

# Renewal of OSS: Why, When and How to Rewrite

B310 Digital GmbH  
Jannik Streek

03.06.2025





# What is a Rewrite?

- Replacing (most of) a legacy system
- Following a potential grace period, the old system or codebase will be retired





**„[...] the single worst strategic mistake that any software company can make:**

**They decided to rewrite the code from scratch.“**

Joel Spolsky (JoelOnSoftware), 2000 [1]



# OSS vs. Closed Source

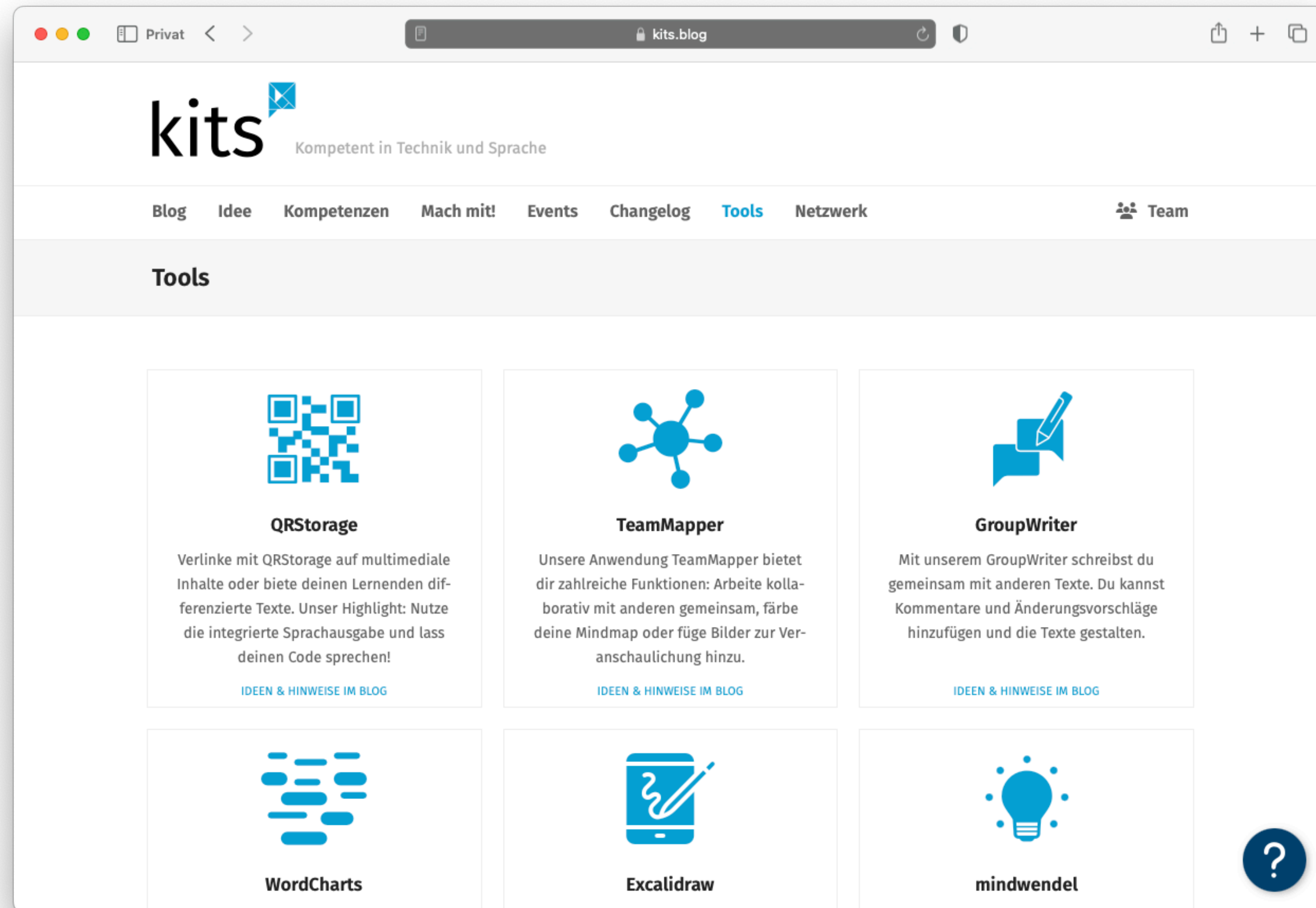
## Some differences:

- Maintainers might leave ... and you don't realize that they have left!
- Diverse opinions in project goals and no higher instance available to resolve issues



