Community-first open source: An action plan!

PyCon US 2023 - Maintainer Summit

Hi!
I’m Pavithra :)

DEVELOPER ADVOCATE, QUANSIGHT

Co-presenter: Tania Allard

CO-DIRECTOR, QUANSIGHT LABS

What is community-first open source?

Company-backed

&

Community-driven

Community-driven:

Driven, developed, and governed
by the *community*.

Community comes first,
and for-profit entities are a part
of the broader community.

Community?

Enthusiasts

Contributors

Users

Advocates

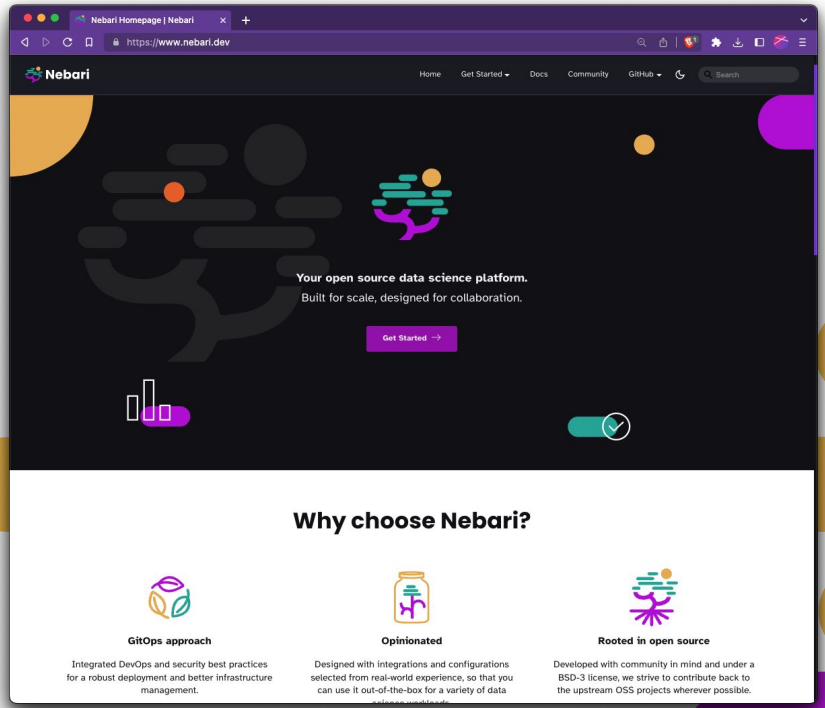
Maintainers & Sustainers

Long-term sustainability

Open+documented decisions

Improvements that benefit everyone

Investment in onboarding new contributors

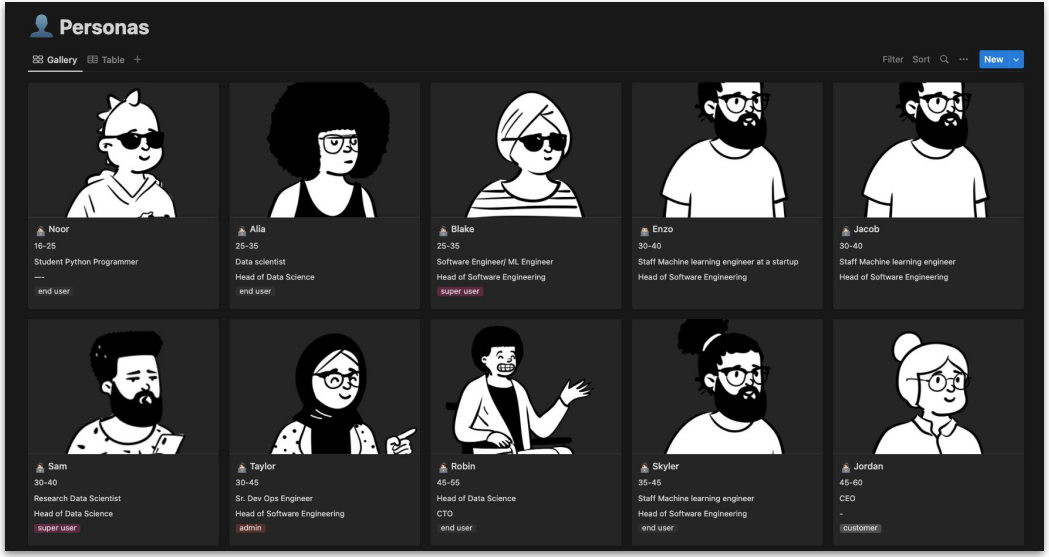


We're moving Nebari to a community-first governance

Step 1

Guiding principles

User personas & journeys



Core values

- Accessibility and inclusion
- Flexibility: Use OSS tools
- Explicit governance
- Contribute back upstream
- Security best practices
- Vendor agnostic dev

Vision board

Vision	All-in-one development and experimentation platform for Data Science teams. Open source platform for data science teams (1-100) to work efficiently and collaboratively. Reducing the deployment and integration friction.	What is your purpose for creating the product? Which positive change should it bring about?

