

 Ad Hoc

Past, present and future

# CMS Design System

# Hey, I'm Scott

Since 2004 I've focused on improving online experiences for US citizens. My focuses include design systems, front-end user experience, web standards, and accessibility.

Currently, I work for [Ad Hoc](#) as a Managing Senior UX Designer 2 : Front-end Specialist and since 2019 I've served as the design lead for the Centers for Medicare & Medicaid Services design system (CMSDS).



# CMSDS: Past, present and future

## → How it started

- ◆ What's a design system and why is it important?
- ◆ Challenges
- ◆ Changes made

## → Version two is released

- ◆ Challenges
- ◆ Changes made

## → Our future vision

- ◆ How are we getting there?



**How it started**

# **In 2017 we were a team of one.**

A single designer/front-end developer

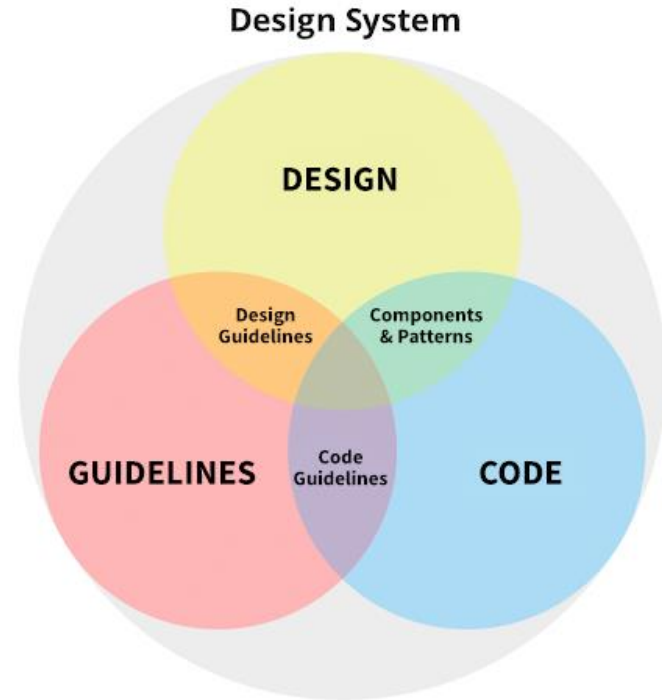
**CMS wanted all teams working on CMS projects to build consistent, responsive, and accessible experiences.**

# **We needed a design system!**

But wait... what's a design system and why did we need one?

# What's a design system?

A design system is a collection of reusable components both in design and in code, guided by clear documentation and standards, that can be assembled together to build web experiences and can be easily updated within products.





# Why is a design system important?

- Allows designers and developers to create quickly and at scale.
- Allows product teams to focus on larger, more complex problems.
- It creates a unified language within and between cross functional teams and products.
- It creates visual consistency across products, departments and experiences.
- It can serve as a tool and reference for junior-level product team members.

# What's the difference between the CMSDS and the USWDS?

# USWDS and CMSDS design systems



USWDS was created by a team at 18F and the U.S. Digital Service in 2015 under the guidance of an advisory board of talented, experienced government workers.



CMSDS was created by contracting teams Ad Hoc and Nava in 2017 in cooperation with the Centers for Medicare & Medicaid Services.

# CMSDS builds on the work of USWDS

The CMSDS is it's own set of components and therefore we can make make changes when we see opportunities.

## **Similarites**

Components, Patterns, CSS class structure, Responsive grid, HTML examples

## **Differences**

React components & patterns, NPM delivery, CDN delivery, Specific to CMS needs

# Initial rollout of CMSDS

The CMSDS was first adopted in several HealthCare.gov products.

These products needed specific customizations, which were originally supported through the concept of site packages.

Site packages contained styles and components unique to a particular CMS site and were built on top of the CMS Design System.

# One design system & site packages



**CMS Design System**



**Medicare  
Site package**



**Healthcare  
Site package**

Medicare product

Medicare product

Healthcare product

Healthcare product

**By mid-2019 we grew to a team of 2.**

One designer and one engineer



**Version one challenges**



# Challenge **Complex and overly technical**

Site packages had a large technical overhead and were difficult to use and maintain.

- Difficult to build, publish, and test due to no standard practices.
- Difficult to update the documentation site due to a high technical proficiency required.

# Challenge Multiple design/dev resources

Their were four overlapping resources with little-to-no coordination or maintenance across them.

1. <https://styleguide.healthcare.gov/> - A digital style guide for the website
2. **A brand guidelines PDF** - primarily for printed material
3. <https://assets.cms.gov/> - A front-end framework for building accessible and responsive websites
4. <https://design.cms.gov> - A documentation site for the design system

# Challenge **Small team size**

The CMSDS team was very small early on and this made it hard to mature the design system simply because there were not enough people to do all the things.

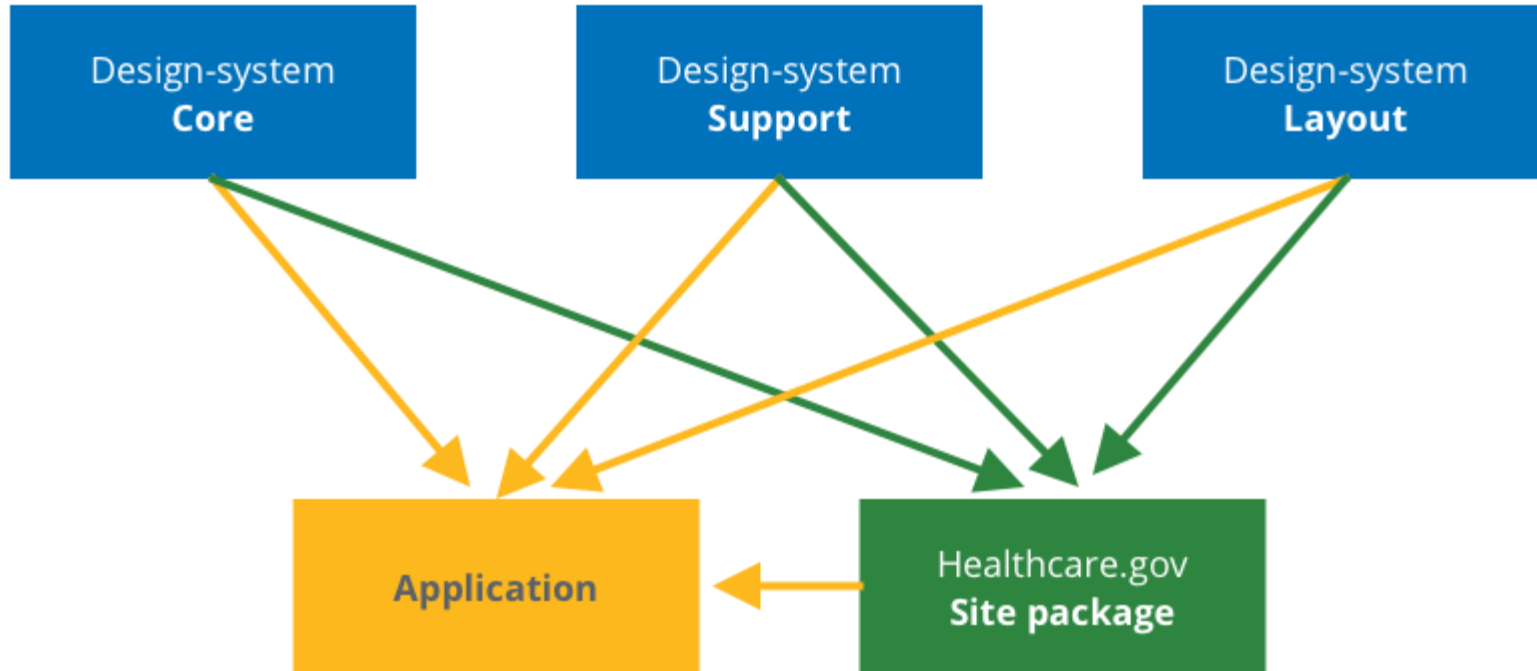
Develop new features, fix bugs, make enhancements, write guidance, create design assets, define processes, offer support, and more...

# Challenge Multiple codebases

The Design system consisted of multiple codebases maintained by different teams, with no method of sharing common code or processes.

- Code is duplicated and a lot of visual inconsistencies were the result.
- Each codebase can't benefit from bug fixes or improvements in another codebase.
- Code isn't easily shareable across codebases

# Challenge Dependencies



A scene from the movie 'The Hobbit and the Moon' showing four young men standing in a mountainous landscape. They are wearing medieval-style clothing, including capes and tunics. The background features rocky, snow-dusted mountains and a stream. A blue text box is overlaid on the image.

