celential.ai

The Ultimate Guide To

Sourcing In-Demand Software Engineers

Backend

Frontend

Full-Stack

DevOps

Table of Contents

Introduction

Despite the uncertain times that we are living in, there will always be a need for developers. They are the backbones of the tech industry which now plays a role in almost every facet of life. And for the general population, coding might as well be a language from a foreign country that no one wants to visit.

That, of course, makes software engineers one of the most sought-after roles in the recruiting industry. Still, their specialized knowledge also limits the ability of any recruiter to source qualified candidates quickly. For starters, you need to know the difference between a coding language and a framework.

We have decided to address this challenge.

Our proprietary Talent Graph covers 5M+ engineering candidates in North and South America, and our CEO/Founder has 20+ years of experience leading and building engineering teams across companies. We leveraged this profound knowledge in the tech recruiting domain to create this ebook for talent acquisition professionals tasked with sourcing developers. It is a quick reference guide for any recruiter diving into sourcing technical roles or a technical recruiter who wants to be prepared before every interview.

With this ebook any recruiter will be able to effectively understand:

1. What is a Frontend, Backend, Full-Stack, or DevOps engineer
2. How to identify one based on their resume and skill set
3. Where are the most common sites to find them
4. How to send an outreach message that gets them to hit reply

Tech Glossary

Here is where it all begins. We have listed below four of the most in-demand software engineering roles and the languages, frameworks, and other skills they need.

A coding **language** is used to write a program, script, or other set of instructions for a computer to execute. **Frameworks** and **libraries** provide additional user-made components, tools and solutions to facilitate development with a particular language. For example, a software developer might use JavaScript with the framework Angular to create the front end of an application.

Of course, a software engineer will need expertise in many other technologies and **skills**. Here are common ones for the most in-demand developer roles.

Role	Language	Framework/Library	Other Skills
Frontend Engineer/Developer	CSS, HTML, JavaScript, TypeScript	Angular, React, Vue	AJAX, JSON, UI/UX
Backend Engineer/Developer	C, C++, C#, Go/Golang, Java, Node, PHP, Python, Ruby, Scala	ASP.NET, Django, Flask, Rails, Spring	API (REST or SOAP), Databases (MySQL, NoSQL, MongoDB, etc.), Git, SDK
Full-Stack Engineer/Developer	C, C++, C#, CSS, Go/Golang, HTML, Java, JavaScript, Node, PHP, Python, Ruby, Scala, TypeScript	Angular, ASP.NET, Django, Flask, Rails, React, Spring, Vue	AJAX, API, Databases, Git, JSON, SDK, UI/UX
DevOps Engineer	Bash, Go (or Golang), Perl, Python		Ansible, AWS, Azure, Chef, CI/CD, Docker, GCP, Git, Jenkins, Kubernetes, Puppet, Salt, Shell, Terraform

Position Definitions

Frontend Engineer/Developer

Frontend engineers/developers work on the part of web applications that you can see and experience in your browser. They can also specialize in websites, internal applications, desktop applications, and mobile. They are in charge of the way the product looks (User Interface), how users feel about using it (User Experience), web performance of the application, cross-browser and cross-device capability, and accessibility.

Popular Job Titles:

- **JavaScript Developer** – As JavaScript is the main language of frontend development, someone with JavaScript in their title is either working exclusively in frontend or is full-stack.
- **UI/UX Engineer** – It is a more specialized frontend role working with user interfaces that often comes with a background in design.
- **HTML/CSS Developer** – Someone working only with HTML and CSS likely does little to no coding and does not possess a computer science background.
- **WordPress Developer** – They create the front ends of websites but WordPress takes much work out of the process that this title is typically associated with small-scale freelance work.

Take me to the complete frontend glossary! ➡

Backend Engineer/Developer

Backend engineers/developers work on the part of web applications that is invisible to the user but supports the functioning of the application, such as the storage and retrieval of data, the architecture and infrastructure of an application, as well as its scalability and availability.

Popular Job Titles:

- **Python/C/C++/C#/Java Developer** – As these are the primary backend languages at the moment, someone with these titles is likely doing modern backend work.
- **Member of Technical Staff** – Some larger companies prefer this title for backend or full-stack roles; in some cases equivalent to a Staff role, though in others the track is from MoTS 1 to MoTS 2 and onward.
- **OS Developer** – Like UI/UX Engineers for frontend roles, Operating System developers use many of the same languages and skills as other backend developers, but have a more specialized skillset.
- **Machine Learning Engineer** – More likely to have the background in math and statistics that goes into Machine Learning. But an appreciable number of engineers flip flop between software development and machine learning.

Take me to the complete backend glossary! ➡

DevOps Engineer

DevOps is made up of “development” (i.e. the coding to build infrastructure that backend engineers do) and “operations” (i.e. ensuring that the production environment is running smoothly).

Popular Job Titles:

- **DevOps Engineer** – A DevOps generalist who could be doing any number of tasks to enable software development, including CI/CD, QA, Cloud Infrastructure, and security.
- **Cloud Engineer** – A subset of DevOps engineers responsible for creating and maintaining an organization’s cloud infrastructure.
- **DevSecOps Engineer** – Short for Developer Security Operations, DevSecOps engineers ensure security protocols remain in place throughout the development cycle.

- **Site Reliability Engineer (SRE)** - Though Site Reliability Engineering is not exactly the same as DevOps, there are many shared skills between the two roles, and many engineers who do one will do the other at some point in their career.
- **Systems Administrator** – A largely archaic title, many DevOps engineers who started their careers prior to the advent of cloud computing began as Systems Administrators.

