



# Report on the 2020 FOSS Contributor Survey

The Linux Foundation &  
The Laboratory for Innovation Science at Harvard

**Frank Nagle**  
Harvard Business School

**David A. Wheeler**  
The Linux Foundation

**Hila Lifshitz-Assaf**  
New York University

**Haylee Ham**  
**Jennifer L. Hoffman**  
Laboratory for Innovation Science at Harvard

## Acknowledgments

This report and the research behind it would not have been possible without the leadership of the Core Infrastructure Initiative’s Advisory Committee, composed of Josh Corman, Steve Lipner, Audris Mockus, Henning Piezunka, and Sam Ransbotham. Frank Nagle would also like to thank his fellow co-directors of the Core Infrastructure Initiative, Jim Zemlin at the Linux Foundation and Karim Lakhani at the Laboratory for Innovation Science at Harvard, for their counsel and direction throughout this project. Gratitude and thanks to Michael Dolan and Kate Stewart at the Linux Foundation for their ongoing commitment to this undertaking. Thank you to James Dana for laying the initial groundwork for this survey.

Finally — and perhaps, most importantly — thank you to all the individuals who contribute to FOSS projects. Without their tireless efforts, our core digital infrastructure and the feats enabled by it would not be sustainable.

REVISED: This report has been updated since its original release on 8 December 2020. This second version, released on 10 December 2020, corrects errors found in the original text and graphics.

# Contents

<b>Executive Summary</b>	<b>4</b>
<b>Introduction</b>	<b>7</b>
<b>Methodology</b>	<b>9</b>
<b>Overview of Findings</b>	<b>10</b>
Demographics	10
Figure 1: Gender and Age of Respondents	10
Figure 2a: Geographic Location of Respondents by Country	11
Figure 2b: Geographic Location of Respondents by Global Region	11
Figure 3: Employer by Sector	12
FOSS at Work	13
Figure 4: Employment Status	13
Figure 5: Contributors Receiving Payment By Contributor Status per Project	14
Figure 6: Countries by Ratio of Paid Contributors to Total Respondents	14
Figure 7: Employer’s IP Policy Related to FOSS Contributions During Free Time	16
Current FOSS Contributions	17
Figure 8: Top Project Languages Reported by Respondent Pool	17
Motivations	19
Figure 9a: Contributor Motivations	20
Figure 9b: Contributor Motivations by Highest Reported Contributor Status & FOSS Experience	20
Figure 10: Future FOSS Contribution	21
Time Allocation	22
Figure 11: Hours per Week Spent on FOSS By Contributor Status	22
Figure 12: Percent of Hours Spent on FOSS Projects Occurring During Paid Work vs. Free Time	23
Figure 13: Time Spent on FOSS Now vs. 5 Years Ago for those with 5-10 Years Experience	24
Figure 14: Time Spent on FOSS Now vs. 5 Years Ago for those with 10+ Years Experience	24
Figure 15: Time Spent on FOSS Now vs. 10 Years Ago for those with 10+ Years Experience	25
Figure 16: FOSS Time Allocation: Actual vs. Ideal	26
Figure 17: FOSS Time Allocation: Maintainers/Core Contributors vs. Occasional/One-Time Contributors	26
Figure 18: FOSS Time Allocation: Paid vs. Unpaid Contributors	27
<b>High-Level Takeaways &amp; Suggested Actions</b>	<b>28</b>
1. Contributors’ Motivations	28
Figure 19: Participation of Core Participants & Maintainers Paid by their Employers in Projects Beyond the Initial 5 Identified	29
2. Need to Increase Security	31
Figure 20: Value of Contributions from External Sources	31
3. Contributions Linked to Employment	34
4. Corporate FOSS Policies	35
<b>Conclusion</b>	<b>36</b>
<b>A Thank You to FOSS Contributors</b>	<b>37</b>
<b>Endnotes</b>	<b>40</b>
<b>Appendix</b>	<b>42</b>

# Executive Summary

This report summarizes the results of a survey of free/open source software (FOSS) developers in 2020. The goal was to identify key issues in improving the security and sustainability of FOSS since the world now depends on it as critical infrastructure that underlies the modern economy. The survey was a collaboration between the [Linux Foundation's](#) Core Infrastructure Initiative (CII) and the [Laboratory for Innovation Science at Harvard](#) (LISH). This work has been recently incorporated into the [Open Source Security Foundation](#) (OpenSSF) working group on securing critical projects.

To capture a cross-section of the FOSS community, the research team distributed the survey to contributors to the most widely used open source projects (as determined by the previous "[CII Census II Preliminary Report — Vulnerabilities in the Core.](#)") and also invited the wider FOSS contributor community through an open invitation. The response distribution was usually similar between these two groups, though there were exceptions (e.g., different programming languages' prominence did vary). A total of 1,196 respondents filled out the demographic section and at least one question about current FOSS contributions, of whom 603 went through the entire survey.

Of the respondents, 27% were in the United States, 12% were in Germany, and almost 7% were in France. The rest were diversified from countries around the world (see [Figure 2a](#)). The vast majority of respondents, nearly 75%, are employed full-time. The bulk of respondents are employed in tech-related industries (61%), although there is representation from other industries, including Finance, Transportation, Construction, Real Estate, Educational Services, and Healthcare. The survey found that over half of all respondents are paid to contribute to FOSS, though this varied greatly by country.

Below are the key insights of the report with corresponding suggestions for action.

## 1. The top three motivations for contributors are non-monetary.

Non-monetary motivations — specifically adding a needed feature or fix, enjoying learning, and fulfilling a need for creative/enjoyable work — were most frequently ranked in respondents' top three motivations for contributing. Conversely, being paid to develop FOSS was the most likely motivation to rank in an individual's bottom three motivations, even for those who reported receiving payment for their contributions.

People need money to have food and a place to live. However, the overwhelming majority (74.87%) of respondents are already employed full-time, and more than half (51.65%) are specifically paid to develop FOSS. This observation must be tempered by remembering that this survey focuses on people, not projects. Some projects may not have anyone paid to contribute to them — even if they are critical and even if some of the contributors are being paid to work on other projects. Even though many contributors are paid for their work on some projects, it is possible that some critical projects could benefit from financial support for their contributors.

When asked the question, “What type of contribution from external sources would be most beneficial?”, the second most common answer was financial contributions. At first, this seems inconsistent with the low priority of payment to contributors. This seeming contradiction can be resolved by an understanding that the financial contributions could often be used in ways other than payment to contributors, such as paying for cloud build servers, travel funding, events, security audits, or other resources for the project community.

### **Suggested Actions:**

1. Recognize the value of the knowledge and skills that employees gain from contributing to FOSS.
2. Support the learning process for new contributors, e.g., by providing project demos and educational materials and free courses on best practices across all open source projects.
3. Balance creative and mundane tasks for all contributors to promote continued engagement through rewarding, fulfilling experiences.
4. Consider support options other than payment to contributors (e.g., security audit, computing resources, and travel) when providing financial support for FOSS projects.

## **2. There is a clear need to dedicate more effort to the security of FOSS, but the burden should not fall solely on contributors.**

All types of contributors reported they spend very little of their time responding to security issues (an average of 2.27% of their total contribution time) and reported that they do not desire to increase this significantly. When asked what would be the most beneficial contribution to their FOSS projects, survey participants pointed to bug/security fixes, free security audits, and simplified ways to add security-related tools to their CI pipelines (see [Figure 20](#)). Efforts focused on dramatically increasing the time current contributors spend on security are unlikely to be welcome; alternative methods for incentivizing security-related efforts should be considered.

### **Suggested Actions:**

1. Fund security audits of critical FOSS projects and require that the audits produce specific, mergeable changes.
2. Rewrite portions or entire components of FOSS projects prone to vulnerabilities to produce a substantially more secure result (e.g., contribute a rewrite in a memory-safe language).
3. Prioritize secure software development best practices.
4. Companies should make secure software development training a requirement for hiring or continued professional development for their paid FOSS developers.

5. Utilize badging programs, mentoring programs, and the influence of respected FOSS contributors to encourage projects and their contributors to develop and maintain secure software development practices.
6. Encourage projects to incorporate security tools and automated tests as part of their continuous integration (CI) pipeline; ideally as part of their default code management platform.

### **3. As more contributors are paid by their employer to contribute, stakeholders need to balance corporate and project interests.**

Over the past few years, there has been debate about the growing influence of money in the FOSS ecosystem. This survey shows that nearly half (48.7%) of respondents are paid by their employer to contribute to FOSS. Although the private sector's increasing role in paid contributions could increase FOSS stability and sustainability, concerns remain about what happens to projects if that support suddenly ceases.

#### **Suggested Actions:**

1. Allay concerns over corporate involvement in FOSS projects through greater transparency and clear commitments to support FOSS in general and specific FOSS projects for several years.
2. Incentivize paid contributors to dedicate time to mentoring new volunteer contributors.
3. Transfer FOSS projects to a foundation with neutral governance to ensure diversity of organizations and control.

### **4. Enhance the positive trend of corporate support for employees' contribution to FOSS.**

Companies' increasing openness towards their employees' involvement in FOSS is encouraging, but there is still work to do. Over 45.45% of respondents stated that they are free to contribute to FOSS without asking permission, compared to 35.84% ten years ago. However, a significant percentage of respondents reported that their firms have unclear policies (17.48%) or that they are unaware (5.59%) of the policies.

#### **Suggested Actions:**

1. Clarify policies on when and how employees can contribute to FOSS projects, ensure the policies are visible to employees, and encourage employees to engage in projects.
2. Promote contributions to FOSS projects' security improvements, either through individual employees engaging directly, or collaborative efforts such as the Open Source Security Foundation (OpenSSF).

# Introduction

Free and Open Source Software (FOSS) has become a critical part of the modern economy. It has been estimated that FOSS constitutes 80-90% of any given piece of modern software,<sup>1</sup> and software is an increasingly vital resource in nearly all industries. This heavy reliance on FOSS is common in both the public and private sectors,<sup>2</sup> and among tech and non-tech organizations alike.<sup>3</sup> Therefore, ensuring the health and security of FOSS is critical to the future of nearly all industries in the modern economy.

To better understand the state of security and sustainability in the FOSS ecosystem, and how organizations and companies can support it, the [Linux Foundation](#)'s Core Infrastructure Initiative (CII) and the [Laboratory for Innovation Science at Harvard](#) (LISH) collaborated to conduct a widespread survey of FOSS contributors as part of larger efforts to take a pre-emptive approach to strengthening cybersecurity by improving open-source software security. These efforts — recently incorporated into the [Open Source Security Foundation](#) (OpenSSF) working group on securing critical projects — aim to support, protect, and fortify open software, especially software that is critical to the global information infrastructure.

This survey's primary goal is to identify how best to improve the security, including the sustainability, of FOSS — especially the FOSS that is widely relied upon by the modern economy. Specifically, the survey seeks to help answer the question, "How can we better incentivize adequate maintenance and security of the most used FOSS projects?"

Importantly, in conducting this survey, the research team sought to take a holistic view of security. It captured more technical aspects of security and also considered the more human side. The survey included questions about contributor motivations and level of involvement, corporate involvement in FOSS, and the role of economic considerations in contribution behavior. Further, the methodology for recruiting survey participants emphasized contributors to FOSS projects that have been identified as widely used via previous research that culminated in the release of "[CII Census II Preliminary Report — Vulnerabilities in the Core.](#)"

The survey focused on the following topics:

- 1. Demographics:** What are the demographics of FOSS contributors? In particular, what are their gender, employment, and geographic location?
- 2. Motivations:** What are their reasons for starting, continuing, or stopping contributions to FOSS? How can projects keep contributors engaged, and do contributors feel that their employers or others value their work?
- 3. Pay:** How many FOSS contributors are paid for their work on FOSS? If paid, by whom (e.g., by employers and/or corporate sponsorship)? If they are not, does the lack of payment lead to significantly poorer security or sustainability?
- 4. Time Spent:** How much time do contributors spend contributing to FOSS, and how would they like to spend it? Is there an interest in increasing time spent on security issues?

5. **Aid:** What kinds of actions from external actors would help improve security (e.g., code contributions and/or money)?
6. **Current activity:** What kinds of security-related activities are already taking place in the FOSS projects represented by the respondents?
7. **Education/training:** How much education/training have FOSS contributors had in secure software development and operations? From which sources did they receive it?

The survey categorizes contributors into the following groups:

- **“Maintainers”** are package maintainers or software maintainers who are the final decision makers over all or portions of source code that goes into a build or release. Maintainers would likely also identify as a subset of core participants.
- **“Core participants”** may have been involved in the project since inception, joined later, and regularly participated in major discussions about project direction, and have significant ongoing roles in the work, possibly including accepting patches to the code base. Core participants may be referred to as “Committers” in a project community.
- **“Occasional participants”** would not normally participate in ongoing or weekly project discussions, but occasionally provide contributions over longer periods of time.
- **“One-time participant”** is someone who provides a specific set of suggestions or contributions and then exits involvement once their work is done; these are sometimes called “drive-by commits.”

The research team invited all kinds of contributors to participate in the survey, whether they contributed software code, documentation, or provided other kinds of contributions. Future iterations of the survey hope to get more specific information on the kinds of contributions provided and improve the display logic of questions that did not apply to their kind of contributions.



# Methodology

This survey aimed to identify how to improve security, including the sustainability, of FOSS critical to the global information infrastructure. A review of other existing surveys of open source contributors revealed a lack of data on employment-related contributions, current security practices, as well as time and task allocation. The research team developed this survey to shine a light on these lesser-understood aspects of the open source community.

To capture a cross-section of FOSS contributors' extensive community, the research team distributed the survey via two separate methods. Building on the findings of "[CII Census II Preliminary Report — Vulnerabilities in the Core](#)," a list of the most widely used open source projects was compiled to create a targeted sample group to which emails were sent inviting those projects' contributors to participate in the survey. This initial distribution offered respondents the opportunity to send a link to the survey to other FOSS contributors. The Linux Foundation and LISH also undertook a marketing campaign to advertise the survey to the open source community at large. This included media outreach and press releases, a video appeal to contributors on The Fourth Industrial Revolution, pushes on social media platforms like LinkedIn and Twitter, as well as guest appearances on various open source-related podcasts, including [CHAOSScast](#). Additionally, the Linux Foundation added appeals to participate in the survey in two of their monthly newsletters and a direct email to over 250 project mailing lists.

This two-pronged approach resulted in 1,866 responses to the survey, though many responses were only partially complete — likely due, in part, to the length of the survey and the depth of the questions it asked. All questions (except a respondent's geographical location) were optional by design, and not all participants chose to answer each question they were presented. Not all participants saw every question in the survey as display and skip logic ensured that respondents did not see questions unrelated to their stated roles, contributions, and experience levels. For example, a participant who responded that they were unemployed did not see subsequent questions related to work-related FOSS contributions.

Therefore, to obtain the most value out of the results, the analysis focused on the answers individuals provided, even if they did not complete the entire survey. The analysis included all responses from the 1,196 participants who completed the demographic questions and at least one question about contributions. Thus, the number of answers varies from question to question, but each respondent pool is detailed under each question in the separate [Appendix](#), which also includes the aggregate participant responses for all questions that were asked.

# Overview of Findings

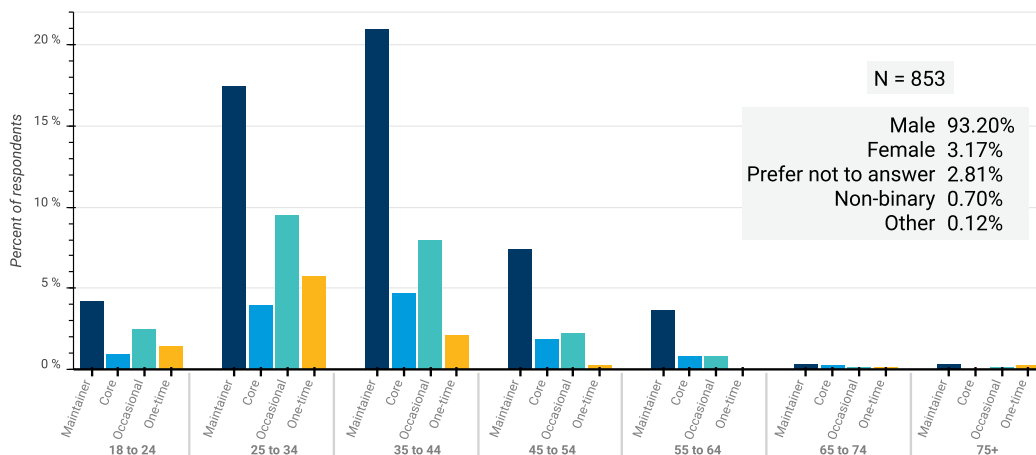
The FOSS Contributor Survey went in-depth on various dimensions about who contributors are, why they contribute, and how they approach security in their FOSS projects. This section highlights some of the most interesting results, while the complete survey results can be found in the [Appendix](#), which go far beyond the results discussed in this section. This section's results fall into the following categories: demographics, FOSS at work, current FOSS contributions, and time allocation.

## Demographics

The survey began with basic demographic questions about the participants responding. Figures 1 and 2 show the gender, age, and geographic location of the survey respondents. Figure 1 breaks down respondents by their age and their self-reported highest level of contribution to any project (e.g., if a respondent reported they are a Maintainer for one project and an Occasional participant for four projects, they would be classified as a Maintainer).

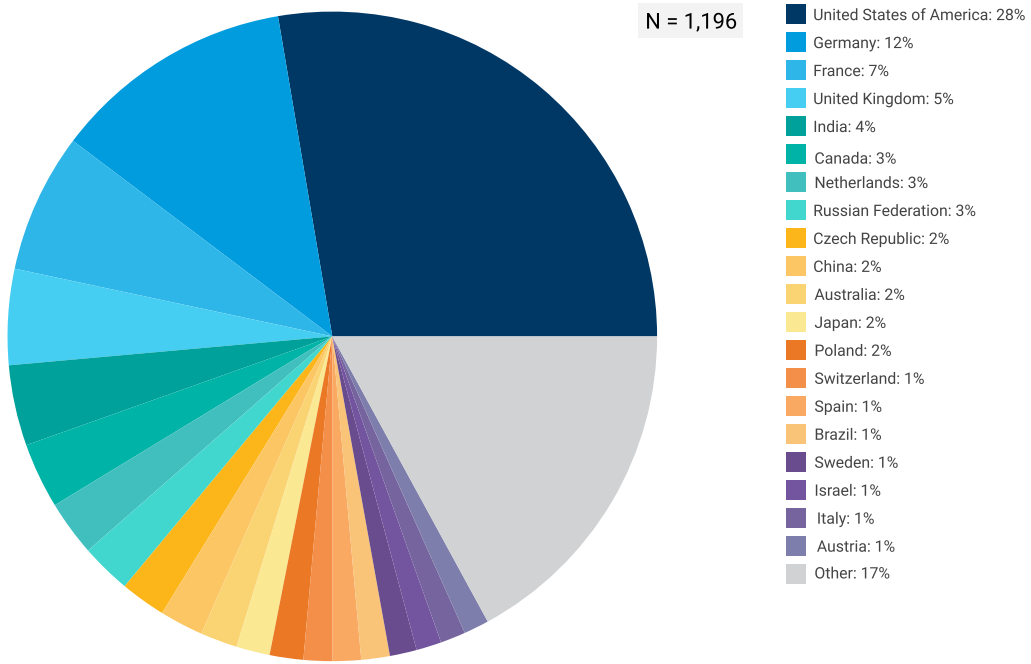
The majority of respondents were male and between 25 and 44 years old. The fact that 91% of respondents reported being male emphasizes the continuing concerns about a lack of female representation in FOSS communities.<sup>4</sup> It further raises possible concerns that these results are biased towards male contributors' FOSS activities and are not fully representative of female contributions to FOSS.<sup>5</sup> The geographic distribution of respondents in Figure 2 indicates that, although nearly a quarter of respondents were located in the United States, most were located elsewhere. There is particularly strong representation from France and Germany, two countries that have long been significant contributors to FOSS. The majority of respondents were from North America or Europe, indicating the results throughout the survey may better represent FOSS contributors' experiences in those areas than contributors in, for example, Asia.

Figure 1: Gender and Age of Respondents



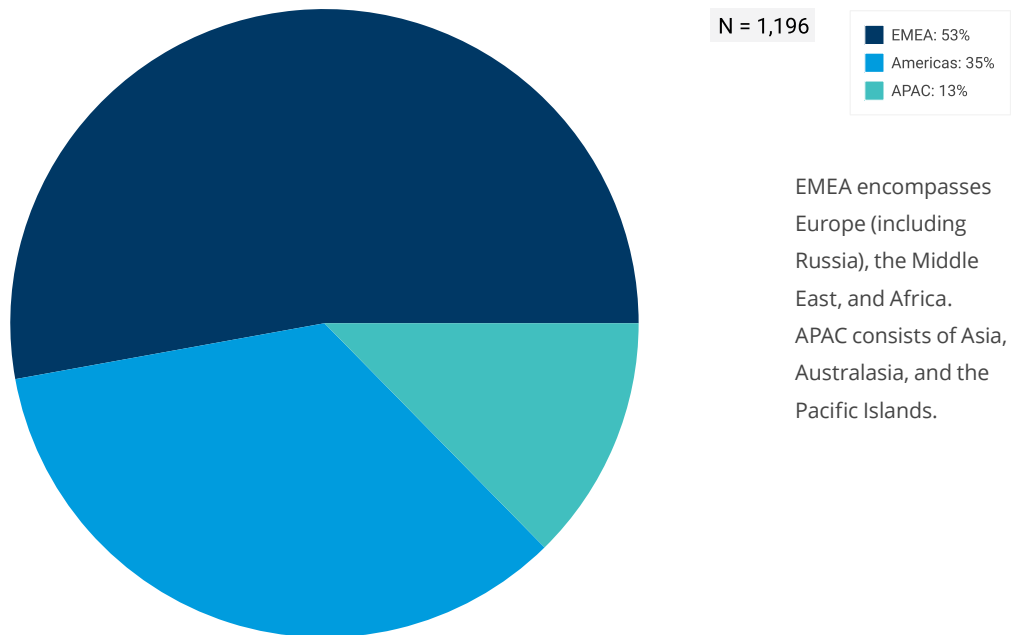
Please note that the gender breakdown for this figure represents a subset of the overall respondents who also answered questions about their age and contributor status. It differs slightly from the gender breakdown of the entire sample of responses seen in the text above and in the Appendix.

Figure 2a: Geographic Location of Respondents by Country



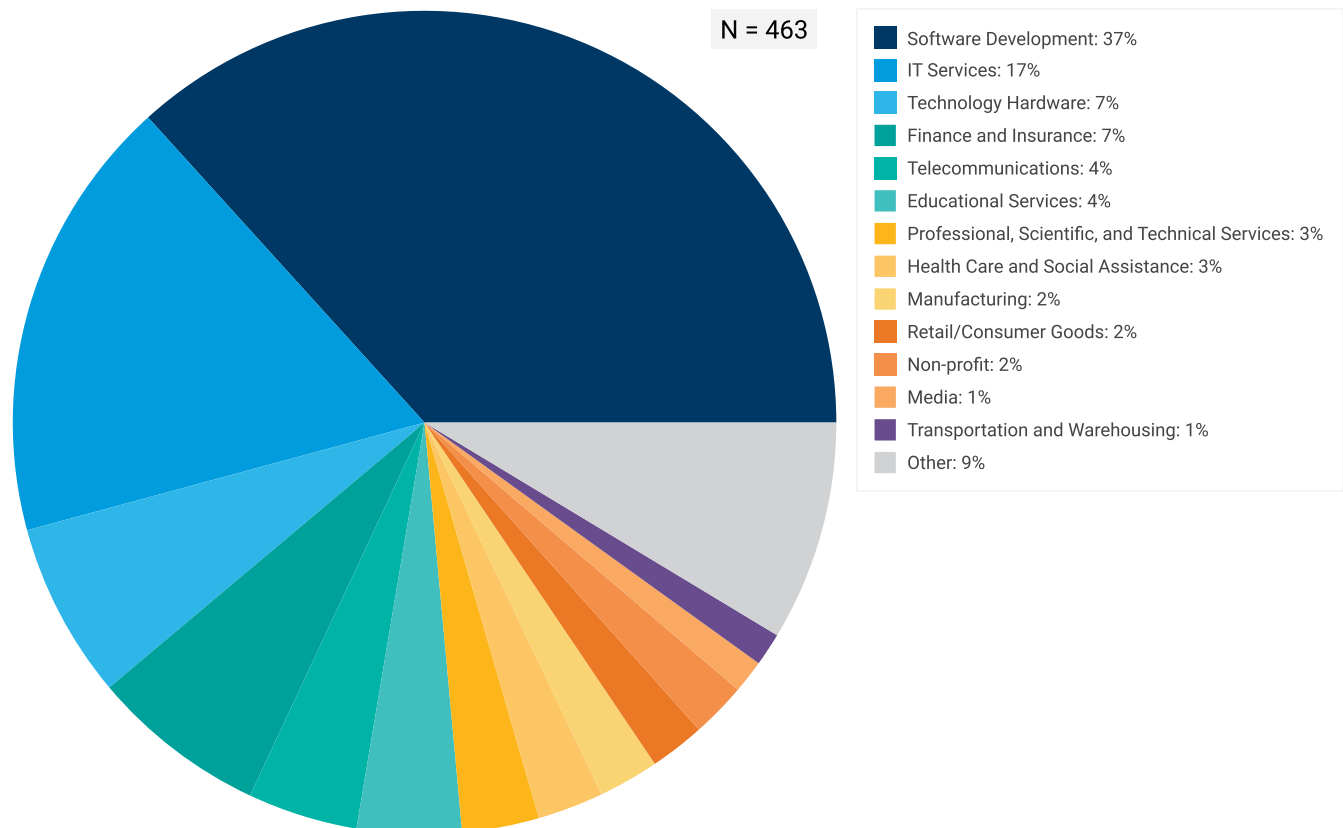
Countries in the “Other” category include: Algeria, Argentina, Bahrain, Bangladesh, Belarus, Belgium, Bulgaria, Cameroon, Chile, Colombia, Croatia, Denmark, Ecuador, Egypt, El Salvador, Estonia, Finland, Georgia, Greece, Honduras, Hong Kong, Hungary, Indonesia, Iran, Ireland, Israel, Italy, Kenya, Latvia, Lithuania, Madagascar, Malaysia, Malta, Mexico, Morocco, New Zealand, Niger, Nigeria, North Korea, Norway, Pakistan, Peru, Portugal, Romania, Saudi Arabia, Singapore, Slovakia, Slovenia, South Africa, South Korea, Sri Lanka, Swaziland, Syria, Thailand, Turkey, Ukraine, United Arab Emirates, Uruguay, Venezuela, Vietnam, Zambia, and Zimbabwe.

Figure 2b: Geographic Location of Respondents by Global Region



In addition to gender, age, and location, the survey also asked participants about their employer’s industry. Figure 3 shows the responses. Perhaps unsurprisingly, most respondents are in tech-related industries (Software, Hardware, IT Services, and Telecommunications). However, there is substantial representation from other sectors, including Finance, Transportation, Construction, Real Estate, Educational Services, and Healthcare.

Figure 3: Employer by Sector



*Sectors in the “Other” section of the pie chart include: Accommodation and Food Services; Administrative and Support Services; Agriculture, Forestry, Fishing and Hunting; Arts, Entertainment, and Recreation; Construction; Energy and Utilities; Mining; Other; Public Administration; and Real Estate Rental and Leasing.*

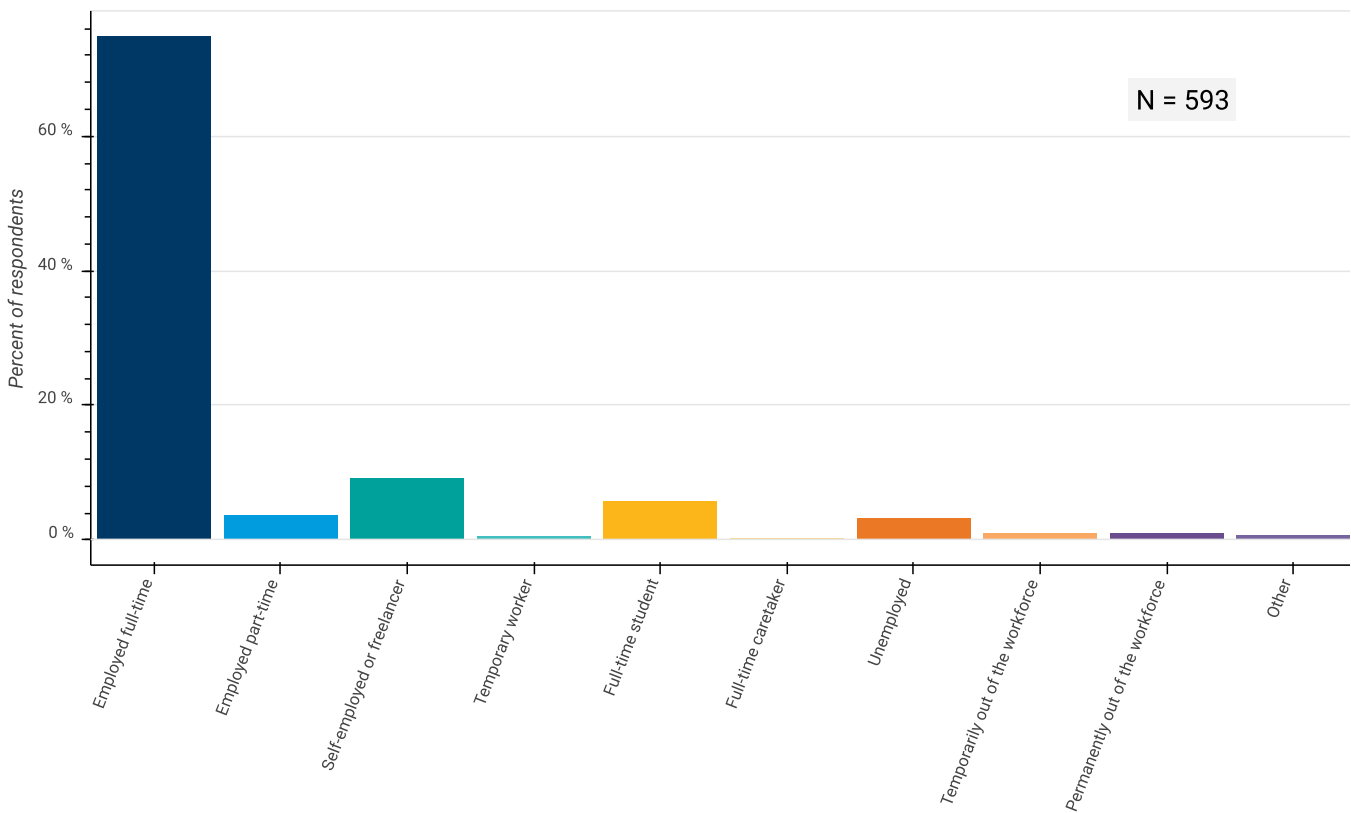
Finally, the survey asked participants about any training they had in software development, and 86.3% reported receiving formal training in software development. However, only 39.8% reported formal training in *secure* software development.

## FOSS at Work

A consistent source of debate about the future of FOSS revolves around the role of money and corporate involvement in the ecosystem. If FOSS contributors cannot support themselves (housing, food, etc.), then they are unlikely to work on FOSS. However, there is also debate around whether or not FOSS contributors are, or should be, compensated directly for their efforts. The survey asked respondents a variety of questions about their employment status, payment for contributions, and their employer's policy towards FOSS.

Figure 4 shows the employment status of the survey respondents. The overwhelming majority are employed full-time. The next two most popular answers were self-employed/freelancer or full-time student. This makes sense as most of the skills necessary to contribute to FOSS are highly valued in today's job market (programming, technical documentation, etc.). Despite the survey being administered during the economic downturn resulting from the COVID-19 pandemic, very few respondents were out of the workforce.

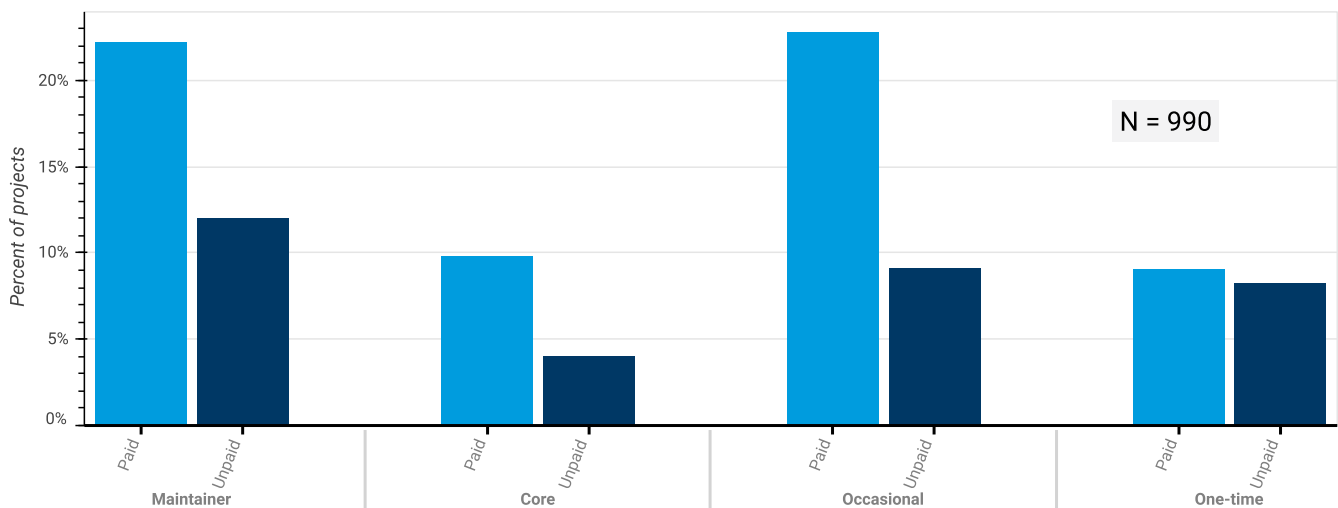
Figure 4: Employment Status



Beyond whether or not the respondents were employed, the survey also aimed to understand whether they were directly compensated for their FOSS efforts. In aggregate, of 577 survey respondents, 48.7% said they are paid for time spent on open source contributions by their current employer, 2.95% said another party pays them, 4.33% said they are not paid because their employment contract prevents them from accepting payment for open source development, and 44.02% said they are not paid for any other reason. Interestingly, most of those paid for working on a FOSS project also contributed to other FOSS projects without being compensated. Further, whether or not a respondent was paid was related to their role in a given FOSS project.

