2.1.10 Merge the Feature into the Development Environment

In this final step, we'll integrate the new feature into the `develop` branch.

First, let's make `develop` the active branch:

```
git checkout develop
```

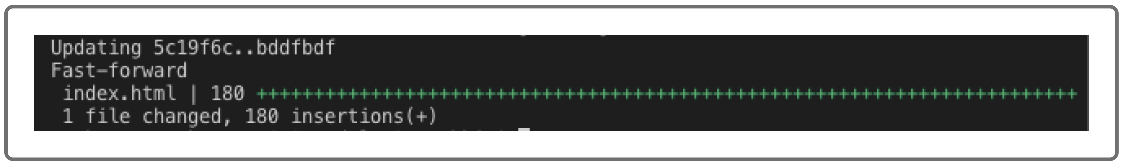
In VS Code, let's look at the hero section in the `index.html` file. Note that the new HTML we just added is not visible here. It only exists in the feature branch, `feature/hero-text`. To add the new code to the `develop` branch, we need to perform a `merge` operation.

Let's do that now:

```
git merge feature/hero-text
```

SHOW PRO TIP

Upon successful merging, the terminal should look like this:



```
Updating 5c19f6c..bddfbdf
Fast-forward
 index.html | 180 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 1 file changed, 180 insertions(+)
```

This should look similar to the `git pull` operation we executed earlier in the lesson because we also performed a `git merge` operation then as well.

Excellent. Now `develop` is updated with our new HTML and has remained a stable codebase throughout. Let's move on to the second issue and tackle the contact form feature.

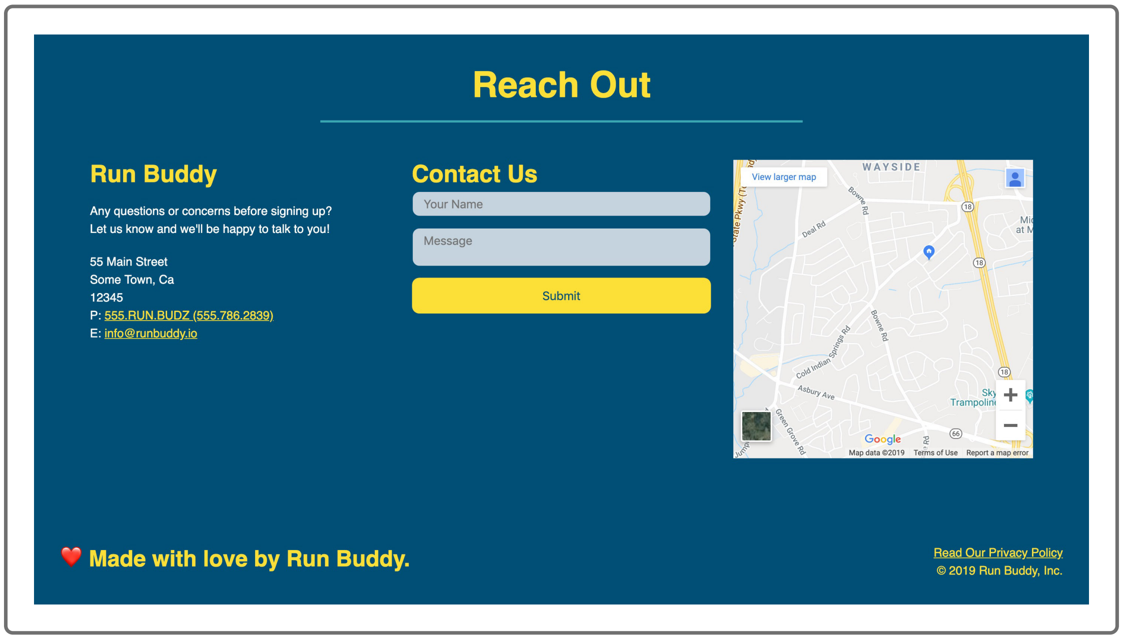
We'll follow the same procedure that we used for the `feature/hero-text` issue:

1. Set up the feature development environment, which you can do now:

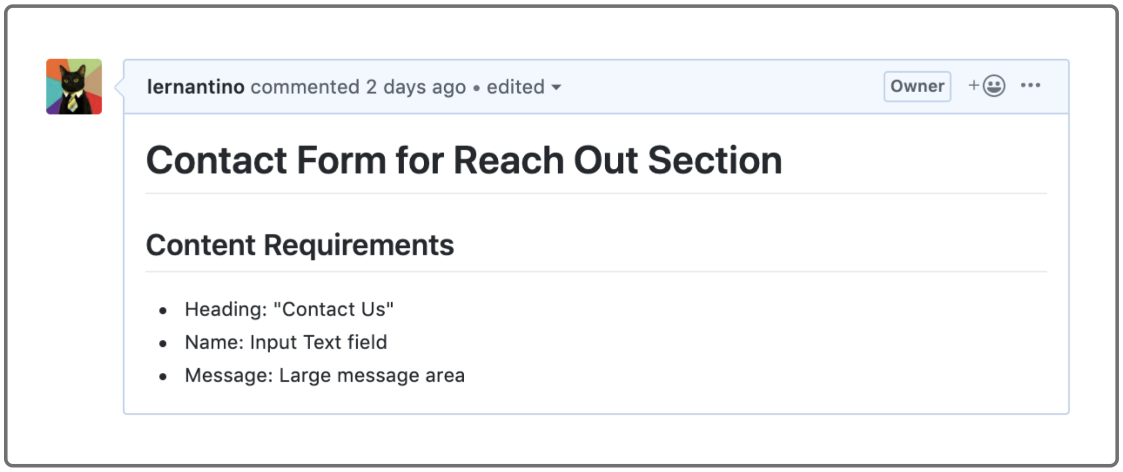
```
git checkout -b feature/contact-form
```

2. Add the HTML according to the requirements.
3. Verify that changes were successful in the browser.
4. Push up the feature branch to the remote repo.
5. Merge the `feature/contact-form` into the `develop` branch.
6. Push the `develop` branch to the repo to update the shared version with your new feature.

Here is the design team's mock-up:



And here is the corresponding issue with the requirements:



lernantino commented 2 days ago • edited

Owner

Contact Form for Reach Out Section

Content Requirements

- Heading: "Contact Us"
- Name: Input Text field
- Message: Large message area

Notice the large text input field for the message. This is a new HTML element that will allow multiple lines to be entered. This element is called `<textarea>`. This element has quite a few attributes that enable configuration parameters such as the size of the rendered input field, maximum character length, and whether it is a required input in order to submit. For more details, see the **MDN web docs on the textarea element (https://developer.mozilla.org/en-US/docs/Web/HTML/Element/textarea)**.

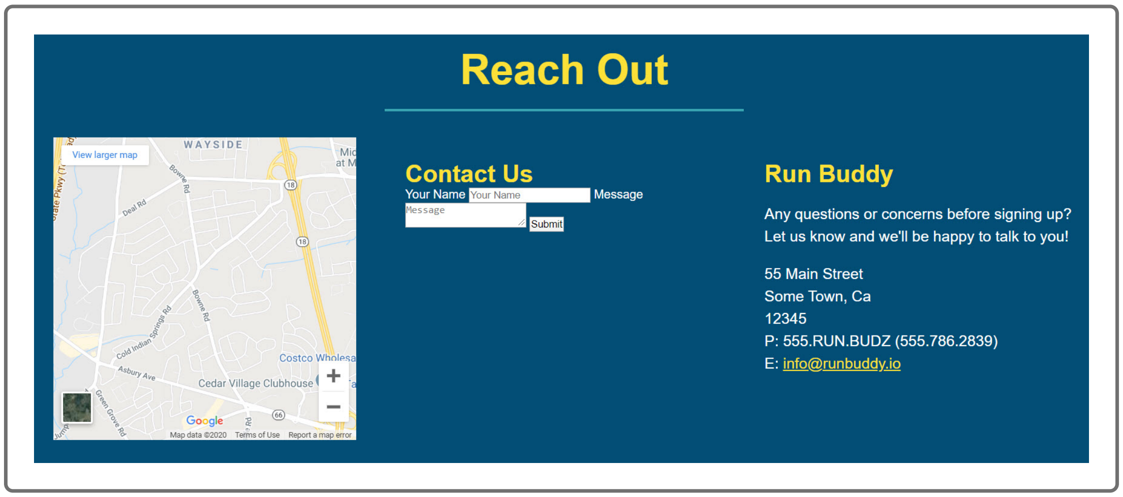
Here is the HTML that you'll add to the `<div class="contact-info">` element in between the existing `<iframe>` and `<div>` elements:

```
<div class="contact-form">
  <h3>Contact Us</h3>
  <form>
    <label for="contact-name">Your Name</label>
    <input type="text" id="contact-name" placeholder="Your Name" />

    <label for="contact-message">Message</label>
    <textarea id="contact-message" placeholder="Message"></textarea>

    <button type="submit">Submit</button>
  </form>
</div>
```

