

Team Big Data

U-tification

DAR Report

JavaScript Chart Library

Date: 12/14/2022

Team Leader: Joseph Armas

Team Members: Joshua Gherman

Rhoy Oviedo

Frank Curry

Ghabrille Ampo

David DeGirolamo

Git Repository: <https://github.com/JosephArmas/cecs-491A-Team-Big-Data>

Technologies:

1. Chart.js | Version 4.0.1
2. Chartist | Version 1.3.0
3. C3.js | Version 0.7.2
4. MetricsGraphics.js | Version 2.15.6

Metrics:

1. Trend Chart
2. Time
3. Json compatible (Track pins created)
4. Loading Performance
5. Multiple Charts on a Single Page
6. Google Chrome Support

Introduction

The problem is the issue with tracking data and creating a trend chart in correlation with users interacting with our application. Therefore, a chart library is required for our application to display a trend chart in the usage analysis dashboard.

Technologies/Metrics	Chart.js	Chartist.js	C3.js	MetricsGraphics.js
Trend Chart (1.2x)	True [4]	True [4]	True [4]	True [4]
Time (configurable x or y axis) (1.1x)	True [4]	True [4]	True [4]	True [4]
Json compatible (Track pins created) (1.2x)	True [4]	True [4]	True [4]	True [4]
Loading Performance (1.2x)	7 ms [3]	~0.4 ms [4]	28 - 32 ms [2]	? [1]
Multiple Charts on a Single Page	True {3+} [4]	True {28+} [4]	True {3+} [4]	True {23+} [4]

(1.1x)				
Google Chrome 102.0.5005.189 (1.2x)	True [4]	True [4]	True [4]	True [4]
Total	26.8	28	25.6	24.4

Valuation was determined 1-4 with 4 being the best option and 1 being the worst option for each metric. The weights were determined by how important they are to make our application work. The score each metric gets is inside of the brackets “[]”.

Discussion

For the Loading Performance using a few demos from an online source the values were derived for how long it took them to completely load. These metrics are not exactly what we are looking for in each technology. The loading performance should be after taking in large amounts of data from a database then displaying a graph. These are from small pre-defined data points. From an extrapolation to a large quantity of data points, we can likely still get within the threshold of 15 of loading. When Chartist.js was tested, it came with a really low value, this is likely because it creates a graph in SVG and does not have built-in event handling like the other technologies. For the Multiple Charts on a Single Page the value within the parentheses “{ }” represents the number of charts that we could find on a single page. While there is no explicit cap on the number of graphs, these were cited as just some of the best represented webpages. Since our usage analysis dashboard requires at least 4 graphs to be represented on a single page all of the technologies fulfill this requirement.

Conclusion

Overall we can see that Chartist.js is the best fit for our application. We can see that chartist.js meets the requirements of having our charts load under 15s, configure time on the x or y axis, provides a trend chart or various other charts onto a single page and supports Google Chrome browser. Although Chart.js comes in second with meeting the requirement of a chart loading under 15s, it is better to allow extensibility and scalability when more charts are needed in a single page with minimal decrease in performance, hence, Chartist.js meets those requirements.

Reference

All

<https://medium.com/sicara/compare-best-javascript-chart-libraries-2017-89fbe8cb112d>

Chartist.js

<https://gionkunz.github.io/chartist-js/>

<https://www.chartjs.org/docs/latest/configuration/interactions.html>

<https://www.npmjs.com/package/chartist?activeTab=versions>

<https://michaelmangial1.medium.com/csv-to-json-to-chartist-js-7c5c9e2c37c4>

<https://www.chartjs.org/docs/latest/general/performance.html>

[https://www.predictiveanalyticstoday.com/chartist-js/#:~:text=js%20\(angular%20directive.-,Chartist.,Safari%206%2C%20iOS%20Safari%207.](https://www.predictiveanalyticstoday.com/chartist-js/#:~:text=js%20(angular%20directive.-,Chartist.,Safari%206%2C%20iOS%20Safari%207.)

Chart.js

<https://www.chartjs.org/docs/latest/charts/line.html>

<https://stackoverflow.com/questions/34677136/get-all-the-data-from-a-clicked-node-in-c3js>

<https://github.com/chartjs/Chart.js/releases>

<https://webflow.com/made-in-webflow/website/No-CodeLow-Code-MemberStack-Ready-Dashboard>

C3.js

<https://c3js.org/gettingstarted.html>

<https://github.com/c3js/c3/releases/tag/v0.7.20>

<https://github.com/c3js/c3/issues/1823>

MetricsGraphics.js

<https://metricsgraphicsjs.org/line>

<https://cdnjs.com/libraries/metrics-graphics/2.15.6>

<https://www.npmjs.com/package/metrics-graphics?activeTab=versions>

<https://cran.r-project.org/web/packages/metricsgraphics/vignettes/introductiontometricsgraphics.html>