Target group	Needs	Product	~~Business~~ Community goals
Users: - Data Scientists and ML Engineers - MLOps experts or practitioners - Software developers Maintenance and deployment: - infra and DevOps teams Customers: Data science teams between 1-100: - research - industry (startups)	Data Science teams: 1. Flexible data science environment 2. Open source tools and multiple integrations (workflow engines, data sources, orchestration, data stores, feature store. version control, visualization and dashboards) 3. High-quality and comprehensive documentation 4. Best practices: security, reproducibility, version control, collaboration DevOps/infra: 1. Reproducible deployments 2. Security-first features 3. Avoid vendor lock-in 4. Transparency regarding configuration, permissions, upgrades, dependencies and supply chain matters	1. Just works experience 2. Open source (built on and itself) 3. Build on the right to replicate for customer 4. Vendor agnostic → multi-cloud, GitHub, GitLab, infrastructure and configuration as a code	Make this a true community-driven open source project 1. Complementary to [redacted] 2. Drive engagement and trust within the open source community 3. Contribute to upstream projects and ecosystem 4. Have solid governance and sustainability objectives
Which market or market segment does this product address? Who are the target customers and users?	What problem does the product solve? Which benefit does it provide?	What product is it? What makes it stand out? Is it feasible to develop the product?	How is the product going to benefit the company? What are the business goals?

Step 2

Project management

1

Licensing

2

Code of conduct

3

Repository management

4

Communicat-ion channels

5

Governance

6

Messaging & branding

1

Licensing

Consider licenses for core, supporting, upstream, and downstream projects

2

Code of conduct

Policy

Enforcement strategy

Contributor Covenant

GitHub's Open Source Guide

SciPy's CoC

Turing Way's CoC (derived from Carpentries)

