Benchmarking JavaScript Frameworks

|  |  |
| --- | --- |
| Lines of Code | (Park, 1992) |
| Cyclomatic Complexity | (McCabe, 1976) |
| Halstead Complexity | (Halstead, 1977) |
| Maintainability Index | (Oman & Hagemeister, 1992) |
| Database metrics | (Calero, Piattini, & Genero, 2001) |
| Page-load/render-time | (Christodoulou & Gizas, 2014), (“Metrics - WebPagetest Documentation,” 2008) |
| Speed index | (“Metrics - WebPagetest Documentation,” 2008) |
| Render and frame measurements | (“Timeline Event Reference | Web,” n.d.) |

Comparative Evaluation of JavaScript Frameworks

Size Metrics: *lines of code (LOC),*

Complexity Metrics - *McCabe’s cyclomatic complexity*

Maintainability Metrics: Halstead

metrics (*Program Volume* and *Program Level*) and

*Maintainability Index (MI)*.

Open google and find papers of every google lighthouse metrics

Available for <https://ieeexplore.ieee.org/abstract/document/8780515> first contentful paint

write about overview of frameworks in some chapter from paper pattern frameworks and middleware

from the paper Benchmarking javascript framework chapter 4.4 use this as why you chose to do speed indexing

Insert referencing for todomvc

IF CONtent is short write about typescript somewhere

UI,DOM,SPA, SLOC & LLOC FCP, SEO

Write about library vs js

Write about version releases

Next section write about paramters choosen and how to compute them

Change thises aim to compare browser performance

Last list item

Code size

Community available

Developers experince

Lighthouse

<https://www.practicalecommerce.com/google-lighthouse-monitor-site-performance-seo-accessibility>

<https://www.magnet4blogging.net/lighthouse-tool/>

<https://www.greengeeks.com/blog/2019/08/15/google-lighthouse-how-you-use-it/>

headers, about home.

Angular 8

React

Vue.js

Ember.js

System architecture

About testing environment

Scope & limitations