

Use Case Testing:

1.

Title: Publish Post

Actor: User

Main Scenario:

1. The user selects the option of creating a new post on their profile page.
2. The system displays the post-creation interface.
3. The user uploads a photo from their device.
4. The system displays the selected image on the post-creation interface, along with input text boxes for caption, rating, and tags.
5. The user inputs the caption, rating, and tags in their corresponding sections.
6. The system displays all entered information about the post.
7. The user clicks on "Publish post"
8. The system produces a post from the user's input and displays it in the feed.
9. The user can see that their post has been published successfully.

Alternatives:

- 2a. System does not display the post-creation interface
- 2a1. User cannot make a post
- 3a. User does not upload a photo
- 3a1. No image will be displayed
- 4a. System does not display the selected image on post-creation interface
- 4a1. User cannot view image
- 5a. User does not enter post details
- 5a1. User remains in post-creation interface
- 6a. System does not display all entered post information
- 6a1. User remains in post-creation interface
- 7a. User does not click "publish post"
- 7a1. User remains in post-creation interface
- 8a. System does not produce a post from users input
- 8a1. User remains in post-creation interface
- 9a. User cannot see their post
- 9a1. Post not successful

Test Situations:

- 1) User selects option that is not create a new post
- 2) Connection error displaying post-creation interface
- 3) User tries to upload file that is not an image
- 4) System displays the wrong post details

- 5) User changes post details
- 6) System displays wrong information
- 7) User clicks different button
- 8) Error producing post
- 9) User sees wrong post

Test Coverage:

Base: number of main and alternative situations = 9

Test situations cover all 9 cases

100% coverage of use case

2.

Title: Comment on a Post

Actor: User

Main Scenario:

1. The user clicks on the comment icon attached to the post.
2. The system displays any comments that are already made on the post.
3. The user clicks on Add Comment.
4. The system activates the comment input field for the user.
5. The user types the comment in the comment input field.
6. The system saves the user's input.
7. The user sends/submits their comment.
8. The system updates the comment section to include the user's new comment.
9. The user can view their comment added to the comment section of the post.

Alternatives:

- 2a. System does not display comments already made on post
- 2a1. User cannot view comments
- 3a. User does not click on add comment button
- 3a1. User can only view post
- 4a. System does not activate the comment input field
- 4a1. User can only view post
- 5a. User does not type comment
- 5a1. No comment added
- 6a. System does not save user input
- 6a1. No comment added
- 7a. User does not submit their comment
- 7a1. No comment added
- 8a. System does not update the comment section to include new comment

- 8a1. No comment added
- 9a. User cannot view their added comment
- 9a1. Comment not successful

Test Situations:

- 1) User clicks on user instead
- 2) Error displaying previous comments
- 3) User clicks on reply button
- 4) Error activating comment input field
- 5) User types comment in wrong field
- 6) Error saving input
- 7) User deleted their comment
- 8) Error including new comment in comment section
- 9) User can't view previous comments anymore

Test Coverage:

Base: number of main and alternative situations = 9

Test situations cover all 9 cases

100% coverage of use case

3.

Title: Register

Actor: User

Main Scenario

- 1. The user enters their username in the username text box
- 2. The user enters their password in the password text box
- 3. The user clicks the "Register" button
- 4. The system creates a new user in the database with the entered username/password
- 5. The user is redirected to the Login page

Alternatives:

- 1a. The user has entered a username that is not unique
 - 1a1. The system displays an error message notifying the user that their username is taken
- 1b. The user has entered a username that is less than 3 characters long
 - 1b1. The system displays an error message saying that usernames must be at least 3 characters long
- 2a. The user has left the password box empty
 - 2a1. The system displays an error message saying that passwords are required for accounts
- 3a. The user clicks "Submit" without fixing errors from steps 1 and 2
 - 3a1. The system displays an error message saying to follow the Login instructions

4a. The database is unresponsive

4a1. The system displays an error message saying that there was an error connecting to the database, and to retry creating an account

5a. The system fails to automatically redirect to Login

5a1. The system displays an error message asking the user to manually click the link to the Login page

Test situations:

- 1) The user enters a username that already exists
- 2) The user enters a username that is under 3 characters long
- 3) The user does not enter a password
- 4) The user tries to submit their credentials with one of the errors from 1-3
- 5) The database's automatic redirect to Login is unavailable

Test Coverage:

Base: number of main and alternative situations = 6

Test situations cover all 6 cases

100% coverage of use case

4.

Title: Like a Post

Actors: User

Main Scenario:

1. The user clicks on the thumbs-up icon attached to the post they are viewing.
2. The system registers the user's like on the post and changes the thumbs-up icon to a heart icon.
3. The user sees that they have successfully liked the post, as the thumbs-up icon has been replaced by a heart icon.

Alternatives:

2a. The system does not display the like icon.

