STATE OF THE DEVELOPER NATION

23RD EDITION

The latest trends from our Q3 2022 survey of 26,000+ developers
About SlashData

SlashData is the leading analyst company in the developer economy, tracking global software developer trends based on more than 30,000 software developers annually in over 160 countries. Our surveys track the changing landscape of mobile, IoT, desktop, cloud, web, AR, VR, games, machine learning developers, and data scientists. Our mantra:

*We help the world understand developers - and developers understand the world.*

SlashData Ltd.
19-21 Hatton Gardens,
London EC1N 8BA
United Kingdom
+44 845 003 8742
www.slashdata.co
@SlashDataHQ

About Developer Nation

Developer Nation is a global community engaging thousands of developers of all shapes and sizes across the globe, enabling them to benchmark themselves against the developer nation. We are committed to facilitating community contribution and knowledge sharing, and promoting diversity and inclusion in the developer ecosystem.

*Our vision is to empower developers to shape the future!*

www.developernation.net
@devnationworld
Can I share data from this report?

1. License Grant
This report is licensed under the [Creative Commons Attribution-NoDerivatives Licence 4.0 (International)](https://creativecommons.org/licenses/by-nd/4.0/). Put simply, subject to the terms and conditions of this license, you are free to:

**Share** — You can reproduce the report or incorporate parts of the report into one or more documents or publications, for commercial and non-commercial purposes. Under the following conditions:

** Attribution** — You must give appropriate credit to SlashData™, and indicate if changes were made. In that case, you may do so in any reasonable manner, but not in any way that suggests that SlashData™ endorses you or your use.

**NoDerivatives** — you cannot remix or transform the content of the report. You may not distribute modified content.

2. Limitation of Liability
SlashData™, believes the statements contained in this publication to be based upon information that we consider reliable, but we do not represent that it is accurate or complete and it should not be relied upon as such. Opinions expressed are current opinions as of the date appearing in this publication only and the information, including the opinions contained herein, are subject to change without notice. Use of this publication by any third party for whatever purpose should not and does not absolve such third party from using due diligence in verifying the publication’s contents. SlashData™ disclaims all implied warranties, including, without limitation, warranties of merchantability or fitness for a particular purpose.

SlashData™, its affiliates, and representatives shall have no liability for any direct, incidental, special, or consequential damages or lost profits, if any, suffered by any third party as a result of decisions made, or not made, or actions taken, or not taken, based on this publication.

---

*The analyst of the developer economy | formerly known as VisionMobile*

SlashData © Copyright 2022 | Some rights reserved
ABOUT THE AUTHORS

**Konstantinos Korakitis**
*Director of Research*

Konstantinos heads the Research Product team at SlashData and is responsible for all syndicated research products and custom research projects. With more than 10 years of experience as an engineer, consultant, and manager, he oversees research planning, survey design, data analysis, insights generation, and research operations.

[Email](konstantinos@slashdata.co)

**Liam Dodd**
*Data Storyteller*

Liam is a researcher with an interest in data analytics and its ability to impact society. He holds a PhD in Physics, and previously worked generating intelligence and insights for the European automotive market.

[Email](liam.dodd@slashdata.co)
ABOUT THE AUTHORS

Richard Muir
Data Journalist
Richard is a Data Journalist with over five years’ experience working in a range of industries, from government policy to automotive. He is responsible for finding and interpreting actionable insights from SlashData’s developer surveys.

Sam Coates
Data Storyteller
Sam is a researcher with a passion for data analysis and exploring what drives change. He holds a PhD in Physics and is the author of over 20 scientific papers, which have been published in internationally recognised journals.

Nikita Solodkov
Data Storyteller
Nikita is a multidisciplinary researcher with a particular interest in using data-driven insights to solve real-world problems. He holds a PhD in Physics and has over five years of experience in data analytics and research design.
ABOUT THIS REPORT

SlashData’s Developer Nation survey is the leading research programme on mobile, desktop, industrial IoT, consumer electronics, embedded, third-party app ecosystems, cloud, web, game, AR/VR, and machine learning developers, as well as data scientists, tracking developers’ experiences across platforms, technologies, programming languages, app and API categories, revenue models, segments, and regions.

The 23rd edition of the Developer Nation global survey ran from June to August 2022 and reached more than 26,000 developers from 163 countries. This research report delves into key developer trends for Q3 2022 and beyond.

The report focusses on six major themes - each with its own visualisations - showing how the data lends insight into the developer community.

1. **The state of blockchain development**: The distributed transaction ledgers and smart contracts that comprise blockchain technologies have applications in a wide range of industries, particularly in finance, logistics, and government. In this chapter, we focus solely on blockchain applications other than cryptocurrencies and we explore how developers are involved in blockchain projects and which platforms they target.

2. **Students’ top career aspirations**: Having long-term career aspirations can provide students with a sense of direction and help them make appropriate choices in their pursuit of knowledge. In turn, this speeds up their professional development and increases the likelihood of them achieving a successful career. In this chapter, we look at the top career aspirations of developers who self-identify as students.

3. **Language communities - An update**: Programming languages are often the kernels of strong communities and the subject of opinionated debates. In this chapter, we provide updated estimates of the number of active software developers using each of the major programming languages, across the globe and across all kinds of programmers.

---

1 Unless stated otherwise throughout the report, by “developers” we mean everyone who self-identifies as being involved in the software development ecosystem, including individuals holding business (e.g. product managers, C-level executives) and creative (e.g. UI/UX designers) roles.
4. **Why developers contribute to vendor-owned open source projects:**

Corporate and vendor-owned open source projects combine the values of sharing code, knowledge, and best practices among peer developers with the support from the world's largest tech companies. In this chapter, we examine what motivates developers to contribute to such projects, across different regions and across different developer segments.

5. **Types of studios game developers work for:**

The nature of professional game developers' work can vary depending on the type of studio they work for. In this chapter, we explore the profile of developers working for different types of game studios, focussing on their level of experience, roles, and technology choices.

6. **The rise and fall of web frameworks:**

Web frameworks simplify and accelerate the web development process by enabling developers to abstract access to underlying system resources through a set of high-level APIs. In this chapter, we look at which web frameworks have risen or declined in popularity in the past two years, and we explore the technology choices of the developers who use them.

We hope you will enjoy this report and find the insights useful! If you have any questions or comments, or are looking for additional data, you can get in touch with Stathis Georgakopoulos, Product Marketing Manager for SlashData at stathis@slashdata.co. You can download and access this report and more free resources at slashdata.co/free-resources.

You can find more developer data at **SlashData Dashboards**. Through the interactive data dashboards, you can user filter combinations to narrow down results to answer the questions you have or to match your audience's profile. Try it out at sdata.me/DashBoard.

We’d like to thank everyone who helped us reach 26,000+ respondents and create this report. Our Media Partners - CodeGym, NeoHR, Microsoft, Salesforce, Samsung, Zup, and so many others.

Our Developer Committee supported our efforts once more, to create the most up-to-date and detailed survey. Special thanks to our members: Akhil, Christopher, Deborah, Dominic, Ioannis, Ivan, Javier, JS, Ken, Kujtim, Mikhail, Mustapha, Niccolò, Rajasekar, Ricardo, and Thien for your help with reviewing survey content, translations, and suggesting prizes.

