

Metrics Report #1

Metric: Average Lines of Code Per Method

Prepared By Victoria Law

Submitted February 10th, 2019

CSC 405 Capstone, Team Flash

Winter 2019 Dr. Da Silva

Frontend LOC

For our frontend, we are using ReactJS.

Frontend			
File Name	Total LOC	Number of Methods	Average LOC
/frontend/src			
App.js	22	2	11
App.test.js	7	1	7
index.js	18	1	18
serviceWorker.js	131	5	26.2
/frontend/src/js			
index.js (2)	142	8	17.75
rootReducer.js	18	2	9
/frontend/src/js/components/header			
index.js (3)	192	5	38.4
/frontend/src/js/popups			
popup_container.js	87	8	10.875
reducer.js	23	1	23
/frontend/src/js/popups/dir_adder			
index.js	109	13	8.384615385
/frontend/src/js/popups/options			
index.js	114	11	10.36363636
/frontend/src/js/toolbox/components/api			
index.js	61	5	12.2
/frontend/src/js/toolbox/components/button			
index.js	258	9	28.66666667
/frontend/src/js/toolbox/components/explorer			
column.js	63	7	9
editable_item.js	40	3	13.33333333
explorer.js	138	6	23
index.js	169	14	12.07142857
item.js	55	4	13.75
preview.js	48	5	9.6

search.js	59	4	14.75
spacer.js	7	1	7
truncate.js	17	1	17
/frontend/src/js/toolbox/components/file_input			
index.js	108	7	15.42857143
/frontend/src/js/toolbox/components/icon			
index.js	51	2	25.5
/frontend/src/js/toolbox/components/options_button			
	53	6	8.833333333
/frontend/src/js/toolbox/components/spinner			
index.js	38	2	19
/frontend/src/js/toolbox/constants			
animation.js	53	1	53
color.js	95	1	95
fontSize.js	11	1	11
index.js	29	1	29
lineHeight.js	4	1	4
spacing.js	10	1	10
transition.js	3	1	3
/frontend/src/js/views			
reducer.js	22	1	22
/frontend/src/js/views/main			
index.js	197	19	10.36842105
input_modal.js	136	11	12.36363636
/frontend/src/js/views/visualizer			
index.js	57	7	8.142857143
TOTAL:	2049	133	15.40601504

Backend LOC

For our backend, we are using Python.

Backend			
File Name	Total LOC	Number of Methods	Average LOC
/backend			
csvToJson.py	75	5	15
server.py	43	2	21.5
/backend/src			
CsvInterpreter.py	59	5	11.8
DataClassifier.py	19	3	6.333333333
Tester.py	10	1	10
Total (All Files)	206	16	12.875

Analysis/Observations

The metric I chose was size, measuring the lines of code for both source files as well as test files. The average lines of code for frontend and backend are pretty similar. For the frontend, since we are using ReactJS, all of the files were javascript files, and the average lines of code for all the files is 15.41. Most of the frontend code is used to communicate with the API, and used to design the UI. And for the backend, all the files were in python, and the average lines of code for all the backend files is 12.88. All the backend code is for connecting to the API, creating endpoints for the frontend to communicate to, conversion of CSV to JSON, and start of implementing our tag system.

It is generally a good practice to have lesser lines of code per function to produce loose coupling. This also makes it easier to debug when a problem arises because you can easily see the code all in one screen, instead of having to keep on scrolling up and down a function. For our backend, we are doing pretty well in terms of having a balanced amount of lines of code per method. But for our frontend, we can refactor the code so that there is less lines of code per function for some of the functions (i.e. animation.js, color.js, and etc). Since both animation.js and color.js relates to how our product looks like, if we reduce the lines of code per function, this would help us in the future if we need to add additional features to the code, or if we want to change the look of the product. I do not see any major problems with our code thus far.