

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

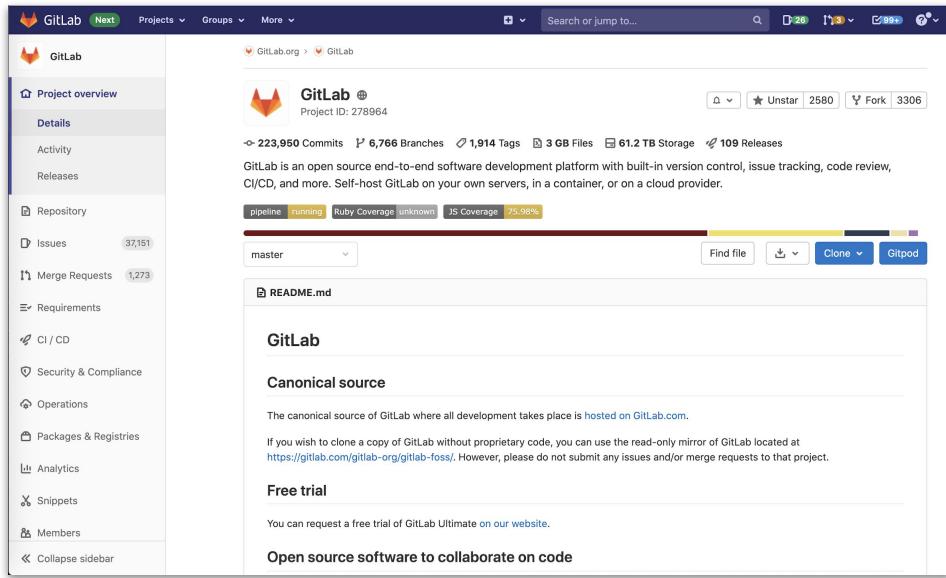
- Conduct strategic and tactical research using a variety of methods
- Teach product designers and product managers how to plan and conduct their own research projects
- Generate actionable insights and help teams define and evolve the product direction





Case Study

Evaluating the GitLab.com navigation experience



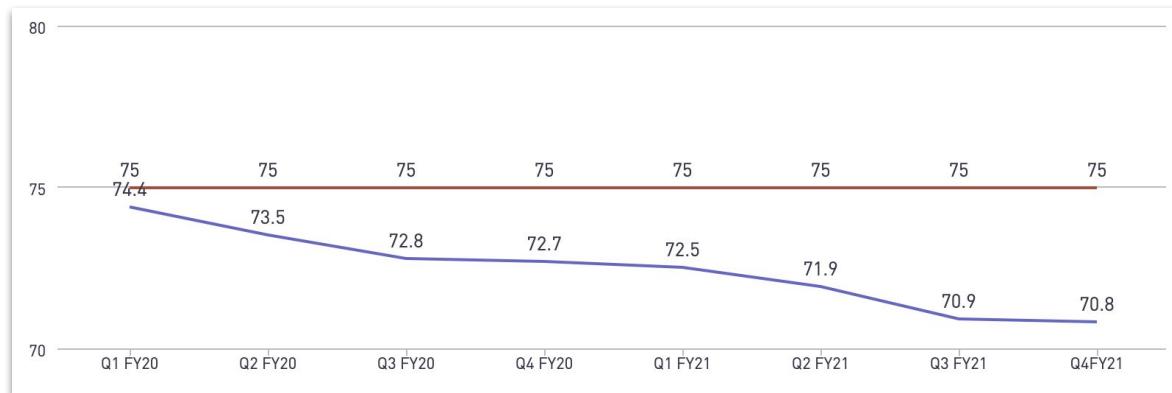
The screenshot shows the GitLab.com project overview for the 'GitLab' project (Project ID: 278964). The sidebar on the left lists various project management sections: Details, Activity, Releases, Repository, Issues (37,151), Merge Requests (1,273), Requirements, CI / CD, Security & Compliance, Operations, Packages & Registries, Analytics, Snippets, and Members. The main content area displays the project's statistics: 223,950 Commits, 6,766 Branches, 1,914 Tags, 3 GB Files, 61.2 TB Storage, and 109 Releases. It also shows a pipeline status bar with 'running' and coverage percentages (Ruby Coverage: unknown, JS Coverage: 75.98%). Below this is a search bar with 'master' selected, and buttons for 'Find file', 'Clone', and 'Gitpod'. The 'README.md' file is shown with the title 'GitLab' and the sub-section 'Canonical source'. It explains that the canonical source is on GitLab.com and provides a read-only mirror at <https://gitlab.com/gitlab-org/gitlab-foss/>. The 'Free trial' section is also visible.

