Keeping design systems alive and healthy with user feedback and data





Jan Toman

Director of Design, Supernova.io

supernova

Supernova helps you build, mature, and scale your design system with ease. From powerful documentation to efficient design system management, it's the ideal end-to-end platform that grows with your design system.

Previously leading design systems at





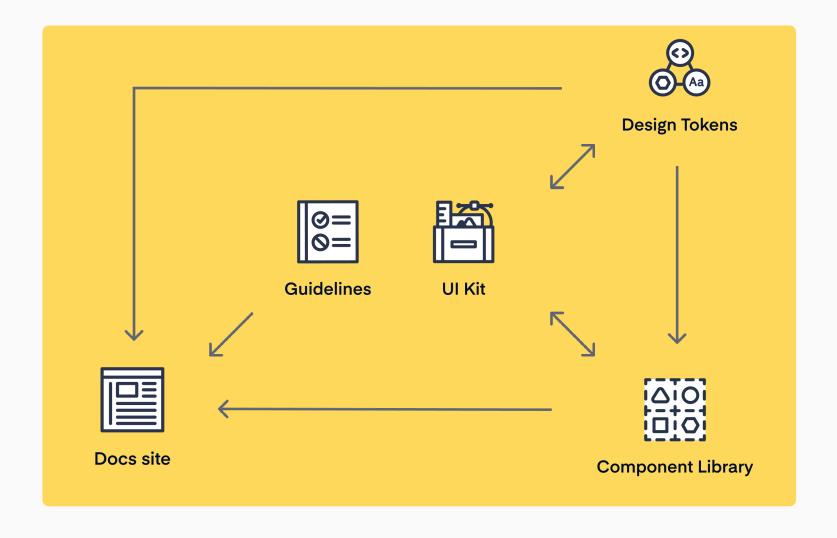


A design system isn't a project. It's a Product, serving products.

— Nathan Curtis

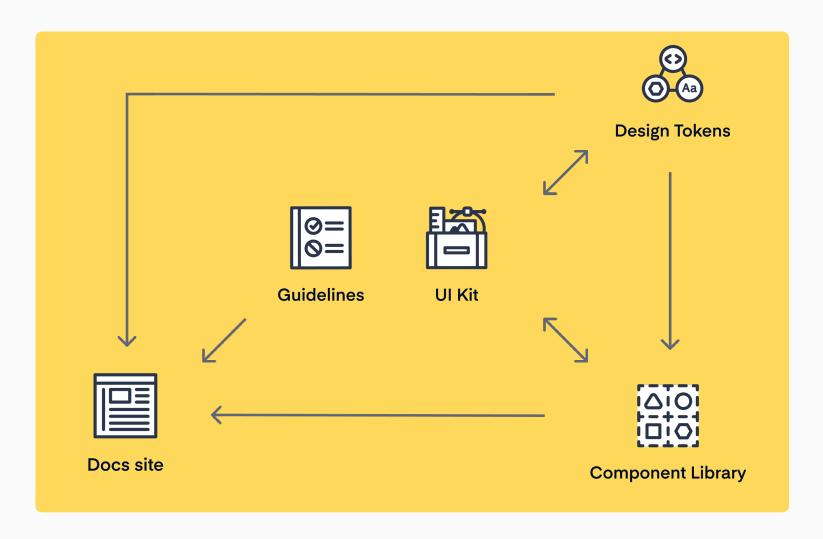
'A design system isn't a project. It's a Product, serving products.

