**What Is Recursive Thinking?**

Recursive Thinking (RT) is a collective of people dedicated to career development in the technology space. In order to learn from others, members needed a website to showcase their skillsets to establish connections in and out of RT meetings. As a way to build skills, developers, designers and technical writers work on the website together. By participating, you will provide RT collaborators with:  
• A schedule of upcoming meetings  
• A place to provide lesson requests and vote for lessons  
• Documentation of interview questions and answers  
• A professional profile  
  
If you’d like to work on the website, keep reading our user guide below for information about how to get started.  
  
**Getting Started**

Before you begin, you will need to have a few tools to work on the site. Make sure you have the ones listed below.

**Integrated Development Environment (IDE):** Depending on your preference, you download one of these IDE options.

* Download VSCode at [https://code.visualstudio.com](https://code.visualstudio.com/).
* Download Brackets at [http://brackets.io](http://brackets.io/).
* Download SublimeText at [https://www.sublimetext.com](https://www.sublimetext.com/).

**Download Node.js** for your version of OS at <https://nodejs.org/en/>.

**Download Git** for your version of OS at <https://git-scm.com/downloads>.

Note: If you’re using Windows, your computer will automatically install Git Bash Terminal.

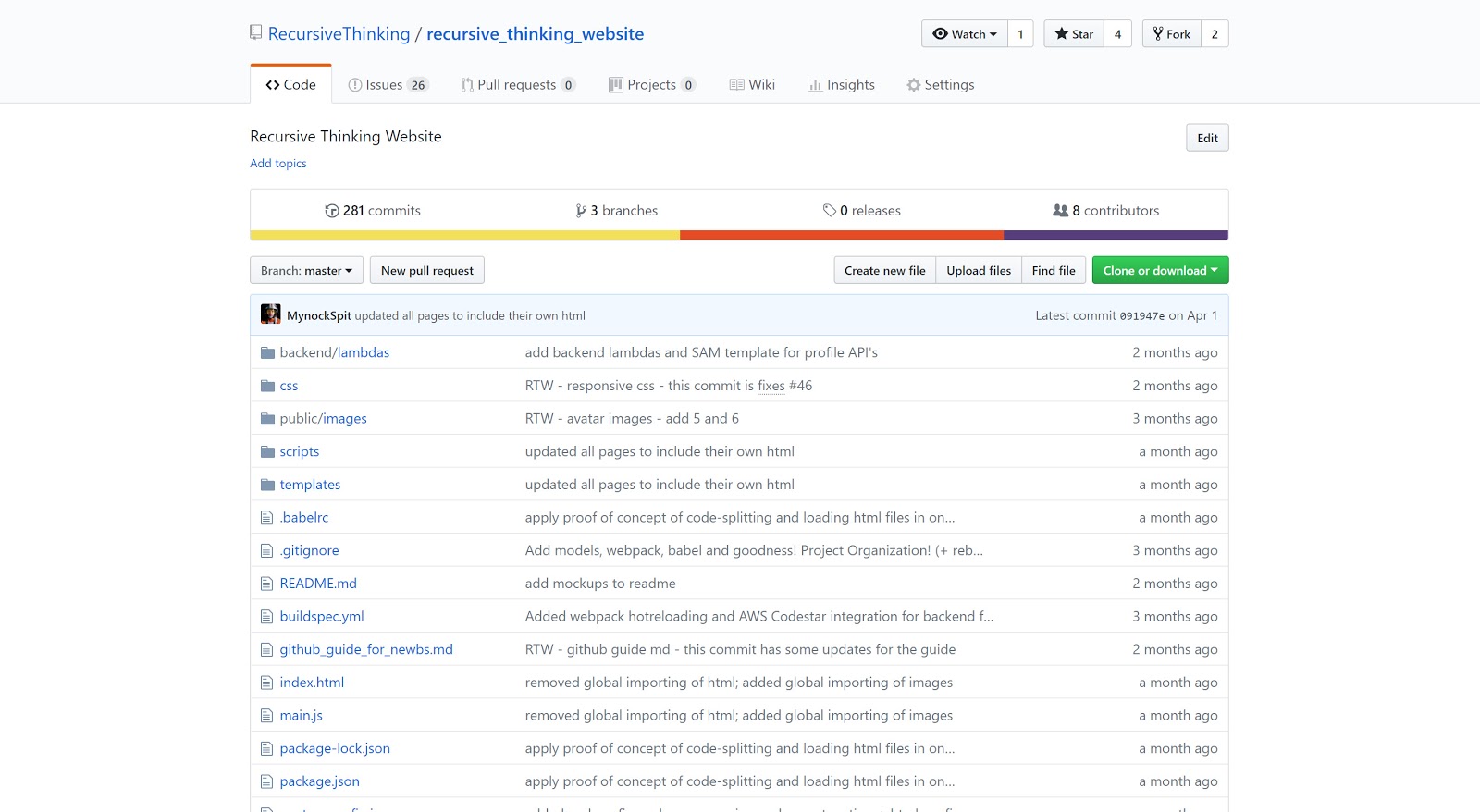
**Getting the Project from Git**

To work on the RT website project, you’ll need to transfer the project files from Git to your computer. This is done through a Git clone.