Background: measuring perceived system usability over time



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.

Content	Tags	Note
Default navigation in GitLab is cumbersome, and navigating groups and projects rarely defaults to what I need. Moving out of the WebIDE back to what you where doing prior is not intuitive.	Editor 12, Navigation 50	Q3 FY20 - Optimizing GitLab
navigating between lots of project may be sometimes really slow. It would be nice to have a "fuzzy search" like the one that you have with "t" to browse files within a repository. Hitting a keyboard shortcut and then being able to enter a fuzzy string to jump right into a group/project will be really useful	Navigation 50, Search 19	Q3 FY20 - Usability
make the interface simpler to use and easier to navigate	Learnability 67, Navigation 50	Q1 FY20 - Usability
Simpler configuration for removing branch-specific containers would help keep the container registry easier to navigate	Navigation 50	Q3 FY20 - Optimizing GitLab
I would like to optimise mobile use, I like to review code and take actions from my mobile device. Sometimes I find navigation a bit more difficult.	Mobile 25, Navigation 50	Q3 FY20 - Optimizing GitLab
Quite a lot of functionality is hidden somewhere in the UI and seemingly	Discoverability 14	Q3 FY20 - Optimizing GitLab

1	A Verbatims - Is there anything you'd like to say about GitLab's usability?	SUS Theme	C New or Mature User?	D Plan	E Interested in Follow-Up?	F		G		H	
						Issue Type	Severity	Issue Type	Severity	Issue Type	Severity
189	Cluttered and counter intuitive. When I need to outside my normal workflow, it is way to much to hidden in the UI, and then about half the time it want - or what I thought it was supposed to do.	Navigation & discoverability	New	Silver	No	UI polish	3 - Major				
190	Compared to GitHub GitLab feels very heavy an. The structural organization is not clear and some inerations are unnesesary.	Navigation & discoverability	New	Free	No	Performance	Unsure				
191	Navigation between sections is a pain, the search to help but is slow, the support for next/previous (eg. filters in issue board)	Navigation & discoverability	Mature	Silver	Yes	Unknown	3 - Major				
192	1. Complaint from VP (who's in charge of purch GitLab) I can never find where I was working last "Here's the activity feed; uh... just scroll down until picture on one of the actions." Having a view of sorted by time would increase our chances of staying in the long run.	Navigation & discoverability	Mature	Silver	Yes	Feature enhancement	4 - Low				
193	The user interface is overloaded with functions know where to find something. Some sense of according to functionality and typical workflows. Colors may help as well. The focus should be content rather than always having all navigation write this with a year of experience in gitlab and experience with github.	Navigation & discoverability	New	Free	Yes	UI polish	Unsure				
194	Sometimes it feels like there is way too much in screen, and is difficult to find what one needs in a team.	Navigation & discoverability	New	Free	No	Usability	Unsure				
195	There should be a easier way navigating issues documentation in a team.	Navigation & discoverability	Mature	Free	Yes	Usability	4 - Low				
196	I can't find my projects from every page, I always the "projects"-side and from there further to my	Navigation & discoverability	New	Free	No	Usability	3 - Major				

Navigation baseline metrics initiative, January 2021- February 2021



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 conduct 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

