Katherine Okpara

Senior UX Researcher

About Me

Who I am

- Senior UX Researcher
- Based in San Francisco, CA
- Brain and behavior enthusiast
- Passionate about connecting with people from all walks of life and understanding their experiences
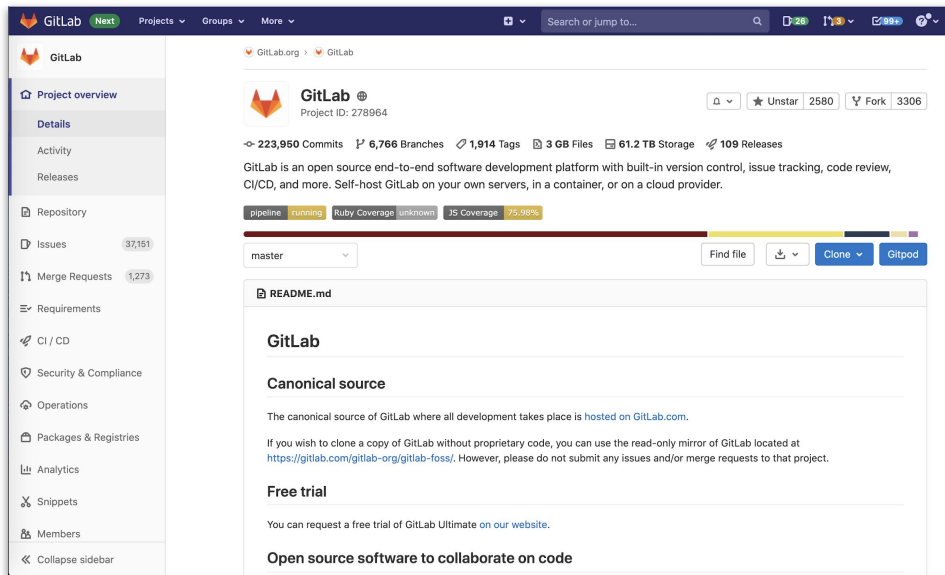
What I do

- Conduct strategic and tactical research using a variety of methods
- Teach product designers and product managers how to plan and conduct their own research projects
- Generate actionable insights and help teams define and evolve the product direction

Case Study

Evaluating the GitLab.com navigation experience

The System Usability Scale (SUS) is a standardized, 10-item questionnaire that measures the ease of use of websites, software, mobile devices, and other technological applications. I conduct a SUS survey every quarter as a way to track the perceived usability of GitLab over time.

GitLab's perceived usability has steadily declined since I first started running the survey. It currently rates as a "C" and GitLab's leadership aims for it to move up to a "B."
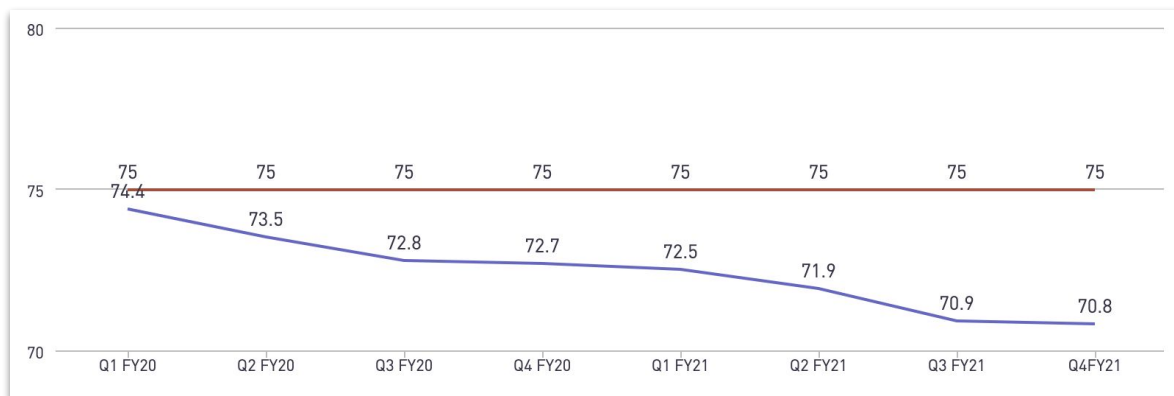
Synthesizing data across quarters

Last year, I synthesized survey data across several quarters and found that we were consistently receiving feedback about challenges with navigation and discoverability.

