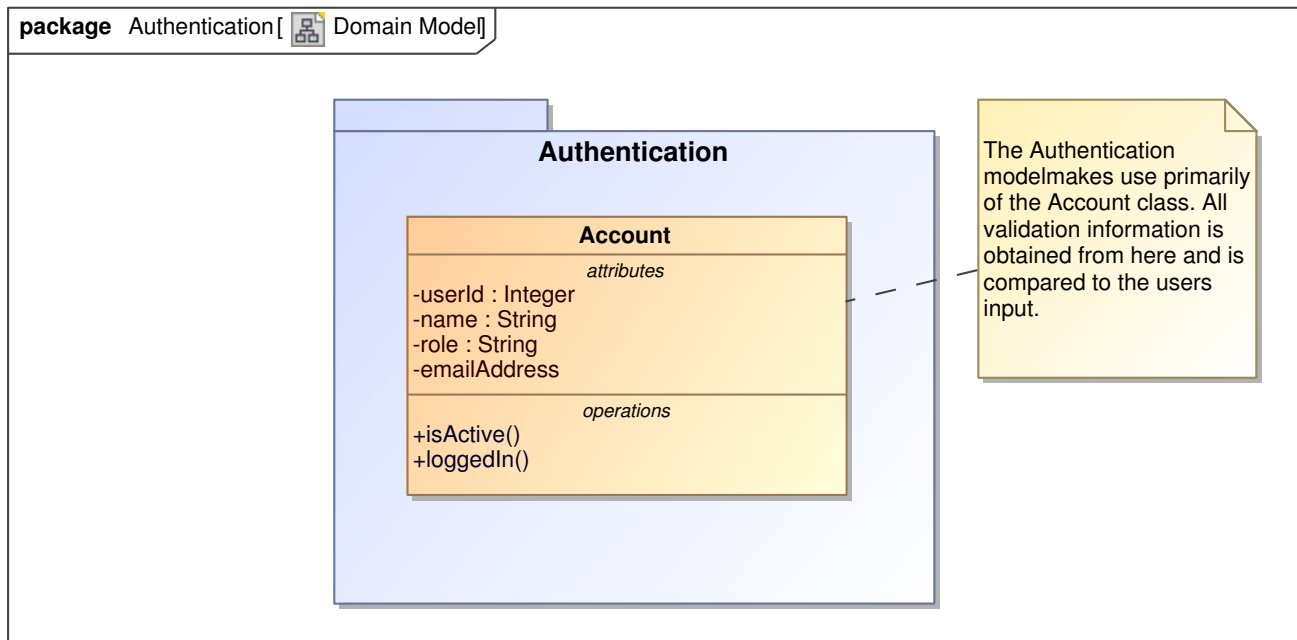

Figure 8: Domain model of Authentication module

2.2   Manuscript Management

2.2.1   Scope

This section provides the detailed service contracts and activity diagrams(where needed) for the use cases offered by the Manuscript Management module. It's services includes all the needed functionality to create, read, evaluate, edit and to send it to the next role in the life-cycle of creating a quality book.
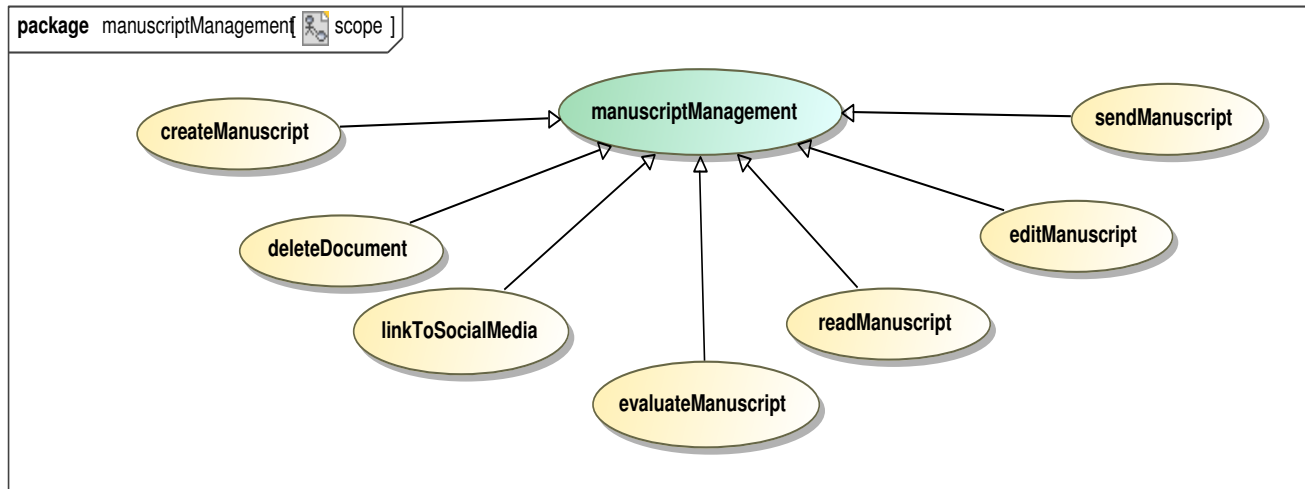

