

## **Spotstitch Web app Testing:**

### User Onboarding Workflow:

1: Possibly using input mask in phone number field.



### Account Information

We're designing and creating an open ecosystem; so innovative ideas can turn into world changing action.

Don

Comeaux

Don\_C

7757717158

**Input Mask** 

As a user, I feel this looks a little unprofessional and would not be sure of the formatting and if I am filling it out correctly.

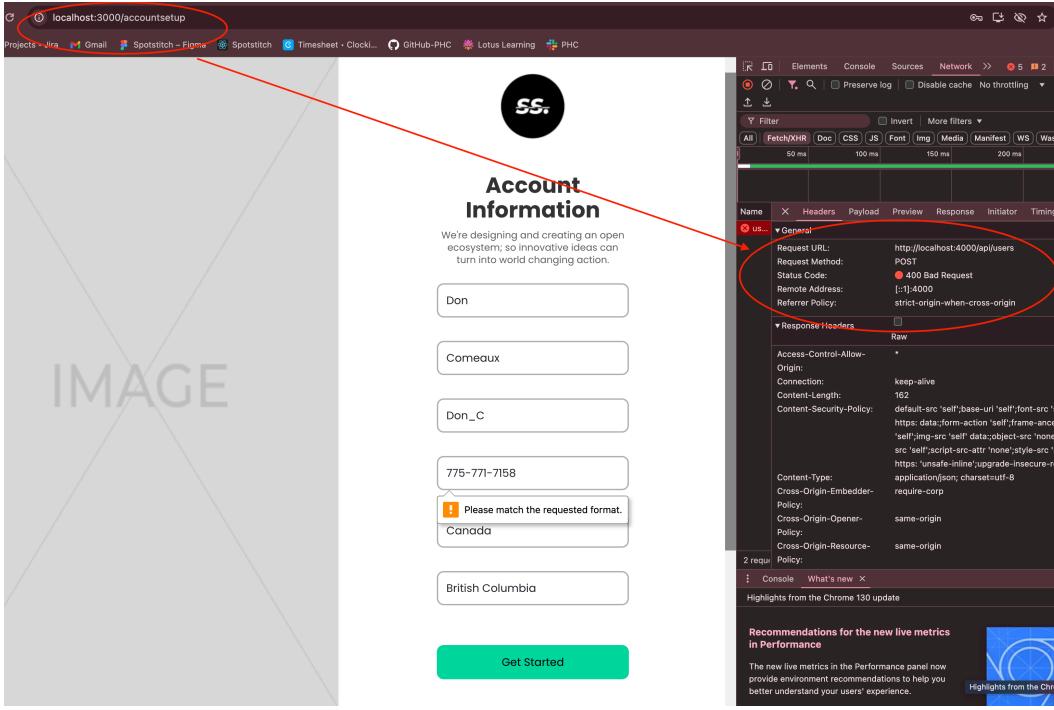
#### Cons:

- \* It needs to be localized for users' specific country or region.
- \* If not working flawlessly, it could cause user abandonment.

#### Benefits:

- \* removing users' uncertainty over how to fill out a particular restricted field
- \* reducing users' anxiety over possibly filling out the field incorrectly, and receiving errors as a result
- \* reducing the number of validation errors all users experience, which can sometimes be the cause of site abandonments.

## 2: Get Started button non-functional.



### Description:

When user clicks the 'Get Started' button, user gets 400 bad request  
I am currently on `localhost:3000/accountsetup` (shown in the browser tab)  
But the API request is going to `localhost:4000/api/users` (shown in the network tab)  
This mismatch in ports (3000 vs 4000) and the difference in endpoints is causing the 400 Bad Request error.

### Preconditions:

Have your network inspector open  
Go to <http://localhost:3000/userssignup>

### Steps to Reproduce:

1. Fill out all account information
2. Click 'Get Started' button
3. Observe status code

### Expected:

User is directed to email verification page

### Actual:

400 Bad Request Error

### 3: Next Button non-functional

The screenshot shows a web application interface for uploading a profile photo. At the top, there's a placeholder image with the text "IMAGE". Below it is a circular logo with "SS.". The main heading is "Add a profile photo." followed by a subtext: "We're designing and creating an open ecosystem; so innovative ideas can turn into world changing action." A placeholder image of a man with a beard holding a glass is shown. Below the image are two buttons: a green "Next" button and a grey "Skip" button. At the bottom, there are links for "Privacy Policy" and "Terms of Service". To the right of the application window is the Chrome DevTools Network tab. A red oval highlights the status code row for a PUT request to "http://localhost:8080/v1/user/image", which shows a status code of 401 Unauthorized. The status bar at the bottom of the browser window also displays "Status 401 Unauthorized".

#### Description:

When user clicks 'Next' button, user gets 401 Unauthorized

User likely not sending the proper authentication credentials with the request.

