

Manual Testing Procedure - Shopist

Table Of Contents

Manual Testing Procedure - Shopist	1
Account Creation and Log in	2
Testing Account Creation	2
Testing Sign In	5
Testing Checkout Functionality, and CRUD	8
Make a Purchase	8
Known Issues with the Checkout	16
Testing submission of a Product Review	18
Testing Search Functionality	22

Account Creation and Log in

Testing Account Creation

To create an account the user clicks ‘Sign Up’ from the nav bar, or clicks ‘Get Started’ on the button contained within the Jumbotron on the main page. Reference Figure 1 and Figure 2.

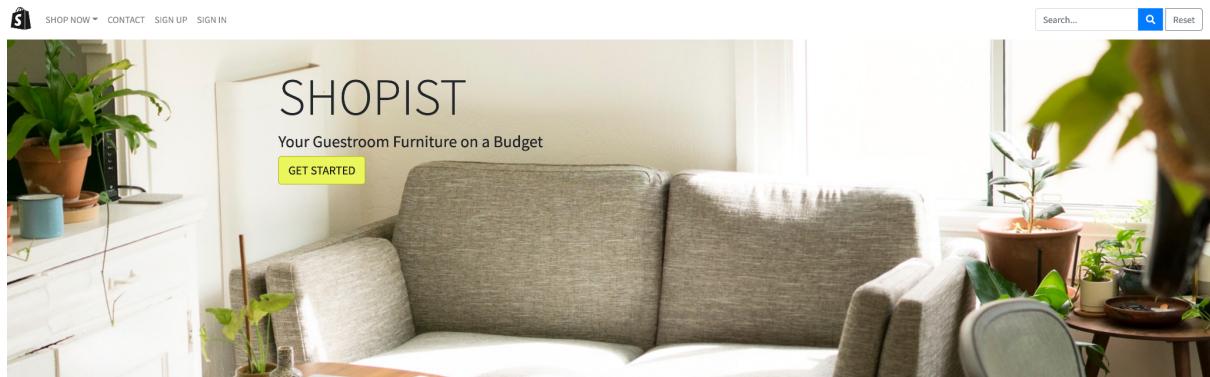


Figure 1: Click ‘Get Started’ to create an account or ‘Sign Up’

CREATE A NEW ACCOUNT

Enter the following information to create a new account

First name*

Last name*

Username*

Required. 150 characters or fewer. Letters, digits and @./+/-/_ only.

Password*

Your password can't be too similar to your other personal information.
Your password must contain at least 8 characters.
Your password can't be a commonly used password.
Your password can't be entirely numeric.

Password confirmation*

Enter the same password as before, for verification.

Email*

eg. youremail@gmail.com

SIGN UP

Figure 2: Sign up page presented to the user

1. Clicking on 'SIGN UP' with none of the fields filled in presents the following feedback to the user. Figure 3.

Enter the following information to create a new account

First name*

Last name*

 Please fill out this field.

Username*

Required. 150 characters or fewer. Letters, digits and @./+/-/_ only.

Password*

Figure 3: Validation feedback on submission of a blank form.

2. Filling in the email address field with an 'incorrect' email type presents the following feedback. Figure 4.

Email*

eg. youremail@gmail.com

Please include an '@' in the email address. 'asdfsadf' is missing an '@'.

SIGN UP

Figure 4: 'Email address' must correspond to a valid email regular expression.

3. Filling in a 'Password' and a 'Repeat Password' that do not match presents the following feedback. Figure 5.

Password confirmation*

The two password fields didn't match.

Enter the same password as before, for verification.

Figure 5: 'Passwords must match' feedback to the user.

4. Filling in a ‘Username’ that already exists results in the following error. Figure 6.

The screenshot shows a registration form with three fields: 'First name*' containing 'asdf', 'Last name*' containing 'asdf', and 'Username*' containing 'a'. The 'Username*' field is highlighted with a red border and has a red exclamation mark icon in the top right corner. Below the field, a red message says 'A user with that username already exists.'

First name*

asdf

Last name*

asdf

Username*

a

A user with that username already exists.

Figure 6: Cannot use an existing ‘Username’

5. Upon Successful Registration, a notification message is presented, along with a User icon with the user’s name on the navbar. Figure 7.

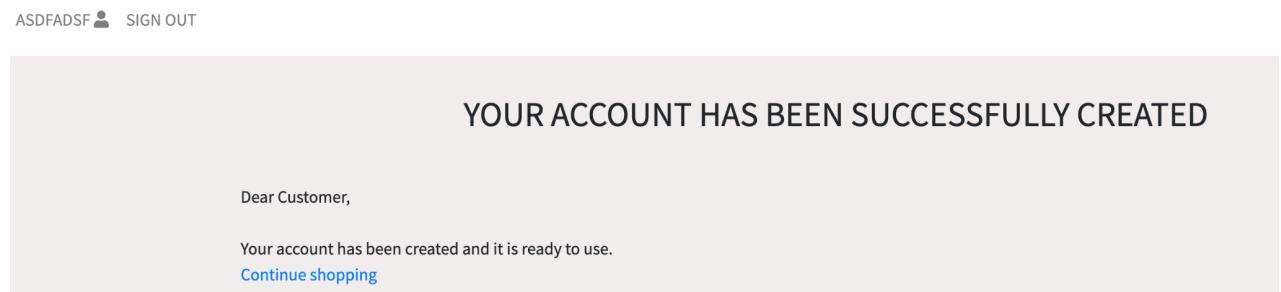


Figure 7: Notification to indicate account has been successfully created.

6. We can confirm that the new user `asdfadsf` has been added to the database by navigating to '<https://shopist-dml.herokuapp.com/admin/auth/user/>'. Figure 8.