Over half (51.65%) of respondents reported that they receive payment for their FOSS contribution from either their employer or a third party.

Figure 5: Contributors Receiving Payment By Contributor Status per Project



Thus, Figure 5 shows whether or not a survey respondent was paid for their work on a specific FOSS project, as well as their level of contribution to that project. Therefore, in Figure 5, if a respondent contributed to two projects, one where they were paid and a maintainer, and the other where they were unpaid and only an occasional contributor, they would show up twice since the data is at the project level. For simplicity, in most other analyses in this report, a contributor is considered to be a “paid contributor” if they are paid for any projects they work on and an “unpaid contributor” if they are not paid for any of their work.

Figure 6: Countries by Ratio of Paid Contributors to Total Respondents

Country	% of Respondents Paid for FOSS	Total Responses
United States of America	63.8%	174
Germany	58.7%	75
France	37.1%	35
United Kingdom	42.9%	28
Canada	57.1%	21
Netherlands	75.0%	20
India	15.8%	19
China	29.4%	17
Austria	63.4%	11
Brazil	45.5%	11
Japan	45.5%	11
Australia	30.0%	10
Other	43.5%	145

For example, Figure 6 considers the percent of respondents that are paid for any of their FOSS contributions to better understand differences across countries in their likelihood to be paid. It is interesting to note the wide variance in the percentage of respondents that are paid for their contributions across countries. However, it should be noted that some of the countries have few respondents, making the findings difficult to generalize.

Importantly, this survey focuses on people, not projects. Some projects may not have anyone paid to contribute to them, even if they are important and even if some of the contributors are being paid to work on other projects. In short, even though many contributors are paid, it is possible that some of the critical projects they work on do not receive financial support or that other contributors working alongside them are still unpaid.

As the analysis below on contributors' motivations illustrates, financial motivation often is not the critical factor spurring FOSS contribution. However, the role of financial incentives in the FOSS ecosystems is difficult to unravel.

More than half (56.12%) of the respondents said that involvement in FOSS projects was moderately important, very important, or extremely important in getting their current job. A sizable minority (32.81%) of respondents reported that involvement in FOSS projects was "not important at all" in getting their current job. Similarly, 54.82% believed that their participation in FOSS projects has positively impacted their salary or job prospects.

Benefits also flowed to employers. Among the respondents, 81.41% reported that the skills that they acquired from working on FOSS were "valuable" or "very valuable" for their current employer. Respondents said that they learned how to write and test production-level code, organize asynchronous work as a team, and communicate and collaborate with a team. Respondents also said that they learned how to perform code reviews, how to use version control, and that they benefit from clean code being prioritized over "ugly workarounds." Respondents benefited from learning about "workflows, issue tracking and resolution, bug

reporting, timezone distributed work, gaining trust in developer communities.” They also said that learning about FOSS libraries helped them know what solutions exist when they are writing code.

Finally, the research team sought to understand how the stance of the respondents’ employers towards FOSS contribution to projects that are unrelated to their work during their free (non-work) time has changed over the past decade. Therefore, the survey asked respondents about the status of contributing to non-work related FOSS projects during their free time in the intellectual property policy of their employer today, five years ago, and ten years ago.

Figure 7: Employer’s IP Policy Related to FOSS Contributions During Free Time

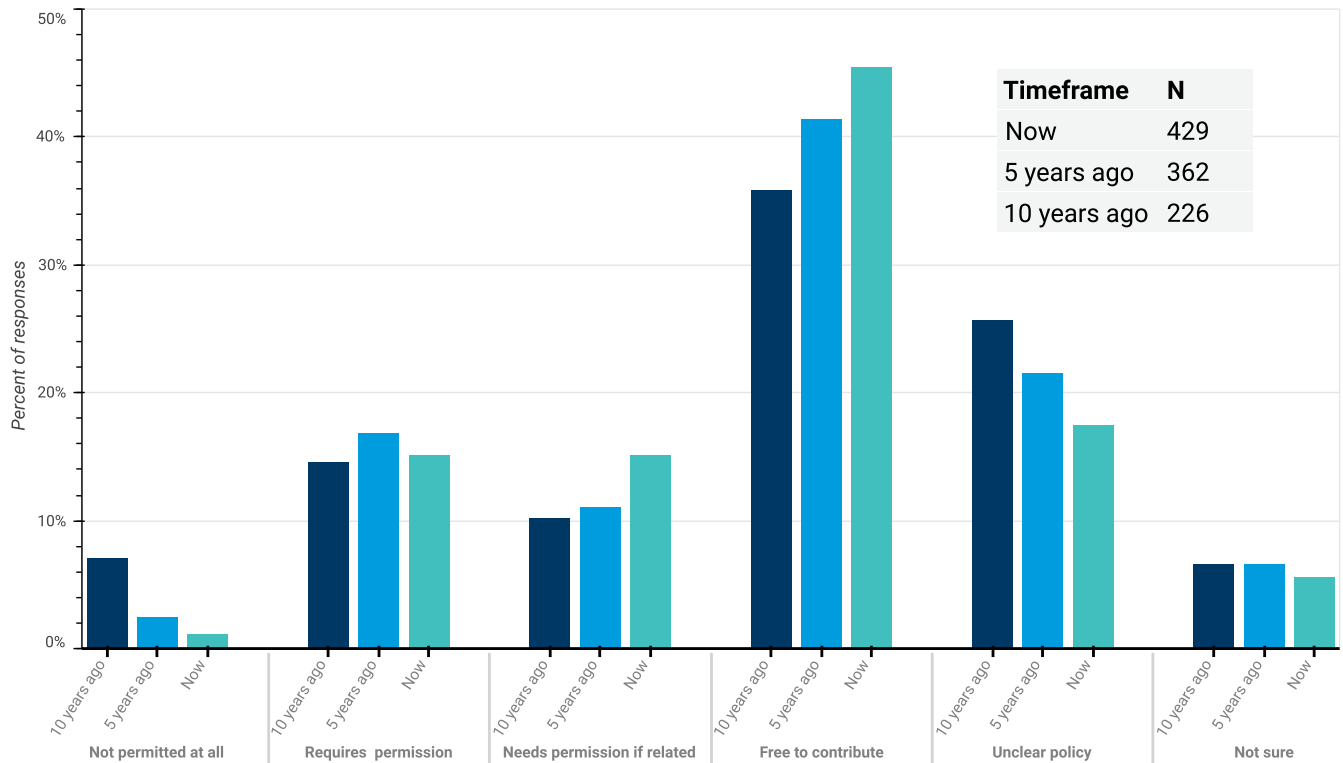


Figure 7 shows the results of these questions. Although it is encouraging that policies preventing individuals from contributing have gone down over the last ten years and that policies saying individuals are free to contribute have gone up, there is still a great deal of uncertainty related to employers’ contribution policies. As the figure below shows, 17.48% of respondents say their current employer does not have a clear policy towards FOSS contribution, while 5.59% say that they are unsure what the policy is.



## Current FOSS Contributions

The survey asked respondents about various aspects of their present (and future) contributions to FOSS. In particular, questions were asked about the projects they contributed to, their motivations for contributing, and their likelihood of contributing in the future.

As mentioned above, survey respondents came from the pool of targeted contributors based on [findings](#) related to the most widely used FOSS projects or the open survey that any FOSS contributor could take. In general, responses across these two groups were fairly consistent. However, because the prior research findings had greater representation from projects written in Java and JavaScript, there is a noticeable skew towards those projects from that group.

Figure 8: Top Project Languages Reported by Respondent Pool

Language	# of Projects (Invited)	Language	# of Projects (Open Survey)
Java	351	C	335
JavaScript	275	Python	221
C#	146	C++	134
Scala	94	Go	126
C++	83	JavaScript	109
TypeScript	81	Java	106
Python	64	Shell	70
Go	62	PHP	40
C	50	HTML	31
Shell	35	Rust	25
Ruby	26	Typescript	22
HTML	21	Ruby	21
PHP	14	C#	16
Kotlin	14	Perl	14
Haskell	13		
Rust	9		

Figure 8 shows the top languages of projects survey respondents contribute to, broken into these two groups. Importantly, the survey asked respondents to list up to five projects they contributed to, so the total number of projects represented is greater than the number of respondents. Figure 8 shows that despite the skew towards Java and JavaScript projects for the direct distribution (invited) respondents, there is a heavy representation from projects in numerous languages in both respondent groups. The diversity of languages used by respondents reflects the diversity of languages used in FOSS projects.

The survey respondents indicated that many communication channels are used by FOSS projects. The most common ones were issue trackers (90.35%), mailing lists (56.80%), and instant messenger applications like Slack and IRC (51.10%). Nearly half (47.03%) have never met face-to-face with their FOSS project partners.

Respondents also indicated how important some characteristics were when considering whether or not to contribute to a FOSS project. A majority (57.89%) said the presence of an open source license was extremely important, with another 25.53% stating it was very important (for a total of 83.42%). This suggests that FOSS projects should make it extremely easy for potential contributors to identify their license. Three other characteristics were considered extremely important or very important by a majority of FOSS developers: responsive maintainers (80.39%), active development (65.75%), and a welcoming community (63.19%). Other characteristics that many consider extremely important or very important (in order) are widespread use (41.2%), presence of a contributing guide (39.91%), employer uses/encourages contributions to it (37.58%), and being neutrally governed (27.69%). FOSS projects would be wise to implement all of these where they can.

Many considered the presence of a Code of Conduct important. It was considered extremely important (8.93%), very important (15.26%), or moderately important (21.59%) by 45.78% of respondents. However, 3.73% tend to avoid projects that have one. It's useful to understand that far more respondents found them important even though they can raise controversy.

The most common response to asking about the importance of a developer's certificate of origin (DCO) was that they did not know what it is (32.73%). DCOs have potential advantages, but many contributors do not know what they are; there may need to be an effort to explain them.

Respondents also noted that they tended to avoid contributing to projects with certain characteristics. The top characteristics they most avoided were projects with a contribution agreement to a for-profit organization (23.16%), a contributor license agreement aka CLA (15.33%), and a contribution agreement to a non-profit organization (10.44%). FOSS projects should consider avoiding these characteristics when practical, especially the first one, as they reduce the potential number of contributors (and thus may put the project at risk).

## Motivations

In asking survey respondents about their motivations for contributing to FOSS, the research team drew upon the extensive academic literature<sup>6</sup> on this topic and presented respondents with the following ten motivations and asked them to rank order them based on importance to why they contribute:

- I enjoy learning
- I am paid to develop FOSS
- I value the recognition of my peers
- Contributing allows me to fulfill a need for creative, challenging, and/or enjoyable work
- I use this piece of FOSS and needed the specific features/fixes I added
- Since I use FOSS, I feel I should contribute back to it
- I believe in the mission of FOSS or the particular area I contribute to (e.g., privacy software)
- I expect my contributions will help me advance my career
- I enjoy working with my peers and my community
- I enjoy helping others

These motivations were displayed in a random order to each respondent to limit possible bias.<sup>7</sup> Figure 9a shows this question's results by focusing on the motivations most commonly included in respondents' top three or bottom three responses.<sup>8</sup> It is interesting to note that all three of the top three responses are non-monetary motivations, while all three of the bottom three responses are extrinsic motivations. The need for a feature or fix and the enjoyment of learning were the two most commonly occurring motivations in respondents' top three choices. Conversely, being paid to develop FOSS was the most likely motivation to show up in an individual's bottom three choices. Since many contributors are not paid to contribute to FOSS, it makes sense that this would be at the bottom of the list. However, even when analysis split out the motivations by contributors that report being paid for FOSS contributions versus those that are unpaid (Figure 9b, middle two columns), payment is still close to the bottom of the motivations.

Figure 9b also shows the results of this question broken out by whether or not a respondent says their maximum level of involvement in any of the projects they contribute to is as a Maintainer or Core Developer versus an Occasional or One-Time Developer (first two columns) as well as by the number of years they have been contributing to FOSS (final two columns). In both tables in figure 9b the cell with the most common answer in its column has the darkest shade, while the least common has the lightest shade. Throughout all of these subsample analyses, the results are fairly consistent with the full sample results. Notable exceptions are that Occasional/One-Time developers and those only contributing to FOSS for five years or less are more likely to say that since they are a user of a FOSS project, they believe they should contribute to it (reciprocity) is a larger motivation than their need for creative work. Further, paid contributors are more likely to rank their belief in the FOSS mission as an important motivator than the need for creative work. Importantly, research<sup>9</sup> has shown that a contributors' motivations can change over time, and thus these sub-sample analyses may reflect shifting motivations as a user gets more involved in the community.

Figure 9a: Contributor Motivations

Most frequently found in respondents' top three		# of Times Ranked in Top 3
Motivation		
I use this piece of FOSS and needed the specific features/fixes I added		278
I enjoy learning		276
Contributing allows me to fulfill a need for creative, challenging, and/or enjoyable work		222
Since I use FOSS, I feel I should contribute back to it		205
I believe in the mission of FOSS or the particular area I contribute to (e.g. privacy software)		184
I enjoy helping others		154
I am paid to develop FOSS		99
I enjoy working with my peers and my community		89
I value the recognition of my peers		38
I expect my contributions will help me advance my career		83

Most frequently found in respondents' bottom three		# of Times Ranked in Bottom 3
Motivation		
I am paid to develop FOSS		326
I expect my contributions will help me advance my career		252
I value the recognition of my peers		216
I enjoy working with my peers and my community		133
I believe in the mission of FOSS or the particular area I contribute to (e.g. privacy software)		128
I use this piece of FOSS and needed the specific features/fixes I added		123
I enjoy helping others		117
Since I use FOSS, I feel I should contribute back to it		110
Contributing allows me to fulfill a need for creative, challenging, and/or enjoyable work		96
I enjoy learning		62

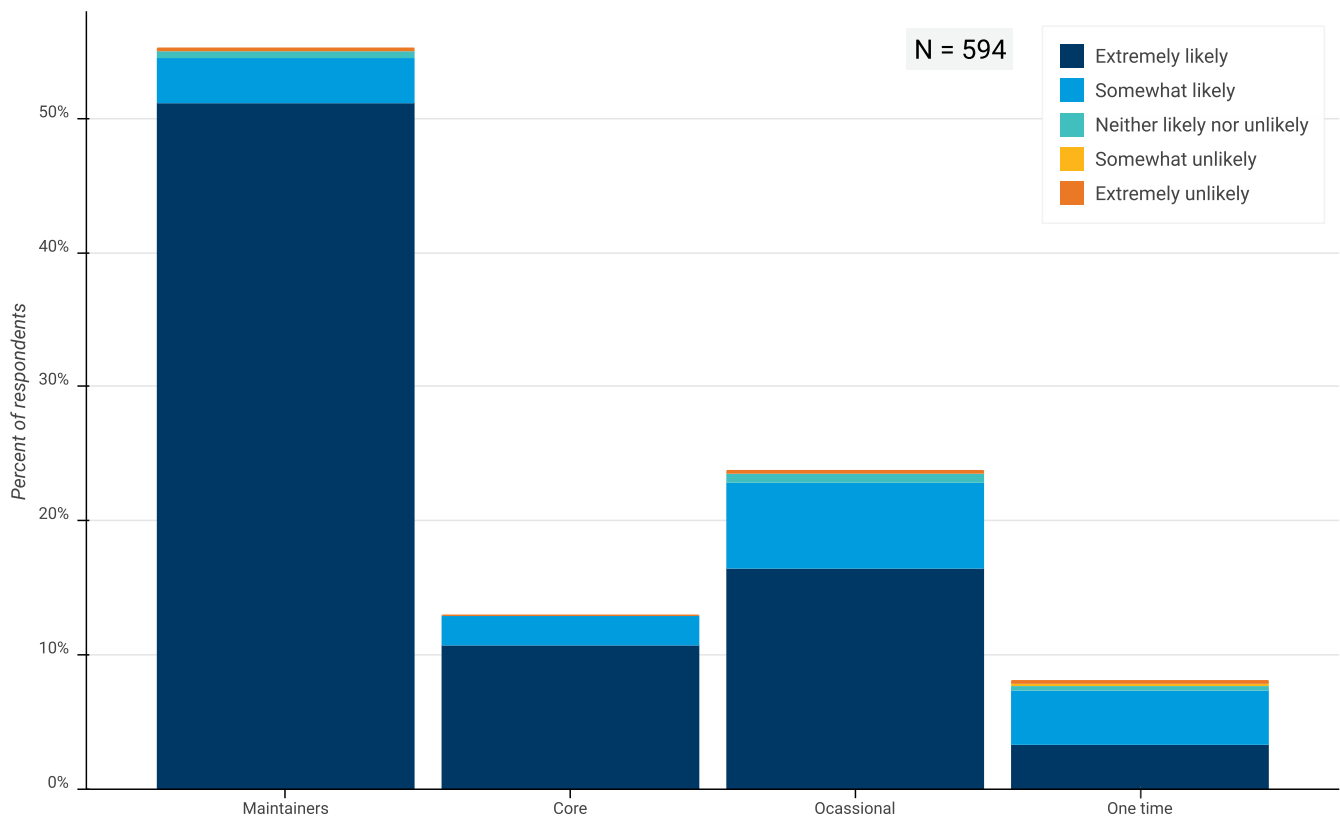
Figure 9b: Contributor Motivations by Highest Reported Contributor Status & FOSS Experience

# of Times Ranked in Top 3	Maintainers / Core	Occasional / 1X	Paid Contributors	Unpaid Contributors	≥10 years in FOSS	≤5 years in FOSS
Enjoy learning	158	107	115	137	120	69
Needed specific features	178	92	146	119	145	47
Need for creative work	160			103	126	
User, feel I should contribute		72				40
Believe in mission of FOSS			106			

# of Times Ranked in Bottom 3	Maintainers / Core	Occasional / 1X	Paid Contributors	Unpaid Contributors	≥10 years in FOSS	≤5 years in FOSS
Paid	188	127	117	189	156	79
Career advancement	172	74	129	115	157	
Peer recognition	147	63	123	82	117	39
Needed specific features						40

Finally, the survey asked respondents how likely they are to contribute to FOSS in the future. Overwhelmingly, respondents said that they are “Extremely Likely” to keep contributing, showing a strong commitment to the FOSS ecosystem (Figure 10). Interestingly, when breaking out responses based on the highest level of contribution to FOSS projects (Maintainers, Core Developers, Occasional Developers, or One-Time Developers), higher contribution levels are related to a higher likelihood of saying “Extremely Likely.” Survey participants who had ever stopped contributing to FOSS (temporarily or permanently) were asked why they had done so. Lack of time (due to professional, family, or social commitments) was the most common reason selected. Additional responses to this question can be seen in the detailed answers for question 9 in the [Appendix](#).

Figure 10: Future FOSS Contribution

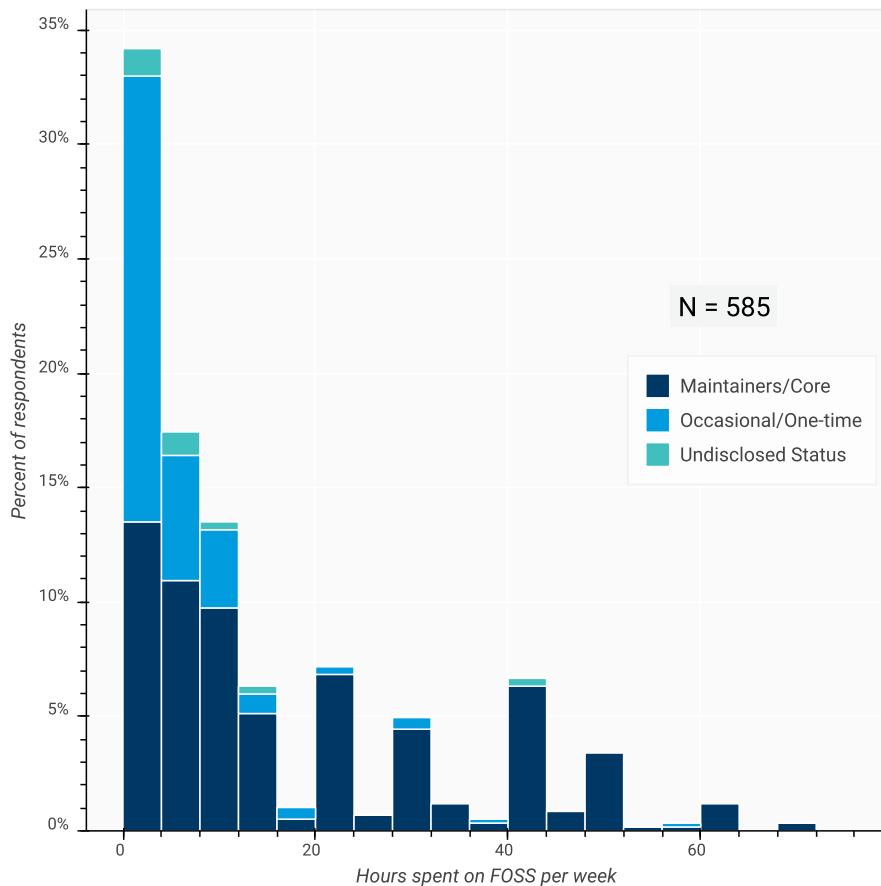


# Time Allocation

To better understand how much time respondents spend on contributing to FOSS, the survey included various questions related to time allocation, including how they spent their time in the past, what FOSS tasks they spend their time on, and how the coronavirus pandemic impacted their time spent on FOSS.

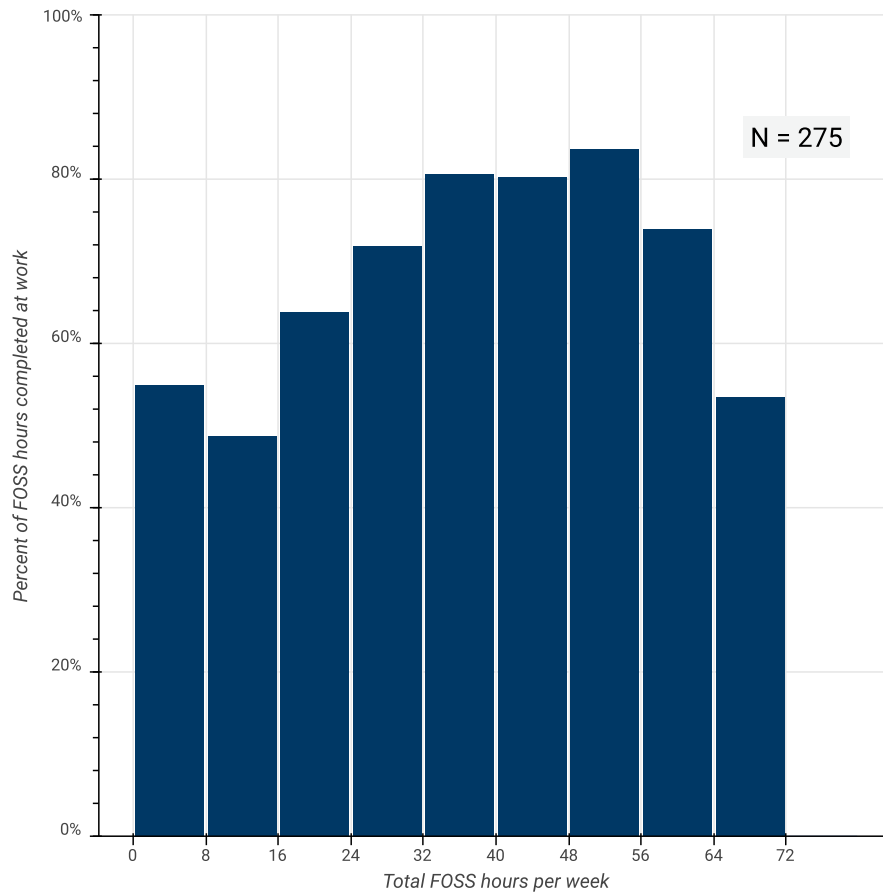
Figure 11 shows the number of hours per week spent on FOSS broken out by the highest contributor status level. As with many things related to FOSS, the hours contributed follow a roughly power-law distribution where most contributors spend a few hours a week. A handful of respondents spend more than 50 hours per week, with one reporting working on FOSS 70 hours per week. The more hours per week a contributor spends on FOSS, the more likely they are to have reported that they are a maintainer or core developer of at least one project. Most respondents who reported only being an occasional or one-time contributor spend less than four hours per week, while nearly all of them spend less than twelve hours per week.

Figure 11: Hours per Week Spent on FOSS By Contributor Status



To better understand how the relationship between hours per week spent on FOSS and whether or not a respondent is being paid for their efforts, the research team looked at the subset of respondents that reported being paid for FOSS work. Figure 12 shows the percentage of hours spent during paid work time versus free time, broken out by the number of hours per week spent on FOSS (in 8-hour blocks).

Figure 12: Percent of Hours Spent on FOSS Projects Occurring During Paid Work vs. Free Time



On average, respondents spending 0-16 hours per week on FOSS do roughly 50% of those hours during paid work. Meanwhile, respondents spending 16-32 hours per week do 65-70% of those hours during paid work. Respondents spending 32-64 hours per week on FOSS do 75-80% of those hours during paid work. The one respondent who reported doing 70 hours of FOSS per week reported that 38 of those hours were during paid work, while the other 32 were during their free time. In aggregate, no matter the number of hours spent on FOSS during paid work time, nearly all respondents also spend some of their free time working on FOSS.

The FOSS ecosystem is continuously evolving over time. To better understand the human side of this evolution, respondents were asked to compare the amount of time they spend on FOSS now to the amount of time they spent five or ten years ago. To be eligible to answer these questions, a respondent first had to indicate that they had been active in FOSS for more than five or ten years, respectively. Therefore, the sample size across these questions varies, and the results are broken out by the length of time an individual has been participating in FOSS.

Figures 13 and 14 show how a respondent's time spent on FOSS today compares to that spent five years ago for respondents with 5-10 years of experience in FOSS and 10+ years of experience in FOSS, respectively.

Figure 13: Time Spent on FOSS Now vs. 5 Years Ago for those with 5-10 Years Experience

5-10 Years Contributing to FOSS	Current Average Hrs/Wk	Current Median Hrs/Wk	Number of Responses
Current weekly time spent for this subset overall	10.78	5	151
Time spent for those responding that they spend <b>more</b> time now than 5 years ago	15.37	8	72
Time spent for those responding that they spend <b>less</b> time now than 5 years ago	3.77	2	49
Time spent for those responding that they spend <b>same</b> amount of time as 5 years ago	11.21	5.5	30

Figure 14: Time Spent on FOSS Now vs. 5 Years Ago for those with 10+ Years Experience

10+ Years Contributing to FOSS	Current Average Hrs/Wk	Current Median Hrs/Wk	Number of Responses
Current weekly time spent for this subset overall	15.78	10	297
Time spent for those responding that they spend <b>more</b> time now than 5 years ago	22.66	20	99
Time spent for those responding that they spend <b>less</b> time now than 5 years ago	7.87	4	114
Time spent for those responding that they spend <b>same</b> amount of time as 5 years ago	18.42	10	84

Figure 15 shows how time spent on FOSS today compares to that spent ten years ago for only those with 10+ years of FOSS experience. In aggregate, these three figures show a few trends. First, it is clear that while some respondents are spending more time than they used to, some are spending less. Second, it is interesting to note the similarities in percentages between Figure 13 (people with 5-10 years of FOSS experience and their contribution activity five years ago) and Figure 15 (people with 10+ years of FOSS experience and their contribution activity ten years ago). In both, close to half of respondents reported spending more time, roughly a third reported spending less time, and the remainder reported spending the same amount of time. This is noticeably different from Figure 14 (people with 10+ years of FOSS experience and their contribution activity five years ago) where the three options are closer to equal, with a slightly higher percentage spending less time and a slightly lower percentage spending the same amount of time.



Figure 15: Time Spent on FOSS Now vs. 10 Years Ago for those with 10+ Years Experience

10+ Years Contributing to FOSS	Current Average Hrs/Wk	Current Median Hrs/Wk	Number of Responses
Current weekly time spent for this subset overall	15.9	10	294
Time spent for those responding that they spend <b>more</b> time now than 10 years ago	21.06	20	139
Time spent for those responding that they spend <b>less</b> time now than 10 years ago	7.65	4	108
Time spent for those responding that they spend <b>same</b> amount of time as 10 years ago	19.59	10	47

Related to the shifting amount of time spent on FOSS over time, respondents were asked whether the current COVID-19/coronavirus pandemic had changed the amount of time they spent on FOSS. The survey found that 66% of respondents said they were spending the same amount of time on FOSS once the COVID-19 pandemic hit, while 24% said they were spending more time, and 10% said they were spending less. However, as mentioned in the Demographics section, given that responses to the survey came overwhelmingly from men (93.2%), these findings may not reflect the experiences of women who contribute to FOSS, particularly those impacted by increased family responsibilities during the pandemic.<sup>10</sup>

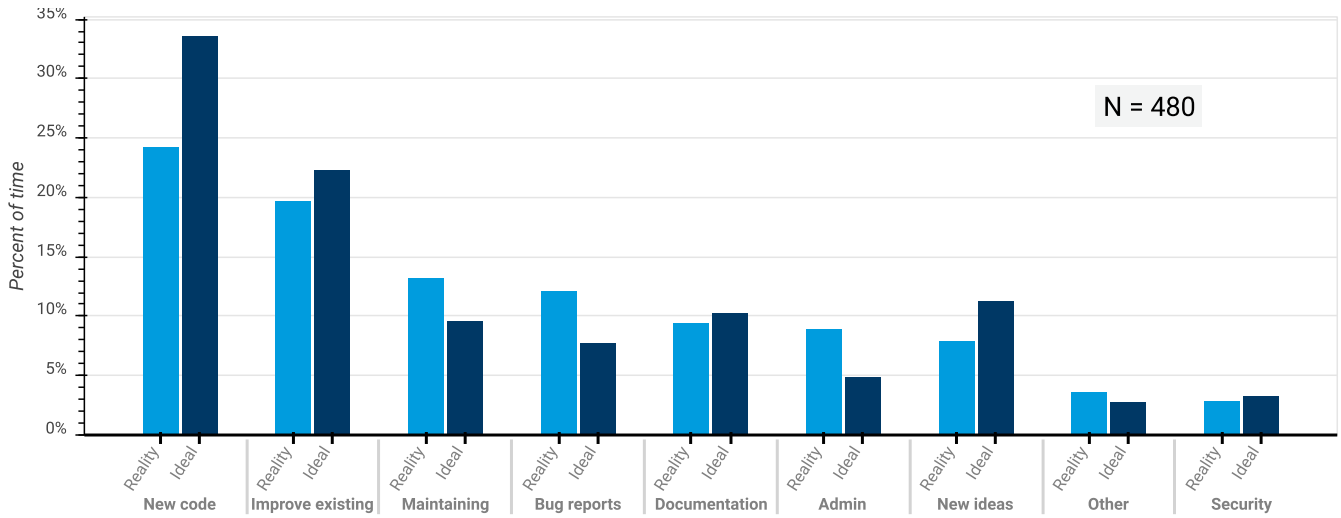
Finally, to understand how FOSS contributors allocate their time spent on FOSS, respondents were asked to share the percentage of their time on FOSS that goes towards the following activities:

- Contributing new code
- Improving existing functionalities
- Maintaining projects
- Performing organizational or administrative functions
- Reporting or documenting bugs and unexpected behaviors
- Offering ideas for new features
- Contributing documentation
- Responding to security issues

Respondents also had the opportunity to enter other tasks such as answering user questions, reviewing code, and project community management under an “Other” category. In aggregate, they spent approximately 5% of their contribution time on all of these miscellaneous tasks.

In addition to asking how they allocate their time, the survey also asked respondents how they would ideally like to allocate their time. Figure 16 shows the differences in preference for actual versus ideal time allocations. For tasks like contributing documentation, improving existing functionalities, and responding to security issues, the time allocated roughly matches the time the respondent desired to spend. However, the “general housekeeping” tasks like maintaining projects, reporting or documenting bugs and unexpected behaviors, or performing organizational or administrative functions tended to take a larger percentage of time than respondents would have liked. Conversely, contributors noted that they did not allocate as much time as desired to the more creative tasks of contributing new code and offering ideas for new features.

Figure 16: FOSS Time Allocation: Actual vs. Ideal



Figures 17 and 18 respectively show the actual time spent on FOSS tasks broken out by maintainers/core contributors versus occasional or one-time contributors and by paid versus unpaid contributors. Overall, it is clear that FOSS contributors spend the largest proportion of their time contributing new code or improving existing code. However, maintainers and core contributors also spend a good deal of time maintaining projects and performing administrative functions while occasional or one-time contributors spend a good deal of time on bug reporting and documentation. Likewise, paid contributors spend noticeably more time on administrative work than unpaid contributors. Importantly, all types of contributors spend very little of their time on responding to security issues, which is a likely area of concern for the future health of the FOSS ecosystem and is discussed further below.

Figure 17: FOSS Time Allocation: Maintainers/Core Contributors vs. Occasional/ One-Time Contributors

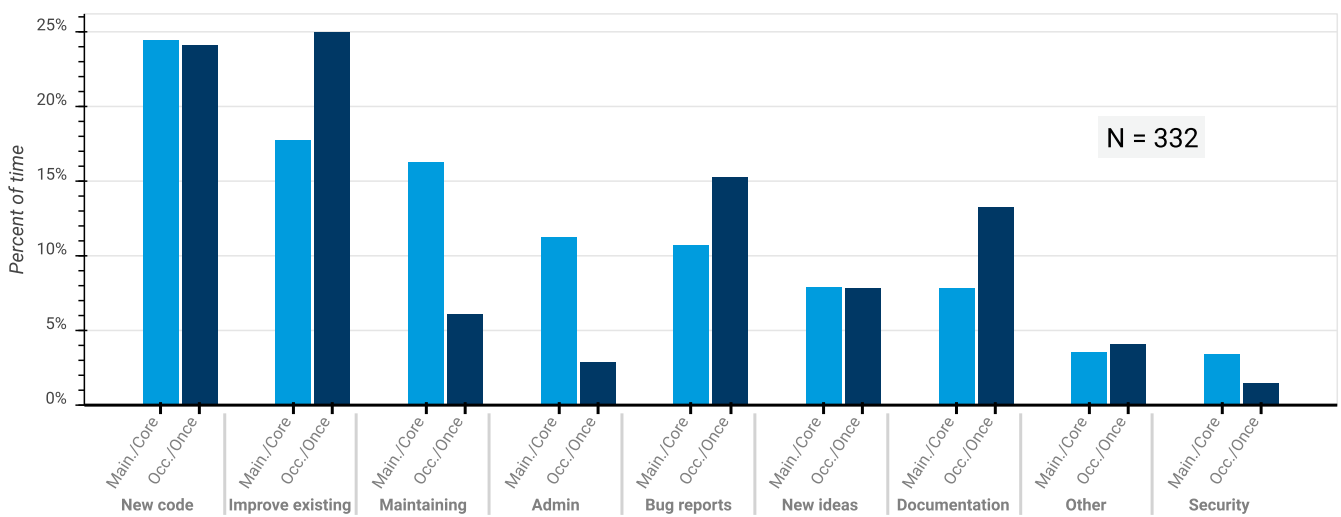
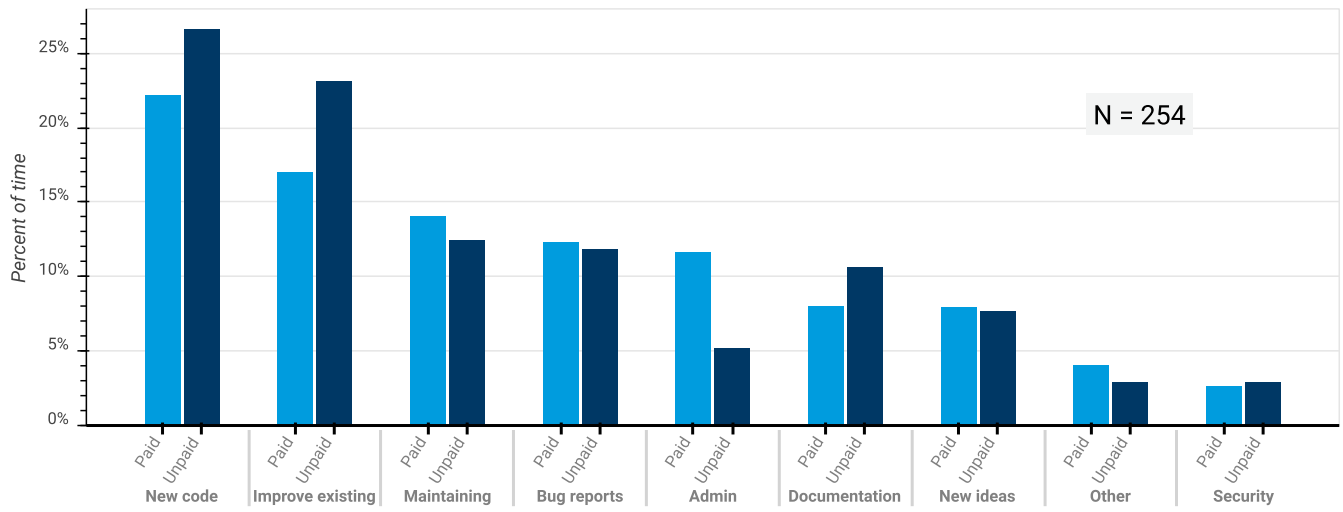


Figure 18: FOSS Time Allocation: Paid vs. Unpaid Contributors



# High-Level Takeaways & Suggested Actions

This survey delved into a great deal of detail on various dimensions of FOSS contributions. Below is a summary of a few key high-level takeaways that emerge from the results. Additionally, there are suggested actions related to each of these takeaways for open source project managers, as well as companies and organizations, looking to help sustain the FOSS ecosystem.

## 1. Contributors' Motivations

**The top three motivations of all contributors are non-monetary motivations.**

The most commonly occurring motivations in respondents' top three choices were "the need for a feature or fix" (a more functional motivation, indicating how FOSS is helpful in most contributors' work), the "enjoyment of learning," and the "need for creative expression." Conversely, being paid to develop FOSS was the most likely motivation to show up in an individual's bottom three choices. Even after splitting out motivations by contributors that report being paid for FOSS contributions versus those that are unpaid, payment remains in the bottom three of each group's motivations. This result may sound counter-intuitive, especially in light of the increased prevalence of paid contributions and the existing assumptions that these contributors might be contributing only because it is part of their work income. This stresses the importance of software professionals using FOSS for learning purposes that can potentially benefit their employers.<sup>11</sup>

### Suggested Actions: Leverage Motivations

The strong desire of contributors, paid and not paid, to learn should guide FOSS projects' and organizations' approaches toward contributors. FOSS contributions need to be perceived and communicated as an important learning mechanism for contributors. With the constant pace of change and new knowledge that contributors need to master, FOSS has become a clear and efficient path for learning and professional development. Companies ought to recognize the important added value of this knowledge. Additionally, FOSS projects should consider making it easy to get started with a FOSS project such as pre-created images or demos so that learning can begin quickly. FOSS projects could also provide some educational materials (such as tutorials or getting started guides) about their projects to help those motivated by a desire to learn.

The need for a feature or fix and the enjoyment of learning were the two most commonly occurring motivations for contributing to FOSS, while receiving payment for contributions was near the bottom—for both paid AND unpaid contributors.

It also appears that many contributors would be interested in learning more general material, such as how to develop secure software (as long as its application is not very burdensome). Several respondents listed a free online course on how to develop secure software as a desirable contribution from an external source. OpenSSF has developed a set of three free courses on how to develop secure software on the non-profit edX learning platform. These courses are part of the Professional Certificate program, [Secure Software Development Fundamentals](#) (for a fee, developers can take various tests to show that they mastered the material).

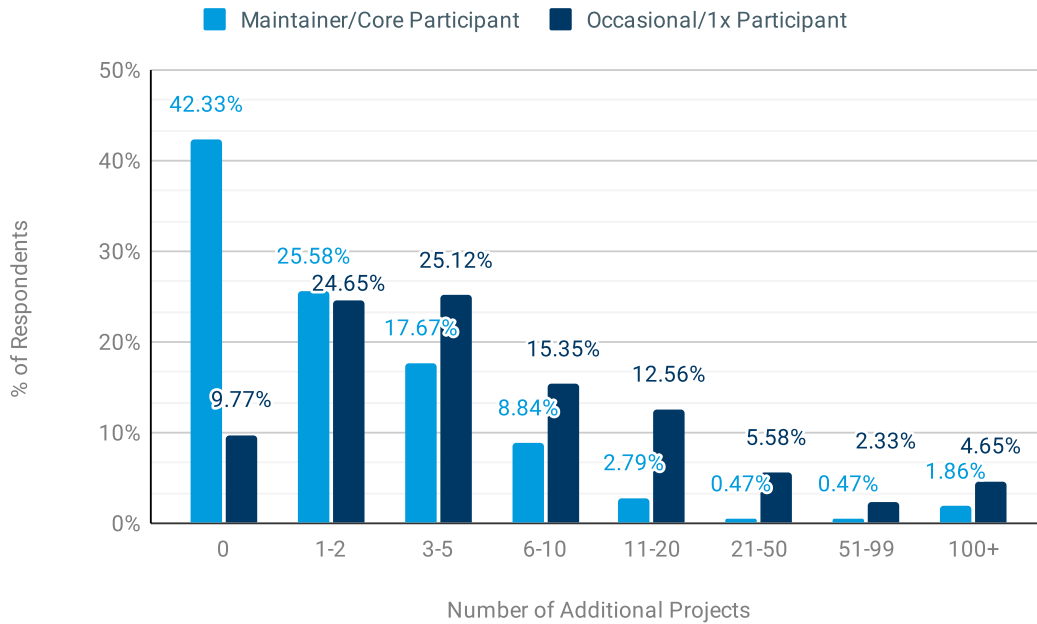
The “need for creative work” was highly rated (3rd motivation), so FOSS projects should keep their work creative to attract unpaid contributors and not focus on making it “less creative.” Balancing creative and more mundane tasks for contributors could keep project members more engaged.

The fact that payment to contributors is generally not reported as a strong motivator is an important finding as recently there have been various efforts to pay FOSS projects and their contributors for their efforts directly. However, it is important to note that this survey’s results show that the vast majority of respondents are employed full-time, and thus basic needs are likely being met for them. Also, more than half of respondents are paid to develop FOSS. Thus it is difficult to precisely discern the importance of being paid for working on FOSS, even though these paid contributors rated being paid as very low on their motivations for contributing.

Although their motivations for contributions are not primarily financial, it would be an overreach to say that money does not matter for FOSS. Note that more than a third of respondents (36.43%) listed financial contributions as potentially beneficial to at least one of the FOSS projects they contribute to, and financial contributions were considered the most important after code contributions (see Figure 20). The research team interprets this as suggesting that financial contributions to support FOSS development could be highly beneficial to increase their security and sustainability if primarily directed toward specific purposes, e.g., for adding security-related tools to the CI pipeline, security audits, and computing resources.

Some projects may not have anyone paid to contribute to them, even if they are important and even if some of the contributors are being paid to work on other projects. The 215 respondents identifying as core participants and maintainers (40 and 175, respectively) contribute — on average — to 2.13 projects for which they are paid and 1.2 projects for which they are not. Figure 19 shows how many FOSS projects these paid maintainers and core participants contribute to in addition to the initial five projects they reported initially in the survey — either regularly or occasionally. Although the survey did not ask whether respondents received payment for work on these additional projects, Figure 19 shows that more than half of those paid by their employer to contribute to FOSS for at least one of their primary projects participate significantly in other FOSS projects as well. Participation in these additional projects could be a natural result of their paid work (e.g., contributing to the upstream dependencies of the project they are paid to work on) or it could result from unrelated interests (e.g., hobby projects).

Figure 19: Participation of Core Participants & Maintainers Paid by their Employers in Projects Beyond the Initial 5 Identified



In short, even though many contributors are paid, it is possible that some critical projects need financial support or that their contributors are unpaid. For less desirable tasks (e.g., less creative), more funding may be required than expected because many contributors are not motivated primarily by money.

## 2. Need to Increase Security

### There is a clear need to dedicate more effort to the security of FOSS.

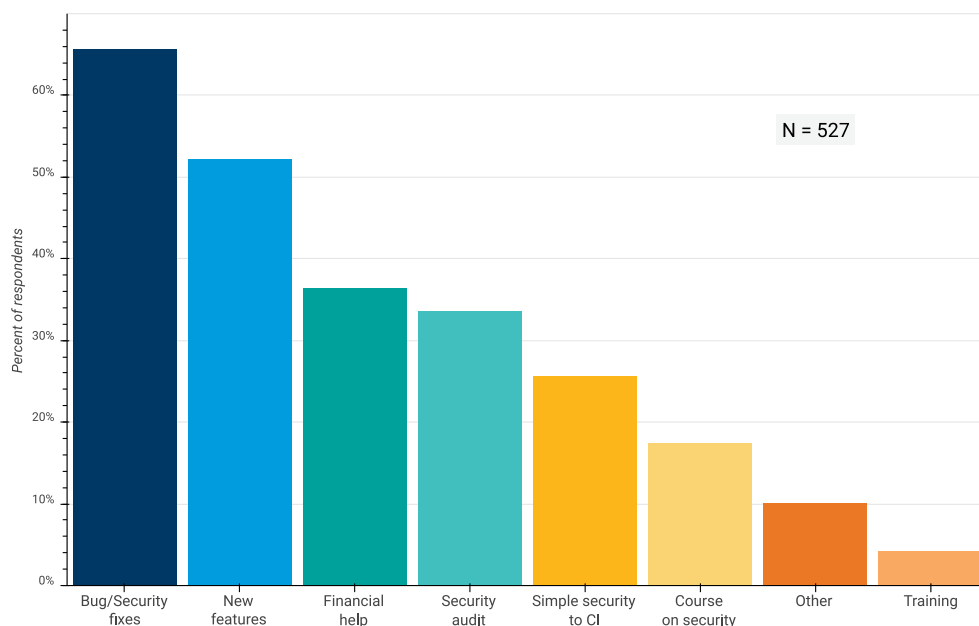
One of the survey goals was to understand the state of security in FOSS, and indeed it found that respondents report spending very little of their time on responding to security issues (an average of 2.27% of their total time spent). Moreover, the respondents do not report a desire to increase this significantly; in fact, the average of percent of time reported they would like to spend on security was only 0.06% higher. In addition to questions about their efforts, survey participants were asked what would be the most beneficial contribution to their FOSS projects from external sources. Some of the topmost requested contributions were proposed bug/security fixes, free security audits, and simplified ways to add security-related tools to their CI pipelines (Figure 19). This indicates that there is a clear need for these efforts, but existing contributors are not interested in dedicating substantial additional time to it.

Text responses indicated that many respondents had no interest in increasing time and effort on security; it was not simply that they wanted to be proactive. One respondent said, “I find the enterprise of security a soul-withering chore and a subject best left for the lawyers and process freaks. I am an application developer.” Another said, “I find security an insufferably boring procedural hindrance.”

Thus, efforts focused on dramatically increasing the time contributors spend on security are unlikely to be welcomed by many existing FOSS contributors, especially if they require significant new steps to be performed by developers. Therefore, alternative methods for incentivizing security-related efforts need to be considered.

Further, given the discussion on incentives, it is unlikely that simply offering money to contributors for focusing on security will move the needle a great deal (although for some smaller, understaffed projects, this could potentially help).

Figure 20: Value of Contributions from External Sources



## Suggested Actions: Improve Security Practices while Limiting the Burden on Contributors

Actions related to security can be thought of in three buckets: improving existing code by addressing security issues, rewriting code, and incorporating security into new code. In all of these cases, developers are not interested in dramatically increasing their time on security, so identifying ways to reduce or distribute the effort will be important.

To improve existing code by addressing security issues, the vulnerabilities must be identified (e.g., using audits and tools), and then the appropriate fixes must be developed and proposed. The CI pipeline also needs to be modified over time to detect problems and prevent recurrences.

Developers generally do not want to become security auditors; they want to receive the results of audits. The Open Source Security Foundation<sup>12</sup> (OpenSSF) is discussing launching efforts to audit critical FOSS projects, possibly including the development of proposed patches to fix problems found. However, these audits require funding, so organizations that do not have the resources to pay their own employees to contribute to FOSS could instead make small contributions towards security audits of the FOSS upon which they rely. These audit reports should include specific mergeable changes to implement those reports.

Organizations that pay employees to contribute to FOSS could redirect some of their efforts to identify and address known security issues in their relevant projects. In particular, organizations can have employees submit patches to correct vulnerabilities identified or incorporate automated tools to detect vulnerabilities without a significant false-positive rate.

Instead of fixing existing code, developers could rewrite portions or entire components of FOSS projects that are prone to vulnerabilities. A rewrite might be advisable for other reasons, e.g., the project's or component's structure might be so convoluted that it's challenging to maintain. Often such rewrites begin with developing tests so that functionality and reliability are not lost. If a rewrite is to be done, there should be an effort to produce a substantially more secure result.

One way to improve a rewrite's security is to switch from memory-unsafe languages (such as C or C++) into memory-safe languages (such as nearly all other languages). This would eliminate entire classes of vulnerabilities such as buffer overflows and double-frees. These vulnerability classes loom large; Microsoft reports that 70% of all security bugs from 2006 through 2018 have consistently been memory safety issues.<sup>13</sup> Some projects are pursuing this approach, e.g., Mozilla's Oxidation project is rewriting portions of Firefox in Rust, a programming language that is memory-safe by default.<sup>14</sup> However, this approach has many challenges. It requires significant effort to rewrite the code, and perhaps more importantly, a significant effort for the existing contributors to learn to use the new language well if they do not know it already. The project may find it challenging to agree on that new language since there are always trade-offs, and the new language selection would have a large impact on all contributors.

While regular audits can ameliorate security concerns in the short term, establishing secure coding and security best practices as every FOSS member's responsibility will be key for the long term.



All of this effort will, at best, have no immediate visible advantage to users since they will typically not see an immediate functional improvement. Rewriting a portion is less risky than trying to rewrite an entire component. Still, that approach is also more complicated because of the increased complexity of calling between components that are written in different languages. Using a different language can also complicate compilation (where applicable) and use of the component, e.g., if its callers are typically written in C, it's often simplest to write the component in C. Switching to a memory-safe language could have significant long-term advantages, but it presents significant shorter-term costs. It's likely to be considered more of an option for network-facing components or those typically used to process untrusted data. However, this might be an area where one-time funding could produce significant long-term improvements.

Incorporating security into new code in the FOSS ecosystem needs to become a priority for all FOSS stakeholders. If organizations, projects, and contributors incorporate secure coding and other security best practices into any new code additions, the foundations of open source will become much more robust.

The following are just a few examples of actions that could improve security, without placing significant additional burdens on developers:

1. Make secure development training a requirement for hiring or continued professional development for paid FOSS developers. As noted earlier, the OpenSSF has developed a set of three free courses on how to develop secure software on the non-profit edX learning platform. These courses are part of the Professional Certificate program, [Secure Software Development Fundamentals](#) (for a fee, developers can take various tests to show that they mastered the material).
2. Use badging programs, like the Core Infrastructure Initiative's Best Practices Badge<sup>15</sup>, as a powerful new norm that encourages projects to develop — and maintain — secure software development practices.
3. Ask influential FOSS contributors to stress the importance of security and specific steps to make the software more secure.
4. Partner with mentoring programs, such as The Linux Foundation's LFX<sup>16</sup> and Google's Summer of Code<sup>17</sup>, to have them incorporate security best practices to help the next generation of FOSS contributors learn secure software development from the start.
5. Encourage projects to incorporate many different security tools and automated tests as part of their continuous integration (CI) pipeline. Platforms (such as GitHub and GitLab) could provide useful defaults and make it especially easy to incorporate these tools and tests. This could include automatically-generated merge/pull requests to change a CI configuration to add such tools, including an appropriate configuration for them in that circumstance. This request could be generated after automatically determining a project's specific technology stack (such as its languages and framework). Once a project successfully addresses problems found in a configuration, later merge/pull requests could be automatically generated that would modify the configuration to increase tools' sensitivities.

Finally, a better understanding of why FOSS developers do not like to spend time on security issues may require follow-up interviews and qualitative deep dives with key maintainers and organizations that employ FOSS developers. Such efforts would likely lead to possible additional mechanisms for addressing security concerns.

### 3. Contributions Linked to Employment

**Nearly half (48.7%) of contributors stated they are paid by their employer to contribute.**

The survey shows just over half of respondents (51.65%) are paid for at least some of their FOSS contributions by their employer or a third-party, while just under half (48.35%) are volunteer contributors. Yet, most paid contributors contribute to multiple projects and frequently make unpaid contributions to other projects. Over the past few years, there has been debate about the growing influence of money in an ecosystem that was once mostly driven by volunteer work. There may be a concern that FOSS projects will be pushed in a direction that is most beneficial for the employer of a paid contributor (especially when that contributor is a project maintainer), rather than in a direction that is best for all users of the project. Further, there is concern about what happens if the organizations paying employees to contribute decide that it is no longer cost-effective to do so. This would be a particularly important concern if there was a large negative economic shock (e.g., a recession) that impacted the entire tech industry which is where most of the paid contributors work.

The survey shows just over half of respondents (51.65%) are paid for at least some of their FOSS contributions by their employer or a third-party, while just under half (48.35%) are volunteer contributors.

If these contributors, especially those who spend most of their paid working hours contributing to FOSS, then were redeployed to spend their working hours on internal proprietary work, would the volunteer contributors pick up the slack, or would there be a large decrease in the total amount of contribution effort towards FOSS? Of course, there are also some potential benefits to paid contribution efforts that could help lead to increased stability and sustainability in FOSS. Suppose more individuals are paid to work on FOSS. In that case, there is a decreased reliance on contributors having enough free time (which can ebb and flow due to life considerations, as discussed above) to maintain critical FOSS projects successfully.

#### **Suggested Actions: Balance Corporate and Project Interests**

To help address concerns related to corporate influence in FOSS, greater transparency of corporate involvement should be available. Although the level of corporate involvement in some large projects is clear, it is often less so in smaller projects. Corporate involvement in FOSS should be made clear to reduce accusations of hidden agendas. Further, to help reduce fears of companies retracting their participation, they can make clear commitments to support FOSS in general and specific FOSS projects for at least several years. Additionally, paid contributors should be incentivized to dedicate time mentoring new volunteer contributors to encourage new contributor development.

An additional option is for FOSS project governance decisions to ensure paid contributors do not crowd out volunteer contributors and prevent a single company from dominating contributions to essential projects. The obvious solution is to transfer the FOSS project to a foundation and establish neutral governance so that no single organization controls it. When respondents were considering whether or not to contribute to

a FOSS project, 52.54% said it was extremely important, very important, or moderately important to have neutral governance (by a foundation or otherwise not controlled by a single company). However, it may be difficult to convince a dominating company to do this (when there is one), such as when one company is the primary contributor to a project because they are the project's primary users. If the company will not yield the project to an independent foundation, the community may be better off that this project is open source than proprietary, but will need to either come to terms with it being dominated by one company or possibly fork it into a competing project.

## 4. Corporate FOSS Policies

### **Despite companies' increasing openness towards their employees' involvement in FOSS, many still do not have clear FOSS policies.**

Over the last ten years, companies have increasingly allowed their employees to contribute to FOSS. 45.45% of respondents stated that they are free to contribute to FOSS without asking permission, compared to 35.84% ten years ago. Today, 30.30% of respondents can contribute after receiving permission, up from 24.78% (for a total of 75.75% who may contribute to FOSS). However, 1.17% of respondents reported that their company does not permit employees to contribute to FOSS projects at all. Additionally, a significant percentage of respondents reported that their firms either have unclear policies (17.48%) or that they are unaware (5.59%) of what the policies are. By comparison, ten years ago only 35.84% were free to contribute without asking permission, 24.78% needed to ask permission, 7.08% were not permitted to contribute at all, 25.66% had unclear policies, and 6.64% were unsure of the policies in place. Although there have been positive changes over the past decade, there is still room for improvement.

### **Suggested Actions: Clearer Policies for More FOSS Contributions**

Companies should embrace clear policies on FOSS. These policies should, ideally, allow FOSS contribution as this can benefit the company. This survey finds that the two top motivations as "love of learning" and "needed a feature in this FOSS" and that these two are highly connected. This means that software professionals gain significant and satisfying learning that is also highly beneficial for their work by using and contributing to FOSS. Previous studies have also shown that contributing to FOSS is associated with higher company productivity in many settings.<sup>18</sup> Allowing employees to contribute to FOSS can also allow companies to attract higher caliber programming talent (65% of respondents said their current employer's support of FOSS influenced their decision to join the organization, and 81% of respondents said the skills the acquired in FOSS were either valuable or very valuable to their current employer). Further, although companies are increasingly allowing their employees to contribute to FOSS, it would be ideal if they also made a push for their employees to contribute to the security of these projects, either individually or through collaborative efforts such as the Open Source Security Foundation (OpenSSF).

# Conclusion

The goals in running this survey were to understand the state of security and sustainability in FOSS and identify opportunities to help improve them and ensure FOSS's viability in the future. In particular, this survey focused on the "human side" of FOSS, more than the technical side, although the two are certainly inter-related, and these findings relate to both. These results identified reasons for optimism about the future of FOSS (individuals are continuing to contribute to FOSS, companies are becoming friendlier to FOSS to the point of paying some employees to contribute, etc.), but also areas of concern (in particular, the lack of security-related efforts, and potential difficulties in motivating such efforts).

In the end, free and open source software is, and always has been, a community-driven effort that has led to the development of some of the most critical building blocks of the modern economy. This survey highlights the importance of the security of this important dynamic asset. Likewise, it will take a community-driven effort, including individuals, companies, and institutions, to ensure FOSS is secure and sustainable for future generations.

By continuing such discussions with, and analysis of, the FOSS contributor community, end-users and other stakeholders will better understand how they can help sustain FOSS into the future. Therefore, the Linux Foundation and the Laboratory for Innovation Science at Harvard aim to run future iterations of this survey annually. If you would like to be involved in such efforts, please sign up here: <http://bit.ly/2021-FOSS-Survey>

# A Thank You to FOSS Contributors

As a thank you to the open source contributors who kindly gave us their time and insight by completing this survey, the authors would like to acknowledge that without the following individual respondents — and many others — this research would not have been possible. The following contributors authorized us to share their names/identifiers, allowing us to publicly thank them for their time.

@KengoTODA	beikov	David Hebbeker
@MarcoRossignoli	Bence A. Toth	davidkhala
@webknjaz (Sviatoslav Sydorenko)	Benjamin Krämer	DAXaholic
2E0PGS	benjamingr	Debajit Roy
Aaron Friel	Bernd Eckenfels	delagen
AaronRobinsonMSFT	Bert Laverman	Denys Tsomenko
adambkaplan	bobfriesenhahn	derekcollison
Aditya Bisoi	boenrobot	dhalperi
adrpo	bokub	Dhruv Aggarwal
Ajith P Venugopal	borgdylan	DiegoRBAquero
Alexander Usyskin	Brian Cipriano	dilawar
Alexey Romanov	Brian London	djjodav
Alexey Rusakov	Brian Warner	drabinowitz
alshopov	Bryce Cronkite-Ratcliff	EdChdX
Alvaro Carballo Garcia	burtonr	Edison Henrique Andreassy
Alvin Smith	BWPanda	eduardbadillo
Andreas Lindhé	C. Masloch	Emmanuel DELOGET
andreevym	Caleb Cushing	ennerf
Andrew Johnson	Cengiz Can	Eric Newberry
Andrew Landsverk	Christian Murphy	Erwin Janssen
andreybleme	chtompki	Eugene Platonov
Andrzej Leśkiewicz	Cogitri	Eugenio Palumbo
Anonymouse64	colltoaction	fahimfarookme
ansvonwa	cooperka	faissaloo
Anthony Zotti	Craig Andrews	Fardale
Anuradha Udunuwara	Crt Mori	felixmulder
arouel	CXuesong	FFY00
Arshad Basha	cyyber	frague59
artem-smotrakov	Daniel Barlow	Francois Marier
aularon	Daniela	Frederick
Baruch Siach	DanielRuf	Gabriel Schulhof
bauen1	danintel	Gary E. Miller
bcornec	daveisfera	gaurav9822

gauravbrills	Josh Cummings	mhewedy
GeorgLink	Joshua Humphries	Michael Hausegger
gesellix	Jostein Kjørnigsen	michelkazi
Ghan Vashishtha	Juan José López Villar	mightyiam
ghane	juliangruber	Miha-x64
giorgioazzinnaro	Justin Szaday	MikeCamel
Giorgos Gaganis	Kahbazi	Miloudi Adel
glyn	KangZhiDong	Mirana Lova
Grant Hutchins	Karen Bennet	miromannino
grexe	Karsten Ohme	Mkeskells
Grzegorz Rożniecki	Ken Guest	mpollmeier
Gsealy	Kevin Croft	myd7349
GuiRitter	kilian hefti	Mykhailo Kaskun
GustavoARSilva	kitplummer	Mykyta Potapenko
H. Chase Stevens	kulabun	myungjoo
Harald Brabenetz	kwwall	NAVA Mathieu
Harald Welte	kyranet	Neal Murphy
Hasan Arous	leon-barrett	neki
Hendrig Sellik	Leonardo Teixeira Menezes	Nicholas Renner
<a href="https://github.com/c-rindi">https://github.com/c-rindi</a>	LeSpocky	NingZhang-e
<a href="https://github.com/enesacikoglu">https://github.com/enesacikoglu</a>	lextiz	NoodleOfDeath
iblamefish	listerenko	nope
IDA Kenichiro	Liyang Jin	nyetwurk
ihrigb	LJHarb	obscurerichard
ilgrosso	Lorenz Nickel	Oleksandr Kozlenko
In Gyu Hwang	Luc Van Oostenryck	Oleksij Rempel
j4ng5y	madcampos	Oliver Sampson
Jack-Works	Mael Le Guen	oprogramador (Piotr Sr)
Jacob Lalonde	marado	osiegmar
JamesNK	Marcel Raad	Oskar Duycz
Jamo Luhrsens	Mario Rugiero (github/Oppen)	Pedro Lino
Jan Henke	MarkDacek	pengisgood
Jan Lübke	markdroth	perrin4869
JanStureNielsen	Martin Fuzzey	Phillip Johnson
japgolly	Mateusz Gozdek	prasanths96
Jason Sturges	mathben	psilospore
Jasper Moeys	mathdesc	rafoof
javad-alipanah	matiaszanolli	rafaelsamenezes
Jeff Sheets	MatthewZMD	ramrunner
Jeffrey Ulatan	Matti Vaittinen	rbergman
jpgonggrijp	mbiesiad	redivo
jmacey	Mc	reinhapa
Joe Pearson	mcgrof	Rhys Perry
Johannes Kohnen	meirotstein	Richard Laager (rlaager)
Jonas Öberg	Mekki MacAulay	Robert Hancock
Joris-van-der-Wel	meskio	Robert J Lipe

Robert Jacobson  
RobertvanderHulst  
RomanHargrave  
romeara  
Rory Graves  
Rosica Dejanovska  
Rouven Wessling  
rtfpessoa  
rweisleder  
rwky  
s-ayush2903  
Samer Masterson  
Sammy Chu  
Santhosh  
sbrl  
schumaml  
seratch  
sibelius  
siblingzconsult  
SieuDoTia  
SinopsysHK  
siriak  
Skjolber  
smbape

Solomon Peachy  
sourabhsparkala  
Stefan Haun  
Stephan van den Akker  
Steve Gagné  
Steve Manuel  
stukselbax  
sumnerevans  
swaldman  
swinslow  
syntheticnightmar3  
Taylor Wood  
tealeg  
Teoman "QuanticDev" Soygul  
theMcQ  
TheRealHau  
TheToolbox  
Thomas Harning Jr.  
Tobias Jakobsson  
Tobias Schlatter  
Toke Høiland-Jørgensen  
Tomáš Janoušek  
trevorlinton  
Trishank Karthik Kuppusamy

tswast  
Ugljesa Jovanovic  
Umer Salman (umer936)  
Utsav Akruwala  
vamsi-kavuri  
vchrombie  
vibhuti019  
Victor Homyakov  
VictorNicollet  
Vishal anand  
Vivek Ganesan  
WalterCouto  
Weixin Wang  
Wentao Liu  
Winston Weinert  
xcatliu  
Yann E. MORIN  
yasserzamani  
Yurii Kadirov (@sirkadirov)  
YUTARO IINO  
z3ntu  
Zeeshan Shahid  
ZZMarquis  
龙腾

# Endnotes

1. "State of the Software Supply Chain," Sonatype, 11 July 2016, [https://www.sonatype.com/hubfs/SSC/Software\\_Supply\\_Chain\\_Inforgraphic.pdf?t=1468857601884](https://www.sonatype.com/hubfs/SSC/Software_Supply_Chain_Inforgraphic.pdf?t=1468857601884).
2. Klint Finley, "Open Source Won. So, Now What?" Wired, 11 August 2016, <https://www.wired.com/2016/08/open-source-won-now/?GuidesLearnMore>.
3. Emily Olin, "Corporate Open Source Programs are on the Rise as Shared Software Development Becomes Mainstream for Businesses," The Linux Foundation, 30 August 2018, <https://www.linuxfoundation.org/uncategorized/2018/08/corporate-open-source-programs-are-on-the-rise-as-shared-software-development-becomes-mainstream-for-businesses>.
4. Klint Finley, "Diversity in Open Source Is Even Worse Than in Tech Overall," Wired, 2 June 2017, <https://www.wired.com/2017/06/diversity-open-source-even-worse-tech-overall/>.
5. For example, StackOverflow's "2020 Developer Survey" (<https://insights.stackoverflow.com/survey>) showed that male and female respondents differed on a variety of reasons that they code and why they choose their current employer.
6. For example, Karim R. Lakhani and Robert G. Wolf, (2005). "Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects." In Perspectives on Free and Open Source Software, edited by Joe Feller, Brian Fitzgerald, Scott Hissam, and Karim R. Lakhani. Cambridge: MIT Press.; Josh Lerner & Jean Tirole, (2002). "Some Simple Economics of Open Source." The Journal of Industrial Economics, 50(2), 197-234.; Jeffrey A. Roberts, Il-Horn Hann, & Sandra A. Slaughter, (2006). "Understanding the Motivations, Participation, and Performance of Open Source Software Developers: A Longitudinal Study of the Apache Projects." Management Science, 52(7), 984-999.; Sonali K. Shah, (2006). "Motivation, Governance, and the Viability of Hybrid Forms in Open Source Software Development." Management Science, 52(7), 1000-1014.
7. These motivations are consistent with the three primary groups of motivations for contributing to user communities and FOSS that are found in the academic literature (as discussed in Sonali K. Shah and Frank Nagle, "Why Do User Communities Matter for Strategy?" Special Issue on Open Innovation. Strategic Management Review 1, no. 2 (2020): 305-353) : self-focused intrinsic motivation (learning, creativity, a feature needed, sense of community), community-focused intrinsic motivation (reciprocity, belief in the mission, helping others), and extrinsic motivation (recognition, payment, career advancement). However, importantly in our context when a user says they were motivated by needing a feature, we do not know if this need is purely intrinsic for a project they are doing in their free time, or if it is needed for their paid work. Thus that particular motivation could be either self-focused intrinsic, or extrinsic. Given the high rating of this motivation, future versions of the survey will seek to disentangle these two possibilities.
8. The "Total of Respondents Ranking It" column contains different values for each response as it was possible for a respondent to not rank some of the motivations. This occurred only a handful of times.



