# WebPage Test

- It a tool by catchpoint which runs synthetic tests like catchpoint, it doesn’t have historic aggregation but when it comes to Catchpoint, we have historic aggregation

Historic Aggregation: It allows us to track how the performance of our web application or web site changed over time

- when it comes to catchpoint, it runs the tests based on the scheduling.

- catchpoint test runs on catchpoint agent and WPT tests run on WPT agent

- WPT captures unique metrics: core web vitals and lighthouse posts metrics

- catchpoint does support traceroute, error handling error

**WebPage Test Folder Creation:**

->Control Centre -> Test -> Product -> New -> WebPage Test

A screenshot of a computer

Description automatically generated

**WebPage Test Properties:**

Name: To name the folder

Description: To describe the folder in more detail

Location: Described In which product the folder is located

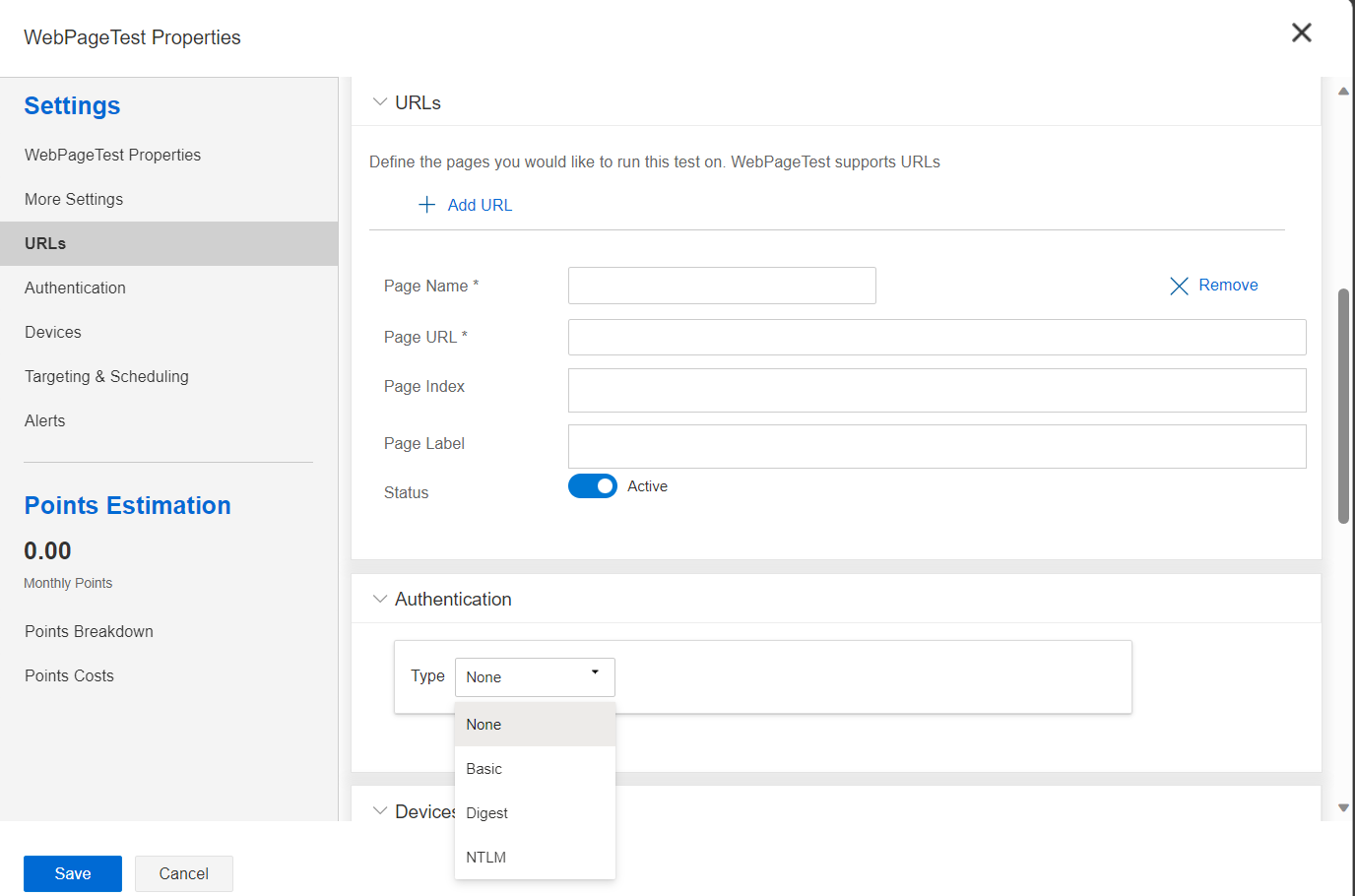
Status: To make the WPT folder active or inactive(if we make it inactive then all the childs of the folder will be inactive)

**More Settings:**

Run From: Time under which the test under WPT folder will be running

Index:

Labels: Used to categorize and tag the folder



**URLs:** This section is designed to input and manage the web pages we want to test with WebPage Test Folder

Page Name: A friendly name for the page to easily identify.

