COS 301 ASSIGNMENT
GROUP 5 B

Sebastian Gerber (12213749)
Godfrey Mathe (13103394)
Shaun Meintjes (13310896)
Duran Cole (13329414)
Isabel Nel (13070305)

February 2015

1 Introduction

The idea behind the 'Buzz' project is to find ways to enhance teaching and improve the learning of students through the use of online discussions. The 'Buzz' will be an online forum where students can express their views on specific work topics, thus students will be able to see the work from other students perspective's and be able to discuss problems and find solutions creating an educational environment while allowing for social interactivity among students.

The idea is to create a fun, rewarding and educational environment, thus students will be rewarded for participating in this online forum by being 'leveled up' according to contributions on the forum. When a student reaches a new level he/she will have more privileges such as being able to start their own thread and having access to more functionality. All users will also be able to see each other's levels - in this way we try to motivate students to 'level up' to also receive those privileges.

By having students communicating about work related to subjects the lecturers will also be able to view the discussions and thus see what problems and questions students have about the work and can address these again in class if needed. Lecturers on the other hand will also have a lot more privileges than students and can at any time start a new thread and discussions and so forth.

The forum needs to be user-friendly to help inexperienced users to easily navigate and find their way around the forum thus it also needs to be well and logically organized. It should also raise excitement in students to want to participate in discussions or just go onto the forum and read about work covered during lectures to get a better understanding.

2 Functional requirements and application design

2.1 Use case prioritization

2.1.1 Critical

- Login
- Logout

- Post
- Thread
- Buzz Space
- User Management(Profile)

2.1.2 Important

- Search
- Voting
- Notifications

2.1.3 Nice to Have

- Request

2.2 Use case/Services contracts

2.2.1 Login

Pre-Conditions :

- Must not be logged in.

Post-Conditions :

- If login succesful, user will be logged in.
- If login failed, user will be asked to resubmit.

2.2.2 Logout

Pre-Conditions :

- Must be logged in.

Post-Conditions :

- User is logged out.

2.2.3 Post

Pre-Conditions :

- Must be logged in.
- A thread should exist.

Post-Conditions :

- Post is created.

2.2.4 Buzz Space

Pre-Conditions :

- To create a Buzz Space, user must be logged in and user must be authorized.
- To delete a Buzz Space the Buzz Space must exist and user must own the Buzz Space.
- To edit a Buzz Space, the Buzz Space must exist and user must own Buzz Space.

Post-Conditions :

- Buzz Space is either created, deleted or edited according to the functionality used.

2.2.5 Request

Pre-Conditions :

- Must be logged in.

Post-Conditions :

- Request is sent.

2.2.6 Thread

Pre-Conditions :

- Only Authorised persons may use this functionality.
- Buzz Space needs to exist . No such thread sould already exist.

Post-Conditions :

- A new Thread now exists under which people can create posts.

2.2.7 Voting

Pre-Conditions :

- Must be logged in.
- User has not yet voted.

Post-Conditions :

- User voted.

2.2.8 Search

Pre-Conditions :

- User needs to enter a search request.

Post-Conditions :

- Search results are returned.

2.2.9 Notifications

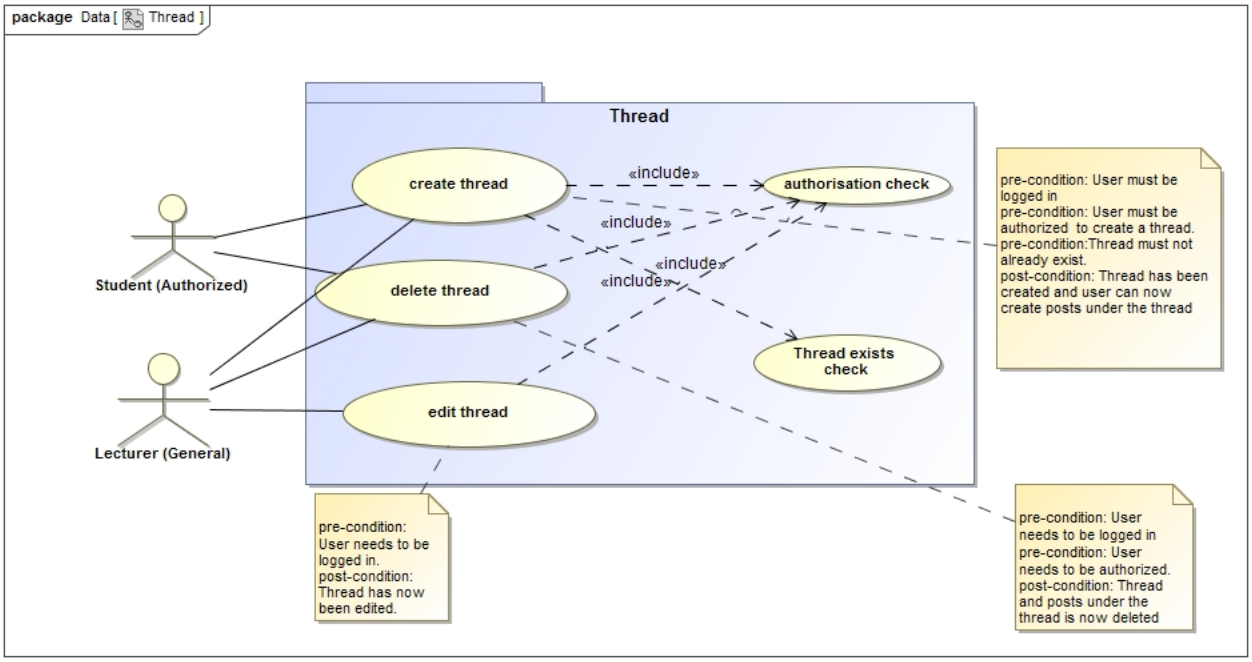
Pre-Conditions :

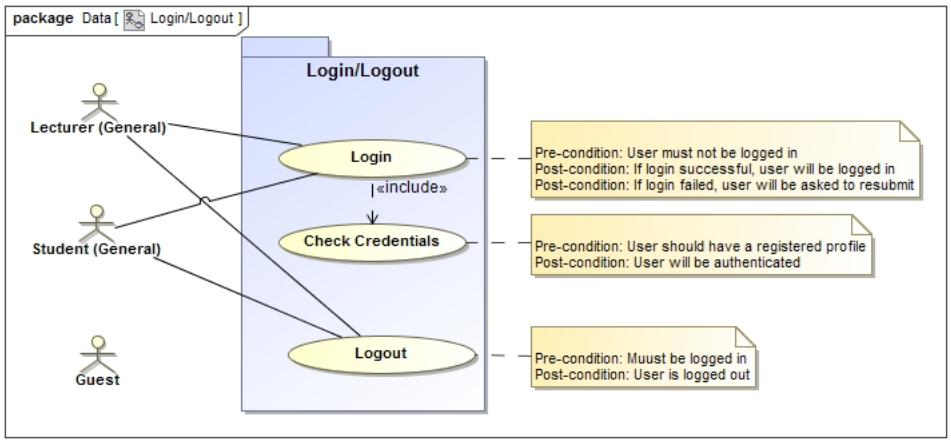
- Must be logged in to see notifications.

