-------------------------------------------------------------------------------

**TERMS AND THINGS YOU NEED TO KNOW**

-------------------------------------------------------------------------------

*These terms will help you understand what is going on in the next few pages.*

**Jekyll -** Jekyll is an open source, simple, blog-aware, static site generator. It takes a template directory containing raw text files in various formats, runs it through a converter (like Markdown) and a Liquid renderer, and spits out a complete, ready-to-publish static website suitable for serving with your favorite web server.

**Homebrew -** Software package management system that simplifies installation of software on Apple MacOS operating system. It allows you to install Ruby.

**Ruby -** Open-source programming language. It is the language used to write Jekyll.

**Terminal -** terminal allows you to use your command line on your mac. *The easiest way to find it if it is not on your dashboard is to search for “terminal” on your whole computer.*

**-------------------**

**FIRST STEPS**

**-------------------**

*Connecting to the MBON server and navigating to the website files.*

File-> go -> connect to server->smb://files.msi.ucsb.edu/sbcbon

*You will need to log into the server using your credentials.*

**Navigate to ->** /volumes/sbcbon/internal/bon\_website/bootstrap

*This is where the local copy of website lives, anything you change here will NOT be live on the actual internet facing website until you copy all the files to the live folder. We will get there.*

--------------------------------------------------------------------------

**INSTALLING JEKYLL AND THE JEKYLL PROTOCOL**

--------------------------------------------------------------------------

*Running the Jekyll protocol and installing jekyll on your computer*

**Find** the Jekyll protocol under Bon\_website *it has instructions for installing Jekyll.*

Open terminal, copy paste code from the jekyll\_protocals and run.

***Installing jekyll:***

1. **Open terminal**

*After opening the terminal you will want to install homebrew following the steps below, simply copy/paste the non-bold text into the terminal and hit enter (if you get stuck in this section Margaret was a great help to me!):*

**2) Install homebrew:**

/usr/bin/ruby -e "$(curl -fsSL <https://raw.githubusercontent.com/Homebrew/install/master/install>)"

**3) Update homebrew:**

brew update

**4) Install ruby**

brew install rbenv

brew install ruby-build

**enter** “sudo gem install jekyll”

*At this point you should have all the software necessary for using Jekyll installed on your computer and ready to work on the website.*

-------------------------------------------------------------------------------

**Initiate server (for local web development):**

-------------------------------------------------------------------------------

*How to log into the local website and necessary jekyll commands.*

**Connect to sbcbon fileshare** (smb://files.msi.ucsb.edu/sbcbon)

**Open Terminal**

*cd to site directory. This is basically where you are telling the terminal to work in the website folder. This allows you to make your changes with Jekyll and build the website.*

**In terminal type->** cd /Volumes/sbcbon/internal/bon\_website/bootstrap

Now the terminal is in bootstrap and when you run commands in jekyll it will make changes in the bon\_website folder.

--------------------------------------------------------------

**IMPORTANT JEKYLL FUNCTIONS**

--------------------------------------------------------------

*These are common commands you will use when working with the website. Honestly these are basically the only things I type into the command line on terminal.*

"**jekyll build**" -- takes all files in current working directory [wd] and compiles them to "wd/\_site/"

**"jekyll serve"** -- locally hosts “[wd]/\_site" for viewing at "<http://127.0.0.1:4000/>

**"jekyll serve -w"** -- watches for changes made in working directory, recompiles site whenever changes are made, locally hosts updated site for viewing (see above). Very useful for development. *Jekyll serve allows you to see the local website files you have been working on, on a web browser without making them live. Just put* <http://127.0.0.1:4000/> *into your web bar and you will see your website changes :) This is the best tool for development.*

--------------------------------

**SUBLIME TEXT**

--------------------------------

*Sublime text is a text editor used to edit HTML and other code languages. Personally I think it is the easiest to use and it is free. Once you change the bottom right code language selector to HTML the software can predict what tags you are going to use and it writes all the code in different colors making each line of code easily to distinguish from the others.*

*It can be downloaded HERE ->* [*https://www.sublimetext.com/*](https://www.sublimetext.com/)

*Sublime text provides a free trial and I have found that if it asks you to download the upgrade or Buy it you can just hit no/cancel and the software still works.*

--------------------------------

**JEKYLL TID-BITS**

--------------------------------

*In Terminal with jekyll installed you can use:*

*Cd change directory:*  /Volumes/sbcbon/internal/bon\_website/bootstrap *(change it if you want to change where you are working)*

**\_site** folder is build by Jekyll and contains the website basically. When it is in the bon\_website/bootstrap folder it is still not live however.

-----------------------------------------------------------------------------------------------------------------

**WORKING LOCALLY TO CHANGE THINGS WITHOUT CHANGING THE LIVE SITE**

-----------------------------------------------------------------------------------------------------------------

*Follow the steps below to start working on the website locally. Remember working locally does NOT change anything in the live website, unless you put the changes in the live folder manually. Also remember if you do not like changes made be sure to change them back because they could show up later on if you forget about them.*

1. **Open terminal**
2. **Type ->** cd (insert path to local copy) If you do not know the path to the local copy **right click** on folder **click** on get info and in text under the header “where” is your file pathway **copy/paste** this text after cd. You will have to write the actual folder name in the directory after writing a /. Unless you change where the website lives you will always direct jeckyll to cd /Volumes/sbcbon/internal/bon\_website/bootstrap when developing the site.
3. Once you are connected linked to the file you want to work in you should run the two following jekyll commands:
   1. **Jekyll build-**> builds site! :)
   2. **Jekyll serve -w** -> builds a copy and makes a psudo serve online and will make changes it you change them on the code.
4. The terminal box will give you a server address-> copy paste server address into chrom which give you a website.

*If you click on anything on the fake site it will go to the real website*

*If you want to go to another page on the fake site type the page name in the web browser for example*

/gallery

/home

/research/imagery *(this is because you have two layers of files before you reach an index.html files)*

*NOTE\*\*\* anything with an underscore is a jekyll behind the scenes thing and won't be displayed on the actual site. As a general rule, DO NOT WORK in ANY folder that has an underscore in front of the folder name.*

Go and make all your changes in folders (DON’T MAKE IT IN \_SITE!!!)

------------------------------------------

**MAKING THE CHANGES LIVE**

------------------------------------------

*Once you are fully satisfied with the changes you have made on the website you can upload the new \_site folder to the public\_html folder and make the changes live on the website.*

*To get the changes to the website do the following:*

1. ***Navigate to the folder:*** finder/connect to server ->smb://files.msi.ucsb.edu/sbcbon-web
2. **Click ->** on the “public\_html”folder
3. **Navigate ->** in a separate finder window navigate to the Volumes/sbcbon/internal/bon\_website/bootstrap/\_site folder and open the \_site folder.
4. **Copy/Paste ->** everything inside the “\_site” folder into the **public\_html** folder
5. **Click ->**  “yes replace all”

*The site folder from your local copy will copy over to your Public\_html folder. Be sure to check the live website sbc.marinebon.org for all the changes you made. If there are changes that did not transfer over or not get saved it’s always best to go back to the local copy and try again. Sometimes it takes copy/pasting into the pulbic\_html folder several times before all the changes are adopted. NEVER work in the public\_html folder.*

*Anything in this folder will be reflected in the website*

Get a log in for LI how the files link to the IO server from GIT hub and how they work with Iframe. sbcbon2017