FUNCTIONAL REQUIREMENTS SPECIFICATION

FOR

Buzz System

Mini Project Phase 1
GitHub

`https://github.com/RenaldoV/COS301_Group5_a`

Khathutshelo Matidza 11072157   Renaldo van Dyk 12204359
Andreas du Preez 12207871   Sean Hill 12221458
Kgomotso Sito 12243273   Hlavutelo Maluleke 12318109
Sboniso Masilela 10416260   Semaka Malapane 13081129

Contents

1 Introduction

This document discusses the application functionality required by users (and other stake-hodlers).

2 Searching and Filtering

2.1 Use case prioritization

2.1.1 Critical

1.1 Search
1.1.1 Search for student
1.1.1.1 Search by name
1.1.1.2 Search by student number
1.1.2 Search for posts
1.1.2.1 Search posts by title

2.1.2 Important

1.2 Filter posts
1.2.1 Filter by title
1.2.2 Filter by module code

2.1.3 Nice-To-Have

1.2.3 Filter by user
1.2.3 Filter by date

2.2 Use case/Services contracts

2.2.1 Pre-Conditions
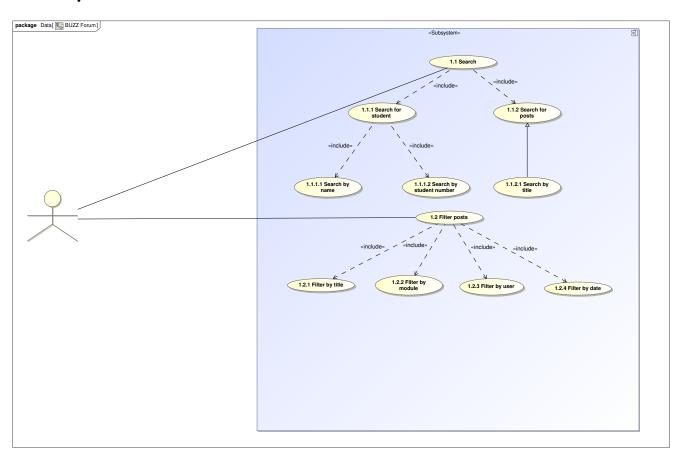
1.1 Search text is invalid
1.1.1 Student does not exist in system
1.1.1.1 Student of specified name does not exist in the system
1.1.1.2 Student of specified student number does not exist in system
1.1.2 Post does not exist
1.1.2.1 Post of specified title does not exist

1.2 a) Criteria to filter is invalid
b) No posts available to filter

1.2.1 Invalid title specified

1.2.2 Module code does not exist

1.2.3 a) Incorrect user specified b) User does not exist

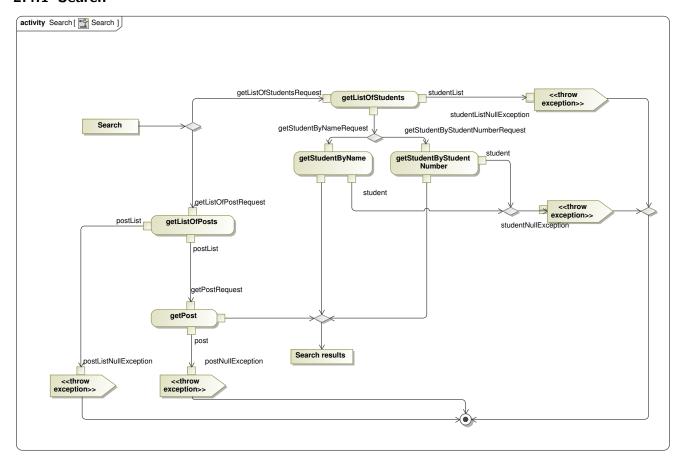1.2.4 a) Date incorrectly specified

2.2.2 Post-Conditions

1.1 Search results returned

1.1.1 Student details of student searched for are returned

1.1.1.1 Student details of student number specified are returned

1.1.2 Post searched for is returned

1.1.2.1 Post of specified title is returned

2.3 Request and Results Data Structures
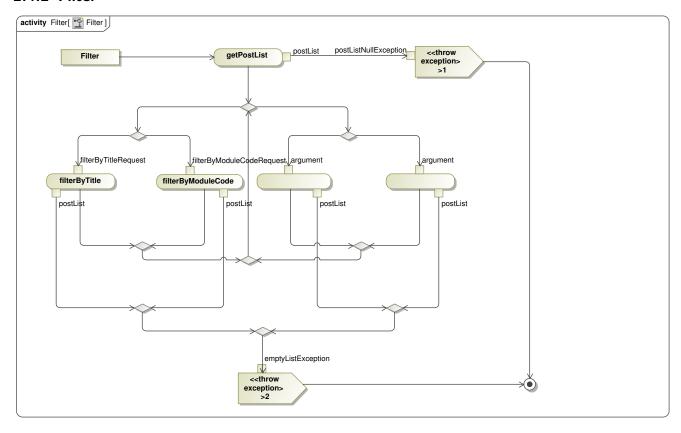
2.4 Process specifications
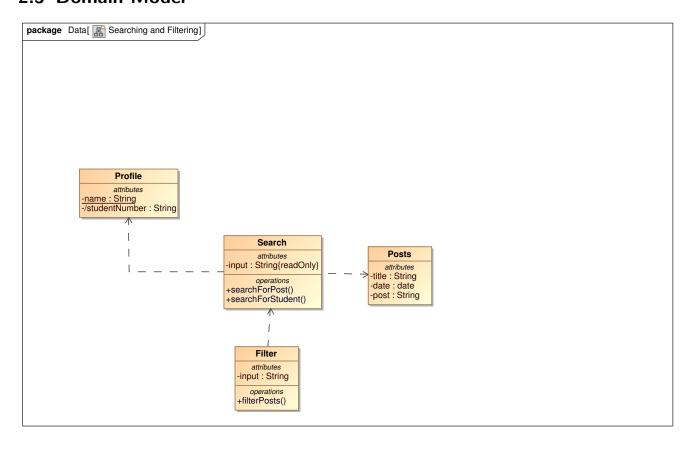
2.4.1 Search

