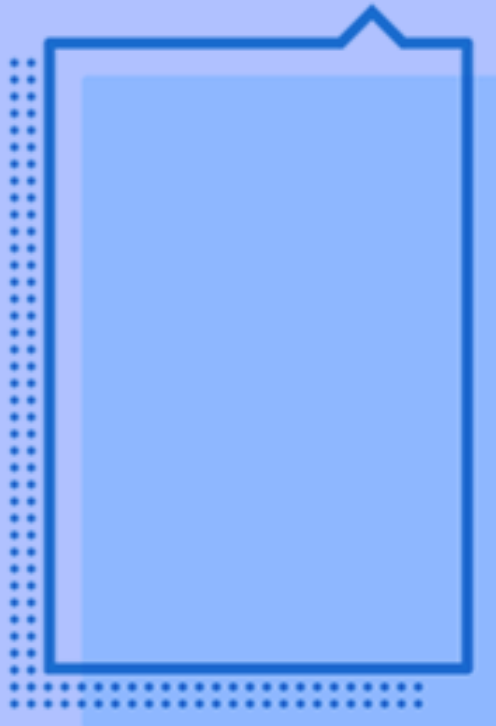
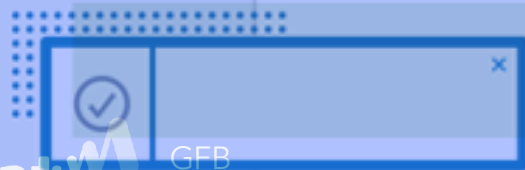




ART/DECO

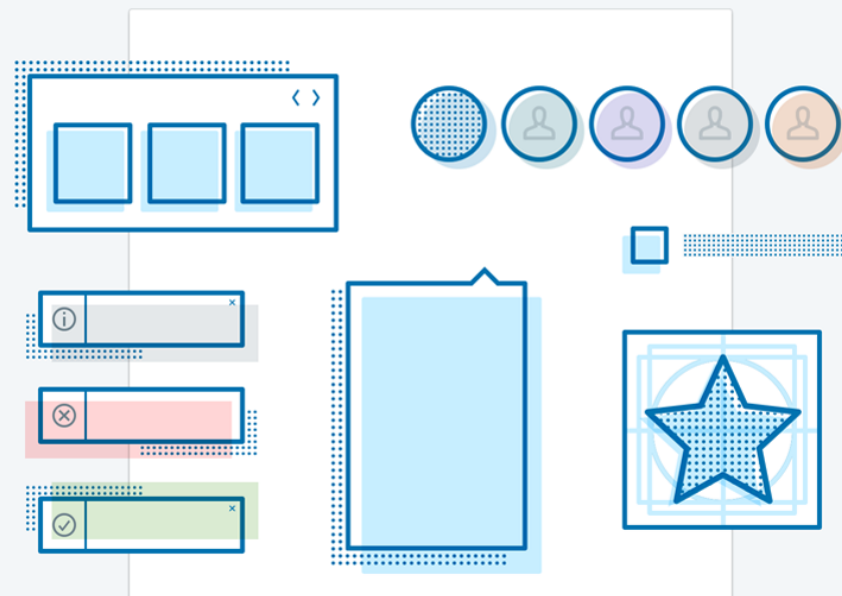


LinkedIn



In summer of 2019, I was invited by LinkedIn to join the Design Systems team and contribute to the Art Deco initiative.

ART/DECO

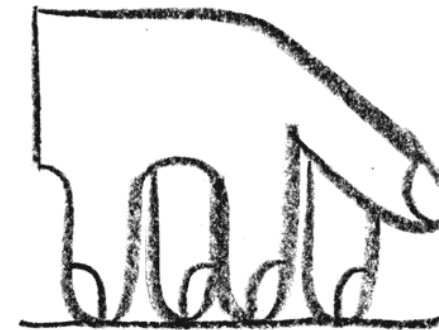


My first week I hit the ground running and needed to understand the current team and design system currently in place.



First... I shadowed my manager to planning meetings:

- Gathered feedback around current workflows & process
- Learned about the history of previous design systems projects



Who is our audience...

- Both Designers and Devs?
- Only the design team?
- Design systems team?





Polaris



Lightning Design System

Research: How are other design teams guidelines documented?



Carbon

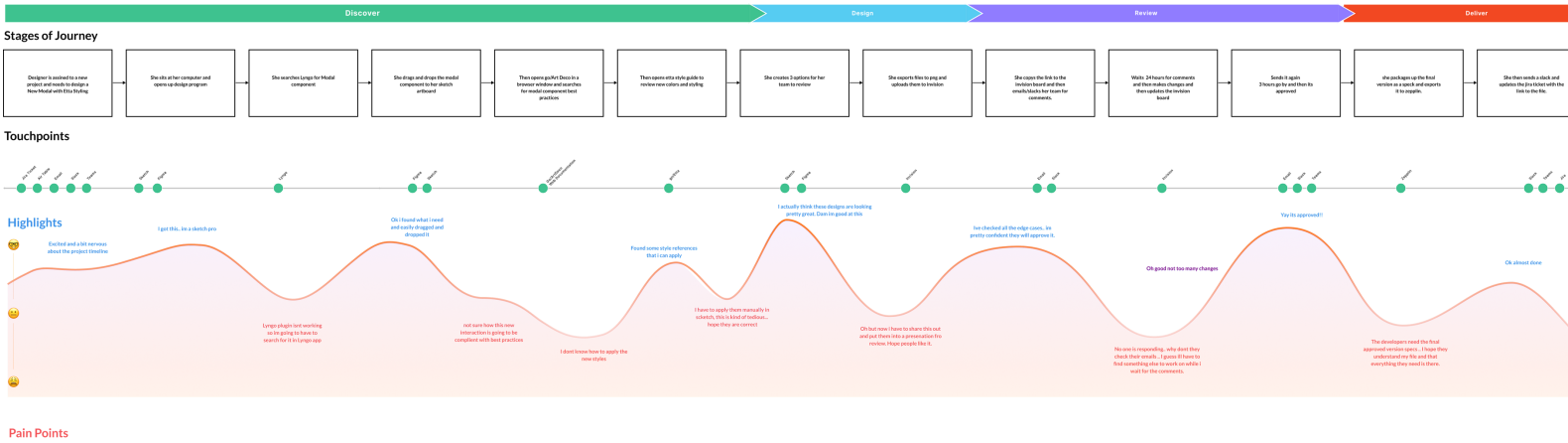


CLARITY™



Atlassian.design

I interviewed **10** product designers and managers to learn more about the designs team process and mapped out their journey.



is feels hard
don't have time

This doesn't feel
flexible... or
allows much Creativity

Too Technical

Too Vague

Designers felt confused because it wasn't written for them.

Where do I

How Current are
these guidelines?

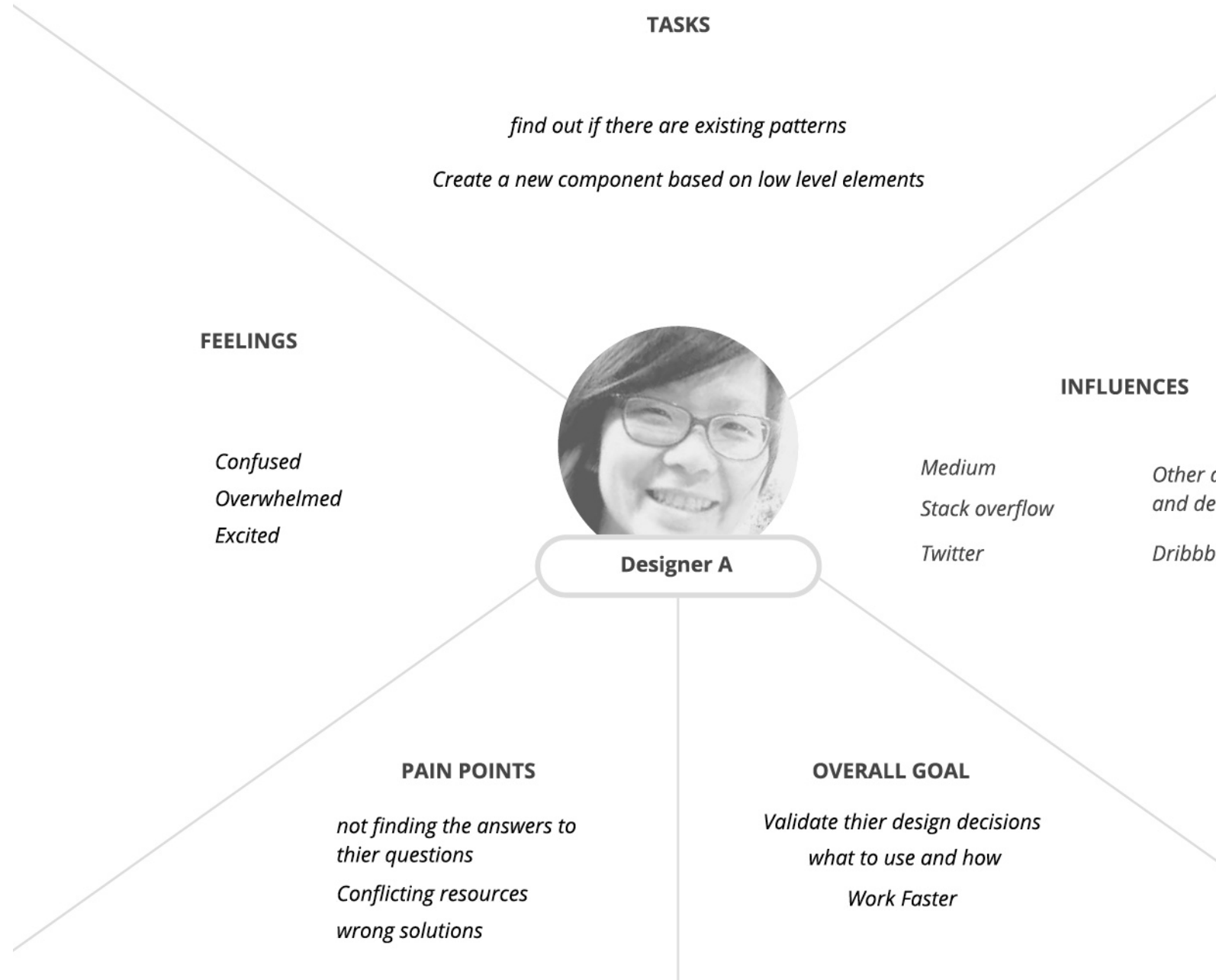
How Can I apply
this to my work

Designer A:

embedded on the LMS team uses sketch and lyngo to iterate on existing patterns
Only has visibility to her current vertical

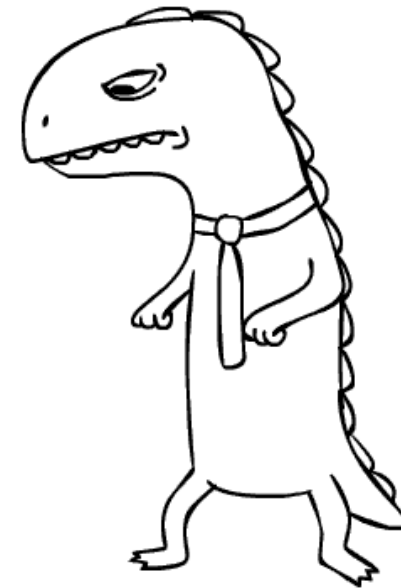
Has tool fatigue from previous initiatives

Works on-site and sometimes remote



Pain Points:

- Communication is difficult between cross-functional teams
- Feels cut off from other designers
- Not sure if assets are up to date or current
- Confusion around component guidelines and documentation in context to her projects
- Lyngo is increasingly buggy and not well organized.



What we discovered: designers had **struggled with context** and was looking for specific guidance around how and when to use a specific styling, component variation, or current **status** in the product.

