# Instructions for working on your web site

This assumes you currently have a working web site (called ‘myapp’ located under csi185website)

If you don’t, do the following.

1. In a terminal, type “git clone https://github.com/mstoth/csi185website.git”
2. Verify you can run the functional tests.
   1. In a terminal, go to the directory called csi185website (You should see it in the directory you typed the clone command)
   2. Run the server by typing “python3 manage.py runserver” and minimize the terminal so it’s out of your way.
   3. In another terminal go to the same directory and run your functional tests by typing “python3 functional\_tests.py” [[1]](#footnote--1)

At this point, you have a working web site. It’s called “myapp” and it’s located under csi185website.

If you now run a browser and type in the URL “localhost:8000/index.html” you should see a computer. When you click on the keyboard, you should be taken to another page showing a teletype. Go ahead and try it. Depending on your browser settings, you may or may not see the title of the web site, “The Title of my Web Site”

## Making your first failure

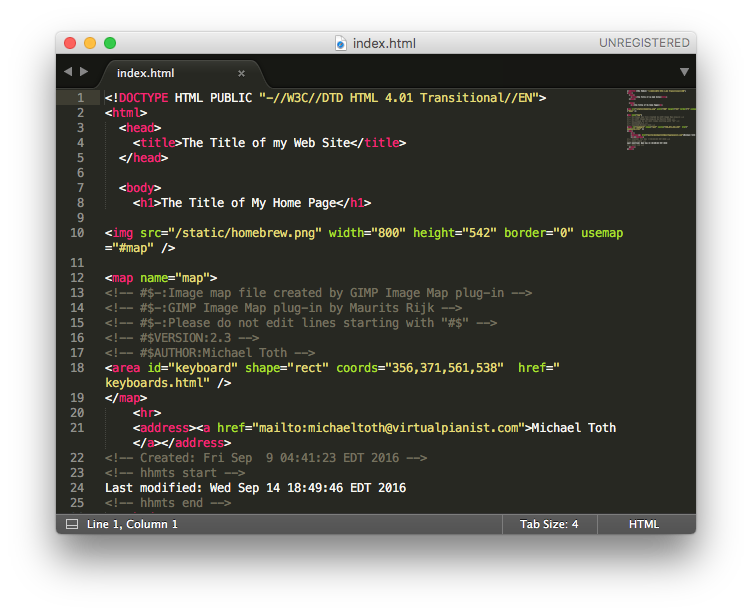
Now we will create our first failure. Using Idle, edit the functional\_tests.py file and change the title of your web site to something you want. Change the following line,

self.assertIn('The Title of my Web Site',self.browser.title)

so it has the title you want (we did this in class but if you weren’t there, you need to do it.)

Once you have done that, re-run your functional tests and verify that you have a failure. [[2]](#footnote-0)

Now we will fix this failure by editing index.html. It’s located under the templates directory under myapp. Use Sublime to edit it.



Change the text between <title> and </title> to match your title you put in functional\_tests.py, save it, then re-run your functional tests. You should now have a passing test again. *Talk to me if you don’t.*

Now the next failure will be the heading. Change the following line in functional\_tests.py

self.assertIn("The Title of My Home Page",h.text)

Change “The Title of My Home Page” to be what you want. Then save it and re-run your functional tests.[[3]](#footnote-1)

Now change index.html to match.

<h1>The Title of My Home Page</h1>

becomes

<h1>The Beginning</h1>

in my case. We are now back to GREEN. This was my goal last Monday.

## Changing the Image

The image is defined by an img tag.[[4]](#footnote-2) In the index.html file, it appears like this.

<img src="/static/homebrew.png" width="800" height="542" border="0" usemap="#map" />

All your images will be stored in the static directory (under myapp) and they will be referenced like ‘/static/<filename>’ as in the example, src="/static/homebrew.png". This is why the HTML class under Lynda is a good thing to do. It explains all this stuff.

After you have decided which image to show on your home page, change your functional tests.

m=self.browser.find\_element\_by\_tag\_name('img')

self.assertIn('homebrew.png',m.get\_attribute('src'))

Change homebrew.jpg to the name of your image file you want to use.

Rerun your functional test and verify you get an error. [[5]](#footnote-3)

Now copy your image file into the static directory and change your index.html file to have the correct image name.

You should now be back in the GREEN condition and when you open localhost:8000/index.html in your browser you should see your picture.

Try it out.

Now that you have changed the image, your mapping is no longer valid. Delete the following lines from your index.html file.