Figure 9: Scope for manuscript management module

2.2.2 Use Cases

1. **Create Manuscript - priority: critical**

   This use case allows an author to create a new project, by creating a new manuscript. The author will be able to give the manuscript a name(title) and specify the privacy setting on it. There may be one or multiple authors collaborating on a manuscript. A manuscript's title should be unique.
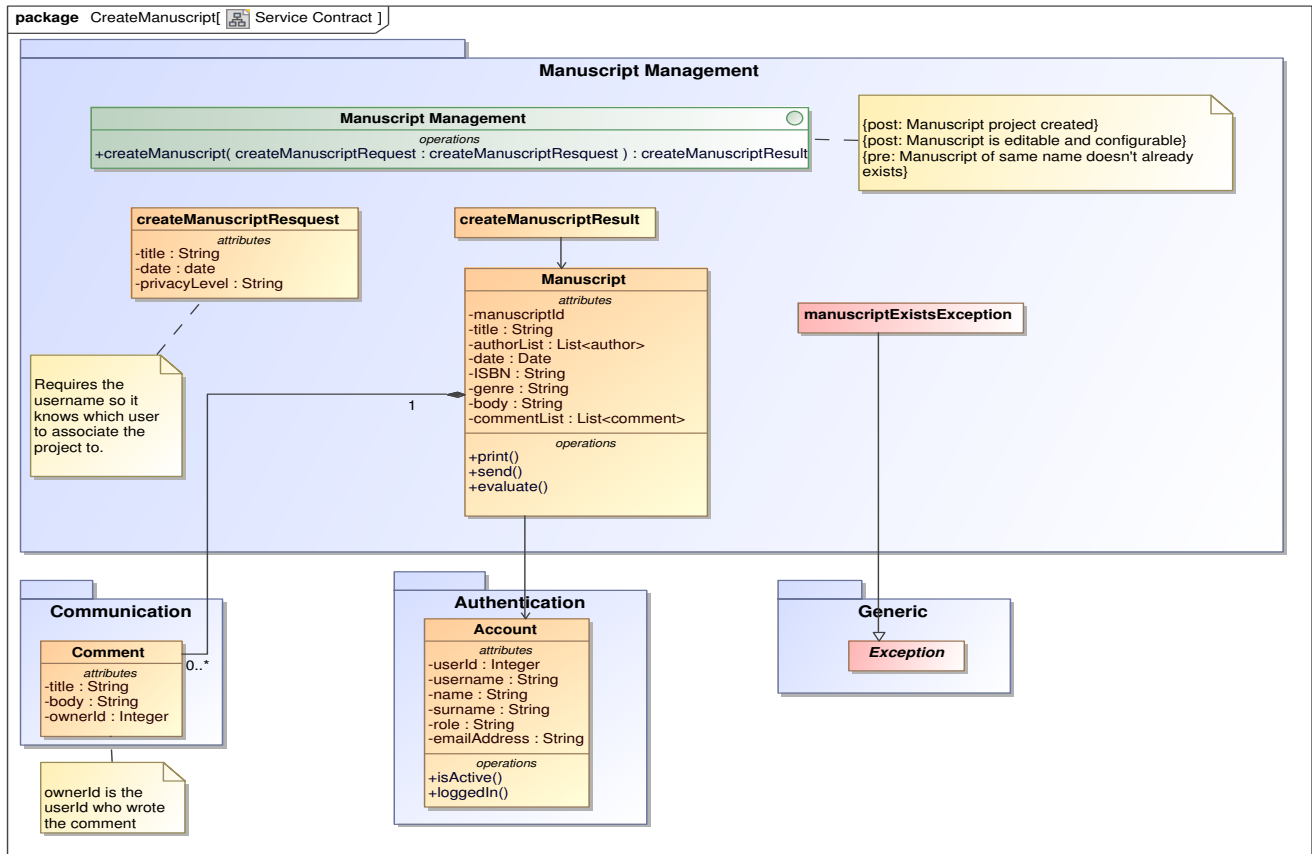
   Service Contract:

Figure 10: Service contract for creating a new project (manuscript)