GitLab.org

Group overview

Details

Activity

Epics 3,638

Issues 51.1k

Merge Requests 2,439

Security & Compliance

Packages & Registries

Analytics

Wiki

Members

« Collapse sidebar

UX-Research-at-GitLab

Project overview

Details

Activity

Releases

Repository

Issues 15

Merge Requests 1

Requirements

CI / CD

Security & Compliance

Operations

Packages & Registries

Analytics

Wiki

Snippets

Members

Settings

« Collapse sidebar

Benchmarking the group- and project-level left side navigation

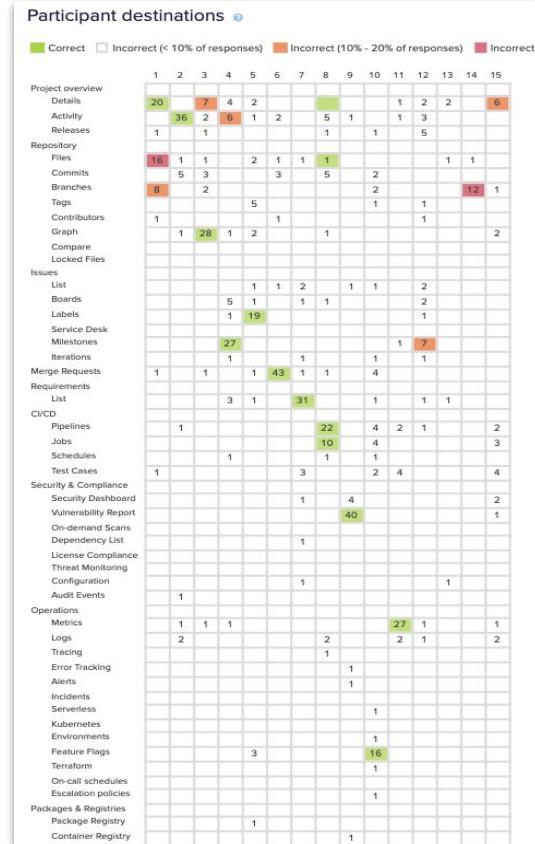


Goal: Assess the findability of the left side navigation structures of GitLab.com and identify areas for improvement.

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
<u>Group-level left navigation tree-test</u> (N = 30)	47%
<u>Project-level left navigation tree-test</u> (N = 52)	52%
<u>Project-level settings tree-test</u> (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

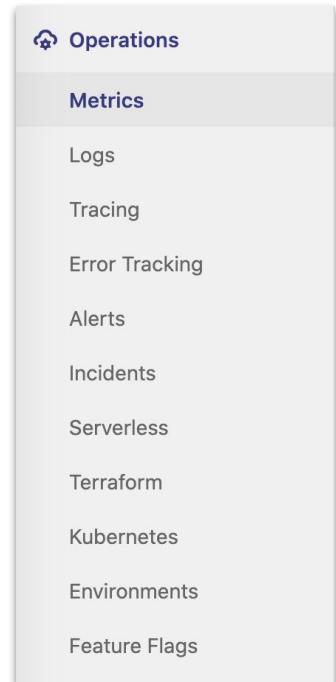
Operations information wasn't where it was expected to be



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%	<ul style="list-style-type: none">Analytics (71% of participants)
Mark an incident as resolved	Operations	23%	<ul style="list-style-type: none">Issues (69% of participants)
Turn a feature flag on	Operations	23%	<ul style="list-style-type: none">CI/CDSettingsRepository



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	Settings
Create a deploy token	Settings>Repository	9%	<ul style="list-style-type: none">Settings>Access TokensCI/CD	General
Define Merge Request approvers	Settings>General	17%	<ul style="list-style-type: none">Settings>RepositoryMerge RequestsMembers	Integrations
Set the project path for a pipeline subscription	Settings>CI/CD	23%	<ul style="list-style-type: none">CI/CD>PipelinesCI/CD>EditorCI/CD>Schedules	Webhooks
Update a project's description	Settings>General	36%	<ul style="list-style-type: none">Project overview>Details (63% of participants)	Access Tokens

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	Analytics
Code coverage statistics for a project	Analytics>Repository	11%	<ul style="list-style-type: none">Analytics>InsightsAnalytics>Code ReviewCI/CD	CI / CD Code Review Insights Issue Merge Request Repository
Time it took to go from idea to production	Analytics>Value Stream	13%	<ul style="list-style-type: none">Analytics>InsightsIssuesProject overview	Value Stream
Summary of the test code coverage for projects in a group	Analytics>Repositories	10%	<ul style="list-style-type: none">Analytics>InsightsGroup overview	

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	Container Host Security
Audit Events	Time Tracking	Code Review	Code Quality	Container Registry	DAST	Pages	Kubernetes Management	Metrics	Security Orchestration	Container Network Security
Audit Reports	Boards	Wiki	Code Testing and Coverage	Helm Chart Registry	Fuzz Testing	Review Apps	Secrets Management	Incident Management	Logging	Container Host Security
Compliance Management	Epics	Static Site Editor	Load Testing	Dependency Registry	Dependency Scanning	Advanced Deployments	ChatOps	Tracing	Error Tracking	Product Analytics
Code Analytics	Roadmaps	Web IDE	Dependency Proxy	License Compliance	Feature Flags	Serverless	Infrastructure as Code	Cluster Cost Management		
DevOps Reports	Service Desk	Live Preview	Browser Performance Testing	Release Evidence	Secret Detection	Release Orchestration				
Value Stream Management	Requirements Management	Snippets	Usability Testing		Vulnerability Management					
Insights	Quality Management		Accessibility Testing	Git LFS						
	Design Management		Merge Trains							

