BCC Final Iteration

Mark Fuller
Kevin Kulda
Connor Woodahl

Dr. Cerny
Software Engineering 1
04-DEC-2019

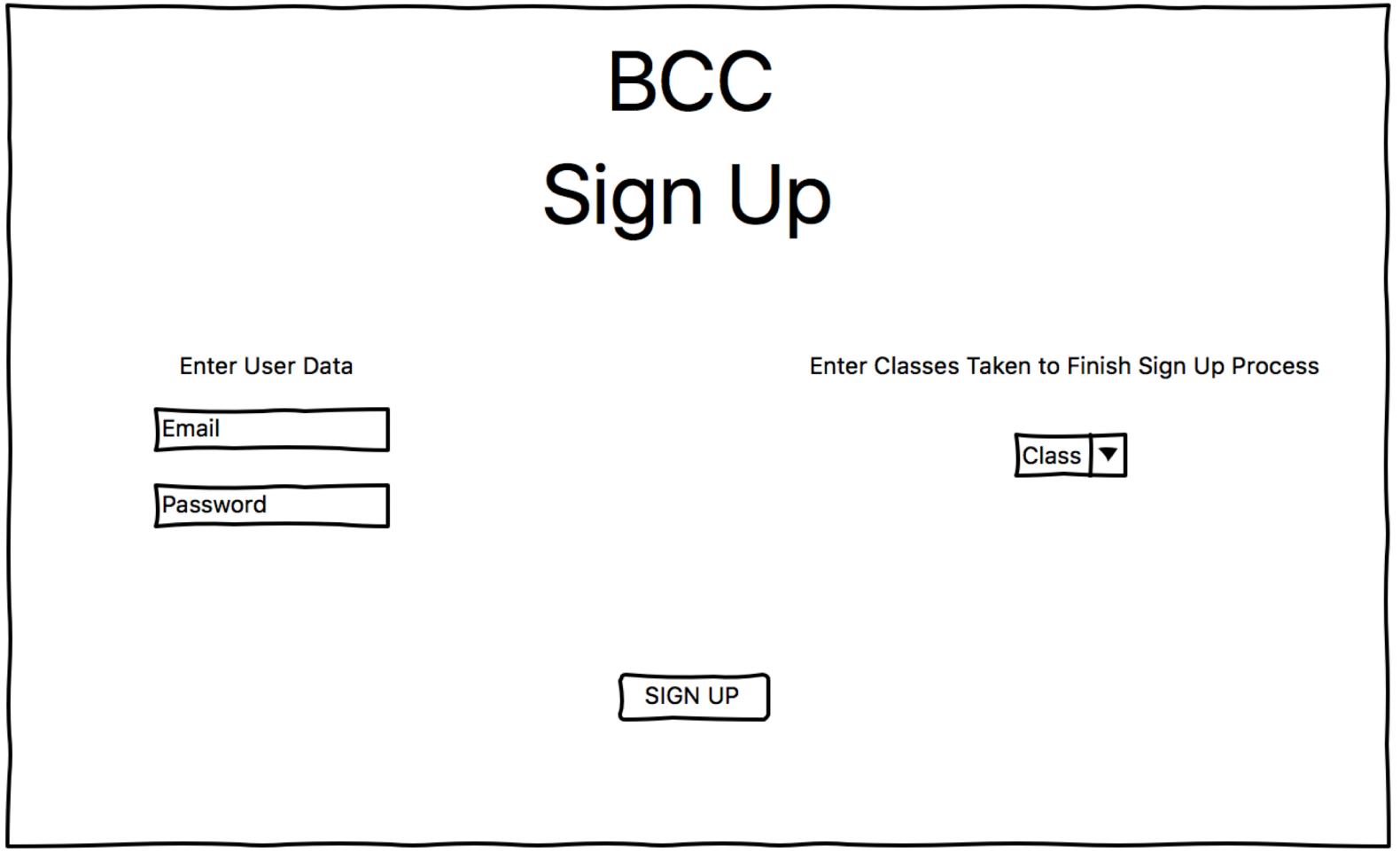
Project Repository: https://github.com/MarkFuller1/BNU

BCC Iteration 1

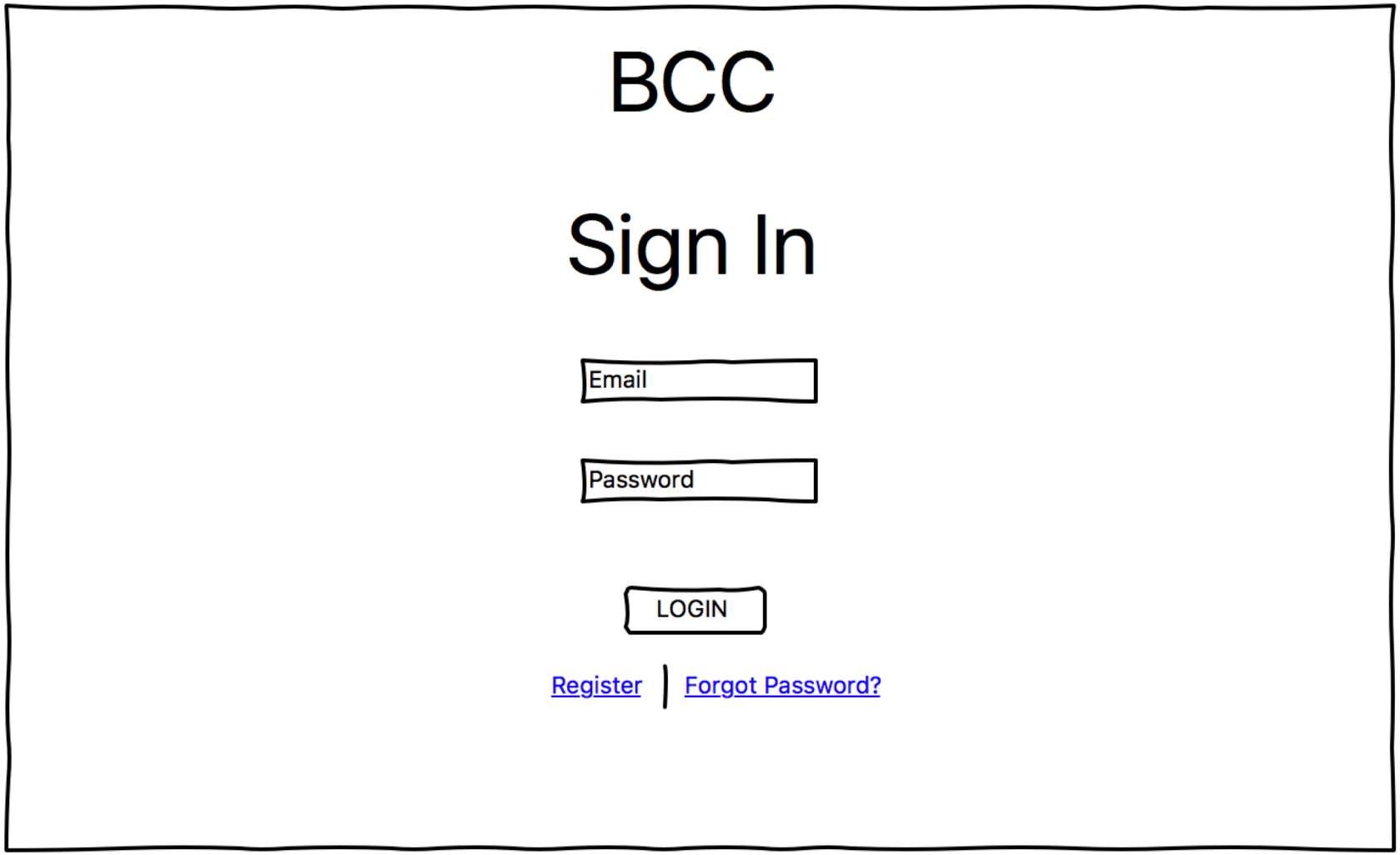
Baylor Class Connect

- BCC is a tool used by students to rate their current teachers, view reviews of future teachers and contact previous students who havehad those teachers.

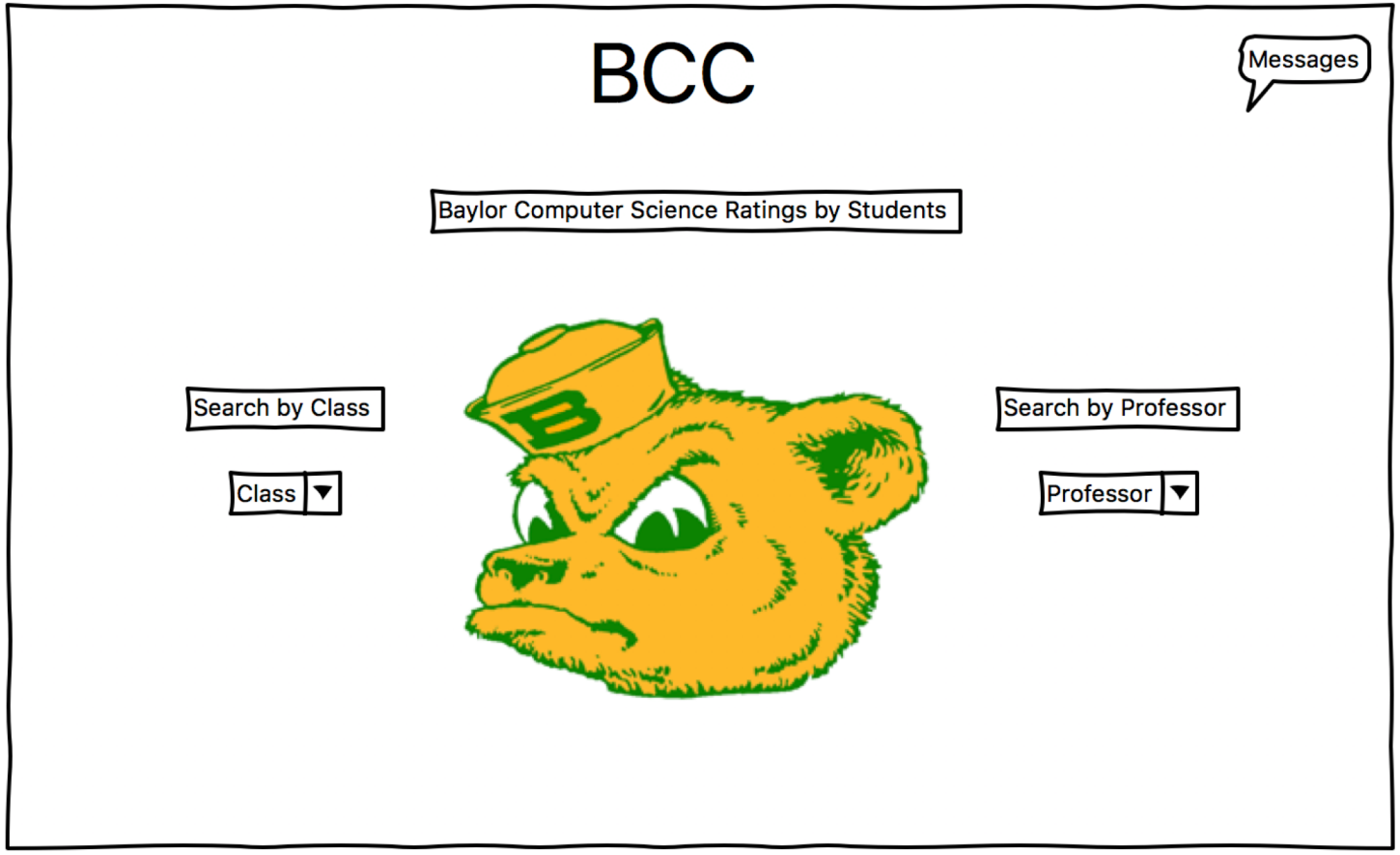
Wireframes


BCC
Sign Up
Enter User Data
Email
Password
Enter Classes Taken to Finish Sign Up Process
Class
SIGN UP

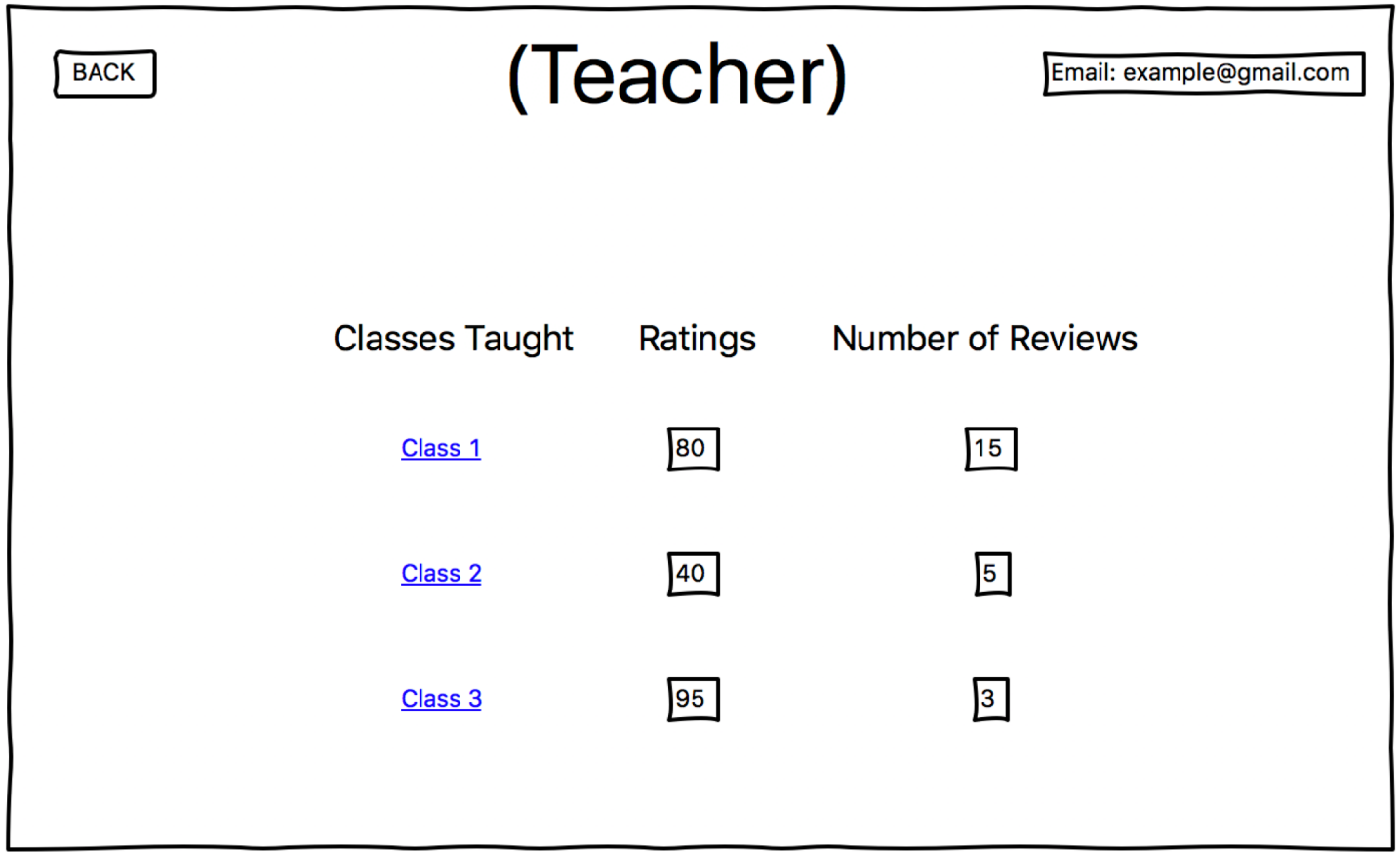
Wireframes



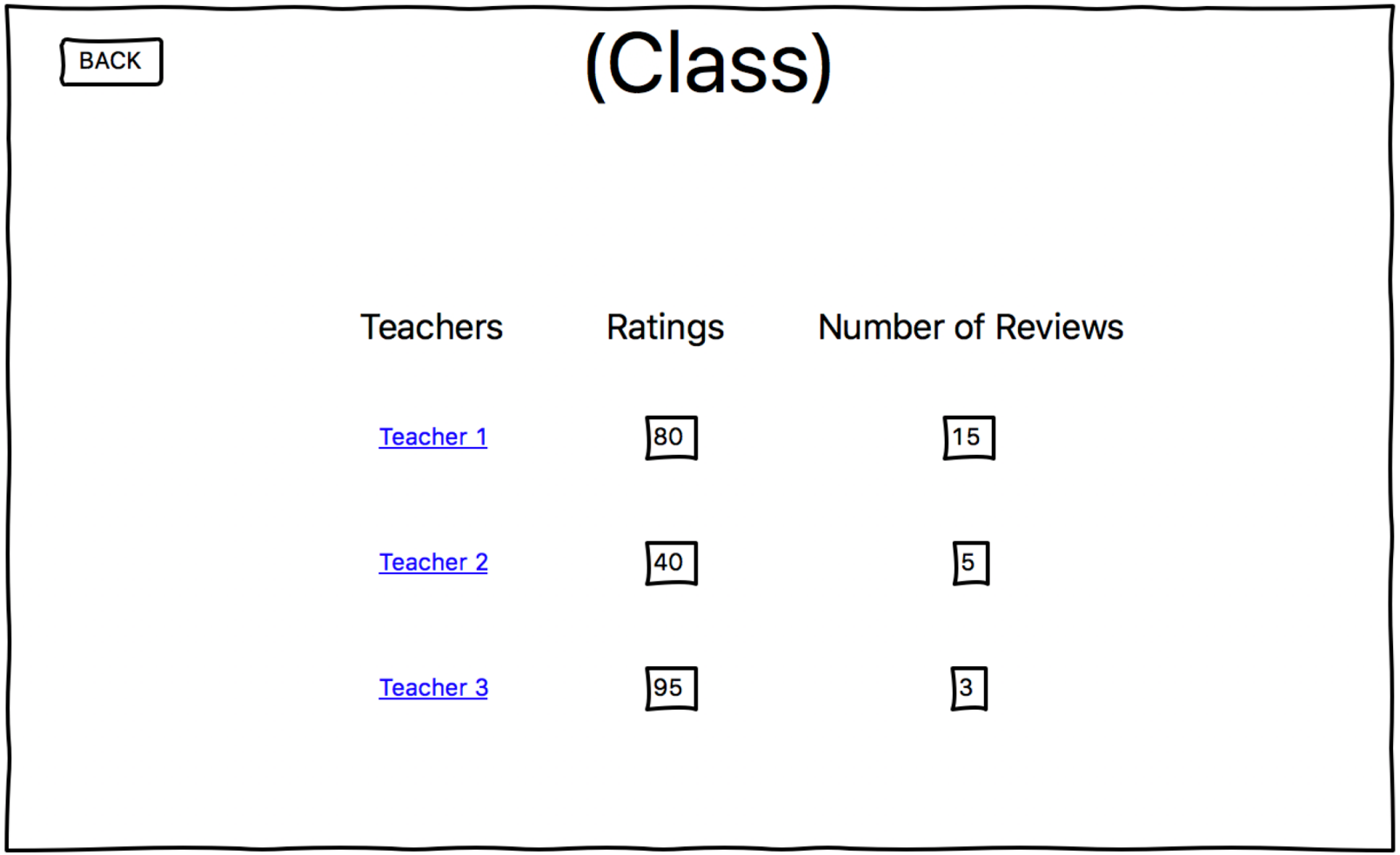
Wireframes


BCC
Messages
Baylor Computer Science Ratings by Students
Search by Class
Class
Search by Professor
Professor

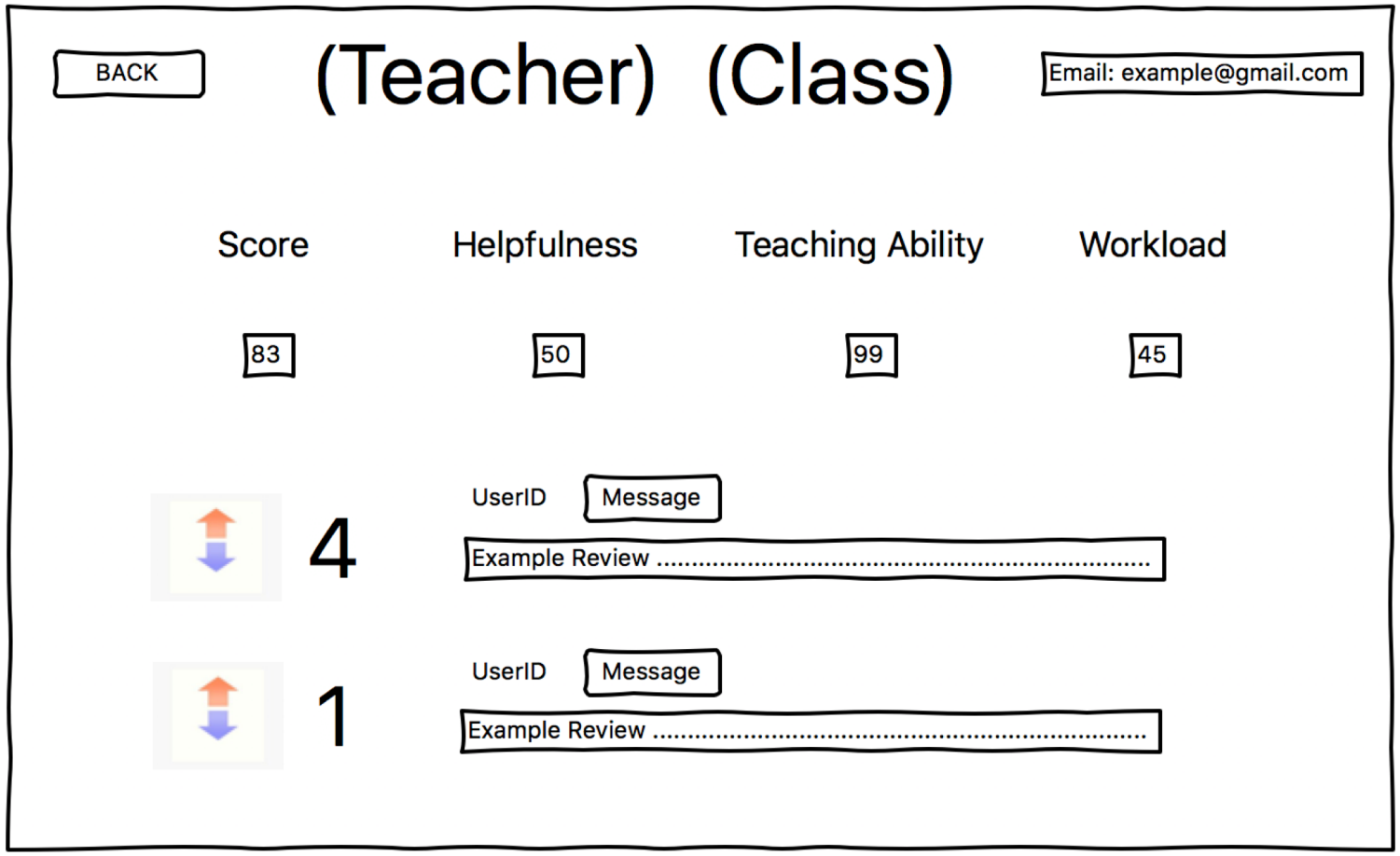
Wireframes


BACK
(Teacher)
Email: example@gmail.com
Classes Taught
Ratings
Number of Reviews
Class 1
80
15
Class 2
40
5
Class 3
95
3

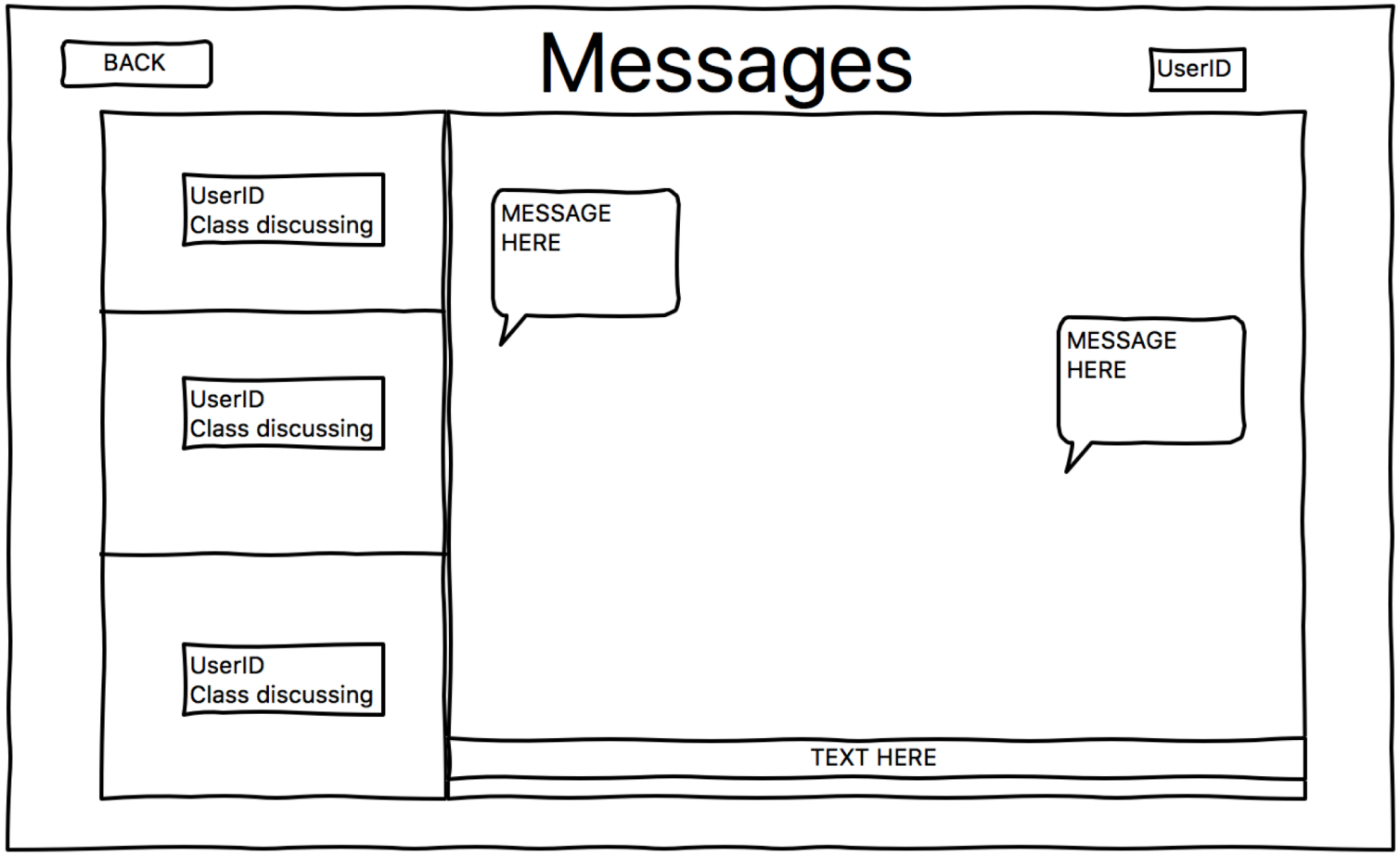
Wireframes


BACK
(Class)
Teachers
Ratings
Number of Reviews
Teacher 1
80
15
Teacher 2
40
5
Teacher 3
95
3

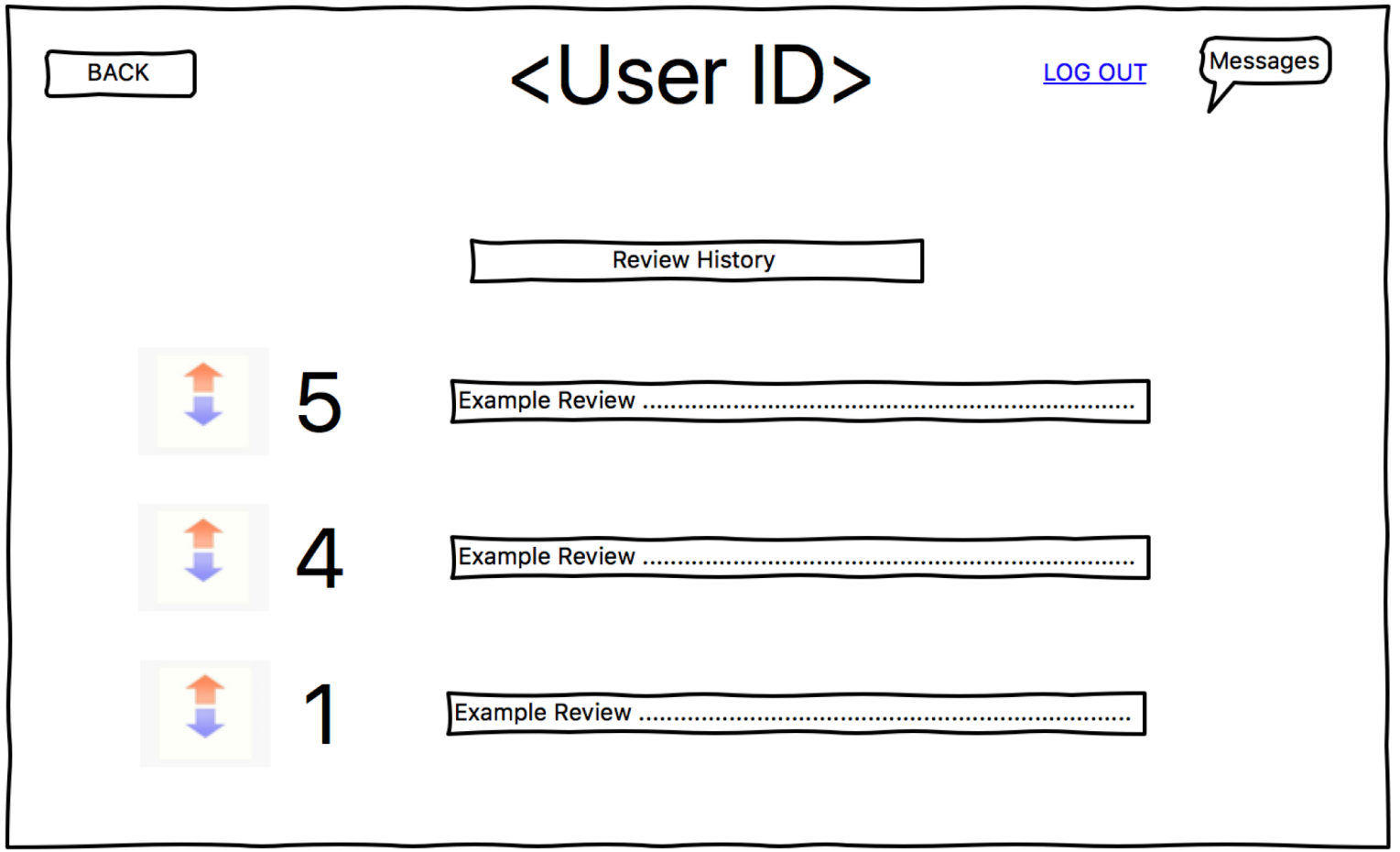
Wireframes


BACK
(Teacher) (Class)
Email: example@gmail.com
Score
Helpfulness
Teaching Ability
Workload
83
50
99
45
4
UserID
Message
Example Review .....
1
UserID
Message
Example Review .....

Wireframes


BACK
Messages
UserID
UserID
Class discussing
UserID
Class discussing
UserID
Class discussing
MESSAGE
HERE
MESSAGE
HERE
TEXT HERE

Wireframes


BACK
<User ID>
LOG OUT
Messages
Review History
5
Example Review .....
4
Example Review .....
1
Example Review .....

Requirements

Functional Requirements

- Users need to be able to register an account.
- Users must be able to search based on class.
- Users must be able to see professor options after searching the class.
- Users must be able to search based on professor.
- Users must be able to write a review on a professor.
- Users must be able to upvote or downvote a review.
- Users must be able to send a message to the author of the review.
- The reviewer must be able to respond to a message.
- User must be able to log out.
- The application must be able to sanitize user input.
- User must be able to navigate bidirectionally in this application.
- Users need to be able to delete an account.
- Users must be able edit a review.

Non-Functional Requirements

- Messages must be able to be sent anonymously
- User data must be kept private
- Application must be fast for the average user
- Application must handle traffic of at most 20 users at once
- User interface should be intuitive
- User interface should be graphically pleasing

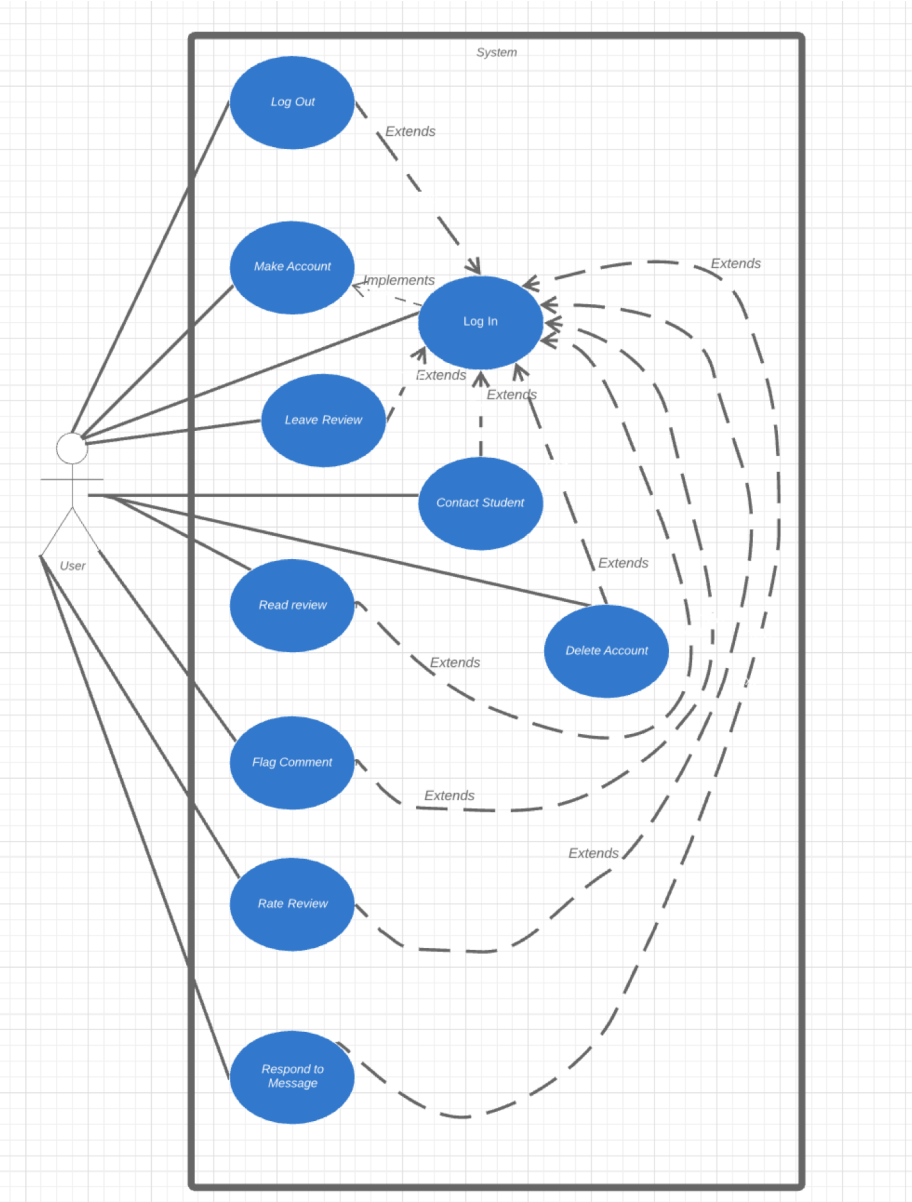
Actors

User

Admin

System

Use Case Diagram



Use Case Division

- Use Cases handled by Kevin– Make Account, Respond to a Message, Flag Comment
- Use Cases handled by Mark – Leave Review, Rate Review, Read Review
- Use Cases handled by Connor – Contact Student, Edit Review, Delete Account, Log Out

Make Account

Make an Account

Scope: GCC Application

Level: User Goal

Primary Actor: Student

Stakeholders and Interests:

Student User: Wants username and password stored and hashed for security.

Preconditions: Student has loaded the site

Success Guarantee: Students data is hashed and saved into database. Immediate access to the application is granted.

Main Success Scenario:

Student loads application.

Student requests a new account.

Student enters Username, Password and, email, etc.

System updates valid students list with new student.

Student Logs in.

Alternative Flows:

a. At any time the student closes the application.

1. Interaction ends.

b. At any time system fails.

1. Student will restart application.

c. While student is on register page, they would like to login with valid credentials(the realized they had an account already)

1. Student clicks `return to login` and logs in with valid username and password.

d. Username and password from account creation do not work.

1. Prompt student to reset password with security question.

1.1. Student answers security question correctly

1.1.1. Student resets password.

1.2. Student does not answer security question correctly

1.2.1 Student is returned to login page.

Leave Review

Leave Review

Scope: GCC Application

Level: User Goal

Primary Actor: Student

Stakeholders and Interests:

Student User:

Wants Username and Password to be recognized.

Wants review to be saved for others to view

Student has valid login credentials

Preconditions:

Student has loaded the program.

Success Guarantee:

Student review is made available to other users on the site.

Main Success Scenario:

Student loads application.

Student logs in.

Student selects professor they would like to review from drop-down menu

Student is brought to professors review page

Student writes review

Student saves review

Alternative Flows:

a. At any time the student closes the application.
1. Interaction ends.
b. At any time system fails.
1. Student will restart application.
c. Student picks wrong professor
1. Student clicks `back`
2. Selects correct professor from drop-down list
d. Username and password do not work.
1. Prompt student to reset password with security question.
1.1. Student answers security question correctly
1.1.1. Student resets password.
1.2. Student does not answer security question correctly
1.2.1 Student is returned to login page.

