

Happy New Year from MUI!

2021 has been another amazing year, not only for MUI but also for the React ecosystem and our users. The digitalization of the economy is accelerating, pushed by COVID-19. Developers and designers have been, more than ever, busy creating apps.

We spent our year making it faster and easier for you to build great UIs and to delight your users with amazing UI experiences. Thank you for being part of this journey!

Growth

It's only with your trust that we could achieve the following outcomes in 2021. Thank you!

- 📦 From 6.4M to [10.8M](#) downloads per month (from 18% to [23%](#) share of the React ecosystem).
- ✅ From 4.6M to 6.0M unique visitors per year on the documentation.
- ★ From 63.8 to 74.0k stars, leave us [yours](#) 🌟.
- 👥 From 2,052 to [2,396](#) contributors.
- 💰 Grew financial support by 2.9X in 2021, compared to 2020.
- 👤 From 6 to 14 full-time equivalent [team members](#).

The numbers speak for themselves. We have reproduced the [traction we got in 2020](#).

In review

When we started 2021, we had recently released the first version of the MUI X data grid and were in the middle of the release of MUI v5. We had committed ourselves to fully execute these two major efforts:

- On MUI Core, major releases have historically been efforts that span over a dozen months. v5 was no different as great work takes time. At the beginning of 2021, we had solved the most important aspect of v5, we had a new style engine to unlock a better styling and customization experience for developers. But we still had most of the work ahead of us, we had to roll out the improvements to all our users. With a large code and user base like MUI has, we knew it would take time. For example, we might have had migrated a few components to the new style engine and fix all the issues that we could find, going to production is unforgiving. Developers tend to surface all the cases that are overlooked in the first iteration with GitHub issues.
- On MUI X, we had successfully sold our first Pro plan licenses and reached 100k downloads/month on the MIT version. We had early signs from the community of a market fit for the product and that the [open core license model](#) was the right one to execute our mission. We were determined to solely focus on the data grid component, and demonstrate that our model was viable to build the best React data grid in the space.

What might have been the biggest change in 2021 is the implications of a larger team for the organization structure. We have reached an inflection point at around 10 people. I could no longer look at all the GitHub issues and PRs. At that point, it was requiring >70h/week to do it correctly. It was not scaling to have one individual aware of all PRs and GitHub issues. Continuing in this direction would prevent us to grow the team, and more importantly, it was distracting me from the core CEO responsibilities (people, vision, funding). So we have broken down the responsibilities We have introduced our first layers of management and defined more clearly the purpose of each role.

Achievements

We have achieved most of what we could have hoped for.

Company

- The most important, we have welcomed 10 [members](#) in the company: Matheus, Michał, Jun, Danilo, Flavien, Benny, Alexandre, Jan, Bharat, Prakhar.
- We have introduced a [public handbook](#) on how we run the company. For example, we are using the OKRs work methodology to prioritize our efforts.
- We held our first company [retreat](#) 🏠 in Lisbon, Portugal PT, for members of the team that were not prevented by COVID-19 related travel restrictions, and who felt safe enough to travel.
- We have [rebranded](#) the company to a clean 3 letters acronym: mui.com. This move was key for us to grow beyond Material Design.

MUI Core

- We have released [v5](#), 21% of the npm downloads are on this major version now.
- We have broken down the demos into smaller and focus on one aspect at a time.
- We have polished and promoted most of the components that were in the lab to `@mui/material`.
- We have introduced many new components:
 - [Masonry](#)
 - [Stack](#)
 - [Trap Focus](#)
 - [Unstyled button](#)
 - [Unstyled slider](#)
 - [Unstyled modal](#)
 - [Unstyled slider](#)
 - [Unstyled tabs](#)

MUI X

- We have released [v5](#), 42% of the npm downloads are on this major version now.
- We have introduced support for [custom locales](#).
- We have introduced many new features while consolidating the existing ones:
 - [Row editing](#)
 - [Cell editing](#)
 - [Column pinning](#)
 - [Tree data](#)
 - [CSV export](#)
 - [Print](#)
- We have grown the MIT downloads on npm and the number of licenses sold per month by a factor of x10 in 2021 compared to 2020.
- We have grown the size of the team from 2 to 4 people (6 in January).

