

## **Meatblog Testing Document**

### **User Stories Testing:**

#### **New External User Goals:**

*As a new user, I want to be able to view recipes to cook*

- There are 2 prompts/CTA's to direct a new (and also returning) user to view all the website recipes so they can find one to cook. The first and main CTA is in the navigation header under 'Find a Recipe.' This link is visible to all new visitors, as well as signed in visitors, so it will always be there. Additionally, there is a Find a Recipe button in the main homepage image

*As a new user, I want to be able to register for the website and post my own recipes*

- Thanks to 3 'Sign Up' buttons on the homepage (in the main navigation, in the About section and in the large 'Submit Your Own Recipe' bar), directing a user to the sign up function is very clear. As well as these buttons, there is an image at the top of the homepage which also states a user can sign up to post their own recipes.
- With the Sign Up button in the main navigation, the user will be able to click on this button no matter where they are currently on the site.
- There is also a link to 'Sign Up' from the 'Sign In' page should the user end up on this page instead.

#### **Frequent User Goals:**

*As a frequent user, I want to be able to share my own recipes to the website and with the community in an easy way*

- Once logged in via one of the many options above, an 'Add Recipe' item will appear in the main navigation. The Add Recipe page is a simple form-based page which the user can fill out to be able to add their own recipes.
- The form also includes some additional instructions for the user to follow when uploading their recipes to assist them with the process.

*As a frequent user, I want to be able to find new recipes to cook*

- Recipes on the 'Find a Recipe' page have been organised to display the newest recipes first, making it easy for a returning visitor to be able to view new recipes, rather than need to look for them.

*As a frequent user, I want to be able to share recipes that I like to my social network pages*

- Social share buttons have been added to the View Recipe page. This allows the website user to share that particular recipe to social pages, as well as share a link in an email.

#### **Returning External User Goals:**

*As a returning user, I want to be able to edit/update my posted recipes*

- Users can edit their own recipes easily. Once on the 'Find a Recipe' page, the user will be able to see an 'Edit Recipe' button to the right, beneath the 'View Recipe'

button. This will take them to a full Edit Recipe page which looks very similar to the Add Recipe page. Users can replace the information in the form, then click update to update the recipe, or cancel should they change their mind.

- Users will only be able to see an 'Edit Recipe' button once they are logged and it will only be displayed on for recipes they added to the website from their account.

*As a returning user, I want to be able to delete my posted recipes*

- Within the Edit Recipe page there is a large button at the bottom 'Delete Recipe.'
- Once clicked, the recipe will not immediately delete, instead a modal will pop up to ask the user to confirm that do in fact want to delete their recipe. Upon agreeing, the recipe is removed from the website, as well as the database.

### **Site Owner:**

*As the site owner, I want to share my meat-based recipes*

- The website owner is able to post, edit and delete their own recipes in the same way that external users are able to do so.

*As the site owner, I want to drive traffic from this site to upsell from the fresh meat delivery business.*

- The navigation includes a button to 'Order Meat'. Once clicked, the user will be taken, via a new tab, to the fresh meat delivery company MeatBox.
- Additionally, there are 2 further CTA's to show the user of the site where they can order fresh meat – in the main images at the top of the page, and in the about section.
- Thanks to the website being a meat based blog, links to an external meat delivery service fit in to the theme perfectly.

Unless otherwise stated, all below tests were tested in Chrome.

### **Navigation Testing**

Meatblog logo:

- When the Meatblog logo is clicked, the user expects to be taken to the homepage.
  - Desktop: Passed, site acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

Find A Recipe:

- When the Find a Recipe nav item is clicked, the user expects to be taken to the full list of recipes available on the website.
  - Desktop: Passed, site acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

About:

- When the 'About' nav item is clicked, the user expects to be taken to a section providing more information about the website.
- The 'About' nav item should take the user to an anchored section on the homepage with information about the website.
  - Desktop: Passed, the site acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