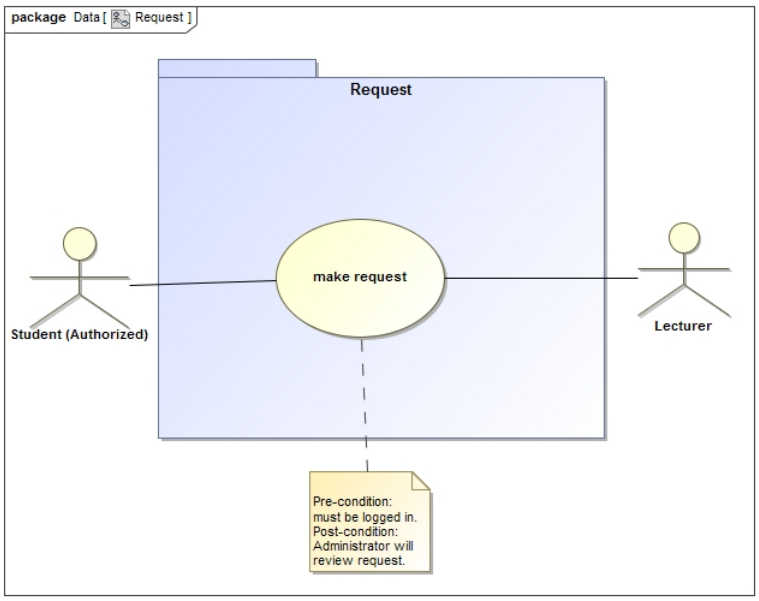
Post-Conditions :

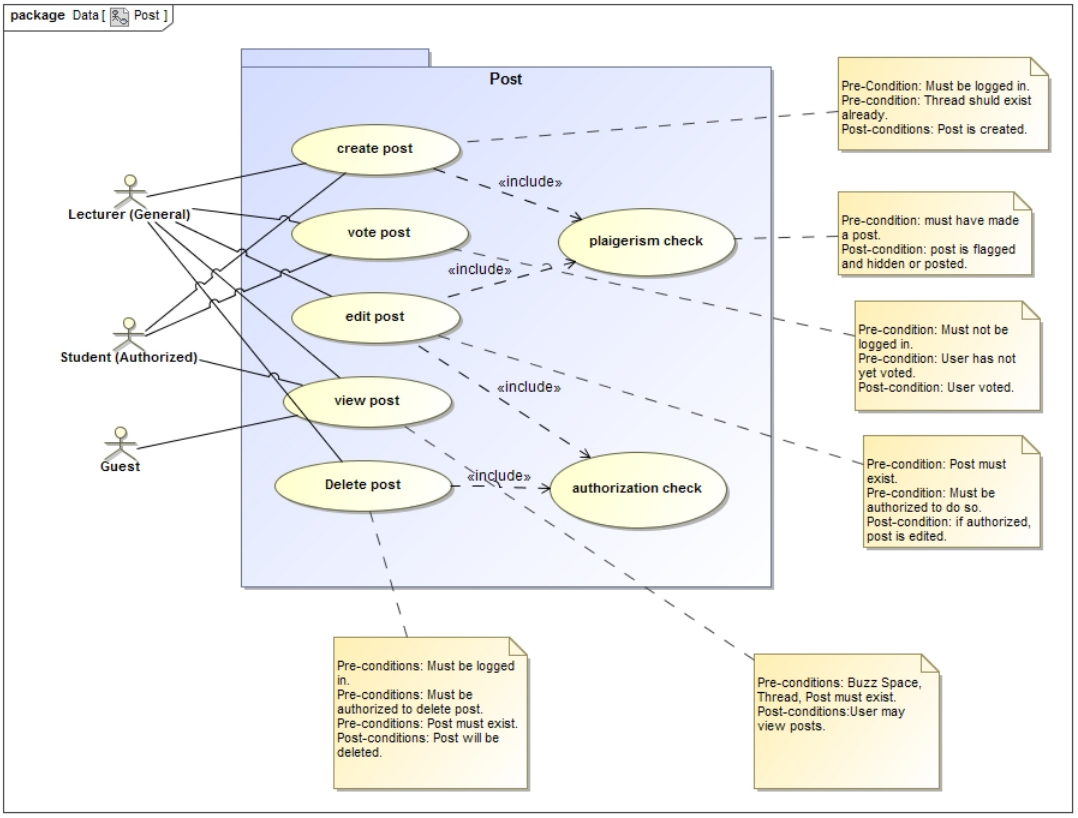
- Notification is displayed to user.

