Congratulations on making it to virtual onsite interviews! In this document you will find everything you need to prepare, practice and feel confident for your upcoming interviews. Let your Recruiter know if you have any questions or concerns not addressed in this document.

Web Solutions Engineer

Virtual Interview Prep Guide

Please use this guide as a resource to aid to help you prepare for your upcoming technical onsite interviews. As a reminder, all of the information in this document is confidential.

What you can expect: Day-of Logistics

Google maintains a casual office environment. Please dress comfortably, a suit is not required, nor expected. We recommend dressing business casual.

You will meet with **3 interviewers** for 60 minutes each. The meeting topics span across various areas that will be defined throughout this document. There may be scheduled breaks but if you need additional time in between interviews, please alert your Recruiting Coordinator.

For a typical virtual onsite, you can expect to meet:

- Web Solutions Engineers (focussing on web development/system design)
- Software Engineers (focussing on coding/data structures/algorithms)
- 1 Hiring Manager

Please note: there will be one Hangout link for you to use. Interviewers will join virtually at their designated time. For the technical and coding portions of your interviews, you will be expected to utilized shared Google Docs, which your Recruiting Coordinator will provide in your confirmation email. *Please let us know **in advance** if you require other accommodations.*

What we look for: 4 Key Attributes

Throughout your onsite interviews, you will be asked a variety of questions that are designed to assess your strengths in the core attributes listed below.

General Cognitive Ability	Leadership	Role-Related Knowledge	Googleyness
We ask open-ended questions to learn how you approach and solve problems. And there's no one right answer—your ability to explain your thought process and how you use data to inform decisions is what's most important.	*Be prepared to discuss how you have used communication and decision-making skills to mobilize others. This might be by stepping up to a leadership role at work or with an organization or by helping a team succeed when you weren't officially the leader.*	*We're interested in how your strengths combine with your experience to drive impact. We don't just look for how you can contribute today, but how you can grow into different roles, including ones that haven't even been invented yet.*	*Share how you work individually and on a team, how you help others, how you navigate ambiguity, and how you push yourself to grow outside of your comfort zone.*

Web Technology

You will meet with 1 current gTech Web Solutions Engineers. They will not necessarily be on your team, but outside of their technical evaluation, you can also get an understanding of a typical "day in the life" of a WSE.

The primary focus of these interviews is to assess your technical abilities outside of coding: webtech, operating systems and technical troubleshooting. The format is similar to the Hangout interview you completed. ***Please continue to leverage the materials sent to you to prepare for the hangout!*** They are full of technical information and resources that will set you up for onsite success.

Web Technology: This will be more in depth than the Hangout. The interviewers will be looking for you to have solid theoretical knowledge of Front-End programming languages (HTML, CSS, JavaScript), of protocols (HTTP and TCP/IP), and a general understanding of how the internet works and how web pages work. ***Think about:***

- *How data is transferred (front and back transfer, nuances, alternatives, transfers between websites, etc.)*
- *Working in asynchronous paradigms*

Keep in mind...

Be detailed but concise in your answers. This is especially the case for your knowledge of Web Tech - try to avoid only superficial answers and having to be prompted.
Think about real world scenarios/time that you've had to solve an issue like the one that you're being presented.
Remember to refer to the hangout prep guide for additional resources!
Think out loud!

- *Applying your knowledge to client-specific information (debugging, compatibility, UI/UX)*

Keep in mind...

Be detailed but concise in your answers. This is especially the case for your knowledge of Web Tech - try to avoid only superficial answers and having to be prompted.
Think about real world scenarios/time that you've had to solve an issue like the one that you're being presented.
Remember to refer to the hangout prep guide for additional resources!
Think out loud!

Software Engineering

The interview will include topics such as coding, data structures, algorithms and computer science theory. You should know at least one language really well, preferably ***Python or Java***. The interviewer will be interested in your knowledge of the following:

Coding practice: You can find sample coding questions on sites like CodeLab, Quora, and Stack Overflow. The book "Cracking the Coding Interview" is also a good resource. Be sure to test your code and ensure it's easily readable without bugs. Don't stress about small syntactical errors like what the correct parameter order is for a substring method (e.g. start, end or start, length) — just pick one and let your interviewer know. You should know at least one programming language really well, preferably Java or Python. You will be expected to know APIs, Object Oriented Design and Programming, how to test your code, as well as come up with corner cases and edge cases for code.

