Sold to
waiyong.low.wy@gmail.com

The SOFTWARE ENGINEERING MANAGER Interview Guide

Vidal Graupera

The Software Engineering Manager Interview Guide

Vidal Graupera

managersclub.com

Copyright © 2019-2020 by Vidal Graupera

All rights reserved.

First Edition: March 2020

Version 1.1, April 2020

No part of this book may be reproduced in any form or by any electronic or mechanical means, including information storage and retrieval systems, without written permission from the author, except for the use of brief quotations in a book review.

Dedication

To Tara, Maggie, Grace, James, Kathleen, Vidal "Joey", Bridget, and Georgia

Table of Contents

Introduction

Interviewing can be challenging, time-consuming, stressful, frustrating, and full of disappointments. My goal is to help make things easier for you so you can get the engineering leadership job you want.

The Software Engineering Manager Interview Guide is a **comprehensive, no-nonsense book** about landing an engineering leadership role at a top-tier tech company. You will learn how to master the different kinds of engineering management interview questions. If you only pick up one or two tips from this book, it could make the difference in getting the dream job you want.

This guide contains a collection of 150+ real-life management and behavioral questions I was asked on phone screens and by panels during onsite interviews for engineering management positions at a variety of big-name and **top-tier tech companies** in the San Francisco Bay Area such as Google, Facebook, Amazon, Twitter, LinkedIn, Uber, Lyft, Airbnb, Pinterest, Salesforce, Intuit, Autodesk, et al. These companies generally pride themselves in holding a high interview bar and rigorous processes. Also included are tips on how to get the interview and succeed at the onsite interview.

In this book, I discuss my experiences and reflections mainly from the candidate's perspective. Your experience will vary. The random variables include who will be on your panel, what exactly they will ask, the level of training and mood of the interviewers, their preferences, and biases. While you cannot control any of those variables, you can control how prepared you are, and hopefully, this book will help you in that process.

I will share with you everything I've learned going through this cycle multiple times while keeping this book short enough to read on a plane ride. I will share all the tips I picked up along the way. If you are interviewing this guide will serve you as a playbook to prepare, or **if you are hiring give you ideas as to what you might ask an engineering management candidate yourself.** Good luck in either case!

Vidal Graupera

December 2019

Chapter 1: Answering Management Questions

In this chapter, I will cover my recommended techniques to answer any behavioral or competency-focused question you will face.

- Think. Why are they **really** asking this question?
- Selling Yourself
- Behavioral questions
- “Disney-like” Versions
- What to say when you don’t have a story
- Asking Clarifying Questions

Think. Why Are They Really Asking This Question?

Before jumping to answer any question pause and consider why your interviewers are asking it. What is the real point of the question? It is likely your interviewers are asking these questions for a reason that may not be what it seems on the surface. For example, “What are you looking for in your next role?” They are probably looking to see if your goals are consistent with the company’s values, see the section on Company Values. And “do you have any questions for me?” is really checking to see what it important to you. I will cover more examples in Chapter 4.

Selling Yourself

Describe your achievements in the most powerful way you can. Focus on your teams accomplishments and impact over implementation details. Level-up your stories and make them into exciting lessons. Rather than just tell what your teams worked on, cite the business metrics that your team moved e.g., "we increased site speed 10x, or conversion rate by 5x, etc." While you want these achievements to sound as impressive as possible, you are not being evaluated on them but your future potential in this role. Therefore, you should explain how these past achievements demonstrate you possess the skills and competencies to be successful in this role. Anytime you can say "I've done that before..." in response to a job requirement, that is good.

Behavioral Questions

The management questions typically asked are behavioral or competency-focused. Behavior questions often start with "Tell me about a time you….". The idea is that previous behavior is a better indicator of how you will perform in the future than vague hypothetical answers. These questions also have the benefit of checking the depth and breadth of your experience. Behavioral questions are sprinkled throughout the interview process, including recruiter screening, phone screens with hiring managers, onsite as well as follow up interviews. These questions play a significant part in the selection process.

There is often no one correct answer. How you handled past situations demonstrates not just your management skills, but whether you would be an excellent cultural fit, a positive influence in the office and your soft skills. All of these are incredibly crucial for managers. Why are behavioral questions difficult? Because you can't show too much modesty or too much confidence. Overconfidence could be interpreted as being a know-it-all or arrogant. If you are too modest, you risk not impressing our interviewers enough.

For each of these questions, you should prepare a short specific story. Many people recommend using the STAR Interview Response Technique to formulate your answer. STAR stands for Situation, Task, Action, Result. This technique helps structure your replies and make sure you cover all the relevant angles. From the listener's perspective, it can make the story more specific and compelling.

Here is an example. When I took over the team at company X, I noticed they were often taking too long to do designs and UX studies. My task was to ensure the team could carry out their work and increase development velocity and predictability. The action I took was to set up training on design and best practices. We started regular design reviews. I brought in designers to help the team. The result was the team learned best practices about design; we got faster; the team felt more confident, and our customers were more satisfied moving up our net promoter score by 0.2 the next quarter.

Ideally, you should prepare two stories for each of these questions. Having at least two different stories for each of the questions will let you try out different versions. I recommend you avoid sharing the same story more than twice during an interview. The reason is that if there is some concern over your qualifications during the debrief, it is good if interviewers have different stories to relate to your experience. It could also be that one of the stories you chose is not one that resonates with your interview panel. Vary your stories; don't take them all from just one previous position or project if you have worked at multiple companies. Having multiple stories can highlight the breadth of your experience.

I recommend you reference actual people, not abstractions. Avoid naming specific people, of course. For example, “When I was working at Company X I had an employee who....” is more powerful than an abstract statement about how to handle such a situation.

“Disney-like” Versions

Tell the best “Disney-like” version of each story, if you have one, where things worked out. You can leave out details that don’t help your story to simplify sharing them. I recommend you eliminate any stories or details that do not paint you in a positive light. However, some stories that have a negative result can highlight your strengths in the face of adversity. In those cases, be sure to discuss what you learned and how you would approach the situation differently next time.

Here is a real-life example. When asked, "How have you managed low performers?" I have a story about someone who was a referral to my team, and I mentioned they were a "strong" referral. Once, an interviewer started digging into how such a referral was not working out. Did I not screen them as diligently because they were are referral? Did I fail to hold the bar at the interview since they were a referral? Interesting points, but that line of questioning was not helping me and had nothing to do with the point on performance management I wanted to make. Going forward, I never mentioned that person was a referral in the future telling of the story.

Finally, watch out for overly complex stories that are difficult to describe or require too much context (e.g., company specific knowledge that will not make sense to the interviewer).

What to Say When You Don’t Have a Story

I do not recommend that you lie or invent stories if you don’t have them.

If you are a new engineering manager you might not have a story for a particular situation. A very short reply to the effect “I don’t know” is not good. If you don’t have a story, here are some things you can try. Assuming you saw another manager or your manager deal with a similar situation, you can say, “This hasn’t happened to me exactly, but here is what I’ve seen…” and explain how you would handle it based on that experience. There are also many blogs and books on engineering management. You can search for answers there and say something to the effect, “I haven't run into this situation myself, but if I were to, from what I've learned this is how I would solve it.” This kind of research requires preparation.

Asking Clarifying Questions

I learned to check for understanding and level of detail. I will ask things like, “did that make sense?”, “is that what you are looking for…?” or “was that enough or too much detail?” I once got feedback that I didn't give enough details so I made it a point to check I am sharing enough details to answer the question.

You should check with the recruiter and interview panel for their preferences to get the right balance between too much and not enough details. Some interviewers want precise time-bound answers. They may have a fixed list of questions they need to make it through. Others may prefer more in-depth answers and have the flexibility to dwell longer on a specific question.

Chapter 2: The Job Interviews

The structure varies by the company, but generally, there was one recruiter call, then one hiring manager call, and then possibly a phone screen with someone other than the hiring manager, and followed by an initial onsite with 5 or 6 people. The most extensive onsite panel I faced was eight people, and the smallest was 3.

If the first onsite was successful, there would often be another onsite with typically 4 to 5 people more senior than the first panel, but this was not always the case.