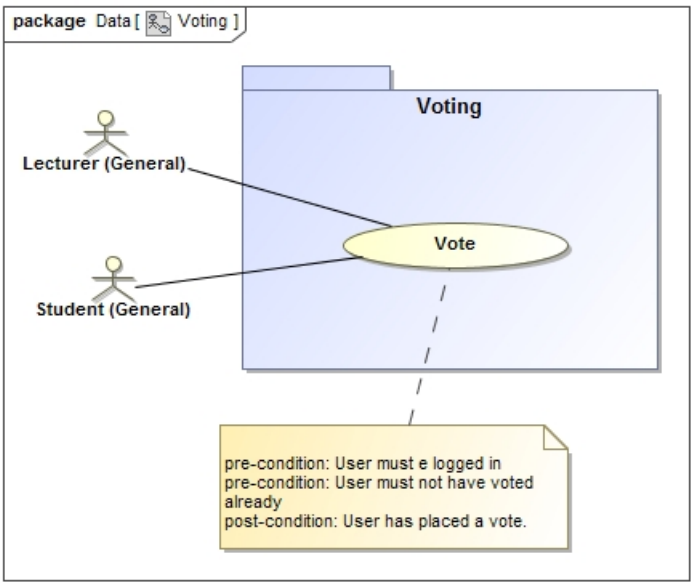
2.3 Required functionality

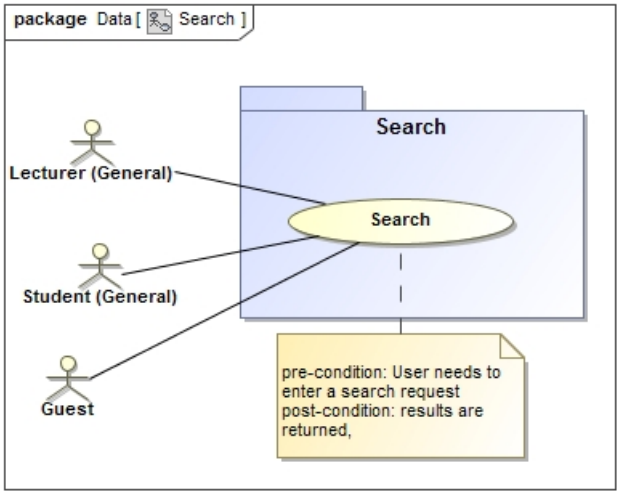


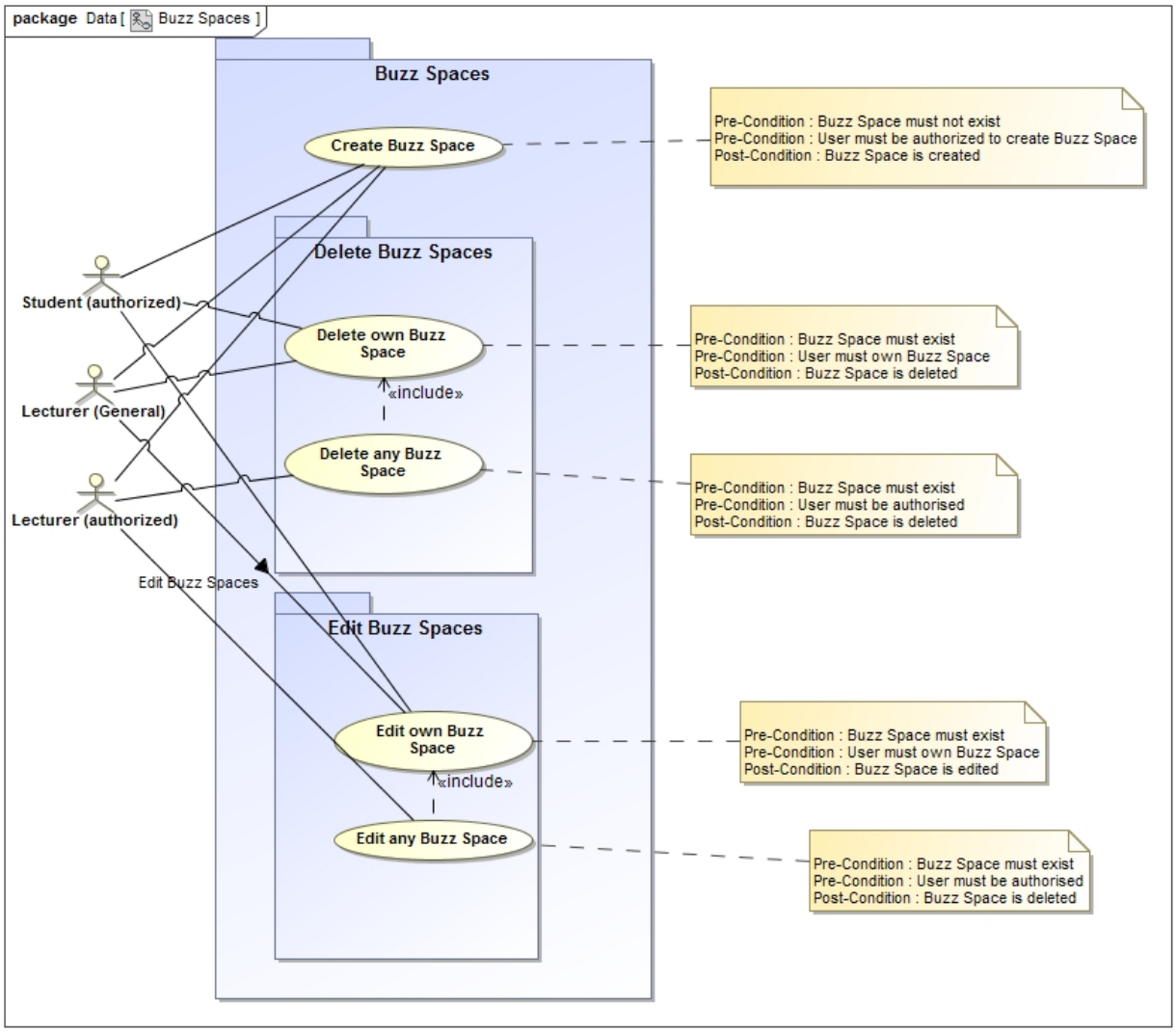



package Data [Request]
Request
make request
Student (Authorized)
Lecturer
Pre-condition:
must be logged in.
Post-condition:
Administrator will
review request.