Django administration						WELCOME, DARRENLEITH
Home : Authentication and Authorization > Users						
AUTHENTICATION AND AUTHORIZATION						
Groups + Add						
Users + Add						
CHECKOUT						
Cart Items + Add						
Carts + Add						
Orders						
STORE						
Categories + Add						
Products + Add						
Reviews + Add						

Select user to change

Action: 0 of 14 selected

<input type="checkbox"/>	USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
<input type="checkbox"/>	a	asdf@adfasf.com	a	a	<input checked="" type="radio"/>
<input type="checkbox"/>	ab	asdf@asdf.com	ab	ab	<input checked="" type="radio"/>
<input type="checkbox"/>	asdf	asdfsadf@ad.com	asdf	asdf	<input checked="" type="radio"/>
<input type="checkbox"/>	asdfasdf	asdfadsf@asdfasdf.com	asdfadsf	asdfasdf	<input checked="" type="radio"/>
<input type="checkbox"/>	billyshake	2beornot2be@yolo.com	Bill	Shakespeare	<input checked="" type="radio"/>
<input type="checkbox"/>	billyshake2323	2beornot2be@yolo.com	Bill	Shakespeare	<input checked="" type="radio"/>
<input type="checkbox"/>	bob	bob@b.com	bob	b	<input checked="" type="radio"/>
<input type="checkbox"/>	darrenleith	darrenleith@test.com			<input checked="" type="radio"/>
<input type="checkbox"/>	j	j@j.com	j	j	<input checked="" type="radio"/>
<input type="checkbox"/>	m	m@m.com	m	m	<input checked="" type="radio"/>
<input type="checkbox"/>	m134	m@m.com	m	m	<input checked="" type="radio"/>
<input type="checkbox"/>	m2	m@m.com	m	m	<input checked="" type="radio"/>
<input type="checkbox"/>	michaelcor	littlemikey@gmail.com	michael	corleone	<input checked="" type="radio"/>
<input type="checkbox"/>	q	q@q.com	q	q	<input checked="" type="radio"/>

14 users

Figure 8: User **asdfadsf** has been created and added to the database.

Testing Sign In

We can re-use the same account just created in the previous section to test Sign In. From the home page click on the ‘Sign In’ navbar link. The following page is presented. Figure 9.

REGISTERED CUSTOMERS ONLY

Username*

Password*

SIGN IN

Figure 9: ‘Sign In’ page presented to the user.

1. Clicking ‘LOGIN’ with none of the fields filled in presents the following feedback to the user. Figure 10.

The screenshot shows a login form titled "REGISTERED CUSTOMERS ONLY". It contains two input fields: "Username*" and "Password*". Both fields have a red border and a placeholder text "Please fill out this field." A green "SIGN IN" button is at the bottom.

Username*

Please fill out this field.

Password*

Please fill out this field.

SIGN IN

Figure 10: Validation feedback on submission of a blank form.

2. Filling in the ‘Email’ field with an incorrect ‘Password’ presents the following feedback. Figure 11.

The screenshot shows a login form titled "REGISTERED CUSTOMERS ONLY". It contains two input fields: "Username*" and "Password*". The "Username" field has the value "asdfadsf". A red callout box above the "Password" field contains the text: "• Please enter a correct username and password. Note that both fields may be case-sensitive." A green "SIGN IN" button is at the bottom.

REGISTERED CUSTOMERS ONLY

- Please enter a correct username and password. Note that both fields may be case-sensitive.

Username*

asdfadsf

Password*

SIGN IN

Figure 11: Validation feedback on submission of an incorrect password.

3. Filling in the ‘Email’ field and ‘Password’ field with a non-existent account likewise presents the following feedback. Figure 12.

The screenshot shows a login interface. At the top, the text "REGISTERED CUSTOMERS ONLY" is displayed in a large, bold, dark font. Below this, a red callout box contains the message: "Please enter a correct username and password. Note that both fields may be case-sensitive." The form has two input fields: "Username*" and "Password*". The "Username" field contains the text "ASDFASDFASDFDSAFAASDF". A green "SIGN IN" button is located below the inputs.

Figure 12: Validation feedback on submission of a non-existent username.

4. A correct combination of ‘Email’ and ‘Password’ presents a user icon on the navbar along with the user’s name. The user can click on this to view their Profile Page. Figure 13



Figure 13: Clicking on username in the navbar presents the users profile page which shows all of the users Order History.

Testing Checkout Functionality, and CRUD

Make a Purchase

1. Only a logged in user can make a purchase. This can be seen in Figure 14.

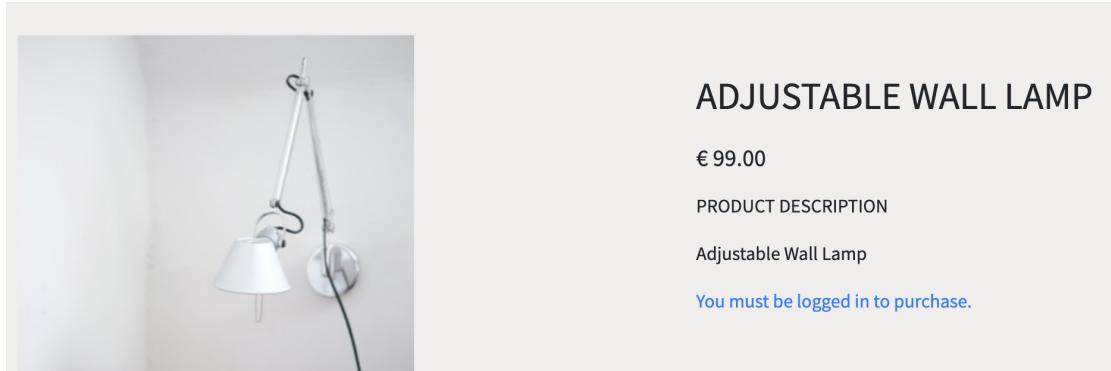


Figure 14: 'You must be logged in to purchase'.