Nathan Curtis





Product #2

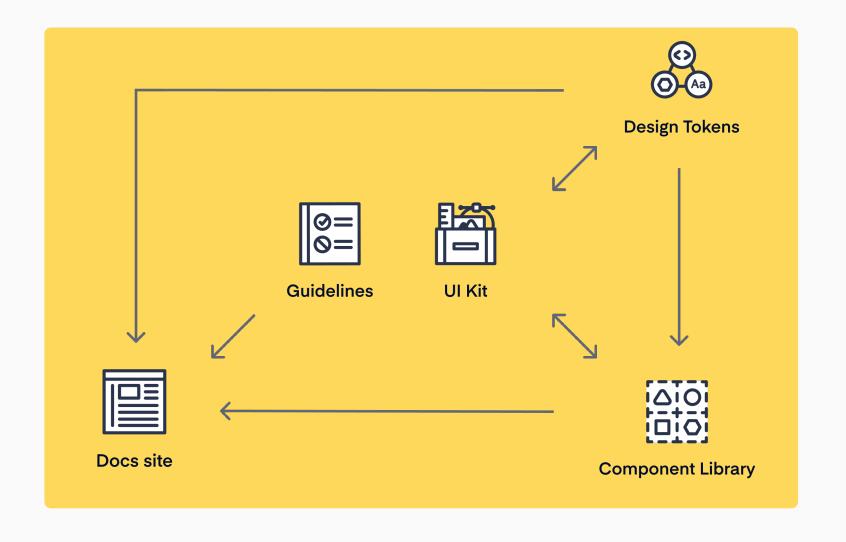


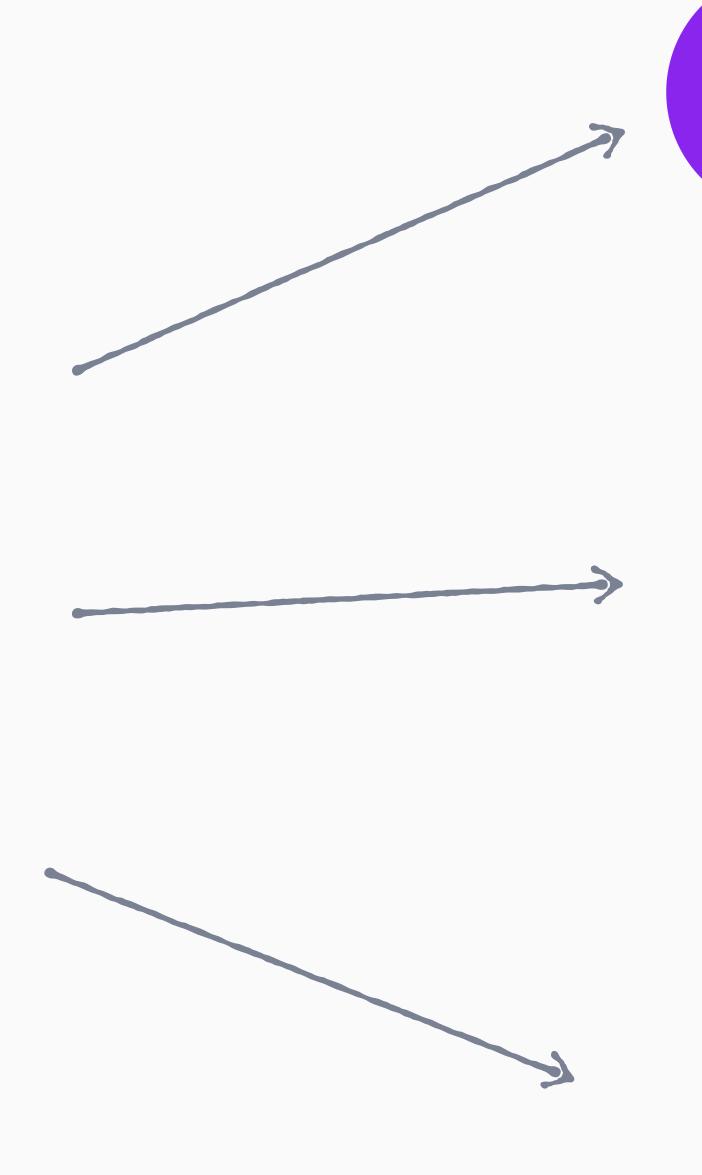


Product #2

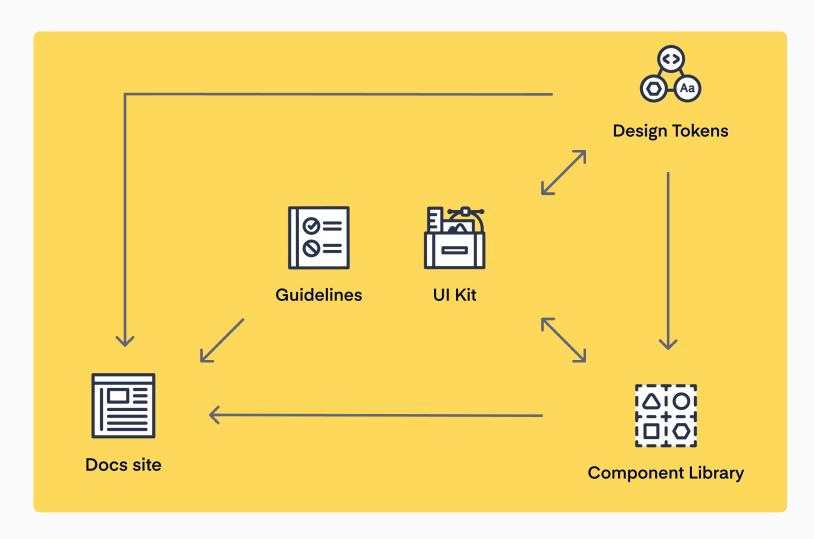
Product #1

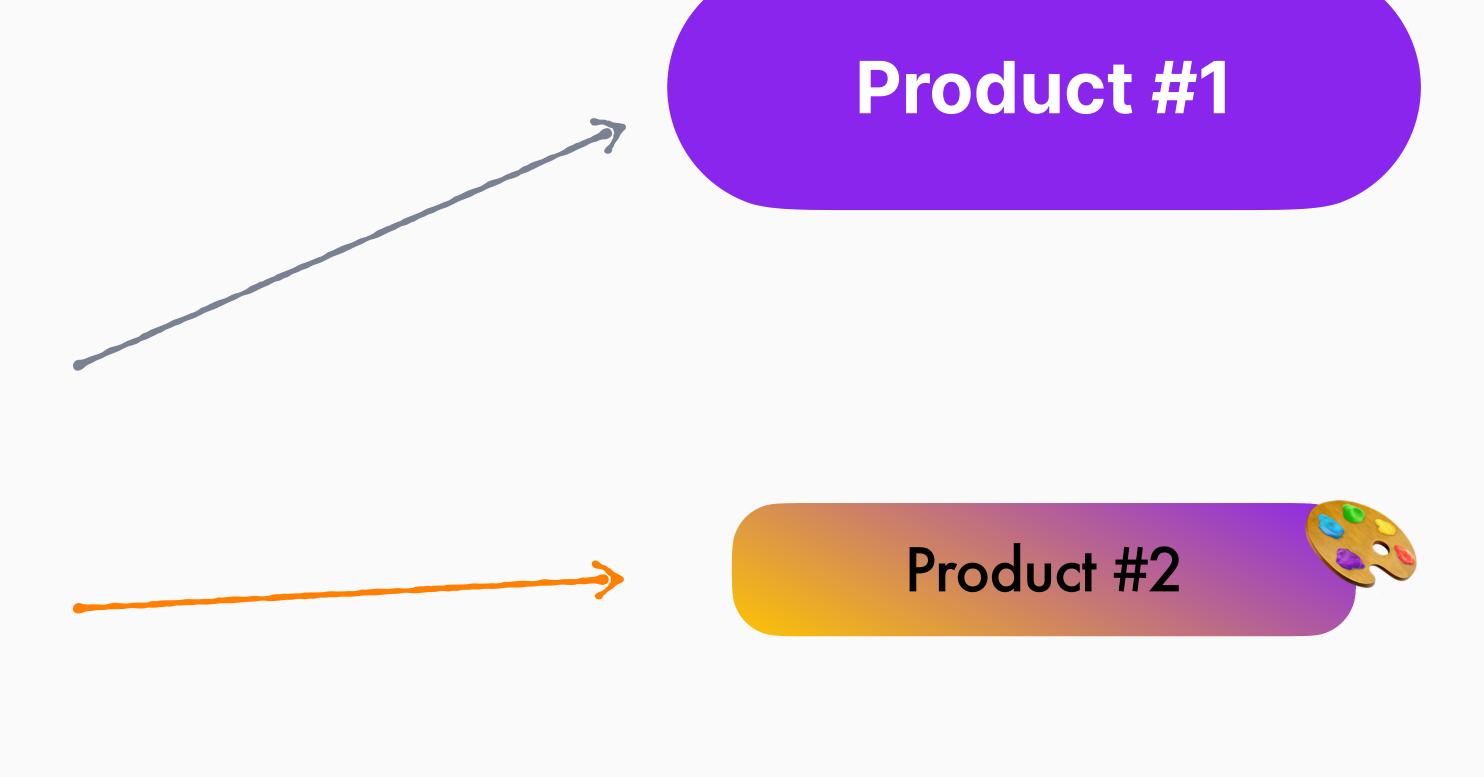
Design system

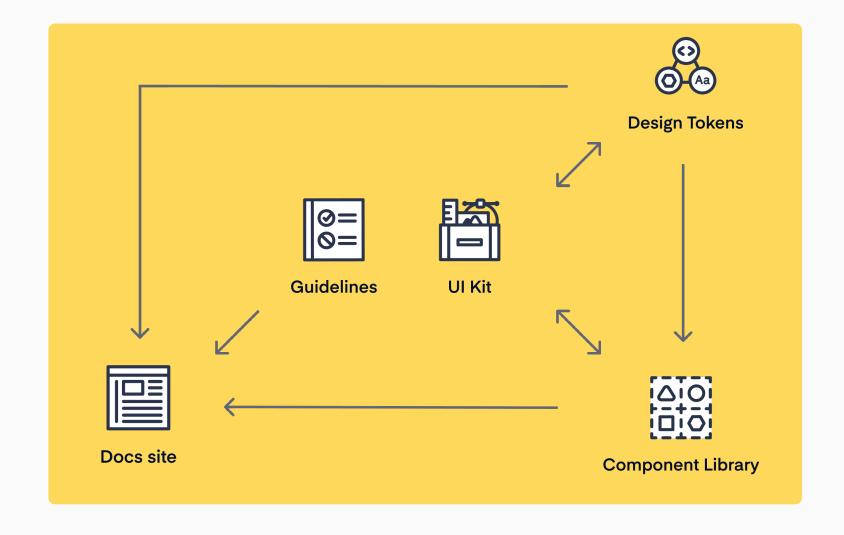


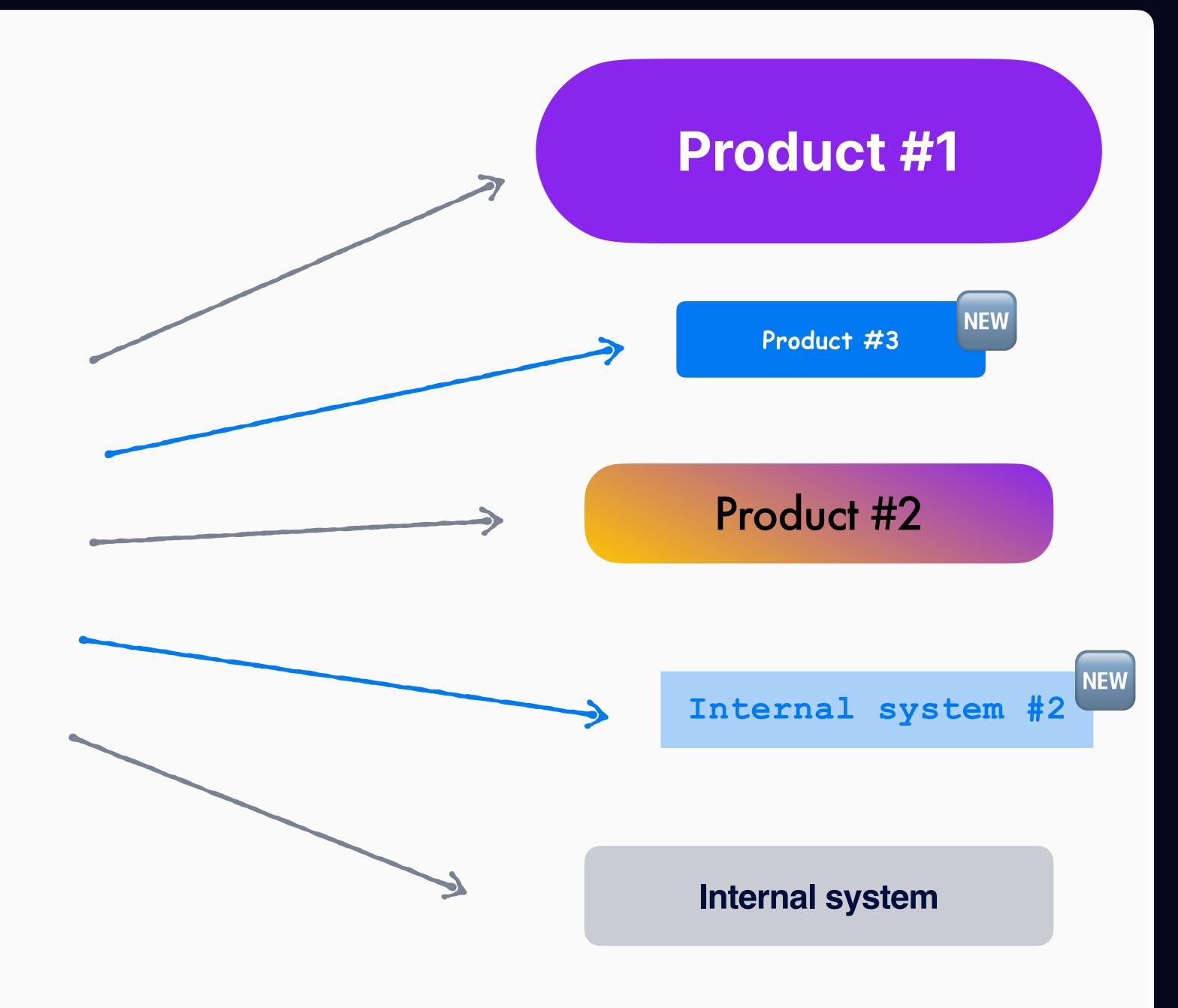


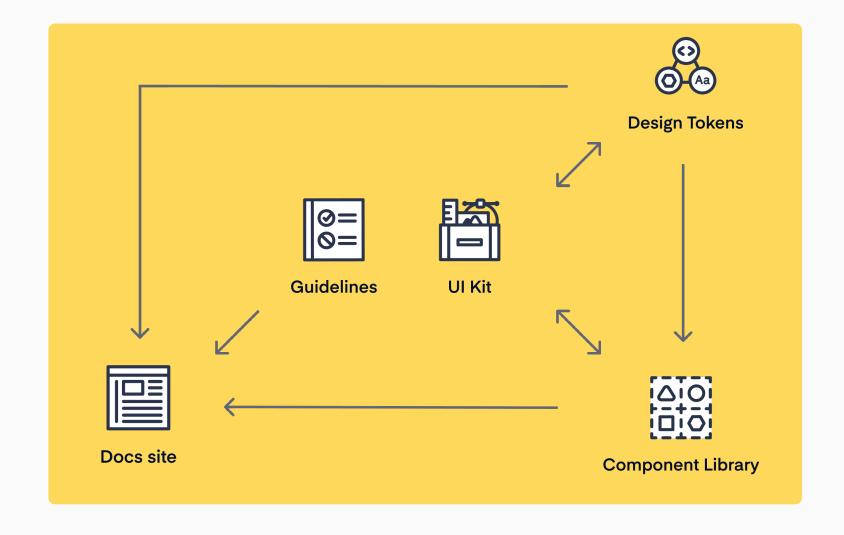
Product #2

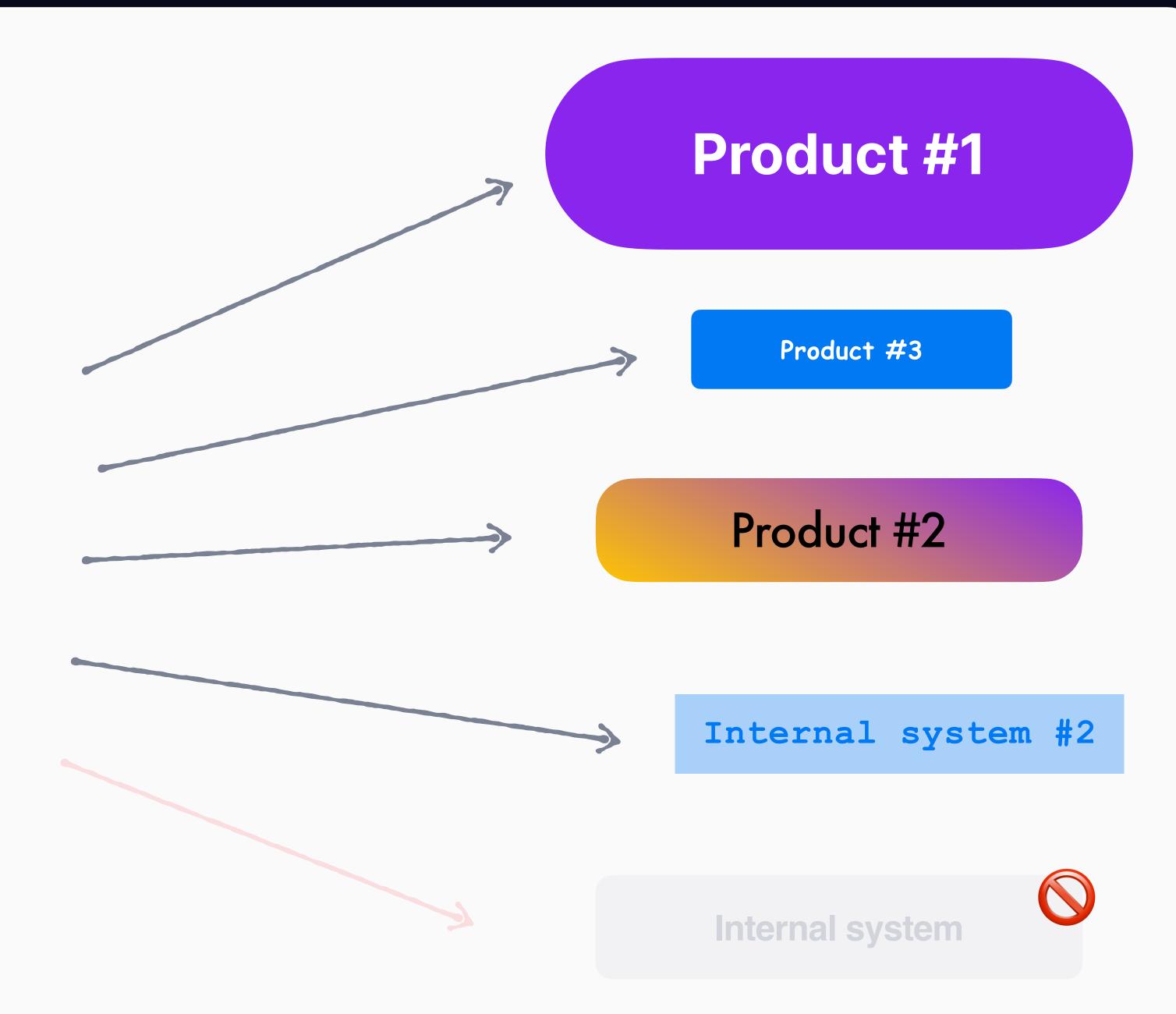




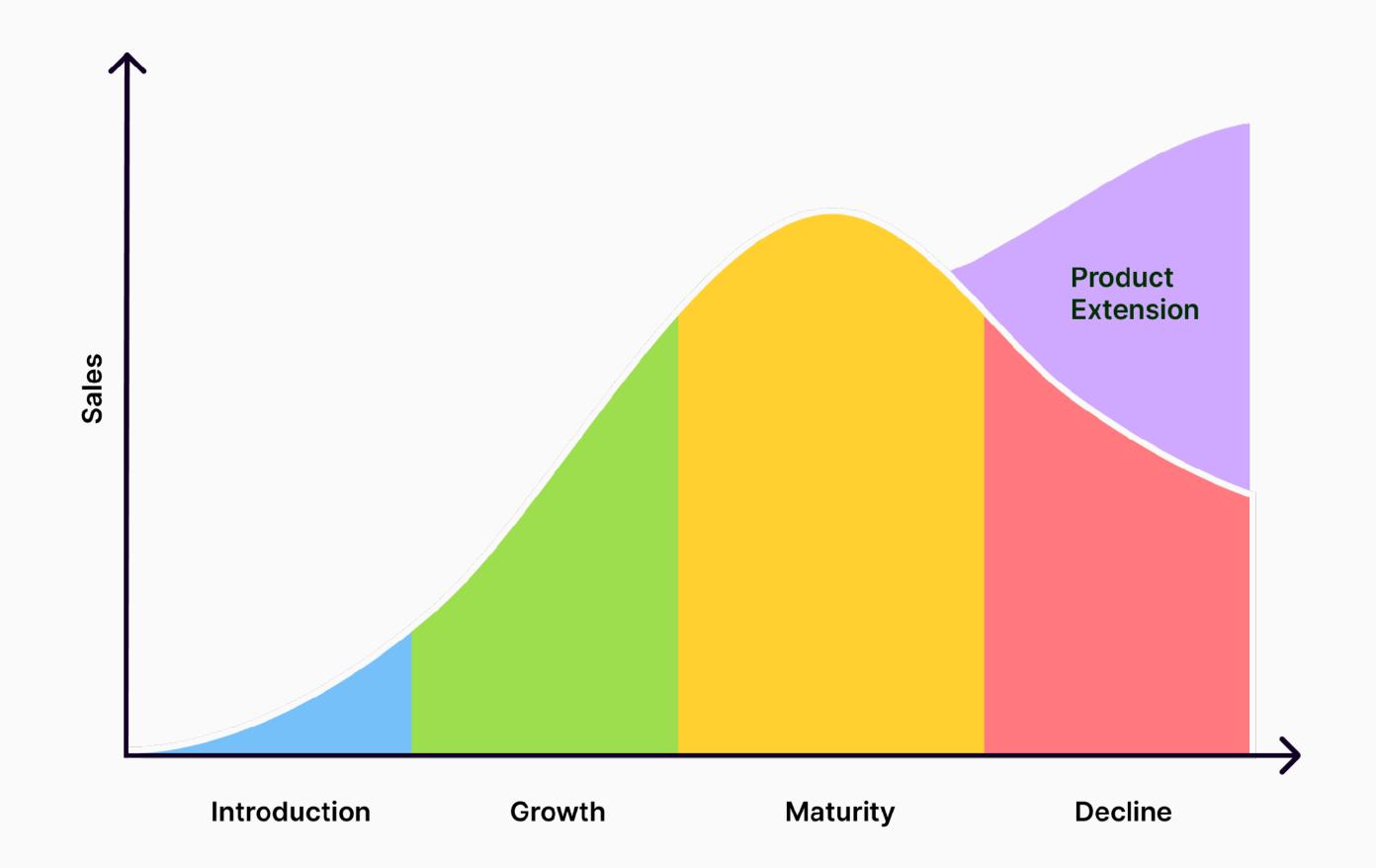


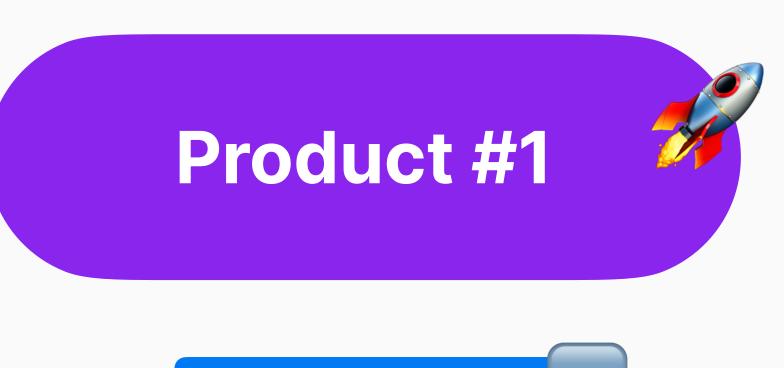






Product lifecycle





Product #3

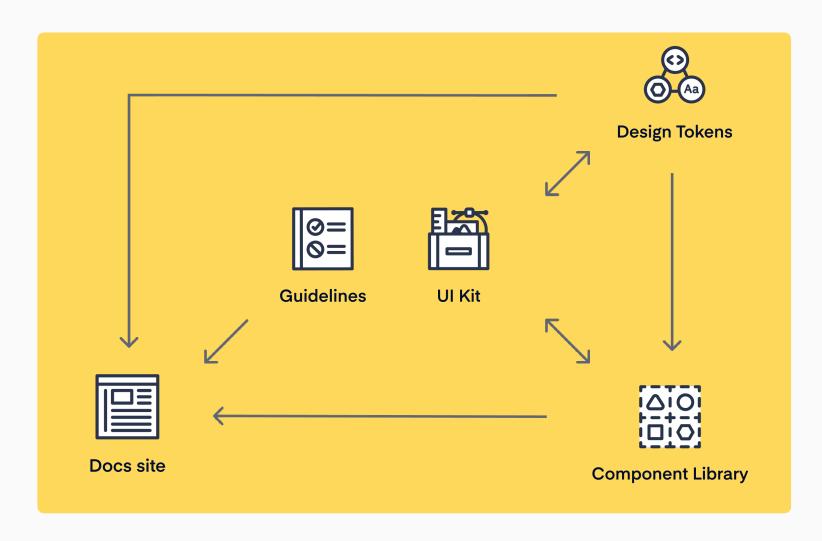
Product #2

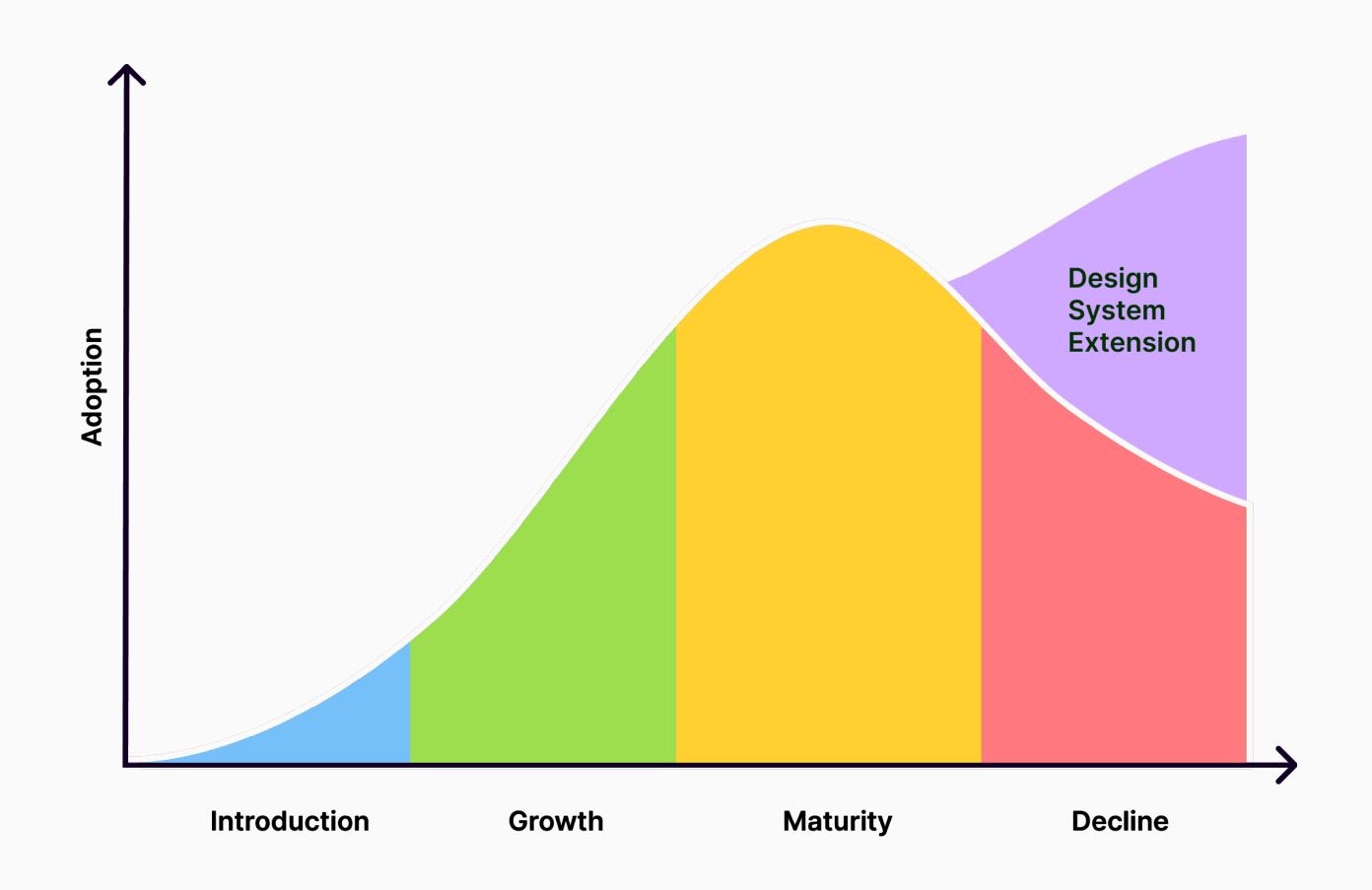
Internal system #2



Product lifecycle for design systems? Yes!