**Version one changes**

# **By early-2020 we grew to a team of 4.**

One designer, two engineers and one product owner

# Change Simplified dependencies

## What we did

The complexity of the dependencies were simplified from 4 down to 1.

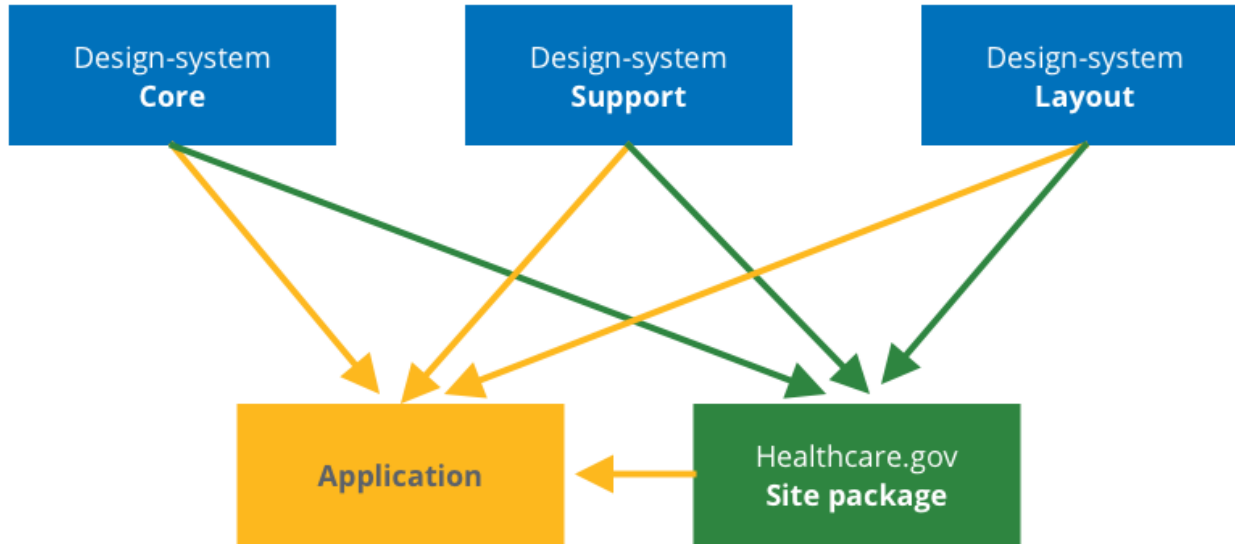
## Why we did it

Made it easier on product teams because they only needed one site package.

Made it easier to track versioning and lessened the complexity of version matching across multiple packages.



# Change Simplified dependencies



# Change Created shared scripts

## What we did

We created shared scripts to use across all 3 codebases for things like linting the code, running the doc site, running tests, and building the doc site.

## Why we did it

This gave us a more consistent approach across all 3 codebases which promotes developer best practices and consistency while making it easier to support, extend and change.

# Change Started using InVision DSM

## What we did

We started using InVision DSM to capture specific documentation and guidance in addition to the documentation sites.

## Why we did it

The InVision DSM allowed for more non-technical users to contribute documentation and guidance to the design system.

# Change Organized doc site source code

## What we did

We better organized our documentation site source code.

## Why we did it

This organization made it easier to find and modify code, styles, examples and documentation across multiple codebases.

# Change Deprecation notice on styleguide

## What we did

Added a deprecation notice to [styleguide.healthcare.gov](https://styleguide.healthcare.gov)

## Why we did it

We added the deprecation notice to help direct users who landed on [styleguide.healthcre.gov](https://styleguide.healthcre.gov) to [design.cms.gov](https://design.cms.gov) instead.

# Change Site package naming

## What we did

We updated the name “site package” to start using “child design system”.

## Why we did it

“Child design system” created a stronger mental model of the relationship between the core and child systems. In addition, this naming tied into larger goals of increasing design system ownership and autonomy within Healthcare.gov and Medicare.gov.

# CMS Design System (core)

The core CMS Design System (CMSDS) provides shared resources (code, styles, guidance) via an NPM package and a public documentation site.

## The CMS Design System contains

- Components
- Styles
- Code examples (React and HTML)
- Documentation site
- Design assets (Sketch)
- Developer tools and scripts for maintaining a Child Design System

# Child Design Systems

A Child Design System provides product teams working under the same CMS sites (i.e. Medicare.gov) a way to share site specific resources like branding, colors, logos, components and guidance.

## Child Design Systems contain

- ***Inherited*** components, styles and code examples from CMSDS
- ***Inherited*** developer tools and scripts from CMSDS
- Specific components, styles and design assets
- Documentation site



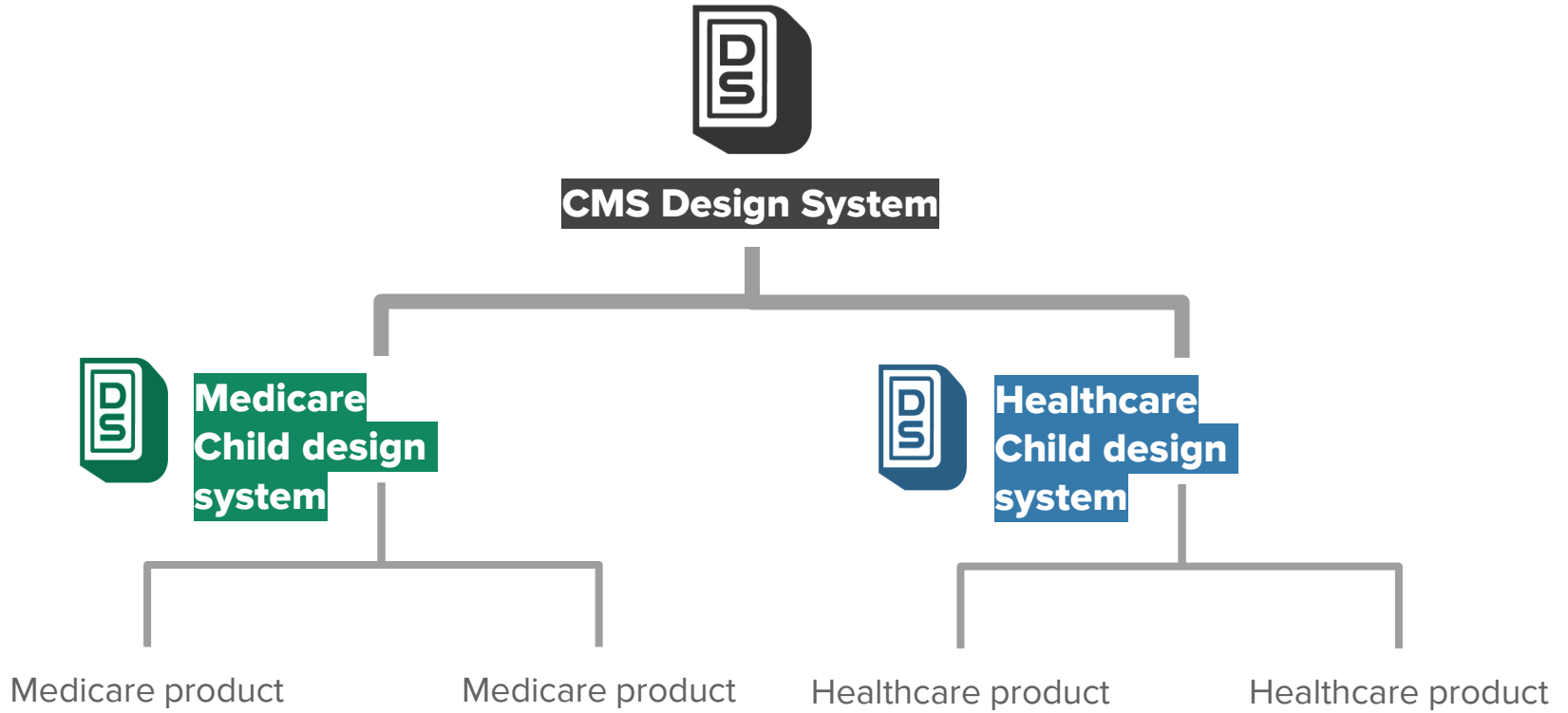


**Version 2 is released!**

# **By mid-2021 we grew to a team of 6.**

Two designers, two engineers and two product owners

# 3 Design Systems



“

**A design system is the official story of how an organization designs and builds digital products.**

”