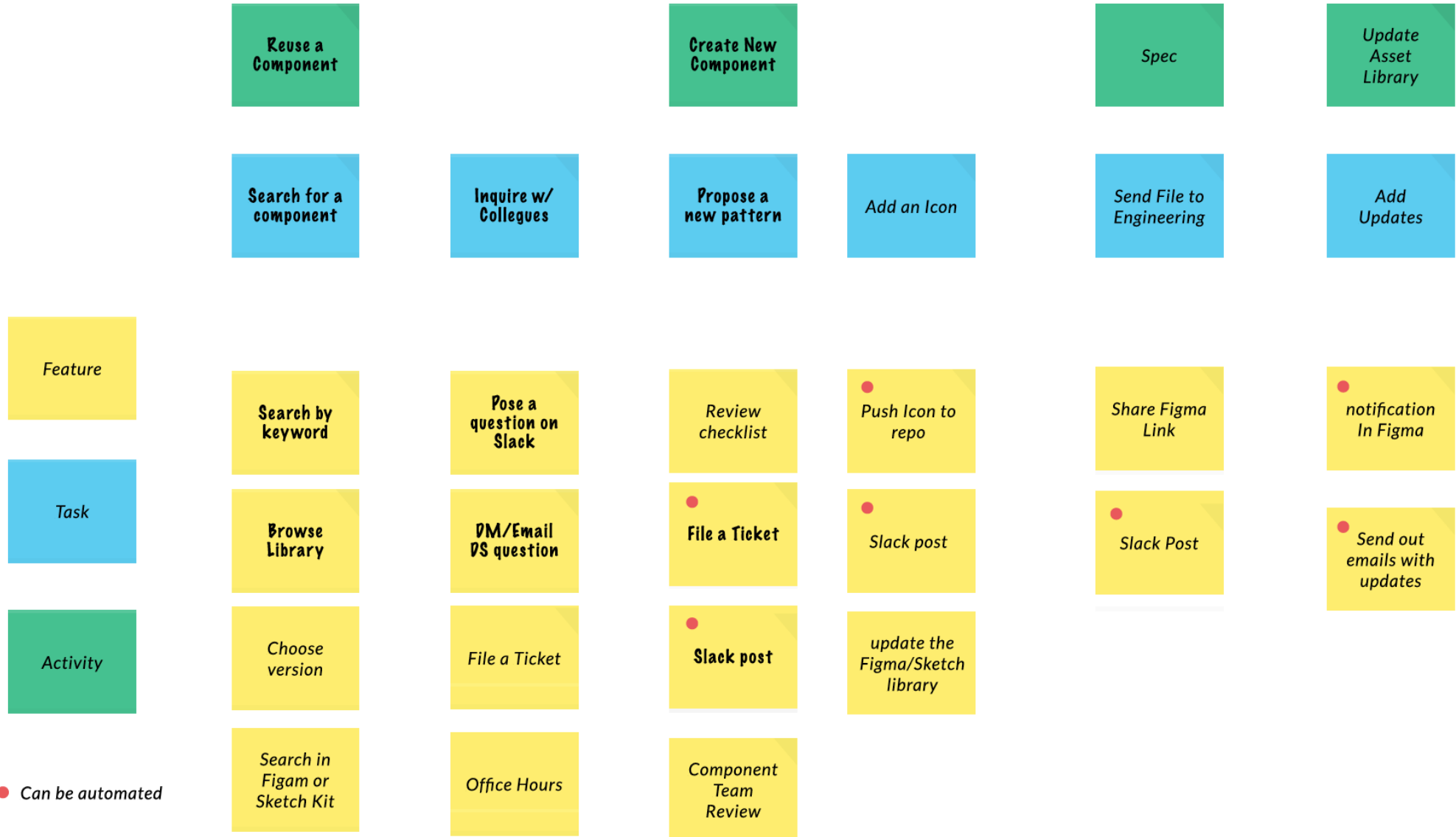
Our goal: provide a centralized place for guidance, on when and how to use components in their design work in a consistent way.



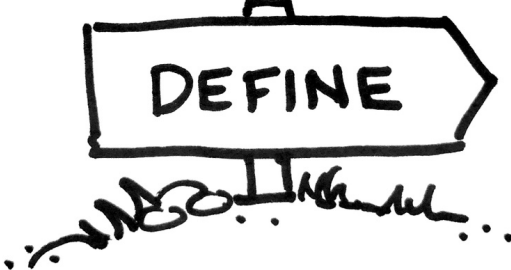
Product Designer Story Map

Persona: UED Designer

Proposed workflow



What are our Goals? How do we measure progress and success?



Planning:

Outlining the goals and building a charter



Project Brief

Business Goals, Funding, Politics, culture, technology, resources, and constraints.

What are the goals?

Fast learning and quick lookup of design resources

What tactics are used to reach these goals?

A Documentation website for a designer handbook

What are the Objectives of this project?

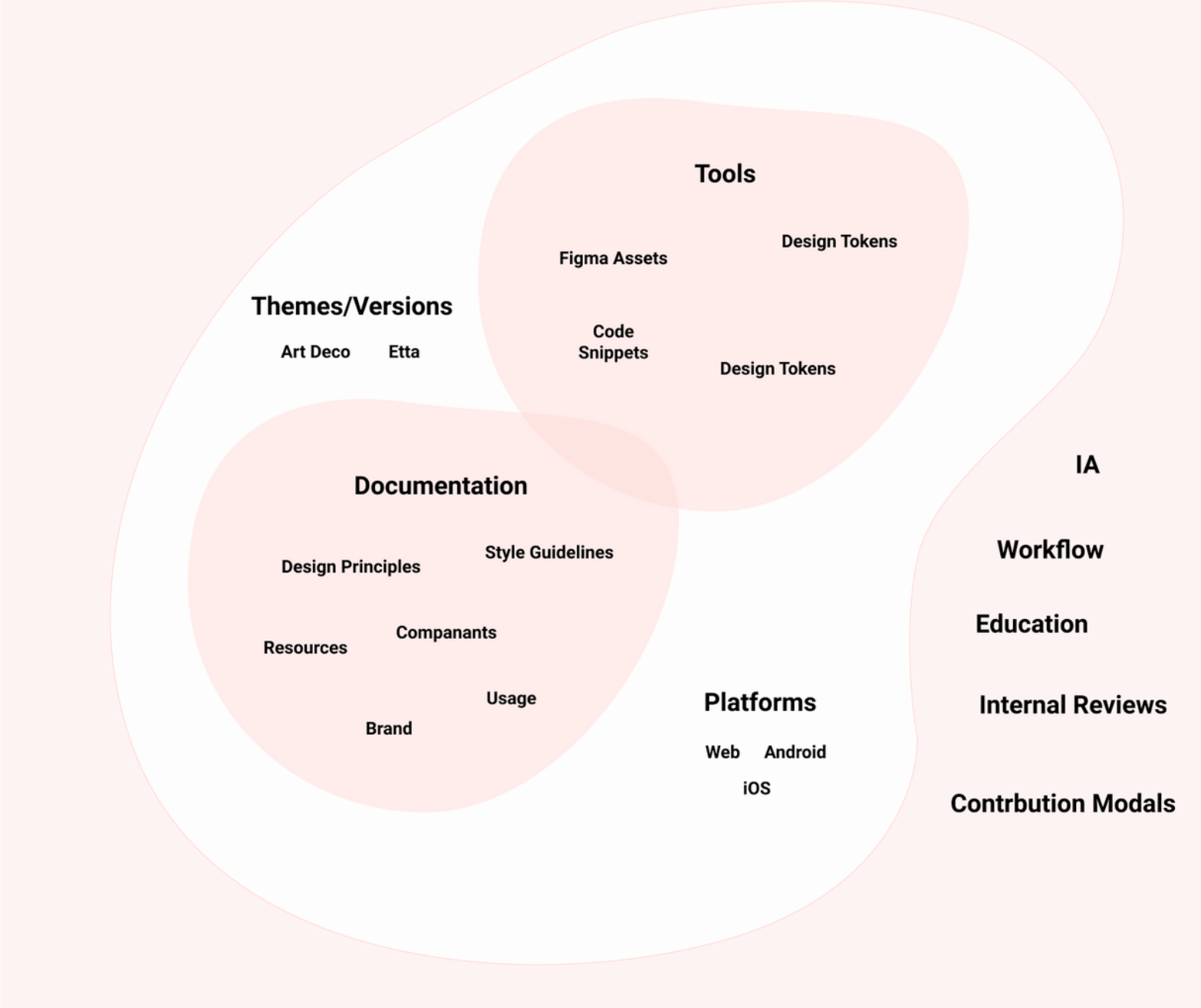
Improve continuity on the design teams

Who will be using this documentation? What are their expectations? What are their pain points and expectations:

- Self service w/ component library, design assets, and usage guidelines.
- Guidance on how to validate their design choices, and the ways they can improve their workflow process
- Syncing design communication and continuity across vertical teams with terminology, versions, and UI a

Workflow, Processes and Standards:

All the things to consider:



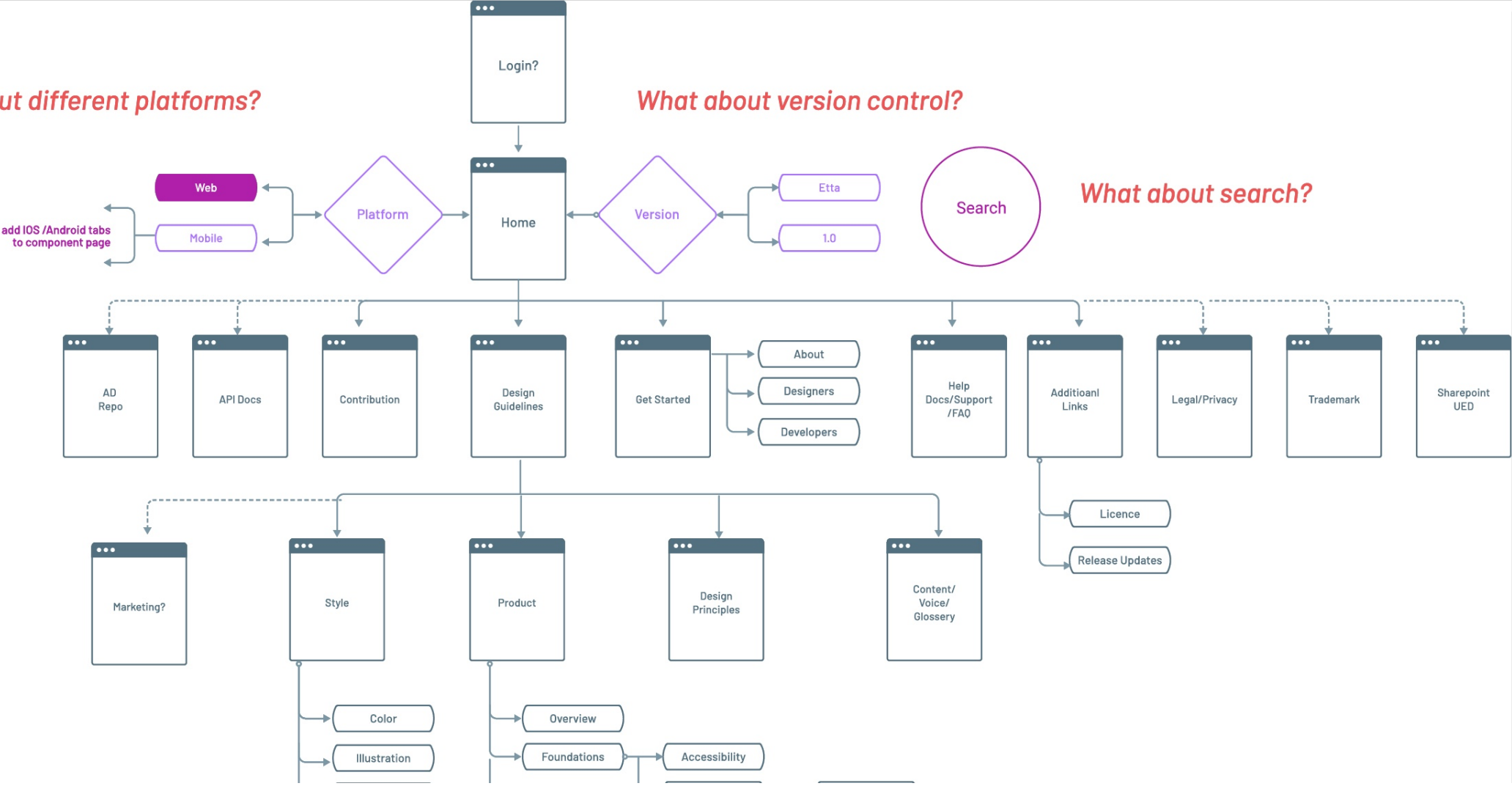
The Documentation Site

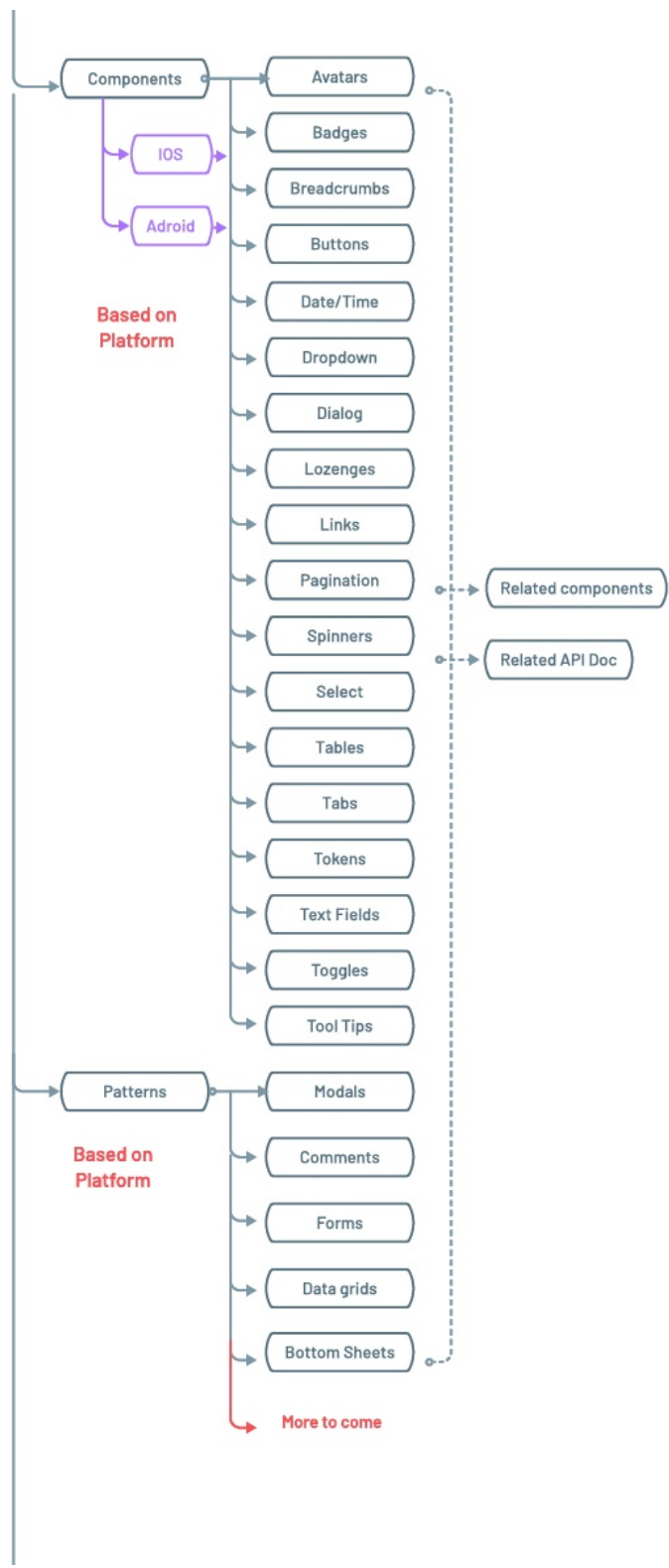


What about different platforms?