It will help if you understand the dynamics of the game you are playing. Companies design the tech interview process to make it easy to evaluate candidates. They are more concerned with making sure not to hire someone they will regret hiring (aka "false positive") than missing out on hiring a great candidate (aka "false negative"). Top-tier tech companies have lots of applicants for open positions. The only cost of missing a good hire for them is the time they invest in the interview. The larger the pool of candidates, the pickier the company can be in their process.

Phone Screens

The phone screen was usually centered on engineering leadership, although I had a few companies that asked for online coding assessments or technical phone screens instead. If you asked for a coding or technical phone screen, then prepare just as you would for a senior engineering role. Your interviewer may be a senior engineer. If you are a software engineering manager, prepare to write code.

More typically if the phone screen is about management, expect a broad array of behavioral questions. You could be asked for example, how do you do planning, how have you dealt with low performers, etc. Refer to Management Questions for a list of these and how to answer. The interviewer is also usually looking to see if you would be a good cultural fit for the organization and if your background is a good match for the team. Here I was almost always asked what I was looking for, why I was leaving my previous job, and why I was interested in the given company, etc. Typically these last for 45 minutes to 1 hour.

Prep Call with the Recruiter

Often you will get a prep call from the recruiter before the onsite. You also may get an interview prep document with sample questions, etc. Take advantage of the prep call by asking good questions of the recruiter. **I would recommend asking what kind of person they are looking for, and what are the typical things that cause candidates to not make it past the onsite**. Recruiters want you to make it and often cannot give you feedback after your interview (not that it matters anymore to them at that point.) Knowing why others haven't made it can be super helpful.

I strongly recommend you get a copy of the job description and read through it very carefully. Make sure you can talk about each of the specific items listed. It's true that many job descriptions are poorly written but don't discount reading through yours to make sure you understand what they are seeking. I've also been to interviews where I was discussing one job with the recruiter, but due to some miscommunication, the job description was different. After that happened to me, I always check not to make sure there is no disconnect.

Ask the recruiter, "can you tell me a little bit more about who I will be meeting? or "how are all these people connected?" Organizational and reporting structure is not easily discoverable via a Linkedin search. The recruiter may know some helpful info about these people's personalities and how they typically assess candidates.

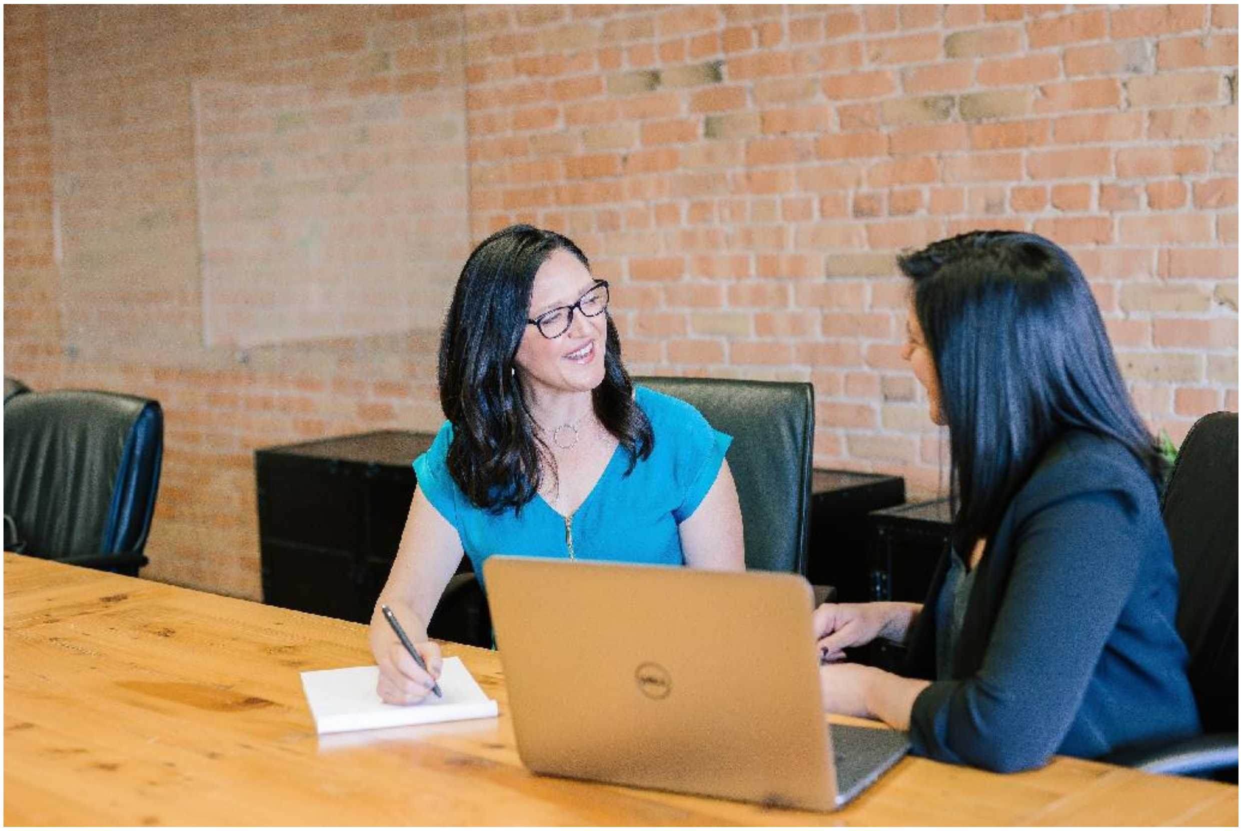
You might also ask why this position is open and how long it has been open. You can ask what the chances are of candidates making past the onsite to the offer. You may be surprised to learn that the pass-through rate, if they even track it or will share it, is generally low. The vast majority of the time, recruiters are calling candidates after the onsite to tell them no, so it helps to understand the odds you are facing. Since the odds are high that you won't make it past this round, focus on the next interview. **Much like sales, interviewing is a numbers game.**

Look up panelists on LinkedIn before the interview. Also, Google them and search for them on YouTube, Twitter, etc. if they appear to be well known in the industry. Scan their blogs if they have one. This prep work will help you unearth common interests or mutual friends and acquaintances. Did you both go to the same school or live in the same city? If they have written articles or given talks, take a moment to scan them. The more you know about the interviewer and the more connections you can build, the better. I found that people reacted well to me telling them that I read about them beforehand. In my experience, this was always a positive way to start the conversation.

The recruiter influences the hiring process. At some companies, the recruiter even has a vote, but I found this is rare. Whether the final decision is "hire" or "no hire" you will want to have a good relationship with them to seek feedback (see below) and reapply later or negotiate an offer. Therefore, it is in your best interest to build a good relationship with the recruiter.

The recruiter is motivated to find the right candidate and close the job opening. With that said, don’t forget that **the recruiter works for the company and not you**. They will be rooting for you to the extent you can fill the opening.

Onsite



The exact structure varies, but a typical format includes possibly one coding interview, up to two design interviews, and three or more managerial/leadership interviews. This mix has become common across the tech industry. A few companies add some unique interviews described below. Unfortunately for candidates, the coding and system design interviews, which can make up a large percentage of the interview, have nothing to do with the day-to-day of engineering management.

Generally, you will be meeting with other managers and individual engineers. You may also meet with a product manager or even an HR manager but that is less typical. While these interviews could all be done remotely over something like Zoom, since I was interviewing locally mine were in person. With that said, it was not uncommon to have 1 or 2 interviewers over Zoom with someone from a remote office or who was working from home that day.

Here are some sample onsite schedules.