<map name="map">

<!-- #$-:Image map file created by GIMP Image Map plug-in -->

<!-- #$-:GIMP Image Map plug-in by Maurits Rijk -->

<!-- #$-:Please do not edit lines starting with "#$" -->

<!-- #$VERSION:2.3 -->

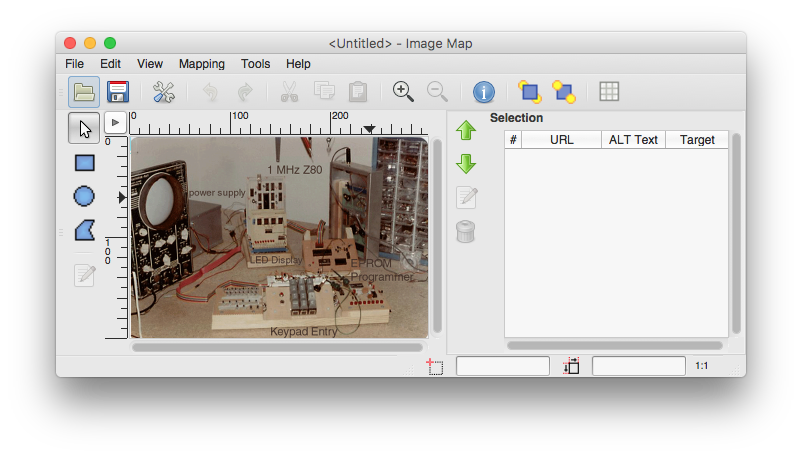
<!-- #$AUTHOR:Michael Toth -->

<area id="keyboard" shape="rect" coords="356,371,561,538" href="keyboards.html" />

</map>

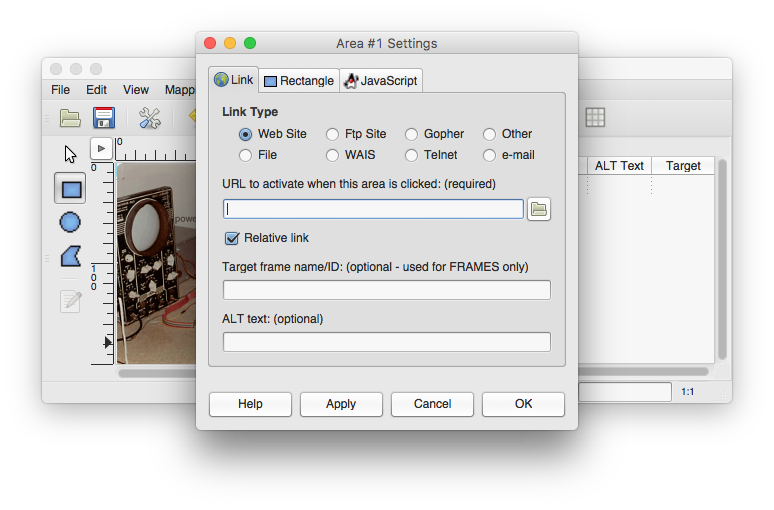
## Mapping an Area of the Picture Using Gimp

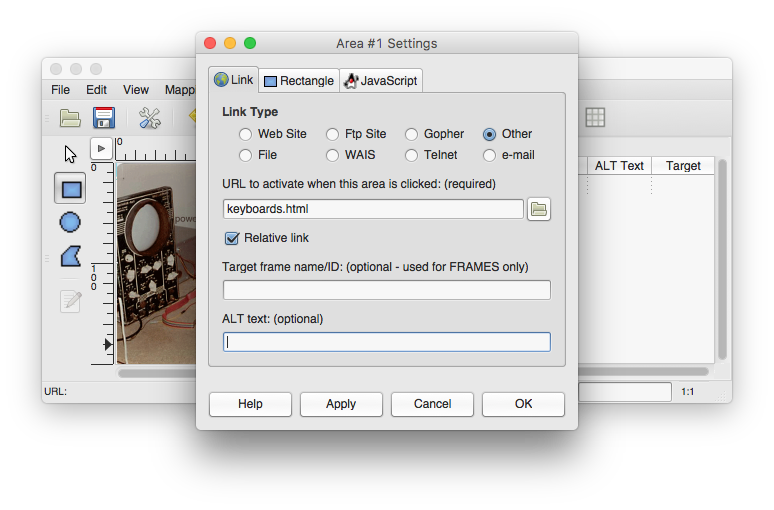
Run Gimp and open your image. Select Filters->Web->Image Map… in the Gimp menus. You should see a new window appear like this.



