Getting Started with Design Systems

Marvel

A free resource made by Marvel Labs

Getting Started with Design Systems

The power of design in everyone's hands.
Marvel provides a cloud based platform that automates the design process, allowing companies and individuals to effortlessly create user-centric digital products and experiences.

marvelapp.com

We've also created a space for our community to share all things design and product, from opinions to processes.

blog.marvelapp.com

Design and Art Direction: Filippo Chiumiento
Illustrations: Diana Stoyanova
Content: Yavor Punchev and Owen Williams

For more like this, visit:

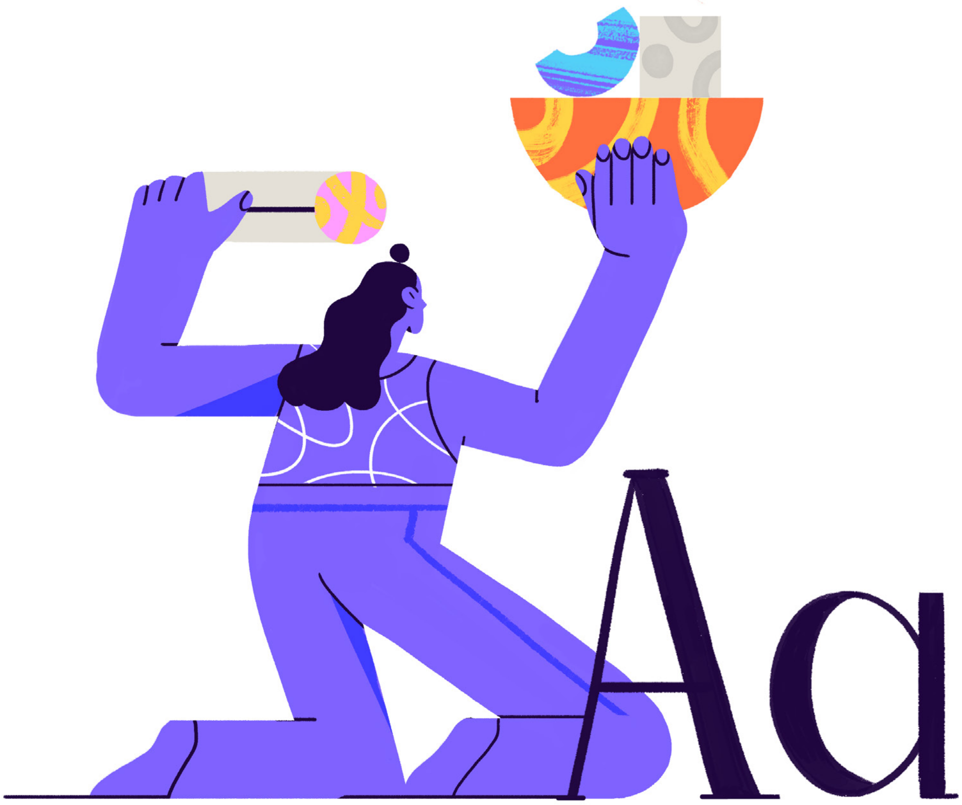
marvelapp.com/resources

Tables of contents

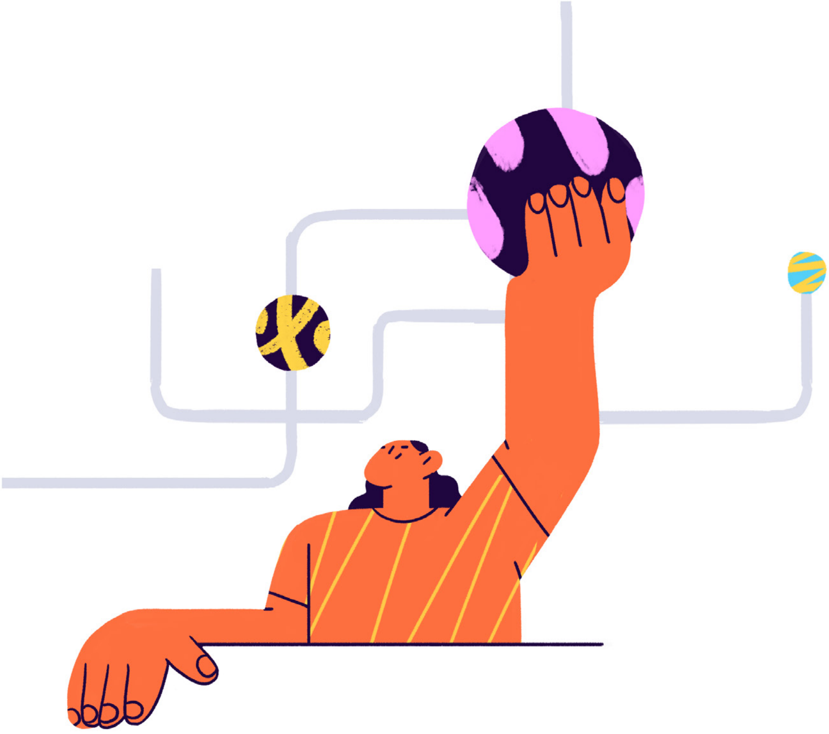
Introduction

As companies focus on design more every year, the concept of 'design systems' has risen in popularity — it's likely you're hearing about them all of the time. Put simply, design systems are a powerful way to help scale design-led development of products and services, making them an important tool to use as your team or business grows.

It can be daunting to build your own design system — particularly if you've never done it before—but it's easier than you think. We've gathered the information you need to start your design system, piece by piece, and how you can go about implementing it in your organization.



Design Systems, explained



A design system is a way to give your organization's design language a common framework that's understood by all, which helps to keep cross-functional teams on the same page. Design systems aren't really about the output, but instead describes a practice in which your team creates a continually-updated 'source of truth.'

Your design system doesn't need to be a final, end-to-end guide on day one, and should be considered a work in progress for the long haul. That means it doesn't need to be 'done' in a few days, nor is it ever truly completed.

> “
>
> *To grossly oversimplify: a design system is the single source of truth for digital products, and should build a place for every team to refer back to when they're creating anything to put into the world on a digital device.*

Design systems, as well as the structure they bring, are important for intentionally managing how visual design elements are implemented at larger companies. It prevents inconsistencies as new products, websites or services are introduced, ensuring that your components look the same regardless of where they appear.

A design system's actual output defines the building blocks of your product that are available on demand. Those blocks are ready to assemble into any final idea, whereas a style guide is a *byproduct* of your design system, not the output.

Additionally, a design system should include more context, such as the processes, tools and rules your team uses to implement new work, as well as the teams that will be involved and the ways you approach problems.