Design system





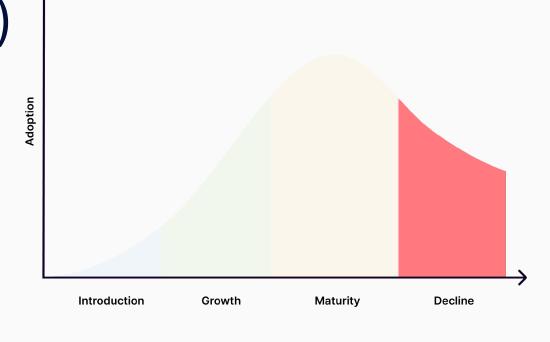
Signals of design system's decline

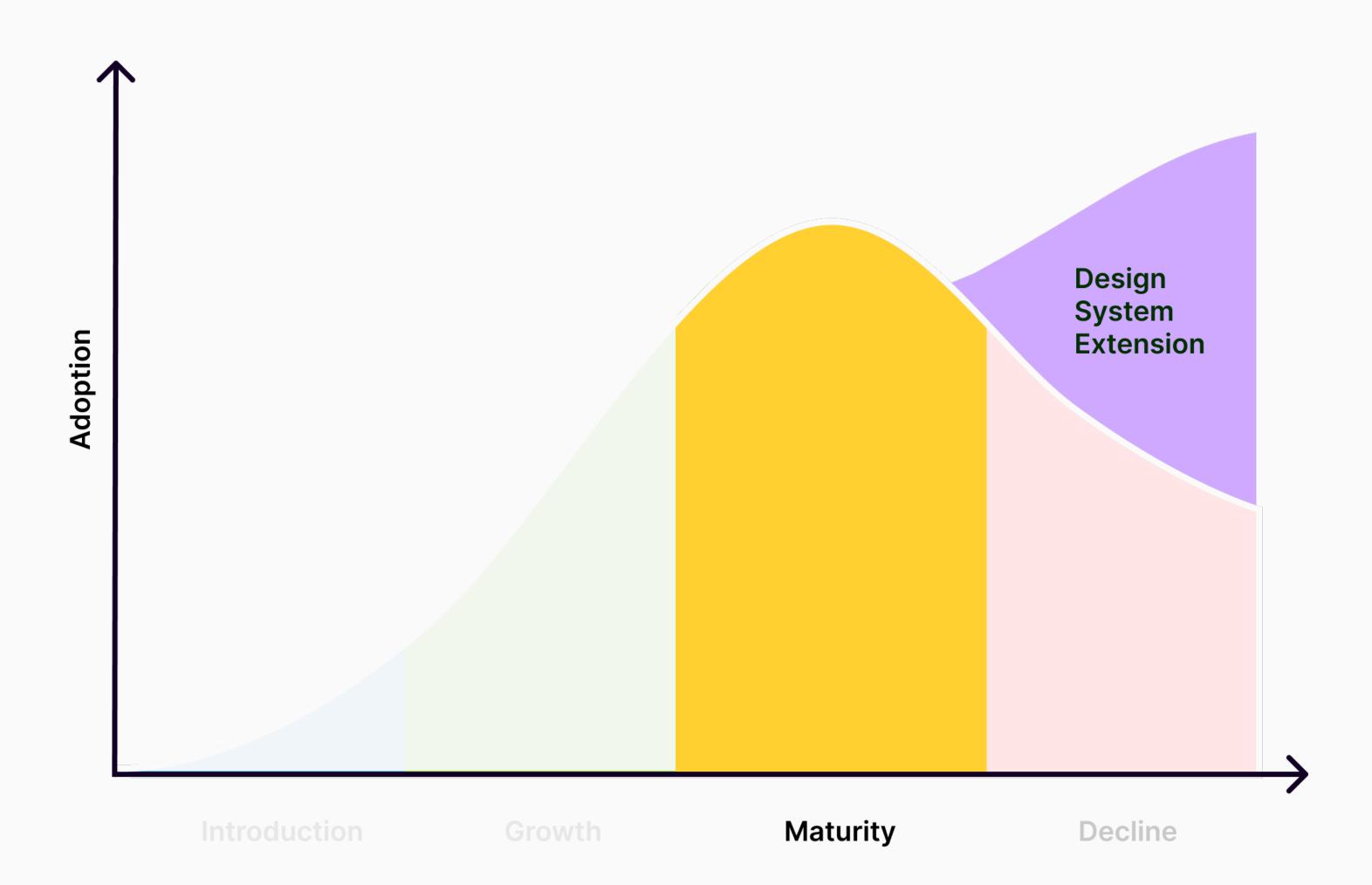
- Many bugs and/or unpredictable breaking changes
- Outdated components and/or documentation
- Support is slow, blocking product teams in their delivery
- Design system is always late with their improvements



Leading to...

- Product teams not using the system
- W Growing design and technical debt within a system and/or product(s)
- We New rogue design systems emerging in the company
- **V** Losing executive support





To understand what works and whatnot.

Research & continuous discovery

To evolve the system.

To understand what works and whatnot.







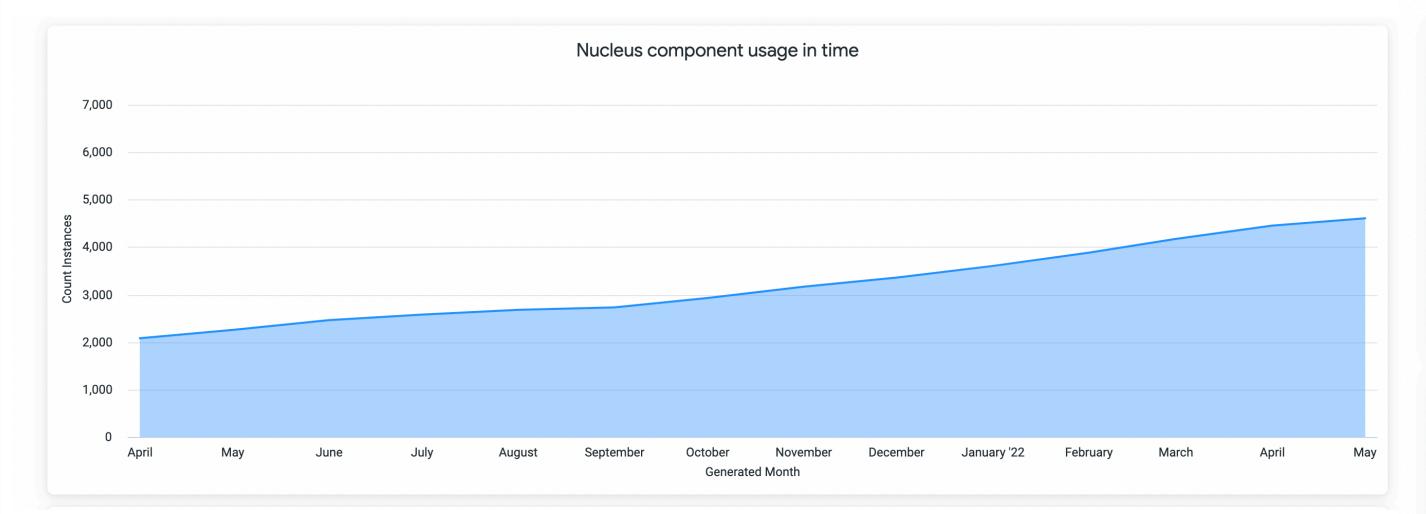
Adoption

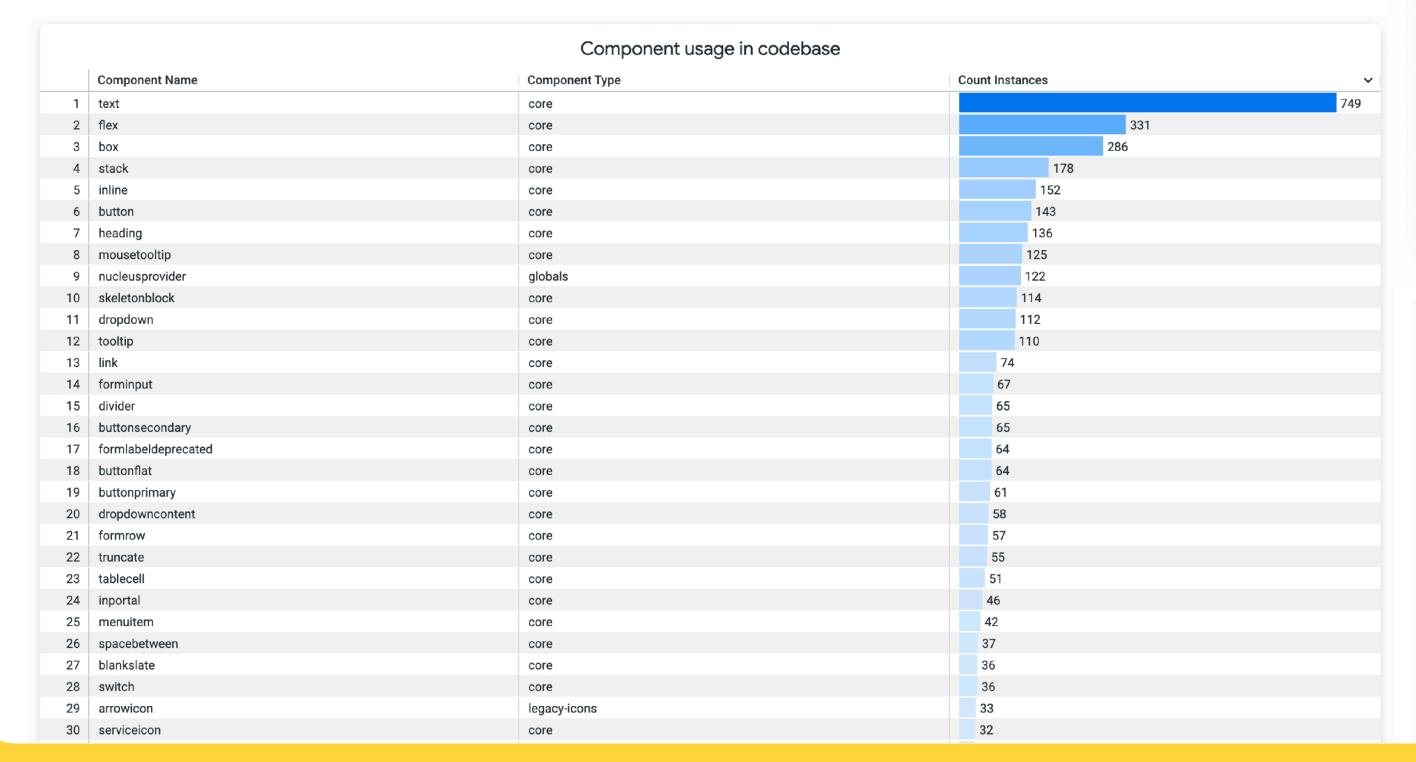
System status

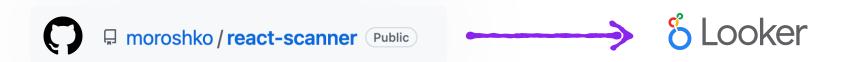
Satisfaction

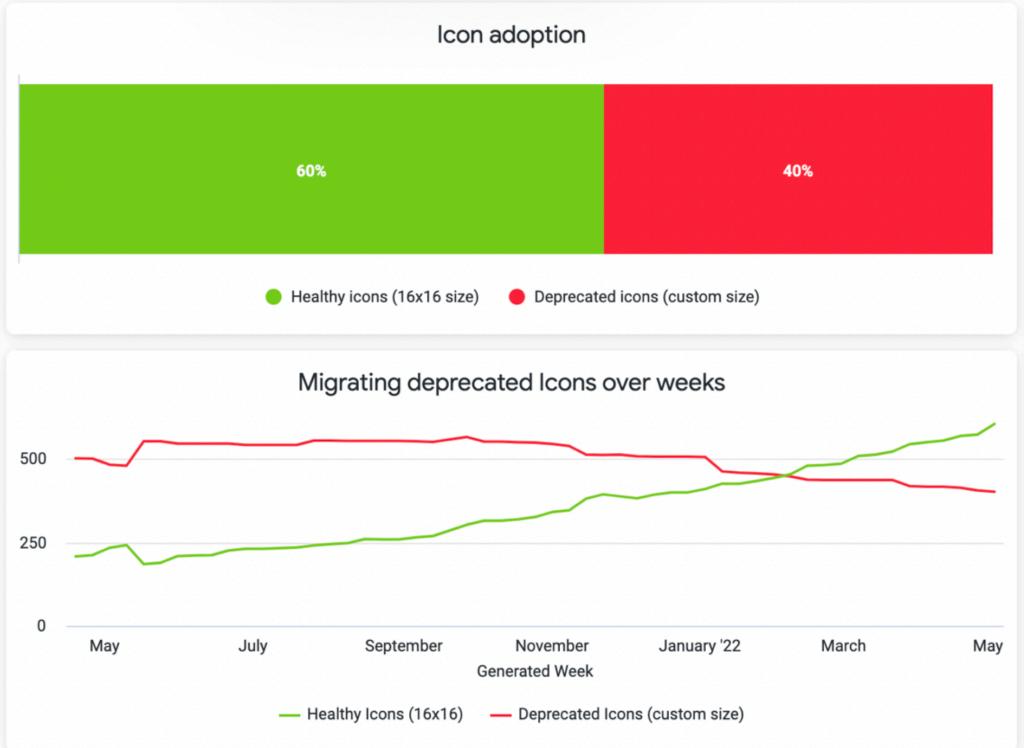
Adoption

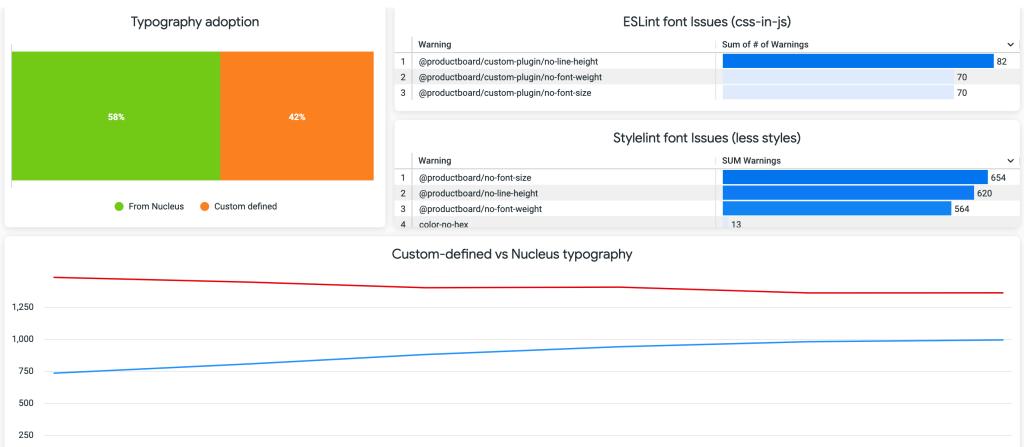
Observability — **Adoption / From Productboard**