What about version control?

What about search?





Are there more natural groupings for these?

What if we organized the components by action? For example we can group buttons and radios into 'controls'.

Content Layout



- Introduction
- Types
- Best Practices
- Style
- Behavior
- Variations
- Accessibility
- Internationalization
- Implementation &

Get Started

Using Components

Status & Updates

Actions

Button

Button Group

Form Controls

Checkbox

Dropdown Selector

Radio Buttons

Text Input

General Guidance

Types

Best Practices

Validation and errors

Styleguide created with

Form Controls

Text Input

Text inputs allow users to enter and select text, providing helpful usability extensions like error indicators and password masking.

Overview

Web

IOS

Android

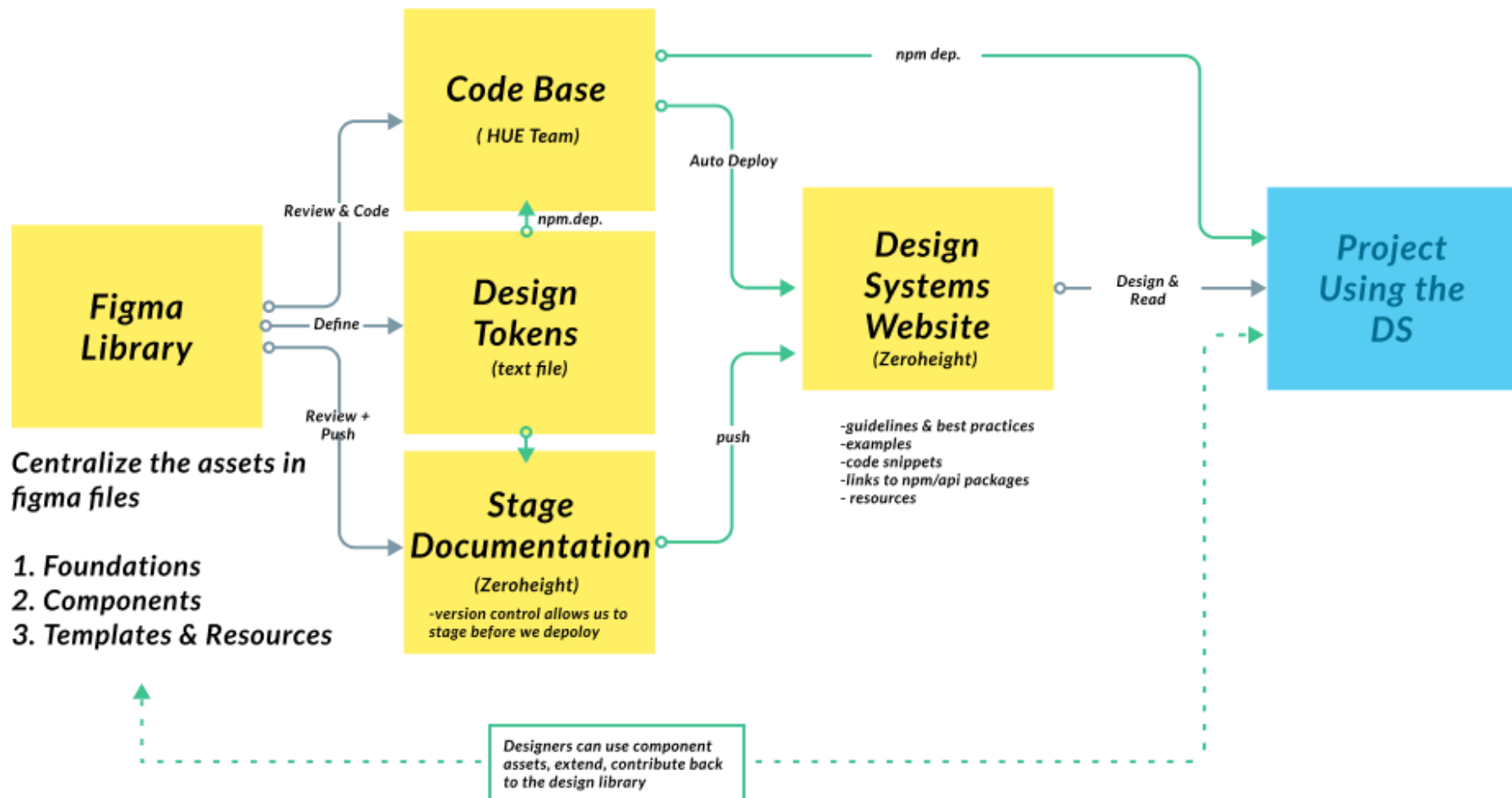
General Guidance

Text inputs should:

- be labeled with clear language that is easily understood by our members so know how to put into the input.
- All fields are required unless marked as "Optional"

Workflow Proposal

- Manual Integration
- Automated Integration



Documenting Content Proposal

- Start Date: 2019-07-01
- RFC's RB:
- Jira Issue:

Note: Please reach out to [Lisamarie Kelly](#) for access, questions and additional information.

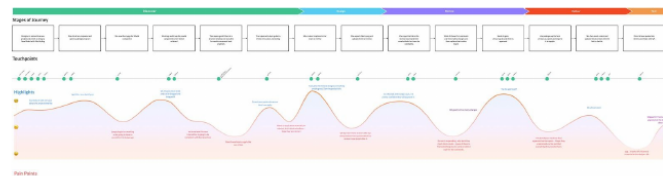
Summary

The goal of the documentation site is to provide clear and concise information for our UED product designers to search, read and take action on. This is a guide on how to build the documentation site by following a content template.

Motivation

Why are we doing this? What use cases does it support? What is the expected outcome?

UED designers were having difficulties navigating and interpreting existing art deco guidance and wanted something more in-context to their needs and workflow. We started this project by exploring the pain points and use cases to better understand our audiences needs. Please take a min to review the user journey and story maps [available here](#).



This doc is for the design systems team members who are looking for direction on how to use this template and how to architect and create design guidance around a component.





Button

Nonwrapping Title

Card title
Subtitle Text
Meta-data

Card title
Subtitle Text
Meta-data

Looker

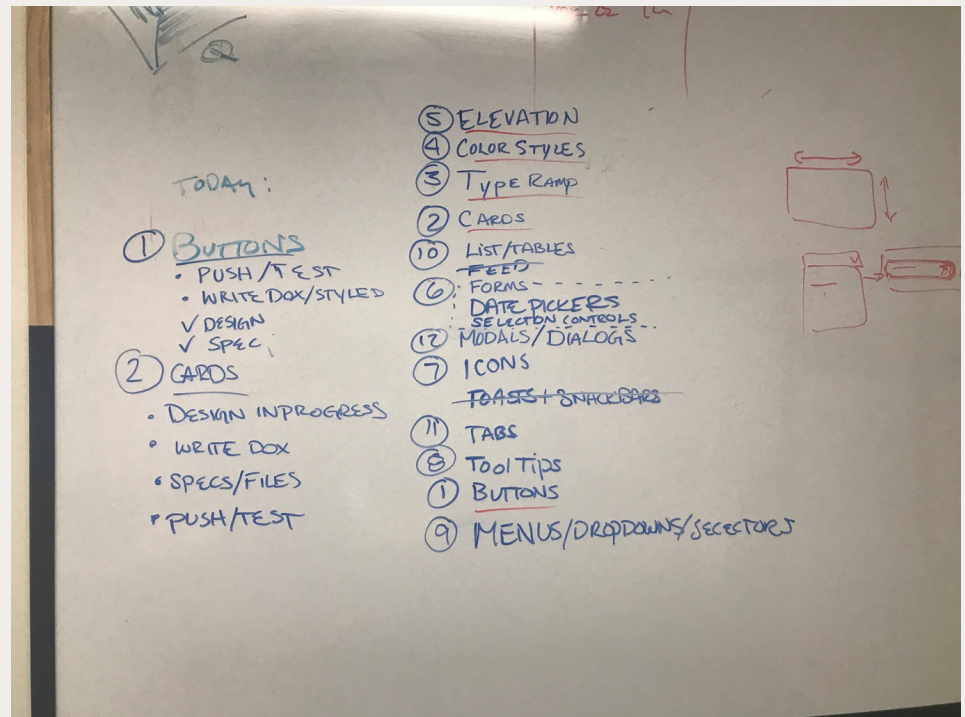
Total Spend by Color

User Base Demographic Profile



Where to start?

Which components are consider low level vs high level



Component Library Foodchain



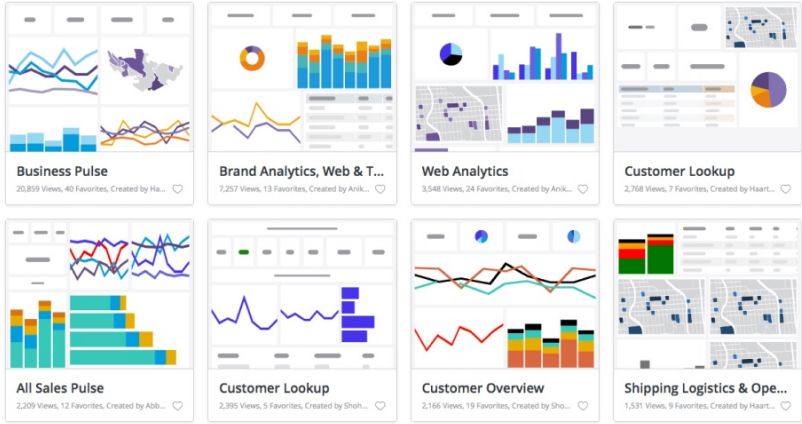
Planning Milestones

Level/Struktur	Strategie und Visionen	Business Case	Strategische Ziele	Maßnahmen für Strategien	Wichtige Funktionen	Technische Spez.	Prozesse und Organisation	IT/Infra	Finanz- & Personalplanung	Risikoprüfung	Rechtliche Aspekte	Marketing & Vertrieb	Produktion & Logistik	Service & Support	Wartung & Erneuerung	Umfeld & Partner	Umfeld	Strategische Projekte	Meilensteine	Zeitplan/Meilensteine	
Strategie	Strategie und Visionen	Business Case	Strategische Ziele	Maßnahmen für Strategien	Wichtige Funktionen	Technische Spez.	Prozesse und Organisation	IT/Infra	Finanz- & Personalplanung	Risikoprüfung	Rechtliche Aspekte	Marketing & Vertrieb	Produktion & Logistik	Service & Support	Wartung & Erneuerung	Umfeld & Partner	Umfeld	Strategische Projekte	Meilensteine	Zeitplan/Meilensteine	
Struktur																					
Prozesse und Organisation																					
Maßnahmen	Maßnahmen für Strategien	Business Case	Strategische Ziele	Maßnahmen für Strategien	Wichtige Funktionen	Technische Spez.	Prozesse und Organisation	IT/Infra	Finanz- & Personalplanung	Risikoprüfung	Rechtliche Aspekte	Marketing & Vertrieb	Produktion & Logistik	Service & Support	Wartung & Erneuerung	Umfeld & Partner	Umfeld	Strategische Projekte	Meilensteine	Zeitplan/Meilensteine	
IT/Infra																					
Finanz- & Personalplanung																					
Risikoprüfung																					
Rechtliche Aspekte																					
Marketing & Vertrieb																					
Produktion & Logistik																					
Service & Support																					
Wartung & Erneuerung																					
Umfeld & Partner																					
Umfeld																					
Strategische Projekte																					
Meilensteine																					
Zeitplan/Meilensteine																					