Mark Fuller, Kevin Kulda, Connor Woodahl ---
https://github.com/markfuller/BNU

Contact Student

Contact Student

Scope: GCC Application

Level: User Goal

Primary Actor: Student

Stakeholders and Interests:

Student User:

Wants Username and Password to be recognized.

Wants review to be saved for others to view

Wants contact with other Student to be over message board

Preconditions:

Student has loaded the program.

Student has valid login credentials

Success Guarantee:

Student is able to message other Student user (two-way anonymity)

Main Success Scenario:

Student loads application.

Student logs in.

Student selects professor they would like to view reviews of

Student is brought to professors review page

Student selects, contact reviewer button

Student creates message

Student Sends message

Opposing student responds to message

Message Notification shows up on login of user

Alternative Flows:

a. At any time the student closes the application.
1. Interaction ends.
b. At any time system fails.
1. Student will restart application.
c. Student picks wrong professor
1. Student clicks `back`
2. Selects correct professor from drop-down list
d. Username and password do not work.
1. Prompt student to reset password with security question.
1.1. Student answers security question correctly
1.1.1. Student resets password.
1.2. Student does not answer security question correctly
1.2.1 Student is returned to login page.

Mark Fuller, Kevin Kulda, Connor Woodahl ---
https://github.com/markfuller/BNU

Casual Use Cases

Respond to a Message

Main Success Scenario

The user clicks on the message board icon on the main page. Selecting the message they want to respond to, they use the messing capability of the application to respond to the message.

Alternative Success Scenario

Edit Review

Main Success Scenario

The user logs in and navigates to their profile where all their reviews are listed. Using the editor, they change the content of their review and save changes.

Alternative Success Scenario The user logs in and navigates to their review by selecting the professor and finding their review under the professors page. Using the editor, they change the content of their review and save changes.

Delete Account

Main Success Scenario

While the user is logged in, the user navigates to their user profile settings and select the delete account button. After confirming your selection they are logged out.

Alternative Success Scenario

Rate Review

Main Success Scenario

User logs in with valid credentials. Selects the professor and class to read reviews for. Clicking the up-vote button if the review is accurate will increase the reviews score.

Alternative Success Scenario

User logs in with valid credentials. Selects the professor and class to read reviews for. Clicking the down-vote button if the review is inaccurate will decrease the reviews score.

Flag Comment

Main Success Scenario

User logs in with valid credentials. Selects the professor and class to read reviews for. If a review is inappropriate or uses vulgar language, the users flags it.

Alternative Success Scenario

Read Review

Main Success Scenario

The user will login with valid credentials. Using the navigation page, selects a professor and the desired class to load reviews for. Reviews load in chronological order.

Alternative Success Scenario

User logs in with valid credentials. Using the navigation page, selects a class and the desired professor to load reviews for. Reviews load in chronological order.

Log Out

Main Success Scenario

While the user is logged in, the user navigates to their user profile settings and logs out.

Alternative Success Scenario

The user is automatically logged out when they close the application.

Mark Fuller, Kevin Kulda, Connor Woodahl ---
https://github.com/markfuller/BNU

Git Hub page

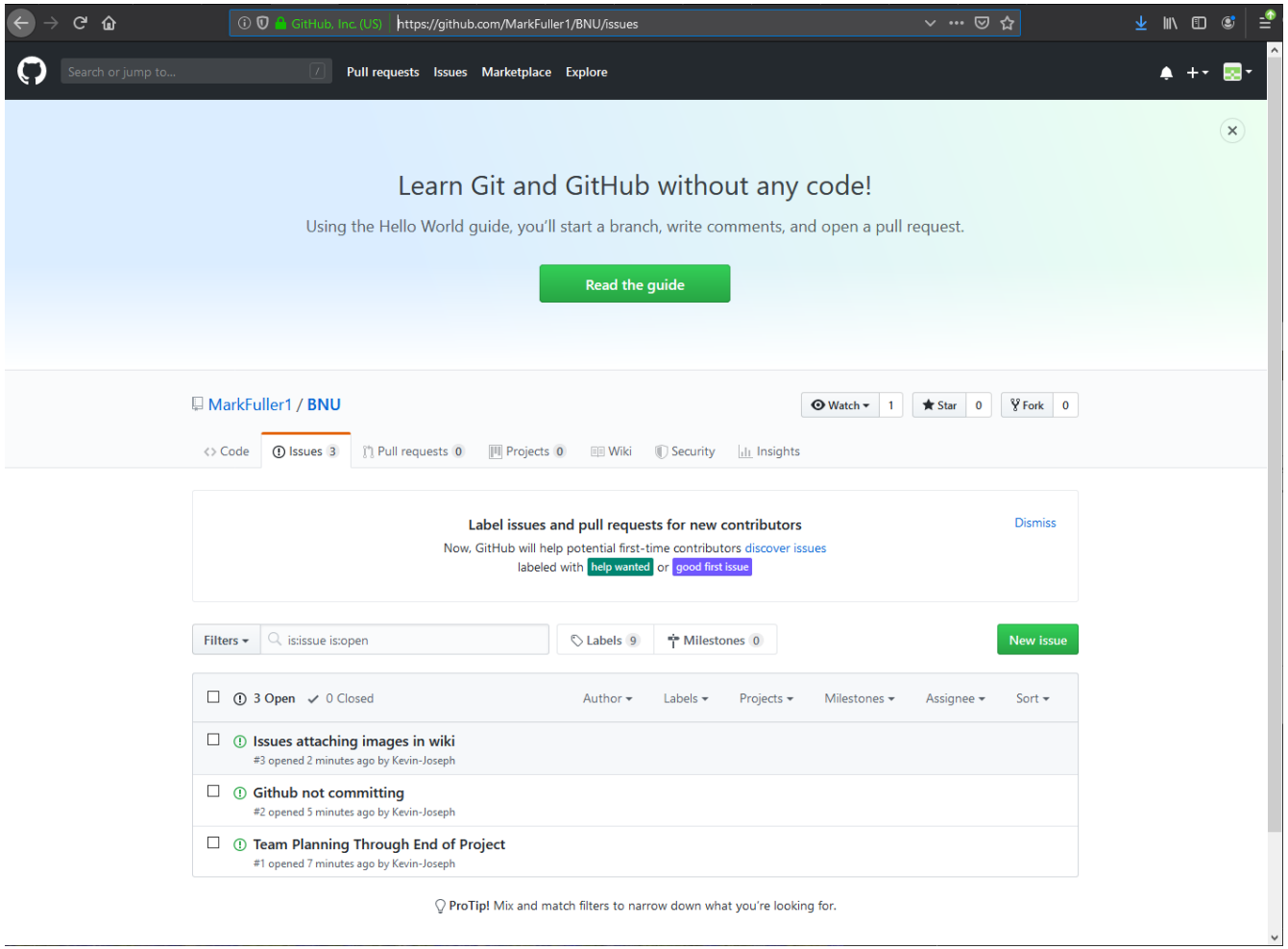
Please see our Github page for the current versions of our development materials and the full repository of our project artifacts.

https://github.com/MarkFuller1/BNU/

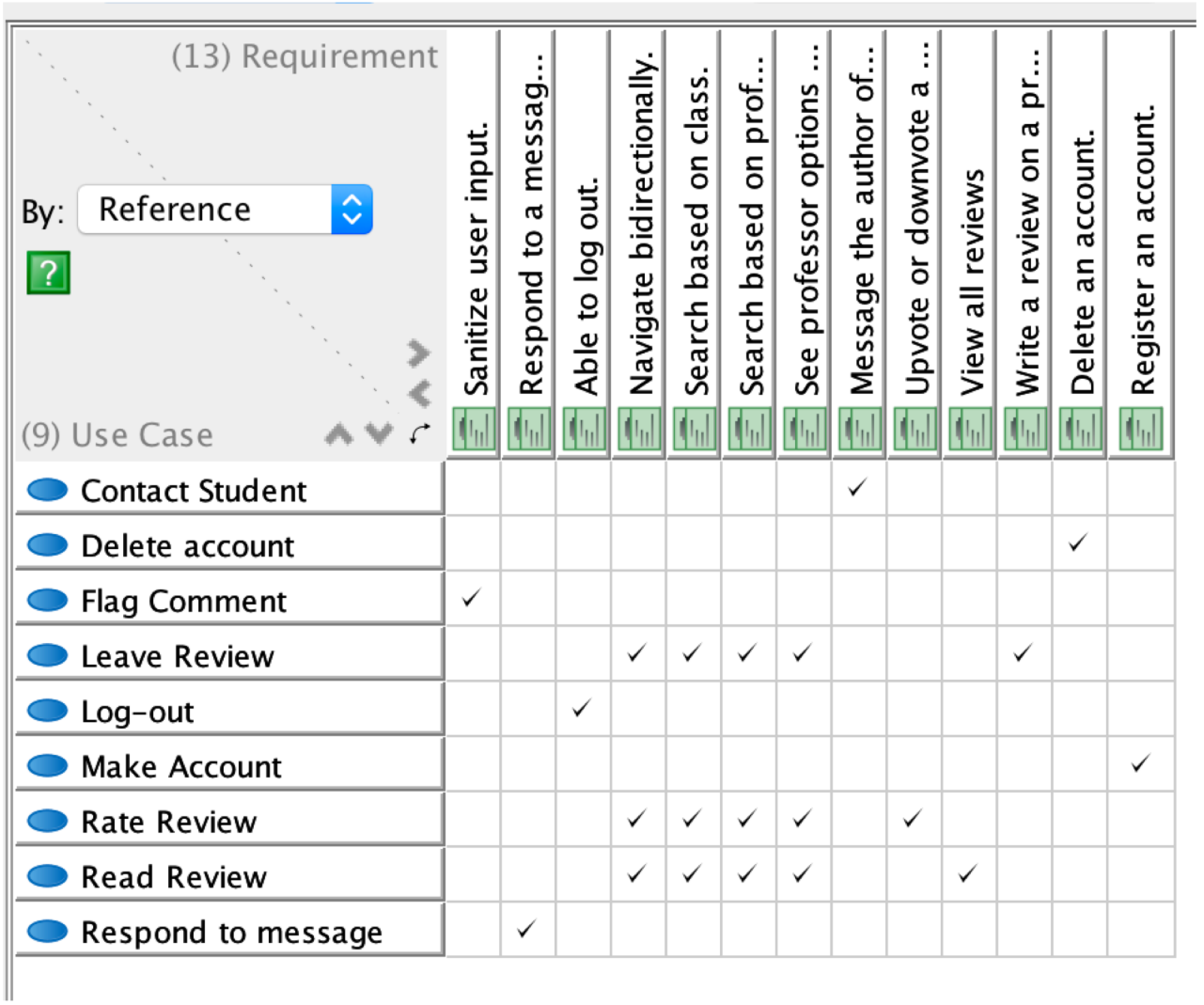
Issue Tracking

- Please see our issue tracking repository on Github at the link below:

https://github.com/MarkFuller1/BNU/issues

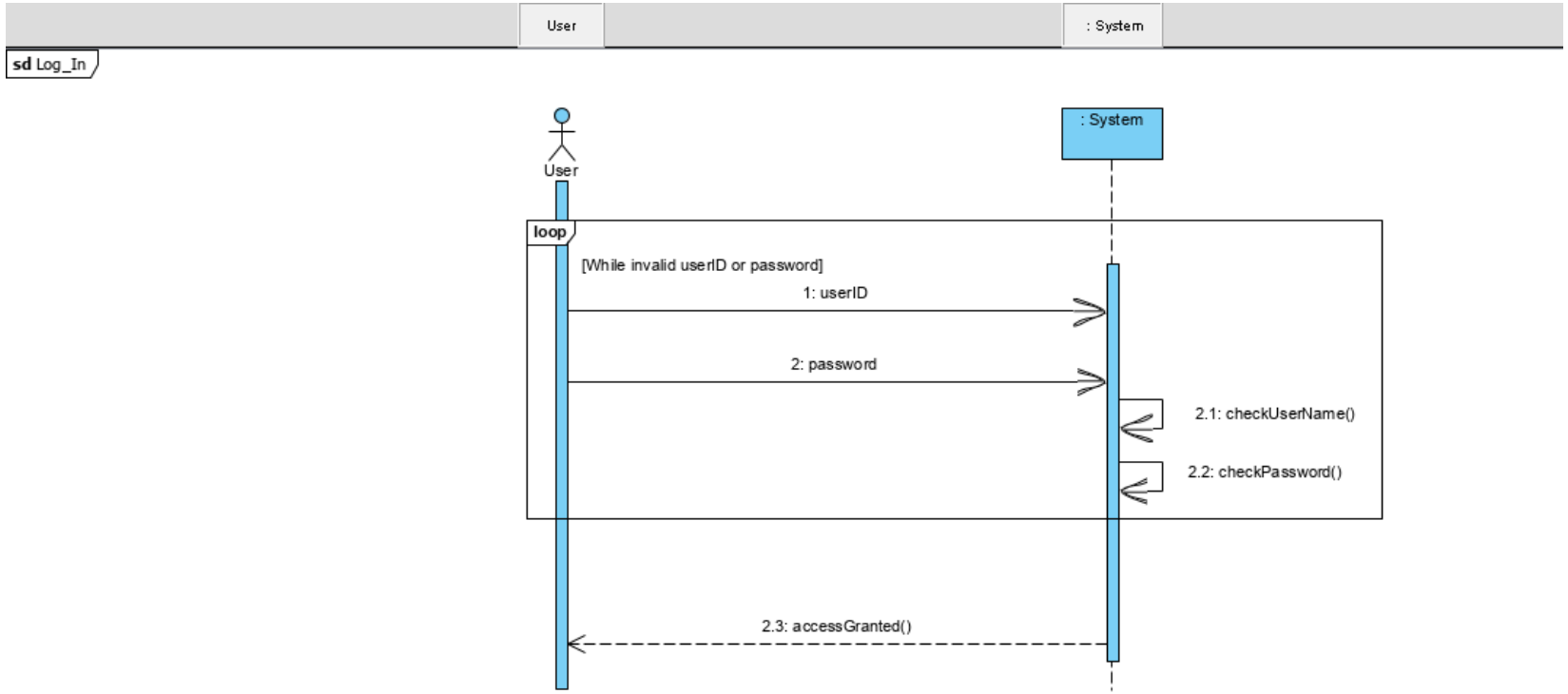


Traceability Matrix

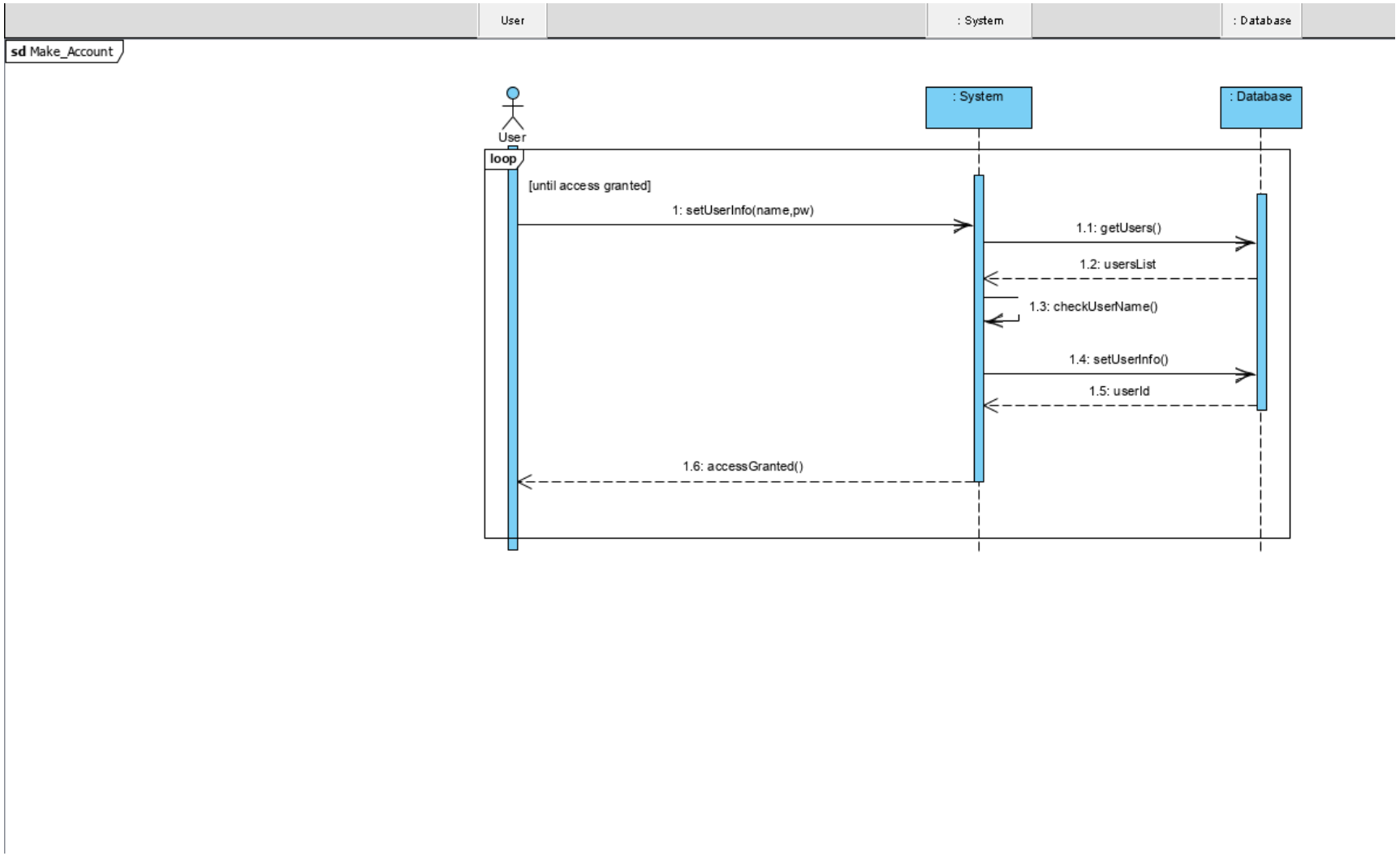


(13) Requirement / By: Reference / (9) Use Case	Sanitize user input.	Respond to a messag...	Able to log out.	Navigate bidirectionally.	Search based on class.	Search based on prof...	See professor options ...	Message the author of...	Upvote or downvote a ...	View all reviews	Write a review on a pr...	Delete an account.	Register an account.
Contact Student								✓					
Delete account												✓	
Flag Comment	✓												
Leave Review				✓	✓	✓	✓				✓		
Log-out			✓										
Make Account													✓
Rate Review				✓	✓	✓	✓		✓				
Read Review				✓	✓	✓	✓			✓			
Respond to message		✓											

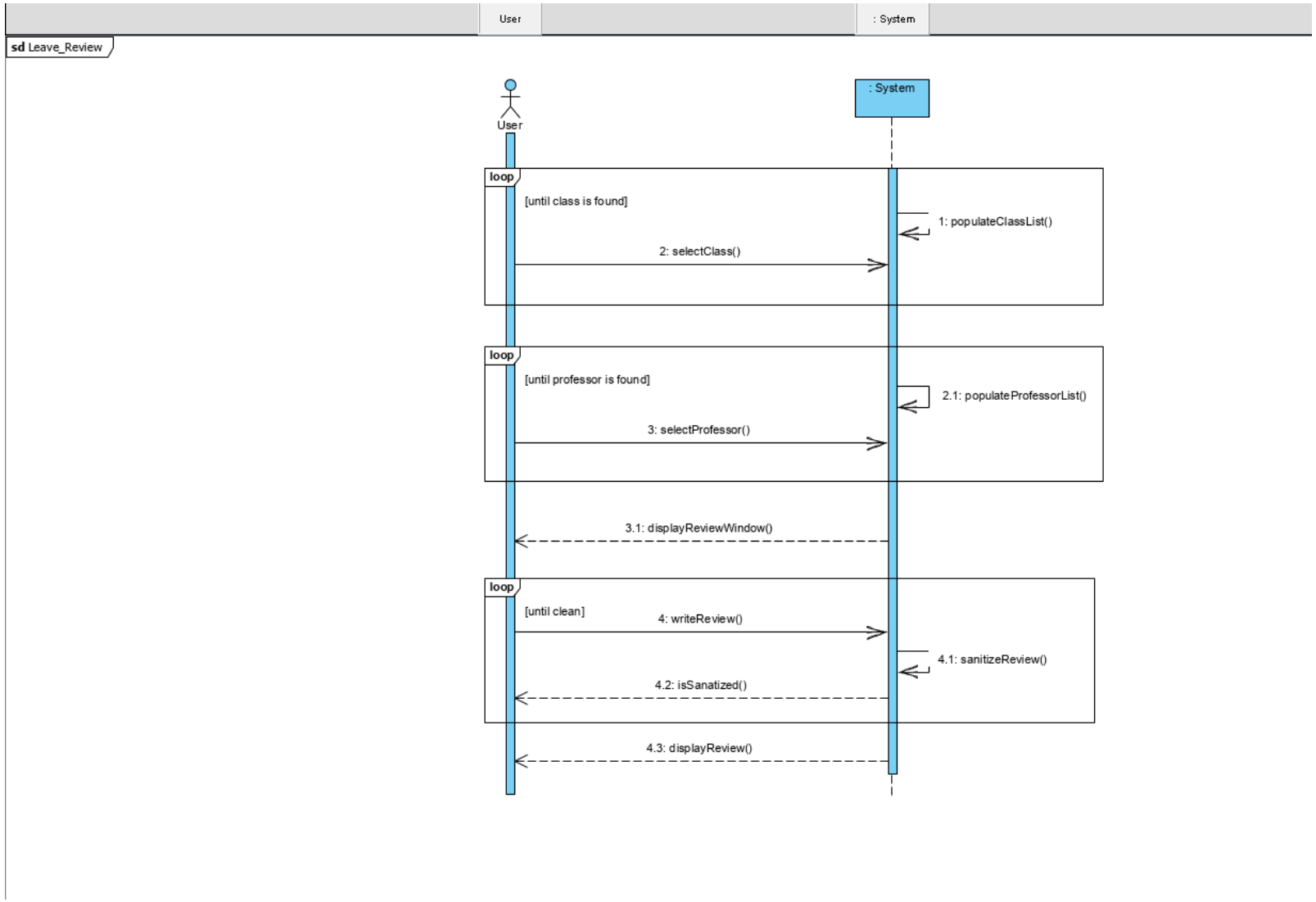
SSD


User
: System
sd Log_In
User
: System
loop
[While invalid userID or password]
1: userID
2: password
2.1: checkUserName()
2.2: checkPassword()
2.3: accessGranted()

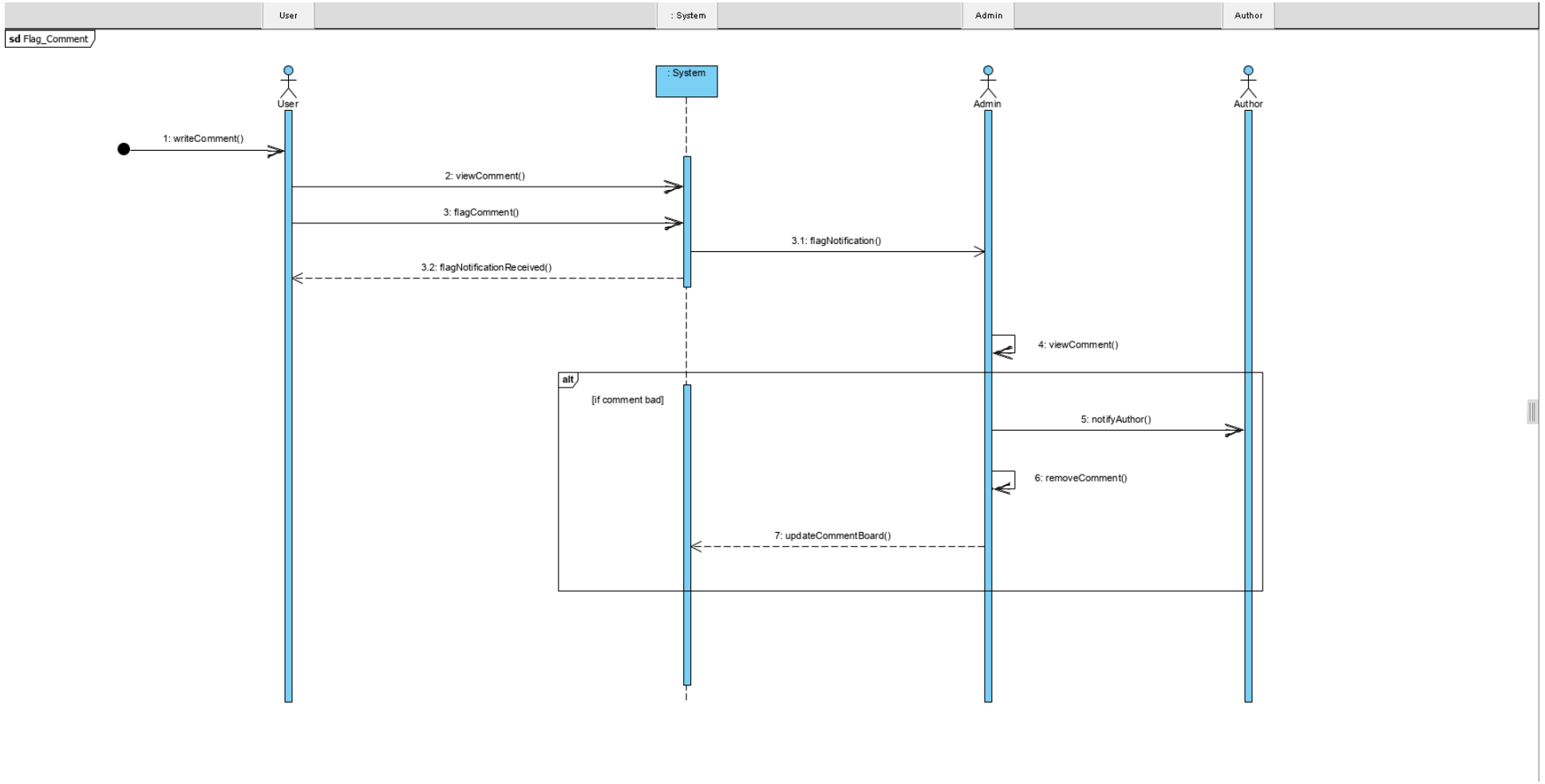
SSD


User
: System
: Database
sd Make_Account
User
: System
: Database
loop
[until access granted]
1: setUserInfo(name,pw)
1.1: getUsers()
1.2: usersList
1.3: checkUserName()
1.4: setUserInfo()
1.5: userId
1.6: accessGranted()

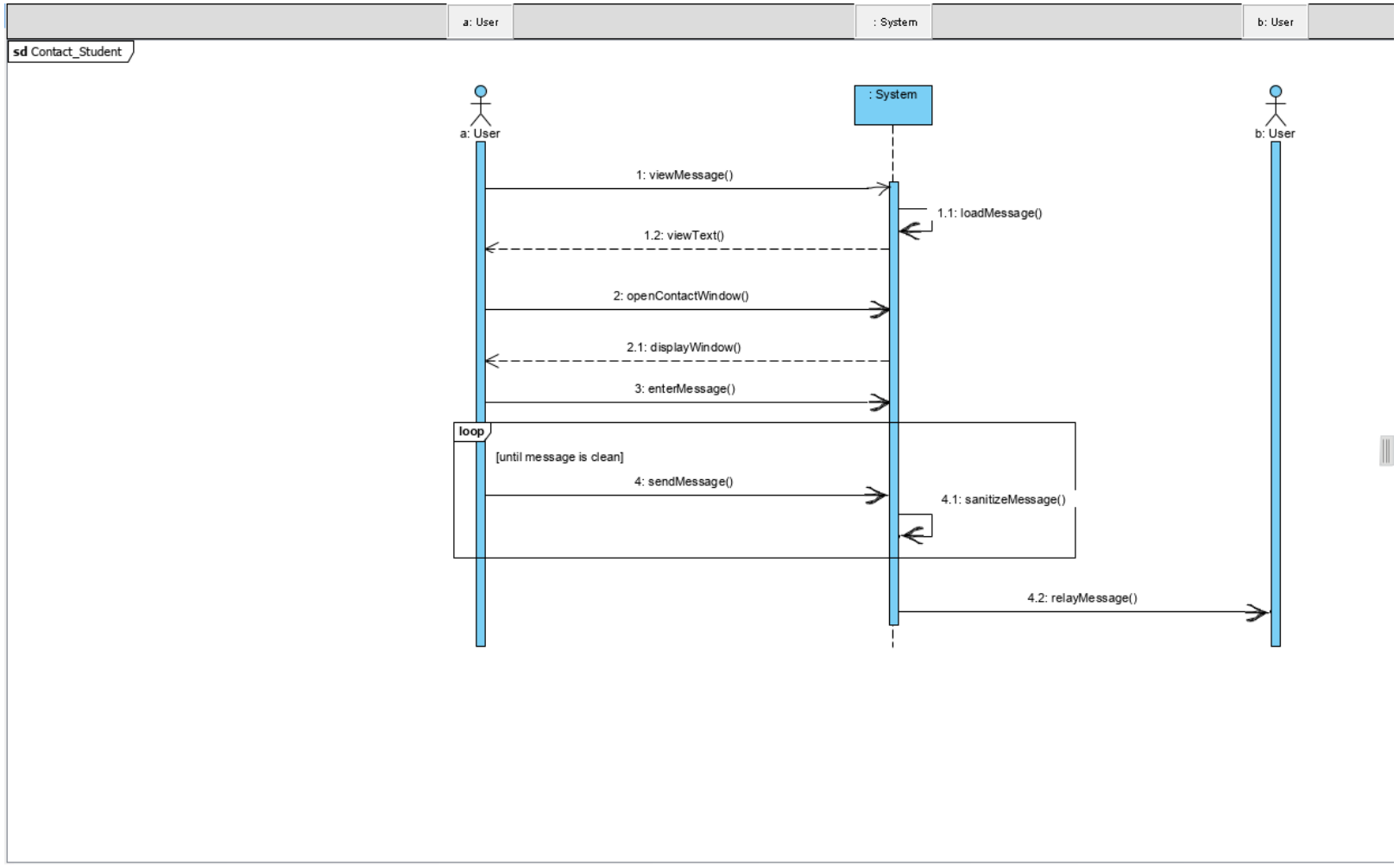
SSD


User
: System
sd Leave_Review
User
: System
loop
[until class is found]
1: populateClassList()
2: selectClass()
loop
[until professor is found]
2.1: populateProfessorList()
3: selectProfessor()
3.1: displayReviewWindow()
loop
[until clean]
4: writeReview()
4.1: sanitizeReview()
4.2: isSanatized()
4.3: displayReview()

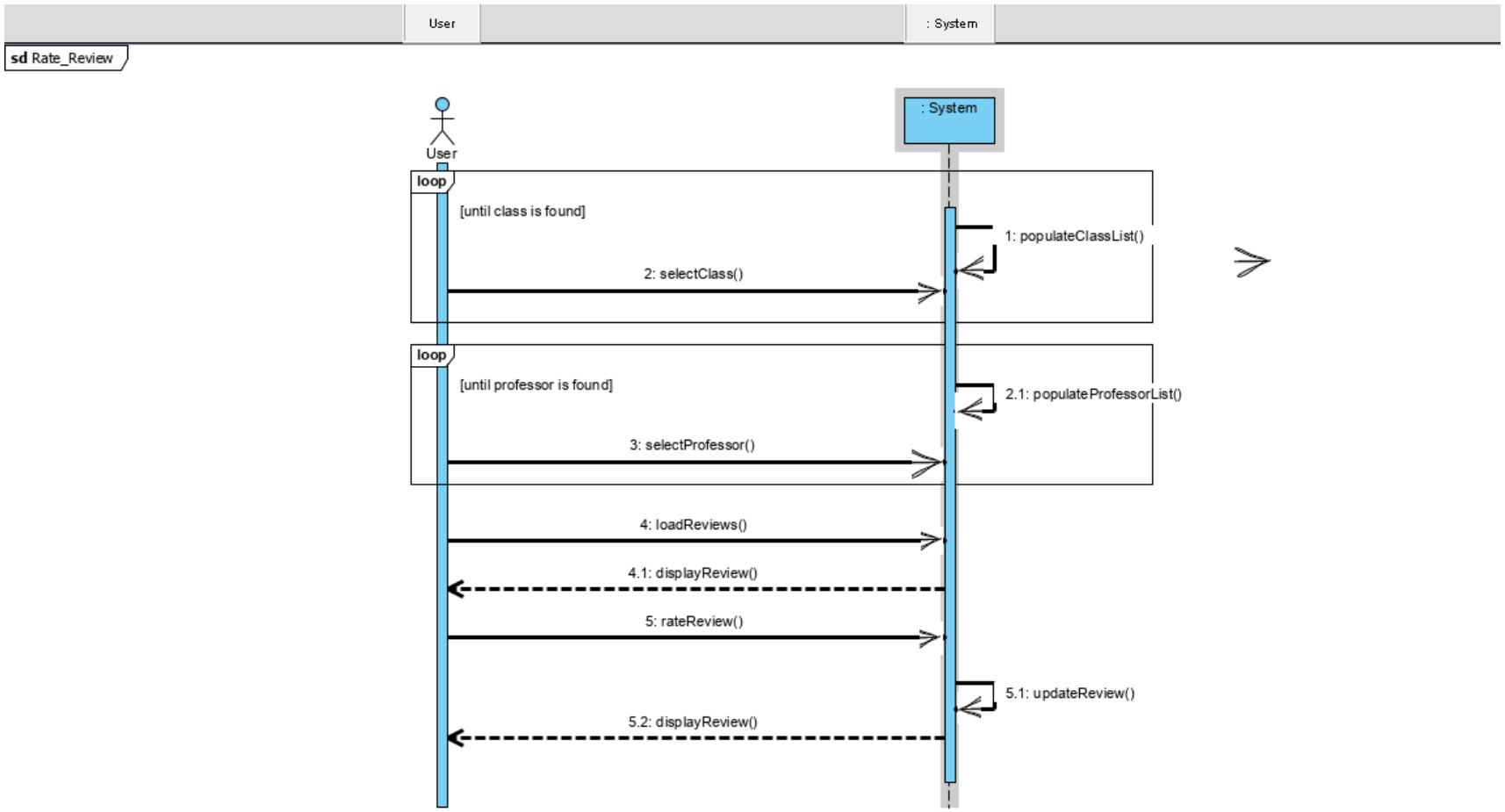
SSD


User
: System
Admin
Author
sd Flag_Comment
1: writeComment()
2: viewComment()
3: flagComment()
3.1: flagNotification()
3.2: flagNotificationReceived()
4: viewComment()
alt
[if comment bad]
5: notifyAuthor()
6: removeComment()
7: updateCommentBoard()

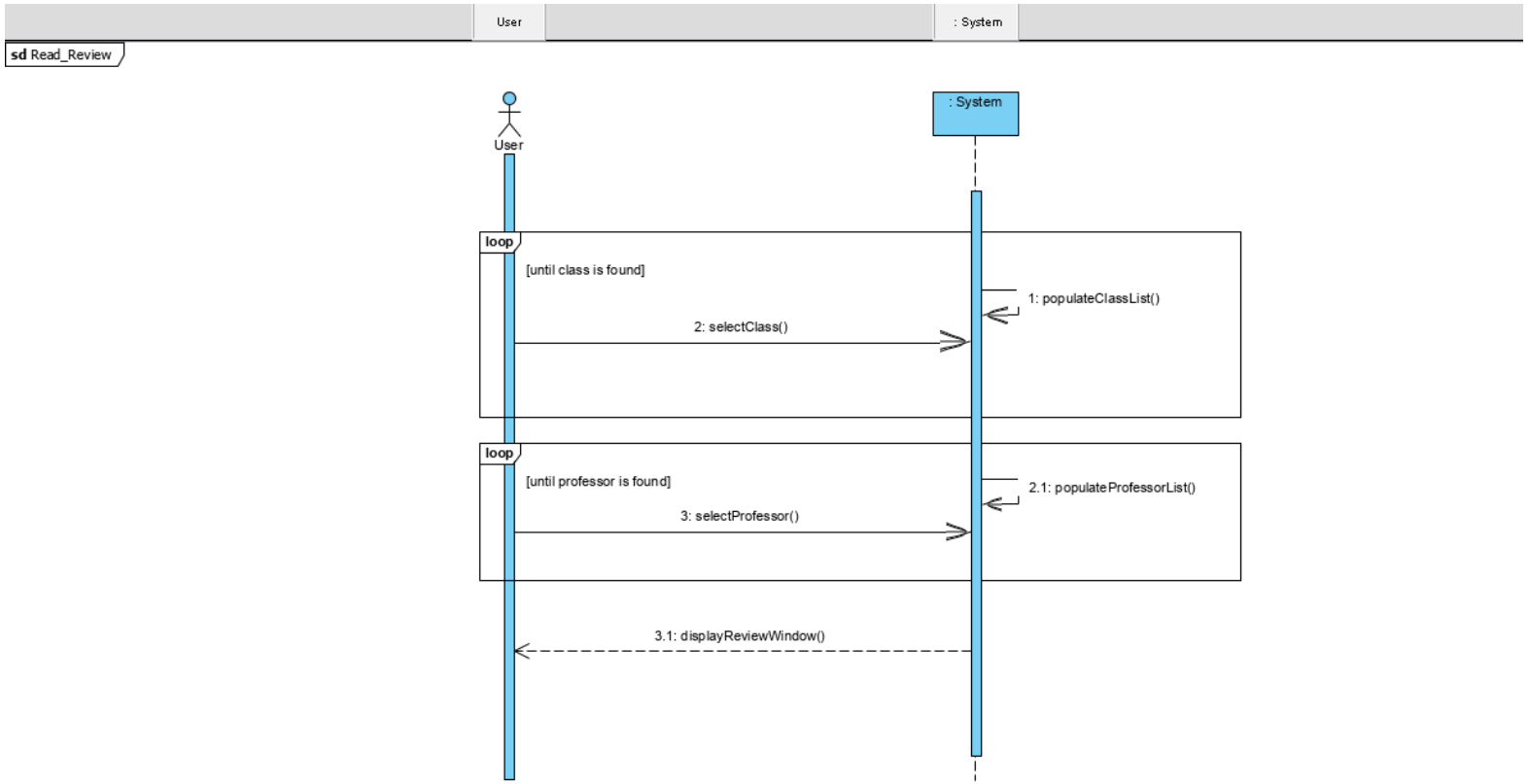
SSD


a: User
: System
b: User
sd Contact_Student
a: User
: System
b: User
1: viewMessage()
1.1: loadMessage()
1.2: viewText()
2: openContactWindow()
2.1: displayWindow()
3: enterMessage()
loop
[until message is clean]
4: sendMessage()
4.1: sanitizeMessage()
4.2: relayMessage()

