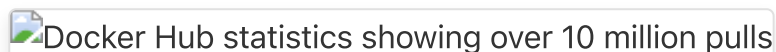




Building a Thriving Open Source Community

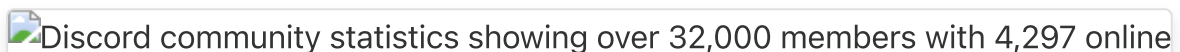
Category: Open Source / Date: 2019-Present

In 2019, I had the opportunity to take over a promising open source project called OhMyForm that was at risk of becoming abandoned. The original developer had created a solid foundation but was unable to continue maintaining it. Recognizing the project's potential and the community's need, I stepped in to not only maintain but significantly expand both the codebase and the community around it. After several years of successful development and growth, I made the decision to sunset OhMyForm in 2024 as the excellent [Formbricks](#) project had vastly exceeded my own ambitions and provided a superior alternative for the community.



Docker Hub statistics showing over 10 million pulls

The project's Docker Hub statistics showing over 10 million pulls



Discord community statistics showing over 32,000 members with 4,297 online

Our thriving Discord community with over 32,000 members

The Challenge

When I took over the project, it faced several critical challenges:

- A codebase that was functional but needed significant modernization
- Limited infrastructure for continuous integration, testing, and deployment
- A small but passionate community with no structured way to communicate or collaborate
- No sustainable funding model to support ongoing development and maintenance
- Technical debt that was hindering new feature development and stability

"The true measure of an open source project's success isn't just code quality or feature completeness—it's the health and engagement of the community that forms around it."

The Approach

I developed a comprehensive strategy that addressed both technical excellence and community building:

1. Technical Infrastructure

My first priority was establishing robust infrastructure to support sustainable development:

- Implemented a comprehensive CI/CD pipeline using GitHub Actions
- Created Docker containers for easy deployment, which have now been pulled over 10 million times
- Established automated testing with coverage requirements for all new code
- Developed clear documentation for both users and contributors
- Set up monitoring and observability tools to track usage and identify issues

2. Community Building

In parallel with technical improvements, I focused on creating a welcoming, active community:

- Established a Discord server that has grown to over 32,000 members, with typically 4,000+ active at any time
- Created structured channels for support, feature requests, showcase, and general discussion
- Implemented community guidelines and moderation systems to maintain a positive environment
- Organized regular community calls and update announcements
- Recognized and celebrated community contributions

3. Sustainable Funding

To ensure long-term viability, I established a sustainable funding model:

- Set up an OpenCollective account for transparent community funding
- Developed clear funding goals tied to specific development milestones
- Created a sponsorship program with appropriate recognition for corporate supporters
- Established a governance model for fund allocation
- Built several years of runway to ensure project stability

The Team

As the project grew, I recruited and mentored a team of over 10 developers who now contribute regularly:

- Established clear contribution guidelines and review processes
- Created an onboarding process for new contributors
- Implemented a mentorship system pairing experienced developers with newcomers
- Developed a governance structure that distributes decision-making authority
- Set up regular team meetings and coordination channels

Results & Impact

The project has achieved remarkable success under this new structure:

- Over 10 million Docker image pulls, demonstrating widespread adoption
- A thriving Discord community with 32,000+ members and 4,000+ regularly active users
- Sustainable funding through OpenCollective with several years of runway
- A stable team of 10+ regular contributors
- Significant expansion of features and capabilities
- Improved code quality, test coverage, and documentation
- Regular release cycles with clear roadmaps

Lessons Learned

This experience provided valuable insights into open source project management:

Technical Lessons

Infrastructure investments pay dividends. The early focus on CI/CD, containerization, and automated testing dramatically improved both code quality and contributor experience. By making it easy to deploy, test, and contribute, we lowered barriers to entry and increased the pace of innovation.

Additionally, the decision to containerize the application early proved prescient, as it significantly simplified deployment across diverse environments and contributed to the project's widespread adoption.

Community Lessons

Community building requires intentional design. By creating structured spaces for different types of interactions (support, development, showcase, etc.), we enabled

community members to engage in ways that matched their interests and expertise. The regular rhythm of updates and community calls helped maintain momentum and excitement.

Transparency in decision-making and fund management built trust with the community, which proved essential for sustainable growth and support.

Future Directions

In 2024, after careful consideration of the project's trajectory and the emergence of more comprehensive alternatives, I made the decision to sunset OhMyForm. The repository was officially archived on GitHub in October 2024. The excellent [Formbricks](#) project had vastly exceeded my own ambitions for OhMyForm and offered the community a superior solution with more features, better support, and a stronger development team.

This decision reflects an important principle in open source stewardship: recognizing when another project better serves the community's needs and being willing to step aside rather than fragmenting efforts. I encouraged our users to migrate to Formbricks and worked to ensure a smooth transition for the community.

The lessons learned from building and eventually sunsetting OhMyForm continue to inform my approach to open source development and community management:

- The importance of clear communication during project transitions
- How to responsibly wind down a project while supporting users
- When to collaborate with or defer to other projects in the ecosystem
- The value of focusing efforts where they can have the greatest impact
- How to transfer community knowledge and relationships during transitions

The success of this project demonstrates how technical excellence combined with intentional community building can transform an at-risk open source project into a thriving ecosystem that benefits thousands of users while providing sustainable opportunities for contributors. It also shows the importance of responsible stewardship throughout a project's entire lifecycle, including its conclusion.

Related Stories

FastAsyncWorldEdit & PlotSquared

The technical challenges overcome in scaling Minecraft world editing from crashing at 50,000 edits to seamlessly handling billions.

[Read Story](#)

Nitric Concepts Leadership

Managing a distributed team of 45 contractors and implementing DevSecOps practices across multiple timezones.

[Read Story](#)

Accessibility: This website is designed and developed to meet WCAG 2.1 Level AAA standards, ensuring the highest level of accessibility for all users. Features include high contrast ratios, keyboard navigation, screen reader compatibility, and responsive design. The site supports both light and dark modes with automatic system preference detection.