Our linguistics partner, Palex Group, supported us to create an inclusive survey, translated into eight different languages - Simplified Chinese, Traditional Chinese, Korean, Portuguese, Russian, Spanish, and Vietnamese. In addition, our prize payments partner, Chimoney.
KEY INSIGHTS

The state of blockchain development
- 25% of developers are currently working on or learning about blockchain applications other than cryptocurrencies.
- Developers with 6-10 years of experience in software development are the most likely to be working on blockchain projects.
- Though Ethereum is the dominant blockchain platform, it is the only one more popular among learners than those currently working on blockchain applications.

Students’ top career aspirations
- Solving problems, gaining expertise, and building innovative solutions are the top career aspirations for student developers.
- Students who aspire to maximise their impact on society show a high level of interest in innovation and building their own businesses.
- Students from North America and the Middle East and Africa are the most likely to aspire to become entrepreneurs in the future.

Language communities - An update
- Javascript remains the largest programming language community, with an estimated 19.6M developers worldwide using it.
- In the last two years, Java has almost doubled the size of its community, from 8.3M to 16.5M. For perspective, the global developer population grew about half as fast over the same period.
- Kotlin and Rust are the two fastest-growing language communities, having more than doubled in size in the past two years.

Why developers contribute to vendor-owned open source projects
- 73% of developers report that they contribute to vendor-owned open source projects, irrespective of what drives them to do so.
- Developers contribute to vendor-owned open source projects primarily to improve their coding skills, to enhance the software that they use, and to contribute to something bigger than themselves.
- Improving software they already use is a dominant driving force behind contributing to corporate open source projects for all developers, but especially so for the most senior ones.

Types of studios game developers work for
- Professional game developers are almost equally distributed across different game studio types.
- There’s a disproportionate concentration of highly experienced developers (with 16+ years of experience under their belts) at indie studios compared to other studio types.
- Game developers working for indie studios tend to wear many hats, self-identifying with multiple roles at the same time to a much larger extent than developers in other company types.

The rise and fall of web frameworks
- Web developers who use frameworks are more likely to be high-performers in software delivery than those who don’t.
- Web developers are gradually settling for a smaller number of frameworks as they stop experimenting with a wide range of tools.
- React is currently the most widely used client-side framework and its adoption has remained stable over the past two years. By comparison, jQuery’s popularity is decreasing rapidly.
THE STATE OF BLOCKCHAIN DEVELOPMENT

The distributed transaction ledgers and smart contracts that comprise blockchain technologies have applications in a wide range of industries, particularly in finance, logistics, and government. In this chapter, we present an overview of who is involved in blockchain development and which blockchain platforms they’re using.
Engagement with blockchain technologies

Of the three blockchain technologies that we track in our survey, non-fungible tokens (NFTs) garner the least attention from developers – 58% of them show no interest, likely due to its perception as a novelty technology. On the other hand, cryptocurrencies are most salient to developers – 27% are either learning about or currently working on such projects and non-cryptocurrency blockchain technologies are very close behind, with 25% of developers similarly involved.

In this chapter, we will focus solely on blockchain applications other than cryptocurrencies, as these technologies have the widest range of use cases and thus the most potential to shape our world. We’ll begin by looking at developers’ engagement with blockchain technologies from a regional standpoint then through the lens of experience in software development. Finally, we’ll give an overview of which blockchain platforms are being used.
A regional view of engagement with blockchain applications

Engagement with blockchain applications other than cryptocurrencies, referred to as blockchain applications from here on, varies greatly depending on where developers are located. North America and East Asia excluding Greater China are hotbeds of blockchain development - 15% and 12% of developers in these regions, respectively, are currently working on blockchain applications, with another 17% learning about the technology in both regions.

Further down the list, we see that while the Middle East and Africa has a smaller proportion of developers actively working on blockchain projects; it has the highest incidence of those learning about them (20%). This suggests that the Middle East and Africa could well become important for blockchain development in the future. Indeed, given the region’s history of rapid adoption of new foundational technologies – exemplified by Africa’s mobile banking revolution – blockchain applications in finance and banking are particularly exciting here – though the data suggests that there may still be some way to go.
Finance and banking professionals in the Middle East and Africa are more interested in blockchain technologies than finance and banking professionals in other regions.

About a quarter of professional developers in the Middle East and Africa who are interested in blockchain technologies are also working in the finance and banking sector. It seems these developers have seen the potential for this technology to shape and disrupt the sector and are getting a head start. Looking at this from the other side, we see that the proportion of finance and banking professionals who are currently working on or learning about blockchain applications is broadly in line with the average for the region (31% vs 33%). However, the proportion who are interested in blockchain applications is 29% higher (37% vs 29%). This is the highest incidence of interest in blockchain applications amongst finance and banking professionals across any region and indicates that blockchain applications could play a pivotal role in this industry in the future.

Tip: You can find more regional data and apply filters in the SlashData Dashboards: [Try it](#)
In North America, almost as many developers are currently working on blockchain projects as are learning about them.

% of developers in each region (Q3 2022 n=24,488)

- **Currently working on**
- **Learning about**
- **Interested in**
- **Not interested**

North America: 41% currently working, 27% learning about, 17% interested in, 15% not interested.

East Asia ex. Greater China: 45% currently working, 17% learning about, 17% interested in, 12% not interested.

Middle East & Africa: 41% currently working, 29% learning about, 20% interested in, 10% not interested.

Oceania: 47% currently working, 30% learning about, 15% interested in, 9% not interested.

Western Europe & Israel: 51% currently working, 26% learning about, 15% interested in, 7% not interested.

South Asia: 46% currently working, 30% learning about, 17% interested in, 7% not interested.

South America: 44% currently working, 34% learning about, 16% interested in, 5% not interested.

Greater China area: 51% currently working, 27% learning about, 17% interested in, 4% not interested.

Eastern Europe, Russia & Former CIS: 61% currently working, 27% learning about, 17% interested in, 10% not interested.
How does experience affect engagement with blockchain applications?

Developers with 6-10 years of experience are the most likely to work on blockchain projects. It’s likely that these developers have reached the point in their career where their technical skills are sufficiently advanced to enable them to work on such demanding projects. On the other hand, we see that the least and most experienced developers are the most likely to be disinterested in such projects. Those with less than a year under their belts have yet to build their knowledge, while the most experienced developers may be looking to maintain some stability in their career and are reluctant to change tack.

Developers with 11-15 years of experience under their belts show the most passive interest in blockchain applications – whilst they aren’t learning about or working on such projects, 32% are interested in some way. These developers are at the zenith of their careers and whilst they are some of the least likely to be learning about the technology, they are also some of the most likely to be currently working on it. These developers are likely keeping a close eye on developments in the space – should they spot an opportunity, they will be able to pivot their considerable experience to become effective contributors to the space.
Interestingly, although we see that the least experienced developers are less likely than their counterparts with 1-15 years of experience to be currently working on blockchain applications, they are only slightly less likely to be learning about these projects. This demonstrates that although they lack the skills to be active contributors, the myriad applications and potential of blockchain applications are a powerful draw. We can expect that, as learning materials improve and the barriers to entry reduce – as is the case with many technologies – over the next few years, developers will be able to get involved in blockchain projects much earlier in their careers.
Which blockchain platforms are most popular?

