



Guide To: Building an MVP Design System

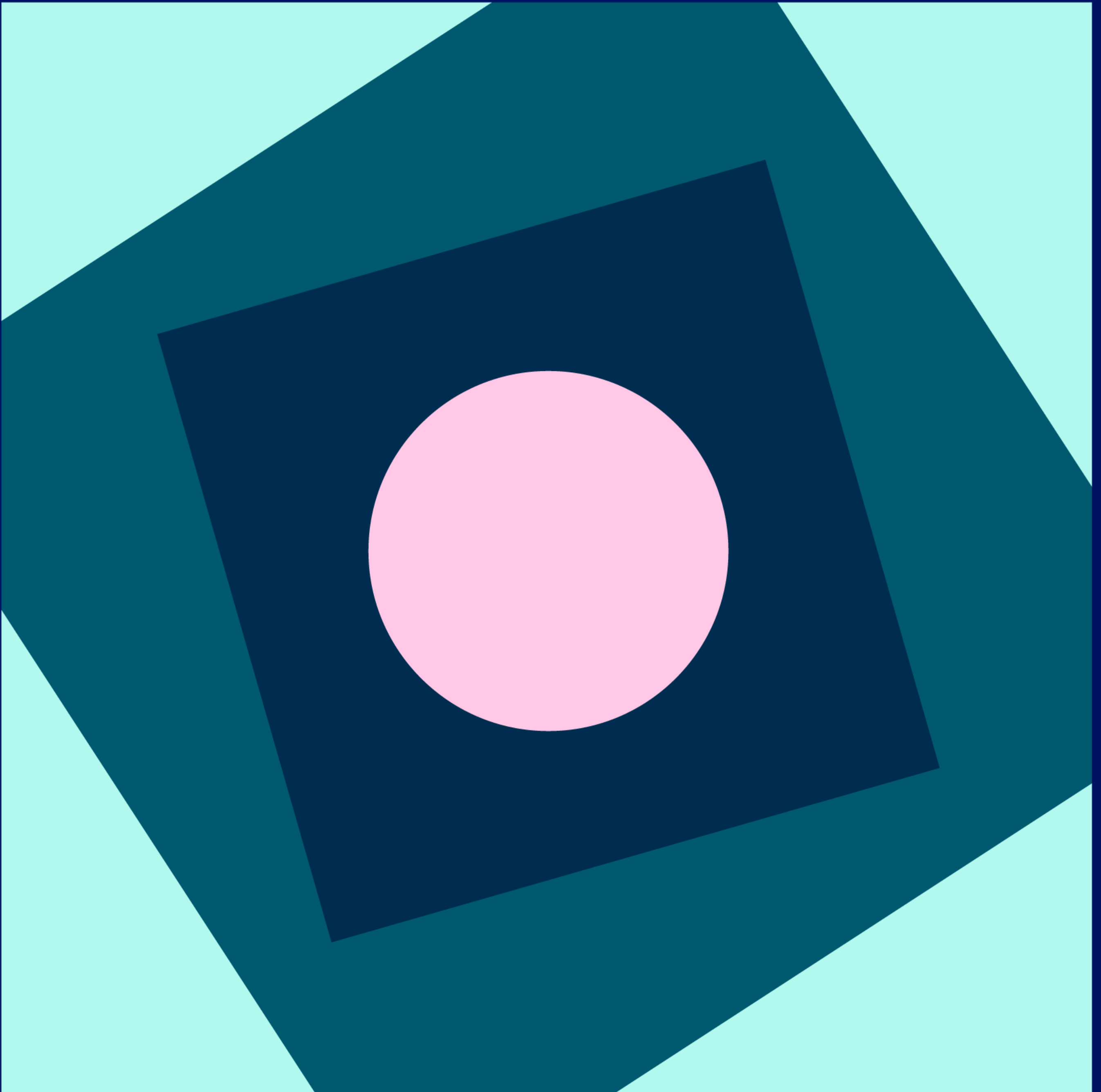


Table of Contents

Introduction	3
Starting an MVP Collection	10
Building an MVP User Base	15
Building an MVP Contribution Plan	25
After MVP Design System Launch	29
Conclusion	30
Contributors	32

This guide is for scrappy design system builders – designers, frontend developers, and other product workflow hackers – who are ready to reap the rewards of a design system but may be starting out with fewer resources and less support than they'd like.

Introduction

The term 'minimum viable product' (MVP) refers to the first of many user-facing iterations. The magic of the MVP idea is the way it helps product teams start small and prevent scope creep. Once a team has defined their MVP, they can hold themselves and their leaders accountable to releasing at a clear cutoff, so they can iterate based on real user data as soon as possible.

It's an especially good way to think about a new design system, because design systems are notoriously complex, sprawling across multiple tech stacks, interconnecting, growing, and evolving more rapidly than other products. Without a clear plan for an MVP design system, it can be tough for teams to begin and even tougher to launch.

Here you'll find a framework to **narrow the scope and build momentum for an MVP design system**. It'll provide steps to create a modest, approachable, but still viable design system with real users and real benefits.

Why start a design system now?

If you're reading this, you probably know that a design system can deliver major benefits, including:

Consistency Unified experiences across products, platforms, regions, and teams.

Scale Company-wide application of code and design norms (including compliance and accessibility standards) and sustainable workflows for all—no matter the number of designers and developers, nor their distance.

Speed Faster time to market, reduction in development and design time, and more room in the day for real innovation.

Why not now?

The reasons not to build a design system can seem just as compelling:

No resources You have no design system buy-in from executives, and therefore no dedicated staff, no budget, nor a divine mandate to support the effort.

No time You have zero sprints to spare in the wash of competing product and brand priorities.

No confidence You're overwhelmed by the potential size and scope of building and maintaining a high-functioning design system, and you don't know where to begin.

These barriers are normal. Start anyway.

Right now might seem like the most inconvenient time to build a design system, but it could also be the most crucial time for your business to have one. When market disruption and competition intensify, that's when consistency, scale, and speed become even more vital to success.

Here's the good news. Our conversations with leading design system builders reveal that, regardless of company size or industry, starting in less-than-ideal conditions isn't a show-stopper. In fact, it's normal.

It's normal to start a design system with zero full-time design system builders.

It's normal to feel that design system work doesn't fit into your current product timelines.

And **it's normal** to feel unequal to the task of building something so comprehensive, universal, and useful to so many teams, products, and edge cases.