Now let's check to see if our contact form has been correctly added. Open the `index.html` file in the browser. You should see the following:



Reach Out

Contact Us
Your Name [Your Name] Message
[Message]
[Submit]

Run Buddy

Any questions or concerns before signing up? Let us know and we'll be happy to talk to you!

55 Main Street
Some Town, Ca
12345
P: 555.RUN.BUDZ (555.786.2839)
E: info@runbuddy.io

Congrats! The HTML has been successfully added. We'll format and re-order this later in the CSS lesson.

Once the HTML has been inserted into the Reach Out section, add, commit, and push the `feature/contact-form` branch into the remote repo:

```
git add .
git commit -m "add contact form"
git push origin feature/contact-form
```

Now let's merge this feature into the `develop` branch using the following commands:

```
git checkout develop
git merge feature/contact-form
```

IMPORTANT

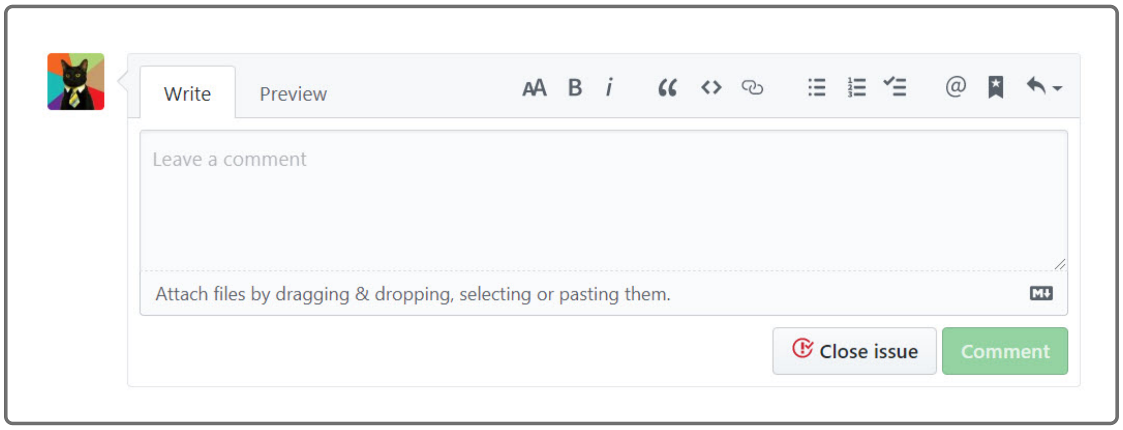
> Don't forget to check which branch is active before merging the feature into the develop branch!

Finally, let's update the remote `develop` branch with the new feature by using the following command:

```
git push origin develop
```

Now that our two feature branches have been completed and merged back into the `develop` branch, what's left to do? We need to mark the related GitHub issues as resolved! This will remove the issues from our to-do list.

Go to the Issues pages in the GitHub repo, open both of the related issues (the contact form and hero content), and click the Close Issue button, as seen below:


Write
Preview
Leave a comment
Attach files by dragging & dropping, selecting or pasting them.
Close issue
Comment

Before we wrap up this lesson, take a moment to work through the following knowledge check: