

Supported platforms

Learn about the platforms, from modern to old, that are supported by MUI.

Browser

MUI supports the latest, stable releases of all major browsers and platforms. You don't need to provide any JavaScript polyfill as it manages unsupported browser features internally and in isolation.

Edge	Firefox	Chrome	Safari (macOS)	Safari (iOS)	IE
<code>>= 91</code>	<code>>= 78</code>	<code>>= 90</code>	<code>>= 14</code>	<code>>= 12.5</code>	11 (partial support)

An extensive list can be found in our [.browserlistrc](#) (check the `stable` entry).

Because Googlebot uses a web rendering service (WRS) to index the page content, it's critical that MUI supports it. [WRS regularly updates the rendering engine it uses](#). You can expect MUI's components to render without major issues.

IE 11

MUI provides **partial** supports for IE 11. Be aware of the following:

- Some of the components have no support. For instance, the new components, the data grid, the date picker.
- Some of the components have degraded support. For instance, the outlined input border radius is missing, the combobox doesn't remove diacritics, the circular progress animation is wobbling.
- The documentation itself might crash.
- You need to install the [legacy bundle](#).
- You might need to install polyfills. For instance for the [popper.js transitive dependency](#).

Overall, the library doesn't prioritize the support of IE 11 if it harms the most common use cases. For instance, we will close new issues opened about IE 11 and might not merge pull requests that improve IE 11 support.

v6 will completely remove the support of IE 11.

Server

MUI supports [Node.js](#) starting with version 12.17 (or 12.0 with `--experimental-modules` enabled) for server-side rendering. This aims to match the [LTS versions that are in maintenance](#) mode.

CSS prefixing

Be aware that some CSS features [require](#) an additional postprocessing step that adds vendor-specific prefixes. These prefixes are automatically added to the client thanks to [jss-plugin-vendor-prefixer](#).

The CSS served on this documentation is processed with [autoprefixer](#). You can use [the documentation implementation](#) as inspiration. Be aware that it has an implication with the performance of the page. It's a must-do for static pages, but it needs to be put in balance with not doing anything when rendering dynamic pages.

React

MUI supports the most recent versions of React, starting with ^17.0.0 (the one with event delegation at the React root). Have a look at the older [versions](#) for backward compatibility.

TypeScript

MUI requires a minimum version of TypeScript 3.5. This aims to match the policy of [DefinitelyTyped](#) with the support of the versions of TypeScript that are less than two years old.