#### Order Meat:

- When the 'Order Meat' nav item is clicked, the user is expecting to be taken to a page to order fresh meat.
- When the 'Order Meat' nav item is clicked, the site should respond by taking the user to an external website called MeatBox, which should open in a new tab.
  - Desktop: Passed, the site acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

#### Sign Up:

- When the 'Sign up' button is clicked, the user is expected to be taken to a page to sign up/register for the website.
  - Desktop: Passed, the site acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

#### Sign In:

- When the 'Sign In' button is clicked, the user is expected to be taken to a page to sign in to the website.
  - Desktop: Passed, the site acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

#### Sign Out:

- When the 'Sign Out' button is clicked (which is only visible once a user has signed in), the user is expected to signed out of the website.
- Upon doing so, the website should redirect the user to the Homepage page and display a flash message to let the user know they have been signed out of the website.
  - Desktop: Passed, the site acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected
- It was discovered by fellow student and Slack member Mark O'Beirne that a user could copy the Sign Out link, and if they tried to load this link whilst not signed into the website, they received an error. To combat this, I added some code to the function that prevented this error occurring. If a user tried to access this link whilst not signed in they would simply be redirected to the homepage.

- Desktop: Passed, the site acted as expected
- Mobile Passed, site acted as expected
- Tablet Passed, site acted as expected

Signed in user:

- When a user is signed in, the 'Sign Up' and 'Sign In' buttons should be hidden, and replaced with a 'Sign Out' button
  - Desktop: Passed, the site acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

Non Signed in user:

- When a user is not signed in, the 'Sign Up' and 'Sign In' buttons should be visible, and the 'Sign Out' button should not.
  - Desktop: Passed, the site acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

## Homepage Testing

Main image:

- The 'Find a Recipe' button on the main image is expected to take the user to the page to find all recipes, the same page as the Find a Recipe nav item.
  - Desktop: Passed, website acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

'Sign up' image

- The image should be fully clickable and take the user to the Sign Up page
  - Desktop: Passed, website acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

'Fresh Meat' image

- The image should be fully clickable and is expected to take the user to the page to be able to order fresh meat – which is an external website MeatBlog. It should also open in a new tab.
  - Desktop: Passed, website acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

### About Meatblog – Sign In button

- When the 'Sign In' button is clicked in this section, the user expects to be taken to the Sign In page.
  - Desktop: Passed, website acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected
- If a user is already signed in, this button has changed to become 'Add Recipe' and should link to the 'Add Recipe' page.
  - Desktop: Passed, website acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

### Order Fresh Meat – Visit Meatbox Button

- When the Visit Meatbox button is clicked, the user is expecting to be taken to the page to be able to order fresh meat – which is an external website MeatBlog. It should also open in a new tab.
  - Desktop: Passed, website acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

### Submit Your Own Recipe – Sign Up Button

- When the user clicks on the 'Sign Up' button, they are expecting to be taken to the 'Sign Up' page of the website.
  - Desktop: Passed, website acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected
- As this section is visible to users whether they are signed in or not, I decided to add a Sign In button next to the Sign Up button so this section is more relevant for all users – new and returning.
- Additionally, I decided to change the visibility of the buttons. If a user is already signed in, the buttons become 'Add Recipe' and should link to the 'Add Recipe' page.
  - Desktop: Passed, website acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

## Footer Testing

### Instagram

- When the user clicks on 'Instagram' they are expecting to be taken to Instagram, opening in a new window, or open the app on mobile.
  - Desktop: Passed, website acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

## Facebook

- When the user clicks on 'Instagram' they are expecting to be taken to Facebook, opening in a new window, or open the app on mobile.
  - Desktop: Passed, website acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

## Twitter

- When the user clicks on 'Instagram' they are expecting to be taken to Twitter, opening in a new window, or open the app on mobile.
  - Desktop: Passed, website acted as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

## Find A Recipe Page Testing

### Display

- Upon landing on the Find A Recipe page, the user expects to see a page full of recipes for them to choose from.
- The user should be able to view the name of the recipe, a way to view the full recipe, see who posted the recipe, and view an image of the recipe (if the user uploaded one).
  - Desktop: the site behaved as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

