A Report on Burnout in Open Source Software Communities

Miranda Heath

18 Nov 2025

Summary

This report draws on a review of the academic literature and a qualitative analysis of OSS community discussion to explore the phenomenon of burnout in the open source software (OSS) community and what can be done to effectively address it.

- In section 1, I explain what burnout is, in line with the psychological literature.
- In section 2, I describe the methodology used to conduct the research that informed this report.
- In section 3, I examine evidence of burnout among OSS developers, concluding burnout is a problem in OSS.
- In section 4, I identify 6 causes of burnout among OSS developers: difficulty getting paid, workload and time commitment, maintenance work as unrewarding, toxic community behaviour, hyper-responsibility and pressure to prove oneself.
- In section 5, I outline 4 recommendations for how to reduce burnout in OSS: pay OSS developers, foster a culture of recognition and respect, grow the community and advocate for maintainers.

I conclude by emphasising the harm that failing to address burnout in OSS poses, both to OSS developers, and to our entire software ecosystem that heavily relies on OSS.

Acknowledgements

I would like to thank Tim Bannister, Swastik Baranwal, Josh Goldberg, Lars Kappert, Mike McQuaid, Joshua Mo and Sergey Shandar for taking the time to contribute their thoughts and experiences regarding burnout in OSS. Their valuable insights as members of the OSS community informed the final draft of this report. Many thanks to Michael Selvidge, Greg Kumparak and Chad Whitacre for helping me share my findings with others, and to Vlad-Stefan Harbuz for diligent editorial support. This research was funded by Sentry as part of their Open Source Pledge initiative, to whom I am very grateful.

1. What is Burnout?

Burnout is a syndrome typically associated with work. It has three components united by one common thread: an exhaustion of physical and mental energy (Maslach et al., 2001; Schaufeli et al., 2023).

1. **Motivational component:** an intolerance of effort. Burnout is associated with fatigue and an inability to motivate oneself to attend to and act on things that need doing. This makes it difficult to engage with and effectively manage work.
2. **Affective component:** feeling emotionally drained and a reduced ability to regulate emotions—a person with burnout may be easily frustrated, irritated or emotionally overwhelmed.
3. **Cognitive component:** a change in how one thinks about work. For example, mentally distancing oneself from it and expressing negativity and cynicism towards it.

Burnout is harmful, and can negatively impact both mental and physical health (Graziotin et al., 2017; Maslach et al., 2001; Singh and Suar, 2013).

Some working conditions are more likely to lead to burnout than others. The risk of burnout is thought to increase with **job demandingness** (e.g, excessive workload, difficulty of work and conflict) and decrease with **job resources** (e.g., support from colleagues or bosses, fair pay, opportunities for learning and development, rewarding work and autonomy over how work gets done) (Demerouti et al., 2001).

2. Method

To inform and guide this report, I undertook three pieces of original research.

1. **Rapid Literature Review:** I conducted a rapid review of the academic literature on burnout among software developers. I did not limit this to OSS developers, as there were few articles looking at burnout in OSS specifically. To find the relevant literature, I conducted a keywords database search and checked the reference lists of discovered materials. I used the keywords *'open source, software, developer, maintainer, engineer, burnout'*, and searched in the databases PsychInfo and ExLibris DiscoverEd. This returned 30 articles. A full list of articles discovered in the rapid literature review is presented in appendix A.
2. **Rapid Thematic Analysis:** To supplement the lack of research on burnout in OSS developers, I also conducted a rapid thematic analysis of non academic materials

discussing burnout in OSS, created by people within the OSS community, reporting on the OSS community, or with an interest in OSS. I searched for these by using a keywords search, following links within discovered materials, and following recommendations by open source community members. As in the rapid literature review, I used the keywords '*open source, software, developer, maintainer, engineer, burnout*'. The search was conducted using Google. After 57 items, I judged that I had achieved adequate data saturation (i.e., it seemed no new themes were emerging) and stopped searching. The breakdown of materials by type was as follows:

Material Type	**Quantity**
Original articles by OSS developers	27
Conference talks by OSS developers	5
Video and podcast interviews with OSS developers	3
Articles published by OSS advocacy bodies (e.g., Tidelift)	9
Articles published by tech news sites	4
Forum and email discussion by OSS community members	4
Guides for OSS developers	2
Reports on burnout in OSS	3

Thematic analysis entails a close reading of materials to identify common themes and subthemes. The causes of burnout identified in section 4 are informed by the themes that emerged from the rapid thematic analysis. Illustrative quotes pertaining to each theme and insights from the rapid review of the academic literature are interwoven throughout. The full table of themes and subthemes with illustrative quotes, plus a list of all the materials included in the rapid thematic analysis, sorted by type, is presented in appendix B.

3. **Community Consultation:** Finally, to check my findings and conclusions were broadly in line with the experiences of the OSS community, I conducted a community consultation. I shared and promoted a first draft of the report among OSS developer networks on Bluesky, Mastodon and the Open Source Pledge Slack channel, alongside a request for feedback. Respondents from the OSS community indicated that my report was a good summary of the issue of burnout in OSS. I had in-depth conversations with 7 respondents to discuss their perspectives on OSS developer burnout. This included developers who had experienced burnout, developers who managed to avoid burnout, developers who actively maintain hugely popular projects, devel-

opers who had decided to step back from OSS, and developers seen as leaders in the OSS community. Conversations were conducted by video call, email, in person and on social media. Where new themes, or important additions to existing themes emerged, I incorporated these into the final draft of this report.

3. Evidence of Burnout in OSS

Burnout is a significant problem in the OSS community.

While there is a lack of data on the rate of burnout in OSS specifically, a 2023 comprehensive survey of 26,348 developers around the world working across both closed and open source found that 73% had experienced burnout at some point in their career (JetBrains, 2023). Burnout is predictive of quitting, and a 2024 survey of over 400 open source maintainers found that 60% had considered leaving open source (Tidelift, 2024). Taken together, this indicates burnout may be widespread in open source.

In my analysis of OSS community discussion, it was clear that OSS developers were experiencing the three components of burnout. Motivation was affected, with OSS developers describing a loss of joy in coding, avoiding work and even eventually quitting OSS entirely.

> *'I don't feel like working on [it] anymore. It went from being one of the most fun experiences in my life to making me feel terrible everyday.'*—Kyle, n.d.

OSS developers showed a change in affect, describing a shift from love for the OSS community to anger and ruder and more perfunctory responses to OSS users and contributors. Feelings of guilt, low self-worth and depressed mood were also cited.

> *'I'm tired and I'm angry, and I've done a reality check and found reality wanting.'*—MacIver, 2015

OSS developers exhibited the cognitive component of burnout. They described how they had become negative and cynical about open source. There was a culture of using dark humour as a means of coping (e.g., the *'three F's of open source: fix it, fork it, fuck off'*). Some also had a sense of directionlessness and loss of meaning in their OSS work.

> *'At a certain point I had to change what my value system was to accommodate my increasing disconnect from the work that I was doing and I somehow made it about like, "if I can just do this more and make it better this will help me get a higher paying job and then I'll have more money and then I can retire and not work anymore" ... I can't even really summon why I thought that that was a good plan but it seemed really good at the time.'*—Kaplan-Moss, 2015

It also appeared that burnout was causing real harm, impacting developers' physical and mental health.

> *'Open source was about sharing the code with fellow developers, learning new skills, and having fun. Somehow, it became for many a threat to their mental health, and an unpaid job.'*—Sapegin, 2023a

4. Causes of Burnout in OSS

From my rapid review of the literature, analysis of OSS community resources and community consultation, I identified 6 factors that play a critical role in burnout among OSS developers:

1. Difficulty getting paid
2. Workload and time commitment
3. Maintenance work as unrewarding
4. Toxic community behaviour
5. Hyper-responsibility
6. Pressure to prove oneself

Troublingly, these factors also appear to be inter-related and mutually reinforcing. For example, difficulty getting paid for OSS can mean having to work a concurrent full-time job, thus increasing one's workload. Toxic community behvaiour makes maintenance work less rewarding, and take more time. In the next section, I discuss each factor in turn and highlight where they reinforce each other.

4.1. Difficulty getting paid

A 2024 survey found that 60% of OSS maintainers do not receive any kind of payment for their work Tidelift, 2024. Difficulty getting paid for OSS work was often cited as a

cause of burnout in OSS community discussion. The reasons for this were twofold.

Firstly, perception of a lack of fair of reward for work is associated with burnout. This relationship has been observed in software engineers (Shih et al., 2013) in the academic literature.

In OSS community discussion, developers often described the fact that they are scarcely if ever financially rewarded for their work as unfair. Some characterised their work as 'free labour'. Some felt exploited by the proprietary software companies that profit from their code, rely on them to maintain it for free and give nothing in return. Some were additionally worried about 'self-exploitation' (i.e., the compulsion to do free labour in OSS to build a portfolio and improve career prospects).

Despite the fact that the software industry is enormously profitable, getting reliable payment for OSS was generally considered to be an unrealistic prospect—one that requires jumping through hoops, compromising on the kind of work one would like to do, and sheer luck. Many developers saw this as unjust. Several took difficulty getting paid in line with the social value of their work to be indicative of market failure.

> *'Unfortunately, many big companies take and take without giving back in proportion to the value they receive, and these problems aren't just fixing themselves under capitalism. If you put purely economics first and aren't conscious of the things that have enabled your success, then you end up just hurtling towards darkness.'*—Grabanski, 2019

Secondly, high workload is predictive of burnout among software developers (Kuutila et al., 2018; Moore, 2000; Sarker et al., 2019; Shih et al., 2013; Sonnentag et al., 1994; Van Oorschot et al., 2018). Given the difficulty of getting paid for OSS work, developers without independent means described how they were reliant on a concurrent, often full-time job in order to make a living, essentially doing a 'double-shift' to support their work in OSS.

> *'I wasn't getting paid to do the work that I felt like I had to do, and I had to do the work that I was getting paid to do because [of] mortgages and bills and car payments and that sort of stuff, so I had to do, like, double work, one out of a feeling of obligation and one out of a feeling of like, we live in a capitalist society and I need money'*—Kaplan-Moss, 2015

This was associated with a lifestyle of long shifts and all nighters to cope with the

workload, affecting OSS developers' physical and mental health. It also reduced the time they could spend with family and friends and limited their capacity for personal relationships. Indeed, loneliness is a recognised problem for people working in OSS (Pagano, 2014) and is itself a risk factor for burnout (Bryan et al., 2024).