2. Once logged in, the user will then have the option to 'Add to Cart'. Figure 15.

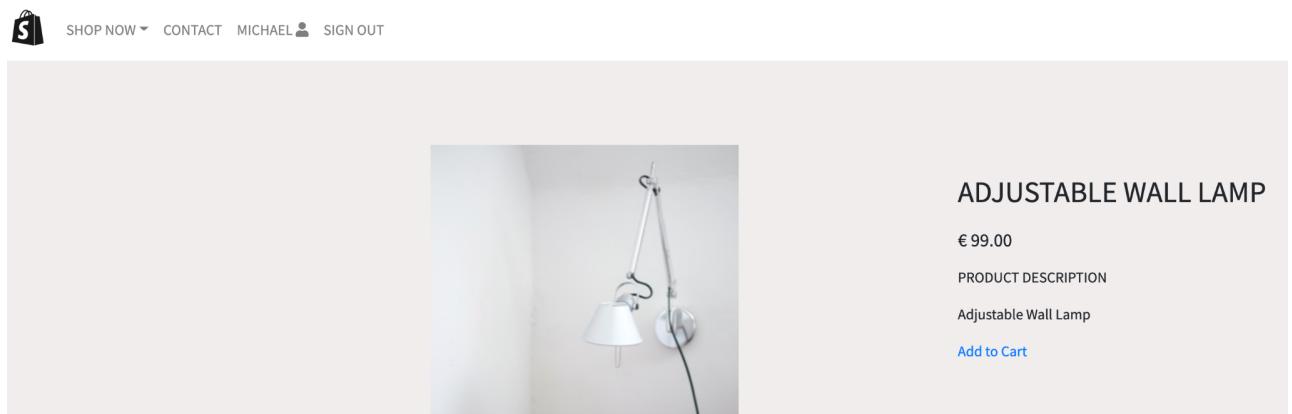


Figure 15: Once logged in, the user has the option to 'Add to Cart'.

3. After clicking on 'Add to Cart' the checkout option in the navbar updates to show the number of items currently in the cart. 'Your Cart' page (<https://shopist-dml.herokuapp.com/cart/>) is also displayed. Figure 16.

SHOP NOW ▾ CONTACT MICHAEL SIGN OUT (1)

YOUR CART

Your items		Check out
	Adjustable Wall Lamp SKU:31 Unit Price: €99.00 Qty:1 x 99.00	€99.00 [+] [trash]
Please review your cart items before proceeding with the payment.		
Your total is: €99.00		
Pay with Card		
CONTINUE SHOPPING		

Figure 16: The 'Your Cart' checkout page.

4. Here, the user can Update the number of items in their cart by clicking on the + or - icon, and can also delete an item entirely from their cart by using the trash icon. Figure 17.

Your items

	Chrome Arc Lamp SKU:24 Unit Price: €320.00 Qty:3 x 320.00	€960.00 [+] [−] [trash]
--	--	--

Figure 17: User can add (assuming there is sufficient stock), remove, and delete items entirely from their cart using the provided icons.

5. If there is insufficient qty of a given item, the '+' icon will no longer show. Figure 18.

SHOP NOW ▾ CONTACT MICHAEL SIGN OUT (3)

Your items

	Adjustable Wall Lamp SKU:31 Unit Price: €99.00 Qty:3 x 99.00	€297.00 [−] [trash]
--	---	---

Figure 18: There are only Qty 3 'Adjustable Wall Lamps' in our store (see Figure 19 showing the database back-end). The '+' icon is no longer present once we have put all x3 in the Cart.

Select product to change

Action: ----- Go 0 of 25 selected

<input type="checkbox"/>	NAME	PRICE	STOCK	AVAILABLE	DESCRIPTION
<input type="checkbox"/>	Adjustable Wall Lamp	99.00	3	<input checked="" type="checkbox"/>	Adjustable Wall Lamp

Figure 19: Database back-end showing there are only Qty 3 'Adjustable Wall Lamps' currently available for sale in our store.

- Once the User is ready to Pay, the User clicks on 'Pay with Card' and the following validation form is shown:

The image shows a payment validation form overlaid on a shopping cart page. The cart summary at the top indicates 1 item: 'Adjustable Wall Lamp' at €297.00, with a total of €297.00. The form itself is titled 'Shopist-Store' and 'Shopis-Store - New Order'. It contains fields for 'Email' (with placeholder '@example.com'), 'Same billing & shipping info' (checkbox checked), 'Name', 'Address', 'ZIP', 'City', and 'United States' (dropdown). A 'Payment Info' button with a right-pointing arrow is at the bottom. The background shows a dark-themed website with a 'Check out' button and a 'CONTINUE SHOPPING' button.

Figure 20: Payment validation form

7. Inputting any incorrect information into the fields results in the erroneous fields being highlighted in red.

The screenshot shows the 'Shopist-Store' application interface for placing a new order. At the top, there's a header with a close button (X) and the text 'Shopis-Store - New Order'. Below the header, there are two input fields: one for email ('asdf') which has a red border indicating an error, and another for a password ('asdf'). A checkbox labeled 'Same billing & shipping info' is checked. Below these, there's a section for address input. It includes a dropdown for country ('United States') and two adjacent input fields for ZIP and City, both of which have red borders. A large blue button at the bottom right says 'Payment Info' with a right-pointing arrow.

Figure 21: Payment validation in action.

8. In order for the transaction to successfully work, the user must input the same email address they registered with. In the example below, user 'michaelcor' has an email address 'littlemikey@gmail.com'. Therefore, this needs to be inputted into the payment form (see Figure 23).

