



# Welcome to Meta

# Our Mission



Meta's mission to give people the power to build community and bring the world closer together. We're looking for talented people who are driven to build new things and bold enough to tackle complex issues that make the world better for everyone. We move fast at Meta, so you'll have the opportunity to make an impact on your very first day and every day after.

**Move  
Fast**

**Be  
Bold**

We are bringing the world closer together in real time. Our teams are small and nimble. We work quickly and collaboratively to build smarter, more meaningful solutions on a global scale.

**Be  
Yourself**



Be unique. Be authentic. However you prefer to say it, we really mean it. Our culture embraces people's diverse perspectives and creates a positive environment where everyone belongs.

# Your Interview Practice

## WATCH:

### Video Tutorial

Meta Mock Interview Prep:  
(Password: 2018prep)



## Coding & Interview Practice

- [Example Interview Questions](#)
- [GeeksforGeeks](#)
- [Cracking the coding interview](#)
- [Meta Coding Portal](#)
- [Meta Engineering](#)

## Meta GitHub

- [Meta Incubator](#)
- [Meta GitHub](#)

# Your Interview Practice

## WATCH:

### Video Tutorial

Cracking the Meta Coding Interview:  
(Password: FB\_IPS)



## Coding & Interview Practice

- [Career Cup](#)
- [CodeChef](#)
- [Project Euler](#)

## Design Interview Prep

- [Github Systems Design Primer](#)
- [HiredInTech](#)
- [Grokking the System Design Interview](#)



# Interview Tips

**Prepare** for technical questions involving coding or algorithms (in your preferred programming language), design patterns, and specific questions to your background. It may also help to review core computer science concepts (data structures, binary trees, linked lists, object-oriented analysis/design, etc.) as well as subjects pertaining to the scale of our technology.

**Practice** writing code and designing systems/networks by hand on paper or a whiteboard. During the interview, explain your thought process out loud. Create a working solution (even if it feels inefficient) and then iterate rather than immediately trying to jump to the clever solution. Tip: If you cannot explain your solution clearly in 5 minutes, it's probably too complex for the interview question at hand.

**Discuss** your solution. If the interviewer gives you hints, be ready to discuss them against other options and implement them if appropriate. It is good to adjust and work through the problems with the interviewer to show your thought process and problem solving ability.

**Complete** your solution. Done is better than perfect, but be able to identify any bugs and assumptions. The interviewer will be looking for an intersection between speed and accuracy.

**Pace Yourself** The duration of the interviews will be ~45 minutes. It helps to time yourself when practicing at home. If your answers are taking longer, aim for simpler solutions.



# Full Interview Loop

## Technical Project / Project Retrospective

Using example(s) from your past, discuss the technical skills needed to deliver a project from start to finish. Showcase success stories, and get to the situations that show how you as a leader navigate the complexity of developing, supporting, and scaling both your people and your teams.

## Coding Interview

Solve a basic computer science problem. The problem will cover basics of algorithms, data structures, design patterns, system design, complexity and basic coding. Many candidates are most concerned with the coding question — especially those who have not coded in many years. Our goal is not to see if you can still write production-quality syntactically perfect code; it is to see how you apply basic CS principles to solve a concrete problem using a language of your choice. Take hints from the interviewer and be open to other solutions as you go. The interviewer will assess efficiency, structure, syntax, bugs, and working code.

## Cross Functional

The Purpose of the XFN Partnership & Communications interview is to assess a candidate's capability to establish and maintain Cross-Functional Partnerships("XFN") and Communicate effectively across the company.

## Design Interview

This interview will test your abilities to design systems/processes that scale in Meta's environment. Explore the design of a complex system and the trade-offs within a design. The scope of the question can vary widely; it's a challenging and deep technical discussion around product ideas, usability issues, scalability, data structures and technologies used. The interviewer may give you a system we currently have in place and have you make suggestions on how it can be improved. For this interview, there is no right or wrong answer; the interviewer wants to observe how you design and architect a system.

## Career Conversation

You'll meet with one of the Engineering leaders or Sr. IC's, and they will ask you questions about any significant projects you have worked on to get a feel of your capabilities and how you work on teams and with other groups across an organization. There may also be questions to learn more about how you've dealt with conflicts in the past or describe or to describe a project you worked on that brings you pride.