Frankly, it would be weird if you didn't feel a little awed by that.

Despite these barriers, people still build design systems that revolutionize the way their teams ship products. They succeed because they've uncovered a secret:

Success doesn't depend on you alone. It's not about your ability to build a massive design system, and then maintain and grow it yourself.

Success depends on you and everyone else. It's about teams and their leaders buying into a thoughtful, sustainable plan that will make their work better, faster, and easier.



Joe Galliford, Designer
IBM iX

CASE STUDY

Yes, you're qualified.

Joe Galliford joined IBM iX as an intern, fresh out of design school. He had no formal design system training or experience, but his first assignment was to create a design system for a multinational client that would unite the work of four teams scattered across continents. He used **InVision Design System Manager** as a best-practice template, and spun up a functioning collection of foundations and components in a matter of days. A few months later, teams were seeing a 50% increase in workflow efficiency.

Joe's advice:

“ Don't worry about getting it perfect right away. Even if you don't have every icon yet, just by having a few components and guidelines available in DSM you're going to start seeing results. ”

This guide is going to help you develop the mindset you need, so you can start strong and help everyone create a truly viable MVP design system.

Let's begin by eradicating a pervasive MVP design system myth.

A collection is not a design system.

Many teams visualize an MVP design system as a collection of foundations and components. They believe if they build a good library, everyone will use it.

A collection is a powerful asset, but it's only one third of a design system. If you want broad adoption, you'll need a solid understanding of designer and developer workflows. And if your design system is going to grow and evolve, it will need growing, evolving guidelines and governance.

What is a Design System?

A collection of ↓	Used by both ↓	Controlled with ↓
<p>Foundations Basic building blocks</p> <ul style="list-style-type: none"> Colors Typography Icons Spacing Grids Motion <p>Components Reusable UI elements made with foundations</p> <ul style="list-style-type: none"> Buttons Cards Input fields Pagination Tabs Tables 	<p>Design Assets created and used in a design tool</p> <ul style="list-style-type: none"> Design Files Libraries Styles <p>Engineering Implementation in front-end development</p> <ul style="list-style-type: none"> APIs Components Tokens 	<p>Guidelines Written usage, rules and recommendations</p> <ul style="list-style-type: none"> Accessibility Examples Patterns Principles Usage Voice & Tone <p>Governance Adoption, maintenance processes, analysis</p> <ul style="list-style-type: none"> Communication Contribution Incentives Measurement Reviews Versions

A collection is only one-third of a design system. To be successful, teams need to include the other two-thirds... the people part. *Source: Guide to: Benchmarking your design system*

It's natural to want to focus on libraries. You can see a library. You can organize it neatly and make a thousand satisfying choices. This blue, not that one, this font with this exact size and spacing in this button state.

It's much harder to picture the more intangible concepts like user needs, workflows, and contribution models... in short, the heart of a living design system.

People bring it to life.

The rest of the definition – the people side – is what animates the entire design system. People have to invest in it and follow a plan to make it grow, evolve, and ultimately transform the entire product workflow. They also have to want to use it. Otherwise all you have is a nice collection.

That means your plans for an MVP design system must include and prioritize people.

What is a Design System?

A collection of ↓	Used by both ↓	Controlled with ↓
<p>Foundations Basic building blocks</p> <ul style="list-style-type: none"> Colors Typography Icons Spacing Grids Motion <p>Components Reusable UI elements made with foundations</p> <ul style="list-style-type: none"> Buttons Cards Input fields Pagination Tabs Tables 	<p>Design Assets created and used in a design tool</p> <ul style="list-style-type: none"> Design Files Libraries Styles <p>Engineering Implementation in front-end development</p> <ul style="list-style-type: none"> APIs Components Tokens 	<p>Guidelines Written usage, rules and recommendations</p> <ul style="list-style-type: none"> Accessibility Examples Patterns Principles Usage Voice & Tone <p>Governance Adoption, maintenance processes, analysis</p> <ul style="list-style-type: none"> Communication Contribution Incentives Measurement Reviews Versions

The other two-thirds of a design system—the people part—brings a design system to life.
Source: Guide to: Benchmarking your design system

We'll explore an approachable way to scope the people work, so that it doesn't seem quite so murky later. But first, let's scope out the easier bit.

The easy part: Starting an MVP collection

Since we're aiming for a quick start, the mantra is "Borrow, not build." Your first library iteration can launch much sooner, with far less deliberation, if you...

Draw on momentum and resources already at play.

Take advantage of a brand redesign or new product version. The project is already approved, funded, and underway. All that remains is adding those redesigned foundation elements and components to the fledgeling design system collection as they are greenlit for product. It's a small dose of extra effort with a big payoff.

Draw from the most fully-baked product in the portfolio. This helps you avoid arguing over the minutiae of colors and button styles, because you're working with precedents already backed by broad consensus and daily use. When choosing between two similar components to add to the collection (e.g. two types of tables) choose the one with the most complex, demanding use case.

If you have no good source for pre-developed components and styles to draw from, consider an open source library like [Google Material](#).

