Lighthouse

A User-Centric Performance Testing Tool

Summer Tang
2020/09/11

Agenda

- Lighthouse
- RAIL
- Metrics
- Goals
- How does Lighthouse monitor DDD?
- Recommendation

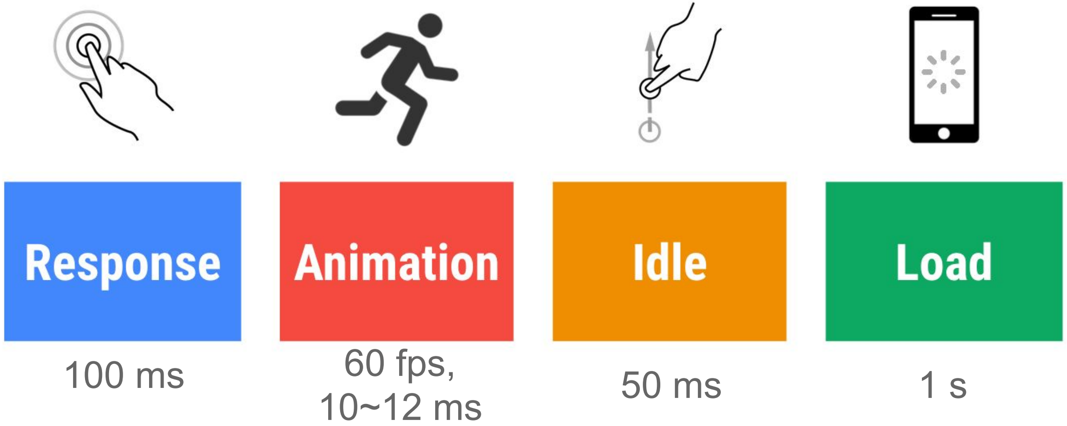
Lighthouse

- Lighthouse is an open-source, automated tool for improving the quality of web pages.
- It has audits for:
  - **Performance**
  - Accessibility 無障礙
  - Progressive Web Apps
  - **Best Practices**
  - SEO (use Search Console Tools would be better)

  and it is continually updated.

- Give Lighthouse an URL to audit, it runs a series of audits against the page, and then it generates a report on how well the page did.
- The report offers opportunities for optimisation and estimated savings.

RAIL (1/2)



Response

100 ms

Animation

60 fps,
10~12 ms

Idle

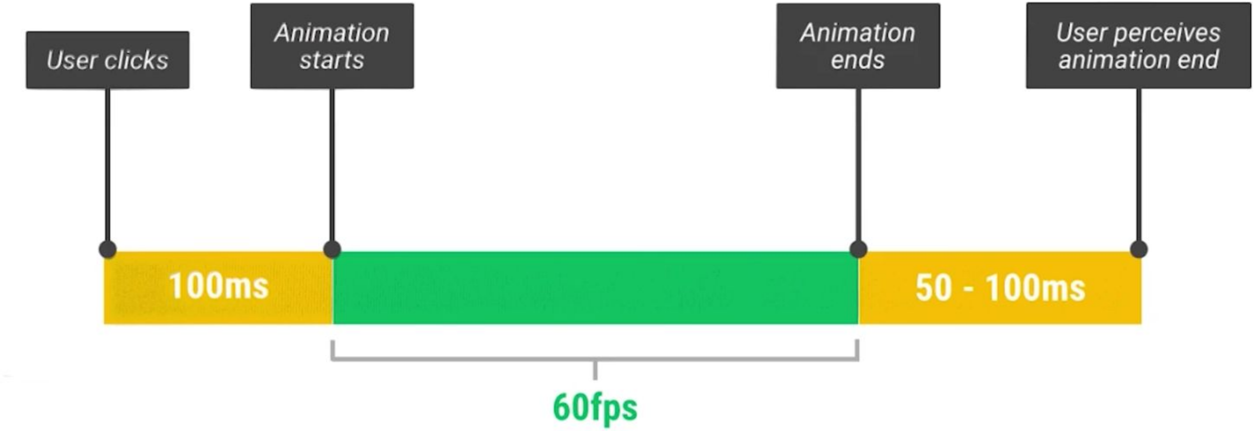
50 ms

Load

1 s

- RAIL is a **user-centric performance model** that provides a structure for thinking about performance.
- RAIL breaks down the user's experience into key actions and combine with different web app life cycle: **Response**, **Animation**, **Idle**, and **Load**. Each of them has its own performance goal.

