Shipt - QA Engineer - Interview Exercise

Upon completion of this exercise, push all your files into github and send us a link.

1. Navigate to www.shipt.com. Choose a feature of the site and write a description or test case to describe how to test the feature.

Test Case: Ability to Update Account Personal Information

This functionality handles the ability to update a user's account personal information, which includes user name, user phone number, and user personal email.

Test Step	**Expected Result**
1.Navigate and log in to www.Shipt.com	User is logged in to test account
2.Navigate to "My Account"	User is directed to the account details page
3.While under the account tab, hit edit on the "Personal Information" window.	"Edit Personal Information" window should appear.
4.Ensure the fields Name, Email, and Phone Number are present in the modal.	All fields are present
5.Under Name, clear and populate with new data. Save and check results.	Name field should be updated with correct new data.
6.Repeat step five using the Email and Phone Number fields.	Fields should be updated with correct new data.
7.Ensure each field follows correct field validation. Required fields cannot be left empty. Phone number field should not allow letters or special characters. Email must be in correct format.	All fields follow correct field validation, and must be populated if field is required.

2. Locate one bug or bad workflow within the app.

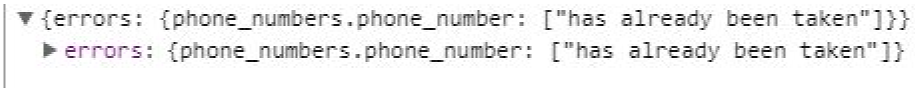
   a. Explain the behavior you are seeing
   b. Include any useful errors or screenshots
   c. Explain why or how it needs to be corrected
   d. What are the steps you would take to report the issue?

Defect: Account Field “Phone Number” can not be updated to previously used number.

Steps to reproduce:

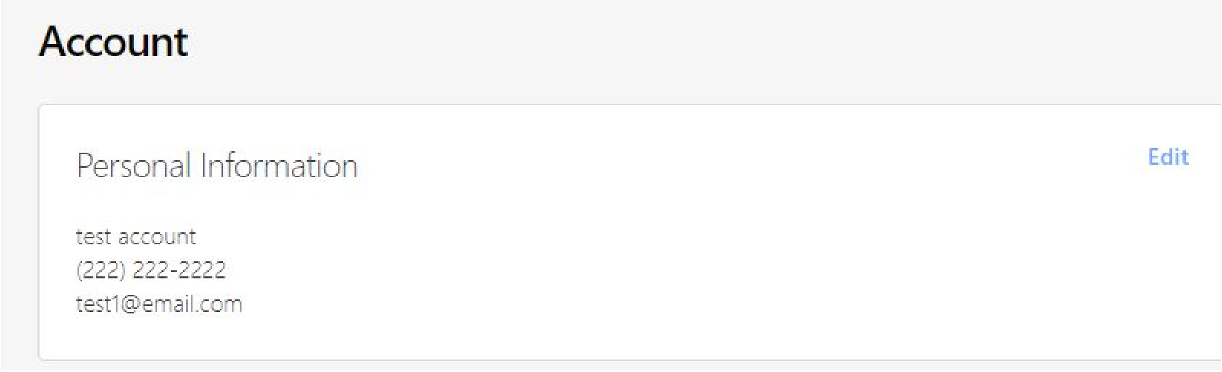
1. Navigate to www.shipt.com, log in to a test user, and go to “My Account”.
2. Edit Personal Information.
3. Take note of current phone number, change to a new number and save.
4. Attempt to edit the phone number back to the original and save.
5. Check phone number.