**Git clone**

1. Before you begin, make sure you’re in the directory/folder where you want the files to go.
2. Perform a Git clone using the following command:

git clone <https://github.com/RecursiveThinking/recursive_thinking_website.git>



1. If cloning, you need to navigate to the root folder (the folder where you want to store your project directory) in your terminal.
2. Install the required node modules to run webpack using the following command: npm install

Note: Make sure you are inside the project (Recursive Thinking website)

directory.

1. Learn more about npm install [here](https://docs.npmjs.com/cli/install).
2. webpack is what we use to compile all of our files into a single page application. You can learn more about webpack [here](https://webpack.js.org/).
3. Run the webpack dev server.
4. Kick off the build sequence with the following command to open a tab in your browser that will show the website: npm run dev-server
5. Now the project should be up and running on your local host.
6. Check the mockups.
7. In order to see the mockups, you’ll need to be added to the Zeplin site. Email [haileyskelcher@hotmail.com](mailto:haileyskelcher@hotmail.com) to gain access.
8. Find the mockups at <https://app.zeplin.io/project/5a35b3271068e9760ef79fc6>

**Creating Changes and Additions**

Now that you’re all set up, let’s take a look at how to do work on the site.

**Committing changes**

If you’ve done some work on the website and you want to push your changes to the repository, perform the following commands.

First time users: If it’s your first time working on the project, you can make sure you’re connected to the project’s repository by running the following command: git remote get-url origin