RAIL (2/2)



- User click UI and expect the result, e.g., animation start, at most 100 ms.
- Animation may take 10 ~ 12 ms and 16 ms per frame (smooth: 60 frames in 1 s).
- Animation end and idle for the next interaction, at most 50 ms; or deliver another feedback, at most 100 ms.

Metrics

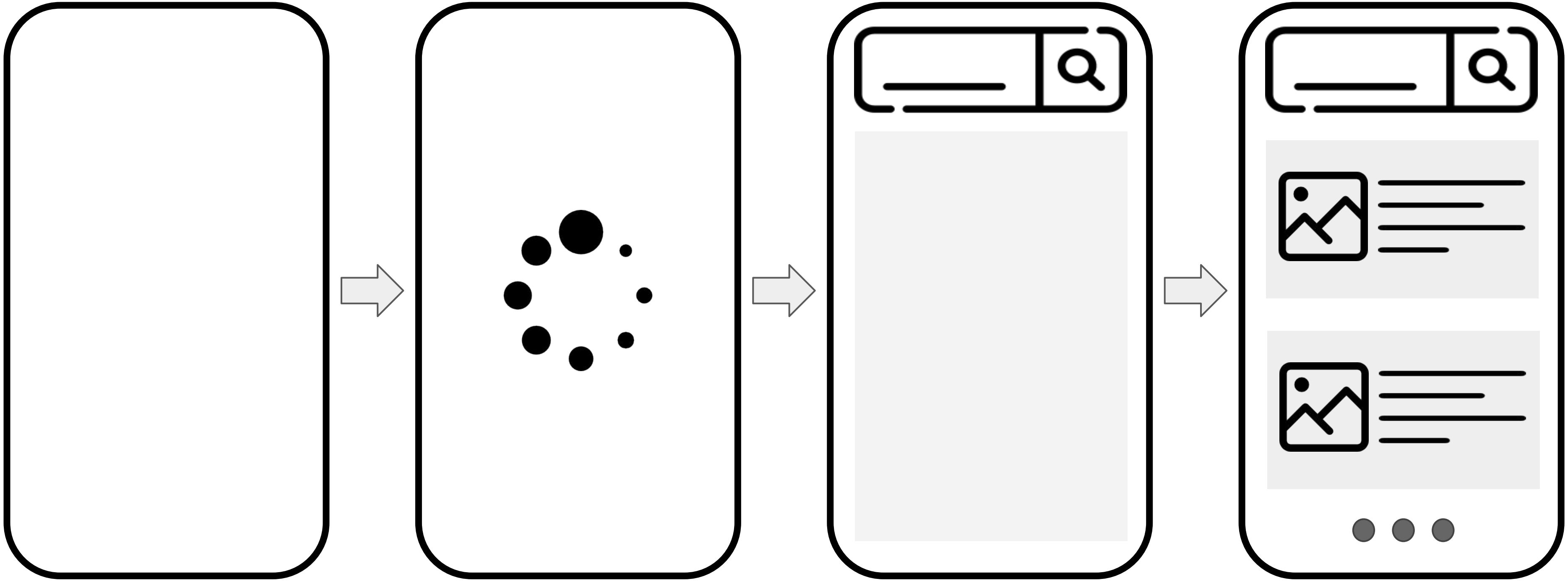
These metrics measure the web app's performance across a number of dimension based on RAIL.

- Perceived Load Speed
  - First Contentful Paint (FCP)
  - Largest Contentful Paint (LCP)
  - Speed Index
- Load Responsiveness
  - First Input Delay (FID)
  - Time to Interactive (TTI)
  - Total Blocking Time (TBT)
- Visual Stability
  - Cumulative Layout Shift (CLS)