Putting Design Systems to work

As you build out your Design System, many teams decide to work towards integrating it into the actual process of how their products are implemented.

At Marvel, for example, our design team invested in embedding our Design System into the deployment processes, so that components are automatically unified across the board, simplifying future changes to the design language that would otherwise take days to implement.

Here’s a great example of a design system you may have heard of: Google’s Material Design, which was introduced in 2014.

Material Design was created to solve a problem within Google where many system applications on Android looked different, despite the fact that everyone worked at the same company. Material Design helped Google unify its design language across its products, from desktop to mobile, by providing a set of guidelines for the wider company to draw upon.

It was such a success that Google refreshed the entire Material Design system in 2018, and rolled out the updated style to all of its apps in just a few months. This was only possible because the components were clearly defined, and teams were required to reference them rather than build their own.

“Whilst your organization might not be Google-scale yet, you can still start your own design system — the sooner you do it, the easier it’ll be to maintain and implement organization wide.”

What are the outputs of a Design System?

As we've mentioned on our blog in the past, it's difficult to describe a Design System because it's largely **a conceptual set of rules, rather than an actual product**. Those rules define how your organization implements and makes changes to a design language, and what it looks like.

We can, however, distil the output of a Design System into a few key components:

1. Clear, up-to-date documentation
2. Guidelines, rules and styles that are easy to understand
3. Documented reasoning about why decisions were made
4. Tools and processes that help you implement the system on a daily basis

> *"It's important to remember at this point that design systems should not complicate or clutter people's workflows with confusing processes. If you find that it's constantly in the way of putting things into the world, it's likely time to re-evaluate your system and look for new ways to improve or optimize your system to help make people more productive."*

How to build your own Design System

Here at Marvel, we've embraced Design Systems wholeheartedly to unify our service regardless of where you might be using it.

We have a number of different products with separate code bases, each with their own team working on them. There's the main Marvel platform, our prototype viewer, design tools, Handoff, mobile apps for both iOS and Android, as well as our Sketch plugin.

There are other surfaces to consider beyond just the product: our marketing website, WordPress blog, email templates, and a number of back-end templates also need to be kept up to date.

> “
>
> *This list only gets longer as the company grows, so it's important to get everything in order early, and keep these updated when new products are introduced.”*

To start building your Design System, it's best to approach it as a set of individual tasks:

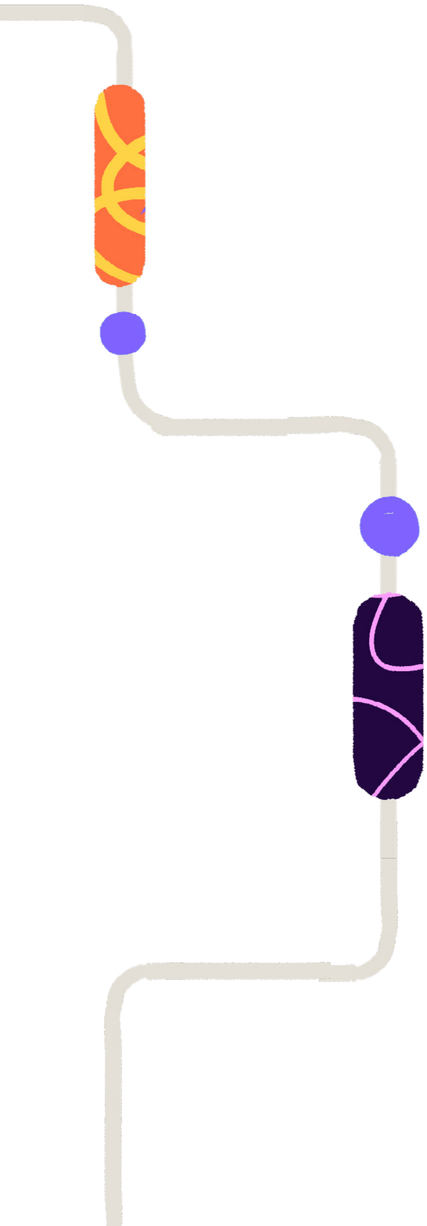
1. Take inventory of what you have today
2. Get your company onboard with your Design System
3. Establish the baseline components of your Design System
4. Define a color palette, typographic scale and every other piece
5. Standardize these across your products
6. Integrate the Design System with your process

Step One
Take inventory

Begin by creating an overview of how your company uses color, fonts, shadows, buttons and every other tiny piece of style. This is a painstaking process that takes hours to work through, but helps you begin to keep track of the random inconsistencies across your products.

At Marvel, we discovered that we had 18 slightly different shades of the same gray, and that colors weren't always balanced properly. By documenting these, we were able to take a first step toward unifying them.

> *"Brad Frost, a web designer and the writer of the Atomic Design book, says that your 'interface inventory' should be a "comprehensive collection" of every tiny piece of your product. Each item, such as buttons, should be documented and categorized, which will help provide you with a high level overview of every pattern in your product today."*



Once you're done with this categorization, you've got a powerful piece of insight into the areas that need addressing, and a great case for the next phase of building your Design System.

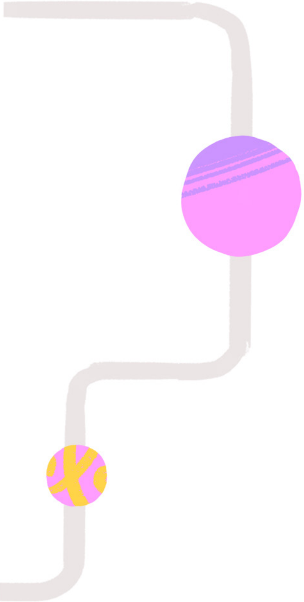
“An important part of building a Design System is getting your wider company onboard — without buy-in, it’ll be a much harder uphill battle in the long haul because **building a Design System isn’t a one time job.”**

Step Two: Get your team onboard

To really benefit from the design system, it'll need to be maintained, and kept up to date as your style evolves — which can become a point of contention.

By getting this buy-in and ensuring that the organization understand the value of a Design System, you're protecting your team when managers or executives later question why you're spending time updating it, rather than shipping products.

You should consult not only your direct team, but stakeholders in the wider organization as well. Explaining the benefits and the challenges you're actually solving will help get them onboard with building a Design System. The inventory you've created and how you'll use the Design System to get more done is an important tool for explaining the value of why a Design System is worth having at this stage.