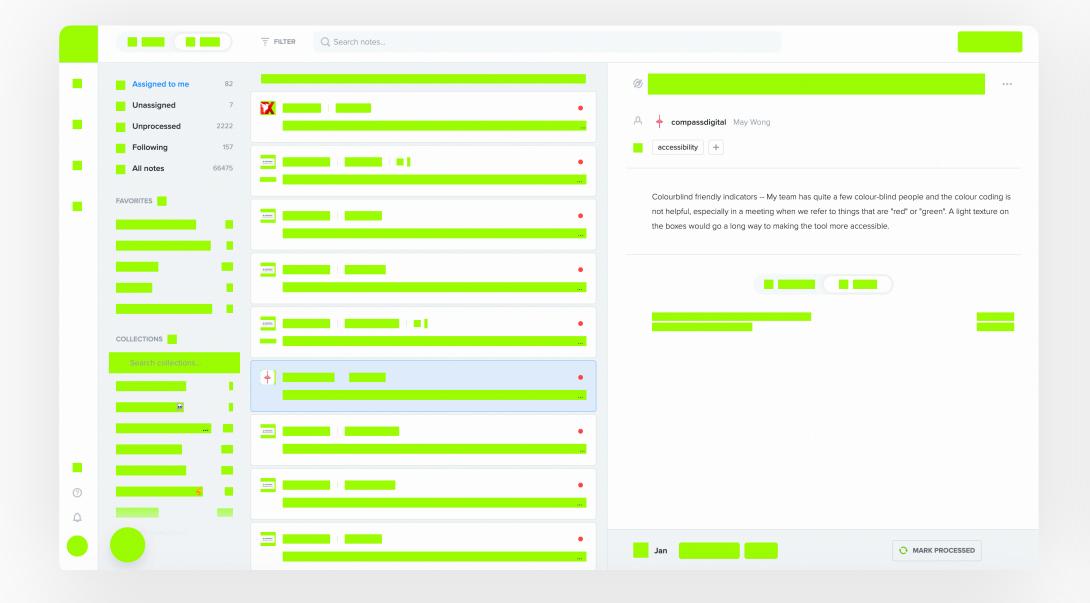


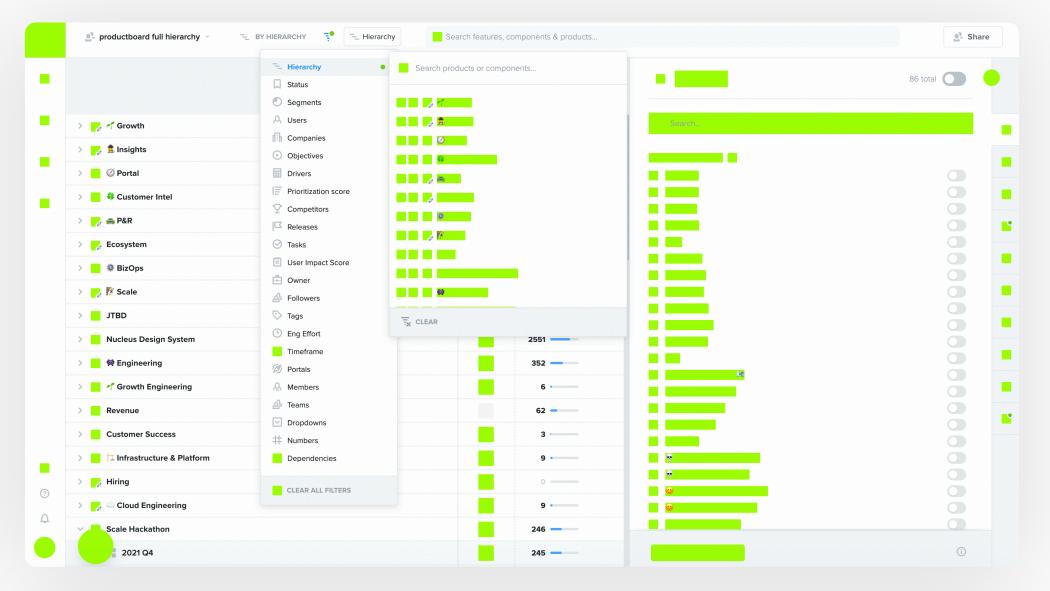


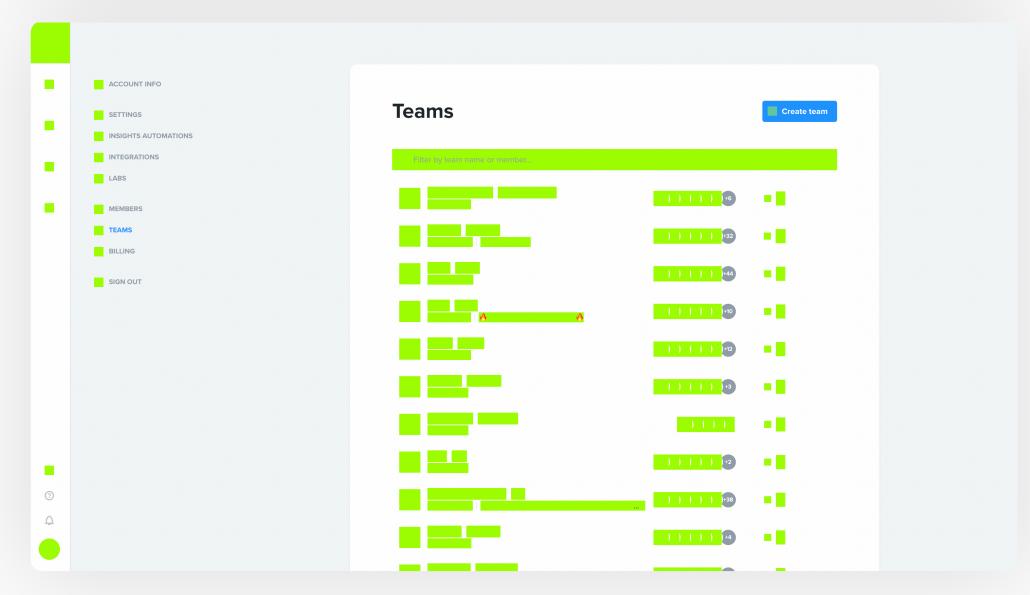


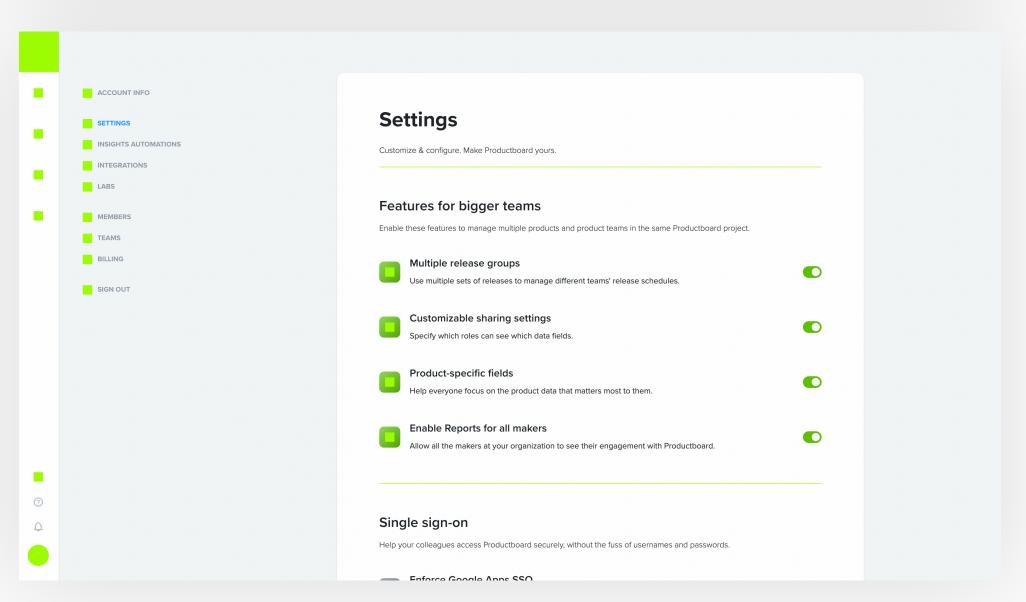


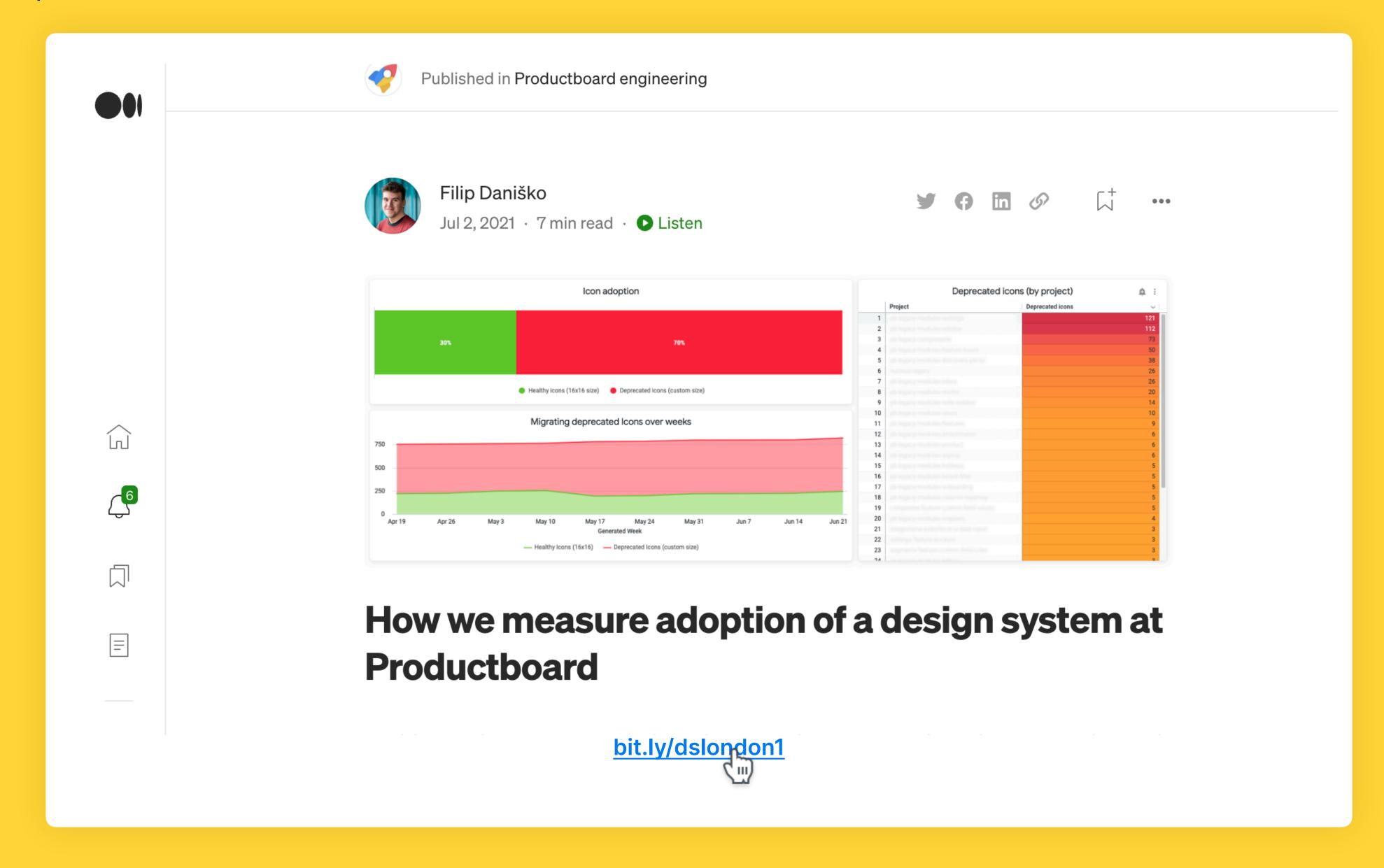
Observability — **Adoption / From Productboard**











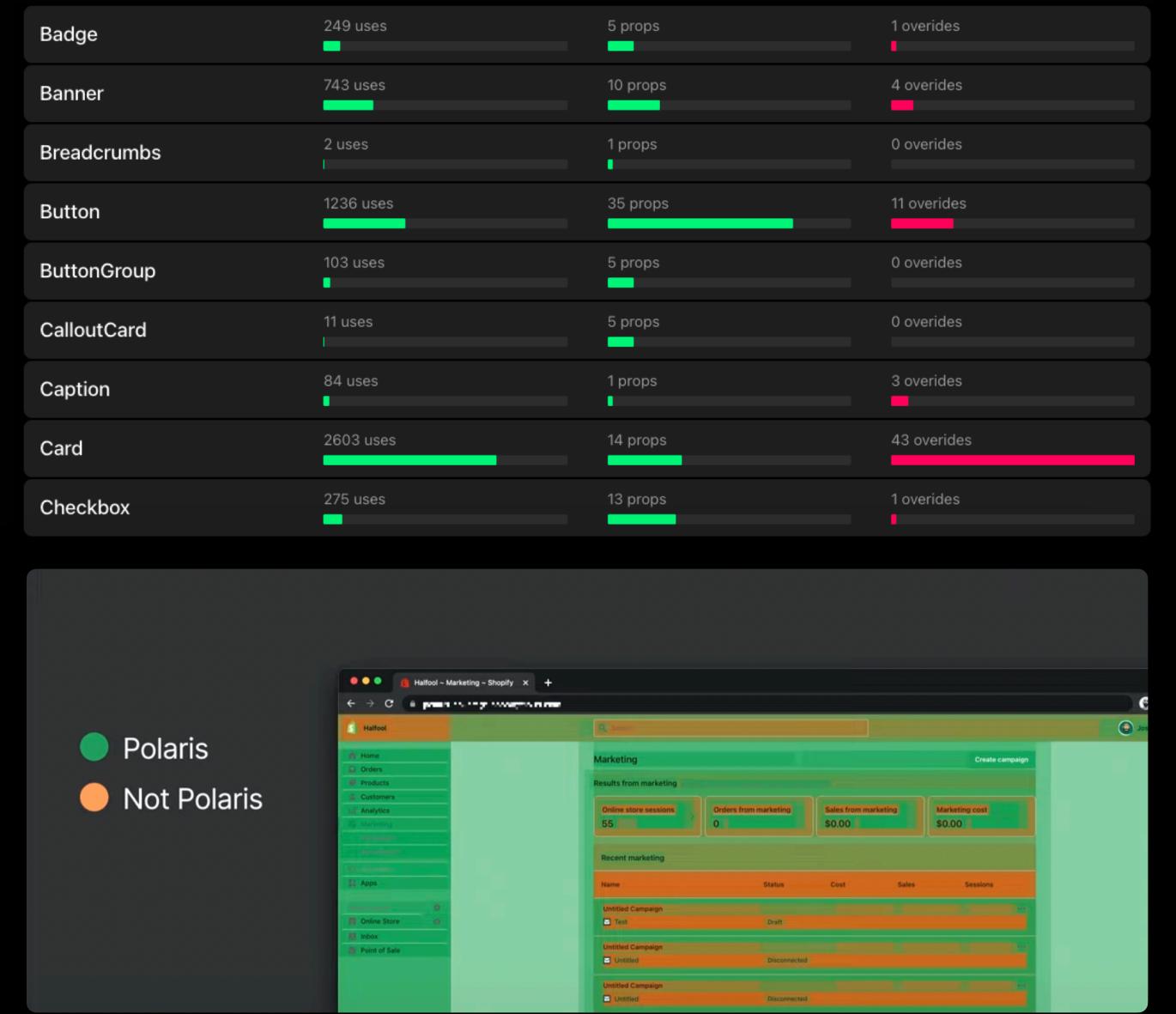
From the community

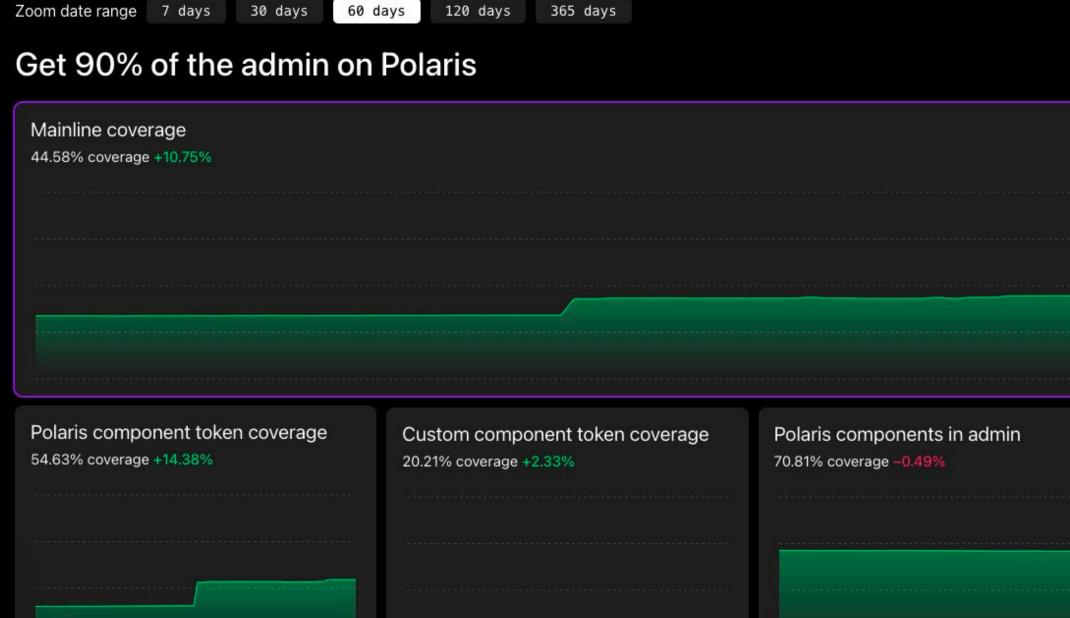


Observability — Adoption / From the community (Onfido)