Perceived Load Speed

First Contentful Paint
Largest Contentful Paint
Speed Index

First Contentful Paint, Largest Contentful Paint



FCP

LCP

Speed Index

Calculate the speed index

$$= \quad \text{Speed Index} = \int_0^{end} 1 - \frac{VC}{100}$$

end = end time in milliseconds
VC = % visually complete

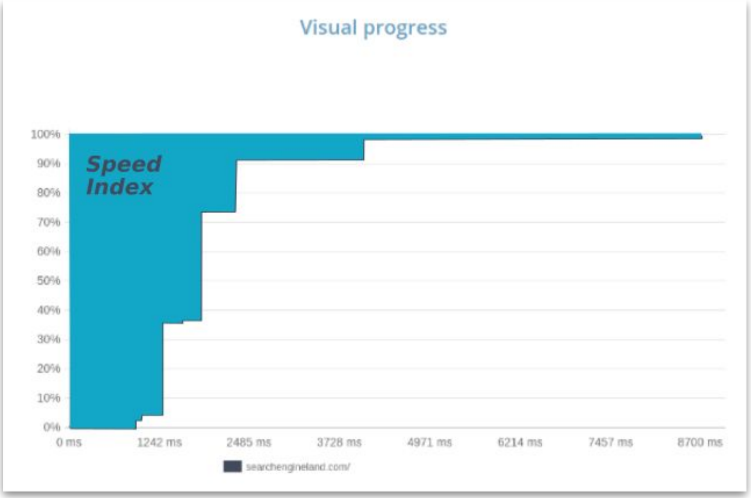


Image source

Page 1 Speed index: **6500**

0%	10%	10%	10%	10%	10%	10%	55%	90%	100%

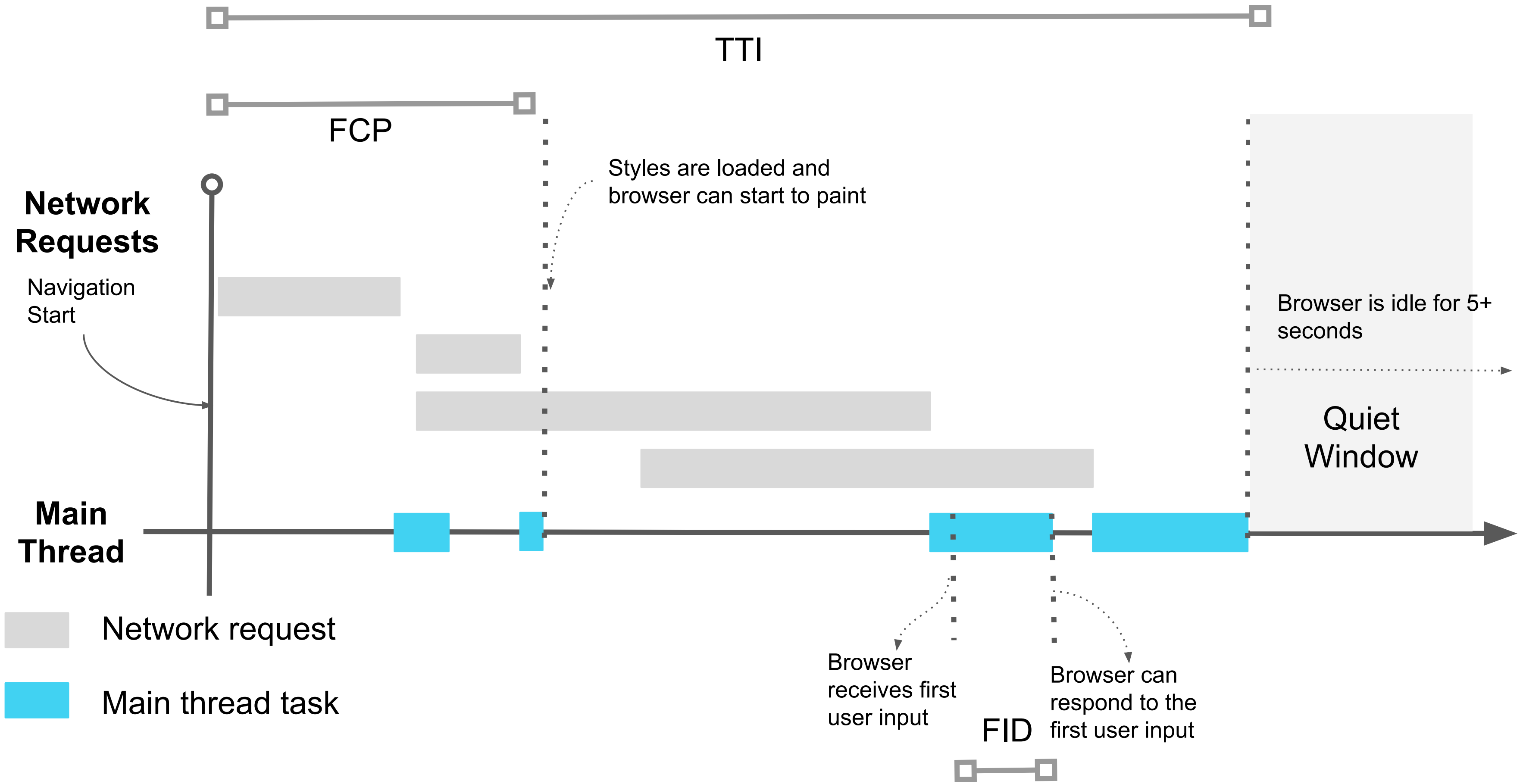
Page 2 Speed index: **5000**