This process should also help decide who will be responsible for championing the Design System, as well as maintaining it, to ensure it doesn't end up on the shelf when it gets too hard to keep up to date.

> *"As we've already mentioned, a Design System is an ongoing investment, and will change over time, so you'll need to actively invest in it. One great tool for embedding this mindset within a team, as Nathan Curtis, the founder of UX Firm Eight Shapes points out, is tying it to your team's goals: "A system's value is realized when products ship features that use a system's parts.""*

Step Three
Choose a minimum viable system

It’s a great idea to establish a baseline of what your initial Design System will cover, so you know where to begin, and it’s clear where the work ends in the first iteration. This will help you focus, and keep moving toward the end-goal as the work gets difficult or tedious.

Airbnb is a fantastic example of an organization that’s embraced design systems, and the company started by creating a basic style guide, the baseline components it uses, as well as the overarching library that it would maintain going forward. Marvel’s own design system covers similar elements, extending to code guidelines, utilities and beyond.

> *“At this point, before building or changing anything, it’s important to decide whether it’s worth building a new system and visual style from scratch, or if it’s better to base it on an existing product. If there’s a way to use your existing product, or even a part of it, as a baseline, it’s much easier than defining this from the ground up at the beginning, which increases the project’s scope dramatically.”*

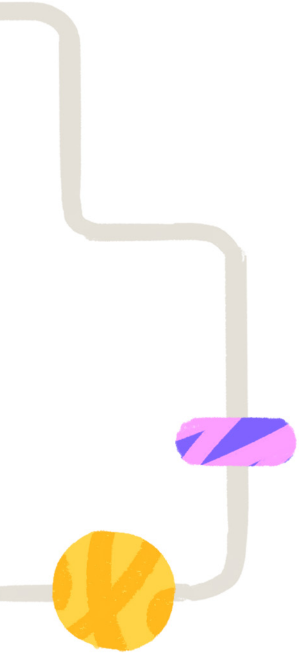
You should also decide *how* the Design System will be implemented. It may require new technology, a shift to component-based tooling like Vue or React, or even a change to how your team releases products into the world. It’s worth pausing and considering this early, before committing to a way of working.

Step Four
Start building

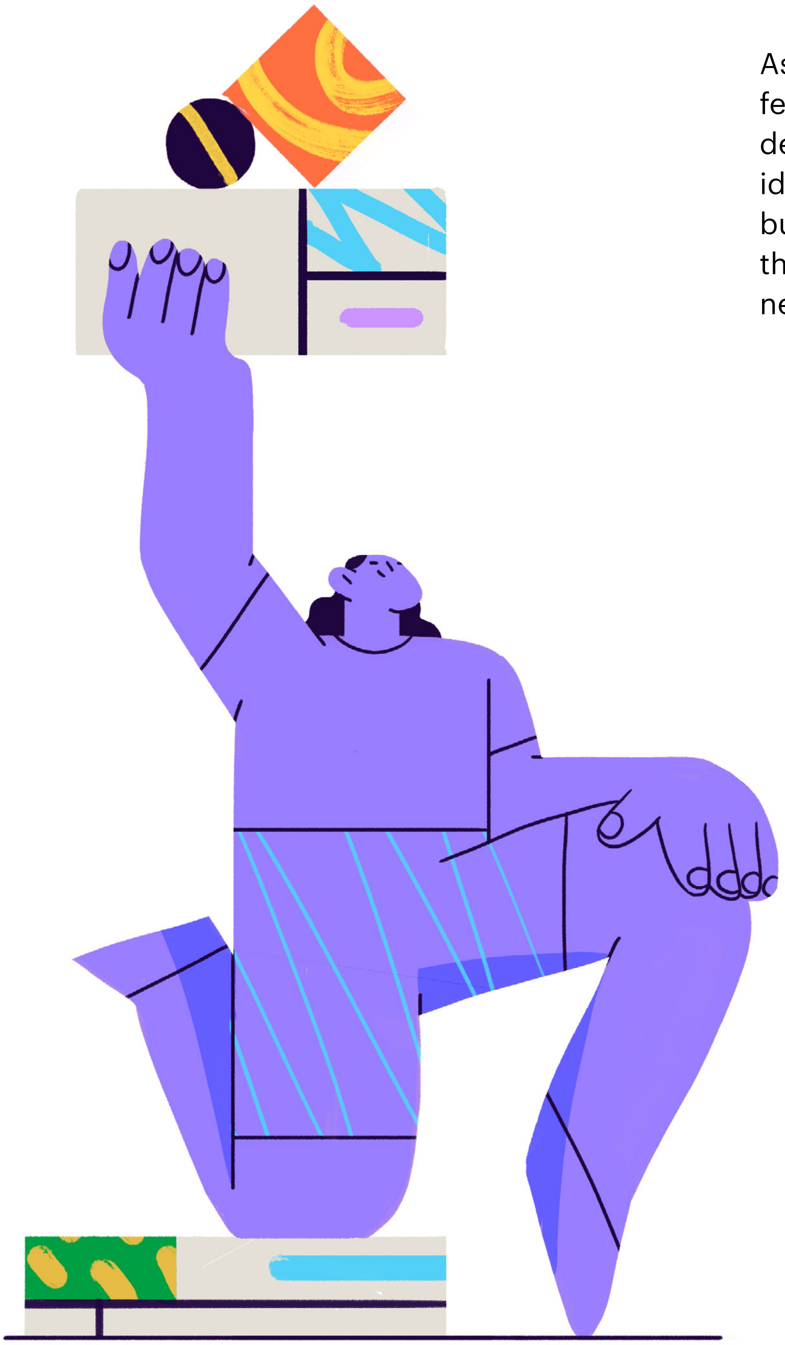
If you've made it this far, it's finally time to start constructing your Design System! Once you've got all the bits inventoried and accounted for, it's time to start defining your color, typography and components across the board.