Possible reasons:

No authentication token/credentials were sent with the request

The authentication token/credentials are invalid or expired

#### Preconditions:

Have your network inspector open

Go to <http://localhost:3000/profileimage>

#### Steps to Reproduce:

1. Click 'Next' button
2. Observe network status code

#### Expected:

User is directed to <http://localhost:3000/accounttype>

#### Actual:

Status 401 Unauthorized

## 4: On Search not identified

The screenshot shows the browser's developer tools open to the 'Console' tab. A red box highlights the error message: "ReferenceError: onSearch is not defined". Below the error, the stack trace shows the function call chain. To the right, the 'Sources' tab is active, displaying the source code for 'CategorySelectionA73.js'. A red box highlights the line where the error occurs: "const handleInputChange = (event) => {". The code is a functional component that handles button clicks and input changes.

```
const handleButtonClick = (index) => {
  if (buttonClickedIndices.includes(index)) {
    setButtonClickedIndices(buttonClickedIndices.filter((clickedIndex) =>
      clickedIndex !== index));
  } else {
    setButtonClickedIndices([...buttonClickedIndices, index]);
  }
};

const handleInputChange = (event) => {
  setQuery(event.target.value);
  onSearch(event.target.value);
};

return (
  <>
    <div className="boss">
      <div className="left">
        <img className="holdering" src={holdering} alt="Placeholder image" />
      </div>
      <div className="right">
        <div className="topspace"></div>
        <div className="topspace"></div>
        <div className="topspace"></div>
        <div className="welcometitle">Let's get you stitched in!
        <div className="topspace"></div>
        <div className="topspace"></div>
        <div className="smalldescription">
          Find your seam. What brings you to Spotstitch?
        </div>
        </div>
        </div>
        </div>
        </div>
        <input type="text" className="searchbox" value={query} onChange={handleInputChange} />
      </div>
    </div>
  </>
);
```

### Description:

When user inputs text into search field, system throws uncaught runtime error

Need to define the **onSearch** function in user component

Error occurs because there's a reference to an **onSearch** function that hasn't been defined or properly passed to your component

### Preconditions:

Have your console inspector open

Go to <http://localhost:3000/categoryselection>

### Steps to Reproduce:

1. Input text into search field
2. Observe console inspector and error message that appears

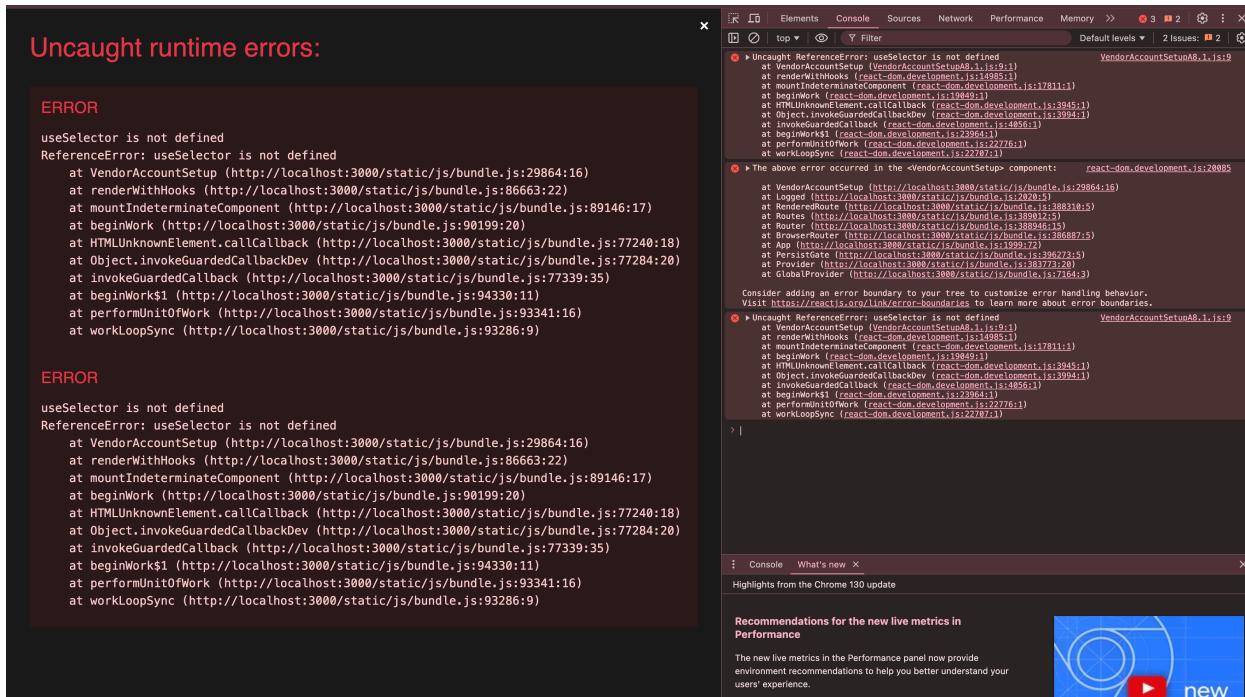
### Expected:

When you type in the search field, a function called **onSearch** should handle the text input

### Actual:

When you type in the search field, you get a "ReferenceError: onSearch is not defined"

## 5: useSelector is not defined



### Description:

When user clicks on 'Next' button system throws uncaught runtime error

Error shows that **useSelector** is not defined in React Redux component. The error is occurring in **VendorAccountSetup** component. Need to import useSelector from React Redux at the top of your VendorAccountSetup component file.

### Preconditions:

Have console inspector open

Go to <http://localhost:3000/findlayer>

### Steps to Reproduce:

1. Click 'Next' button
2. Observe console inspector and error message that appears

### Expected:

**useSelector** hook from Redux should be available to select data from your Redux store  
The component should be able to access Redux state data

### Actual:

"ReferenceError: useSelector is not defined"

Error occurs in VendorAccountSetup component

This is as far as I can go on Onboarding workflow.

The Vendor account setup page is blank after clicking Next, or Skip button on <http://localhost:3000/findlayer> and then closing "ReferenceError: useSelector is not defined".

