Unit Test Report

Product Name: Clothr
Team Name: Clothr Team
Date: December 2, 2017

Database testing:
    -creating user
        A) Create a test user with a username (test), password (test), and email (test@gmail.com). After registering, checked if the user goes to the next page and has the ability to swipe clothes. Checked on the parse-dashboard if the user exists in the database in the user tab with the username and email fields filled. The password field in the database should be hidden.
        B) Created a user that gives an error
            a) Duplicated a user that already exists (test)
            b) Used emails that is not an an email format (testEmail, testing.com,)
    -login after creating
        A) Used an existing user's credentials (username: test, password: test) and clicked the "Login" button and went to the swiping page
        B) Used a non existing user to receive an error
            - Appropriate error messages appear if the login button is pressed w/o any fields filled or if any fields are missing, if the username/password is incorrect, or if the username/password is wrong
    -delete user
        A) Created user samTest and then view it in the database then deleted the user from the database and then checked if the user was deleted. Repeated with multiple users
    - making sure clothes save per user
        A) Check if the "checkout" button on the bottom of the swipe page leads to the users list of clothes they saved. Check if swiping right to save the clothes adds the clothing to the checkout page and if swiping left to delete the clothes didn't make it on the checkout page
    -making sure clothes details are loaded from the database
        A) Swiped multiple clothes and double clicked to see if all the information was populated and was correct. Went to the website to double check if the details matched what was being shown in the app.
Filter testing:
    -making sure filters selected stay highlighted (UI)
        A) Select a filter, scroll down and up. Check if the selected filter is still highlighted.
        B) Search for a filter (only for Brand and Retailer filters). Select the filter searched and scroll down to that specific filter. Check if the selected filter is highlighted. Repeat test 1 for this filter
        C) Select a filter. Check if it appears in the saved filters collection view.

D) Go back to the main page. Go to filters view page and check again to see if the filter is selected. Repeat step 1 for this filter.

-deselecting filters

A) Select a filter. Scroll up and down to make sure it's highlighted. Select the highlighted filter to deselect it. Scroll up and down to make sure it's not highlighted anymore.

B) search for a filter (only for Brand and Retailer filters). Deselect the filter. Scroll down to that filter and check to see if the filter is deselected. Repeated test 1 for this filter.

C) Deselect a filter from the saved filters collection view. Scroll down to the deselected filter in the collection view containing all filters and see if the filter is deselected.

D) Go back to the main page. Go to filters view page and check again to see if the filter is deselected. Repeat test1 for this filter

-adding filters to array

A) Check the filters array count. Select a filter and check the filter id. Check the filters array count. If it's greater by 1, look into the array and check the filter id. Compare this id to the filter id when it was selected.

-removing filters to array

A) Check the filters array count. Select a filter and check the filter id. Check the filters array count. If it's decreased by 1, look into the array and check the end filter id(if there is more than one, otherwise skip this and next step). Compare this id to the filter id when it was selected.

-adding filters to api calls

A) Search for an item in the main page. Check the array count of the filters. Go to filters page. Select a filter and look at its ID. Go back to the main page, where the api call is made in the background. In the API call, check the array count again. If it is greater by 1, look into the array and compare the ID to the selected filter.

-make sure main page is updated when return

A) Search for an item in the main page. Go to filters page. Select a filter. Go back to main page and wait for it to load. If it shows up with a different picture, filter works. If the picture is the same, filter didn't work.

Searching within filter testing:

-typing in a word leaves only proper results

A) Tested multiple search items (supreme, pokemon, shirt, pants, abcdefg, 1234, @@@@, abc1, blue, red , purple5, blue pokemon pants, pink dolphin, macys, water bottle, cheez it, red supreme), made sure that the clothes shown are relevant to the item searched. If nothing is found an error message pops and and says "Sorry your search item was not found."

Swiping testing:

- make sure thumbs-up and thumbs-down show up when swiping

A) On main page, tap and drag the piece of clothing in any direction to test if thumbs-down/thumbs-up image appears. If the image is dragged to any area on

the left half of the page, the thumbs-up image appears. If the image is dragged to any area on the right side of the page, the thumbs-down image appears.Observe that it does.
- make sure animation works
    A) On main page, tap and drag on a piece of clothing to test the animation moves and behaves how it was intended. Dragged the clothing item left, right, up down, and checked the corners of the screen.
    -make sure images shown are linked to products in array
    A) Swiped a product and made sure that the product object being shown is the right product object in the buffer.

Checkout testing:
    -go to the website of the clothing item selected and being able to buy it
    A) Swipe an item to the right. Go to the save items page. Click on a cell. If it brings user to site, check if it's the right product.

Continuous Login:
    -closing the app and staying logged in
    A) From the beginning page, check the username and user ID when user logs in. After logging in, quite the app. When the app opens again, check the current user ID and compare it to the user ID from before.

Search Bar Testing:
- make sure api search team matches what the user types into the search bar
    A)

Logout Testing:
    -button function works and redirects to homepage
    A) When a user is logged in, make sure they are able to go into the settings page and tap the "logout" button. After pressing it, make sure they are redirected to the homepage. After closing and reopening the app, make sure the user still has to login or register instead of starting on the swipe page.

Tutorial Feature Testing:
    -dropdown menu works
    A) Check if the data for each settings section/cell loads in the right area. So the Change Info displays the user's username and email, and two buttons to reset their password and delete their account. The tutorial displays tutorial images and text. The about us section has text describing our app.
    B) Check if each cell does not display anything when entering the settings page and displays the cell's data with a dropdown if the cell section is tapped on.
    -making sure the scroll bar works with pictures
    A) When a cell/section is tapped on, check that the information/data only shows within the box of the section. Make sure any data outside of the section box does not overlap the other cells and have the data outside the box be displayed when the user scrolls through the box.

Deleting Clothes from Checkout
- Successfully deletes a single item of clothing from the checkout page

A) On the checkout page, choose any item of clothing and swipe left until delete animation occurs and item disappears from the page. Leave the page via clicking on back and return to checkout page. Observe that the item is gone.
- delete clothes details from database
    A) Go to the database and look at current item details. Go to app, and go to the checkout page. On the checkout page, choose any item of clothing and swipe left until delete animation occurs and item disappears from the page. Leave the page via clicking on back and check back into the database to ensure the item is deleted.

Reset Password
- In the settings page, make sure that tapping "reset password" displays an alert for the user to confirm the action.
    A) If the user clicks "cancel", the alert is dismissed and nothing happens
    B) If the user clicks "continue", the alert is dismissed and nothing happens

Testing if clothes have loaded and are swipeable:
-check if continuously calling api searches results in a bigger current product array

Details testing:
-product shown in the details view page is corresponding to the image in the main page
    A) Load the image. Check the productID. Double tap the picture. Check the productID on this page and compare to the productID from the main page.