9. Cliff Lampe, Rich Wash, Alcides Velasquez, & Elif Ozkaya,(2010, April). "Motivations to Participate in Online Communities." In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 1927-1936).
10. Deepa Mahajari, Olivia White, Anu Madgavkar, and Mekala Krishnan, "Don't Let the Pandemic Set Back Gender Equality", Harvard Business Review, 16 September 2020, <https://hbr.org/2020/09/dont-let-the-pandemic-set-back-gender-equality>
11. As discussed in Frank Nagle, (2018). "Learning by Contributing: Gaining Competitive Advantage through Contribution to Crowdsourced Public Goods." Organization Science, 29(4), 569-587, and Frank Nagle, (2019). "Open Source Software and Firm Productivity" Management Science, 65(3), 1191-1215, companies can reap benefits from the learning that occurs when they use FOSS and allow their employees to contribute to it.
12. Open Source Security Foundation (OpenSSF), <https://openssf.org/>.
13. Catalin Cimpanu, "Microsoft: 70 Percent of All Security Bugs are Memory Safety Issues," ZDNet, 11 February 2019, <https://www.zdnet.com/article/microsoft-70-percent-of-all-security-bugs-are-memory-safety-issues/>.
14. "Oxidation," Mozilla Wiki, <https://wiki.mozilla.org/Oxidation>.
15. "Best Practices Program," The Linux Foundation Projects: Core Infrastructure Initiative, <https://www.coreinfrastructure.org/programs/best-practices-program/>.
16. "Mentorship," The Linux Foundation: LFX Tools, <https://lfx.linuxfoundation.org/tools/mentorship/>.
17. "Google Summer of Code," Google, <https://summerofcode.withgoogle.com>.
18. As discussed in Frank Nagle, (2018). "Learning by Contributing: Gaining Competitive Advantage through Contribution to Crowdsourced Public Goods." Organization Science, 29(4), 569-587, companies can reap benefits from the learning that occurs when they allow their employees to contribute to FOSS.

# Appendix: Full Survey Text and Detailed Data

As mentioned in the Methodology section, the pool of submitted surveys included many partially completed submissions. The first question — about geographic location — was the only question that all participants were required to answer, in order to comply with international research policies regarding data. Therefore, the results below each survey question represent an analysis of the answers that respondents did provide, even if those same respondents left other questions unanswered. Overall, the respondent pool for each question (N) will represent a subset of the overall pool of 1,196 individuals who answered the demographic questions and at least one question about their FOSS contributions. Separate from each question's respondent pool is the raw number of participants who saw the question and chose to leave it blank. Each potential response is followed by the percentage of that question's respondent pool that selected it. Text responses have been summarized to highlight major trends or unique insights. Display and skip logic has been added in under the question text in italics.

# Demographics

## 1. To determine which privacy law stipulations apply (such as GDPR), would you please select the country in which you currently reside?

N = 1196

Country	% of Respondents	Country	% of Respondents	Country	% of Respondents
USA	27.68%	Bulgaria	0.59%	Sri Lanka	0.17%
Germany	12.04%	Ireland	0.59%	United Arab Emirates	0.17%
France	6.94%	Hungary	0.59%	Uruguay	0.17%
United Kingdom	4.77%	Portugal	0.59%	Zambia	0.17%
India	4.01%	Slovakia	0.59%	Bahrain	0.08%
Canada	3.26%	Greece	0.50%	Bangladesh	0.08%
Netherlands	2.76%	Hong Kong	0.50%	Cameroon	0.08%
Russia	2.51%	Turkey	0.33%	Colombia	0.08%
Czech Republic	2.26%	Argentina	0.33%	Egypt	0.08%
China	2.17%	Iran	0.33%	El Salvador	0.08%
Australia	1.84%	Pakistan	0.33%	Georgia	0.08%
Japan	1.67%	Singapore	0.33%	Honduras	0.08%
Poland	1.67%	South Africa	0.33%	Madagascar	0.08%
Brazil	1.42%	Belarus	0.25%	Malaysia	0.08%
Spain	1.42%	Chile	0.25%	Malta	0.08%
Switzerland	1.42%	Estonia	0.25%	Morocco	0.08%
Sweden	1.34%	Indonesia	0.25%	Niger	0.08%
Austria	1.25%	Kenya	0.25%	Nigeria	0.08%
Israel	1.25%	Lithuania	0.25%	Peru	0.08%
Italy	1.25%	Romania	0.25%	Saudi Arabia	0.08%
Ukraine	1.17%	Vietnam	0.25%	Slovenia	0.08%
Finland	0.84%	Algeria	0.17%	Syria	0.08%
Norway	0.84%	Croatia	0.17%	Swaziland	0.08%
New Zealand	0.84%	Ecuador	0.17%	Thailand	0.08%
Mexico	0.75%	Latvia	0.17%	Venezuela	0.08%
Denmark	0.59%	North Korea	0.17%	Zimbabwe	0.08%
Belgium	0.59%	South Korea	0.17%		

## 2. How old are you?

N = 1192 Saw but didn't respond: 4

Age Range ▲	% of Respondents
18 to 24 years	9.90%
25 to 34 years	36.83%
35 to 44 years	34.14%
45 to 54 years	12.16%
55 to 64 years	4.87%
65 to 74 years	0.84%
75 years or older	0.08%
I prefer not to say	1.17%

## 3. What is your gender?

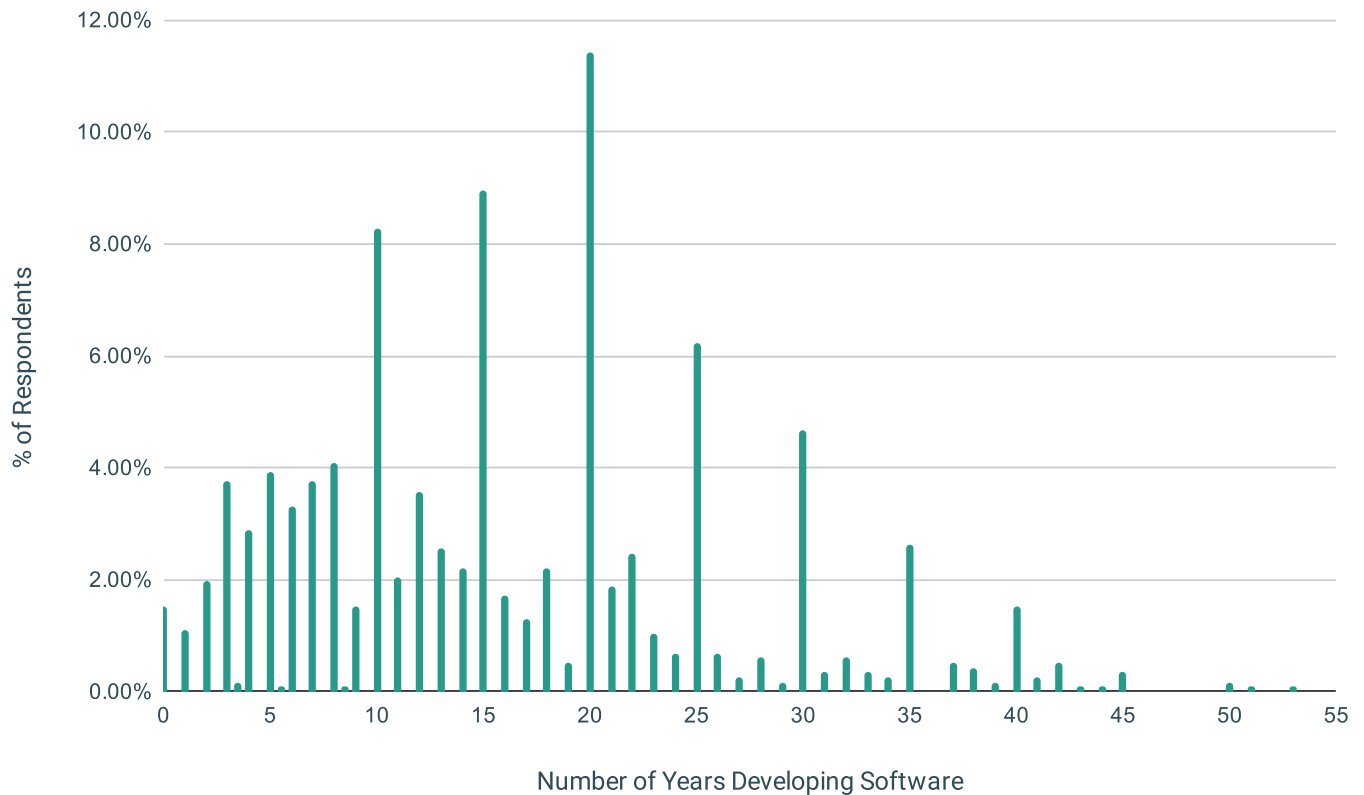
N = 1192 Saw but didn't respond: 4

Gender	% of Respondents ▼
Male	91.28%
Prefer not to Answer	4.28%
Female	3.19%
Non-binary	1.01%
Other	0.25%

# Software Development and FOSS Background

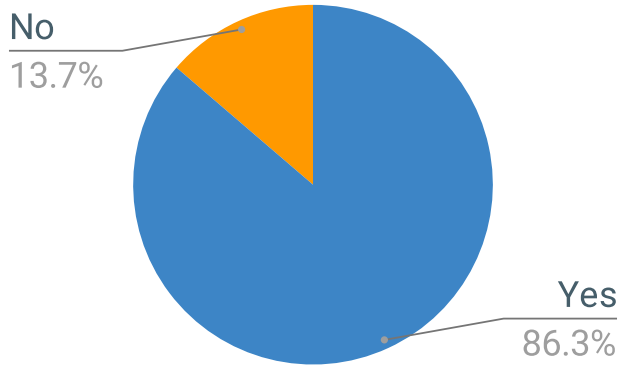
## 4. How many years have you been developing software?

N = 1174 Saw but didn't respond: 22



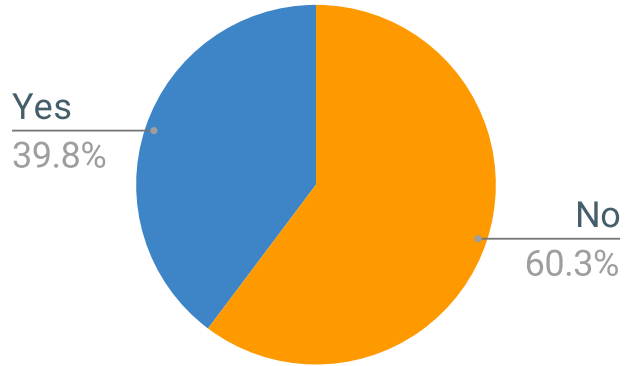
**5. Have you received formal training in software development? (e.g., classes in high school or university, coding boot camps, etc.)**

N = 1181 Saw but didn't respond: 15



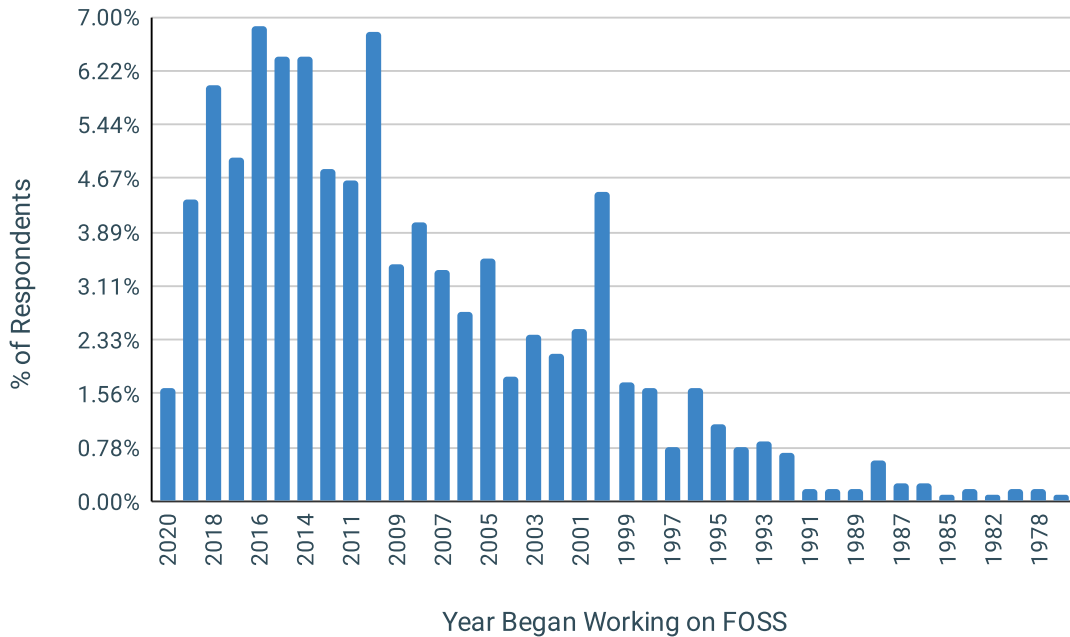
**6. Have you received formal training in developing secure software? (e.g., classes, badges, coding boot camps, etc.)**

N = 1180 Saw but didn't respond: 16



**7. What year did you begin working on open source (FOSS) projects? (e.g., 2009)**

N = 1166 Saw but didn't respond: 30



Started FOSS	% of Respondents
2020	1.63%
2019	4.37%
2018	6.00%
2017	4.97%
2016	6.86%
2015	6.43%
2014	6.43%
2013	4.80%
2011	4.63%
2010	6.78%
2009	3.43%
2008	4.03%
2007	3.34%
2006	2.74%
2005	3.52%
2004	1.80%
2003	2.40%
2002	2.14%
2001	2.49%
2000	4.46%
1999	1.72%

Started FOSS	% of Respondents
1999	1.72%
1998	1.63%
1997	0.77%
1996	1.63%
1995	1.11%
1994	0.77%
1993	0.86%
1992	0.69%
1991	0.17%
1990	0.17%
1989	0.17%
1988	0.60%
1987	0.26%
1986	0.26%
1985	0.09%
1984	0.17%
1982	0.09%
1979	0.17%
1978	0.17%
1953	0.09%

**8. Are you currently contributing to any FOSS projects actively? (e.g., at least one commit in the last 36 months)**

N = 1179 Saw but didn't respond: 17

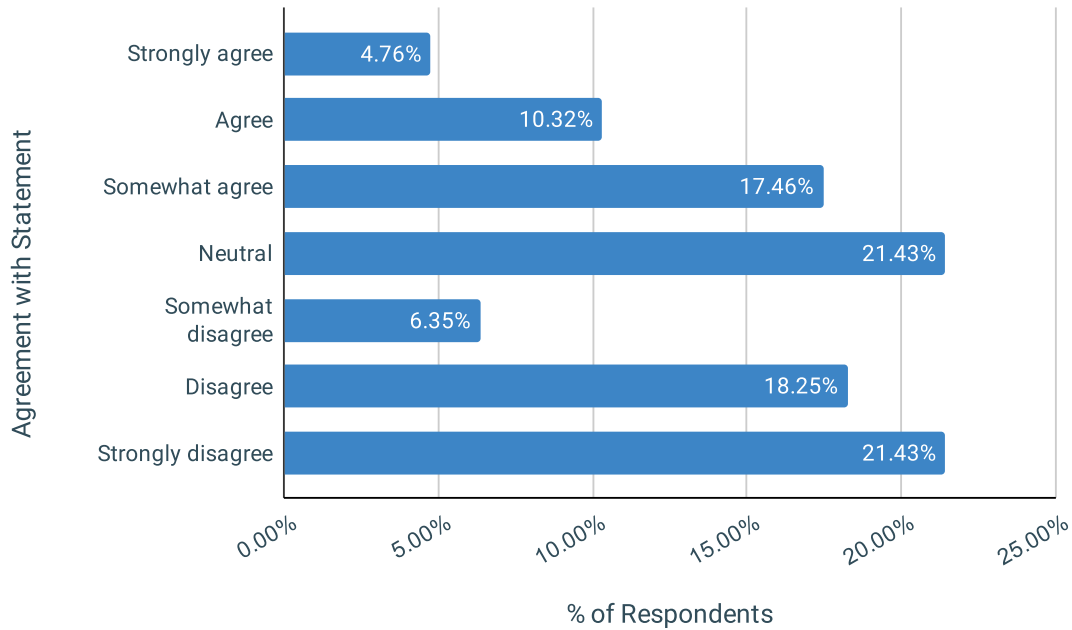
Currently contributing?	% of Respondents
Yes, I contribute on a regular basis	50.72%
Yes, but I contribute more sporadically	36.73%
No, I stopped contributing temporarily	7.46%
No, I no longer contribute to any FOSS projects	5.09%

**9. There are many reasons that people stop participating in an open source project. If your participation in FOSS has ever stopped, please rate your agreement with the following reasons as they apply to your personal situation.**

*This question was only shown to those who responded to question #8 that they are not currently contributing to FOSS.*

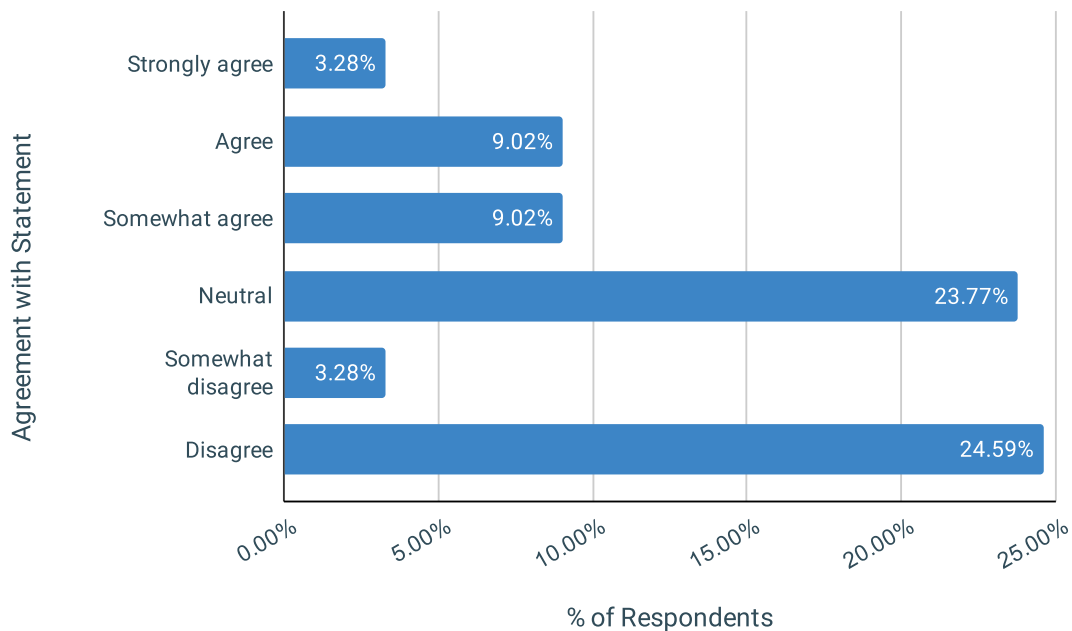
**a. I lost interest in the project**

N = 126 Saw but didn't respond: 22



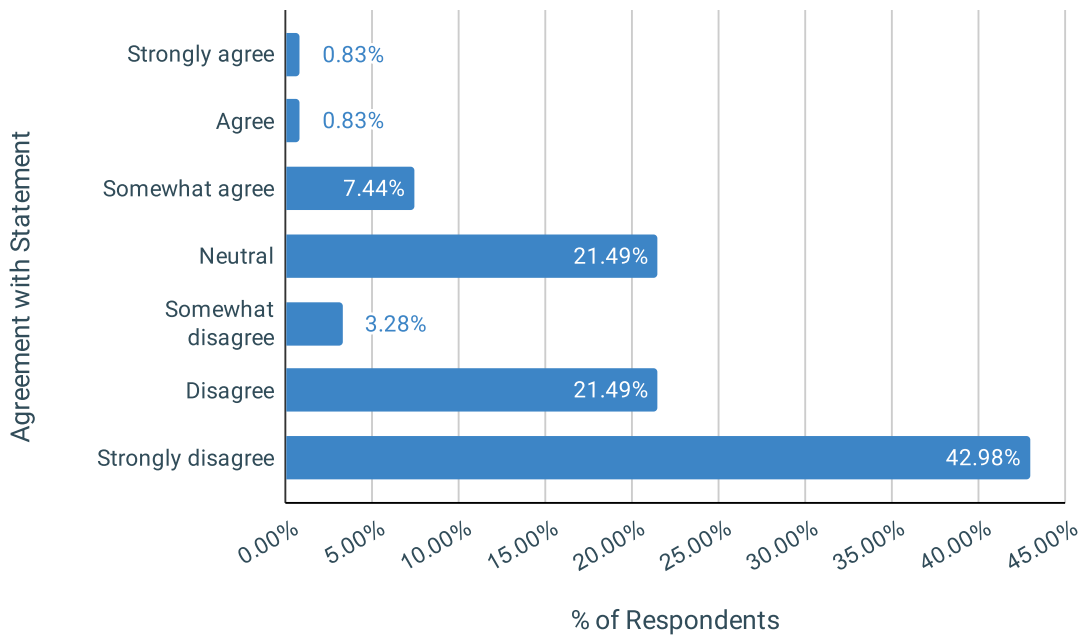
**b. I started to contribute to another open source project**

N = 122 Saw but didn't respond: 26



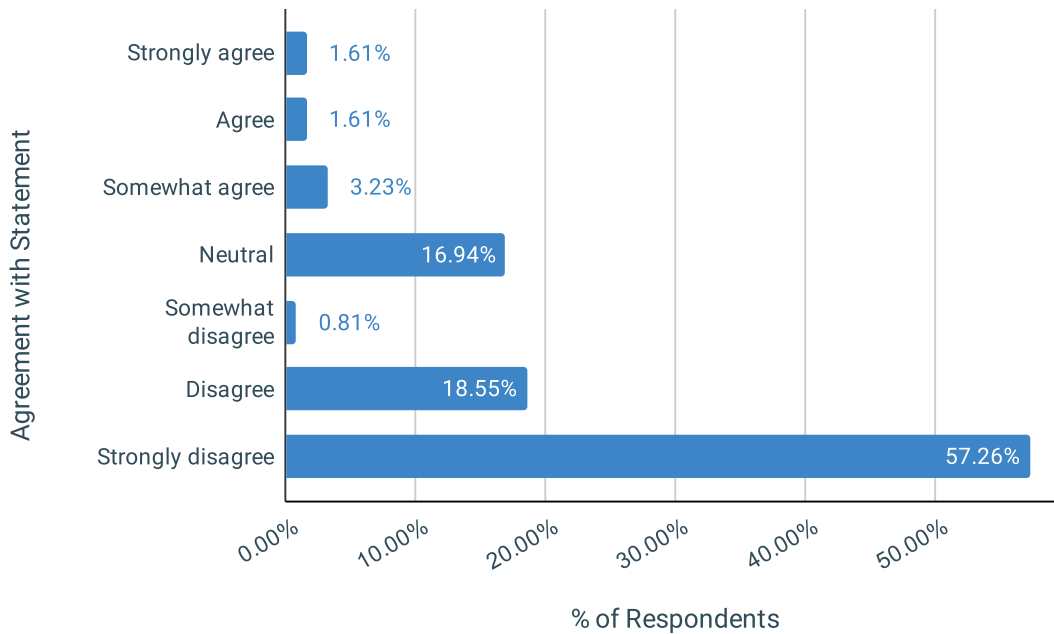
**c. I disagreed with the technical direction of the project**

N = 121 Saw but didn't respond: 27



**d. I had a personal disagreement with some of the participants**

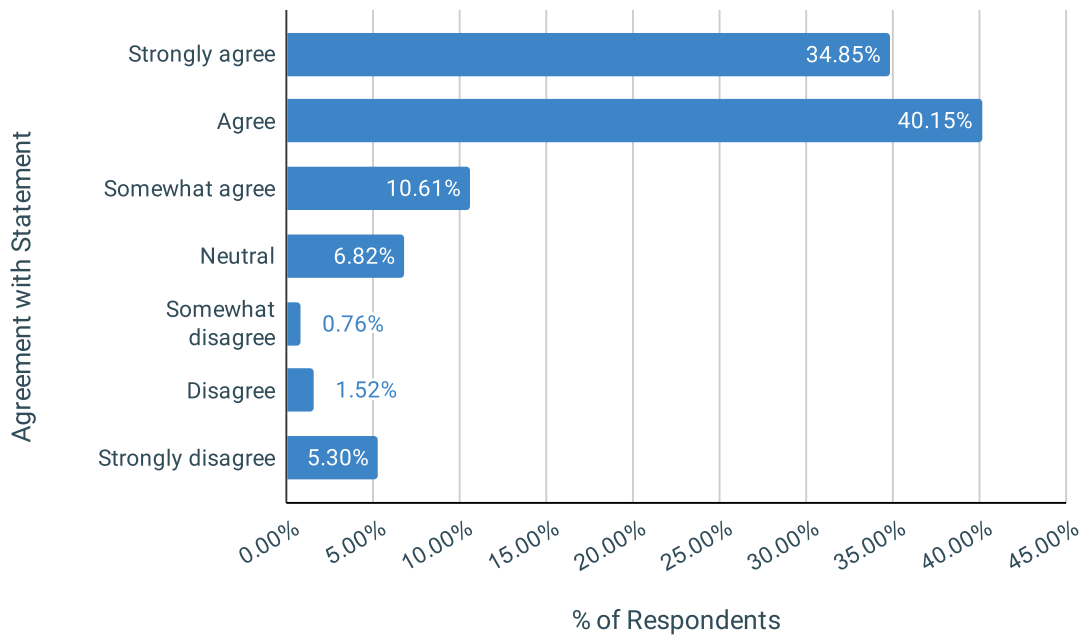
N = 124 Saw but didn't respond: 24





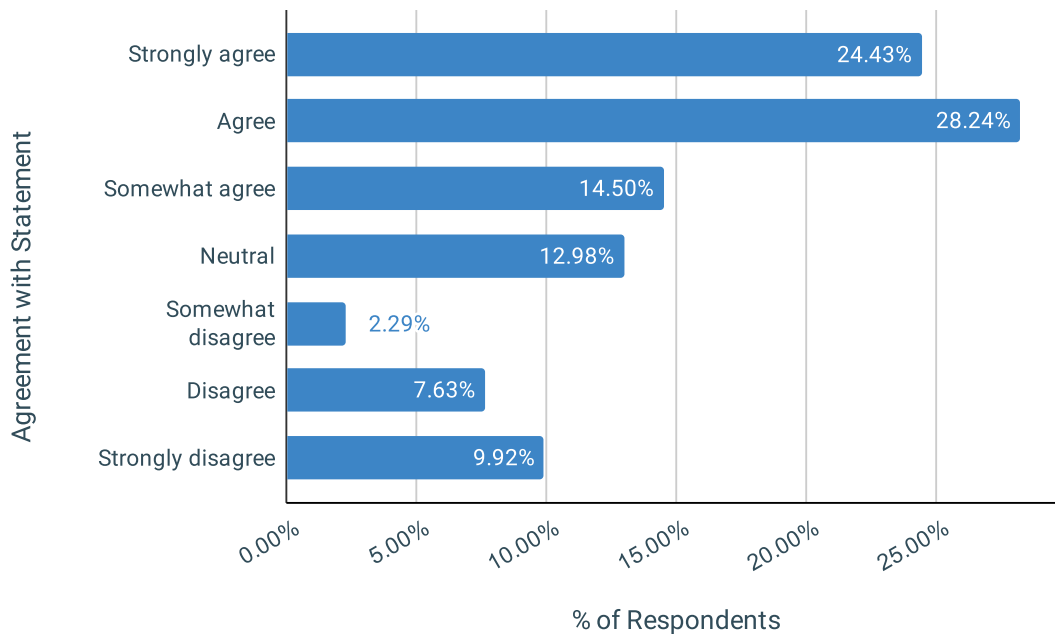
**e. My time became constrained by other professional commitments**

N = 132 Saw but didn't respond: 16



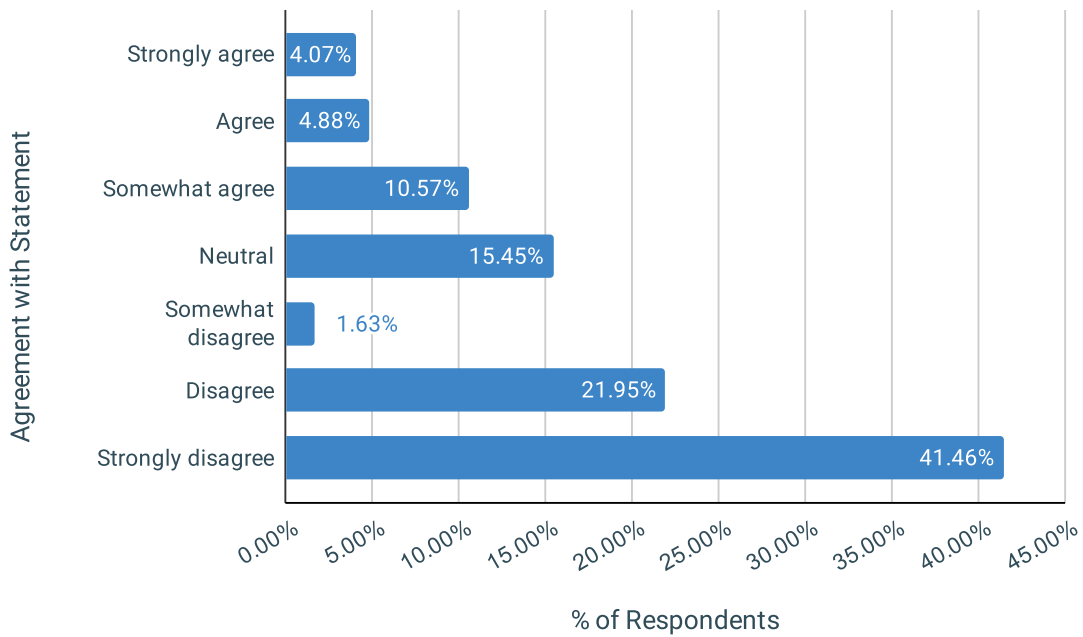
**f. My time became constrained by family or social commitments**

N = 131 Saw but didn't respond: 17



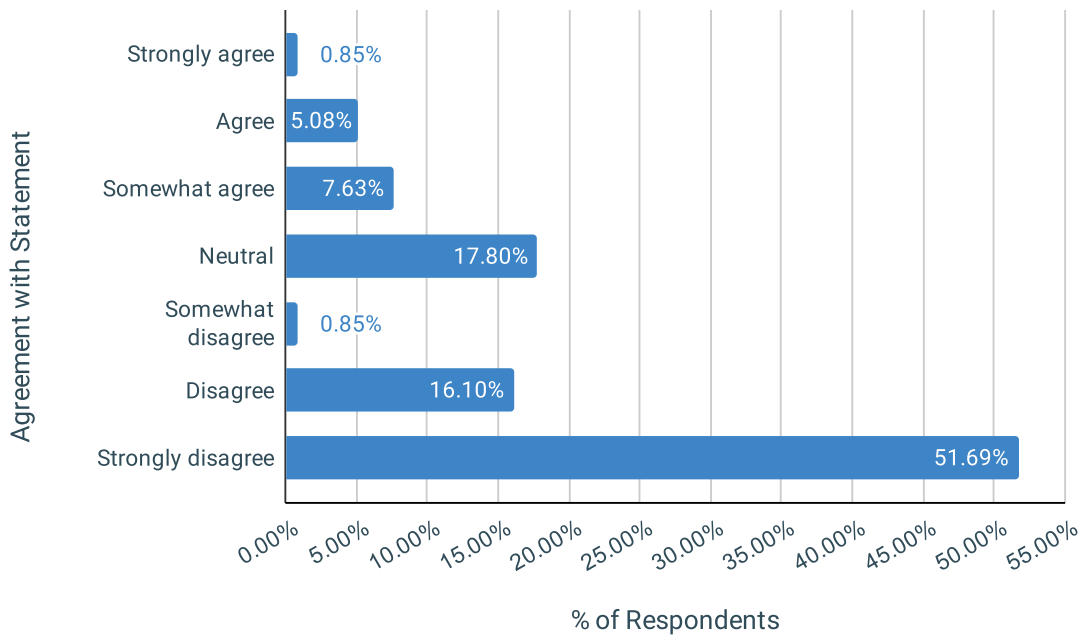
**g. My employment contract prevented me from participating in open source projects**

