Codility

A GUIDE FOR ENGINEERING MANAGERS AND LEADERS

Breaking Down Technical Skills Assessments:

Here's What You Need to Change

Overview

Tech hiring is becoming more strategic. Companies need to adequately assess soft skills and hard skills all while ensuring that no bias is being added to the process.

It's simply not effective to just have a "chat" anymore. But with so many technical skills assessments out there, where do you start?

of an individual's annual earnings is the average cost for just one bad hire.

Holding interviews that are not correctly planned and executed could feed into a bad hire - but it's completely avoidable if you use the right assessment technique.

Finding the best candidate for each position means rejecting a "one-size-fits-all" approach. Hiring teams should instead fine-tune their hiring strategies for each of their open roles so that they can build a skilled, collaborative, and motivated engineering force.

About Codility

Codility works with over 1,500 businesses, and we facilitate over 60,000 candidate evaluations monthly. Codility is the pioneer in the tech assessment space - we've tested 3x more candidates than newer entrants in the space.

The insights we've gained over the years form the content of this ebook.

- **Over 1,500 clients** ranging from startups to global enterprises

- Facilitating **over 60,000 candidate assessments** monthly

Unity Deloitte. AMERICAN EXPRESS

intel PayPal zalando

Contents

CHOOSING THE RIGHT TECHNICAL ASSESSMENT

Common Challenges You'll Face

Do you feel like your interview process is outdated? Perhaps you're no longer seeing the types of software engineers that meet - and exceed - qualifications for the job.

Instead of blaming the quality of candidates, analyze your own process to ensure that you're considering true potential fits.

Traditional recruiting methods rely heavily on subjective insights, which help build a "gut feeling" about candidates. These indicators can be useful, but they don't paint the full picture of a candidate's qualifications for the job. You need to check for coding expertise and problem-solving skills too.

The goals of a technical skills assessment are straightforward: identify weak points, provide an opportunity for the candidate to improve, give feedback, and let hard skills shine through in a "real-world" environment.

Tech hiring challenges:

Combination of tests

There's a wide variety of developer positions - from front-end developers to DevOps engineers, the day-to-day tasks of these positions differ significantly. Screen DevOps engineers for their knowledge in Linux/Windows OS administration. Platforms like Codility tailor assessments to check for knowledge of algorithms, data structures, bug-fixing, and refactoring.

Candidate experience

This is essential to retention. No one is going to set expectations better than the actual team that the candidate will be working with. A majority of developers reported that the job they were hired for was not accurately portrayed to some extent compared to its reality. The more the candidate knows about you, the higher the chances that they have a solid understanding of job expectations.

Bias

Both genders tend to use time limits similarly, but data from our candidate research suggests that a higher proportion of women edit and test their solution more often than men. Review your existing skills assessment tools and confirm that they support your diversity efforts, not hinder them.

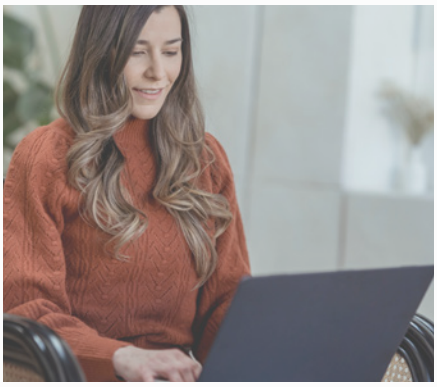
Plagiarism

Candidates can share tasks with one another so you need to be able to refresh assessments otherwise you risk hiring unqualified people. Since engineers often don't have time to manage if a task is leaked, consider partnering with third-parties like Codility who can screen for any task leakage and plagiarism.

CHOOSING THE RIGHT TECHNICAL ASSESSMENT

Key Factors of a Technical Assessment

Remote Versus Onsite Interviews



Hiring at scale

Are you trying to attract top engineering talent in the world? Chances are they don't live in your city. Remote hiring is a more cost-effective option to traditional onsite interviews.

Nowadays many candidates are asked to complete a take-home coding assessment on their own time - and from the comfort of their own equipment. For senior developers, who tend to have other responsibilities, having this autonomy is particularly valuable.

Personal Connection

In an onsite interview it's possible to see nonverbal cues, which are important to how the candidate responds to the question or receives feedback.

An ideal candidate will acknowledge feedback, be open to discussion, and find a way to learn from it. Face-to-face communication also makes it easier to establish a personal connection.

Accelerate time-to-hire

Remote interviews can reduce time-to-hire, making for a more positive candidate experience.

Recent data shows that 60% of candidates have quit an application process because it was too long. Leverage a platform that has a video feature like CodeLive so you can see how an applicant thinks in real-time.

Team Versus Individual Interviews

Imagine that you just graduated from college with a Computer Science degree and get invited to your first onsite interview at Amazon. You follow the hiring manager into the conference room to find five other developers, four of whom you have not met.

The first task they ask you to complete is a whiteboard test. If you're a junior developer, there's a high chance that you feel anxious and skeptical - a surefire way to get turned off by the job, and company.

While the structure of a team interview works for some developer roles, you need to tailor this type of interview to the experience level of the candidate.

For senior developers, this might be your chance to build a collaborative relationship so you'll want to involve several other key members of the team. For junior developers, having more than one person watching and interacting with you can be intimidating.