Action:	USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
<input type="checkbox"/>	michaelcor	littlemikey@gmail.com	michael	corleone	
1 user					

Figure 22: The user must use the same email address they registered with. Note: In a future version of the app, validation could be employed to ensure this is adhered to.

9. A correctly filled out payment form.

The screenshot shows a mobile application interface for 'Shopist-Store'. At the top, it says 'Shopist-Store' and 'Shopis-Store - New Order'. Below that is an input field with an envelope icon and the email 'littlemikey@gmail.com'. There is a checked checkbox labeled 'Same billing & shipping info'. Below that is a dropdown menu with a person icon and the name 'Mike'. Inside the dropdown, there is another dropdown with a location pin icon and the text 'foo foo bar bar'. Below that is a table with two columns: '90210' and 'Beverly Hills'. At the bottom of the dropdown is a dropdown menu with the text 'United States'. At the very bottom is a large blue button with the text 'Payment Info' and a right-pointing arrow.

Figure 23: A correctly filled out payment validation form. Note: In a future version of the app, validation will be employed for countries that do not use a ZIP Code.

10. The user can then submit a test card as shown in Figure 24.

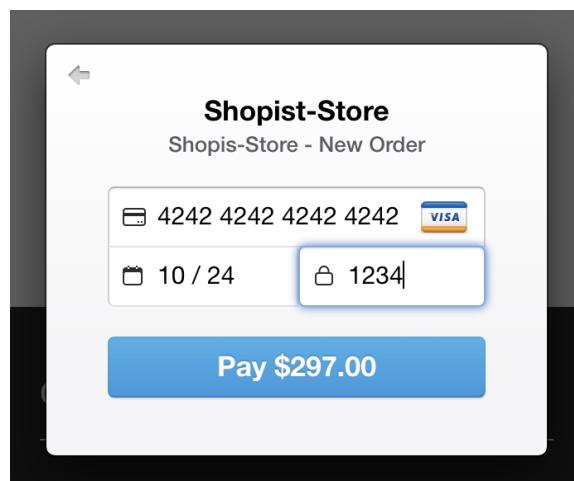


Figure 24: Test card submission.

11. If an order successfully goes through, a green check mark will show on the submit button, indicating success. Figure 25.

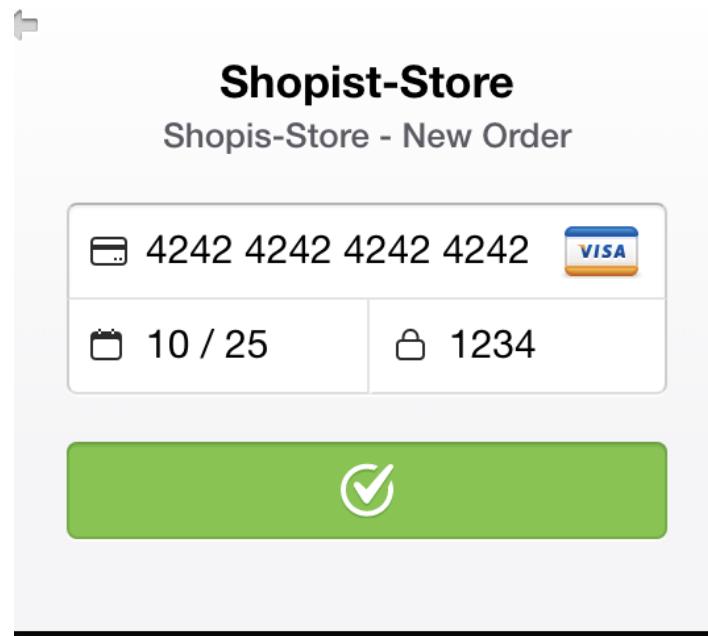


Figure 25: Successful submission of payment.

12. Likewise, form validation ensures all fields are correctly filled out. Figure 26.

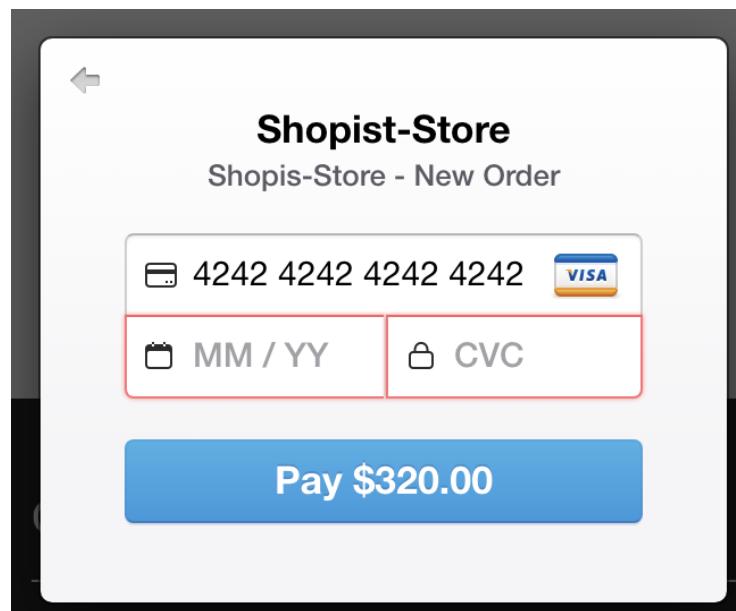


Figure 26: Invalid input.