Page URL: Address of the web page

Page Index:

Page Label: To Identify and categorize WPT Test

Status: To make the test active or inactive.

**?Authentication :**

None, Basic, Digest, NTML

A screenshot of a computer

Description automatically generated

**Devices:** It allows us to define which devices we want to emulate when testing web pages, this functionality is important cause it lets us see how website works on different devices, here each URL and script will be run on each device

**Targeting and Scheduling:** It defines how often our tests will run and from what location

**Frequency:** Defines how frequently the test will occur.

Time Zone: Time Zone used for scheduling the test

Run Between: Specific time between which the test should run

Run Per Test: No of times the test will be executed, we can set it to some value to check consistency of the test or redundancy.

Run From: To set the nodes from which we want to run the tests in this WPT folder

Maintenance Schedule: We can inherit any maintenance schedule we created, it indicates here to avoid running the test during this time

**Alerts:** Notifications based on certain performance benchmark during the WPT test.

This is how we create the WPT folder

To Analyse the metrics:

Analyse -> Smartboard -> Select WPT Test ->

A screenshot of a computer

Description automatically generated

Overview Tab

Latest Performance

1. Core Web Vitals: These are the specific features considered as important in webpage overall user experience –
2. First contentful paint : The time from when the page start loading to when any part of the page rendered on the screen.

1.8 sec – good

1.8 to 3 sec – need improvement

>3 sec – poor

1. Largest Contentful paint: The time taken for the largest element in the viewport to become visible.

<2.5 – good

2.5-4 – needs improvement

>4 – poor

1. Total Blocking time: The amount of time that the page is blocked from responding to user input such as mouse clicks or screen taps or keyboard presses.

0.1ms – good

0.1 to 0.25ms – need improvement

>0.25ms – poor

A screenshot of a computer screen

Description automatically generated

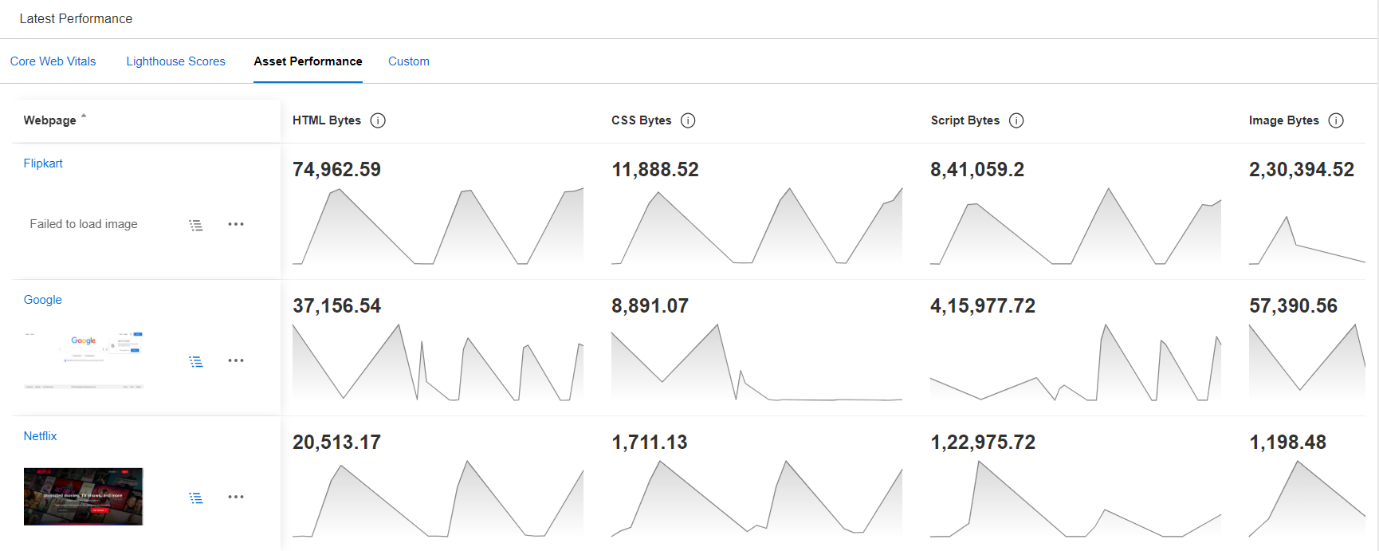
1. Cumulative Layout Shift: helps us to measure the visible stability of the page. It quantifies how often user experience unexpected layout shift during the lifespan of the page session. A layout shift occurs when the visible element changes its position from one rendered frame to next, this can be frustrating for users especially if they are about to click the button and it moves, resulting in mis click.