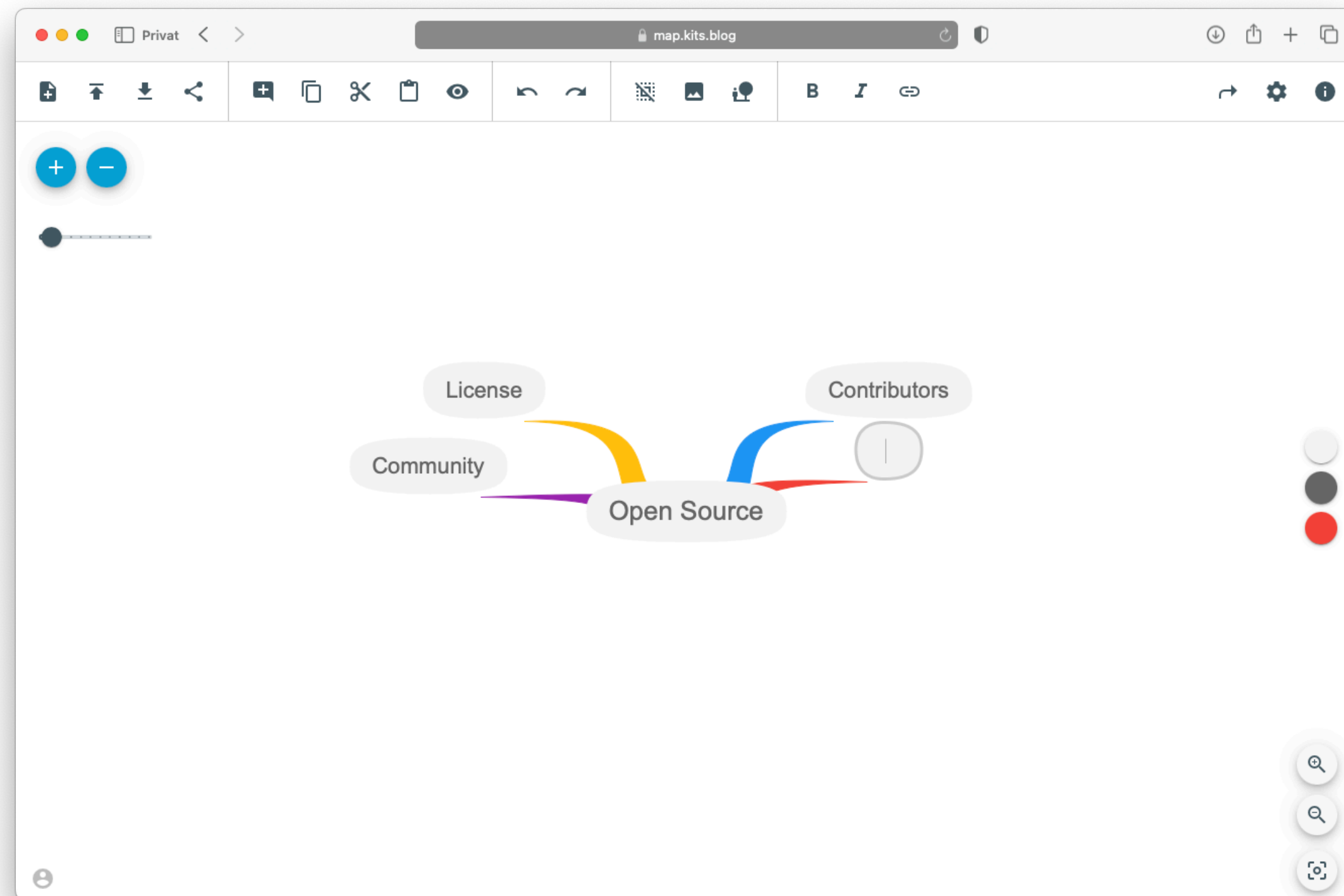


# About Us: Maintaining and Hosting OSS Apps for a State Institute in Lower Saxony



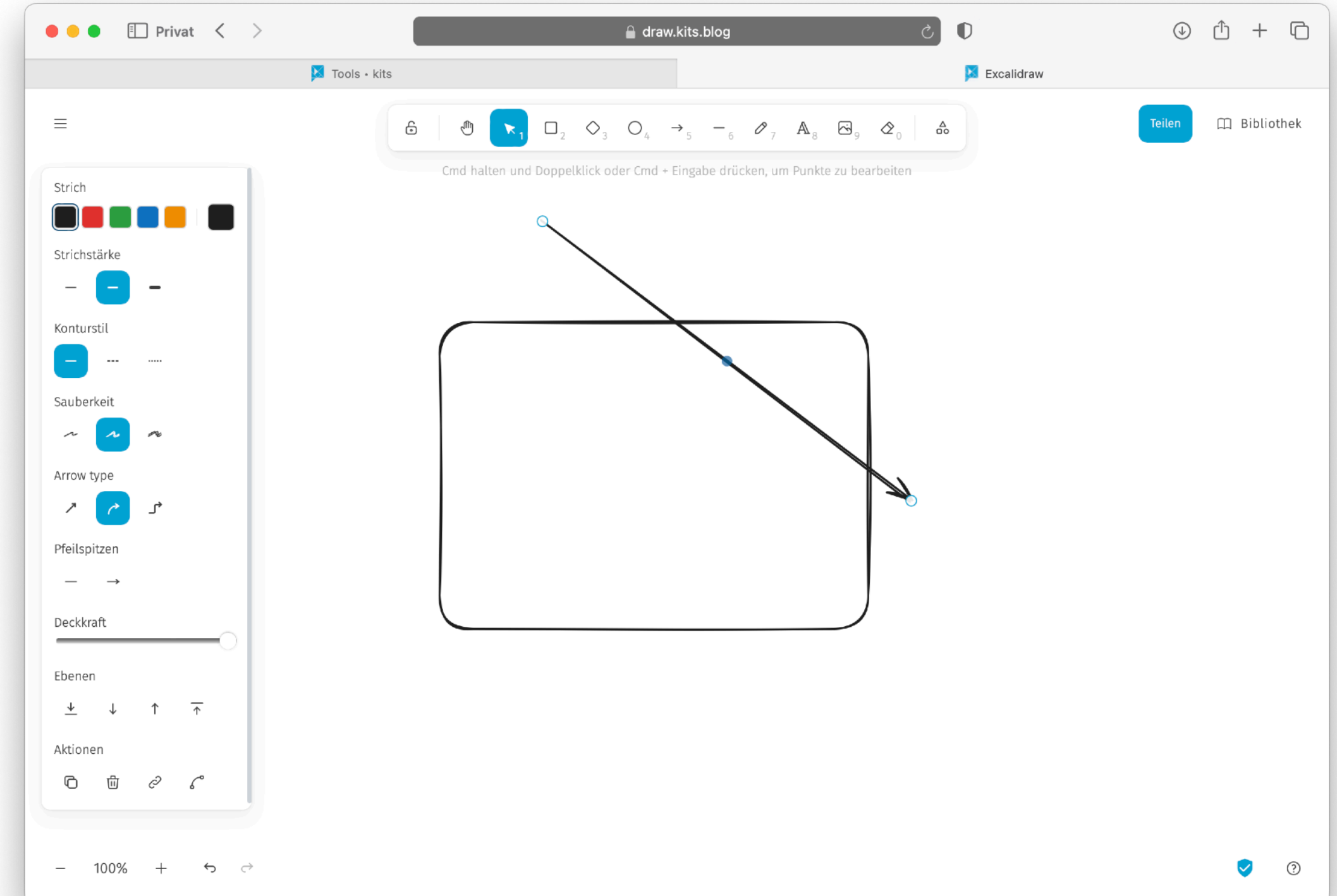


# Apps: Mixture of Originals and Forks



## TeamMapper:

Extended the unmaintained project „Mindmapp“ with collaborative features



## Excalidraw: Fork without Firebase



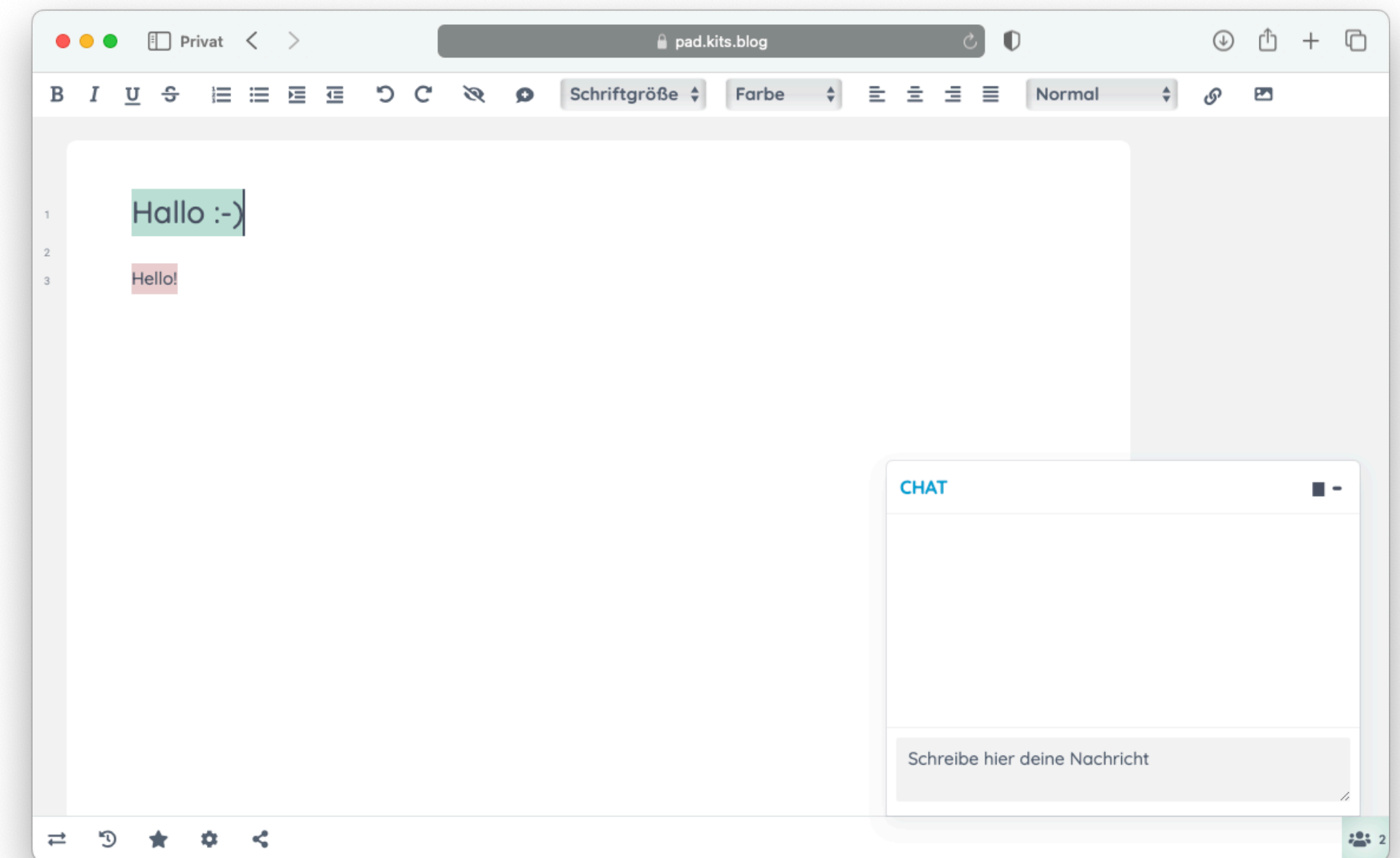
# Case Study: Etherpad (Lite)