in DSM Scope tip

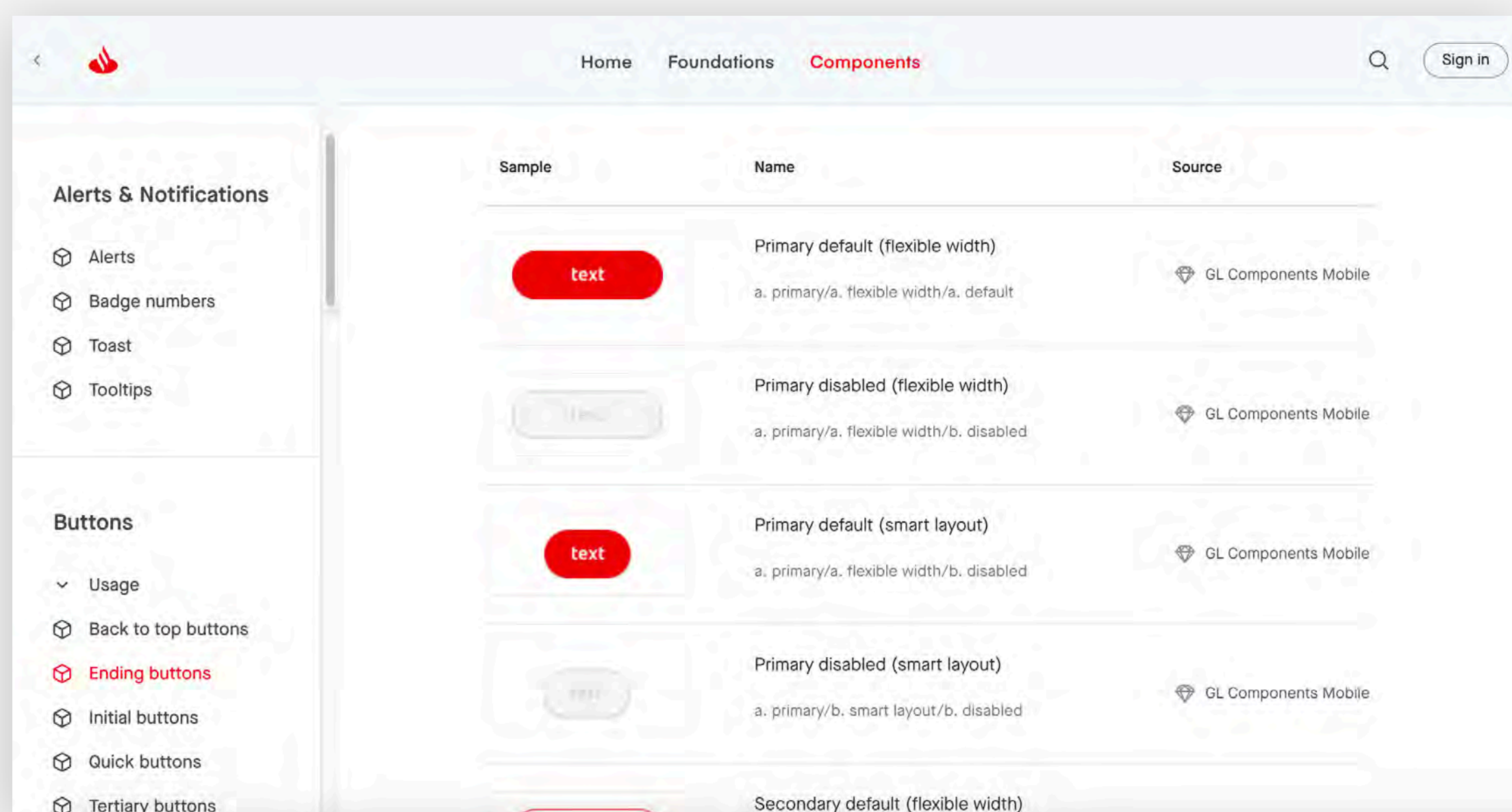
InVision DSM provides a templated place to organize your collection. You can also use it to scope your work. Limit your first collection effort to these DSM sections:

- ✓ All *Foundations* sections
- ✓ Only three *Components* groups



“I recommend people include buttons as one of their first three component groups. People assume buttons will be easy to manage, but there are so many button state variations that it’s actually a really good place to start. Other components will seem simple after that.”

– Zoe Adelman, Design Systems Consultant

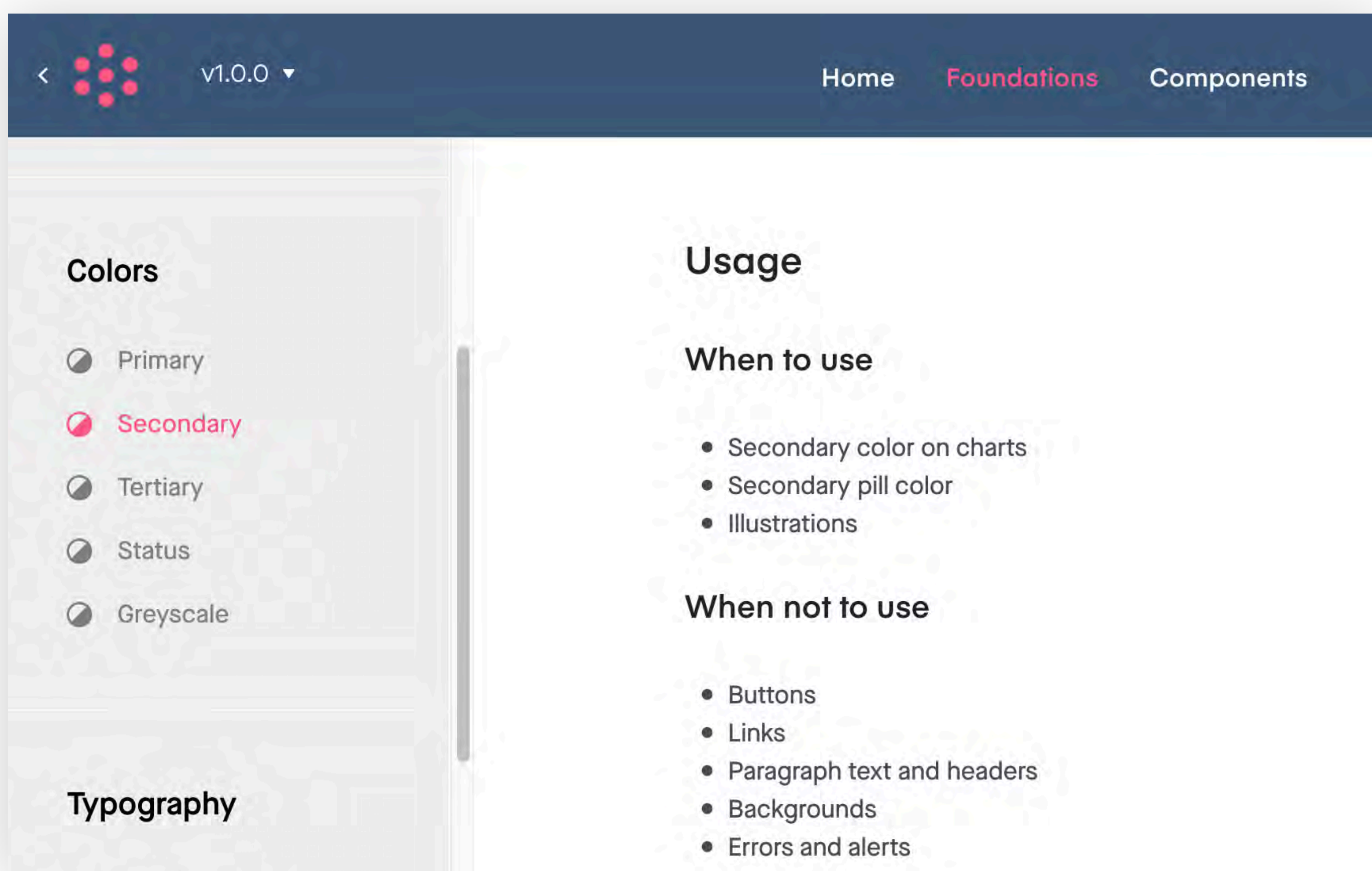


Talk to an expert like Zoe about how DSM can help

Add documentation. You don't need to have every bit of documentation loaded in your MVP collection, but you do need the minimum. Without that it's only a sticker sheet.