### View Recipe Button

- When the 'View Recipe' ('View' on mobile) button is clicked the user expects to be taken to view the full recipe of that item.
  - Desktop: the site behaved as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

### Expandable content

- When a recipe item is clicked on expandable content should appear.
- The content will display just the ingredients needed for that recipe, plus a 'View Recipe' button, which takes the user to view the full recipe details page.
  - Desktop: the site behaved as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected
- Thanks so slack member Mark O'Beirne's suggestion, whilst this section worked correctly, he suggested I stop the accordion expanding if the 'View Recipe' or 'Edit Recipe' button's to the right of the accordion are clicked. I added some additional jQuery code to implement this.

- Desktop: the site behaved as expected
- Mobile Passed, site acted as expected
- Tablet Passed, site acted as expected

Edit Recipe Button (only visible when a user is logged in)

- When a user clicks the 'Edit Recipe' ('Edit' on mobile) button, they expect to be taken to a page whereby they are able to update the details of their own posted recipe.
  - Desktop: the site behaved as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

## Add Recipe page

Select Meal Category – Drop down:

- When this section is clicked the website is expected to respond by displaying a drop down menu of the different meal categories the user can select.
- The user is then able to select the category of their recipe, which will then show in this section at the top.
  - Desktop: the site behaved as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected
- If a category is not selected and the page and recipe is submitted anyway (subject to the rest of the fields being completed correctly).
  - Desktop: the site behaved as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

Recipe name

- The user is expected to write the name of the recipe they want here. If the user does not add a recipe name, the line will appear red. If the user attempts to submit the form without a recipe name provided in the field, an error will display.
  - Passed: the site behaved as expected

Recipe Ingredients

- The user is expected to write in all the ingredients in the recipe, separating them on a new line for each item, as explained in the helper text. If the user does not add recipe ingredients, the line will appear red. If the user attempts to submit the form without a recipe name provided in the field, an error will display.
  - Error, page submitted without ingredients
    - Error fixed by adding a 'required' class to the recipe ingredients input field.
  - 2<sup>nd</sup> test: Passed: the site behaved as expected

### Recipe Method

- The user is expected to write in the method of cooking their recipe in the Recipe Method section. As explained by the helper text, the user is expected to input each step in the recipe on a new line.
  - Passed: site behaved as expected

### Add an image

- The user is requested to enter the url of an image into the Add an Image section. As explained by the helper text, the url must be full, including https://. If a user does not input a url, or leaves this section blank completely, the form will not submit and an error will display.
  - Passed: site behaved as expected

### Submit Recipe button

- When a user clicked on the Submit Recipe button (and provided all fields are filled out correctly) the user expects to be taken to the Find a Recipe page where they will be able to see their recipe, and they are also expecting to see a success message so they know the recipe has been submitted.
- Once the submit button is pressed on a completed form, the user will be redirected to the Find a Recipe page and a success message will appear in the form of a flash message at the top of the page.
  - Passed: site behaved as expected

### Edit Recipe Page

#### Change Category

- Upon clicking the category drop down, the user is expected to be able to select a new category for their recipe, and the recipe will be updated upon clicking 'Update Recipe'
- A flash message will appear at the top of the page to notify the user that the update has been successful.
  - Passed: website responded as expected

#### Update recipe name

- When the user types in a new recipe name into the Recipe Name section and submits the form, they expect the recipe name to be updated and to receive a flash notification to confirm the change has been successful.
- If the user leaves this section blank, the update will not be successful, and an error will appear
  - Passed: website responded as expected

#### Update Recipe Ingredients

- When the user makes changes to the recipe method and submits the form, they expect the recipe method to be updated and to receive a flash notification to confirm the change has been successful.