# Qualities We Look For In Engineers

## Ownership of projects from start to finish

Meta engineers are always building new things, and have a demonstrated ability of owning projects from start to finish. Our engineers are able to take on every aspect of a project – from design and development to implementation and support.

## Ability to thrive in a flexible, flat environment

Meta has a relatively flat organization, and all of our engineers are flexible and agile enough to stop what they're doing and jump into code.

## Daringness to be bold

Meta engineers have consistently taken on challenging projects throughout their education and careers. When you speak with engineers on our teams, you'll hear stories about times when they weren't quite sure about how they were going to accomplish a project, but dared to be bold, iterated quickly and found a solution. Our engineers love working on the most difficult challenges. They embrace uncertainty, take risks and learn from failure.



# Coding Interview (One 45 minute interview)

## What does the interview cover?

Be prepared for 1-2 basic coding questions covering CS coding, algorithms, data structures, design patterns, system design and scalability. You will use Coderpad (execution disabled) for your solution using a programming language of your choice. Solutions should focus on the basics in each area, not the more advanced and/or esoteric computer science skills — e.g., we are not focusing on syntactically perfect code nor advanced language usage (e.g., C++ templates, multiple inheritance).

## How to prep for the interview

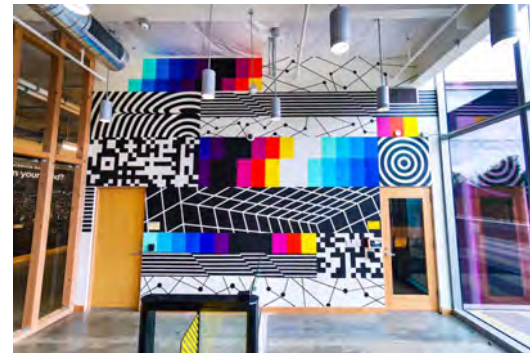
- Review basic problem solving skills: how do you break down a problem, how do you map a problem to practice, how do you express a solution in code, how do you test, what is a good API (and why), what are the trade-offs you are making in your solution?
- Practice using your favorite language to solve some basic programming problems (e.g., print a binary tree in-order, add two binary numbers passed to a function as a string with the function call: `string AddNumbers(string firstNum, string secondNum)`). Practice away from the computer, writing code and thinking out loud as you try to solve the problem, using basic data set(s) to test and debug your solution. Do NOT learn a new language just for the interview.

## What to do in the interview

- Think out loud as you work through a solution, as the engineer/interviewer will want to know how you approach and troubleshoot problems. If the interviewer gives you hints to improve your code, take them and run with it; it's good to adjust and work through problems to show the interviewer your problem-solving abilities. Don't try to fake it nor worry about speed. But we do expect you to be able to solve 1 or 2 basic coding problems in 45 minutes.

## Interview Assessment Criteria

- **Communication:** Think out loud while writing code; ask questions to confirm understanding.
- **Problem solving:** Check your approach before writing code.
- **Coding:** Check for bugs; try to not over-complicate your code.
- **Complexity analysis:** Time and space complexity.
- **Debugging:** Check your code proactively.





# Design Interview [AI/ML] (Two 45 minute interviews)

## What does the interview cover?

This is a domain expertise interview based on machine learning algorithms.

Sample questions could include:

- Build a ranking or recommender system
- Design a entity matching based on taxonomy system
- Design a face tagging system based on computer vision & Image

## How to prep for the interview

- Drive the conversation. Keep improving. Own the design, and not just to answer our questions.
- Prepare to answer the why's in any important choices they made. "That's how we did it" or "I saw xxx did it" is not enough. Conscious about the cost of each decision, the trade-off they are making, and why it's a good one for the situation.
- If it's a research/specialist role, prepare to deep dive into expertise matters (they should already know what this means).
- If it's a product oriented role, be able to make strategic connections between the design and the product.
- Think beyond the status quo. Think about fundamental flaws of what others might take for granted, and have an opinion towards the future, and be able to defend it.