13. A message will appear to say thank-you, along with the order number. Figure 27.

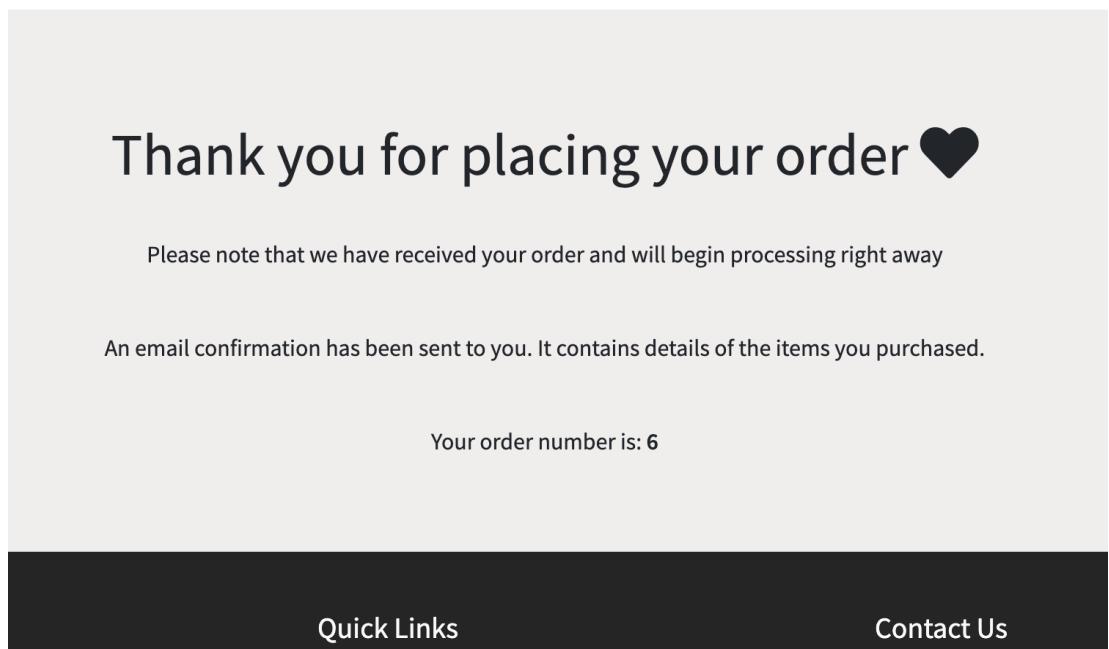


Figure 27: Feedback to the user, indicating a successful order.

14. The user can then go to their Profile page and see their Order History, with a list of all of their Orders. Figure 28.

A screenshot of a user profile page for "michaelcor's Profile Page". The page features a large header with the user's name. Below the header, there are sections for "YOUR REVIEWS" and "YOUR ORDER HISTORY". The "YOUR ORDER HISTORY" section displays a table with one row of data.

Order Number	Order Date	Total Amount	Action
5	05 Sep 2021	297.00	View Order

Figure 28: List of User's Order History

15. After clicking on any of the Orders (Order Number 5 in this test), we can see all the details of the transaction. Figure 29.

MICHAEL SIGN OUT

Order Details

Order: #5 Date: 05 Sep 2021 Order Total: \$297.00 Order Status: ✓ Complete.	Billing Address: Mike foo foo bar bar Beverly Hills 90210 US		
Product Description	Qty	Unit Price	Sub-Total
Adjustable Wall Lamp	3	\$99.00	\$297.00
			Total \$297.00
			Total Paid \$297.00
Shipping Address: Mike foo foo bar bar Beverly Hills 90210 US	Payment Details: The order #5 has been paid successfully		

[Print Order](#) [Back to Profile](#)

Figure 29: User's Order Details. This form is available for every Order the user has performed. Note the User also has the option to 'Print Order'.

16. User has the option to print the Order. Figure 30.

shopist-dml.herokuapp.com/accounts/profile/order/5

W Reading List

W CONTACT MICHAEL

Order Details

Order: #5 Date: 05 Sep 2021 Order Total: \$297.00 Order Status: ✓ Complete.	Billing Address: Mike foo foo bar bar Beverly Hills 90210 US		
Product Description	Qty	Unit Price	Sub-Total
Adjustable Wall Lamp	3	\$99.00	\$297.00
			Total \$297.00
			Total Paid \$297.00
Shipping Address: Mike foo foo bar bar Beverly Hills 90210 US	Payment Details: The order #5 has been paid successfully		

[Print Order](#) [Back to Profile](#)