0%	10%	20%	30%	40%	50%	65%	78%	95%	100%

Load Responsiveness

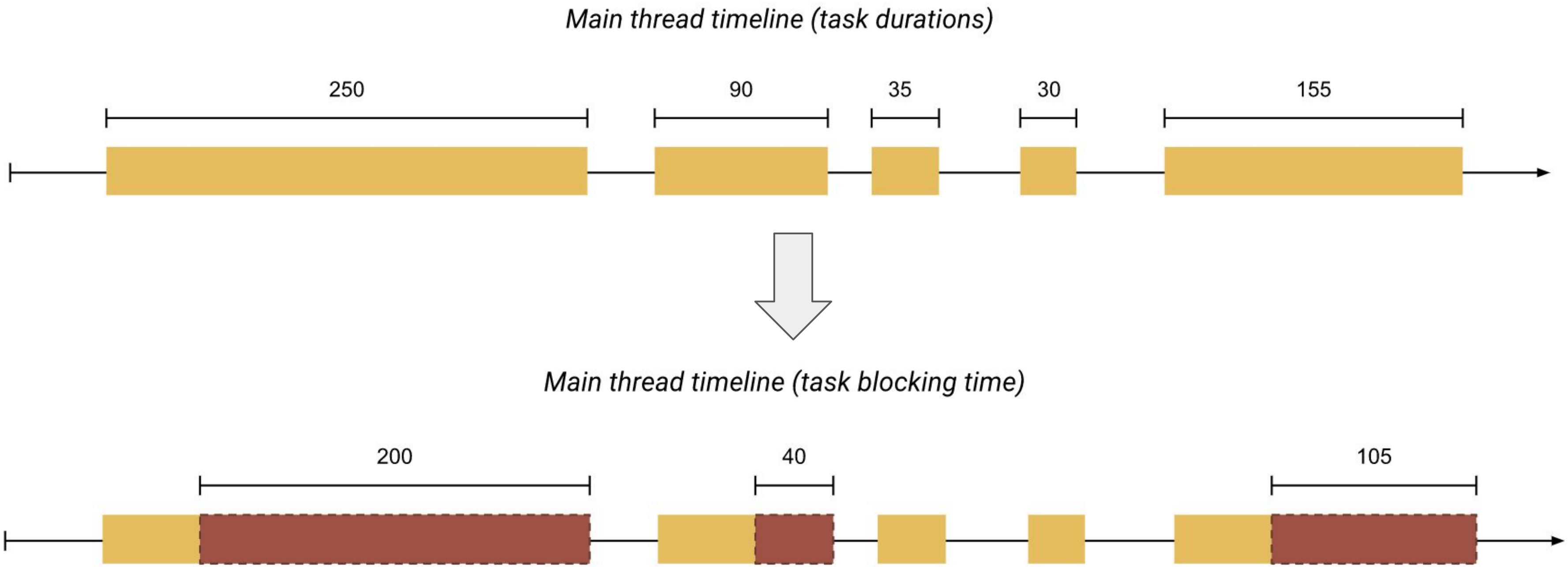
First Input Delay
Time to Interactive
Total Blocking Time

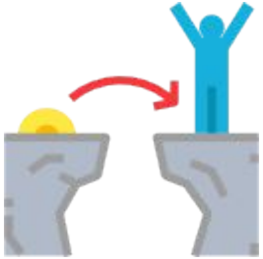
Time to Interactive, First Input Delay


TTI
FCP
Styles are loaded and browser can start to paint
Network Requests
Navigation Start
Browser is idle for 5+ seconds
Quiet Window
Main Thread
Network request
Main thread task
Browser receives first user input
FID
Browser can respond to the first user input