2a1. The user cannot like any post.

3a. The user does not click on the like icon.

3a1. The system does not register any likes, and the icon remains unchanged.

3b. The like icon does not change to a heart icon after clicking.

3b1. The user sees the thumbs-up icon remain unchanged even though they click on the like.

Test Situations:

1. The user clicks the thumbs-up button multiple times on the same post.
2. The system fails to register the user's likes.
3. The system fails to update the icon after the user clicks.

Test Coverage:

Base: number of main and alternative situations = 3

Test situations cover all 3 cases

100% coverage of use case

5.

Title: Following users.

Actors: User

Main scenario:

1. The user clicks on the "Follow" button next to the username of the post creator.
2. The system registers the user's action to follow the username and updates the follow button to unfollow.
3. The user sees the updated button, confirming that they have successfully followed the post creator.

Alternatives:

2a. The system does not display the follow button.

2a1. The user cannot click on the button.

3a. The system does not register the user's action to follow the post creator.

3a1. The system doesn't update the "Follow" button after the user's click.

Test situations:

1. The user clicks on a different button.
2. The system encounters an error and fails to register the user's click action.
3. The follow button isn't displayed.

Test Coverage:

Base: number of main and alternative situations = 3

Test situations cover all 3 cases

100% coverage of use case

6.

Title: View Profile

Actors: User

Main Scenario:

1. The user clicks on a username from the post page
2. The system redirects the user to the profile page associated with the clicked username
3. The user can see the profile page

Alternatives:

2a. The system does not redirect the user to the profile page

2a1. The system displays an error message notifying the user that redirecting failed.

3a. The user cannot see the profile page.

3a1. The system displays an error message asking the user to try again.

Test Situations:

- 1) User clicks a button that is not a username
- 2) Error redirecting user to profile page
- 3) User sees wrong page

Test Coverage:

Base: number of main and alternative situations = 3

Test situations cover all 3 cases

100% coverage of use case

7.

Title: Showcase Favorite Posts

Actor: User

Main Scenario:

1. The user selects the change favorite posts button on their profile page.
2. The user selects which favorite post to change.
3. The system displays all the posts that the user has created.
4. The user clicks on the post they would like to showcase.
5. The system updates the database with the new favorite post id.
6. The user can see the post they selected as their new favorite post.

Alternatives:

2a. The user does not select which post to change.

2a1. The system displays an error message notifying the user that they need to select a post to change.

3a. The user has not created any posts.

3a1. The system displays an error message notifying the user that they need to create a post to showcase their favorite post.

4a. The user does not select a post to showcase.

4a1. The system displays an error message that the user needs to select a post.

5a. The database is unresponsive

5a1. The system displays an error message that their favorite post could not be updated, and to retry.

6a. The user cannot see the post they selected.

6a1. Favorite post was not updated successfully.

Test Situations:

1) User selects incorrect button.

2) User select wrong option

3) Error displaying the posts that the user has created.

4) User selects incorrect post.

5) Error updating database.

6) User sees wrong favorite post.

Test Coverage:

Base: number of main and alternative situations = 6

Test situations cover all 6 cases

100% coverage of use case

Unit Testing:

Username (valid)	Username (invalid)
length \geq 3 characters (1) is unique (2)	length $<$ 3 characters (3) is not unique (4)

Rating (valid)	Rating (invalid)
$0 \leq \text{value} \leq 11$ (1) Is an integer (2)	value < 0 (3) value ≥ 11 (4) Not an integer

--	--

Post Items (valid)	Post Items (invalid)
Has image url (1) Has a caption (2) Has a tag (3) Has a rating (4)	No image url (5) No caption (6) No tag (7) No rating (8)

Caption(valid)	Caption (invalid)
0<length<400 characters (1)	length=0 characters(2) Length >400 characters (3)

Comment(valid)	Comment (invalid)
0<length<200 characters (1)	length=0 characters(2) Length >200 characters (3)

Tag (valid)	Tag (invalid)
In {Book, Music, Movie, Game} (1)	Not in {Book, Music, Movie, Game} (2)

Login Credentials (valid)	Login Credentials (invalid)
Username in Database (1) Password matches the username's password in the database (2) Both fields filled in (3)	Username not in Database (4) Password does not match the username's password in the database (5) One or more fields empty(6)

Bio(valid)	Bio (invalid)
0<length<300 characters (1)	length=0 characters(2) Length > 300 characters (3)

Profile page (valid)	Profile page(Invalid)
Has a username(1) Has a bio (2) Has a profile picture(3) Displays any posts the user has featured. (4)	Doesn't have a username (5) Doesn't have a bio button (6) Doesn't have a profile picture (7) Doesn't show a post the user has featured (8)

Delete Post(valid)	Delete Post (invalid)
Post is removed from the feed(1) Post is removed from the user's profile (2)	Post is still in the feed (3) Post is still on the user's profile (4)