Looking at 2022

I have been involved on MUI for 7 years and almost 3 years full-time now. It's amazing to see the progress the library has done [since then](#). But at the same time, I'm astonished by how far we can still push the mission forward. My experience has been that the more we improve the library, the more we envision future ways to do it better. It's still [day 1](#).

What's our mission? To empower as many people as possible to build great UIs, faster.

2021 was great, 2022 is going to be even more exciting! We will continue in the same direction. Here is a breakdown of our [roadmap](#).

MUI Core

The release of v5 has introduced a significant API churn in the community. While our [versioning frequency](#) aims for one major per year, we hope we can iterate on v5 during the whole year of 2022 without any breaking changes, e.g. taking full advantage of the new style engine.

MUI Base

To continue improving the customization experience, we will double down on the version of the components without styles: `@mui/base`. These components and hooks contain the main functionalities and accessibility, without being opinionated about how styles are applied or what styles. We still have work to do to have a full set of unstyled components.

You can read more about them in [the docs](#) and keep track of our progress in [#27170](#).

Our high-level plan is to use the unstyled components and hooks as the basis of the Material Design components and our second design system.

System

There a number of improvements that we can bring to the system (the npm package `@mui/system` used to create the styled components). Hopefully, we will bring native support for CSS variables, improve DX, and improve performance.

Material UI

Google is rolling out [Material Design 3](#), we have to determine how to respond to it. Should we implement it later on for v6? Should we ignore it? Should we implement Google's UI that looks cleaner instead? In the meantime, we will keep improving the components, increasingly relying on `@mui/base`.

Joy UI

We will continue [in the direction of 2021](#). While Material Design is a predominant design choice for building rich client-side applications, it's not the only popular design trend. We plan to provide a [second design system](#) to expand the range of use cases for MUI, striking a balance between:

- alignment with the best-looking design trends of hot startups in the US.
- optimization of its usage for rich client-side applications while keeping it good enough for landing pages.
- simplicity of the design for allowing simpler customizations.
- complementarity with Material Design.

It will be built on top of the unstyled components: MUI Base and our styling library: MUI System.

MUI X

The recent data grid product initiative is working, it's time to scale MUI X by growing the team and the product scope.

Data Grid

We will keep the data grid as our primary effort, there are [many features left to be built](#) 🚧. Our efforts will include: doubling down on the documentation, releasing the Premium features, and working on a hook-only API.

Date Picker

MUI Core had not enough bandwidth to work on the pickers components. So we will expand the scope of MUI X to the date picker components, taking ownership and resuming where the work was left.

Low-code

Pro-code (writing code in a regular programming language) has been an incredibly effective way to deliver on our mission so far. The two main benefits reported by developers in our 2021 survey for using MUI are "moving faster" and "ease of use". However, the data points we can find suggest that the demand for developers in the next 5-10 years is growing a lot faster than the growth of the workforce. The importance of these two benefits is only going to increase, it's critical for MUI to keep pushing the envelope of what development speed and ease of use look like for developers.

Entering low-code, we have a small team that will continue explore in 2022 how MUI can empower developers and knowledge workers to build apps more visually, up to x10 faster, with no advanced SQL or JavaScript knowledge. The goal is to write fewer lines of code, while still empowering developers to extend it with custom logic.

Scale

If we sustain the current growth rate in 2022, we should be able to grow the team from 14 to 40 members in the company by the end of the year (based on the self bootstrap revenues). The primary purpose will be to strengthen all the efforts that we have started, not starting any new ones until we reach a great level of execution on all the above.

At a high level we are guided by the following rock beliefs:

- React dominance in the UI development landscape will continue to increase in the next 5 years.
- Developers will always be looking to build faster, more appealing UIs, and with less specific knowledge.
- Low-code will progressively become mainstream for professional developers and knowledge workers.

If like us, you are excited about bringing joy to developers and enabling more people to build beautiful apps faster, check our positions, there are 8 open roles as of now, [we are hiring!](#)

You can help us shape these new packages by taking part in discussions. [RFCs](#) are waiting for your feedback. Don't hesitate to let us know what you think!

See you in 2022, thank for being part of this journey 🚀