Question & Answer Versus Code Testing

Many employers are relying on technical coding interviews because they are objective, cost-effective, and skills-based. It's simple to alter time limitations and tailor the level of difficulty to the different roles.

Traditional hiring relies on question and answer (Q&A) as a screening method - a strong way to test a candidate's communication skills and their ability to explain technical concepts.

The issue with Q&A is that you could potentially miss out on key technical areas like the ability to write readable code. For example, how will you know a junior candidate's ability to debug if you didn't see their process first-hand?

Skills that companies need the most:

- Middle and integration software
- Web development and framework
- Artificial intelligence
- Mobile application design
- Cloud and distributed computing

BREAKDOWN OF TECHNICAL ASSESSMENTS BY TYPE

Types of Technical Assessments

Behavioral phone interviews

Although this type of assessment mainly looks at soft skills, there's a small technical component. Behavior-based assessments give you an understanding of how the candidate works on a team and responds to feedback. It's a chance for the candidate to get to know the hiring manager, or technical lead.

Behavior-based interviews are limited in that they don't show how the candidate codes. It's easy for candidates to throw around technical buzzwords to sound like they have certain capabilities that they don't actually possess.

Take-home code tests

In the real-world, developers spend their time reading and maintaining existing code. Coding is the best place to start when screening developers. If a developer is an expert in one language, chances are they can learn other languages.

Utilize finely-tuned, role-specific language and framework testing to determine their true knowledge. Candidate experience is improved when they can complete the test on their own time and in their own environment, with access to commonly-used resources like Stack Overflow. Platforms like CodeCheck offer automatic feedback and grading to speed the screening process further.

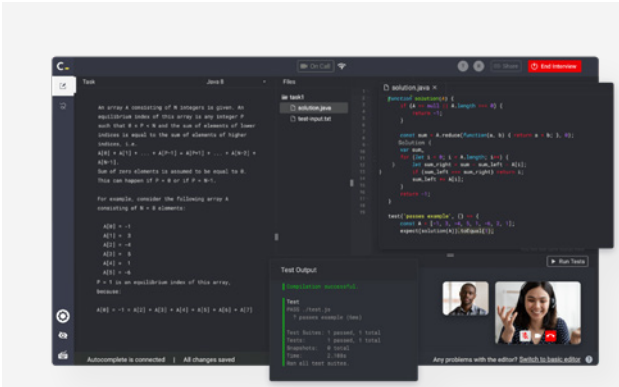
Take home code tests are limited in that they can be time-consuming and feel like "homework" to candidates, a red flag when it comes to maintaining a positive employer brand. You need to send the right signal to candidates in order to attract top talent.

Whiteboard interviews

There's plenty of proof, especially on Twitter, that most developers condemn whiteboard interviews - yet a majority of companies still do this.

The only pro to whiteboard interviews is that you can assess a candidate's thought process and problem-solving approach - but this is limited by whether or not a candidate is reciting algorithms from memorization. If a candidate can find the algorithm best for the question that's asked, then they will be more effective in how they approach their daily tasks.

Whiteboard interviews are outdated and they don't olistically assess technical skills. It also puts more pressure on developers, who tend to be more introverted, and is likely not a part of their day-to-day work or experience.



Collaborative coding interviews

This is comparable to pair programming. People learn from each other - and developers specifically have to be people-oriented in order to work through problems. Collaborative coding through tools like CodeLive gives both the candidate and employer a chance to code with people that'd potentially work with.

This is one of the best ways to see how a candidate works through problems in real-time. It also adds a personal touch to remote hiring, which can benefit remote senior developer recruitment.

Collaborative coding interviews can pose potential issues with bias, similar to phone interviews. It's important to ensure that you have strong internet connectivity so you can use interviewing tools to the best of their ability.

Conclusion

Technical skills assessments are crucial for hiring teams looking to assess candidates' skills and validate their experience.

Candidates you eventually hire will contribute to the company's code base, so their code needs to meets your expectations.

It may take several iterations to design a skills assessment that both achieves your goals and keeps the candidate at the center of every step. Measuring and iterating the process will take time and strategy.

Forward-looking insights

Various factors are impacting the way that companies hire engineers today, and shaping what the future of hiring looks like. We're seeing employer branding and CX as increasingly significant factors needed to attract candidates.

If you're building an engineering team, your hiring process needs to balance assessing candidates and selling to them. Find methods that adequately gauge a candidate's skills so your company is set up to grow technical capabilities and problem-solving capacity.

Key takeaways

- Avoid bad patterns and unstructured interviews. Have a plan that you can reflect on, and optimize.
- Collaborative coding sessions are best for senior developers. At this stage, they should have soft skills in persuasion and collaboration, among others.
- Candidate experience should be a priority. This will organically help you improve employer branding.
- Whiteboard interviews have a bad reputation. They don't accurately reflect the work environment and real-life problems you might face.
- Long interview processes can hurt you. A majority of candidates quit an interview process because it takes too long. Stick to 1-2 hours.
- Behavioral interviews should be paired with a technical assessment. Inefficient hiring is costly so you need to paint a holistic picture of the candidate and their skills.

Where the best engineering teams are built.