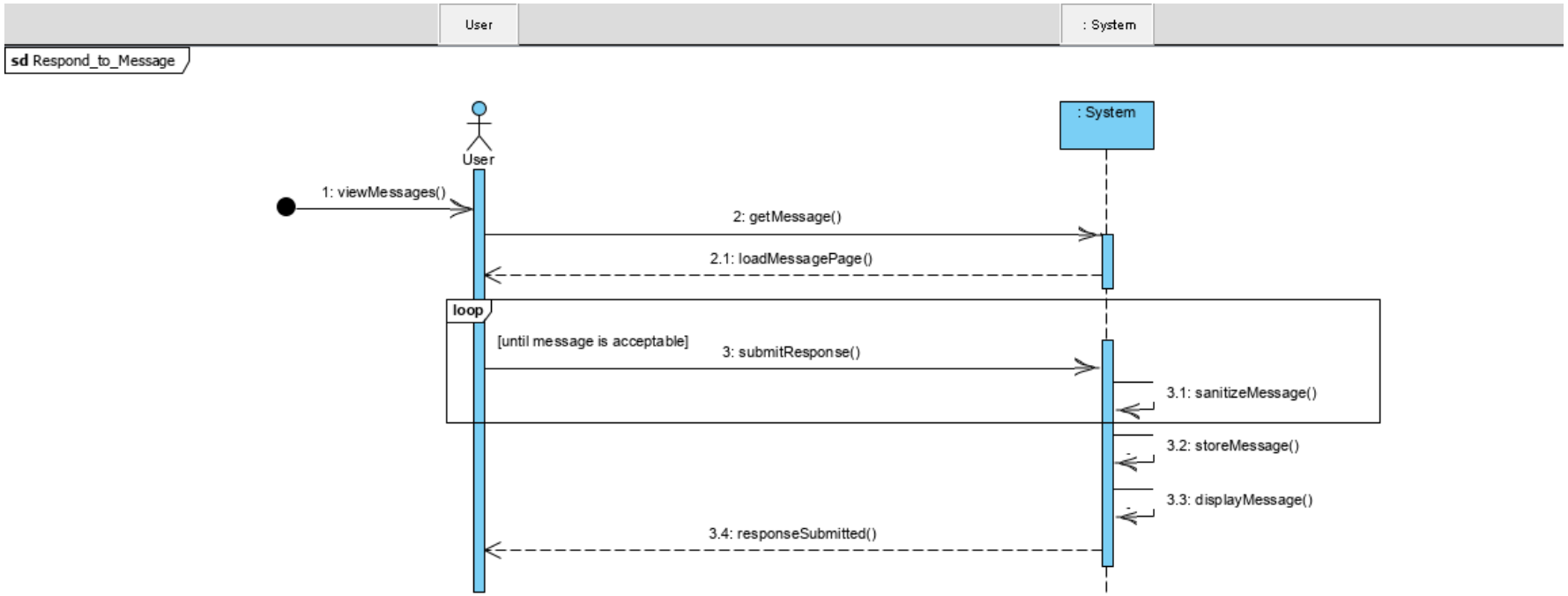
SSD


User
: System
sd Rate_Review
User
: System
loop
[until class is found]
1: populateClassList()
2: selectClass()
loop
[until professor is found]
2.1: populateProfessorList()
3: selectProfessor()
4: loadReviews()
4.1: displayReview()
5: rateReview()
5.1: updateReview()
5.2: displayReview()

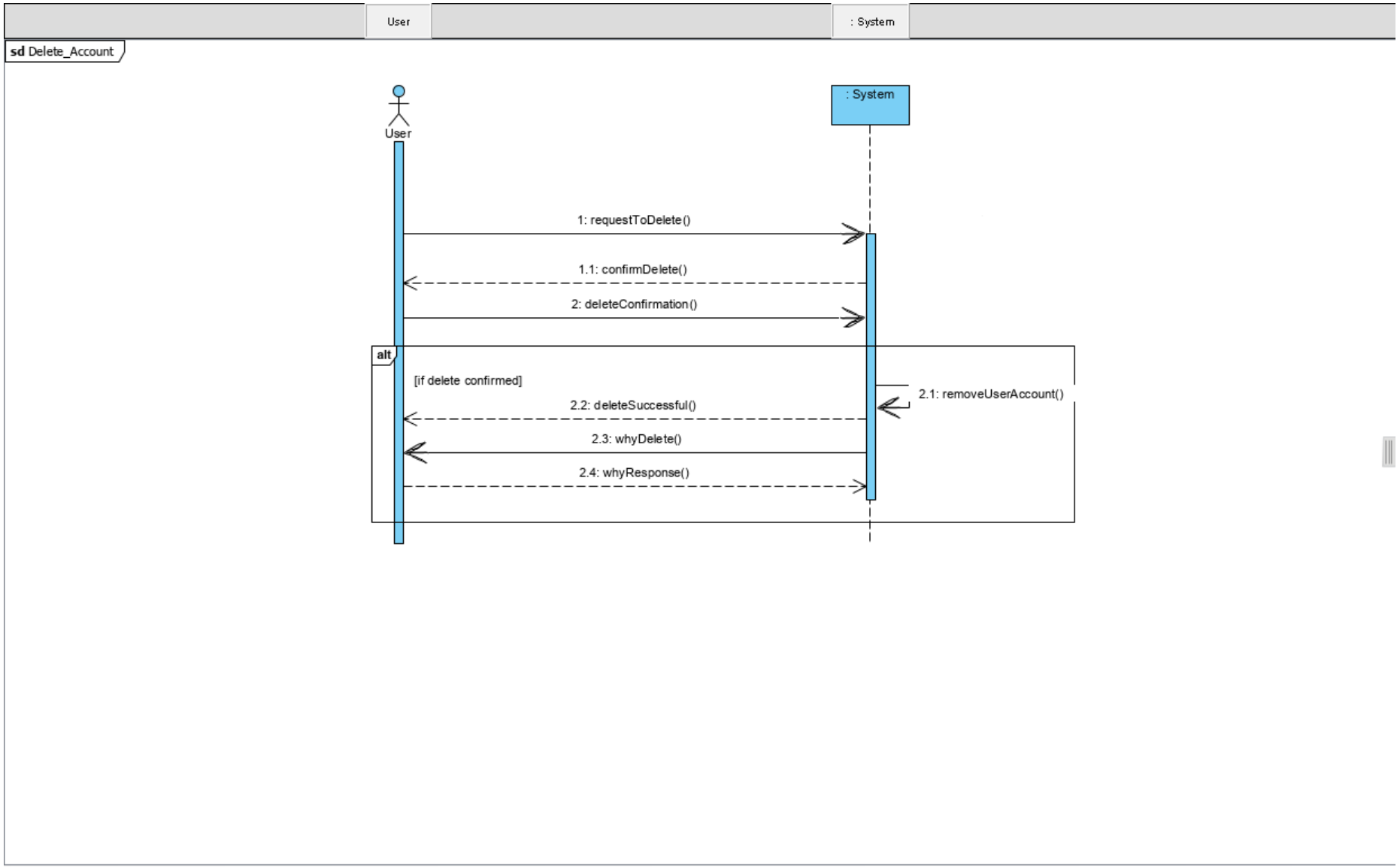
SSD


User
: System
sd Read_Review
User
: System
loop
[until class is found]
1: populateClassList()
2: selectClass()
loop
[until professor is found]
2.1: populateProfessorList()
3: selectProfessor()
3.1: displayReviewWindow()

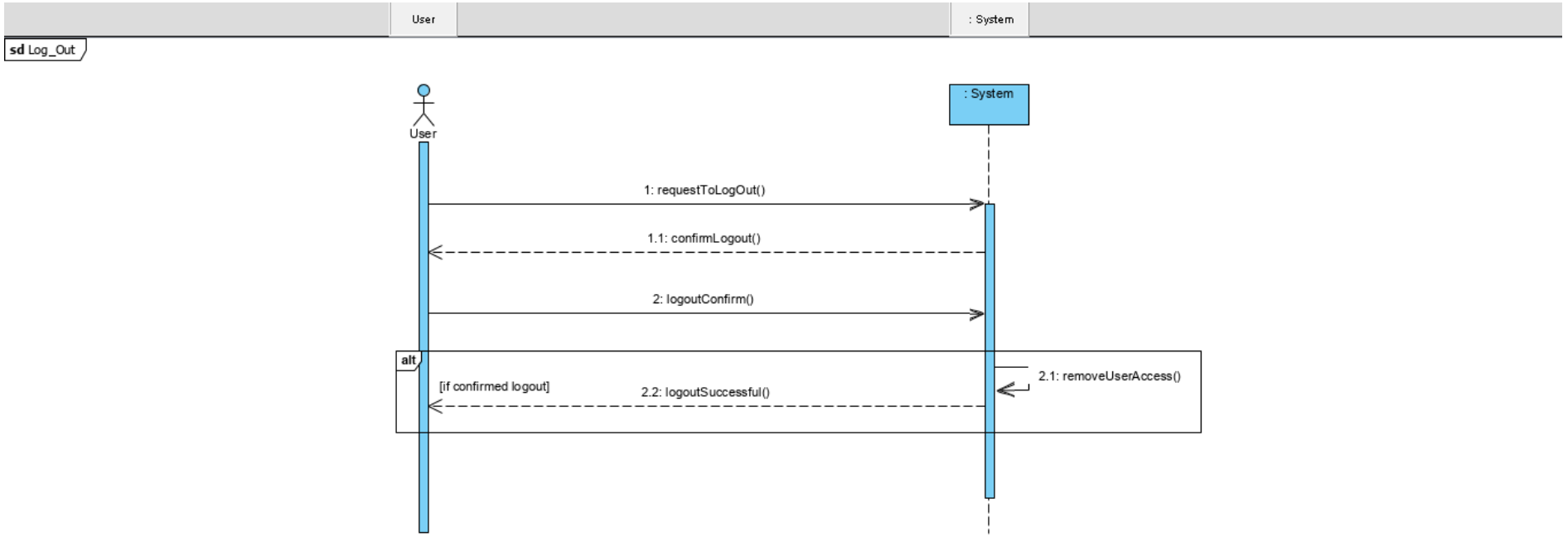
SSD


User
: System
sd Respond_to_Message
User
: System
1: viewMessages()
2: getMessage()
2.1: loadMessagePage()
loop
[until message is acceptable]
3: submitResponse()
3.1: sanitizeMessage()
3.2: storeMessage()
3.3: displayMessage()
3.4: responseSubmitted()

SSD


User
: System
sd Delete_Account
User
: System
1: requestToDelete()
1.1: confirmDelete()
2: deleteConfirmation()
alt
[if delete confirmed]
2.1: removeUserAccount()
2.2: deleteSuccessful()
2.3: whyDelete()
2.4: whyResponse()

SSD


User
: System
sd Log_Out
User
: System
1: requestToLogOut()
1.1: confirmLogout()
2: logoutConfirm()
alt
[if confirmed logout]
2.2: logoutSuccessful()
2.1: removeUserAccess()

OPERATIONS

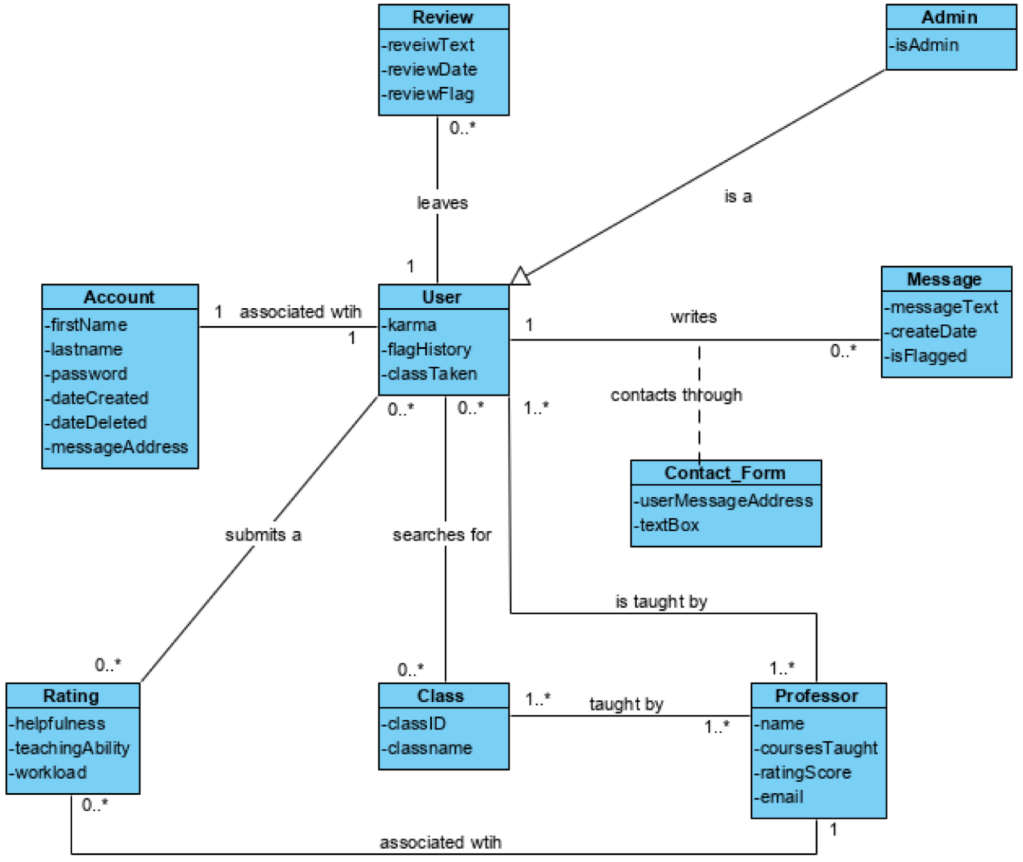
Defining the system operations helps us plan how we will implement the BCC application.

Examples

- checkUsername()
- checkPassword()
- populateClassList()
- populateProfessorList()
- sendMessage()
- viewMessage()

There are ~70 more ☺

Domain Model



BCC Iteration 2

Mark Fuller

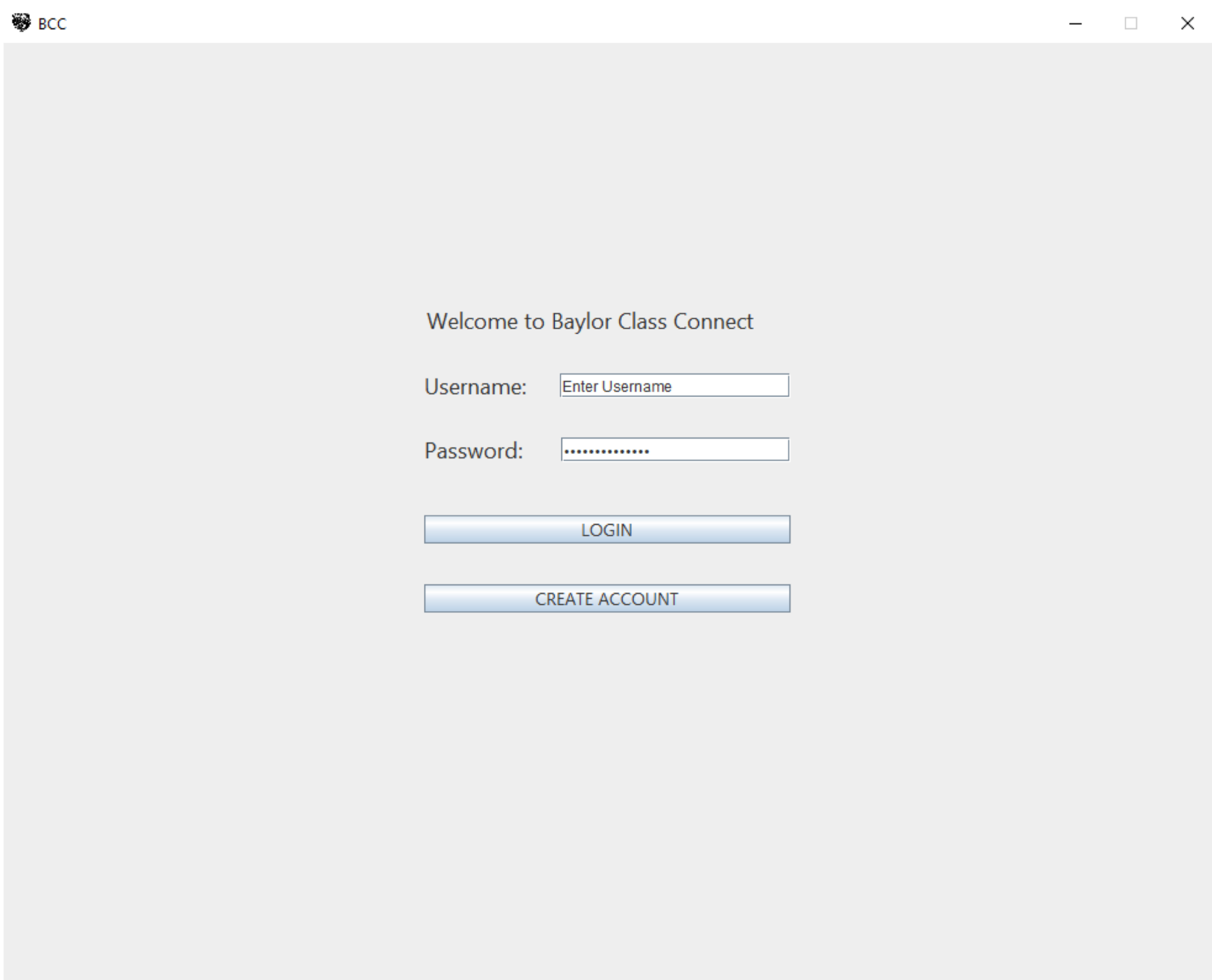
Kevin Kulda

Connor Woodahl

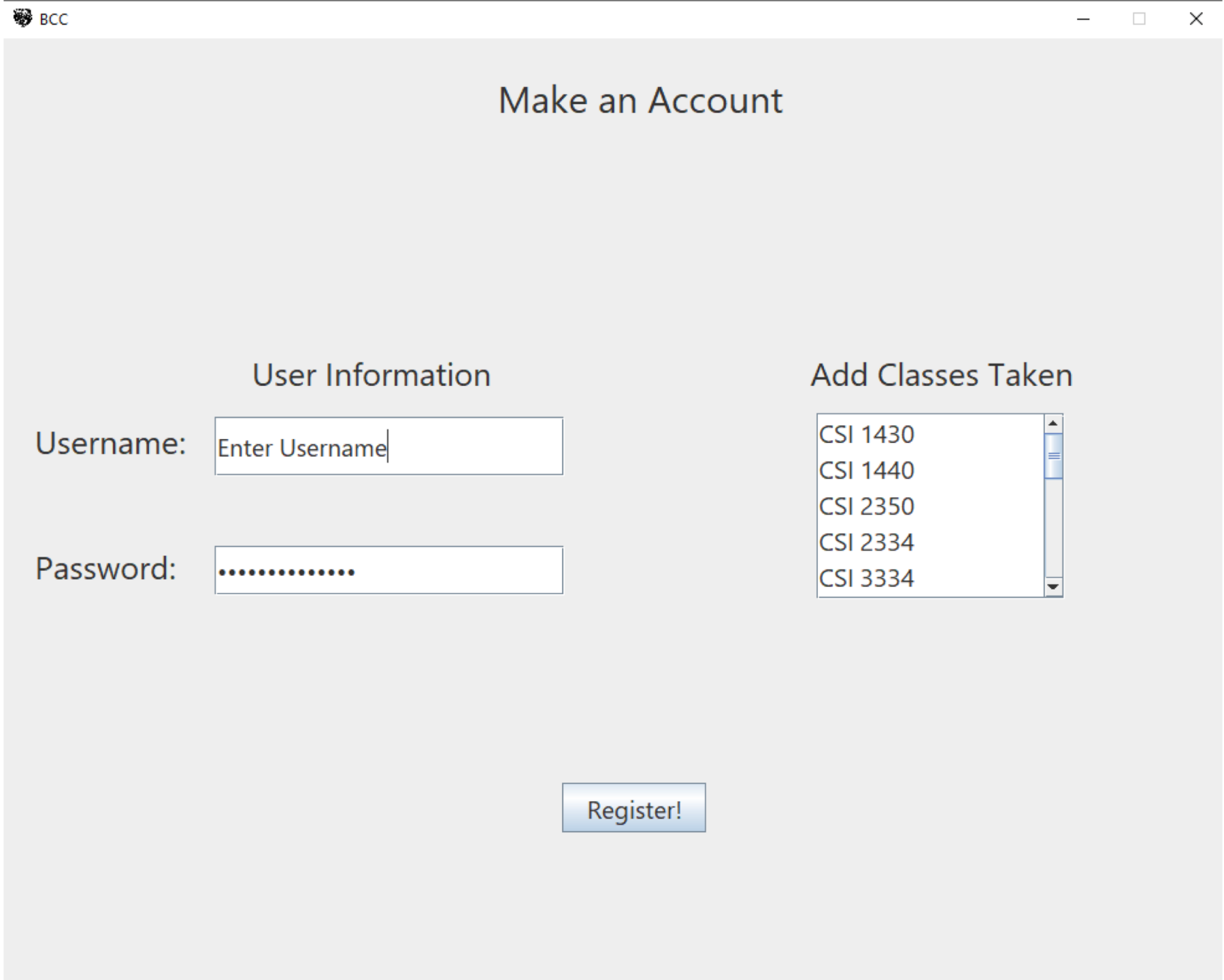
Project Repository: https://github.com/MarkFuller1/BNU

Application Interface

Login


BCC
Welcome to Baylor Class Connect
Username:
Enter Username
Password:
LOGIN
CREATE ACCOUNT

Register


BCC
Make an Account
User Information
Add Classes Taken
Username:
Enter Username
Password:
CSI 1430
CSI 1440
CSI 2350
CSI 2334
CSI 3334
Register!