N = 123 Saw but didn't respond: 25



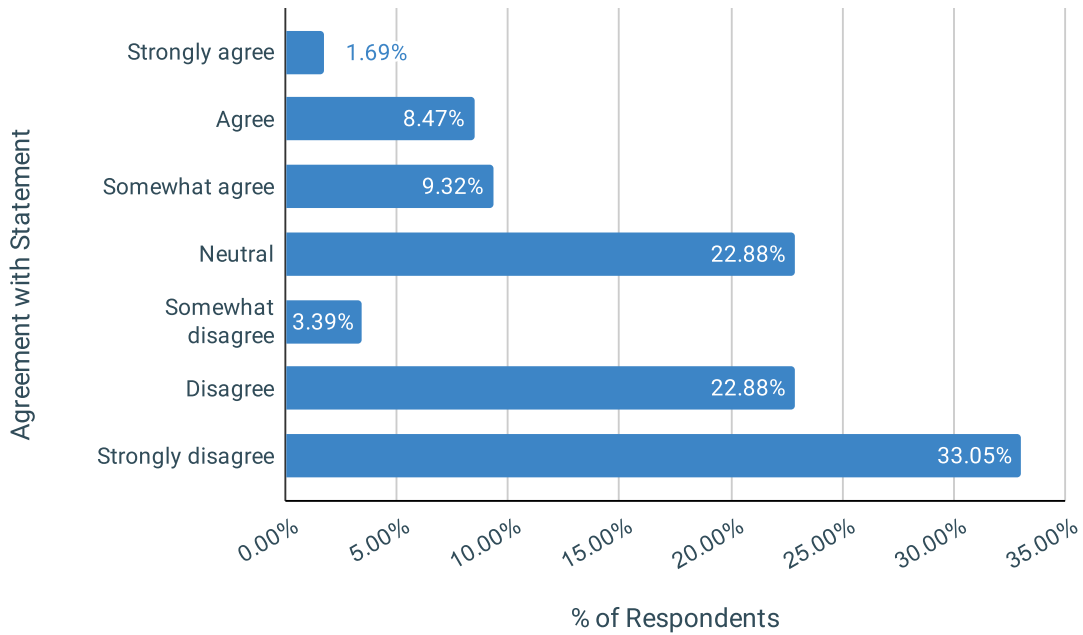
**h. My work supervisor did not approve of my participation in open source projects**

N = 118 Saw but didn't respond: 30



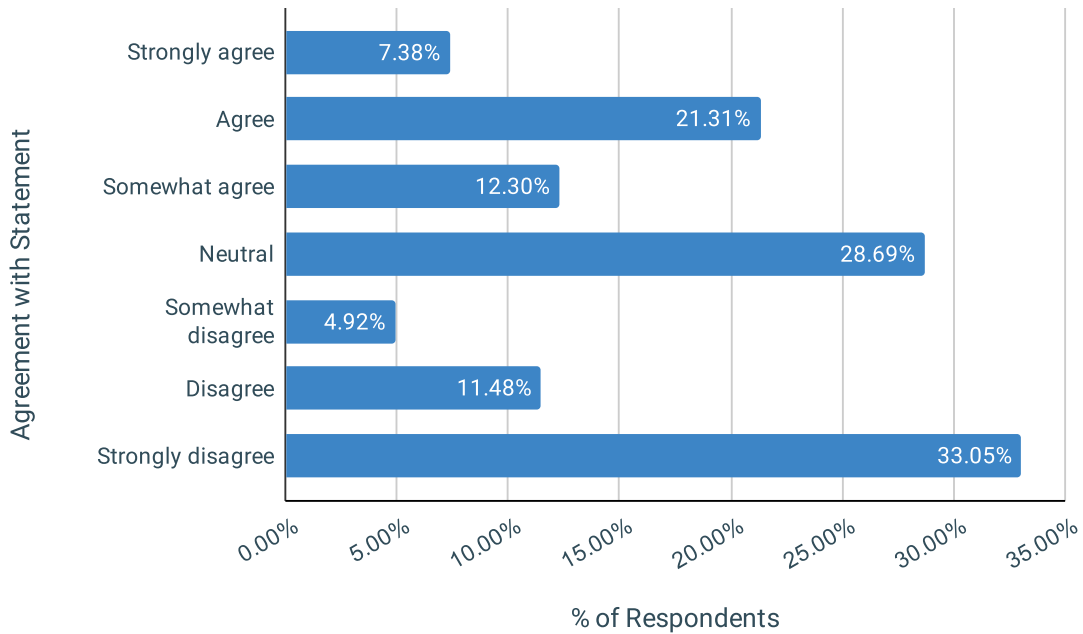
**i. Participation by others in the project stopped**

N = 118 Saw but didn't respond: 30



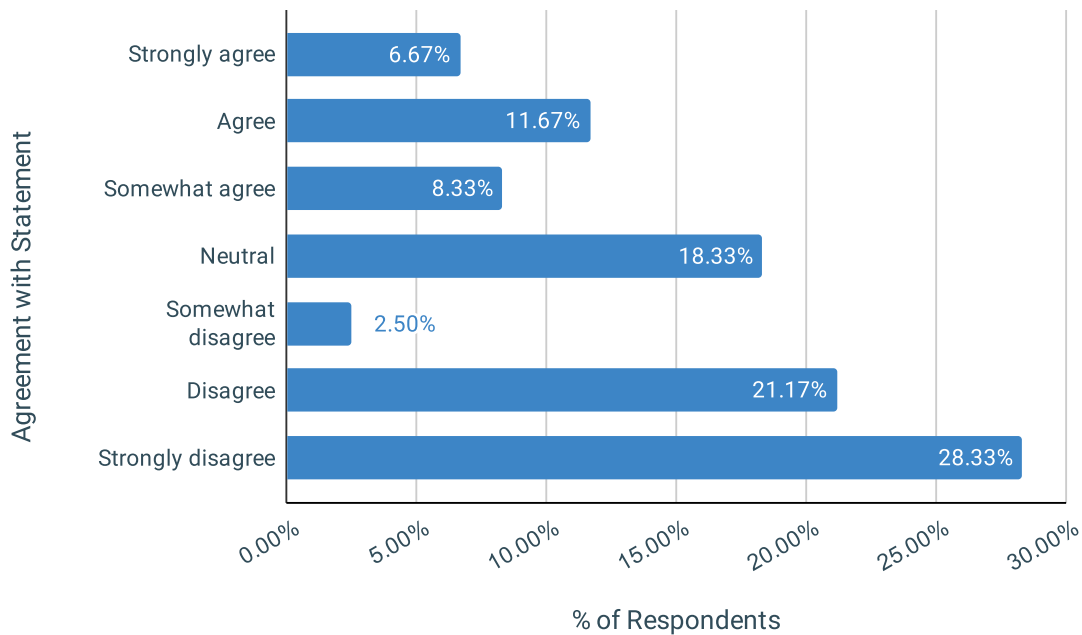
**j. The project accomplished all of its goals that were of concern to me**

N = 122 Saw but didn't respond: 26



**k. My employer's technical needs changed and I was redeployed to another project**

N = 120 Saw but didn't respond: 28



**l. Other (please enter below)**

N = 54 Saw but didn't respond: 94

**Text Response Summary:**

In addition to the common response that time constraints led to the end of their contributions, a few also noted that economic constraints or the end of payment for their contributions were the cause. Several noted that contributions were tied to the respondent's employment; when they changed jobs or companies, their contributions ended. Still others left projects when they lost interest or felt as though they had fixed all the issues that they wanted to change. Perhaps most troubling, but less frequently mentioned, was that contributing had become "unattractive," as their efforts were met with unreasonable demands from end users, attacks from others, as well as "negative personal and professional outcomes."

# Current FOSS Contribution

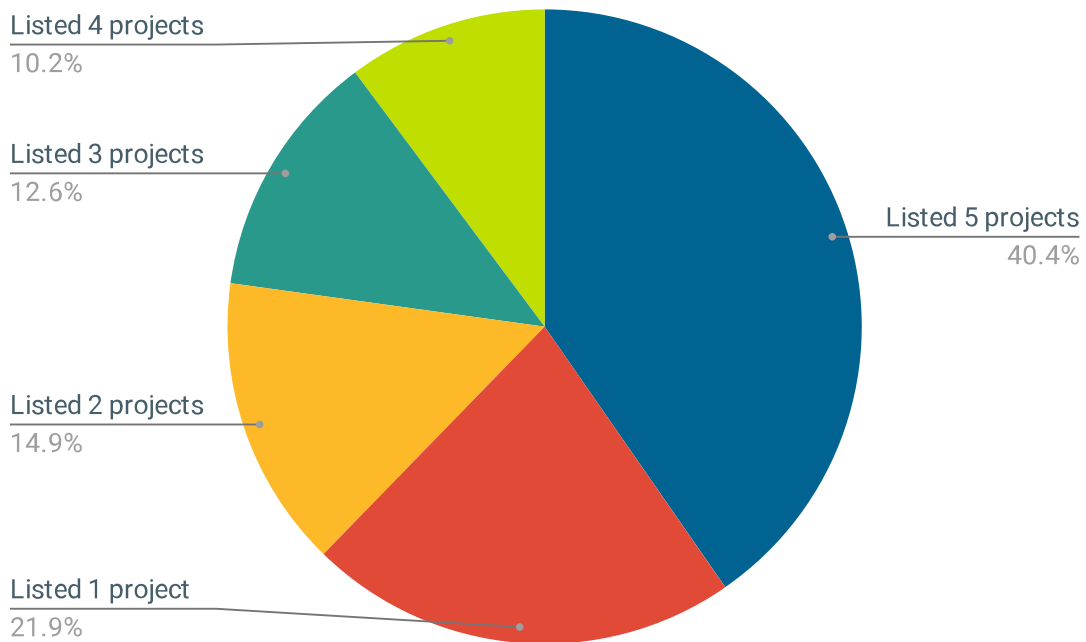
10. Please list the FOSS projects to which you contribute in the formats modeled below. If more than five, please list only the five in which you are currently most active. (e.g., at least one commit in the last 36 months)

**GITHUB** example format for antlr/antlr4: <https://github.com/antlr/antlr4>

**GITLAB** example format for sublime-music: <https://gitlab.com/sublime-music/sublime-music>

**SOURCEFORGE** example format for gpshell: <https://sourceforge.net/projects/globalplatform>

**OTHER:** please just list the full repository url for the project



11. Did you know anyone involved in the following project(s) prior to your participation?

N = 903 respondents answered this question about 2,998 projects

4 individuals saw (but did not answer the question) about 7 total projects

**37.12%** of projects had someone the respondent knew on the project before joining

**62.88%** of projects did not have someone the respondent knew on the project before joining

## 12. In the FOSS projects to which you contribute, which channels are generally used for communication? (please select all that apply)

N = 912 Saw but didn't respond: 11

**90.35%** use issue trackers (e.g., GitHub Issues, BugZilla, etc.)

**56.80%** use email/ mailing lists

**51.10%** use instant messenger (e.g., Slack, IRC, etc.)

**40.79%** use documentation

**21.27%** use video conferencing (e.g., Zoom, Skype, etc.)

**19.30%** use in person

**7.24%** use "Other"

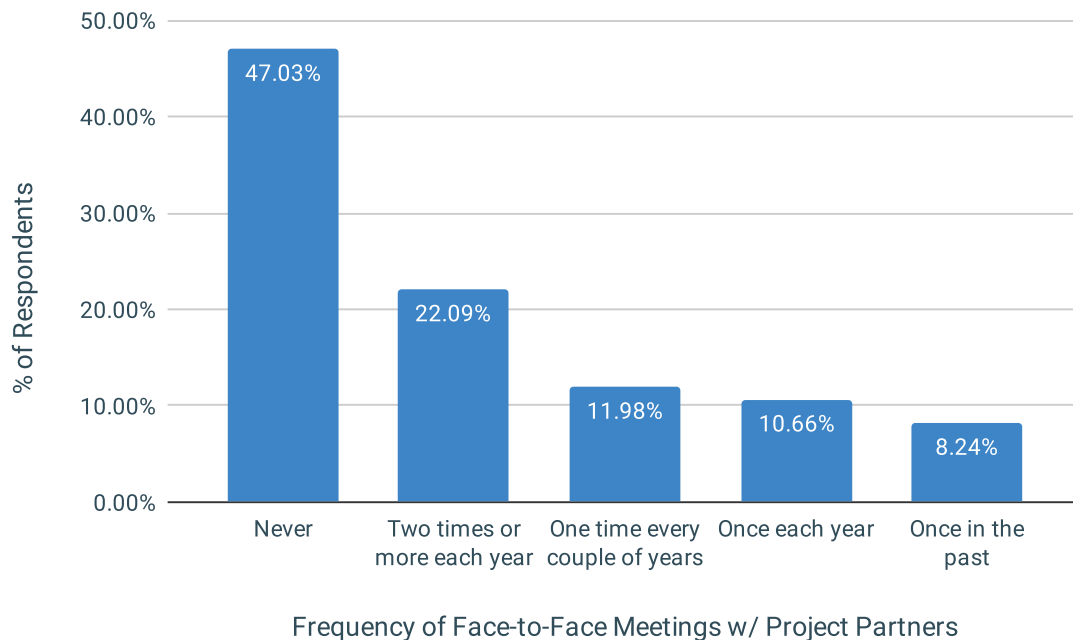
### Text Response Summary (N = 65)

Most text responses entered for "Other" listed various chat, messaging, and networking platforms (including Gitter, WeChat, Discord, LinkedIn etc). One frequently mentioned social media platform used for general communication was Twitter. General forums (e.g., StackOverflow, Reddit) and project-specific forums were often named as well.

Other means of communication listed were audio conferencing or calls, hack weekends and conferences, unit tests, code reviews, pull or merge requests, and online pairing.

## 13. How often in the past have you met face-to-face with FOSS project partners?

N = 910 Saw but didn't respond: 13



**14. In the FOSS projects to which you contribute, which channels are most effective for resolving technical disagreements (e.g., accepting new features, how to fix bugs, etc.)? (please select all that apply)**

N = 896 Saw but didn't respond: 27

**74.22%** use issue trackers (e.g., GitHub Issues, BugZilla, etc.)

**38.06%** use instant messenger (e.g., Slack, IRC, etc.)

**37.83%** use email/ mailing lists

**20.42%** use in person

**17.97%** use video conferencing (e.g., Zoom, Skype, etc.)

**12.28%** use documentation

**2.68%** use "Other"

**Text Response Summary (N = 23)**

While some responses here were similar to those for general communication (including Gitter, Discord, etc), forums tended to be the preferred method. Gerrit code reviews and pull or merge request comments were commonly mentioned as well.

**The next set of questions pertains to your level of engagement in each FOSS project named. Active contributors to a FOSS project are identified as either "maintainers", "core participants", "occasional participants" or "one-time participants."**

*"Maintainers" are software maintainers or package maintainers who are the final decision makers over all or portions of source code that goes into a build or release. Maintainers would likely also identify as a subset of core participants.*

*"Core participants" may have been involved in the project since inception or joined later and regularly participated in major discussions about project direction, and have significant on-going roles in the work, possibly including accepting patches to the code base. Core participants may be referred to as "Committers" in a project community.*

*"Occasional participants" would not normally participate in ongoing or weekly project discussions, but occasionally provide contributions over longer periods of time.*

*"One-time participants" is someone who provides a specific set of suggestions or contributions and then exit involvement once their work is done; these are sometimes called "drive-by commits."*

**Given these definitions, please answer the following questions:**

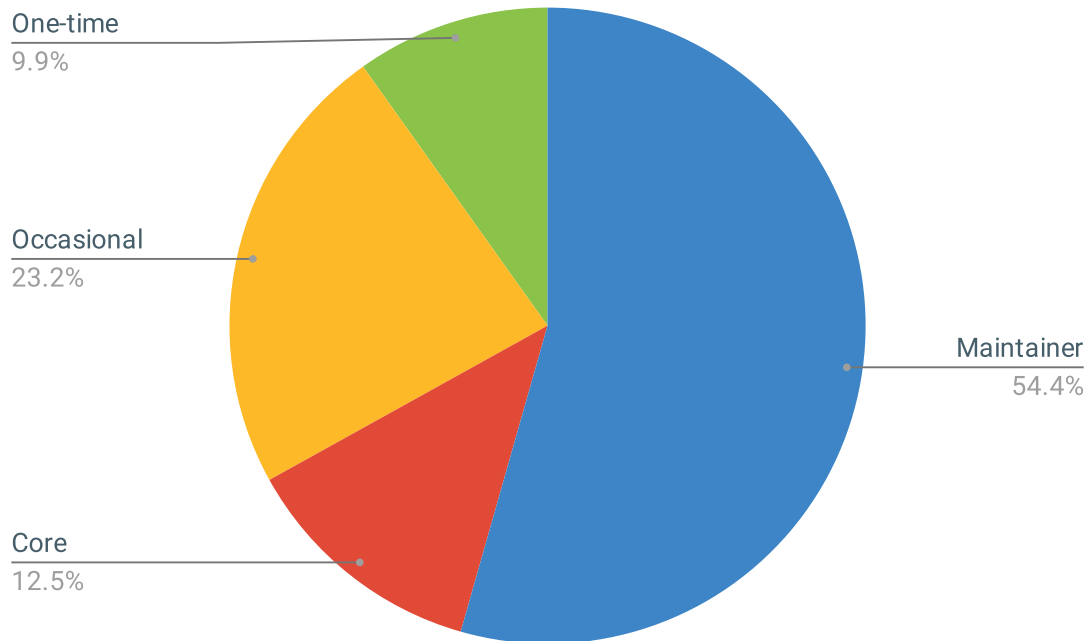
**15. Do you regard yourself as a maintainer, core, occasional, or one-time participant on the following project(s)?**

Respondents were asked to identify their level of contribution for each of the projects they had listed in question 10

N = 853 respondents answered this question about 2,885 projects

4 respondents saw but did not answer the question about 7 total projects

Here the individual's contributor status is determined as the highest status indicated by that individual on any listed project:



**16. Aside from the projects you have already identified, in how many other FOSS projects would you consider yourself to be a maintainer or core participant?**

N = 887 Saw but didn't respond: 20

Number of Projects as Maintainer or Core Participant	% of Respondents
0	52.76%
1-2	22.21%
3-5	13.53%
6-10	4.96%
11-20	2.48%
21-50	1.58%
51-99	0.90%
100+	1.58%

**17. Approximately how many other FOSS projects would you consider yourself to be an occasional participant?**

N = 889 Saw but didn't respond: 18

Number of Projects as Occasional Participant	% of Respondents
0	16.76%
1-2	28.91%
3-5	25.08%
6-10	13.16%
11-20	8.32%
21-50	3.71%
51-99	1.01%
100+	3.04%



### 18. Do you have authority to give commit rights in the project(s) below?

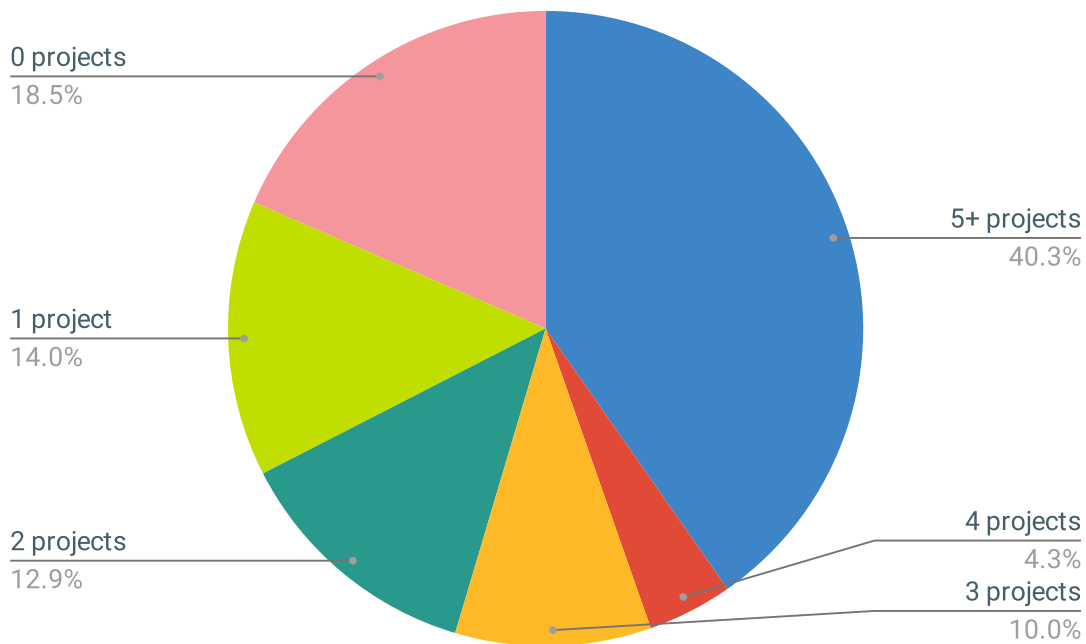
*This question was only shown to respondents who indicated that they were a maintainer or core participant on at least one of the projects they listed in question #15*

N = 475 respondents answered this question about 1,009 projects  
9 respondents saw but did not answer the question about 21 projects

Respondents have commit authority to **81.17%** of the projects  
Respondents *do not* have commit authority to **18.83%** of the projects

### 19. How many total FOSS projects have you had commit rights to?

N = 762 Saw but didn't respond: 13



### 20. When developing software, what are your main sources for security best practices? (please select all that apply)

*Respondents were also asked to elaborate on which specific sources in each resource category they use*  
N = 606 Saw but didn't respond: 169.

#### Responses and Summaries of Entered Text:

##### a. 10.73% In-person Classes

Overwhelmingly, the text responses referred to college and university courses that the respondents had taken, either as part of a degree or an elective. Another popular source were lectures at workshops, camps, meet-ups, conferences, job-related training courses, or even impromptu mentorship from colleagues. Some of the other training courses and certifications mentioned were Sensepost, SANS and Global Information Assurance Certification (GIAC).

**b. 15.51% Corporate training**

Beyond general descriptions of internally organized training sessions on privacy, software security, cybersecurity, data protection and latest industry trends, some respondents specifically cited receiving training on cross-site scripting prevention and secure software development. Formal training often occurred through outside organizations like Open Web Application Security Project's Security Shepherd, Watchcomm, the Linux Foundation, Microsoft, Google, IBM and Red Hat.

**c. 25.91% Open Web Application Security Project — OWASP**

By far the most cited OWASP resource was the Top 10 Web Application Security Risks, followed by the OWASP Cheat Sheet Series and Best Practices guides. Other resources — including the Application Security Verification Standard (ASVS), dependency checker, wikis, video courses, and vulnerability scanning tools — were also mentioned frequently.

**d. 46.54% Online articles / blogs**

Several popular blogs came up in the text responses, including LWN, Y Combinator, Schneier on Security, Google's Project Zero, GitHub's blog, Red Hat security blog, Troy Hunt's blog, Paragon Initiative Enterprises blog, Niebezpiecznik.pl, Heise Online, DZone, and a variety of blogs on Medium. Instead of mentioning specific blogs, others reported finding useful links on Reddit, Hacker News, StackOverflow, StackExchange, Dev.to, Twitter, and Google search results related to their issues of interest. Several respondents also turned to "the documentation of the libraries" they use and project specific blogs, such as the Mozilla Developer Network, Django, and Linux pages. A few "official" sources came up as well, including Common Vulnerabilities and Exposures (CVE), the Internet Engineering Task Force (IETF), and the French National Cybersecurity Agency (ANSSI).

**e. 16.17% Books**

Bruce Schneier's works Applied Cryptography and Cryptography Engineering featured prominently in the respondents' lists. Beyond those two books, only a handful of others received multiple mentions, including the following:

- Security Engineering: A Guide to Building Dependable Distributed Systems by Ross Anderson
- UNIX Network Programming by W. Richard Stevens
- The Art of Unix Programming by Eric S. Raymond
- The Linux Programming Interface: A Linux and UNIX System Programming Handbook by Michael Kerrisk
- Writing Secure Code by Michael Howard

**f. 50.66% Forums, like Stack Overflow or Reddit**

By far, StackOverflow was the forum that respondents most often reported consulting for security best practices. Other sites in the Stack Exchange Network also featured prominently, including those dedicated to cryptography, software engineering, and information security. Respondents mentioned Reddit second most often citing several subreddits like r/programming/,

r/reverseengineering, r/crypto, r/netsec, r/security, and r/hacking. A few other sites came up in responses, such as Discord, Baeldung, Google groups, Lobste.rs, and other project- or language-specific forums.

**g. 8.42% CII Best Practices Badge**

*No text responses were requested when respondents selected this option*

**h. 8.75% Online Classes**

Online courses related to security on Coursera, Pluralsight, Udemy, and Udacity appeared frequently in respondents' text entries. Again, social media channels like YouTube and LinkedIn (via the Learning platform) received significant mentions. Some respondents cited the official documentation, tutorials and guidance from security team members as their preferred "classes."

**i. 7.26% Podcasts**

Some of the podcasts that respondents cited most often were This Week in Tech's Security Now and FOSS Weekly. Other notable inclusions were CppCast, Changelog's GoTime, Talk Python to Me, Software Engineering Radio, Open Source Security Podcast, and Säkerhetspodcasten. Two YouTube channels — LiveOverflow and Computerphile — appeared as well.

**j. 13.20% Mailing Lists**

Unsurprisingly, most text responses cited project-specific lists most often. However, there were a few broader, security-related mailing lists that received specific attention: <https://seclists.org/fulldisclosure>, <https://seclists.org/oss-sec/>, and <https://www.openwall.com/lists/oss-security/>. Security-specific announcements from CVE, Debian, SuSE, OpenJDK, RedHat, Linux Kernel, and Apache were also highlighted on the list.

**k. 30.03% Other**

One of the most valuable resources was identified by one respondent as "experience combined with common sense." Along with formal education and on-the-job training, these resources are substantial time commitments and may not be possible for every contributor. However, the second most cited resource for security best practices was advice from other FOSS contributors, co-workers, and friends in cybersecurity. Peer review — whether in person, over IRC, via Request For Comments (RFCs) or through mailing lists — was another resource several respondents mentioned. One respondent pointed out that "working through problems collaboratively" was invaluable: "I without a doubt couldn't know what I know about security today if it weren't for all the people who were willing to point out errors, listen to me walk through potential security issues, or correct me by pointing out articles of current best practices."

Conferences, hacking challenges, capture the flag games, and other collaborative learning experiences were cited as well. Even for individuals that are unable to attend these events in real time, videos of the talks and presentations online provide valuable information that contributors can access at their convenience.

A third useful resource that many respondents noted was the integration of automated security tools and tests into continuous integration pipelines. They specifically identified static analysis tools and penetration test tools as helpful for improving security.

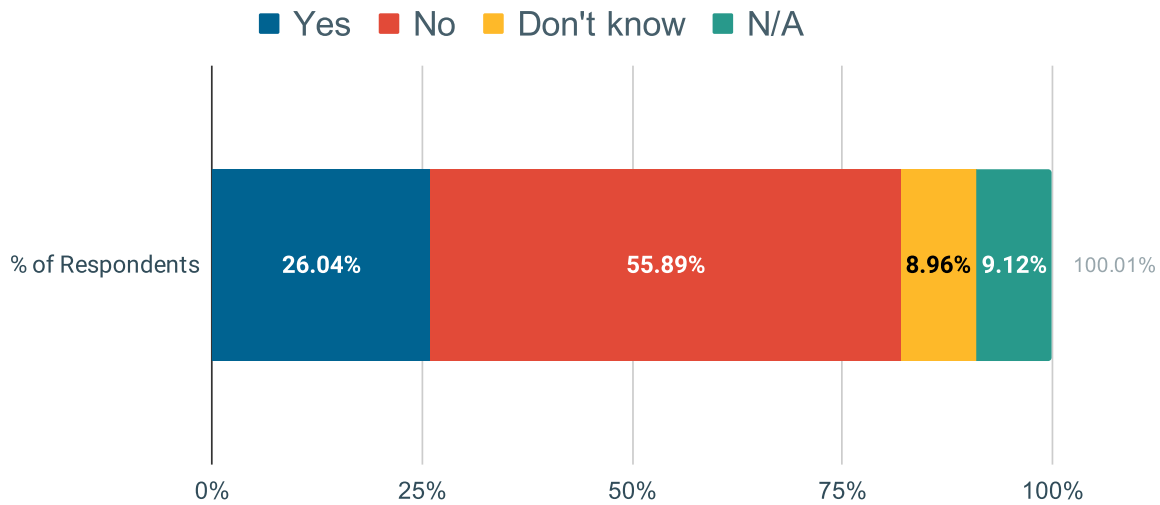
Not all respondents were enthusiastic about discussing security best practices. One respondent said that they “find security an insufferably boring procedural hindrance.” Another is quoted in full here: “I find the enterprise of security a soul-withering chore and a subject best left for the lawyers and process freaks. I am an application developer.”