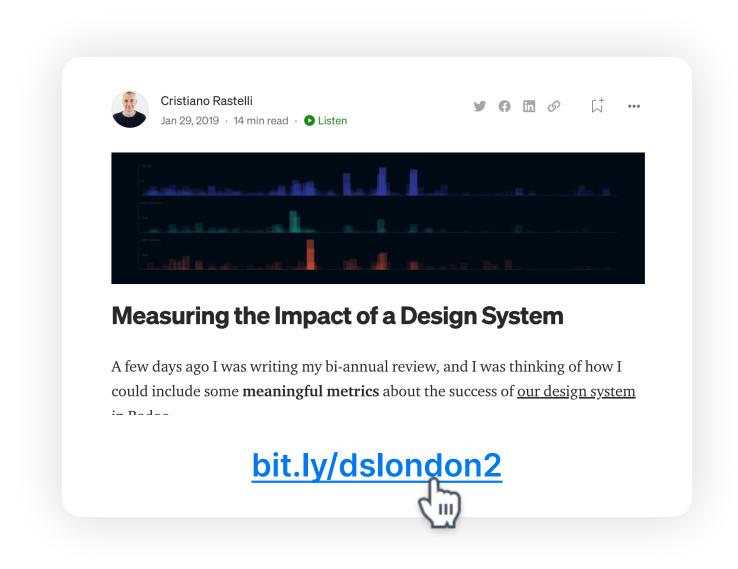
Observability — Adoption / From the community (Shopify Polaris)

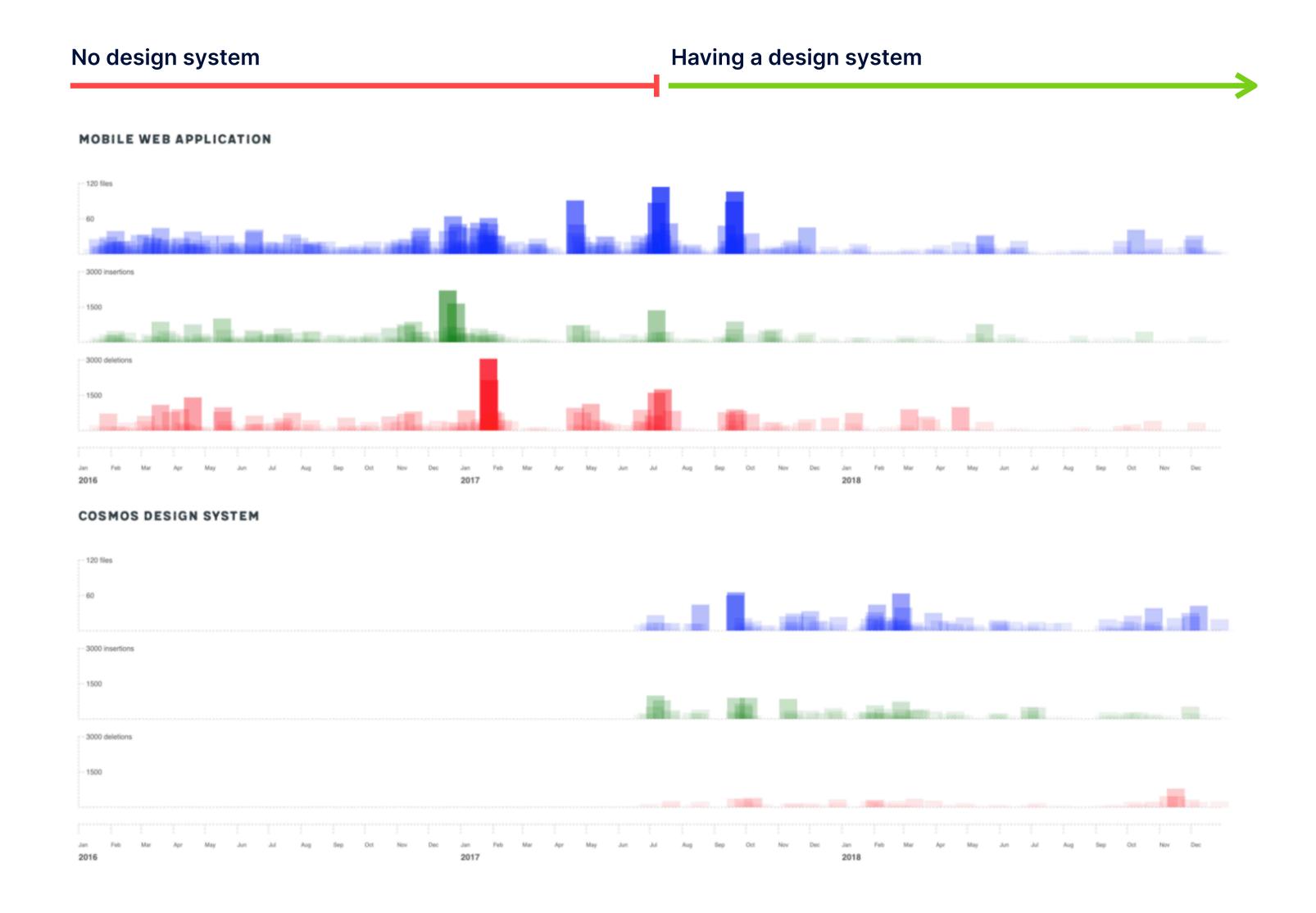




Code browser Components Elements Working hours

Observability — Adoption/ From the community (Bumble)







Status

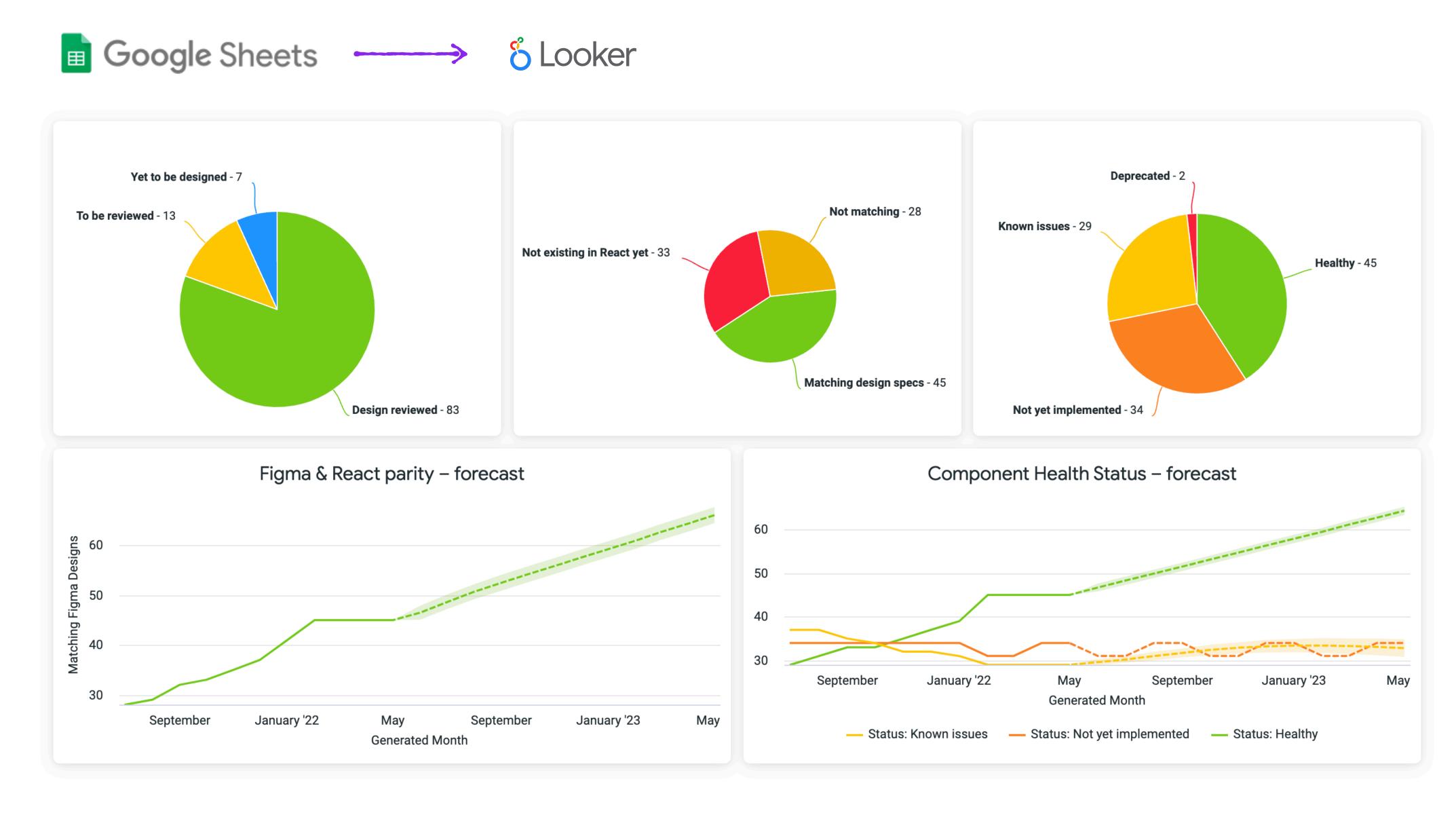
	In Figma	In React	In Swift
Accordion	Yes	No	No
Button	Yes	In progress	Yes
Dropdown	Yes	In progress	No
Modal	In progress	No	No

• •

Observability — Status / From Productboard

A1	→ fx Component				^			
	A	В	С	D	E ▼	F	G	
1	Component =	Figma =	Design status =	React =	React status =	Health statu: 🍸	Type 🝸	Note
9	Audit Log / Item	Released	Reviewed	To be done	n/a	Not yet impleme	UI component	
10	Avatar	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	Legacy implen
11	Avatar / Organization	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	Note: Loads Ic
12	AvatarGroup	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	
13	Badge / Badge Labs	Released	Reviewed	Released	Needs review	Known issues	UI component	Legacy name:
14	Badge / Badge Notification	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	
15	Badge / Badge	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	
18	BarIndicator	To be done	Needs review	Released	Needs review	Known issues	UI component	It's from the se
22	Вох	n/a	n/a	Released	Needs review	Known Issues	UI component	
23	BoxHighlight	n/a	n/a	Released	Needs review	Known issues	UI component	
24	Bulk Edit Checkbox	n/a	n/a	Deprecated	Deprecated	Deprecated	UI component	
27	Button / Button Close	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	
28	Button / Button Danger	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	
29	Button / Button Flat	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	
30	Button / Button Floating	Released	Reviewed	Released	Needs review	Known issues	UI component	Legacy name:
31	Button / Button Integration	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	
32	Button / Button Login	Released	Reviewed	To be done	n/a	Not yet impleme	UI component	
33	Button / Button Primary	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	
34	Button / Button Secondary	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	
35	Button / Button Segmented	Released	Reviewed	To be done	n/a	Not yet impleme	UI component	
36	Button / Button Status	Released	Reviewed	To be done	n/a	Not yet impleme	UI component	
37	Button / Button Thin	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	
39	Card	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	
41	Checkbox	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	
45	Coachmark	Released	Needs review	Released	Needs review	Known issues	UI component	
46	ColorField	Released	Needs review	Released	Needs review	Known issues	UI component	Legacy name:
47	Comment	Released	Reviewed	Released	Matching Figma designs	Healthy	UI component	
48	ContextMenu	Released	Reviewed	Released	Matc	Healthy	UI component	
50	Detailed Label	Released	Reviewed	Released	Needs review	Known issues	UI component	

Observability — **Status** / **From Productboard**



Tools used: Google Sheet data connected to Looker.com (custom script for data import)

Health Status for each component



Healthy component matches Figma implementation and meets the highest quality standards for our UI.

• Known Issues

There are known issues, but the component is generally recommended to use. If you want to improve this component, pick any missing criteria and improve its implementation.

Issues to be solved:

Switch is passing classname, thus being able to be extended with any style. We're working on removing this. DS-477

Switch uses a theme prop which can effect visuals of the component. This capability should not be needed anymore and will be removed.

Deprecated

This components is deprecated and shouldn't be used anymore.

Component(s) to use instead:

RadioGroup

Available in Figma

OPEN IN FIGMA 🔀

All component variants and states are also in Figma.

Code matches design specification

All component variants and states are implemented in React.

Latest implementation standards

Implementation complies with coding and API conventions for the design systems components.

Code is properly documented

API is documented, together with the functional specifics of each component and interactive playground.

Design tokens

All design attributes (color, typography, layout, animation, etc.) are connected to Nucleus design tokens.

Accessible contrast

Follows WCAG 2.0 standards for contrast (AA).

Keyboard interactions

Follows WCAG 2.0 standards for keyboard accessibility and includes a description of the keyboard interactions.

Responsive behavior

It's correctly displayed in supported screen sizes.

Usage guidelines

Includes a list of Dos and Don'ts that highlight best practices and common misuses.

Covered by tests

All functionalities of the component are covered by unit tests.

