2021 Scroll Survey Report

- Noteworthy results
  - Overall satisfaction with web scrolling
  - Difficulty working with scroll
  - Importance of touch interactions
  - Difficulty of tab key or gamepad navigation
  - Learning touch-action
  - Cyclical scrolling
  - Are scrollable areas important
  - Carousels
  - Infinite scroll
  - Scroll-linked or scroll-triggered animations
  - Compete with built-in scrolling
  - Overall satisfaction building scroll interactions on the web
- Survey Takeaways
  - Compatibility
  - Education
  - APIs
  - Features
  - Resources

In April, the Chrome team released a scroll and touch-action survey based on top reported issues from the 2019 MDN Web DNA Report. The 2021 Scroll Survey Report is ready, and the Chrome team would like to share some thoughts and action items we've gleaned from the survey results. We hope these results will help browser vendors and standards groups understand how to improve web scrolling.

View the 2021 Scroll Survey Report.

Noteworthy results

The survey anonymously collected 880 submissions, with 366 answering every question.

While getting started with scrolling is one line of CSS, like `overflow-x: scroll;`, the surface area of scroll APIs and options is large, spanning JavaScript to CSS. The following results help to highlight the issues web developers are encountering.

Overall satisfaction with web scrolling

Question 27

45%

are **somewhat** or **extremely dissatisfied overall**
with **web scrolling**.

This question was placed near the end of the survey intentionally, after questions on 26 scroll use cases and features. From the response, it's clear that the web community struggles with scroll. Almost half of the respondents report a level of overall dissatisfaction.

We believe overall sentiment about working with scroll should not be this low. This metric needs to be changed; it's a clear signal there's work to be done.

Difficulty working with scroll

Question 2

43%

reported it's **somewhat** or
extremely difficult
to work with scrolling.

From our research, these difficulties come from the multitude of use cases for scroll. When we talk about scrolling, that might include:

- Positioning elements within scrollable areas
- Infinite scroll
- Scroll linked animation
- Carousels

- Scrollview padding
- Cyclical scroll
- Virtualized scroll

Missing browser features, complex JavaScript, and the need to support input modes including touch, keyboard, and gamepads, make all of these things harder.

Importance of touch interactions

Question 3

51%

report **touch interactions** as
very or **extremely important**
to their work.

With mobile web users still on the rise in visit statistics, it wasn't surprising to see half of the respondents report that touch is very important to their work on the web. This signaled that web features like CSS scroll snap and `touch-action` need extra attention so the web can deliver on high-quality touch interaction.

Difficulty of tab key or gamepad navigation

Question 5a

44%

report **somewhat** or **extremely difficult**
to do **gamepad** and **tab navigation**.

Scrolling includes navigation methods such as keyboard arrows, tab keys, spacebar presses, and gamepads, and it can be difficult to include these when doing custom scroll work. Almost half of the respondents report it's somewhat or extremely difficult to include these inputs.

Learning `touch-action`

Question 9

50%

report **learning** about
`touch-action: manipulation`
from the survey.

Some of the survey questions asked about using certain APIs with a possible answer of Yes, No, or "today I learned." One notable piece of feedback was the number of people who reported learning about `touch-action` from the survey, as it's a critical property when building custom touch gestures that need to interact within scroll.

Cyclical scrolling

Question 27

58%

report **sometimes, often** or on **every project**
using **cyclical scrolling**.

> The video shows cyclical seconds scrolling,
> after 60 seconds it begins at 0 again.

Those numbers are high for a scrolling feature with little or no support provided by the web platform. The feature often incurs high amounts of technical debt because of this, with duplication or JavaScript injected to force the effect. It's popular for product carousels and when selecting time in seconds or minutes to offer cyclical scrolling.

Are scrollable areas important

Question 2

55%

very or
extremely important

16%

report **not at all**
or **slightly important**

Respondents felt strongly about the importance of scrollable areas, giving another signal about the struggles required to deliver high-quality scrolling.

Carousels

Question 20

87%

have used carousels.

24%

report they're
easy to manage.

Nearly every respondent delivers carousels in their web work, while only 25% find them easy to manage. Off-the-shelf carousels were popular during our research, but this statistic surprised us, as it doesn't sound very solved.

Infinite scroll

Question 22

65%

use it **sometimes**
to **every project**

60%

somewhat or
extremely difficult.

Two-thirds of respondents deliver infinite scroll in their web work, and an equal amount report it's difficult to do. Another example of high usage paired with high difficulty, which indicates to us an area needing attention.

While `content-visibility` and `contain-intrinsic-size` can be combined to reduce render costs for long scrollable areas, it doesn't seem to be helping with "load more" infinite scroll UX.

Scroll-linked or scroll-triggered animations

Question 24

47%

use it **sometimes**
to **every project**

56%

report **somewhat** or
extremely difficult

Almost half of all respondents use scroll-orchestrated animations and half the respondents find it difficult, once again linking high usage with difficulty.

Compete with built-in scrolling

Question 26

32%

always or
most of the time

50%

sometimes

The built-in scroll and touch interactions of phone and tablet applications are often touted as a clear place where the web can catch up. The features include scroll-linked animations, programmatic interfaces, voice integration, scroll hints, and pull-to-refresh APIs.

Just half of the respondents felt it was only sometimes possible to match the experience of built-in scrolling.

Overall satisfaction building scroll interactions on the web

Question 27

A pie chart showing 5 sections: 6.3% extremely dissatisfied, 2.7% extremely satisfied, 23.4% somewhat satisfied, 28.8% neither satisfied nor dissatisfied, 38.7% somewhat dissatisfied.

Survey Takeaways

The survey results are segmented into four categories: compatibility, education, APIs, and features.

Compatibility

The Chrome team has declared a goal to decrease the number of web compatibility issues, including scroll compatibility.

The first three compatibility issues to focus on:

1. Horizontal scrolling compatibility.
2. `overscroll-behavior` cross browser.
3. Removing prefixes from `-webkit-scrollbar` and following the standard.

Education

The survey results showed that there needs to be more education around `touch-action` and logical properties. The browser is at the forefront of international layout, and it's apparent it's underutilized or misunderstood.

Areas to focus on:

1. `touch-action`
2. Logical properties

APIs

Usage of scroll snapping is growing, and developers have responded that they want to use features interoperably with popular libraries and plugins. Shrinking this gap between CSS and plugin libraries will help the satisfaction of scroll snap developer and user experience.

We'll focus API work on `scroll-snap`:

1. API availability and compatibility across browsers.
2. Begin work on new CSS APIs like `scroll-start`.
3. Begin work on new JS events like `snapChanged()`.

Features

The survey results showed that users struggle with some specific types of scroll-related components on the web, as the platform doesn't provide the primitives they need to build them without plugins or a high level of effort. This is an area that we hope to explore more deeply.

The features developers struggle to build include:

1. Carousels
2. Virtual scroll
3. Infinite scroll

Resources

- Scroll Survey Report
- Survey Announcement
- Mozilla DNA Reports
- Compat2021: Eliminating five top compatibility pain points on the web

Thumbnail image: photo by Taylor Wilcox on Unsplash.

□□□ #3
□ □□□ □□□ 2026-10-31 21:35:05 UTC
□ cocobook-sync □□□ 2026-11-04 17:36:26 UTC