10:15 AM - 10:30 AM	Greeting with Recruiter
10:30 AM - 11:15 AM	Engineering Management & Leadership
11:15 AM - 12:00 PM	Architecture & Systems Design
12:00 PM - 12:45 PM	Practical Data Structures & Algorithms
12:45 PM - 01:45 PM	Hiring Manager Lunch
01:45 PM - 02:30 PM	Product Sense & Working with a PdM
02:30 PM - 02:45 PM	Wrap-up
10:00 AM - 10:15 AM	Recruiter - Greet
10:15 AM - 11:00 AM	Engineering Manager – Design Interview
11:00 AM - 11:45 AM	Director of Engineering - People Management
11:45 AM - 12:30 PM	Engineering Manager - Lunch
12:30 PM - 01:15 PM	Engineering Manager - Coding
01:15 PM - 02:00 PM	Software Engineer – Design Interview
02:00 PM - 02:45 PM	Engineering Manager – Career Motivation
02:45 PM - 03:00 PM	Break
03:00 PM - 03:45 PM	Engineering Manager - Project Retrospective
03:45 PM - 04:00 PM	Engineering Manager - Walk Out

Take frequent breaks. You may be offered bathroom or water breaks between each segment. I recommend taking advantage of a lot of these even if you don't need to use the restroom or get water to have a few minutes to clear your head between each interview. Walking to get coffee with your interviewer can also give you a few minutes to build rapport. While I think this is an extremely bad practice, you may even be left alone between interviews! Sometimes the next interviewer is running late and people are in such a hurry they will leave you alone to wait in the conference room or even to wander the halls. I found this a very poor signal on the company and how the interview is going. I never abandon candidates when I am interviewing.

Company Values

Some companies, e.g., Amazon, have a series of company values or leadership principles. Below, for example, are the Amazon.com Leadership Principles. During an interview at Amazon, you can even ask which principle someone is asking about if it is not clear to you from their questions.

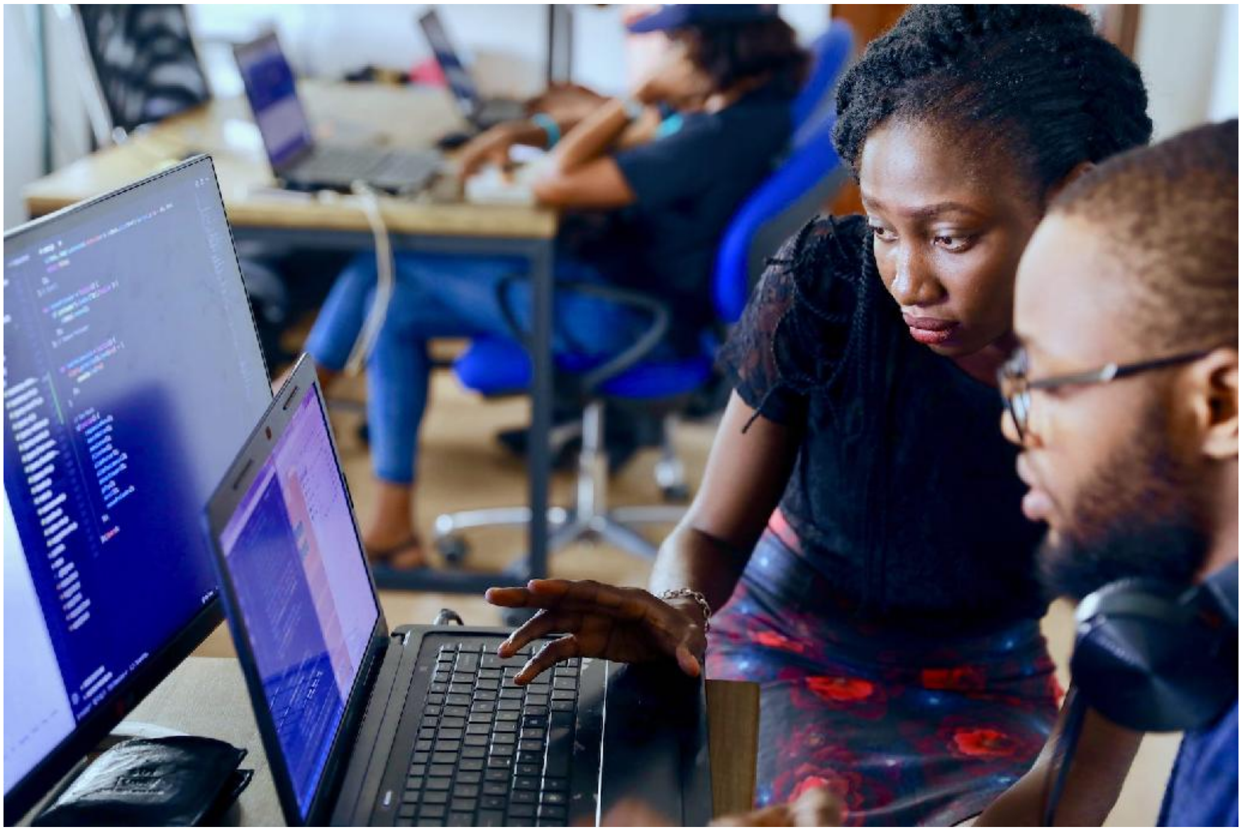
- Customer Obsession
- Ownership
- Invent and Simplify
- [Leaders] Are Right, A Lot
- Learn and Be Curious
- Hire and Develop the Best
- Insist on the Highest Standards
- Think Big
- Bias for Action
- Frugality
- Earn Trust
- Dive Deep
- Have Backbone; Disagree and Commit
- Deliver Results

Visit the company's website. Review their core values and what they find most important, and speak to these during your interview. No company values posted? Check their blog to find valuable information about company culture. Look for YouTube talks or podcasts featuring the Founder, CEO or

CTO. If it is a public company, you can look at their investor information site. This research also allows you to ask informed questions that show interest.

I recommend that you have 2-3 stories prepared ahead of time that are specific to each of the company values if you are interviewing at such a company. Don't try to make them up on the spot – it will be very hard and you can get it wrong.

Coding, Algorithms and Data structures



About 50% of the companies I interviewed for a software engineering manager position required a coding interview on a whiteboard or computer. A few companies even asked me for two coding interviews! To me, this became an indicator as to whether managers are expected to be more than people managers. On the other hand, other companies, without any programming expectations, may also do this just to assess the candidate’s technical level.

If you have not been coding for a while or actively interviewing, I sincerely recommend you study a few months for this part of the interviews. This more technical section is often about data structures and programming challenges that managers and engineers do not deal with in real life. It has nothing to do with management. Think of it as something like the SAT or GMAT for college entrance.

Even though recruiters may tell you that engineering managers are not “expected to code,” or “don’t spend too much time preparing,” you should not interpret this to mean that the coding questions asked will be easy ones. Therefore, **do not study only easy problems**. I recommend you do prepare and it might take a significant amount of time. Contrary to what some will tell you, there is also little or no credit for effort, or obvious brute force solutions. Expect that you need to produce a working solution using algorithms with a decent time and space complexity and appropriate data structures in approximately 45 minutes to 1 hour. Well-calibrated panelists will have asked the same question to many other candidates, including senior and staff level engineers, so your answer will be judged in comparison.

While it is essential to solve the problem, it is also important how you work with the interviewer and connect with them. They will be subjectively evaluating you on how you interact with them and how you appear to code. If they are on your side, things will go better, and your solution will be graded more positively, and the gaps more easily overlooked. What clarifying questions did you ask? How confident at coding do you seem? Did you listen to their instructions and tips? Listen for clues and hints. If your interviewer gives you hints or prompts you in a certain direction, run with it!

If you appear nervous, scatterbrained in your approach, and overall lacking confidence in your coding skills, you will not do well even if you happen to come up with a working solution. I've failed a coding interview because of how I was unable to connect with the interviewer even though I managed to deliver a working solution before time ran out. I've also passed coding interviews where my answer was not so great, but I connected well with the interviewer. Again you must solve the problem, but don't focus on it so much you become oblivious to the human factors.

Prepare as if you are interviewing for a senior software engineer or higher. Interviewers can choose from a large set of programming problems and classes of problems. There is a fair amount of random luck in the question you will get. If you solved a similar problem before, you will have a significant advantage! If you never saw this kind of question or solved anything like it, then passing this segment will be a lot harder. The best way to minimize the chance of getting a problem; you have no clue how to solve is to practice a lot of different kinds of problems. **One of the best resources I found for preparing was https://www.algoexpert.io/. Use this code jojfu-46 for 15% off.**

There are popular books such as *Cracking the Coding Interview* and sites like the following where you can practice:

- https://leetcode.com/
- https://www.careercup.com
- https://www.interviewcake.com/

Also interesting is Coding Interview University a GitHub repo with over 100K stars containing a massive amount of resources to prepare you well for a technical interview at just about any software company. There is also a popular repo with System Design prep material.