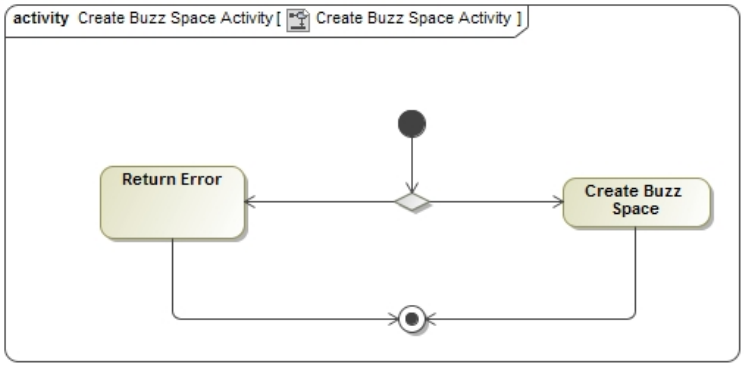

package Data [Post]
Post
create post
vote post
edit post
view post
Delete post
plaigerism check
authorization check
«include»
«include»
«include»
«include»
«include»
Lecturer (General)
Student (Authorized)
Guest
Pre-Condition: Must be logged in.
Pre-condition: Thread shuld exist already.
Post-conditions: Post is created.
Pre-condition: must have made a post.
Post-condition: post is flagged and hidden or posted.
Pre-condition: Must not be logged in.
Pre-condition: User has not yet voted.
Post-condition: User voted.
Pre-condition: Post must exist.
Pre-condition: Must be authorized to do so.
Post-condition: if authorized, post is edited.
Pre-conditions: Must be logged in.
Pre-conditions: Must be authorized to delete post.
Pre-conditions: Post must exist.
Post-conditions: Post will be deleted.
Pre-conditions: Buzz Space, Thread, Post must exist.
Post-conditions:User may view posts.

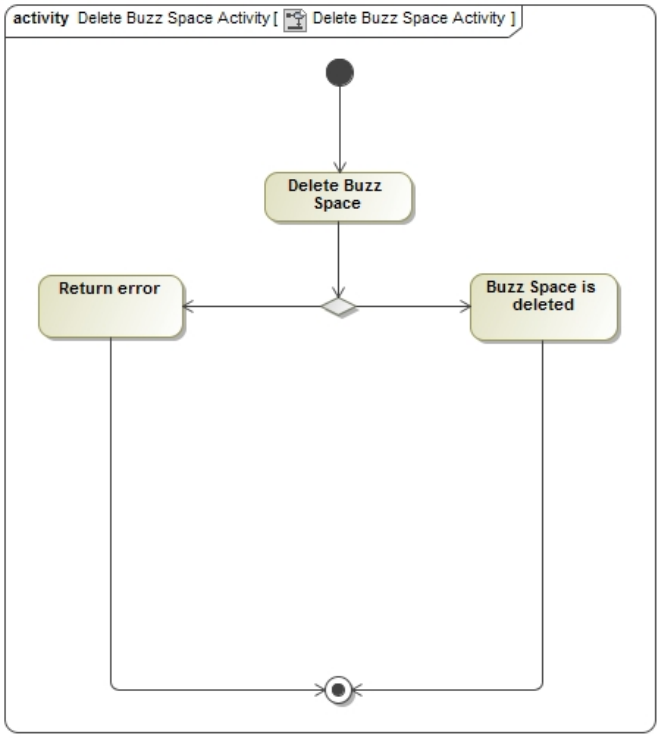

package Data [Voting]
Voting
Lecturer (General)
Vote
Student (General)
pre-condition: User must e logged in
pre-condition: User must not have voted already
post-condition: User has placed a vote.

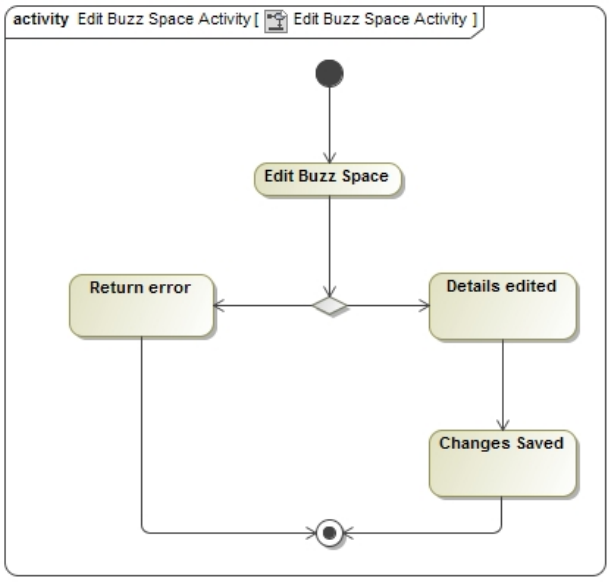

package Data [Search]
Search
Lecturer (General)
Search
Student (General)
Guest
pre-condition: User needs to enter a search request
post-condition: results are returned,

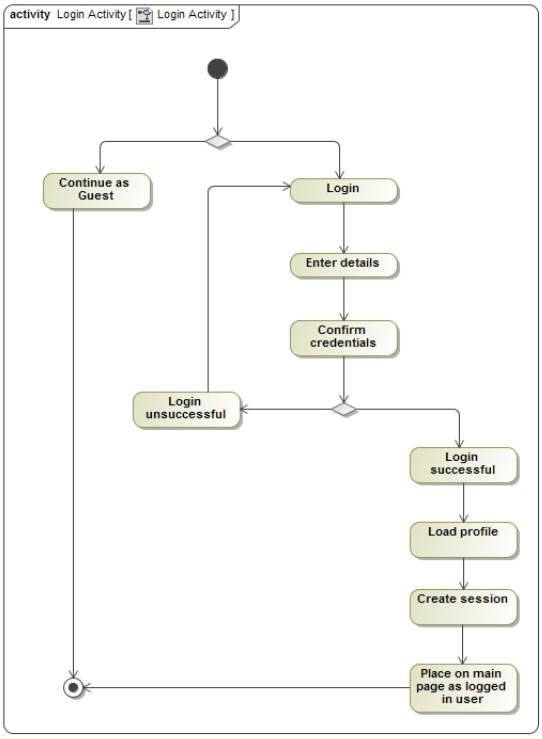

package Data [Buzz Spaces]
Buzz Spaces
Create Buzz Space
Pre-Condition : Buzz Space must not exist
Pre-Condition : User must be authorized to create Buzz Space
Post-Condition : Buzz Space is created
Delete Buzz Spaces
Student (authorized)
Delete own Buzz Space
Pre-Condition : Buzz Space must exist
Pre-Condition : User must own Buzz Space
Post-Condition : Buzz Space is deleted
«include»
Lecturer (General)
Delete any Buzz Space
Pre-Condition : Buzz Space must exist
Pre-Condition : User must be authorised
Post-Condition : Buzz Space is deleted
Lecturer (authorized)
Edit Buzz Spaces
Edit Buzz Spaces
Edit own Buzz Space
Pre-Condition : Buzz Space must exist
Pre-Condition : User must own Buzz Space
Post-Condition : Buzz Space is edited
«include»
Edit any Buzz Space
Pre-Condition : Buzz Space must exist
Pre-Condition : User must be authorised
Post-Condition : Buzz Space is deleted