2.4.2 Filter

2.5 Domain Model

package Data[🔲 Searching and Filtering]

Profile

attributes
-name : String
-/studentNumber : String

Search

attributes
-input : String{readOnly}

operations
+searchForPost()
+searchForStudent()

Posts

attributes
-title : String
-date : date
-post : String

Filter

attributes
-input : String

operations
+filterPosts()

3 Rating

3.1 Use case prioritization

3.1.1 Critical

1.2 Rate
1.1.1 Like comment
1.1.2 Dislike comment
1.1.3 Store rating

3.1.2 Important

3.1.3 Nice-To-Have

3.2 Use case/Services contracts
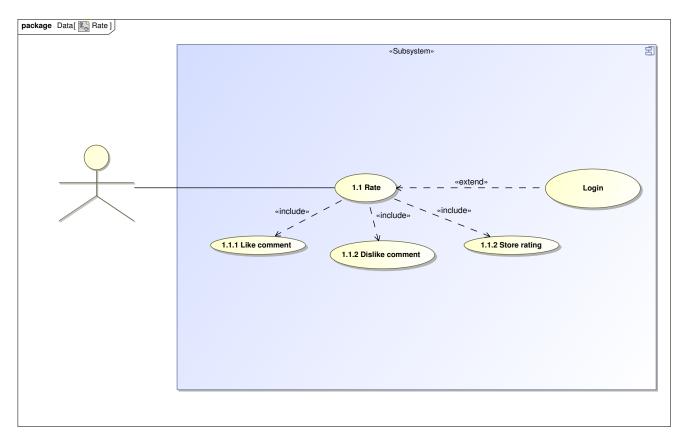
3.2.1 Pre-Conditions

1.2 User not logged in 1.1.1 Post does not exist 1.1.2 Post does not exist 1.1.3 Rating invalid
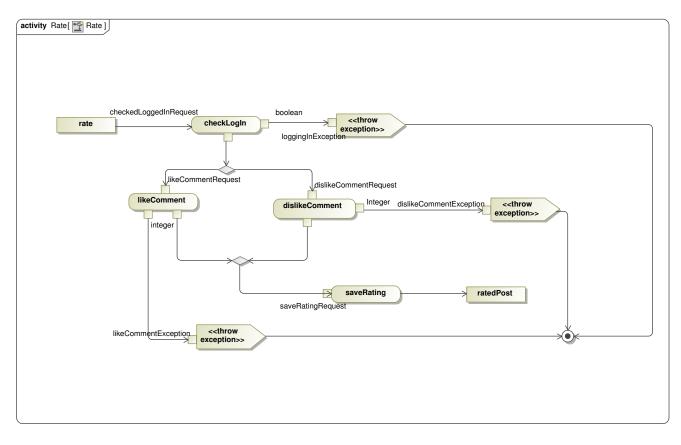
3.2.2 Post-Conditions

1.2 User not logged in 1.1.1 Post does not exist 1.1.2 Post does not exist 1.1.2 Rating invalid

3.3 Request and Results Data Structures

3.4 Process specifications

activity Rate[Rate]

rate —checkedLoggedInRequest→ **checkLogIn** —boolean→ <<throw exception>>

loggingInException

checkLogIn → likeCommentRequest → **likeComment**

dislikeCommentRequest → **dislikeComment** —Integer— dislikeCommentException → <<throw exception>>

likeComment → integer

saveRating saveRatingRequest → **ratedPost**

likeCommentException → <<throw exception>>

3.5 Domain Model

package Data[🔲 Rating]

```
                              ┌──────────────┐
                              │    Login     │
                              └──────────────┘
                                     ▲
                                     ┊
                                     ┊
    ┌────────────────────────┐      ┌──────────────────┐
    │        Profile         │      │       Rate       │
    ├────────────────────────┤      ├──────────────────┤
    │       attributes       │      │    attributes    │
    │ -name : String         │◇─────│ -user : Profile  │
    │ -/studentNumber : String│ userProfile│ -rating : Integer│
    └────────────────────────┘      └──────────────────┘
```