Main Page

BCC

Log Out

Profile

Baylor Class Connect

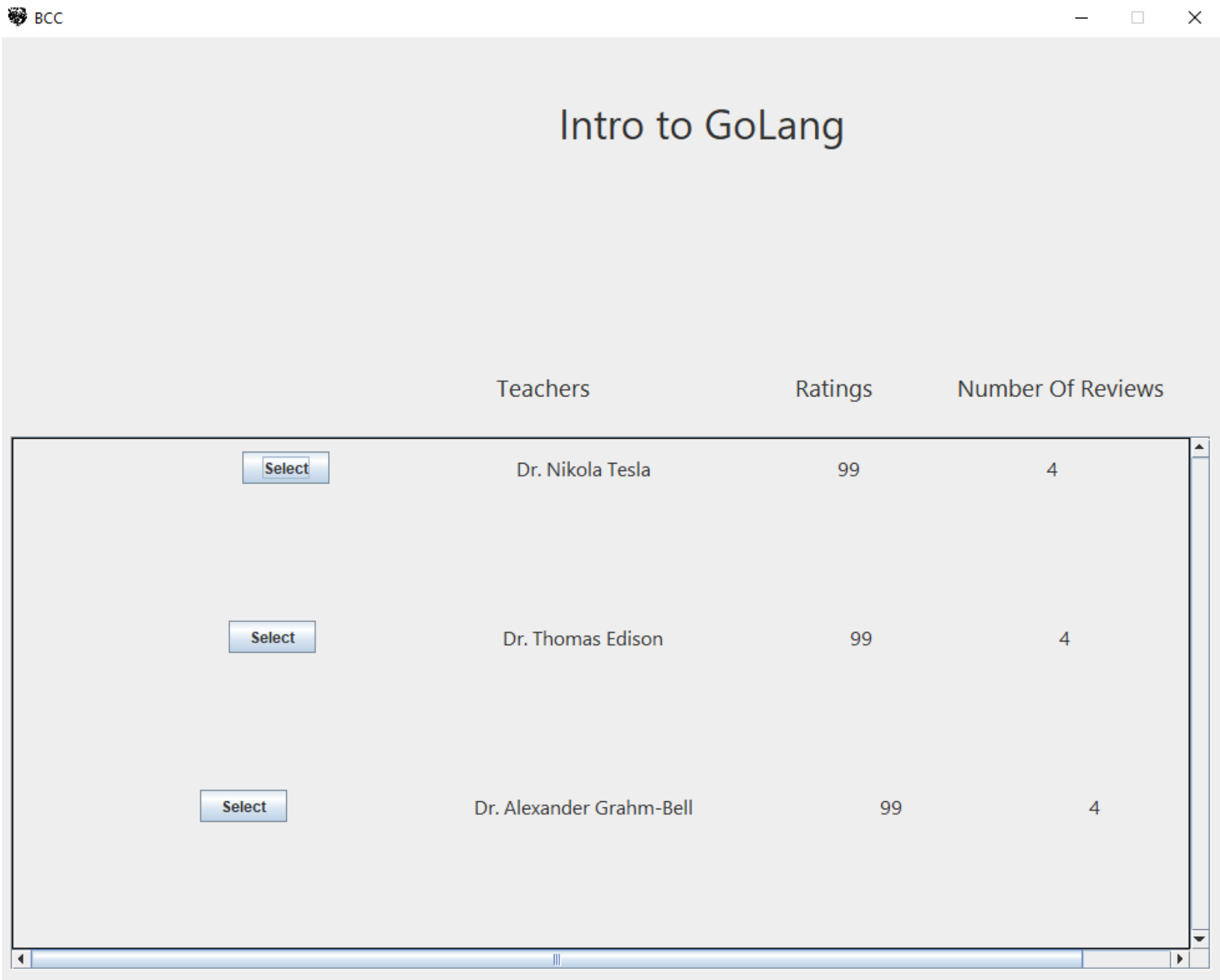
Search By Class

Adventures in Multi-Threading

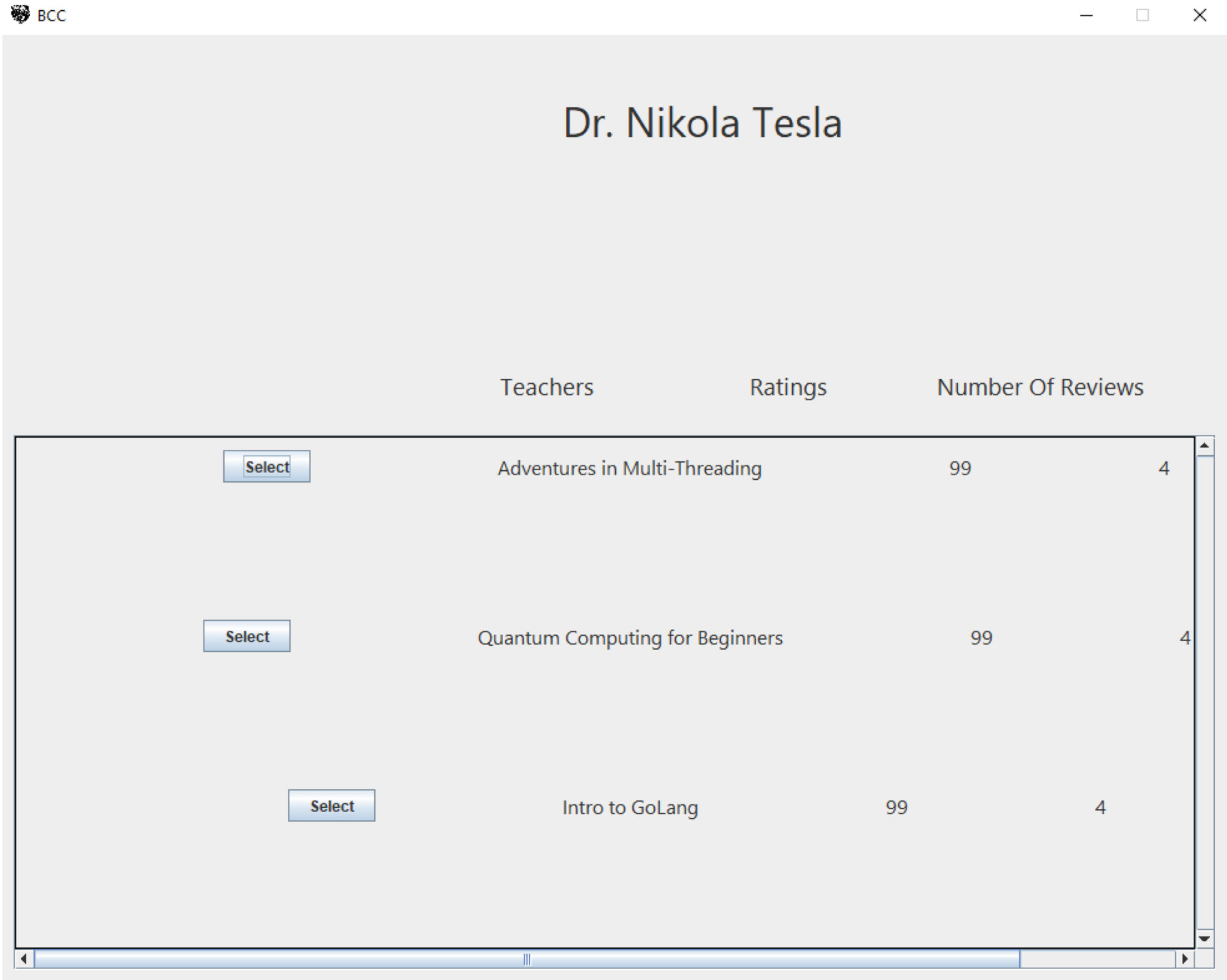
Search By Professor

Dr. Nikola Tesla

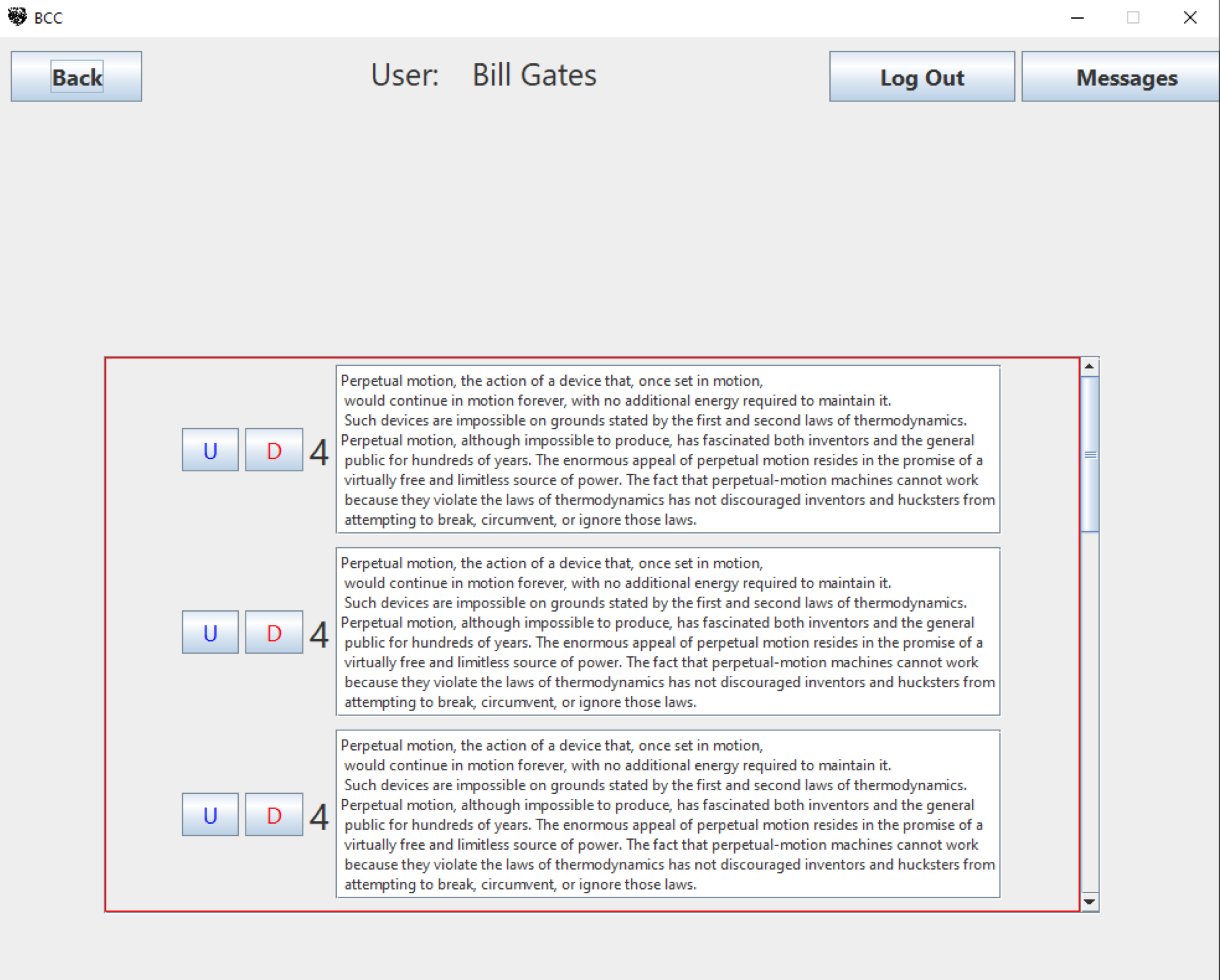
Search by Class



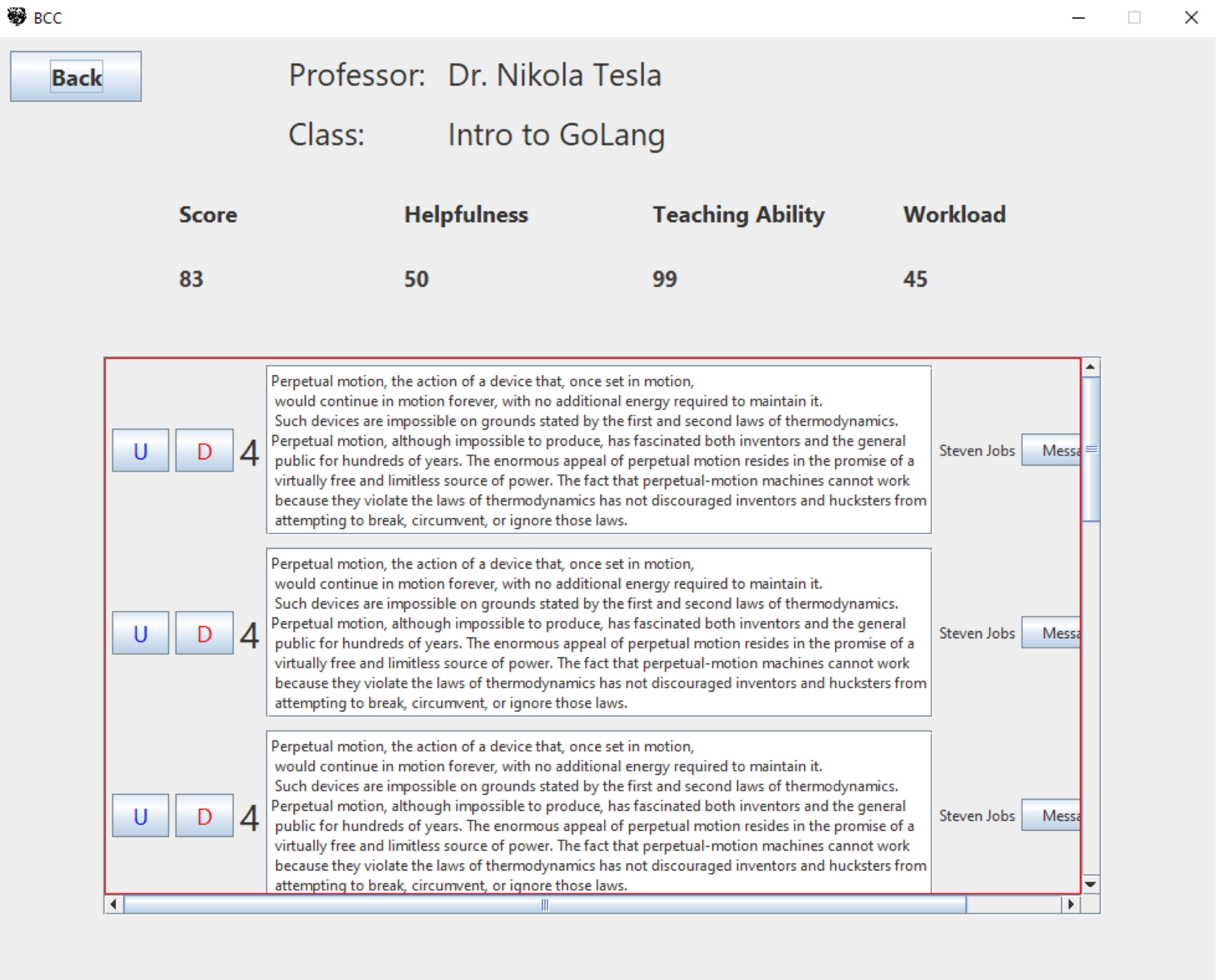
Search by Professor



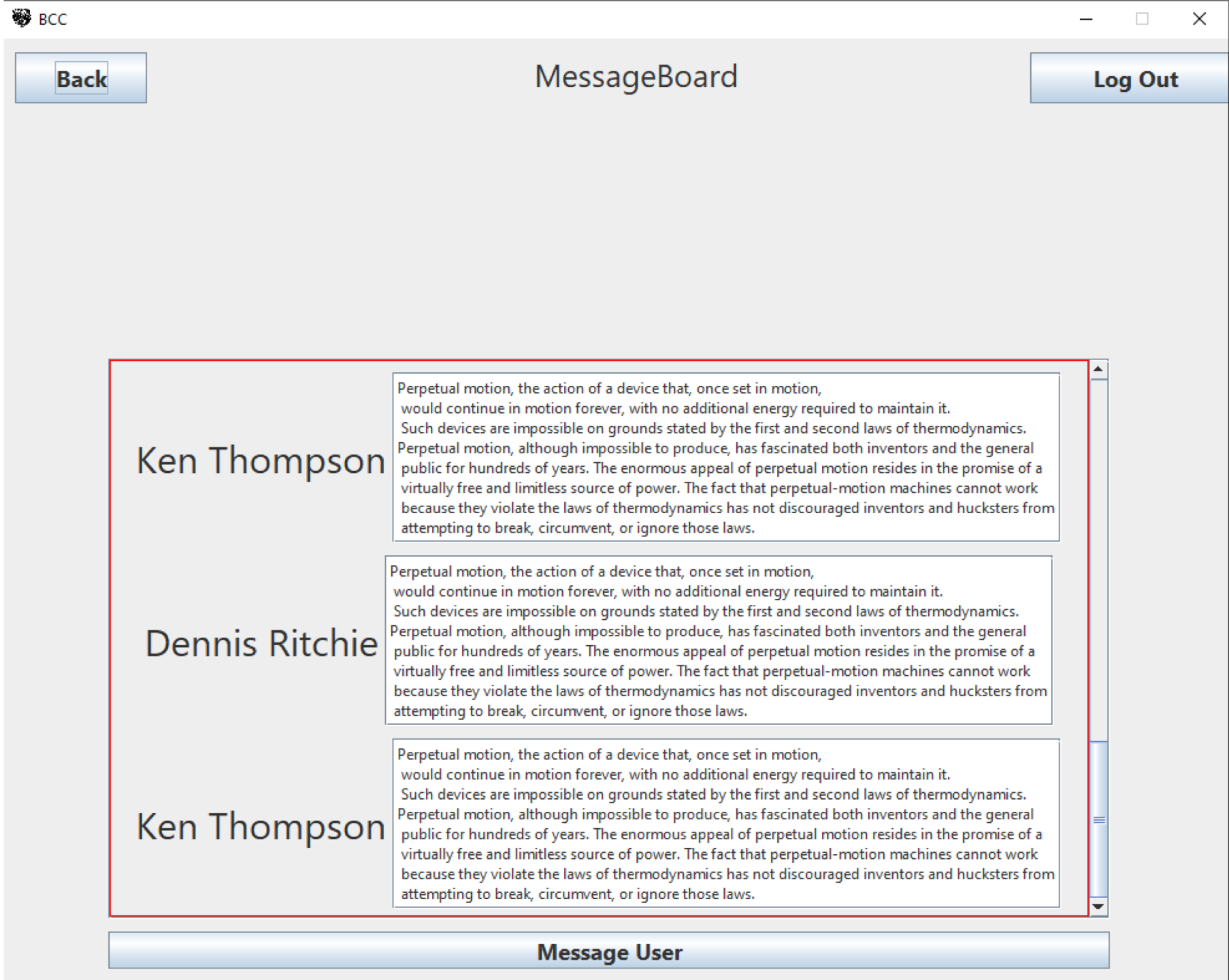
User Reviews



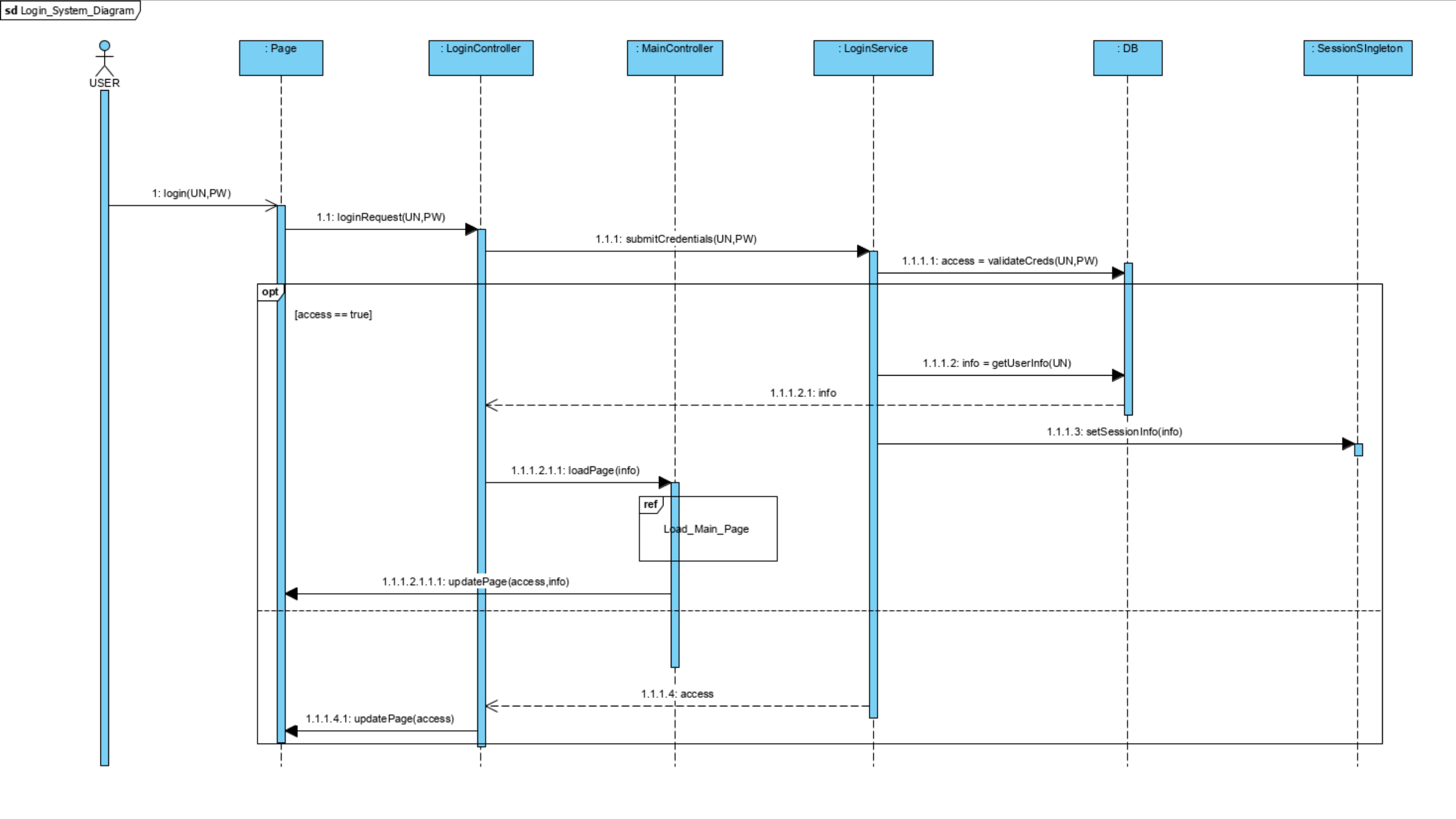
Class Reviews

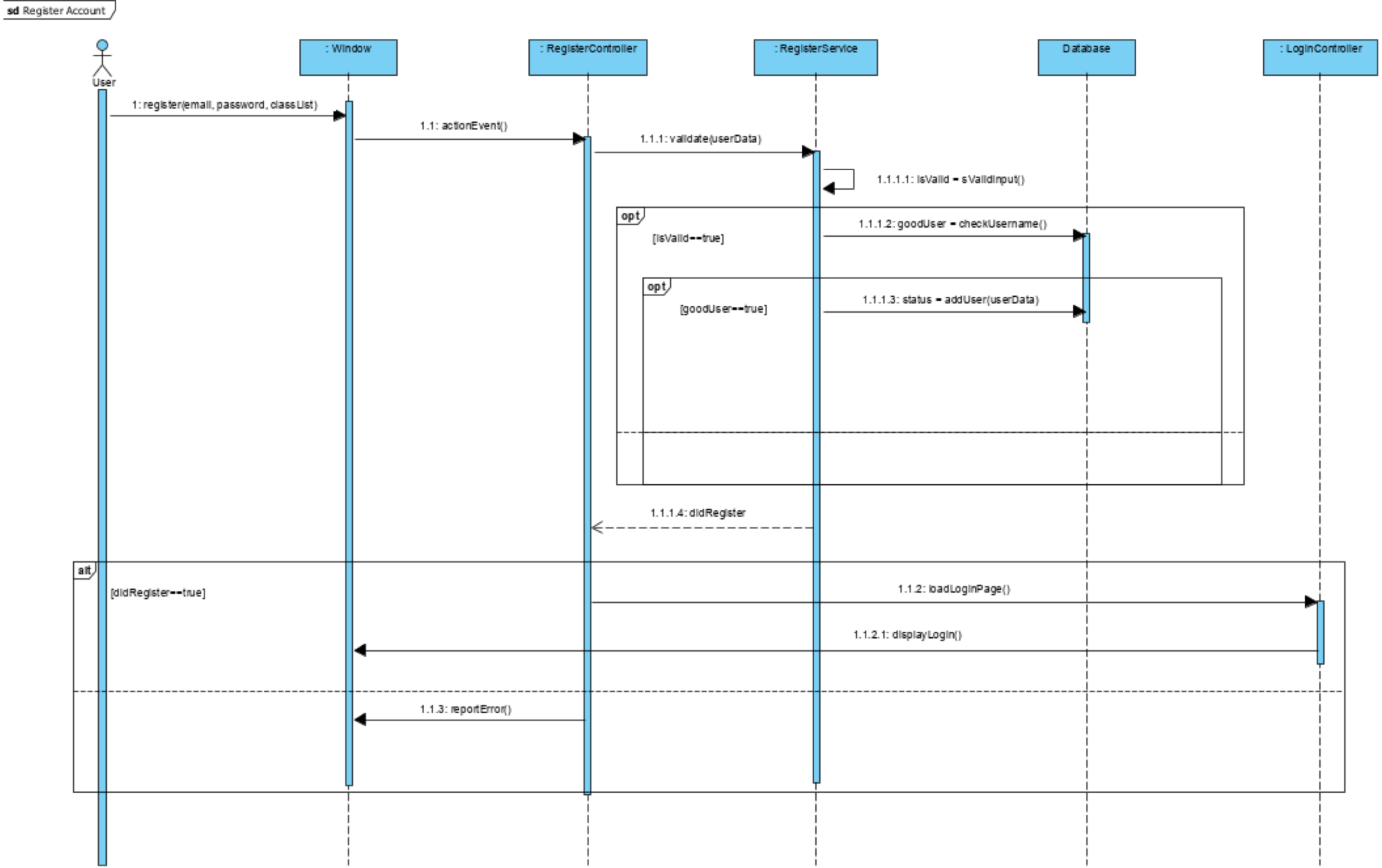


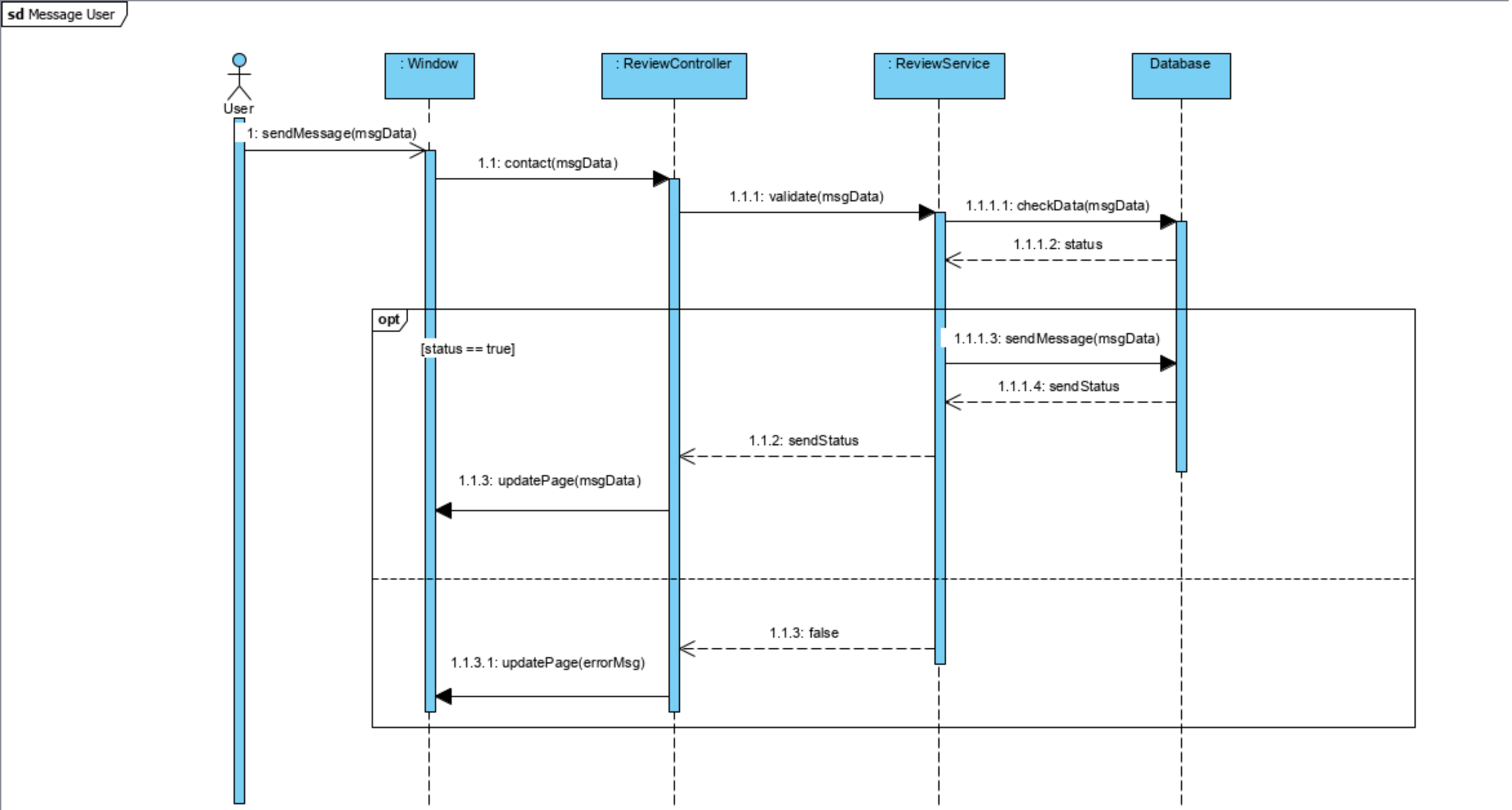
Message Board

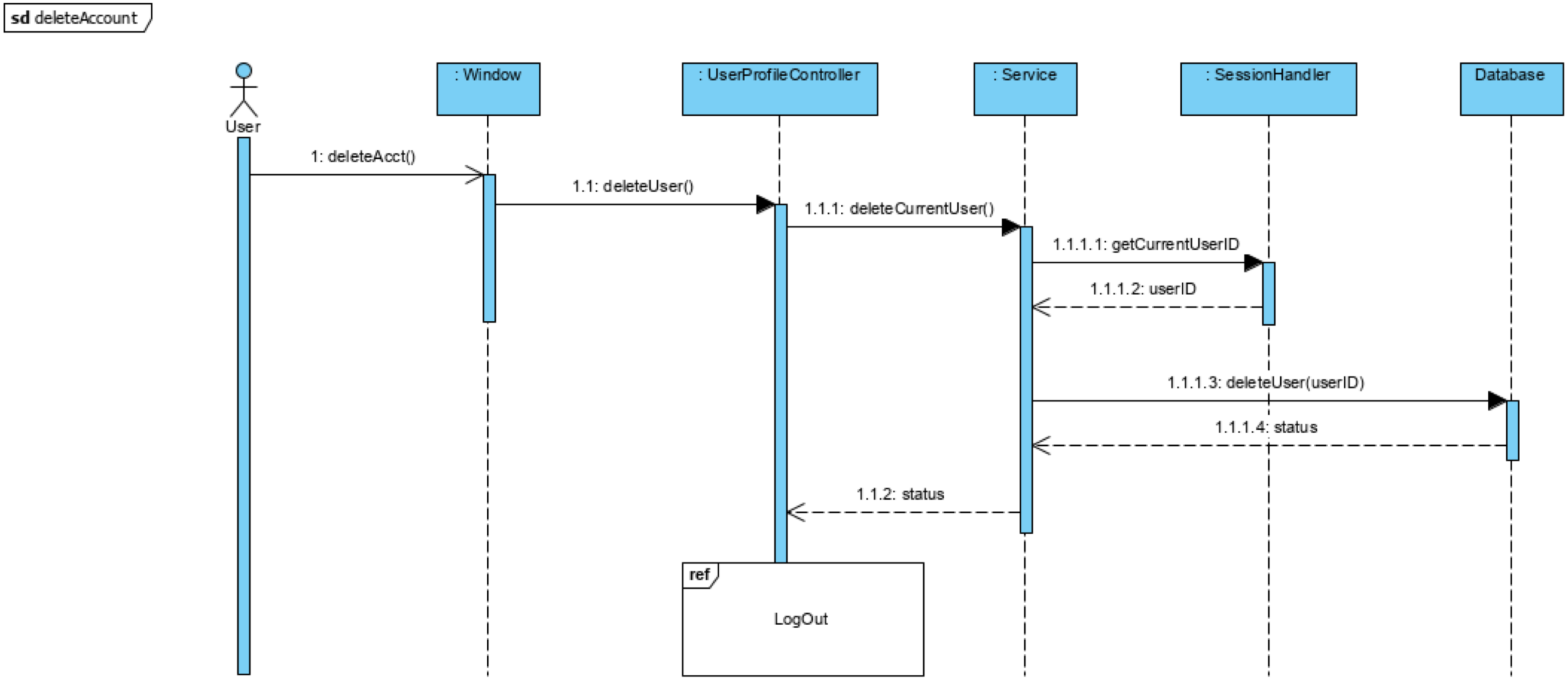


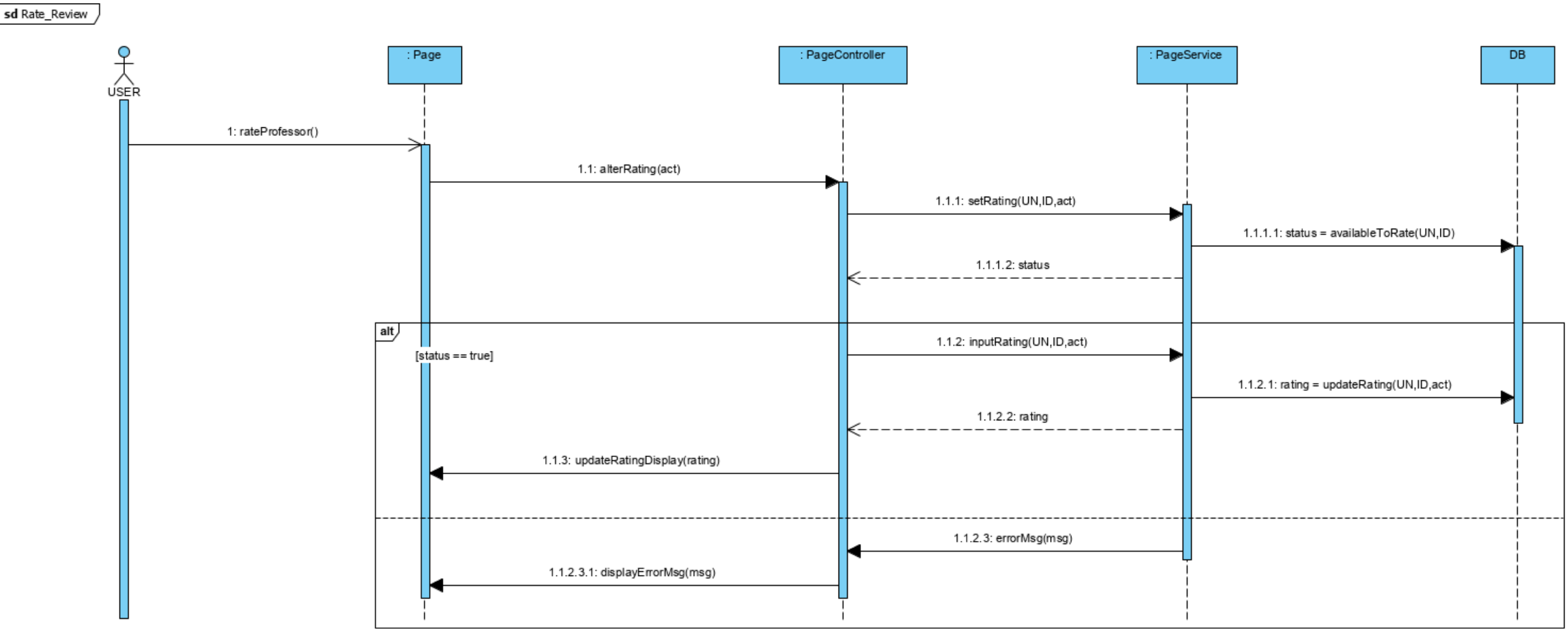
Sequence Diagrams

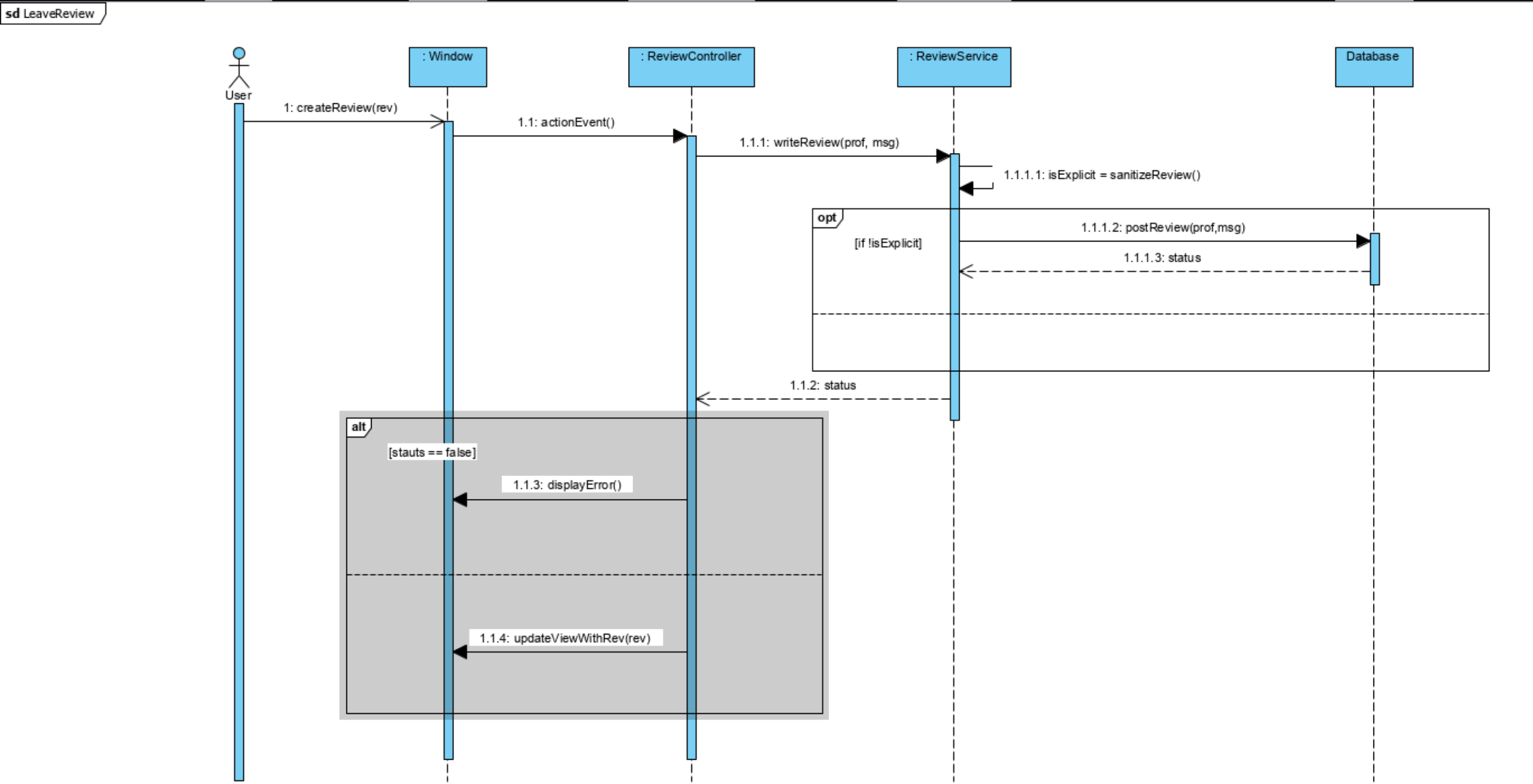

sd Login_System_Diagram
USER
: Page
: LoginController
: MainController
: LoginService
: DB
: SessionSIngleton
1: login(UN,PW)
1.1: loginRequest(UN,PW)
1.1.1: submitCredentials(UN,PW)
1.1.1.1: access = validateCreds(UN,PW)
opt
[access == true]
1.1.1.2: info = getUserInfo(UN)
1.1.1.2.1: info
1.1.1.3: setSessionInfo(info)
1.1.1.2.1.1: loadPage(info)
ref
Load_Main_Page
1.1.1.2.1.1.1: updatePage(access,info)
1.1.1.4: access
1.1.1.4.1: updatePage(access)

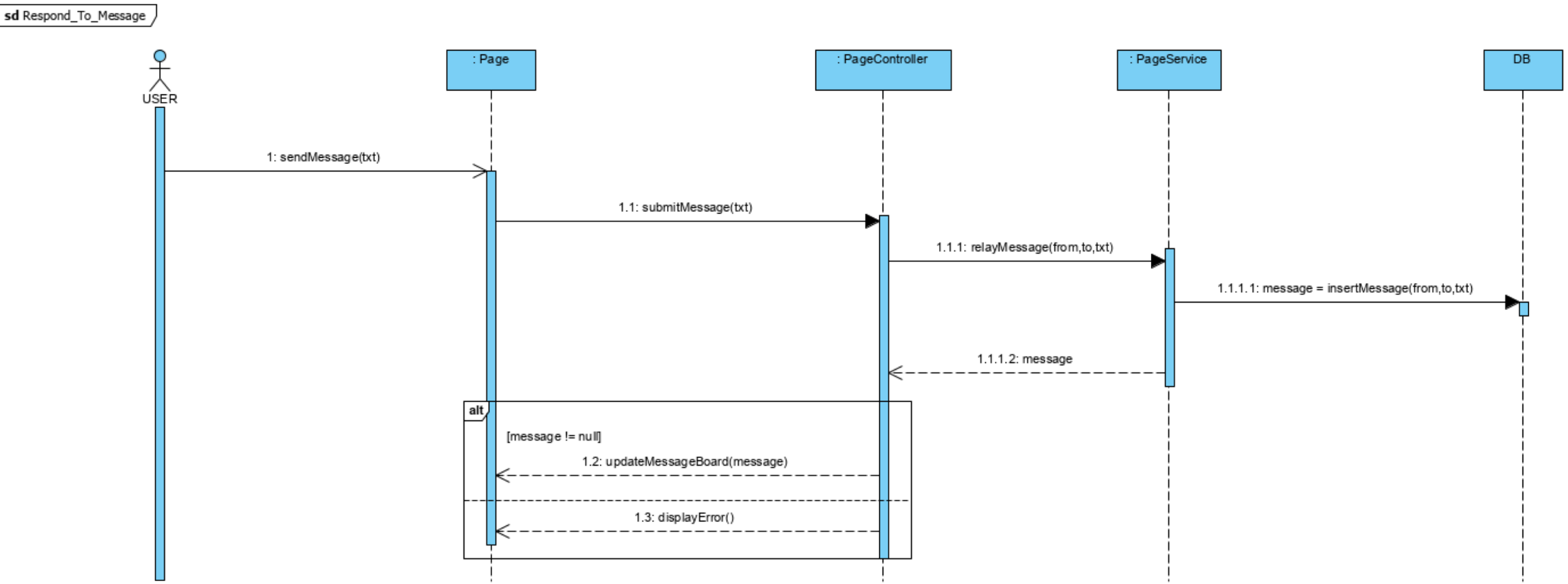

sd Register Account
User
: Window
: RegisterController
: RegisterService
Database
: LoginController
1: register(email, password, classList)
1.1: actionEvent()
1.1.1: validate(userData)
1.1.1.1: isValid = isValidInput()
opt
[isValid==true]
1.1.1.2: goodUser = checkUsername()
opt
[goodUser==true]
1.1.1.3: status = addUser(userData)
1.1.1.4: didRegister
alt
[didRegister==true]
1.1.2: loadLoginPage()
1.1.2.1: displayLogin()
1.1.3: reportError()

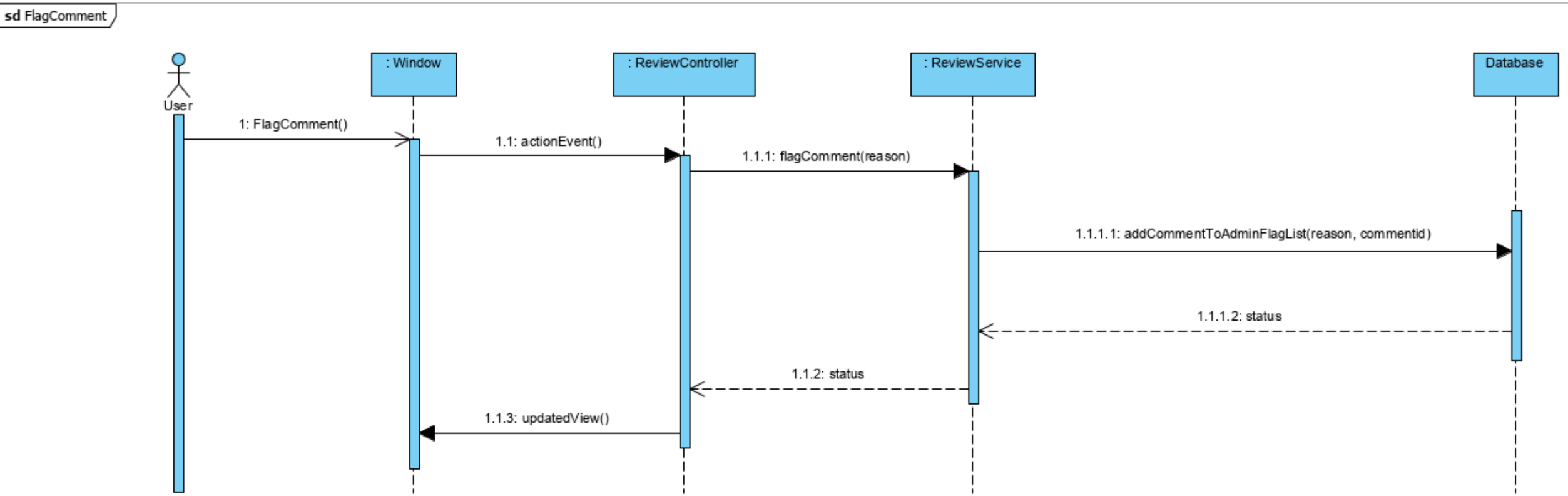

sd Message User
User
: Window
: ReviewController
: ReviewService
Database
1: sendMessage(msgData)
1.1: contact(msgData)
1.1.1: validate(msgData)
1.1.1.1: checkData(msgData)
1.1.1.2: status
opt
[status == true]
1.1.1.3: sendMessage(msgData)
1.1.1.4: sendStatus
1.1.2: sendStatus
1.1.3: updatePage(msgData)
1.1.3: false
1.1.3.1: updatePage(errorMsg)

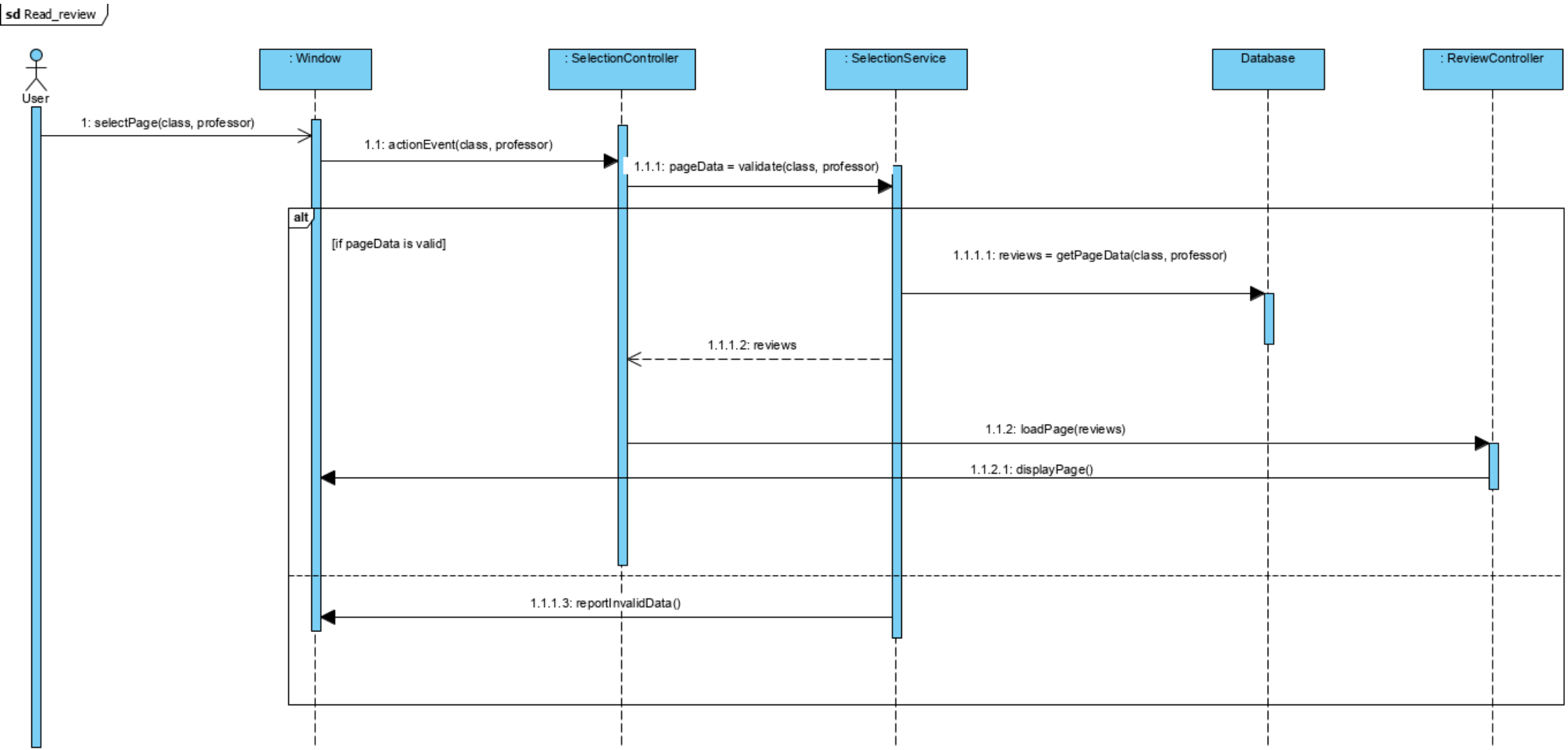

sd deleteAccount
User
: Window
: UserProfileController
: Service
: SessionHandler
Database
1: deleteAcct()
1.1: deleteUser()
1.1.1: deleteCurrentUser()
1.1.1.1: getCurrentUserID
1.1.1.2: userID
1.1.1.3: deleteUser(userID)
1.1.1.4: status
1.1.2: status
ref
LogOut