Expected Result: Phone number is updated to the previously used value.
Actual Result: The value will not be returned to the previously used phone number. No UI error message is thrown, but in the Network Dev tools tab, an error is present:



```
▼{errors: {phone_numbers.phone_number: ["has already been taken"]}}
  ▶errors: {phone_numbers.phone_number: ["has already been taken"]}
```

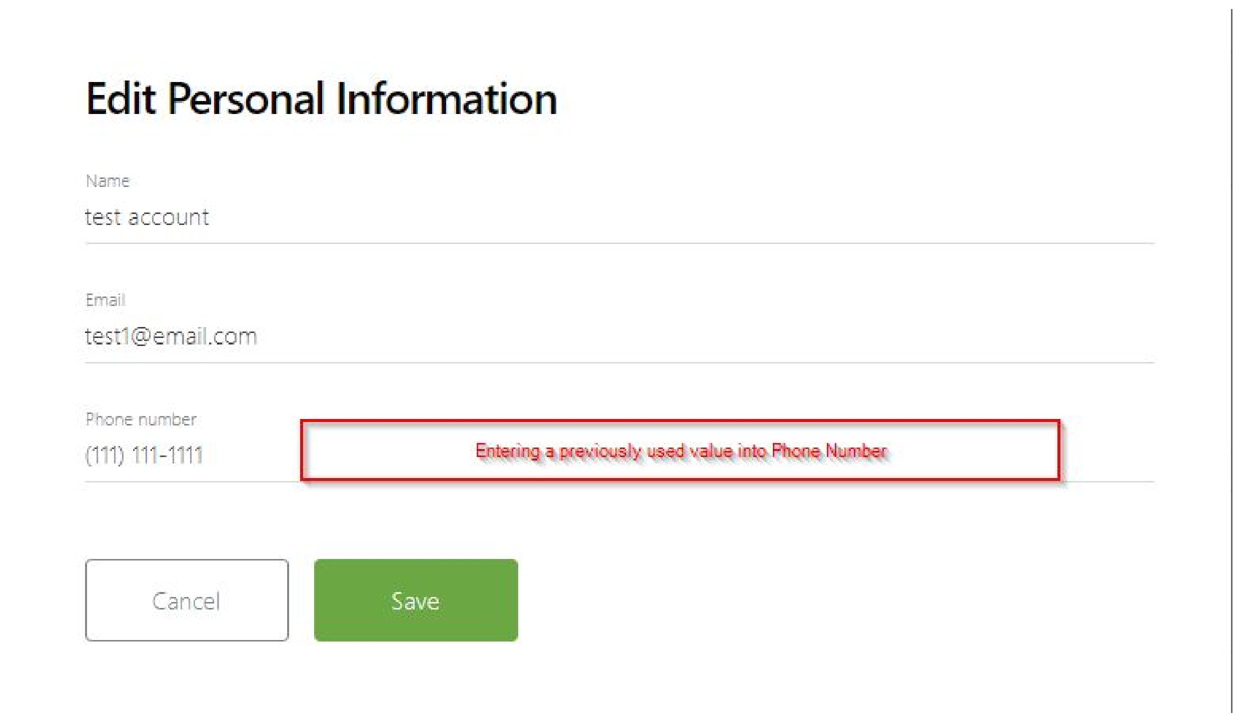
More images:



Account

Personal Information

test account
(222) 222-2222
test1@email.com



Edit Personal Information

Name

test account

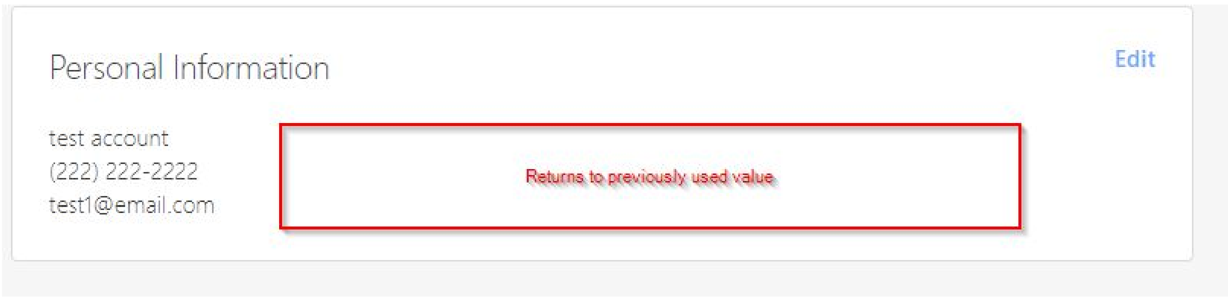
Email

test1@email.com

Phone number

(111) 111-1111

Entering a previously used value into Phone Number



Personal Information

test account
(222) 222-2222
test1@email.com

Returns to previously used value

If retaining previously used phone numbers in the database is a must, then it should allow the reuse of those values as well.