PDF

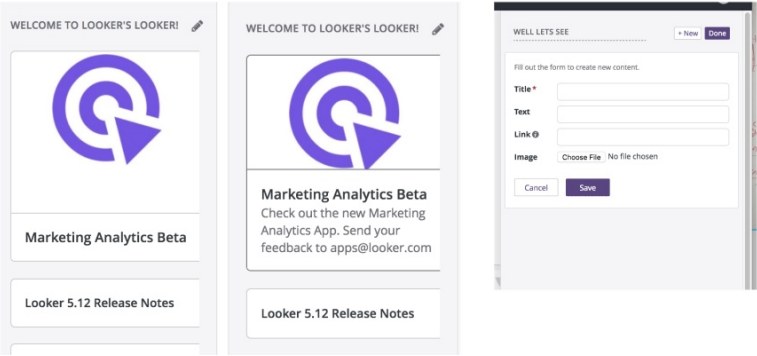
Audit Existing Card UI

SUGGESTED FOR YOU

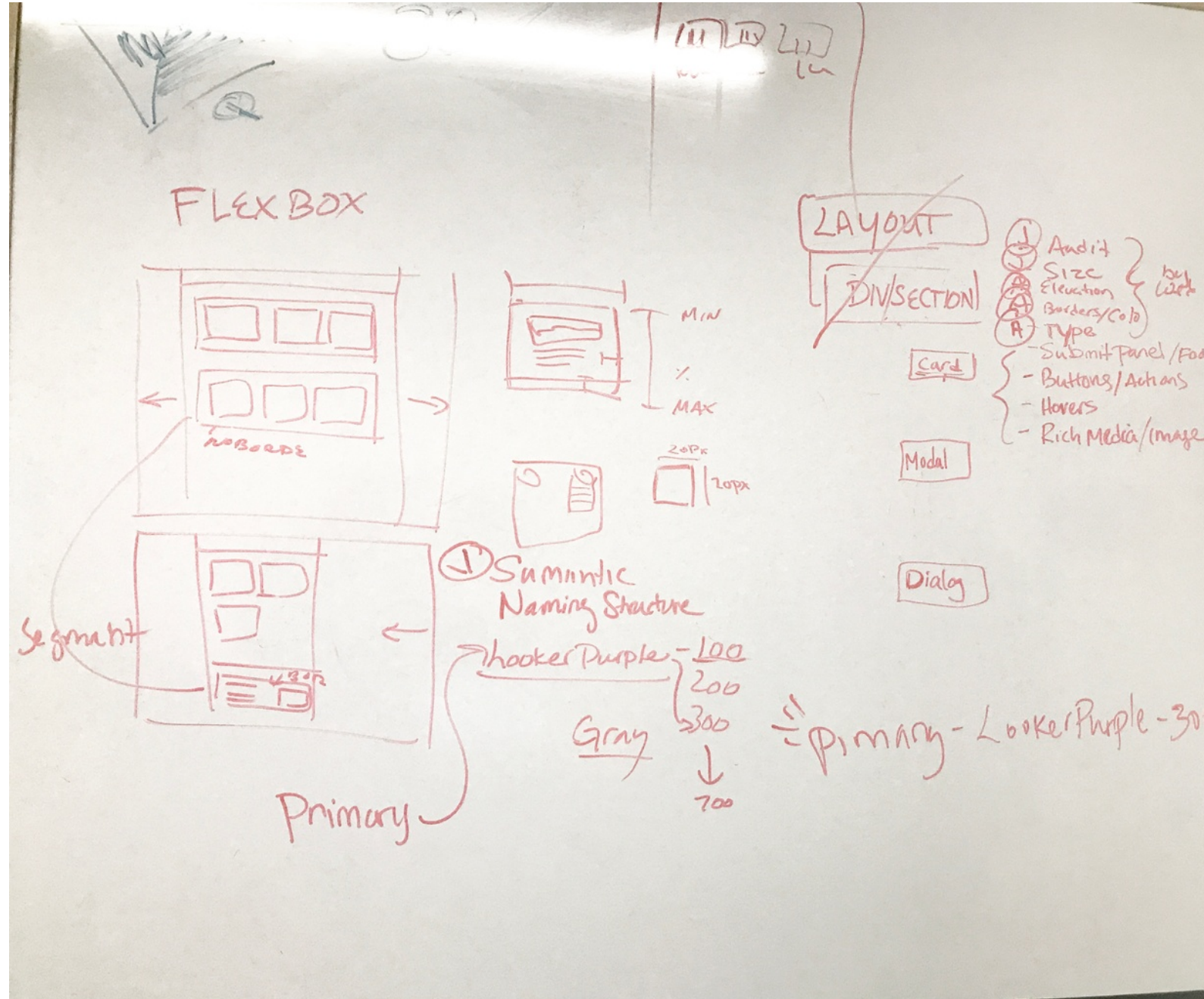


Sidebar cards, with hover to reveal details action

Edit card in sidebar



Naming Semantic Layers



Design & Specs

Cards: Container Specs

OVERVIEW

WHAT IS A CARD?

Cards provide scannable organized groups of content that is limited to one concept or thought per a card. These collections of content are a set and are made up of related content.

BEST PRACTICES

- Cards are bite sized previews of one focused concept.
- Contains: images, text, video, audio, list groups, actions & links.
- The entire card is clickable.
- There can be a header and a footer/submit panel.
- Cards have rounded corners and have elevation.
- Keep it simple and not crammed with content.
- Cards can be sorted and filtered.
- They should always sit on a background of light gray.

INSTALLATION
npm install @lens/card

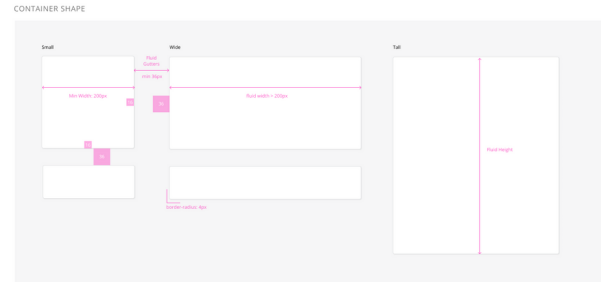
`card`

RELATED COMPONENTS

[Grid Layout](#)
[Typography](#)
[Elevation](#)

HTML STRUCTURE

```
<div class=".card">
  Simple
</div>
```



CONTENT BLOCKS

Cards are made up of blocks of content and are optional.

Rich Media: Photos, Video, or Audio.
16:9 or 1:1 square ratio

Titles
Left justified, Non-wrapping, & centered vertically

Subtitles
Left justified & Non-wrapping

Text Block
Left justified, wrapping, 12px font-size/18px line-height

Actions: Buttons, icon action, links, & controls.

Visual examples of content blocks: Rich Media (gray box), Title (bold text), Subtitle (smaller text), Text Block (wrapping text), and Action (button).

ACTIONS

- Keep actions limited to 1-2 items
- A card can have a CTA button or a link action but not both

Visual examples of action buttons: 'Action' (simple button), 'Action 1 Action 2' (button with two actions), and 'Small' (smaller button).

Cards: Styling

BASE CONTAINER

STACKED CONTAINER

COLOR BAR

MEDIA CARD

```
background: #ffffff;
border: 1px, solid, #E4E5E6
box-shadow: 1px 1px 2px gray-200
```

TYPOGRAPHY

Visual examples of typography: H2 - Card Title, H3 - Subtitle, H3 - Meta Data, and Nonwrapping Title. Each example shows the text style and its placement within a card layout.

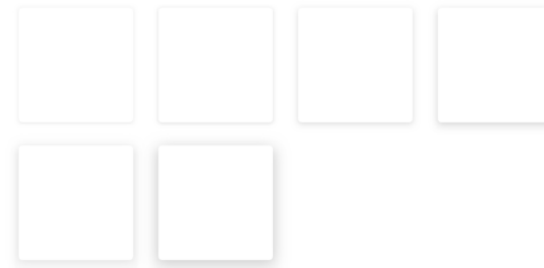
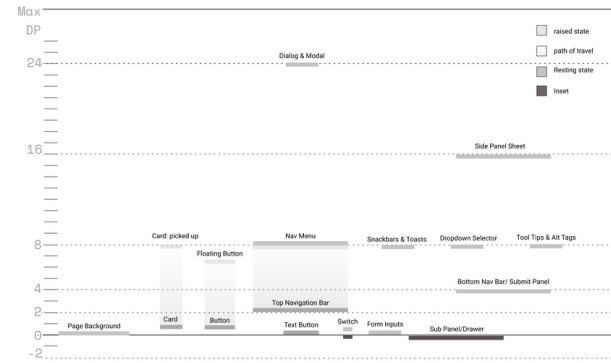
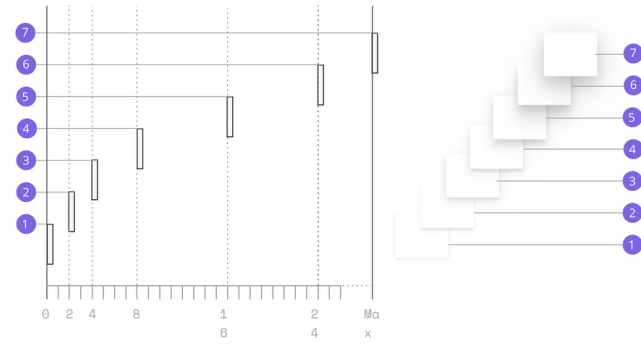
H2 - Card Title	H3 - Subtitle	H3 - Meta Data	Body	Tag
font: Open Sans font-size: 16px font-weight: semi-bold 600 line-height: 24px color: #3c4345 font-weight: 600 text-overflow: ellipsis	font: Open Sans font-size: 12px font-weight: semi-bold 600 font-size: 12px line-height: 18px color: text-overflow: ellipsis	font: Open Sans font-size: 10px line-height: 24px color: #797672 text-overflow: ellipsis	font: Open Sans font-size: 12px line-height: 18px color: #3c4345	font: O font-si line-he color: #Eff on text-ov

FLUID MODULAR SCALE HEADERS
Example: <https://codepen.io/MadeByMike/pen/VvwqW>

- A quick read and not over descriptive
- Scannable with clear simple language
- Single sentence / no word wrapping
- Use an ellipsis when
- Title Casing Only / not All Caps

Visual examples of fluid modular scale headers: Nonwrapping Title (54), Nonwrapping Title (58), Nonwrapping Title (64), and Nonwrapping Title (72). Each example shows the text size and its placement within a card layout.


Elevation



Comms: Outreach w/ Partners

A Developer wants to search/find an existing first level component and iterate or change it slightly to form an extended subclass component.



I want to use this card pattern somewhere else in the product but change it a bit. Where do I start to reuse that pattern?

Enable

DigitalOcean- Stop Droplet

Stop a DigitalOcean droplet.

Works with

- Fields:  `digitalocean_droplet_id`
- Queries:  `digitalocean_droplet_id`

Do you Search for it?

Pop open developer tools and check out the markup and styles?

The screenshot displays a grid of dashboard cards on a web page. The cards include:

- Looker Pulse**: 43,519 Views, 128 Favorites, Created by Andrew Sea...
- Customer Lookup**: 28,627 Views, 108 Favorites, Created by Ryan Lee
- New EMEA Dashboard**: 2,634 Views, 4 Favorites, Created by Alex Mehrtens
- Chatter Dashboard v2**: 2,001 Views, 12 Favorites, Created by Brecht Vermeil...