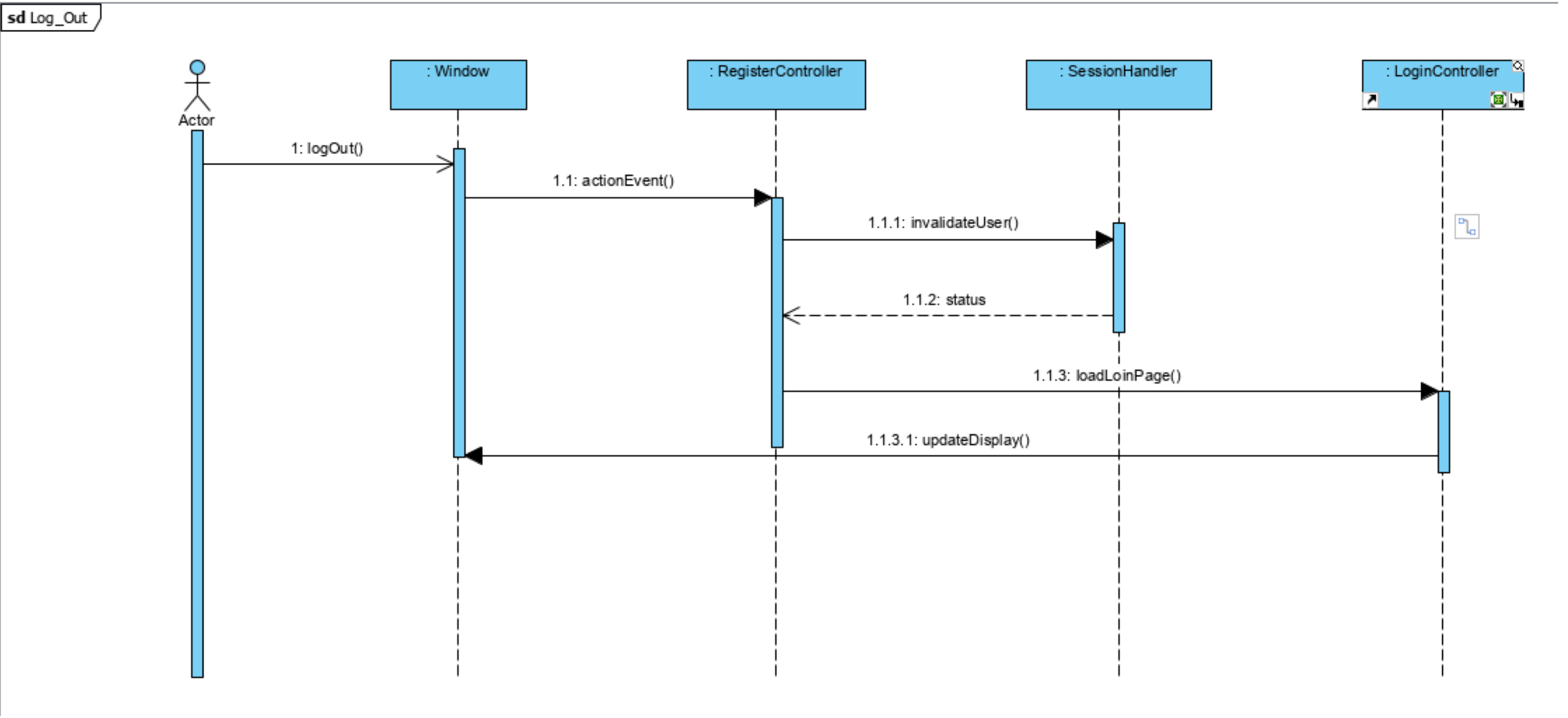

sd Rate_Review
USER
: Page
: PageController
: PageService
DB
1: rateProfessor()
1.1: alterRating(act)
1.1.1: setRating(UN,ID,act)
1.1.1.1: status = availableToRate(UN,ID)
1.1.1.2: status
alt
[status == true]
1.1.2: inputRating(UN,ID,act)
1.1.2.1: rating = updateRating(UN,ID,act)
1.1.2.2: rating
1.1.3: updateRatingDisplay(rating)
1.1.2.3: errorMsg(msg)
1.1.2.3.1: displayErrorMsg(msg)


sd LeaveReview
User
: Window
: ReviewController
: ReviewService
Database
1: createReview(rev)
1.1: actionEvent()
1.1.1: writeReview(prof, msg)
1.1.1.1: isExplicit = sanitizeReview()
opt
[if !isExplicit]
1.1.1.2: postReview(prof,msg)
1.1.1.3: status
1.1.2: status
alt
[stauts == false]
1.1.3: displayError()
1.1.4: updateViewWithRev(rev)


sd Respond_To_Message
USER
: Page
: PageController
: PageService
DB
1: sendMessage(txt)
1.1: submitMessage(txt)
1.1.1: relayMessage(from,to,txt)
1.1.1.1: message = insertMessage(from,to,txt)
1.1.1.2: message
alt
[message != null]
1.2: updateMessageBoard(message)
1.3: displayError()

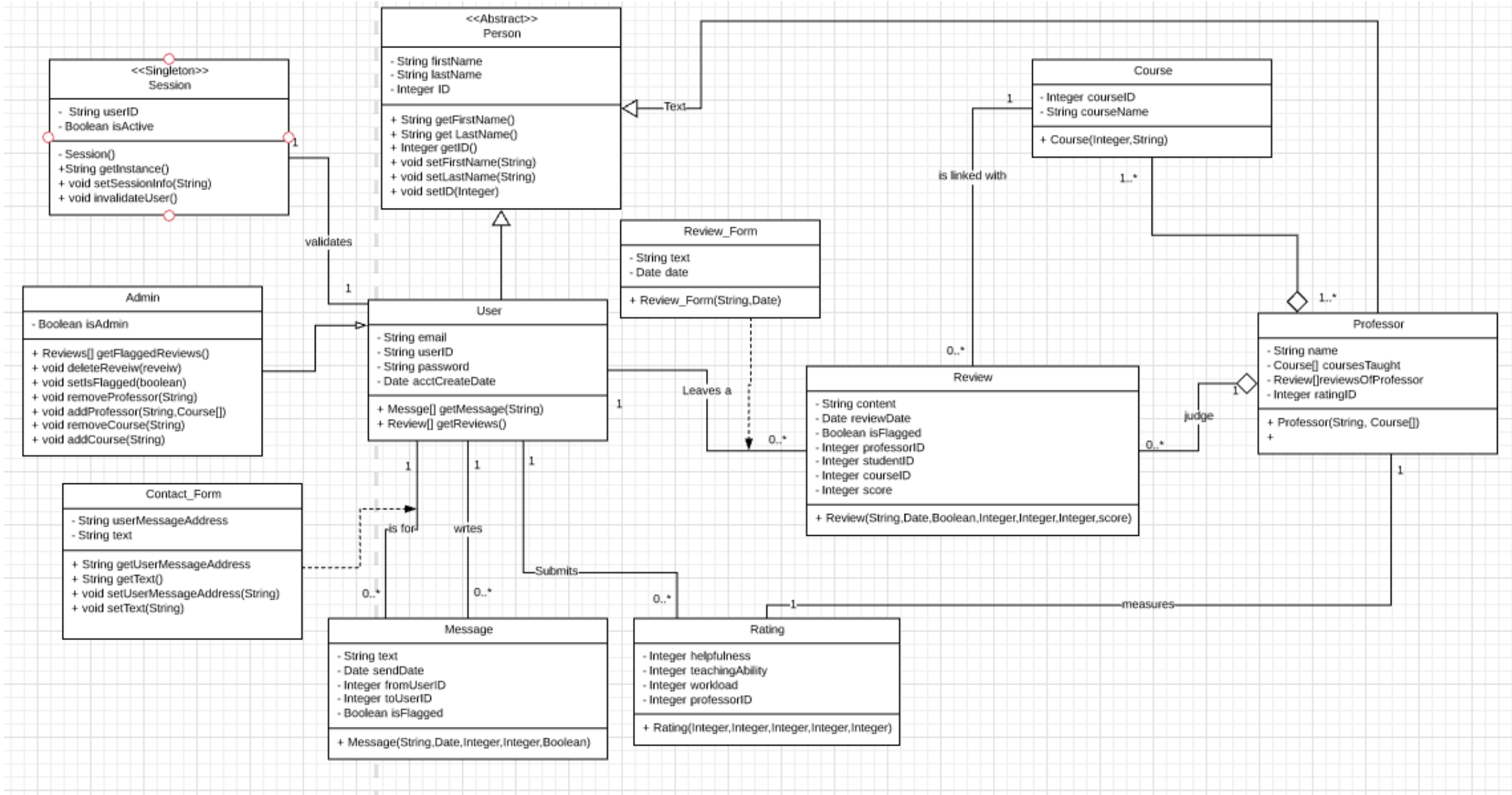

sd FlagComment
User
: Window
: ReviewController
: ReviewService
Database
1: FlagComment()
1.1: actionEvent()
1.1.1: flagComment(reason)
1.1.1.1: addCommentToAdminFlagList(reason, commentid)
1.1.1.2: status
1.1.2: status
1.1.3: updatedView()


sd Read_review
User
: Window
: SelectionController
: SelectionService
Database
: ReviewController
1: selectPage(class, professor)
1.1: actionEvent(class, professor)
1.1.1: pageData = validate(class, professor)
alt
[if pageData is valid]
1.1.1.1: reviews = getPageData(class, professor)
1.1.1.2: reviews
1.1.2: loadPage(reviews)
1.1.2.1: displayPage()
1.1.1.3: reportInvalidData()

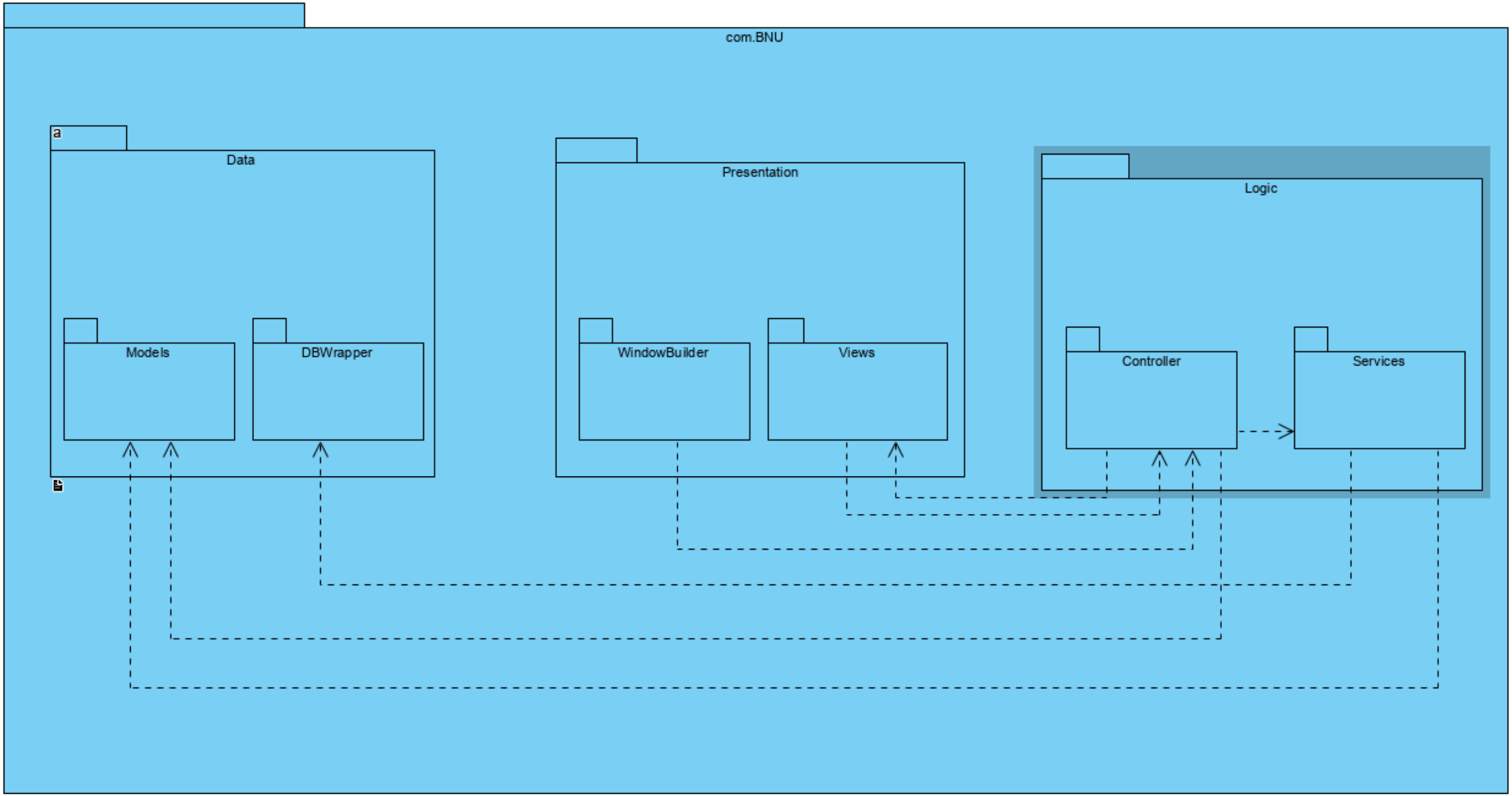

sd Log_Out
Actor
: Window
: RegisterController
: SessionHandler
: LoginController
1: logOut()
1.1: actionEvent()
1.1.1: invalidateUser()
1.1.2: status
1.1.3: loadLoinPage()
1.1.3.1: updateDisplay()

Design Class Diagram

BCC Application



Iteration 2 Package Diagram


com.BNU
a
Data
Models
DBWrapper
Presentation
WindowBuilder
Views
Logic
Controller
Services

Doxygen Project Documentation

Software_Engineering_1 - file:/C:/Users/Kevin/Documents/Software_Engineering_1/BNU/BCC/dioxy/html/annotated.html - Eclipse IDE

File Edit Navigate Search Project Run Window Help

Quick Access

Package Explorer

- html
  - search
  - _class_by_teacher_controller_8java.html
  - _class_by_teacher_model_8java.html
  - _class_by_teacher_view_8java.html
  - _course_8java.html
  - _course_list_entry_8java.html
  - _course_professor_8java.html
  - _database_mock_8java.html
  - _input_message_8java.html
  - _login_controller_8java.html
  - _login_model_8java.html
  - _login_view_8java.html
  - _main_8java.html
  - _main_controller_8java.html
  - _main_model_8java.html
  - _main_view_8java.html
  - _message_board_controller_8java.html
  - _message_board_model_8java.html
  - _message_board_view_8java.html
  - _message_model_8java.html
  - _page_controller_8java.html
  - _professor_8java.html
  - _professor_course_8java.html
  - _register_controller_8java.html
  - _register_model_8java.html
  - _register_view_8java.html
  - _review_item_8java.html
  - _teacher_list_entry_8java.html
  - _teacher_review_controller_8java.html
  - _teacher_review_model_8java.html
  - _teacher_review_view_8java.html
  - _teachers_by_class_controller_8java.html
  - _teachers_by_class_model_8java.html
  - _teachers_by_class_view_8java.html
  - _user_review_2_review_model_8java.html
  - _user_review_controller_8java.html
  - _user_review_model_8java.html
  - _user_review_view_8java.html
  - _window_builder_8java.html
  - annotated_dup.js
  - annotated.html
  - bc_s.png
  - bdwn.png
  - classcom_1_1_b_n_u_1_1database_1_1_database_
  - classcom_1_1_b_n_u_1_1database_1_1_database_
  - classcom_1_1_b_n_u_1_1database_1_1_database_
  - classcom_1_1_b_n_u_1_1database_1_1_database_
  - classcom_1_1_b_n_u_1_1pages_1_1_page_control
  - classcom_1_1_b_n_u_1_1pages_1_1_page_control
  - classcom_1_1_b_n_u_1_1pages_1_1_page_control

WorkerFileHandleLeak.java | WorkerMemory.java | WorkerCPU.java | Worker.java | WorkerPath.java | Main.java | My Project: Packages

file:///C:/Users/Kevin/Documents/Software_Engineering_1/BNU/BCC/dioxy/html/namespaces.html

My Project 1.0

Main Page | Packages | Classes | Files | Search

- My Project
  - Packages
    - Packages
  - Classes
  - Files

Packages

Here are the packages with brief descriptions (if available):

[detail level 1 2 3 4]

- com
  - BNU
    - database
    - pages
      - classesByTeacher
      - login
      - main
      - message_board
      - register
      - teacher_review
      - teachersByClass
      - UserReview
    - startmain
    - windowbuilder

Generated by doxygen 1.8.14

Console

No consoles to display at this time.

Software_Engineering_1 - file:/C:/Users/Kevin/Documents/Software_Engineering_1/BNU/BCC/dioxy/html/annotated.html - Eclipse IDE

File Edit Navigate Search Project Run Window Help

Quick Access

Package Explorer

- html
  - search
  - _class_by_teacher_controller_8java.html
  - _class_by_teacher_model_8java.html
  - _class_by_teacher_view_8java.html
  - _course_8java.html
  - _course_list_entry_8java.html
  - _course_professor_8java.html
  - _database_mock_8java.html
  - _input_message_8java.html
  - _login_controller_8java.html
  - _login_model_8java.html
  - _login_view_8java.html
  - _main_8java.html
  - _main_controller_8java.html
  - _main_model_8java.html
  - _main_view_8java.html
  - _message_board_controller_8java.html
  - _message_board_model_8java.html
  - _message_board_view_8java.html
  - _message_model_8java.html
  - _page_controller_8java.html
  - _professor_8java.html
  - _professor_course_8java.html
  - _register_controller_8java.html
  - _register_model_8java.html
  - _register_view_8java.html
  - _review_item_8java.html
  - _teacher_list_entry_8java.html
  - _teacher_review_controller_8java.html
  - _teacher_review_model_8java.html
  - _teacher_review_view_8java.html
  - _teachers_by_class_controller_8java.html
  - _teachers_by_class_model_8java.html
  - _teachers_by_class_view_8java.html
  - _user_review_2_review_model_8java.html
  - _user_review_controller_8java.html
  - _user_review_model_8java.html
  - _user_review_view_8java.html
  - _window_builder_8java.html
  - annotated_dup.js
  - annotated.html
  - bc_s.png
  - bdwn.png
  - classcom_1_1_b_n_u_1_1database_1_1_database_
  - classcom_1_1_b_n_u_1_1database_1_1_database_
  - classcom_1_1_b_n_u_1_1database_1_1_database_
  - classcom_1_1_b_n_u_1_1database_1_1_database_
  - classcom_1_1_b_n_u_1_1pages_1_1_page_control
  - classcom_1_1_b_n_u_1_1pages_1_1_page_control
  - classcom_1_1_b_n_u_1_1pages_1_1_page_control

WorkerFileHandleLeak.java | WorkerMemory.java | WorkerCPU.java | Worker.java | WorkerPath.java | Main.java | My Project: Class List

file:///C:/Users/Kevin/Documents/Software_Engineering_1/BNU/BCC/dioxy/html/annotated.html

My Project 1.0

Main Page | Packages | Classes | Files | Search

- My Project
  - Packages
  - Classes
    - Class List
    - Class Index
    - Class Hierarchy
    - Class Members
  - Files

Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

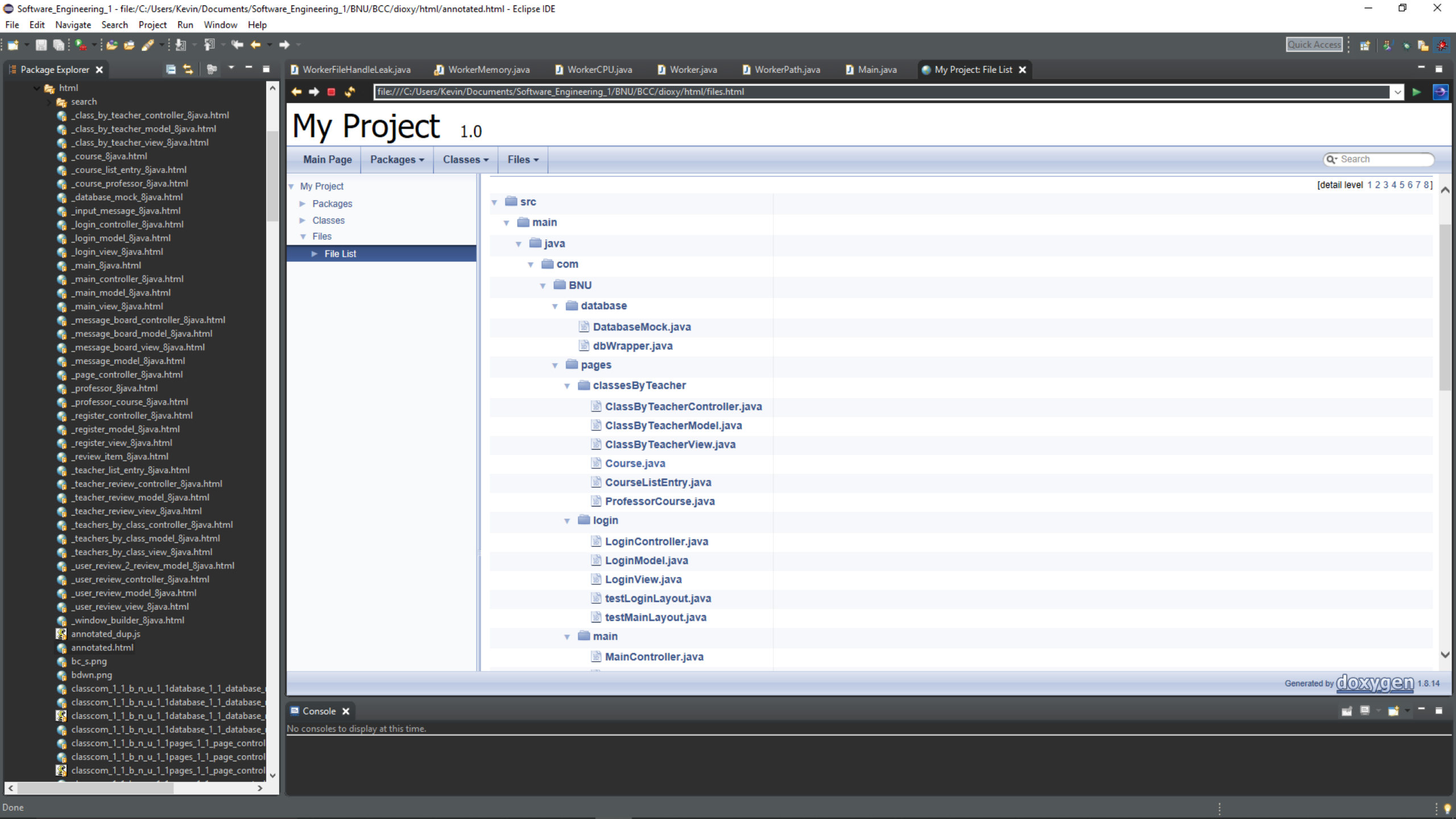
[detail level 1 2 3 4 5]

- com
  - BNU
    - database
      - DatabaseMock
      - dbWrapper
    - pages
      - classesByTeacher
        - ClassByTeacherController
        - ClassByTeacherModel
        - ClassByTeacherView
        - Course
        - CourseListEntry
        - ProfessorCourse
      - login
        - LoginController
        - LoginModel
        - LoginView
        - testLoginLayout
        - testMainLayout
      - main

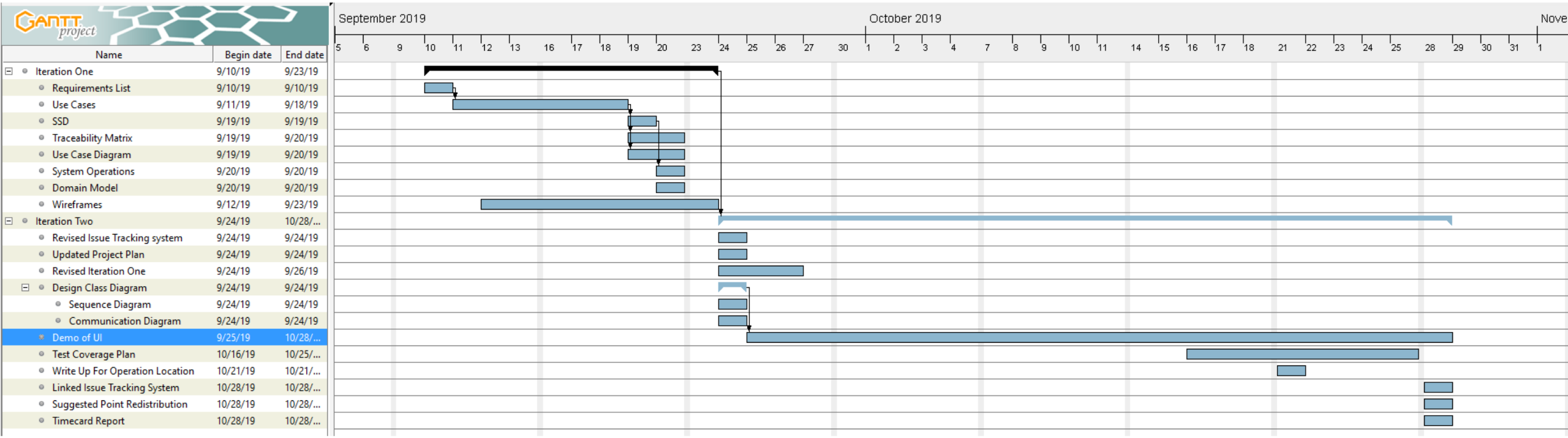
Generated by doxygen 1.8.14

Console

No consoles to display at this time.


Software_Engineering_1 - file:/C:/Users/Kevin/Documents/Software_Engineering_1/BNU/BCC/dioxy/html/annotated.html - Eclipse IDE
File Edit Navigate Search Project Run Window Help
Quick Access
Package Explorer
html
search
_class_by_teacher_controller_8java.html
_class_by_teacher_model_8java.html
_class_by_teacher_view_8java.html
_course_8java.html
_course_list_entry_8java.html
_course_professor_8java.html
_database_mock_8java.html
_input_message_8java.html
_login_controller_8java.html
_login_model_8java.html
_login_view_8java.html
_main_8java.html
_main_controller_8java.html
_main_model_8java.html
_main_view_8java.html
_message_board_controller_8java.html
_message_board_model_8java.html
_message_board_view_8java.html
_message_model_8java.html
_page_controller_8java.html
_professor_8java.html
_professor_course_8java.html
_register_controller_8java.html
_register_model_8java.html
_register_view_8java.html
_review_item_8java.html
_teacher_list_entry_8java.html
_teacher_review_controller_8java.html
_teacher_review_model_8java.html
_teacher_review_view_8java.html
_teachers_by_class_controller_8java.html
_teachers_by_class_model_8java.html
_teachers_by_class_view_8java.html
_user_review_2_review_model_8java.html
_user_review_controller_8java.html
_user_review_model_8java.html
_user_review_view_8java.html
_window_builder_8java.html
annotated_dup.js
annotated.html
bc_s.png
bdwn.png
WorkerFileHandleLeak.java
WorkerMemory.java
WorkerCPU.java
Worker.java
WorkerPath.java
Main.java
My Project: File List
file:///C:/Users/Kevin/Documents/Software_Engineering_1/BNU/BCC/dioxy/html/files.html
My Project 1.0
Main Page
Packages
Classes
Files
Search
My Project
Packages
Classes
Files
File List
[detail level 1 2 3 4 5 6 7 8]
src
main
java
com
BNU
database
DatabaseMock.java
dbWrapper.java
pages
classesByTeacher
ClassByTeacherController.java
ClassByTeacherModel.java
ClassByTeacherView.java
Course.java
CourseListEntry.java
ProfessorCourse.java
login
LoginController.java
LoginModel.java
LoginView.java
testLoginLayout.java
testMainLayout.java
main
MainController.java
Generated by doxygen 1.8.14
Console
No consoles to display at this time.
Done

Project Planning


GANTT project
Name
Begin date
End date
Iteration One 9/10/19 9/23/19
Requirements List 9/10/19 9/10/19
Use Cases 9/11/19 9/18/19
SSD 9/19/19 9/19/19
Traceability Matrix 9/19/19 9/20/19
Use Case Diagram 9/19/19 9/20/19
System Operations 9/20/19 9/20/19
Domain Model 9/20/19 9/20/19
Wireframes 9/12/19 9/23/19
Iteration Two 9/24/19 10/28/...
Revised Issue Tracking system 9/24/19 9/24/19
Updated Project Plan 9/24/19 9/24/19
Revised Iteration One 9/24/19 9/26/19
Design Class Diagram 9/24/19 9/24/19
Sequence Diagram 9/24/19 9/24/19
Communication Diagram 9/24/19 9/24/19
Demo of UI 9/25/19 10/28/...
Test Coverage Plan 10/16/19 10/25/...
Write Up For Operation Location 10/21/19 10/21/...
Linked Issue Tracking System 10/28/19 10/28/...
Suggested Point Redistribution 10/28/19 10/28/...
Timecard Report 10/28/19 10/28/...
September 2019
October 2019
Nove
5 6 9 10 11 12 13 16 17 18 19 20 23 24 25 26 27 30 1 2 3 4 7 8 9 10 11 14 15 16 17 18 21 22 23 24 25 28 29 30 31 1

Grasp Justification

Each page is an individual package containing all the functionality needed required to load that page. Using the MVC model we are able to atomize the datatypes, logic, and view of each page.

Each pages Controller contains instances of the view, model, and panel that that page will display, as well as a reference to the database where data can be fetched for displaying.

Each Model contains the datatypes required to build the page. If the model represents a Java Swing pane then it contains mostly the JComponents that will be displayed on that page.

Each View builds the model JComponents into the pane that is stored by the Controller.

The database is currently wrapped with an API that will mimic the database when it is fully implemented. However it is currently hard-coded.

Coverage Plan

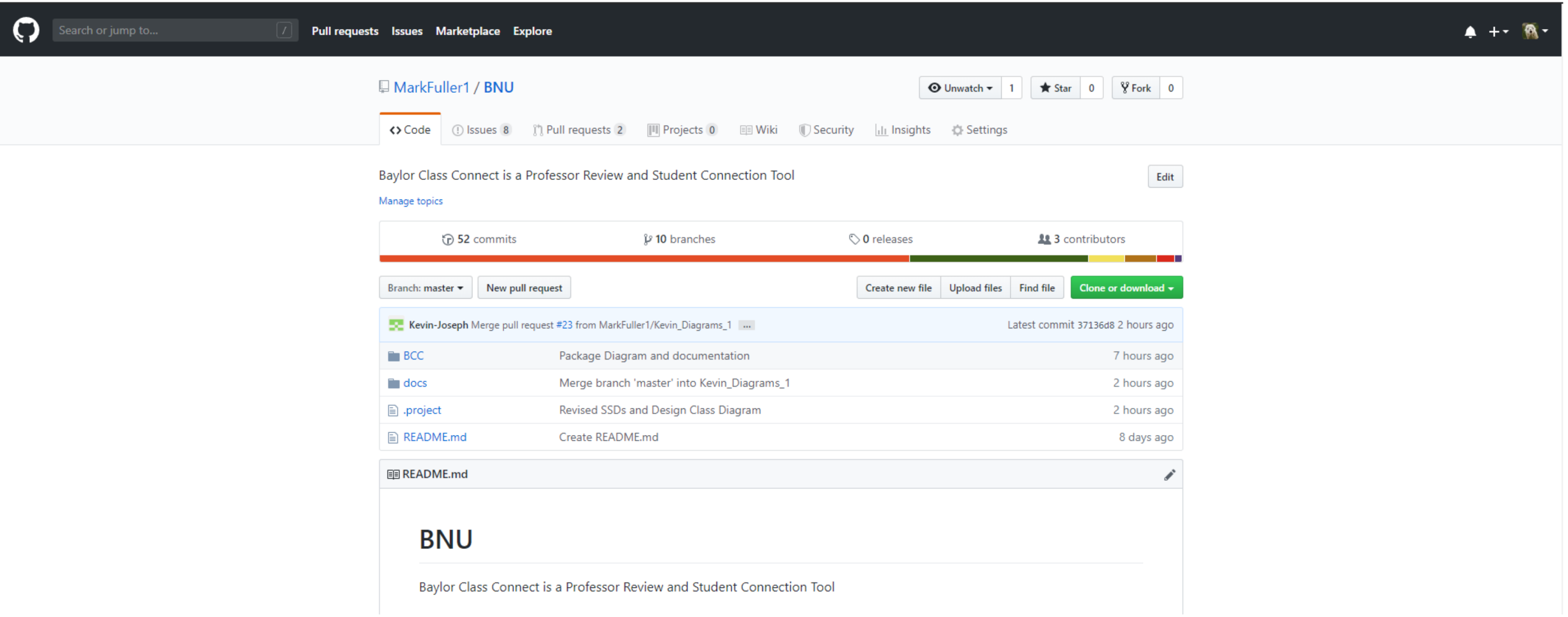
Each Controller and View will have test classes dedicated to them.

The Model currently has no implementation other than getters and setters, therefore does not need a testing suite.

The View of each page will be tested for loading errors by mocking the Controller for values.

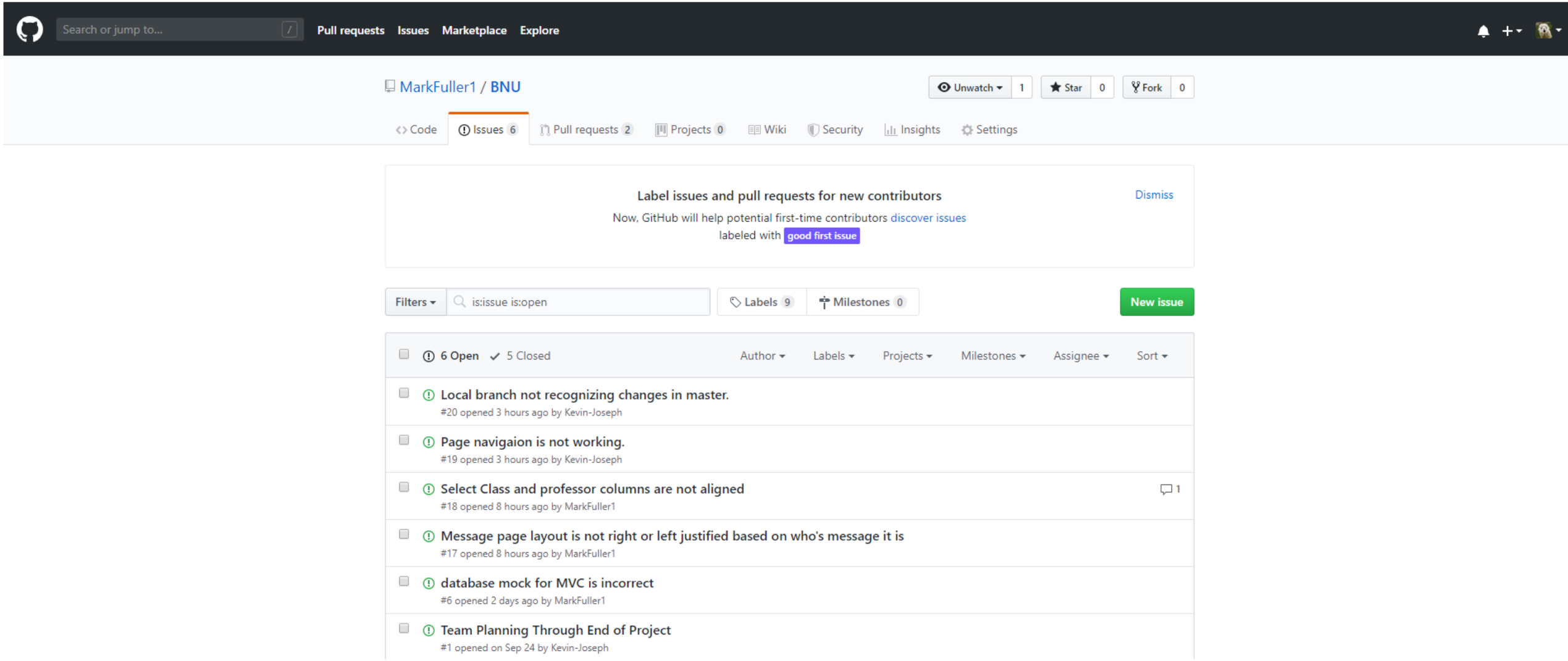
The Controller will be tested using a mocked view to check for valid GUI generation and exception handling, It will also have tests for the functionality it contains within itself for database query routing.