Print 1 sheet of paper

Destination EPSON L355 Series

Pages All

Copies 1

Color Color

More settings

Cancel Print

Figure 30: Printing the Order

Known Issues with the Checkout

1. An error will occur if trying to Purchase from a country that does not have a Zip Code.

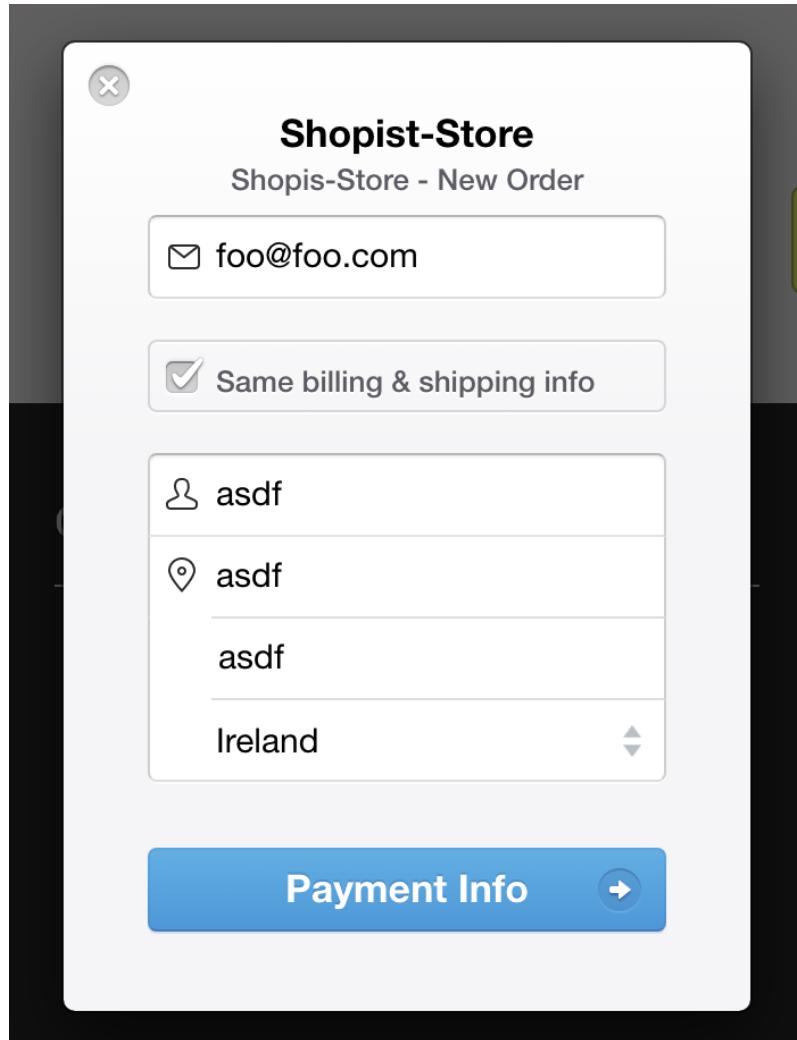
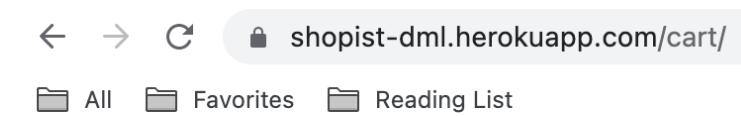


Figure 31: Re-creating a bug when purchasing from a country that does **not** have a Zip Code.



Server Error (500)

Figure 32: A generic 500 error will present itself when purchasing from a country without a ZipCode. In this version of the app, we have not provided validation to handle this bug. See Figure 33.

MultiValueDictKeyError at /cart/

'stripeBillingAddressZip'

Request Method: POST
Request URL: http://127.0.0.1:8000/cart/
Django Version: 3.2.5
Exception Type: MultiValueDictKeyError
Exception Value: 'stripeBillingAddressZip'
Exception Location: /Library/Frameworks/Python.framework/Versions/3.9/lib/python3.9/site-packages/django/utils/datastructures.py, line 7
Python Executable: /usr/local/bin/python3
Python Version: 3.9.5
Python Path: ['/Users/darrenleith/Documents/web-development/shopist', '/Library/Frameworks/Python.framework/Versions/3.9/lib/python39.zip', '/Library/Frameworks/Python.framework/Versions/3.9/lib/python3.9', '/Library/Frameworks/Python.framework/Versions/3.9/lib/python3.9/lib-dynload', '/Users/darrenleith/Library/Python/3.9/lib/python/site-packages', '/Library/Frameworks/Python.framework/Versions/3.9/lib/python3.9/site-packages']
Server time: Sun, 05 Sep 2021 19:43:10 +0000

Traceback [Switch to copy-and-paste view](#)

```
/Library/Frameworks/Python.framework/Versions/3.9/lib/python3.9/site-packages/django/utils/datastructures.py, line 76, in __getitem__  
76.         list_ = super().__getitem__(key)  
► Local vars
```

During handling of the above exception ('stripeBillingAddressZip'), another exception occurred:

```
/Library/Frameworks/Python.framework/Versions/3.9/lib/python3.9/site-packages/django/core/handlers/exception.py, line 47, in inner  
47.             response = get_response(request)  
► Local vars
```

```
/Library/Frameworks/Python.framework/Versions/3.9/lib/python3.9/site-packages/django/core/handlers/base.py, line 181, in _get_response
```

Figure 33: A 'stripeBillingAddressZip' error. This is a known bug present in this iteration of the app.

Testing submission of a Product Review

1. Similar to purchasing, only a logged in user can post a review. Figure 34.

Figure 34: There is no option to submit a review as a guest. Guests can only read.