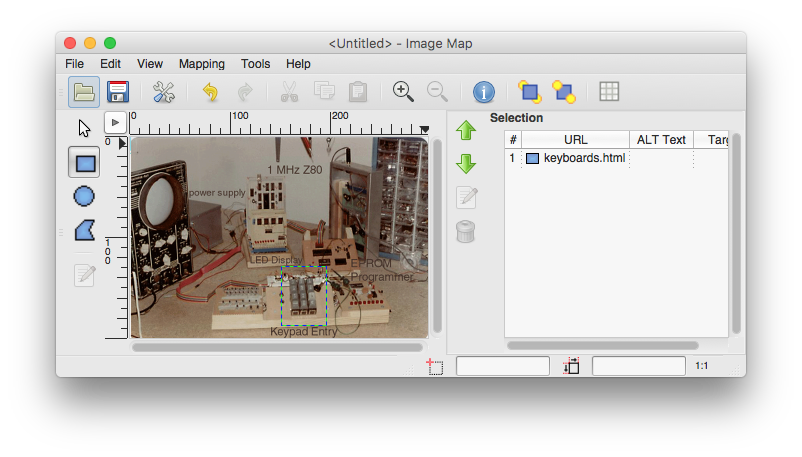
With your picture, not mine of course. Pick a shape on the left and draw the shape on the image where you want it to react to a mouse click.

You should see another window appear like this.



Type in the URL you want to go to when you click on the area. This will be a new page on your web site. You can name it anything you want. Just put the name of the file here, nothing in front of it. In my case the file is called keyboards.html and my area is around the keyboard of the homebrew computer.

Click Apply then OK.

You should see your area on the right hand side of the window.

Select File->Save and navigate to your templates directory. Save the file as <something>.map where <something> means something to you. No spaces. You now have a map file in your templates directory. In my case it looks like this.

<img src="[homebrew2] (imported)" width="300" height="203" border="0" usemap="#map" />

<map name="map">

<!-- #$-:Image map file created by GIMP Image Map plug-in -->

<!-- #$-:GIMP Image Map plug-in by Maurits Rijk -->

<!-- #$-:Please do not edit lines starting with "#$" -->

<!-- #$VERSION:2.3 -->

<!-- #$AUTHOR:Michael Toth -->

<area shape="rect" coords="151,129,196,188" href="keyboards.html" />

</map>

1. You should see something like this

   hello-mac:csi185website michaeltoth$ python3 functional\_tests.py

   .

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

   Ran 1 test in 2.581s

   OK

   hello-mac:csi185website michaeltoth$ [↑](#footnote-ref--1)
2. My failure looks like this

   hello-mac:csi185website michaeltoth$ python3 functional\_tests.py

   F

   ======================================================================

   FAIL: test\_home\_page (\_\_main\_\_.NewVisitorTest)

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

   Traceback (most recent call last):

   File "functional\_tests.py", line 41, in test\_home\_page

   self.assertIn('Computer History',self.browser.title)

   AssertionError: 'Computer History' not found in 'The Title of my Web Site'

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

   Ran 1 test in 3.322s

   FAILED (failures=1)

   hello-mac:csi185website michaeltoth$ [↑](#footnote-ref-0)
3. My change to self.assertIn("The Beginning",h.text) resulted in this:

   hello-mac:csi185website michaeltoth$ python3 functional\_tests.py

   F

   ======================================================================

   FAIL: test\_home\_page (\_\_main\_\_.NewVisitorTest)

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

   Traceback (most recent call last):

   File "functional\_tests.py", line 55, in test\_home\_page

   self.assertIn("The Beginning",h.text)

   AssertionError: 'The Beginning' not found in 'The Title of My Home Page'

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

   Ran 1 test in 2.495s

   FAILED (failures=1)

   hello-mac:csi185website michaeltoth$ [↑](#footnote-ref-1)
4. http://www.w3schools.com/tags/tag\_img.asp [↑](#footnote-ref-2)
5. In my case I get

   hello-mac:csi185website michaeltoth$ python3 functional\_tests.py

   F

   ======================================================================

   FAIL: test\_home\_page (\_\_main\_\_.NewVisitorTest)

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

   Traceback (most recent call last):

   File "functional\_tests.py", line 48, in test\_home\_page

   self.assertIn('homebrew2.png',m.get\_attribute('src'))

   AssertionError: 'homebrew2.png' not found in 'http://localhost:8000/static/homebrew.png'

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

   Ran 1 test in 5.566s

   FAILED (failures=1)

   hello-mac:csi185website michaeltoth$ [↑](#footnote-ref-3)