I hypothesized that these unresolved challenges were **negatively impacting** the perceived usability of the product.

After discussing these challenges with Product & UX leadership, "Navigation & Settings" became a dedicated category within my assigned product area. I launched a research initiative to establish a baseline measurement of the GitLab.com navigation experience.

What

Research to conducted to learn:
- How users expect to navigate throughout GitLab
- What content is easy to find
- What content is difficult to find

Why

- Better understand the mental models of GitLab users
- Uncover the most challenging aspects of navigation
- Establish a baseline for comparison

How

- Tree-testing
- Card-sorting
- Usability testing
- Analysis of feedback from research participants

Navigation baseline metrics initiative: high-level overview

GitLab is a robust application that has multiple types and layers of navigation. In order to get a comprehensive view of the existing user experience, I evaluated the findability of multiple navigation structures in the product.

- **Duration:** 2 months (January 2021 - February 2021)

- **Mixed-method approach:** Conducted 3 tree-testing studies, 1 card-sorting study, 1 moderated usability testing study, and synthesis of qualitative data gathered from various sources

- **Communication:** I led the research initiative and consistently communicated progress and early insights to the product manager and product designer who were directly responsible for navigation areas

- **Results and impact:** Delivered a report on baseline metrics and key challenges that laid the foundation for improvements

Problem Space

The left navigation structures had been growing organically since 2019

Navigation was an "unowned" category so there were no dedicated resources

GitLab's product offering had expanded to DevSecOps and beyond - doubling the amount of features displayed in the left navigation (including the 2nd-level items)

Changes were being made to individual sections without analysis of the impact to the overall structure
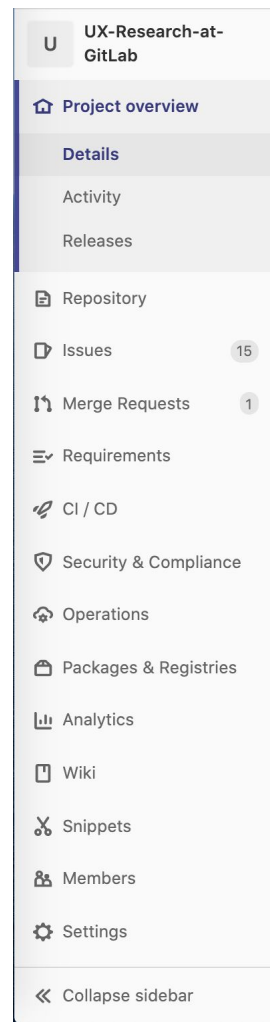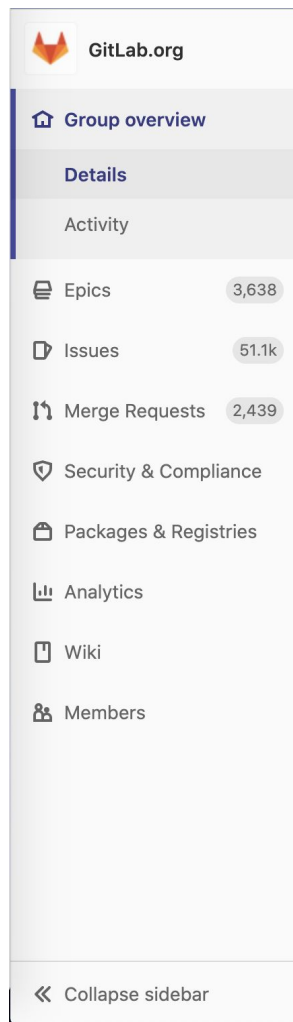
Benchmarking the group- and project-level left side navigation

Goal: Assess the findability of the left side navigation structures of GitLab.com and identify areas for improvement.

Methodology: 3 unmoderated tree-testing studies conducted with 30+ GitLab users each. Analysis of qualitative data gathered through open-text feedback and UserTesting.com recordings.

Participant characteristics: Random sample of users sourced from GitLab's data warehouse - primarily engineers using GitLab for 1+ years.