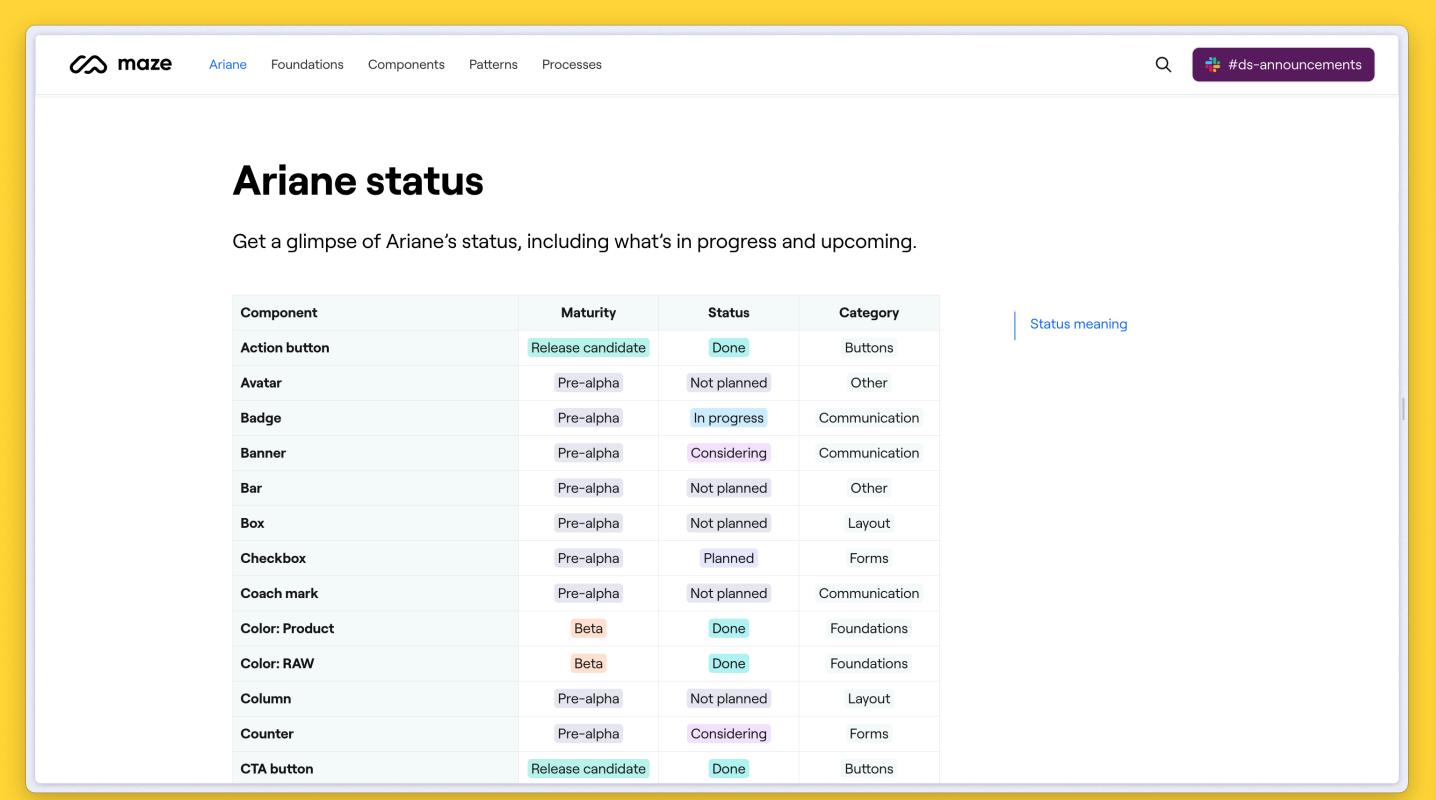
```
import type { HealthType } from '../docs-blocks';

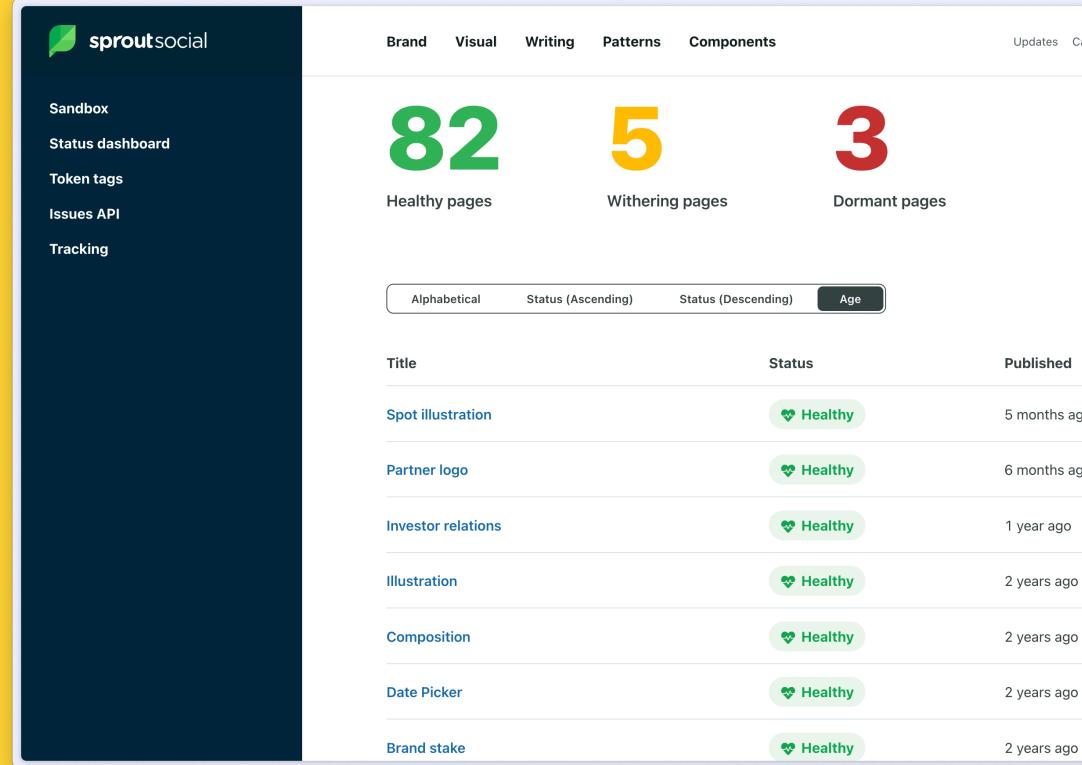
export const health: HealthType = {
  componentName: 'Button',
  criteria: {
    figma:'https://www.figma.com/file/...',
    matchesDesignSpec: true,
    matchesCodeStandards: true,
    hasDocumentation: true,
    hasGuidelines: false,
    hasDesignTokens: true,
    hasAllyKeyboard: true,
    hasTestsCoverage: true,
    hasPasspansivity: true
```

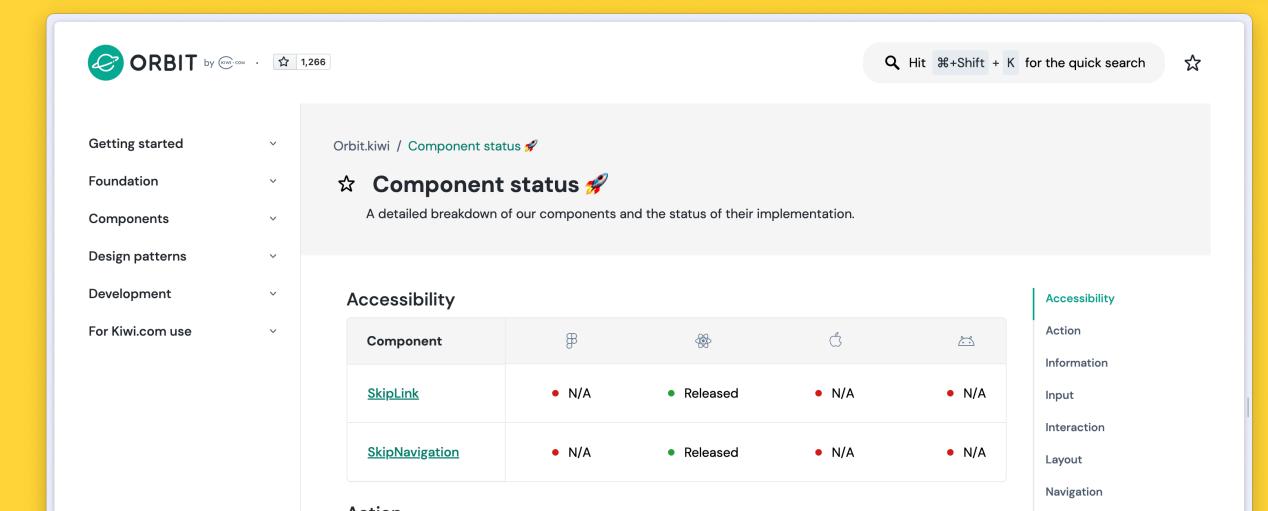
From the community

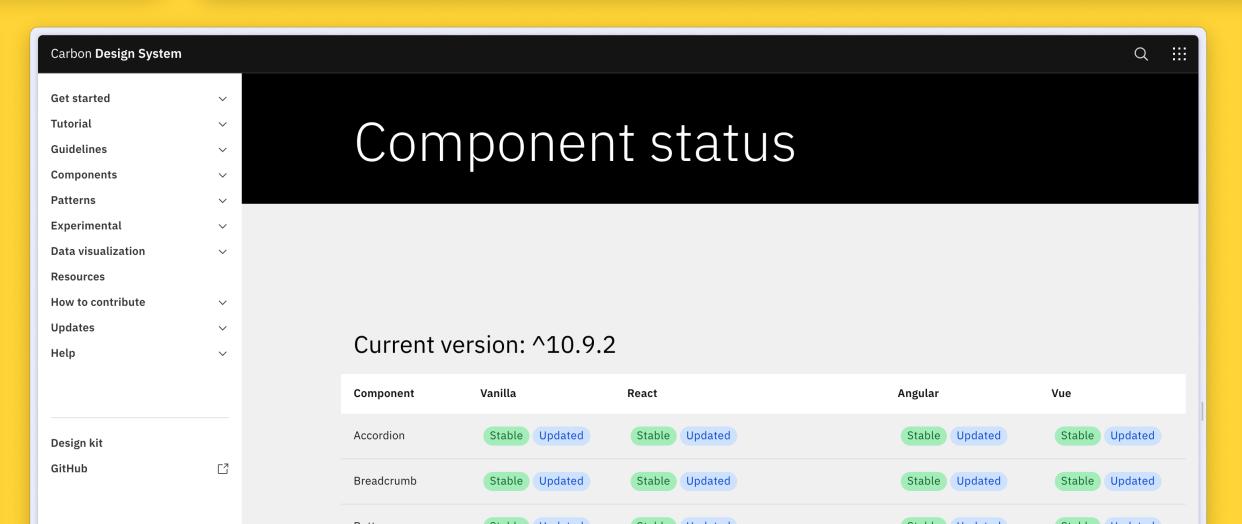


Observability — Status / From community (Orbit by Kiwi.com, Ariane by Maze.co, Carbon by IBM, Seeds by SproutSocial)









Observability — Status / From community (Spectrum by Adobe)

Design checklist

N/A All interactive states

Includes all interactive states that are applicable (hover, down, focus, keyboard focus, disabled).

All color themes

Works properly across all four color themes (lightest, light, dark, darkest).

✓ All platform scales

Includes a desktop scale (UWP, macOS, web desktop) and a mobile scale (iOS, Android, web mobile).

Accessible use of color

Color is not used as the only visual means of conveying information (WCAG 2.0 1.4.1).

Accessible contrast for text

Text has a contrast ratio of at least 4.5:1 for small text and at least 3:1 for large text (WCAG 2.0 1.4.3).

Accessible contrast for UI components

Visual information required to identify components and states (except inactive components) has a contrast ratio of at least 3:1 (WCAG 2.11.4.11).

Content design

UI language and information design considerations have been incorporated into component design.

Defined options

Includes relevant options (variant, style, size, orientation, optional iconography, decorations, selection, error state, etc.)

Defined behaviors

Includes guidelines for keyboard focus, layout (wrapping, truncation, overflow), animation, interactions, etc.

Usage guidelines

Includes a list of dos and don'ts that highlight best practices and common mistakes.

Writing guidelines

Includes content standards or usage guidelines for how to write or format in-product content for the component.

Internationalization guidelines

Works properly across various locales and includes guidelines for bidirectionality (RTL).

Keyboard interactions

Follows WCAG 2.0 standards for keyboard accessibility guidelines and includes a description of the keyboard interactions.

Design tokens

All design attributes (color, typography, layout, animation, etc.) are available as design tokens.

UI kit

Includes a downloadable XD file that shows multiple options, states, color themes, and platform scales.

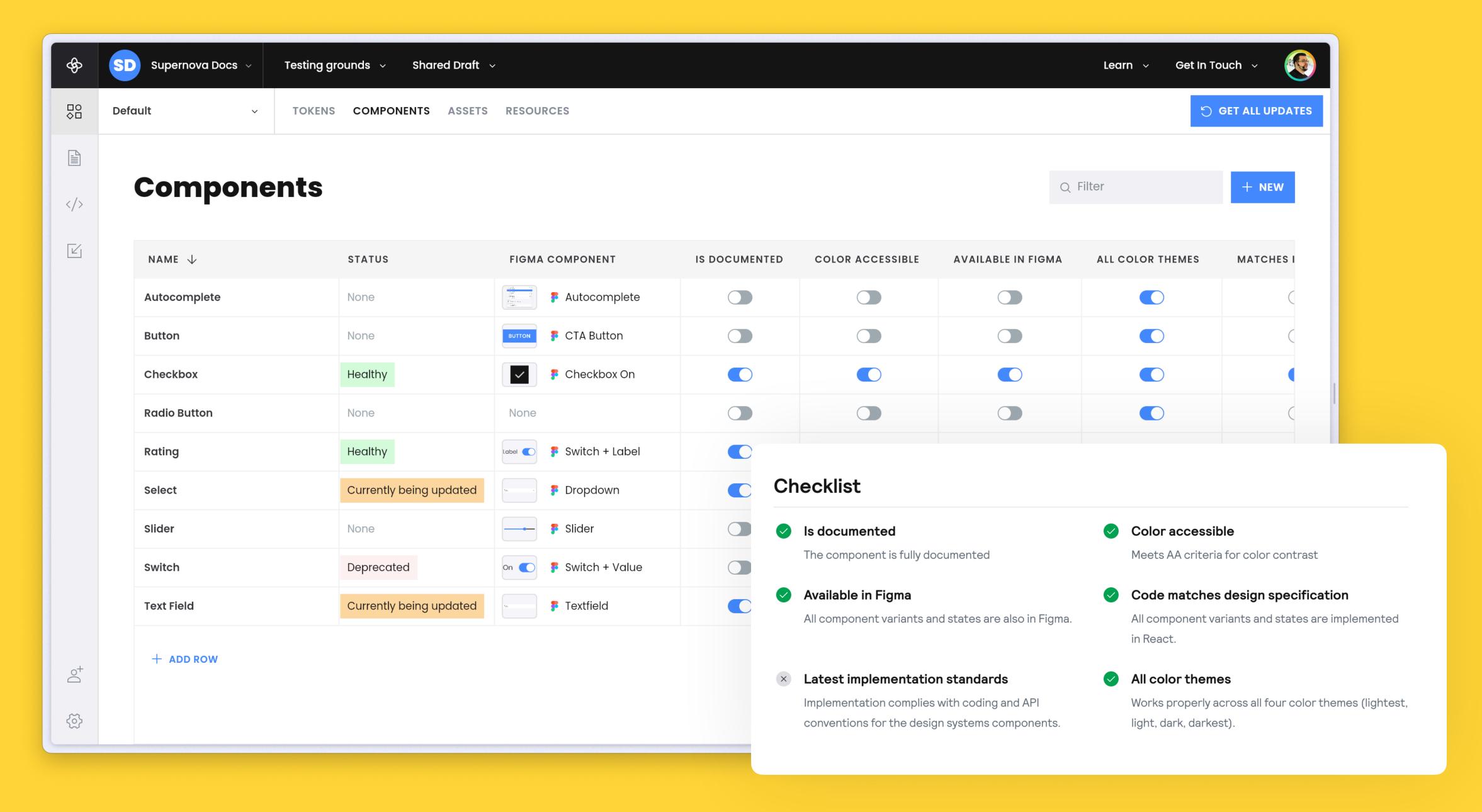
Generated UI kit

Includes a downloadable XD file, generated by code using design tokens defined in Spectrum DNA, and shows multiple options, states, color themes, and platform scales.

In Spectrum for Adobe XD plugin

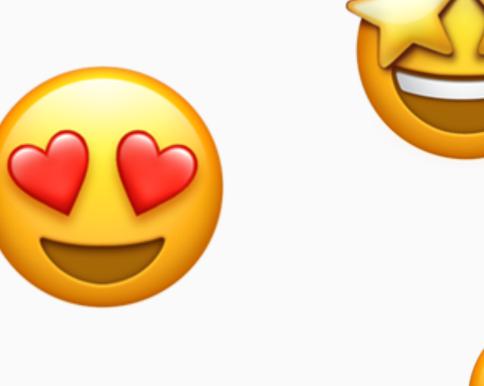
Component is included in the Spectrum for Adobe XD plugin.







Satisfaction







Observability — Satisfaction / From Productboard

Focus	Impact ▼	Question	Factor	<u></u>	Favorable score	Trend	Comparison
	N/A	Nucleus documentation gives me all the information I need to use Nucleus.	Nucleus engagement	2	65	-	-
	N/A	I am interested in contributing to Nucleus.	Nucleus engagement	1	82	-	-
	N/A	Nucleus is flexible enough for what I need.	Nucleus engagement	0	76	-	-
	N/A	Using Nucleus saves my time significantly.	Nucleus engagement	0	100	-	-
	N/A	I have noticed inconsistencies in Nucleus.	Nucleus engagement	3	40	-	-
	N/A	It's easy to find information about Nucleus I need for my work.	Nucleus engagement	0	71	-	-

Things that are working well for Productboard

These were the most agreed with questions across the survey.

Nucleus engagement

Using Nucleus saves my time significantly.

100% agreed

82% agreed

I am interested in contributing to

Nucleus engagement

Nucleus.

76% agreed

need.