## History [2]

- 2008: Etherpad Release (Scala, Java, Javascript)
- 2009: Acquired by Google and open sourced shortly after
- 2011: Official Release of Etherpad Lite, a rewrite in Javascript

## Technology of Etherpad Lite

- Rewrite in Node.js
- Mixture of versioned libraries and npm packages
- Synchronization: Changeset Library [3] (easysync protocol), based on OT
- In this presentation referred to as „Etherpad“





# 01

## Why





# Active Maintenance

„Only 11% of OSS projects are actively maintained“

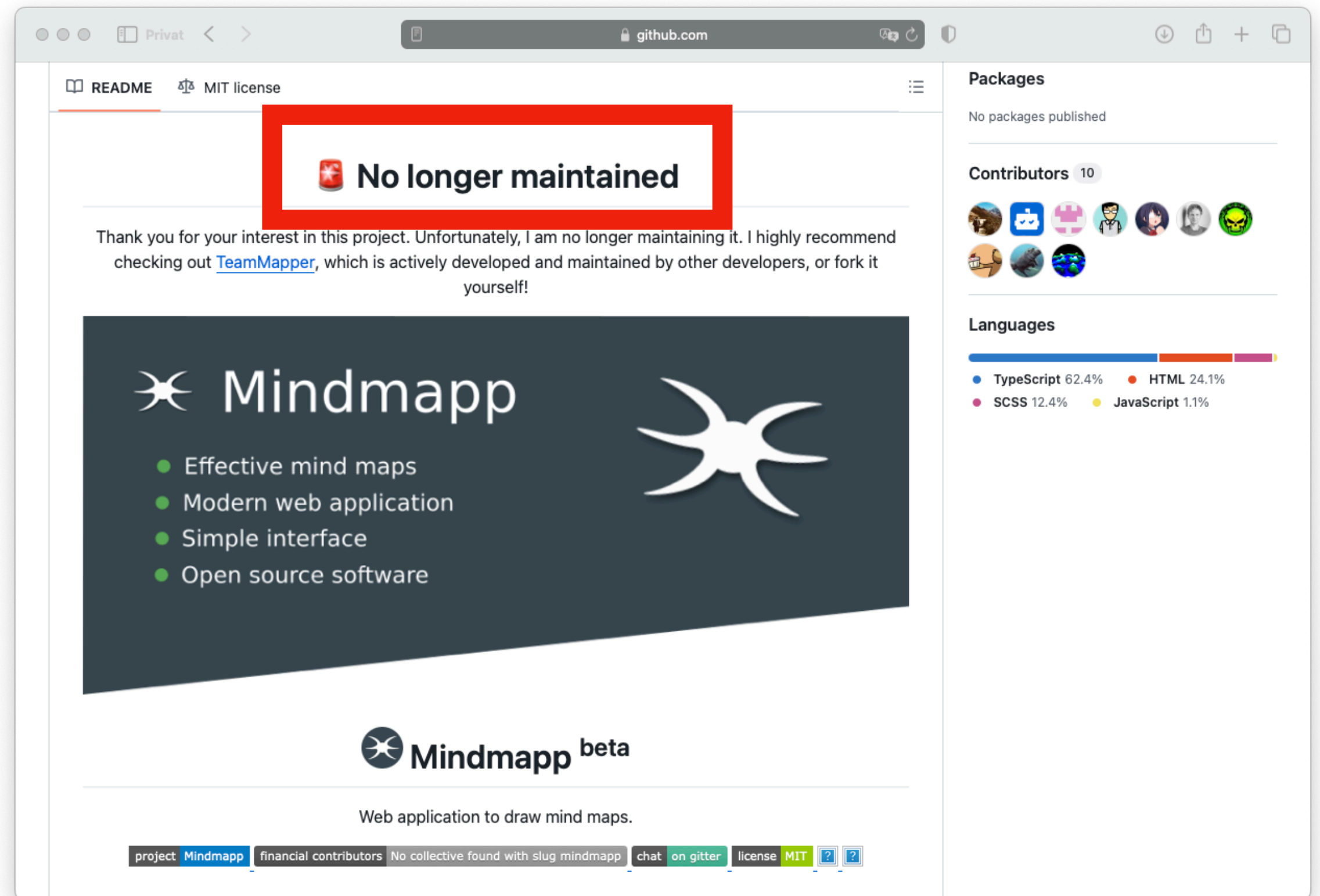
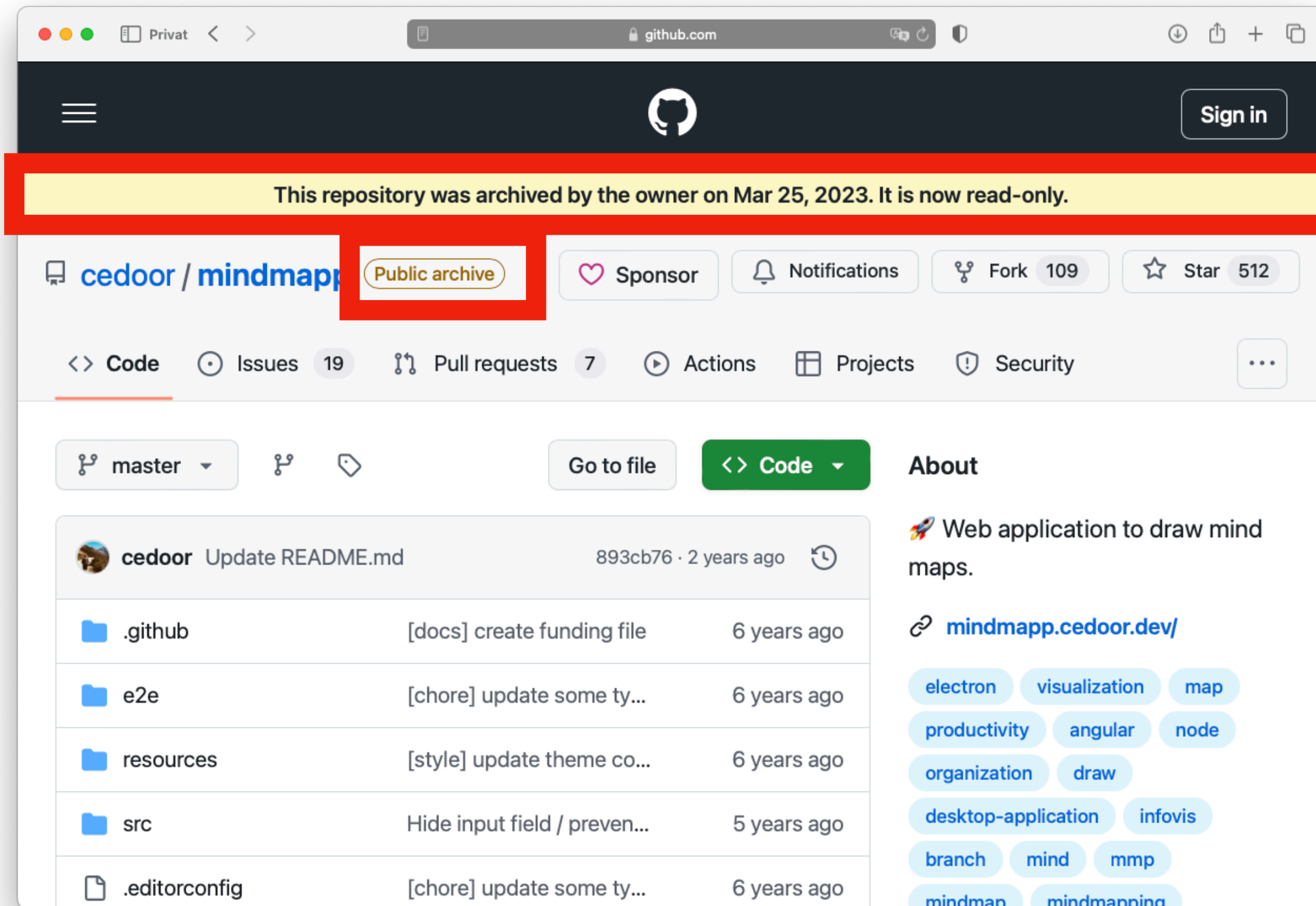
9th Annual State of the Software Supply Chain, sonatype, 2023 [4]