2. Delete Manuscript priority: important

This use case removes an existing manuscript. The pre-conditions are enforced (raising the appropriate exception should they not be met) and the manuscript of interest is deleted and changes are persisted through to database.
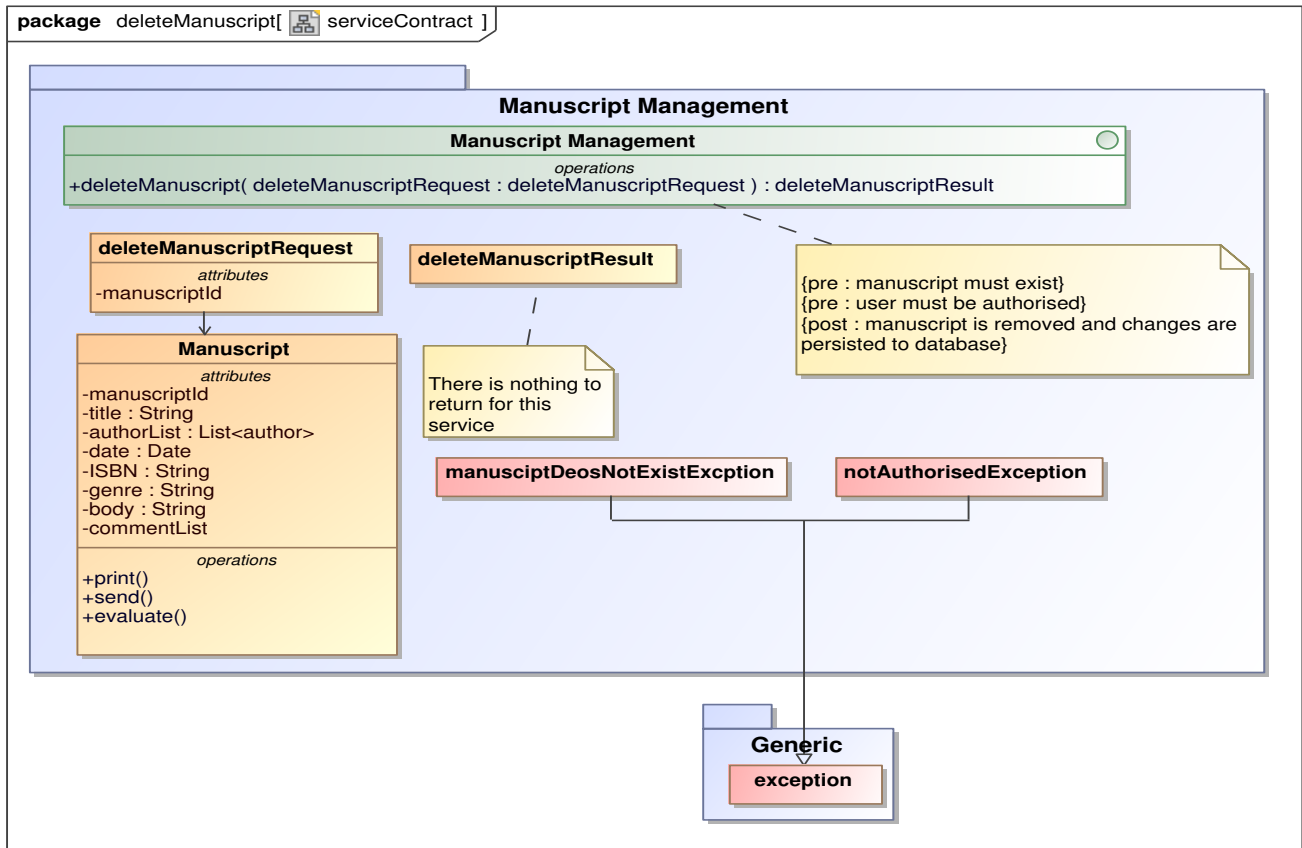
Service Contract:

Figure 11: Service contract for deleting a project (manuscript)