— *Brad Frost*

# The design systems are siloed



Components



Patterns



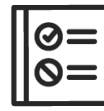
Component library



Reference site



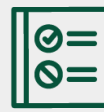
Guidelines



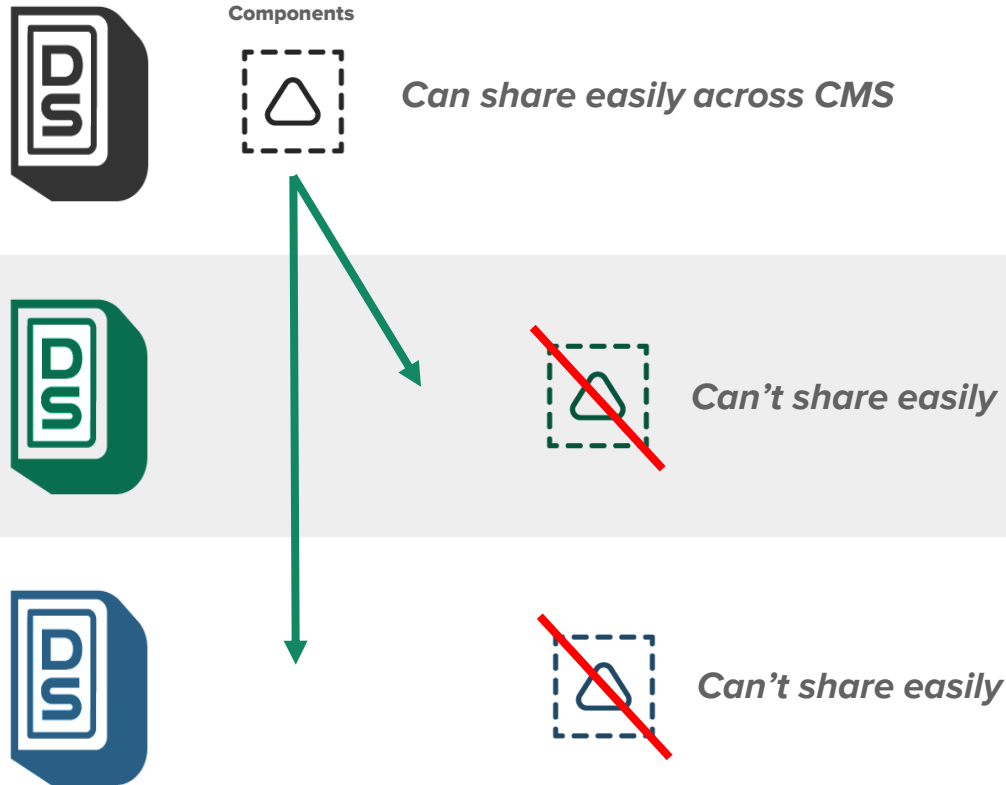
Governance



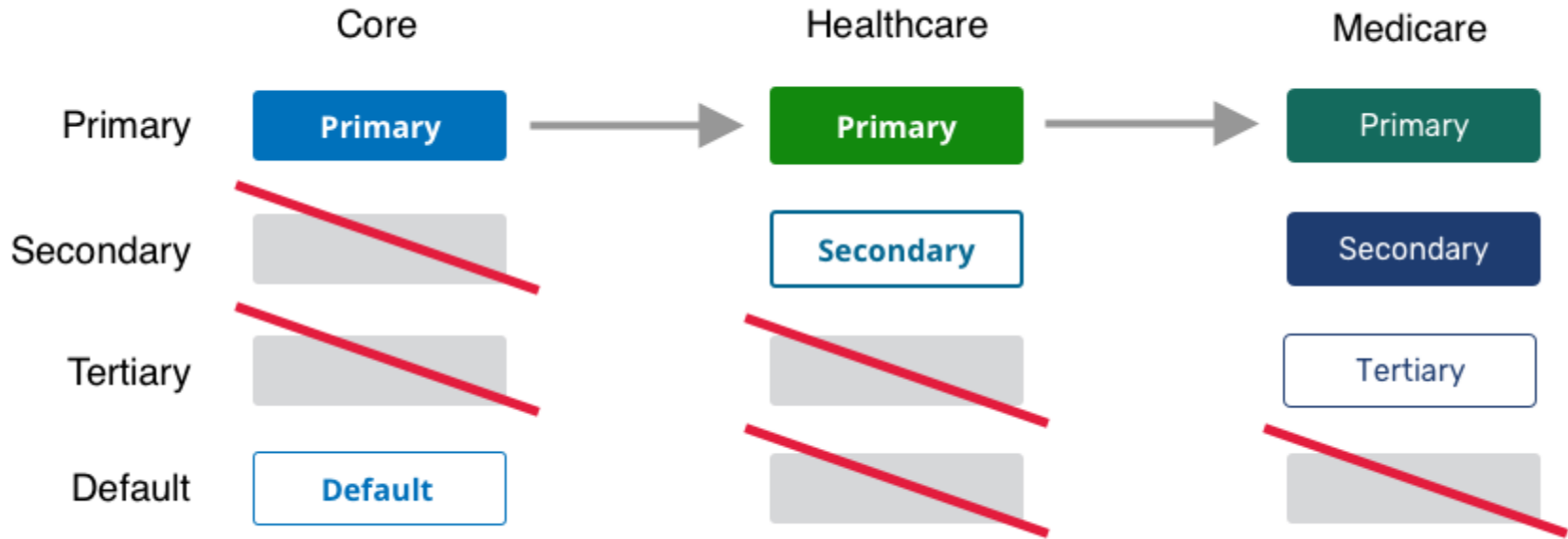
UI Kit



# Sharing across the CMSDS is still hard



**Buttons across the CMS Design System are a great example of these silos.**





# Current button variations are inconsistent

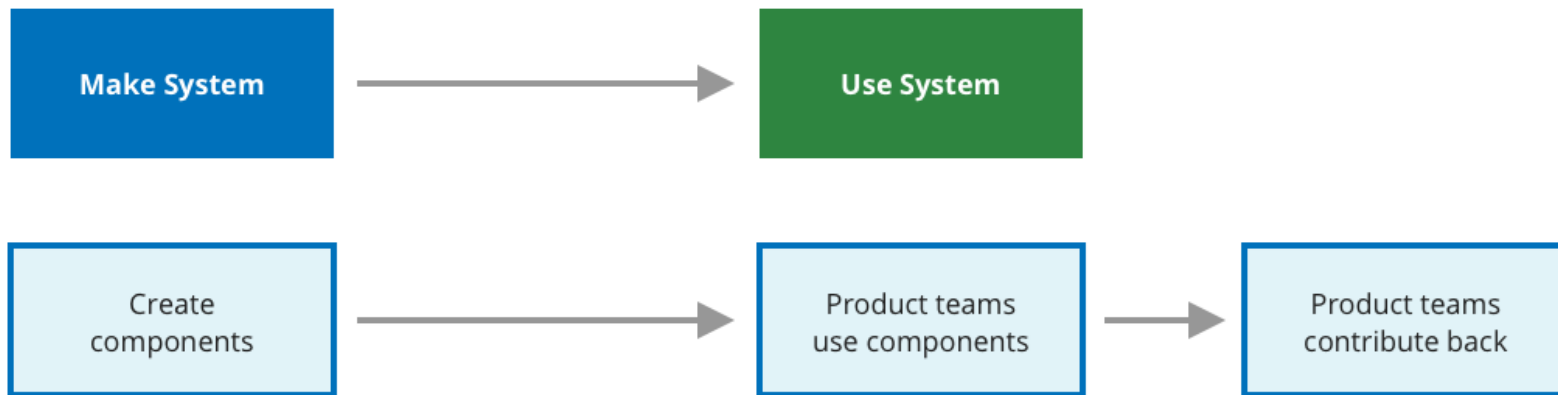
Current button variations are inconsistent across the design system and child design systems...but they don't have to be.

- **Styles** are inconsistent.
- **Naming** is inconsistent.

This button architecture doesn't stand up well to the flexibility needed within a multi-branded design system. We need a more systematic approach!

**Our process for adding components into the design system had a lot of friction.**

# Our process



# Our current process doesn't work

## Create components first

- This is hard because we don't have real use cases and it is an abstract exercise.
- This is also hard because it involves knowing what product teams need before they need it.

## Product teams contribute back

- Product teams don't have time or capacity to do this work.
- Product teams need to learn the design system processes.
- Product teams need to adopt a systems thinking mindset where they are focused on the Design System as a whole.

**What happens when product teams run into a limitation in the design system?**

# Where can I?

- Find documentation
- Find design assets
- Ask questions
- File a bug
- Propose something new
- Request a change
- Contribute





**Version 2 challenges**

# Challenge Large maintenance overhead

There is a lot of maintenance overhead to maintaining multiple design systems. Most of what gets published is duplicated effort (doing the work 3 times).

- 3 separate codebases
- 3 separate documentation sites
- Duplication of guidance leads to confusion or incorrect usage of components
- Duplication of efforts and solving the same problems across CMS leads to inconsistent experiences.