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

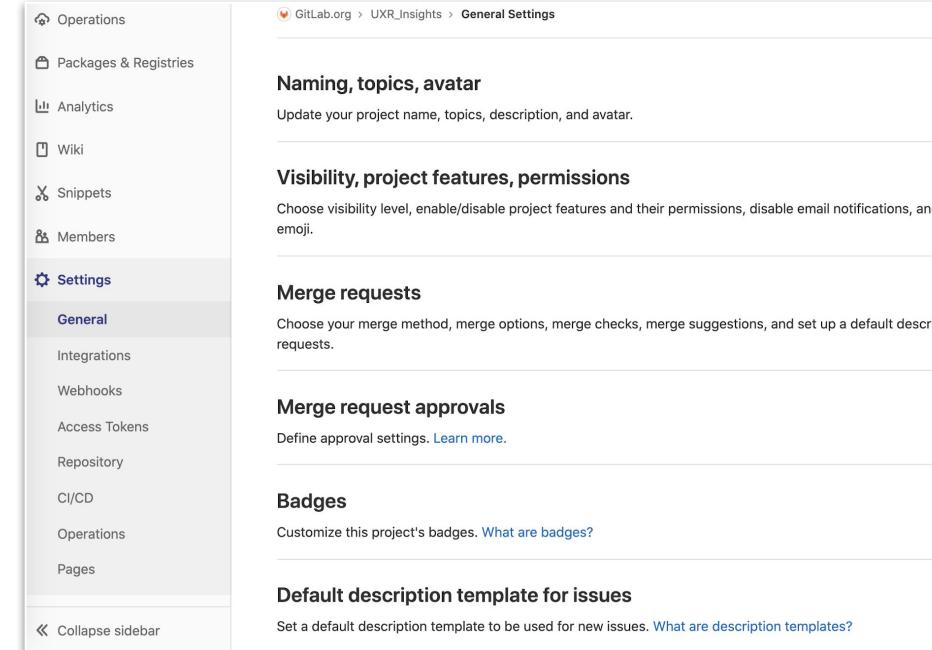
Users often describe challenges with managing settings in the SUS survey feedback. This research showed us that figuring out where settings are located is a large part of that challenge.

“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.



The screenshot shows the 'General Settings' page on GitLab. The left sidebar lists several categories: Operations, Packages & Registries, Analytics, Wiki, Snippets, Members, Settings (which is selected), General, Integrations, Webhooks, Access Tokens, Repository, CI/CD, Operations, and Pages. The main content area is titled 'Naming, topics, avatar' with a sub-section for updating project name, topics, description, and avatar. Another section, 'Visibility, project features, permissions', allows setting visibility level, project features, and permissions. The 'Merge requests' section covers merge methods, options, and suggestions. The 'Merge request approvals' section defines approval settings. The 'Badges' section allows customization of project badges. The 'Default description template for issues' section sets a template for new issues. A 'Collapse sidebar' button is at the bottom of the sidebar.



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.

The diagram illustrates the evolution of the GitLab left navigation over three years. It features two main panels: a '2018' panel on the left and a '2021' panel on the right, connected by a large black arrow pointing from left to right.

2018 Panel (Left): This panel shows a vertical list of items under the 'Overview' header. The items are: Details, Activity, Cycle Analytics, Repository, Registry, Issues, Merge Requests, CI / CD, Wiki, Snippets, and Settings. The 'Activity' item is highlighted with a light gray background.

2021 Panel (Right): This panel shows a more extensive and organized navigation structure. It includes a 'Project overview' header with a 'Details' section containing 'Activity' and 'Releases'. Below this, the navigation is organized into several categories: 'Repository' (containing 'Issues' with 4 items and 'Merge Requests' with 1 item), 'Requirements', 'CI/CD', 'Security & Compliance', 'Operations', 'Packages & Registries', 'Analytics', 'Wiki', 'Snippets', and 'Members'. The 'Settings' item is also present in this panel.

A large black arrow points from the 2018 panel to the 2021 panel, indicating the progression and growth of the navigation structure over time.

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.

A screenshot of a Dovetail project titled "GitLab Navigation/IA Research Synthesis". The project description states: "This Dovetail project contains insights from research studies that help us better understand the mental model of GitLab users. Understanding how users conceptualize our product helps us make meaningful improvements to navigation design and information architecture." The slide deck format is a Google Slides presentation. The project grid contains the following items:

Defining information architecture and navigation	Top challenges in GitLab's information architecture and navigation design	Wayfinding and top navigation insights
Left navigation insights	Measuring overall success at a high level	Appendix

Each item has a small profile picture and the date "13 Mar 2021".

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?