Area and Methodology	Overall Success Rate
Group-level left navigation tree-test (N = 30)	47%
Project-level left navigation tree-test (N = 52)	52%
Project-level settings tree-test (N = 70)	35%
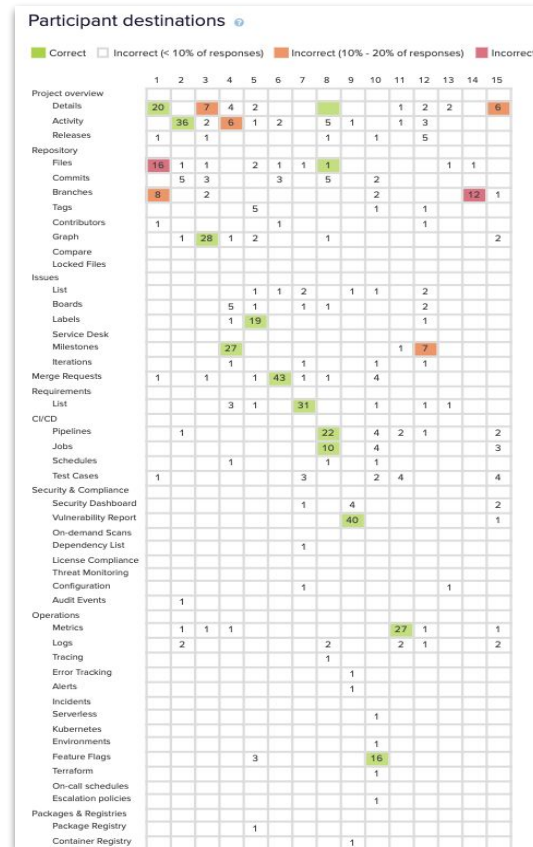
Individual task success rates varied greatly by product area

Though overall success rates were low for the left navigation, I found that certain product areas were more difficult to navigate than others.

Low Success

Participants struggled on most tasks related to these areas.

- **Operations**
- **Settings**
- **Analytics**

Moderate Success

Participants struggled on some tasks but were successful on others.

- **Issues**
- **Project overview**
- **Repository**

High Success

Participants easily completed tasks related to these areas.

- **Merge Requests**
- **Wiki**
- **CI/CD**
- **Security & Compliance**

Operations information wasn't where it was expected to be

In a test of Operations navigation items, I saw a low overall success rate of 30%.

Why? On several tasks, there were stark differences between participants' expectations and the current organization structure.

Where would you go to…	Answer	Success Rate	Top incorrect choices
View your metrics dashboard	Operations	15%	• Analytics (71% of participants)
Mark an incident as resolved	Operations	23%	• Issues (69% of participants)
Turn a feature flag on	Operations	23%	• CI/CD • Settings • Repository

Operations

Metrics

Logs

Tracing

Error Tracking

Alerts

Incidents

Serverless

Terraform

Kubernetes

Environments

Feature Flags
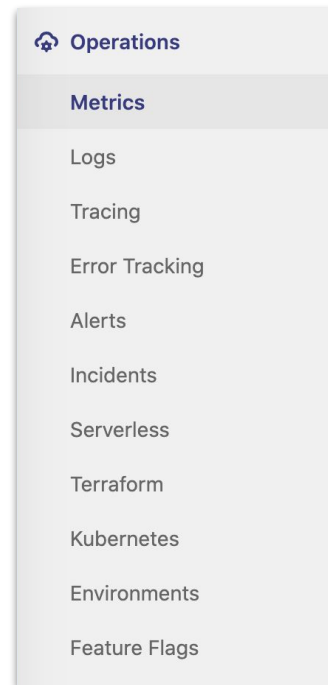
Users weren't aligned on where they expected to find Settings information

In a tree-test of project-level settings, I saw a low overall success rate of 35%.

Why? Participants often chose the wrong category under Settings or looked for the setting directly in the feature area.

Where would you go to...	Answer	Success Rate	Top incorrect choices
Create a deploy token	Settings> Repository	9%	• Settings>Access Tokens • CI/CD
Define Merge Request approvers	Settings> General	17%	• Settings>Repository • Merge Requests • Members
Set the project path for a pipeline subscription	Settings> CI/CD	23%	• CI/CD>Pipelines • CI/CD>Editor • CI/CD>Schedules
Update a project's description	Settings> General	36%	• Project overview>Details (63% of participants)

⚙ **Settings**

General

Integrations

Webhooks

Access Tokens

Repository

CI / CD

Operations

Pages
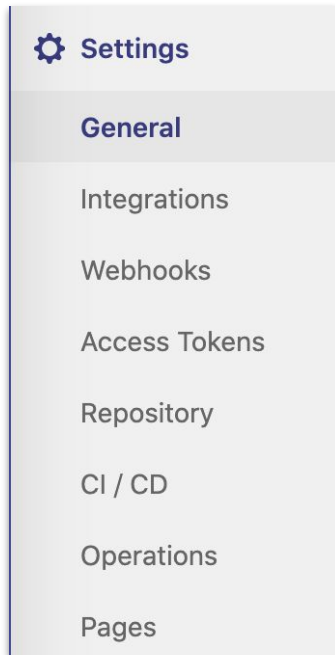
Analytics information had the highest failure rate

In both the project and group-level tree-tests, participants struggled to find analytics information.