# The landscape



Sketch Kit - Mgov



GitHub



Invision DSM - HCgov



Invision DSM



Design.cms.gov



HCgov doc site



brand guidelines PDF



GitHub - HCgov



Assets.cms.gov



Sketch Kit



styleguide.healthcare.gov



Invision DSM - Mgov



GitHub - Mgov

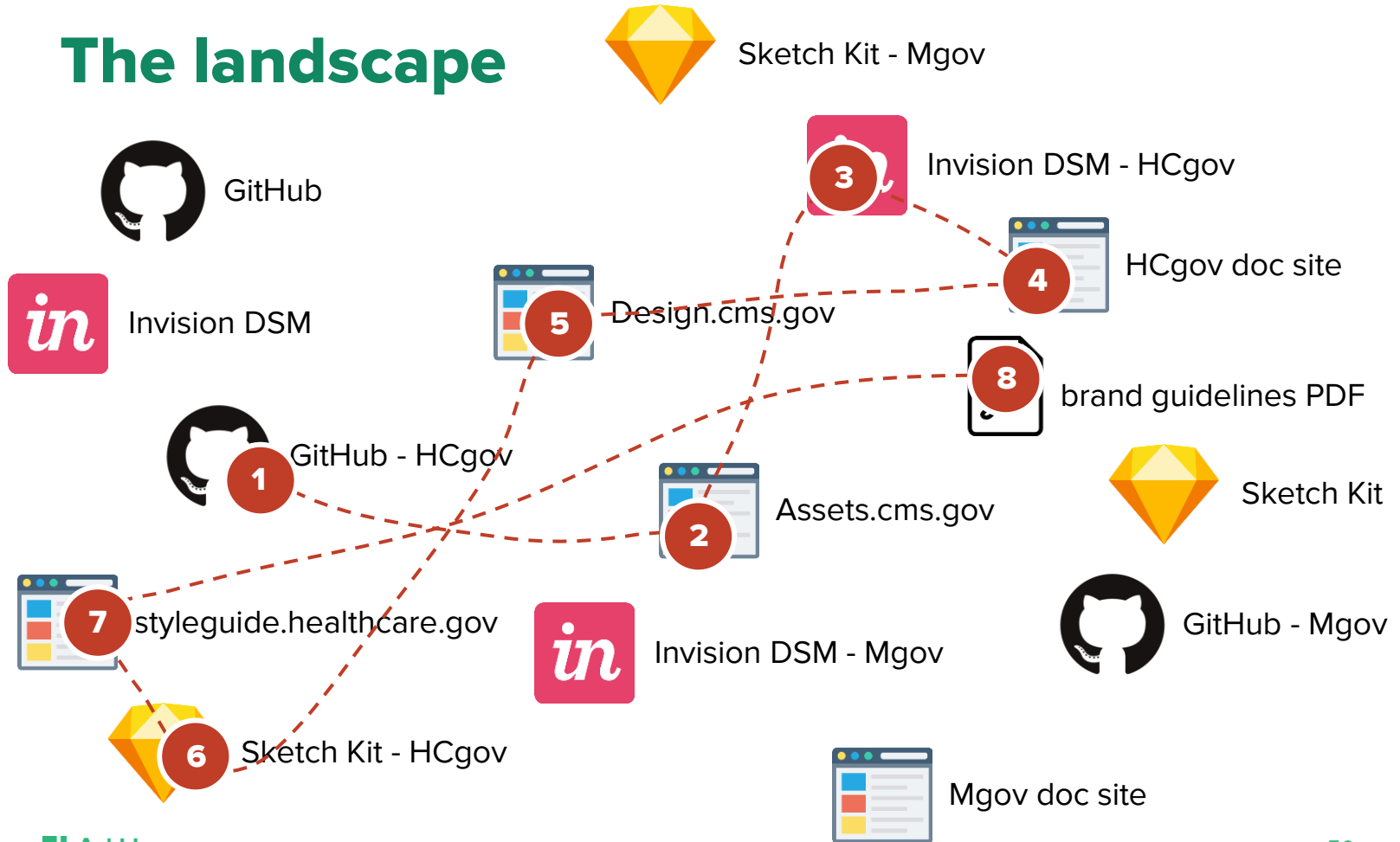


Sketch Kit - HCgov

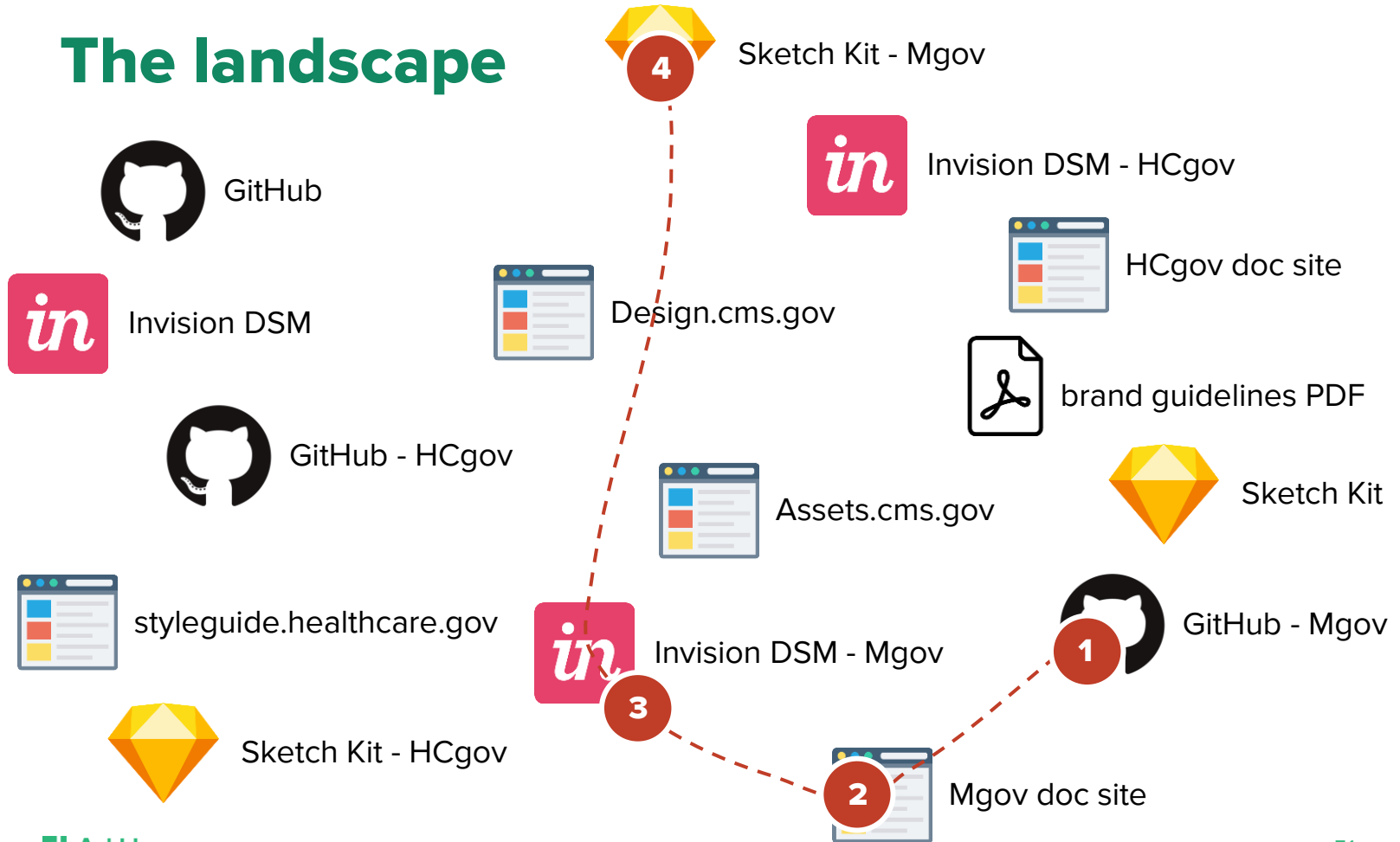


Mgov doc site

# The landscape



# The landscape



# Challenge Digital sprawl and tech debt

Over time the design systems will grow apart leading to digital sprawl and technical debt.

- Increased time and energy used updating child design systems to the latest core version
- Multiple layers of updates need to happen to get updates to product teams
- Multiple versions of the same component or pattern with differing accessibility and UI adherence

# Challenge Complicated to contribute

Complicated or confusing contribution model leads to lack of contribution from product teams.

- Multiple places and processes for contribution
- Unsure which where to focus energy - core or child design system
- Encourages siloed thinking Healthcare/Medicare instead of system thinking

# Challenge Child design system ownership

No ownership of a child design system results in a wild west mentality where decisions are made arbitrarily.

