



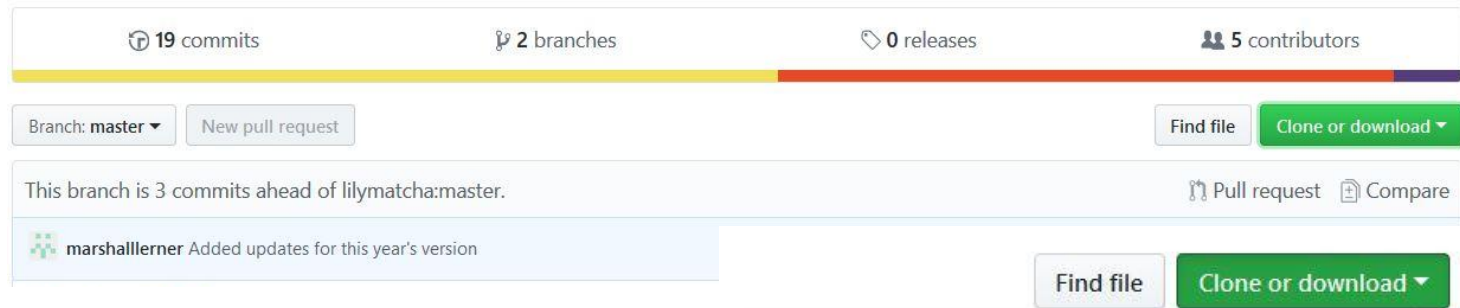
# Heatmap and Replay Gear-up

With your host, Marshall

# Step 1: Download the files using Github

<https://github.com/marshalllerner/1300-eye-tracking>

*No description, website, or topics provided.*



19 commits   2 branches   0 releases   5 contributors

Branch: master   New pull request   Find file   Clone or download

This branch is 3 commits ahead of lilymatcha:master.   Pull request   Compare

marshalllerner Added updates for this year's version

Find file   Clone or download

## Clone with HTTPS ?

Use Git or checkout with SVN using the web URL.

<https://github.com/marshalllerner/1300-eye-tracking>

Open in Desktop

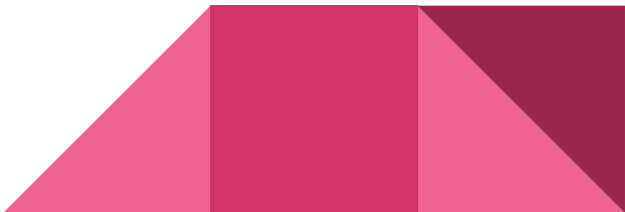
Download ZIP

6 days ago

## Step 2: Put your screenshots and log files into the extracted Github folder

We will provide you with screenshots on the day of your eyetracking session, so you can use samplescreen.png for the purpose of this gearup.

If you don't have log files yet, you can use the sample one here: [www.eternote.com/sample](http://www.eternote.com/sample)



## Step 3: Check out the files with a text editor

If you don't have a text editor already installed, we recommend Atom or Sublime Text

Heatmap.html: 4 TODOs

Replay.html: 3 TODOs



# Heatmap.html TODOs

1. Put your CSV filename in the constant CSV\_NAME

```
const CSV_NAME = 'TODO: put your CSV filename here';
```

This would be the relative path to your CSV file from your heatmap.html. For example, if your folder looks like this, it will be “log.csv”:

> Desktop > 1300-eye-tracking

Name	Date modified
.git	11/11/2018 11:27
heatmap.js-2.0.5	11/11/2018 11:11
heatmap	11/11/2018 11:11
heatmap	11/11/2018 11:23
log	11/17/2018 11:20
replay	11/11/2018 11:26
samplescreen	11/11/2018 11:11