Git Repository

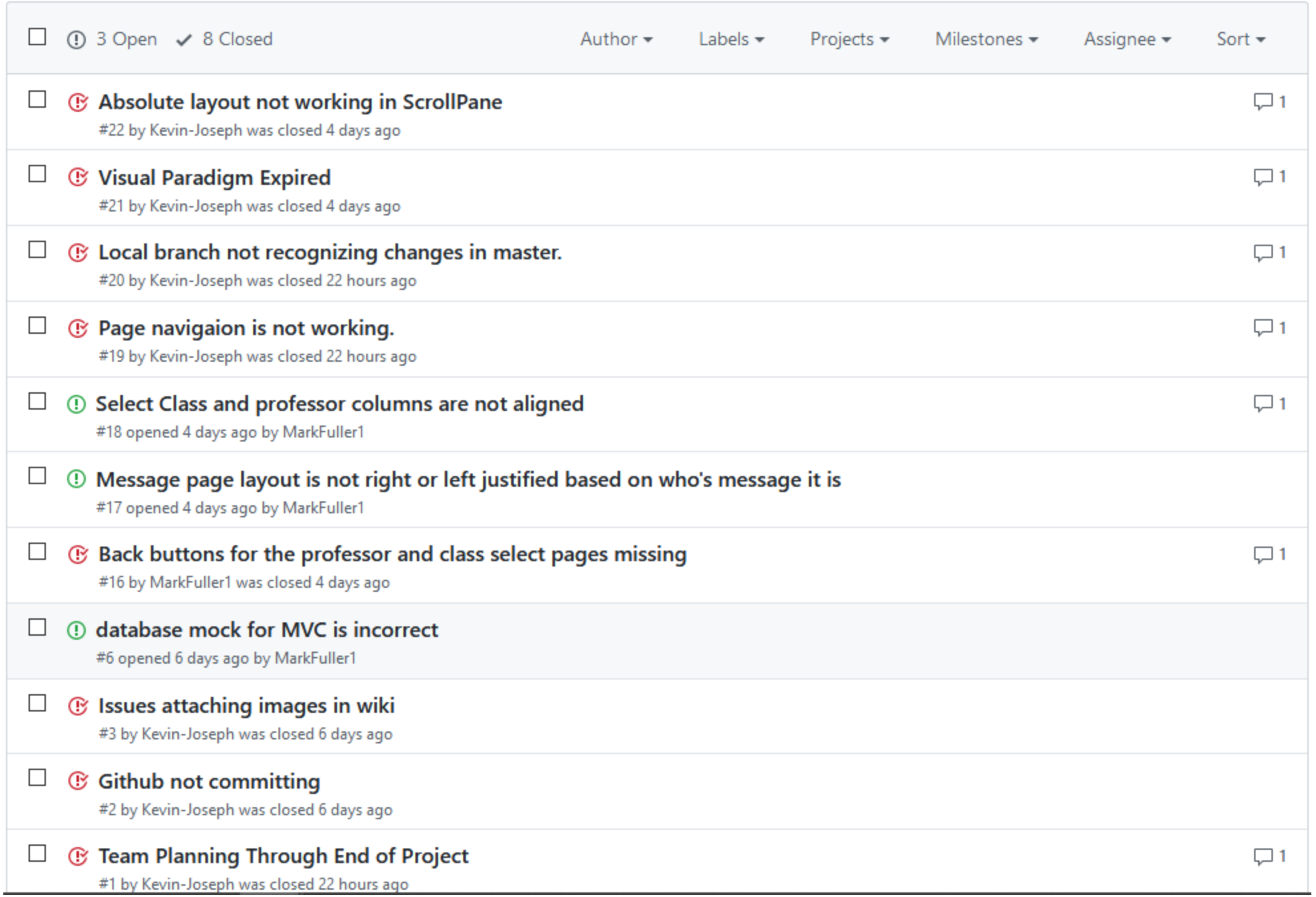


Repository Link: https://github.com/MarkFuller1/BNU/

Issue Tracking



Repository Link: https://github.com/MarkFuller1/BNU/issues



Repository Link: https://github.com/MarkFuller1/BNU/issues

BCC Iteration 3

Mark Fuller

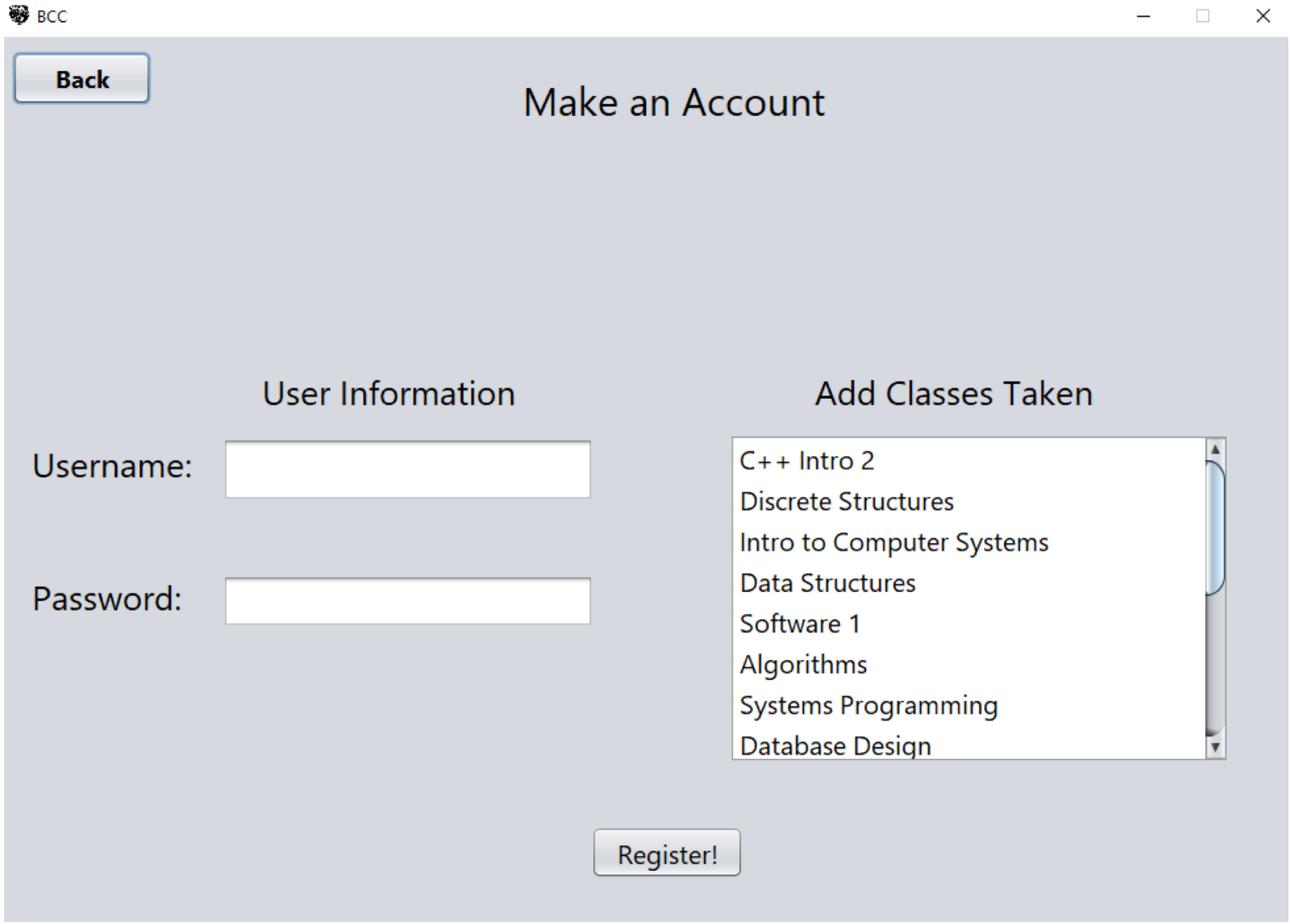
Kevin Kulda

Connor Woodahl

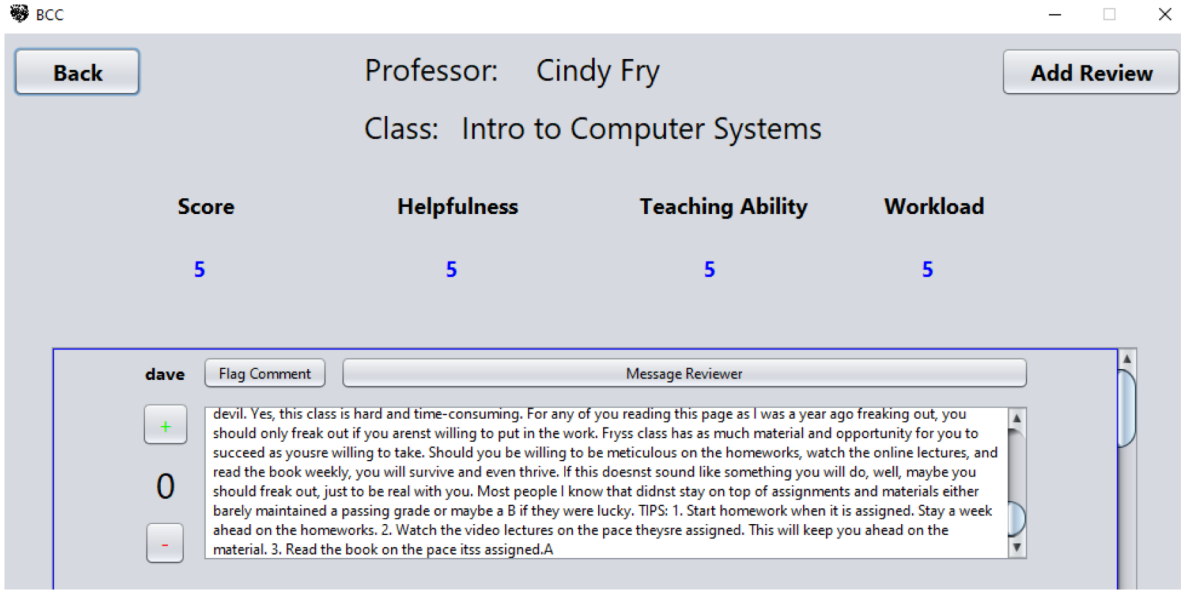
Project Repository: https://github.com/MarkFuller1/BNU

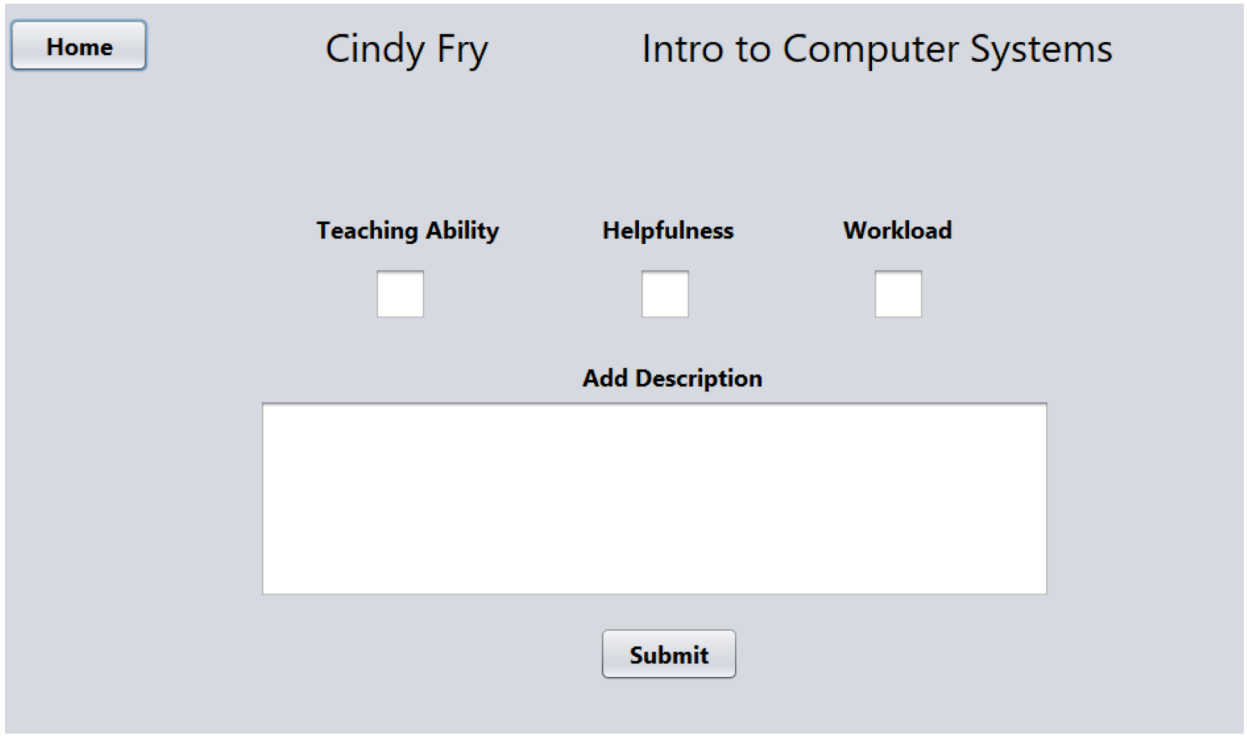
Use Case Screenshots from GUI

Make Account

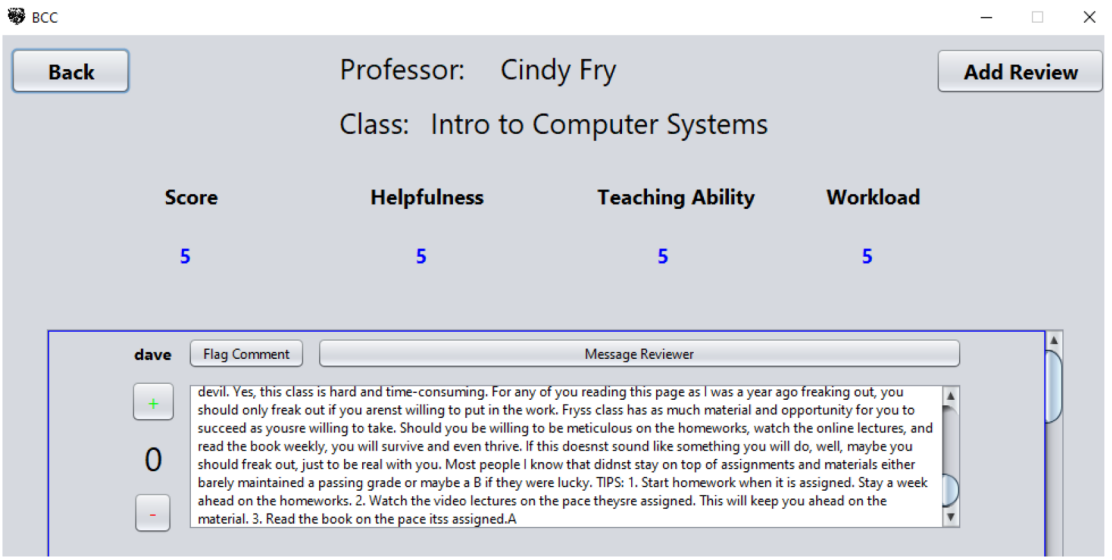


Leave Review

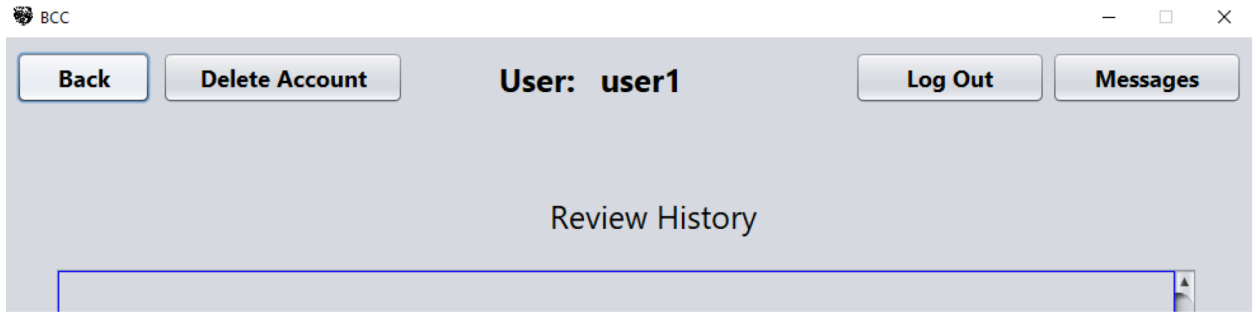




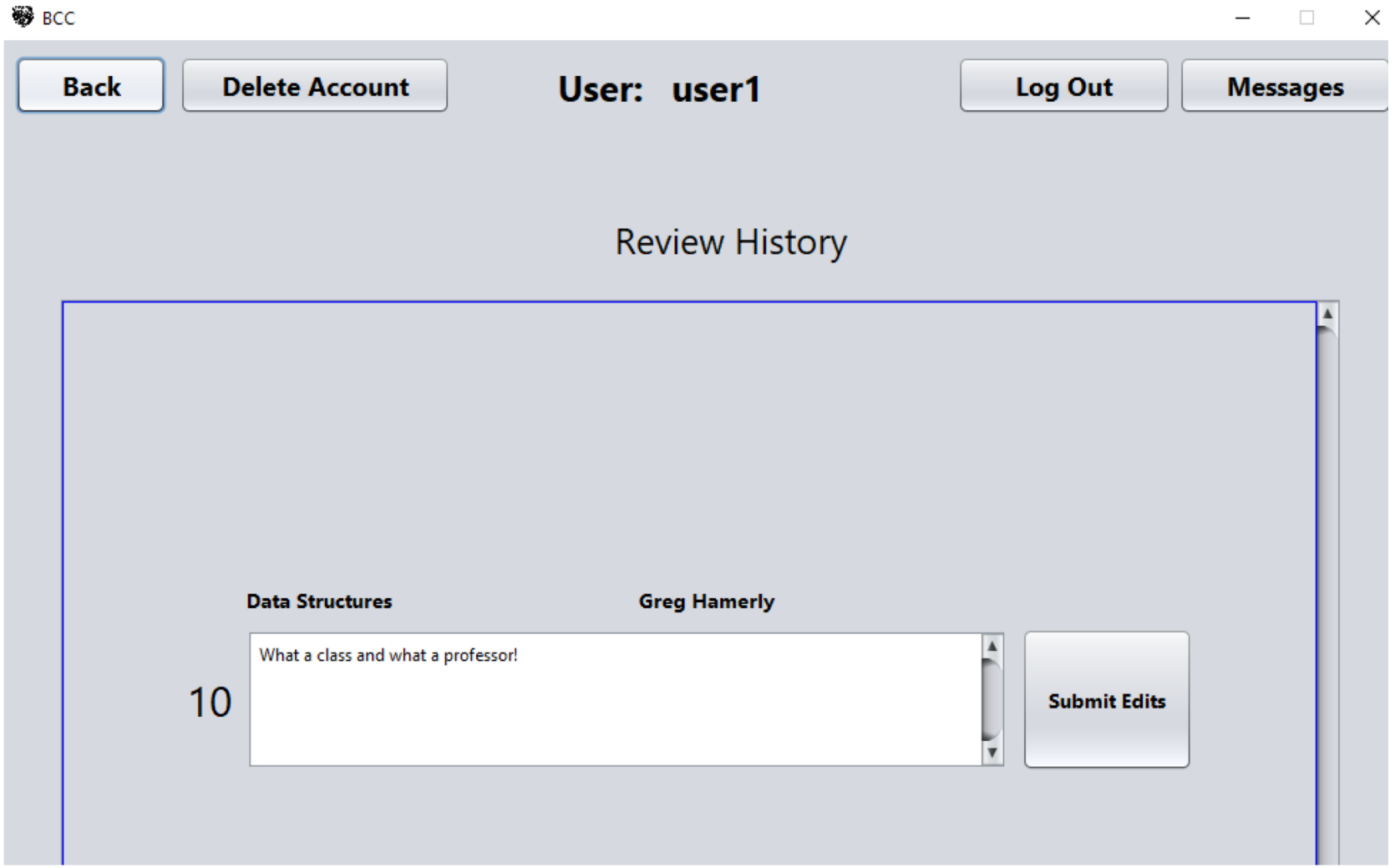
Contact Student



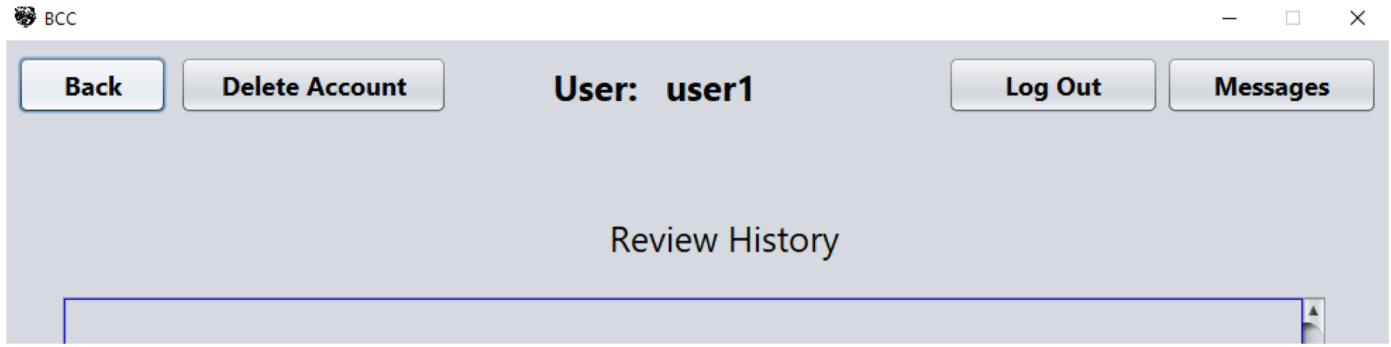
Respond to a Message

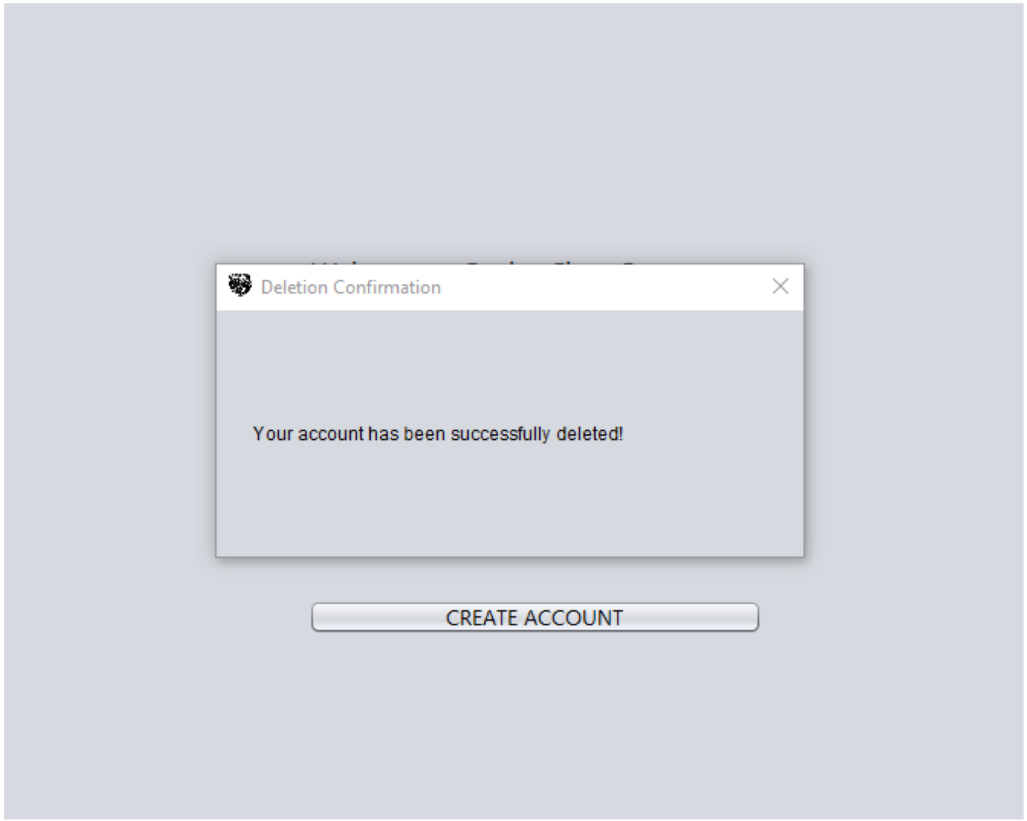


Edit Review

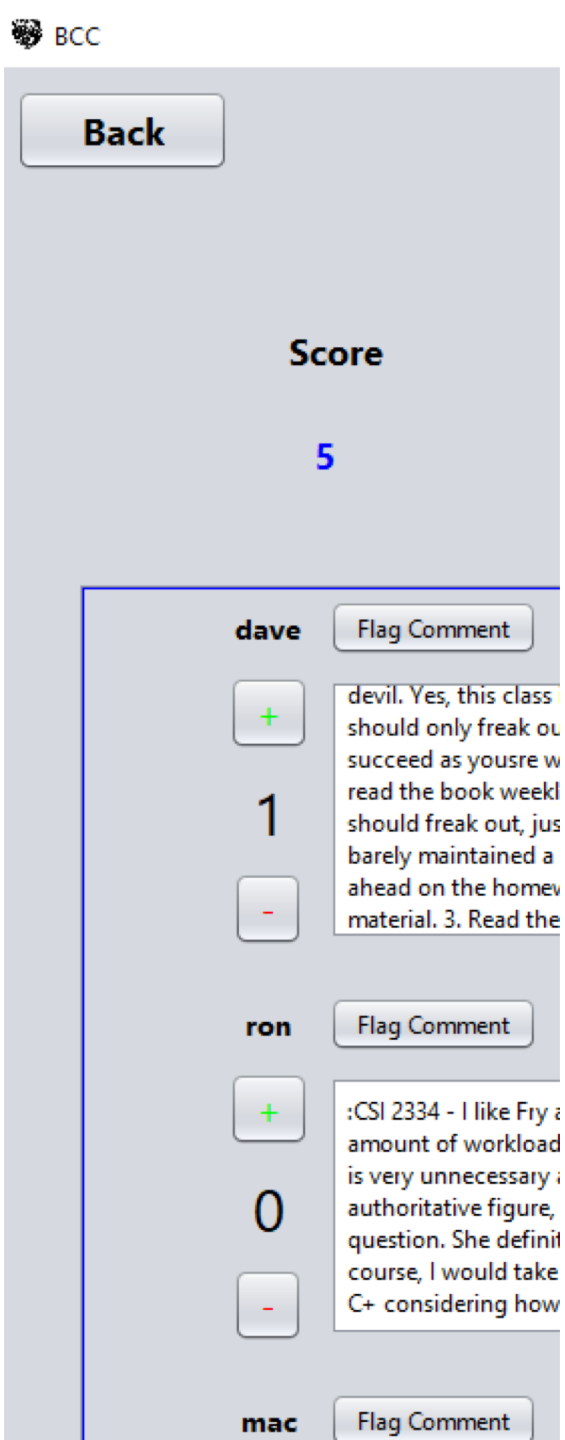

BCC
Back
Delete Account
User: user1
Log Out
Messages
Review History
Data Structures
Greg Hamerly
10
What a class and what a professor!
Submit Edits

Delete Account

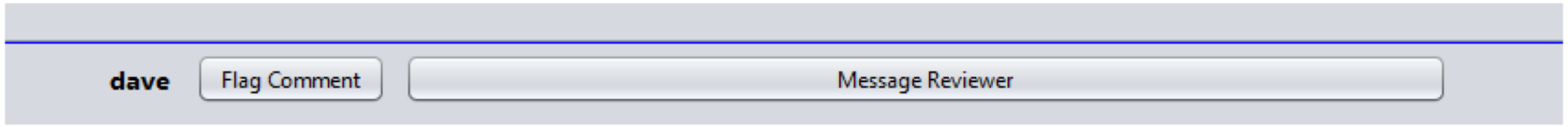




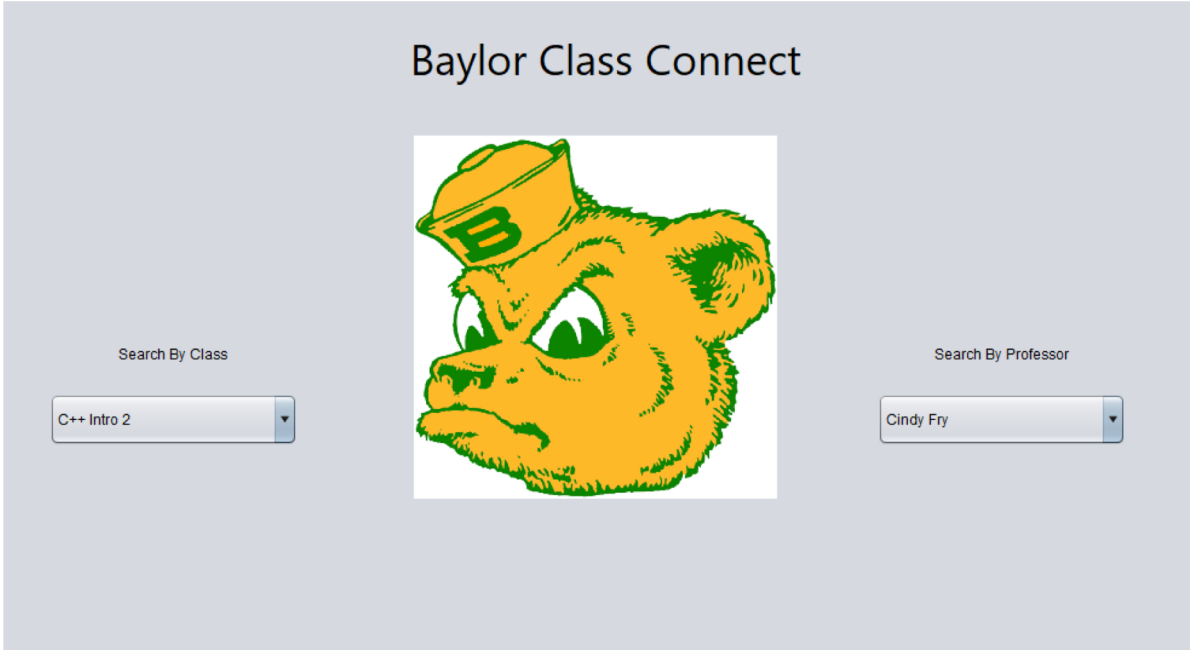
Rate Review

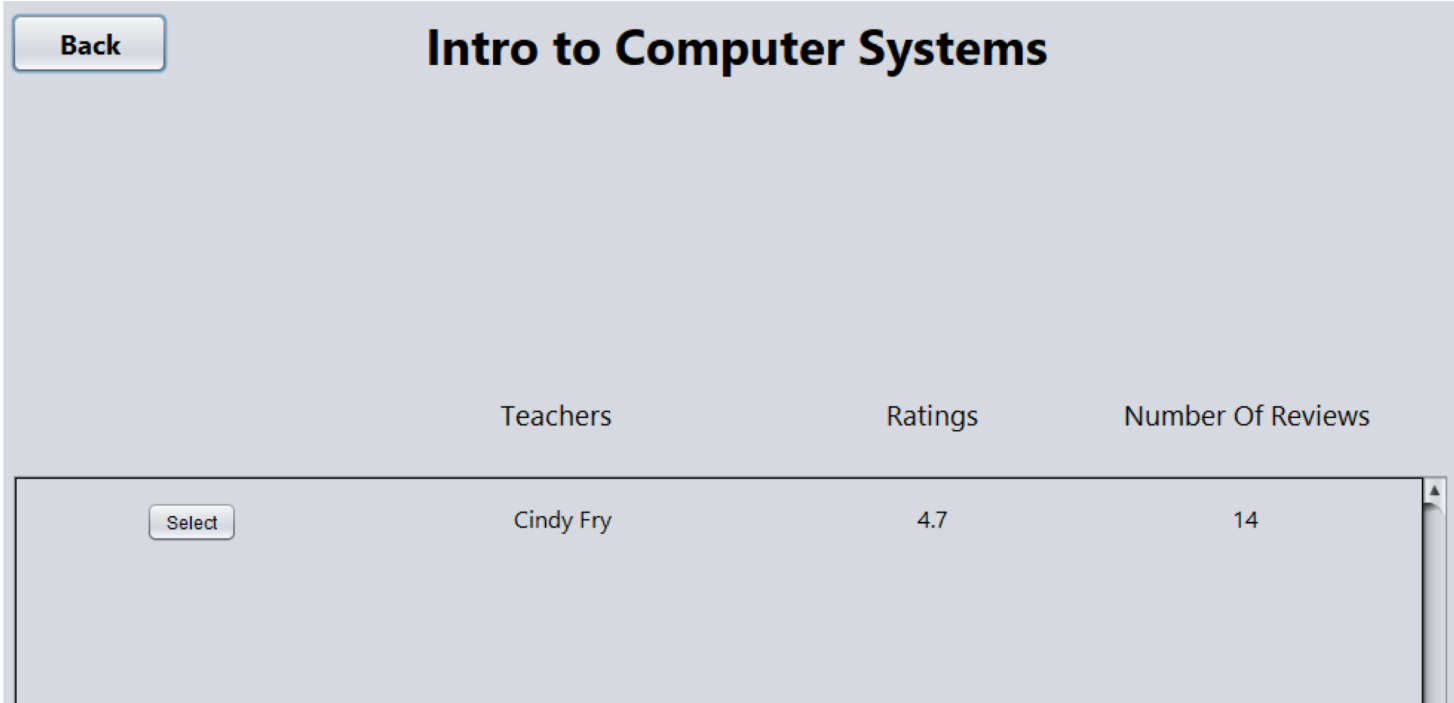

BCC
Back
Score
5
dave
Flag Comment
+
devil. Yes, this class
should only freak ou
succeed as yousre w
read the book weekl
should freak out, jus
barely maintained a
ahead on the homev
material. 3. Read the
1
-
ron
Flag Comment
+
:CSI 2334 - I like Fry a
amount of workload
is very unnecessary
authoritative figure,
question. She definit
course, I would take
C+ considering how
0
-
mac
Flag Comment

Flag Comment

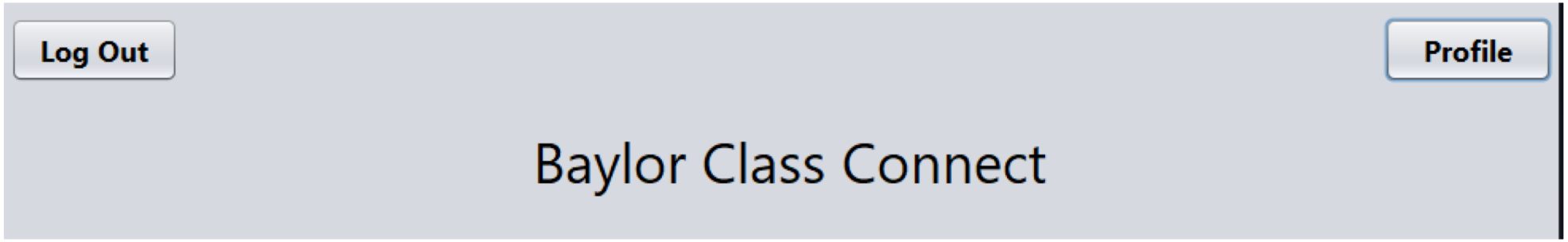


Read Review


Baylor Class Connect
Search By Class
C++ Intro 2
Search By Professor
Cindy Fry


Back
Intro to Computer Systems
Teachers
Ratings
Number Of Reviews
Select
Cindy Fry
4.7
14

Log Out



Link to Use Case Video

https://youtu.be/wOC9_yUQSi8

Link to Application User Guide

https://github.com/MarkFuller1/BNU/wiki/User_Guide_For_BCC

User_Guide_For_BCC

Edit New Page

Kevin-Joseph edited this page 3 minutes ago · 11 revisions

USER GUIDE

BCC

Mark Fuller

Kevin Kulda

Connor Woodahl

BCC is an application to read and write reviews about Baylor Computer Science professors. Additionally, users may upvote, downvote, and flag comments, and may message other users. The following functions are necessary to fully utilize the BCC application: register account, login, find class reviews, flag a review, upvote and downvote a review, write a review, message a user, view your reviews, and delete account.

- How To Run The Application: Navigate to the jar page on this wiki [https://github.com /MarkFuller1/BNU/wiki/jar] and download the application jar. Next, open a terminal/command prompt and navigate to the location of the jar. In the terminal enter the following command to run the jar: >java -jar [jar file name]

- Register Account: When you navigate to the BCC application you will see the login page. If you have login credentials you may enter them to gain access to the application. If you do not have login credentials but want access to the application, you must register an account. To register an account select the "Create Account" button on the login screen. You will then be taken to the registration page.

+ Add a custom sidebar

Link to Download Project Jar

https://github.com/MarkFuller1/BNU/wiki/jar

Testing

Command Prompt

```
INFO: Login page loaded correctly
Dec 04, 2019 10:53:30 PM BNU.presentation.LoginView BuildLoginView
INFO: Login page loaded correctly
Dec 04, 2019 10:53:30 PM BNU.data.database.DatabaseApi isAdminImpl
WARNING:
[INFO] Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.749 s - in BNU.logic.service.proxy.ClassTest
[INFO] Running BNU.logic.service.RegisterServiceTest
Dec 04, 2019 10:53:30 PM BNU.presentation.LoginView BuildLoginView
INFO: Login page loaded correctly
Dec 04, 2019 10:53:30 PM BNU.data.database.DatabaseApi validateUserImpl
INFO: SELECT * FROM users WHERE users.user_name = 'admin' AND users.password = 'admin'
admin admin
Dec 04, 2019 10:53:30 PM BNU.presentation.LoginView BuildLoginView
INFO: Login page loaded correctly
Dec 04, 2019 10:53:30 PM BNU.data.database.DatabaseApi validateUserImpl
INFO: SELECT * FROM users WHERE users.user_name = 'user3' AND users.password = 'kk'
user3 kk
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.59 s - in BNU.logic.service.RegisterServiceTest
[INFO] Running BNU.logic.service.TeacherByClassServiceTest
Dec 04, 2019 10:53:30 PM BNU.presentation.LoginView BuildLoginView
INFO: Login page loaded correctly
Dec 04, 2019 10:53:31 PM BNU.data.database.DatabaseApi getAllProfessorsForClassImpl
INFO: SELECT first_name, last_name from course, professor_course, professor WHERE course.title = 'Algorithms' AND course.course_id_pk = professor_course.course_id AND professor.professor_id_pk = professor_course.professor_id
Dec 04, 2019 10:53:31 PM BNU.presentation.LoginView BuildLoginView
INFO: Login page loaded correctly
Dec 04, 2019 10:53:31 PM BNU.data.database.DatabaseApi getAllTeacherInfoByCourseImpl
INFO: select professor.first_name, professor.last_name, review.* FROM review, professor, course, professor_course WHERE professor.professor_id_pk = review.professor_id  AND course.title = 'Algorithms' AND course.course_id_pk = professor_
course.course_id  AND review.course_id = professor_course.course_id
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.64 s - in BNU.logic.service.TeacherByClassServiceTest
[INFO] Running BNU.logic.service.TeacherReviewViewServiceTest
Dec 04, 2019 10:53:31 PM BNU.presentation.LoginView BuildLoginView
INFO: Login page loaded correctly
Dec 04, 2019 10:53:31 PM BNU.data.database.DatabaseApi getAllReviewsForTeacherClassImpl
INFO: select content, review.score as score, user_name, review_id_pk  FROM review, professor, course  WHERE professor.first_name = 'Bill'  AND professor.last_name = 'Booth'  AND professor_id = professor.professor_id_pk  AND course.title
= 'Algorithms'  AND review.course_id = course.course_id_pk
Dec 04, 2019 10:53:32 PM BNU.presentation.LoginView BuildLoginView
INFO: Login page loaded correctly
Dec 04, 2019 10:53:32 PM BNU.data.database.DatabaseApi getOverallProfessorRatingsImpl
INFO: SELECT Avg(x.teaching_ability) as ta, Avg(x.workload) as w,              Avg(x.helpfulness) as h,                    ((avg(x.teaching_ability) + avg(x.workload) + avg(x.helpfulness)) / 3) as avg                  FROM   (SELEC
T review.professor_id,                review.teaching_ability,              review.workload,              review.helpfulness          FROM   review,              professor          WHERE  professor.first_name = 'Bill'
       AND professor.last_name = 'Booth'              AND review.professor_id = professor.professor_id_pk) AS x
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.694 s - in BNU.logic.service.TeacherReviewViewServiceTest
[INFO] Running BNU.logic.service.UserReviewServiceTest
Dec 04, 2019 10:53:32 PM BNU.presentation.LoginView BuildLoginView
INFO: Login page loaded correctly
Dec 04, 2019 10:53:32 PM BNU.data.database.DatabaseApi getAllReviewsForUserImpl
INFO: select content, (teaching_ability + helpfulness + workload) / 3 as avg,  course.title, first_name, last_name, review_id_pk  from review, professor, course Where user_name = 'user1'  AND review.professor_id = professor.professor_id_
pk AND review.course_id = course.course_id_pk
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.34 s - in BNU.logic.service.UserReviewServiceTest
[INFO] Running BNU.singleton.ClassTest
[INFO] Tests run: 5, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.01 s - in BNU.singleton.ClassTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 72, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  13.406 s
[INFO] Finished at: 2019-12-04T22:53:33-06:00
[INFO] ------------------------------------------------------------------------

C:\Users\Kevin\Documents\Software_Engineering_1\BNU\BNU\BCC>git status
On branch kevin_service_2
Your branch is ahead of 'origin/kevin service 2' by 22 commits.
```

Note: tests that are dependent on data in the database may fail if we altered the data during development.