At the bottom, the browser's developer tools are open, showing the HTML markup for a card. The selected element is:

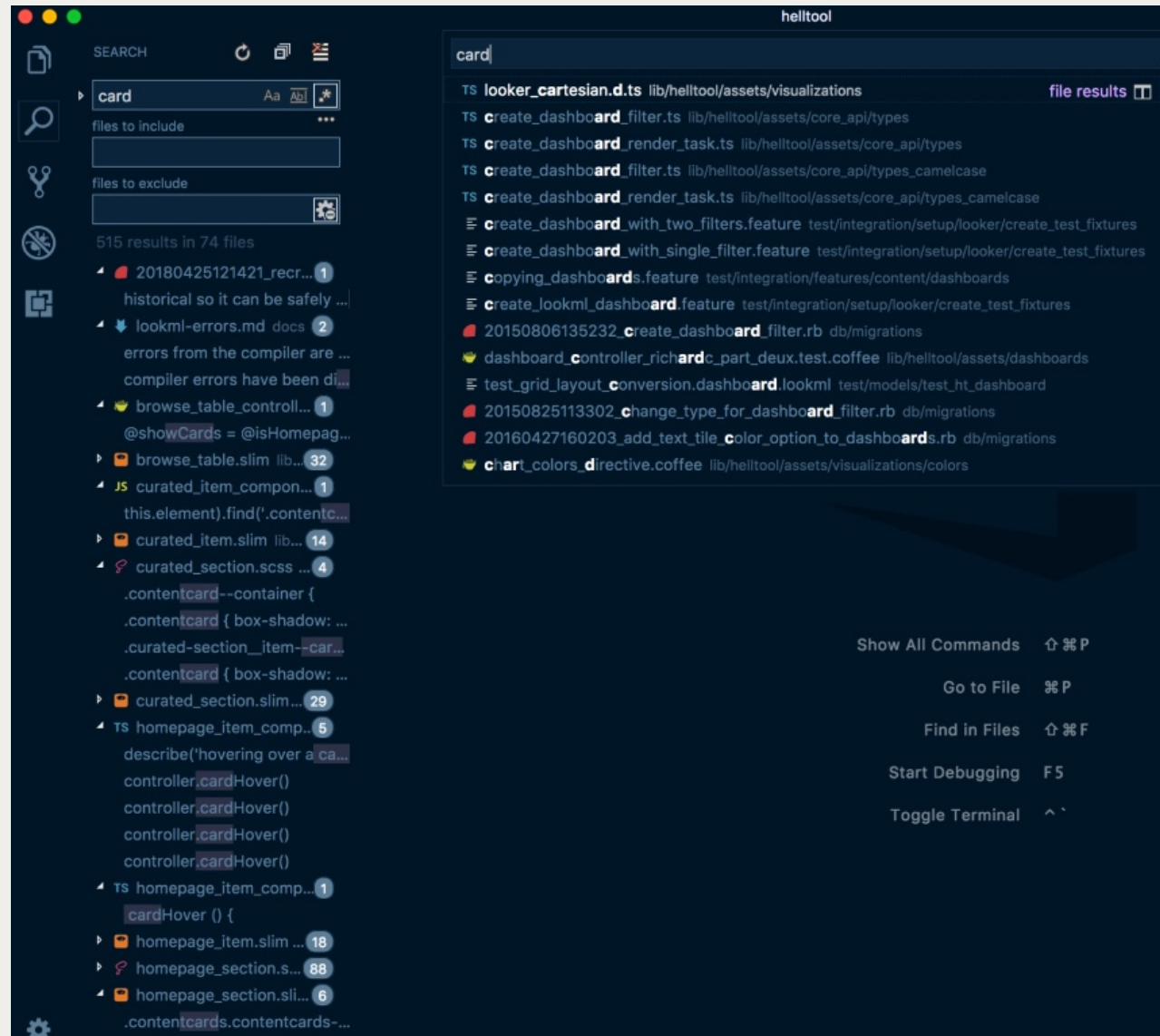
```
<div class="contentcard--container ng-scope contentcard--suggested" ng-class="{true: 'contentcard--container_small contentcard--favorites', false: 'contentcard--suggested'}|section,title = 'Your Favorites'| ng-repeat: item in items | orderBy: ['order'] >= 50 <!-- ngRepeat: item in items | orderBy: ['order'] --> <!-- ngIf: ctrl.itemImageUrl --> <div class="contentcard--hoverarea" ng-mouseover="$ctrl.cardHover()"> <a class="contentcard" lk-track-action="Suggested for you" lk-track-name="homepage_item" ng-href="/dashboards/451" href="/dashboards/451"> <div class="contentcard--thumb"> <!-- ngIf: ctrl.itemImageUrl -->  <!-- end ngIf: ctrl.itemImageUrl --> ::after </div> <div class="contentcard--info"> </a> <div class="contentcard--favorite"> </div> </lk-homepage-item> </div> <!-- end ngRepeat: item in items | orderBy: ['order'] --> <div class="contentcard--container ng-scope contentcard--suggested" ng-class="{true: 'contentcard--container_small contentcard--favorites', false: 'contentcard--suggested'}>
```

The Styles panel on the right shows the following styles for the selected element:

```
Filter  
element.style {  
}  
@media screen and (min-width: 1140px) {  
  .contentcard--container, .contentcard--browse-table, .contentcard--container, .contentcard--container {  
    width: 25%;  
  }  
}  
@media screen and (min-width: 980px) {  
  .contentcard--container, .contentcard--browse-table, .contentcard--container, .contentcard--container {  
    width: 33.333%;  
  }  
}  
@media screen and (min-width: 706px) {  
  .contentcard--container, .contentcard--browse-table, .contentcard--container, .contentcard--container {  
    width: 50%;  
  }  
}
```

Search for it?

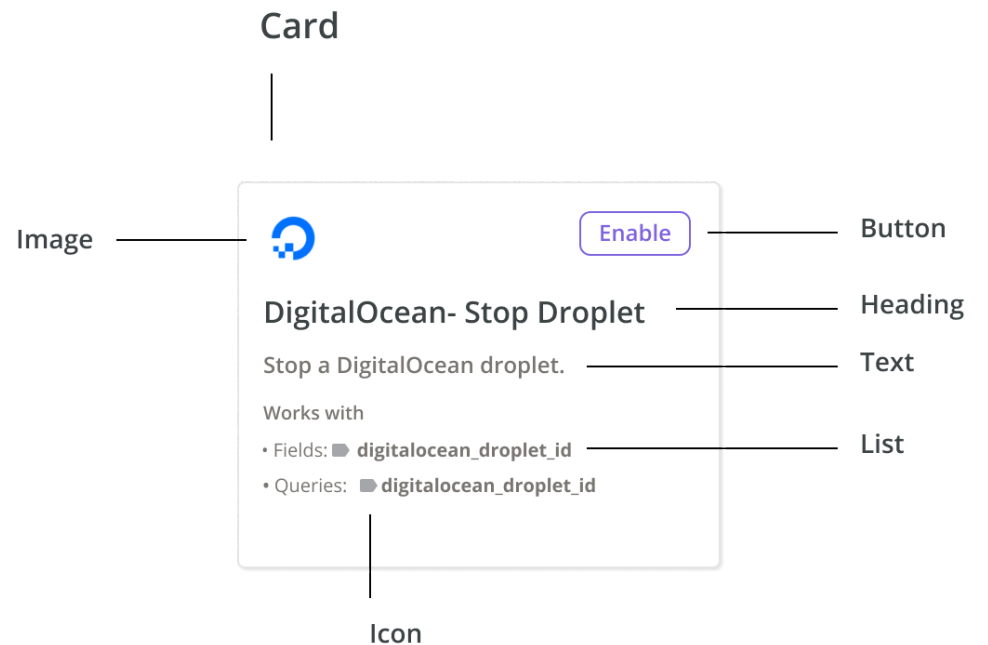
Open your code editor and search for card...



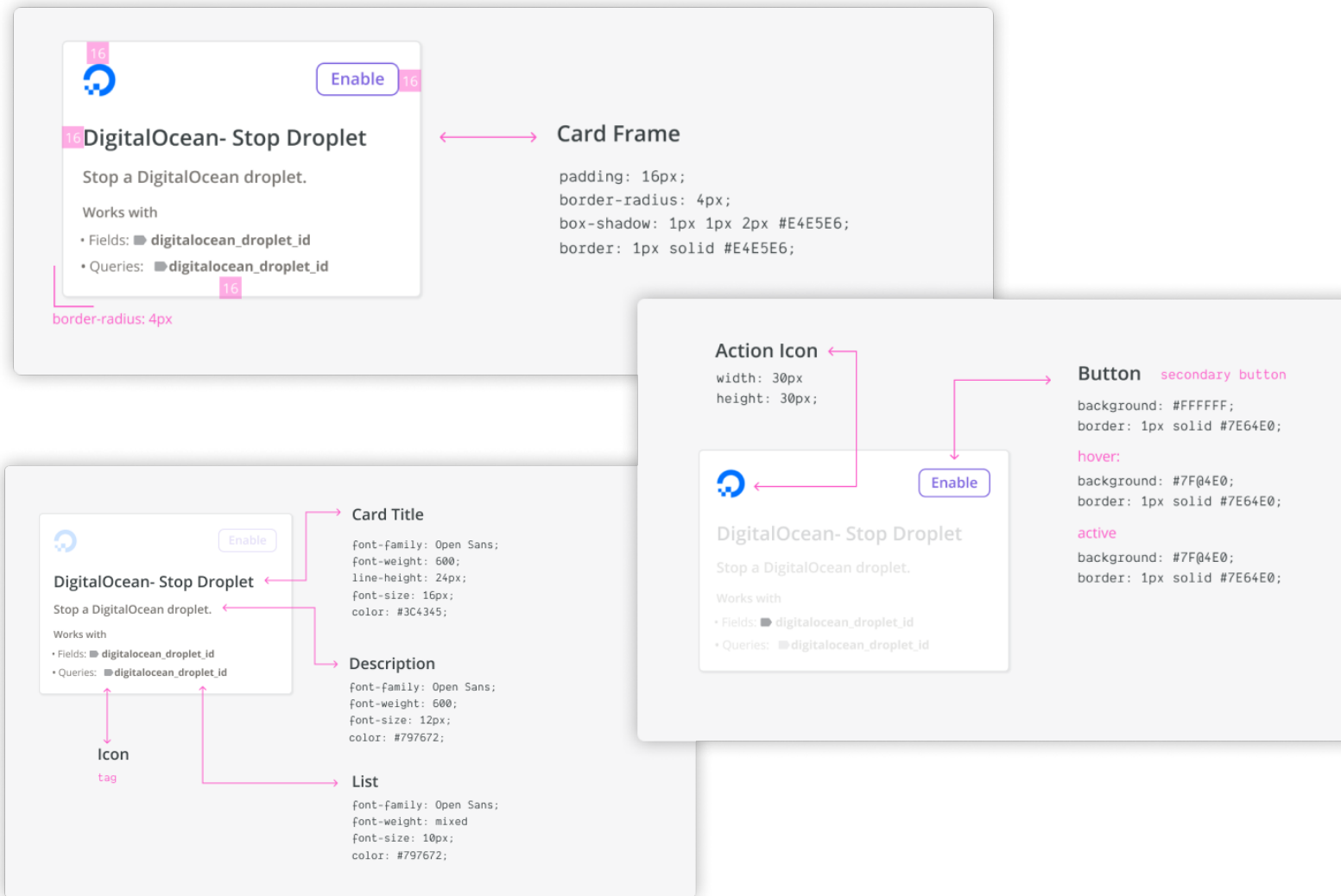
Hop on slack and search or ask in a channel?

The screenshot shows a Slack interface for a workspace named 'Looker'. The left sidebar lists several channels: #looker-benefits, #monday-club, #musicalookers, #ops, #places, #product, #proj-census-data-app, #random, #santa-cruz-office, #security, #sme-databases, #sql-sorcerers, #team-administration, #team-content-creation, #team-content-org, #team-core-infra, #team-data-delivery, #team-model-dev, #team-result-gen, and #ux-research. Below these are 'Direct Messages' including 'slackbot'. The main channel view is '#engineering', which has 102 members and 42 unread messages. A search bar at the top right contains the text 'card ui'. The channel content includes a 'Today' section with several announcements: 'Eng talk: Open Source at Looker - Roland' (2:00 PM to 2:25 PM), 'Nate Nate Paternity Leave' (April 30th - June 22nd, 2018), 'Jesse Callaway OOO' (May 17th - May 22nd, 2018), 'Elijah WF Austin' (May 21st - May 22nd, 2018), 'Rodney OOO - Mini Vacay' (May 21st - May 23rd, 2018), 'Kai OoO: Family Vacation (London & Croatia)' (May 21st - June 8th, 2018 at Croatia), and 'Adam WFH' (May 22nd, 2018). Below these is a message from 'aryeh' at 11:14 AM with a link to a GitHub pull request: <https://github.com/looker/helltool/pull/37330>. The message has 2 reactions and 2 replies. The right sidebar shows 'Search Results' for 'card ui', 'People & Channels' with 1 person, 'Top Files' including 'Looker UI Audit', and 'Top Messages' with a message from 'Ash Cunningham' at 12:10 PM mentioning 'card specs/styling roadmap, and UI audit/check day this afternoon.. but will o the...'. The bottom of the interface shows a message input field with a plus sign, '@', and a smiley face icon.

If we take a step back and look at our card to be built it can be broken up into discrete pieces. Which on the surface looks simple.



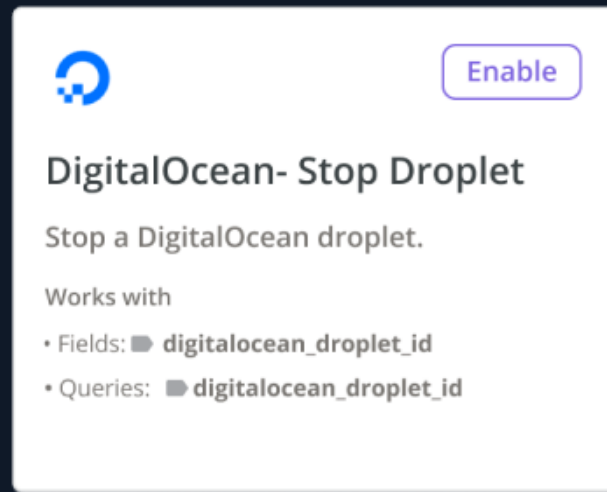
Hand off is where this can go wrong, those discrete pieces can actually be quite complicated



Instead of building with blocks you end up with something fragile and disorderd.



With Lens we want the card to be represented in a declarative language. Where the pieces can map to trusted components and to design elements.