3. **Link To Social Media priority: important**

This use case links a project(manuscript) to a social media of choice if conditions are met and creates an automated post on that social media.
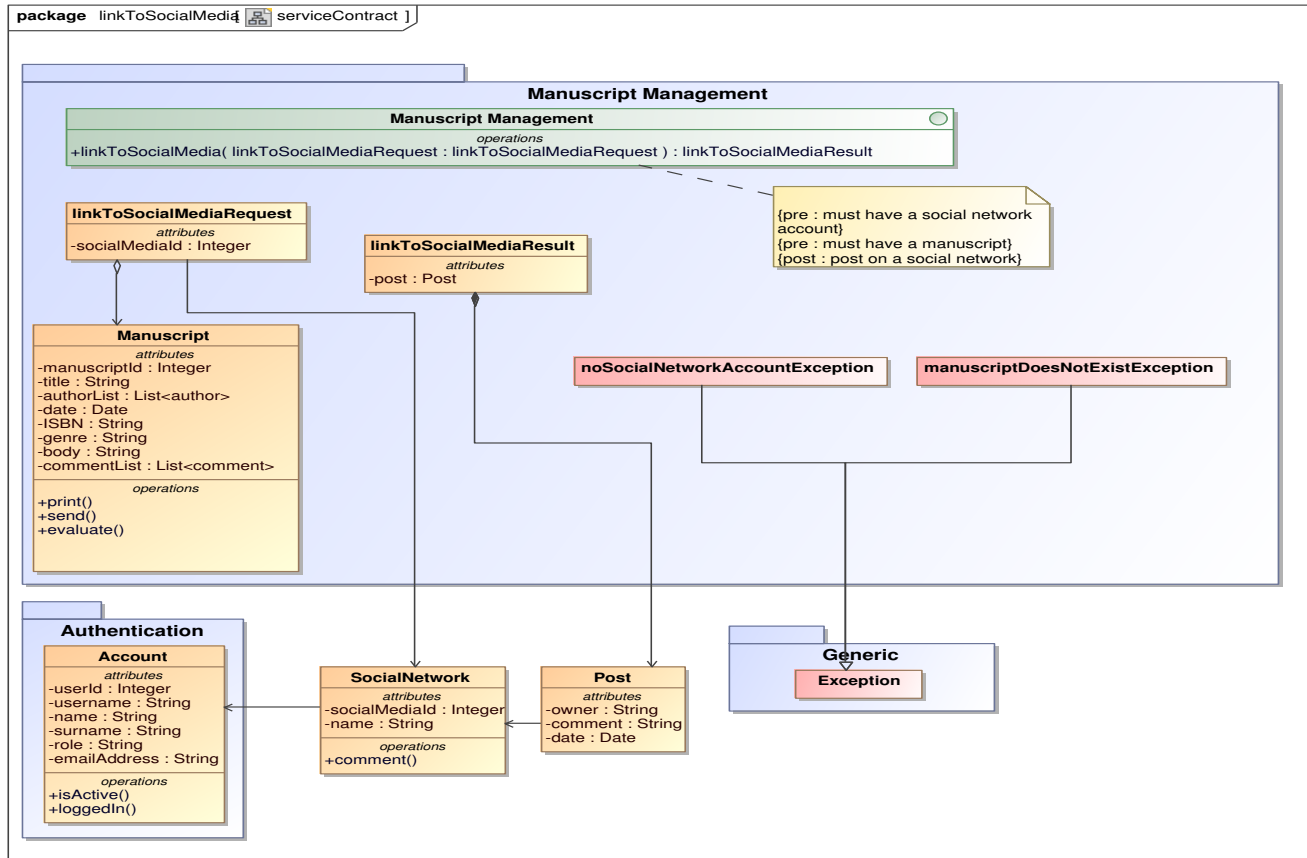
Service Contract:

Figure 12: Service contract for linking a project (manuscript) to a social network

4. **Edit Manuscript priority: critical**

   This use case which allows one to edit a manuscript if authorized and afterwards changes are persisted to database.
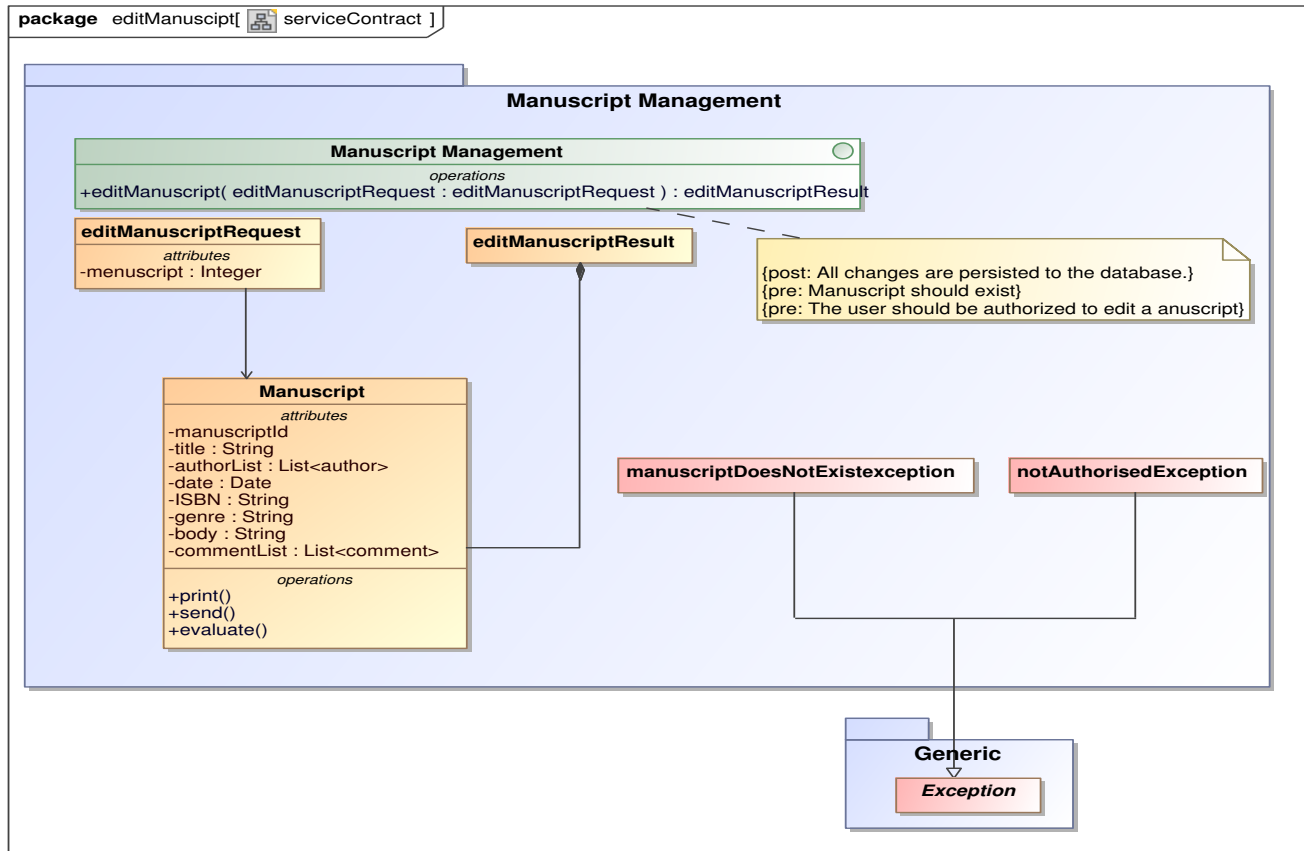
   Service Contract:

Figure 13: Service contract for editing a project (manuscript)

5. **Evaluate Manuscript - priority: critical**

   This use case allows an agent or an editorial to evaluate the manuscript and to give the according feedback to the author, such as a comment or editorial letter.
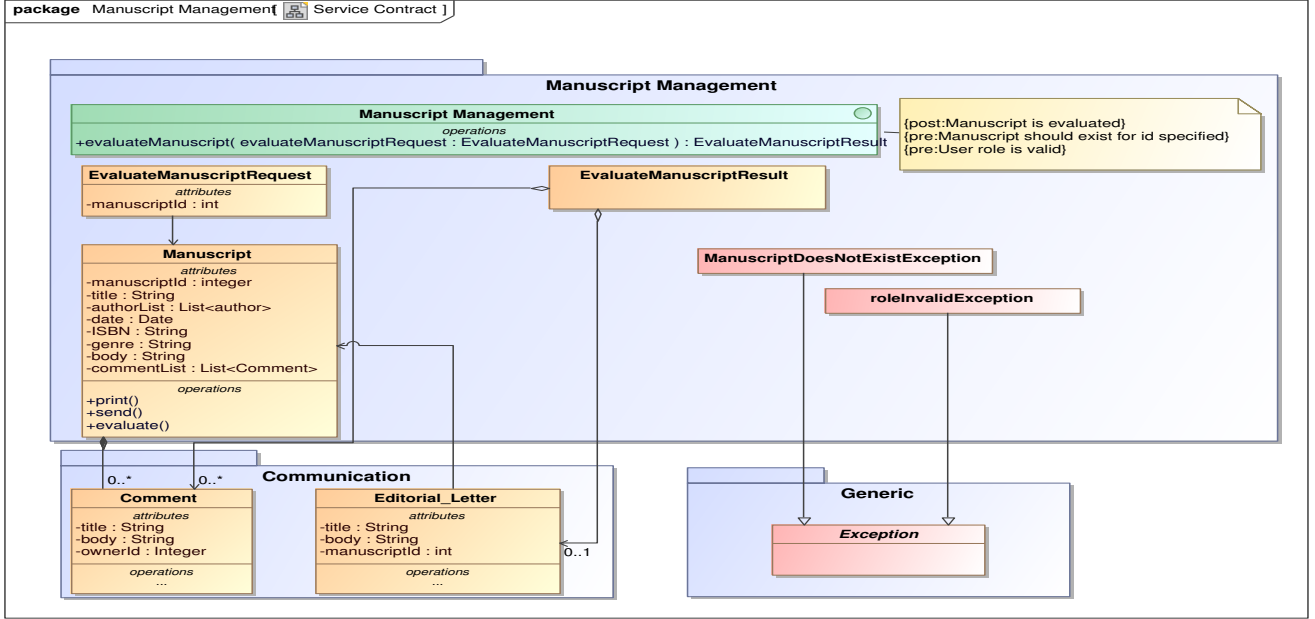
   Service Contract:

Figure 14: Service contract for evaluating a manuscript
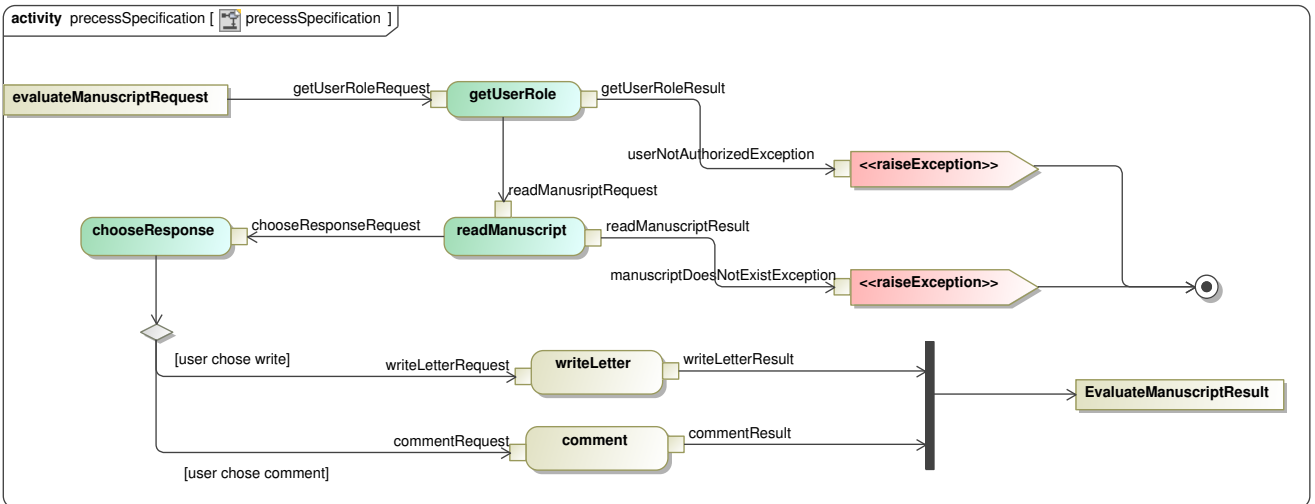
Process Specification (Activity diagram):

Figure 15: Process Specification for evaluating a manuscript

6. **Read Manuscript - priority: critical**

The use case grants access to users to view/read a manuscript, under the conditions that the user is authorized and that the manuscript still exists.
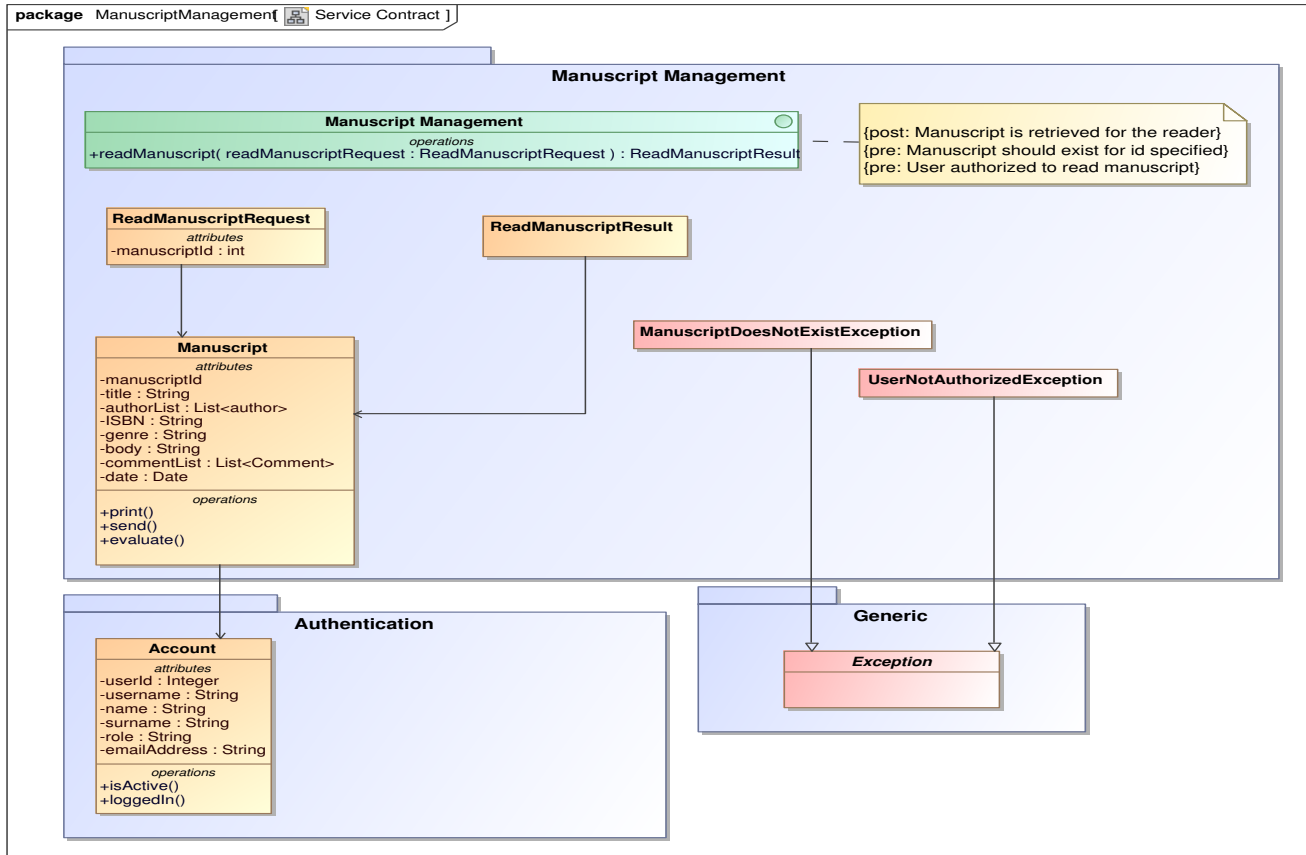
Service Contract:

Figure 16: Service contract for reading a manuscript

7. **Send Manuscript - priority: critical**

   This use case allows a user link another user on the system to the manuscript. This allows the recipient to access it and, depending on the privileges afforded by the owner of the manuscript, be able to read/modify it.
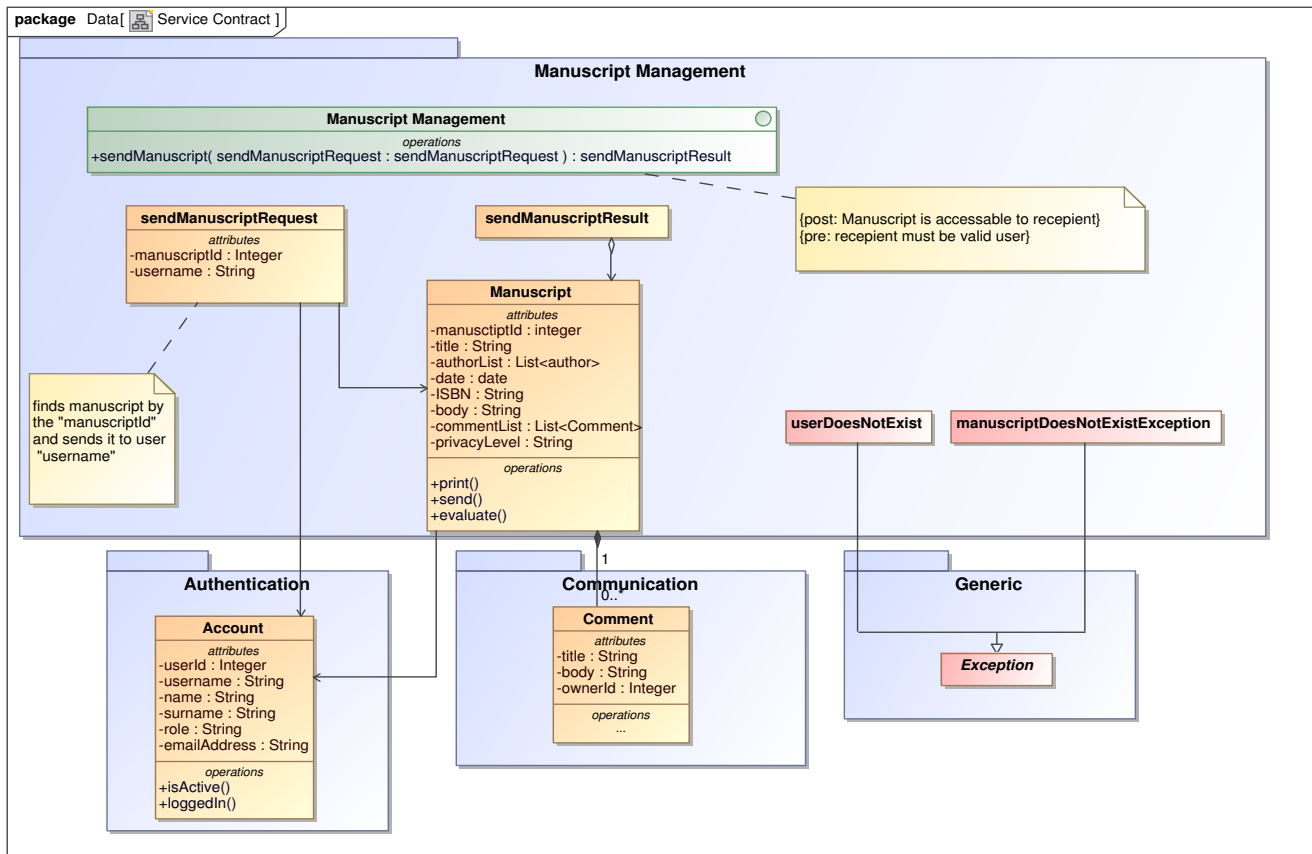
   Service Contract:

Figure 17: Service contract for sending a manuscript

2.2.3 Domain Model

This module will keep a manuscript in the database as well as be connected to all the other modules needed to perform the necessary tasks such as communication, adding authors,editors,agents etc, the modules connected to this module at the current stage of management are: Communication, Authentication as well as the external classes required for linking the manuscript to social media.
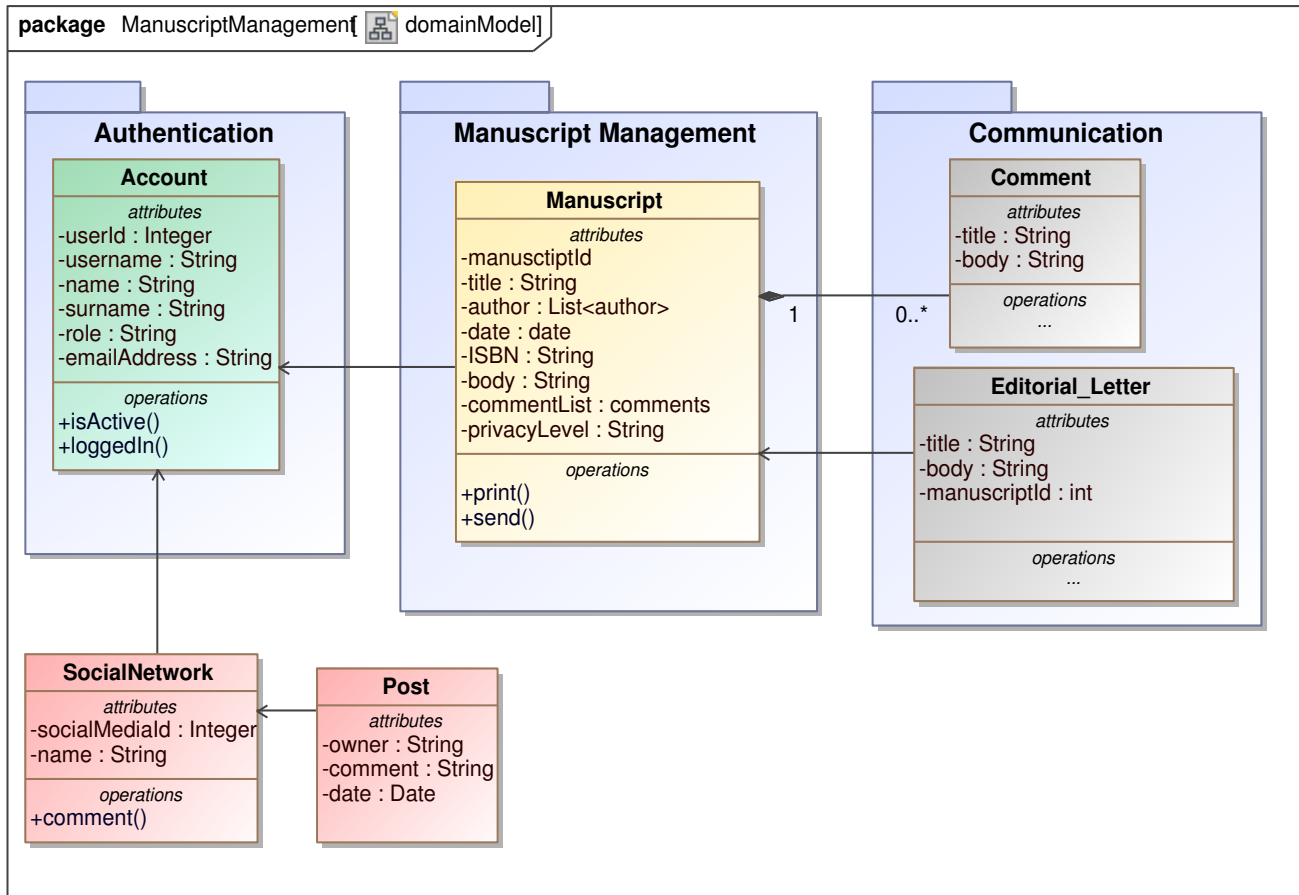

Figure 18: Domain model for managing a manuscript