- If the user leaves this section blank, the update will not be successful, and an error will appear
  - Passed: website responded as expected

#### Update Recipe Method

- When the user makes changes to the recipe ingredients and submits the form, they expect the recipe ingredients to be updated and to receive a flash notification to confirm the change has been successful.
- If the user leaves this section blank, the update will not be successful, and an error will appear
  - Passed: website responded as expected

#### Update Recipe Image

- When the user makes changes to the image URL and submits the form, they expect the recipe ingredients to be updated and to receive a flash notification to confirm the change has been successful.
- If the user leaves this section blank, the update will not be successful, and an error will appear
  - Passed: website responded as expected

#### Delete Recipe Button

- Upon pressing the Delete Recipe Button, the user expects their recipe to be deleted from the website
- An additional deletion step has been added in the form of a pop up modal to confirm the user wants to delete their recipe and has not clicked the button by accident. By clicking Delete Recipe on the modal, the user is taken back to the Find a Recipe page and a flash message appears at the top of the page to inform the user that the recipe has been successfully deleted.
- The modal also has a Cancel Button which takes the user back to the Edit Recipe page and the recipe has not been delete.
  - Passed: website responds as expected

#### Sign Up page testing

- On this page the user expected to be able to sign up for a new account to log into the website and post their recipes.
- The user is supplied with instructions to help them sign up to the website. They will not be able to sign up and submit the form if these instructions are not followed, or if they leave a field empty. An error will appear should they try to submit the form incorrectly.
- If a user tries to sign up with a username which has been taken, the page will reload and a flash message will display to inform them of the error.

- One the user has successfully signed up, they will be redirected to the homepage with a flash message stating they have successfully signed up and can post their own recipes.
- For all of the above:
  - Desktop: the site behaved as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

