**SPLASH NOTES:**

**TO START SPLASH (via docker) from Terminal Window:**

**At Command Line enter following:**

docker run -it -p 8050:8050 scrapinghub/splash

**This starts the server in the background and now you just go to the url port that it is listening on (Note this address will appear in your terminal window once above command runs):**

<http://0.0.0.0:8050>

**PLEASE NOTE THAT YOU MUST LEAVE THE TERMINAL WINDOW OPEN & RUNNING IN BACKGROUND SINCE IT SERVES AS THE WEB SERVER!**

**TO STOP A DOCKER CONTAINER (i.e.: kill the process $ re-open the container fresh):**

killall Docker && open /Applications/Docker.app

Note: To see a list of your current running docker container processes do this command:

docker container ls -a

**TO KILL A SPECIFIC CONTAINER PROCESS, YOU CAN DO THIS:**

**Get the Containter ID:**

sudo docker ps \*\*note you may be asked for password (just use your machine login: ie: I used jr. developer mac login)

**Kill the process by entering the container ID**

Using Kill command (note the below ID (f6f86bef02ee ) is an example of what the ID would look like after you run the ‘sudo docker ps’ command above):

sudo docker kill f6f86bef02ee

Sample Splash Script:

<http://small-change.caftest.org/calendar/>

PE\_Link = splash:select(//\*[@id="pagecontent"]/main/section[1]/div[1]/div[1]/header/div/div/h1)

PE\_Link:mouse\_click()

splash:mouse\_click(“Programs & Events”)

splash:wait(0.1)  
 return splash:html()

Screenshot = splash:jpeg()

Local link = splash:select(‘Programs & Events’)

Link:mouse\_clck()

**function** main(splash)

splash:go("http://example.com")

splash:wait(0.5)

**return** {html=splash:html()}

**end**

**function** main(splash)

**local** element = splash:select('.element')

**local** bounds = element:bounds()

assert(element:mouse\_click{x=bounds.width/3, y=bounds.height/3})

**end**

a.button.button\_black

Calendar dimensions (link button) 76.31 x 50

<http://small-change.caftest.org/calendar/>

class=”nav\_link nav\_link\_icon nav\_link\_active”

\*\*\*

Use below command in console (during inspect mode in browser) to get scroll position for element on page (x and y)

$0.getBoundingClientRect()

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

**Calendar** (Scroll Position)

DOMRect {x: 547.21875, y: 0, width: 76.3125, height: 50, top: 0, …}

splash:mouse\_press(547.21875, 0)

splash:mouse\_press(547.21875, 0)

splash:mouse\_click(320.125,65)

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

**Tours** (Scroll Position)

x: 320.125, y: 65

splash:mouse\_hover(320.125, 65)