Each of your three component groups needs at least:

- ✓ A definition
- ✓ Do's and don'ts
- ✓ Rules for accessibility and compliance

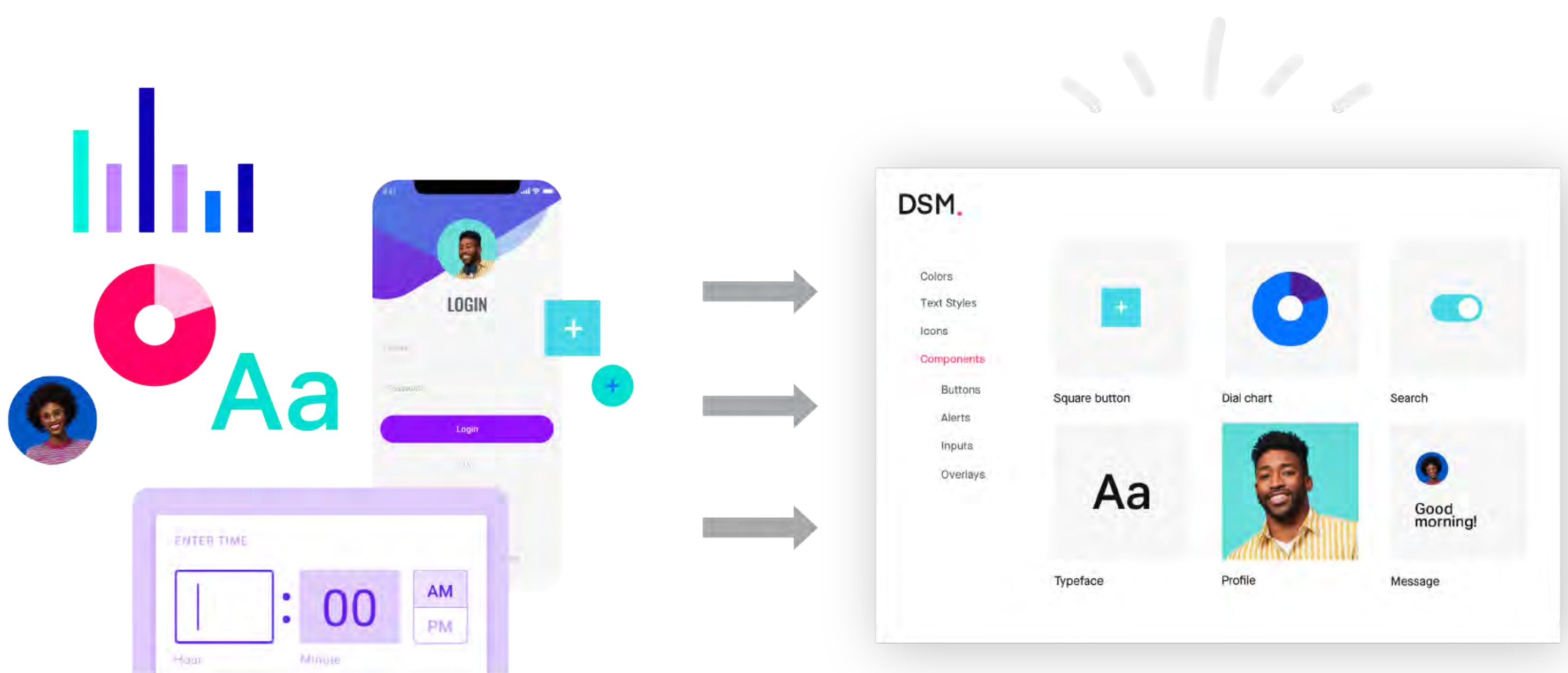


The rest can be added as you go, after launch.

Recap: Creating an MVP collection

Remember:

- ✓ Find components already in play
- ✓ Draw from the most fully-baked product in the portfolio
- ✓ Consider an open source library like Google Material
- ✓ Aim for iteration, not perfection



Success looks like this:

You take advantage of existing momentum to start a strong collection of borrowed foundations and components, with minimum documentation.



Joe Stutzman, Art Director
Host Gator

CASE STUDY

No dev help? No problem.

Joe Stutzman, Art Director had been wishing for a design system for his team at [HostGator](#) for years, but lacked developer resources to build a monument site.

“Other teams have to call in a dev every time they need to update something in their design system,” he said. “I don’t want to commit developers to that.”

The latest version of [DSM](#) was exactly the tool he needed, empowering him to publish and version a design system without any developer help. “I was able to quickly get something out in the world – something I could demo to show people how a design system changes things, how you can create a new page in ten minutes.” Now he’s using that momentum to gain executive support and ultimately to get the resources he needs to hook up coded components for developers.

Joe’s advice:

“You’ve got to go in with a mindset to get your hands dirty. Put it on paper, mess with it, break things, iterate.”

The people part: Building an MVP user base

Without users, a design system is dead on arrival. But you don't need a large user base from the start. For the MVP, focus your efforts where they can have the biggest long-term impact: with executives and developers.

Prepare an executive sponsor (or three).

An executive sponsor can help you secure time, tools, staff, and users. But for the MVP stage, you only need support for an experiment.

Instead of attempting to get buy-in for your dream design system right away, aim to stoke executive curiosity. Later, you can wow them with real results and make a bigger request.

Here's how to spot an executive sponsor and prepare them to help:

Find an influential leader, one who has plenty to gain from improving design and development workflows. Likely candidates include:

- ✓ **A product leader** assigned to the product you're using to kickstart the MVP collection
- ✓ **A design leader** with budget and/or sway
- ✓ **A frontend development leader** who benefitted from design systems in a prior role, or who preaches the wisdom of reuse instead of building from scratch

Frame the potential impact in terms of success metrics they care about.

For a product leader, that would be higher quality, sustainable consistency, and speed for future iterations.

For a design leader, it could be more time to invest in big-picture user experience work, and less time moving pixels – plus consistency between products and teams.

For a development leader, it's usually about increasing speed and reducing frustration. Ask questions and listen hard to pinpoint the right angle for the right person.

Sample impact framing script:

"We're already spending X developer hours and X designer hours to get X product out the door. If we invest just 5% more energy and add the work we do into a reusable system, we could save ourselves half the time down the road."



Tessa Rodes, Senior Design Manager
IBM

CASE STUDY

Latch onto a big executive goal.

Whenever possible, the IBM Cloud team used Carbon, IBM's central design system, one of the most advanced in the world. But Tessa Rodes and a few teammates hoped to develop their own branch of the design system. They called it a pattern asset library (Cloud PAL). It would draw from Carbon, but it would include special components, variations, and guidelines designed to cover the unique needs of IBM Cloud product offerings.

Even though designers and developers understood the value of working with a design system, the would-be Cloud PAL builders struggled for six months to find enough time and momentum to launch their MVP system.

Their opportunity arrived in the form of an announcement from the VPs of product and design. "They said every product in our org had to use Carbon 10 within the year, meaning everything had to be updated to look and feel the same across all of Cloud. We latched onto that and said, 'Let's not make this a one-off redesign; let's do this in a smart, sustainable way.' That unlocked all the doors."

Tessa's advice:

" Groundswell efforts can get you halfway there, but it's hard to reach true consistency without a clear directive from the top. Without executive support and a view into their strategic timelines and goals, you end up aligning only the pieces that were already flexible, and you lack the ability to escalate those large, complex pieces where inconsistency is most likely to hide."

Scope expectations. Make sure the leader understands this is the MVP design system, just phase one of a game-changing product with no expiration date. You can promise the MVP will deliver enough functionality to test and prove effectiveness, but remind them it's not a comprehensive solution—yet.

Your message: "Like any new product endeavor, this is an experiment."

Don't stop at one. Think of this as building an executive coalition, not winning a single supporter. Reach out to leaders from product, design, and business, so that support for the design system is built on more than one pillar.

Nurture a developer fanbase.

It may seem backward to prioritize developers as users of your fledgling design system. After all, the system's coded side might not even be in scope yet. Here's why you should look at developers first anyway:

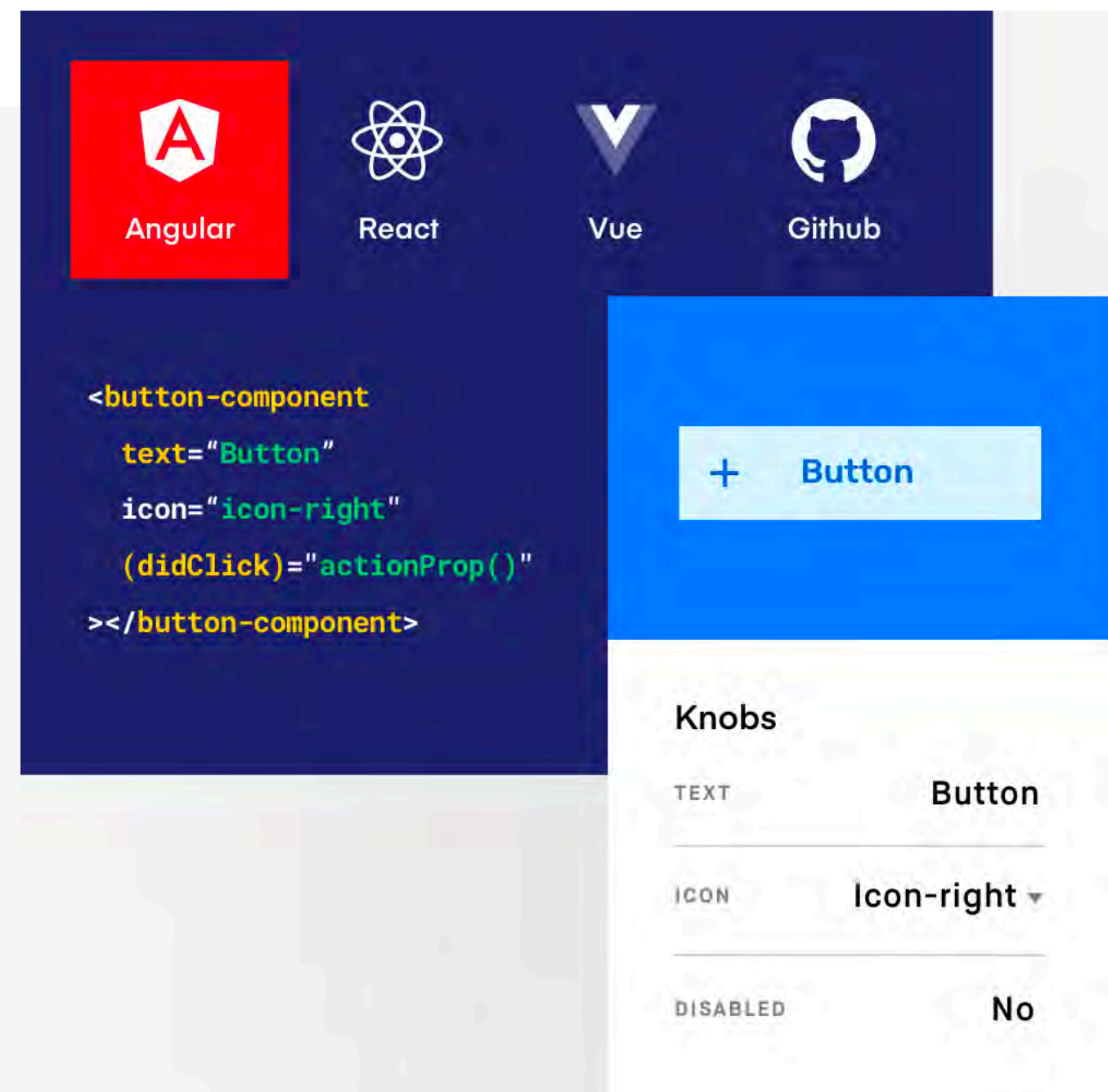
- 1. Developers are more expensive and more numerous.** Save a designer one hour of work, and you've done well. Save 100 developers one hour of work, and you've done miracles. Translation: Dollar sign success metrics to shore up your business case.
- 2. They're underserved.** If your design crew has been around for a year or two, you probably have a few reusable components and guidelines in place. It's far less likely that you have linked those same components to reusable code and guidelines for developers. But that is precisely what could transform their entire workflow.
- 3. They're disconnected.** Developers often work on the other side of the globe (or just the building) from designers. A design system keeps everyone in lockstep, no matter their locations.
- 4. They care about quality, speed, and efficiency.** Happily, these are all outcomes a design system delivers.
- 5. They're invaluable partners.** Developers can help you scale the design system to the hard reality of all your digital products – code. On top of that, they hold the keys to historical and current product architecture, important knowledge when you're prioritizing and standardizing components.

in DSM Collaboration tip

InVision DSM offers a [Storybook integration](#), linking design components to coded components in one shared hub for designer-developer truth.

Components appear alongside the documentation, variations, and guidelines devs need. And with [InVision Inspect](#), devs can easily spot existing components in specs, so they don't waste time building something that already exists.

[Talk to an InVision expert about how DSM can help](#)



Let's walk through some lightweight methods you can use to begin building a rabid fan base of developers before your MVP launches.

Turn user research into guerilla marketing. Like any product, a design system requires research into user workflows and needs. You can double the benefits of your research by asking developers questions that build expectation and curiosity.

Sample questions for developers:

What types of things do you find yourself building again and again?

Walk me through your workflow from the moment specs arrive.

Where do you spend the most time? Which meetings or messages feel redundant?

What's the most frustrating part of the process?

How do you find the compliance guidelines, assets, details you need?

If you begin without preamble, this line of questioning inevitably causes developers to ask, "Where's all this going?" And that's when you can paint the vision of design system utopia.

Give a teaser of the end product, not the MVP. It's okay at this point to blast far beyond your initial design system's scope. The goal is to help developers catch your vision for the future, when the fully-fledged design system has all the core components and guidelines they need to build products quickly, with superior quality and consistency, and pass QA in one go.

This is also a good time to float ideas about how developers might contribute new changes and components as part of their natural workflow, and ask for early feedback on your governance plans.

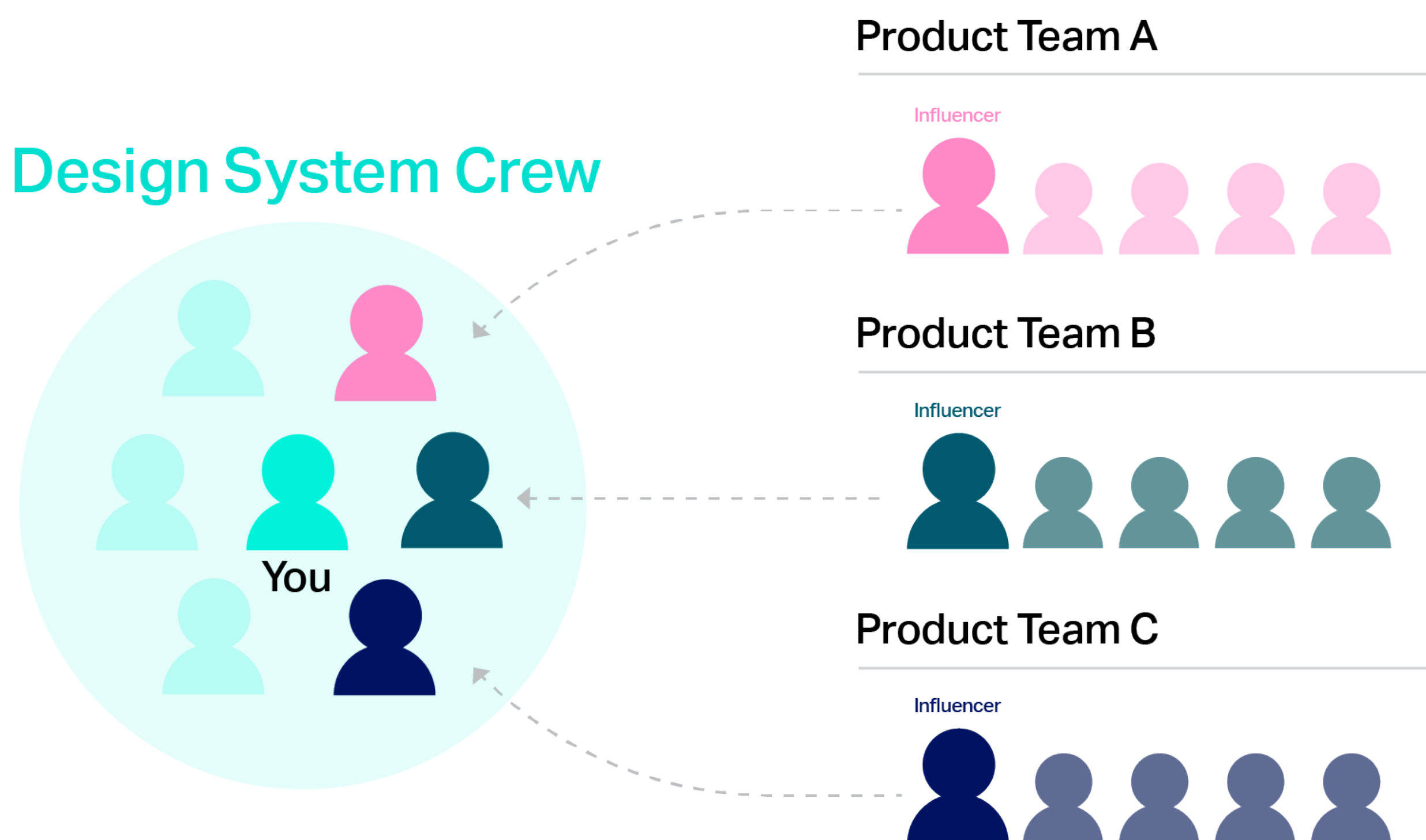
Once they've seen the light, it's time to tell them about your MVP effort, and make your request...

Ask new enthusiasts to be beta testers. Make their time commitment as light and asynchronous as you can to keep them coming back for more. No one knows it yet, but you're building a user base of developers who will ultimately build half the system for you.

Recap: The people part

Remember:

- ✓ Find influencers from as many teams as possible
- ✓ Entice them with success metrics they care about
- ✓ Use research to build your first fanbase
- ✓ Turn new enthusiasts into beta testers



Success looks like this:

At least one influential leader is excited to see what happens with your design system experiment. Also, a large cohort of developers know that an MVP is in motion. Some of them have joined the early-adopter club and are thrilled about how the future design system can make their work easier.