- A lot of conversations happen at a child design system level but there is not a team or group leading the effort at that level
- Nobody to fix component issues or bugs
- Nobody to update to the latest version of the design system

# Challenge Our foundation needed work

The Design System was built on a SCSS architecture and wasn't very scaleable or systematized.

- SCSS conventions in child design systems differed
- Components did not have a solid theming layer
- Our colors were not systematized
- There was not integration between design and code assets

A group of characters from The Lord of the Rings standing in a forest. In the foreground, four Hobbits (Sam, Merry, Pippin, and Legolas) are looking towards the right. Behind them stand several other characters, including Gandalf the White, Galadriel, and Legolas. A blue box with white text "Version 2 changes" is overlaid on the image.

**Version 2 changes**



# **By late-2021 we grew to a team of 9.**

Three designers, four engineers and two product owners

# Change Consolidated codebases

## What we did

We combined the two child design system repos into one monorepo for the design system.

## Why we did it

This streamlined and simplified the codebase and made it easier to maintain and for product teams to contribute to.

# Change Took ownership of child systems

## What we did

The Experience team took oversight of the Healthcare and Medicare Child Design Systems.

## Why we did it

This gave the Experience team the ability to mature the child systems by getting them back in sync with the overall design system.

In addition, this gives product teams someone to communicate with about the design system.

# Change **Began exploring design tokens**

## What we did

We starting looking into updating the foundational elements of color, type, spacing, etc... within the design system to use design tokens.

## Why we did it

Standardizing and implementing design tokens sets up the CMSDS and any other consumer of our design language with a set of standards that will improve design consistency and workflow organization using nearly any tool-stack available now or for years to come.

# What are tokens?

Basically, Key Value pairs.

Key	Value
“lightblue-500”	“#0000AA”
“spacer-2”	“16px”
“font-size-xl”	“2rem”

*But we already have Sass variables for customization why another external dependency?*

With a mature design system supporting multiple brands, teams and codebases, a more flexible option for governance and configuration of variance between those systems becomes a necessity.

We also need a *source of truth* as a set of guidelines for teams & brands to follow.

# Change Systematized our color palette

## What we did

We created a systematized color palette for the entire CMS Design System build from existing theme and brand colors from Healthcare and Medicare.

## Why we did it

We needed a more robust and predictable foundational color palette to offer product teams designing experiences. This palette gives us stronger guidance, language, accessibility and design options.

#FFFFFF	#0071BC	#046791	#e0f2f1	#E1F3F8	#CFEFFC	#E8F1FF
#F7F7F7	#205493	#035578	#409e90	#9BDAF1	#3E94CF	#4F6B9B
#F3F3F3	#112E51	#023247	#146A5D	#02BFE7		#1E3C70
#F1F1F1			#0C3F37	#00A6D2	#F9DEDE	#12284D
#D6D7D9	#E7F4E3	#FFF1D2	#0f4b42	#046B99	#F8DDDD	#081120
#AEB0B5	#E7F4E4	#FAD980	#041412		#F5ADB9	
#BAC5CF	#94BFA2	#F9C642		#DD3603	#E59393	
#72777F	#4AA564	#F8C41F	#BD13B8		#E31C3D	
#5B616B	#2E8540	#F1AD15			#CD2026	
#323A45	#2A7A3B	#FDB81E			#B31E22	
#212121	#266E35	#E4A61B			#B20000	
#1f252d	#12890E	#CA9318			#981B1E	
#000000	#107B0D	#B18115			#8D0000	
	#0E6D0B					
	#0D5E0B					
	#174320					

# Our design token color system

Granite	Rose	Crimson	Persimmon	Goldenrod	Spring	Teal	
#f2f2f2	#fce8ec	#f7e6e6	#fcebe6	#fef9e9	#e7f3e7	#e8f0ef	
#d9d9d9	#f7bbc5	#e8b3b3	#f5c3b3	#fdedbc	#b8dcb7	#b9d2ce	
#c0c0c0	#f18e9e	#d98080	#ee9b81	#fce28f	#89c487	#8ab5ae	
#a6a6a6	#eb6077	#c94d4d	#e7724f	#fad662	#59ac56	#5b978e	
#8d8d8d	#e63350	#ba1a1a	#e04a1c	#f9ca35	#2a9526	#2c796d	
#808080	#E31C3D	#B20000	#DD3603	#F8C41F	#12890E	#146A5D	
#737373	#cc1937	#a00000	#c73103	#dfb01c	#107b0d	#125f54	
#5a5a5a	#9f142b	#7d0000	#9b2602	#ae8916	#0d600a	#0e4a41	
#404040	#720e1f	#590000	#6f1b02	#7c6210	#094507	#0a352f	
#262626	#440812	#350000	#421001	#4a3b09	#052904	#06201c	
#0d0d0d	#170306	#120000	#160500	#191403	#020e01	#020b09	
Sky	Dark Sky	Lapis	Ocean	Sapphire	Cerulean	Windsor	Orchid
#e6f9fd	#ecf4fa	#e9ecf1	#e6f1f8	#e6f0f4	#eaf8fe	#edeaf4	#f8e7f8
#b3ecf8	#c5dff1	#bcc5d4	#b3d4eb	#b4d1de	#c0e9fb	#c9c0de	#ebb8ea
#81dff3	#9fcae7	#8f9eb8	#80b8de	#82b3c8	#97daf8	#a696c9	#de89dc
#4ed2ee	#78b4dd	#62779b	#4d9cd0	#4f95b2	#6dcbf5	#826bb3	#d15acd
#1bc5e9	#519fd4	#35507e	#1a7fc3	#1d769c	#43bcf2	#5e419d	#c42bbf
#02bfe7	#3E94CF	#1E3C70	#0071BC	#046791	#2eb4f0	#4c2c92	#BD13B8
#02acd0	#3885ba	#1b3665	#0066a9	#045d83	#29a2d8	#442883	#aa11a6
#0186a2	#2b6891	#152a4e	#004f84	#034866	#207ea8	#351f66	#840d81
#016074	#1f4a68	#0f1e38	#00395e	#023449	#175a78	#261649	#5f0a5c
#013945	#132c3e	#091222	#002238	#011f2b	#0e3648	#170d2c	#390637
#001317	#060f15	#03060b	#000b13	#000a0e	#051218	#08040f	#130212



**At the beginning of 2022 we put out a survey to product teams using the design system.**

# 42.1%

of users either find our documentation not complete or thorough *or* are unsure where to find documentation and guidelines

**Design System Survey, 2022**

“

**To be honest, I never know where to look for what I need. Should it be in Sketch, the DSM, the public-facing site, elsewhere?**

”

“

**I know where to find the documentation...but find it inconsistent in what is there. What shouldn't be used vs. what is okay to use?**

”

What we need is

**a single source of truth.**



**Our future vision**



# A unified and themable CMS Design System

Components



Patterns



Component library



Documentation site



Guidelines



Governance



UI Kit



Design Tokens



Healthcare  
theme



Medicare  
theme



CMS.gov  
theme



Additional  
theme

# Benefits of a unified design system

- Less overhead to maintain a single design system
- Energy is focused towards a single consistent system that everyone benefits from
- Less technical debt and digital sprawl
- A consistent contribution model for all teams
- A consistent documentation site experience for product teams
- Design tokens integrated into code and design



Our hypothesis:

**Medicare and Healthcare products are more similar than they are different.**

# Medicare and Healthcare similarities

We think we can align the following similarities at a Design System level to support all of CMS (Medicare, Healthcare, etc...)

- **General guidelines**
- **Components & Patterns**
- **Processes**

# Medicare and Healthcare differences

We think visual styles are where Medicare and Healthcare differ the most. These styles should be modified to fit each unique brand.