- Note that we focus on conceptual understanding rather than memorization.
- How well does your code work?

Algorithms: Approach the problem with both bottom-up and top-down designs. You will be expected to know the complexity of an algorithm and how you can improve/change it. Algorithms that are used to solve Google problems include sorting, searching, divide-and-conquer, dynamic programming/memoization or algorithms linked to a specific data structure. Know Big-O notations (e.g. run time complexity) and be ready to discuss complexity of algorithms like Dijkstra and A*. We recommend discussing or outlining the algorithm you have in mind before writing code. Be familiar with common sorting functions and on what kind of input data they're efficient on or not. Think about efficiency means in terms of runtime and space used. For example, in exceptional cases insertion-sort or radix-sort are much better than the generic QuickSort/MergeSort/HeapSort answers.

Graphs: Consider if a problem can be applied with graph algorithms like distance, search, connectivity, cycle-detection, etc. There are three basic ways to represent a graph in memory (objects and pointers, matrix, and adjacency list) — familiarize yourself with each representation and its pros and cons. You should know the basic graph traversal algorithms, breadth-first search and depth-first search. Know their computational complexity, their tradeoffs and how to implement them in real code.

Data Structures: You should study up on as many data structures as possible. Data structures most frequently used are arrays, linked lists, stacks, queues, hash-sets, hash-maps, hash-tables, dictionary, trees and binary trees, heaps and graphs. You should know the data structure inside out, and what algorithms tend to go along with each data structure.

Mathematics: Some interviewers ask basic discrete math questions. This is more prevalent at Google than at other companies because counting problems, probability problems and other Discrete Math 101 situations surround us. Spend some time before the interview refreshing your memory on (or teaching yourself) the essentials of elementary probability theory and combinatorics. You should be familiar with n-choose-k problems and their ilk.

Recursion: Many coding problems involve thinking recursively and potentially coding a recursive solution. Use recursion to find more elegant solutions to problems that can be solved iteratively.

Computer science principles: Data structures / algorithms and how they can be used in your solutions. As well as studying computer science principles, be prepared to discuss algorithms in depth - how complex is an algorithm, how to optimise it.

Analytical Skills: Analysis of your own code (can you spot a bug, can you tell the interviewer what is the most optimal solution?) and analysis of the question (do you understand it, do you ask clarifying questions)

Problem Solving Skills: They want you to think out loud. Tell them what you are doing and what steps you are taking to solve the problem.

*****Be sure to refer back to the hangout prep guide for additional resources*****

Web Technology and Coding are the most important and you would be checked with a lot of depth in these 2 areas. You might also be checked on your basics for the following :

Troubleshooting: You will be given scenarios, again using a combination of behavioral and hypothetical questions where the interviewers want to see the steps you take to solve a problem. This will give them a good idea of how you would approach solving problems for clients. The interviewers may be purposely vague to encourage you to ask for more information and to clarify unknowns. Be prepared for open ended questions and to give specific examples or rationales for your decisions.

Databases: You may be probed about your knowledge of SQL, specifically around syntax and commands, as well as other database languages, big data and data analysis. You should be prepared to write complex queries. This Resource will help you prepare.

Keep in mind...

Be detailed but concise in your answers. This is especially the case for your knowledge of Web Tech - try to avoid only superficial answers and having to be prompted.
Think about real world scenarios/time that you've had to solve an issue like the one that you're being presented.
Remember to refer to the hangout prep guide for additional resources!
Think out loud!

Analytical Skills: Ability to analyse data and present it back to an audience, experience of doing this and having the ability to spot bugs and analyse and correct your own work.

Keep in mind...

Be detailed but concise in your answers. This is especially the case for your knowledge of Web Tech - try to avoid only superficial answers and having to be prompted.
Think about real world scenarios/time that you've had to solve an issue like the one that you're being presented.
Remember to refer to the hangout prep guide for additional resources!
Think out loud!

