**Feature: As a user I want to search for another user on Bookshelf so that I can see their opinions about the media in their collection.**

Scenario: The user makes a search for another user on Bookshelf to see their reviews.

* Given: the user navigates to the home page.
* When: the user enters a username in the search bar.
* When: the user clicks the search button.
* When: the media in that user’s collection populates.
* When: the user clicks on the username attached to a populated media title.
* When: the page changes to the other user’s page of their media collection.
* Then: the other user’s media and reviews will populate the page.

Scenario: The user makes a search for another user – user searched for does not populate.

* Given: the user navigates to the home page.
* When: the user enters a username in the search bar.
* When: the user clicks the search button.
* When: no media populates.
* Then: an alert populates telling the user that this username does not exist in Bookshelf.

Scenario: As a user I want to contact the moderators so that I can get help with an issue.

Given the user is on the contact us page.

When the user fills in the email input.

When the user fills in the name input.

When the user fills in the message textarea.

When the user clicks the submit button.

Then assert alert comes up for message sent.

Scenario: As a user I want to log out so that my info remains safe.

When the user clicks on the logout button.

Then assert the user is redirected to login page.

Feature System Control

Scenario: As the system I should be able to validate requests

Given users are on the log in screen

When user enters incorrect email

When user enters incorrect password

When user clicks the submit button

Then user gets an alert

Feature: as a moderator, logging in and moderating the website.

Scenario: As a moderator I want to login so that I can determine the validity of media and reviews.

Given the moderator is on the login page.

When the moderator fills in the email input field.

When the moderator fills in the password input field.

When the moderator clicks the login button.

Then assert the moderator is redirected to the moderator page.

Scenario outline: As a moderator I want to review media insertion requests so that I can make sure they are legitimate requests.

When the moderator clicks the pending media button.

When the moderator fills the <title> in the input field.

When the moderator clicks the <status> button.

Then assert alert pops up for successful change.

Scenario: As a moderator I want to see all users so that I can view all users that have signed up.

When the moderator clicks the users button.

Then assert users information populates.

Scenario: As a moderator I want to validate reviews made on media so that I can make sure the reviews are appropriate.

When the moderator clicks the pending reviews button.

When the moderator fills the review <id> in the input field.

When the moderator clicks the <status> button.

Then assert alert pops up for successful change.

Scenario: As a moderator I want to check my messages so that I can make sure users are not having problems.

When the moderator clicks the moderator home button.

Then assert the messages populate.

Scenario: As a moderator I want to view past approved media insertions so that I can view all past decisions.

When the moderator clicks the pending media button.

Then assert past media populates.

Scenario: As a moderator I want to view past denied and approved reviews so that I can view all past decisions.

When the moderator clicks the pending reviews button.

Then assert past reviews populate.

Scenario: As a moderator I want to logout so that my information remains safe.

When the moderator clicks the logout button.

Then assert moderator is redirected to t