- **Color** - Brand colors and supporting palette
- **Type** - Type style and settings
- **Spacing** - Content formatting and space settings
- **Branding** - Specific logos, illustrations and imagery
- **Specific guidance** - Specific guidance as it relates to Medicare or Healthcare

# One unified Design System

## Design system

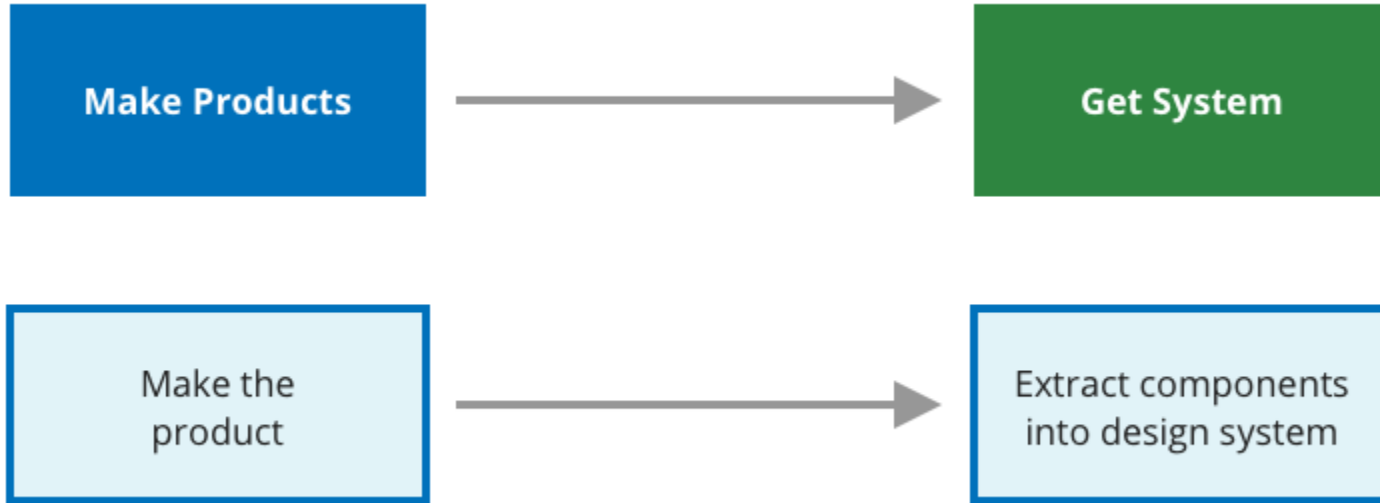
- Components & patterns
- Documentation site
- Usability, UI, & accessibility guidelines
- Sketch Library
- Governance
- Design tokens for colors, fonts, spacing and other customizable styles

## Theme

- Theme variables
- Themed Sketch Library
- Site specific assets
- Specific guidelines

# Changing our process for incorporating components into the design system

# Our updated process



# Our updated process

## **Product teams build components in their product**

Nothing new here, let teams do what they're good at and love doing. Build products!

## **Extract components into the design system**

Design system team takes components with real world use cases and moves them into the core design system with possible modifications.



**How are we getting there?**



# Retiring old sites



InVision DSM



Styleguide.healthcare.gov



Assets.cms.gov

Retire legacy documentation and guidance sites and start using **design.cms.gov** for everyone that is using the design system no matter what product they are building for CMS.