On this page, I keep the most up-to-date resources for preparing for the coding interview https://www.managersclub.com/practicing-for-coding-algorithms-and-data-structures-interviews/.

I recommend you ask if you can bring your laptop with you to the interview because you'll be more comfortable in your environment versus on a laptop provided by the company. Bring a power cable and any necessary adapters to connect your laptop to an external monitor or projector, e.g. HDMI adapter, so you can quickly show the screen to your interviewer if necessary. Don't assume they will have adapters easily available in the interview room.

You should always start on the whiteboard. I bought a whiteboard for my house to practice. The whiteboard is easier to show problem-solving as well as communication/collaboration skills. At the very least, the (algorithmic) design phase should be carried on the whiteboard. Once that is agreed with the interviewer, the next step is either coding on a computer or to finish on the whiteboard.

If you are well-practiced and confident about how to solve the problem and have at least 45 minutes, and your interviewer is expecting working code over pseudocode, I personally would choose a computer over a whiteboard to do the actual coding. If you have less time or are not very confident about how to solve the problem, the whiteboard gives you more fudge factor and can be faster. In either case, do not start coding unless you are confident in your approach. An advantage of writing out the real code is you can write tests to check if your algorithm works. It takes longer and is more difficult to debug algorithms on a whiteboard in my experience.

If you have a choice of the time slot, I would recommend doing the coding interview first — you will be fresher than after speaking for many hours.

If you happen to contribute any open source projects on GitHub, for example, I recommend pointing out and sharing these with the interviewer. Engineers generally do coding interviews, and this may help build rapport with them and impress them. However, I would not invest time in creating such resources since, in my experience, if you are given a coding interview, you will be graded on your ability to solve the problem in the room, not on external code contributions.

System Design and Architecture Interviews

It is common to see at least one and often two design interviews. These interviews are the same as those for any senior software engineer or higher. You may be asked the following:

- Generic design of a popular system
- Design specific to a domain

- Design of a system your team worked on

These interviews are very interactive and open-ended with not a single answer. The purpose is to evaluate how knowledgeable you are about design and tradeoffs. You will likely be drawing diagrams and perhaps data structures on a whiteboard.

Generic Design Of A Popular System

Examples might be to design any popular website or application such as Facebook, Facebook's News Feed, Twitter, YouTube, Google docs, WhatsApp, Uber, Tinder, etc. No one designs such systems by themselves as part of their job, so these questions also require advance preparation.

There is a fair amount of random luck into what kind of popular system you may be asked to design. The best way to minimize the chance of getting a design problem you have no clue about is to review the design of a lot of different kinds of systems and apps. Practice them on the whiteboard.

Read about how hyper-scale companies such as Amazon, Google, Uber, etc. have implemented their systems. You can often find posts on medium.com, etc. summarizing these. You can also read some widely cited papers such as Dynamo: Amazon's Highly Available Key-value Store, The Google File System, and Bigtable to get design ideas. Another good place also to prepare for these interviews is to watch YouTube videos on system design. This channel has good system design videos, Tech Dummies — Narendra L and so does this playlist System Design — Gaurav Sen.

You should study up and understand how the internet works and be familiar with the various pieces (e.g., routers, domain name servers, load balancers, firewalls, etc.) You should understand how, for example, Twitter and Facebook implement "fan-out" in their feeds, and how chat services like WhatsApp work. These are popular and variations come up in many design interviews. You should brush up on when to use SQL and NoSQL databases, load balancers, API gateways, and API design. A good resource for brushing up on this is https://docs.microsoft.com/en-us/azure/architecture/patterns/. A good book is also "Designing Data-Intensive Applications: The Big Ideas

Behind Reliable, Scalable, and Maintainable Systems”. It is common to be asked how you would design a database (schema, tables, indices, etc.) for such a system including choice of database, e.g. SQL and NoSQL, and how you would scale it.

You should be comfortable with asking additional questions like, “Am I allowed to use DB, caching, distributed solutions, cloud, etc.?” You don't want to go with an "expensive" solution only to be told they were trying to see how frugal you can be with an MVP. As you ask clarifying questions, write the list of requirements on the whiteboard. The list should typically be the first thing you add to the whiteboard.

A Design Specific To A Domain

For example, if you are interviewing for a team that processes payments online, you may be expected to know about encryption algorithms and how to design a system that keeps data at rest and in transit secure.

At the very least, think about how to design systems that are relevant to the team you are interviewing for and learn about design patterns that could be applicable. Again, for example, if you are interviewing a ride-sharing company, e.g., Uber or Lyft, think about how you might implement the rider and driver apps, those companies offer.

Design Of A System Your Team Worked On

Finally, you may be asked to diagram and explain the design and architecture of a system that your team worked on, and discuss the design decisions and tradeoffs. You should refresh your memory of all the details and be able to draw out and explain the systems and the tradeoffs convincingly. The purpose of this is to understand what kinds of systems you have experience with and how you approach making tradeoffs.

Lunch Interview

You may have a lunch “interview.” You may meet some engineers or product managers from the team at lunch. Mostly these are non-evaluative interviews which means they will not be officially quizzing you or writing up feedback for the debrief. With that said, the lunch interview is still a real interview. The people at lunch will be evaluating you for cultural fit within the team and your motivations for joining the company. It is possible to say the wrong thing and lose the job at lunch, however, so be careful. If any of the panelists are also at lunch making a great impression on them can impact their subjective assessment of you and improve your chances.

Managerial and Leadership

These interviews were divided into a variety of topics with each panelist drilling into a particular topic with behavioral interview questions. You can find the topics in Chapter 4.

The Bar Raiser

Many companies involve a bar raiser, a concept popularized by Amazon. A bar raiser is a person who does not have a stake in filling the job opening. They receive special training and are generally selected due to their experience and high skill in performing interviews. Their role is to assess the level of the candidate compared to others in different organizations in the company. In other words, is the company keeping a consistently high bar across different teams? Often they may also run the debrief meeting. At some companies, the function of the bar raiser is replaced by a hiring committee.

Typically you can identify who is the bar raiser because they do not work in the organization you are interviewing for. Another dead giveaway is if they are remote on Zoom or teleconference from another office. Needless to say, you must pass this interview. They will almost always be asking managerial/leadership questions. Bar raisers carry a lot of weight and if you impress them, they can advocate for you. Be sure to make a good impression and be ready to shine. Usually, they don't ask technical questions but are trying to see if you are a good cultural fit.

Unique One-Off Interviews

In place of one of the usual kinds of interviews above, I was also asked to do by different companies the following:

- Mock one-on-one. In this interview, the interviewer posed me with some challenges you might hear from a direct report as a manager in a 1:1, and we role-played coaching them through it.
- Mock code review. I was given a piece to code and asked to review it.
- Mock interview. Here the tables are reversed and you interview someone from the company and explain later why you would hire them or not. The purpose is to see how well you interview candidates. I personally found it super awkward.
- Culture fit interview.
- Technical presentation on a topic of choice.
- Project Retrospective

No company I interviewed asked for more than one of these kinds of interviews. These were overall rare.

Chapter 3: Tips To Succeed

How To Get The Interviews

Referrals

The best approach I found is getting a **referral** from a friend or a friend of a friend. The person referring you will often receive a bonus if you are hired. Well-known high profile tech companies often get thousands of online applications for each job posting. Coming in as a referral is the best way to get your resume on the radar of the recruiter and hiring manager. Your resume can easily get lost if you only send it in via an online application.

Note the person referring you will be asked how they know you. Are you a casual acquaintance, or someone they worked with? Can they vouch for your skills? Obviously the more they know you and can vouch for your skills the better. Nevertheless, I still think even a casual referral can be enough to get you noticed and in the door and is far superior to an online application.

The person referring you can tell you a lot about the culture and the team you will be interviewing with. They can give you insights into the interviewing process since they have gone through it.

Networking

Reach out to colleagues you haven't talked to in years. I did a lot of lunch and coffee meetings while I was searching. Even if your friends are not hiring, they may know of unadvertised openings within their companies.