0.1ms – good

0.1 to 0.25ms – need improvement

>0.25ms – poor

1. Lighthouse Core: These cores represent different aspect of web quality
2. Performance Lighthouse: This determines how quickly the page becomes interactive
3. ??PWA Lighthouse: Stands for Progressive Web App, accessing the page against standard for progressive web applications which include factors like responsiveness and offline capability.
4. Accessibility Lighthouse: Determines how accessible is the page to users with disabilities.
5. Best Practices Lighthouse: Scores based on development best practices and not using deprecated libraries. Like using up to date APIs and avoiding deprecated features, performances best practices implementing the techniques that improves the speed, coding best practices means avoiding common mistakes that could introduce some bugs or security issues.
6. SEO(Search Engine Optimization) Lighthouse: Is a metric that gives us how well a webpage is optimized for search engines, this analyses the page for SEO best practices that can help increase the page’s visibility in search engine results.
7. Asset Performance :



Provides metrics on size of the various assets that make up a webpage.

HTML Bytes: The amount of data used by the HTML code of the page.

CSS Bytes: The size of the Cascading Style Sheets used to style the webpage.

Script Bytes: The amount of data used by JavaScript scripts on the page.

Image Bytes: The total size of image files loaded by the webpage.

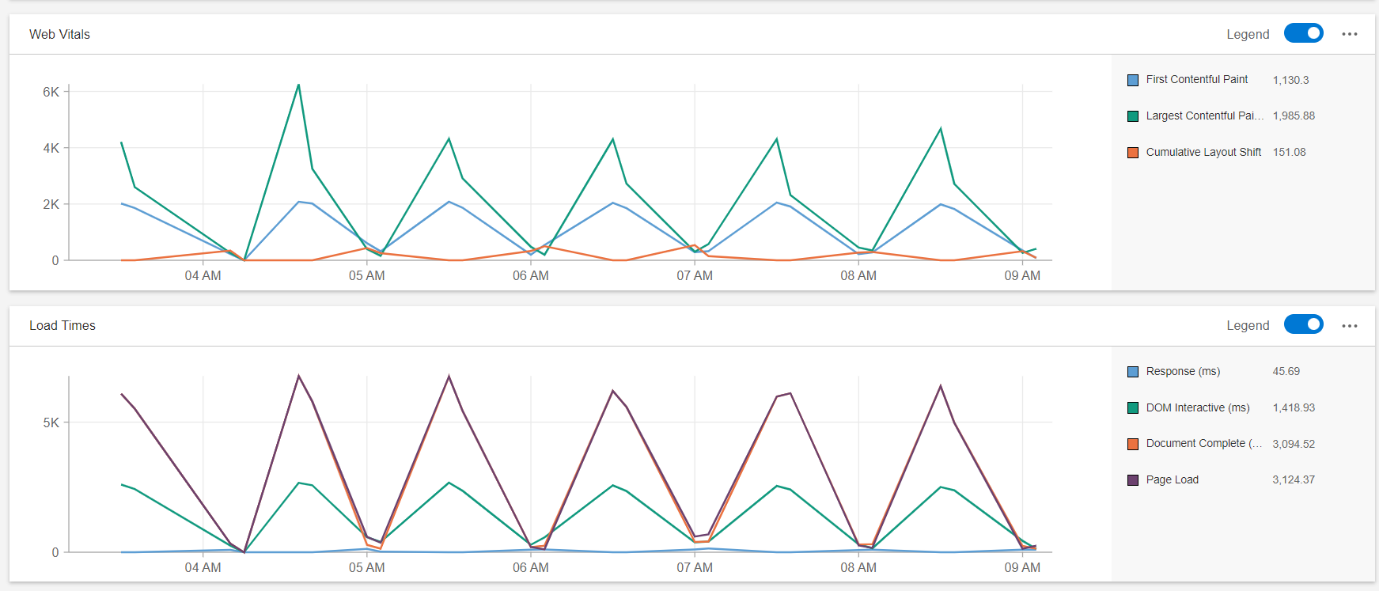
1. Custom : to customize the metrics

Webpage Tab

1. Visual Display: shows the visual loading performance of the web page.

A graph of different colored lines

Description automatically generated



A screenshot of a graph

Description automatically generated

Render Start (ms): The time taken for the page to start rendering.