Mozilla's consequence ladder

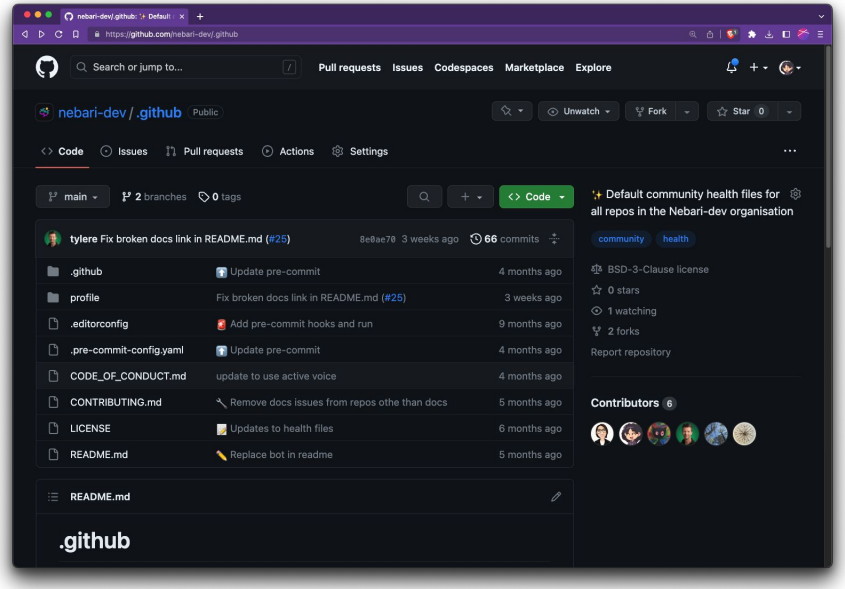
3

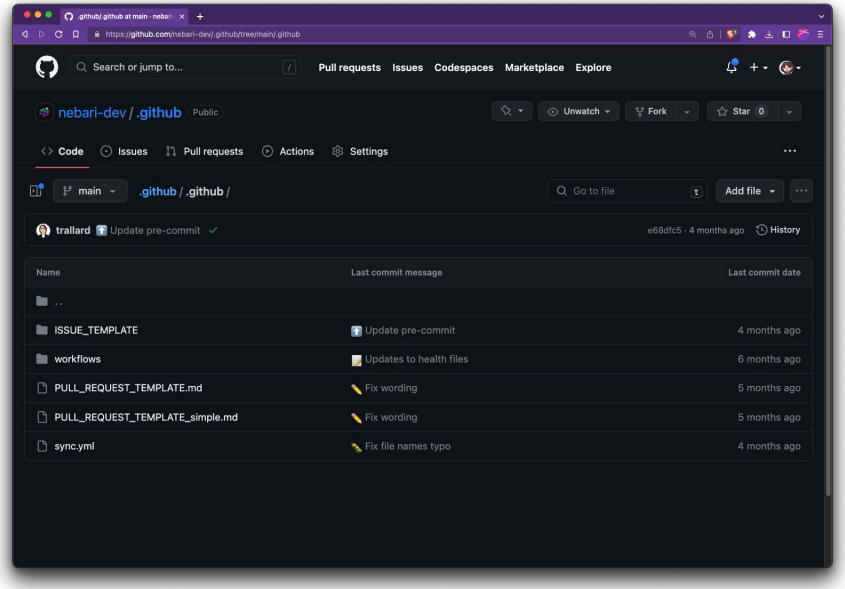
Repository management

Organization, repository, team structure

Issues / PRs – labels & templates

Nebari’s repository management





4

Communication channels

Support channels

Development chat

Meetings & notes

5

Governance

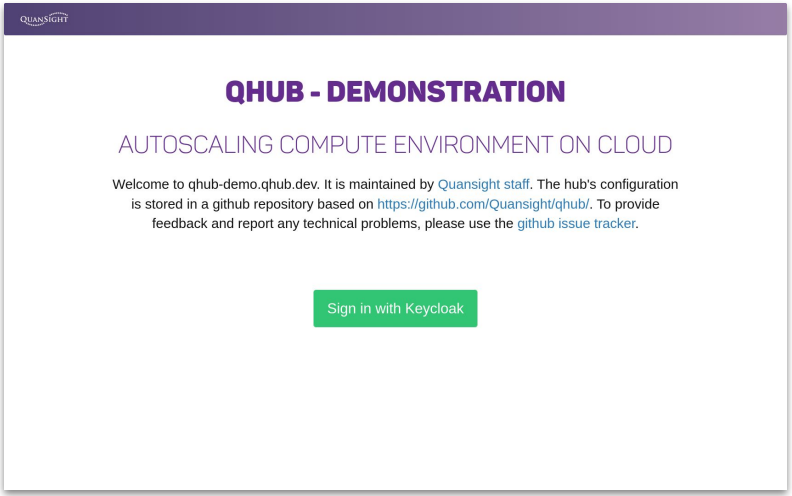
A process for making design, development, and management decisions

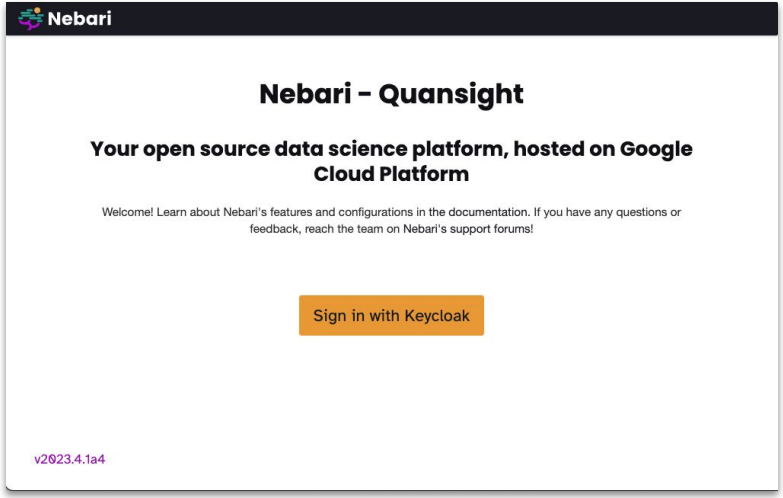
6

Messaging and branding

Alignment with new values & vision

Nebari's messaging and branding





Step 3

Enabling Contributions

1

Contributor guidelines & pathways

2

Maintainers guidelines & pathways

3

Architecture information

4

Roadmap

1

Contributor guidelines and paths

Guidelines for code and low/no-code contributions

Issues / PRs best practices, reply templates, etc.