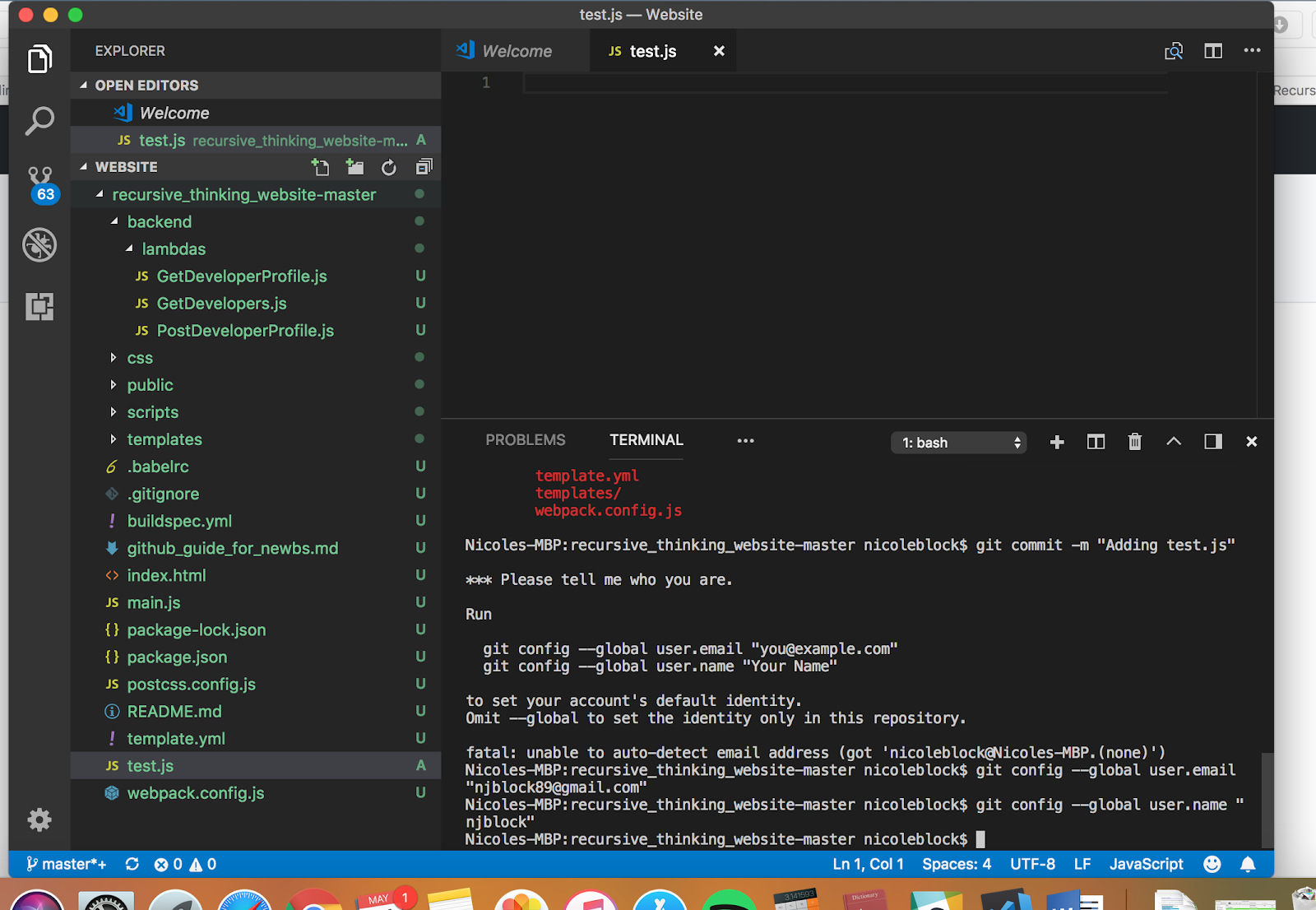
This should produce the address for the RT site. If it doesn’t, run the following command:

git remote add origin <https://github.com/RecursiveThinking/recursive_thinking_website.git>

1. Initialize a Git repository in your project folder using the following command: git init
2. Add untracked files.
3. If you’ve created new files or made changes to existing ones, you’ll need to add your changes.
4. First, use the command git status to make sure that you are on the master branch. This will also allow you to see what files have not yet been added, as they will appear in red.
5. To add files, run the following command: git add [your file name]

For example, if you created a file named test.js, you would add the file by running: git add test.js

Once added, they will appear green when you run the git status command. See image below.



1. When you’re ready to commit your changes, you can run the commit command, which includes a mandatory message detailing what action you have taken:

git commit -m “” (Inside of the quotation marks, you will type your

message details.)

For example, once you are ready to commit test.js, you would run:

git commit -m “adding test.js”

1. Push up your changes.
2. To make the final step in pushing your changes to Git, where they will show up in your commit history and in the project directory on GitHub, run the following command: git push origin master
   * 1. Errors? Like:
        1. Whatever the BS error is for merge Conflicts
           1. deal with these BS Errors
        2. Other Errors
           1. merge conflict/git stash
           2. ???
3. Front End
   1. Templater
4. Back End
   1. YAML Templating