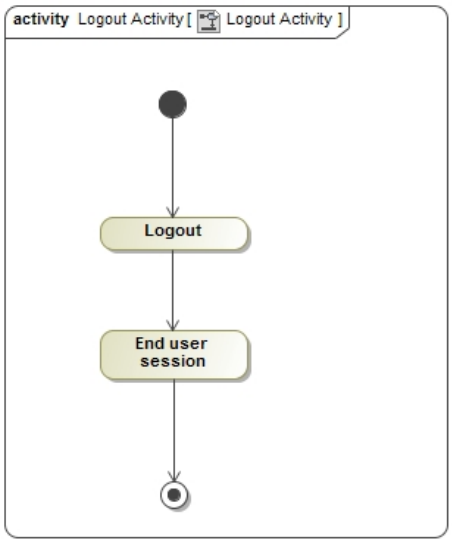
2.4 Process specifications

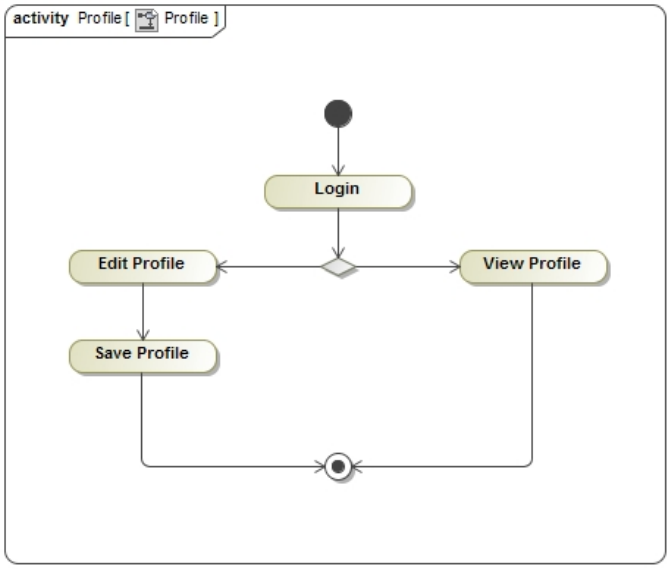


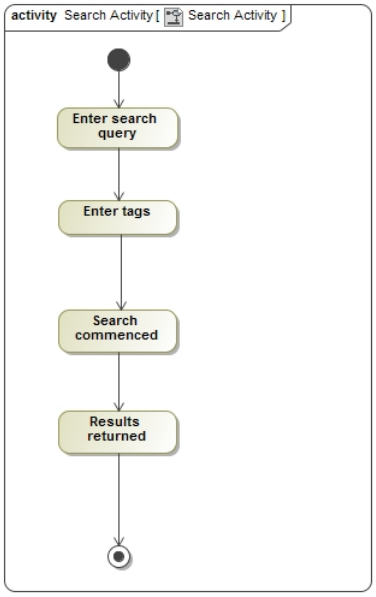


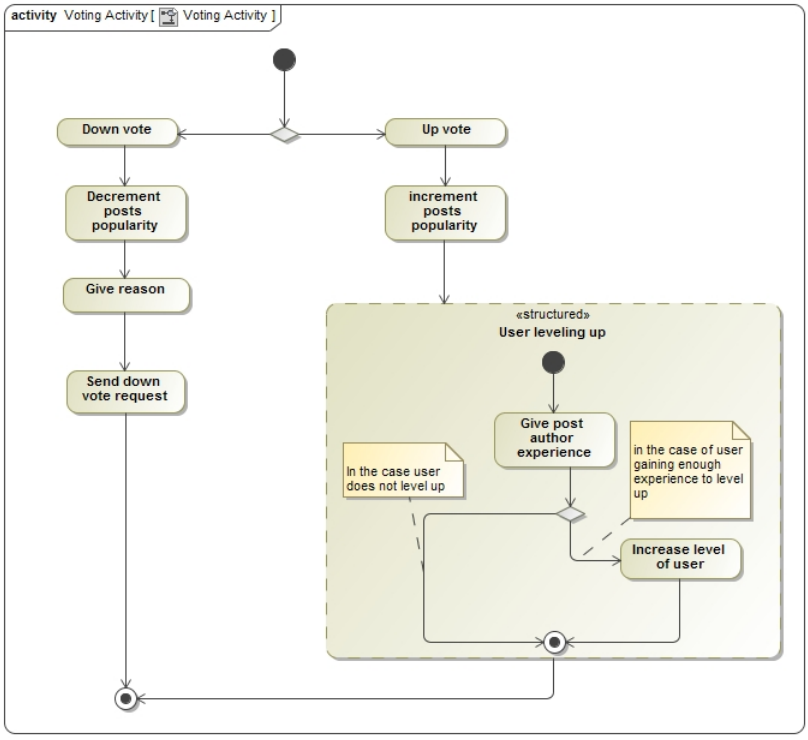

activity Edit Buzz Space Activity [Edit Buzz Space Activity]
Edit Buzz Space
Return error
Details edited
Changes Saved