Looking at the specific blockchain platforms that developers report using, we see that Ethereum is clearly dominant amongst learners and those actively developing alike. It’s also unique amongst the blockchain technologies that we ask about, in that it is the only one which is more popular amongst those learning about the technology than those who are currently working on it. This indicates that Ethereum’s ecosystem is in good shape – not only is it large, but it also has a healthy pipeline of new contributors.

Although new contributors are certainly good news – the utility of a blockchain rises with the number of applications that use it – a large influx can also create problems. For example, too many transactions on a network can slow things down severely and greatly increase the price of a transaction. For example, when CryptoKitties surged in popularity, the cost of a transaction on the Ethereum network increased ninefold, from around $50 to over $450. Such is the price of success. This said, Ethereum’s recent transition to a proof-of-stake model is expected to reduce energy consumption by 99.95% and makes the platform more scalable, secure, and sustainable, potentially mitigating this pitfall.

Further down the list, Binance Smart Chain is the second most widely used blockchain platform and is used significantly more by active developers than learners. Here, the Binance Smart Chain not only benefits from its association with the Binance cryptocurrency trading platform, but also its interoperability with the Ethereum blockchain. We also see a similar story with the IBM Blockchain platform – this platform is based on open source blockchain technology managed by the Linux foundation and clearly benefits from the backing of these two large organisations and their developer and business communities.
Despite the hype, blockchain technologies are still somewhat in their infancy. Blockchain, much like cloud computing fifteen or so years ago, has the potential to underpin and enable many other technologies and experiences, but as we saw earlier, only 9% of developers are currently working on such projects. Rather than affecting an instant technological transformation, blockchain technologies have the opportunity to become a foundational technology on which our digital experience sits, much like TCP-IP – the building blocks of the internet – and developers will be key players in shaping this particular view of the future.
Having long-term career aspirations can provide students with a sense of direction and help them make appropriate choices in their pursuit of knowledge. In turn, this speeds up their professional development and increases the likelihood of them achieving a successful career. In this chapter, we will take a closer look at the top career aspirations of developers who self-identify as students.
In the latest edition of our Developer Nation survey, we asked student developers about their top career aspirations in software development. As it turns out, the vast majority (88%) of students have at least one specific career aspiration, while only 7% are not planning to become professionals. Furthermore, an average of 2.5 career aspirations were mentioned by the former group of respondents, with a large variety of combinations.

Solving problems is the top career aspiration among student developers, mentioned by nearly a third (32%) of them. This is closely followed by becoming an expert in a domain or technology (29%), building innovative products or services (27%), and working on challenging projects (26%). These findings suggest that, despite an apparent financial appeal, curiosity and chasing innovation are the primary motivators for students in their journeys to becoming professional software developers. However, it is worth noting that maximising their earning potential is important for students too, ranking sixth on the list of top career aspirations.
Having good problem-solving skills when working on challenging projects is usually the key to building innovative products and services. As such, it is no surprise that these aspirations are frequently mentioned together by students who want to become professional developers.

Those who want to build innovative products or services show an above-average level of interest in becoming entrepreneurs or working at esteemed companies. Further analysis reveals that aspiring entrepreneurs are more likely to focus on maximising their impact on society, while those who express a desire to work for acclaimed companies show more interest in pursuing challenging projects.
Students who seek to become tech executives or company leaders place significantly lower importance than average on the most popular career aspirations. Instead, they prioritise building their own businesses while also showing an above-average level of interest in getting a specific job title. This is likely due to them being naturally career-focused and prioritising pathways that will allow them to achieve their long term goals.

On the other end of the spectrum, those who want to maximise their impact on society show a similar level of interest in building their own businesses. These developers are the most likely to aspire to build innovative products or services while also showing a high level of interest in mentoring and helping others grow.

*Students who aspire to maximise their impact on society show a high level of interest in innovation and building their own businesses*
Students who aspire to become company leaders show the highest entrepreneurial spirit

% of student developers reporting each career aspiration that also report each additional aspiration (n=4,181)

<table>
<thead>
<tr>
<th>Additional career aspirations</th>
<th>Build innovative products / services</th>
<th>Build my own business / become an entrepreneur</th>
<th>Work at an esteemed company</th>
<th>Maximise my impact on society</th>
<th>Become a tech executive / company leader</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solve problems</td>
<td>41%</td>
<td>32%</td>
<td>32%</td>
<td>37%</td>
<td>27%</td>
</tr>
<tr>
<td>Become an expert in a domain or technology</td>
<td>34%</td>
<td>27%</td>
<td>30%</td>
<td>34%</td>
<td>25%</td>
</tr>
<tr>
<td>Build innovative products/services</td>
<td>34%</td>
<td>24%</td>
<td>32%</td>
<td>26%</td>
<td>23%</td>
</tr>
<tr>
<td>Work on challenging projects</td>
<td>32%</td>
<td>17%</td>
<td>29%</td>
<td>24%</td>
<td>32%</td>
</tr>
<tr>
<td>Build my own business / become an entrepreneur</td>
<td>30%</td>
<td>20%</td>
<td>17%</td>
<td>14%</td>
<td>23%</td>
</tr>
<tr>
<td>Maximise my earnings potential</td>
<td>25%</td>
<td>21%</td>
<td>21%</td>
<td>26%</td>
<td>18%</td>
</tr>
<tr>
<td>Work at an esteemed company/organisation</td>
<td>23%</td>
<td>20%</td>
<td>19%</td>
<td>15%</td>
<td>15%</td>
</tr>
<tr>
<td>Mentor and help others grow</td>
<td>20%</td>
<td>13%</td>
<td>11%</td>
<td>10%</td>
<td>14%</td>
</tr>
<tr>
<td>Maximise my impact on society</td>
<td>22%</td>
<td>18%</td>
<td>13%</td>
<td>10%</td>
<td>15%</td>
</tr>
<tr>
<td>Become a tech executive / company leader</td>
<td>13%</td>
<td>10%</td>
<td>15%</td>
<td>12%</td>
<td>15%</td>
</tr>
<tr>
<td>Get a specific job title</td>
<td>11%</td>
<td>11%</td>
<td>14%</td>
<td>12%</td>
<td>11%</td>
</tr>
<tr>
<td>Get recognition and credibility</td>
<td>11%</td>
<td>11%</td>
<td>14%</td>
<td>12%</td>
<td>11%</td>
</tr>
</tbody>
</table>
As with most other topics, regional differences in culture and socio-economic circumstances manifest themselves as significant differentiators in career aspirations amongst student developers. For instance, South Asia is the region with the highest concentration of developers who self-identify as students (40%). In this region, 30% of students aspire to work on challenging projects — their top motivation — but are significantly less likely than students in other regions to show an interest in solving problems (26%) or building innovative products/services (22%). Instead, South Asian students are the most likely to prioritise obtaining a specific job title (16%) while also showing an above-average level of interest in becoming tech executives or company leaders (15%).