Unit Testing Screenshot:

```

(base) brettpalmer@Bretts-MacBook-Pro-3 src % npm test

> react-lab@0.1.0 test
> react-scripts test --env=jsdom --verbose --watchAll=false

PASS src/Login.test.js
  Login Component
    ✓ valid login navigates to "/Post" (126 ms)
    ✓ username not in database shows error (40 ms)
    ✓ incorrect password shows error (31 ms)
    ✓ empty fields show error message (9 ms)

PASS src/Register.test.js
  ✓ successful registration navigates to login page (26 ms)
  ✓ short username is rejected (9 ms)
  ✓ empty password field is rejected (8 ms)
  ✓ existing username registration shows username is already taken (11 ms)

PASS src/caption&rating&tag/Rating.test.js
  Rating component
    ✓ calls onChange with valid rating when input changes (8 ms)
    ✓ does not call onChange with invalid rating (2 ms)

PASS src/caption&rating&tag/Tag.test.js
  Tag component
    ✓ calls onChange with valid tag (9 ms)
    ✓ does not call onChange with invalid tag (2 ms)

PASS src/caption&rating&tag/Caption.test.js
  Caption component
    ✓ calls onChange with valid caption when input length is between 1 and 399 characters (8 ms)
    ✓ does not call onChange with invalid caption when input length is less than or equal to 0 characters or more than 399 characters (1 ms)

Test Suites: 5 passed, 5 total
Tests: 14 passed, 14 total
Snapshots: 0 total
Time: 1.047 s
Ran all test suites.

```

Acceptance Testing:

-Both test subjects gave permission to share their test results

Jensen Aspinall Testing

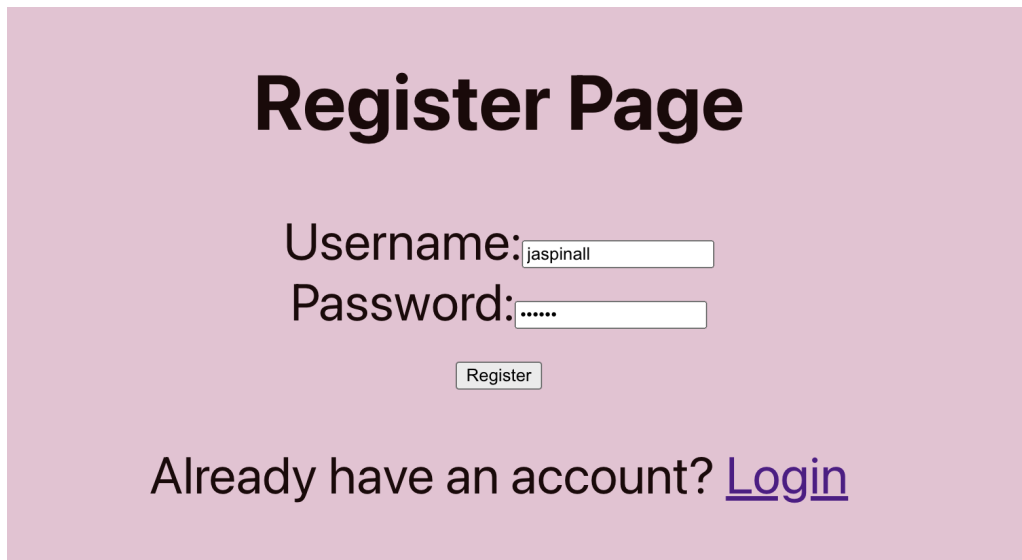
Pros:

- Jensen loved the review feature, and had fun posting about Gossip Girl.

- She likes the simple UI and review structure

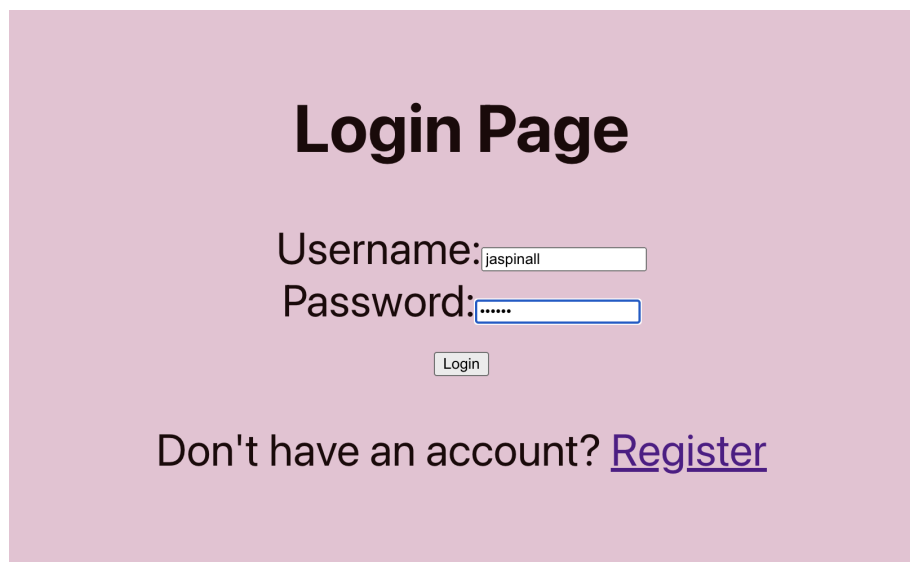
Cons:

- Going from Register to Login is confusing, going from Register to Feed makes more sense.
- Posting with the URL Wasn't clear, Also posting itself was small at the bottom.
- Adding feature post broke the site



A screenshot of a web page titled "Register Page" in large, bold, black font. Below the title, there are two input fields: "Username:" followed by a text box containing "jaspinall", and "Password:" followed by a text box containing ".....". Below these fields is a small button labeled "Register". At the bottom of the page, it says "Already have an account? [Login](#)". The entire page has a light purple background.

Jensen created a Poppin account

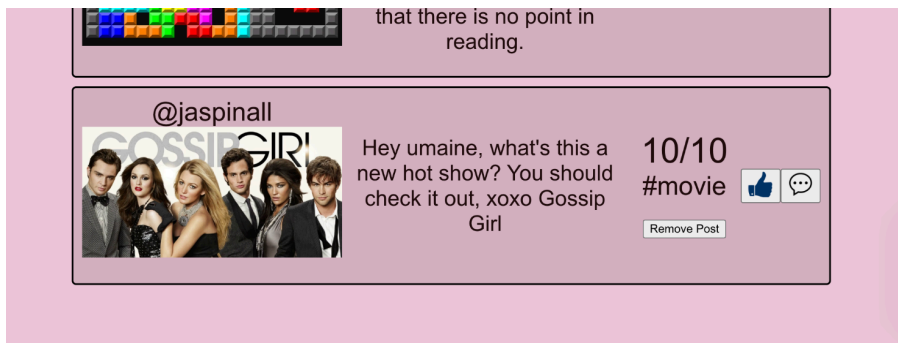


A screenshot of a web page titled "Login Page" in large, bold, black font. Below the title, there are two input fields: "Username:" followed by a text box containing "jaspinall", and "Password:" followed by a text box containing ".....". Below these fields is a small button labeled "Login". At the bottom of the page, it says "Don't have an account? [Register](#)". The entire page has a light purple background.

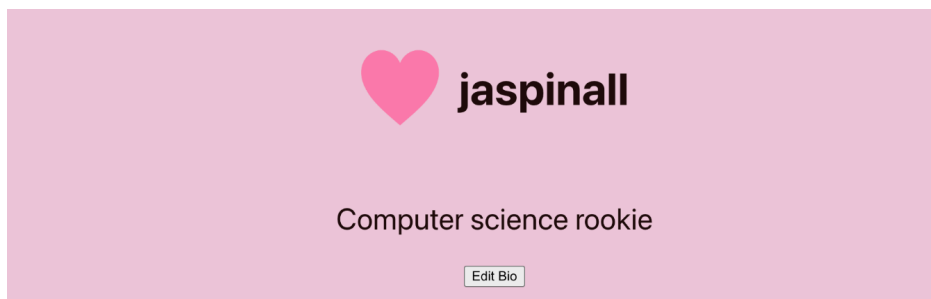
Jensen logged into her account



Jensen liked and commented on a post



Jensen created a post about Gossip Girl



Jensen visited her profile and made a custom bio

Sarah Stutzman Testing

Pros:

Sarah enjoyed how easy it was to make an account and login. She also enjoyed how there were multiple post features and being able to interact with other peoples posts.

Cons:

Sarah suggested that the posts of the accounts you follow should be at the top of the feed. Filtering the feed was a feature we wanted to add but didn't have enough time to implement.

Poppin

Register Page

Username:

Password:

Register

Already have an account? [Login](#)

Sarah made an account

Login Page

Username:

Password:


Login

Don't have an account? [Register](#)

Sarah logged in with her account


@jaspinall

Follow



Hey umaine, what's this a new hot show? You should check it out, xoxo Gossip Girl

10/10 #movie



Add a comment:

Submit Comment

Manuela 4/18/2024, 11:30:12 AM

One of the best show ever!!!!

Delete

sarah 4/18/2024, 2:17:57 PM

omg love this show

Delete


Sarah interacted with Jensen's post by commenting on it

Poppin

Your Profile

Create a Post

@sarah Follow





love island

I love to watch this show
with my friends

10/10
#movie

Remove Post



Finally, Sarah added to the feed page by making her own post