Scott Arnold, Design System Lead
Motorola Solutions

CASE STUDY

Act as if there's already a shared language.

When he began building the design system for [Motorola Solutions](#), Scott Arnold knew that bridging the gap between design and development would be critical.

He decided to act as if there was already a shared language for design and code. Now all he needed to do was discover it. He took his engineering partner Nathan Babcock on a research roadshow, visiting developer workspaces from Krakow to Chicago.

Why a roadshow? "The message we were sending was, 'We're coming to you. We're gonna meet in your space, respect your ethos, and understand your dynamics as an org,'" Scott explained.

He and Nathan observed firsthand how developers were already making their work reusable, and together they began to find commonalities with existing design components. As they dove into conversations, developers latched onto the possibility of reducing overhead and creating a flexible, shared language for frontend product development. Many became avid proponents for the design system cause.

Scotts's advice:

“Talking to devs on their turf helps you avoid looking like the designer coming down from an ivory tower to hand them a thing they don't want. A design system should feel like a spirit of collaboration, rather than a product they have to use.”

The growth part: Building an MVP contribution plan

Eventually you'll need to establish clear models for governance, adoption, and contribution. How do guidelines ladder downward from universal rules, to product-level decisions, to edge cases? Which teams and products should adopt the design system? You'll begin to discover answers to these questions as you build your collection and conduct user research.

But you don't need to launch your MVP with a perfectly detailed plan, because even if you did, it would only continue to evolve.

You do, however, need to dig into your user research and write your best answer (for now) to this question:

How will people contribute changes or new components?

One of the most common traps new design system builders fall into is that they hold onto responsibility too tightly. They become the bottleneck for everything. Soon they get overwhelmed, and the design system simply stops growing.

A healthy contribution model keeps the design system growing even after you leave. It embeds design system growth into everyday workflows, meetings, and communication. Every team is building new components, helping to decide where and how they are used, and continuously giving insights back to the system.

Your user research can inform an initial contribution plan. **Be on the lookout for mentions of the following topics during your conversations:**

- ✓ Meetings or review sessions that could include a design system moment each week
- ✓ Points in the developer workflow when components could be evaluated for possible addition to the system
- ✓ Shared communication tools that could be used for ongoing design system conversations

You can also explore [Nathan Curtis's article](#) about common contribution models to frame your ideas against best practices.



Alexandra Clotfelter, Senior UI/UX Designer
Apartments.com

CASE STUDY

Keep contribution simple.

Alexandra Clotfelter built the first [Apartments.com](#) design system in [InVision DSM](#), working with frontend engineer Jake Anderson to add Storybook components.

When it was time to plan for design system governance, Alexandra and Jake knew they needed to keep their workload manageable and help teams self-serve whenever possible.

Alexandra created a [collaborator's guide](#) to explain why, when, and how to add a new component to the system. She shared decision flow charts that help everyone determine if a new component is really the right answer to their problem. If it is, the official component request is made by filling out a simple Google Form decorated with photos of brand spokesperson Jeff Goldblum.

Alexandra's advice:

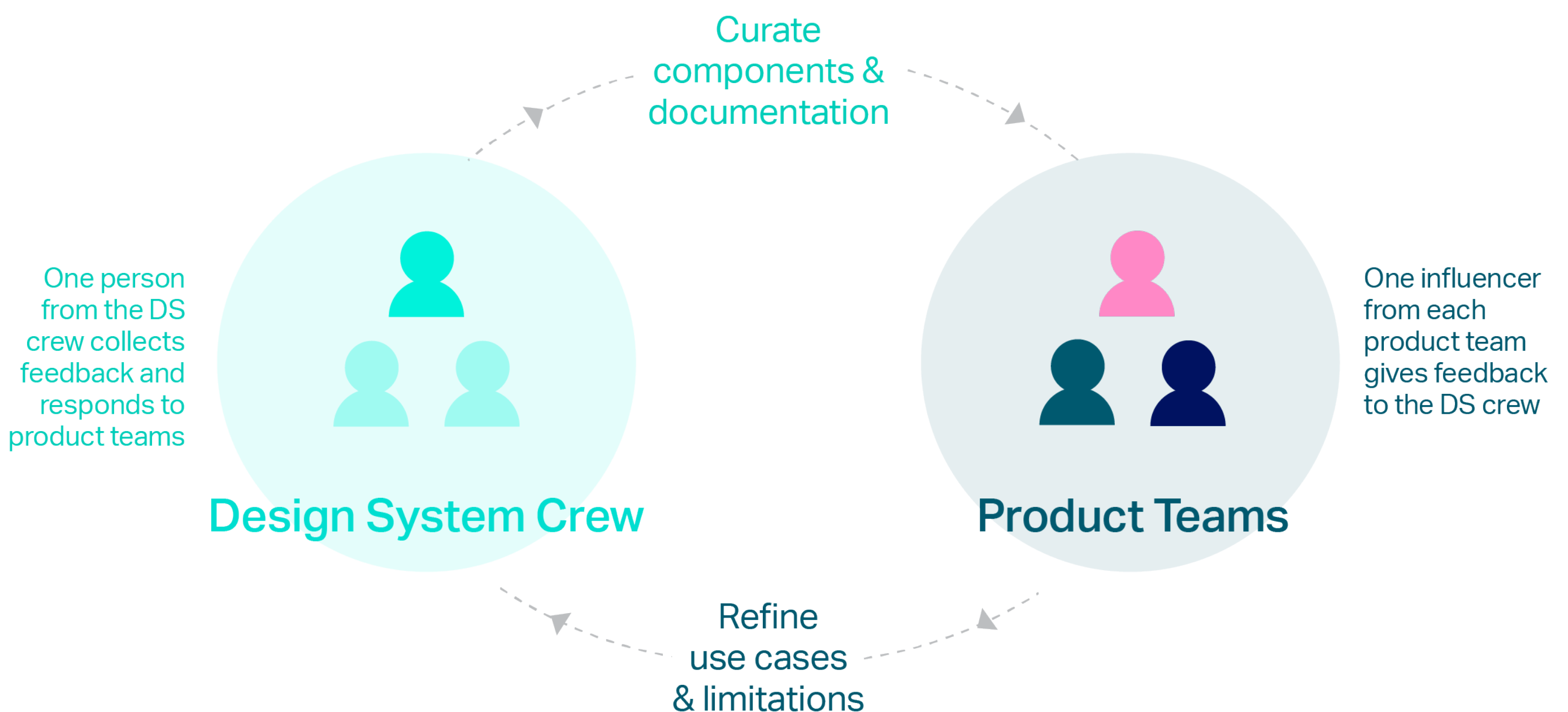
" It helps to go through the DSM Kickstart process. The InVision consultant met with us for a series of how-to sessions about best practices, governance, everything we'd need to drive adoption."