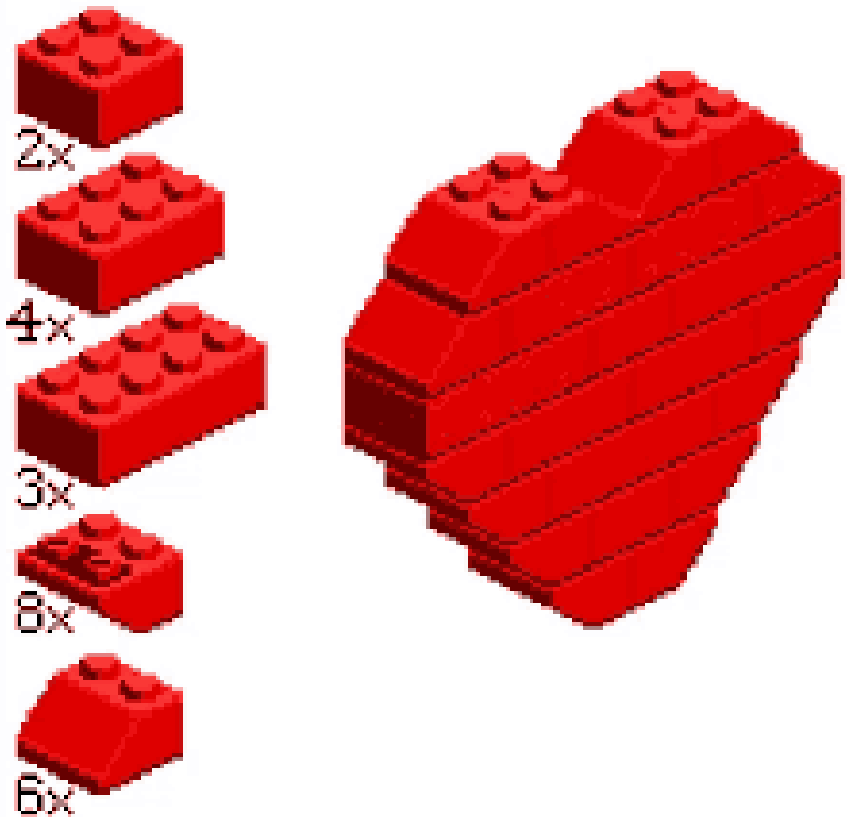
```
Card: {
  Button: {
    mode: 'light',
    size: 'xsmall',
    text: 'Enable',
    raised: true,
  },

  Heading: {
    level: '3',
    size: '4',
    text: ...
  }

  Text: {
    size: '5',
    mode: 'subdued',
    text: ...
  }

  List: {
    ListItem: {
      text: Field <Icon glyph="tag" /> ..
    },
    ListItem: {
      text: 'Queries: <Icon glyph="tag" />
    }
  }
}
```


Its a set of blueprints
that enable design and
engineering to build
with confidence.



Back to our new design,
we have found some
code to reuse, what
happens today?

The screenshot shows the Looker Actions page. The top navigation bar includes the Looker logo, 'Browse', 'Explore', and 'Develop' menus, along with search and help options. The left sidebar lists navigation categories: General (Settings, Labs), Users (Users, Groups, Roles, Content Access, User Attributes), Database (Connections, Queries, Persistent Derived Tables, Datagroups), Scheduler (Plans, History, External Emails), and Platform (Actions). The main content area is titled 'Looker Action Hub' and contains a list of integrations, each with an 'Enable' button and details on how to use them.

Integration	Description	Fields	Queries
Airtable	Add records to an Airtable table.	Any query	
AWS EC2 - Stop Instance	Stop an EC2 instance.	aws_resource_id	aws_resource_id
Azure Storage	Write data files to an Azure container.	Any Query, Any Dashboard	
DigitalOcean- Stop Droplet	Stop a DigitalOcean droplet.	digitalocean_droplet_id	digitalocean_droplet_id
DigitalOcean Spaces	Write data files to DigitalOcean's Spaces storage.	Any Query, Any Dashboard	
Google Cloud Storage	Write data files to a Google Cloud Storage bucket.	Any query, Any Dashboard	
Hipchat	Send a message to a Hipchat room referencing data.	Any Query	
JIRA	Create a JIRA issue referencing data.	Any Query	

You copy some markup
and adjust it to fit the
new requirement.



Modify it, tinker with
CSS, hunt for values,
compile, test and review
with team.



Just waiting for assets.json to compile



Wil Gieseler 2:42 PM

yall ever waiting for assets.json manifest to be built?



15 replies Last reply 7 days ago



Adam Markowitz 7 days ago

yeah. Takes a long time nowadays 😞 I'm usually at about 3m-5m or so. I typically try to build it once and then start the app with `--no-generate-assets` like @ian suggested. Would be awesome to get that time down to < 1m if someone happens to have some spare time on their hands 😊

Hope it works, no confidence in the system



How it works w/ Lens

You can come search Lens and see documented components and patterns that are easy to compose and extend

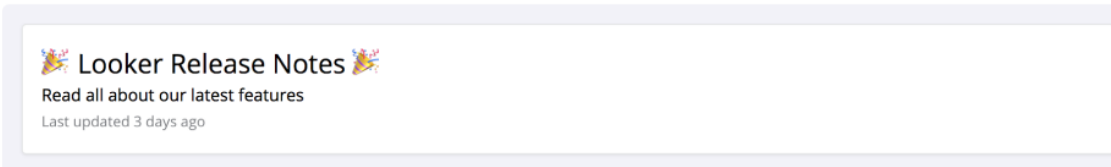
LENS

Components

- Card
- CardContent
- CardGroup
- CardMedia

A Card with content

Using the `CardContent` component quickly gives your content consistent spacing inside your `Card`. You can then use other components to layout your `Card`.



Looker Release Notes
Read all about our latest features
Last updated 3 days ago

VIEW CODE

```
<Card raised url="https://docs.looker.com/relnotes">
  <CardContent>
    <Heading size="2">🚀 Looker Release Notes 🚀 </Heading>
    <Heading level="4" size="5">Read all about our latest features</Heading>
    <Text size="6" mode="subdued">Last updated 3 days ago</Text>
  </CardContent>
</Card>
```


There are components for almost everything anyone would need and they map to the new Lens handoff language.

LENS

Filter by name


Lens

Style

- Color
- Typography
- Spacing

Components



- Block
- BlockLayout
- Button
- Card
- CardContent
- CardGroup
- CardMedia
- DataTable
- DataTableBody
- DataTableHead
- Heading
- Icon
- Link
- List
- ListItem
- Text

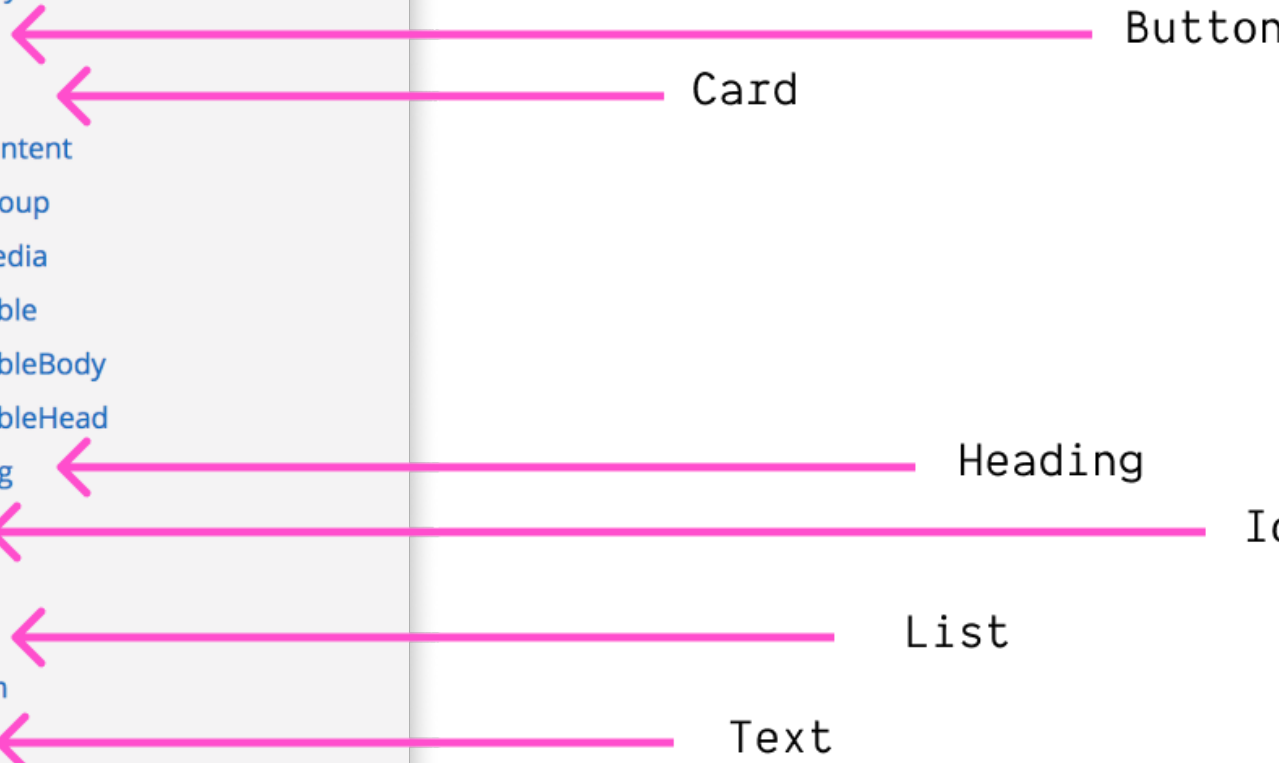
 Enable

DigitalOcean- Stop Droplet

Stop a DigitalOcean droplet.

Works with

- Fields:  digitalocean_droplet_id
- Queries:  digitalocean_droplet_id



You are able to tweak card settings in real time, instant feedback, didn't have to write a line of CSS.

You can now copy this piece of code and put into codebase and will work just as seen in Lens

The screenshot shows the LENS design tool interface. On the left is a sidebar with a search bar labeled 'Filter by name' and a list of components under 'Components' including Block, BlockLayout, Button, Card, CardContent, CardGroup, CardMedia, DataTable, DataTableBody, DataTableHead, Heading, Icon, Link, List, ListItem, and Text. The main area on the right is titled 'Card' and shows the file path 'src/components/Card/Card.tsx'. It contains a description: 'Cards provide scannable organized groups of content that is limited to one concept or thought per a card.' Below this is a section for 'PROPS & METHODS'. The next section is 'A basic Card', which explains that a Card is a slightly styled container. Below the text is a visual example of a basic card, which is an empty rectangular box with a light purple border and a subtle shadow. Underneath the example is a 'VIEW CODE' button and a code block containing the following code:

```
// A Card with Block component to add spacing
// Try typing some content into the Block or Card to see how it handles content by default

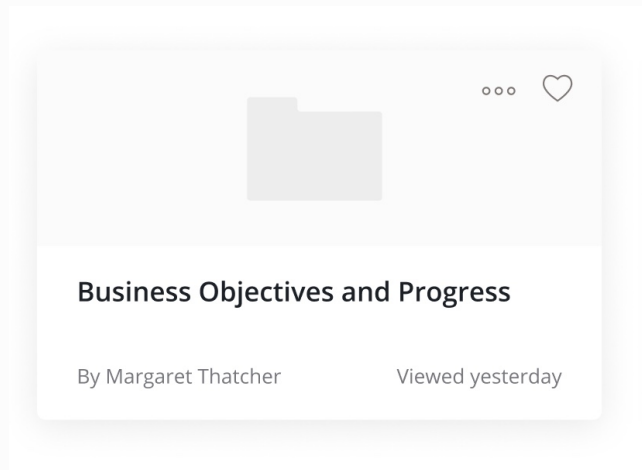
<Card raised>
  <Block p="4xl"></Block>
</Card>
```

A Card with content

Using the `CardContent` component quickly gives your content consistent spacing inside your `Card`. You can then use other components to layout your `Card`.