Understanding + upgrading unmaintained OSS might be challenging

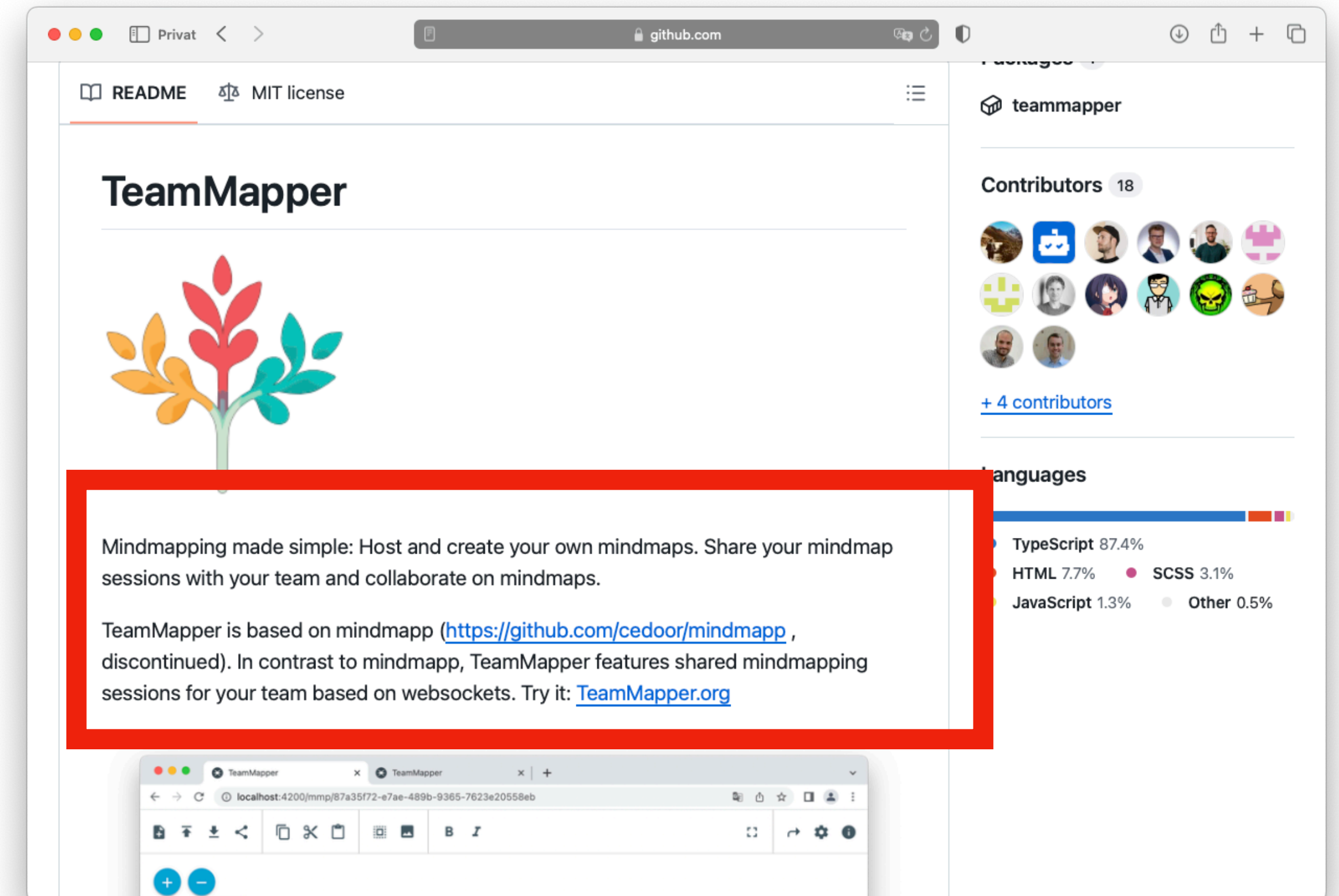
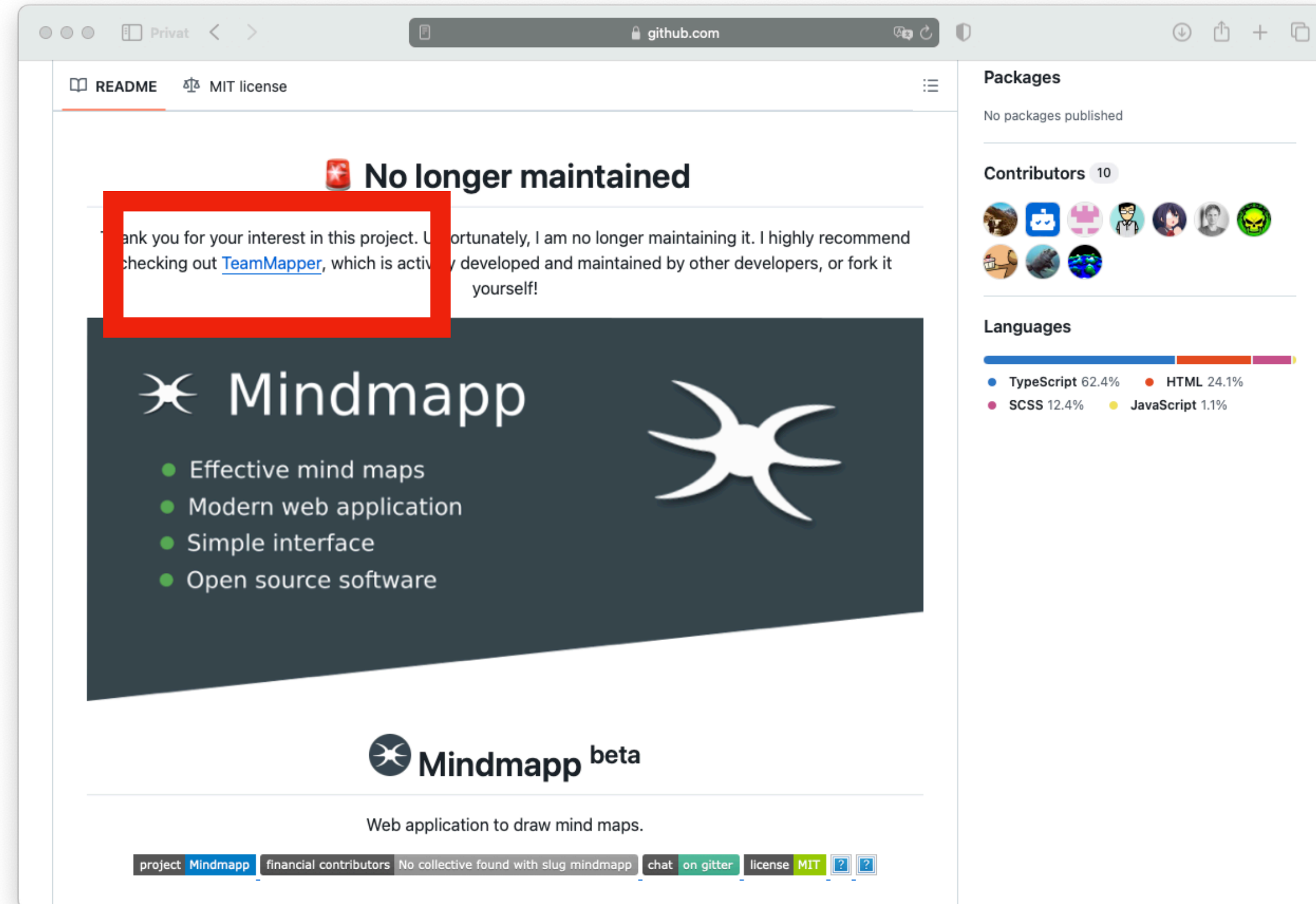