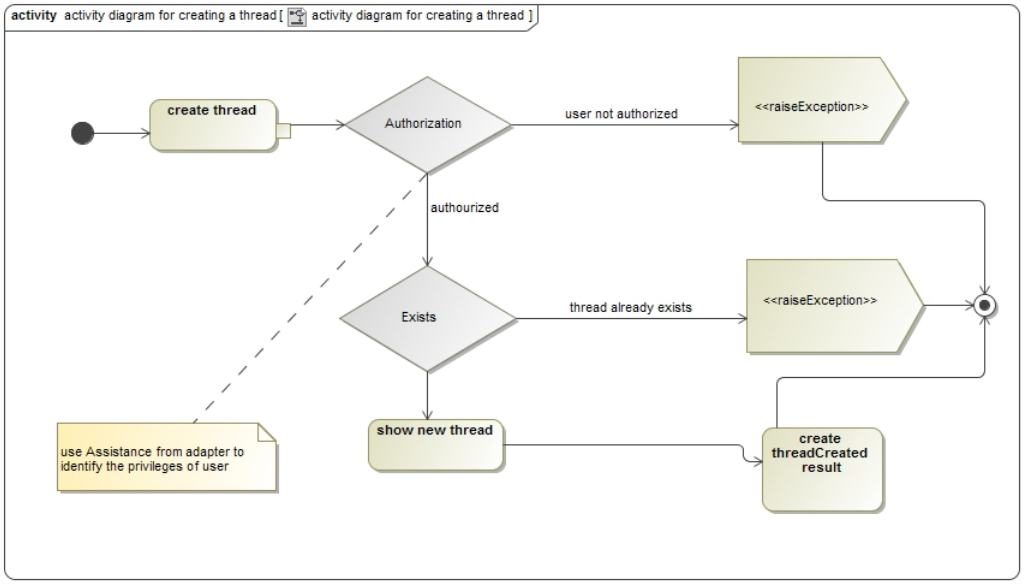

activity Login Activity [Login Activity]
Continue as Guest
Login
Enter details
Confirm credentials
Login unsuccessful
Login successful
Load profile
Create session
Place on main page as logged in user

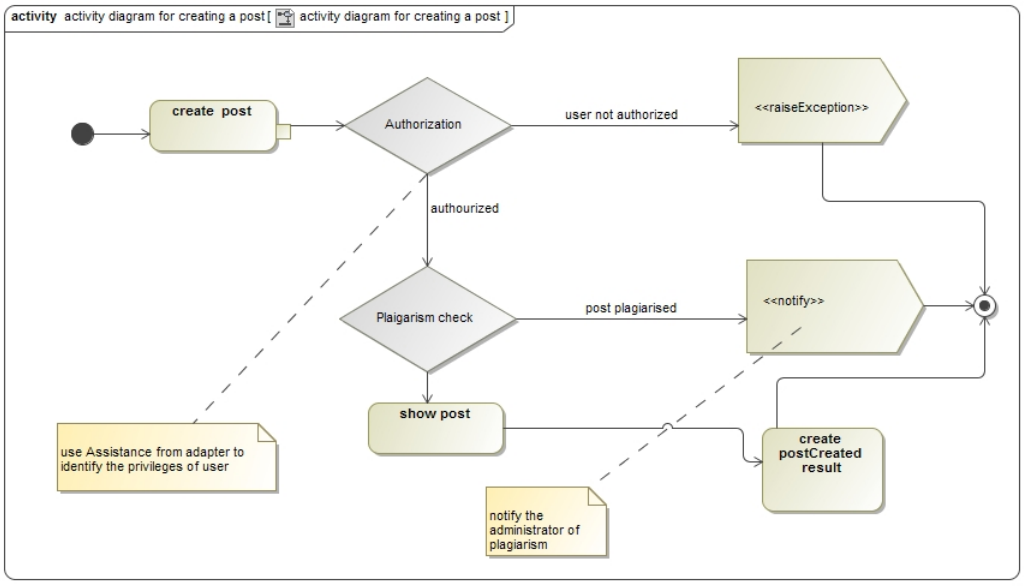

activity Logout Activity [Logout Activity]
Logout
End user session

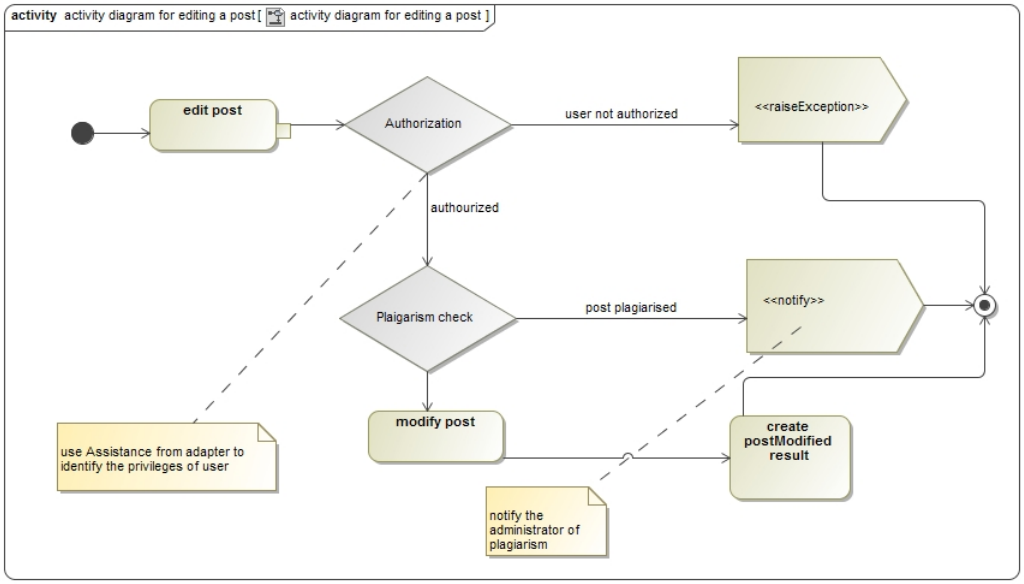

activity Profile [Profile]
Login
Edit Profile
View Profile
Save Profile

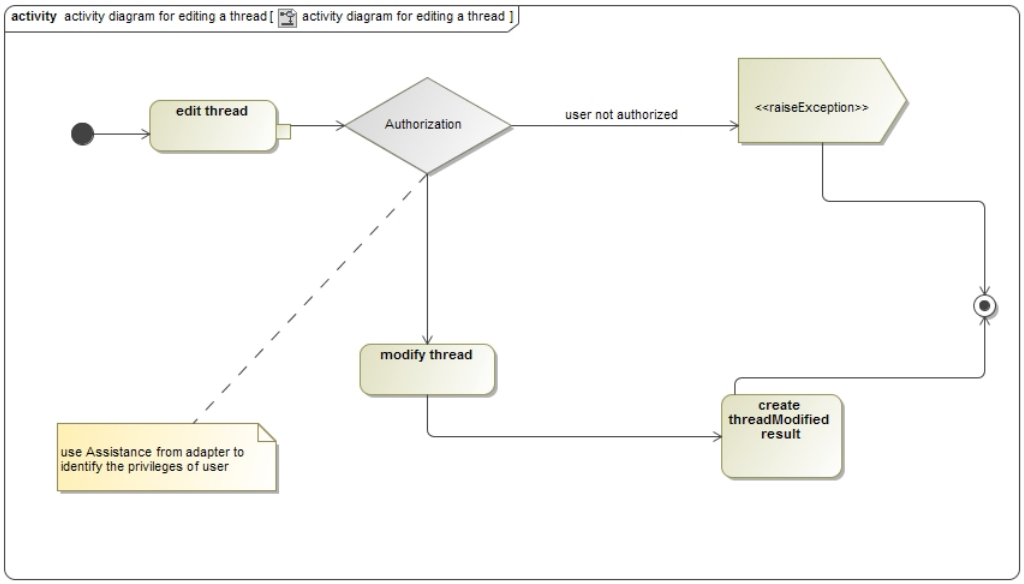

activity Search Activity [Search Activity]
Enter search query
Enter tags
Search commenced
Results returned

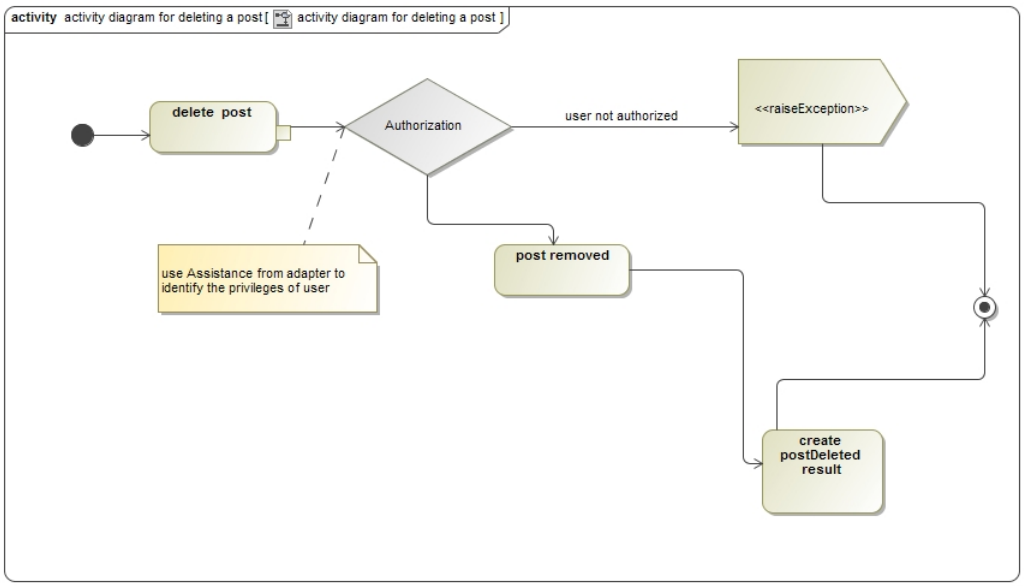

activity Voting Activity [Voting Activity]
Down vote
Up vote
Decrement posts popularity
increment posts popularity
Give reason
Send down vote request
«structured»
User leveling up
Give post author experience
In the case user does not level up
in the case of user gaining enough experience to level up
Increase level of user

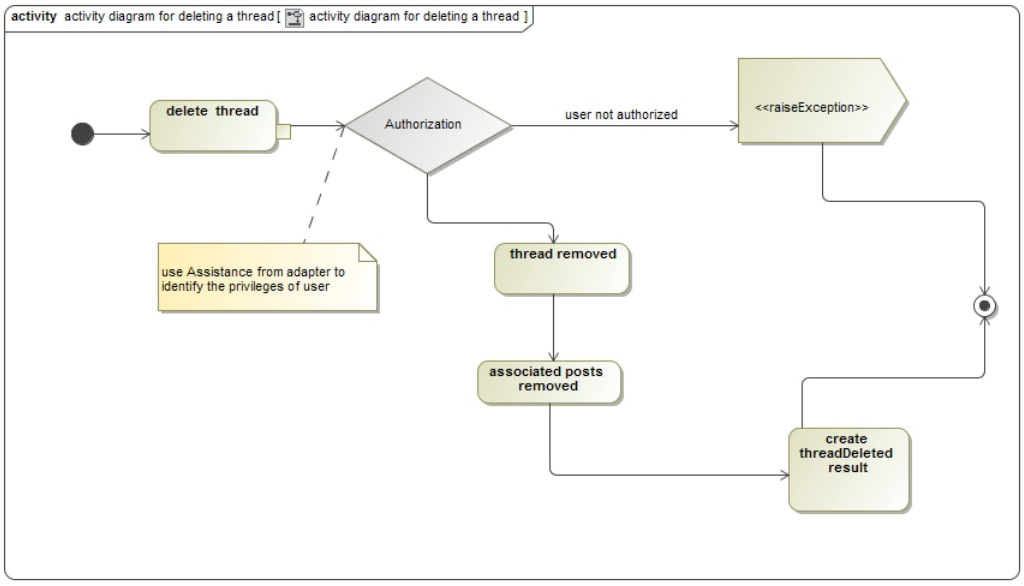

activity activity diagram for creating a thread [activity diagram for creating a thread]
create thread
Authorization
user not authorized
<<raiseException>>
authourized
Exists
thread already exists
<<raiseException>>
use Assistance from adapter to identify the privileges of user
show new thread
create threadCreated result