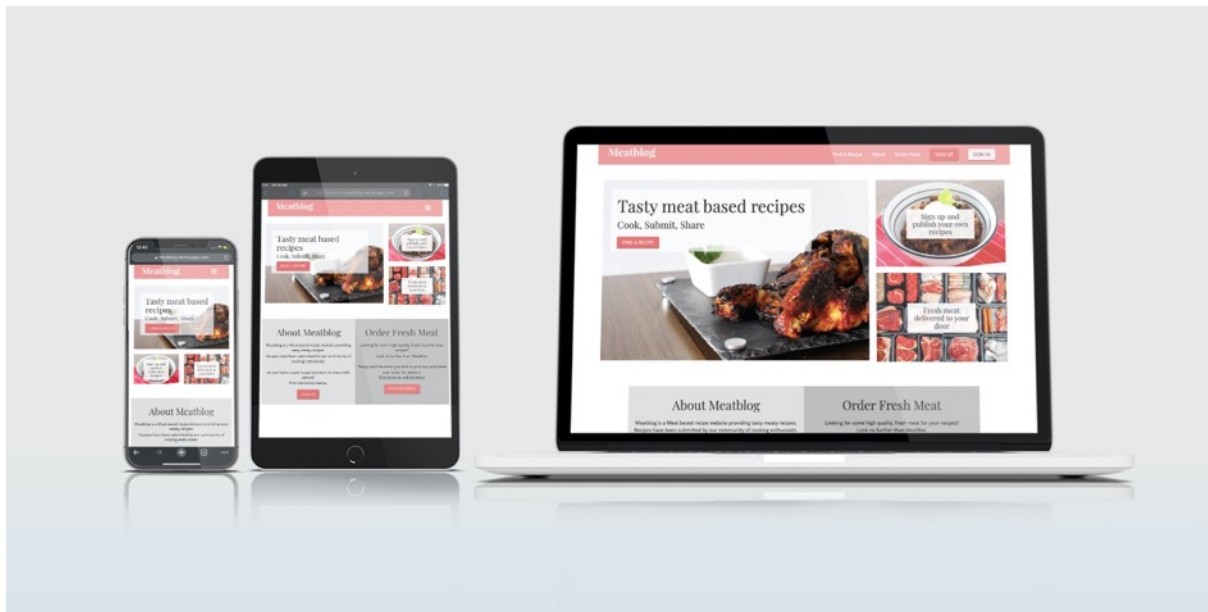
### **Sign In page testing**

- On this page the user expected to be able to sign in to the website and post their recipes.
- The user will not be able to sign into the site if they do not correctly fill out the username or password details with their original registration details.
- Missing one of the fields will display an error
- If the user does not enter their username or password correctly, an error will appear in the way of a flash message at the top of the screen and the page will be refreshed so they can enter their details again.
- If a user tries to sign up with a username that already exists in the database, a flash message will be displayed to inform the user of this error.
- If a user successfully signs into the website, they will be redirected to the homepage and a flash message will appear at the top of the screen to let them know they have signed in successfully.
- If a user lands on this page accidentally instead of the Sign Up page, there is a link at the bottom of the form to direct users to the correct page. This link should take the user to the Sign Up page.
  - Desktop: the site behaved as expected
  - Mobile Passed, site acted as expected
  - Tablet Passed, site acted as expected

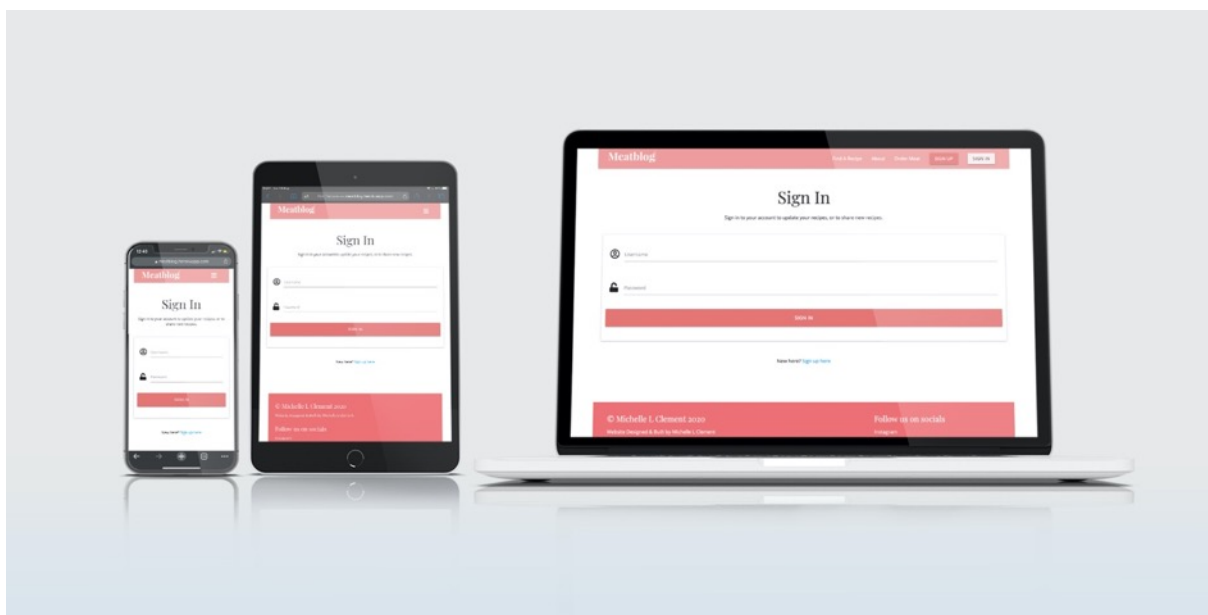
### **Responsive Testing – Screen shots from devices, not Dev Tools**

Testing the responsive view of each page on desktop (Apple Macbook Air), mobile (iPhone 8 Plus) and tablet (iPad)

