*Florida International University*

*School of Computing and Information Sciences*

Software Engineering Focus

Feature Document

User Story #1332

**Name:** Leon Liang

**Team Member(s):** Leon Liang, Dale Keith, Andres Moser, Steve Hirabayashi, Adam Levy

**Project:** V.I.P. Website 7.0

**Product Owner(s)**: Francisco Ortega, Masoud Sadjadi

**Mentor(s)**: Jose Ponce

**Instructor**: Masoud Sadjadi

**User Story Name: Search Functionality For Inbox and Student Review**

As an user, I want to be able to search based on keywords for student project reviews, general messages, user registration review, and student application review in order to find individual requests

Acceptance Criteria

* User must have a search box on the page section to search for individual keyword
* Inbox should return all messages if the search box is empty

**Use Cases**

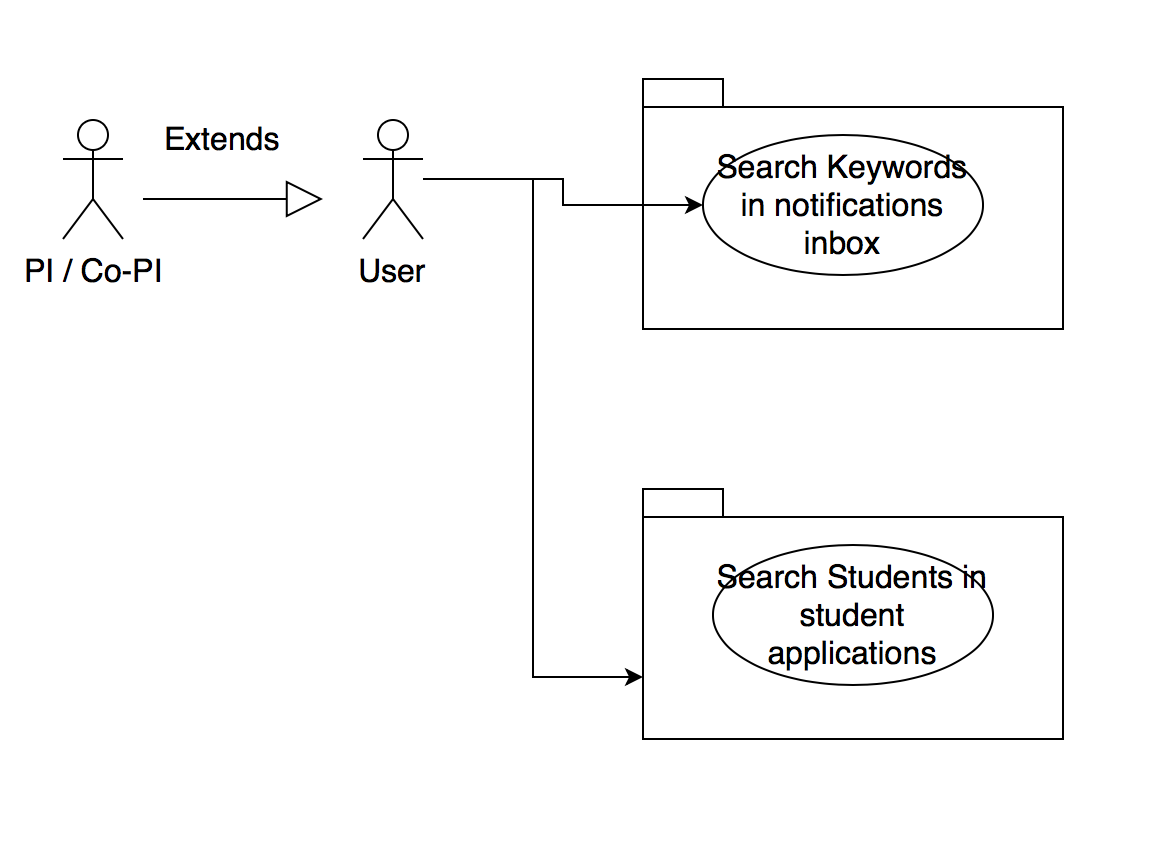
* Name: User searches for messages in inbox
* Actor: User (Pi/CoPi/Faculty/Student)
* Preconditions:
* User must be logged in
* User must be at the notifications page
* Description:

1. User types in keyword, whether it be name or project in the search box
2. All notifications will be filtered based on algorithm to find the keywords in all the texts
3. User deletes search phrase in the keyword
4. All results will be returned