Software_Engineering_1 - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Package Explorer

- BCC [BNU kevin_service_2] (59)
  - src/main/java (59)
  - src/test/java
    - BNU.data.database
      - DatabaseApiTest.java
    - BNU.logic.service
      - ClassByTeacherServiceTest.java
      - LoginServiceTest.java
      - MainServiceTest.java
      - MessageBoardServiceTest.java
      - RegisterServiceTest.java
      - TeacherByClassServiceTest.java
      - TeacherReviewViewServiceTest.java
      - UserReviewServiceTest.java
    - BNU.logic.service.proxy
      - ClassTest.java
    - BNU.singleton
      - ClassTest.java
  - JRE System Library [JavaSE-1.8]
  - Maven Dependencies
  - JUnit 4
  - dioxy
  - resources
  - src
  - target
  - BCC.doxyfile
  - BCC.log
  - BCC.log.1
  - BCC.log.2
  - BCC.log.3
  - BCC.log.4
  - BCC.log.5
  - BCC.log.6
  - BCC.log.7
  - BCC.log.8
  - database.log
  - log.log
  - log.log.1
  - log.log.2
  - log.log.3
  - pom.xml

Software_Engineering_1 - BCC/src/test/java/BNU/singleton/ClassTest.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Package Explorer

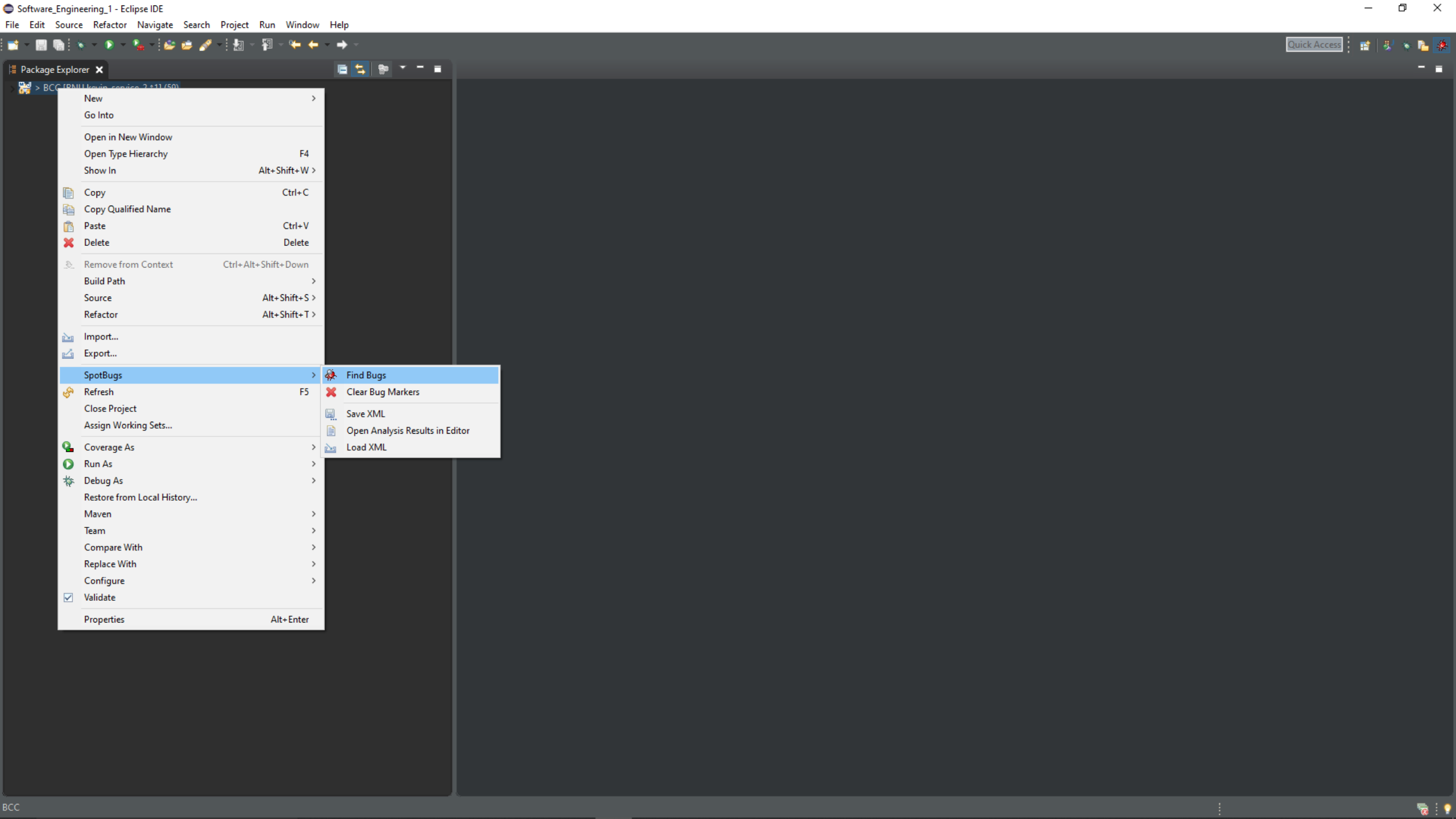
- BCC [BNU kevin_service_2] (59)
  - src/main/java (59)
  - src/test/java
    - BNU.data.database
      - DatabaseApiTest.java
    - BNU.logic.service
      - ClassByTeacherServiceTest.java
      - LoginServiceTest.java
      - MainServiceTest.java
      - MessageBoardServiceTest.java
      - RegisterServiceTest.java
      - TeacherByClassServiceTest.java
      - TeacherReviewViewServiceTest.java
      - UserReviewServiceTest.java
    - BNU.logic.service.proxy
      - ClassTest.java
    - BNU.singleton
      - ClassTest.java
  - JRE System Library [JavaSE-1.8]
  - Maven Dependencies
  - JUnit 4
  - dioxy
  - resources
  - src
  - target
  - BCC.doxyfile
  - BCC.log
  - BCC.log.1
  - BCC.log.2
  - BCC.log.3
  - BCC.log.4
  - BCC.log.5
  - BCC.log.6
  - BCC.log.7
  - BCC.log.8
  - database.log
  - log.log
  - log.log.1
  - log.log.2
  - log.log.3
  - pom.xml

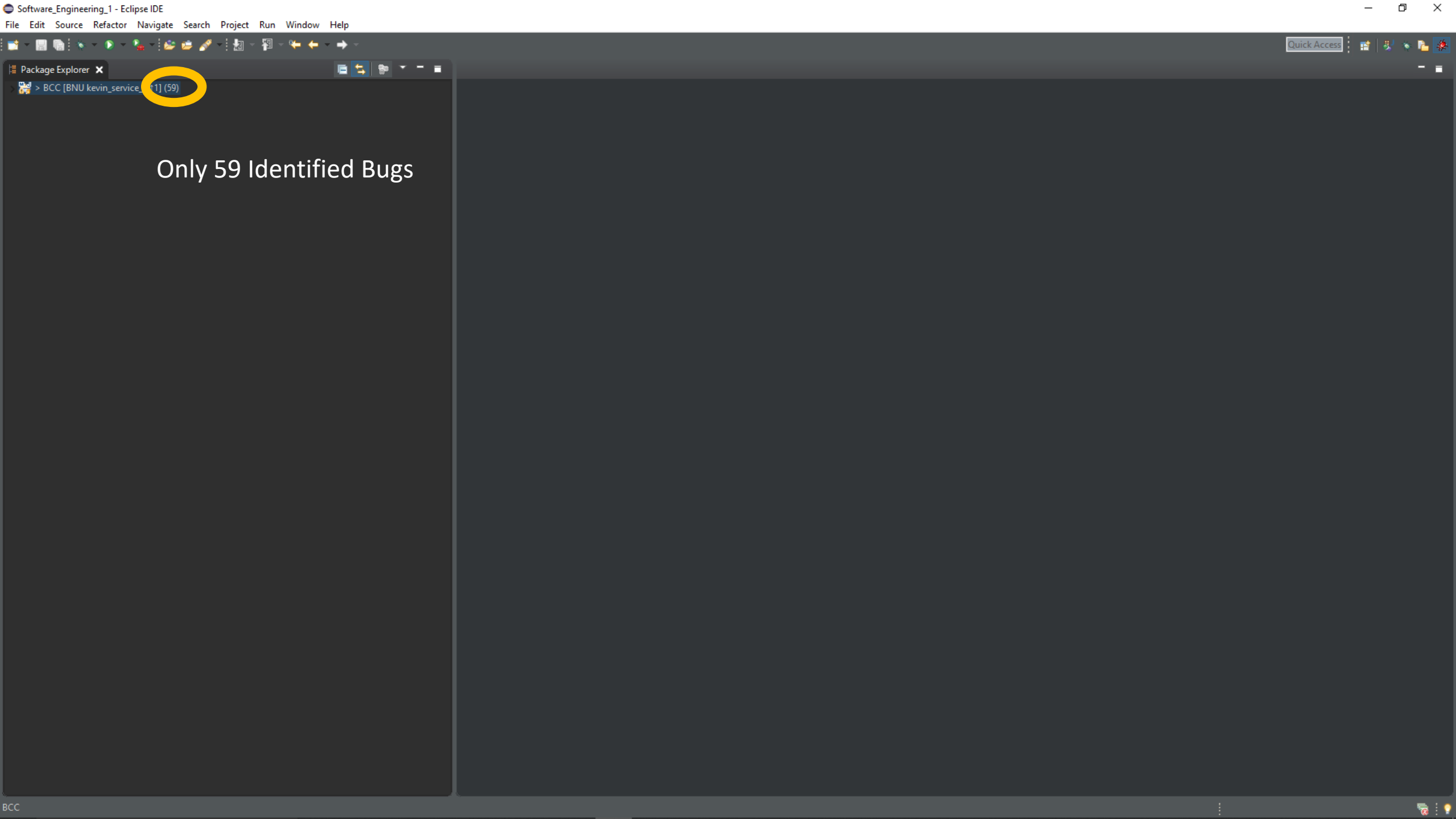
ClassTest.java

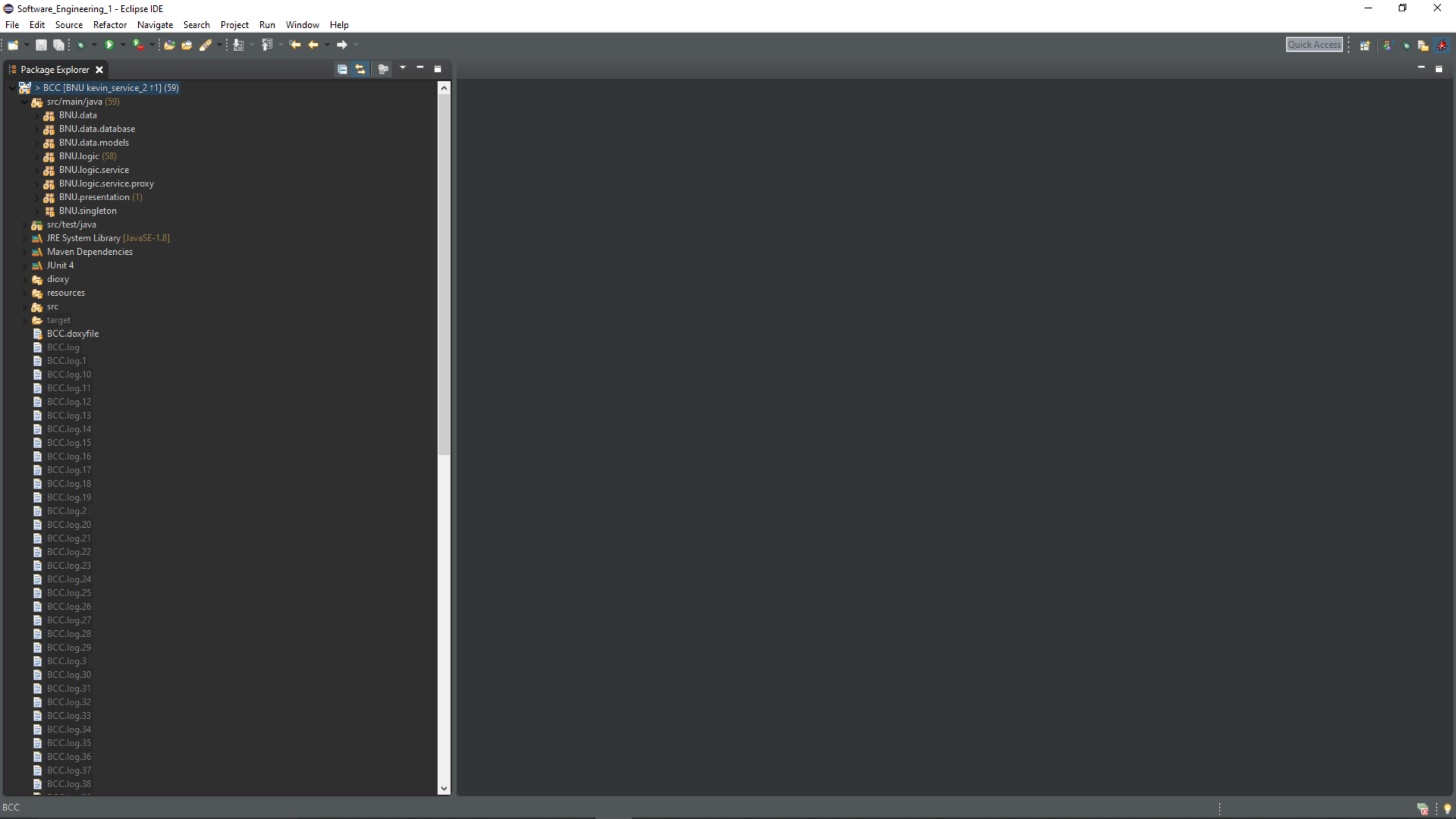
```java
package BNU.singleton;

import static org.junit.Assert.assertFalse;



public class ClassTest {

    @Before
    public void setUp() {
        SingletonSession session = SingletonSession.getInstance("User", true);
    }

    @Test
    public void testSingletonGetInstance() throws Exception {
        SingletonSession.getInstance();
        //SingletonSession session = SingletonSession.getInstance("User", true);
        assertTrue(SingletonSession.getUserName().equals("User"));
        SingletonSession.getInstance();
        assertTrue(SingletonSession.isAdmin());
    }

    @Test
    public void testGetUserName() throws Exception {
        SingletonSession.getInstance();
        assertTrue(SingletonSession.getUserName().equals("User"));
    }

    @Test
    public void testGetAdmin() throws Exception {
        SingletonSession.getInstance();
        assertTrue(SingletonSession.isAdmin());
    }

    @Test
    public void testSetUserName() throws Exception {
        SingletonSession.setUserName("changed");
        SingletonSession.getInstance();
        assertTrue(SingletonSession.getUserName().equals("changed"));
    }

    @Test
    public void testSetAdmin() throws Exception {
        SingletonSession.setAdmin(false);
        SingletonSession.getInstance();
        assertFalse(SingletonSession.isAdmin());
    }
}
```

Writable Smart Insert 52 : 6

Spot Bugs Static Code Analysis


Software_Engineering_1 - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Package Explorer
New
Go Into
Open in New Window
Open Type Hierarchy F4
Show In Alt+Shift+W
Copy Ctrl+C
Copy Qualified Name
Paste Ctrl+V
Delete Delete
Remove from Context Ctrl+Alt+Shift+Down
Build Path
Source Alt+Shift+S
Refactor Alt+Shift+T
Import...
Export...
SpotBugs
Refresh F5
Close Project
Assign Working Sets...
Coverage As
Run As
Debug As
Restore from Local History...
Maven
Team
Compare With
Replace With
Configure
Validate
Properties Alt+Enter
Find Bugs
Clear Bug Markers
Save XML
Open Analysis Results in Editor
Load XML
BCC


Software_Engineering_1 - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Package Explorer
> BCC [BNU kevin_service_ 1] (59)
Only 59 Identified Bugs
BCC


Software_Engineering_1 - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Package Explorer
> BCC [BNU kevin_service_2 t1] (59)
src/main/java (59)
BNU.data
BNU.data.database
BNU.data.models
BNU.logic (58)
BNU.logic.service
BNU.logic.service.proxy
BNU.presentation (1)
BNU.singleton
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
JUnit 4
dioxy
resources
src
target
BCC.doxyfile
BCC.log
BCC.log.1
BCC.log.10
BCC.log.11
BCC.log.12
BCC.log.13
BCC.log.14
BCC.log.15
BCC.log.16
BCC.log.17
BCC.log.18
BCC.log.19
BCC.log.2
BCC.log.20
BCC.log.21
BCC.log.22
BCC.log.23
BCC.log.24
BCC.log.25
BCC.log.26
BCC.log.27
BCC.log.28
BCC.log.29
BCC.log.3
BCC.log.30
BCC.log.31
BCC.log.32
BCC.log.33
BCC.log.34
BCC.log.35
BCC.log.36
BCC.log.37
BCC.log.38
BCC

Software_Engineering_1 - BCC/src/main/java/BNU/logic/AdminController.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Bug Explorer | Package Explorer

Write to static field BNU.logic.AdminController.model from instance method new BNU.lo
Write to static field BNU.logic.AdminController.panel from instance method BNU.logic.A
Write to static field BNU.logic.AdminController.panel from instance method new BNU.lo
Write to static field BNU.logic.AdminController.view from instance method BNU.logic.Ad
Write to static field BNU.logic.AdminController.view from instance method new BNU.log
Write to static field BNU.logic.ClassByTeacherController.model from instance method BN
Write to static field BNU.logic.ClassByTeacherController.model from instance method ne
Write to static field BNU.logic.ClassByTeacherController.panel from instance method BNU
Write to static field BNU.logic.ClassByTeacherController.panel from instance method new
Write to static field BNU.logic.ClassByTeacherController.view from instance method BNU
Write to static field BNU.logic.ClassByTeacherController.view from instance method ne
Write to static field BNU.logic.LoginController.ls from instance method new BNU.logic
Write to static field BNU.logic.LoginController.model from instance method BNU.logic
Write to static field BNU.logic.LoginController.model from instance method new BNU.lo
Write to static field BNU.logic.LoginController.panel from instance method BNU.logic.Lo
Write to static field BNU.logic.LoginController.panel from instance method new BNU.log
Write to static field BNU.logic.LoginController.view from instance method BNU.logic.Log
Write to static field BNU.logic.LoginController.view from instance method new BNU.logic
Write to static field BNU.logic.MainController.model from instance method BNU.logic.M
Write to static field BNU.logic.MainController.model from instance method new BNU.log
Write to static field BNU.logic.MainController.panel from instance method BNU.logic.Ma
Write to static field BNU.logic.MainController.panel from instance method new BNU.logi
Write to static field BNU.logic.MainController.view from instance method BNU.logic.Mai
Write to static field BNU.logic.MainController.view from instance method new BNU.logic
Write to static field BNU.logic.MessageBoardController.checkDbForNewMessages from i
Write to static field BNU.logic.MessageBoardController.mbs from instance method new B
Write to static field BNU.logic.MessageBoardController.model from instance method BN
Write to static field BNU.logic.MessageBoardController.model from instance method new
Write to static field BNU.logic.MessageBoardController.panel from instance method BNU
Write to static field BNU.logic.MessageBoardController.panel from instance method new
Write to static field BNU.logic.MessageBoardController.view from instance method BNU.
Write to static field BNU.logic.MessageBoardController.view from instance method new B
Write to static field BNU.logic.RegisterController.rs from instance method new BNU.logic
Write to static field BNU.logic.SetReviewController.panel from instance method new BNU
Write to static field BNU.logic.SetReviewController.srs from instance method new BNU.lo
Write to static field BNU.logic.SetReviewController.view from instance method new BNU.
Write to static field BNU.logic.TeacherReviewController.model from instance method BN
Write to static field BNU.logic.TeacherReviewController.model from instance method new
Write to static field BNU.logic.TeacherReviewController.panel from instance method BNU
Write to static field BNU.logic.TeacherReviewController.panel from instance method new
Write to static field BNU.logic.TeacherReviewController.rs from instance method new BN
Write to static field BNU.logic.TeacherReviewController.view from instance method BNU.
Write to static field BNU.logic.TeacherReviewController.view from instance method new B
Write to static field BNU.logic.TeachersByClassController.model from instance method BI
Write to static field BNU.logic.TeachersByClassController.model from instance method ne
Write to static field BNU.logic.TeachersByClassController.panel from instance method BN
Write to static field BNU.logic.TeachersByClassController.panel from instance method ne
Write to static field BNU.logic.TeachersByClassController.view from instance method BNU
Write to static field BNU.logic.TeachersByClassController.view from instance method new
Write to static field BNU.logic.UserReviewController.model from instance method BNU.lo
Write to static field BNU.logic.UserReviewController.model from instance method new BI

AdminController.java

```
1  package BNU.logic;
2
3  import java.awt.event.ActionEvent;
12
13 public class AdminController extends PageController{
14     static AdminView view;
15     static AdminModel model = new AdminModel();
16     static JPanel panel;
17     public JFrame mainF;
18     static AdminService ac;
19
20     public AdminController(){
21         model = new AdminModel();
22         panel = new JPanel();
23         view = new AdminView();
24     }
25
26     @Override
27     public void dispatchBuilder(JFrame mainFrame) {
28         this.mainF = mainFrame;
29         try {
30             AdminView.BuildAdminView(mainFrame, this);
31         } catch (SecurityException e) {
32             e.printStackTrace();
33         }
34     }
35
36
37
38     public static AdminService getAc() {
39         return ac;
40     }
41
42     public static void setAc(AdminService ac) {
43         AdminController.ac = ac;
44     }
45
46     public static AbstractDB getDb() {
47         return db;
48     }
49
50     public JPanel getPanel() {
51         return panel;
52     }
53
54     public void setPanel(JPanel panel) {
55         AdminController.panel = panel;
56     }
57
58     public AdminView getView() {
59         return view;
60     }
61
62     public void setView(AdminView view) {
63         AdminController.view = view;
64     }
65
66     public AdminModel getModel() {
67         return model;
68     }
69
70     public void setModel(AdminModel model) {
```

Bug Info

AdminController.java: 21

Navigation

Write to static field BNU.logic.AdminController.model from instance method new BNU.logic.AdminController()
Field BNU.logic.AdminController.model

B: Write to static field BN.logic.AdminController.model from instance method new BNU.logic.AdminController()

This instance writes to a static field. This is tricky to get correct if multiple instances are being manipulated, and generally bad practice.

Rank: Of Concern (15), **confidence**: High
Pattern: ST_WRITE_TO_STATIC_FROM_INSTANCE_METHOD
Type: ST, **Category**: STYLE (Dodgy code)

XML output:

```
<BugInstance type="ST_WRITE_TO_STATIC_FROM_INSTANCE_METHOD" priority="1" rank="15" abbrev="ST" categ
  <Class classname="BNU.logic.AdminController">
    <SourceLine classname="BNU.logic.AdminController" sourcefile="AdminController.java" sourcepath="B
  </Class>
  <Method classname="BNU.logic.AdminController" name="<init>" signature="()V" isStatic="false">
    <SourceLine classname="BNU.logic.AdminController" start="20" end="24" startBytecode="0" endBytec
  </Method>
  <Field classname="BNU.logic.AdminController" name="model" signature="LBNU/data/models/AdminModel;"
    <SourceLine classname="BNU.logic.AdminController" sourcefile="AdminController.java" sourcepath="B
  </Field>
  <SourceLine classname="BNU.logic.AdminController" start="21" end="21" startBytecode="11" endBytecod
  <SourceLine classname="BNU.logic.AdminController" start="21" end="21" startBytecode="11" endBytecod
</BugInstance>
```

Write to static field BNU.logic.AdminController.model from instance method new BNU.logic.AdminController() [Of Concern(15), High confidence]

Writable | Smart Insert | 21 : 34

Software_Engineering_1 - BCC/src/main/java/BNU/presentation/MainView.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Bug Explorer | Package Explorer

- BCC (59) [BNU kevin_service_2 ↑1]
  - Scary (1)
    - Normal confidence (1)
      - Method call passes null for non-null parameter (1)
        - Null passed for non-null parameter of new javax.swing.ImageIcon(Image) in BNU.presentati
  - Of Concern (58)

MainView.java

```
75          //controller.getModel().setCb_SearchClass(new JComboB
76          controller.getModel().setCb_SearchClass(new JComboBox
77          controller.getModel().getCb_SearchClass().setActionCo
78          controller.getModel().getCb_SearchClass().addActionLi
79          controller.getModel().getCb_SearchClass().setBounds(4
80          controller.getModel().getWest_Panel().add(controller.
81
82          controller.getPanel().add(controller.getModel().getWe
83
84          //build east panel
85          controller.getModel().setEast_Panel(new JPanel(null))
86          controller.getModel().getEast_Panel().setPreferredSiz
87
88          controller.getModel().setTxt_SearchProfessor(new JLab
89          controller.getModel().getTxt_SearchProfessor().setHor
90          controller.getModel().getTxt_SearchProfessor().setBou
91          controller.getModel().getEast_Panel().add(controller.
92
93          //controller.getModel().setCb_SearchProfessor(new JCo
94          controller.getModel().setCb_SearchProfessor(new JComb
95          controller.getModel().getCb_SearchProfessor().setActi
96          controller.getModel().getCb_SearchProfessor().addActi
97          controller.getModel().getCb_SearchProfessor().setBoun
98          controller.getModel().getEast_Panel().add(controller.
99
100         controller.getPanel().add(controller.getModel().getEa
101
102         //build center panel
103         addImage(mainFrame, controller);
104
105         mainFrame.setContentPane(controller.getPanel());
106         mainFrame.setVisible(true);
107     }
108
109     private static void addImage(JFrame mainFrame, MainContro
110         BufferedImage image = null;
111         try {
112             image = ImageIO.read(new File("resources" + File.
113         } catch (IOException e) {
114             System.out.println("image not loaded");
115             e.printStackTrace();
116         }
117         controller.getModel()
118                 .setTxt_Image(new JLabel(new ImageIcon(getSca
119         controller.getModel().setCenter_Panel(new JPanel());
120         controller.getModel().getCenter_Panel().add(controlle
121         controller.getPanel().add(controller.getModel().getCe
122     }
123
124     private static Image getScaledImage(Image srcImg, int w,
125         BufferedImage resizedImg = new BufferedImage(w, h, Bu
126         Graphics2D g2 = resizedImg.createGraphics();
127
128         g2.setRenderingHint(RenderingHints.KEY_INTERPOLATION,
129         g2.drawImage(srcImg, 0, 0, w, h, null);
130         g2.dispose();
131
132         return resizedImg;
133     }
134 }
135
```

Bug Info

MainView.java: 118

Navigation

Null passed for non-null parameter of new javax.swing.ImageIcon(Image) in BNU.presentation.MainView.addImage(JFrame, MainController)
Method invoked at MainView.java:[line 118]
Called method new javax.swing.ImageIcon(Image)
Argument 1 might be null but must not be null
Value loaded from image
Null value at MainView.java:[line 110]
Known null at MainView.java:[line 113]
Known null at MainView.java:[line 114]
Known null at MainView.java:[line 115]

Bug: Null passed for non-null parameter of new javax.swing.ImageIcon(Image) in BNU.presentation.MainView.addImage(JFrame, MainController)

This method call passes a null value for a non-null method parameter. Either the parameter is annotated as a parameter that should always be non-null, or analysis has shown that it will always be dereferenced.

Rank: Scary (8), **confidence**: Normal
Pattern: NP_NULL_PARAM_DEREF
Type: NP, **Category**: CORRECTNESS (Correctness)

XML output:

```
<BugInstance type="NP_NULL_PARAM_DEREF" priority="2" rank="8" abbrev="NP" category="CORRECTNESS"
  <Class classname="BNU.presentation.MainView">
    <SourceLine classname="BNU.presentation.MainView" sourcefile="MainView.java" sourcepath="BNU/
  </Class>
  <Method classname="BNU.presentation.MainView" name="addImage" signature="(Ljavax/swing/JFrame;L
    <SourceLine classname="BNU.presentation.MainView" start="110" end="122" startBytecode="0" end
  </Method>
  <Method classname="javax.swing.ImageIcon" name="<init>" signature="(Ljava/awt/Image;)V" isStati
    <SourceLine classname="javax.swing.ImageIcon" start="238" end="245" startBytecode="0" endByte
  </Method>
  <Int value="1" role="INT_MAYBE_NULL_ARG"/>
  <LocalVariable name="image" register="2" pc="37" role="LOCAL_VARIABLE_VALUE_OF"/>
  <SourceLine classname="BNU.presentation.MainView" start="118" end="118" startBytecode="71" endB
  <SourceLine classname="BNU.presentation.MainView" start="110" end="110" startBytecode="0" endBy
  <SourceLine classname="BNU.presentation.MainView" start="113" end="113" startBytecode="40" endB
  <SourceLine classname="BNU.presentation.MainView" start="114" end="114" startBytecode="47" endB
  <SourceLine classname="BNU.presentation.MainView" start="115" end="115" startBytecode="51" endB
  <Property name="edu.umd.cs.findbugs.detect.NullDerefProperty.ALWAYS_ON_EXCEPTION_PATH" value="t
</BugInstance>
```

Null passed for non-null parameter of new javax.swing.ImageIcon(Image) in BNU.presentation.MainView.addImage(JFrame, MainController) [Scary(8), Normal confidence]

Writable | Smart Insert | 118 : 117

7 Design Patterns

Software_Engineering_1 - BCC/src/main/java/BNU/singleton/SingletonSession.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Package Explorer

- BCC [BNU kevin_service_2 ↑1] (59)
  - src/main/java (59)
    - BNU.data
    - BNU.data.database
    - BNU.data.models
    - BNU.logic (58)
    - BNU.logic.service
    - BNU.logic.service.proxy
    - BNU.presentation (1)
    - BNU.singleton
      - SingletonSession.java
  - src/test/java
  - JRE System Library [JavaSE-1.8]
  - Maven Dependencies
  - JUnit 4
  - dioxy
  - resources
  - src
  - target
  - BCC.doxyfile
  - BCC.log
  - database.log
  - log.log
  - pom.xml

Console | SingletonSession.java

```java
package BNU.singleton;

import BNU.logic.service.proxy.SmartProxy;

/**
 *
 * @author Kevin
 *
 * Singleton GoF Design Pattern
 * Creational
 *
 * This class follows the singleton design pattern because only one instance
 * of it will exist. The existence of only one object is controlled by the
 * private constructor. This program uses this singleton as a session
 * monitor to track which user is logged into the application and whether that
 * user is an admin. This is useful because the program will need to know
 * in certain instances who the user is and whether they are an admin.
 */

public class SingletonSession {
    static String userName;
    static boolean isAdmin;

    private static final SingletonSession SINGLE_INSTANCE = new SingletonSession();

    private SingletonSession() {
    }

    private SingletonSession(String userName, boolean isAdmin) {
          SingletonSession.setUserName(userName);
          SingletonSession.setAdmin(isAdmin);
    }

  public static SingletonSession getInstance() {
      return SINGLE_INSTANCE;
    }

  public static SingletonSession getInstance(String userName, boolean isAdmin) {
      SingletonSession.setUserName(userName);
      SingletonSession.setAdmin(isAdmin);
      return SINGLE_INSTANCE;
    }

public static String getUserName() {
    return userName;
}

public static void setUserName(String userName) {
    SmartProxy sp = new SmartProxy();
    String cleanUN = sp.sanatizeInput(userName);
    SingletonSession.userName = cleanUN;
}

public static boolean isAdmin() {
    return isAdmin;
}

public static void setAdmin(boolean isAdmin) {
    SingletonSession.isAdmin = isAdmin;
}
```

Writable | Smart Insert | 50 : 49

Software_Engineering_1 - BCC/src/main/java/BNU/logic/service/proxy/SmartProxy.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Package Explorer

- BCC [BNU kevin_service_2 ↑8] (59)
  - src/main/java (59)
    - BNU.data
    - BNU.data.database
    - BNU.data.models
    - BNU.logic (58)
    - BNU.logic.service
    - BNU.logic.service.proxy
      - SmartProxy.java
    - BNU.presentation (1)
    - BNU.singleton
      - SingletonSession.java
  - src/test/java
  - JRE System Library [JavaSE-1.8]
  - Maven Dependencies
  - JUnit 4
  - dioxy
  - resources
  - src
  - target
  - BCC.doxyfile
  - BCC.log
  - BCC.log.1
  - BCC.log.2
  - BCC.log.3
  - BCC.log.4
  - BCC.log.5
  - BCC.log.6
  - BCC.log.7
  - BCC.log.8
  - database.log
  - log.log
  - log.log.1
  - log.log.2
  - log.log.3
  - pom.xml

SmartProxy.java

```java
package BNU.logic.service.proxy;

import java.math.BigInteger;

/**
 *
 * @author Kevin
 *
 * PROXY GoF Design Pattern
 * Structural
 *
 * This class follows the proxy design pattern. This class works as
 * a proxy because it is used in place of another class to add
 * functionality. The controllers think they are talking directly with
 * the services but they are talking with this proxy in some instances
 * instead. This is a smart proxy because it is designed to add a layer
 * of security. This proxy is intended to sanatize input from the user
 * to prevent SQL Injection attacks.
 */

import BNU.logic.LoginController;
import BNU.logic.MessageBoardController;
import BNU.logic.ReviewController;

public class SmartProxy {

    public String sanatizeInput(String input) {
        String clean = input.replaceAll("'", "");
        clean = clean.replaceAll("\\\\", "");
        clean = clean.replaceAll(";", "");
        clean = clean.replaceAll("`", "");
        return clean;
    }

    public boolean sanatizeAndCheckCreds(String userName, String password) {
        boolean isValidated = false;
        String cleanUN = sanatizeInput(userName);
        String cleanPW = sanatizeInput(password);
        isValidated = LoginController.db.validateUser(cleanUN, cleanPW);
        System.out.println(cleanUN + " " + cleanPW);
        return isValidated;
    }

    public boolean sanatizeAndCheckIsAdmin(String userName) {
        boolean isAdmin = false;
        String cleanUN = sanatizeInput(userName);
        //confirm how I should be getting the db
        if(LoginController.db.isAdmin(cleanUN)) {
            isAdmin = true;
        }
        return isAdmin;
    }

    public void sanatizeAndSendMessage(String message, String from, String to, BigInteger date) {
        message = sanatizeInput(message);
        MessageBoardController.db.sendMessage(message, from, to, date);
    }

    public void sanatizeAndCreateReview(String userName,String professorName,String className,
            String content,String TA, String H, String WL) {
        String cleanContent = sanatizeInput(content);
```

Writable | Smart Insert | 14 : 71

Software_Engineering_1 - BCC/src/main/java/BNU/logic/MessageBoardController.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Package Explorer

- BCC [BNU kevin_service_2 ↑8] (59)
  - src/main/java (59)
    - BNU.data
    - BNU.data.database
    - BNU.data.models
    - BNU.logic (58)
      - AdminController.java (6)
      - ClassByTeacherController.java (6)
      - CustomComparator.java
      - LoginController.java (7)
      - Main.java (1)
      - MainController.java (6)
      - MessageBoardController.java (8)
      - PageController.java
      - RegisterController.java (1)
      - ReviewController.java
      - SetReviewController.java (3)
      - TeacherReviewController.java (7)
      - TeachersByClassController.java (6)
      - UserReviewController.java (6)
      - WindowBuilder.java (1)
    - BNU.logic.service
      - AdminService.java
      - ClassByTeacherService.java
      - LoginService.java
      - MainService.java
      - MessageBoardService.java
      - RegisterService.java
      - SetReviewService.java
      - TeacherByClassService.java
      - TeacherReviewViewService.java
      - UserReviewService.java
    - BNU.logic.service.proxy
    - BNU.presentation (1)
    - BNU.singleton
  - src/test/java
  - JRE System Library [JavaSE-1.8]
  - Maven Dependencies
  - JUnit 4
  - dioxy
  - resources
  - src
  - target
  - BCC.doxyfile
  - BCC.log
  - BCC.log.1
  - BCC.log.2
  - BCC.log.3
  - BCC.log.4
  - BCC.log.5
  - BCC.log.6

MessageBoardService.java | MessageBoardView.java | MessageBoardController.java

```java
package BNU.logic;

import java.awt.event.ActionEvent;

/**
 *
 * @author Kevin
 *
 * LAZY LOADING GoF Design Pattern
 *
 * This controller reveals how the message board follows the lazy
 * loading design pattern. Specifically, it follows the lazy
 * initialization implementation because the program waits to load
 * messages between the user and another user until the user
 * selects who is the other party in the conversation. In contrast,
 * the program could load all the data at once and store it in case
 * the user wants to look at all threads of messages. By lazily
 * initializing the message board model, however, we improve memory use
 * and processing speed.
 */
public class MessageBoardController extends PageController {
    static MessageBoardView view;
    static MessageBoardModel model = new MessageBoardModel();
    static JPanel panel;
    public JFrame mainF;
    public static MessageBoardService mbs;
    public static Thread checkDbForNewMessages;
    private static final Logger LOGGER = Logger.getLogger(MessageBoardController.class.getName());

    static {

        try {
            LOGGER.addHandler(new FileHandler("log.log"));
            LOGGER.setLevel(Level.FINEST);
        } catch (SecurityException | IOException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }

    // static private InputMessage im;

    public MessageBoardController() {
        model = new MessageBoardModel();
        panel = new JPanel();
        view = new MessageBoardView();
        mbs = new MessageBoardService();
    }

    @Override
    public void dispatchBuilder(JFrame mainFrame) {
        this.mainF = mainFrame;
        try {
            checkDbForNewMessages = new Thread(new Runnable() {
                public void run() {
                    try {
                        mbs.observer(MessageBoardController.this, SingletonSession.getInstance().getUserName());
                    } catch (InterruptedException e) {
                        LOGGER.info("Thread Stopped");
                    }
                }
            });
```