Some developers expressed a desire for payment for OSS work so they can continue contributing to OSS without sacrificing their free time. Others, despondent about the possibility of reliable payment, saw the quitting as the only pathway to reclaiming it.

> *'If and when we decide to start having kids, I will probably quit open source for good. I just can't see how I'll be able to make the time for both raising a family and doing open source.'*—Lawson, 2017

Despite the widespread recognition that not being paid for OSS work contributes to burnout, payment seemed a taboo topic in the OSS community, with most conversation around the importance of paying developers led by advocacy bodies (e.g., Tidelift and Open Collective), rather than the developers themselves. I identified two possible reasons for this.

Firstly, identity, community norms and values motivate participation in OSS (Krishnamurthy, 2006). Introducing payment for OSS might be seen as a move away from the original values of open source. Indeed, one influential developer has written polemically on the dangers of introducing payment for OSS (Hansson, 2021). These values can be an important part of developers' identity and sense of belonging, making it harder for them to talk publicly about payment.

Secondly, developers that *do* manage to get sufficient and reliable payment for OSS work often feel privileged and like they ought not to complain, which can make it uncomfortable for them to weigh in on the topic of payment.

Silence around working conditions is a risk factor for burnout (Lainidi et al., 2025), so the taboo around payment for open source may *itself* contribute to the problem of burnout in OSS.

4.2. Workload and time commitment

The stress caused by a high workload and associated lack of free time is a significant risk factor for burnout, and many OSS developers described their workload as remarkably high. This was for two reasons.

Firstly, as discussed in section 4.1 above, many OSS developers are obligated to undertake additional work alongside their OSS work in order to make a living. High workload as cause of burnout is thus bound-up with difficulty getting paid for OSS.

Secondly, on its own, OSS work can come with a very high workload. Maintainers of popular packages described being swamped with requests and emails from users for support, bug-fixes, updates and features. This was compounded by the fact that many were the sole maintainer on a project, and found it difficult to attract new contributors capable of high quality work. Moreover, reviewing poor quality contributions was described as taking a great deal of time and effort. 'Email overload' has been singled out specifically in the academic literature for its role in burnout (Reinke and Chamorro-Premuzic, 2014), and it was clear that many maintainers were struggling against the tide of notifications.

> *'I was doing nights and weekends, it was wrecking my health and I was just devastated, so after a while, enough issue follow-ups, like, "is this maintained?", "are you gonna fix my issue?", I had to say that I had to give it up.'*—Williams, 2024

4.3. Maintenance work as unrewarding

Work is less likely to lead to burnout if it is intrinsically motivating, i.e., inherently rewarding and meaningful (Gaines and Jermier, 1983). Academic research has shown that software engineers are highly intrinsically motivated (Rubin and Hernandez, 1988), and a love of the creative process of coding was clearly apparent in OSS community discussion.

However, OSS developers noted that software maintenance is a very different kind of work to building software, requiring less creative coding, more repetition and drudgery, and more communication and people management skills—skills not all OSS developers possess or enjoy exercising.

> *'I would rather be remembered as a bad artist than a good programmer. … now I'm asked more and more… to express myself less and to maintain the project more.'*—Asay, 2020

Echoing concerns about the high workload associated with software maintenance, maintenance work was often described as taking up time developers would rather spend doing coding that they actually enjoyed and could learn something from. Software engineers are at greater risk of burnout when they have a lack of autonomy over their work (i.e., the ability to choose what to work on, when and how) (Shih et al., 2013), and a lack of opportunity to learn (Trinkenreich, Santos, and Stol, 2024).

Work that is not intrinsically motivating needs to be extrinsically rewarded; when there is an imbalance, such that the magnitude of *effort put in* is far greater than the *reward received*, the risk of burnout increases (Van Vegchel et al., 2005). Unfortunately, maintenance work is rarely extrinsically rewarded. As noted in section 4.1, most OSS maintainers are not rewarded financially for their work. Moreover, as will be discussed further in section 4.4, maintenance work can feel emotionally unrewarding due to a lack of appreciation and entitled and uncivil behaviour among the OSS userbase. As one developer put it, there are:

> *'[s]o many projects that their creator poured their energy in, but for one reason or another didn't get the recognition they deserved or the support they needed.'*—Kappert, 2024

In sum, maintenance work comes with a high risk of burnout, as OSS developers tend to find it less inherently rewarding that other software development tasks, and it is insufficiently externally rewarded to compensate for this. The more time OSS developers feel obliged to spend doing maintenance work instead of engaging in creative coding, the greater the risk of burnout. This is compounded by the fact that many must spend their remaining time doing additional paid work to supplement their work in OSS, rather than working on creative coding projects of their choosing.

A Note on AI use

There was a sense in the OSS developer community that, like it or not, AI use in coding is here to stay. When it came to the effect of AI use on burnout, the developers in my research were divided into two camps.

One camp felt that AI use had increased their workload by making it easier for contributors to submit low quality 'slop' code. They felt the time that contributors save by using AI is time that maintainers have to spend fixing it at the review stage. Reviewing AI generated code was described as 'mind-numbing', suggesting it is particularly unrewarding and unedifying to engage with work created by an intentionless algorithmic process. If AI use increases developers' sense of unfairness and makes maintenance work even less rewarding, the rise of AI use in coding could worsen maintainer burnout.

The other camp saw AI in more neutral terms as a *tool*. Whether this tool makes burnout better or worse depends its usage. For example, AI can be used to make workflows more efficient, saving developers time and energy and thus reducing their risk of burnout. On the other hand, it can serve as a barrier to education, as it is easy to reach for AI solutions instead of putting in the time and effort to learn. This could lead to lower quality submissions and limit the pool of talented OSS developers, leaving those that remain struggling with burnout-inducing workloads.

Looking at the arguments on both sides of the debate, it seems there are two things that can be done to prevent AI use exacerbating maintainer burnout. Firstly, by cultivating means of filtering out and refusing poor quality AI-generated contributions. Secondly, by improving education among contributors and developers on how, and how not, to use AI in collaborative coding, such that it does not prevent developers learning new skills and can change their workflows for the better.

In sum, AI use could drastically change the experience of being an OSS maintainer, for better or for worse. OSS developers are sounding the alarm, and the quicker action is taken, the greater the chance of the rise in AI having a positive, rather than a negative, effect on burnout.

'*We need to reduce the amount of sand in the machine. We must do something to drastically reduce the temptation for users to submit low quality reports. Be it with AI or without AI.*'—Stenberg, 2025

4.4. Toxic community behaviour

Positive, supportive relationships with colleagues and feelings of belonging decrease the risk of burnout, while difficult communication and conflict increase it. This relationship has been observed among software engineers generally and people working in OSS (Rutner et al., 2011; Shih et al., 2013; Sonnentag et al., 1994; Trinkenreich, Gerosa, et al., 2024).

Unfortunately, hostility and toxic behaviour is a significant issue in the OSS community (Cohen, 2021), and many OSS developers cited toxic community behaviour as

contributing to their burnout.

Toxic behaviour is also relatively common. In a sample of non-technical emails from the Linux kernel mailing list, two thirds were uncivil (Ferreira et al., 2021). Sentiment analyses have estimated the rate of toxic comments in GitHub issues at .07-1.5% (Cheriyan et al., 2021; Raman et al., 2020), and that toxicity is higher in OSS than corporate software.

One source of toxic behaviour is entitled users. Developers often described having to deal with users demanding features and bug fixes and turning to insults when their needs aren't met. This was supported by the academic literature, which suggests the most prevalent forms of OSS community toxicity are insults, entitlement and demandingness (Miller et al., 2022).

Demanding users left developers feeling that maintenance work is thankless and underappreciated—that good work goes unrecognised and mistakes incite anger or even public shaming. Several wondered if users might act differently if they knew many projects are maintained by a small team or single developer labouring voluntarily in their free time. Others noted that demands for features and fixes often came from proprietary software companies who are well aware of this, leaving them feeling exploited by those with the capacity to pay them for their work.

> *'[N]othing is stopping people to bash [sic] us, for work we aren't paid for, for a feature we all dislike yet needed to keep due to backward compatibility concerns.'*

The combination of great expectations and little recognition, respect or reward from the users contributes to maintainers' feelings of unfairness and perception of maintenance as unrewarding, putting them at risk of burnout.

> *'I just gave up, because what's the point when all you get is constant issues? You give and give and give, and people just take and take and take.'*—Grabanski, 2019

Another source of toxicity is developers themselves. Emotional dissonance (i.e., having to act differently towards people than you actually feel) is associated with burnout (Abraham, 1998). Unable to keep up the pretense, some developers described becoming ruder to users the more burnt out they felt.

> *'I started identifying [my burnout] as other people's problem,"if only these these idiots would stop submitting bad code that I would have to review","if only these these jerks would stop asking for help", like, it wasn't my fault that that I couldn't muster the energy to work on this, it was all these these stupid people out there.'*—Kaplan-Moss, 2015

The academic literature suggests that the ruder user-developer interactions are, the longer problems take to solve, the more they occupy developers' time (Destefanis et al., 2016). As noted in 4.2, time and workload play a significant role in contributing to OSS developer burnout. This suggests there is a risk of a burnout death-spiral, in which rudeness from users exacerbates burnout in developers, which makes developers ruder to users and makes issues take longer to solve, further entrenching developer burnout.

Developer-developer interactions can also be toxic. Open source is unusual in that it is massively co-operative, and yet few developers are ever physically present in the same space as each other. There is no one to offer training in collaboration, conflict resolution or leadership. Unless a project team decides to create them, there are no formal support structures, no complaints processes, no guiding rules or principals of governance. Developers sometimes cited these peculiarities as increasing the risk of toxicity within OSS project teams, and making it harder to resolve it once it takes hold.

To make matters worse, fears of reprisal or burning professional bridges made it challenging for developers to call out the bad behaviour of others, and there was a perceived shortage of safe spaces for developers to talk about the painful feelings community toxicity arouses.

> *'At first, I tried to deny my feelings and "just do my job". That lead to severe burnout and depression. In order to deny my feelings of powerlessness, my anger at the community, I had to stuff everything into an emotional box and close the lid. I usually follow a lot of technology activists, but in order to get my job done, I had to deny that the problem even existed. In order to mentally survive in a community that would not allow change, I could not acknowledge that change was possible or valued.'*—Anonymous Author, n.d.

The prevalence of toxic behaviour within OSS gives the impression of a community in which developers are hurting because they are overstretched and underappreciated, causing them to spill that hurt out onto others. This can create an unsupportive working environment that is particularly conducive to burnout.