**21. For the FOSS projects where you are a maintainer or core participant, do you have the following security processes in place?**

*This question was only shown to respondents who indicated that they were a maintainer or core participant on at least one of the projects they listed in question #15*

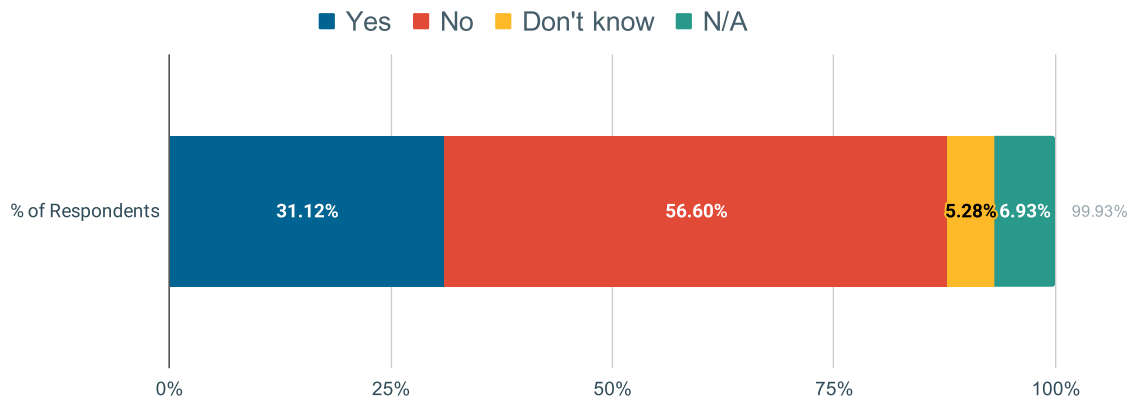
**a. Do you have a security policy in place?**

N = 312 respondents answered the question about 603 projects  
 415 respondents saw but did not answer the question about 1,202 projects



**b. Do you have a vulnerability disclosure policy in place?**

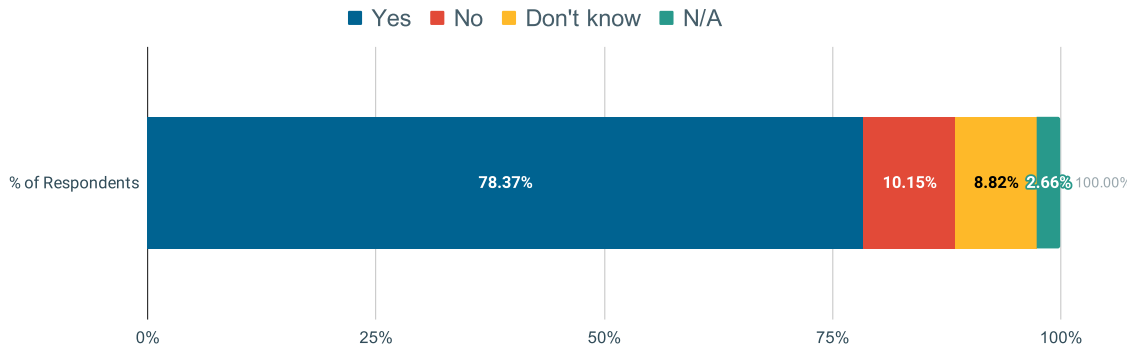
N = 314 respondents answered the question about 606 projects  
 415 respondents saw but didn't answer the question about 1,199 projects



**c. Do you have support for SSL/TLS on the website?**

N = 316 respondents answered the question about 601 projects

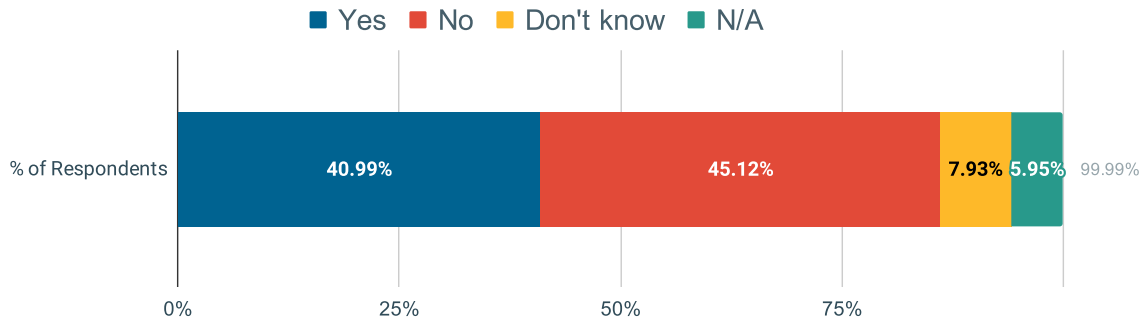
414 respondents saw but didn't answer the question about 1,204 projects



**d. Is there a maintainer/core participant with a security focus?**

N = 315 respondents answered the question about 605 projects

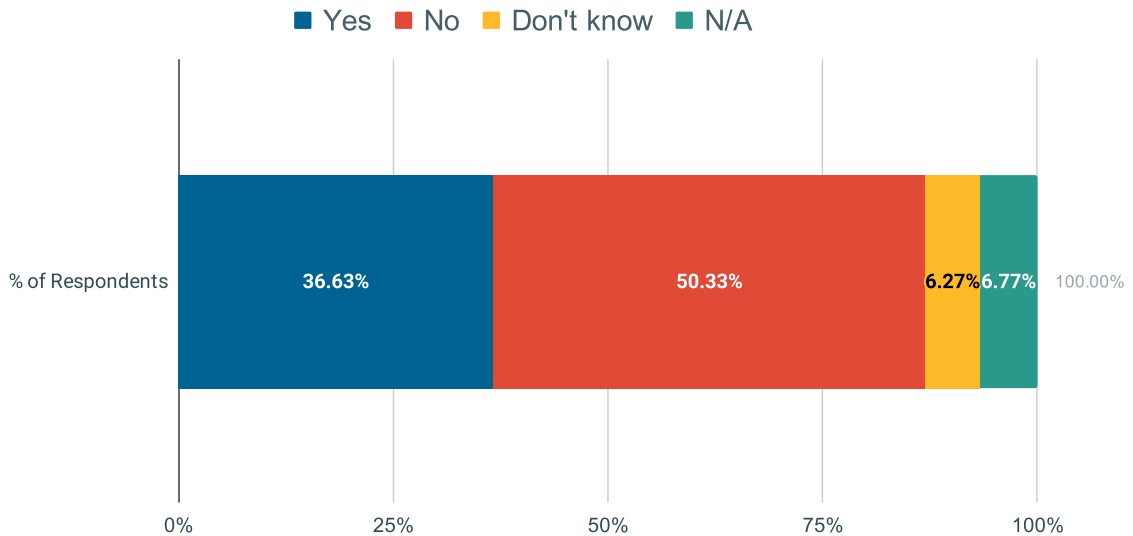
415 respondents saw but didn't answer the question about 1,200 projects



**e. Do you use a static analysis tool?**

N = 317 respondents answered the question about 606 projects

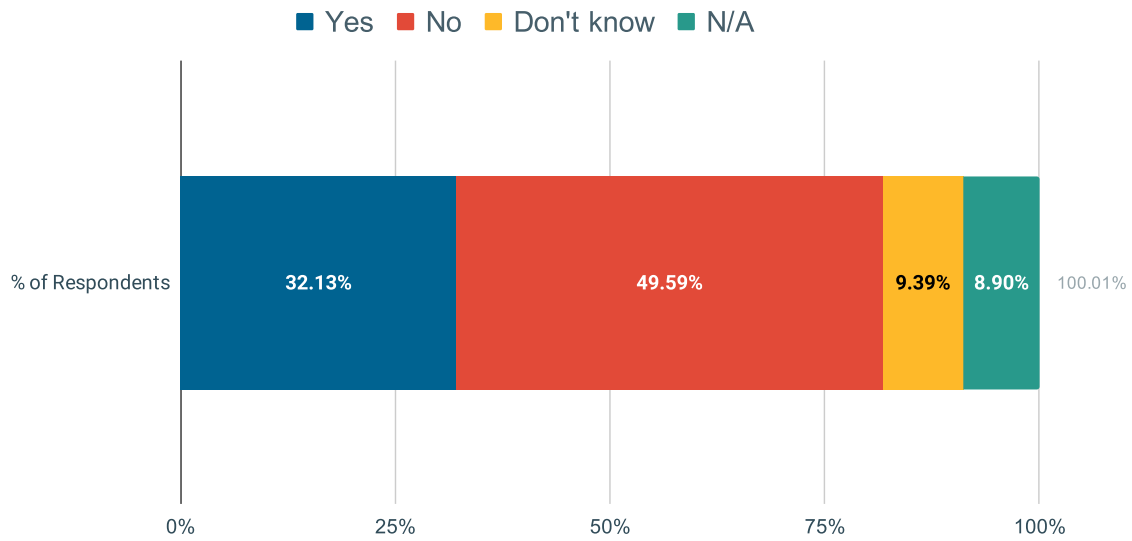
413 respondents saw but didn't answer the question about 1,199 projects



**f. Do you use a software component/dependency analysis tool?**

N = 315 respondents answered the question about 607 projects

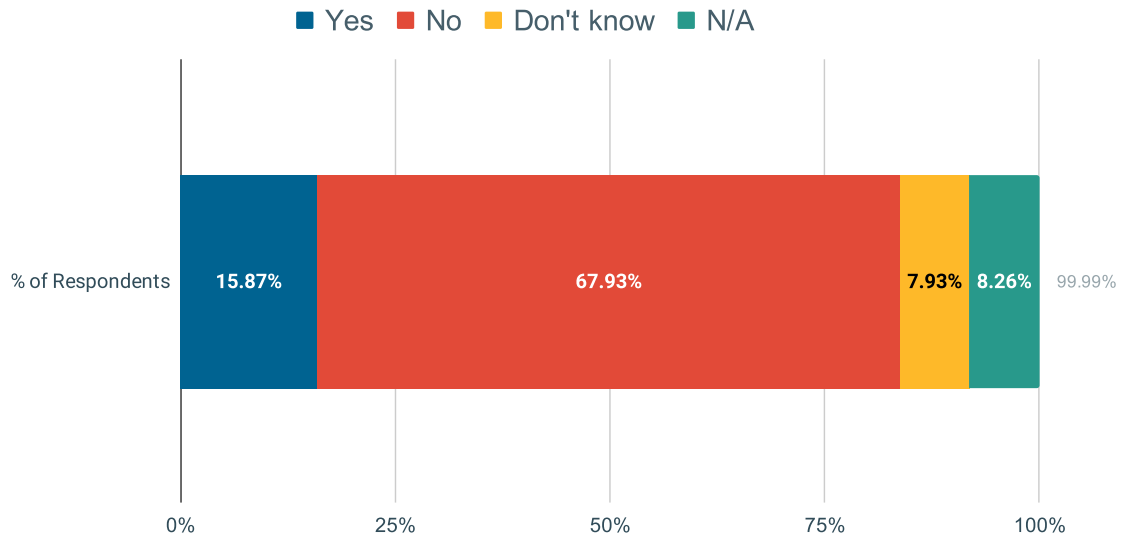
414 respondents saw but didn't answer the question about 1,198 projects



**g. Do you use a dynamic analysis tool?**

N = 312 respondents answered the question about 605 projects

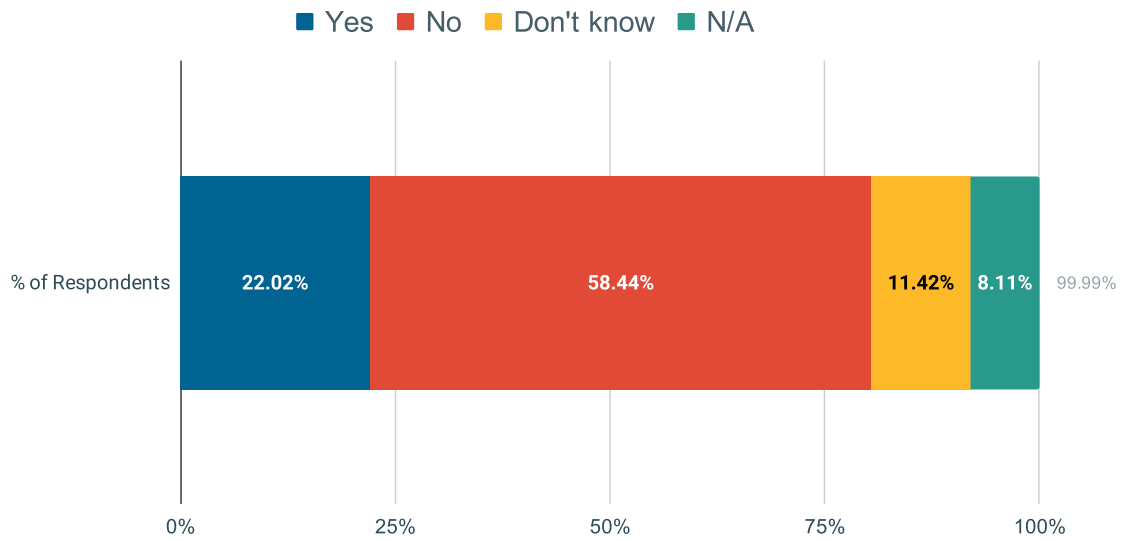
415 respondents saw but didn't answer the question about 1,200 projects



#### h. Do you use a threat model for the project?

N = 314 respondents answered the question about 604 projects

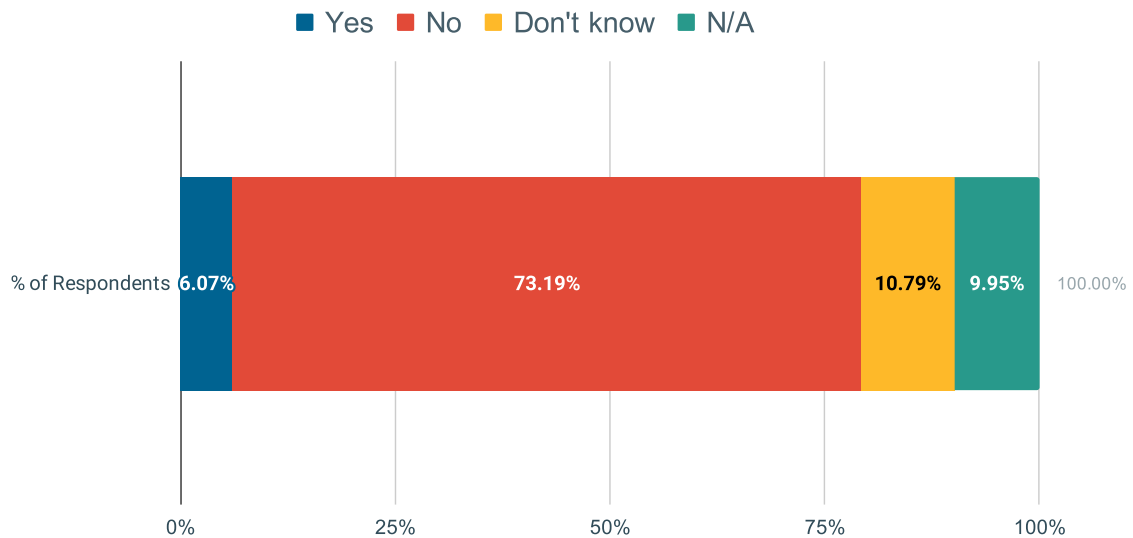
415 respondents saw but didn't answer the question about 1,201 projects



#### i. Does the project have a CII Best Practices Badge?

N = 308 respondents answered the question about 593 projects

416 respondents saw but didn't answer the question about 1,212 projects



**22. For each of the following security practices, please indicate which specific tools do you use:**

*This question was only shown to respondents who reported that one or more of the projects to which they contribute used the named security practice(s) in question #21*

**a. Static Analysis Tools (e.g., Coverity Scan)**

Text responses to this question overwhelmingly cited Coverity Scan and clang security checkers, like Clang Static Analyzer, clang-tidy, and clang-analyze. SonarQube and SonarCloud appeared several times as well. Other respondents cited Cppcheck, LGTM, a variety of linting tools (e.g., pylint and ESLint), and GitHub code scanning. Custom internal tools were included in the list also, though less frequently. Often respondents stated that they used multiple static analysis tools.

**b. Dynamic Analysis Tools (e.g., OWASP ZAP)**

Tools like Google's OSS-Fuzz, syzkaller, and LLVM's libFuzzer topped the list of text responses for this question. Valgrind, american fuzzy lop, and the Go Toolchain were also mentioned frequently, along with various sanitizers like ASan, UBSan, MSan, KASAN, and clang sanitizers.

**c. Software Component / Dependency Analysis Tools (e.g., Snyk)**

Security features offered through GitHub frequently appeared in these responses, including Dependabot, security alerts, vulnerability warnings and scans. The second most often cited tool was Snyk (others like WhiteSource, Sonatype's Nexus, and Synopsys' BlackDuck CoPilot were mentioned as well, though not as frequently). Respondents also cited npm audit and OWASP Dependency-Check. However, in the words of one respondent, "tools are pretty bad in that they misidentify vulnerabilities very often which is very frustrating as a maintainer."

**23. Do the FOSS projects you work on require or strongly encourage digital signatures on commits to be accepted (e.g., as pull requests, merge requests, and/or patches), to know who (by real name or username) is proposing the change?**

N = 722 Saw but didn't respond: 21

- 50.42% No, none of them do
- 32.27% Some of them do, some do not
- 17.31% Yes, all of them

**24. Do the FOSS projects you work on perform digital signatures on released versions (e.g., use cryptographically signed git tags ("git tag -s") or cryptographically signed release packages wherever the project releases the package), so that recipients can verify who released it even if the distributing repo might be subverted?**

N = 720 Saw but didn't respond: 23

- 41.53% Some of them do, some do not
- 35.97% No, none of them do
- 22.5% Yes, all of them do



**25. Please select the projects that do not require digital signatures for change requests to be accepted.**

*Only respondents who answered “Some of them do, some do not” to question #24 saw this question. They were asked to select which of the projects they had listed in question #10 do not require digital signatures. For security reasons, the results of this survey question are not included in this public report.*

**26. For the projects that you indicated do not require digital signatures, what influenced that decision?**

*This question was only shown to respondents who reported that one or more of the projects to which they contribute do not use digital signatures in questions #23 and #24. For security reasons, the results of this survey question are not included in this public report.*

N = 405 Saw but didn't respond: 206

**Text Response Summary**

Many respondents replied that there were no specific reasons, simply inertia and/or acceptance of the default (no digital signatures) of the platform they use. Others mentioned that the measure seemed unnecessary — either due to the size of the project, current review processes, trust in the platform used for commits, the perception that risks are low, or difficulty of faking identities. As one respondent explained, “Trust is placed in the subsystem maintainers who review, sign-off, and forward changes, and in the public review system, rather than trusting individual contributors.”

Several respondents expressed concern that the additional barrier could deter contributors (both new and existing), given the time and effort required to implement the system and educate contributors on how to use it. “No projects require it, because the friction that [it] would cause for contributions is never worth the security benefits of having it.”

**27. What would encourage you to require digital signatures in projects that don't currently?**

*This question was only shown to respondents who reported that one or more of the projects to which they contribute do not use digital signatures in questions #23 and #24*

N = 387 Saw but didn't respond: 225

**Text Response Summary**

Many respondents replied that they would be motivated to require digital signatures if it were easier to implement, e.g., “an easy process that I can use, understand, and insist upon for contributors to supply them over channels like github.” Again, many expressed concern over the requirement becoming a barrier for would-be FOSS contributors, due to the lack of a “user-friendly workflow.” A significant portion also pointed out that they do not see the benefit of digital signatures when code reviews would reveal any “bad-faith contributions.” One respondent stated that they “suspect most security failures are due to buffer overflow than nefarious people hacking git commits. Think you'd get a lot more secure software if you banned C than if you required signatures.” Some respondents indicated that they were unsure of how to implement digital signatures and would benefit from a tutorial.

**28. Do the FOSS projects you work on require the use of two-factor authentication (2FA) at some point (e.g., on login, merge acceptance, push to public tree, or using a previously-unseen device) in order to accept a change request (e.g., pull request, merge request, or patch) in order to reduce the risk of accepting a malicious change from an account with a stolen password? Examples would include using the GitHub organizational setting “Require two-factor authentication for everyone in the organization” or ensuring 2FA is enabled for all users who can accept change requests.**

N = 654 Saw but didn't respond: 34

- 47.55%** No, none of them do
- 32.11%** Some of them do, some do not
- 20.34%** Yes, all of them

**29. Do the FOSS projects you work on require the use of two-factor authentication (2FA) at some point (e.g., on login, merge acceptance, push to public tree, or using a previously-unseen device) to release a new version in order to reduce the risk of a malicious release from an account with a stolen password?**

**Examples would include using the GitHub organizational setting “Require two-factor authentication for everyone in the organization” or requiring 2FA to release to a language-specific package repository.**

N = 643 Saw but didn't respond: 45

- 44.95%** No, none of them do
- 32.66%** Some of them do, some do not
- 22.4%** Yes, all of them

**30. For the projects that you indicated do not use two-factor authentication (2FA) to accept change requests or release new versions, what influenced that decision?**

*This question was only shown to respondents who reported that one or more of the projects to which they contribute do not use digital signatures in questions #27 and #28*

N = 323 Saw but didn't respond: 191

#### **Text Response Summary**

The majority of respondents indicated that not including 2FA was a lack of decision rather than a decision. They said they were either unaware it was an option or that because it is not the default behavior, it was not considered, or was considered too restrictive to require. “It wasn't a decision, it was the default.”

**31. What would encourage you to use two-factor authentication (2FA) in projects that don't currently?**

*This question was only shown to respondents who reported that one or more of the projects to which they contribute do not use digital signatures in questions #27 and #28*

N = 387 Saw but didn't respond: 225

### **Text Response Summary**

Respondents indicated that they would be incentivized to use 2FA if there was “evidence of need for low-risk projects.” Many said that the projects they work on have only a handful of contributors and they do not see the need for 2FA on small projects. Others pointed out that they find the process difficult to use and said that they would like it to “work smoothly with git-push.” Respondents do not want to add to the barriers that stop people from contributing to FOSS.

One respondent said, “Adding extra hoops through which to jump would be detrimental to the project generally. Our goal is to make the contribution process as easy as possible while limiting who has commit access so that we can sufficiently read over the code being contributed, specifically for [that] sake.”

## **32. What steps would you recommend that the Linux Foundation take to help improve FOSS security?**

N = 380    Saw but didn't respond: 285

### **Text Response Summary**

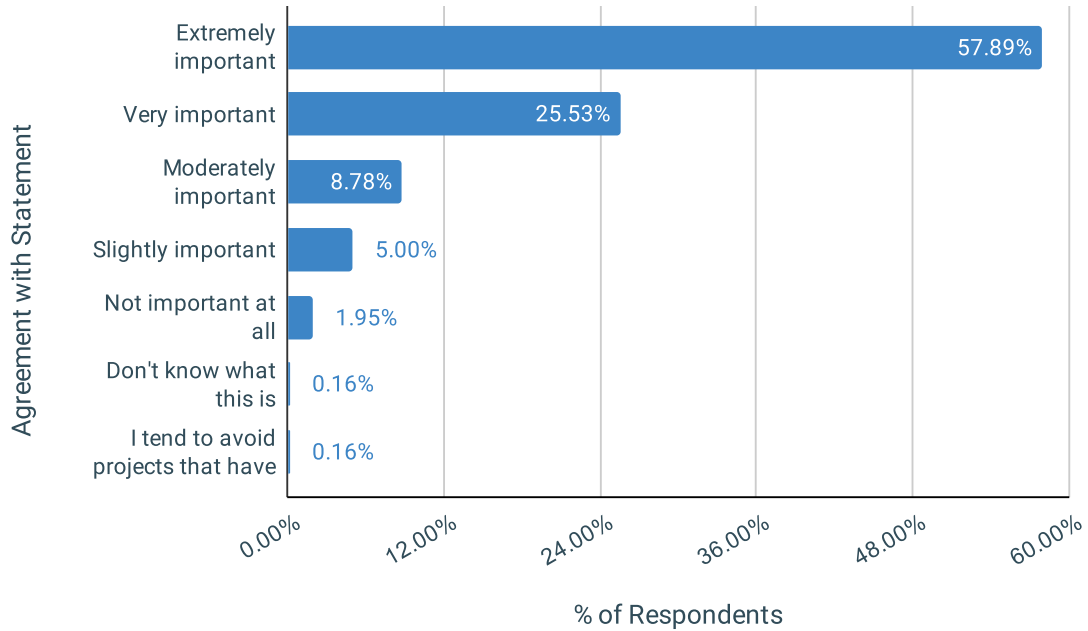
Many respondents mentioned that the Linux Foundation should provide training and free resources promoting and simplifying static analysis, security scans, best practices, digital signatures, and 2FA. The idea of automating security checks was also brought up by several respondents, including automated health checks that could be run by contributors and maintainers.

“I think companies like GitHub are in a great position to try and tackle that by providing some sort of “validated package.” GitHub provides a full ecosystem — repos, build agents and package feeds. So a validated package could be built only using validated packages, not have access to the internet during build, signed by a trusted party, etc.”

### 33. When thinking about whether to contribute to a FOSS project, how important are the following things?

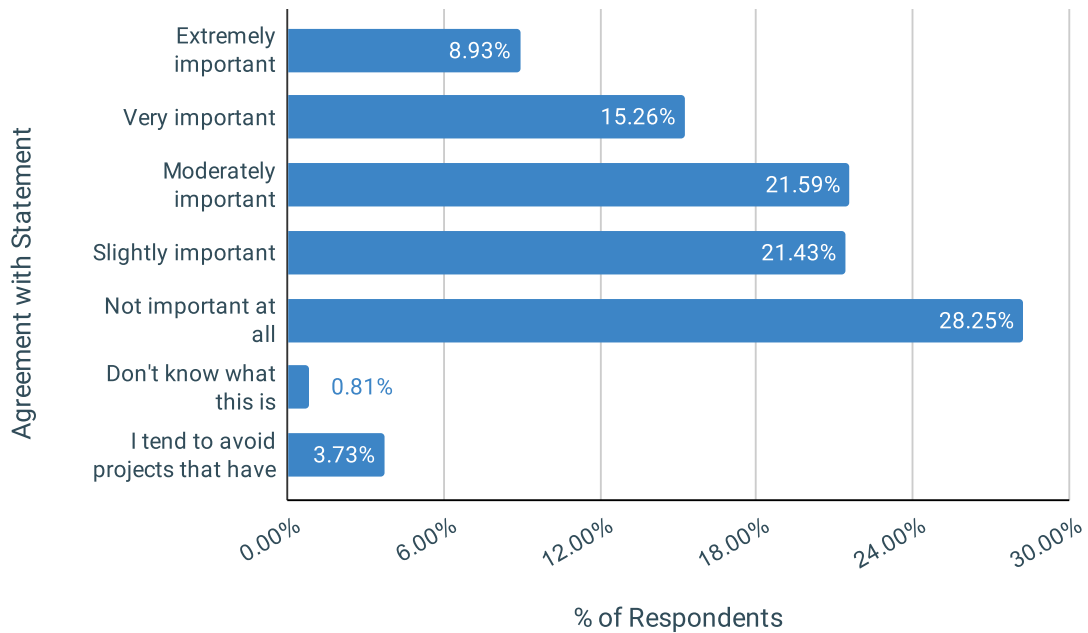
#### a. Presence of an open source license

N = 615 Saw but didn't respond: 24



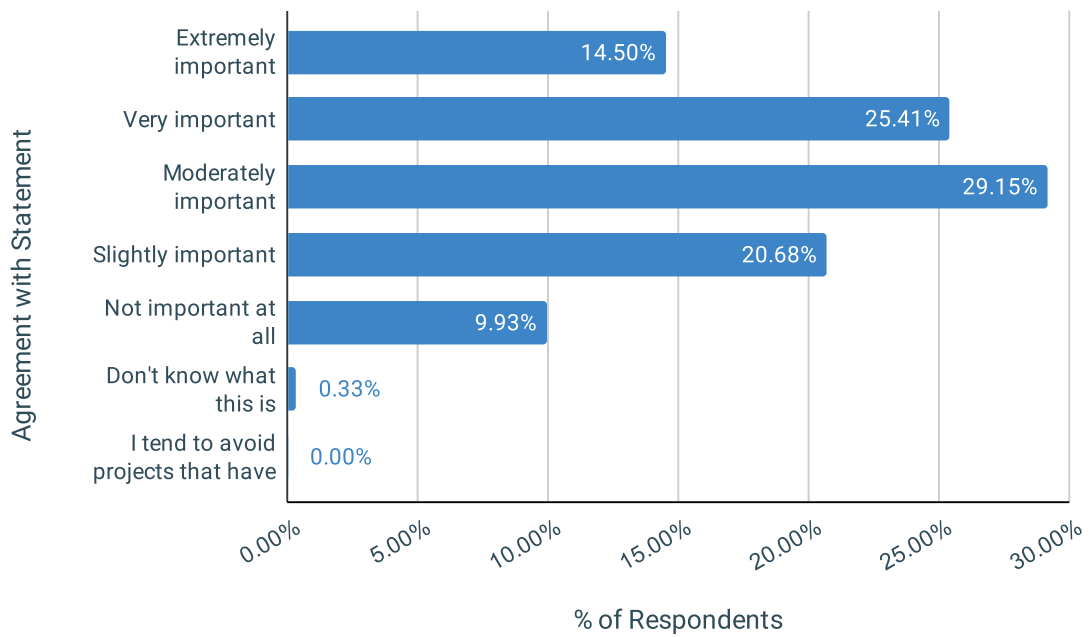
#### b. Presence of a code of conduct

N = 616 Saw but didn't respond: 23



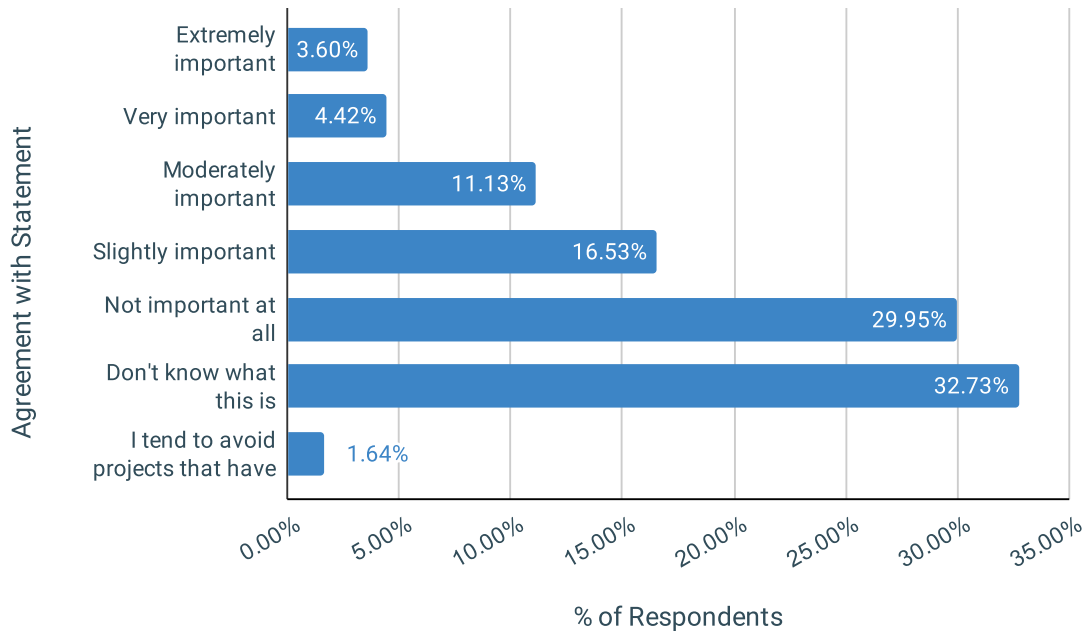
**c. Presence of a contributing guide**

N = 614 Saw but didn't respond: 25



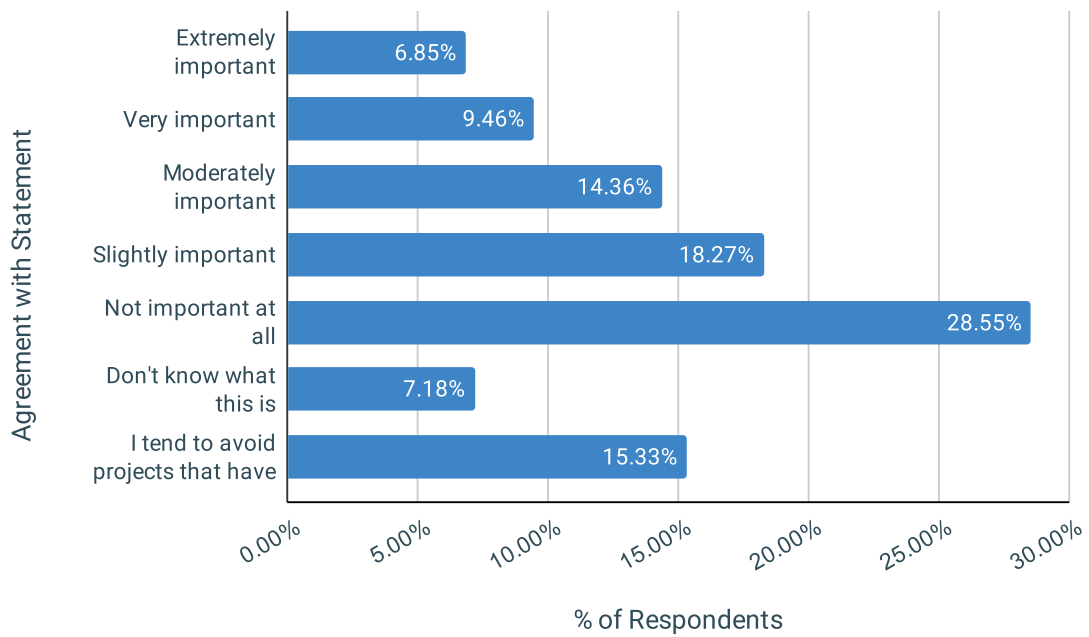
**d. Presence of a developer certificate of origin (DCO)**

N = 611 Saw but didn't respond: 28



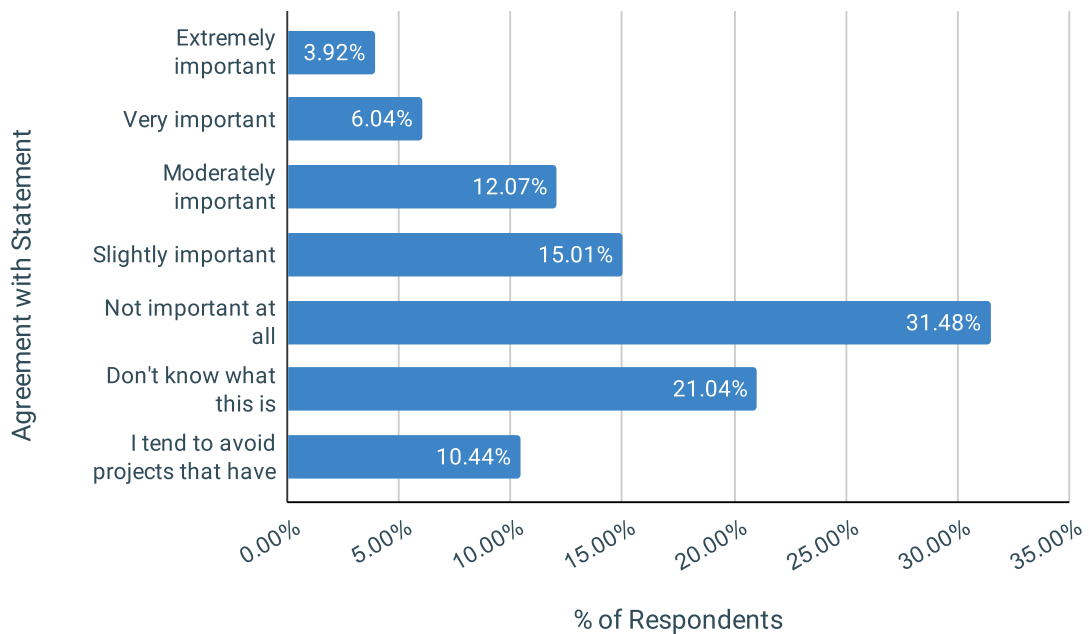
**e. Presence of a contributor's license agreement (CLA)**

N = 613 Saw but didn't respond: 26



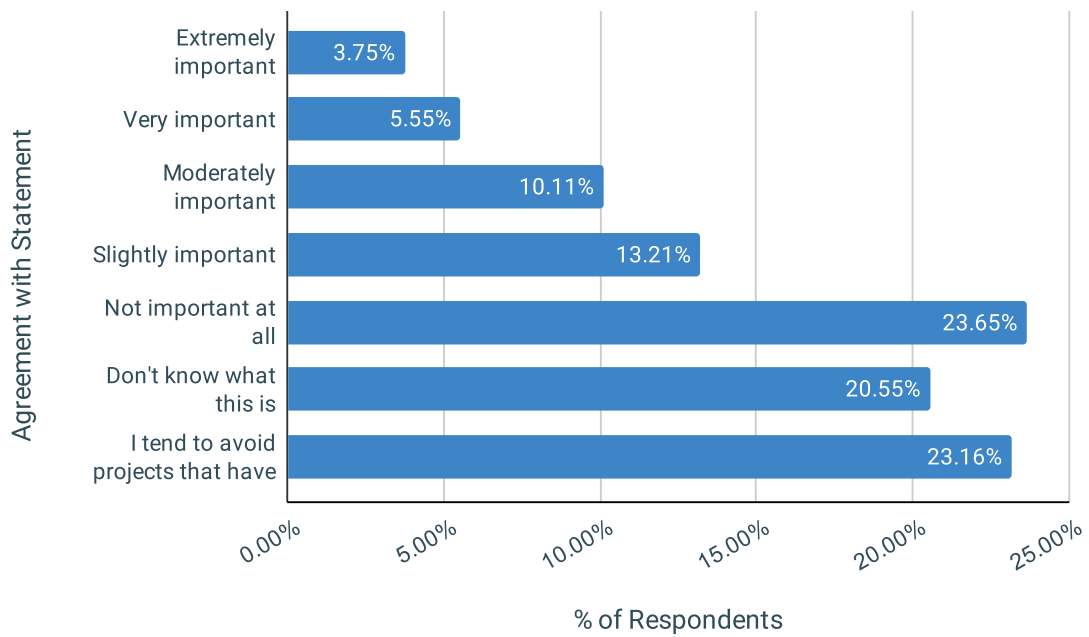
**f. Presence of a contributor's assignment agreement (CA) to a non-profit**