Nucleus engagement

⊙ You and 18% were neutral

Nucleus is flexible enough for what I

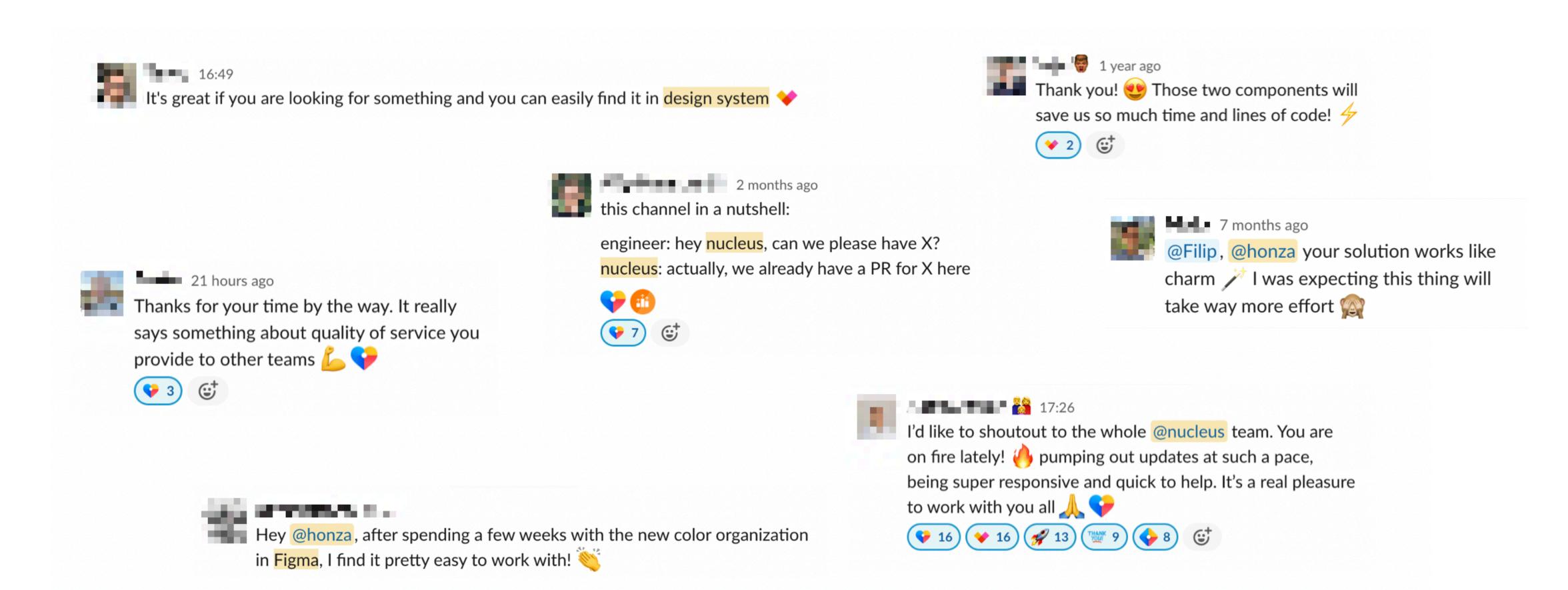
You and 0% were neutral **⊙** You and 18% were neutral

Results from System Usability Score survey

Neither agree nor			
	Q2/Q3 diff FE	Q2/Q3 diff Des	
I like to use Orbit frequently.	0.36	-0.29	Strong improvement
Orbit is unnecessarily complex.	0.01	0.43	Stayed same basically
Orbit is easy to use.	0.02	-0.20	Strong impairment
I've noticed inconsistencies in Orbit.	-0.41	-0.37	
Orbit is well documented.	0.43	-0.26	
Orbit doesn't work the way I want it to.	-0.03	0.09	
It's simple to integrate Orbit into existing projects.	-0.06	0.60	
Orbit makes my job harder.	-0.08	0.14	
Orbit saves my time.	0.01	-0.14	
I needed to learn a lot of things before I could use Orbit.	-0.14	0.06	



Tip: Collect praises from the users





Research & continuous discovery

To evolve the system.











Look for impactful opportunities

You have a lot of data and insights already.







Go back to your notes from initial research. Interview people again.

On contributions

- "If I know that our button looks like this, I will t should propose it to our design system, and it
- "If I come up with a new component or variation and how it should be prepared in Figma so the ensure that it will get to dev phase - that it will
- Described ideal process:
- It should be clear where to go with new present
- A designer will describe the needs of a needs and how to adjust it if needed. Or to confir
- Then we will push it to development, or we system lead should be a facilitator of the w. something changed, documentation will be
- And that there will be some design review standards we need and that it's easy to wo

On typography

- "I just don't know what to use there is a hug different things, and it's a lot about my gut fee
- "I don't want to use 12 or 13px. Ideally, the sma
- "A design system should answer when to use

On colors

"I struggle with using our color palette. There details and information on how to use them, w

Other

On icons

 "Currently, there are different scopes of comp Textfield, but also some specific Portal compo

On documentation

- "What if I have four buttons next to each other be structured?"
- "I can have some component guidelines open be easy to find those components in Figma."
- "Our tooltips have several different delays, bu that are. It's not documented anywhere."

 "I create small Loom videos, where I explain why some" so far (a couple of people commended it on their own),

- On the current state that is still in progress." Note: context menu ⇒ list of There are components from Sketch import, adjusted a
- are more mature (buttons use auto-layout, etc.) "Petur started with dropdown, Zdeněk with text field –
- "Typography isn't clear even to us, designers. It's hard

- "Unifying approach to Ul is needed it was manageab
- growing, there is also growing risk that we won't be ab "To enable changes in our designs, specifically usability freshness and adding a delight – it's connected to poss

the individual styles - which makes also onboarding ha

- "It should make future changes less expensive."
- "It should increase my productivity easier prototyping or prototypes, have focus on the new feature vs. recrea On expectations

On prototyping

- "Sometimes in prototypes, we need to use as real data worth it for some cases - because customers won't ge They aren't able to imagine it in their context."
- "The faster I can build a prototype, the better."
- "It would be great if we could prototype in code Fign possibilities and freedom in code. It's also more trustwo

On visual style

- Fear that our product will feel outdated and that the "Visually, we didn't innovate much in the past year."
- "People are judging us with an optic of Atlassian, Herol complex tools with a lot of good things, their UI is impre
- "We want to keep the brand and product close to each

On contributions

- "When I use something more often, I try to push it to th need to recreate them repeatedly."
- "We should keep the barrier of entry low" if you give more willing to meet for 30 minutes, design a compone would help me to deliver it."
- "Don't assume or be too harsh on new contributions guidelines when needed."

On design handoff to developers

On the current state

- "Currently, there is a lot of things, that could be solve
- "It's really easy to use current dropdown that is in de
- "We miss a lot of tests jest or snapshot tests."
- "We have a problem with accessibility it doesn't ex

On documentation

- "docz or storybook should show what is in the design components – "When I started, it was quite hard t
- what. Something was hidden, and not everything "It should be clearly described what can be used
- "It's great when I can click through all variations a

- "It would be great if the design system could solve th
- "There should be a clear guide on how to write com; what to belongs to our design system and what does
- "Design system should contain components that are
- "We don't have a unified design across our sections. implemented several times, and it's hard to use them deprecated things or older visual style or technology
- "It should bring consistency for our customers and m

On contributions

- "I know that there were some situations when one te but it was getting delayed by combination of comple priority work in that team. So we were delaying a rele end, we just copied what they had and released it. A ready, we replaced it."
- "I think it's possible that it will work with some kind o something that works for their case and someone els And when it's ready, we can replace it."
- "Basic things should be solved so we know how it we don't." Note: from technology point of view
- "I am for some hybrid approach. A fully centralized a be able to deliver it fast enough. But platform can se linters or similar - and everyone should use it."
- "I miss someone who could code review design syst

On working with design "Typically, I get a solution that is 80% done; a lot of ir

Expectations

- "We should start the contribution process work on de us. And you'll be the person who will be an owner of m system - like how we create components, how we app component, or should we override?") - on both sides: o
- "It could help accessibility and general usability so us and they could expect it will work like that everywhere.
- "Design system should give us less options so every de using their own variant of a component."
- "We want to refresh our visual style and it is easier or can quickly recreate all screens, and then you try how

- "We're designing with white-and-grey UI, combined wit other products, or, e.g. how can we now work with bore use shadows instead, and it could make our UI a bit lig
- "We're not in the situation that we need to make some to have some tools like modular and systematic UI so v
- "productboard is very content-heavy applications so it should be a secondary thing. The primary thing is the users and that users know how to work with it."
- "Check Figma for example their UI is not heavily brar web, marketing. And that can be our case too - that we because of the design system, and then when we disc

On design debt

 "Typically, I imagine a situation where I need to solve s solution how to solve the problem. But I can't apply this some behaviour in eight more places. But I will never c design debt gets created...

On contributions

- "We didn't learn yet how to push some helper componunified library - or how to approach the naming of thes
- "All designers should understand what obligations tow. creating new stuff - what to create, how to define it, w "When we were building a design system in Y Soft, we
- and a broader team who then implemented it." "Design changes could be approved when there is a m say four designers out of seven. If there's gonna be an
- "We should start with the contribution process and clea and then we can change or adjust our direction if need how things should be done, structured, etc. Let's start : Other notes to mention

What should a design system solve?

- solve "atoms" like typography, white space, accessibili "It would be great if developer like me doesn't need to use what is already in the system and build it from that
- sidebar with toggles, I have everything available in the not sure how to solve line-heights in typography – me is not sure if it's even a good thing - e.g. Figma or Ske like overengineering.
- "Could a design system help us with z-indexes? We ha values in millions just to override something else."
- The sidebar has probably the biggest CSS debt, a lot of

is polished a lot. Although everyone is afraid to touch i

- "Dropdown is something I need probably most often."
- "Hopefully it decreases a cadence of design feedback mentioned that it happens in code and Figma – proba
- "It speeds up my job already, with the components we more, and I will have time for other things."
- "I expect it to make our UI more consistent e.g. that f expect that it will make productboard 'more mature' count on when they pay big money for it."
- "In the short-term, I'd like to have things that I can requ need to rush it. Quality is important, foundations need
- "In the future, it can be a nice promo for productboard

Where can it fail? Where can it complic

- Generally, it can fail if no one will build foundations, bu it's now) – so the whole activity will take years to finish change designs. So the whole thing will break apart.
- That it won't be able to react on design changes quick ■ That there won't be a will to refactor old code – e.g. th
- replace the old ones. Like it happened with tooltips or "It's great that in productboard, there are mostly senior work on removing technical debt. But usually, there ar clean-up weeks after each initiative, but now they hap
- "It can bring asynchronicity to my work" for example the bottom, he will need to wait when it's done. He like encouraged to contribute. Now it's encouraged that ev need to adjust something quickly, I won't have time to attend design systems syncs, so I have some overview overwhelming for some people and they maybe don't
- "We could have shared code ownership that I may o someone then adjusts what's needed."

to it later. Then Kaja and Pavel started to push it more." "We are somehow starting with a design system - we don't have a clear vision, how it should look, what the requirements are, and how our contribution process should look like."

On the current state

"When I have some new design, so most of the components there is already defined in the design

"Newer components like tooltip or popups are quite okay; they don't have a business logic inside."

"That idea of design system is here for a few years already, to be more systematic in the

"Peter with Majo created some basic foundations like typography, colours, paddings, but the

whole activity silenced then - we pushed a design system tasks down, and we weren't able to get

- system on design part and code part too. "It should speed up how we work with design and designers – we can even skip Figma and implement some small changes directly in the code. We can have an ad-hoc meeting where a designer sketches something on the paper, and we can implement it in parallel."
- "It should bring uniformity on how we call components and elements not everyone is using the same name for dialog, popup, modal. So the communication is tricky sometimes because I talk about modal and the other talks about dialog, but we all have the same component in mind."
- "Modal is something that informs me about some state, dialog requires a reaction." "I am afraid of the situation that when we need to ship a feature, and it will take several weeks to core design systems team before they deliver a new component for us. And it will block us. This needs to be solved somehow."

- "I'd like it if the whole engineering contributed but from a practical point of view, it's problematic When we have deadlines, we have to deprioritize everything that's not a feature. Which also
- "Idea of decentralized contributions is great, but I am a bit sceptical about that. I can imagine that if someone from the core team would help me to finalize my component if something is missing."
- possible edge cases, or we don't do performance testing. We just don't have that much of time for it; we need to make it work for the use case we have in the initiative, not for all other use cases."
- "The biggest issue is a time of devs work on the initiative is always a priority."
- "I am not sure if we had any direct contribution to our design system. Usually, when we agree that something should be in design system, we need to ship, so we don't have time to sync with everyone. So we ship it with the intent of follow-up that we will bring it to design systems sync, but in the end, it happens rarely."

On components in design system

- "Components in our design system should have solved possible edge cases, should be covered
- "Components should be shielded from any business logic. Now it happens that there is some custom feature logic baked into components and it shouldn't be like that - I expect that I will get a visual representation of how a component should look like and I don't need to hack it for my

On running experiments

In growth team, mostly because we need to move fast and we're not sure if the experiment will be a success, we don't want to spend much time perfecting some solution. So we need to code it



Be close to where your users are and track their feedback continuously



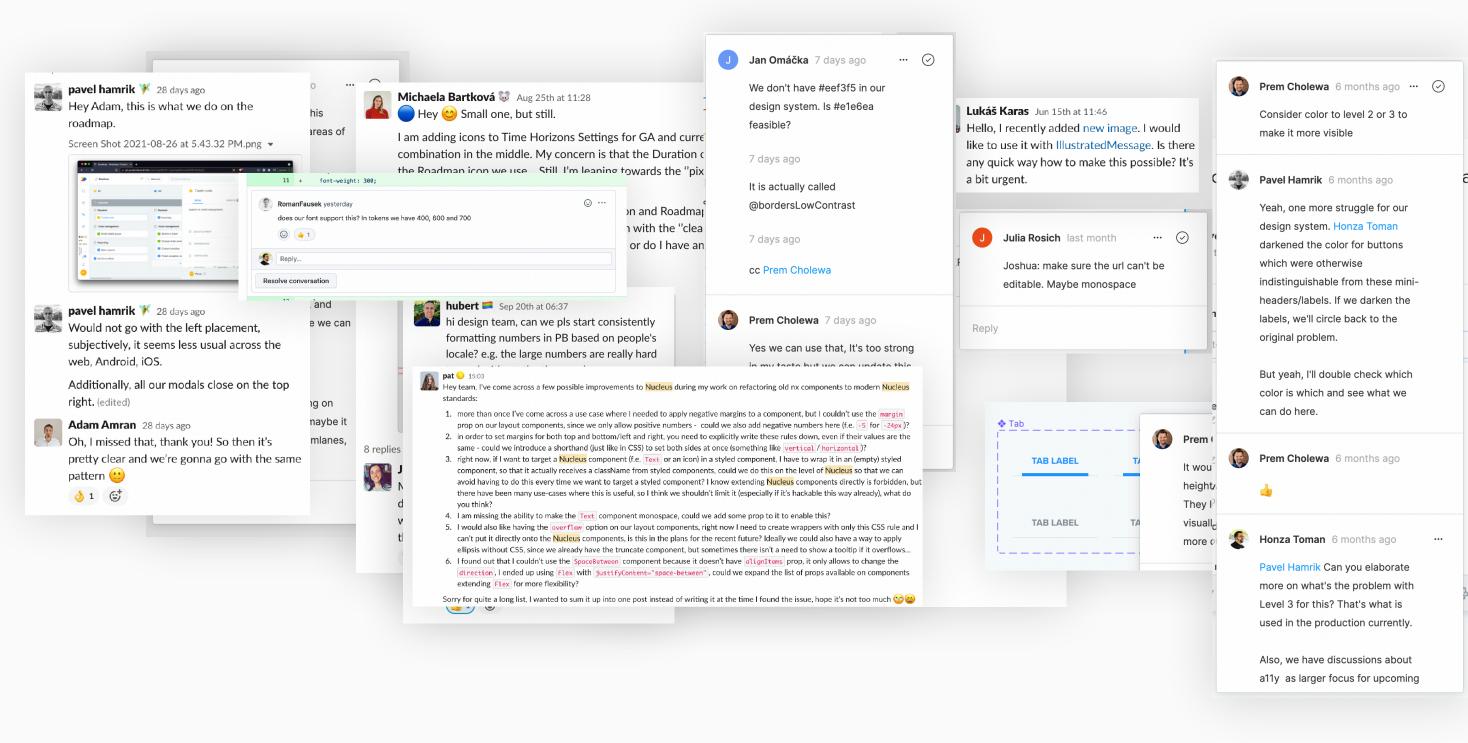


GitHub

We need weekly interactions with customers, because we are making product decisions every day.