Hiring Manager Interview

Typically, this interview is conducted by the Hiring Manager you will be working directly with at Google. This interview will focus on your ***leadership qualities, thought processes, and communication skills*.** You will also have a chance to learn more about your potential team and projects. In addition, you will have the opportunity to learn about the types of customers you would be supporting, projects you'll be working on and what a "typical" week looks like (*note, no week is the same as the last!*)

We're looking for Googlers who do the right thing, respect diversity of thought and who want to continue to develop. Below are the primary areas that will come up during this interview:

Leadership: Experience leading projects and/or teams. Showing thought leadership, initiative and influencing cross functional/senior stakeholders without official authority. Leadership at Google is not always just about being able to manage a team, but to be able to show leadership skills in general, or in projects and in a team.

How You Think/Decision Making: Questions designed to see how you think on your feet and your approach to dealing with complexity and ambiguity. Ability to use data in your decision making and to make difficult decisions. How do you make decisions and what do you use to make informed decisions (i.e data, consultation etc).

Fit: The Hiring Manager wants to know you are the best person for this job. Can they envision working and managing you? Why are you passionate about this role? What can you bring to this role immediately? How do you think that you will fit into the team? Why are you specifically, the right person for this role?

Project & TIme Management: Web Solutions Engineers support client projects from pre-Sales through ongoing post-Sales support. Think of scenarios that demonstrate your ability to manage multiple, maybe conflicting deadlines. Your communication skills need to be strong and to the point.

Be prepared for a mix of behavioral and hypothetical questions:

Behavioral questions looks at how you've handled a specific challenge in the past to assess if you'll be a good match for the role.

Ex: Give me an example of how you utilized your analytical ability to solve a problem at work. What was the result?

Hypothetical questions are designed to uncover how you would think or handle a specific challenge or situation that you have not yet encountered.

Ex: Suppose a colleague comes to you with [abc business problem]. Walk me through the steps that you would take in order to [xyz]. How would you approach this task? What types of data and metrics might you want to gather before determining your strategy? How would you evaluate whether your approach was successful?

Keep in mind…

Though this will largely focus on what was mentioned above, the Hiring Manager will also be asking technical questions as well, so be prepared to answer those.
Don't be afraid to ask questions! Ensure you clarify questions you don't fully understand, and ask for more data if you need it. Make assumptions if you need to, but make them out loud.
We aren't always looking at the end result. Often, we are trying to see the steps you take to get to a solution, the reasoning and decisions you make (essentially, the journey is just as important as the ending).

Tips

- ➔ **Predict the future:** You can anticipate 90% of the interview questions you're going to get. "Why do you want this job?" "What's a tough problem you've solved?" If you can't think of any, Google "most common interview questions." Write down the top 20 questions you think you'll get.
- ➔ **Plan:** For every question on your list, write down your answer. That will help them stick in your brain, which is important because you want your answers to be automatic.
- ➔ **Have a backup plan:** Actually, for every question, write down THREE answers. Why three? You need to have a different, equally good answer for every question. You want the next interviewer to hear a different story and become your advocate.
- ➔ **Explain:** We want to understand how you think, so explain your thought process and decision making throughout the interview. Remember we're not only evaluating your technical ability, but also how you approach problems and try to solve them. Explicitly state and check assumptions with your interviewer to ensure they are reasonable.
- ➔ **Be data-driven:** Every question should be answered with a story that demonstrates you can do what you're being asked about. "How do you lead?" should be answered with "I'm a collaborative/decisive/whatever leader. Let me tell you about the time I … "
- ➔ **Clarify:** Many of the questions will be deliberately open-ended to provide insight into what categories and information you value within the technological puzzle. We're looking to see how you engage with the problem and your primary method for solving it. Be sure to talk through your thought process and feel free to ask specific questions if you need clarification.
- ➔ **Improve:** Think about ways to improve the solution you present. It's worthwhile to think out loud about your initial thoughts to a question. In many cases, your first answer may need some refining and further explanation. If necessary, start with the brute force solution and improve on it — just let the interviewer know that's what you're doing and why.
- ➔ **Practice:** Everyone gets better with practice. Practice your interview answers—out loud—until you can tell each story clearly and concisely.
- ➔ **Speak Up!** Talk through your thought process about the questions you are asked. In all of Google's interviews, our engineers are evaluating not only your technical abilities but also how you approach problems and how you try to solve them.

About Google's Products and Services

Google Ads

Google Marketing Platform

Google Display Campaigns

Youtube

Google Products

About our Company

Google

The Google story

Life Google