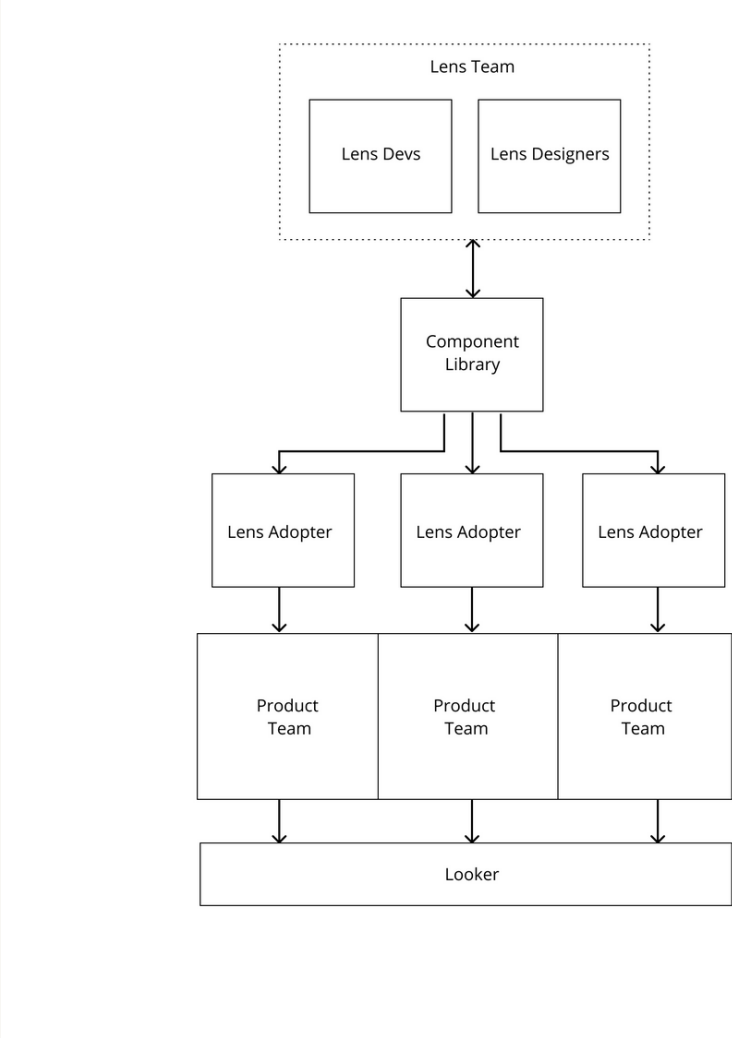
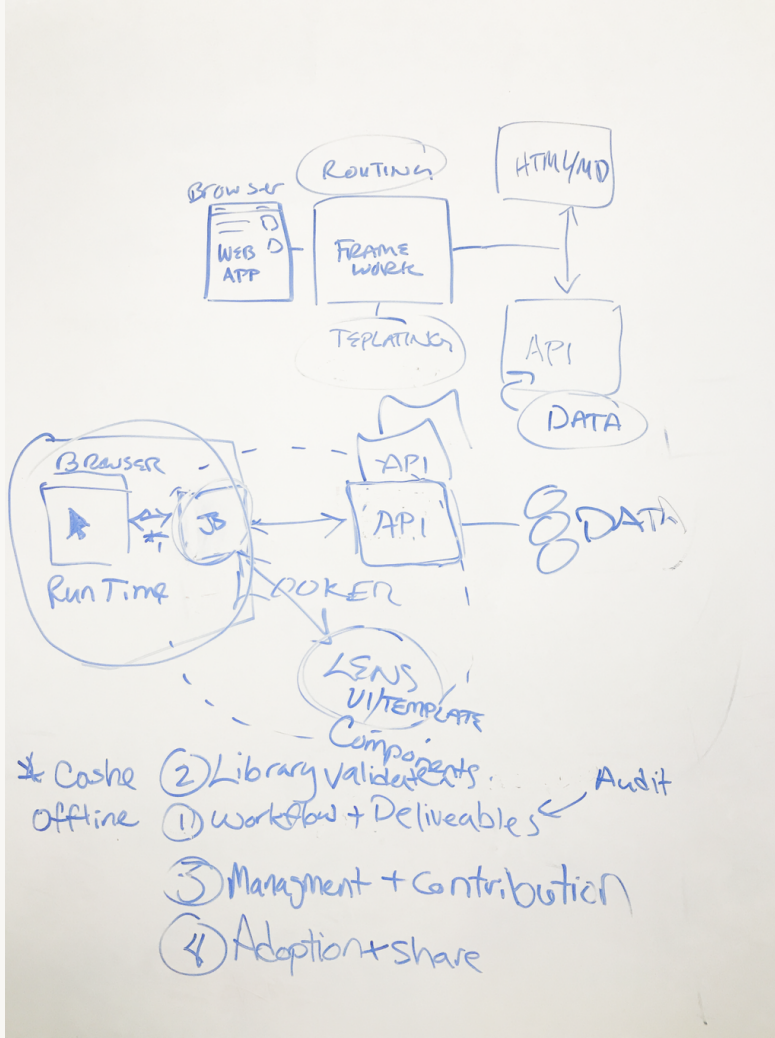
The visual example shows a card with a light purple border and shadow. Inside the card, there is a title 'Looker Release Notes' with a colorful icon on either side. Below the title is a subtitle 'Read all about our latest features' and a date 'Last updated 3 days ago'.

VIEW CODE

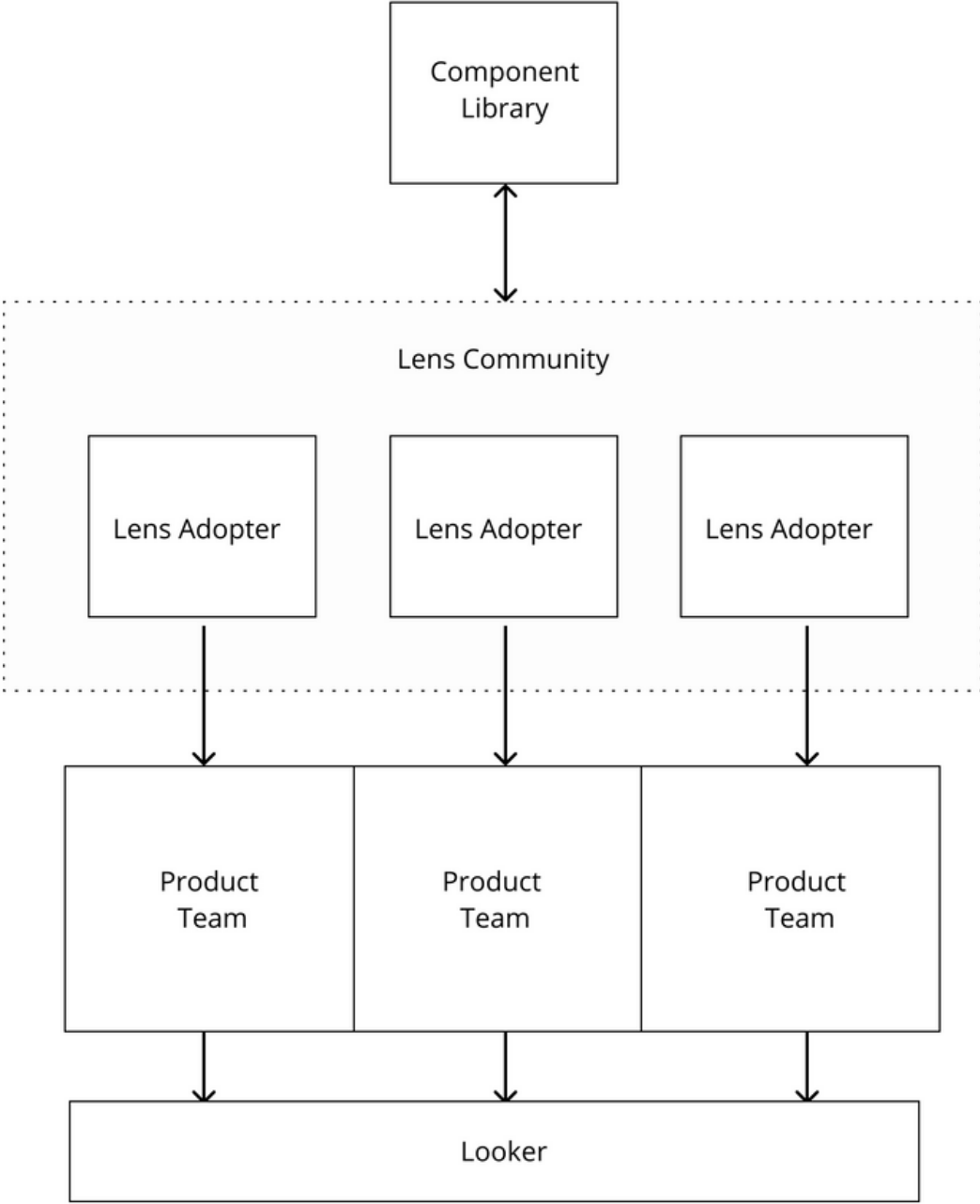


.. or take the base component and add some of their own markup to give new functionality and opens a PR to share it back to the library.

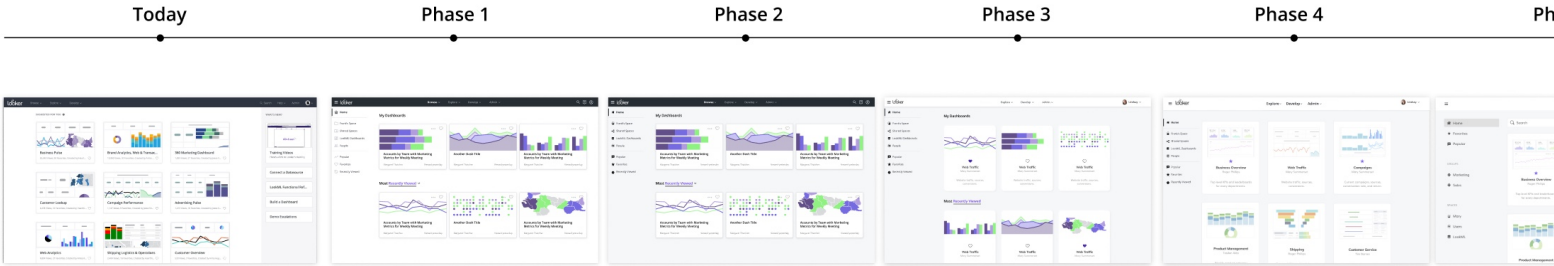
Workflow at the for the first milestone.



which evolves into a community driven model.



How to get to there: Phased Approach



Mary ▾

Overview

Website traffic, sources, conversions.

Web Traffic

Mary Summerset

Website traffic, sources, conversions.

Campaigns

Mary Summerset

Current campaigns, sources, conversation rate, and return.

Management

Top level KPIs and leaderboards for every departments.

Shipping

Roger Phillips

Top level KPIs and leaderboards for every departments.

Customer Service

Tim Barnes

Current campaigns, sources, conversation rate, and return.

Business Leaderboard

Tim Barnes

Quarterly sales tracking

looker

Sign In

looker Support tickets

Support tickets ☆

Support tickets, daily, last 7 days

Support tickets, average call time, in minutes

Support tickets, last 7 days, top 5 categories

Support tickets, average call time

Support tickets, average call time

Support tickets, average call time

Support tickets

Support tickets, daily

1,165

Support tickets, daily

Mary ▾

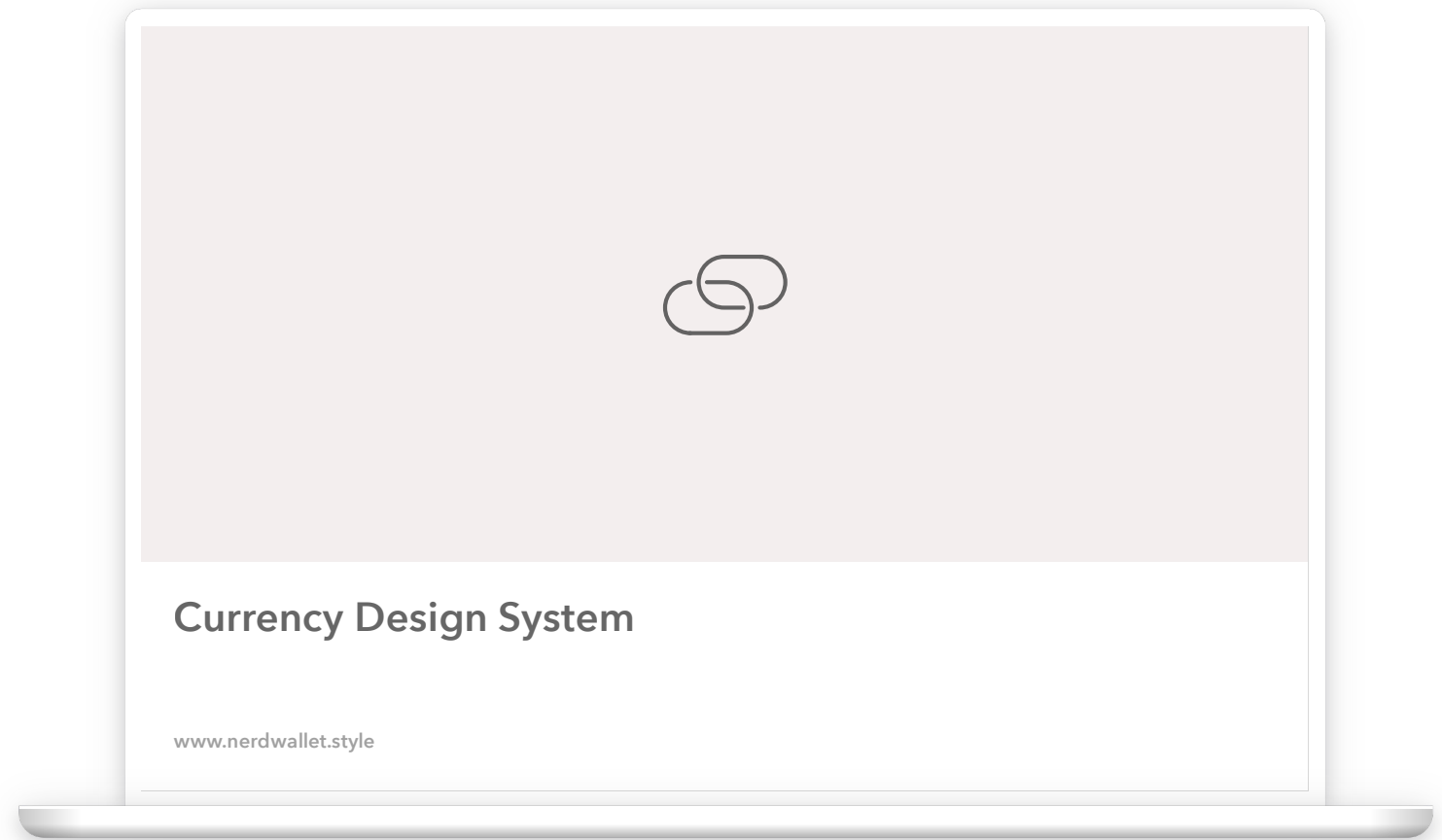
General > Business Overview >

Orders ▾ Viewed by Day ▾

Orders, Last 30 Days

NerdWallet

Currency Launch
Pattern Library &
Component
Documentation




Grid Design & Education

nerdwallet CREDIT CARDS BANKING INVESTING MORTGAGES LOANS INSURANCE MONEY [Join or log in](#)