Steps to report:

1. Enter a defect card into whichever service company uses (i.e. Jira).
2. Judgement call, depending on priority of issue, further communication may be required, such as slack message or email to a superior.
3. If there is a developer currently working on a story in the affected area, communication with said developer would also be needed.

3. What priority would you give this bug (Scale of 1-5, 1 being highest) and why?

Priority 3. I would assume there is a workaround on the back end. Could not figure out a workaround from a user standpoint. The defect possibly prevents users from using correct phone number for orders.

4. In the database, you have the following tables (columns listed on each)

    a. How would you define the relationships between each table? (1 to many, many to many, 1 to 1)

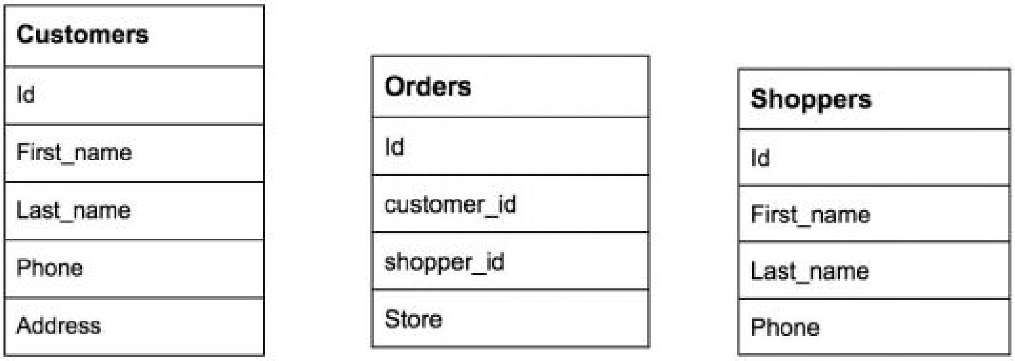
    b. Write a SQL query which would return the ID and Store of an order tied to a specific Customer and Shopper

Relationships:

a. **Customer to Orders:**1 to many. **Orders to Shoppers:** 1 to 1. **Customers to Shoppers:** Many to many.

b.

```
SELECT Id, Store
From Orders
WHERE customer_id='45' AND shopper_id='305886';
```



Customers
Id
First_name
Last_name
Phone
Address

Orders
Id
customer_id
shopper_id
Store

Shoppers
Id
First_name
Last_name
Phone

Automation

Write an automated test which accomplishes the following

1. Access www.shipt.com in the browser
2. Using the feature you chose for step 1, write an automated test
3. The test should validate the feature works as expected

We recommend you choose from the following languages/tools. Feel free to use other tools if you feel they lend better to this type of testing.

- Selenium
- Calabash
- Typescript/Javascript
- Ruby

Automation Assessment

1. If you chose to use a tool or language other than the recommended, briefly explain why.

I chose Protractor. It uses a selenium server, and is written in Typescript/javascript. It has a built in reporter and implicit waits, and I have experience using it, so it made getting the scripts up and running much quicker for me.

2. How did you go about locating the elements for your tests?

Using the DOM in the Dev tools. Honestly, I mostly copied the Xpaths straight from the DOM, and exported them as variables in my config file, for the sake of the demo. For more elegant targeting, though. I would build more flexible Xpaths or jquery targeting for reusability.

3. What do you believe are the most common causes for instability in UI automation?

Elements with non unique ID's. Performance spikes on published pages. Format differences across different browsers.

4. How do you make your tests consistent and easy to debug?

I try to make my test cases as granular as possible, with in depth descriptions in their describe() functions. Also, baking expectations and assertions straight into my exported functions allows to track down failed test to the exact click or field entry that it failed on.