I went to a lot of meetups and events during my job search. A lot of people at meetups are also there to hire or know hiring managers at their companies. If an event is at the offices of a company that you want to work for this a perfect opportunity to meet the recruiters and people who work there in person! There are conferences for engineering leadership that you can attend to build your network. You can find a list of major conferences on my site at https://www.managersclub.com

There are also several online Slack communities and public LinkedIn groups for engineering managers you can join and network virtually, e.g., eng-managers.slack.com and rands-leadership.slack.com. Hiring managers will often post jobs on those forums. You can also post that you are looking and even your resume. Finally, it is possible to ask and get a referral from people you meet on these communities.

LinkedIn

You can get inbound leads from your Linkedin profile. Your LinkedIn profile should be updated and have a recent photo. Recruiters and sourcers look for candidates by doing keyword searches. You should make sure to have all the right keywords. However, do it naturally, so it doesn't look like you are keyword stuffing. Ask for a few recommendations if you don’t have any for your most recent position. There is a profile setting on LinkedIn to say you are open to new opportunities that you should turn on. A recently updated profile is also a signal that you are in the market that recruiters can search on. I don’t think it is necessary to explicitly write on your profile that you are looking for a job if you do these steps.

LinkedIn will also show you jobs that it thinks would be a good match for your skills. This is a powerful feature that can find positions at companies you might not have considered. Before immediately applying I would check if I know someone at the company that can refer me.

Get your resume ready. Make sure it has no typos and zero grammatical errors. Have someone proofread it. Focus your experience on what was the effect and impact of your work not just what you did. A lot of this will be the same as what you want on your Linkedin profile so you can do both at the same time. Make your resume and LinkedIn profile very easy to scan. Recruiters and hiring managers may only spend a few seconds scanning it to see if they want to contact you.

I think it is best to position yourself as a manager within a specialty, e.g., dev ops, front-end engineering, etc. and highlight those experiences. While engineers can often be hired as generalists within a given area and later assigned to teams, management positions are usually specific to a team and a business need. A company may have a particular group that they need a new manager for or to grow a new team with specific capabilities—therefore, positioning yourself as a generalist is less likely to match those job requirements and stand out. The closer you can position yourself to the job opening, the more likely it is to get in the door.

Ageism

Ageism is, unfortunately, a big issue in tech. In my experience, anyone over age 40 falls in this bucket. Here are a few things you can do to mitigate it if ageism is a concern for you. Age neutralize your resume and LinkedIn profile. Remove all graduation years. College graduation year is the most common way that people infer your age. Recruiters can also search for people who graduated from college since ____ year.

Shorten up your resume and LinkedIn profile. Your experience may be extensive but resist the urge to list everything you've done in your career. Hiring managers are going to weigh your recent experience, e.g., your last 3 positions most heavily. It is therefore only essential to detail the last 10 to 15 years of your work experience.

Focus your job search on larger well-established companies that are hiring a lot of people. There are several reasons for this. Companies that need to hire a lot of people have fewer incentives to discriminate since they need people. Larger well-established companies recognize they should not discriminate and often prove training to their staff and attempt to correct for this. Newer and smaller companies often do not have this level of process maturity. Finally, larger well-established companies likely already have a number of older workers in their workforces. With that said, it is a huge issue in tech and a big concern for workers over 40.

Scheduling and Timelines

In my experience, it can take 3 to 4 months or more to complete a job search for an engineering manager position from scratch. While the number of tech jobs is large, there naturally exists an order of magnitude or fewer openings for managers than individual contributors. The significant elements of the search include preparation, searching, networking, phone screening with various companies, onsite interviewing (which can consist of multiple rounds) again with multiple companies, and then finally offer negotiation, and determining a start date.

You will undoubtedly have some companies that you would love to work for more than others. I recommend you not schedule any interviews with them when you are just starting. You will perform better and feel more comfortable after doing several complete interviews. I know I did. Put off interviewing at your dream companies until you have practiced. I made this mistake myself. It is easy to overestimate how well prepared you are or underestimate the difficulty of questions you will be asked. Even better if you can have an offer from one of your not first choice companies.

I found it useful to keep a spreadsheet with all the companies I was talking with and where I was with each, what was the next step, etc. Otherwise, it can be easy to lose track and let something slip through the cracks.Here is the template I used.

I recommend not scheduling onsite interviews on back to back days. Each onsite can be very exhausting and it is good to have an entire day to recover and prepare for the next one.

Interview Feedback

You will be flying blind. This is a huge challenge and frustration as a candidate. Most companies will give you very little specific feedback on your interview results besides a “hire” or “no hire.” If you are able to build a relationship with the recruiter and the company does not have a strong policy against it, and you ask nicely, you may be given some details.

You can try to ask your panelists at the end of each interview for their assessment of how you did but this is not effective for a few reasons. First, it is unlikely you will get a truthful answer. They probably just want to get out of there and back to work, especially if they have made up their mind that you are a no-hire. Most people are not comfortable giving negative feedback which is what you need to improve. And secondly, I think you are better served by using the last few minutes of the interview making your case why they should hire you than asking for interview feedback which will not change the result.

One interesting technique I've heard is to ask, "What doubts do you have in your mind that I am the best candidate for this role?" This question invites feedback, isn't necessarily negative feedback, and also allows you to get the chance to sell yourself one last time.

If you are interested in reading more about why companies do not share feedback, see Forbes The Real Reasons Why Job Seekers Are Not Given Feedback.

Mock Interviews

A way to get feedback on your interview performance is to do mock interviews. A mock interview is when someone asks you questions just like in a job interview but for practice and training. You ideally want someone who has experience as a hiring manager at the kinds of companies you are applying for and has sat in actual interviews and debrief panels.

You can ask your friends to conduct a mock interview or pay someone to do it. There are people who sell this as a service. The advantage of having a friend do it is that it is free. The disadvantage is that they might not have the right experience and being your friend may not give you completely candid feedback.

Professional mock interviewers will probably give you better feedback but you have to pay. Costs can run $200 or more per hour. A few companies I know that sell mock interviews are https://www.careercup.com/interview and https://www.interviewkickstart.com/.

Panelists

Understand that most panelists (except the hiring manager and recruiter) probably don't want to be in the interview room. They have a lot of other work to do, and interviewing you is not helping them get their job done. While getting the job might be a life-changing event for you, it does not mean the same thing to them. Therefore, they may not always be paying 100% attention to what you say; they may be in a hurry, etc. Some may have received extensive interview training and done a lot of interviews, others none, and have little experience.

Do not assume that anyone read your resume more than a few seconds before walking into the interview room. They may have read an old, outdated resume! If they are holding a resume, check it is the latest up to date version. This happened to me more than once.

When you are asked if you have any questions, or to talk about yourself, I recommend you summarize what you want them to know and remember that they might have missed reading from your resume or they forgot to ask. Just because something is on your resume doesn't mean they picked up on it unless you discussed it in the interview. You can say something like, "I don't know if you picked this up on my resume, but I wanted to call out…."

Making a Great First Impression

Interviewing is a highly subjective human-centered process. First impressions are not specific to interviewing. People will make a snap decision about you in the first few seconds of meeting you. Their impression of you will influence how they think you will behave in the future. Since the interview is basically about how they judge you will act in the future, your impression will significantly influence their evaluation one way or the other.

First impressions are often based on stereotypes. Even people who are aware of stereotypes have difficulty in seeing past them. It could be based on how you are dressed, so check with the recruiter as to the dress code. You do not want to dress too differently than is typical at the company. Being dressed appropriately is easy and will make it seem you fit right in.

All you say will be filtered through a lens of confirmation bias to confirm their initial positive or negative impressions. You will have to work hard to reverse this if it is not in your favor during the interview. They may even see you as a threat if you are interviewing to be their manager or peer!

There are positive psychological techniques to help you make the best impression and be perceived the way you want to be perceived. Even with something apparently as black and white as a coding question, if the interviewer is on your side it can go much better than if they have already set their minds against you.

Many books have been written just on this topic. The most famous book on making a good impression is, "How To Win Friends and Influence People" by Dale Carnegie. An online resource I found useful for learning about making a good impression and displaying confidence is The Charisma on Demand YouTube channel. The videos are full of valuable and actionable tips. Many engineers are introverts and this channel has relevant content addressing making a good impression while being an introvert. There is an amazing TED Talk by Amy Cuddy that discusses how body language affects how others see us, but it may also change how we see ourselves. Try doing a "power pose" for a few minutes before your interview!