You can build this as a UI kit in a tool like Sketch or Marvel, then store the versioned files in Abstract to ensure they're tracked as they change over time, or define it in code that's compiled back into Sketch files. A tool like Marvel is a great way to hand-off the components between your designers and developers, to keep both sides on the same page.

> *“The most important part of this step is covering all of the different ways your components might appear in a product. If you're defining a button, for example, all of its states from disabled to pressed should be outlined as well.”*



When everything is defined and the system is locked in, start pushing those changes out across your products with discipline. Some companies choose to update their visual assets only when they're already addressing a particular feature of a product, but it's much easier to clean things up as an intentional, wider exercise.



Because you created an inventory at the start of this process, you should have a clear list of places that need attention. This list can be used to track the state of the Design System's roll-out, and whether or not it's properly implemented, making it the ultimate source of truth.

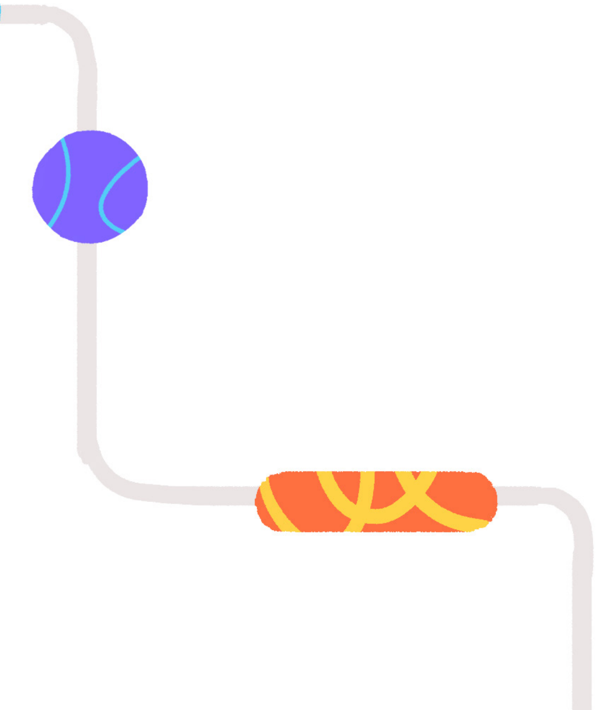
As you begin work on new products or features, use your Design System to guide you. Don't let it get in the way of the ideation process as a rigid set of rules, but instead consider it a living document that can be added to when new components are designed.