4.5. Hyper-responsibility

While intrinsically motivating work is generally associated with a decreased risk of burnout, it is *more* likely to lead to burnout under conditions of high stress, since when one really cares to get the job done well, one might willfully miss signals that they are overstretched and need to slow down (Sonnentag et al., 1994).

OSS developers are not only highly intrinsically motivated by the joy of coding, they are also motivated by a desire to benefit others, their membership in the OSS community, and social and political beliefs about the value of open source (Hertel et al., 2003). OSS developers often described having an enormous sense of responsibility to be good stewards of their projects: on behalf of their users, the community, and the principle of open source itself. The thought of stepping back left some developers feeling guilty, like they would be letting the community down.

This hyper-responsibility puts developers at greater risk of burnout, as they are more likely to persist with work beyond a point that is healthy when under stress (for example, from high workloads discussed in 4.2, or maltreatment by the community discussed in 4.4).

Maintainers of popular packages sometimes characterised their software gaining popularity as a curse disguised as a blessing, owing to the extra responsibility that comes with having so many people depend on their project.

> *'My open source success went from a major blessing to a great curse. It was one of the darkest times in my life. Something that started out with such hope and light ended up just being about getting thousands of emails. People told me their whole life stories and how it's all been leading up to this one feature they really need me to add.'*—Grabanski, 2019

To make matters worse, many felt *solely* responsible for their project, and that it would 'die' without their work, since it is so hard to attract new quality contributors. This problem is compounded by the toxic community behaviour discussed in 4.4, since toxic community interactions have been shown to deter newcomers from contributing (Jensen et al., 2011; Steinmacher et al., 2013).

> *'I'm the only person holding this together, if I leave who will do this?'*—Rogers, 2017

Developers that did manage to stay clear of burnout discussed having to learn that, despite

their conscientiousness, it is okay to say no, and to match the energy they put in to the energy put in by others (e.g., by not replying to users that don't provide minimal reproducible examples, or not spending hours combing over poorly-written patches).

Finally, developers also described feeling pressure to ensure their software is secure in a climate where industry security standards are increasingly challenging to meet. This highlights how important it is that security legislation protects OSS developers from liability for security issues relating to their code. Not being able to say no to upholding these standards would put developers in an unreasonable position of responsibility and at even greater risk of burnout.

4.6. Pressure to prove oneself

External pressures can make work more stressful, increasing the risk of burnout (Sonnentag et al., 1994). OSS developers described feeling pressure to constantly prove themselves, both to the community and to prospective employers to whom their OSS work could serve as a portfolio. They felt pressure to constantly skill-up to keep pace with an increasingly complex software development landscape—the high learning requirements of software development are a noted risk factor for burnout (Sonnentag et al., 1994). They also felt pressure to show their worth both as a developer and to the OSS community through their contributions. Some called out GitHub for playing a role in this with features like achievements and badges that were taken to gamify contributions.

> *'It's all in the numbers — followers, contributions, comments, stars.'*—Szczur, 2015

Fear of losing the reputation they had built for themselves, the connections they had made in open source, and their identity as members of the OSS community created pressure for developers to continue working beyond the point of burnout.

> *'it becomes your identity, you rub shoulders with really influential people in the industry, and I don't want to get kicked out of the github organisation so I gotta keep going—that's how bad it gets … one day I was like, 'I can't do this anymore'*—Stacoviak and Santo, 2018

Since the pressure OSS developers feel to prove themselves is intertwined with their sense of worth, their identity, and their position in the community, it may be a particularly pernicious source of burnout, making it extraordinarily difficult for developers to take a step back from work when overburdened.

5. Recommendations

There is an upside to the causes of burnout in OSS being inter-related and mutually reinforcing: addressing burnout on one front could go some way towards addressing it on others, leading to improvements that cascade throughout the system.

Having identified some key causes of burnout in OSS and where they intersect, I will now recommend 4 directions that could be pursued in order to address them. Each undoubtedly comes with its own unique set of challenges and could be realised in multiple ways; for the purposes of this report, I keep my suggestions broad and am not dictatorial about the best way to implement them in practice.

In recognition of the fact that the problem of burnout in OSS is structural and systemic, my recommendations will focus on changes that can be made at the system level, not on how individual OSS developers can change their behaviour to avoid burning out.

5.1. Pay OSS developers

Routine, reliable payment for OSS work would reduce the risk of OSS burnout on multiple fronts. It would remove developers' reliance on a concurrent full-time job to make a living, lightening their workloads and freeing up time for them to spend with friends, family and working on projects they love. It would change developers' perception that they are being unfairly rewarded for their OSS and alter the effort-reward imbalance associated with maintenance work. It would make it easier and potentially more appealing for people to participate in open source, increasing the chances of developers finding talented contributors with whom to share their workload and responsibilities. Finally, it could decrease the pressure developers feel to continuously prove themselves and remain competitive in the hope of getting work that pays in the future.

It is important to note that some developers fear the wrong model of payment could create pressure to act in the best interests of funders, rather their own interests or the interests of the community. This could in fact make burnout worse–research suggests software developers are at greater risk of burnout when they lack autonomy over their work (Shih et al., 2013; Whitacre, 2013).

This fear is not unfounded: pressure to appease a source of funding seems to have played a role in the recent Ruby Gems takeover, resulting in a change in project priorities and maintainers losing control over a project they stewarded for many years (Page, 2025).

As such, the model of payment for OSS that is most protective against burnout will

be one that allows maintainers to make a living while maintaining creative control over their work. This could be achieved through a decentralised funding model, in which projects are not reliant on a single source of industry funding for survival. Collective governance may also have a role to play by ensuring everyone involved project has a say in its direction.

It is also important that payment be predictable and regular; it is hard to plan effectively plan one's life and time around sporadic tips and donations. It is for this reason that some of the developers I spoke to favoured salaried jobs with time set aside specifically for open source, or some version of a universal basic income.

For a discussion of the challenges associated with payment for OSS, see Eghbal, 2016, pp. 106–108. For discussion of current payment models, see Eghbal, 2016, pp. 109–123 Tidelift, 2024, pp. 13–16.

5.2. Foster a culture of recognition and respect

A shift in the OSS community from entitlement and toxicity towards a culture of recognition and respect would go a long way in reducing OSS developer burnout. While cultural shifts are undoubtedly hard to achieve, there are some promising avenues through which to pursue it.

Firstly, psychological research suggests community leaders (i.e., people the community look up to) are particularly effective at shaping group norms, values and behaviours (Steffens, Haslam, Reicher, et al., 2014). Groups with leaders who foster a positive shared identity are less prone to burnout (Steffens, Haslam, Kerschreiter, et al., 2014; van Dick et al., 2021). Empowering OSS community leaders who model respectful conduct to share their knowledge and insights with others—for example, through mentoring, talks, or creative writing—could help help us move towards a kinder culture in OSS. Promisingly, some respected community members are already beginning to engage in this kind of positive identity leadership (Czaplicki, 2018; Mastery Learning, 2024).

Secondly, entitled users seem to be treating OSS developers as if they were a company providing a service (Depierre, 2022). Shaking users out of their default mode of consumption to realise that open source is something like a gift economy rather than a market economy might make them less inclined to be hostile and demanding. This would make maintenance work more rewarding for developers, reducing the risk of burnout.

Institutions like GitHub are well-positioned to educate users about the nature of open source, and their Maintainer Month initiative shows they are willing to engage in this

kind of work. Several OSS community members suggested GitHub could also introduce features that help manage user expectations, for example, by highlighting when a project is maintained by one person in their free time (Sapegin, 2023b).

Practical measures to address how users engage with OSS developers could be supplemented by theoretical work that considers what we owe them ethically. This could entail framing OSS as a gift for which gratitude or future reciprocation is appropriate (Whitacre, 2025), or applying ethical principles such as the ethic of care to OSS maintenance (Harbuz, 2024b; The Information Maintainers et al., 2021).

Finally, overstretched and isolated developers can end up being rude and unhelpful to both users and the developers they work with. Improving the availability of social support, emotional support and training in collaboration, communication and leadership could reduce toxic behaviour among developers, making OSS working environments less conducive to burnout.

Companies and foundations wishing to give thanks to OSS developers could facilitate access to coaching, mental health support, and training resources that enable developers to feel socially supported and equipped to deal with conflict with users and within developer teams. They could also sponsor events that that focus on fostering a sense of community—safe spaces where OSS developers can gather, meet face-to-face, support each other, express their feelings and frustrations and find paths through them together. Community events like these are highly valued by developers, but are sadly few and far between and can be costly to attend.

Happily, whatever measures we take to reduce burnout in OSS, these are also likely to reduce community toxicity, as less burnout means OSS developers will have greater cognitive and emotional bandwidth to communicate helpfully and politely.

5.3. Grow the community

Encouraging more people to participate in or remain in open source could help reduce burnout by increasing the pool of developers willing and able to share the workload and responsibility for existing projects.

As noted in recommendation 5.1, paying developers and reducing toxic community behaviour could go a long way in empowering more people to join and remain in OSS (Jensen et al., 2011; Soelton et al., 2020; Steinmacher et al., 2013).

Improving access to education would also ensure newcomers are able to gain the skills necessary to make quality contributions to OSS (Rogers, 2017). While there are

lots of online resources for learning programming skills in general, some skills that are particularly relevant to open source are harder to find instruction in. These include the tools and practices necessary for successful collaborative coding, and how to get orientated with new codebases.

These sorts of skills may be best learnt through mentoring relationships with senior project members. However, senior project members may not have the time to mentor others if they are already struggling with enormous workloads. Equally, they may be unwilling or unable to invest emotional energy in coaching newcomers if they are have been left wary by toxic community behaviour, low-effort contributions and malicious actors exploiting the state of burnout in OSS to install backdoors, as in the XZ utils hack (Tidelift, 2024). This shows how it is necessary to tackle the causes of burnout in OSS on multiple fronts in order to create the conditions for healthy community growth.

5.4. Advocate for developers

Since the issues causing OSS developers to burn out exist at the system-level, we need bodies that are capable of advocating for system-level change.

As noted in recommendation 5.2, organisations like GitHub could support developers by using their position of power in the OSS ecosystem to educate users about the realities of working OSS and manage their expectations. Since it has become such a key component of OSS infrastructure, GitHub can advocate for developers though how it is designed, including features that empower and protect OSS developers and removing features that put them under increased pressure. Indeed, the developers in my research seemed to respond more positively to GitHub making structural changes that address the causes of burnout—like the inclusion of triage and maintenance roles, or the ability to block users—than GitHub promoting individual changes maintainers themselves can make to avoid burning out (Sapegin, 2023b; swyx, n.d.).

It is also important that there are advocacy bodies capable of representing the interests of OSS developers politically. Burnout in open source is a political issue. Much of the world's critical infrastructure relies on open source. When maintainers burn out and are unable to steward it effectively, the functioning and security of this critical infrastructure is at risk (Harbuz, 2024a; O'Neill, 2021; Saarinen, 2018). Effective advocacy could raise the profile and the urgency of the issue of burnout in open source at the level of government, potentially attracting funding and research dedicated to addressing it.

It is also important that someone is there to advocate for OSS developers when legislative decisions are made that could affect their lives. One area where this is particularly

likely to be the case is cybersecurity legislation, such as the EU Cybersecurity Resilience Act. It is critical that security legislation does not leave developers liable for security issues relating to their code, as this would put them under enormous pressure and increase the risk of burnout.

Encouragingly, there are already a number of organisations and initiatives advocating for payment for OSS developers (e.g., thanks.dev, the Open Source Pledge, the Open Source Endowment, Github Sponsors, Open Collective, Tidelift (acquired by Sonar in 2024) and Liberapay). As recommendation 5.1 describes, this could go a long way in improving working conditions for OSS developers and reducing their risk of burnout.

A Note on Diversity

A noted limitation of the methodology of this report is that the majority of OSS community discourse analysed in the rapid literature review was authored by men, and people who are ethnically white. While this may be representative of the field—gender and racial diversity is lower in OSS than in programming as a whole (Finley, 2017)—it is important to include the experiences of marginalised groups if we are to understand and address the barriers to participation that uphold this disparity.

Burnout is likely to affect marginalised groups (e.g. women, ethnic minorities, the working class and LGBT+ people) differently. For every cause of burnout, it is plausible they are at greater risk. They may be less likely to have independent means and therefore more reliant on a second job to support their OSS. They may be expected to take on extra labour such as childcare and domestic work, placing greater demands on their time. They may have less social support in the OSS community (Qiu et al., 2019) or experience mistreatment on the basis of their identity—indeed, harassment of women was a noted issue in OSS community discussion (Anonymous Author, n.d.; Kaplan-Moss, 2015). Against this backdrop, they may feel a greater pressure to prove themselves to the OSS community (Pagano, 2014), have a harder time saying no to additional work, and be uncomfortable speaking out about their working conditions.

Paying OSS developers and fostering a culture of respect and recognition may be especially important for improving burnout among marginalised groups. Efforts to grow the community and advocate for maintainers should be sensitive to the additional barriers the marginalised face. Happily, some work is already being done in this direction, e.g., Outreachy, who provide internships to underrepresented groups in OSS.

6. Conclusion

Failing to address the problem of burnout in OSS poses the possibility of real harm to OSS developers. Burnout damages their physical and mental health. By affecting emotional regulation, it can negatively impact their relationships. By increasing the chances of quitting, it can shape the very direction of their lives.

Since OSS is the backbone of our critical software infrastructure, failure to address the problem of burnout in OSS also puts our entire software ecosystem at risk. Burnout is associated with a decline in work quality, threatening the gains in innovation and reliability from which we all benefit from when developers choose to make their code available open source. Burnt out developers are more likely to quit, leaving the potentially millions of users that rely on their code unsupported if no one else is able to take up the mantle. Even more worryingly, an increasing number of security incidents with the potential to affect a vast proportion of our software ecosystem have resulted from malicious actors exploiting developers struggling with burnout.

Burnout is not just a problem for OSS developers, it is a problem for all of us. It is also a problem that can be most effectively addressed by working collaboratively to achieve system-wide change.

References

Abraham, R. (1998). Emotional dissonance in organizations: Antecedents, consequences, and moderators. *Genetic, Social, and General Psychology Monographs*, *124*(2), 229–246.

Anonymous Author. (n.d.). "Leaving toxic open source communities". Model View Culture. https://modelviewculture.com/pieces/leaving-toxic-open-source-communities

Asay, M. (2020). "What does an open source maintainer do after burnout?" InfoWorld. https://www.infoworld.com/article/2258588/what-does-an-open-source-maintainer-do-after-burnout.html

Bryan, B. T., Andrews, G., Thompson, K. N., Qualter, P., Matthews, T., & Arseneault, L. (2024). Loneliness in the workplace: A mixed-method systematic review and meta-analysis. *Occupational Medicine (Oxford, England)*, *73*(9), 557–567. https://doi.org/10.1093/occmed/kqad138

Cheriyan, J., Savarimuthu, B. T. R., & Cranefield, S. (2021). Towards offensive language detection and reduction in four Software Engineering communities. *Evaluation and Assessment in Software Engineering*, 254–259. https://doi.org/10.1145/3463274.3463805

Cohen, S. (2021). Contextualizing toxicity in open source: A qualitative study. *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, 1669–1671. https://doi.org/10.1145/3468264.3473492

Czaplicki, E. (2018). "The hard parts of open source". Strange Loop Conference. https://www.youtube.com/watch?v=o_4EX4dPppA

Demerouti, E., Bakker, A. B., Nachreiner, F., & Schaufeli, W. B. (2001). The job demands-resources model of burnout. *Journal of Applied Psychology*, *86*(3), 499–512. https://doi.org/10.1037/0021-9010.86.3.499

Depierre, T. (2022, December 31). "I am not a supplier". Musings about software. https://softwaremaxims.com/blog/Not-A-Supplier

Destefanis, G., Ortu, M., Counsell, S., Swift, S., Marchesi, M., & Tonelli, R. (2016). Software development: Do good manners matter? *PeerJ Computer Science*, *2*, e73. https://doi.org/10.7717/peerj-cs.73

Eghbal, N. (2016). "Roads and bridges: The unseen labor behind our digital infrastructure". Ford Foundation. https://www.fordfoundation.org/wp-content/uploads/2016/07/roads-and-bridges-the-unseen-labor-behind-our-digital-infrastructure.pdf

Ferreira, I., Cheng, J., & Adams, B. (2021). The "Shut the f**k up" phenomenon: Characterizing incivility in open source code review discussions. *Proceedings of the ACM on Human-Computer Interaction*, *5*, 1–35. https://doi.org/10.1145/3479497

Finley, K. (2017). "Diversity in open source is even worse than in tech overall". Wired. https://www.wired.com/2017/06/diversity-open-source-even-worse-tech-overall/

Gaines, J., & Jermier, J. M. (1983). Emotional exhaustion in a high stress organization. *Academy of Management Journal*, *26*(4), 567–586. https://doi.org/10.2307/255907

Grabanski, M. (2019). “Frontend Masters: “We teach open source, we use open source, and we support open source.””. Open Collective. https://blog.opencollective.com/frontend-masters/

Graziotin, D., Fagerholm, F., Wang, X., & Abrahamsson, P. (2017). Consequences of unhappiness while developing software. https://doi.org/10.48550/ARXIV.1701.05789

Hansson, D. H. (2021). “I won’t let you pay me for my open source”. David Heinemeier Hansson. https://world.hey.com/dhh/i-won-t-let-you-pay-me-for-my-open-source-d7cf4568

Harbuz, V.-S. (2024a). “Not paying open source maintainers is expensive”. Open Source Pledge. https://opensourcepledge.com/blog/not-paying-open-source-maintainers-is-expensive/

Harbuz, V.-S. (2024b). “The philosophy of the Open Source Pledge”. Vlad-Stefan Harbuz. https://vlad.website/the-philosophy-of-the-open-source-pledge/

Hertel, G., Niedner, S., & Herrmann, S. (2003). Motivation of software developers in Open Source projects: An internet-based survey of contributors to the Linux kernel. *Research Policy*, *32*(7), 1159–1177. https://doi.org/10.1016/S0048-7333(03)00047-7

Jensen, C., King, S., & Kuechler, V. (2011). Joining free/open source software communities: An analysis of newbies’ first interactions on project mailing lists. *2011 44th Hawaii International Conference on System Sciences*, 1–10. https://doi.org/10.1109/HICSS.2011.264

JetBrains. (2023). “The state of developer ecosystem 2023”. https://www.jetbrains.com/lp/devecosystem-2023

Kaplan-Moss, J. (2015). “What part of for life don’t you understand?” Open Source and Feelings 2015. https://www.youtube.com/watch?v=EqcuzSwySR4

Kappert, L. (2024, June 28). “Privileged, but not a clown”. WebPro. https://webpro.nl/articles/privileged-but-not-a-clown

Krishnamurthy, S. (2006). On the intrinsic and extrinsic motivation of free/libre/open source (FLOSS) developers. *Knowledge, Technology & Policy*, *18*(4), 17–39. https://doi.org/10.1007/s12130-006-1002-x

Kuutila, M., Mäntylä, M. V., Claes, M., Elovainio, M., & Adams, B. (2018). Using experience sampling to link software repositories with emotions and work well-being. *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, 1–10. https://doi.org/10.1145/3239235.3239245

Kyle, J. (n.d.). “Dear JavaScript”. Jamie Kyle. https://jamie.build/dear-javascript.html

Lainidi, O., Johnson, J., Griffin, B., Koutsimani, P., Mouratidis, C., Keyworth, C., & O'Connor, D. B. (2025). Associations between burnout, employee silence and voice: A systematic review and meta-analysis. *Psychology & Health*, 1–21. https://doi.org/10.1080/08870446.2025.2509074

Lawson, N. (2017). "What it feels like to be an open-source maintainer". Nolan Lawson. https://nolanlawson.com/2017/03/05/what-it-feels-like-to-be-an-open-source-maintainer/

MacIver, D. R. (2015). "Throwing in the towel". https://www.drmaciver.com/2015/08/throwing-in-the-towel/

Maslach, C., Schaufeli, W. B., & Leiter, M. P. (2001). Job Burnout. *Annual Review of Psychology*, *52*(1), 397–422. https://doi.org/10.1146/annurev.psych.52.1.397

Mastery Learning (with Torvalds, L.). (2024). "Linus Torvalds addressing maintainer fatigue". https://www.youtube.com/watch?v=eWfUaFNSPhM

Miller, C., Cohen, S., Klug, D., Vasilescu, B., & KaUstner, C. (2022). "Did you miss my comment or what?": Understanding toxicity in open source discussions. *Proceedings of the 44th International Conference on Software Engineering*, 710–722. https://doi.org/10.1145/3510003.3510111

Moore, J. E. (2000). One road to turnover: An examination of work exhaustion in technology professionals. *MIS Quarterly*, *24*(1), 141. https://doi.org/10.2307/3250982

O'Neill, P. H. (2021). "The internet runs on free open-source software. Who pays to fix it?" MIT Technology Review. https://www.technologyreview.com/2021/12/17/1042692/log4j-internet-open-source-hacking/

Pagano, J. (2014, April 24). "It's dangerous to go alone: Battling the invisible monsters in tech". PyCon 2014. https://www.youtube.com/watch?v=1i8ylq4j_EY

Page, C. (2025). "Open source to closed doors: RubyGems control fight erupts". https://www.theregister.com/2025/09/25/open_source_to_closed_doors/

Qiu, H. S., Nolte, A., Brown, A., Serebrenik, A., & Vasilescu, B. (2019). Going farther together: The impact of social capital on sustained participation in open source. *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*, 688–699. https://doi.org/10.1109/ICSE.2019.00078

Raman, N., Cao, M., Tsvetkov, Y., Kästner, C., & Vasilescu, B. (2020). Stress and burnout in open source: Toward finding, understanding, and mitigating unhealthy interactions. *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: New Ideas and Emerging Results*, 57–60. https://doi.org/10.1145/3377816.3381732

Reinke, K., & Chamorro-Premuzic, T. (2014). When email use gets out of control: Understanding the relationship between personality and email overload and their impact

on burnout and work engagement. *Computers in Human Behavior*, *36*, 502–509. https://doi.org/10.1016/j.chb.2014.03.075

Rogers, M. (2017, June 12). "Maintainer vs. community". Open Source Communities. https://medium.com/open-source-communities/maintainer-vs-community-97edc28387ad

Rubin, H. I., & Hernandez, E. F. (1988). Motivations and behaviours of software professionals. *Proceedings of the 1988 ACM SIGCPR Conference*, 62–71. https://doi.org/10.1145/57216.57225

Rutner, P., Riemenschneider, C., O'Leary-Kelly, A., & Hardgrave, Bi. (2011). Work exhaustion in information technology professionals: The impact of emotion labor. *ACM SIGMIS Database: the DATABASE for Advances in Information Systems*, *42*(1), 102–120. https://doi.org/10.1145/1952712.1952718

Saarinen, J. (2018). Unknown dev gets rights to popular module, adds crypto stealer. *IT News*.

Sapegin, A. (2023a). "Healthier way to open source your code". dev.to. https://dev.to/sapegin/healthier-way-to-open-source-your-code-2el6

Sapegin, A. (2023b). "Why I quit open source". dev.to. https://dev.to/sapegin/why-i-quit-open-source-1n2e

Sarker, F., Vasilescu, B., Blincoe, K., & Filkov, V. (2019). Socio-technical work-rate increase associates with changes in work patterns in online projects. *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*, 936–947. https://doi.org/10.1109/ICSE.2019.00099

Schaufeli, W., Hakanen, J., & Shimazu, A. (2023). Burning questions in burnout research. In N. De Cuyper (Ed.), *Job Insecurity, Precarious Employment and Burnout*.

Shih, S.-P., Jiang, J. J., Klein, G., & Wang, E. (2013). Job burnout of the information technology worker: Work exhaustion, depersonalization, and personal accomplishment. *Information & Management*, *50*(7), 582–589. https://doi.org/10.1016/j.im.2013.08.003

Singh, P., & Suar, D. (2013). Health consequences and buffers of job burnout among Indian software developers. *Psychological Studies*, *58*(1), 20–32. https://doi.org/10.1007/s12646-012-0171-9

Soelton, M., Lestari, P. A., Arief, H., & Putra, R. L. (2020). The effect of role conflict and burnout toward turnover intention at software industries, work stress as moderating variables. *Proceedings of the 4th International Conference on Management, Economics and Business (ICMEB 2019)*. https://doi.org/10.2991/aebmr.k.200205.034

Sonnentag, S., Brodbeck, F. C., Heinbokel, T., & Stolte, W. (1994). Stressor-burnout relationship in software development teams. *Journal of Occupational and Organi-*

zational Psychology, *67*(4), 327–341. https://doi.org/10.1111/j.2044-8325.1994.tb00571.x

Stacoviak, A., & Santo, J. (2018). *A call for kindness in open source with Brett Cannon (Changelog Interviews #318)*. https://changelog.com/podcast/318

Steffens, N. K., Haslam, S. A., Kerschreiter, R., Schuh, S. C., & van Dick, R. (2014). Leaders enhance group members' work engagement and reduce their burnout by crafting social identity. *German Journal of Human Resource Management*, *28*(1–2), 173–194. https://doi.org/10.1177/239700221402800110

Steffens, N. K., Haslam, S. A., Reicher, S. D., Platow, M. J., Fransen, K., Yang, J., Ryan, M. K., Jetten, J., Peters, K., & Boen, F. (2014). Leadership as social identity management: Introducing the Identity Leadership Inventory (ILI) to assess and validate a four-dimensional model. *The Leadership Quarterly*, *25*(5), 1001–1024. https://doi.org/10.1016/j.leaqua.2014.05.002

Steinmacher, I., Wiese, I., Chaves, A. P., & Gerosa, M. A. (2013). Why do newcomers abandon open source software projects? *2013 6th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*, 25–32. https://doi.org/10.1109/CHASE.2013.6614728

Stenberg, D. (2025). "Death by a thousand slops". daniel.haxx.se. https://daniel.haxx.se/blog/2025/07/14/death-by-a-thousand-slops/

swyx. (n.d.). "Breaking apart the monolith". The ReadME Project. https://github.com/readme/guides/maintainer-monolith

Szczur, K. (2015). "The dark side of open source". Karolina Szczur. https://medium.com/@fox/the-dark-side-of-open-source-ba5a66c8a4c3

The Information Maintainers, Olson, D., Meyerson, J., Parsons, M. A., Castro, J., Lassere, M., Wright, D. J., Arnold, H., Galvan, A. S., Hswe, P., Nowviskie, B., Russell, A., Vinsel, L., & Acker, A. (2021). "Information maintenance as a practice of care". https://themaintainers.org/wp-content/uploads/2021/05/Information-Maintenance-as-a-Practice-of-Care.pdf

Tidelift. (2024). "The 2024 Tidelift state of the open source maintainer report". Tidelift.

Trinkenreich, B., Gerosa, M. A., Sarma, A., & Steinamcher, I. (2024). Guidelines for cultivating a sense of belonging to reduce developer burnout. *IEEE Xplore*. https://doi.org/110.1109/MS.2024.3404361

Trinkenreich, B., Santos, F., & Stol, K.-J. (2024). Predicting attrition among software professionals: Antecedents and consequences of burnout and engagement. ACM Transactions on Software Engineering and Methodology. *ACM Transactions on Software Engineering and Methodology*, *33*(8), 1–45. https://doi.org/10.1145/3691629

Van Oorschot, K. E., Sengupta, K., & Van Wassenhove, L. N. (2018). Under pressure: The effects of iteration lengths on agile software development performance. *Project Management Journal*, *49*(6), 78–102. https://doi.org/10.1177/8756972818802714

Van Vegchel, N., De Jonge, J., Bosma, H., & Schaufeli, W. (2005). Reviewing the effort–reward imbalance model: Drawing up the balance of 45 empirical studies. *Social Science & Medicine*, *60*(5), 1117–1131. https://doi.org/10.1016/j.socscimed.2004.06.043

van Dick, R., Cordes, B. L., Lemoine, J. E., Steffens, N. K., Haslam, S. A., Akfirat, S. A., Ballada, C. J. A., Bazarov, T., Aruta, J. J. B. R., Avanzi, L., Bodla, A. A., Bunjak, A., Černe, M., Dumont, K. B., Edelmann, C. M., Epitropaki, O., Fransen, K., García-Ael, C., Giessner, S., … Kerschreiter, R. (2021). Identity leadership, employee burnout and the mediating role of team identification: Evidence from the Global Identity Leadership Development Project. *International Journal of Environmental Research and Public Health*, *18*(22), 12081. https://doi.org/10.3390/ijerph182212081

Whitacre, C. (2013). "Open Call w/ David Heinemeier Hansson". https://www.youtube.com/watch?v=p1E-svVd9Xc

Whitacre, C. (2025). "Is there a crisis in Open Source software?" https://www.youtube.com/watch?v=eTcF5dPSSDQ

Williams, A. (2024). "2 Wolves". Monktoberfest 2024. https://www.youtube.com/watch?v=gR2hjbh0RUk

Appendix A: Rapid Literature Review Sources

Atouba, Y. C., & Lammers, J. C. (2020). Examining the relationships between participative organisational communication practices and burnout among IT professionals. *Total Quality Management & Business Excellence*, *31*(7–8), 814–828. https://doi.org/10.1080/14783363.2018.1447367

Cheriyan, J., Savarimuthu, B. T. R., & Cranefield, S. (2021). Towards offensive language detection and reduction in four Software Engineering communities. *Evaluation and Assessment in Software Engineering*, 254–259. https://doi.org/10.1145/3463274.3463805

Cohen, S. (2021). Contextualizing toxicity in open source: A qualitative study. *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, 1669–1671. https://doi.org/10.1145/3468264.3473492

Destefanis, G., Ortu, M., Counsell, S., Swift, S., Marchesi, M., & Tonelli, R. (2016). Software development: Do good manners matter? *PeerJ Computer Science*, *2*, e73. https://doi.org/10.7717/peerj-cs.73

Ferreira, I., Cheng, J., & Adams, B. (2021). The "Shut the f**k up" phenomenon: Characterizing incivility in open source code review discussions. *Proceedings of the ACM on Human-Computer Interaction*, *5*(CSCW2), 1–35. https://doi.org/10.1145/3479497

Filippova, H. C. A. (2016). The effects and antecedents of conflict in free and open source software development. *CSCW '16: Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing, 705–716*. https://doi.org/10.1145/2818048.282001

Graziotin, D., Fagerholm, F., Wang, X., & Abrahamsson, P. (2017). Consequences of unhappiness while developing software. *2017 IEEE/ACM 2nd International Workshop on Emotion Awareness in Software Engineering (SEmotion), 42–47*. https://doi.org/10.1109/SEmotion.2017.5

Hars, A., & Ou, S. (2002). Working for free? Motivations for participating in Open-Source Projects. *International Journal of Electronic Commerce*, *6*(3), 25–39. https://www.jstor.org/stable/27751021

Hertel, G., Niedner, S., & Herrmann, S. (2003). Motivation of software developers in Open Source projects: An internet-based survey of contributors to the Linux kernel. *Research Policy*, *32*(7), 1159–1177. https://doi.org/10.1016/S0048-7333(03)00047-7

Jensen, C., King, S., & Kuechler, V. (2011). Joining free/opensSource software communities: An analysis of newbies' first interactions on project mailing lists. *2011 44th Hawaii International Conference on System Sciences*, 1–10. https://doi.org/10.1109/HICSS.2011.264

JetBrains. (2023). *The State of Developer Ecosystem 2023*. https://www.jetbrains.com/lp/devecosystem-2023

Krishnamurthy, S. (2006). On the intrinsic and extrinsic motivation of free/libre/open source (FLOSS) developers. *Knowledge, Technology & Policy*, *18*(4), 17–39. https://doi.org/10.1007/s12130-006-1002-x

Kuutila, M., Mäntylä, M. V., Claes, M., Elovainio, M., & Adams, B. (2018). Using experience sampling to link software repositories with emotions and work well-being. *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, 1–10. https://doi.org/10.1145/3239235.3239245

Miller, C., Cohen, S., Klug, D., Vasilescu, B., & KaUstner, C. (2022). "Did you miss my comment or what?": Understanding toxicity in open source discussions. *Proceedings of the 44th International Conference on Software Engineering*, 710–722. https://doi.org/10.1145/3510003.3510111

Moore, J. E. (2000). One road to turnover: An examination of work exhaustion in technology professionals. *MIS Quarterly*, *24*(1), 141. https://doi.org/10.2307/3250982

Qiu, H. S., Nolte, A., Brown, A., Serebrenik, A., & Vasilescu, B. (2019). Going farther together: The impact of social capital on sustained participation in open source. *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*, 688–699. https://doi.org/10.1109/ICSE.2019.00078

Raman, N., Cao, M., Tsvetkov, Y., Kästner, C., & Vasilescu, B. (2020). Stress and burnout in open source: Toward finding, understanding, and mitigating unhealthy interactions. *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: New Ideas and Emerging Results*, 57–60. https://doi.org/10.1145/3377816.3381732

Rubin, H. I., & Hernandez, E. F. (1988). Motivations and behaviours of software professionals. *Proceedings of the 1988 ACM SIGCPR Conference*, 62–71. https://doi.org/10.1145/57216.57225

Rutner, P., Riemenschneider, C., O'Leary-Kelly, A., & Hardgrave, Bi. (2011). Work exhaustion in information technology professionals: The impact of emotion labor. *ACM SIGMIS Database: The DATABASE for Advances in Information Systems*, *42*(1), 102–120. https://doi.org/10.1145/1952712.1952718

Sarker, F., Vasilescu, B., Blincoe, K., & Filkov, V. (2019). Socio-technical work-rate increase associates with changes in work patterns in online projects. *2019 IEEE/ACM 41st*

International Conference on Software Engineering (ICSE), 936–947. https://doi.org/10.1109/ICSE.2019.00099

Shih, S.-P., Jiang, J. J., Klein, G., & Wang, E. (2013). Job burnout of the information technology worker: Work exhaustion, depersonalization, and personal accomplishment. *Information & Management*, *50*(7), 582–589. https://doi.org/10.1016/j.im.2013.08.003

Singh, P., & Suar, D. (2013). Health consequences and buffers of job burnout among Indian software developers. *Psychological Studies*, *58*(1), 20–32. https://doi.org/10.1007/s12646-012-0171-9

Soelton, M., Lestari, P. A., Arief, H., & Putra, R. L. (2020). The effect of role conflict and burnout toward turnover intention at software industries, work stress as moderating variables. *Proceedings of the 4th International Conference on Management, Economics and Business (ICMEB 2019)*, 185–190. https://doi.org/10.2991/aebmr.k.200205.034

Sonnentag, S., Brodbeck, F. C., Heinbokel, T., & Stolte, W. (1994). Stressor-burnout relationship in software development teams. *Journal of Occupational and Organizational Psychology*, *67*(4), 327–341. https://doi.org/10.1111/j.2044-8325.1994.tb00571.x

Steinmacher, I., Graciotto Silva, M. A., Gerosa, M. A., & Redmiles, D. F. (2015). A systematic literature review on the barriers faced by newcomers to open source software projects. *Information and Software Technology*, *59*, 67–85. https://doi.org/10.1016/j.infsof.2014.11.001

Steinmacher, I., Wiese, I., Chaves, A. P., & Gerosa, M. A. (2013). Why do newcomers abandon open source software projects? *2013 6th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*, 25–32. https://doi.org/10.1109/CHASE.2013.6614728

Trinkenreich, B., Gerosa, M. A., Sarma, A., & Steinamcher, I. (2024). Guidelines for cultivating a sense of belonging to reduce developer burnout. *IEEE Xplore*. https://doi.org/10.1109/MS.2024.340436

Trinkenreich, B., Santos, F., & Stol, K.-J. (2024). Predicting attrition among software professionals: Antecedents and consequences of burnout and engagement. *ACM Transactions on Software Engineering and Methodology*, *33*(8), 1–45. https://doi.org/10.1145/3691629

Tulili, T. R., Capiluppi, A., & Rastogi, A. (2023). Burnout in software engineering: A systematic mapping study. *Information and Software Technology*, *155*, 107116. https://doi.org/10.1016/j.infsof.2022.107116

Van Oorschot, K. E., Sengupta, K., & Van Wassenhove, L. N. (2018). Under pressure: The effects of iteration lengths on agile software development performance. *Project Management Journal*, *49*(6), 78–102. https://doi.org/10.1177/8756972818802714

Appendix B: Rapid Thematic Analysis Themes and Sources

Themes

Themes and subthemes	**Illustrative quotes**	**Sources**
Difficulty getting paid ● reliable payment unrealistic prospect for most people ● have to be realistic: lack of money unsustainable, leads to giving up on 'dream' OS projects ● can only keep up OSS work if paid ● desire to maintain as day job if paid	*'[W]hen asked "What do you dislike about being an open source maintainer?" the response, "Not financially compensated enough/at all for my work" was the second most selected reason.'* *'The next most common reason for quitting or considering quitting included not getting paid enough to make it worthwhile, which rose from 32% to 38% of maintainers citing it in this year's survey versus our previous results.'* *'This isn't a lifestyle that's available to most people'* *'This makes any attempt to get money for tooling such an uphill struggle that it's really not worth the effort. Plans which are predicated on changing the world before anyone will pay you any money are decidedly bad plans.'* *'I feel that the main goal of the article (github maintaining balance for open source maintainers) is to convince maintainers to keep doing what they are doing for as long as possible, meaning to keep working for free. The article briefly mentions sponsoring but for most maintainers it's unrealistic to rely on sponsoring or donations.'*	Sonar (citing Tidelift report) Sonar (citing Tidelift report) Remy Sharpe David R. MacIver Artem Sapegin

• uncompensated, free labour • exploitation by the market and software companies • difficulty getting paid indicative of market failure	*'At some point you need sustainability, to either hire people to help or to enable you to do open source full time. If there had been something like Open Collective back then maybe I could have kept going, but I had to find other avenues.'*	Open Collective, (quoting Marc Grabanski, founder of Frontend Masters)
	'I was like, I'm going to keep this project, I'm going to keep working on this project. I fell in love with this project. But it didn't work out. I had to get a job.'	Ashley Williams
	'I think Hypothesis will make the world a better place, and I have a lot emotionally invested it … but I've really lost all desire to continue giving away so much of my labour for free, so I won't.'	David R. MacIver
	'There's something sadly ironic about developers trying to get hired elsewhere, just to sustain their work on OSS. If we talk about sustainable open source, this isn't it.'	Lars Kappert
	'Somewhere in the last decade, the corporate world discovered the wealth of free labour embedded in FOSS and has capitalized heavily on it. It's no longer devs sharing code with each other. A lot of these panicked demands for more free labour come from users in corporate context. Either putting pressure on maintainers at the direct request of their boss... or trickle-down pressure to finish job X which needs open source project Y.'	Artem Sapegin
	"I personally get regular demands for unpaid work (Discussions about payment for work always stall) by healthy high profit companies large and small for [my projects]. If I don't respond in a timely fashion, if I'm not willing to accept a crappy pull request, I/we get labeled a jerk. There is nothing like having core	Nadia Eghbal, Roads and Bridges (quoting Daniel Roy Greenfeld)

	Python/PyPA maintainers working for Redhat [sic] demanding unpaid work while criticizing what they consider your project's shortcomings to ruin your day and diminish your belief in open source.'	
	'Open source was about sharing the code with fellow developers, learning new skills, and having fun. Somehow, it became for many a threat to their mental health, and an unpaid job. Multi-million corporations take advantage of thousands of developers working for free around the globe. And on top of this, we have a generation of developers who demand that open source maintainers fix their issues for free.'	Artem Sapegin
	'Yeah, I could probably eke out a living. Particularly if I was prepared to burn a lot of bridges and sacrifice most of what actually makes me want to work on it, but basically we've built an industry on free labour, and we've concluded that we'd much rather make people work for free in their spare time to produce adequate software and shame them into supporting it when somehow it surprisingly doesn't do exactly what we want than fairly compensate for their labour and get good software out of it.'	David R. MacIver
	'There's an immense amount of money. The problem is that it's not evenly distributed. So a lot of people don't usually think about this when they think about in a market, but the health of a market—its whole purpose—is to distribute resources efficiently. So when I look at how much money there is in open source, and then how much struggle there is, I can't help but think this is a failed	Ashley Williams

	market. So I think we need to stop using this term, "open source sustainability" and we need to start saying, this is a market failure.'	
	'[T]he one that annoys me personally is Facebook buys Instagram for a billion dollars, and Instagram has not donated a cent or a line of code to Django or to Python which they built that on top of. There is so much money around open source, there is not a company out there that is not directly more profitable because they're using free software and yet very very few people actually are getting paid for their work on open source.'	Jacob Kaplan-Moss
	'Unfortunately, many big companies take and take without giving back in proportion to the value they receive, and these problems aren't just fixing themselves under capitalism. If you put purely economics first and aren't conscious of the things that have enabled your success, then you end up just hurtling towards darkness.'	Open Collective, (quoting Marc Grabanski, founder of Frontend Masters)
	'[I]t's really important that an independent organization with clear thinking towards sustainability of the commons, like Open Collective, continues to exist and thrive. I don't want to live in a world where only one or two companies have a monopoly on where we get our goods, software and knowledge. If we don't fund smaller initiatives and decentralized software, that seems to be where we're headed.'	Open Collective, (quoting Marc Grabanski, founder of Frontend Masters)
• lack of pay associated with burnout	*'Open Source has created an economic value vacuum. Our society depends utterly on the common pool resource of Open Source*	Chad Whitacre

	software, and this commons is severely underprovisioned. How do we know? The real indicator of the Open Source sustainability crisis as I define it is maintainer burnout.'	
	'many also feel like it adds to their stress and that they are not financially compensated for the work'	Chris Grams (citing Tidelift report)
	'There was a long time where I was doing open source almost more time than my full time job, and getting paid nothing. I just burnt out. I stopped writing and contributing to open source.'	Open Collective, (quoting Marc Grabanski, founder of Frontend Masters)
	'I wasn't getting paid to do the work that I felt like I had to do, and I had to do the work that I was getting paid to do because [of] mortgages and bills and car payments and that sort of stuff, so I had to do, like, double work, one out of a feeling of obligation and one out of a feeling of like, we live in a capitalist society and I need money.'	Jacob Kaplan-Moss
• not paying maintainers is a security risk, paying maintainers as insurance	*'In the Tidelift state of the open source maintainer survey, when asked to provide their reasons as to why they do not plan to meet industry standards, 38% of maintainers said they don't have the time, closely followed by 37% who said they weren't being paid to do it.'*	Sonar (citing Tidelift report)
	'I ultimately think–and this is maybe a little controversial–but I think most memberships in open source foundations are selling insurance.'	Ashley Williams

• payment for OSS polarising topic	*'There is a polarizing reaction to money within an open source project, specifically with regards to paying for a contributor's time.'*	Pia Mancini
	'The MIT license to a large extent is the anti-license. The utopia of socialized programs, one that embraces the lack of marginal cost for software goods. It's an explicit rejection of the strong-property rights approach taken by both Gates and Stallman at their respective ends of the libertarian spectrum. It's the language of giving without expecting anything in return. It's the language of sincere charity. A charity without strings attached, neither commercial nor reciprocal. With the risk of sounding sanctimonious, I read it as a pure projection of altruism'	David Heinemeier Hansen
	'I'm sharing details about my progress to hopefully popularize the model, and eventually help other maintainers adopt it, although I'm not quite ready to recommend anyone else drop everything to try this just yet.'	Filippo Valsorda
• getting paid, or being able to do OSS without getting paid, is a privilege	*'Honestly, I didn't mean to earn anything with any of this. My projects aren't extraordinary, I'm a developer in a safe and rich country, and so many other developers deserve at least the same. Being involved in OSS is simply a great way to keep learning and stay hirable with an open profile, which is an invaluable privilege in itself.'*	Lars Kappert

Workload and time commitment ● unable to enjoy free time, time with family and friends, time to choose direction of own life ● balancing time with other projects, time off now is more work later, more time if could do OSS as day job ● lack of sleep, all nighters, long shifts	*'The top reason why maintainers considered quitting was that other things in their life and work took priority (mentioned by 54% of respondents).'* *'I've already told my partner that, if and when we decide to start having kids, I will probably quit open source for good. I just can't see how I'll be able to make the time for both raising a family and doing open source. I anticipate that ultimately this will be the solution to my problem: the nuclear option. I just hope it comes in a positive form, like starting a new chapter of my life, and not in a negative form, like unceremoniously burning out.'* *'For me, my family comes first. Work and code isn't even a distant second. It's taken me many years of working silly hard and silly hours to work that out. Now that I understand that, life is better.'* *'You've considered doing open source as your day job, but from talking with folks who actually do open source for a living, you know that this usually means permission to work on a specific open-source project as your day job. That doesn't help you much, because you have dozens of projects across various domains, which are all vying for your time.'* *'The team is working around the clock … and my 6 a.m. to 4 a.m. (no, there is no typo in time) shift has just ended.'* *'I was doing nights and weekends, it was wrecking my health and I was just devastated, so after a while, enough issue follow-ups, like, "is this maintained?", "are you gonna fix my issue?", I had to say that I had to give it up.'*	Sonar (citing Tidelift report) Nolan Lawson Remy Sharpe Nolan Lawson Patrick Howell-O'Neill (quoting Volkan Yazici, Log4J member) Ashley Williams

	'the idea of a stealth job, when the work you're doing is masquerading as a hobby you don't realise you've added on suddenly 10, 15 20 hours of work a week on-top of your full-time job'	Kathleen Danielson
Maintenance can be unrewarding • maintaining is a different type of work/skill to coding • spend time responding to issues rather than creating • repetitive, needless, mind-numbing • lack of creativity/complexity is unrewarding • increasingly complex (JavaScript, tooling incompatibilities)	*'[P]eople are hard, code is easy to write, code you have a right answer and you have a wrong answer. People, relationships are hard and being able to work with other developers and work with other maintainers … can be very stressful'* *'I would rather be remembered as a bad artist than a good programmer. … now I'm asked more and more... to express myself less and to maintain the project more.'* *'I have so much work to do and I just need to focus on one thing and I need to not code. I need to focus on maintaining the project rather than coding the project.... It burns you out really, really easily.'* *'When you do too much of what you don't like, work gets depressing very fast.'* *'I only hope to have more time because I really like to build open source software.'* *'Open source maintainer burnout is a huge problem for any developer community. If the community is truly fatigued by a new set of tools every couple of years, then they need to help us fight this problem. Because in the end, this kind of criticism only ends up hurting the community.'*	Linus Torvalds Mat Asay (quoting Salvatore Sanfilipo, Redis founder) Mat Asay (quoting Jim Bailey, OBS Project founder) Artem Sapegin Nolan Lawson James Kyle

Community can be toxic ● users hostile, toxic, demanding, public shaming ● expectation without recognition: mistakes noticed but no recognition for good work ● people don't realise maintainers are human, uncompensated, overworked ● treatment affects self-esteem/self-worth ● burnt out makes OSS developers ruder ● toxicity on developer teams ● hard to speak up, especially if from group that is marginalsied or under-represented in tech ● isolation: work online, don't meet colleagues, lonely, no time to go out and socialise	*'[T]he angry response has been overwhelming. Every single day I'm reading someone else rant about how awful of a job we're doing. It's been hard to stay motivated—I've practically stopped looking at issues and pull requests'* *'[N]othing is stopping people to bash [sic] us, for work we aren't paid for, for a feature we all dislike yet needed to keep due to backward compatibility concerns.'* *'I don't want to help corporations make millions on free code, and receive rude comments instead of any kind of recognition.'* *'[A]lmost half of maintainers feel underappreciated and like the work is thankless'* *'Rewards can come in many shapes or forms. But whatever constitutes low rewards to you is what fuels burnout.* *'I just gave up, because what's the point when all you get is constant issues? You give and give and give, and people just take and take and take.'* *'That's the big thing for me ... It's knowing you did something for free, out of love, and there's an endless stream of people going 'more! more!' and getting angry when you won't accommodate their edge case.'* *'You may get emotional about your code. You may feel ashamed of what you did, and do. After all, your code has bugs, there are numerous issues opened at your bug tracker, and people are*	James Kyle Patrick Howell-O'Neill (quoting Volkan Yazici, Log4J member) Artem Sapegin Chris Grams (citing Tidelift report) Lars Kappert Open Collective, (quoting Marc Grabanski, founder of Frontend Masters) Nadia Eghbal, Roads and Bridges (quoting David Michael Ross) George Stavracas

	complaining non-stop. (Oh and, naturally, there will be someone who will try their best to put you down with that.)'	
	'We want to be seen and have our work acknowledged. We're humans after all.'	Karolina Szczur
	'I started identifying it as other people's problems, "if only these these idiots would stop submitting bad code that I would have to review, if only these these jerks would stop asking for help", like, it wasn't my fault that that I couldn't muster the energy to work on this, it was all these these stupid people out there and I started you know making it about about other people's problem.'	Jacob Kaplan-Moss
	'At first, I tried to deny my feelings and "just do my job". That lead [sic] to severe burnout and depression. In order to deny my feelings of powerlessness, my anger at the community, I had to stuff everything into an emotional box and close the lid. I usually follow a lot of technology activists, but in order to get my job done, I had to deny that the problem even existed.'	Anonymous Author
	'[W]hen asked "What do you dislike about being an open source maintainer?" the response, "It can be lonely," was the third most selected reason.'	Sonar (citing Tidelift report)
Responsibility ● people depend on you ● software popularity blessing that becomes a curse ● maintenance as a long-term commitment	*'[B]ut there are so many people out there that use (depend, even?) on the project for their apps, that I feel obligated to be a good steward of it.'*	Nadia Eghbal, Roads and Bridges (quoting Arash Payan)

• concern for community • guilty/letting community down • strain of responsibility for security, maintainer burnout is a security risk • single maintainers have sole responsibility • community support a blessed relief • hard to attract new contributors, maintainers are an aging population • if maintainer stops the project dies	*'Once something gets popular, you only hear from people who are having a problem with it. If you break anything - easy to do - you'll have many people suddenly upset at you.'* *'My open source success went from a major blessing to a great curse. It was one of the darkest times in my life. Something that started out with such hope and light ended up just being about getting thousands of emails. People told me their whole life stories and how it's all been leading up to this one feature they really need me to add.'* *'Open source is supposed to be our passion, something we work on even in our spare time. We talk about community, collaboration, and working on projects for the greater good. To walk away from an open source project for mental health reasons seems somehow selfish.'* *'He got bullied into giving maintainership to somebody else. And that person installed a backdoor into XZ, and it's so easy for that to happen.'* *'[M]aintaining compliance with security standards and regulations adds extra pressure to the role. In the Tidelift state of the open source maintainer survey, when asked to provide their reasons as to why they do not plan to meet industry standards, 38% of maintainers said they don't have the time, closely followed by 37% who said they weren't being paid to do it.'*	Juha Saarinen (quoting Dominic Tarr, maintainer of the event-stream module) Open Collective, (quoting Marc Grabanski, founder of Frontend Masters) Anonymous Author Ashley Williams Sonar (citing Tidelift report)

	'[M]any projects struggle to bring in new developers, limiting the project's growth and sustainability. In the same Tidelift survey, almost half of maintainers surveyed said they were a solo maintainer.	Sonar (citing Tidelift report)
	'[Y]ou can fork it if you want. And no one did. But I was incredibly sad, and the worst part about this is that it actually kinda haunted me for a while.'	Ashley Williams
	'[I]f the projects I created lives on without me, then open source has worked.'	Remy Sharpe
	'The moment I click that unsubscribe button on GitHub, the project essentially dies'	Artem Sapegin
	'I'm the only person holding this together, if I leave who will do this?'	Mikael Rogers

Pressure to prove yourself ● pressure to prove yourself: contribute more, work harder, know everything ● pressure to build an OSS portfolio to be employable, OSS as internship ● attention capture, GitHub complicit for gamifying contributions, hard to switch off, desire to go out in nature and live rustic life after burnout	*'[I]t starts kind with a compulsion to to prove oneself, I mean for me I felt like we had we had open source this thing and I wanted it to be really awesome and I needed to make this really great and the more I lost interest the harder I worked to try to like overcome it, you know, "this isn't fun anymore, but if I just do it harder I'll make it fun again"*	Jacob Kaplan-Moss
	'There is distortion in the web community that you need to learn more, contribute more, work more, know everything, be on the latest technology, and so on'	Remy Sharpe
	'Open Source often is a constant, vicious cycle of ego games. Ever-present expectations of performance and arbitrary success bring people to a breaking point. We confuse approval with love and self-worthiness, which becomes dependent on achieving. At the lowest levels of behaviour we engage in a phenomenon called impression management — we're always thinking about how we appear to others, even when no one's around.'	Karolina Szczur
	'It's all in the numbers — followers, contributions, comments, stars.'	Karolina Szczur
	'It's actually pretty typical for open-source work to help build a portfolio that then leads to paid jobs. In some ways the structure resembles unpaid internships in other industries—a system increasingly seen as unethical, exploitative, and unfairly advantageous to people who can afford to take on heaps of uncompensated work at the expense of those who cannot.'	Patrick Howell-O'Neill

Sources

Original articles by OSS developers

antirez. (2019). The struggles of an open source maintainer. <*antirez*>. https://antirez.com/news/129

Anonymous Author. (n.d.). Leaving Toxic Open Source Communities. *Model View Culture*. https://modelviewculture.com/pieces/leaving-toxic-open-source-communities

Boyter, B. E. C. (2019). *The Three F's of Open Source Development*. https://boyter.org/posts/the-three-f-s-of-open-source/

Caleb, P. (2020). *Introducing Sponsorware: How A Small Open Source Package Increased My Salary By $11k in Two Days*. Caleb Porzio. http://calebporzio.com/sponsorware

Cannon, B. (2024). *Saying thanks to open source maintainers*. https://snarky.ca/saying-thanks-to-open-source-maintainers/

Cannon, B. (2021). *The social contract of open source*. https://snarky.ca/the-social-contract-of-open-source/

Depierre, T. (2022). *I am not a supplier*. https://softwaremaxims.com/blog/Not-A-Supplier

Fu, A. (2022). *Why reproductions are required*. https://antfu.me/posts/why-reproductions-are-required

Fu, A. (2024). *Mental health in open source.* https://antfu.me/posts/mental-health-ossHansson, D. H. (2021). *I won't let you pay me for my open source*. https://world.hey.com/dhh/i-won-t-let-you-pay-me-for-my-open-source-d7cf4568

Kappert, L. (2024). *Privileged, but not a clown*. https://webpro.nl/articles/privileged-but-not-a-clown

Kyle, J. (n.d.). *Dear JavaScript*. https://jamie.build/dear-javascript.html

Lawson, N. (2017). *What it feels like to be an open-source maintainer*. https://nolanlawson.com/2017/03/05/what-it-feels-like-to-be-an-open-source-maintainer/

Lehnardt, J. (2017). *Sustainable Open Source: The Maintainers Perspective or: How I Learned to Stop Caring and Love Open Source*. https://writing.jan.io/2017/03/06/sustainable-open-source-the-maintainers-perspective-or-how-i-learned-to-stop-caring-and-love-open-source.html

MacIver, D. R. (2015). *Throwing in the towel*. https://www.drmaciver.com/2015/08/throwing-in-the-towel/

Mensching, R. (2024). *A Microcosm of the interactions in Open Source projects*. https://robmensching.com/blog/posts/2024/03/30/a-microcosm-of-the-interactions-in-open-source-projects/

ThePrimeagen. (n.d.). *My Burnout Experience—YouTube*. Retrieved 17 September 2025, from https://www.youtube.com/watch?v=KEWV5yaj_2o

Rogers, M. (2017). *Time to leave.* Medium. https://medium.com/@mikeal/time-to-leave-a68294ccb2af

Sapegin, A. (2018). *No complaints policy in open source by Artem Sapegin*. https://sapegin.me/blog/no-complaints-oss/

Sapegin, A. (2022). *Going offline*. Artem Sapegin. https://sapegin.me/blog/going-offline/

Sapegin, A. (2023a). *Healthier way to open source your code*. https://dev.to/sapegin/healthier-way-to-open-source-your-code-2el6

Sapegin, A. (2023b). *Why I quit open source*. DEV Community. https://dev.to/sapegin/why-i-quit-open-source-1n2e

Sharp, R. (2015). *'Why I don't like open source' – my thoughts*. https://remysharp.com/2015/01/09/dont-like-open-source

Stavracas, G. (2019). On Being a Free Software Maintainer. *Feaneron*. https://feaneron.com/2019/03/28/on-being-a-free-software-maintainer/

Sternberg, Daniel. (2025). Death by a thousand slops. https://daniel.haxx.se/blog/2025/07/14/death-by-a-thousand-slops/

Szczur, K. (2015). *The Dark Side of Open Source*. https://medium.com/@fox/the-dark-side-of-open-source-ba5a66c8a4c3

Valsorda, F. (2023). *I'm Now a Full-Time Professional Open Source Maintainer*. https://words.filippo.io/full-time-maintainer/

Conference talks by OSS developers

Michelle Bakels. (n.d.). *Tech Broke My Heart*. Retrieved 9 November 2025, from https://www.youtube.com/watch?v=vSXLDvvIpYE

Danielson, K. (2015). *Avoiding Burnout, and other essentials of Open Source Self-Care* [Video recording]. https://www.youtube.com/watch?v=RbeHBnWfXUc

Kaplan-Moss, J. (2015). What Part of '... For Life' Don't You Understand? [Video recording]. Confreaks. *https://www.youtube.com/watch?v=EqcuzSwySR4*

David Whitney (2024). Open-Source Exploitation [Video recording]. https://www.youtube.com/watch?v=9YQgNDLFYq8

Williams, A. (2024). 2 Wolves [Video recording]. RedMonk Tech Events. https://www.youtube.com/watch?v=gR2hjbh0RUk

Video and podcast interviews with OSS developers

Mastery Learning. (2024). *Linus Torvalds addressing maintainer fatigue* [Video recording]. https://www.youtube.com/watch?v=eWfUaFNSPhM

Stacoviak, A., & Santo, J. (2018). *A call for kindness in open source with Brett Cannon (Changelog Interviews #318)* [Broadcast]. https://changelog.com/podcast/318

SustainOSS. (2024). *Nolan Lawson of PouchDB on what it feels like to be a maintainer* [Broadcast]. https://podcast.sustainoss.org/252

Articles published by OSS advocacy bodies

Grabanski, M. (2019). *Frontend Masters: 'We teach open source, we use open source, and we support open source.'* Open Collective. https://blog.opencollective.com/frontend-masters/

Grams, C. (2025). *The open source maintainer community is getting grayer—DEV Community*. Tidelift. https://dev.to/tidelift/the-open-source-maintainer-community-is-getting-grayer-1gc2

Harbuz, V.-S. (2024). *Not Paying Open Source Maintainers Is Expensive*. Open Source Pledge. https://opensourcepledge.com/blog/not-paying-open-source-maintainers-is-expensive/

Open Collective. (2017, November 3). *SustainOSS: The Report*. Open Collective. https://blog.opencollective.com/sustainoss-the-report/

Sonar. (n.d.). *Who are open source maintainers?* Retrieved 1 September 2025, from https://www.sonarsource.com/learn/open-source-maintainers/

Sonar. (2023). *Maintainer burnout is real. Almost 60% of maintainers have quit or considered quitting maintaining one of their projects*. https://www.sonarsource.com/blog/maintainer-burnout-is-real/

Whitacre, C. (2017). *Your Company Should Probably Pay $2,000 per Person for Open Source*. Medium. https://gratipay.news/your-company-should-probably-pay-2000-per-person-for-open-source-9205443e209d

Whitacre, C. (2024c). *The Open Source Sustainability Crisis*. https://openpath.quest/2024/the-open-source-sustainability-crisis/

Hawrot Weigel, B. (2024). *The Silent Crisis in Open Source: When Maintainers Walk Away.* OpenSauced. https://opensauced.pizza/blog/when-open-source-maintainers-leave

Articles published by tech news sites

Asay, M. (2020). *What does an open source maintainer do after burnout?* InfoWorld. https://www.infoworld.com/article/2258588/what-does-an-open-source-maintainer-do-after-burnout.html

Joslyn, H. (2023). Open Source Needs Maintainers. But How Can They Get Paid? *The New Stack*. https://thenewstack.io/open-source-needs-maintainers-but-how-can-they-get-paid/

O'Neill, P. H. (2021). *The internet runs on free open-source software. Who pays to fix it?* MIT Technology Review. https://www.technologyreview.com/2021/12/17/1042692/log4j-internet-open-source-hacking/

Saarinen, J. (2018). *Unknown dev gets rights to popular module, adds crypto stealer*. IT News. https://www.itnews.com.au/news/unknown-dev-gets-rights-to-popular-module-adds-crypto-stealer-516106

Forum and email discussion by OSS community members

Bastien. (2025). *Open Source Challenges bzg/opensource-challenges* [Computer software]. https://github.com/bzg/opensource-challenges

Collin, L. (2022). *Re: [Xz-devel] XZ for Java*. Mail-Archive. https://www.mail-archive.com/xz-devel@tukaani.org/msg00567.html

Whitacre, C. (2024a). *Funding common pool resources · Issue #14 · chadwhitacre/openpath*. GitHub. https://github.com/chadwhitacre/openpath/issues/14

Whitacre, C. (2024b). *Post-scarcity economics · Issue #15 chadwhitacre/openpath*. GitHub. https://github.com/chadwhitacre/openpath/issues/15

Guides for OSS developers

Best Practices for Maintainers (n.d.). Opensource.Guide. https://opensource.guide/best-practices/

Cabunoc Mayes, A. (n.d.). *Maintaining Balance for Open Source Maintainers*. Open Source Guides. https://opensource.guide/maintaining-balance-for-open-source-maintainers/

Reports on burnout in OSS

Eghbal, N. (2016). *Roads and Bridges: The Unseen Labor Behind Our Digital Infrastructure*. Ford Foundation.

SustainOSS. (2017). *Sustain: A one day conversation for open source software sustainers*. SustainOSS. https://sustainoss.org/assets/pdf/SustainOSS-west-2017-report.pdf

Tidelift. (2024). *The 2024 Tidelift State of the Open Source Maintainer Report*. Tidelift.