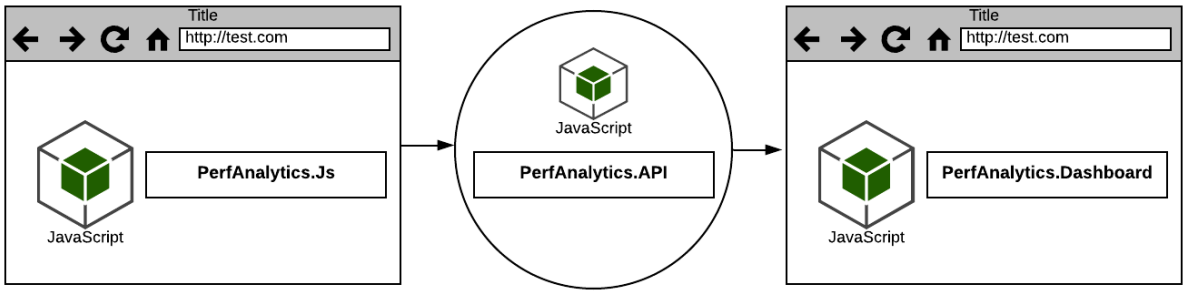
PERFANALYTICS
BECAUSE PERF MATTERS

PerfAnalytics is an ecosystem which collects and criticizes web performance data. The ecosystem consists of 3 subsystem;

- **PerfAnalytics.Js**
- **PerfAnalytics.API**
- **PerfAnalytics.Dashboard**



PerfAnalytics.JS is a client-side library, which collects some performance related key metrics from browser and sends to the PerfAnalytics.API

Acceptance Criteria

- It does not harm clients performance
- It should measure TTFB, FCP, Dom Load, and Window Load events
- It should measure Network timings for Document, Image, Font, JS, and CSS
- It should work on all modern browsers Except Microsoft Products
- It should send performance metrics to API in a proper way.
- It should be generic and can be used in any web application (big plus)
- It should smaller than 3KB Gzip (big plus)

trendyol
tech

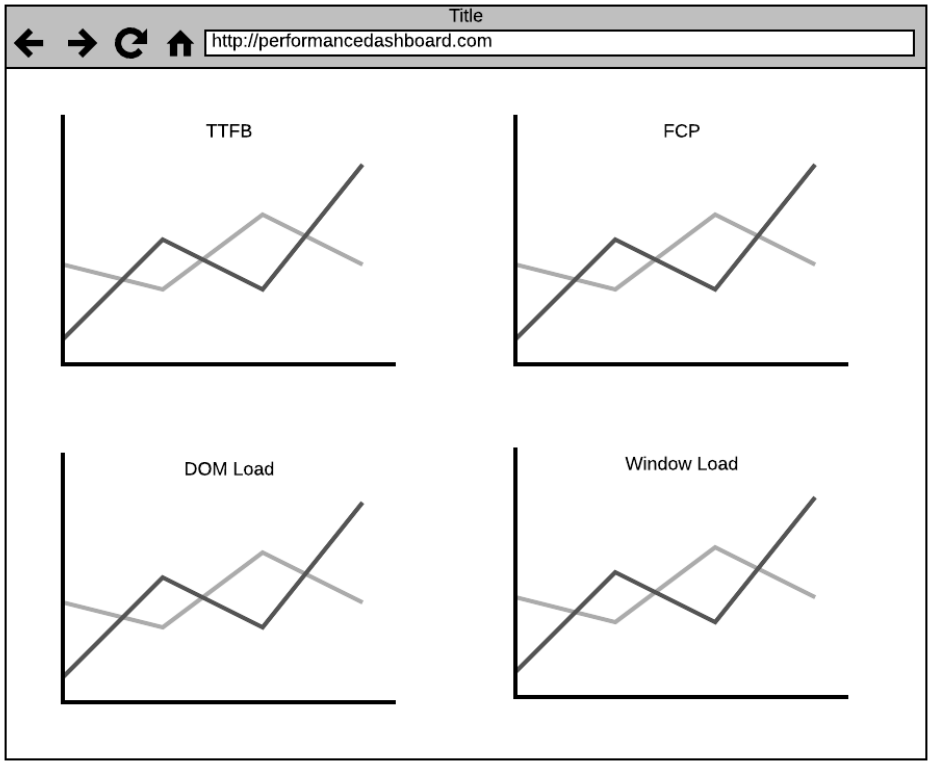
PerfAnalytics.API is a restful API which saves data, posted from **PerfAnalytics.JS** and returns time specific filtered data. *Acceptance Criterias*

- It should handle min 200 RPS (a proof is a big plus)
- It should calculate dashboard metrics < 1 second
- It should return data between specific dates or only last 30 minutes(big plus)
- It should be well designed for future implementations.
- It should be well tested
- It should be properly designed for REST standards

PerfAnalytics.Dashboard is a dashboard which shows perf related metrics in a visualized way.

Acceptance Criteria

- It should be written using React
- It should have component based architecture
- It should have a responsive layout
- It should show the last 30 minutes of TTFB, FCP, Dom Load, and Window Load as in charts
- It should get data from **PerfAnalytics.API**
- It should be rendered within 2 seconds



trendyol
tech

CI / CD *Acceptance Criteria*

- The application should be dockerized
- Automated deployment and testing should be implemented. Free providers can be used like Heroku etc.
- Tests should run before each deployment