N = 613 Saw but didn't respond: 26



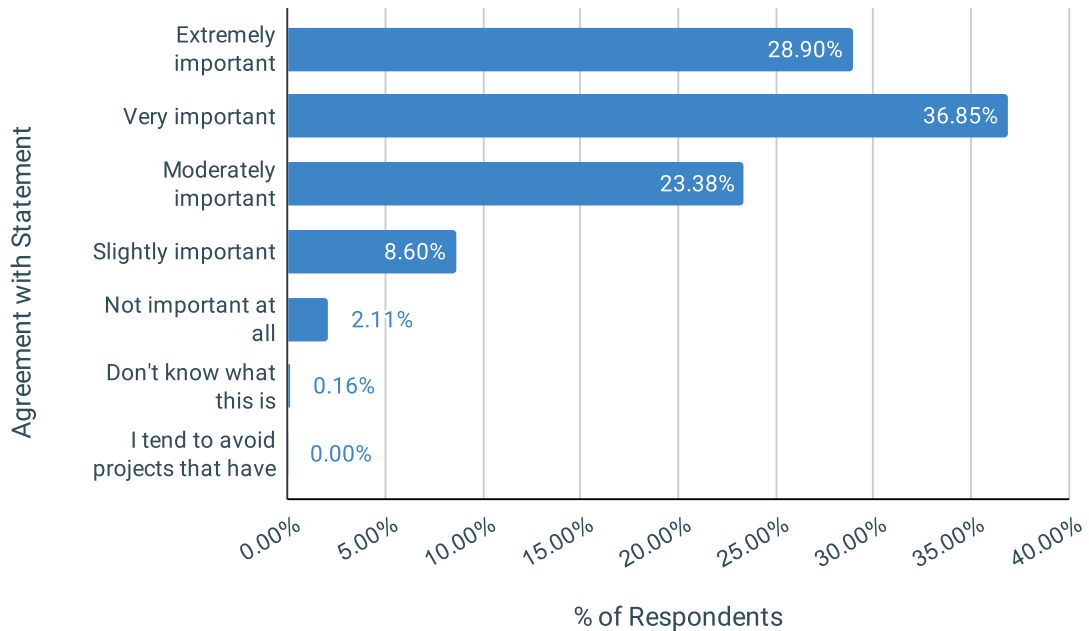
**g. Presence of a contributor's assignment agreement (CA) to a for-profit**

N = 613 Saw but didn't respond: 26



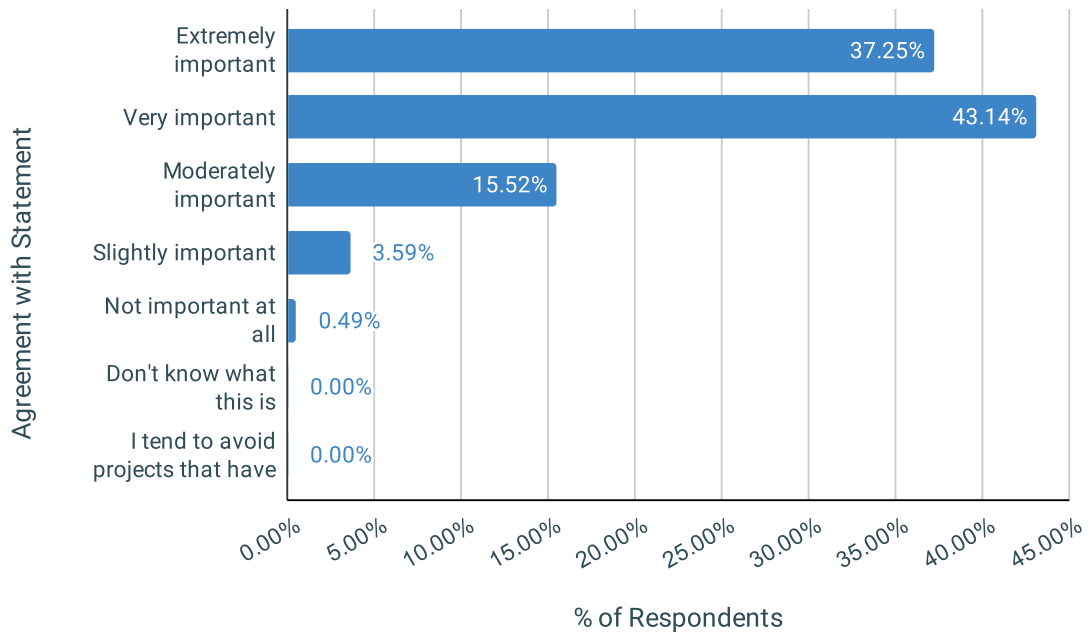
**h. Active development**

N = 616 Saw but didn't respond: 23



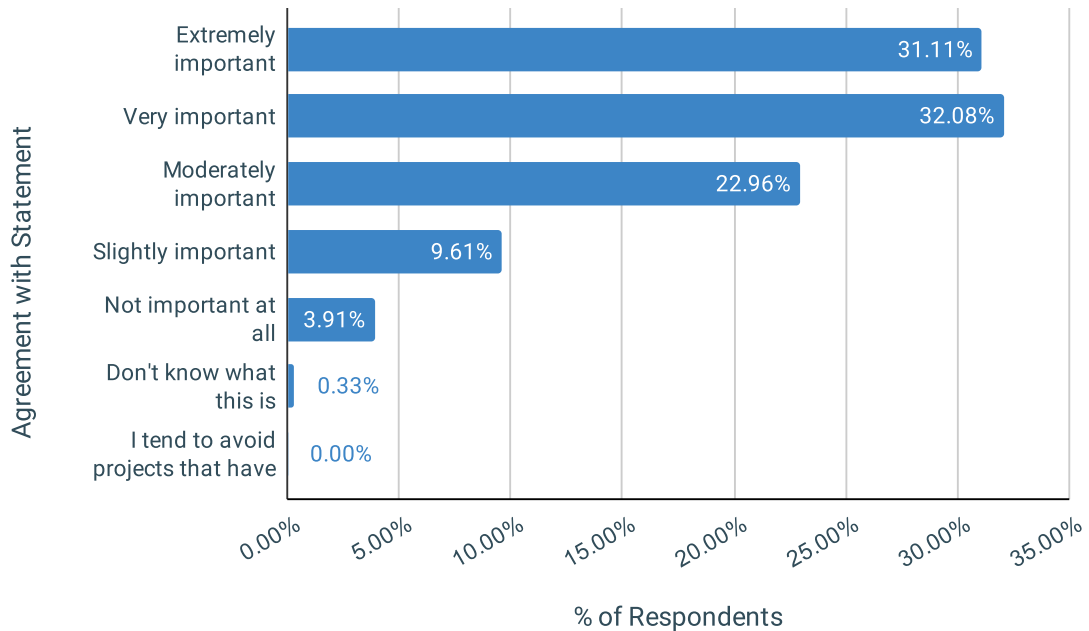
**i. Responsive maintainers**

N = 612 Saw but didn't respond: 27



**j. A welcoming community**

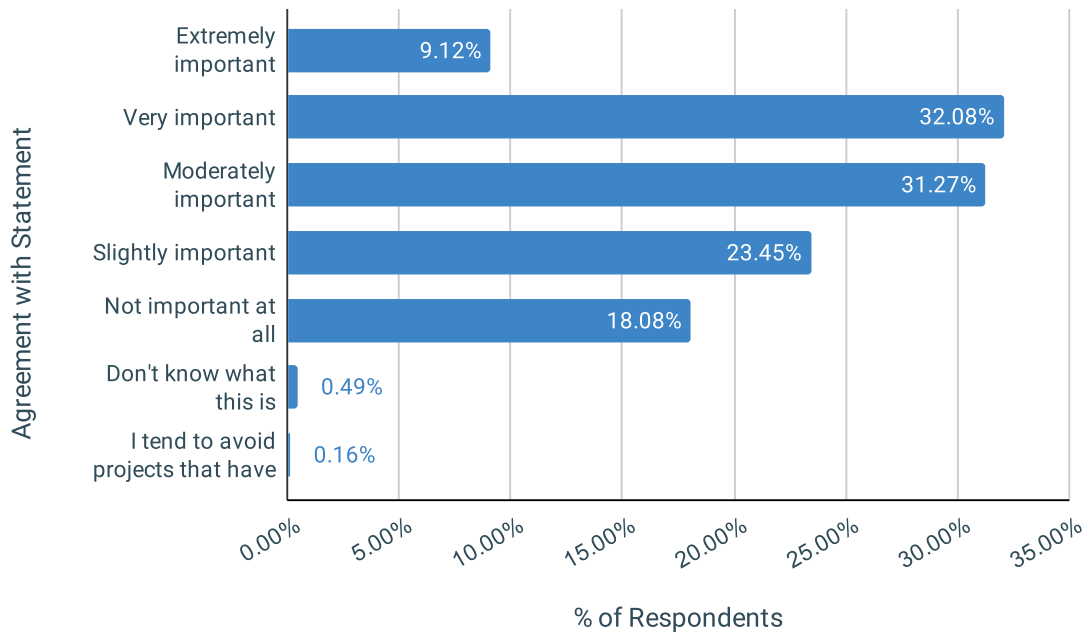
N = 614 Saw but didn't respond: 25



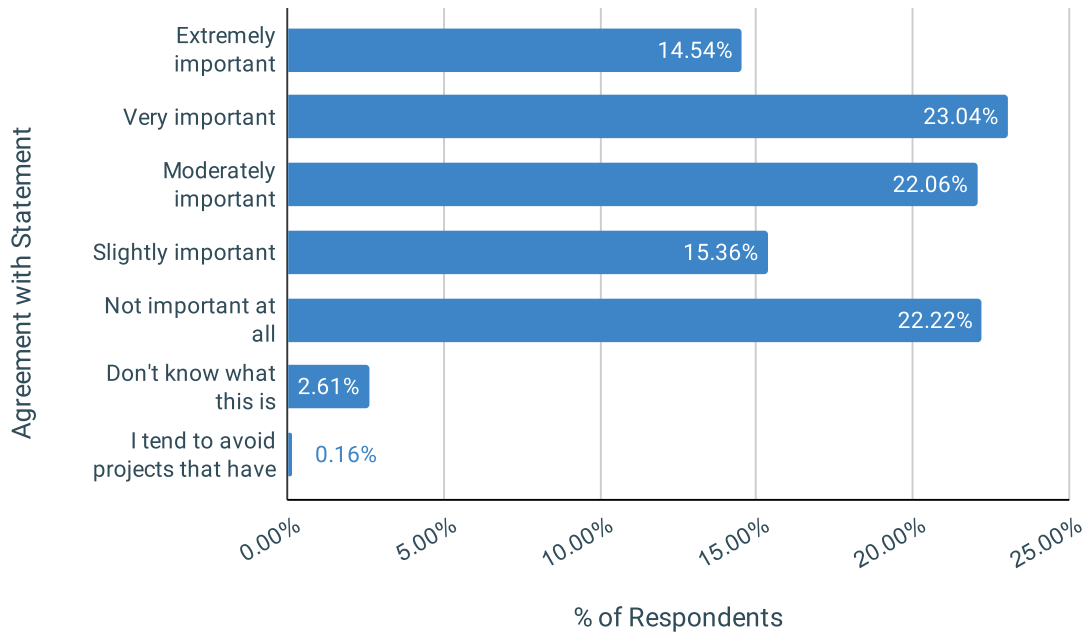


### k. Widespread use

N = 614 Saw but didn't respond: 25

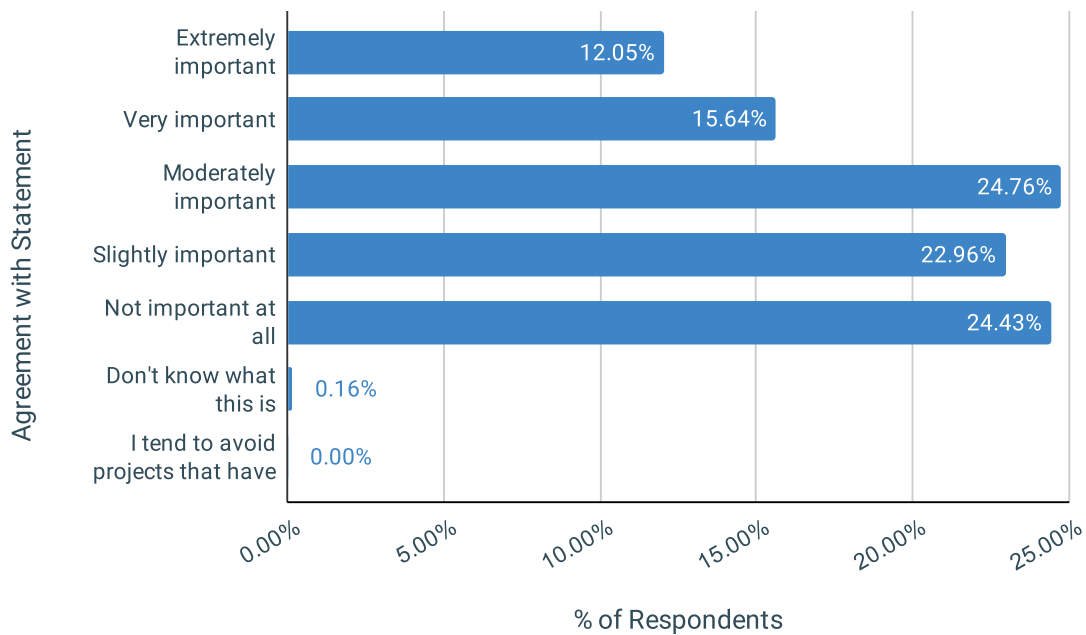


### l. Employer uses/encourages contribution to this FOSS project



**m. Neutrally governed, e.g., by a foundation or not controlled by a single company**

N = 614 Saw but didn't respond: 25



**n. Other? (Please specify below)**

N = 119 Saw but didn't respond: 520

**Text Response Summary**

Respondents indicated that they also consider whether contributions are likely to be accepted, whether the project is something they use professionally or personally, and whether it is easy to build the project and run existing tests.

**Please select the response that most closely reflects your experience for each of the following statements:**

**34. I feel as though I am a member of the FOSS community.**

N = 615 Saw but didn't respond: 12

I feel as though I am a member of the FOSS community	% of Respondents
Strongly agree	50.24%
Somewhat agree	31.87%
Neither agree nor disagree	13.50%
Somewhat disagree	2.60%
Strongly disagree	1.79%

### 35. When I contribute to a FOSS project, I generally...

N = 602 Saw but didn't respond: 25

When I contribute to a FOSS project, I generally...	% of Respondents ▼
Use my real name	62.62%
Use a consistent screen name or email address easily linked to my real name online	27.57%
Use a consistent screen name or email address that is not linked to my real name anywhere online	6.15%
Use several different screen names on different platforms	2.49%
Don't provide any identifying information	1.16%

### 36. Please rank your primary motivations for contributing to the FOSS project you spend the most time on by clicking and dragging the items below. (#1 indicates the most important, #10 is the least important)

N = 534 Saw but didn't respond: 87

Motivation	# of Times a Motivation Was Ranked in Top 3 ▼
I use this piece of FOSS and needed the specific features/fixes I added	278
I enjoy learning	276
Contributing allows me to fulfill a need for creative, challenging, and/or enjoyable work	222
Since I use FOSS, I feel I should contribute back to it	205
I believe in the mission of FOSS or the particular area I contribute to (e.g. privacy software)	184
I enjoy helping others	154
I am paid to develop FOSS	99
I enjoy working with my peers and my community	89
I value the recognition of my peers	83
I expect my contributions will help me advance my career	83

Most frequently found in respondents' bottom three	# of Times a Motivation Was Ranked in Bottom 3 ▼
I am paid to develop FOSS	326
I expect my contributions will help me advance my career	252
I value the recognition of my peers	216
I enjoy working with my peers and my community	133
I believe in the mission of FOSS or the particular area I contribute to (e.g. privacy software)	128
I use this piece of FOSS and needed the specific features/fixes I added	123
I enjoy helping others	117
Since I use FOSS, I feel I should contribute back to it	110
Contributing allows me to fulfill a need for creative, challenging, and/or enjoyable work	96
I enjoy learning	62

### 37. How likely are you to contribute to FOSS projects in the future?

N = 617 Saw but didn't respond: 9

How likely are you to contribute to FOSS projects in the future?	% of Respondents
Extremely likely	80.55%
Very likely	16.69%
Neither likely nor unlikely	1.94%
Somewhat unlikely	0.32%
Extremely unlikely	0.49%

### 38. Why do you feel that way?

N = 474 Saw but didn't respond: 150

#### Text Response Summary

Respondents who indicated that they would contribute in the future cited their desire to give back to the community and to help to improve software that is used around the world. One frequently cited motivation is that there will always be more bugs to fix. One respondent noted the benefits of the collaboration: "... I think this improves my skills both as a technician as well as a human being by working together with previously unknown people." Another noted the satisfaction contributing brought:

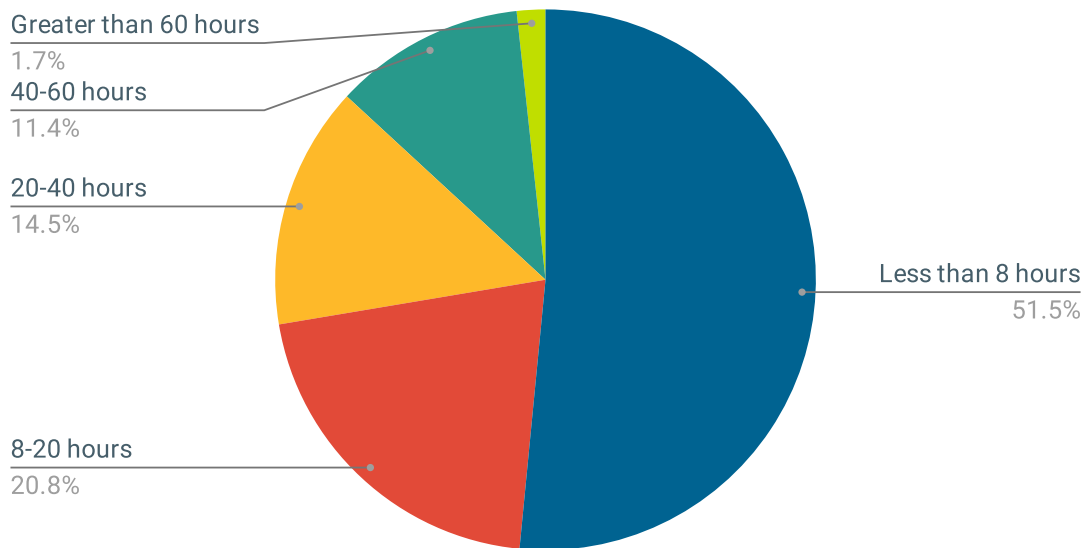
"Working on FOSS is one of the ways I feel I make a contribution to the world. It is much more satisfying than working on proprietary software." Those who said they would not be contributing to FOSS in the future predominantly cited their lack of time, e.g. "I am overburdened and need to focus on other things."

## Time Allocation

### 39. How many hours per week — on average — do you spend on all your FOSS projects combined? Please add all time spent on any kind of contribution including active development, documentation, performing organizational or administrative functions, etc.

*This question was presented to respondents as a text box where they could enter a numeric value, but for simplicity in this report, the results are separated into the five categories below: < 8 hours, 8-20 hours, 20-40 hours, 40-60 hours, and >60 hours.*

N = 586 Saw but didn't respond: 33



### 40. Is this more or less time than you were spending on FOSS projects on average 5 years ago?

*This question was only shown to respondents who reported that they had started contributing to FOSS at least 5 years ago in question #7*

N = 574 Saw but didn't respond: 45

**44.08%** More  
**33.45%** Less  
**22.47%** The same

### 41. Is this more or less time than you were spending on FOSS projects on average 10 years ago?

*This question was only shown to respondents who reported that they had started contributing to FOSS at least 10 years ago in question #7*

N = 302 Saw but didn't respond: 17

**48.01%** More  
**36.42%** Less  
**15.56%** The same

**42. Have the circumstances created by the coronavirus changed the number of hours per week you spend on all your FOSS projects combined?**

N = 554    Saw but didn't respond: 65

- 65.88% No, there has been no change
- 24.37% Yes, I spend more hours per week on FOSS now
- 9.75% Yes, I spend fewer hours per week on FOSS now

**43. What percentage of those [reported number from #38] hours spent on FOSS projects do you devote to the following tasks? Please ensure that the sum of the values you entered in each column equal 100 percent.**

Task	Average % of Total Time Spent	Average % of Time Desired to Spend
Contributing new code	24.20%	33.52%
	N = 480	N = 428
	Saw, didn't respond = 120	Saw, didn't respond = 172
Contributing documentation	9.39%	10.22%
	N = 481	N = 428
	Saw, didn't respond = 119	Saw, didn't respond = 172
Maintaining projects	13.20%	9.60%
	N = 480	N = 427
	Saw, didn't respond = 120	Saw, didn't respond = 173
Improving existing functionalities	19.64%	22.24%
	N = 480	N = 428
	Saw, didn't respond = 120	Saw, didn't respond = 172
Reporting or documenting bugs & unexpected behaviors	12.07%	7.67%
	N = 479	N = 427
	Saw, didn't respond = 121	Saw, didn't respond = 173
Offering ideas for new features	7.86%	11.27%
	N = 479	N = 428
	Saw, didn't respond = 121	Saw, didn't respond = 172
Performing organizational or administrative functions	8.84%	4.87%
	N = 481	N = 428
	Saw, didn't respond = 119	Saw, didn't respond = 172
Responding to security issues	2.83%	3.26%
	N = 481	N = 428
	Saw, didn't respond = 119	Saw, didn't respond = 172
Other	3.62%	2.77%
	N = 481	N = 4
	Saw, didn't respond = 119	Saw, didn't respond = 172

### Text Response Summary

Other tasks that respondents reported related to project communication (such as answering user questions, contributing to discussions, marketing, project websites, and “emotional support”), reviewing code, or community management (organizing votes, devising project progression, discussing proposed behavior changes, working on ethical and social issues, as well as writing proposals for changes). Contributors also mentioned mentoring and teaching others, reading mailing lists and keeping up-to-date with project developments, as well as working on improvements to infrastructure, architectural design, bug fixes, testing and continuous integration and continuous delivery (CI/CD).

## Employment

### 44. What is your current employment status?

N = 593 Saw but didn't respond: 7

What is your current employment status?	% of Respondents
Employed full-time	74.87%
Self-employed or freelancer	9.11%
Full-time student	5.73%
Employed part-time	3.71%
Unemployed	3.20%
Temporarily out of the workforce (e.g. medical or parental leave)	1.01%
Permanently out of the workforce (e.g. retired, due to disability)	1.01%
Other	0.67%
Temporary worker	0.51%
Full-time caretaker	0.17%

### 45. Does your current employer encourage you to contribute to FOSS as part of your work?

*This question was only shown to those who reported their employment status in question #42 as full-time, part-time, temporary worker, temporarily out of the workforce, or other*

N = 586 Saw but didn't respond: 9

**57.51%** Yes

**42.49%** No

**46. How does your current employer encourage you to contribute to FOSS? Please select all that apply.**

*This question was only shown to those who reported their employer encourages them to contribute to FOSS as a part of their work in question #43*

N = 327 Saw but didn't respond: 10

How does your current employer encourage you to contribute to FOSS?	% of Respondents
Allows some of paid work hours to contribute to FOSS project(s)	67.58%
Requires some paid work hours to contribute to FOSS project(s)	34.68%
Funds training in FOSS components (online or in-person classes, etc.)	24.77%
Funds participation in FOSS events (conferences, meet-ups, etc.)	47.70%
Other	22.94%

**Text Response Summary**

Some respondents stated that their employers encourage them to contribute to FOSS through strictly verbal encouragement. Some respondents said the employer allows contribution if it is useful for the project and many said their employer encourages releasing internal projects as FOSS.

**47. Did your current employer's support for FOSS influence your decision to join that organization?**

*This question was only shown to those who reported their employer encourages them to contribute to FOSS as a part of their work in question #43*

N = 331 Saw but didn't respond: 6

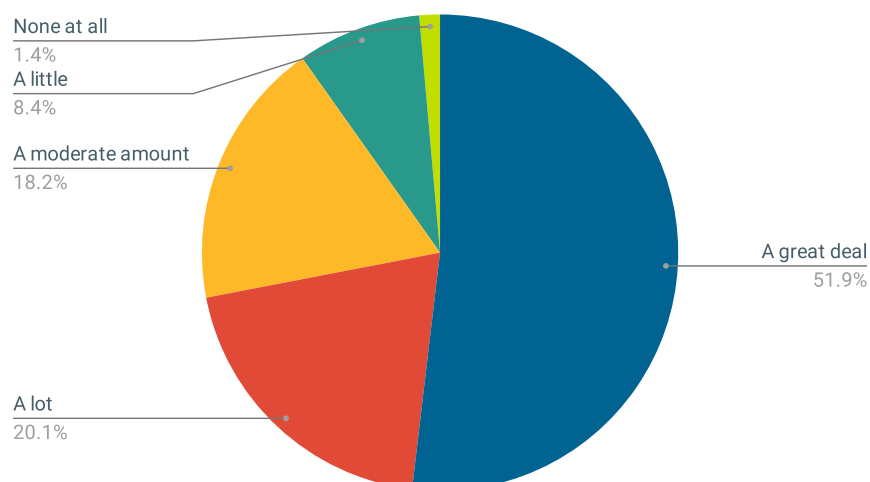
**65.26% Yes**

**34.74% No**

**48. How much did their stance on FOSS influence your decision**

*This question was only shown to those who reported yes, their employer's stance on FOSS influenced their decision to join that organization in question#47*

N = 214 Saw but didn't respond: 2





**49. What percentage of those [reported number from #38] hours spent on FOSS projects occurred during paid working time versus during your free time? Please ensure that the sum of the values you entered equal 100 percent.**

*This question was only shown to those who reported their employment status in question #42 as full-time, employed part-time, temporary worker, temporarily out of the workforce, or other*

N = 531 Saw but didn't respond: 0

**39.14%** Average paid time

**51.56%** Average free time

**50. Is this more or less time than you were spending on FOSS projects during work hours 5 years ago?**

*This question was only shown to those who reported their employment status in question #42 as full-time, part-time, temporary worker, temporarily out of the workforce, or other AND reported starting their FOSS contributions at least 5 years ago in question #7.*

N = 414 Saw but didn't respond: 10

**36.96%** More

**33.82%** The same amount

**22.22%** Less

**7.0%** Not applicable

**51. Is this more or less time than you were spending on FOSS projects during work hours 10 years ago?**

*This question was only shown to those who reported their employment status in question #42 as full-time, part-time, temporary worker, temporarily out of the workforce, or other AND reported starting their FOSS contributions at least 10 years ago in question #7.*

N = 277 Saw but didn't respond: 5

**46.57%** More

**24.91%** The same amount

**21.66%** Less

**6.86%** Not applicable

**52. Does your supervisor consider this allocation of work time to FOSS projects as part of your core job tasks?**

*This question was only shown to those who reported any percentage of time spent on FOSS occurring during work hours.*

N = 334 Saw but didn't respond: 3

**71.26%** Yes

**28.74%** No

### 53. What is your current occupation?

This question was only shown to those who reported their employment status in question #42 as full-time, part-time, self-employed or freelancer, temporary worker, temporarily out of the workforce, or other. Respondents were given a drill-down menu of occupations from the [Bureau of Labor Statistics Standard Occupational Classification system](#)

N = 493 Saw but didn't respond: 36

Occupation	% of Respondents
Software Developer, Applications	28.40%
Software Developer, Systems Software	23.94%
Computer Programmer	12.78%
Web Developer	7.10%
Computer Occupation, All Other	4.46%
Computer or Information Research Scientist	4.26%
Network or Computer Systems Administrator	1.62%
Computer or Information Systems Manager	1.62%
General and Operations Manager	1.01%
Engineer, All Other	1.01%
Architectural or Engineering Manager	1.01%
Chief Executive	1.01%
Electrical Engineer	1.01%
Other occupations were chosen by less than 1% of respondents	

### 54. Under which sector would your current employer fall?

This question was only shown to those who reported their employment status in question #42 as full-time, part-time, temporary worker, temporarily out of the workforce or other.

N = 463 Saw but didn't respond: 10

Employer's Sector	% of Respondents
Software Development	36.72%
IT Services	17.49%
Finance and Insurance	6.91%
Technology Hardware	6.91%
Telecommunications	4.32%
Educational Services	4.10%
Other	3.24%
Professional, Scientific, and Technical Services	3.02%
Health Care and Social Assistance	2.59%
Manufacturing	2.38%
Non-profit	2.16%
Retail/Consumer Goods	2.16%
Transportation and Warehousing	1.30%
Media	1.30%
Arts, Entertainment, and Recreation	1.08%
Administrative and Support Services	1.08%
Agriculture, Forestry, Fishing and Hunting	0.86%
Public Administration	0.86%
Energy and Utilities	0.65%
Accommodation and Food Services	0.65%
Construction	0.00%
Mining	0.00%
Real Estate Rental and Leasing	0.00%

### 55. What type of department are you in?

This question was only shown to those who reported their employment status in question #42 as full-time, part-time, temporary worker, temporarily out of the workforce or other.

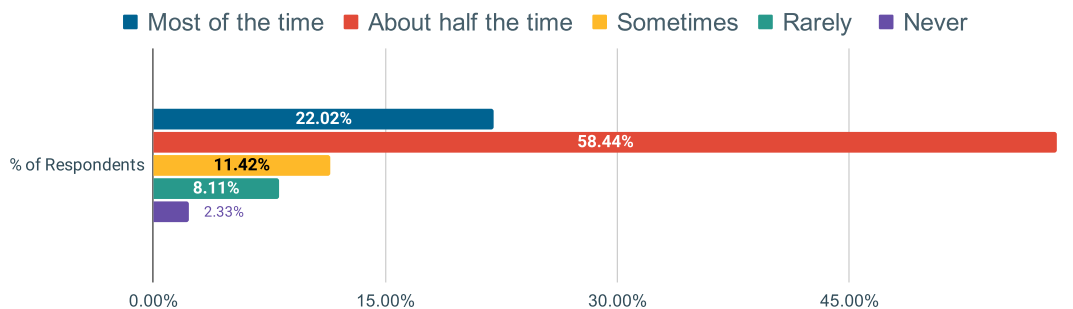
N = 463 Saw but didn't respond: 10

Respondent's Department	% of Respondents
Software/Application Development	37.37%
Engineering	15.77%
Research and Development	10.58%
DevOps   Infrastructure	6.48%
IT Leadership (CTO, CIO, IT Director)	4.32%
Web Development	4.32%
Architecture	3.67%
Information Security	3.46%
Other	2.60%
Cloud Computing Engineering	2.38%
Open Source Program Office	1.73%
Production	1.30%
Sales	1.08%
Computer Support	1.08%
Computer Network	0.86%
Marketing	0.86%
Operations	0.65%
Safety	0.43%
Database Administration	0.43%
Administration	0.43%
Legal	0.22%

### 56. How often do you write software or otherwise contribute to software productions in your current job? Note: This could be either closed (proprietary) or open (FOSS) software.

This question was only shown to those who reported their employment status in question #42 as full-time, part-time, self-employed or freelancer, temporary worker, temporarily out of the workforce or other.

N = 516 Saw but didn't respond: 11



**57. What is (or was) your most recent annual salary? (Please specify the currency. If your currency is not listed, please convert it to USD [here](#)) While you can choose not to answer, please note that the answer to this question will be kept strictly confidential and will be used only in an aggregated form without any personal identifiers.**

Respondents were presented with three dropdown menus: "Currency", "Salary Range", and "Gross (pre-tax) or Net (post-tax)." The ten offered currency options came represented the the [countries with the most FOSS contributions](#): US Dollars, Euros, Australian Dollars, Canadian Dollars, Pound Sterling, Japanese Yen, Indian Rupee, Hungarian Forint, Russian Rubles, Brazilian Reals, and Chinese Yuan Renminbis

N = 475 Saw but didn't respond: 118

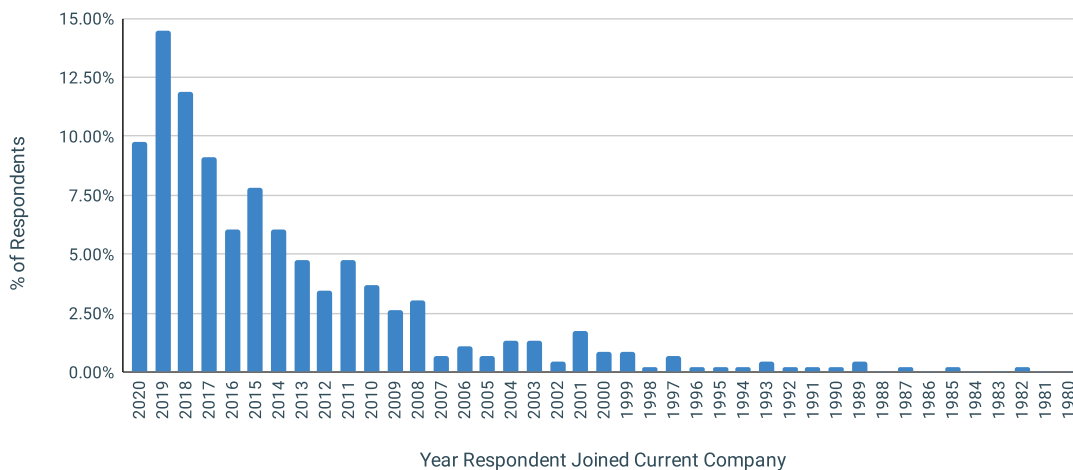
Mode of the salary range for each currency option, in order of most often selected currency to least often selected currency:

Currency	Symbol	Modal Salary Range
US Dollar	US\$	120,001 - 150,000
Euro	€	36,001 - 54,000
Pound Sterling	£	31,201 - 46,900
Canadian Dollar	C\$	80,101 - 106,800
Chinese Yuan Renminbi	¥	280,001 - 420,000
Indian Rupee	₹	0 - 1,450,000
Brazilian real	R\$	0 - 89,500 and 89,501 - 178,950
Japanese Yen	¥	6,480,001 - 8,640,000
Australian Dollar	A\$	91,801 - 122,400
Russian ruble	₽	1,333,001 - 2,666,000
Hungarian forint	Ft	0 - 6,080,000

**58. What year did you join your current company?**

This question was only shown to those who reported their employment status in question #42 as full-time, part-time, self-employed or freelancer, temporary worker, temporarily out of the workforce or other.