Total Blocking Time

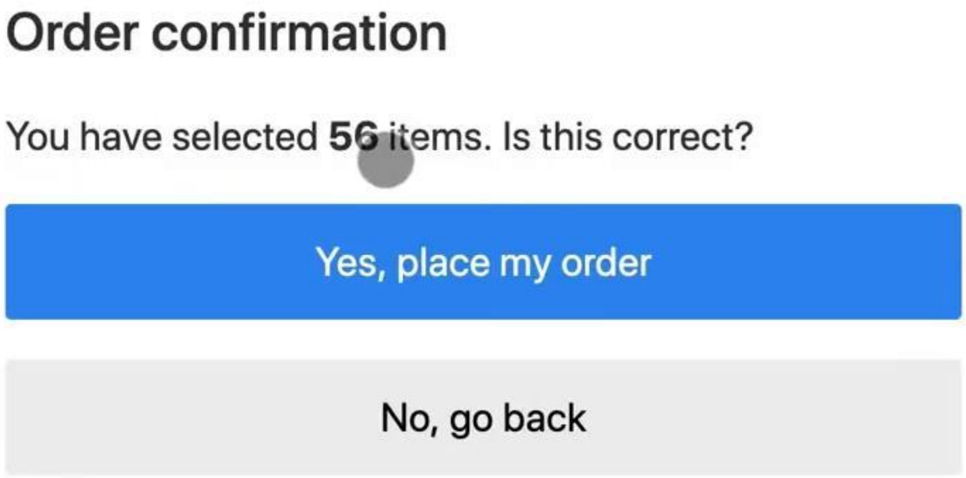
Total blocking time (> 50ms) between FCP and TTI.


Main thread timeline (task durations)
250
90
35
30
155
Main thread timeline (task blocking time)
200
40
105



Visual Stability

Cumulative Layout Shift

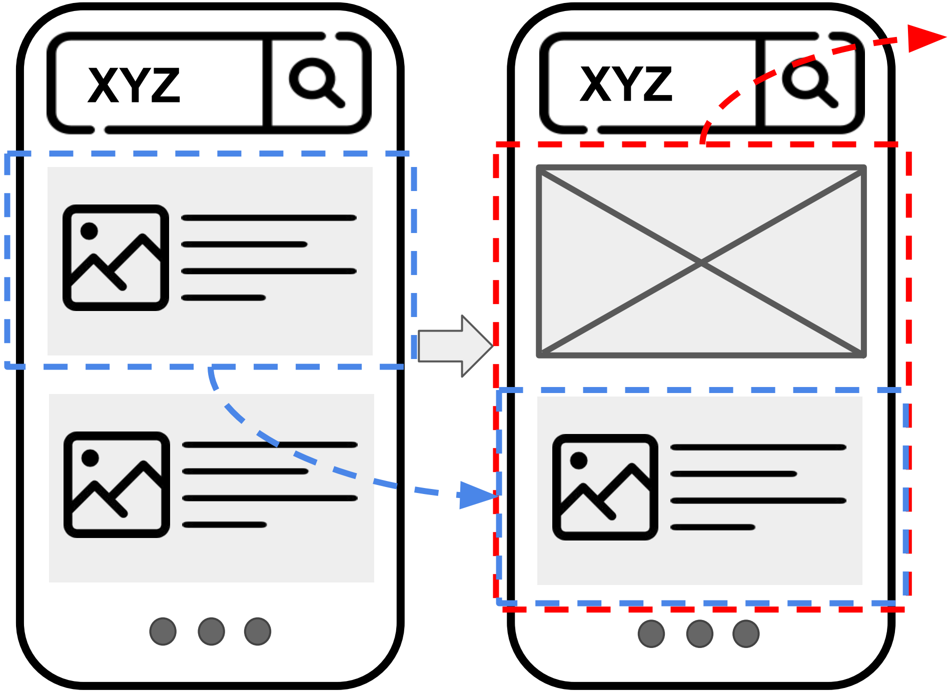


Order confirmation

You have selected **56** items. Is this correct?

video source

Cumulative Layout Shift



After the search result load complete...

Ad shows, and the search result block moves downward...

CLS
= impact fraction * distance fraction
= ⅔ * ⅓ = 2/9 ~ 0.22

- impact fraction
  = impact area / whole area
  = ⅔
- distance fraction
  = the greatest distance has moved
  = ⅓

Goals

Key performance metrics related to user experience.