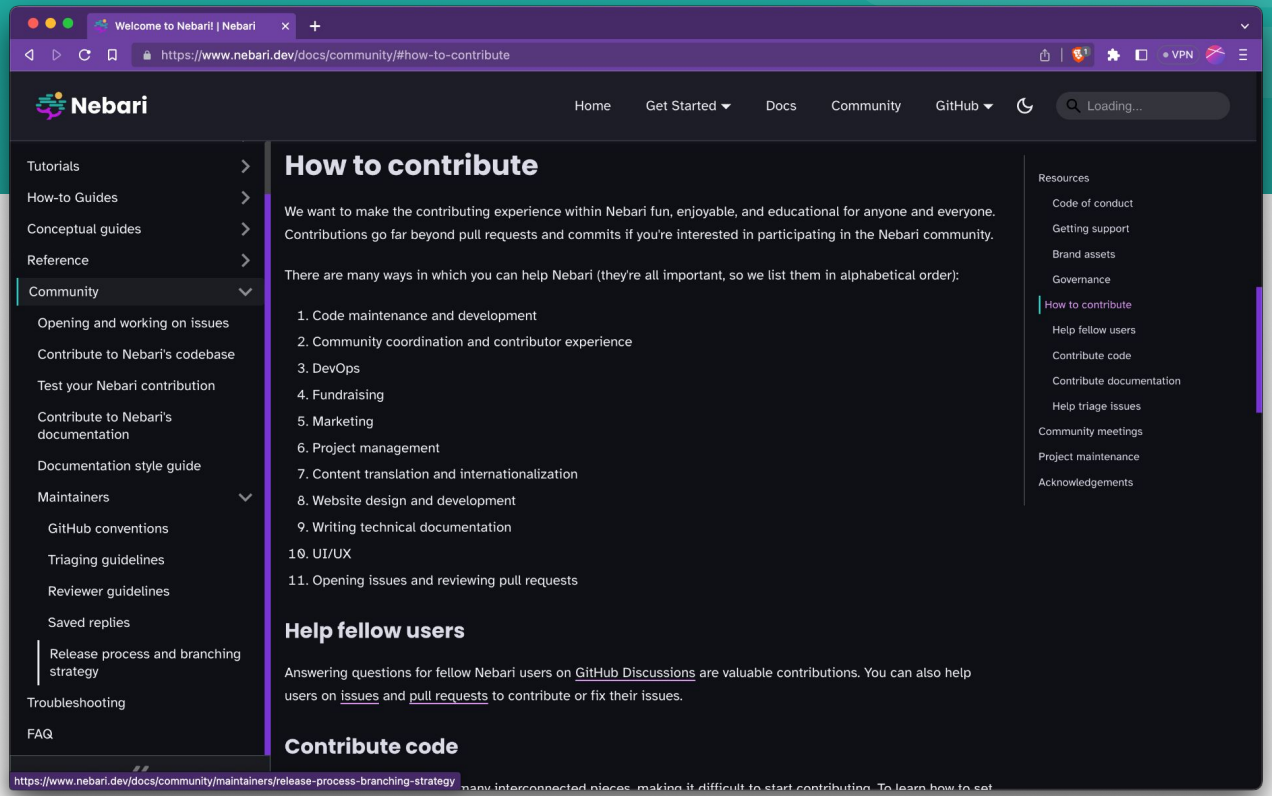
2

Maintainer guidelines and paths

Code review guidelines

Testing & debugging workflows

Contributor and maintainer docs



3

Architecture information

Underlying architecture diagrams

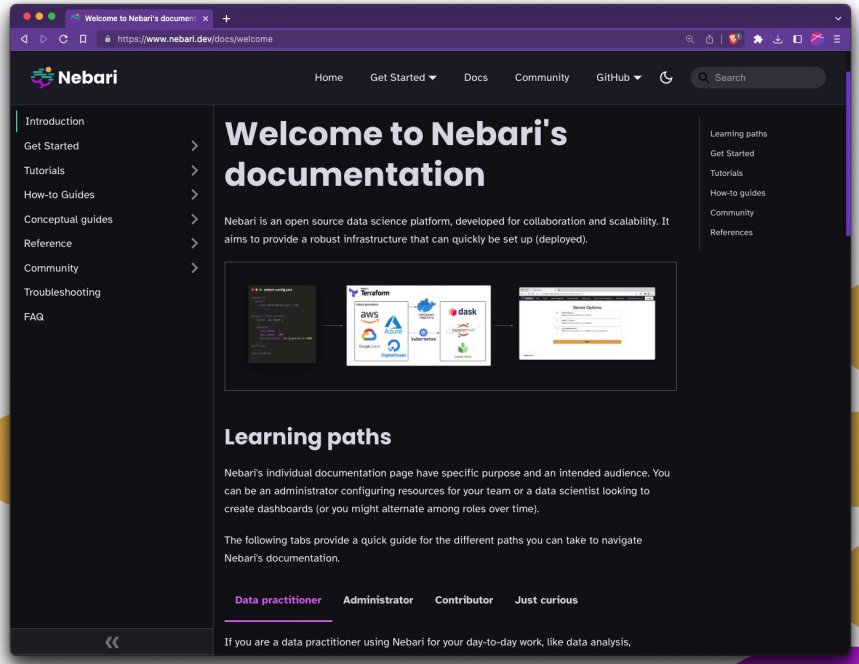
Release mechanism

4

Roadmap

Milestones

Priorities



Address *all* community members

> Docs style guide

> Framework (Diátaxis)

These will evolve as your community grows :)

Thank you