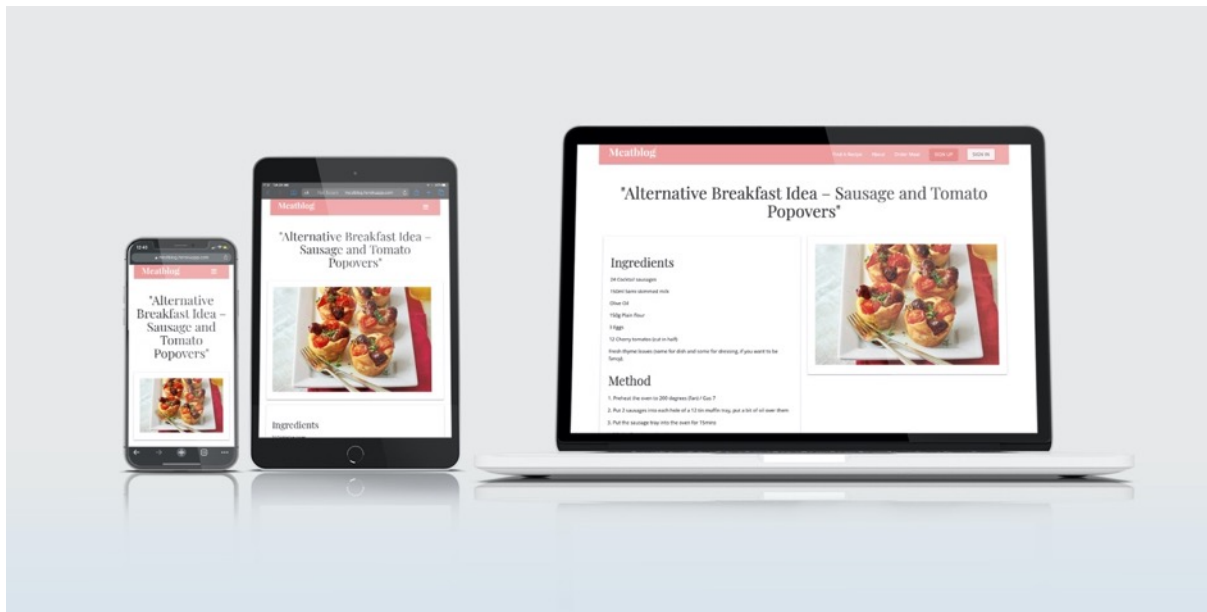
- Homepage: Passed: Website responded as expected



- Find A Recipe page: Passed: Website responded as expected
- Edit Recipe page: Passed: Website responded as expected
- Sign In page: Passed: Website responded as expected



- Sign Up page: Passed: Website responded as expected
- View Recipe page: Passed: Website responded as expected



### Firefox Browser testing:

All above website testing was repeated on desktop in Firefox.

- Passed. Website responded as expected

### Safari Browser testing:

All above website testing was repeated on desktop in Firefox.

- Passed. Website responded as expected

### PEP8 Compliance

I used the website <http://pep8online.com/> to check my app.py files complied with the PEP8 requirements.

1<sup>st</sup> test of code: 2 Errors occurred:

- Line 2: line too long
  - I corrected this error by adding a backslash part way through the line and moving part of the text to the line below, making sure to add the correct indentation.
- Line 136: line too long
  - I corrected this error by splitting up the long line of text across a further two lines making sure to add the correct indentation.

2<sup>nd</sup> test: passed with no errors

### W3C HTML Validator

I used the W3C HTML Validator (<https://validator.w3.org>) to check the validity of my HTML code on all 8 pages.

- Index.html
  - Passed with no errors
- Add\_recipe.html
  - Passed with no errors
- Base.html
- Edit\_recipe.html
  - Error: Bad value for attribute 'value' on element 'input': Line feed not allowed.
    - I believe this error is being displayed due to the Edit Recipe page being pre-filled with content in order to edit it. The value being pulled into the value attribute comes from the recipe database which is then formatted to display on the view\_recipe page with splitlines. When I paste the html code directly into the validator, this error does not appear.
- Recipes.html (Find a Recipe page)
  - Passed with no errors
- Sign\_in.html
  - Passed with no errors
- Sign\_up.html
  - Passed with no errors
- View\_recipe.html
  - Passed with no errors

### **W3C CSS Validator**

I used the W3C CSS Validator (<https://jigsaw.w3.org/css-validator/>) to check the validity of my CSS code.

- 1<sup>st</sup> test: Passed with no errors.