Visually Complete (ms): The time taken for the contents of the page to be fully rendered.

Speed Index: A score that represents how quickly the contents of a page are visibly populated.

First Contentful Paint: The time taken for the first text or image to be painted on the screen.

Largest Contentful Paint: The time taken for the largest text or image to be painted.

1. Web Vitals:

First Contentful Paint (FCP): The time from when the page starts loading to when any part of the page's content is rendered on the screen. The lower the time, the better the perceived load speed.

Largest Contentful Paint (LCP): Measures the time taken for the largest content element visible within the viewport to be fully rendered. This is important because it gives a good indication of when the main content of the page has loaded.

Cumulative Layout Shift (CLS): Quantifies the amount of unexpected layout shift of visible page content. A low CLS score is good as it means there is less unexpected movement on the page.

1. Load Times:

Base Response (ms): How long it takes for the server to respond to a request.

DOM Interactive (ms): The time until the DOM is fully constructed and ready for interaction.

Document Complete (ms): The time until the browser has finished loading HTML and sub resources like images, CSS, and JavaScript but before any scripts have run.

Page Load (ms): The total time taken for the page to fully load.

1. CPU Time: Visualizes the time spent in various activities that affect CPU usage on a webpage, such as:

Scripting: Time spent executing JavaScript.

Layout: Time spent calculating the layout of the page.

Painting: Time spent painting pixels to the screen.

Loading: Time taken to load resources.

Idle: Time when the CPU is not doing any work.

1. Lighthouse Scores: "Lighthouse Scores" chart displays scores from the Lighthouse tool for various performance aspects:

LH Performance: Score for general performance (speed and load times).

LH PWA: Score for Progressive Web App criteria (offline work, responsiveness).

LH Accessibility: Score for accessibility features for users with disabilities.

LH Best Practices: Score for general best practices in web development.

LH SEO: Score for search engine optimization.

Assets Tab

A screenshot of a graph

Description automatically generated

A screenshot of a graph

Description automatically generated

A screenshot of a computer

Description automatically generated

The "Assets" section of the dashboard is tracking the sizes and types of resources loaded by a webpage over time:

1. Content Sizes: Tracking the sizes and types of resources loaded by a webpage
2. HTML Bytes: The amount of data for HTML documents.

CSS Bytes: The size of CSS files used for styling the page.

