

# Customized Timing

## Overview

Empower your RUM (Real User Monitoring) measurements with a controllable timer that allows you to focus on the metrics that matter most to your application. Optimize performance management with customizable timing solutions tailored to your needs.

- **Auto-Start Timer:** Automatically measure the time difference between when the page loads and your defined endpoint, rather than the browser's default `onload` event. This feature is ideal for tracking page load times, gaining insights into performance, and ensuring a seamless user experience. Monitor the duration required for all necessary data to be fully loaded and ready for user interaction.
- **Manual Timer:** Measure custom events such as component load times, function execution durations, and interaction timings, to gain deeper insights into your application's performance beyond standard browser metrics.

Together, these timers offer flexibility and precision for effectively optimizing web applications.

## Auto-Start Timer

### Use cases

- **Actual load time measurement** - Track when all product images and details on an e-commerce page are fully visible, ensuring a precise reflection of the user experience.
- **Data loading completion tracking** - Monitor the duration from page load to the completion of specific network calls or data population. This provides critical insights for pages reliant on external APIs or subject to slow network responses.

### Examples

- **Add timing for the first click event** Measure the time from the initial page rendering to the user's first click. In this use case, the function calculates the time elapsed until it is triggered.



```
window.addEventListener('click', function handler(){
  window.removeEventListener('click', handler);

  CoralogixRum.addTiming('first-click');
});
```

- **Add custom timing** In this scenario, the user provides a value along with the function triggering.

```
window.addEventListener('click', function handler() {
  window.removeEventListener('click', handler);

  CoralogixRum.addTiming('first-click', 1000);
  CoralogixRum.addTiming('number-of-pixels', 35);
});
```

## Manual Timer

### Use cases

- **Track asset visibility** - Measure the time required for key images or assets to load and become visible on the page.
- **Monitor resource availability** - Track the time it takes for a video to become ready to play and when the user begins watching it.
- **Analyze form submission performance** - Monitor form submission durations to identify potential delays.
- **Monitor rendering time** - Measure the rendering time of a modal window after a user interaction.
- **Evaluate dynamic content loading** - Analyze the loading speed of dynamically generated content, such as search results.

### Examples

- **Track user login duration** - Track the time between two points by starting a timer with `startTimeMeasure` and stopping it with `endTimeMeasure`, using a unique name to identify each measurement. Labels can be added at the start for additional context.

```
CoralogixRum.startTimeMeasure('login', {
  paymentMethod: 'visa',
  gridRows: 4
```

```
});
```

```
CoralogixRum.endTimeMeasure('login');
```

## Limitations

- Duplicate or empty keys are ignored.
- Measurements left open are cleared when the session expires.
- Labels must be added when starting the timer and cannot be updated later.

## Custom measurement data

Your custom timing measurements are stored in the following fields:

- `cx_rum.custom_measurement_context.name` - The unique identifier you chose for your timer.
- `cx_rum.custom_measurement_context.value` - The value you assigned.
- `cx_rum.labels` - The labels you added for additional context.

## Examples

```
custom_measurement_context: {  
  name: "Number of users in grid"  
  value: 32  
}
```

```
labels: {  
  age: 27  
  name: "name"  
}
```

## Support

### Need help?

Our world-class customer success team is available 24/7 to walk you through your setup and answer any questions that may come up.

Feel free to reach out to us **via our in-app chat** or by sending us an email to [support@coralogix.com](mailto:support@coralogix.com).

## Was this helpful?

Leave your feedback here.

☐ **Yes**   ☐ **No**

Send

---

© 2025 Coralogix. All rights reserved.

Generated on: November 18, 2025

Source: <https://coralogix.com/docs/user-guides/rum/sdk-features/customized-timing/>