# Learning 1: Unmaintained? Make It Transparent!



# Learning 2: Fork It & Tell the Old Maintainer

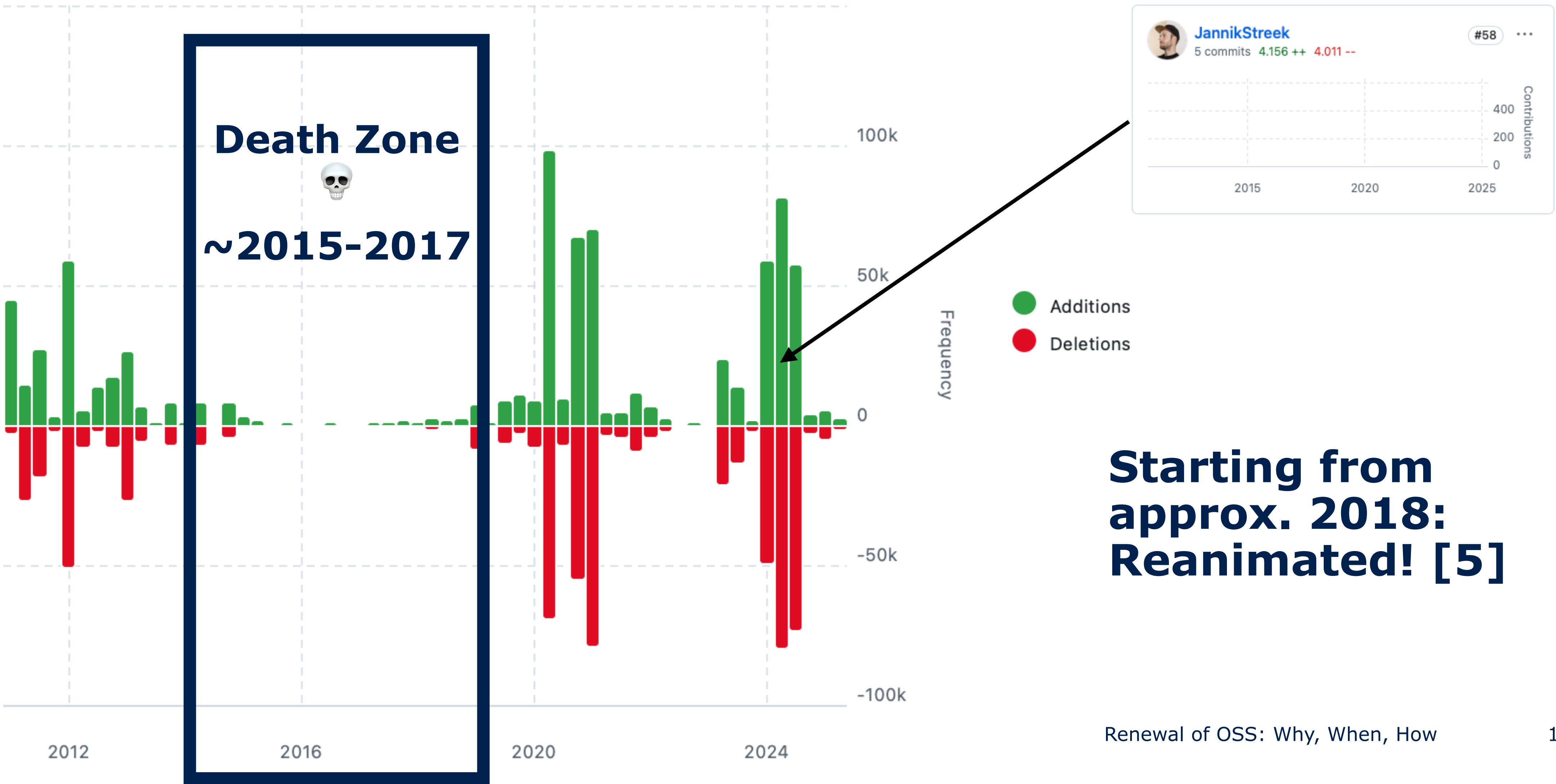




# Case Study Etherpad: Maintenance

## Code frequency

Additions and deletions per week





# Diverging Project Goals

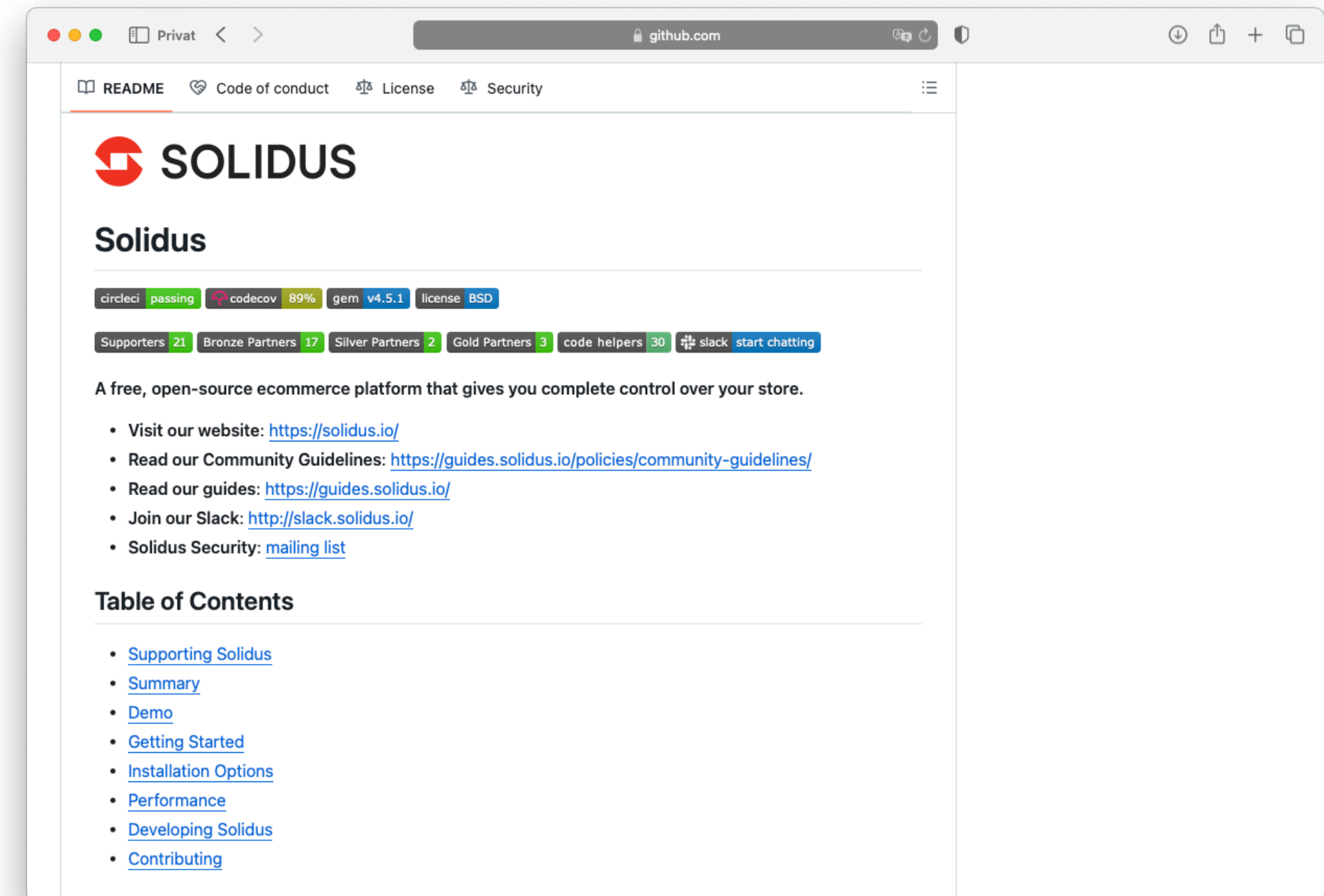
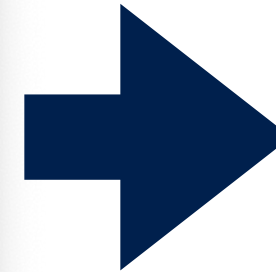
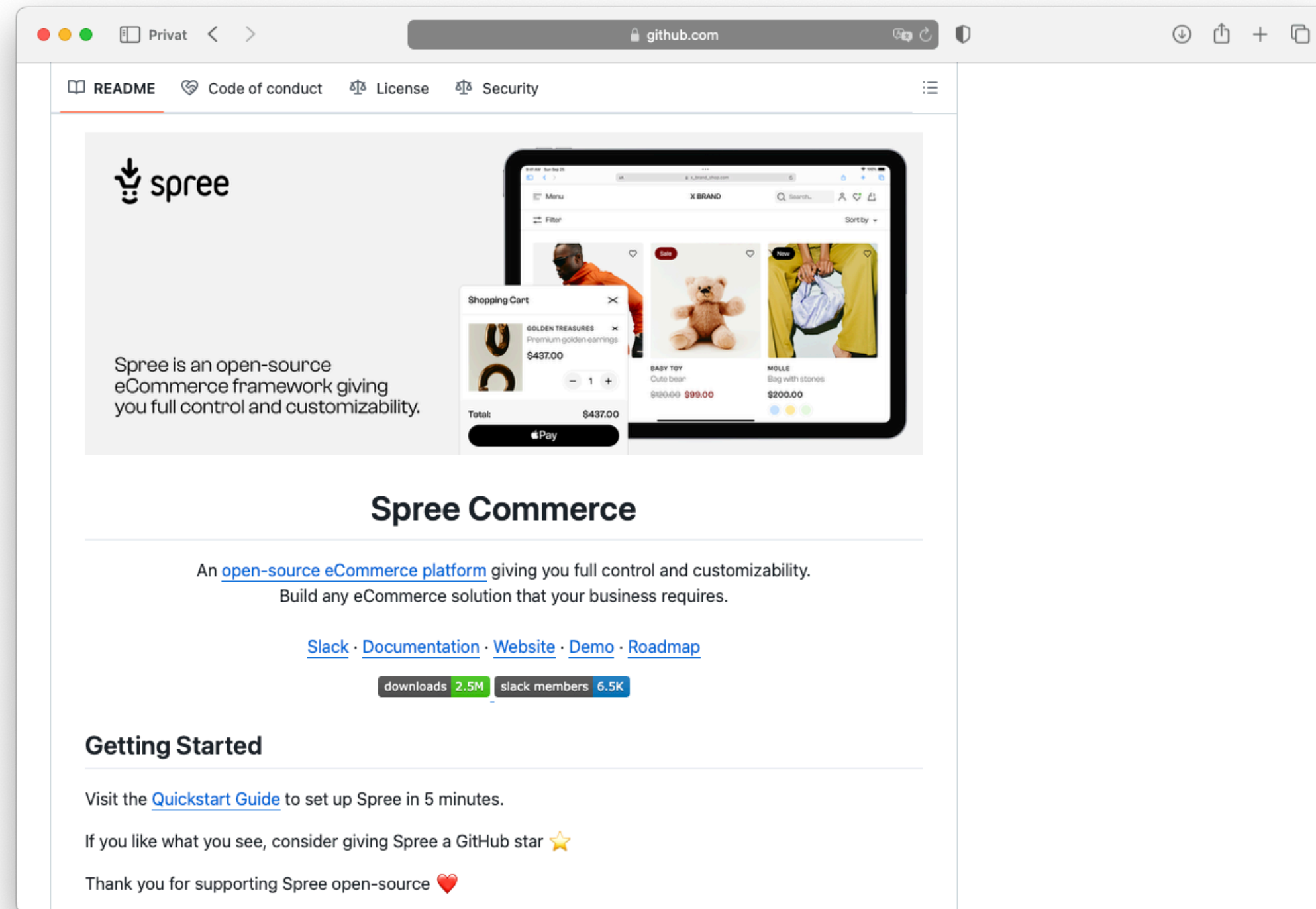
Pain to remove new features > gain of new improvements

Unstable releases and backward compatibility



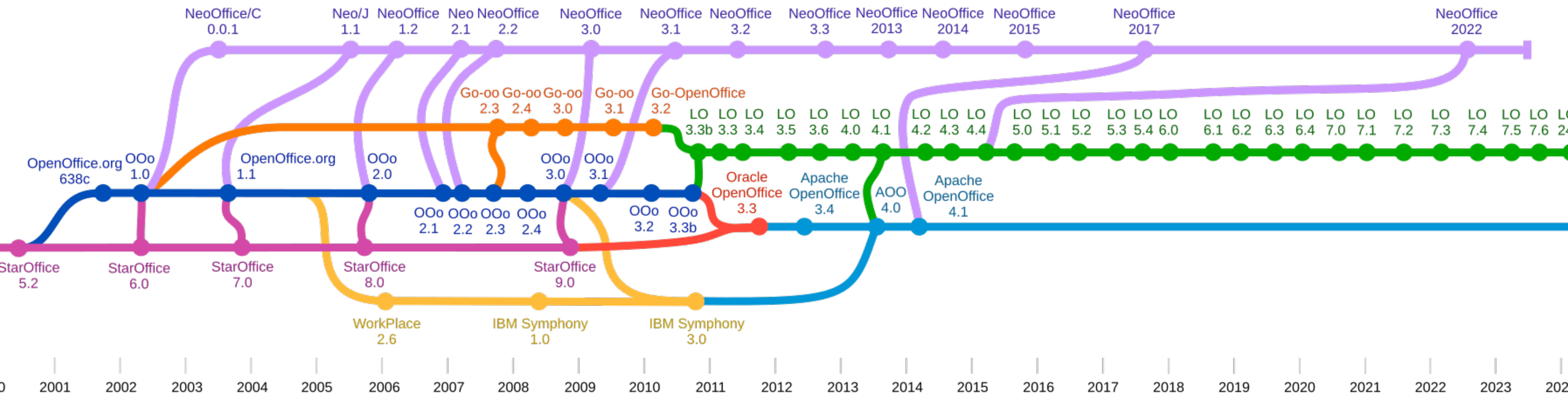


# Learning 3: Consider Joining or Forking Before



"Spree Commerce [...] has announced that it will stop investing in the open source project and that there currently is no core team. We [...] forked Spree and called it Solidus." [6]

# Learning 4: Be Aware of Fork Complexity



## StarOffice Derivatives [7]



# Case Study Etherpad: Project Goals

- At first: Fork with minor changes, still easy to consolidate
- Joined the effort, e.g. docker setup
- Plugin-Mechanism in Etherpad used for additional features
- More and more work required to remove unwanted new features



# Refactoring & Modernization

New technologies can make parts of older projects obsolete

Changing best practices, dead code, bad code, ...

License!





# Case Study Etherpad: Refactoring & Modernization

- Ongoing efforts to upgrade the project, e.g. pnpm, docker, TypeScript ...
- A lot of unmaintained and undocumented legacy code



# Performance, Scalability and Security

Identifying and resolving  
performance / scalability  
issues, aligned with business  
needs [8]

Debugging





# Learning 5: Making Clever Use of Technology Improvements

**Porting can be an alternative to rewriting!**



**A 10x Faster TypeScript [9]**

# Case Study Etherpad: Performance, Scalability and Security

- Poor performance
- Crashes
- Security concerns (admin area, plugin mechanism, binary file import)
- Hard to debug



# 02

## When





# Reality Check

Get in touch with the maintainers

Research project alternatives



# Case Study Etherpad: Reality Check

- No better alternatives found during the time of research
- Friendly maintainers, discussions around how to cooperate
- Problem: Different viewpoints on scope
- Fear of user confusion with a differently scoped app under the same name (based on experience with old Etherpad Lite rewrite)



# Feasibility

Complexity of the project

Existing user-base





# Learning 6: Be Realistic About the Effort When You Want to Rewrite

- Netscape Example [10]
  - Release of Netscape 4, June 1997
  - Release of Netscape 5, Abandoned
  - Release of Netscape 6, November 2000 (Rewrite from scratch)