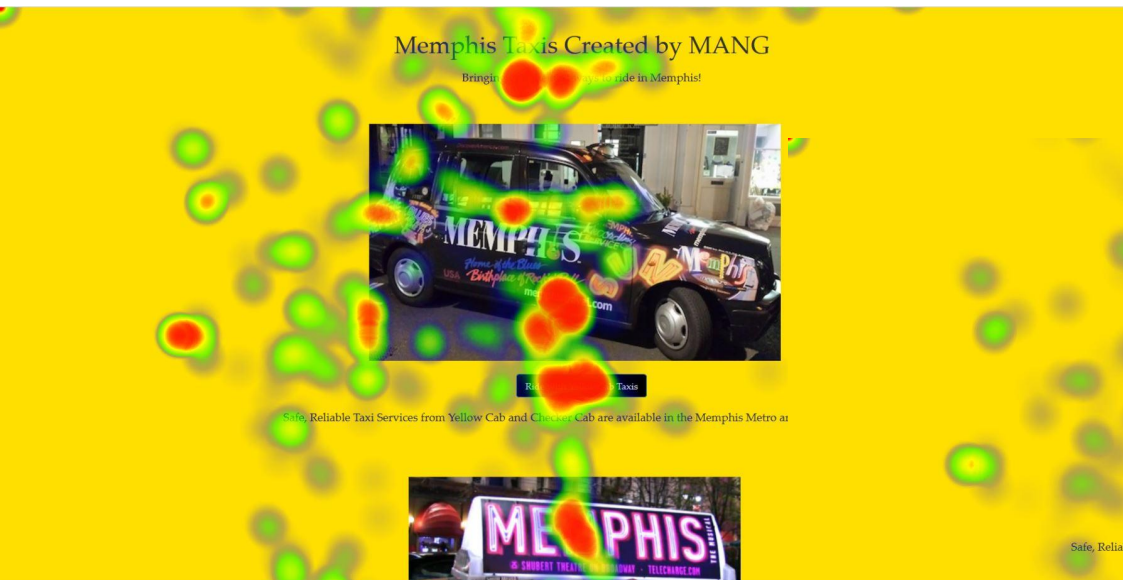
# Heatmap.html TODOs cont'd

2. This is the meat of the assignment. This is where you take each line from your CSV and put the data points into the datapoints array. This shouldn't be more than 20 lines of code



# Heatmap.html TODOs cont'd

- This is the a value to change the threshold for the colors of the heatmap. In the example below, the left image for max is a lower value than the right image. You want a few red areas

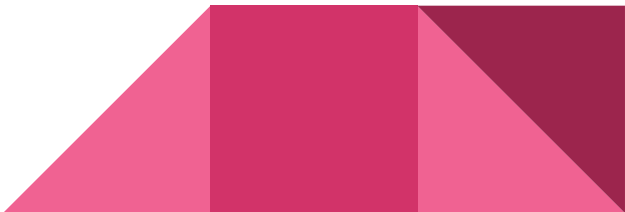


# Heatmap.html TODOs

4. Put your screen filename in the placeholder just like the CSV file from step 1 (using the relative path). In our previous example, the placeholder would be "samplescreen.png"

```
</script>
<body style="margin:0">
<h2>Heatmap</h2>
<div id="one" class="heatmap" style="position: relative;">

</div>
</body>
</html>
```





# Replay.html TODOs

1. Put your CSV filename in the constant CSV\_NAME. This would be the relative path to your CSV file from your heatmap.html. For example, if your folder looks like this, it will be "log.csv".
2. Put your screenshot relative path filename in the url("") pointed at by an arrow below.

```
<script src="heatmap.js-2.0.5/build/heatmap.min.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
<link rel="stylesheet" href="heatmap.css"/>
</head>
<body style="margin:0; background-image: url('TODO: put your screenshot filename here'); background-repeat: no-repeat;">
<canvas id="myCanvas" width="1920" height="1200" style="position: absolute; left: 0px; top: 0px">
<script>
// TODO: in the line two lines above this one, replace the width and height values with the original resolution of your screenshot

const CSV_NAME = 'TODO: put your CSV filename here';
```



# Replay.html TODOs cont'd

3. This is the meat of the assignment. This is where you take each line from your CSV and put the data points NOT into the datapoints array this time, but use `ctx.fillRect(x_value, y_value, width_dot, height_dot)`. This shouldn't be more than 20 lines of code

Note: You must set the variable `ctx.fillStyle`. To set the variable, you say

`ctx.fillStyle = ____`

where you fill in the \_\_\_\_ with the color of the shapes you are filling, like `"rgb(255, 0, 0)"` for red (keep the quotes).

You must set this before calling `ctx.fillRect()`



# Viewing the actual heatmap or replay of your data

1. If you do not have Python, **download** and install it [for Mac](#) or [for Windows](#).
2. To test your code, open *Terminal*, **navigate to the folder that contains “heatmap.html,”** and **run the following command to launch your Python server:**  
  
**If you have Python2:** `python -m SimpleHTTPServer 8000`  
**If you have Python3:** `python -m http.server 8000`
3. Once your Python server is running, the visualizations will be displayed at this URL:  
<http://localhost:8000/heatmap.html> and <http://localhost:8000/replay.html>