Why? First-click rates were split between Analytics, Project/Group overview, and other feature areas for these tasks. If participants continued to look in Analytics, they often selected Analytics>Insights as their final choice.

Where would you find...	Answer	Success Rate	Top incorrect choices
Code coverage statistics for a project	Analytics> Repository	11%	• Analytics>Insights • Analytics>Code Review • CI/CD
Time it took to go from idea to production	Analytics> Value Stream	13%	• Analytics>Insights • Issues • Project overview
Summary of the test code coverage for projects in a group	Analytics> Repositories	10%	• Analytics>Insights • Group overview

Analytics
CI / CD
Code Review
Insights
Issue
Merge Request
Repository
Value Stream
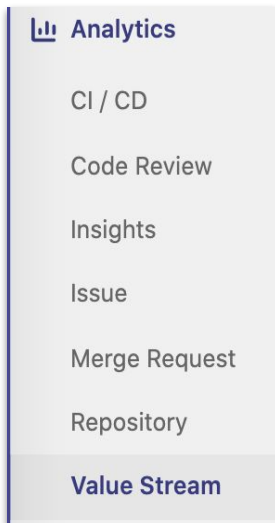
Follow-up research to evaluate and implement recommended improvements

I outlined several follow-up initiatives to dive deeper into trends from the data. I also created a plan for conducting iterative research to evaluate targeted improvements to the product areas where participants struggled.

- Iterative research plan to evaluate proposed left side navigation structures:

    - https://gitlab.com/groups/gitlab-org/-/epics/5687

- Exploratory initiatives focused on improving information architecture:

    - Explore improvements to categorization of Settings

    - Explore improvements to the categorization of Analytics

Key takeaways from the navigation research initiative

Key challenges in GitLab's navigation experience

After analyzing and synthesizing the data gathered across the research studies, I was able to identify the top 3 challenges that negatively impact the navigation experience:

Challenge 1

Unintuitive naming and structure of information

Challenge 2

Difficulty finding and managing settings

Challenge 3

Overwhelming amount of items in the left navigation

Challenge 1: Unintuitive naming and structure of information

We're organizing a lot of information in the way we communicate about it at GitLab (Conway's Law). As a result, the terminology and groupings we use in the product do not always resonate with our users.

"gitlab is capable of a lot, I mean A LOT of things and I can't understand more than just some basic ones."

"When I need to access a function outside my normal workflow, it is way too much [effort] to find where it is hidden in the UI."

"The different options are not necessarily intuitive to find...When one is searching for some option, one needs to search through several options to make it work."

Manage	Plan	Create	Verify	Package	Secure	Release	Configure	Monitor	Protect
Subgroups	Issue Tracking	Source Code Management	Continuous Integration (CI)	Package Registry	SAST	Continuous Delivery	Auto DevOps	Runbooks	Container Scanning
Audit Events	Time Tracking	Code Review	Code Quality	Container Registry	DAST	Pages	Kubernetes Management	Metrics	Security Orchestration
Audit Reports	Boards	Wiki	Code Testing and Coverage	Helm Chart Registry	Fuzz Testing	Review Apps	Secrets Management	Incident Management	Container Host Security
Compliance Management	Epics	Static Site Editor	Load Testing	Dependency Proxy	Dependency Scanning	Advanced Deployments	ChatOps	Logging	Container Network Security
Code Analytics	Roadmaps	Web IDE	Browser Performance Testing	Release Evidence	License Compliance	Feature Flags	Serverless	Tracing	
DevOps Reports	Service Desk	Live Preview	Usability Testing	Git LFS	Secret Detection	Release Orchestration	Infrastructure as Code	Error Tracking	
Value Stream Management	Requirements Management	Snippets	Accessibility Testing		Vulnerability Management		Cluster Cost Management	Product Analytics	
Insights	Quality Management		Merge Trains						
	Design Management								

What we should do: Leverage research insights to structure information in a way that aligns with the mental models of GitLab users.