[Advertiser disclosure](#)

Compare **rewards** credit cards for **excellent** credit

Filters: No annual fee No foreign transactions fee [+ More filters \(3\)](#) Sort by: [Editor's rating](#)



Partner name/product name goes here

★★★★★ 4.3 NerdWallet Review

TEXT

on XXXXXXX's secure website

LABEL	LABEL	LABEL	RECOMMENDED CREDIT SCORE
0%	00% - 00%	Sed ut perspiciatis unde omnis iste natus	690 850 Good - Excellent
Sed ut perspiciatis unde omnis iste natus	Sed ut perspiciatis unde omnis iste natus	Sed ut perspiciatis unde omnis iste natus	See if you may qualify

[WHY WE LIKE IT](#) [PROS & CONS](#) [PRODUCT DETAILS](#)

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt.

Renders as a div

Renders as a span

```

<Stack>
  <Box color="blue">Renders as a div</Box>
  <Box component="span">Renders as a span</Box>
</Stack>

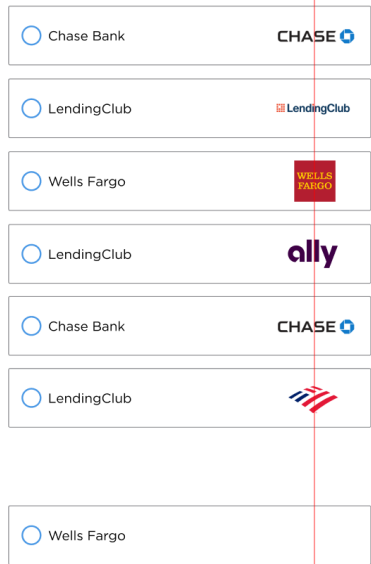
```

▶

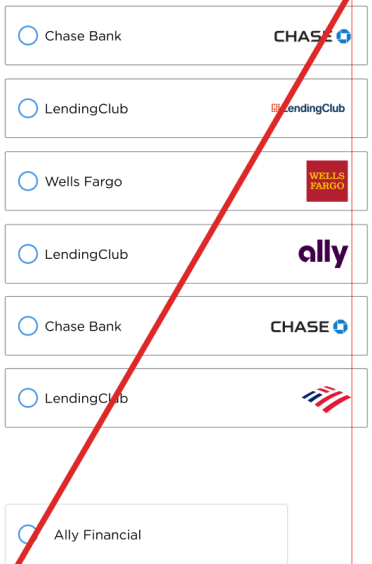
Interactive playroom

w/ Storybook

center ●

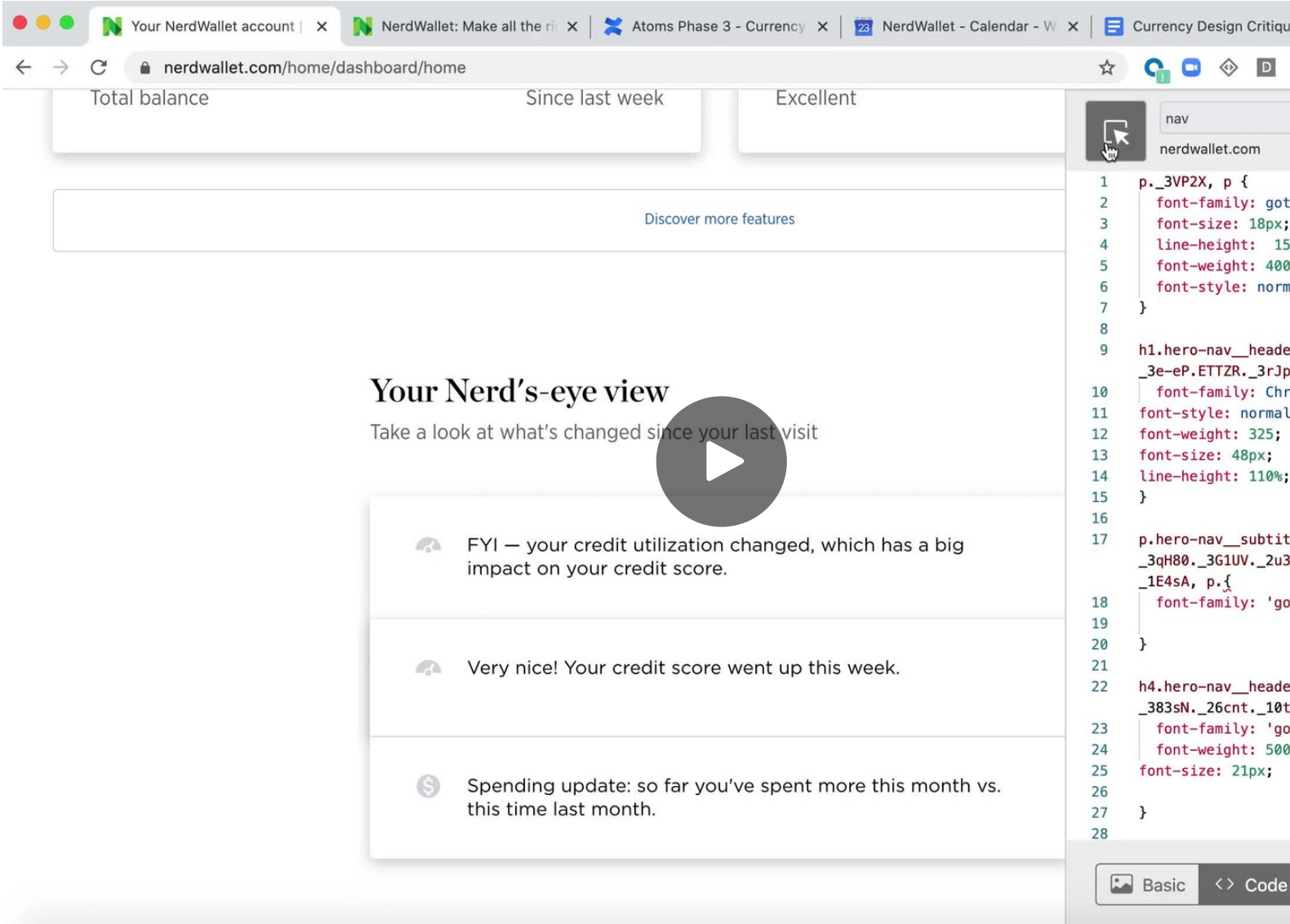


right alignment



Usage + Dos & Donts

HMW: Redesign the typeface of the product & blog demo



Figma File Versioning

The screenshot shows the NerdWallet Figma workspace interface. At the top, there is a navigation bar with the NerdWallet logo, a search bar, and menu items: NerdWallet, Home, Recent, Spaces, People, Apps, Templates, and a Create button. On the left, a sidebar lists various design assets under the 'Currency' section, including items like 'How does PEDD tran...', 'Adding Weightless in...', 'Fonts & Type - User ...', 'Consolidating Curren...', 'Page Templates, Pag...', 'Color - Semantic Col...', 'Design for Accessibil...', 'Style Guide Content ...', 'Roadmap', 'Project Charter', 'Decision log', and 'Figma Migration'. Below this is an 'Archived pages' section.

In the main workspace, a diagram shows a hierarchy of files. A box labeled 'POD Project File' is connected to two other boxes above it, illustrating a branching structure for file versions.

Pros:

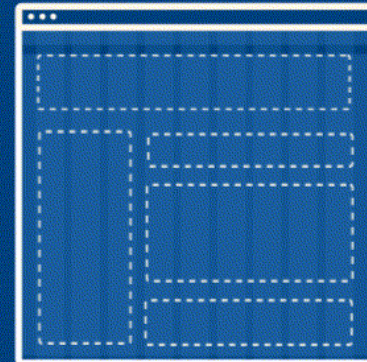
- Automatically overrides type with new styles in components
- Less risk of breaking links
- Versions are visually side by side in each page section.
- Fast and easy to push a type style updates
- Great for new systems when you are not sure what the look-and-feel and are still exploring

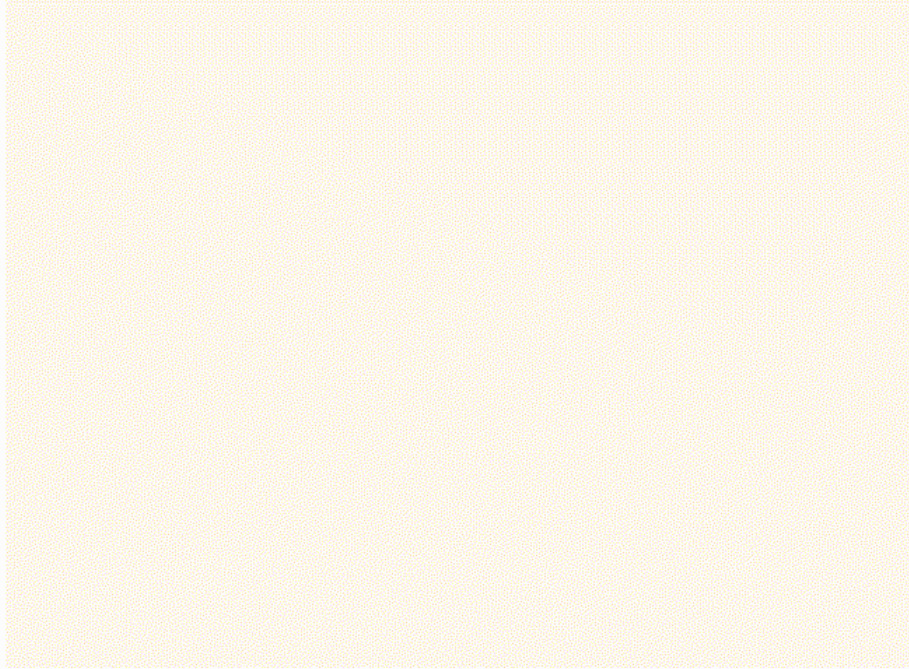
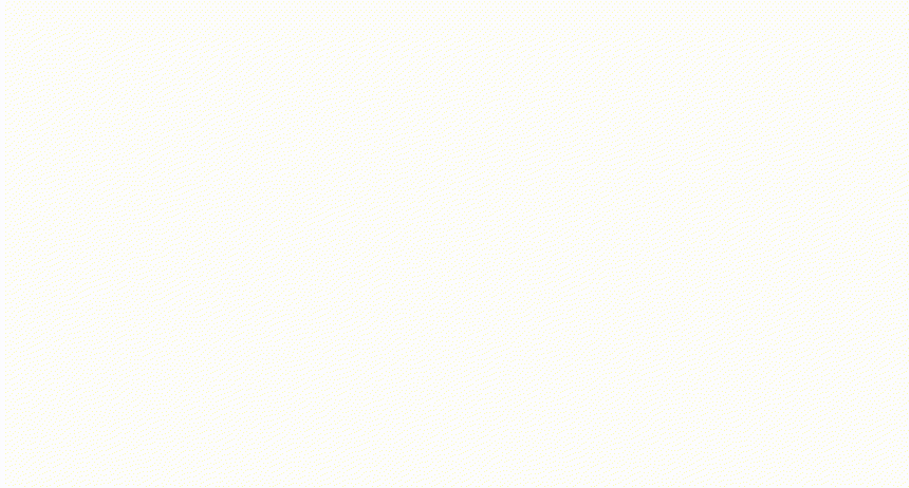
Cons:

- Page and layers names will need to be explicitly clear, organized by version
- Unclear to users which version they are using
- New versions may override components
- Analytics in Figma more difficult to track adoption
- Long list of styles in type panel for users to scroll through

Proposed: New File Versioning + Component Versioning Model

Animated Assets





Thank You