Students in South America, Eastern Europe, and Western Europe show the highest levels of interest in the global top-four career aspirations. However, while European students are significantly more focussed than average on maximising their earning potential (>30%), only 21% of their South American counterparts prioritise this when planning their careers. Instead, South American students are highly focussed on securing job opportunities at esteemed companies/organisations. In turn, they may be more likely to accept junior positions and lower salaries in exchange for job security and a promising career path.
Students from North America and the Middle East and Africa show the highest entrepreneurial spirit

In North America, a high portion of students aim to become entrepreneurs while also showing the highest level of interest in maximising their impact on society. This is likely due to the highly prominent startup culture in this region. This effect is most apparent in Silicon Valley, which has established itself as the nexus of technological innovation and is home to many tech giants and startups. Similarly, students from the Middle East and Africa also show high entrepreneurial spirit in a region that is primed for a booming startup scene. In addition to this, the students in this region are the most likely to prioritise transferring their knowledge and experience to the next generation (28%).

East Asian students are the least likely to express a specific interest in becoming professional software developers. This suggests that students in this region have doubts about pursuing careers in software development and may be looking at other, unrelated jobs. Similarly, students from the Greater China area show below-average levels of interest in many of the top choices. In particular, we find that only 6% of them prioritise building their own businesses or becoming entrepreneurs, which is 19 percentage points below the average of the other regions. However, maximising their earning potential appears to be the key driving force behind why 33% of the students from the Greater China area want to become professional developers.
European students are by far the most likely to want to become problem solvers

% of student developers in each region reporting each career aspiration and concentration of students in each region (n=4,746)

<table>
<thead>
<tr>
<th>Region</th>
<th>Solve problems</th>
<th>Become an expert in a domain or technology</th>
<th>Build innovative products/services</th>
<th>Work on challenging projects</th>
<th>Build my own business / become an entrepreneur</th>
<th>Maximise my earnings potential</th>
<th>Work at an esteemed company/organisation</th>
<th>Mentor and help others grow</th>
<th>Maximise my impact on society</th>
<th>Become a tech executive / company leader</th>
<th>Get a specific job title (e.g. product manager)</th>
<th>Get recognition and credibility</th>
<th>Not planning to become a professional</th>
<th>No specific aspirations</th>
</tr>
</thead>
<tbody>
<tr>
<td>South Asia</td>
<td>26%</td>
<td>29%</td>
<td>22%</td>
<td>30%</td>
<td>19%</td>
<td>23%</td>
<td>21%</td>
<td>16%</td>
<td>14%</td>
<td>15%</td>
<td>16%</td>
<td>9%</td>
<td>7%</td>
<td>4%</td>
</tr>
<tr>
<td>Middle East &amp; Africa</td>
<td>35%</td>
<td>28%</td>
<td>27%</td>
<td>22%</td>
<td>32%</td>
<td>16%</td>
<td>17%</td>
<td>36%</td>
<td>19%</td>
<td>18%</td>
<td>10%</td>
<td>15%</td>
<td>10%</td>
<td>5%</td>
</tr>
<tr>
<td>South America</td>
<td>35%</td>
<td>38%</td>
<td>40%</td>
<td>33%</td>
<td>24%</td>
<td>21%</td>
<td>36%</td>
<td>18%</td>
<td>15%</td>
<td>9%</td>
<td>11%</td>
<td>15%</td>
<td>11%</td>
<td>5%</td>
</tr>
<tr>
<td>East Asia ex. Greater China</td>
<td>20%</td>
<td>16%</td>
<td>18%</td>
<td>16%</td>
<td>6%</td>
<td>17%</td>
<td>13%</td>
<td>12%</td>
<td>8%</td>
<td>9%</td>
<td>11%</td>
<td>14%</td>
<td>11%</td>
<td>5%</td>
</tr>
<tr>
<td>Greater China area</td>
<td>26%</td>
<td>29%</td>
<td>16%</td>
<td>27%</td>
<td>32%</td>
<td>33%</td>
<td>21%</td>
<td>21%</td>
<td>7%</td>
<td>16%</td>
<td>11%</td>
<td>11%</td>
<td>4%</td>
<td>4%</td>
</tr>
<tr>
<td>North America</td>
<td>34%</td>
<td>29%</td>
<td>16%</td>
<td>27%</td>
<td>21%</td>
<td>22%</td>
<td>21%</td>
<td>19%</td>
<td>21%</td>
<td>19%</td>
<td>11%</td>
<td>11%</td>
<td>4%</td>
<td>1%</td>
</tr>
<tr>
<td>Eastern Europe, Russia &amp; Former CIS</td>
<td>43%</td>
<td>30%</td>
<td>34%</td>
<td>30%</td>
<td>31%</td>
<td>33%</td>
<td>33%</td>
<td>30%</td>
<td>30%</td>
<td>38%</td>
<td>11%</td>
<td>20%</td>
<td>11%</td>
<td>10%</td>
</tr>
<tr>
<td>Western Europe &amp; Israel</td>
<td>42%</td>
<td>33%</td>
<td>38%</td>
<td>30%</td>
<td>33%</td>
<td>30%</td>
<td>30%</td>
<td>20%</td>
<td>19%</td>
<td>16%</td>
<td>11%</td>
<td>15%</td>
<td>15%</td>
<td>10%</td>
</tr>
</tbody>
</table>

Concentration of students: 40% to 10%
The choice of programming language matters deeply to developers because they want to keep their skills up to date and marketable. Languages are a beloved subject of debate and the kernels of some of the strongest developer communities. They matter to toolmakers too, because they want to make sure they provide the most useful SDKs.
It can be hard to assess how widely used a programming language is. The indices available from players like Tiobe, Redmonk, Stack Overflow’s yearly survey, or GitHub’s State of the Octoverse are great, but offer mostly relative comparisons between languages, providing no sense of the absolute size of each community. They may also be biased geographically or skewed towards certain fields of software development or open source developers.

The estimates we present here look at active software developers using each programming language; across the globe and across all kinds of programmers. They are based on two pieces of data. First is our independent estimate of the global number of software developers, which we published for the first time in 2017. We estimate that, as of Q3 2022, there are 33.6 million active software developers in the world.

Second is our large-scale, low-bias surveys which reach tens of thousands of developers every six months. In these surveys, we have consistently asked developers about their use of programming languages across 13 areas of development, giving us rich and reliable information about who uses each language and in which context.
In 2020, Python overtook Java as the second most widely used language and now counts nearly 17M developers in its community. Python has continued to show strong growth, having added about 8M net new developers over the last two years. The rise of data science and machine learning (ML) is a clear factor in Python’s growing popularity. To put this into perspective, about 63% of ML developers and data scientists report using Python. In comparison, less than 15% use R, the other language often associated with data science.

JavaScript remains the most popular programming language for the 11th survey in a row, with over 19.5M developers worldwide using it. Notably, the JavaScript community has been growing in size consistently for the past several years. Between Q3 2020 and Q3 2022, Javascript experienced a 59% increase as 7.3M developers joined the community – one of the highest growths in absolute terms across languages. Not only do new developers see it as an attractive entry-level language, but existing ones are also adding it to their skillset. JavaScript’s popularity extends across all sectors, with at least a quarter of developers in every sector using it.
Java is one of the most important general-purpose languages as well as the cornerstone of the Android app ecosystem. Although it has been around for more than two decades, it continues to experience strong growth. In the last two years, Java has almost doubled the size of its community, from 8.3M to 16.5M. For perspective, the global developer population grew about half as fast over the same period. Within the last year alone, Java has added 6.3M developers, the largest absolute growth of any language community. Our data suggests that Java’s growth is supported not only by the usual suspects, i.e. backend and mobile development, but also by its rising adoption in AR/VR projects, likely due to Android’s popularity as an AV/VR platform.