Challenge 2: Difficulty finding and managing settings

Users often describe challenges with managing settings in the SUS survey feedback. This research showed us that figuring out where settings are located is a large part of that challenge.

"I find myself clicking through all of the different settings sections to find something."

"Some settings seem like they can be grouped in multiple places. I generally get confused between Settings > General and Settings > Repository."

"What setting is where under the Settings menu seems arbitrary at times and often requires searching each likely category."
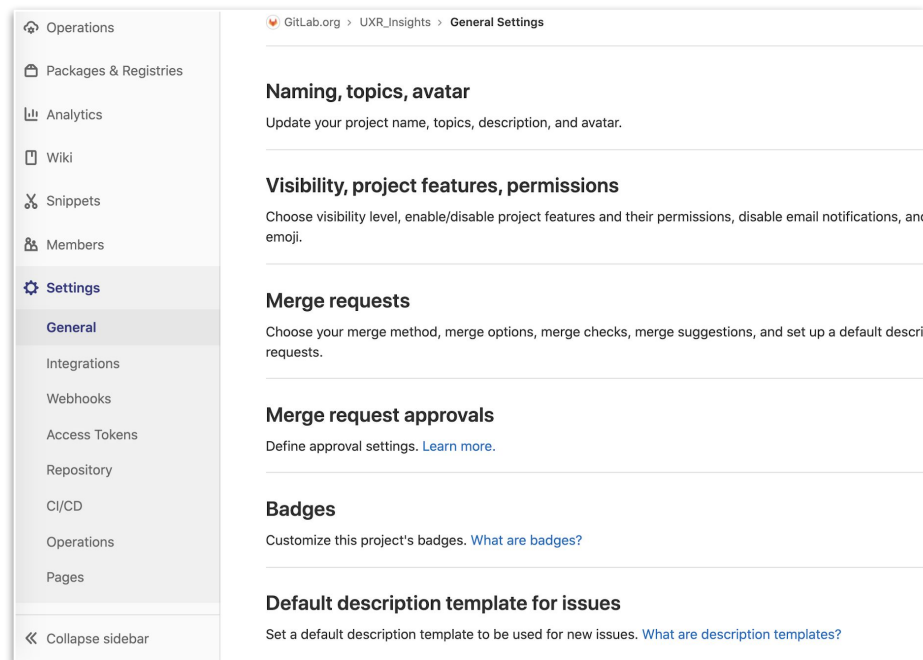
What we should do: Improve the categorization of settings. Provide contextual access to settings. Make settings searchable.

Challenge 3: Overwhelming amount of items in the left navigation

Usage of the features varies by persona and user goals. Users can feel overwhelmed by the amount of items displayed in the left navigation, especially if they don't have use cases for the entire DevOps lifecycle.

"The user interface is overloaded with functions and it is hard to know where to find something."

"Sometimes, it would be helpful to be able to 'hide' aspects related to programming even more, so that non-technical users [don't] get confused."

"Showing every single tool on all repositories without an option to hide some of them makes the interface feel intimidating."
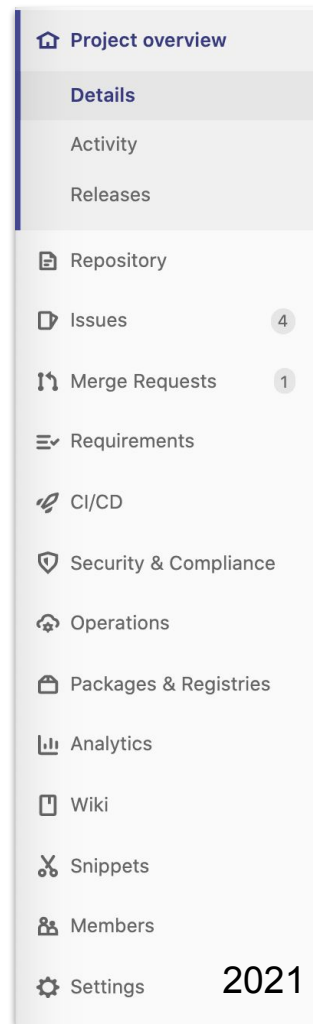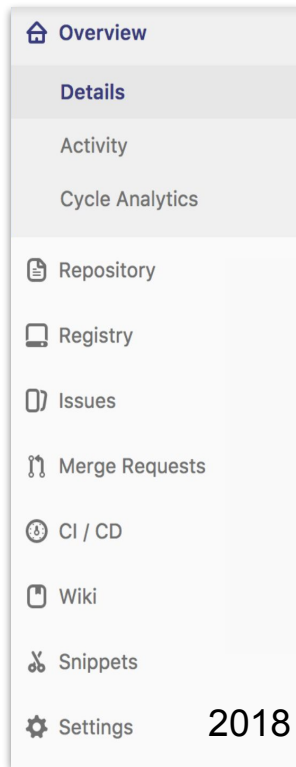
What we should do: Carefully evaluate decisions to add new items to the navigation. Consider impact on the findability of existing content.