Take me to the complete DevOps glossary! ➡

Full-Stack Engineer/Developer

True full-stack engineers/developers tend to be rare, as it is difficult to balance and maintain skills on both the backend and frontend. As such, full-stack engineers are often application developer generalists rather than specialists in specific areas such as Data or Machine Learning.

Titles shared between frontend, backend, and full-stack

- **Junior Software Engineer** – Many developers will only have this as their first title before getting promoted internally or moving to a new job; typically denotes 0-2 years of experience.
- **Senior Software Engineer** – Often developers achieve this title in about 3 years via internal promotion or some companies call all their experienced developers Senior, this job typically is associated with 5-7 years of experience.
- **Staff Software Engineer** – The next level up from Senior, and typically associated with 7+ years of experience. Often the highest level for individual contributors, though can come with managerial components as well.
- **Lead Software Engineer** – A role that often straddles individual contributor and management duties.

Teach me how to recruit full-stack engineers! ➡

Popular Specializations

A Frontend Engineer might specialize in:

- **Mobile** - applications for use on operating systems such as Android and iOS but in this case they are typically called mobile/UI engineers and will have more experience with graphics, design, and gaming.

- **Web** - applications, **Internal** applications, and **Desktop** applications and focus on the user interface and user experience.

A Backend Engineer might specialize in:

- **Applications** (Mobile or Web) - work on applications, or “apps”, programs designed to perform a specific function for an end user.

- **Data/ETL** - build systems, infrastructure, and “pipelines” for collecting, transferring, storing, processing, and accessing data.

- **Embedded** - work on low-level or “embedded” systems close to the physical hardware of a computer. For example, **Networking** engineers work on computer networks (routers, switches, and firewalls), and **Storage** engineers work on file systems.

- **Game** - develop video, computer or mobile games with systems such as Unity.

- **Infrastructure** - work on the foundational infrastructure (servers and networking) underlying all systems.

- **Machine Learning** - build infrastructure and models for Artificial Intelligence systems.

- **Platform** - work on the platform layer, in between the underlying infrastructure and the software on top.

- **Security** - protect data, electronic assets, and infrastructure from intentional or unintentional harm or misuse.

A DevOps Engineer might specialize in:

- **Build/Release** - work with developers to test and deploy software. They are also involved with version control, automation, and configuration management.

- **Cloud Architecture** - oversee the overall cloud computing strategy, converting technical requirements into the design, development, and management of cloud architecture and infrastructure.

- **Infrastructure** - work on overall infrastructure, including design, security, virtualization, containerization, automation, deployment, and reliability.

- **Quality Assurance/Testing Automation** - ensure that software releases are reliable and tested to be error-free before being released to users.

- **Security (DevSecOps)** - incorporate security controls in order to protect information systems from misuse, fraud, information theft, and damage.

- **System Administration/IT support** - work on the maintenance, configuration, and operation of computer systems and services, focusing on performance, security, and troubleshooting.

Resume Examples and Seach Tips

The average recruiter receives 250 resumes for an open position. That's a lot of reading!

A software engineer's resume may appear dense and jargon-filled at first glance. And unfortunately for recruiters, software engineers' titles don't always make a role or specialization explicit. For example, a Front End, DevOps, and Machine Learning engineer might all be titled "Software Engineer".

However, with the right strategies, it's more than possible to effectively evaluate candidates' skills and experiences from a resume or online profile.

We have prepared some examples to show you how to read resumes for our four in-demand software engineering roles and highlighted the key terms you need to take a closer look at.

Frontend Sample Resume

Robin Powell

Senior Software Engineer

*Creating intuitive, appealing **UI/UX** and scalable and performant frontend at Etsy*

Professional Experience:

Senior Software Engineer at Etsy (2018-)

- Frontend rendering infrastructure and web performance
- Scalable, user-facing applications for both mobile and desktop

Frontend/UI Developer at Ebay (2014-2018)

- Create and design responsive web layouts and reporting tools
- Used Backbone, Require, Grunt and jQuery to develop web interfaces for product purchasing pages

UI Developer at Paypal (2014-2015)

- Maintain, standardize and refactor existing products, APIs and web applications
- Coordinated with backend developers to create highly responsive and optimized **applications** for mobile devices

Education:

- Carnegie Mellon University: **BS in Computer Science** (2009-2013)
- Carnegie Mellon University: MS in Computer Science (2014)

Skills:

JavaScript, HTML, Java, CSS, Servlets, Front End, UI, API, UX, Android SDK, XML, AngularJS, React, Redux, RequireJS, GruntJS, Backbone.js, Underscore.js, Ember.js, JSON, jQuery, PHP

- Unfortunately for recruiters, Software Engineers' titles don't always reference an explicit role. However, you can successfully identify a Front End Engineer by looking for JavaScript frameworks and libraries (AngularJS, React.js, Vue.js, etc.).
- Keywords to identify a Front End Engineer include AJAX, Animations, Cross-browser, Graphics, HTML, JavaScript, JavaScript frameworks, JavaScript libraries, jQuery, Layouts, Mobile design, Responsive design, User-facing, UI, UX, Web Design.
- The most common Front End engineering titles include Software Engineer, Front End Engineer or Front End Developer.
- A Front End Engineer is distinct from a Web Developer. Front End Engineers have development skills and write the code for web applications. While Web Developer's use programs to generate the code for their websites.
- Not all software engineers have a Computer Science or related degree — others might have picked up programming skills from a bootcamp such as App Academy or Hack Reactor, or might be self-taught. However, a CS degree is typically preferred by most employers.

Backend Sample Resume

Amir Williams

Tech Lead at Stripe

*Amir Williams is an experienced Tech Lead with superpowers in both team leadership and backend engineering. Currently leading the development of complex backend service architecture at **Stripe**.*

Professional Experience:

Tech Lead at Stripe (2018-)

- Reduced paging volume and tech debt by consolidating multiple identity verification services into one
- Contributed to solving scaling issues by helping build out a new payments search service — breaking the legacy monolithic service into scalable, available and well-monitored **microservices**

Senior Software Engineer at PayPal (2014-2018)

- Designed, implemented, and grew an intelligent shopping experience backend serving over 100 million requests per day
- Co-designed PayPal's first real-time Kafka streaming application

Software Engineer at LinkedIn (2010-2014)

- Backend system design to ensure high stability, performance, and reliability
- Building new features on LinkedIn's Service Oriented Architecture

Education:

- University of California, Los Angeles - BS in Computer Science (2006-2010)

Skills:

Ruby, Git, MyQSL, PHP, C++, **HTML, CSS**, Java, JavaScript, Node.js, Python, **Django**, Flask, PostgreSQL, Apache Kafka, MongoDB, Linux, Unix, Pandas, APIs, Express.js,

At a small startup, a back end engineer is likely to be a generalist who will wear many different hats. However, at a large company like Stripe, back end engineers typically specialize in a particular area such as infrastructure, data, or networking.

Some keywords to look for are: API, business logic, data access, database, distributed systems, architecture, infrastructure, microservices, platform, server side, services, scalability.

Don't be thrown off by the presence of frontend skills — backend engineers frequently require a basic understanding of skills such as HTML, CSS, and JavaScript.

One way to recognize Back End Engineers is by the Back End frameworks they use. Some of the most popular today are Django (Python), Flask (Python), Spring (Java), Node (JavaScript), Rails (Ruby) and .Net (C#).

Full-Stack Sample Resume

Amy Moore

Staff Engineer at Precisionly.ai

I am a full-stack software developer with an emphasis on modern JavaScript frameworks — specializing in Java for developing back end and building RESTful API endpoints and React for building visually appealing and user-focused front end.

Professional Experience:

Staff Engineer at Precisionly.ai (2020-present)

- Led frontend team in migrating the platform from Angular to React
- Implemented enterprise service architecture which can be split into microservices if needed

Staff Engineer at Galvanize (Feb 2020 - Sep 2020)

- Responsible for the entire revenue optimization frontend.
- Maintain and improve on Java services

Senior Engineer at Galvanize (2018-2020)

- 3rd party authentication integration, various visualization tools as well as app infrastructure
- Resolution of identified security vulnerabilities, conducting code reviews for team, streamlining and improving SDLC practices

Education:

- North Carolina State University - BS in Computer Science (2009-2013)

Skills:

JavaScript, HTML, CSS, Java, Python, Angular, React, JavaScript, APIs, Microservices, Docker, Kubernetes, Node, Express, Apache Spark, C, C++, C#.Net, MySQL, MongoDB, Microsoft SQL Server, Git, Redis, RESTful API

- At the Senior level, a recruiter would expect a candidate to demonstrate ownership of a feature or product from end to end, including design, planning, troubleshooting and implementation. Senior engineers are also typically expected to participate in leading projects, interview processes and mentorship of junior developers.

  From the Senior Level, an engineer might be promoted to Engineering Manager. Otherwise, Principal, Staff, Tech Lead and Architect are all individual contributor roles, with ownership of increasingly complex problems and systems, as well as an impact on repeatable processes, overall engineering culture and the direction of the team.

- Full Stack Engineers are recognizable for their mix of frontend and backend skills.

DevOps Sample Resume

Adam Roy

Senior Engineer at Zonova

Hi, I'm Adam. I'm a DevOps Engineer at Cloudery.io, working on our cloud strategy, cloud infrastructure, and automation frameworks.

Professional Experience:

Senior Devops Engineer at Cloudery.io (2017-)

- **Deploying, managing, maintaining, scaling, and monitoring** multi-tenant SaaS applications on Azure
- Manage operational databases, primarily focusing on Couchbase and Elastic, but also transitioning data from Mongo and Redis
- Troubleshoot live production issues, perform root cause analysis and script remediation

Software Engineer at AveriTech (2014-2017)

- Design **REST APIs** for Virtual Machine setups
- Configure and Maintain DNS and Proxy Servers on Linux
- Executed test cases on bug fixes for storage addition, multitenant deployment, software updates, FedRamp images, and new features.

Site Reliability Engineer at Cisco Meraki (2013-2014)

- Built tooling for developers to safely test and deploy code
- Assisted the development and deployment of JVM, NodeJS and Python applications while maintaining availability
- Automation of configuration, logging, monitoring and DNS to help reduce provisioning times

Education:

- UC Irvine, B.S. in Computer Engineering (2009-2013)

Skills:

Linux, C, C++, MySQL, Jenkins, DNS, PHP, Developer Tools, Verilog, Unix, Python, Ansible, Docker, Kubernetes, Java, RESTful API, Containerization, Virtual Machines, Bash Scripting, Azure, AWS, Hadoop, Apache Airflow, Redis, Mongo, Couchbase, ElasticSearch, Terraform, Node

Keywords to identify a DevOps engineer are Automation, Availability, Build & Release, Cloud Architecture, Configuration, Containers, Continuous Integration Continuous Deployment (CI/CD), DevSecOps, DNS, Logging, Monitoring, Infrastructure, Scripting, Security, Site Reliability, Quality Assurance (QA), Testing, Tooling and Virtual Machine.

You will likely also find many Backend keywords on a DevOps Engineer's profile.

There is a good deal of similarity and overlap between DevOps and Site Reliability Engineering, and different organizations may mean different things by each title. It is typically necessary to read the job description carefully to compare with the elements on a DevOps engineer's resume.

Where to Source Developer Candidates Outside of LinkedIn

01 Upwork

It's a freelance platform that allows candidates new and experienced alike to compete for jobs, so expect to have to sift a little bit to hire developers up to the technical bar a startup demands.

02 Toptal

Only more seasoned freelance developers will show up on Toptal. It's definitely a faster way to search for candidates with the right skills than Upwork, but you can expect that search to return more expensive hourly rates too.

03 AngelList

Half of the battle in hiring a developer full-time for your startup involves finding people willing to work at a startup, and AngelList solves this problem by being the platform for people seeking startup jobs. That said, it comes with the same challenges as other job boards: you're going to have to do some filtering.

04 GitHub

GitHub's name is wonderfully intuitive—it's a hub for maintaining code via Git, so to understand how to find candidates on GitHub, you need to understand Git.

Git creates forking branches through versions of a project, preserving older versions of code in case they're useful later. It also allows team leaders to see the whole picture, only merging pieces of code to the main code base once everyone's sure that the merge won't break something.

GitHub is just (by far) the most popular place to store that code and to run Git commands on it.

05 Stack Overflow

Stack Overflow is a Q&A board for technical information and theoretical discussions about coding.

Operating since 2008 with 14 million registered users who have asked 21 million questions, it serves as an encyclopedic repository of knowledge for programmers. Common issues related to a wide variety of technical topics and complex problems have been addressed on the site.

- GitHub

How to Find Software Engineering Candidates on GitHub

STEP #1: Create an account

STEP #2: Compose an Advanced Search

For example, if you'd like to find a software engineer living in San Francisco who programs in Java or Ruby with more than 15 followers, you might use the following string:

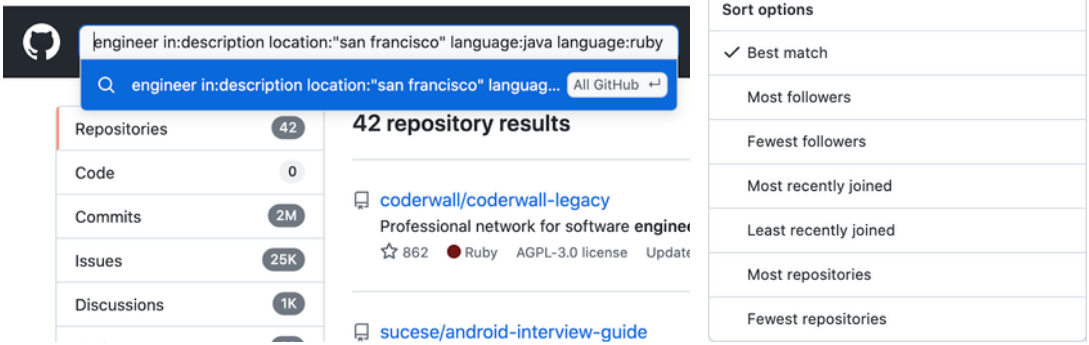
engineer in:description location:"San Francisco" language:java language:ruby followers:>15

STEP #3: Sort Profiles

GitHub search results return repositories by default. Click on "users" at the bottom of the left sidebar to view profiles that fit the search criteria.

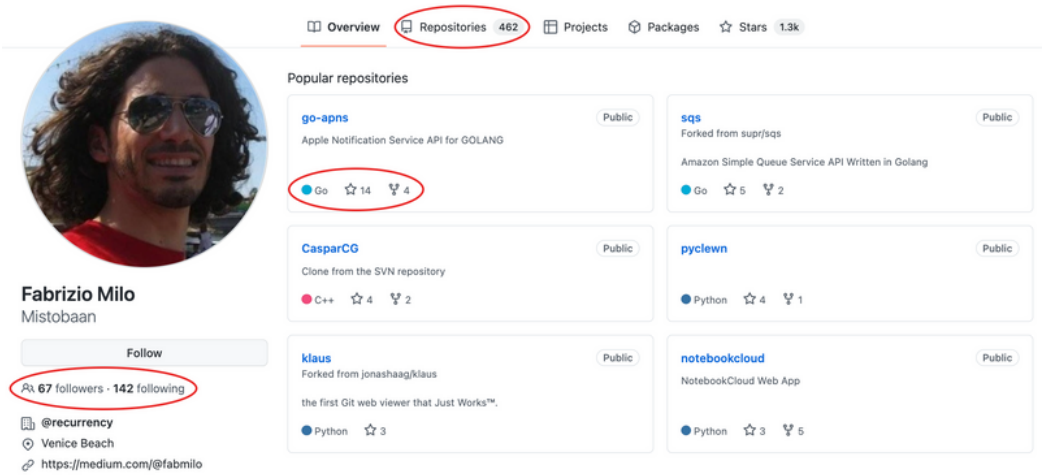
STEP #4: Evaluate Users and record profiles

GitHub doesn't currently offer a way to record and track candidates, so upon evaluating each profile, be sure to note down important information such as name and email addresses. Not every user includes their email address on their profile, but many unlisted addresses can be located manually.



How to Evaluate Profiles on GitHub

A quick way to gauge the popularity of a contributor is to scan the stats under their username.



- **Followers** (67 is impressive on GitHub; even 10 followers would be eye-catching)
- **Stars** (analogous to Facebook likes) can also help you gauge how many meaningful and interesting contributions they've made.
- **Repositories** (open-source coding projects) can indicate the languages and skills a developer has demonstrated use of, as well as breadth of experience.
- **The number of times a repository has been starred or forked** (i.e. “copied”, allowing another user to make their own changes and experiment freely without touching the original code) can indicate the scale of the project and how valuable or interesting others find it.

- Github

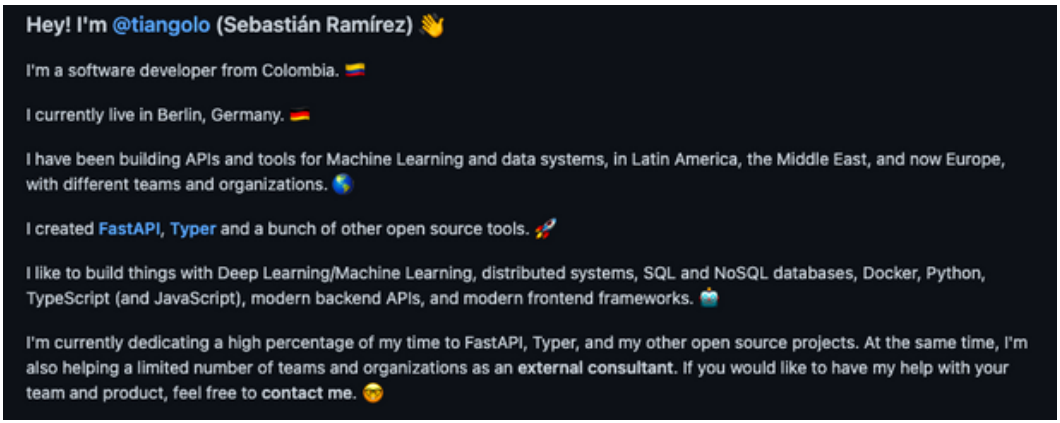
How to Determine the Potential of Candidates via Their Repositories

There's no barrier to entry on GitHub, so we wanted to provide a few criteria for what should be considered an impressive unforked repository, and what kind of forked repositories might be a good sign, check out our blog "Sourcing on Github in 2022" for the full rundown. **Here are the questions you need to answer:**

Unforked repositories

Does this project address a need? - If the developer explains a specific, sophisticated problem they're addressing in their about page, then you can count on them to understand how to translate business needs into software solutions.

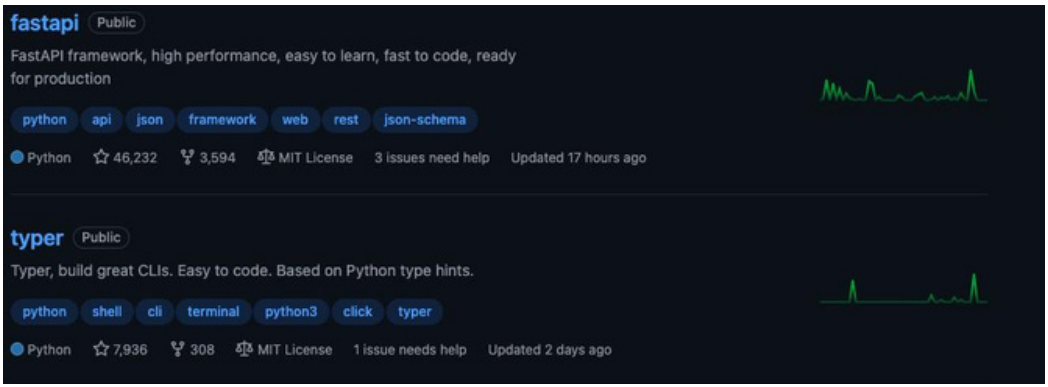
Can this project scale? - Fresh graduates and junior developers don't typically need to think about how their work will perform if a ton of people use it. An about page that explains how this code has been written to run efficiently can show that its author is attuned to these challenges.



Forked repositories

Would this developer be working with this tool in the role I have for them? - Developers won't necessarily put every little piece of technology that they work with on their LinkedIn. But if a developer has forked the source code for FastAPI, you know that they're using FastAPI.

Is there a pattern to this profile's forked repositories? - You can piece together what someone's been working on based on what they've forked. For example, someone who has forked from Docker and Kubernetes is most likely working in DevOps managing microservices.



Projects that shouldn't be taken as a strong positive signal of technical prowess:

- Solutions to data structures and algorithms problems.
- "Clones" of major websites like Facebook or Twitter.
- To-do list apps, simple games, and personal websites.

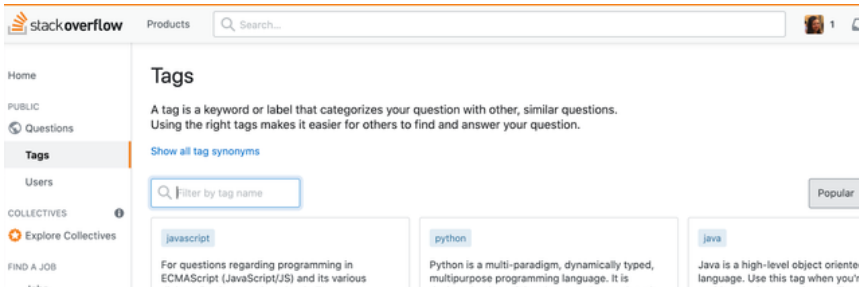
How to Find Profiles on Stack Overflow (For Free)

There are multiple ways to search for candidates on Stack Overflow, including a paid option for recruiters, but we have decided to show you how to do it here for free. **Check out our full guide on sourcing candidates on Stack Overflow**.

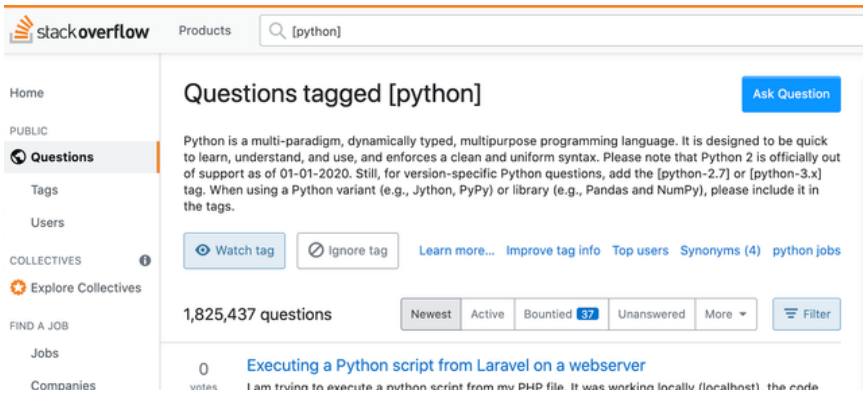
Tags

One option to search Stack Overflow for free involves tags.

1. Click on “Tags” on the left-hand sidebar. Then click on the desired language, skill, or domain, such as “python”.



2. Click on “Top users” to show user profiles. The results are limited to the 20 top users all-time and from the past 30 days, so the initial search might seem pretty limited.



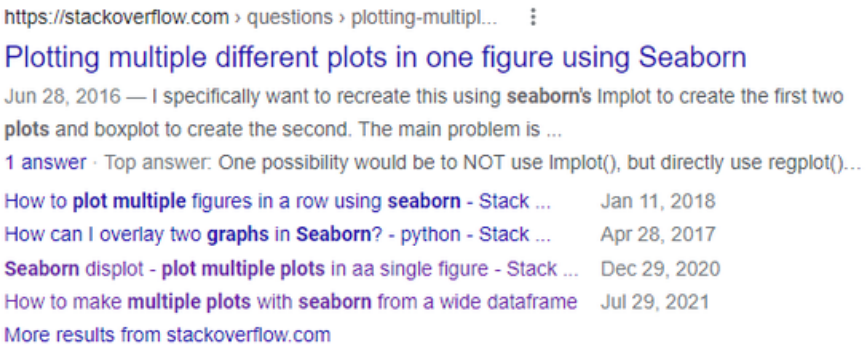
Searching for candidates on Stack Overflow with Google

If you're location agnostic, you can pretty easily find engineers just by running a Google search for the kinds of problems the people you're recruiting need to be able to solve.

For example, let's say you're hiring a Data Scientist whose day-to-day will involve producing data visualizations in Python with the Seaborn library.

1. You ask the hiring manager for the role what kinds of things an experienced professional should be able to do in Seaborn on the job with no issue. Among them, you learn that anyone reasonably familiar with the tech should know how to display multiple plots in one figure.

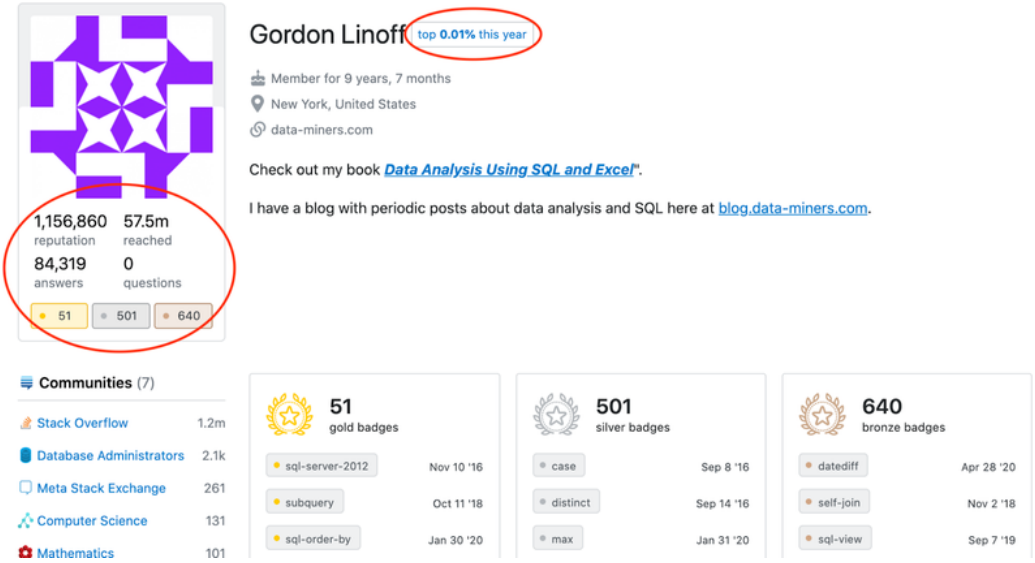
2. So you search "Seaborn display multiple plots" on Google and get this:



3. Didn't even have to include "Stack Overflow" in the search! And indeed, you see that somebody has asked this question annually on Stack Overflow since 2017. If you check out the reply from 2021, you'll find a user who not only codes out the solution for the asker but provides a deep dive into the technical documentation of Seaborn.

- Stack Overflow

How to Evaluate Profiles on Stack Overflow



Here are some data points to consider while you search for and evaluate user profiles to find candidates.

1. **Ranking** gives you an idea of how active the user has been.

2. **Reputation** is awarded for upvoted questions and answers (or lost by downvoted ones), serving as a metric of how trusted a member is by the Stack Overflow community.

3. **Badges** are awarded for being helpful (or for meeting sometimes obscure combinations of various criteria).

Great Answer	Answer score of 100 or more	100.4k awarded
Populist	Highest scoring answer that outscored an accepted answer with score of more than 10 by more than 2x	25.9k awarded
Revival	Answer more than 30 days after a question was asked as first answer scoring 2 or more	505.3k awarded
Necromancer	Answer a question more than 60 days later with score of 5 or more	783.9k awarded

4. **Top tags** reveal what a user most frequently discusses on the site. To the right of each tag is a score (number of upvotes minus number of downvotes in the user's tagged answers), total number of tagged posts, and the percentage of the users' posts that relate to the tag. Tags can help you infer what kind of engineering role the user has.

This gets more and more important if you're not only sourcing developers for general roles like Machine Learning or web development, but for specific skills within those roles like data visualization or API integrations.

Caveat: Stack Overflow users are judged for their ability to communicate theoretical knowledge and expertise of a certain technology rather than the strength of their coding. While these skills are correlated and ideally the candidate would have both, the distinction may be important when sourcing for a hands-on programming role.

How to Conduct Outreach for Software Engineers

3.4%

With only 3.4% of software engineers currently looking for a job, pursuing passive talent is vital to successfully filling an open position.

Persuading candidates to reply to your cold recruiting emails can feel almost impossible — let alone perennially in-demand full stack software engineers who are already happy at their current position.

However, we've found that it's possible to boost your reply rate to 30% — even for those elusive full stack engineers.

How do you do it?

By telling developers a personalized story about why they should work with you.

1. **Explain how this opportunity fits into a candidate's career trajectory** and what your company can offer them in terms of culture, values, and personal growth
2. **Reference the candidate's domain experience, technical skills, work history, and other background details**. And don't forget to personalize your subject line — this can lead to up to 50% better results.
3. **Mention common backgrounds** such as education, career history, geographic location, and mutual connections
4. **Send 2-3 follow-ups to increase your response rate** — according to our own company recruitment data, two-thirds of replies come from follow-ups.

To make it easy, here is a **hyper-personalized template** we've put together that we've used to source and hire real engineering candidates.

---------- *Subject line* ----------

[Candidate first name,] come join the eng team at Celential (early-stage/AI/recruiting/ remote-friendly)

----------*Personalized greeting* ----------

Hi [Candidate first name,]

I came across your profile while looking for exceptional engineers to join our team. I understand you're probably happy at [Company name], but figured that with your [years worked] coming up you might be wondering what other new and exciting opportunities are out there.

---------- *Company quick introduction* ----------

Celential.ai has developed a talent graph of all product and engineering talent in the US, combined with a matching algorithm to identify the best candidates.

We then use NLP AI to engage these candidates in a highly personalized way automatically. In fact, you're experiencing the platform right now.

---------- *Candidate role match* ----------

I was really impressed with your strong startup experience at [Company name] and your previous ML and engineering accomplishments at [Company name] and [Company name]. Also your stellar education at [University name] and expertise in Python, Deep Learning. NLP, Scikit-Learn and TensorFlow, made me think there would be a good mutual fit. There is a lot of common background between you and our team. Some of our team members have worked at [Company name] in the past too.

---------- *Opportunity pitch & mutual background* ----------

Our team is founded by ex-[Founders' company name] executives, alums from top schools like [University name] and your alma mater [University name], as well as ex-Google engineers who are experts in delivering data/ML-driven products. Our company is in hypergrowth with dozens of happy annual customers including [Customer] and [Customer].

---------- *Call to action* ----------

I've attached a job description. I'd really appreciate it if you could find some time for a quick call with us! Looking forward to hearing from you.

How You Can Automate Your Software Engineering Sourcing Process

One thing to take into consideration is that you're unlikely to be able to consistently use just Stack Overflow to search for software engineer candidates with a high degree of precision. It's only one of many online resources that can be pulled together to establish a comprehensive profile for developers.

Sourcing through GitHub can also be a challenge without a technical background. It doesn't replace a resume, and often needs to be cross-checked with other sources to get a full picture of a candidate's experience.

To top things off, the total cost for hiring a Full-Stack Developer can be around $30,000 or more in 2022.

Any reduction an employer can get in hiring costs is a huge advantage especially during cost-saving times when talent acquisition department budgets are thrown out the window and their teams are understaffed.

AI-based solutions can help companies slash time and cost by:

1. **Streamlining their recruitment processes**
2. **Providing access to vast amounts of data and extensive talent networks**
3. **Hyper-personalizing candidate outreach and engagement at scale.**

Our AI recruiting solution pulls together hundreds of data points on developers and composes personal outreach at scale for your sourcing efforts. We deliver high-quality and engaged talent with zero effort or learning curve on your part — freeing your talent acquisition team to build and nurture candidate relationships and make top-quality hires.

We offer:

80%

Present-to-acceptance Ratio

Hiring teams accept 80% of the candidates we submit for interviews.

3 day

Average turnaround

You receive warm talent ready for interviews in 1-5 business days.

50%

Cost saving

Our 'Virtual Recruiter' helps clients save 50% recruiting costs on average compared to other hiring methods.

With concerns about an economic slowdown, record-setting inflation, and VCs slowing down investing, 2022's hiring market involves a lot of uncertainty and lower recruiting budgets for tech companies.

Try Celential.ai today and get the flexibility to scale up and down your recruiting efforts instantaneously as your hiring needs shift.

Tech Glossary Appendix (Full Term Definitions)

Frontend Definitions

Frontend Languages, Frameworks and Skills

- **AJAX (Asynchronous JavaScript and XML)** is a set of web development techniques used to create asynchronous applications. With AJAX, webpages can send and receive information from the server without reloading the full page — allowing for dynamically updated content such as a map or other object.

- **Angular** is a TypeScript-based free and open-source framework for building web applications, created and maintained by Google.

- **CSS (Cascading Style Sheets)** is a style language used to express the presentation and appearance of a document written in a markup language such as HTML. CSS tells the browser which HTML tags to style and how they should be formatted in terms of color, font, layout, etc.

- **JavaScript** is a high-level, object oriented, just-in-time compiled language used as the scripting language for almost all webpages. JavaScript is used for interactive elements on a webpage such as drop down menus or animated images.

- **JSON (JavaScript Object Notation)** is an open standard format for storing and transmitting data. JSON stores information as arrays of name-value pairs in human readable format.

- **React** is a JavaScript-based free and open-source library for building user interfaces, created and maintained by Facebook. Currently, React is the most popular library/framework for frontend development.

- **TypeScript** is a strongly typed programming language that builds on JavaScript. Typically, TypeScript is preferred for large-scale web applications and JavaScript is used to develop interactive webpages.

- **UI (User Interface)** is the space of interaction between humans and machines. User interfaces are designed to be maximally intuitive and visually appealing.

- **UX (User Experience)** focuses on the usability and overall experience of the user of a product.

- **Vue** is a JavaScript-based open-source framework for building user interfaces which uses a model-view-viewmodel architectural pattern.

- Appendix II

Backend Definitions

Backend Languages, Frameworks and Skills

- **API (Application Programming Interface)** is a connection between software programs. The term API can refer to either (1) the specification of how software components should interact with each other, or the (2) implementation of the API's service

- **REST (Representational State Transfer) API** is a software architectural style designed to make the Web more scalable. This is the predominant web API model.

- **SOAP (Simple Object Access Protocol) API** is a messaging protocol specification for exchanging structured data in the implementation of web services.

- **ASP.NET** is a web framework (part of the .NET framework) used for building dynamic web applications. It is frequently used with Microsoft products such as C#.

- **C** is one of the original compiled, general-purpose programming languages. It is still used widely today in mostly low-level applications such as operating systems and hardware drivers.

- **C++** is an object-oriented, compiled, general-purpose programming language designed as an improvement on **C**. Though its popularity has declined somewhat, it is still used in applications that require high performance such as servers and search systems.

- **C#** is an object-oriented, compiled programming language designed as an improvement on **C++**. Developed by Microsoft in 2000, it is used with Windows applications, in many large enterprises, and by gaming companies.

- **Django** is an open-source Python web application framework.

- **Flask** is an open-source Python web application framework.

- **Go (or Golang)** is a statically typed, compiled programming language designed by Google.

- **Java** is an object-oriented, compiled, general purpose programming language developed by Oracle. It is often used in enterprise software.

- **MySQL** is an open source relational database management system, and the most widely used open source SQL database. It is currently owned by Oracle.

- **Node** is an open source, cross-platform, runtime environment designed to run JavaScript as a server-side language. Node has an event-driven architecture which enhances scalability to concurrent users.

- **NoSQL (not only SQL)** is a database for storing and managing data on a large scale. Popular examples include MongoDB, CouchDB, CouchBase, Cassandra, and Redis

- **PHP** is an open source server-side scripting language designed for web development. Though it was popular in the 1990s and early 2000s, today it is considered to be somewhat outdated.

- **Python** is an open-source interpreted, object-oriented, general-purpose programming language. It is currently the most popular programming language and used widely at both tech companies and in scientific research.

- **Ruby** is an open-source interpreted, object-oriented, general-purpose programming language designed for fast development time. Though it peaked in popularity around 2008, it is still used by many tech companies and startups. Ruby is paired with the **Rails** framework (**Ruby on Rails**).

- **Scala** is a statically typed, general purpose programming language used for both object oriented and functional programming. It is interoperable with **Java** and intended to be compiled on a Java Virtual Machine.

- **SDK (Software Development Kit)** is a collection of software development tools provided by the manufacturer of an operating system, hardware platform or language that allow for the creation of applications.

- **Spring** is the most popular open source framework for Java.

DevOps Definitions

DevOps Skills (In addition to backend languages/frameworks)

- **Ansible** is an open source IaaS (infrastructure as a service) automation tool for cloud provisioning, configuration management and application deployment. It is similar to Chef, Puppet and Salt.
- **AWS (Amazon Web Services)** is an on demand cloud computing platform operated by Amazon. Currently, it is the most popular cloud computing web services company.
- **Azure** is a cloud computing platform operated by Microsoft. Currently, it is the second most popular cloud computing web services company.
- **Chef** is a commercial tool for configuring, managing and deploying cloud infrastructure. It is similar to Ansible, Puppet and Salt.
- **CI/CD (Continuous Integration/Continuous Delivery)** is software development practice of rapidly building, testing and releasing software to users by frequently and incrementally introducing changes. Automation and continuous monitoring ensure that code is released quickly and reliably.
- **Docker** is a PaaS (platform as a service) product for building, running and managing containers on the cloud.
- **GCP (Google Cloud Platform)** is a cloud computing platform. Currently, it is the third most popular cloud computing web services company.

- **Git** is software used for distributed revision control and tracking and managing source code changes. It places an emphasis on speed and data integrity, allowing developers to track revisions and history and coordinate their work on a coding project. **GitHub, BitBucket, GitLab** and **SourceBucket** offer Git repositories as a service.

- **Jenkins** is an open source automation software which enables developers to build, test and deploy software, facilitating **CI/CD**.

- **Kubernetes** is an open-source system for automating the deployment, scaling, and management of containerized applications.

- **Puppet** is an automation and configuration management tool similar to Ansible, Chef and Salt.

- **Salt** is open source software used for event-driven IT automation and configuration management. It is similar to Ansible, Chef and Puppet.

- **Terraform** is an open source IaaS automation tool for cloud provisioning, configuration management and service orchestration.