## Here are some resources to help you prepare:

- [Design a Recommendation System](#)
- [Machine Learning System Design](#)
- [Build an Activity Feed](#)
- [Machine Learning Lecture](#)
- [Machine Learning Interview Questions](#)



# Technical Project Interview (One 45 minute interview)



This is a discussion-based interview covering the architecture and implementation of a technical project. You will present a project that you have previously worked on, providing enough detail to explain the project at both high and low levels. This will take the form of a discussion and not a presentation. Slide decks are not needed and discouraged, but code examples and architectural diagrams are reasonable.

## What does the interview cover?

Using an example from your past, you are asked to discuss the technical skills needed to deliver a project. Areas we cover include:

- Setting context - The purpose of the project/product (e.g., we needed to build a new file system that scaled to 100EB because we were entering the photo-sharing market).
- A high-level description of product/project requirements.
- A fairly in-depth discussion of the architecture and the trade-offs that were made in designing the system. Clarity with detail is important here — this is the key part of the interview.
- Answers to questions as the interview will dig into the details of the architecture to see how deep you can go with respect to the various design points. For example: in a distributed system we might ask what types of communications protocols you used, how they worked, etc.

## How to prep for the interview

Recall one or two different projects you have been responsible for and map out the various aspects of the project, focusing on how best to describe both the key high and low-levels of the project.

## What to do in the interview

With only 40 minutes, it's important to have a concise technical description. If you are describing a large project (e.g., building a new rocket ship to mars), you might need to focus on specific areas to ensure you can cover both high and low-level details.

## Interview assessment criteria

- Demonstrate breadth and depth in system design.
- Demonstrate clarity of thought in presenting complex systems.
- Demonstrate good judgment in design trade-offs both in your project description and on-the-fly as we ask questions about “what if you did things a different way?”.



# Cross Functional Interview (One 45 minute interview)

## What does the interview cover?

The purpose of this interview is to assess your capability to establish and maintain cross-functional partnerships (“XFN”) and communicate effectively across the company.

The team is interested in how you think about and interact with other people on other teams inside and outside your broader org. We like this to be “here’s a story about a time when I worked cross team or cross org - including both things that went well and things that could have gone better.”

A typical starting point is: “Walk me through a project or initiative you have lead that required collaboration across multiple functions. How did you ensure the effectiveness of the overall project team? How did you keep the overall team informed and aligned?”

You’ve probably picked up that we’re big on self-awareness, so it’s really helpful here to be able to ‘fess up to working relationships you screwed up and learned from, conflicts you couldn’t resolve, and anything else that helps round out our picture of you. Can you work across the company? Can you handle different approaches, agendas and facilitate positive partnerships? How do you manage difficult partnerships?

## How to prep for the interview

- Think about a couple examples from your past.
- Make sure to have a pithy setup to describe the situation and provide important context.
- Think about what you learned from the XFN experience(s) and how that has changed how you now approach collaboration.

## How not to prep for the interview

Books are not going to help here. We are looking for your history, your story, and how you approach XFN collaboration.



# Career Interview (One 45 minute interview)

## What does the interview cover?

The purpose of the career conversation interview is to assess your capability in supporting organizational health and influencing the technical direction of your project/company. Showcase success stories, and get to the situations that show how you as a leader has navigated the complex business problems that affected the company at large.

## How to prep for the interview

- This interview will focus on you and your history, your resumé, and your motivation. Take the time to review your own resume as your interviewer will almost certainly ask about key events in your work history.
- Support each answer with examples or anecdotes.
- Use the STAR methodology to structure your answers.
- Be humble and focus on team work, leadership and mentorship qualities. Be open and honest about your successes and failures.
- Practice solid answers for the following questions (picked from the article) well ahead of time and review the day before the interview:
  - How do you deal with conflict?
  - What were some excellent collaborations you've had?
  - Can you tell me about four people whose careers you have fundamentally improved?
  - Describe a few of your peers at your company and what type of relationship you have with each of them.
  - What did you do on your very best day at work?
  - What does office politics mean to you, and do you see politics as your job?
  - Tell me about a project that you led that failed. Why did it fail and what did you learn?

## How not to prep for the interview

- Assume that since you've been doing this for years, you know everything off the top of your head. Taking a bit of time to organize your thoughts before the interview can greatly improve the structure of your answers, allowing you to relate them back to basic leadership principles AND demonstrate your experience by coupling them with solid examples from your past.