# Case Study Etherpad: Feasibility

- Rewriting a real-time text merging algorithm from scratch out of time and budget
- But: Availability of new technologies (algorithms, libraries, ...)
- Control over our existing user base



# 03

## How





# Scope Management

Clear objective and goals

Minimum Viable Product  
Mindset vs. Scope Creep





# Case Study Etherpad: Scope Management

- Clear idea about the scope
- Features that were added via plugins can now be baked in
- Superfluous features can be removed (e.g. chat)
- Side note: Etherpad Lite reduced scope, but added it back over time



# Research

Technology that provides a smarter, more elegant solution

Modern programming stacks better suited for the problem





# Case Study Etherpad: Research

- Researched possible technologies
  - Y.js [11] for synchronizing data structures (e.g. text), based on CRDT
  - ProseMirror [12] / TipTap Editor [13]
  - Hocuspocus Server [14]
- Result: Integrating & customizing existing OSS tools as replacement



# Follow Engineering Best Practices

Breaking up the problem  
[16]

Gradual process of replacement „Strangler Fig“ [16]

Usage of patterns, e.g.  
Transitional Architecture  
[15]

...





## Learning 7: Be Aware of the Challenges a Rewrite Poses

- Expensive, often takes a long time
- Undocumented requirements, unclear need of legacy features
- Feature freeze of an old system might be a strategic mistake in case of delay or being stuck



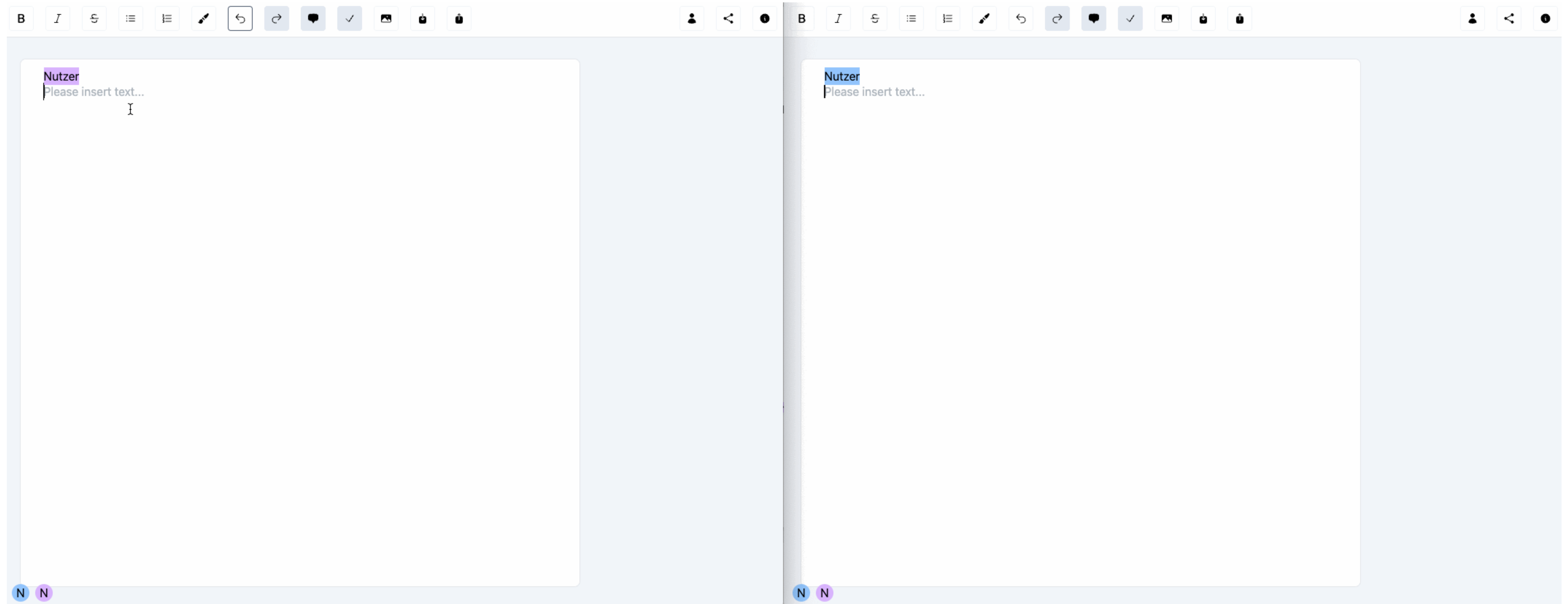
# 04

## GroupWriter





# GroupWriter



**write.kits.blog**

[github.com/b310-digital/groupwriter-frontend](https://github.com/b310-digital/groupwriter-frontend)