One more thing you can do to make a good impression and offset any negative stereotypes is to build some rapport or connection with the interviewer. For example, if you went to the same school, grew up in the same town, etc. you should mention that. While this is generally no "right" first impression, when interviewing as a manager, I feel it is essential to smile and show confidence. I found it works best always to keep eye contact with your interviewer. Many times companies hire based on overall personalities and culture fit so it is important to stay cool under the pressure of the interview.

Debrief

The interview panel will meet following your onsite interviews and discuss their feedback. Afterward, your recruiter will be in touch with an update. Depending on the company, this can take from a few days to many weeks. You should ask the recruiter how long this typically takes to set your expectations. Then I would add another 50 to 100% buffer onto that. There is nothing you can do during this period except wait patiently.

It is possible that you may get hired with dissenting votes but this would be rare. Thus the importance of being super well prepared. Even if you pass all the interviews, a job offer is in no way guaranteed. You may have answered everything correctly, but no one on the panel may feel that excited about your candidacy or there might be other candidates they prefer more. In general, companies are more conservative in hiring engineering managers than individual contributors due to the fear of a bad hire of a manager may "tank" an entire team.

Even if you performed awesomely, there are factors outside of your control that might lead to you not getting the job. I recommend you spend your time immediately looking for and preparing the next interview, and not waiting around for anyone to call you.

Thank You Notes

After the interview, send a thank you note and route it through the recruiter. People may be disappointed if you don't send one which could hurt you. I would send it within 24 hours. A handwritten note is not necessary and anyways it would take too long to arrive. An email is fine. If you happen to have the email addresses of the hiring manager or other panelists you can include them.

With that said, a thank you note is a checklist item. I recommend you do not spend a lot of time writing one. Invest your time to prepare for the next interview!

Here is an example template I have used.

Subject: Thank you for your time and hospitality

Thank you for the opportunity to interview at ______.

It was a pleasure to meet everyone on the panel. I had a great experience. The more I learned about ______ the more excited I am about the company and the opportunity.

<here you can summarize why you are the right person for the job>

I am confident that my skills will allow me to come in and succeed as __________, and it’s a position I’d truly be excited to take on.

I’m looking forward to hearing from you about the next steps, and please don’t hesitate to contact me in the meantime if you have any questions.

Thank you again, and I hope to hear from you soon.

All the best,

<Your name here>

Studying

It never hurts to study up on management best practices in general. Don't assume that since you've managed for years, you know everything. Take a bit of time to organize your thoughts before the interview to improve the structure of your answers and relate them to basic management principles.

Pick one or two of your favorite management books and skim them to refresh yourself with the various aspects of management. At this page, I keep a list of Top Recommended Books for Engineering Managers.

Chapter 4: Example Behavioral and Competency Questions

Below I will list the questions I encountered with suggestions for how to approach and answer several of the most popular. No one asked all these, although some of them are popular. I didn’t write notes in the interviews, so these are mostly taken from memory.

I will give examples of good answers and also not so good answers to some of the most important but not all questions. I am not saying these are the absolute best answers ever. You might know a better answer and if so please let me know so others can benefit in the next update, see contact info.

You always want to give thoughtful, well-prepared answers. Some people write out answers to the most popular interview questions ahead of time to prepare. If you have time, that might be a good idea. Having your answers written out, whether electronically or on paper, allows you to refer to them in the future to practice and refine them. You can even share them with your friends also to review.

The good news about these questions is you can prepare for them ahead of time. There is no single correct answer as everyone's experience is unique.

General Questions

Frequently Asked and Important

Tell me about yourself.

This question is the most common opening question asked by about 90% of panelists. I recommend you have a smooth, well-rehearsed answer ready.

This answer is also sometimes called an "elevator pitch."

Why are they asking this? Maybe they didn't read the resume, or want you to summarize it? Perhaps it was not clear. Or this is just a conversation starter. In any event, the interviewer is not asking you to rehash your entire career history. They are asking how to interpret your resume and what is the bottom line to take away from it. Keeping it short also does not risk losing the interest of your interviewer. Your answer should be at most 2 minutes.

I recommend starting in **reverse chronological order** and summarizing your last three positions. You aim to leave the impression of how you progressed through roles of greater responsibility and achievements. I think it is best to limit it to your most recent experience because it is likely going to be weighted most heavily, and time is limited. Explain how these experiences relate to you what you bring to the table and professional goals.

You might start with something along the lines of, *"I am very passionate about the field of _____. In my last three positions, I lead teams working in _____. At <last company>, I lead a team of <whatever the number of engineers> to achieve X. Before that, I was <another company>, and my team(s)...."*

Why are you interested in this opportunity?

I recommend you research the company to prepare a strong, enthusiastic answer to this question. While it is a given that you may be there because you need a job, it is better to talk about why you are excited to work at the company. If you have used the company's products, this is a perfect opportunity to convey how passionate you are about the company or product. In general, people react well to hearing that others are fans of their work or company. This can start the interview off on a very positive note. If you are a referral mention that your friend(s) works there.

Tell me something that is not on your resume.

While this sounds like a very innocent question, if you are concerned about discrimination or ageism be careful what you answer.

The interviewer is asking you to tell them something that will impress them. You are going to need to make an informed guess based on what you know about them as to what might "impress" them. Ideally something memorable and positive. If the panelists are talking about you as the candidate who is an award-winning whatever that is great. It distinguishes you from the pack.

With that said, this is not the time to say e.g. that you like playing video games, or enjoy cooking, unless you know the interviewer is very into video games, or cooking themselves. In that case, this could be good for building a relationship and positive goodwill. It is ideal if you have some impressive achievement that shows you align with one of their company values. Safe and positive things could be your involvement in volunteer work or philanthropy. You could share that you won awards in college, high school, etc. but not if they were very long ago.

What are you not good at? [This is similar to what is your greatest weakness?]

The interviewer is fishing to see if you will give up something disqualifying you from the job but also how self-aware, honest and forthcoming you come across. Definitely prepare an answer to this common question. Everyone has limitations.

Don't say something cliche like, *"I work too hard"* or *"I'm a perfectionist."* Even if this is true, it will sound inauthentic and that you are hiding something or dodging the question. However, this is not the time to share your most profound darkest character flaws, either! The area of improvement you discuss should obviously not be a weakness in an area that is listed explicitly in the job description.

I recommend you pick a weakness where you have a plan to improve, so while it may be a weakness today, it may not always be one. An example might be you are not good at public speaking, and you are taking a program like Toastmasters to improve.

What is your management style? A common variant is also, "What is your management philosophy?"

This is a very broad and open-ended question. The interviewer is screening you to determine if your management approach is going to mesh well with what they understand works in the organization. They are trying to imagine if you can work with members of the existing team and other managers.

This question is a tremendous opportunity to engage the interviewer in a discussion about management and show that you have an excellent grasp of it. You want to give a confident and well thought out answer. If you immediately jump in and answer with details of your style, it might or might not match what they are looking for, and you may not pass this section.

Instead, take a step back. If you can involve your interviewer in a discussion about management principles and get them nodding their head in agreement, you are off to a better start. This discussion may also help build rapport. If you can even teach them something about management they had not considered, you will clearly distinguish yourself.

I would start by discussing what good management and the purpose of management is to you. Once you establish the aim of management, you can give your take on how you approach it and why you do what you do. Setting this background is better in my experience than just jumping into practices. I might open with something to the effect that the objective of the manager is to help the team win, and the manager should be a force multiplier for the team. You could also, for example, open with making an analogy to being the head coach of a professional sports team.

If you've done your research and know something about the company, you can highlight where you think your management style matches. For example, are decisions made by consensus, or is top-down? Is being able to work with remote workers important?

I think it is important not to leave this discussion at an entirely theoretical level. Give examples of how your management style has worked well, and how your teams have reacted well to your management style.

I would then list the general things I do to help the team win. You don't want to appear as someone who is rigid or dogmatic. I strongly recommend adding a caveat that a one-size-fits-all style doesn't work for everyone, and I must adapt my management style to each individual. For example, I would say that I aim to be hands-off, but can be very hands-on when the situation or individual requires it.