The group of major, well-established languages is completed with C/C++ (12.3M), C# (10.6M), and PHP (8.9M). PHP has seen the slowest growth rate of all languages over the last year, growing just 22%, adding 1.6M net new developers. PHP is a common choice for cloud and web developers, but has seen decreasing popularity, particularly amongst web developers where it has gone from the second most popular language in Q3 2021 behind JavaScript, to the fourth most popular in Q3 2022, with Python and Java becoming more popular choices.

There has been slow growth for PHP over the last year, as Python and Java have become more popular choices for web development

C and C++ are core languages in embedded and IoT projects, for both on-device and application-level coding, but also in mobile and desktop development, which are sectors that attract 17.7M and 15.6M developers respectively. C#, on the other hand, has maintained its popularity among multiple different areas of software development, particularly among desktop and game developers. C/C++ added 4.3M net new developers in the last year and C# added 2.8M over the same period.
Ranking of programming languages communities over the last 24 months
(Q3 2020 n=17,241 | Q1 2021 n=17,801 | Q3 2021 n=19,319 | Q1 2022 n=20,041 | Q3 2022 n=26,183)

* JavaScript includes CoffeeScript and TypeScript
Rust and Kotlin continue their rise in popularity

We have previously identified Rust and Kotlin as two of the fastest-growing language communities and this continues to be the case. Rust has more than tripled in size in the past two years, from just 0.8M developers in Q3 2020 to 2.8M in Q3 2022. Rust has added 0.7M developers in the last six months alone and is close to overtaking Objective C to become the 11th largest language community. Rust has formed a strong community of developers who care about performance, memory safety, and security. As a result, it has seen increased adoption in IoT software projects, but also in desktop and game development, where Rust is desired for its ability to build fast and scalable projects.

The Rust community has more than tripled in size in the last two years

Kotlin has also seen large growth in the last two years, more than doubling in size from 2.3M in Q3 2020 to 6.1M in Q3 2022. As such, it went from the ninth to the seventh largest language community during this period of time, overtaking Swift and those using visual development tools. This growth can largely be attributed to Google’s decision in 2019 to make Kotlin its preferred language for Android development and it is currently used by a fifth of mobile developers and is the second most popular language for mobile development. Despite Google’s preference for Kotlin, the inertia of Java means that, after three years, it is still the most popular language for mobile development.
Swift currently counts 4.2M developers and is the default language for development across all Apple platforms. This has led to a phase-out of Objective C (3M) from the Apple app ecosystem. However, Objective C has maintained its place among IoT developers and an increasing adoption for on-device code, and AR/VR developers, leading to a similar increase in the number of Swift and Objective C developers in the last two years; 1.8M and 1.6M respectively.

The more niche languages – Go, Ruby, Dart, and Lua – are still much smaller, with less than 4M active developers each. Go and Ruby are important languages in backend development, but Go is the third fastest growing language community and has added more than twice as many developers as Ruby in the last two years; 2.3M and 1.0M new developers, respectively. This is likely due to the fast development cycle it offers even though it is a compiled language. Dart has seen steady growth in the last two years, predominantly due to the increasing adoption of the Flutter framework in mobile development, with 13% of mobile developers currently using Google’s language. Finally, Lua is among the fastest growing language communities, mainly drawing in IoT, game, and AR/VR developers looking for a scripting alternative to low-level languages such as C and C++.
Open source software (OSS) development is a major part of developer culture and represents a unique collaborative environment. In this chapter, we will explore the motivations behind vendor-owned OSS contributions through the lenses of experience, global region, and the use of Stack Overflow.
OSS projects represent the power of community: collaborative efforts to develop code and software which positively impacts a wider audience than the individuals involved. Vendor-owned OSS projects, e.g. TensorFlow and Visual Studio Code, combines this sense of community with financial backing from the world’s largest tech companies – a powerful combination of stability and open cooperation.

For every developer involved in vendor-owned OSS, there is a different motivating factor – why do developers contribute to these projects? The big picture is that the top-three motivators for vendor-owned OSS contributors are: wanting to learn how to code better (38%), to improve the software that they use (29%), and to contribute to something bigger than themselves (22%).

How does experience affect vendor-owned OSS contribution?

When compared to beginners, those with six years of experience or more are around 13 percentage points more likely to contribute in order to improve a software they use. These experienced and improvement-focussed developers are also much more likely to hold specialist roles compared to their less experienced peers. For example, they are six times more likely to be software architects and five times more likely to be either tech/engineering team leads or site reliability engineers. They not only believe that the software they use can be improved, but that they also have the capability and skills to improve it.
Experienced developers are highly focussed on improving vendor-owned OSS that they use

In fact, improving software seems to be the main motivation for many senior developers – those with 16 years of experience or more are the least likely to contribute for the majority of the other reasons we list. Learning to code better, getting noticed by their company, and getting their code reviewed are much lower priorities among seasoned developers. This is to be expected given the amount of expertise and recognition they have typically accumulated by that stage of their career.

At the other end of the scale, those most willing to contribute for their own education are developers with 1-2 years of experience. Compared to those with even less experience, these developers are 58% more likely to be exclusively professionals and 48% less likely to be exclusively students. In other words, at this stage of their careers, they have enough professional know-how and confidence to contribute to vendor-owned OSS software – yet are pursuing further education for their coding skills by giving back to the community.
How motivation to contribute to vendor-owned OSS projects changes with developer experience

% of developers who contribute to vendor-owned OSS projects by experience level (Q3 2022 n=22,914)

<table>
<thead>
<tr>
<th>Years of general software development experience</th>
<th>&lt; 1 year</th>
<th>1-2 years</th>
<th>3-5 years</th>
<th>6-10 years</th>
<th>11-15 years</th>
<th>16+ years</th>
</tr>
</thead>
<tbody>
<tr>
<td>Getting noticed by the company</td>
<td>13%</td>
<td>16%</td>
<td>18%</td>
<td>14%</td>
<td>11%</td>
<td>8%</td>
</tr>
<tr>
<td>Getting my code reviewed by knowledgeable people</td>
<td>17%</td>
<td>23%</td>
<td>26%</td>
<td>22%</td>
<td>18%</td>
<td>10%</td>
</tr>
<tr>
<td>Learning to code better</td>
<td>38%</td>
<td>44%</td>
<td>40%</td>
<td>37%</td>
<td>34%</td>
<td>27%</td>
</tr>
<tr>
<td>Contributing to something bigger than me</td>
<td>17%</td>
<td>20%</td>
<td>23%</td>
<td>25%</td>
<td>22%</td>
<td>21%</td>
</tr>
<tr>
<td>Contributing to something that has company backing</td>
<td>13%</td>
<td>16%</td>
<td>20%</td>
<td>20%</td>
<td>15%</td>
<td>8%</td>
</tr>
<tr>
<td>I use that software and want to improve it</td>
<td>18%</td>
<td>24%</td>
<td>28%</td>
<td>36%</td>
<td>34%</td>
<td>38%</td>
</tr>
<tr>
<td>Translating documentation</td>
<td>10%</td>
<td>14%</td>
<td>17%</td>
<td>14%</td>
<td>13%</td>
<td>6%</td>
</tr>
<tr>
<td>Building community support around a corporate open source software project</td>
<td>14%</td>
<td>21%</td>
<td>22%</td>
<td>19%</td>
<td>15%</td>
<td>11%</td>
</tr>
</tbody>
</table>

- >5pp below the average of other levels of experience
- 2.5 – 5pp below the average of other levels of experience
- ±2.5pp around the average of other levels of experience
- 2.5 – 5pp above the average of other levels of experience
- >5pp above the average of other levels of experience
Vendor-owned OSS contribution around the world

According to our data, 73% of developers contribute to vendor-owned OSS globally, but the level of contribution varies around the world. Developers in South Asia are the most likely to contribute (85%), while those in Eastern Europe are the least likely (67%). As for the two largest regional developer communities, North America and Western Europe, 78% and 70% of developers contribute to corporate OSS projects, respectively.

As for specific motivations, there are a couple of hotspot regions that stand out from the crowd. Nearly half (47%) of OSS contributors in the Middle East and Africa and South Asia are motivated by learning to code better and similarly, about one in four by the opportunity to have their code reviewed by more experienced colleagues: 10 and 5 percentage points above the global average, respectively.

South Asia and the Middle East and Africa are hotspots for developers contributing to vendor-owned OSS projects in order to level up their coding skills

Tying in with our previous analysis: these regions also hold the two largest shares of developers with less than two years of experience – 52% for the Middle East and Africa and 73% for South Asia.
However, to see how motivations towards vendor-owned OSS change across the globe, we take a wider perspective. In doing so, we group motivations into three broad categories: individual-focussed\(^1\), collaboration-focussed\(^2\), and business-focussed\(^3\). In this manner, we can get a view of how sentiments towards vendor-owned OSS change around the world.

For instance, we see that developers in Oceania are at least 5 percentage points more likely than any other region to have business-focussed motivations when contributing to vendor-owned OSS projects. This may be linked to the financial success/focus of developers in this region – 9% of OSS contributors in Oceania report that they or their organisation generate more than $1M of revenue every month on average, compared to the global average of 4%.

An interesting note on gender: we see that globally, female developers are 26% more likely than male developers to be business-focussed in their approach to vendor-owned OSS contribution. This observation is particularly strong in Europe: 54% of female developers in Western and Eastern Europe are business-focussed, compared to 33% of male developers. However, as the proportion of OSS-contributing female developers (22%) is only slightly higher than the global proportion (21%), it’s unlikely that they drive business-focussed regional behaviour.

---

\(^{1}\) Individual-focussed: Getting noticed by the company, learning to code better, I use that software and want to improve it.

\(^{2}\) Collaboration-focussed: Getting my code reviewed by knowledgeable people, contributing to something bigger than me, translating documentation.

\(^{3}\) Business-focussed: Contributing to something that has company backing to succeed, building community support around a corporate open source software project.
Developers in Oceania are the most likely to be driven by business-related motivators

% of developers who contribute to vendor-owned OSS projects by region (Q3 2022 n=16,931)
How do OSS contributors use Stack Overflow?

Let’s look at the usage of a website that is synonymous with cooperation in programming and software development and see how the proportion of OSS contributors changes with varying levels of interaction. For users of Stack Overflow, we see a behavioural trend – those who are more active on the website are more likely to contribute to vendor-owned OSS.

Diving into the specific usage patterns of Stack Overflow, those who don’t use or visit the site are the least likely to contribute to vendor-owned OSS for any reason, compared to those who use the site at any level. This is again related to experience: 39% of those who don’t use Stack Overflow have less than a year of software development experience and only 5% have an account with a badge; these developers are the least likely to contribute to vendor-owned OSS projects, after those with more than 16 years of experience.

Higher activity on Stack Overflow is associated with higher vendor-owned OSS contribution rates

% of developers (Q3 2022 n=12,708)

- I don’t use/visit Stack Overflow: 32% contribute to open source, 68% do not contribute.
- I visit Stack Overflow, but I don’t have an account: 25% contribute to open source, 75% do not contribute.
- I have an account on Stack Overflow, but no badges: 19% contribute to open source, 81% do not contribute.
- I have an account and I have earned at least one badge: 16% contribute to open source, 84% do not contribute.
Likewise, there are differences in motivations to contribute to vendor-owned OSS between those with or without Stack Overflow badges. For example, only 28% of OSS-contributing developers without a badge want to improve the software they use, in contrast to 40% of developers with badges. A possible driver here is professional status – 74% of those without a badge are professionals. For those with a badge, 91% are professionals: these developers are not only more focussed on improvement, they are more willing to engage with the community to do so.

Vendor-owned OSS projects represent the power of community: collaborative efforts to develop software which positively impacts a wider audience than the individuals involved. Here, we’ve shown that developers involved in vendor-owned OSS have different motivations depending on their experience, gender, and region, which in turn reflects how they use collaborative environments like Stack Overflow.
The nature of professional game developers’ work can vary depending on the type of studio they work for. In this chapter, we will explore the profile of developers working for different types of game studios, focussing on their experience, roles, and technology choices.
An initial point to note is that game developers are relatively equally spread between the different types of studio. More specifically, 21% work at large scale studios, 19% at third-party studios, and 18% at both game publishers and indie studios.

Game developers may be equally spread between different studio types, but we observe some important differences in their profiles. Developers with 3-5 years of experience are most commonly found in all types of studio except third-party developers; for this type of studio, their team is most likely to have 1-2 years of experience. Those with more than 11 years of experience in game development make up only a small proportion of the general game developer population (11%), but have a much higher representation at indie studios (20%). The most experienced developers in the industry are able to choose where they work. They may be choosing to work for indie studios due to greater creative control over projects or a preference for working within small teams now rarely found in AAA development. They may also be making personal decisions to work for studio types that are less likely to be involved in ‘crunch’.

For the first time in the latest edition of our Developer Nation survey, we asked professional game developers to describe the type of game studio they work for. The modern game development space has stratified itself into several different studio types, and within these studios the types of tools and technologies used can vary significantly. Further to this, the profile of developers also shows differences between studio types.

We break down professional game developers as belonging to the following studio types:

1. Game publishers, who outsource the majority of their development;
2. Large-scale studios, who develop and self-publish a collection of games;
3. Third-party developers, who work on a wide range of games for different publishers;
4. Indie studios that publish and develop a small number of games.
Mid-career game developers – those with 6 to 15 years of experience – make up more than a third (37%) of the developer workforce in large-scale studios, compared to only a quarter (24%) among professional game developers in general. The scope and complexity of the properties that large-scale studios work on may drive a demand for more experienced developers. Despite this, the most experienced developers still more frequently choose indie studios, indicating that the previously suggested factors may outweigh the offers large-scale studios can make to these developers.

Developers at indie studios are twice as likely to be highly experienced – 16 or more years under their belts – than those working for other other studio types

Want to see a regional breakdown of this data? Go to SlashData Dashboards and apply the “region” filter to see where developers live: Try it
Third-party developers have an interesting distribution of developer experience. 36% of their developers have between one and two years of experience and 8% have 11 to 15 years, compared to 26% and 6% among game developers in general, respectively. This indicates that third-party studios are more willing to hire less experienced developers. New developers may also view third-party studios as elevator studios, giving them exposure to multiple different projects so that they can transfer to other studio types. However, these studios still require a healthy proportion of experienced developers to manage multiple projects, leading to hiring more as a proportion compared to the overall game developer population.
Of the many roles game developers may hold, we consider game designers, artists, UI designers, programmers, and QA engineers as the ones that consumers would likely identify with game development, and are often the ‘core’ roles for producing games. Differences in studios can be seen by the different percentages of developers identifying themselves with these roles.