# The landscape



Sketch Kit - Mgov



GitHub



Invision DSM - HCgov



Invision DSM



Design.cms.gov



HCgov doc site



brand guidelines PDF



GitHub - HCgov



Assets.cms.gov



Sketch Kit



styleguide.healthcare.gov



Invision DSM - Mgov



GitHub - Mgov

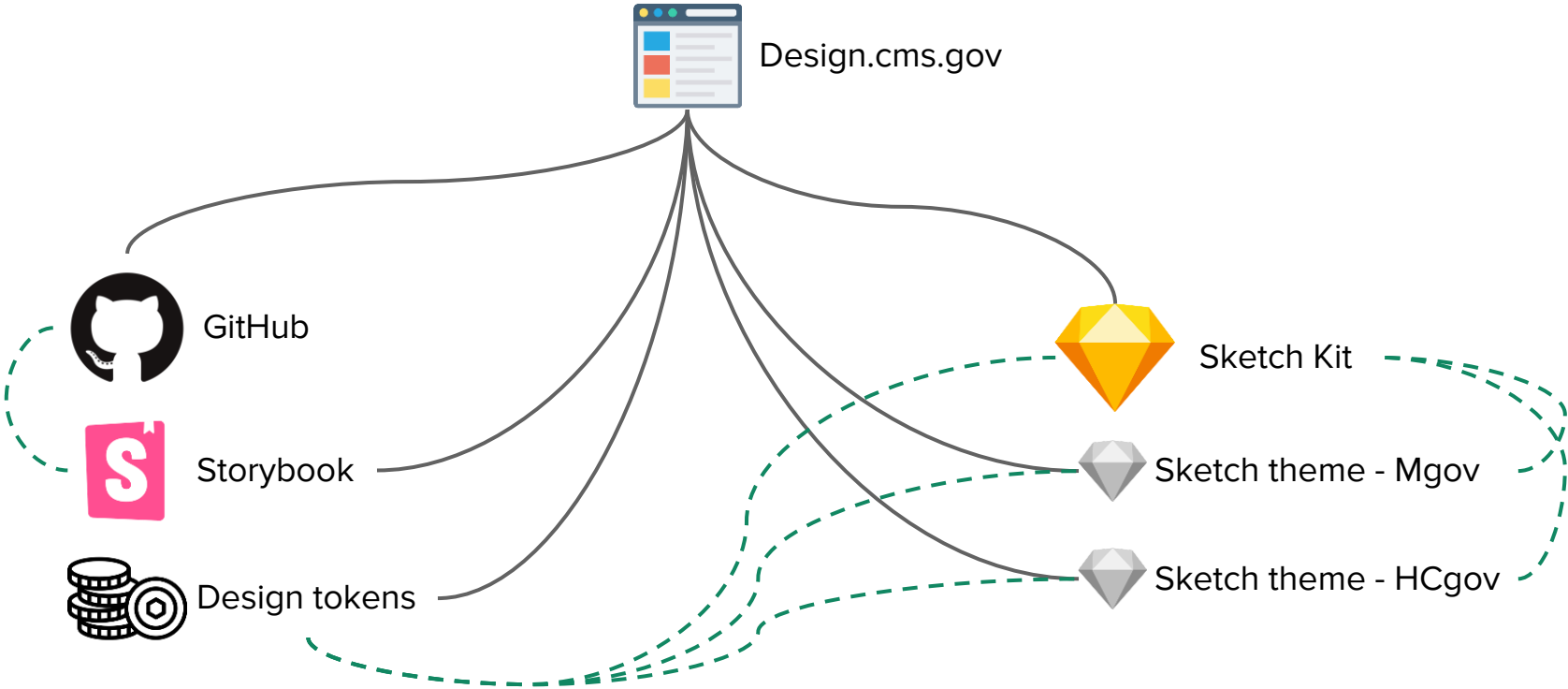


Sketch Kit - HCgov

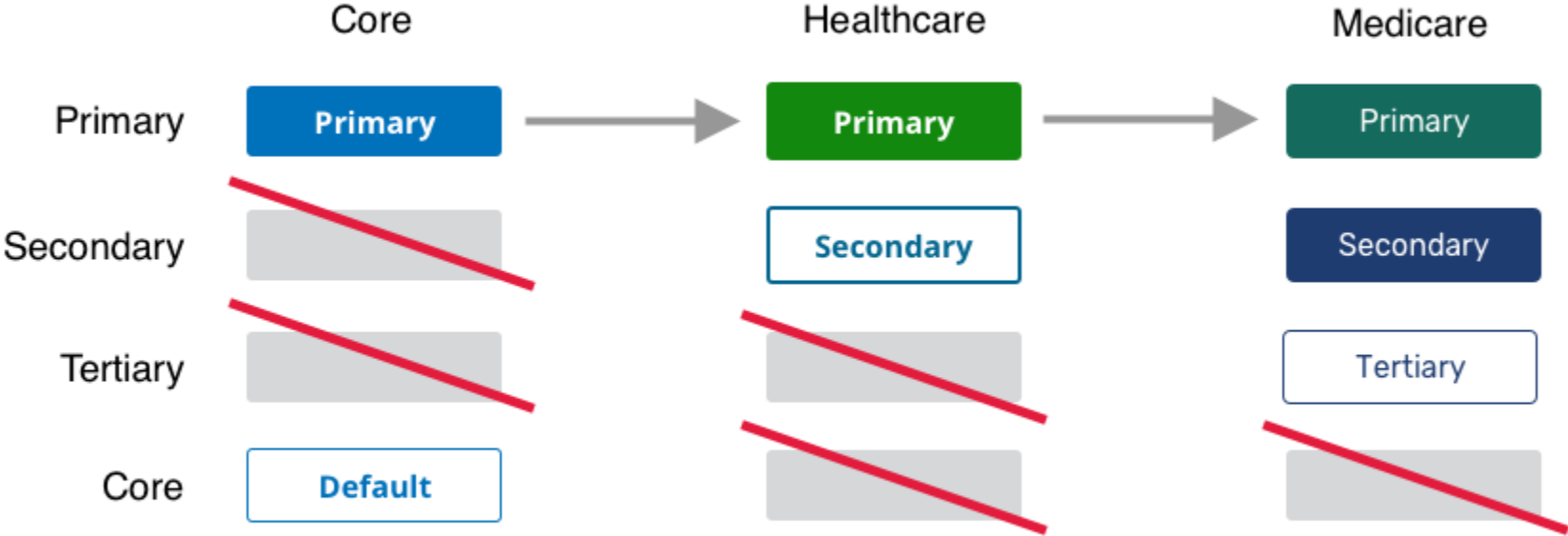


Mgov doc site

# The landscape

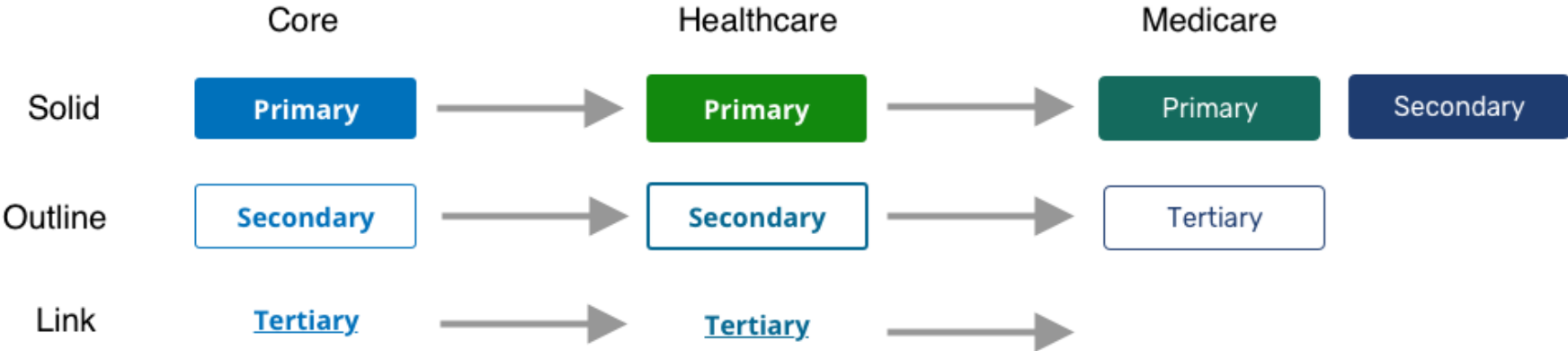


# Systematizing button architecture



# Systematizing button architecture

\*Releasing button architecture updates (Mid-Summer) 😊



# Future button variations are aligned

Making sure we have a systematic approach that supports consistent usage across all of CMS reinforces a united approach to design systems that supports the needs of both Medicare and Healthcare.

- **Styles** are comparable.
- **Naming** is consistent.
- **Guidance** is both general and theme specific.

# Improving the documentation experience

- Theme switching between (Core, Healthcare, Medicare, etc...)
- Showing general and theme level guidance in context
- One place for designers, developers to go
- Live code examples you can modify with Storybook
- Links to relevant Sketch Symbols in sketch Cloud

## CMS Design System

### Select theme

Core

Get started ^

Developers ^

Installation

Using Sass/CSS

Code conventions

Using components

Internationalization

Theming

Examples

Designers

Contributing

# Alert

Alerts keep users informed of important and sometimes time-sensitive changes.

Draft



[Github](#)



[Storybook](#)



[Sketch](#)

## Types

### Standard alert

Type something



#### This is a simple heading

Lorem ipsum dolor sit [link text](#), consectetur adipiscing elit, sed do eiusmod. Alerts can have children, or they can be left off and used with just a heading prop.



[React code](#)



[View in Storybook](#)

### On this page

[Types](#)

[- Standard alert](#)

[- Lightweight alert](#)

[Configurations](#)

[- With heading and](#)

[content](#)

[- With heading](#)

[- With content](#)

[- No icon](#)

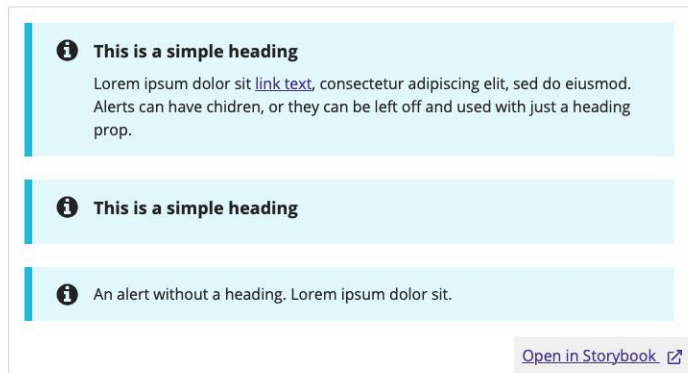
[- Status](#)



# Interactive examples with Storybook

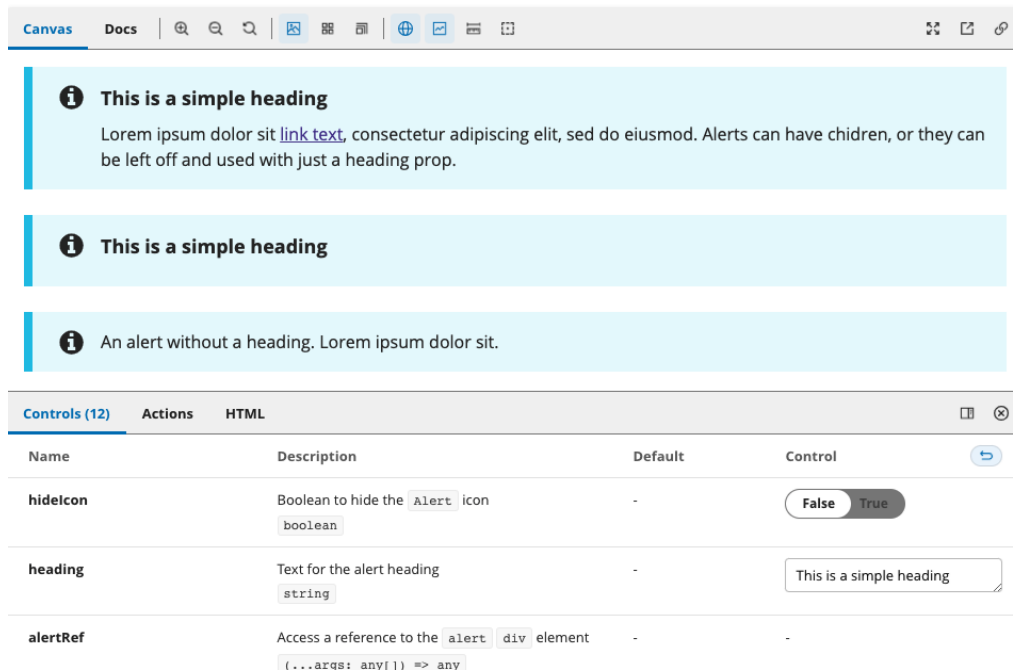
## Default

[View Source File](#)



The screenshot shows a Storybook component preview for an alert. It contains three light blue alert boxes, each with a dark blue vertical bar on the left side. The first alert has an information icon and the heading "This is a simple heading", followed by a paragraph of Lorem ipsum text. The second alert has the same heading. The third alert has an information icon and the text "An alert without a heading. Lorem ipsum dolor sit." Below the preview is a button labeled "Open in Storybook" with an external link icon.

> Code snippet



The screenshot shows a Storybook component preview with a controls panel. The preview area contains three light blue alert boxes. The first two have the heading "This is a simple heading" and the third has the text "An alert without a heading. Lorem ipsum dolor sit." The controls panel is open, showing a table of properties for the component.

Name	Description	Default	Control
hideIcon	Boolean to hide the <code>Alert</code> icon <code>boolean</code>	-	<input type="checkbox"/> False <input checked="" type="checkbox"/> True
heading	Text for the alert heading <code>string</code>	-	<input type="text" value="This is a simple heading"/>
alertRef	Access a reference to the <code>alert</code> <code>div</code> element <code>(...args: any[]) =&gt; any</code>	-	-

# Using design tokens as our foundation

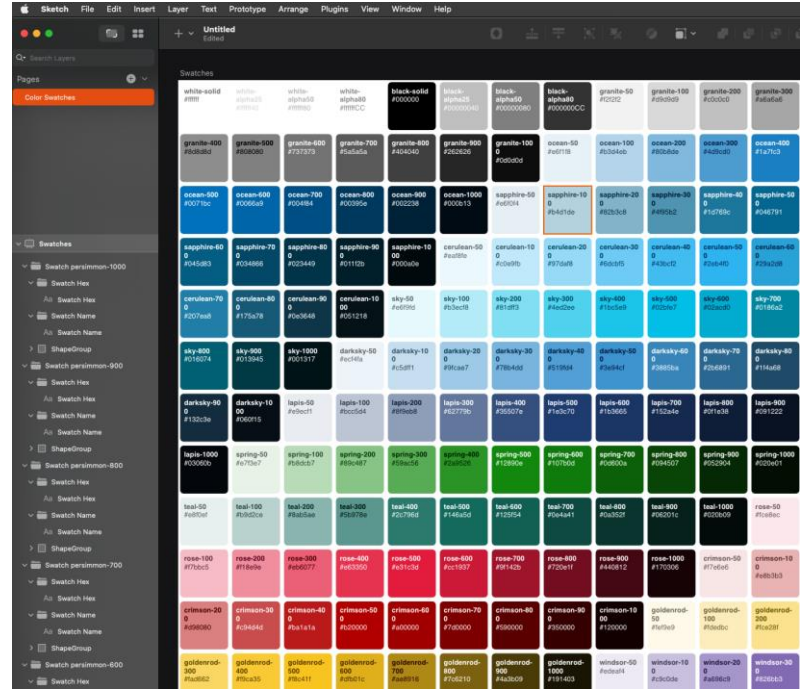
Standardizing, implementing and tool-building for a future with design tokens sets up the CMSDS and any other consumer of our design language with a set of standards that will improve design consistency and workflow organization using nearly any tool-stack available now or for years to come.