Do you have any questions for me?

This was asked by almost 100% of panelists. The only time this was not asked was if we ran out of time.

This is a reverse interview question and a critical one to hit out of the park. The interviewer is asking because **they are looking to grade you on the quality and kinds of questions you ask**. How interested do you seem in the job based on your questions? And are you applying for the right reasons? What is important to you?

It is super important to enthusiastically ask thoughtful questions. You must ask a question. Saying anything to the effect, “no questions” or asking trivial questions will fail this part of the interview. It is perfectly fine to re-ask the same questions you already asked a previous interviewer if you cannot think of anything! You likely may get a different answer plus again you are being evaluated on what you ask. Finally, of course, this is your opportunity to learn more about the company and if you want to take the job if it were offered.

Your interviewer has been controlling the line of questioning until now so this is your chance to present what you want to them. I recommend starting by thanking them for their time. Then quickly summarize why you think you would be a good fit for the role. You need to keep this short as you will often only have a few minutes left. Point out anything relevant to your experience that perhaps they didn't touch upon. **Think of this as your closing argument.**

It easier to come up with a few questions in advance rather than think of them on the spot. I like to ask questions that will leave the interviewer in a positive state of mind and show enthusiasm for the opportunity. I recommend avoiding any negative questions about what they don't like about working there or about anything negative in the news you might have read or heard about the company. You are unlikely to get an honest answer, and it will not help you in closing. Avoid asking HR related questions (benefits, compensation, etc.).

Here are some generic positive questions you can ask if you have nothing specific to ask.

- What do you like most about working here?
- What has been your biggest success while working here?
- What distinguishes successful managers here?
- Assuming you are speaking to the hiring manager, what are your immediate goals and priorities for this position?
- What constitutes success in this position in the next year? In the next two years?

Here is also a good list: How to answer 'Do you have any questions for me?': 25 great questions to ask your future boss

Other Questions You May Be Asked

1. What is the role of an engineering manager?
2. What in your mind are the responsibilities of a manager?
3. What was your purpose in moving into management?
4. What are you looking for in your next role? Or what would you like to do differently in your next role?
5. Why are you leaving your current role? [People are hunting if you left or were let go for some reason that will disqualify you. As long as you have a reasonable answer most people will let this line of questioning

drop. What doesn’t work here is to say bad things about your previous employer or manager.]

6. Where do you see yourself in 5 years? Or describe your ideal career progression.
7. What is the difference between leadership and management?
8. What is the largest team you have ever managed?
9. What size team are you looking to manage?
10. How big a team are you comfortable to manage?
11. What are the differences and tradeoffs between managing a small team and a large team?
12. What is the composition of your current (or last) team, and how is your team organized?
13. How is managing other managers different from managing individual contributors?
14. How do you know that you are doing a good job as a manager?
15. How would your current (or last) team describe you?
16. What would a report/peer/manager feedback on you be? Strengths, areas for development, etc.
17. How hands-on are you with the team? Are you involved in coding, design reviews, architecture, etc.?
18. How much are you coding on a daily basis?
19. Do you have experience managing remote teams or individuals? What is different about managing remote teams?
20. How did you organize your team? [I was even asked to draw out org charts of my previous teams, so you should be comfortable doing that and explaining why you organize your teams that way.]
21. What kinds of meetings do you hold to run your team?
22. If you could only have one meeting with your team what would it be? In other words, what is the most important meeting?
23. What is the hardest lesson you have learned as an engineering manager?
24. What would you say is the main responsibility or most important thing for an engineering manager to do?
25. What was your biggest failure?
26. Think of a mistake or failure you’ve made in the past two years. What did you learn from it and/or do differently in the future?

27. How would you assess if someone is a good manager?
28. Why is management attractive to you?
29. What is a professional area you are working on improving?
30. How have you grown as a manager over the last year?
31. How do you measure your success?
32. How do you make decisions regarding build versus buy?
33. How do you describe your job to people outside the industry?
34. Tell me when you got feedback about your team and had to make a change based on that feedback.
35. Tell me about a difficult decision you've made in the last year.
36. What kind of process have you followed in the past and what has worked well for you? What hasn't worked?
37. How do you institute the right process at the right time? How do you know when you need something more formal?
38. Have you ever removed a process? Why or why not?
39. Tell me how you helped drive a process change or cultural change across teams or organizations.
40. Tell me about a time you received guidance from your manager and had to get buy-in from your team.
41. Tell me about a time you had to lead a team through an organizational change.
42. Tell me why you transitioned from job X to job Y on your resume?
43. Explain to me the progression in your title.

Feedback and Performance Management

Frequently Asked and Important

How have you managed low performers?

This is the most popular question in this category by far. The interviewer is probably asking because, in their opinion, there are some low performers on the team, and they want to know you will be able to step in and deal with them effectively and compassionately. Or the interviewer is wise enough to recognize that all managers face this sooner or later, and they want to be sure you have and will handle it effectively.

You should have a good framework for dealing with low performers, including 30/60/90 day plans and when to go with PIPs, etc. I prepared multiple stories about when I was able to turn around low performers and also when I had to manage people out.

Tip: I recommend leaving out exactly when these examples happened. It is not relevant. I once got feedback that a story I shared was not in the opinion of the interviewer recent enough, and it left the impression that I did not manage out low performers aggressively or often enough. Ideally, therefore, it is best if you have recent stories or leave out the dates altogether.

Other Questions You May Be Asked

1. How do you manage and motivate high performers or star performers?
2. Tell me about a time you turned around a low performer.
3. How do you keep your people motivated?
4. What was some difficult feedback you had to give recently? And why was it hard to deliver?
5. What was some difficult feedback that you received? And why was it hard to receive?
6. How do you do coaching and career development?
7. Tell me about a few people on your team and the career development plans you created with them.
8. How do you coach engineers on your team that are smarter and better (more senior) engineers than you?
9. Have you ever promoted anyone?
10. How do you deal with difficult team members?
11. Have you ever had to fire someone?
12. Have you ever had to fire a friend?
13. What is the point of a one-on-one meeting (1:1)?
14. How often do you do 1:1 meetings?
15. What is the structure of your 1:1 meetings?
16. How do you deal with performance issues?

17. Have you ever had to implement a PIP (performance improvement plan)? What are your thoughts on it?
18. How do you identify engineers who could make good managers? How do you help them develop?
19. How do you handle a great engineer with communication problems?
20. How do you handle someone who really wants a promotion but isn't ready?
21. Tell me about the worst mistake someone on your team made and what happened
22. What core values do you bring to your teammates?

Prioritization and Execution

TIP: I recommend you balance how much you talk about what you did versus what your team, tech lead, et al. did. It's natural in an interview to talk about what you did and not what others did. They are asking about you, after all, right? However, I have seen this backfire in a manager interview. I've made that mistake and talked too much about my contributions and not enough about what others contributed. Engineering is a team sport so remember to give credit to your team.

Frequently Asked and Important

Tell me about a project you are most proud of, and why.

The specific project is not that important. The interviewer is in my experience more interested in understanding why that project was important to you.

A common follow up line of question can include is there anything you would have done differently if you had this project to do over. It is important to think of something to share that shows how you learned even from positive experiences.

How do you manage multiple requests for your team? How do you deal with competing priorities?

I think it is important to acknowledge that every engineering team gets more requests than it can ever do. This is completely normal. Do you have a framework for ranking requests? How do you push back on the requests that you cannot handle? When do you escalate? Here you can also speak about your planning process.

Other Questions You May Be Asked

1. Did you ever disagree with your product manager about the priority of tasks or if something needed to be done at all?
2. How do you see the tradeoffs between scope, quality, and schedule?
3. How do you work with product managers, UX team, QA, etc.?
4. Tell me about a time you exceeded expectations and went above and beyond.
5. Tell me a time when a project you were responsible for was late or not meeting expectations. What did you do?
6. Tell me a mistake you made that hurt the business.
7. How do you balance process versus execution?
8. What are signals that too much or too little process is in place?
9. How do you evaluate the success or failure of a project?
10. Explain to me the roles and responsibilities of the members of a Scrum team. (This company was very into Agile and Scrum.)
11. Tell me about a project that did NOT go as planned and what you learned from it.
12. Tell me about a time you had to make a difficult resource tradeoff.
13. What was your most successful product or project and why?
14. What was your worst product or project and why?
15. Explain your quarterly or long term planning.
16. How do you manage dependencies in your team and across teams?
17. How do you manage existing tech debt?
18. How involved were you in developing features and tech spec creation?
19. How do you ensure quality?