Those who self-identify as programmers account for 39% of those working for indie studios, which is significantly higher than the professional game developer average (24%). Similarly, the proportion of developers working for indie studios describing their roles as either game designer (46%), artist (27%), or UI designer (14%) is nearly double that of the population average, 23%, 15%, and 8% respectively. The proportion of these roles between studios makes sense considering the nature of development within these types of studios. Indie studios are typically smaller than other studios, with 66% of indie developers working for companies with up to 20 employees, compared to only 43% and 34% of developers working for game publishers and third-party studios, respectively. This can lead to more employees in a studio being directly involved in game design and development.

Third-party studios have twice the proportion of developers in test roles compared to other game studios

A counter example can be seen with QA engineers, who are twice as prevalent in third-party studios (10%) compared to the population average (5%). Third-party studios having a larger proportion of QA engineers corresponds to their role in development cycles. These studios do a lot of development in-house, often work on larger projects, and can undertake contract work for larger studios, all of which require dedicated QA departments.
Developers working for indie studios are more likely to wear many hats, taking on both programming and design responsibilities

% professional game developers working for each studio type that hold each role (n=1,117)

<table>
<thead>
<tr>
<th>Role</th>
<th>Game publisher</th>
<th>Large-scale developer</th>
<th>Third-party developer</th>
<th>Indie developer</th>
<th>All professional game developers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programmer / software developer (incl. frontend, backend, full-stack)</td>
<td>12%</td>
<td>25%</td>
<td>24%</td>
<td>46%</td>
<td>12%</td>
</tr>
<tr>
<td>Game designer</td>
<td>9%</td>
<td>11%</td>
<td>10%</td>
<td>27%</td>
<td>19%</td>
</tr>
<tr>
<td>Artist/creator</td>
<td>15%</td>
<td>10%</td>
<td>14%</td>
<td>5%</td>
<td>10%</td>
</tr>
<tr>
<td>UI designer</td>
<td>6%</td>
<td>9%</td>
<td>5%</td>
<td>8%</td>
<td>5%</td>
</tr>
<tr>
<td>Test/QA developer or engineer</td>
<td>6%</td>
<td>5%</td>
<td>10%</td>
<td>4%</td>
<td>5%</td>
</tr>
</tbody>
</table>

12% of professional game developers working for each studio type that hold each role (n=1,117)
Further differences between game studios can be found in the choices of game engines. Unity and Unreal Engine are the most used game engines, with 33% and 15% of game developers using them as their primary engine, respectively. However, among developers who work for indie studios, these two game engines account for 48% and 20% of developers, compared to less than 28% and 12% for developers at other studios. Both engines are widely used and popular, with Unity being the most used engine for all studio types, but have specific business and technical aspects that appeal to indie developers.

Due to its flexibility and ease of use, Unity was the common choice for indie developers when the scene was emerging. This has led to a large online community and marketplace to support indie developers with tutorials, assets, and customised libraries. Unreal Engine is also popular amongst indie developers with it being considered one of the most powerful out-of-the-box engines and having a licensing structure that doesn’t require royalties to be paid until a game makes more than $1 million in revenue. The difference in popularity between the two game engines for indie developers likely lies in the availability of assets in the Unity store. Unity Asset Marketplace has over 77,000 assets and tools, compared to Unreal Marketplace’s 22,000, allowing small indie studios to offset development time with ready-made assets and tools.
Unity and Unreal Engine are the most popular engines for game developers, with particularly high adoption by developers at indie studios

Unity and Unreal have a large number of tools and utility, but are not capable of performing every possible aspect of game design. In contrast, in-house or custom tools allow developers to focus on working with engines designed around the specific requirements of their games, as well as develop tools to optimise both development and performance. Amongst the large-scale and third-party studios, the second most popular engine choice is the use of either an in-house or proprietary engine – 16% and 12%, respectively. These studios have the resources, time, and business motivation to focus on developing their own engines. Use amongst the large-scale developers is further incentivised by allowing assets and developers to move between projects more seamlessly, with a greater familiarity with the engine.

Game publishers have a similar level of resources and time, but their in-house engines are often pivoted to commercial engines. Unreal Engine was developed by publisher Epic Games, Source from Valve, and RedEngine from CD Projekt Red. Among game publishers, the engines initially developed by a publisher but are no longer in-house, make up another 15% of primary engine choices.

Less than 5% of developers working at indie studios use in-house engines; instead, Godot is the next most common engine choice. Godot is an open source game engine that has built a strong community of developers around it. It has created supporters due to its dedicated 2D engine and its own Python-like language GDScript that accommodates the many Python users in the world, which is especially popular among student developers. The open source nature of the engine also means indie developers do not have to worry about licensing or subscription changes, reducing financial demands and worries.
Nearly half of indie studio developers use Unity as their primary game engine, by far the highest adoption across different studio types.

% of professional game developers working for each studio type that use each game engine as their primary tool (n=1,117)

- Unity: 48%
- Unreal Engine: 28%
- In-house / proprietary game engine: 19%
- Amazon Lumberyard: 17%
- GameMaker: 13%
- Godot: 14%
- Source: 22%
- Other Engines: 36%

Game publisher: Dark blue
Large-scale developer: Light blue
Third-party developer: Teal
Indie developer: Grey
All professional game developers: Orange
Web frameworks speed up and simplify the web development process by providing developers with a set of high-level APIs that allow abstract access to underlying system resources and other low-level functionalities. In this chapter, we look at how web frameworks have risen or declined in popularity and we explore the profiles and technology use of the developers who use them.
The big picture of web framework usage

Born out of the necessity to provide consistent web experiences, frameworks revolutionised how web developers create our online world. With standardised approaches to development and framework-specific communities available for support, they are still a popular choice amongst web developers. According to our survey data, 60% of web developers use either client-side or server-side frameworks.

Indeed, web frameworks often present a time-saving option for web developers to do their jobs in simple and efficient ways. When turnaround times are tight, developing a website or a web app from the ground up may not necessarily be the best option, particularly when working with demanding clients. Indeed, when we look at deployment frequency, lead time, and time to restore service, framework users are more likely to be at the elite level of efficiency. 12% of web developers who use frameworks deploy their code on demand, 8% have a lead time of less than an hour, and 51% take less than a day to restore service. For web developers who don’t use frameworks, 9% deploy their code on demand, 5% have a lead time of less than an hour, and 43% take less than a day to restore service.

However, there are disadvantages to using frameworks. For example, it is unlikely that one framework will provide everything a web developer needs, just as it is unlikely that a web developer will use everything that a framework or a library has to offer. The proliferation of different frameworks with different standards and guidelines only further muddies these waters. In this case, it may be easier to stick to one or two frameworks and supplement their use with custom code, rather than creating a bloated working environment.

Web developers using frameworks are more likely to deploy code on demand, have smaller lead times, and are quicker to restore service.
Indeed, that’s what our data shows – the share of web developers using three or more frameworks is gradually dropping. Now, the average number of frameworks used is approximately the same for developers of all experience levels, around 3.2. However, usage habits change depending on specific years of experience – and may give an indication of what’s in store for the future of web development.