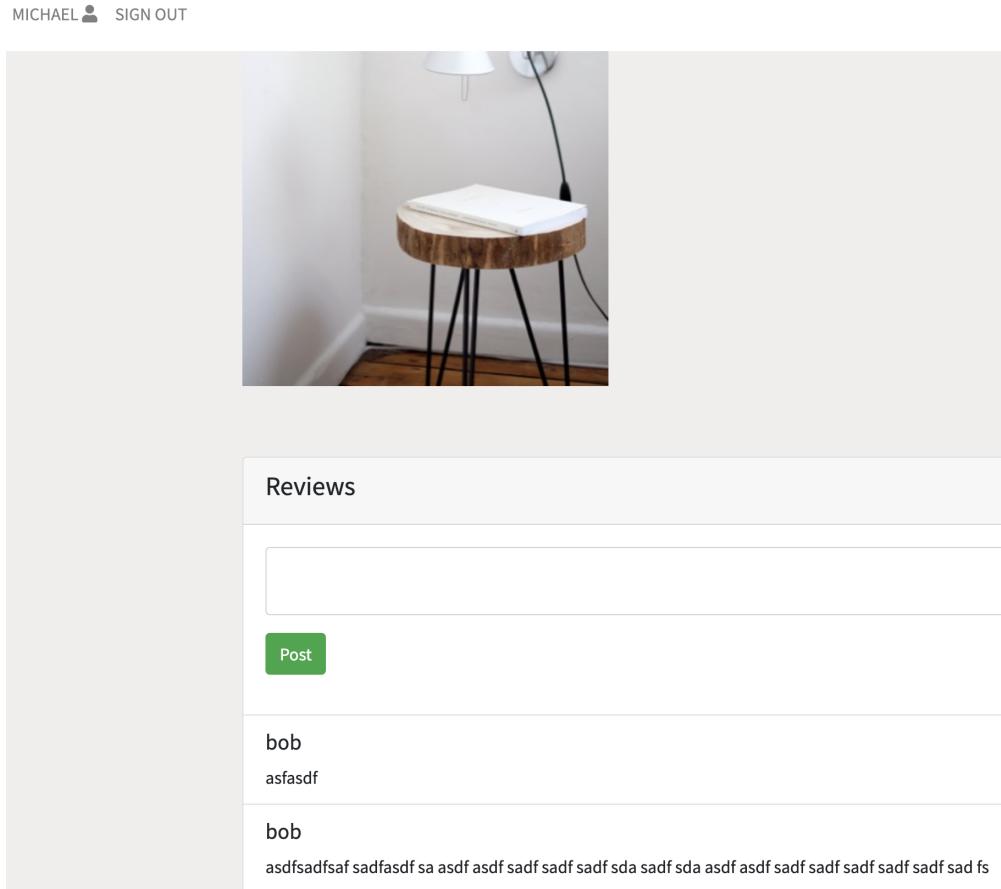


Figure 35: A logged in user has the option to Post.

2. When submitting a review, blank text will simply not update the review section. In Figure 36 we submit the comment 'foo foo foo'

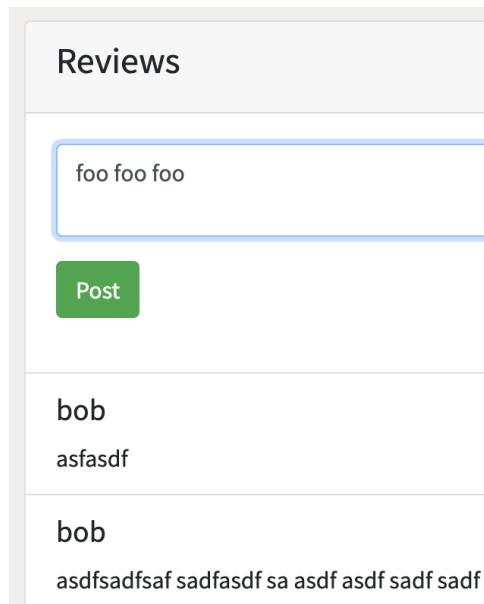


Figure 36: Testing submission of a review

3. This review then updates the review field below it.

Reviews

bob
asfasdf

bob
asdafsadsaf sadfasdf sa asdf asdf sadf sadf sda sadf sda asdf asdf sadf sadf sadf sadf sadf sad fs

michaelcor
foo foo foo

Figure 37: Review successfully submitted.

- The user can then see a list of all reviews they have made on their profile page.
Figure 38.

michaelcor's Profile Page

Figure 38: A list of all of the user's reviews is available on the Profile page. The customer also has the option to delete any review (see Figure 39).

5. The user can delete any of their reviews by clicking the Delete button. This presents a modal as shown in Figure 39.

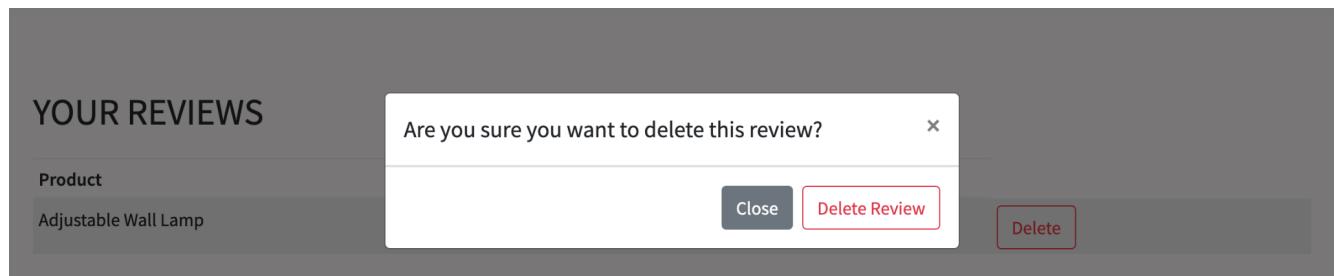


Figure 39: Modal to confirm deletion.

6. Once selected, the review is permanently deleted from the back-end. Figure 39.

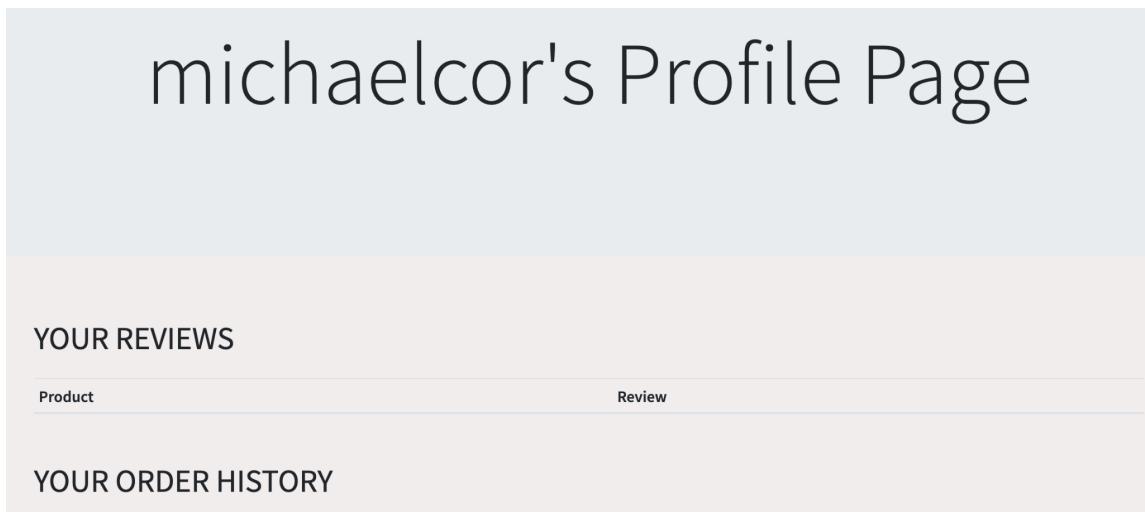


Figure 40: Review has been permanently deleted.