20. Tell me about a time when you had to go above and beyond the call of duty in order to get the job done.
21. Tell me about a time when you had too many things to do and you were required to prioritize your tasks.
22. Give me an example of a time when you had to make a split-second decision.
23. Give me an example of a time when something you tried to accomplish failed.
24. Give me an example of when you showed initiative and took the lead.
25. Give me an example of a time when you motivated others.
26. Tell me about a time when you delegated a project effectively.
27. Tell me about a time when you delegated work to someone on your team. How did you ensure it was done correctly, and on time, and how did you decide who to delegate it to?
28. Tell me about a time when you delegated a task to someone and they were not able to complete it; what did you learn and what was the impact of it?
29. How did you introduce a new process or technology?
30. Give me an example of a time when you used your fact-finding skills to solve a problem.
31. Tell me about a time when you missed an obvious solution to a problem.
32. Describe a time when you anticipated potential problems and developed preventive measures.
33. Tell me about a time when you were forced to make an unpopular decision.
34. Describe a time when you set your sights too high (or too low).

Strategy and Vision

1. Tell me how you communicated your roadmap.
2. How did you go about aligning the experience, skills, and expertise in your team to achieve the goals and vision laid out for your company/group?
3. Have there been occasions when you have not agreed with the strategy for how your organization/team must achieve the goal/vision? If so how

did you handle it?

4. What are some strategies for handling competing visions on how to execute a project?
5. Give me an example of a time when you set a goal and were able to meet or achieve it.

Hiring Talent and Building a Team

1. How do you build a team?
2. How did you build your current team?
3. What do you look for when hiring?
4. How do you work with your recruiters?
5. How do you recruit and hire in very competitive markets e.g. San Francisco Bay Area?
6. How much time do you spend hiring and sourcing candidates? What do you do?
7. Tell me when you took a risk hiring someone and how it worked out.
8. What are you looking for in an engineer when you recruit? Do you prefer to hire only experienced folks? Are you more comfortable with generalists or specialists?
9. How have you optimized the recruiting process in the past? Have you thought about the recruiting funnel and how you can optimize different parts of it?
10. What are your thoughts on structured interview loops versus unstructured free-form loops?
11. Tell me about a borderline case where you decided to hire or not hire someone, and how did you make the call.
12. How would you make your first hire, how would you make your 10 hires after that and the next 100 after that
13. When do you outsource or hire contractors?
14. How do you choose and negotiate with contractors and vendors?
15. Have you hired remote engineers?
16. Have you opened an office in a new city or country?
17. How do you spot burnout?

Working With Tech Leads, Team Leads and Technology

1. What is the role of a tech lead or team lead?
2. What is the relationship between the engineering manager and tech lead?
3. What if all your team is new and junior and you don't have a tech lead? What if no one on your team wants to be a tech lead, or do the things tech leads do?
4. How do you grow and develop tech leads?
5. Have you ever disagreed with one of your tech leads?
6. A tech lead on your team tells you "I want to be a manager." How do you respond?
7. What do you do, or say, if one of your engineers is really pushing hard for new, shiny (potentially risky) technology *(assume: you don't agree it is the right choice)*?
8. How would you convince a team to adopt a new technology?
9. How do you ensure (code) quality and keep a lid on tech debt?
10. How do you establish ownership in your teams?

Dealing With Conflicts

Most people are not comfortable with conflicts. As a people manager, you will inevitably encounter lots of conflicts, and it is part of the job. Do you come with humility to listen and understand, or to always win the argument? Do you rely on data? Do you have any framework? How willing are you to compromise to keep harmony?

Really great managers are skilled in the art of difficult conversations. There are lots of books and training classes on this. Some good books include:

- Radical Candor: Be a Kick-Ass Boss Without Losing Your Humanity
- Crucial Conversations Tools for Talking When Stakes Are High
- Difficult Conversations: How To Discuss What Matters Most
- Conversational Intelligence: How Great Leaders Build Trust & Get Extraordinary Results

Finally, prepare a story where you were in the wrong and what you learned from it. A good story would be e.g. there was a disagreement about priorities. You met with the other person and heard their point of view. After learning some new information, you both aligned and the conflict was resolved. This shows that you are flexible and open-minded when new information is presented. Being in the wrong due to bad information is completely understandable.

Frequently Asked and Important

Tell me about a time you had a conflict with another engineering manager and how you resolved it.

Tell me about a time you did not see eye to eye with your manager and how you resolved it.

Tell me about a time you had a conflict with a product manager and how you resolved it.

These are an opportunity to show your influence and negotiation skills. It is important to show leadership and how you can stand your ground without being too stubborn. When answering these questions it is important to not tell stories where the conflicts quickly went into escalation or were resolved by disagreeing and commit. Those kinds of resolutions do not highlight either influence or leadership.

Other Questions You May Be Asked

1. Tell me about a time there was a conflict between members of your team and how you resolved it.
2. Tell me about a recent situation in which you had to deal with a very upset customer or co-worker.
3. Tell me about a time when you had to use your presentation skills to influence someone’s opinion.

4. Give me an example of a time when you had to conform to a policy with which you did not agree.
5. What is your typical way of dealing with conflict? Give me an example.
6. Tell me about a time you were able to successfully deal with another person even when that individual may not have personally liked you (or vice versa).
7. Describe a situation in which you were able to use persuasion to successfully convince someone to see things your way.
8. Describe a time when you were faced with a stressful situation that demonstrated your coping skills.

Diversity and Inclusion

This is a hot topic in high tech and most companies are looking for ways to increase their diversity. You should be prepared to speak about why this is important and share any personal stories. Here are some articles about diversity. Why Workplace Diversity Is So Important, And Why It's So Hard To Achieve, and Research: When Gender Diversity Makes Firms More Productive.

1. Tell me about the diversity of your team.
2. What do you do to ensure you have diversity?
3. What do you tell people when they come to you and ask why diversity is important?
4. Tell me about your last diversity hire or promotion and why did you hire or promote that person.

Contributors and Acknowledgements

I would like to thank the following people for contributing to and reviewing this document.

- Tom Graber
- Vikram Karumbaiah
- Sribharathi Modukuru
- Maggie Graupera
- Ashok Ramaraj
- Dmitry Yesin
- Anna Winkler
- Kiran Hegde
- Joao Antunes
- Noor-E-Gagan Singh
- Michael Lee
- Tim Worboys
- Sanjay Zope
- Amy Woodward
- Ron Lichty

Photos by Amy Hirschi, NESA by Makers and Evan Dennis on Unsplash

About the Author

Vidal Graupera is an experienced engineering leader and author. Vidal is an Engineering Manager at LinkedIn. Previously he was an engineering leader at Uber, Autodesk, and WalmartLabs.

He hosts the popular "Managers Club" website and podcast, where you will find interviews with successful engineering leaders and managers at https://www.managersclub.com.

Vidal holds engineering degrees from Carnegie Mellon University and the University of Southern CA, and also an MBA from Santa Clara University. He served in the US Navy Reserve as an Engineering Duty Officer. A son of Cuban immigrants, Vidal was born in Miami, FL.

Vidal is a co-author of "Pro Smartphone Cross-Platform Development: iPhone, Blackberry, Windows Mobile and Android Development and Distribution" from Apress, and author of "Engineering Leadership Interviews: Lessons Learned and Best Practices Gathered from Interviews with Top Engineering Leaders Around the World."

Contact Info

Reach out to me if you have any questions or feedback on this guide, or if you need help preparing for an interview. I invite you to join the community for Engineering Managers at my website: https://www.managersclub.com.

Email: vgraupera@gmail.com

LinkedIn: https://www.linkedin.com/in/vidalgraupera/

Website: https://managersclub.com

Twitter: @vgraupera