Step Five
To automation and beyond

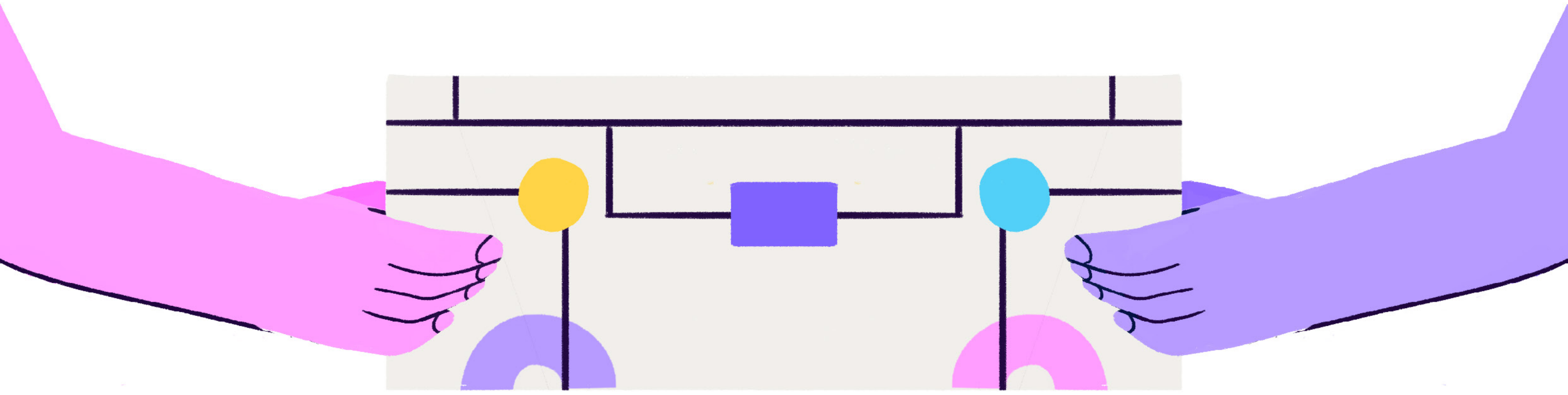
An important part of making a design system easy to use is helping integrate it into the ways you work, and will help set it up for success. Automation in both design and engineering is the ultimate way to keep your components up to date as they change over time.

There are varying schools of thought on how you should go about doing this. Airbnb has experimented with automating its design system end-to-end, building tools that help reliably produce consistent outputs and ease the hand-off process.

Marcelo Somers, User Experience Designer and co-creator of PatternPack outlines the pros and cons of integrating the design system into a codebase, and is a great guide for deciding which level of integration you're willing to strive for.



Behind Marvel’s Design System



At Marvel, we've deeply embedded our Design System at the core of our product creation flow, which automatically turns into code (and vice-versa) as changes are made on either side. This helps keep the system updated without needing constant manual changes, and means it's always the 'source of truth' for our company.

We use a Sketch UI Kit as our source of truth, which is then compiled into CSS and React Elements for use across the product. This leverages a tool called HTML-SketchApp to create Sketch files from CSS, and vice-versa, which helps keep both sides up to date as the system evolves.

From there, we use Marvel to hand our designed components off to developers, who are able to grab the real CSS, measurements and everything else they need to implement the final solution. When it's time to get the changes into the real world, our automated Git workflow pushes it into version control, which then takes care of the rest.

We also created a custom library called Swift Style Guide Generator to automatically generate a style guide for our iOS app from a JSON data file, reducing the risk of inconsistency when implementing the app.

This whole process ensures that our styles are uniformed, and major changes to colors or typography are much easier to implement in a single release. We mentioned the ideal Design System should get out of the way, and we've optimized wherever we can avoid manually changing things along the way.

This flow chart may help you understand the moving parts of this process. It might seem intimidating, but it's a simple way to illustrate how our Design System is transformed into a component users actually see.

We're experimenting with ways to embed our Design System even further into the way we work, using frameworks like Styled Components and Styled System to further modularize our components.

Your Design System may look different

Design Systems are not a one-size-fits-all solution, so you may find that ours looks different from yours! We recommend beginning by taking inventory, and opening the discussion with your team before jumping in head-first; you can't do it alone, and starting the process will quickly reveal the importance of building a design system for your team.

There's lots of other great reading about design systems, including the Design Systems book by Alla Kholmatova, and Brad Frosts's Atomic Design. We encourage you to explore building your own design system; it'll help unify the tools you use and build a framework that makes the design process easier, not harder.

Marvel

For more resources visit:
marvelapp.com/resources