N = 462 Saw but didn't respond: 11



**59. What was your annual salary when you joined your current company?**

**(Please specify the currency. If your currency is not listed, please convert it to USD here)**

**While you can choose not to answer, please note that the answer to this question will be kept strictly confidential and will be used only in an aggregated form without any personal identifiers.**

*This question was only shown to those who reported their employment status in question #42 as full-time, part-time, self-employed or freelancer, temporary worker, temporarily out of the workforce or other. Similar to question # 55, respondents were presented with three dropdown menus: "Currency", "Salary Range", and "Gross (pre-tax) or Net (post-tax)." The eleven offered currency options came represented the countries with the most FOSS contributions: US Dollars, Euros, Australian Dollars, Canadian Dollars, Pound Sterling, Japanese Yen, Indian Rupee, Hungarian Forint, Russian Rubles, Brazilian Reals, and Chinese Yuan Renminbis  
N = 377 Saw but didn't respond: 96*

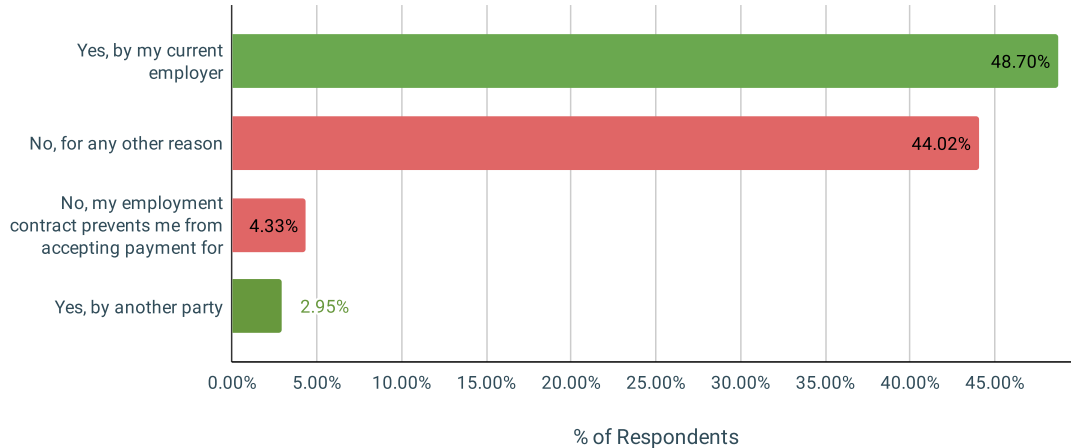
Mode of the salary range for each currency option, in order of most often selected currency to least often selected currency:

Currency	Symbol	Modal Salary Range
US Dollar	US\$	120,001 - 150,000
Euro	€	36,001 - 54,000
Pound Sterling	£	31,201 - 46,900
Canadian Dollar	C\$	53,401-80,100
Chinese Yuan Renminbi	¥	0-140,000 and 140,001-280,000
Indian Rupee	₹	0 - 1,450,000
Brazilian real	R\$	0 - 89,500
Japanese Yen	¥	6,480,001 - 8,640,000
Australian Dollar	A\$	91,801 - 122,400
Russian ruble	₽	0 - 1,333,000
Hungarian forint	Ft	0 - 6,080,000

# FOSS at Work

## 60. Are you paid for any of your time spent on open source contributions?

N = 577 Saw but didn't respond: 16



## 61. Please select the FOSS project(s) to which you contribute as part of your paid employment.

Respondents were asked to select which of the projects they had listed in question #10 they contributed to as part of their paid employment

N = 262 respondents answered the question about 992 projects

Saw but didn't respond: 18

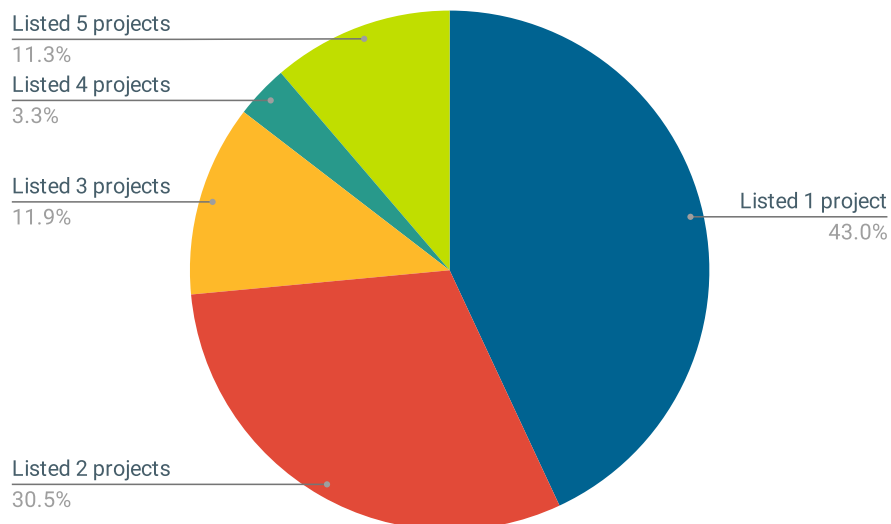
**65.52%** of projects are contributed to as part of a respondent's paid employment

**34.48%** of projects are not contributed to as part of a respondent's paid employment

## 62. 5 years ago, what FOSS project(s) did you contribute to as part of any paid employment.

This question was only shown to those who reported starting their FOSS contributions at least 5 years ago in question #7. They were asked to type in FOSS projects they had contributed to as part of their paid employment.

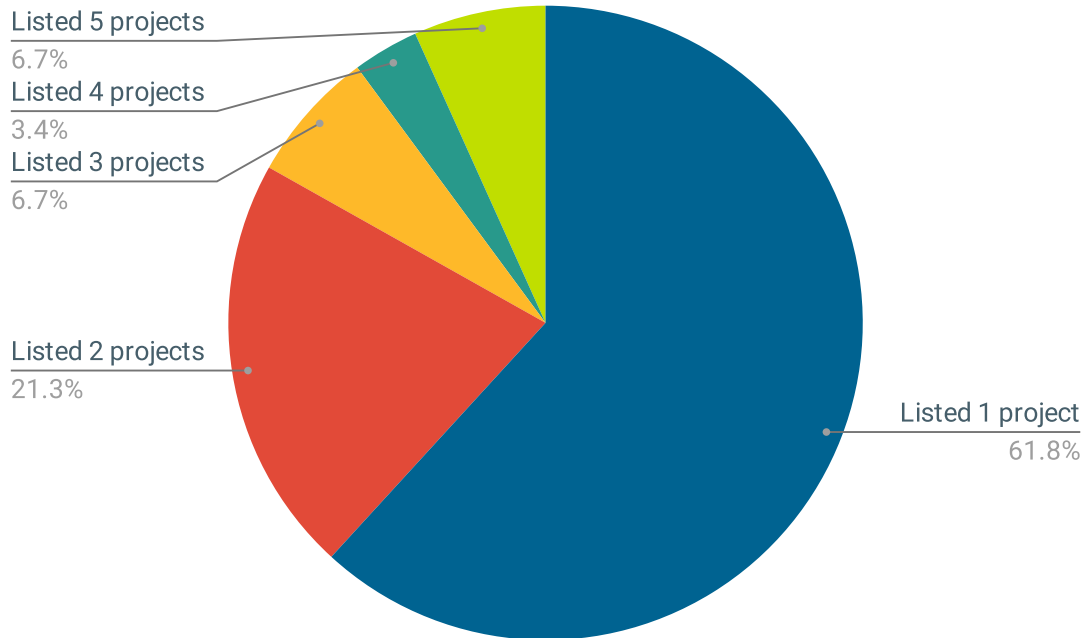
N = 151 Saw but didn't respond: 304



### 63. 10 years ago, what FOSS project(s) did you contribute to as part of any paid employment.

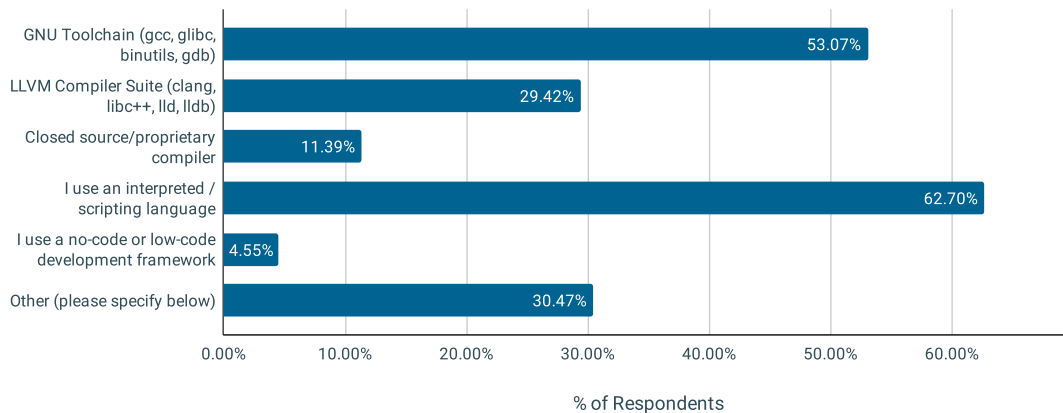
This question was only shown to those who reported starting their FOSS contributions at least 10 years ago in question #7 and they were asked to type in FOSS projects they had contributed to as part of their paid employment.

N = 89 Saw but didn't respond: 213



### 64. Which compiler(s) do you typically use? (Please select all that apply.)

N = 571 Saw but didn't respond: 18



### Text Response Summary

Many respondents listed Roslyn, Kotlin, JVM, Go, TypeScript, OpenJDK, AdoptOpenJDK, Scala, Javac, and MSBuild.

**65. In your current job, are the internal software projects (e.g., applications used within your employer's organization) that you work on closed (proprietary), open source (FOSS) or a mix of both?**

*This question was only shown to those who reported their employment status in question #42 as full-time, part-time, temporary, temporarily out of the workforce or other AND that they write software or otherwise contribute to software production in their current job in question #58.*

N = 417 Saw but didn't respond: 42

**59.95%** work on a mix of both closed and open software projects internally

**27.82%** work on closed software projects internally

**12.23%** work on open software projects internally

**66. In your current job, are the software products (e.g., solutions that your employer sells to customers) that you work on closed source (proprietary), open source (FOSS) or a mix of both?**

*This question was only shown to those who reported their employment status in question #42 as full-time, part-time, temporary, temporarily out of the workforce or other AND that they write software or otherwise contribute to software production in their current job in question #58.*

N = 445 Saw but didn't respond: 14

**38.2%** work on closed software products

**37.98%** work on a mix of both closed and open software products

**12.36%** work on open software products

**11.46%** reported that they don't develop software for sale (N/A)

**67. Do you build on or incorporate FOSS components into closed (proprietary) software?**

*This question was only shown to those who reported their employment status in question #42 as full-time, part-time, temporary, temporarily out of the workforce or other AND that they write software or otherwise contribute to software production in their current job in question #58 AND if they indicated that the internal software projects (question #64) and/or the software products (question #65) they work on contained closed software or a mix of closed and open software.*

N = 390 Saw but didn't respond: 2

78.72% Yes

21.28% No

**68. Which FOSS components do you usually build on or incorporate into non-FOSS projects? (Please use a comma to separate multiple projects.)**

*This question was only shown to respondents who reported that they build on or incorporate FOSS components into closed software in question #66*

N = 229 Saw but didn't respond: 75

**Text Response Summary**

Respondents answered that they use components such as various Apache components, Antlr4, Spring Boot, Spring Cloud, .NET Core, U-Boot, Protobuf, ReactJS, Ruby on Rails, PostgreSQL, Django, JSON.NET, Pandas, AngularJS, Bootstrap, Tensorflow, and Node.js. A common answer was expressed by one respondent as: "Tons of things! What tech stack isn't like 90% FOSS at this point?"



### 69. Which of the following best categorizes your work on open source (FOSS) projects?

*This question allowed for multiple selections and was only shown to those who indicated that the internal software projects (question #64) and/or the software products (question #65) they work on contained open software or a mix of closed and open software.*

N = 306 Saw but didn't respond: 14

Which of the following best categorizes your work on open source (FOSS) projects?	% of Respondents
(Multiple selections per respondent possible)	
Operating Systems	28.76%
DevOps Tools	27.78%
Application Security & Utilities	26.14%
Web Application Framework	24.84%
JavaScript Libraries	21.57%
Build Management tools	17.97%
Database	15.03%
HTTP Modules	14.05%
Analytics	11.77%
Big Data	11.11%
Batch Processing	10.13%
Source Code Management (SCM)	9.48%
Integrated Development Environment (IDE)	8.50%
Cascading Style Sheets (CSS)	6.21%
Content Management Systems	5.88%
Mobile API	5.23%
Other (please specify below)	31.37%

#### Text Response Summary

Respondents listed categories such as networking libraries, compilers, simulation modeling, Java libraries, Python libraries, scientific computing, machine learning, blockchain, documentation, spatial analysis, middleware, developer tools, graphics, and accessibility.

## 70. Which of the following best categorizes your work on closed source (proprietary) projects?

*This question allowed for multiple selections and was only shown to those who indicated that the internal software projects (question #64) and/or the software products (question #65) they work on contained closed software or a mix of closed and open software.*

N = 348 Saw but didn't respond: 40

Which of the following best categorizes your work on closed source (proprietary) projects?	% of Respondents
(Multiple selections per respondent possible)	
Cascading Style Sheets (CSS)	66.10%
DevOps tools	25.86%
Web Application Framework	23.85%
Application Security & Utilities	22.70%
Analytics	21.26%
Database	18.39%
Batch Processing	14.94%
Big Data	14.94%
JavaScript Libraries	14.10%
Build Management tools	12.36%
Operating Systems	10.63%
Content Management Systems	10.35%
HTTP Modules	10.35%
Mobile API	8.62%
Integrated Development Environment (IDE)	5.75%
Source Code Management (SCM)	5.46%
Other (please specify below)	30.75%

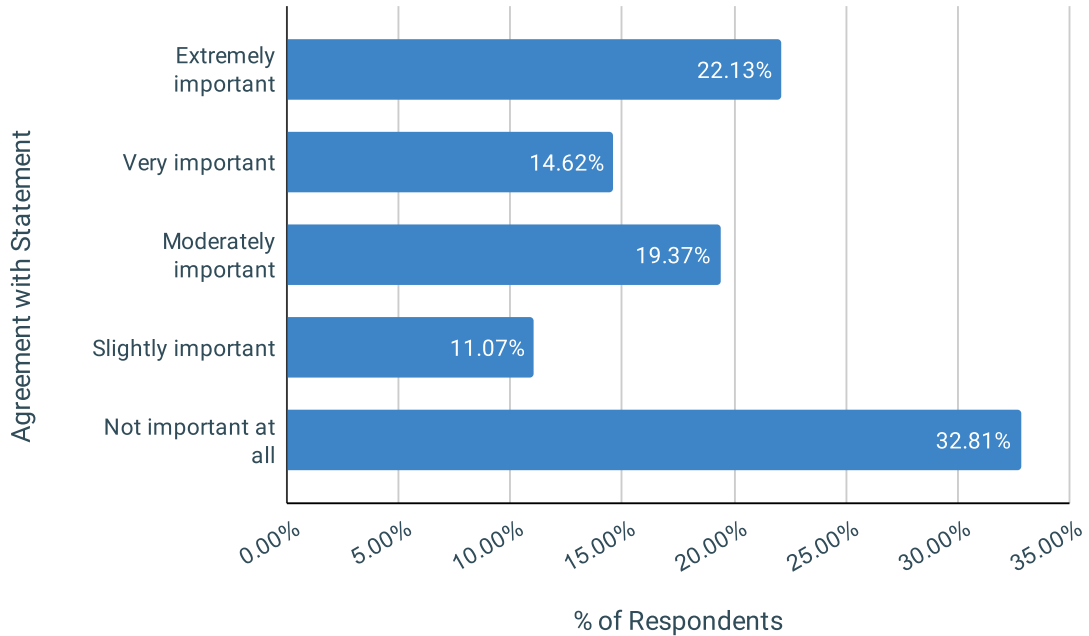
### Text Response Summary

Respondents listed categories such as compilers, machine learning, developer tools, portals, inventories, embedded devices and software, web services, desktop applications, trading systems, cryptography, scientific computing, documentation, graphics, and web applications.

**71. How important do you think your involvement in FOSS projects was to getting your current job?**

*This question was only shown to those who reported their employment status in question #42 as full-time, part-time, temporary, temporarily out of the workforce or other.*

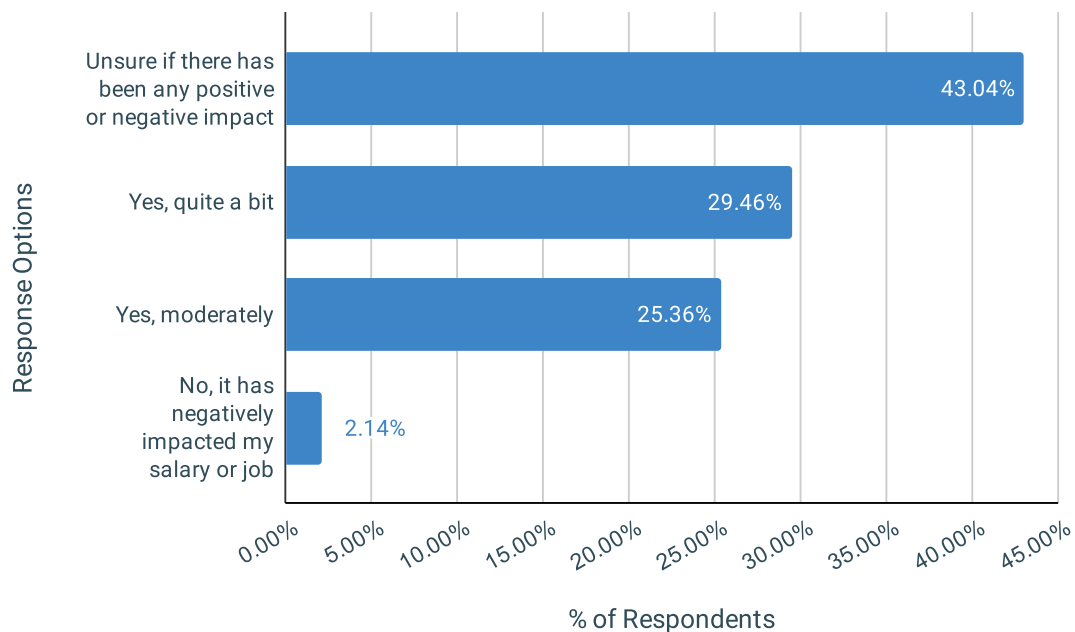
N = 506 Saw but didn't respond: 11



**72. Do you feel your participation in FOSS projects has positively impacted your salary or job prospects?**

*This question was only shown to those who reported their employment status in question #42 as anything other than full-time caretaker or permanently out of the workforce.*

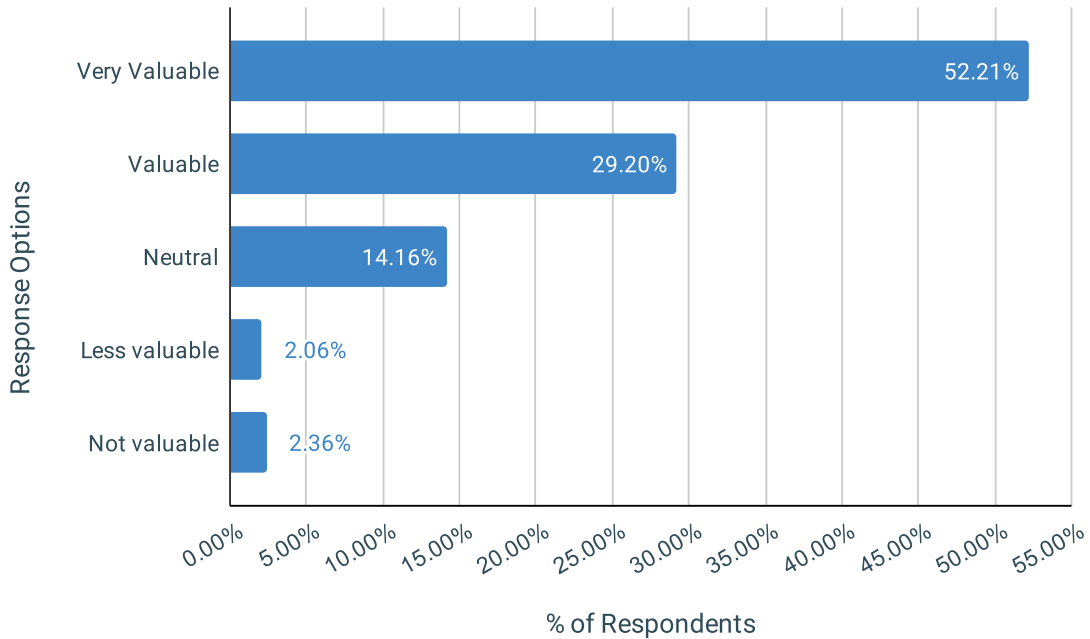
N = 560 Saw but didn't respond: 10



### 73. How valuable were the skills that you acquired from working on FOSS for your current employer?

*This question was only shown to respondents who indicated that FOSS contributions is considered part of their core job in question #51 or if they answered that they received any kind of payment for FOSS contributions in question #59*

N = 339 Saw but didn't respond: 13



### 74. Could you explain or give an example of how these skills were valuable?

*This question was only shown to respondents who answered anything except "Not valuable" in question #70*

N = 200 Saw but didn't respond: 131

#### Text Response Summary

Respondents said that they learned how to write and test production-level code, organize asynchronous work as a team, and communicate and collaborate with a team. Respondents also said that they learned how to perform code reviews, how to use version control, and that they benefit from clean code being prioritized over "ugly workarounds." Respondents benefited from learning about "workflows, issue tracking and resolution, bug reporting, timezone distributed work, gaining trust in developer communities." They also said that learning about FOSS libraries helped them know what solutions exist when they are writing code.

# Employer Policies

All of the questions in the “Employer Policies” section were only shown to those who reported their employment status in question #42 as full-time, part-time, temporary, temporarily out of the workforce or other.

## 75. How does your employer’s intellectual property agreement/policy affect your contributions to FOSS projects, unrelated to your work, during your free time?

N = 429 Saw but didn’t respond: 34

## 76. Thinking back to five years ago, how did the intellectual property agreement/policy of your employer at that time affect your contributions to open source unrelated to your work, during your free time?

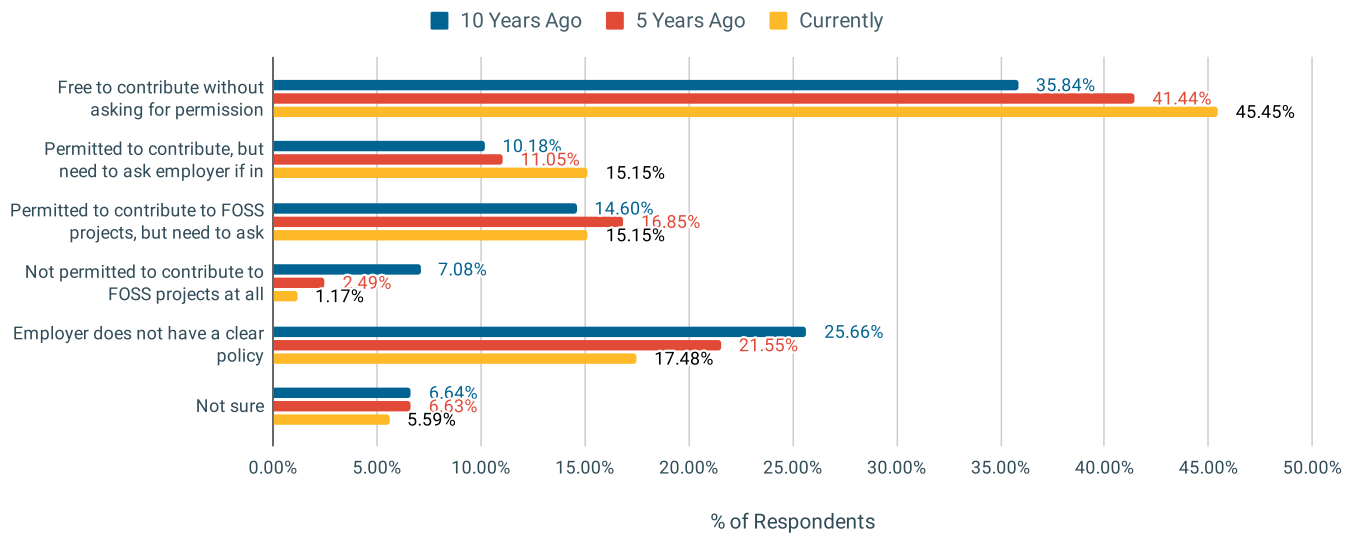
This question was only shown to those who reported starting their FOSS contributions at least 5 years ago in question #7.

N = 362 Saw but didn’t respond: 89

## 77. Thinking back to ten years ago, how did the intellectual property agreement/policy of your employer at that time affect your contributions to open source unrelated to your work, during your free time?

This question was only shown to those who reported starting their FOSS contributions at least 10 years ago in question #7.

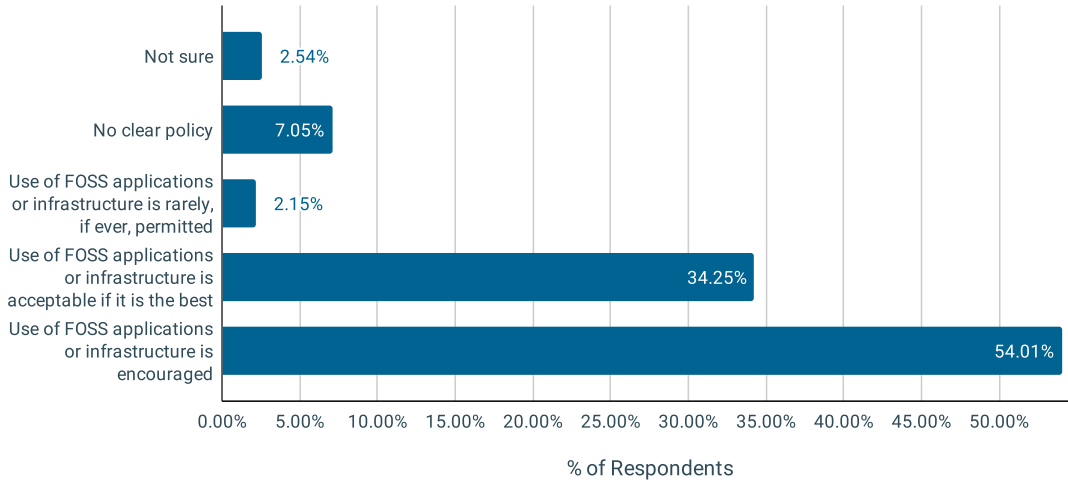
N = 226 Saw but didn’t respond: 74



**78. Please select the statement that is closest to your current employer's policy on using FOSS applications or infrastructure.**

*This question was only shown to those who reported that they write software or otherwise contribute to software production in their current job in question #58.*

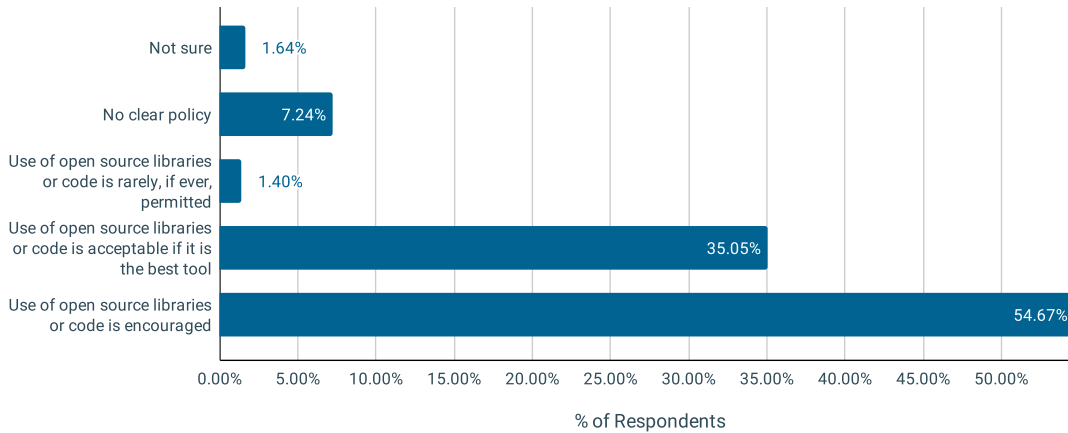
N = 511 Saw but didn't respond: 54



**79. Please select the statement that is closest to your current employer's policy on incorporating open source libraries or code into your code base.**

*This question was only shown to those who reported that they write software or otherwise contribute to software production in their current job in question #58.*

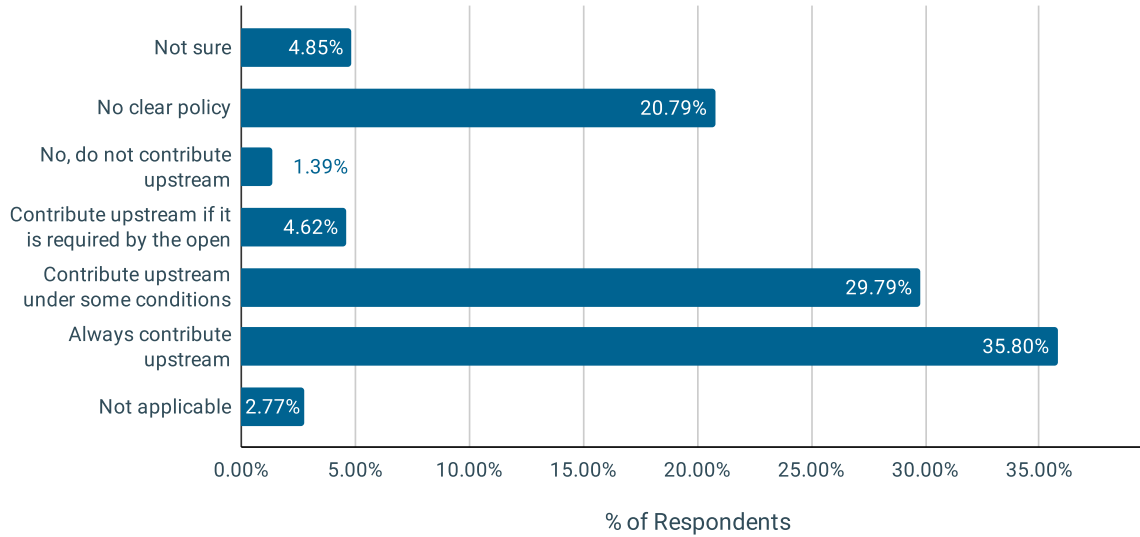
N = 428 Saw but didn't respond: 21



**80. Please select the statement that is closest to your current employer's policy on contributing upstream (e.g., sending any changes you make to an open source project back to the original maintainers for inclusion into an upcoming release of the software).**

*This question was only shown to those who reported that they write software or otherwise contribute to software production in their current job in question #58.*

N = 433 Saw but didn't respond: 16



# Suggested Action

## 81. Think of the FOSS project to which you contribute that needs the most assistance. What type of contribution from external sources would be most beneficial? (Please select all that apply.)

N = 527 Saw but didn't respond: 46

**36.43%** included financial contributions in their response

**52.12%** included code contribution — new features in their response

**65.66%** included code contribution — bug/security fixes in their response

**33.59%** included a free security audit in their response

**17.46%** included a free online course on how to develop secure software in their response

**25.62%** included simplified ways to add security-related tools to the continuous/integration pipeline

**4.18%** included training in their response (see text response summary below):

### Text Response Summary

Most respondents requested online, live session, demos featuring training on best practices, quick start guidance (both for those new to a project and those new to FOSS in general), security, threat modeling, and secure software development. One respondent suggested, “free training on sites such as CodeAcademy, Udemy, and Pluralsight.”

**10.06%** selected to include other suggestions (see text response summary below):

### Text Response Summary

Additional suggestions:

1. Assistance with infrastructure and development workflow, including hosting services, built-in security features in package managers, automatic testing, tools to audit dependency vulnerabilities, access to build and test machines, as well as performance audits and optimizations
2. Sponsoring and simplifying security tools — like encryption, two-factor authentication, and digital signatures — to increase their use
3. Creating, standardizing, and implementing project documentation
4. General assistance with code contribution, reviewing/maintaining, and triaging issues (e.g., bugs, new features, questions)
5. Information on how to improve funding and professionalize the donation process (e.g., invoices or receipts)
6. Marketing assistance for FOSS projects, as well as raising public awareness about open source in general