Overview
- Details
- Activity
- Cycle Analytics
- 📄 Repository
- 🖥 Registry
- 〔〕 Issues
- ⌥ Merge Requests
- ◉ CI / CD
- 📖 Wiki
- ✂ Snippets
- ⚙ Settings

2018

- ⌂ Project overview
  - Details
  - Activity
  - Releases
- 📄 Repository
- 🗇 Issues   4
- ⌥ Merge Requests   1
- ☱ Requirements
- ⌕ CI/CD
- 🛡 Security & Compliance
- ☁ Operations
- 💼 Packages & Registries
- 📊 Analytics
- 📖 Wiki
- ✂ Snippets
- 👥 Members
- ⚙ Settings

2021

Sharing research insights with the team and broader company

After I finalized the baseline metrics and identified the top 3 challenges, I created presentation materials to help me easily share research insights with the UX and Product departments and broader company.
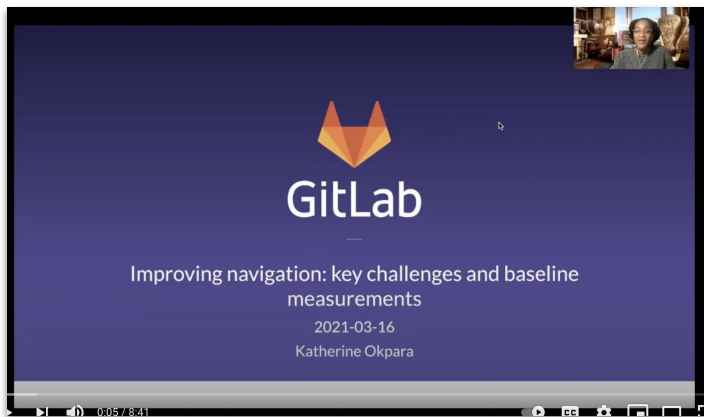
YouTube Video: Key challenges and baseline metrics gathered from the navigation initiative.
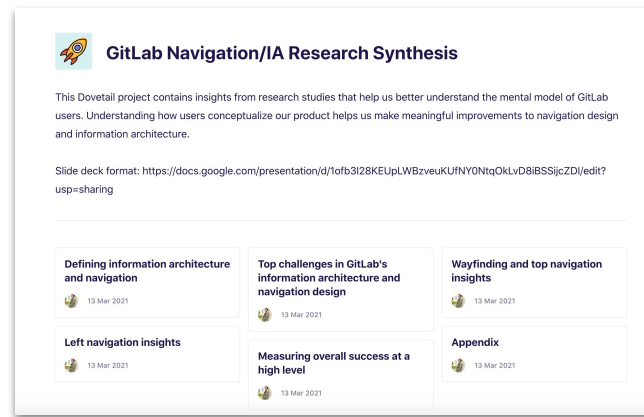
Dovetail: Data synthesis and guidelines for maintaining IA & navigation structures going forward.

What I learned

I had to work fast on this project so that my team could start incorporating insights into their work. If I had the opportunity to slow down and do this project again, I'd keep the following in mind...

- When conducting information architecture research, it's best to get well-acquainted with the full scope of the available content in order to define relationships and plan for growth over time

- Balancing the needs and priorities of different personas in a single application can be very challenging - investigate the needs of primary users first and follow-up with research for secondary user groups

- Gather sufficient qualitative data along with quantitative metrics for as many studies as possible - this helps you answer *why* users have certain expectations and strengthens recommendations

Thank you!

Any Questions?