Key	Value
“lightblue-500”	“#0000AA”
“spacer-2”	“16px”
“font-size-xl”	“2rem”

# Design tokens and Sketch

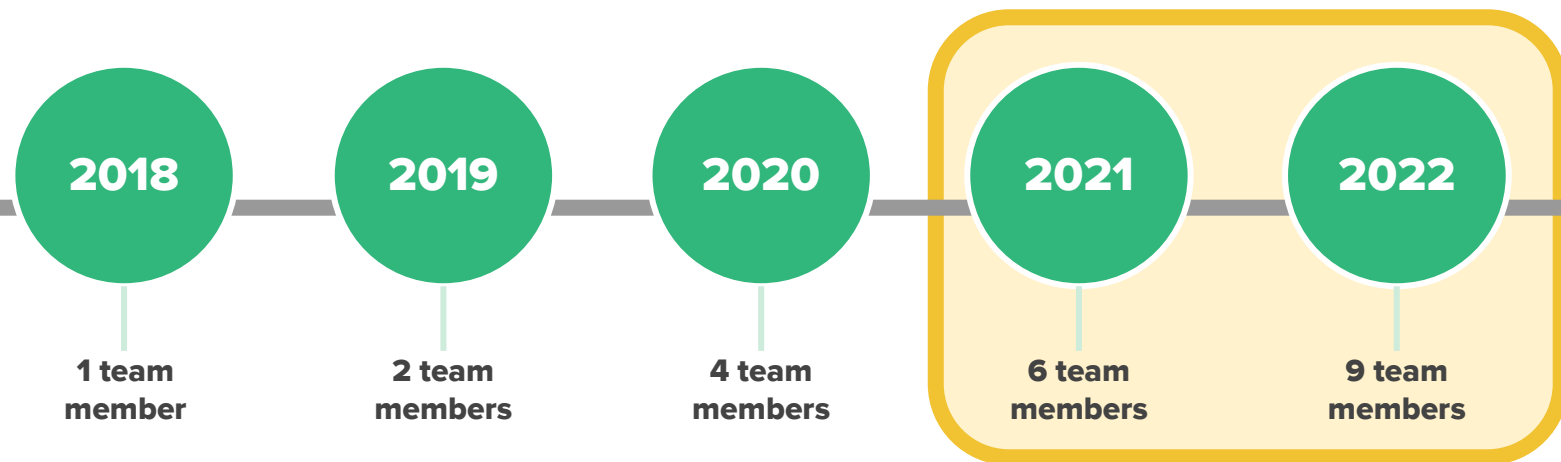
Develop tooling to offer improved workflows:

- Checking for spacing accuracy
- Color correction / suggestion
- Up-to-date versions of design system components represented as sketch objects exported with DS releases.
- Theme switching / variation validation



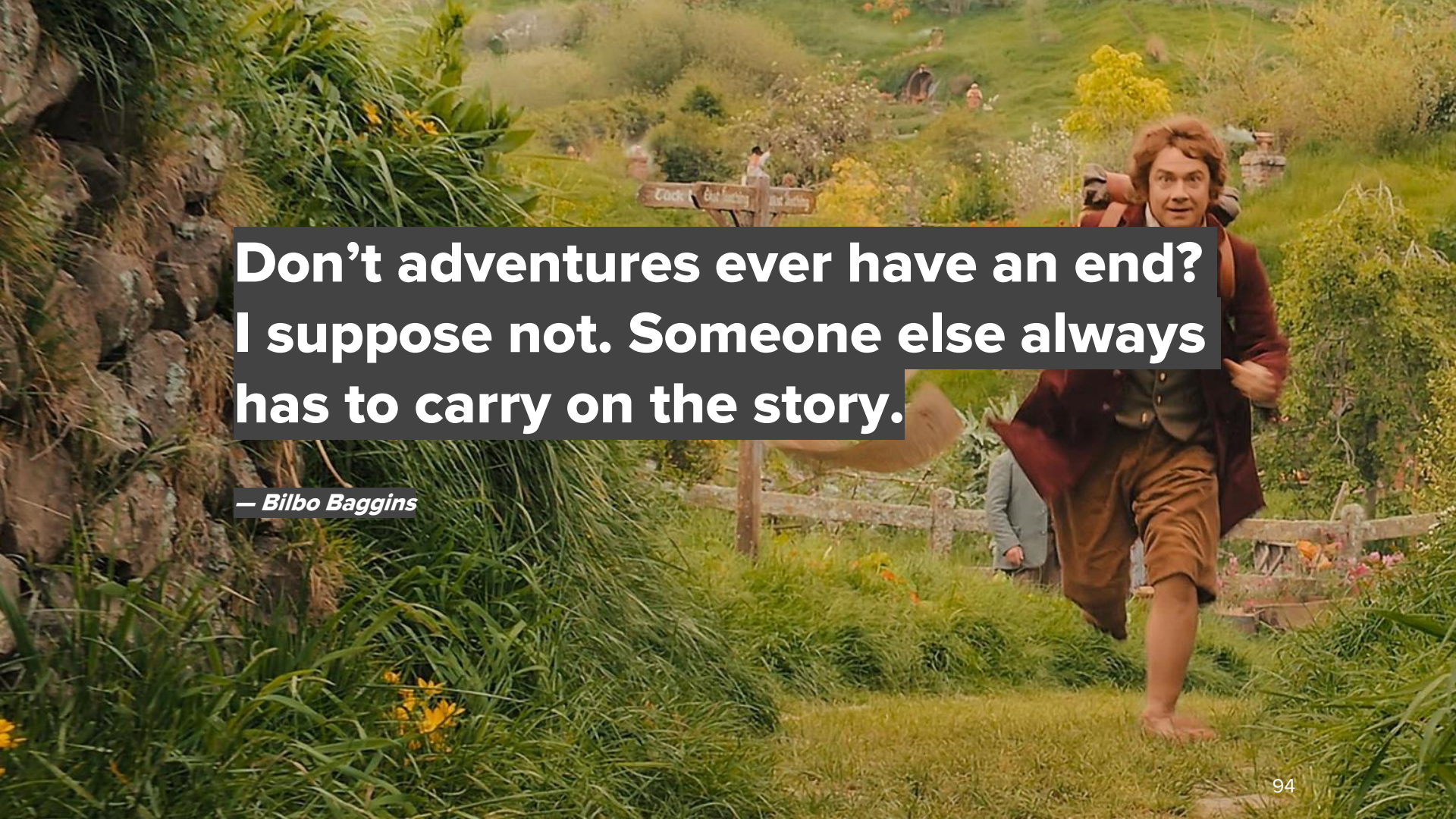
# Staffing the design system

A design system is an evergreen product that is always changing and never done. We will continue to advocate for a cross discipline staffed design system team to ensure the success of the design system into the future.



# Some ways we can measure success

- Users, no matter what discipline (design, development, product, etc...), can all find complete & thorough guidelines on [design.cms.gov](https://design.cms.gov).
- Legacy documentation sites have been deprecated.
- CMS experiences become more consistent in regards to color, spacing, type, icons and other visual elements.
- Design and code feedback lessens for design system components used within products.
- Content updates are easier to make and discoverable by product teams.
- Design patterns are documented easier and more quickly.

A scene from 'The Hobbit' showing Bilbo Baggins running through a lush, green landscape. He is wearing a brown coat and shorts, and has a backpack on his back. In the background, there are rolling hills, a wooden signpost with directions to 'Cock', 'Bat', 'Fishing', and 'The South', and a stone archway. The scene is filled with vibrant green grass and various plants.

**Don't adventures ever have an end?  
I suppose not. Someone else always  
has to carry on the story.**

*— Bilbo Baggins*

**Thank you!**

# Resources

- **The CMS Design System documentation site**  
<https://design.cms.gov>
- **Check out the GitHub for issues and discussions**  
[Design System GitHub](#)
- **Find us in CMS Slack**  
[#cms-design-system](#)