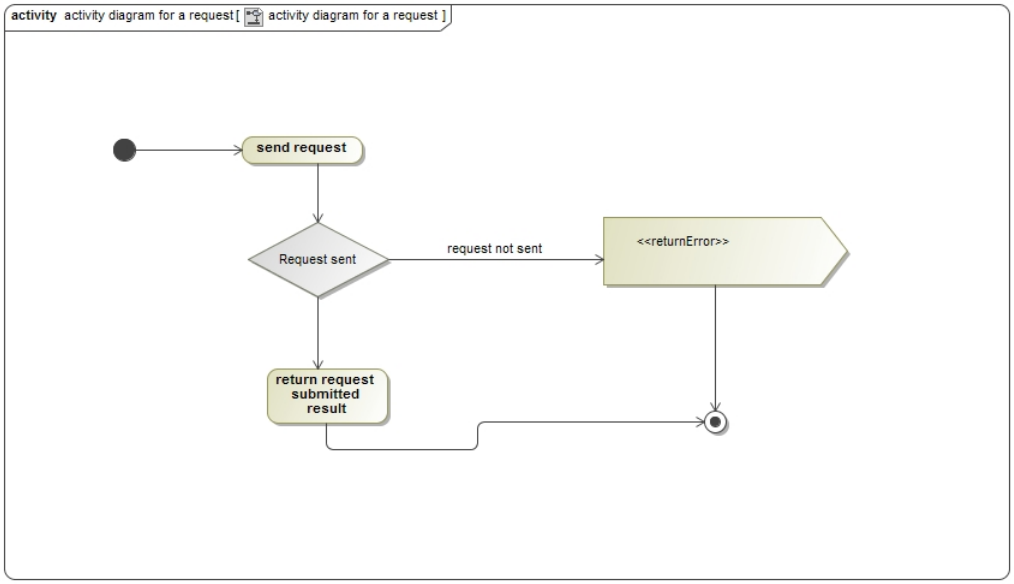
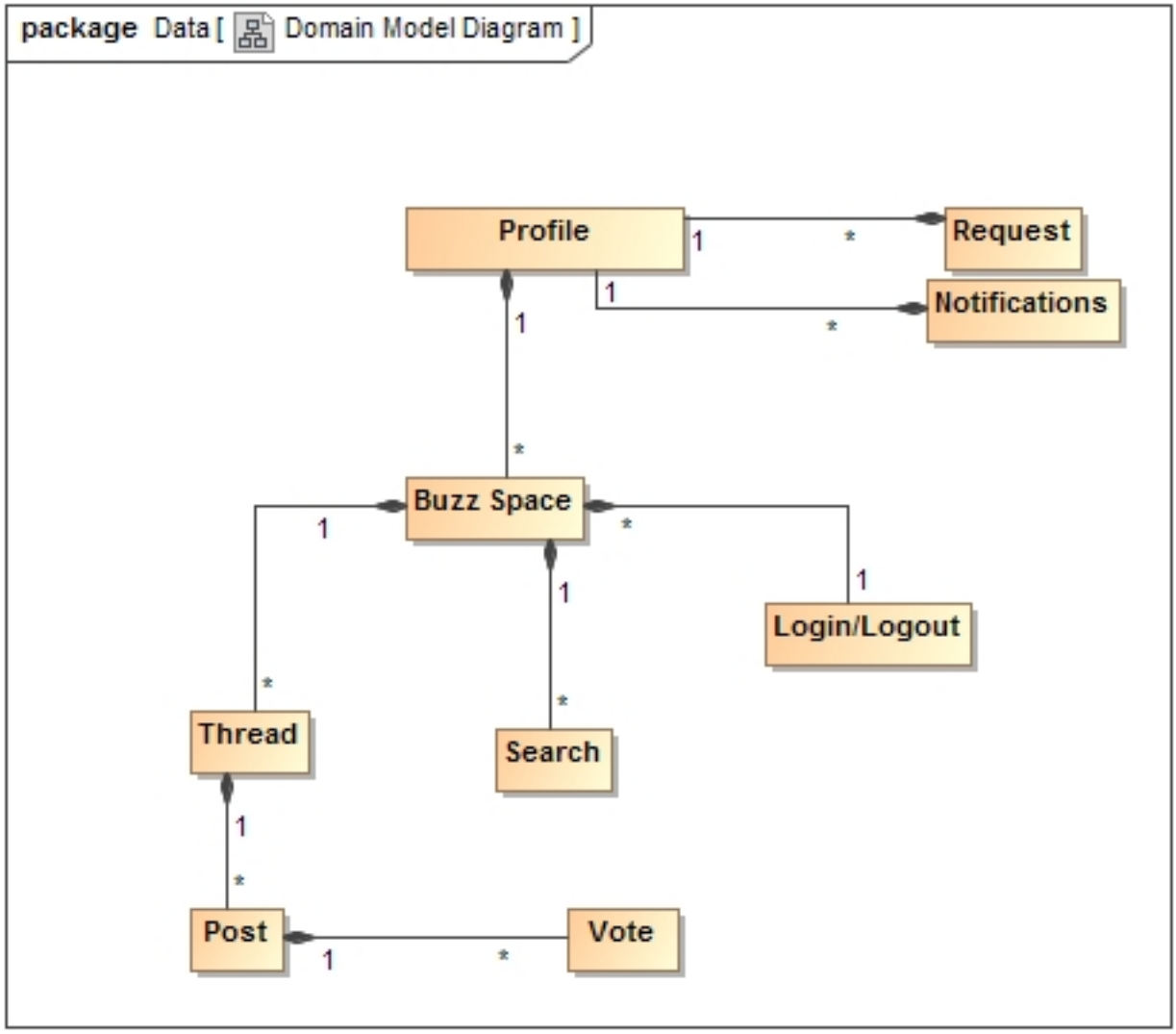

activity activity diagram for creating a post [activity diagram for creating a post]
create post
Authorization
user not authorized
<<raiseException>>
authourized
Plaigarism check
post plagiarised
<<notify>>
use Assistance from adapter to identify the privileges of user
show post
create postCreated result
notify the administrator of plagiarism


activity activity diagram for editing a post [activity diagram for editing a post]
edit post
Authorization
user not authorized
<<raiseException>>
authourized
Plaigarism check
post plagiarised
<<notify>>
modify post
create postModified result
use Assistance from adapter to identify the privileges of user
notify the administrator of plagiarism


activity activity diagram for editing a thread [activity diagram for editing a thread]
edit thread
Authorization
user not authorized
<<raiseException>>
modify thread
create threadModified result
use Assistance from adapter to identify the privileges of user


activity activity diagram for deleting a post [activity diagram for deleting a post]
delete post
Authorization
user not authorized
<<raiseException>>
use Assistance from adapter to identify the privileges of user
post removed
create postDeleted result


activity activity diagram for deleting a thread [activity diagram for deleting a thread]
delete thread
Authorization
user not authorized
<<raiseException>>
use Assistance from adapter to identify the privileges of user
thread removed
associated posts removed
create threadDeleted result


activity activity diagram for a request [activity diagram for a request]
send request
Request sent
request not sent
<<returnError>>
return request submitted result

2.5 Domain Model