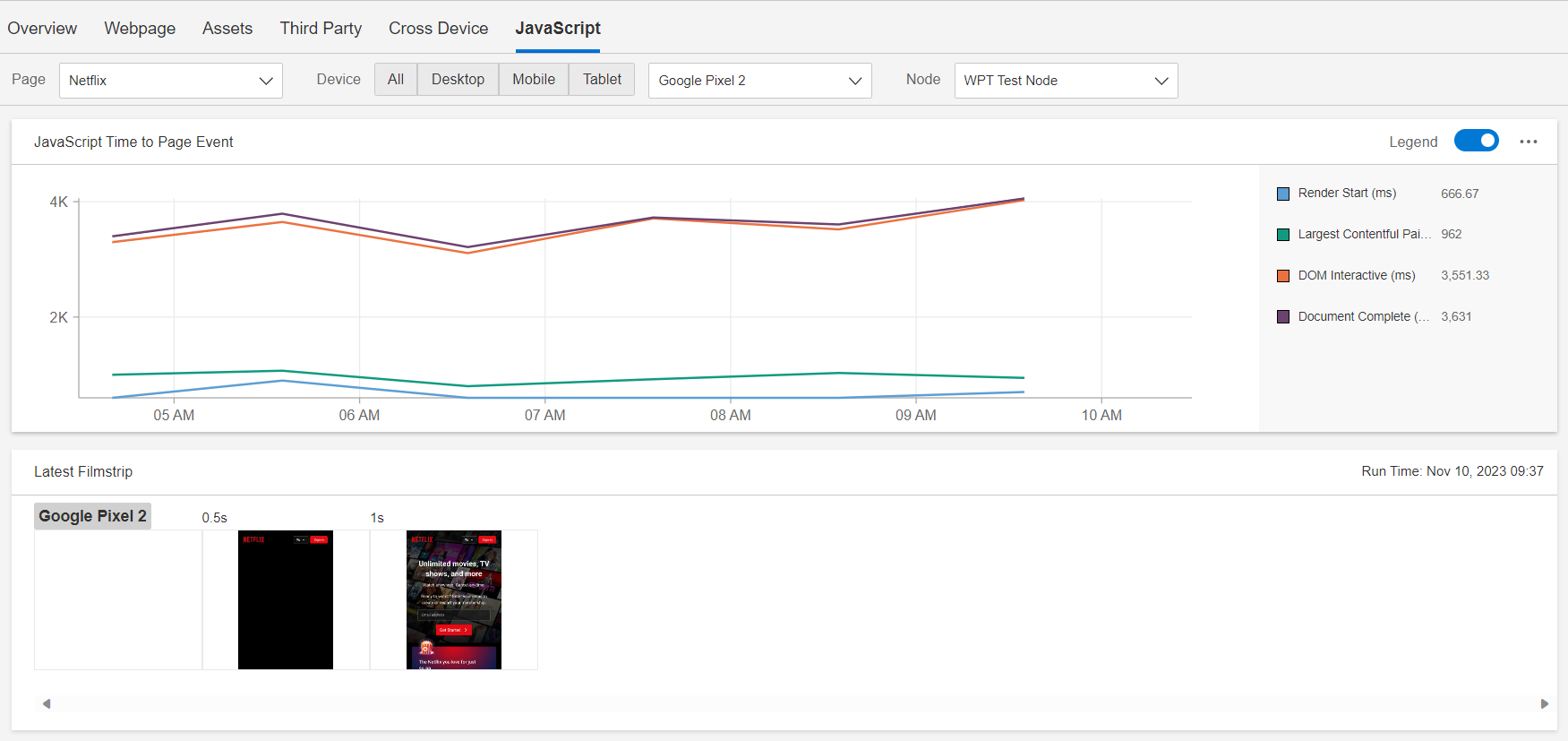
1. Script Bytes: The size of JavaScript files for functionality and interactivity.
2. Image Bytes: The data size of all images loaded on the page.
3. Media Bytes: The size of media files, such as audio and video, though it appears to be 0 in this chart.
4. Other Bytes: The size of other types of resources (like fonts or SVG files).
5. Total Requests: The "Total Requests" graph shows the number of HTTP(S) requests made by a webpage for all the necessary resources to load completely. This includes requests for HTML, CSS, JavaScript, images, media files, and other resources. Each file type's requests are color-coded and stacked, allowing you to see the composition of requests over time.
6. Blocking Requests: The "Blocking Requests" graph indicates the number of requests that are considered "blocking" — meaning they must be processed before the browser can continue with rendering the page. Typically, "blocking" refers to resources that prevent the page from being displayed until they are fully loaded and parsed, such as certain JavaScript scripts or CSS files.

* Below that it just shows the request made by browser to lead the page and those requests are separated into two different columns such as one column representing request based on their sizes and other with the CPU time.

Cross Device Tab: Its giving the same metrics as webpage but for particular device and lesser details.

JavaScript Tab:

1. JavaScript time to page event:



Graph is showing the time taken for certain page events to occur, with a focus on those related to JavaScript execution.

1. Latest Filmstrip: It show a series of screenshots captured at intervals during the page load process. This visual representation, often called a filmstrip, is used to see how a webpage loads and renders over time from the user's perspective.
2. Latest Script time by domain: The "Latest Script Time by Domain" chart displays information about the time taken by scripts from different domains to execute on a webpage. Here's an explanation of the key components:

Domain: This refers to the web domains or servers from which scripts are being loaded. Each domain represents a different source of scripts, which can include your own website's domain, as well as external domains for resources like libraries or third-party services.

Script Time (ms): This represents the time, measured in milliseconds (ms), that scripts from each domain take to execute. It's a measure of how much CPU processing time is used by scripts from each domain.

Date/Time: This is the timestamp when the data was recorded, allowing you to see how script execution times change over time.

By displaying script execution times by domain, this chart can help identify if any particular domain is contributing to slow script execution, which can impact page load times and user experience.