Writable | Smart Insert | 16 : 1

Software_Engineering_1 - BCC/src/main/java/BNU/data/database/dbWrapper.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Package Explorer

- BCC [BNU kevin_service_2 ↑8] (59)
  - src/main/java (59)
    - BNU.data
    - BNU.data.database
      - AbstractDB.java
      - DatabaseApi.java
      - DatabaseConnectionException.java
      - DatabaseMock.java
      - DatabaseOperationException.java
      - dbWrapper.java
    - BNU.data.models
    - BNU.logic (58)
    - BNU.logic.service
    - BNU.logic.service.proxy
      - SmartProxy.java
    - BNU.presentation (1)
    - BNU.singleton
      - SingletonSession.java
  - src/test/java
  - JRE System Library [JavaSE-1.8]
  - Maven Dependencies
  - JUnit 4
  - dioxy
  - resources
  - src
  - target
  - BCC.doxyfile
  - BCC.log
  - BCC.log.1
  - BCC.log.2
  - BCC.log.3
  - BCC.log.4
  - BCC.log.5
  - BCC.log.6
  - BCC.log.7
  - BCC.log.8
  - database.log
  - log.log
  - log.log.1
  - log.log.2
  - log.log.3
  - pom.xml

SmartProxy.java | dbWrapper.java

```java
package BNU.data.database;

import java.util.ArrayList;

/**
 *
 * @author Kevin
 *
 * FACADE GoF Design Pattern
 * Structural
 *
 * This class follows the facade design pattern. This class works as
 * a facad because it hides the complexity of the database
 * implementation from the program. Using this facade, the program
 * does not need to know all the detail regarding CRUD operation
 * involving the database. The program simply needs to know what
 * these methods return and trust the methods to do the secret
 * backend work.
 */

public interface dbWrapper {
    boolean validateUser(String userName, String password); //new

    String[] getAllProfessors();

    String[] getAllClasses();

    String[] getAllProfessorsForClass(String className);

    Professor getProfessor(String Prof);

    String[] getAllClassesForProfessor(String professorName);

    Course getCourse(String course);


    String[][] getAllCoursesByProf(String[] courses); //new

    String[][] getAllTeachersByCourse(String[] professorNames); //new

    boolean submitCredentials(String userName, String password); //new

    //return array of review content,score,reviewerID
    String[][] getAllReviewsForTeacherClass(String professorName,String className);

    // return in this order: overall score, helpfulness, teaching ability, workload
    String[] getOverallProfessorRatings(String professorName);

    // return review content, score, reviewerID, course name
    String[][] getAllReviewsForUser(String userName);

    // my added functions
    String[][] getAllMessages(String sender, String receiver);

    String[] getAllFlagged();

    String getReceiver();

    String getSender();

    String[] getAllUserMessagers(String receiver);
```

Writable | Smart Insert | 32 : 5

Software_Engineering_1 - BCC/src/main/java/BNU/logic/PageController.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Package Explorer

- BCC [BNU kevin_service_2 ↑8] (59)
  - src/main/java (59)
    - BNU.data
    - BNU.data.database
    - BNU.data.models
    - BNU.logic (58)
      - AdminController.java (6)
      - ClassByTeacherController.java (6)
      - CustomComparator.java
      - LoginController.java (7)
      - Main.java (1)
      - MainController.java (6)
      - MessageBoardController.java (8)
      - PageController.java
      - RegisterController.java (1)
      - ReviewController.java
      - SetReviewController.java (3)
      - TeacherReviewController.java (7)
      - TeachersByClassController.java (6)
      - UserReviewController.java (6)
      - WindowBuilder.java (1)
    - BNU.logic.service
      - AdminService.java
      - ClassByTeacherService.java
      - LoginService.java
      - MainService.java
      - MessageBoardService.java
      - RegisterService.java
      - SetReviewService.java
      - TeacherByClassService.java
      - TeacherReviewViewService.java
      - UserReviewService.java
    - BNU.logic.service.proxy
    - BNU.presentation (1)
    - BNU.singleton
  - src/test/java
  - JRE System Library [JavaSE-1.8]
  - Maven Dependencies
  - JUnit 4
  - dioxy
  - resources
  - src
  - target
  - BCC.doxyfile
  - BCC.log
  - BCC.log.1
  - BCC.log.2
  - BCC.log.3
  - BCC.log.4
  - BCC.log.5
  - BCC.log.6

PageController.java

```java
package BNU.logic;

import java.awt.event.ActionListener;

/**
 *
 * @author Kevin
 *
 * BRIDGE GoF Design Pattern
 * Structural
 *
 * This class follows the bridge design pattern because it separates
 * the abstraction from the implementation of the abstract methods.
 * This class implements the ActionListener interface and provides the
 * abstraction for classes with refined abstraction and implementers to
 * use. With this abstraction, implementors will know how specific
 * concrete implementors must be structured.
 */

public abstract class PageController implements ActionListener{
    public static AbstractDB db;
    public abstract void dispatchBuilder(JFrame mainFrame);

}
```

BNU.logic.PageController.java - BCC/src/main/java

Software_Engineering_1 - BCC/src/main/java/BNU/logic/service/MessageBoardService.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

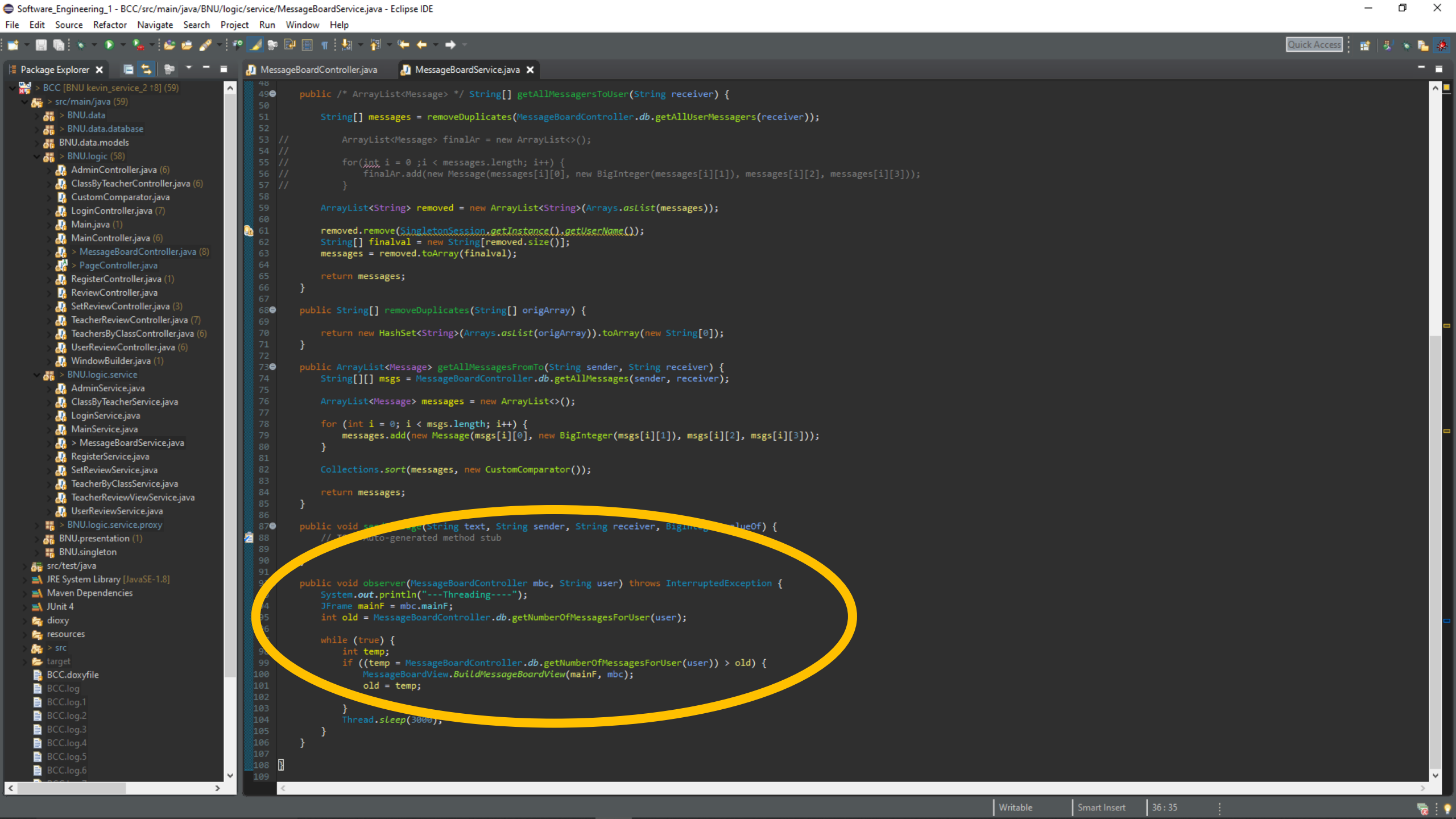
Package Explorer

- BCC [BNU kevin_service_2 ↑8] (59)
  - src/main/java (59)
    - BNU.data
    - BNU.data.database
    - BNU.data.models
    - BNU.logic (58)
      - AdminController.java (6)
      - ClassByTeacherController.java (6)
      - CustomComparator.java
      - LoginController.java (7)
      - Main.java (1)
      - MainController.java (6)
      - MessageBoardController.java (8)
      - PageController.java
      - RegisterController.java (1)
      - ReviewController.java
      - SetReviewController.java (3)
      - TeacherReviewController.java (7)
      - TeachersByClassController.java (6)
      - UserReviewController.java (6)
      - WindowBuilder.java (1)
    - BNU.logic.service
      - AdminService.java
      - ClassByTeacherService.java
      - LoginService.java
      - MainService.java
      - MessageBoardService.java
      - RegisterService.java
      - SetReviewService.java
      - TeacherByClassService.java
      - TeacherReviewViewService.java
      - UserReviewService.java
    - BNU.logic.service.proxy
    - BNU.presentation (1)
    - BNU.singleton
  - src/test/java
  - JRE System Library [JavaSE-1.8]
  - Maven Dependencies
  - JUnit 4
  - dioxy
  - resources
  - src
  - target
  - BCC.doxyfile
  - BCC.log
  - BCC.log.1
  - BCC.log.2
  - BCC.log.3
  - BCC.log.4
  - BCC.log.5
  - BCC.log.6

MessageBoardController.java | MessageBoardService.java

```java
package BNU.logic.service;

import java.math.BigInteger;

/**
 *
 * @author Kevin
 *
 * OBSERVER GoF Design Pattern
 * Behavioral
 *
 * This class follows the observer design pattern because it
 * establishes a one-to-many relationship where the program
 * observes the message board to notify a user if they
 * receive a message. When a user is on the message board
 * viewing a message thread, this observer will constantly
 * check to see if the message thread must be updated. The subject
 * is the other user sending the message and the observer
 * essentially monitors this user because it is dependent
 * on their behavior. The purpose of this observer is to
 * create real-time messaging between two users.
 */
public class MessageBoardService {

    public void messageSend(String message, String from, String to) {
        SmartProxy sp = new SmartProxy();
        BigInteger i = BigInteger.valueOf(System.currentTimeMillis());
        sp.sanatizeAndSendMessage(message, from, to, i);
        // MessageBoardController.db.sendMessage(message);
    }

    public String getReceiver() {
        return SingletonSession.getInstance().getUserName();
    }

    public /* ArrayList<Message> */ String[] getAllMessagersToUser(String receiver) {

        String[] messages = removeDuplicates(MessageBoardController.db.getAllUserMessagers(receiver));

//          ArrayList<Message> finalAr = new ArrayList<>();
//
//          for(int i = 0 ;i < messages.length; i++) {
//              finalAr.add(new Message(messages[i][0], new BigInteger(messages[i][1]), messages[i][2], messages[i][3]));
//          }

        ArrayList<String> removed = new ArrayList<String>(Arrays.asList(messages));

        removed.remove(SingletonSession.getInstance().getUserName());
        String[] finalval = new String[removed.size()];
        messages = removed.toArray(finalval);

        return messages;
    }

    public String[] removeDuplicates(String[] origArray) {

        return new HashSet<String>(Arrays.asList(origArray)).toArray(new String[0]);
    }

    public ArrayList<Message> getAllMessagesFromTo(String sender, String receiver) {
        String[][] msgs = MessageBoardController.db.getAllMessages(sender, receiver);
```

Writable | Smart Insert | 36 : 35


Software_Engineering_1 - BCC/src/main/java/BNU/logic/service/MessageBoardService.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
MessageBoardController.java
MessageBoardService.java
BCC [BNU kevin_service_2 ↑8] (59)
src/main/java (59)
BNU.data
BNU.data.database
BNU.data.models
BNU.logic (58)
AdminController.java (6)
ClassByTeacherController.java (6)
CustomComparator.java
LoginController.java (7)
Main.java (1)
MainController.java (6)
MessageBoardController.java (8)
PageController.java
RegisterController.java (1)
ReviewController.java
SetReviewController.java (3)
TeacherReviewController.java (7)
TeachersByClassController.java (6)
UserReviewController.java (6)
WindowBuilder.java (1)
BNU.logic.service
AdminService.java
ClassByTeacherService.java
LoginService.java
MainService.java
MessageBoardService.java
RegisterService.java
SetReviewService.java
TeacherByClassService.java
TeacherReviewViewService.java
UserReviewService.java
BNU.logic.service.proxy
BNU.presentation (1)
BNU.singleton
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
JUnit 4
dioxy
resources
src
target
BCC.doxyfile
BCC.log
BCC.log.1
BCC.log.2
BCC.log.3
BCC.log.4
BCC.log.5
BCC.log.6
public /* ArrayList<Message> */ String[] getAllMessagersToUser(String receiver) {
String[] messages = removeDuplicates(MessageBoardController.db.getAllUserMessagers(receiver));
// ArrayList<Message> finalAr = new ArrayList<>();
// for(int i = 0 ;i < messages.length; i++) {
// finalAr.add(new Message(messages[i][0], new BigInteger(messages[i][1]), messages[i][2], messages[i][3]));
// }
ArrayList<String> removed = new ArrayList<String>(Arrays.asList(messages));
removed.remove(SingletonSession.getInstance().getUserName());
String[] finalval = new String[removed.size()];
messages = removed.toArray(finalval);
return messages;
}
public String[] removeDuplicates(String[] origArray) {
return new HashSet<String>(Arrays.asList(origArray)).toArray(new String[0]);
}
public ArrayList<Message> getAllMessagesFromTo(String sender, String receiver) {
String[][] msgs = MessageBoardController.db.getAllMessages(sender, receiver);
ArrayList<Message> messages = new ArrayList<>();
for (int i = 0; i < msgs.length; i++) {
messages.add(new Message(msgs[i][0], new BigInteger(msgs[i][1]), msgs[i][2], msgs[i][3]));
}
Collections.sort(messages, new CustomComparator());
return messages;
}
public void
// Auto-generated method stub
public void observer(MessageBoardController mbc, String user) throws InterruptedException {
System.out.println("---Threading----");
JFrame mainF = mbc.mainF;
int old = MessageBoardController.db.getNumberOfMessagesForUser(user);
while (true) {
int temp;
if ((temp = MessageBoardController.db.getNumberOfMessagesForUser(user)) > old) {
MessageBoardView.BuildMessageBoardView(mainF, mbc);
old = temp;
}
Thread.sleep(3000);
}
}
Writable
Smart Insert
36 : 35

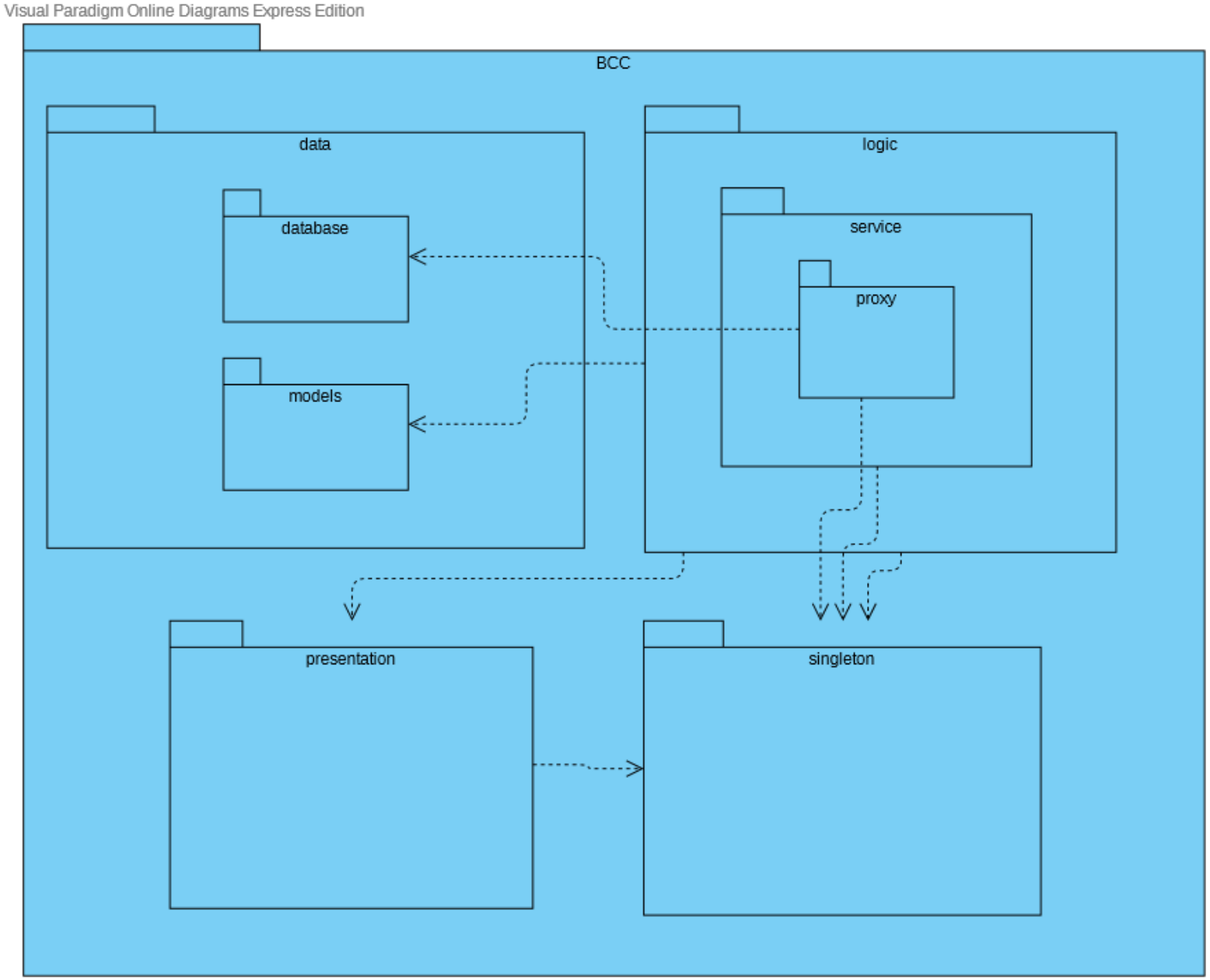
Template Method

```
public abstract class AbstractDB {

    // Design Pattern held here - Template Method


    // All the methods with final in this class, that are then calling
    // methods within DatabaseApi to do the implementation, are Template Methods.
    // For example public final void deleteUserAccount() is a template method
    // as it calls the deleteUserAccountImpl() (in DatabaseApi, which is this class's child class),
    // to do the actual implementation of the function.
```

```java
public final void deleteUserAccount(String user) {
    try {
        con = getRemoteConnection();

        deleteUserAccountImpl(user);

        if (con != null) {

            con.close();

        }

    } catch (DatabaseConnectionException | SQLException | DatabaseOperationException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
```

```
protected void deleteUserAccountImpl(String userId) throws DatabaseOperationException {
    String query = "update users set user_name = \'deleted" + Math.random() * 1000 + "\' where user_name = \'" + userId + "\'";

    try (PreparedStatement stmt = con.prepareStatement(query)) {
        int rs = stmt.executeUpdate();

        if (rs != 1) {
            con.rollback();
            throw new DatabaseOperationException(query);
        }

    } catch (SQLException e) {
        LOGGER.warning(e.getMessage());
        throw new DatabaseOperationException(query);
    }
}
```

Package Diagram



Javadoc

C:\Users\Kevin\Documents\Software_Engineering_1\BNU\BNU\docs\Javadoc

File Home Share View

Pin to Quick access | Copy | Paste | Cut | Copy path | Paste shortcut — Clipboard
Move to | Copy to | Delete | Rename — Organize
New folder | New item | Easy access — New
Properties | Open | Edit | History — Open
Select all | Select none | Invert selection — Select

This PC › Documents › Software_Engineering_1 › BNU › BNU › docs › Javadoc ›

Search Javadoc

- Quick access
  - Documents
  - Downloads
  - Pictures
  - iCloud Photos
  - cmake-build-debug
  - Design_Patterns_Hi
  - docs
  - Screenshots
- Creative Cloud Files
- OneDrive
- This PC
  - 3D Objects
  - Desktop
  - Documents
  - Downloads
  - Music
  - Pictures
  - Videos
  - Windows (C:)
- Network

Name	Date modified	Type	Size
BNU	12/5/2019 5:08 AM	File folder	
index-files	12/5/2019 5:08 AM	File folder	
jquery	12/5/2019 5:08 AM	File folder	
resources	12/5/2019 5:08 AM	File folder	
allclasses-index.html	12/5/2019 5:08 AM	HTML File	22 KB
allpackages-index.html	12/5/2019 5:08 AM	HTML File	6 KB
constant-values.html	12/5/2019 5:08 AM	HTML File	4 KB
deprecated-list.html	12/5/2019 5:08 AM	HTML File	4 KB
element-list	12/5/2019 5:08 AM	File	1 KB
help-doc.html	12/5/2019 5:08 AM	HTML File	10 KB
index.html	12/5/2019 5:08 AM	HTML File	6 KB
member-search-index.js	12/5/2019 5:08 AM	JavaScript File	77 KB
member-search-index.zip	12/5/2019 5:08 AM	Compressed (zipp...	8 KB
overview-summary.html	12/5/2019 5:08 AM	HTML File	1 KB
overview-tree.html	12/5/2019 5:08 AM	HTML File	19 KB
package-search-index.js	12/5/2019 5:08 AM	JavaScript File	1 KB
package-search-index.zip	12/5/2019 5:08 AM	Compressed (zipp...	1 KB
script.js	12/5/2019 5:08 AM	JavaScript File	6 KB
search.js	12/5/2019 5:08 AM	JavaScript File	13 KB
serialized-form.html	12/5/2019 5:08 AM	HTML File	7 KB
stylesheet.css	12/5/2019 5:08 AM	Cascading Style S...	22 KB
type-search-index.js	12/5/2019 5:08 AM	JavaScript File	4 KB
type-search-index.zip	12/5/2019 5:08 AM	Compressed (zipp...	1 KB

23 items

Uses of Class BNU.singleton.SingletonSession

Packages that use SingletonSession

Package	Description
BNU.singleton	

Uses of SingletonSession in BNU.singleton

Methods in BNU.singleton that return SingletonSession

Modifier and Type	Method	Description
static SingletonSession	SingletonSession.getInstance()	
static SingletonSession	SingletonSession.getInstance(java.lang.String userName, boolean isAdmin)	

Git Analysis

Analysis

	Additions				Commits
	GitHub Insights	Git Blame	Lines of code merged to master	Merges to Master	GitHub Insights
Mark Fuller	11,156	3,592	9,241	31	95
Kevin Kulda	6264	3388	3,009	31	82
Connor Woodahl	1,152	959	3,631	23	23
Total Lines of Java Code in final Project	9190				

Mark Fuller: 58 hours

Kevin Kulda: 52 hours

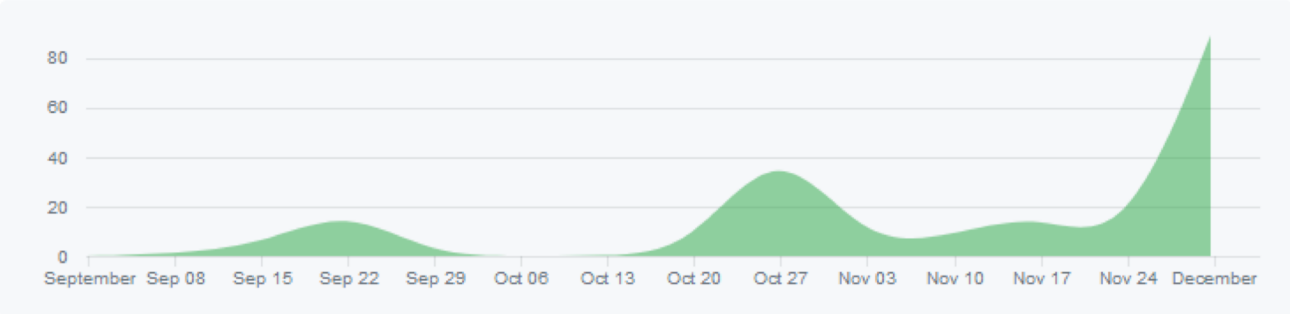
Connor Woodahl: 55 hours

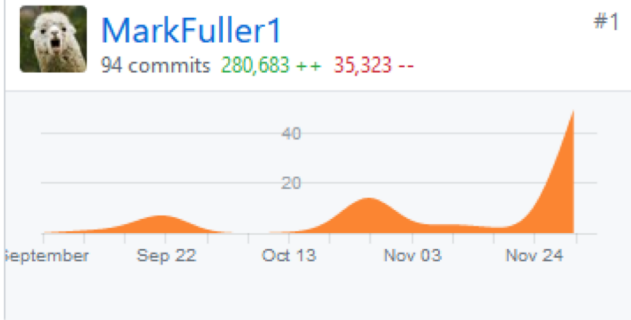
Pulse
Contributors
Community
Traffic
Commits
Code frequency
Dependency graph
Network
Forks

Sep 1, 2019 – Dec 5, 2019

Contributions: Commits

Contributions to master, excluding merge commits





MarkFuller1 #1

94 commits 280,683 ++ 35,323 --

Kevin-Joseph #2

81 commits 108,439 ++ 8,991 --

Connor-woodahl #3

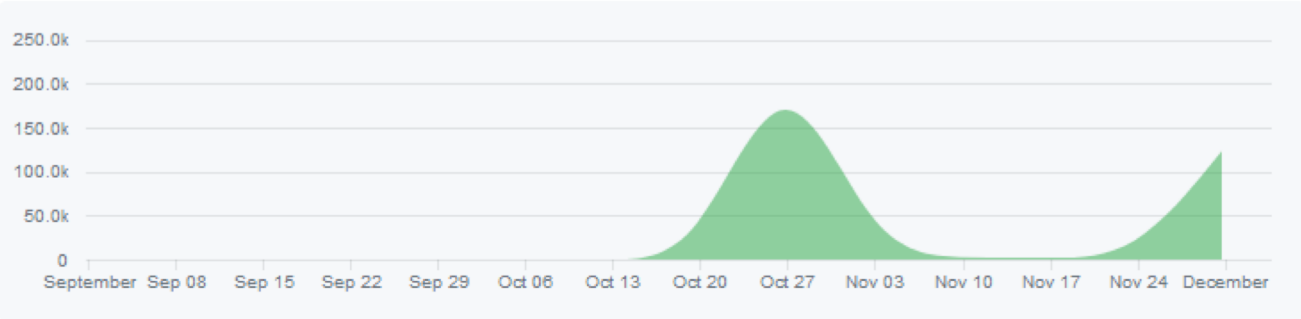
23 commits 1,478 ++ 1,044 --

Pulse
Contributors
Community
Traffic
Commits
Code frequency
Dependency graph
Network
Forks

Sep 1, 2019 – Dec 5, 2019

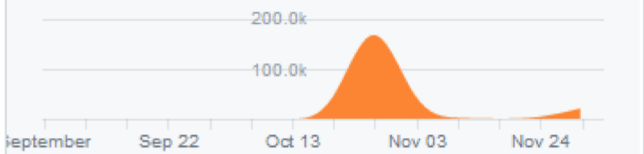
Contributions: Additions

Contributions to master, excluding merge commits



MarkFuller1 #1

94 commits 280,683 ++ 35,323 --

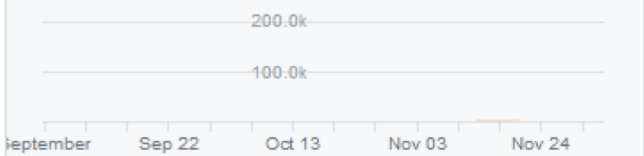


Kevin-Joseph #2

81 commits 108,439 ++ 8,991 --

Connor-woodahl #3

23 commits 1,478 ++ 1,044 --



Pulse
Contributors
Community
Traffic
Commits
Code frequency
Dependency graph
Network
Forks

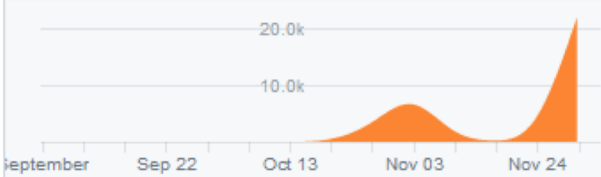
Sep 1, 2019 – Dec 5, 2019

Contributions: Deletions

Contributions to master, excluding merge commits

MarkFuller1 #1

94 commits 280,683 ++ 35,323 --



Kevin-Joseph #2

81 commits 108,439 ++ 8,991 --

Connor-woodahl #3

23 commits 1,478 ++ 1,044 --

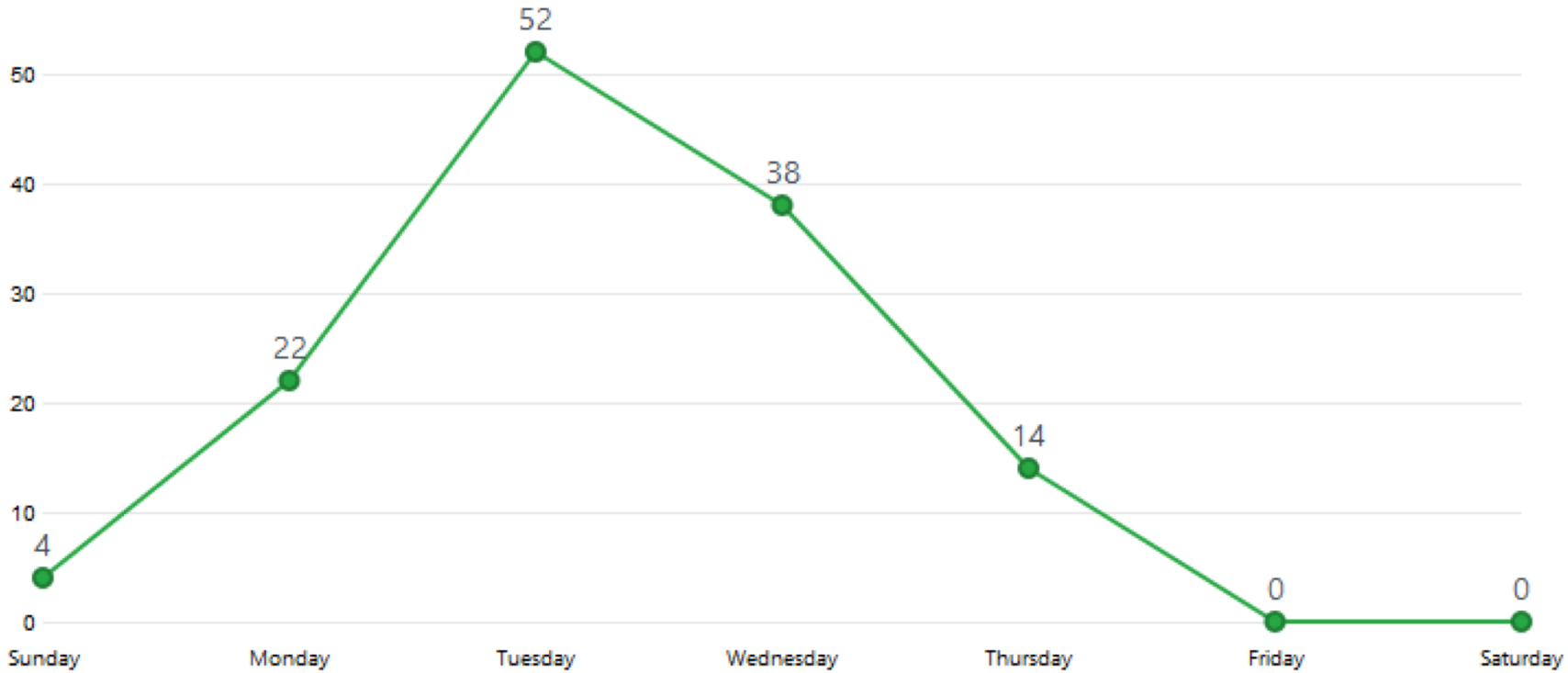
Referring sites

Site	Views	Unique visitors
github.com	42	3

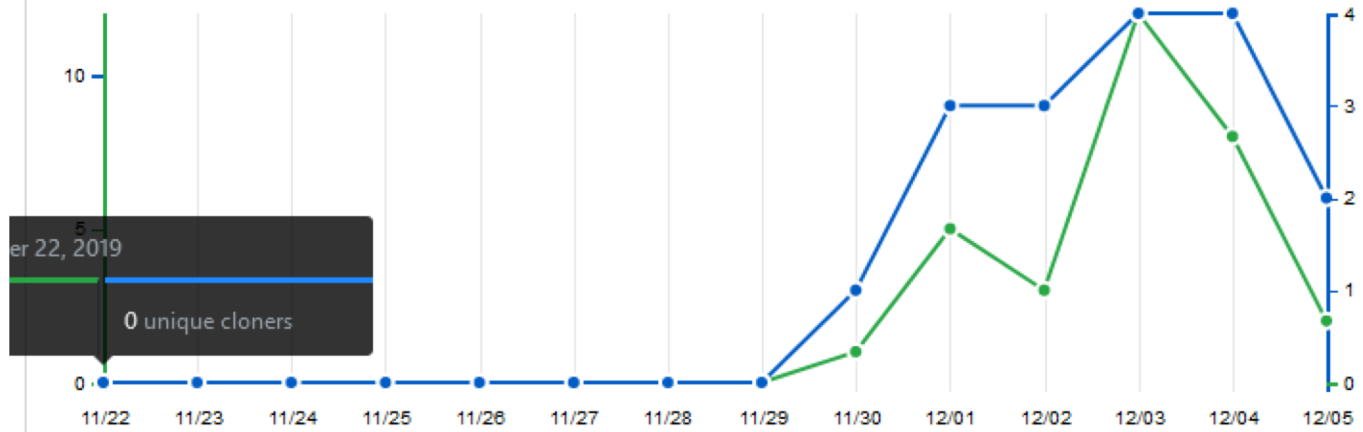
Popular content

Content	Views	Unique visitors
Pull Requests	128	3
MarkFuller1/BNU: Baylor Cla...	92	3
Flag Button by Connor-woo...	18	3
Pulse	15	3
Contributors to MarkFuller1/...	14	2
Issues	14	2
addid by MarkFuller1 · Pull R...	12	3
MarkFuller1/BNU at kevin_se...	10	1
Commits	9	2
update db by MarkFuller1 · P...	9	2

- Pulse
- Contributors
- Community
- Traffic
- **Commits**
- Code frequency
- Dependency graph
- Network
- Forks

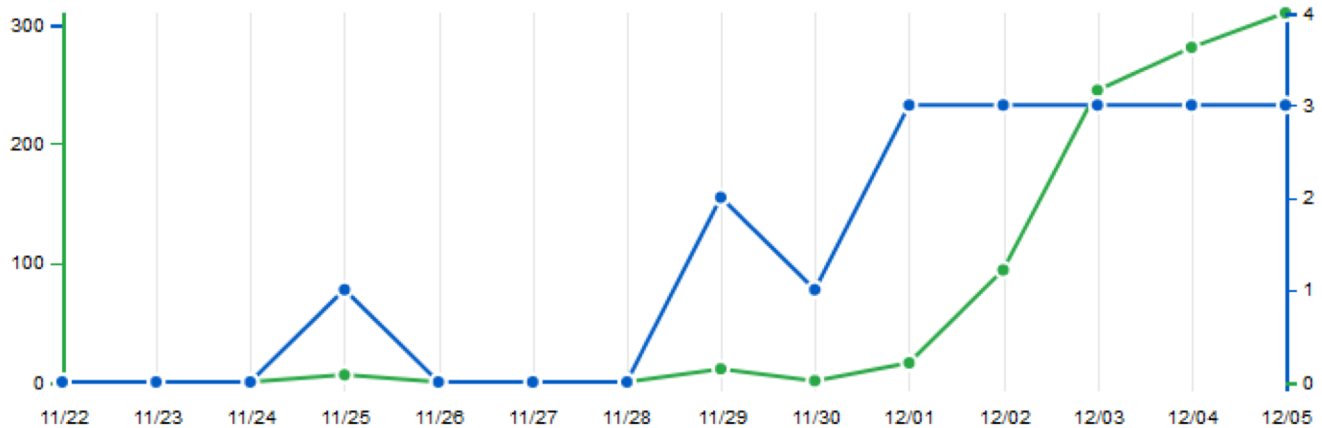


Git clones



31	14
Clones	Unique cloners

Visitors



964	3
Views	Unique visitors

Time Tracking

- Mark Fuller: 58 hours
- Kevin Kulda: 52 hours
- Connor Woodahl: 55 hours

Point Distribution

- Mark Fuller: 33.4%
- Kevin Kulda: 33.3%
- Connor Woodahl: 33.3%