Source: "Continuous Discovery" by Gerard Chiva

Internal conversations about design, patterns, and UI development



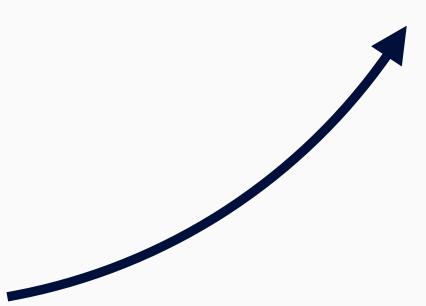


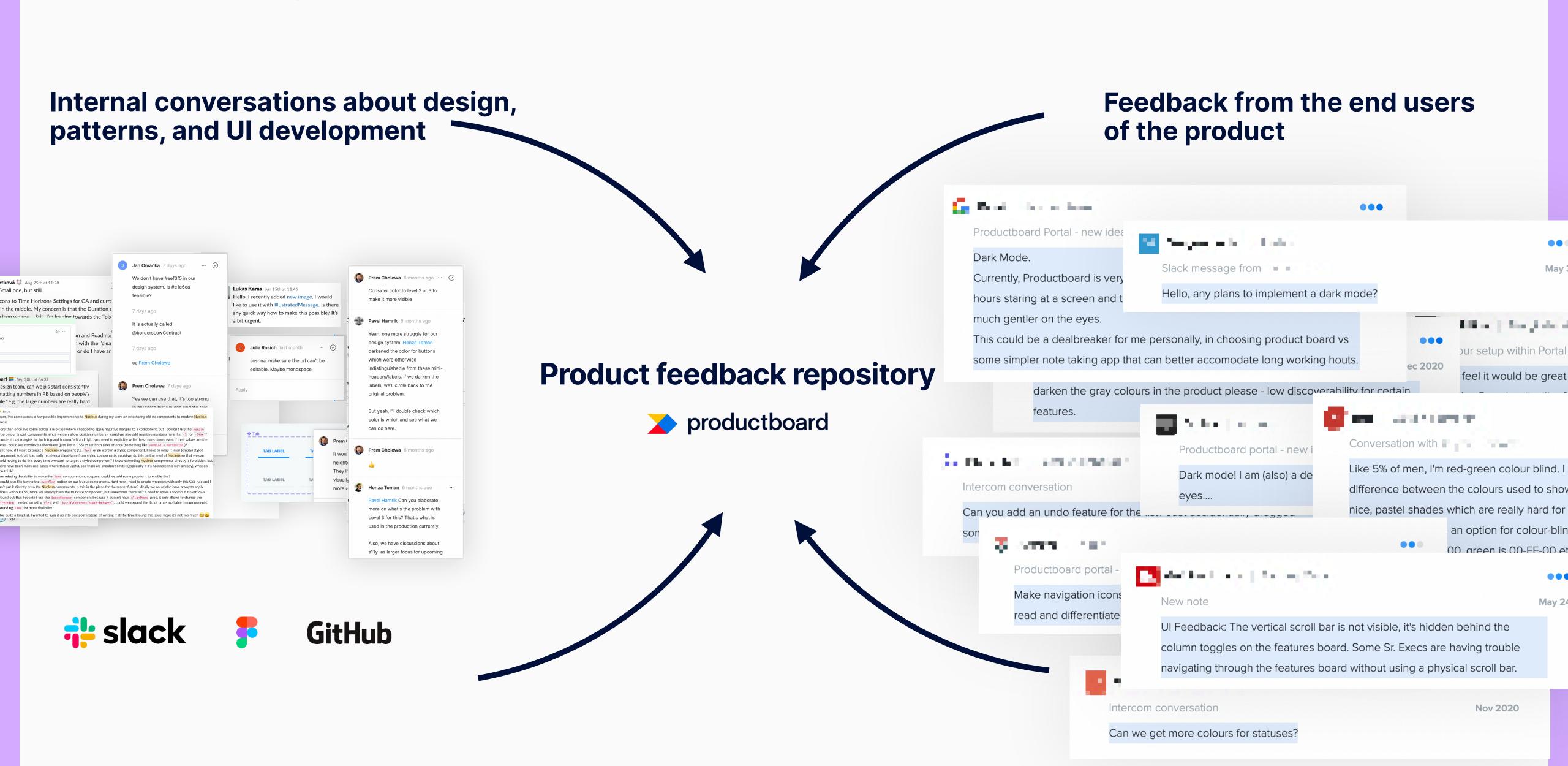




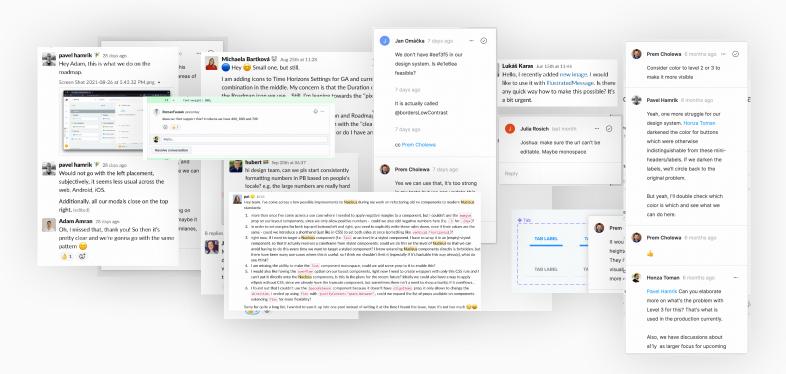






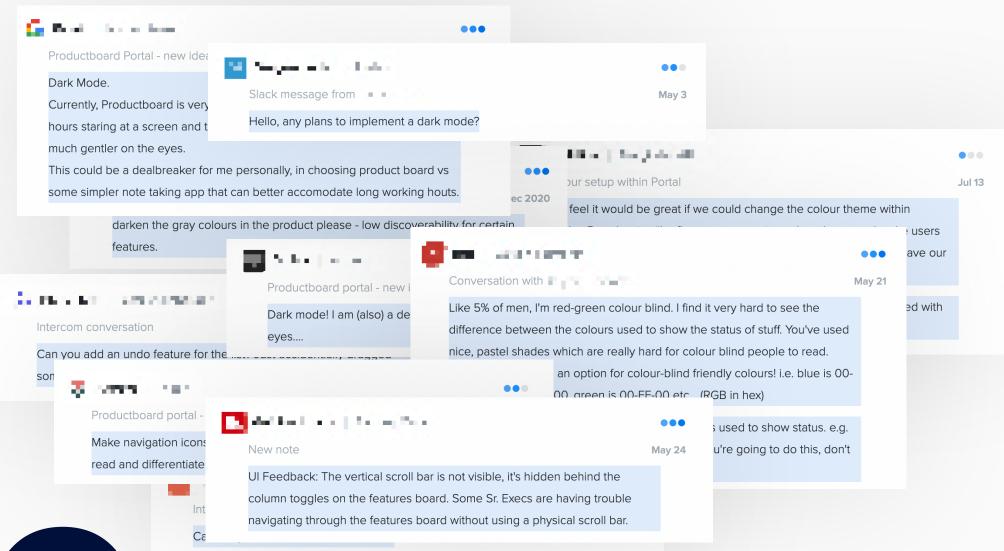


Research & continuous discovery / From Productboard



Product feedback repository

productboard



650000H

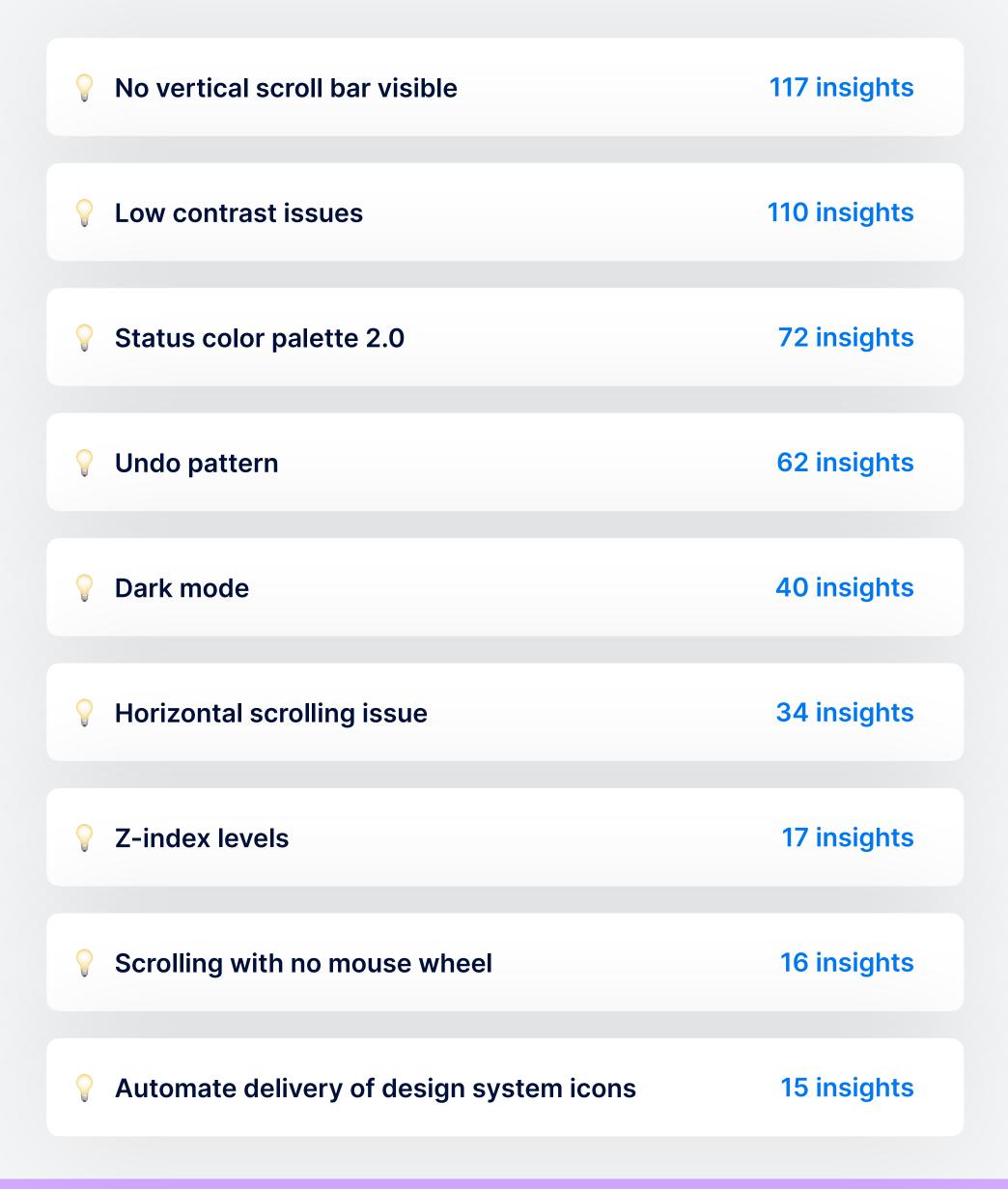
pieces of feedback ("insights") collected over past 8 years

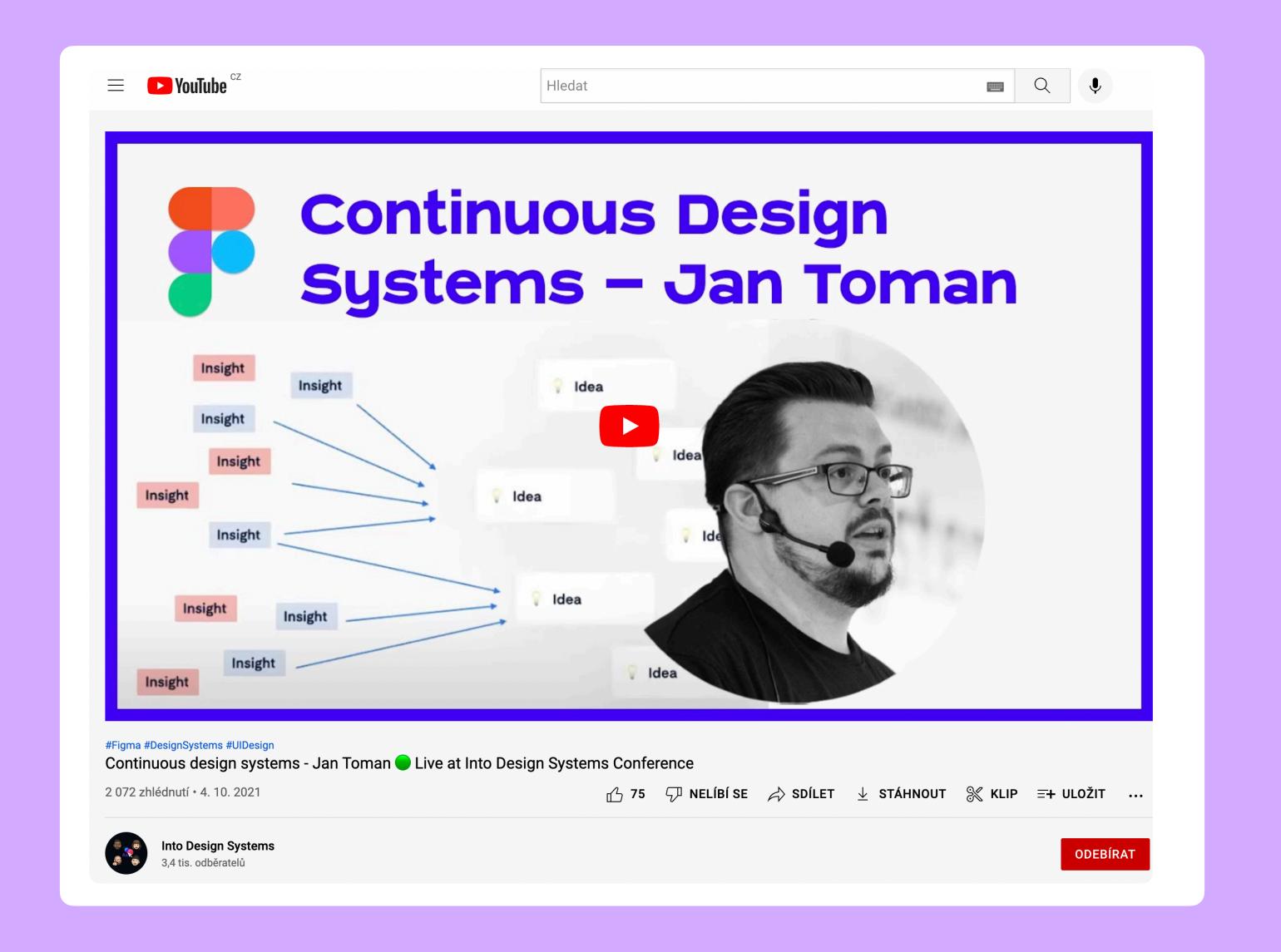
5-8%

was somehow related to a design system and UI/UX

Result

We had a strong discovery backlog backed by feedback from the users







Let's wrap it up



A lot can be tracked for design systems 👀

Identify which data are important Adoption for your design system to have. Components seen by end users Focus on those. Screen adoption overlay Component usage Amount of "snowflakes" Figma component inserts Amount of local libraries Component props usage DS version adoption Time to render Percentage adoption Component overrides Design tokens usage Figma component detaches NPM package uses Usage by version Accessibility score Documentation visits CSS Audit Qualitative Quantitative Deprecated components usage Speed to market Google Core Web Vitals Feedback from users External contributions UI consistency Bugs backlog Number of components Task time for bigger patterns User satisfaction UX issues backlog Platform status Amount of requests from users Test coverage User interviews OKR completion 👸 ROI SUS surveys Usability testing Ticket request volume Components Stakeholder interviews Component average cost Developer productivity First response time Support Lines of CSS Designer productivity
 Parity between code and design Business Linter warnings

Average resolution time Component Health Code excellence End-user related Overall resolution rate Code complexity Code platform Governance Other

Summary

- Understand how people use the system
- Track how the system is extended and customized
- Have strong feedback loops with users of a design system
- Partner with your data team you don't need to figure it out alone
- Keep stakeholders informed about key metrics (e.g. adoption)





Thank you!

All mentioned links and resources on bit.ly/dslondon-overview