Testing Search Functionality

1. Using the navbar, a user can search the list of Products. Figure 41

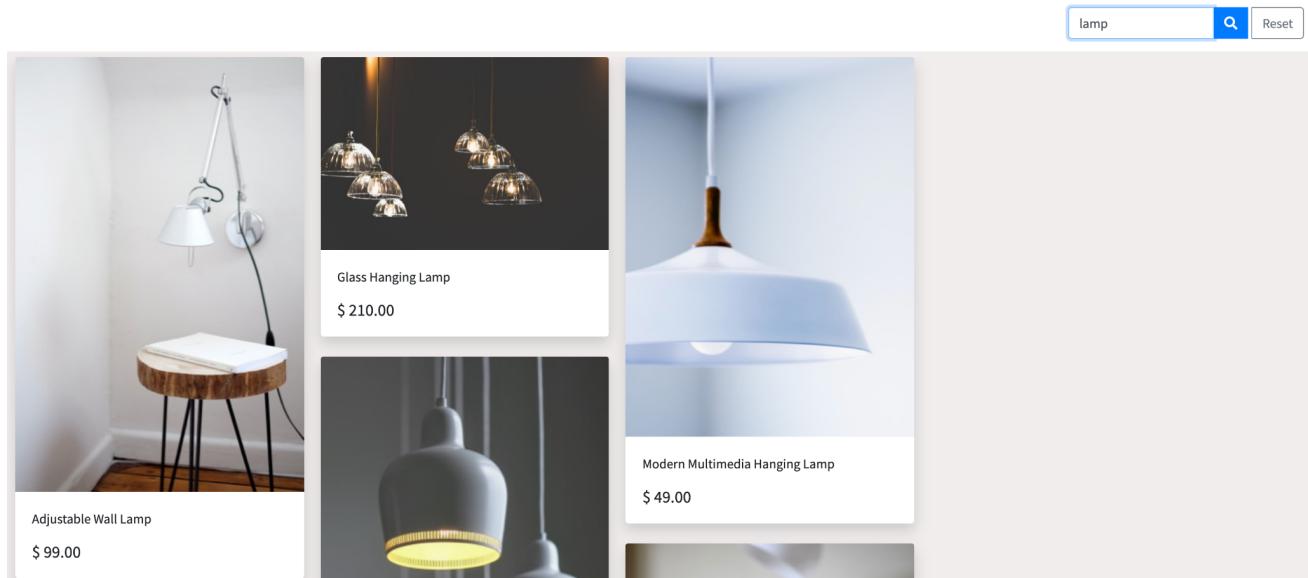


Figure 41: Searching on 'lamp' only retrieves lamps. Likewise, clicking the Reset button will retrieve all Products from the database.

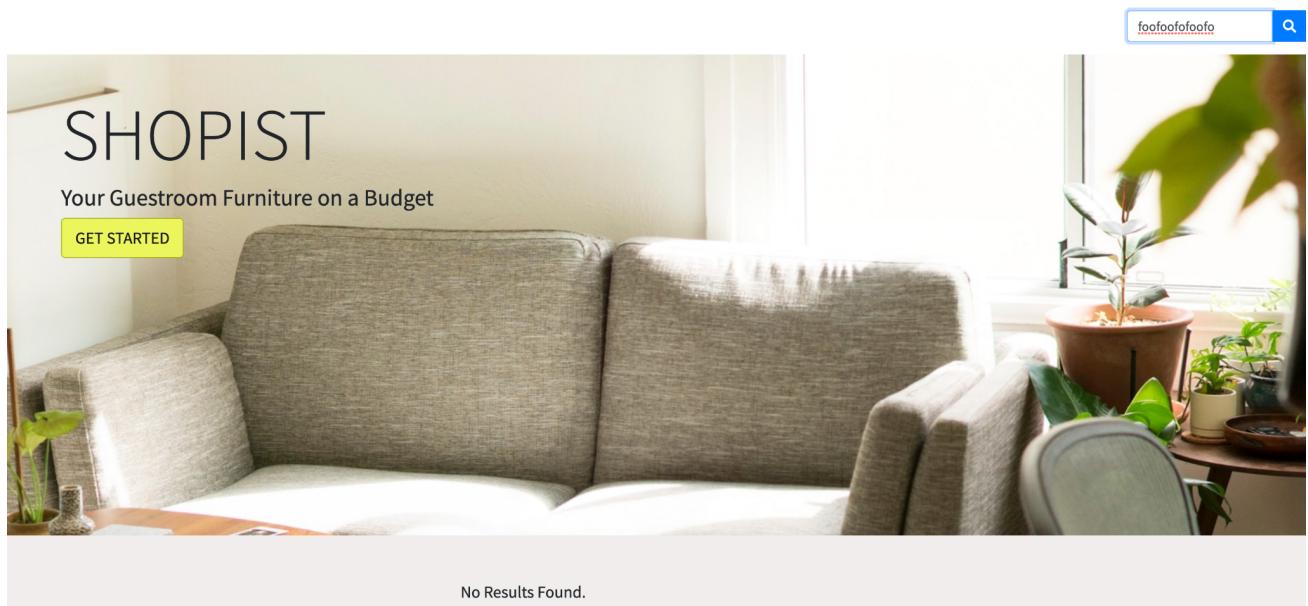


Figure 41: 'No Results Found' message if match not found.