For example, those with less than five years of experience are more likely to use one framework (22%) than those with six years or more (17%). While age is not necessarily correlated with experience, we see a similar trend for those aged 24 and under (22% use one framework) compared to those 45 and above (17%).
We can take a look at specific frameworks to see who is using what. React is by far the most popular client-side library as it is used by 58% of web developers who use client-side web frameworks. React’s stable popularity as a library is contrasted by jQuery’s decline, which has experienced a 13 percentage point drop in usage in the past 12 months. Comparing the two, React is perhaps more capable at handling the modern web development environment – it’s faster, has a larger library of npm packages, and is efficient at creating larger web applications.

Who are the developers using jQuery? Looking closely, we find that those with more than 11 years of web development experience are 9 percentage points more likely to use jQuery than those with less than 10 years (49% vs 40%). Similarly, these seasoned web developers are less likely to use React (52%) compared to their peers (58%). As jQuery was created nearly a decade before React, it makes sense that those with more web development experience are sticking to what they know.

What about server-side frameworks? Next.js and Spring have been on the rise in the past 12 months. Next.js’ popularity is likely linked with React – after all, Next.js is a framework built upon React. In fact, 86% of those who use Next.js also use React; for comparison, only 45% use jQuery. As for Spring – a Java-based framework – its increase in usage may be explained by a similar increase in Java use amongst web developers – 8 percentage points in the past 12 months (27% vs 19%).
Spring and Next.js are the only frameworks to have grown in popularity in the past six months
% of web developers using server-side or client-side frameworks
(Q1 2021 n=3,246 | Q3 2021 n=3,747 | Q1 2022 n=3,525 | Q3 2022 n=3,753)

Client-side frameworks

Server-side frameworks

- React
- Bootstrap
- jQuery
- Vue.js
- Tailwind CSS
- Express
- Next.js
- Spring
- Laravel
- Django

(Q1 2021 n=3,246 | Q3 2021 n=3,747 | Q1 2022 n=3,525 | Q3 2022 n=3,753)
A framework user’s skillset

What does framework use mean for the technology use of web developers? If a web developer is forced to configure or write an application in a particular way, it may narrow their skillset as it forms a reliance on a particular architecture.

When we look at the web developer population, it seems that those who use frameworks are actually more likely to be self-driven, have technical skills, or use web technologies when compared to those that don’t use frameworks. Framework users are more likely to have learnt how to code through self-education (66% vs 55%) and are more likely to manually download packages from npm (45% vs 36%). Similarly, they are more likely to use each of the top-ten web development technologies listed in our survey.

Of these technologies, continuous integration/deployment (CI/CD) services are particularly worth mentioning – framework users are more than twice as likely to use these compared to non-framework users. As we’ve previously highlighted, framework users are more efficient when it comes to code deployment. However, framework users that also use CI/CD tools are 5 percentage points more likely to deploy on demand (15%) than framework users who don’t use CI/CD tools (10%).
Web framework users are more likely to use each of the top-ten web technologies in their development

% of web developers using each technology (Q3 2022 n=6,225)

- IDEs: 61% uses a web framework, 35% does not use a web framework
- Continuous integration/deployment services: 32% uses a web framework, 14% does not use a web framework
- Progressive web apps: 30% uses a web framework, 21% does not use a web framework
- Backend as a Service: 26% uses a web framework, 15% does not use a web framework
- Online payments: 25% uses a web framework, 23% does not use a web framework
- Push notifications: 20% uses a web framework, 16% does not use a web framework
- User analytics: 17% uses a web framework, 15% does not use a web framework
- Messaging platforms: 16% uses a web framework, 15% does not use a web framework
- Prototyping tools: 16% uses a web framework, 11% does not use a web framework
- App performance management: 11% uses a web framework, 11% does not use a web framework

Finally, while the share of web developers who use low-code or no-code tools has increased by 9 percentage points in the past six months (54%) – for framework users this share is 40%. This corresponds to an increase of only 5 percentage points in the same timeframe. In other words, those who are using frameworks are more likely to rely on old-fashioned coding by hand and have the skills to do so.
The 23rd edition of the Developer Nation survey reached 26,000+ respondents from 163 countries around the world. As such, the Developer Nation series of surveys continues to be the most global independent research on mobile, desktop, industrial IoT, consumer electronics, embedded, third-party app ecosystems, cloud, web, game, augmented and virtual reality, and machine learning developers and data scientists combined, ever conducted. The report is based on a large-scale, online developer survey designed, produced, and carried out by SlashData over a period of nine weeks between June and August 2022.

Respondents to the online survey came from 163 countries, including major app and machine learning development hotspots such as the US, China, India, Israel, the UK, and Russia, even stretching all the way to Kenya, Brazil, and Jordan. The geographic reach of this survey is truly reflective of the global scale of the developer economy. The online survey was translated into eight languages in addition to English, namely simplified Chinese, traditional Chinese, Spanish, Portuguese, Vietnamese, Russian, Japanese, and Korean, and was promoted by more than 80 leading community and media partners within the software development industry.

Our respondents came from a broad age spectrum, from young coders and creators who are under 18 to the seasoned ones over 55.

Respondents were asked which types of projects they are involved in out of the 13 under study, namely web apps / SaaS, mobile apps, desktop apps, backend services, augmented reality, virtual reality, games, data science, machine learning / artificial intelligence, industrial IoT, consumer electronics devices, embedded software, and apps/extensions for third-party app ecosystems. They also told us if they are into their areas of involvement as professionals, hobbyists, or students - or as any combination of these - and how many years of experience they have in each.

To eliminate the effect of regional sampling biases, we first weighted to correct for over-represented individual countries within regions. We then weighted the regional distribution across nine regions by a factor that was determined by the regional distribution and growth trends identified in our Developer Nation research. Each of the separate branches: mobile, desktop, industrial IoT, consumer electronics, embedded software, third-party app ecosystems, cloud, web, games, augmented and virtual reality, and data science and machine learning were weighted independently and then combined.

To minimise other important sampling biases across our outreach channels, we weighted the responses to derive a representative distribution for technologies used and developer segments. Using ensemble modelling methods, we derived a weighted distribution based on data from independent, representative channels, excluding the channels of our research partners, to eliminate sampling bias due to respondents who were recruited via these channels. Again, this was performed separately for each of mobile, industrial IoT, consumer electronics, embedded software, third-party app ecosystems, desktop, cloud, web, games, augmented and virtual reality, and data science and machine learning.

For more information on our methodology please visit https://www.slashdata.co/methodology https://www.youtube.com/watch?v=Yhum3ouMnsk
Are you curious about the future developer trends?
Let’s partner up!

- **Get insights into global developer trends**
  Understand how your audience stacks against the global average by inviting your developer community to participate in our research. Get more than 25 graphs with the key trends based on the data from our surveys, which will help you shape your developer outreach strategies.

- **Reach new developer audiences**
  Get your offering in front of our Developer Nation community with 70,000+ developers in 165 countries. Promote your products, tools, events, and developer programs via newsletter inclusions, goody bag sponsorship, content amplification on our blog, or brand placement in our free reports. Or contribute to our Developer Nation Community as a content partner and get to organise workshops, join our podcast, record how-to videos and many more.

LET’S TALK!
Contact us partnerships@slashdata.co