* Name: User searches student in the student applications page
* Actor: Pi/CoPi
* Preconditions:
* Admin must be logged in with administrative rights
* User must be in the reviewStudentApplications page
* Description:

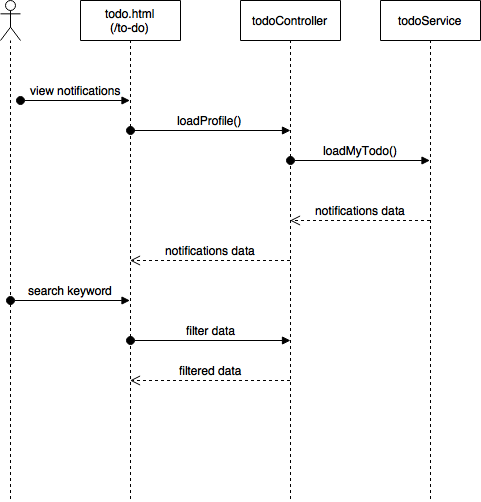
1. User types in keyword, whether it be name or project in the search box
2. All notifications will be filtered based on algorithm to find the keywords in all the texts
3. User deletes search phrase in the keyword
4. All results will be returned

**Use Case Diagram**

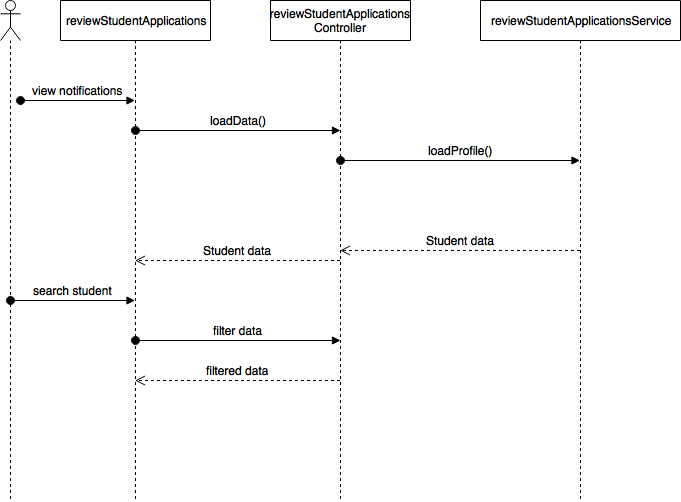


**Sequence Diagrams**

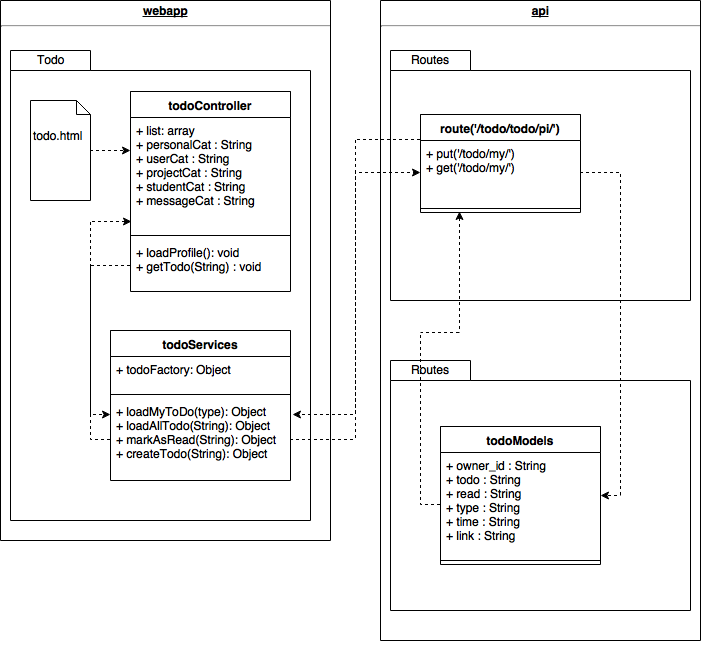
Search notifications by keywords



Search student applications by keywords



**Class Diagram**

****

**Unit Tests**

* Test case ID: VIP-1332-U1
* Description: User searches for notification based on keyword
* Summary of Test:

1. Tester modifies search input

* Test Data:
* Keyword: String
* Pre-Condition:

1. User has logged into the VIP website
2. User is at the todo notifications page

* Expected Results:

1. Notification inbox will be repopulated
2. Notification will contain the keyword as substring in the text

* Actual Result:

1. Notification inbox will be repopulated
2. Notification will contain the keyword as substring in the text

* Status (Fail/Pass): PASS
* Test case ID: VIP-1332-U2
* Description: User searches for student registration based on keyword
* Summary of Test:

1. Tester modifies search input

* Test Data:
* Keyword: string
* Pre-Condition:

1. User has logged into the VIP website
2. User is at the todo notifications page

* Expected Results:

1. Student registrations will be repopulated
2. Entries will only show if the keyword is contained in one of the input texts

* Actual Result:

1. Student registrations will be repopulated
2. Entries will only show if the keyword is contained in one of the input texts

* Status (Fail/Pass): PASS

**Integration Test**

* Test case ID: VIP-1332-G1
* Description: Updating user in admin user panel should update Projects page with the user as a member
* Summary of Test:

1. Tester selects ‘edit user’ button
2. Tester modifies user with test data in edit user dialog
3. Tester selects ‘save changes’ button
4. Tester selects ‘projects’ button in navigation bar
5. Tester selects the project the user is a current member of

* Test Data:
* First Name: FirstNameTest
* Last Name: LastNameTest
* Pre-condition:

1. Admin with Pi/CoPi privileges has logged into the VIP website
2. Admin is at the admin panel page - user maintenance tab
3. User must be a current member of an active project

* Expected Results:

1. User will be updated in the database
2. Project will be updated in the database
3. Projects page will be updated with user name change

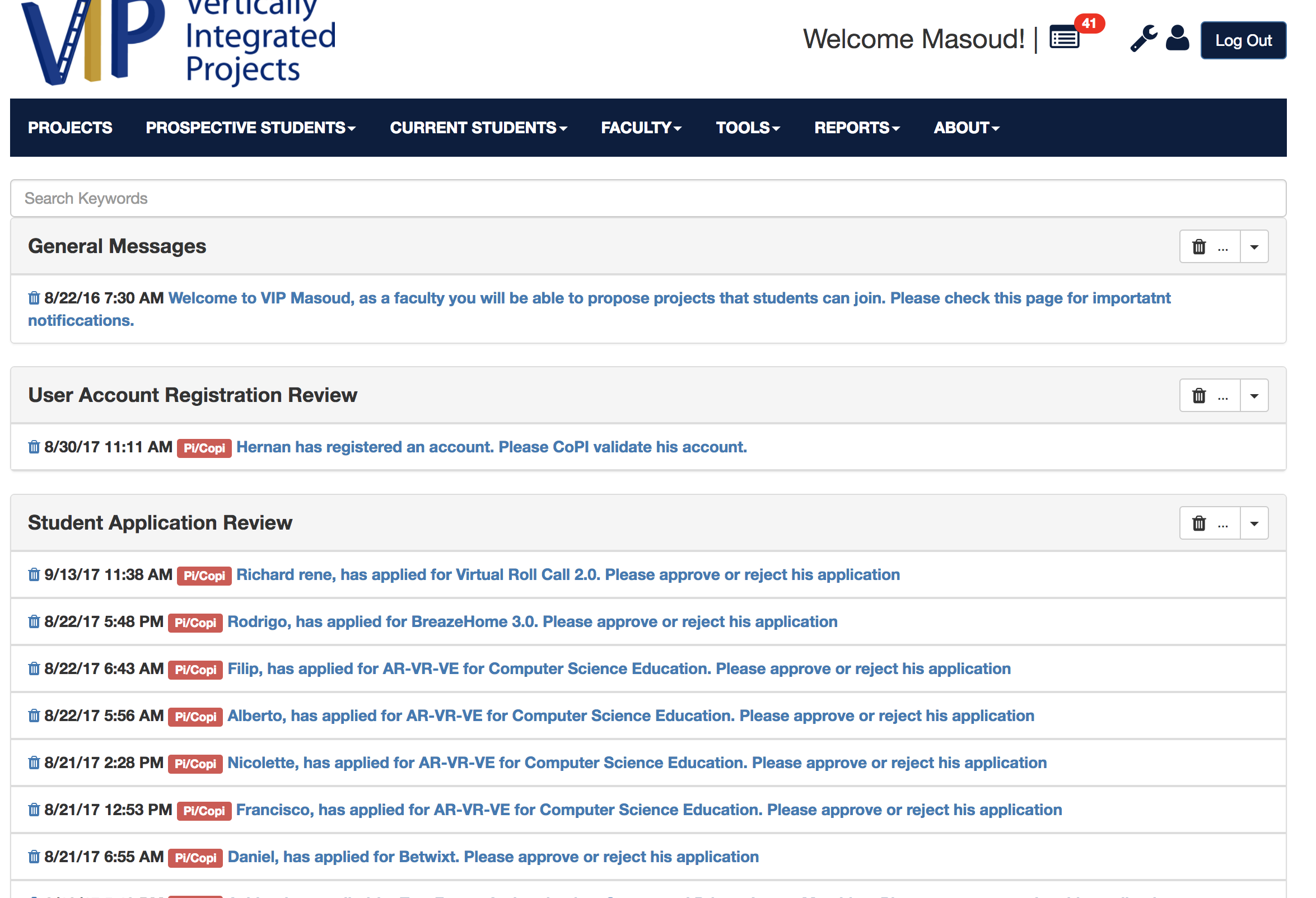
* Actual Result:

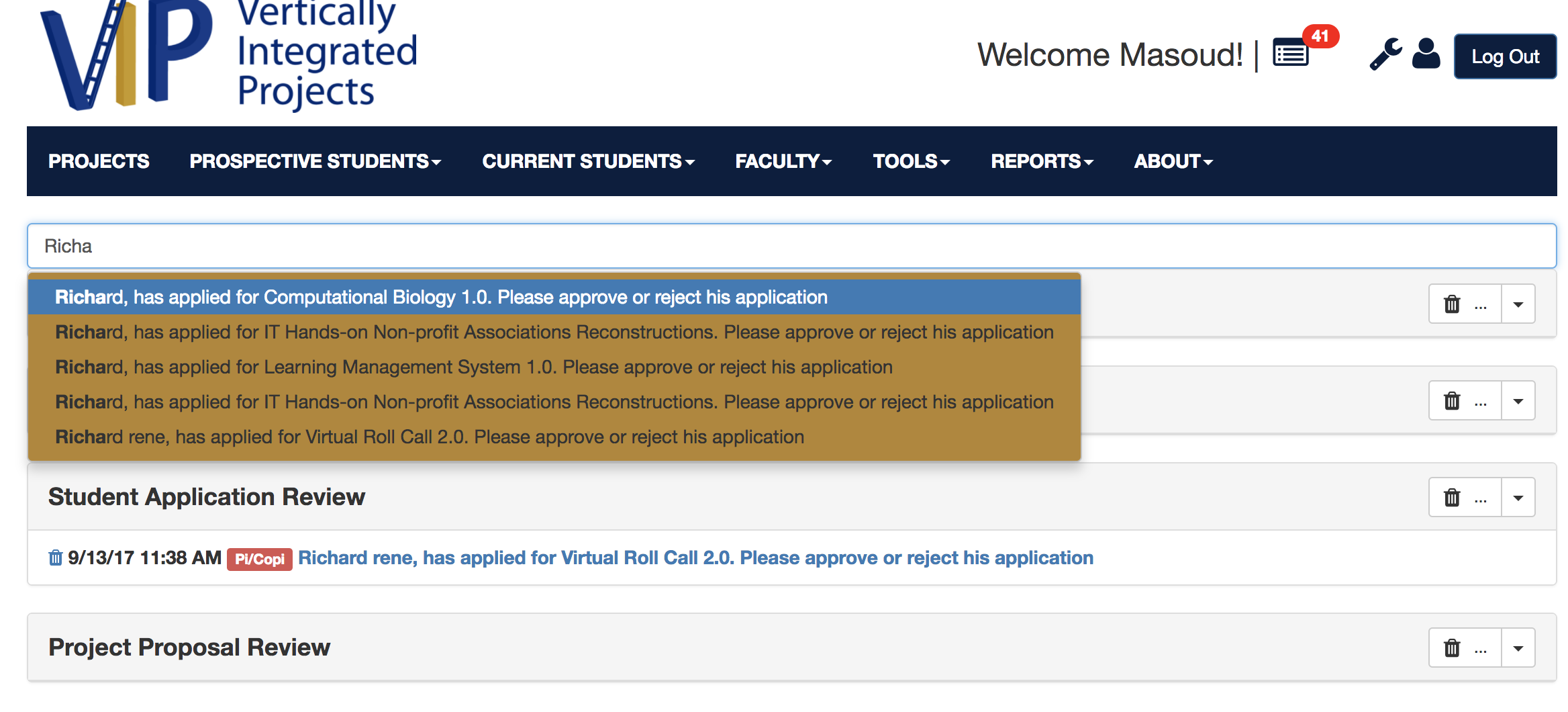
1. User will be updated in the database
2. Project will be updated in the database
3. Projects page will be updated with user name change

* Status (Fail/Pass): PASS

**Visual User Guide**

First Example:





Second example:

