



## Career Advising @ ACM

# Glossary of Tech Roles

**Friendly ACM Asterisk\*\*\*:** These descriptions include ideal skills broken loosely into “important,” “important depending on application,” and “nice to have” in descending order of priority. You definitely do not need to have all or even most of them to be well qualified for positions, especially as an intern and different companies will also have different expectations and preferences — this is more so that you have a broader idea of what kind of technologies and skills you may be working with or may want to pick up if you go into these roles :)

**Note:** Timeline dates reflect summer internship positions, new grad should be similar but full-time recruiting may happen at different times. Companies often also recruit interns during the fall and spring and sometimes in the winter — this varies on a per-company basis, but students usually will take semesters off in order to do these internships. Occasionally companies may do micro-internships that can be done entirely during winter break, but this is fairly rare. Note, however, that Princeton organizes Princeternship, which is essentially a list of winter internships that take place entirely over winter break that are hosted by companies founded or heavily connected with alumni. Great opportunity, especially for underclassmen to get their first internship, or just for anyone interested in getting experience in possibly a new field!

**Note:** Often interviewers will assess your technical skills to the extent that you claim proficiency and knowledge of them in your application and resume: thus, including those skills can be a double-edged sword. You may appear more desirable if you have certain skills, but you may then be asked to demonstrate them through technical questions or problems in interviews, so make sure to take care!

## SOFTWARE

- Often begins recruiting August-November, interviews may continue through the winter and some companies (e.g. Amazon) may still recruit in the spring
- Some first- and second-year programs (e.g. Google STEP) may recruit later in the year, ostensibly because they want candidates to complete their first semester of classes before interviewing
- Interviews generally consist of coding/algorithms, system design, occasionally behavioral, i.e. talking about resume, experiences, challenges, etc.

## Software Engineer (SWE)

- **Designs, implements, tests, productionizes, and maintains software — possibly for a product, database, or application**
- Important
  - One of Java, C, C++
  - Version command, i.e. git
- Important depending on application
  - Possibly Python, Node.js/Javascript, SQL depending on the domain
- Nice to have
  - Experience with systems, architecture, linux, databases, networks is useful but not always required

(Note: the above SWE role is a blanket, generalized role; sometimes larger companies will divide this into subroles with more specialized responsibilities, expounded on below)

## Software Engineer (SWE) — Backend

- **Develops the server which provides data on request, the application which channels it, and the database which organizes the information**
- Important
  - One of Java, C, C++ or C# (important for interviews and for the job itself)
- Important depending on application
  - Node.js
  - Database technologies (SQL, MySQL, Oracle, Postgres, or their NoSQL variants: Redis, Cassandra, DynamoDB, MongoDB)
- Nice to have
  - Ruby or Sinatra, Python, PHP (Typically more popular with front-end or middle-tier engineers, and often popular among self-taught engineers)
  - Experience with APIs
  - System experience – system administration, Linux, cloud computing services (AWS, Rackspace, OpenStack)

## Software Engineer (SWE) — Frontend

- **Front-end developers design and construct the user experience elements on the web page or app including buttons, menus, pages, links, graphics and more.**
- Important
  - HTML, CSS
  - Javascript (Including JavaScript frameworks like Ember.js, Angular.js, Backbone, etc.)
- Important depending on application
  - Ruby on Rails, Python or Django, PHP, ASP
- Nice to have

- Tools like Figma, Photoshop

(Note: more commonly, companies will merge frontend and backend responsibilities into a single Full Stack role, especially on a smaller project, e.g. in the context of Web Development)

### **Software Engineer (SWE) — Mobile/App/iOS/Android**

- **A flavor of software engineering that focuses on developing for and deploying on mobile devices, specifically on the iOS or Android platforms. Some roles work cross-platform using tools like Flutter, but this often requires knowledge of tools relating to both. Also has front-end and back-end components.**
- Important
  - iOS: Swift or Objective-C
  - Android: Java, C++, or C#
- Important depending on application
  - Have a good grasp on build views; can handle user interaction; can display data; show alerts; handle navigation; display images, labels and text views; and integrate with REST APIs and parse JSON.

(Note: iOS vs. Android, some insight from a development/manager perspective)

### **Software Engineer (SWE) — Web**

- **A flavor of software engineering that focuses on developing for and deploying on the Internet using a client-server model. Also has front-end and back-end components.**
- Important
  - Experience in Java, C++, or C#, object-oriented design
  - Architecture and design (architecture, design patterns, reliability and scaling) of new and current systems
  - HTML5, CSS, JavaScript, Responsive Design, web services, etc.
- Important depending on application
  - Experience developing AJAX interfaces with AJAX libraries and frameworks (e.g. ReactJS, JQuery, AngularJS, etc.)
  - Experience with MVC frameworks
  - Understanding of web services technologies such as REST, SOAP, HTTP, JSON
  - Ability to work in a fast-paced and agile development environment and to learn new frameworks/stacks

(Note: in particular in web development, but also generally, frameworks and languages are always changing, so it is always more important to have a solid background in object-oriented programming, algorithms, systems, networks, design, etc. and to be able to pick up new syntax quickly than to know a particular technology)

## SYSTEMS

- *Similar recruiting timeline as software, i.e. primarily August-November, and a few in winter and spring*
- *Interviews generally consist of coding/algorithms, system design, occasionally behavioral, i.e. talking about resume, experiences, challenges, etc.*

### Database Administrator (DBA)

- **Uses software to store and organize data, such as financial information and customer shipping records. They make sure that data is available to users and is secure from unauthorized access.**
- Important
  - Ability to craft and optimize SQL queries
  - Designing data schemas
  - Experience with one or more of the major relational databases: SQL Server, MySQL, Oracle
- Important depending on application
  - Database configuration and administration experience

### DevOps Engineer

- **Introduces processes, tools, and methodologies to balance needs throughout the software development life cycle, from coding and deployment, to maintenance and updates.**
- Important
  - Basic scripting experience in languages such as Python, Perl, Ruby or shell scripting (aka Bash or Unix), potentially other languages such as Java, C as well
  - SQL knowledge
- Important depending on application
  - Experience with one or more cloud service providers, such as AWS, Rackspace, OpenStack, etc.
  - Operational best practices for deployments, releases and maintenance
- Nice to have
  - Experience with monitoring tools such as Nagios, Munin, Splunk, Zabbix, etc.

### System Administrator

- **Knowledgeable in both hardware and software – configuring resilient, secure architecture to ensure the success of the business. System administrators are normally tasked with the installation, maintenance, configuration and repair for servers, networks and other computer systems**
- Important
  - Unix, Linux or Microsoft system administration
  - Basic scripting or programming experience (shell scripting, Perl, Python, Ruby, etc.)

- Important depending on application
  - Operational best practices for releases and maintenance
  - Experience with internal IT, such as VPNs, printers, IP-based phones, tech support, etc.
  - Experience setting up and backing up servers and databases
- Nice to have
  - Experience with monitoring tools such as Nagios, Munin, Splunk, Zabbix, etc.

(Note: there is a difference between DevOps vs System Admin)

### Network Administrator

- **Similar to System Administrator but focusing on networks, including local area networks (LANs), wide area networks (WANs), network segments, intranets, and other data communication systems.**
- Important
  - Experience setting up networks (corporate networks, data centers, etc.)
  - Knowledge of switches, routers, firewalls and other networking hardware

(Note: network admin likely has some overlap with system administrator skills, particularly the system administration pieces)

## DATA

- ➔ *Similar recruiting timeline as software and systems*
- ➔ *Database interviews involving SQL queries, sometimes machine learning interviews, sometimes coding and system design interviews*

### Data Scientist

- **Solves problems by generating insights from large-scale datasets using tools from math, statistics, computer science, and machine learning.**
- Important
  - Strong statistics knowledge and experience, potentially other advanced math as well
  - Ability to construct SQL queries
  - Some programming experience likely in Python, Matlab, R, Java, Ruby, or others
  - Potentially specialized machine learning or data mining skills

(Note: some insight from a recruiting perspective)

### Data Engineer

- **Develops, constructs, tests and maintains architectures, such as databases and large-scale processing systems.**

- Important
  - SQL
  - Understanding of ETL and data modeling
  - Build scalable data infrastructure and understand distributed systems concepts from a data storage and compute perspective
  - Be proficient in at least one scripting/programming language to handle large volume data processing
- Important depending on application
  - Build different types of data warehousing layers based on specific use cases
  - Ensure the accuracy and availability of data to customers and understand how technical decisions can impact their business's analytics and reporting

(Note: [Data Scientist vs. Data Engineer](#))

## TESTING

### Quality Assurance Engineer (QA), Software Engineer in Test (SET/SEDET)

- **Monitors every phase of the software development process so as to ensure design quality, making sure that the software adheres to the standards set by the development company. SDETs often write more software, although this may be changing.**
- Important
  - Ability to write test plans and test cases, know best practices for testing (For example, black box testing, white box testing, unit tests, smoke tests, regression tests, etc.)
- Important depending on application
  - Best practices for software deployments, releases
  - Object-oriented programming language like Java, C, C++
- Nice to have
  - Selenium, or other test automation tools
  - At least one REST API testing tool like SoapUI, Postman or others
  - Knowledge on Modbus, BACNet, MQTT protocols
  - SQL

### Site Reliability Engineer (SRE)

- **“To protect, provide for, and progress the software and systems [...] with an ever-watchful eye on their availability, latency, performance, and capacity.” - [Google](#)**
- Important
  - Proficiency working with Amazon Web Services (AWS) (e.g. EMR, Glue, Lambda, EC2, EBS, ELB, S3, Route 53, RDS, Redshift in a highly available and scalable production environment)

- Important depending on application
  - SQL (MySQL, PostgreSQL), Linux systems administration and architecture
  - Scripting experience with Shell, Python or Ruby
  - Object-oriented programming language like Java, C, C++
- Nice to have
  - Experience with Big Data open source technologies (Hadoop, Scala, Spark, Kafka, Hadoop, Hbase, Zookeeper, Oozie) is a plus
  - Experience with continuous integration and deployment automation tools such as Jenkins, Rundeck, AWS CloudFormation, Terraform
  - Network knowledge (TCP/IP, UDP, DNS, Load balancing) and prior network administration experience is a plus

## HARDWARE

### Hardware Engineer

- **Computer hardware engineers develop, design, and test computer hardware components for computer and electrical systems.**
- Important
  - Experience with systems concepts that hardware engineers will need to keep in mind during the design process (fault-tolerance, parallelism, etc.)
  - Some form of coding experience in a “common language” (C++, Python, Java, etc): interviews may ask you to code a solution to problem motivated from hardware
  - Be familiar with the basic structure of computer chips and how they work
  - Be prepared to discuss trade-offs of any design you propose
- Important depending on application
  - Familiarity with hardware programming tools like Verilog, or with low-level languages like C
  - Understanding how to deal with commonly encountered hardware constraints. For example, what principles underlie the design of wearables and small electronics?
  - Solid understanding of circuits, analog design, digital design, optical design, and photonics (requires a solid electrical engineering background)
- Nice to have:
  - A substantial project over many months that you can talk about: Carlab (ECE 301: Building Real Systems), for example is a great one.

## RESEARCH

- ➔ *These roles are fundamentally different from the above: instead of focusing on concretely designing and implementing functional and efficient systems, which is a more linear process, work in research is more focused on designing and running experiments, reading papers, presenting and discussing with other researchers, and overall going*

*through an iterative process to generate, prototype, and evaluate new ideas to improve systems and models in all kinds of tech domains*

### **(Applied) Research Scientist/Research Engineer**

- **Set up large-scale tests and deploy promising ideas quickly and broadly, managing deadlines and deliverables while applying the latest theories to develop new and improved products, processes, or technologies. From creating experiments and prototyping implementations to designing new architectures, research scientists work on real-world problems that span the breadth of computer science, such as machine learning, data mining, natural language processing, hardware and software performance analysis, improving compilers for mobile platforms, etc.**
- **Warning: typically recruit only MS/PhD students, or very advanced undergraduates**
- Important
  - Research experience in Speech, Robotics, Computer Vision, or Machine Learning/Deep Learning
  - Fluency in at least one programming or scripting language (e.g., Python, Java, C, C++) — if ML, Python using Tensorflow and PyTorch is by far most common
  - Deep expertise in at least one of the related science disciplines: Optimization – LP, MIP, Statistics, Machine Learning, Process Control, Combinatorial Optimization
  - Excellent written and verbal communication skills in research setting
  - Experience working with data mining on large datasets (“big data”)
- Important depending on application
  - Familiarity with the CS theory/mathematical concepts: Probability, Linear Algebra, Analysis of Algorithms, Data Compression/Dimension Reduction, Differential Equations, Dynamical Systems, etc.
- Nice to have
  - Not necessary but a very nice plus: publications at top-tier peer-reviewed conferences or journals

## **FINANCE**

- ➔ *A very different domain to apply your technical skills in, often with a much higher focus on math and problem-solving than programming. Nevertheless, it would be hard to find a position in a modern hedge fund that casts off computer science knowledge altogether, seeing as high-frequency trading has become so intimately tied to computers, low-latency systems, and algorithmic computation.*
- ➔ *Often recruits June-November, with some firms recruiting into the winter and spring.*
- ➔ *Interviews involve solving math problems, often in probability and combinatorics, logic puzzles, mental math and estimation (mostly for trading), as well as algorithms and occasionally system design and behavioral, i.e. talking about resume, experiences, challenges, etc.*



- Some firms (e.g. Citadel) like to give especially hard coding or math questions and are interested more in your problem solving process than getting to the final result quickly.
- Notably, having experience in finance or options trading is generally not necessary, although it may be a plus — all you need is quantitative skills.

### (Quantitative) Trader

- **Quantitative traders use mathematical models to identify trading opportunities and buy and sell securities. Often recruit candidates from diverse quantitative backgrounds in academia, software development, and engineering.**
- *Note: Generally recruit from a large variety of quantitative majors: Mathematics, Engineering, Computer Science, Physics, etc*
- **Warning: often recruit only upperclassmen**
- Important
  - Broadly, ability to think about the world systematically and quantitatively, as well as ability to deal with uncertainty in a rigorous and statistical way.
  - Experience with Python, C, C++, Java or comparable programming language
  - Ability to make fast paced, intuitive decisions
- Nice to have
  - Mental math can be useful, especially for certain interviews; similarly Fermi estimation comes up in interviews and is a useful skill
  - Competitive math background is helpful, or equivalently the ability to solve discrete math problems involving probability and combinatorics quickly and accurately
  - Interest in applying technology to solve complex trading problems.

### Quantitative Researcher

- **Apply research techniques and technology in math, computer science to generate and monetize insights about the future, e.g. in optimal trading and market making.**
- **Warning: often recruit only MS/PhD students, or very advanced undergraduates**
- *Note: Generally recruit from a large variety of quantitative majors: Mathematics, Engineering, Computer Science, Physics, Economics, etc*
- Important
  - Broadly, ability to think about the world systematically and quantitatively, as well as ability to deal with uncertainty in a rigorous and statistical way.
  - Experience with Python, C, C++, Java or comparable programming language
  - Strong analytical skills; experience working with and analyzing large datasets
  - Strong mathematical and statistical modeling skills (i.e. time-series and cross-sectional skills) preferred
  - Knowledge of probability (e.g. ORF309) and basic proof-based math is very helpful in interviewing; the position itself is closer to data science
- Nice to have

- Proficiency in coding, with experience using statistical packages (e.g. R, Matlab)
- Exposure to scripting (e.g. Python, Perl); C/C++ a plus but not required
- Demonstrated ability to complete high level, investment related research
- Prior experience in a quantitative role within a trading environment or experience in a position applying advanced quantitative techniques in solving highly complex data intensive problems is always a big plus, but not required

(Note: the term “quant” may refer to quantitative traders or researchers, or both as a whole)

## Additional References

- <https://insights.dice.com/employer-resource-center/tech-hiring-guide-cheat-sheet/>