Metrics	Good	Needs Improvement	Bad
First Contentful Paint (FCP)	< 1 s	(UDSO: 4.3 s)	
Largest Contentful Paint (LCP)	< 2.5 s	2.5 ~ 4 s	> 4s (UDSO: 4.9 s)
First Input Delay (FID)	< 100 ms	100 ~ 300 ms	> 300ms
Time to Interactive (TTI)	< 5 s	(UDSO: 5.8 s)	
Total Blocking Time (TBT)	< 300 ms	300 ~ 600 ms (UDSO: 590 ms)	> 600 ms
Cumulative Layout Shift (CLS)	< 0.1	0.1 ~ 0.25	> 0.25 (UDSO: 0.311)
Speed Index	< 4.4 s	4.4 ~ 5.8 s (UDSO: 5.2 s)	> 5.8 s

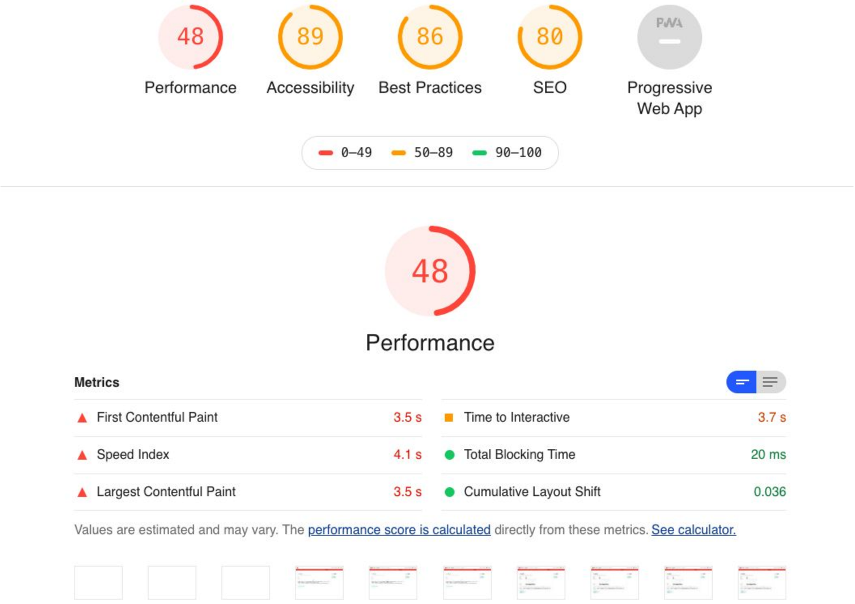
How does Lighthouse monitor DDD?

- Test specific pages (current: dashboard, UDSO, so report, help) by using Lighthouse and Puppeteer.
- Generate reports at 11 AM every Friday by Jenkins.
- Improve UI performance based on the metrics and suggestions from the reports.

Performance Report

- Metrics: The metrics measure the web app's performance across a number of dimension, including FCP, LCP, FID, TTI, TBT and CLS.
- Opportunities: The suggestions can help the page load faster. List the estimated savings behind the Opportunities.
- Diagnostics: More information about the performance of the application, e.g., remove unused files.

Lighthouse Performance Report



48 Performance

89 Accessibility

86 Best Practices

80 SEO

PWA – Progressive Web App

▬ 0–49 ▬ 50–89 ▬ 90–100

48

Performance

Metrics

▲	First Contentful Paint	3.5 s	■	Time to Interactive	3.7 s
▲	Speed Index	4.1 s	●	Total Blocking Time	20 ms
▲	Largest Contentful Paint	3.5 s	●	Cumulative Layout Shift	0.036

Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

https://adc.github.trendmicro.com/pages/summer-tang/ddd-fe-performance-testing/

Recommendation (1/2)

#	Suggestions	Impact Metrics	Solution
1	Avoid an excessive DOM size A large DOM will increase memory usage, cause longer style calculations, and produce costly layout reflows.	FCP, LCP, TTI	● Create DOM nodes only when needed, and destroy nodes when they're no longer needed. ● Minimize unnecessary re-renders.👍 ● Simplifying CSS selectors if cannot reduce DOM nodes.👍
2	Minimize main-thread work Reducing the time spent parsing, compiling and executing JS.	TTI, TBT	● Reduce DOM nodes and simplifying CSS rules to lessen the calculation burden. 👍 ● Use web worker instead of main thread mainly. 👍 ● Split bundled js files by page.

👍 Easy to apply in DDD

Recommendation (2/2)

#	Suggestions	Impact Metrics	Solution
3	Avoid large layout shifts PbAgGrid move from topmost and then move 334px downward.	CLS	Re-allocate space for header component or render PbAgGrid later.
4	Serve static assets with an efficient cache policy	FCP, LCP	Set HTTP caching policy properly.

Lighthouse optimization is not a one-time task.
Ongoing optimization is the new norm.

What might score 100% today, will not score 100% tomorrow.

Quote from Polly Pospelova’s sharing in
How To Get a 100% Lighthouse Performance Score