Don't overbake your new model. People often get overwhelmed at the moment of launch, thinking they need an answer for every eventuality. Just remember the contribution plan is in an MVP state just like your new components, documentation, foundations, and every other part of the system.

Recap: Creating a contribution plan

Remember:

- ✓ Don't hold onto responsibility too tightly
- ✓ Use existing workflows and meetings to collect feedback
- ✓ Keep contribution simple (create a guide)



Success looks like this:

You get conversations and practices in place, and trust that you're going to iterate.

After MVP launch

Now is the moment to circle back to those executive sponsors you prepped and touch base with your new developer fanbase. The goal is to increase momentum and launch the next iteration at a broader scale than the first. Here's your post-launch checklist:

- ✓ **Circle back to your beta users.** Find out how the new design system is starting to affect their workflows, how it can improve, and how they think it will change their work in the future. Collect numbers (time saved, eliminated meetings/tasks/emails) and sentiments (how much easier/less frustrating/more efficient the work felt).
- ✓ **Show leaders the success metrics.** Compile your observations, and project the broader success they can anticipate in the next bigger, better phase.
- ✓ **Use the [benchmarking guide](#) to plan phase two.** It provides outside context about where you stand, and how the system and its benefits could scale with the right investment.
- ✓ **Tell leaders what you need.** In order to make that next phase a reality, someone has to invest. Thankfully, you've already created a compelling business case for whatever you need, whether it's more hands on deck, more time and autonomy to pursue the work, or louder support from the top for design system usage.
- ✓ **Equip beta users as groundswell advocates.** Arm them with the success metrics you've gathered, and show them the plan for phase two, so they can promote it to their own leaders.

InVision DSM: Kickstart your design system.

The DSM Kickstart program delivers expert design system consultants to your doorstep. They partner with you to set the system up for ongoing success. They walk you through the mechanics of auditing, collecting, documenting, and organizing libraries. But more importantly, they help you think through the people parts: user workflows, adoption, governance, and contribution models.

[Talk to an InVision expert about how DSM can help](#)

Conclusion

You started with small steps. Now continue.

Like design, like product management, like development, a design system is never done. The work you did to launch your MVP and build your user base was the first of many steps.

But the benefits also start now, and they're only going to multiply from here. Keep going.



Stacey Eddy, Senior UX Manager
RealSelf

CASE STUDY

One baby step each week.

In July 2019, a group of passionate designers and developers at [RealSelf](#) began to meet once a week. Stacey Eddy explained, “The goal was to reimagine our processes and standards, in an effort to launch more consistent UI patterns.”

They had attempted to build a big, cohesive design system before, but this time they took a different approach. Their motto became, “Take one baby step each week.”

The baby step approach gave them the mental space to tackle a BHAG (big hairy audacious goal) one idea, one thought, one moment at a time.

Slowly, meticulously, the team defined elements like nomenclature standards, UI pattern standards, optimal processes, communication best practices, collaboration cadence... a never-ending list of ideas to consider and debate.

“But after just ten months,” Stacey said, “We’ve successfully aligned our developer Storybook component library to our design Sketch pattern library.”

We named our new RealSelf design system Radiance because it exemplifies positivity, joy, and beauty, and fits into our brand ethos perfectly."

The newly aligned system created a master source of truth, ended duplicative work in design/dev handoffs, and enabled one seamless update in production, across the site, for future UI iterations. **"We calculate Radiance saves us upwards of 1250 salaried hours per year, by reducing frustration and rework between groups.** Plus the new modern UI feels clean and fresh in that artistic sense, where it's hard to explain why you like something, but you just do."

Stacey's advice:

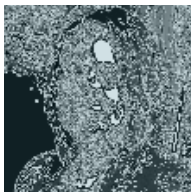
" Find developers and designers who really dig UI, people who care deeply about consistency, simplicity, details, and beauty. They'll be your strategic partners and will help build the backbone of your design system."

We're here for you.

Whether it's your first attempt or your fiftieth iteration, InVision DSM can help you evolve your design system. **Reach out any time and let our experts help you take your next step toward greater connection, greater efficiency, and greater consistency, across all your products and teams.**

The image shows a screenshot of the InVision Design System Manager (DSM) interface. At the top left is the InVision logo. The main content area is titled "DSM." and features a sidebar menu with categories: Colors, Text Styles, Icons, Components (highlighted in red), Buttons, Alerts, Inputs, and Overlays. The main area displays a grid of component examples with labels below them: a square button with a plus sign, a dial chart, a search bar with a toggle, a typeface showing "Aa", a profile picture of a man, and a message card with a profile picture and the text "Good morning!". Above the grid are two large decorative graphics: a donut chart and a bar chart. At the bottom right, the letters "Aa" are displayed in a large, light blue font.

Contributors



Rebecca Kerr, Principal Conversation & Content Strategist

Exploring major product team transformations through interviews, stories, and conversation.



Liz Steelman, Editor

Running InVision's blog Inside Design and helping the design community tackle the new world of work.



Marci Pasenello, Director of Product

Product lead for InVision's DSM who's been experimenting with the product development process for many years.



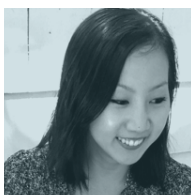
Zoë Adelman, Design System Consultant

Helping InVision customers create and evolve design systems for scale.



Nick Hahn, Director, Design Systems Consultant

Excitedly helping teams evolve their design systems and bring forth the design system revolution as Director of Design Systems Consulting.



Micky Teng, Director of Product Marketing

Marketing InVision DSM to help customers solve key design system challenges.



Catriona Shedd, Director of Product Design

Design lead for InVision DSM and design system advocate specializing in helping organizations improve design system workflows to increase adoption, foster collaboration, and elevate the efficacy of design systems at scale.



Ehud Halberstam, Product Manager

Working on InVision's DSM and these days focusing on developer-designer collaboration.