Renewal of OSS: Why, When, How

36

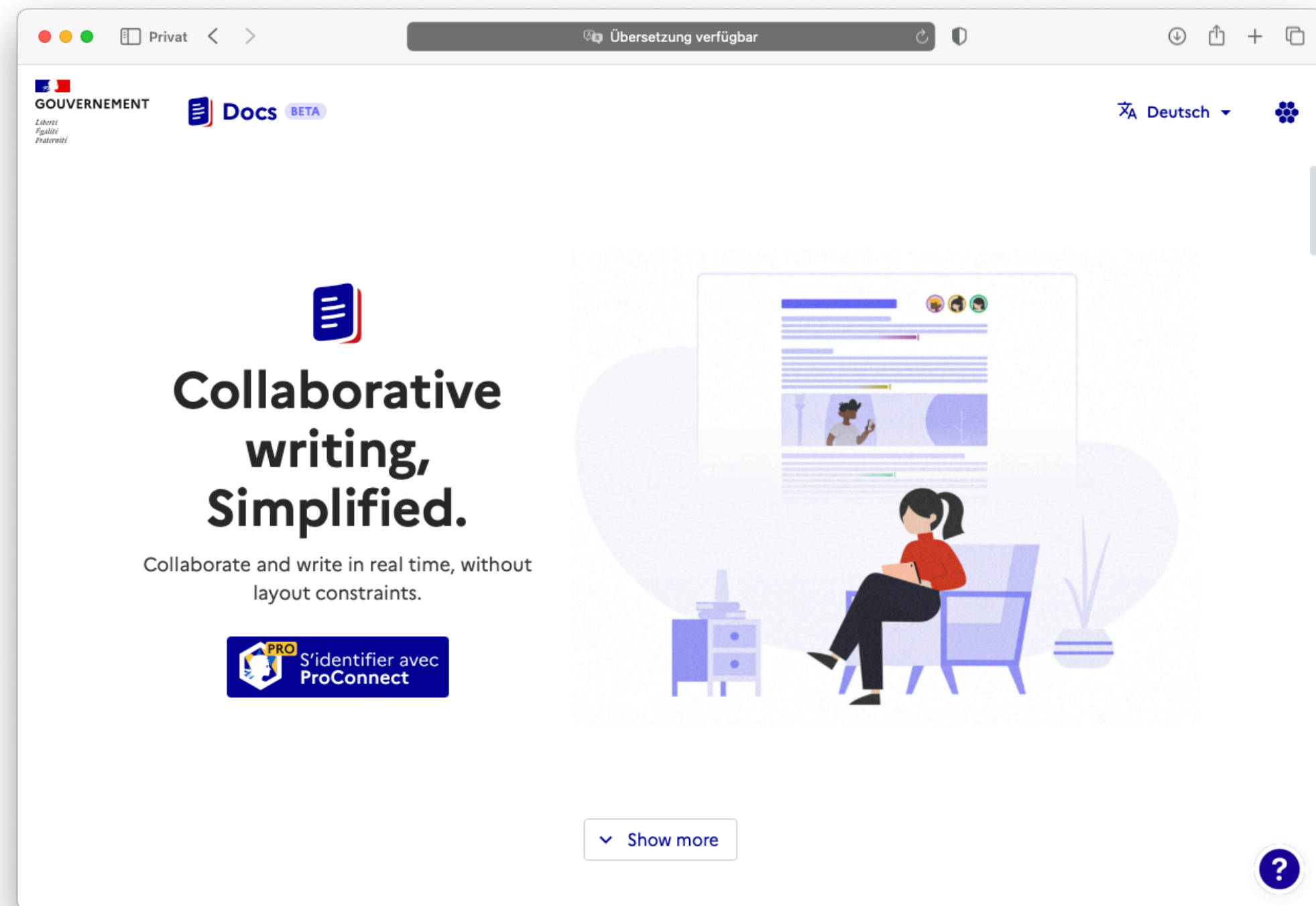


# A Full Rewrite?

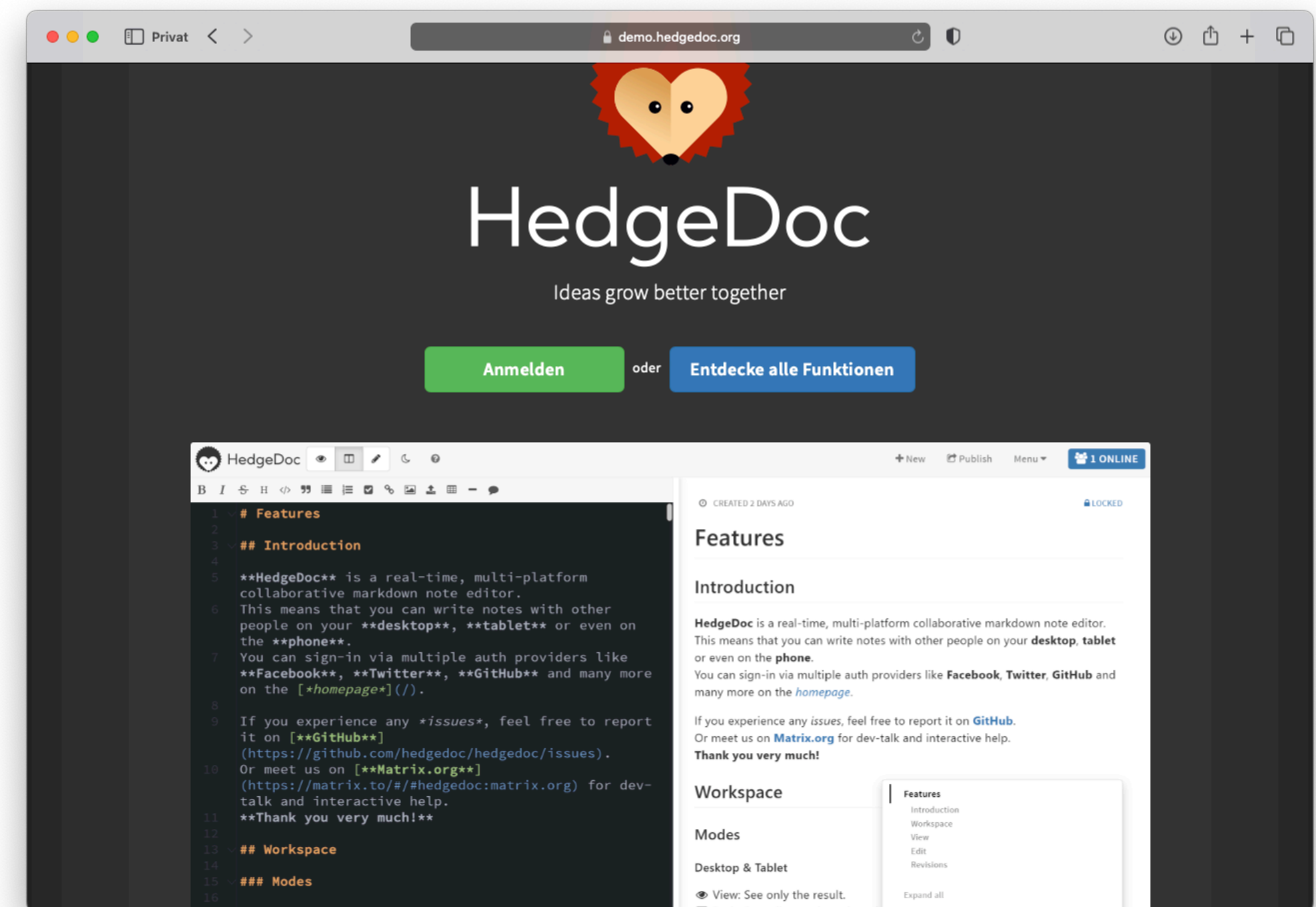
- Retirement of our own Etherpad Lite fork, currently in a grace period
- Feasible effort, mainly integration and customization
- New features: Comments and encrypted image upload
- However: Lost potential to achieve more „good“ for the Etherpad Community by contributing to the project



# Alternatives



Numerique [17]



HedgeDoc [18]



# Food for Thought

**„We’re programmers. Programmers are, in their hearts, architects, and the first thing they want to do when they get to a site is to bulldoze the place flat and build something grand.“**


Joel Spolsky (JoelOnSoftware), 2000 [1]




# Thank you.

**Jannik Streek**

Full-Stack Software Engineer  
B310 Digital - [www.b310.de](http://www.b310.de)

 @jstreek.bsky.social

 [jannik@b310.de](mailto:jannik@b310.de)

 [github.com/JannikStreek](https://github.com/JannikStreek)





# Sources

- [1] = Things You Should Never Do, Part 1, Accessed at 22nd of May, <https://www.joelonsoftware.com/2000/04/06/things-you-should-never-do-part-i/>
- [2] = History of Etherpad, Accessed at 22nd of May 2025, <https://en.wikipedia.org/wiki/Etherpad>
- [3] = Changeset library, Accessed at 22nd of May 2025, <https://github.com/ether/etherpad-lite/blob/develop/src/static/js/Changeset.ts>
- [4] = Sonatype 9th Annual Report (2023), Accessed at 22nd of May 2025, <https://www.sonatype.com/hubfs/9th-Annual-SSSC-Report.pdf>
- [5] = Code Frequency for the Etherpad Project, Accessed at 22nd of May 2025, <https://github.com/ether/etherpad-lite/graphs/code-frequency>
- [6] = Future of Spree, Accessed at 23rd of May 2025, <https://solidus.io/blog/future-of-spree>
- [7] = History of StarOffice derivatives, Accessed at 22nd of May 2025, [https://de.wikipedia.org/wiki/Apache\\_OpenOffice#/media/Datei:StarOffice\\_major\\_derivatives.svg](https://de.wikipedia.org/wiki/Apache_OpenOffice#/media/Datei:StarOffice_major_derivatives.svg)
- [8] = Legacy Displacement, Accessed at 22nd of May, <https://martinfowler.com/articles/patterns-legacy-displacement/>
- [9] = A 10x Faster TypeScript, Accessed at 22nd of May 2025, <https://devblogs.microsoft.com/typescript/typescript-native-port/>
- [10] = Netscape, Accessed at 22nd of May 2025, [https://en.wikipedia.org/wiki/Netscape\\_\(web\\_browser\)#Netscape\\_Communicator\\_5.0\\_\(canceled\)](https://en.wikipedia.org/wiki/Netscape_(web_browser)#Netscape_Communicator_5.0_(canceled))
- [11] = Yjs, Accessed at 22nd of May 2025, <https://yjs.dev/>
- [12] = ProseMirror, Accessed at 22nd of May 2025, <https://prosemirror.net/>
- [13] = TipTap, Accessed at 22nd of May 2025, <https://tiptap.dev/>
- [14] = Hocuspocus, Accessed at 22nd of May 2025, <https://tiptap.dev/docs/hocuspocus/introduction>
- [15] = Legacy Displacement, Accessed at 22nd of May, <https://martinfowler.com/articles/patterns-legacy-displacement/>
- [16] = StranglerFigApplication, Accessed at 22nd of May, <https://martinfowler.com/bliki/StranglerFigApplication.html>
- [17] = Numerique, Accessed at 22nd of May, <https://docs.numerique.gouv.fr/>
- [18] = Hedgedoc, Accessed at 22nd of May, <https://demo.hedgedoc.org/>