

INTERVIEW PREP KIT

Software Quality Engineer, Platforms, Mastercard Data and Services

MAY 2021



Table of Contents

What to expect.....	2
First round interview	2
Final round interview	2
How to prepare	3
Tips for the interview.....	3
Background	3
Communication.....	3
Navigating a remote interview	4
Practice problems	5
Practice interview #1: Testing weather.com	5
Narrow down the scope of the question	5
Ask clarifying questions	5
State your assumptions	5
Break the problem into components	5
Practice interview #2: Clock Angle	5
Ask clarifying questions	5
State your assumptions	6
Break the problem into components	6



Introduction

We are excited that you are considering the Software Quality Engineering role at Mastercard! This document is an optional but highly recommended resource to familiarize yourself with technical interviews. Our technical interviews are purposefully standard, so this advice should serve you well when interviewing at Mastercard or other companies.

What to expect

Every interview begins with introductions and ends with a chance for you to ask questions to get your interviewer's perspective about work and life at Mastercard.

First round interview

The first round will consist of two 45-minute back-to-back interviews. Each interview will have the same format:

- **Background discussion:** The first 10 minutes will be your chance to introduce yourself. We will ask a number of questions about your experiences.
 - The goal of this discussion is to evaluate your communication skills, for example:
 - Ability to precisely and clearly explain what you're talking about and respond to what is being asked
 - Ability to communicate with appropriate context, e.g. explaining an internship to an interviewer who is not familiar with the company or their code
 - Depth/breadth of understanding, e. g. knowing the broader business context behind your project or important details of a technology you learned
 - Ability to reflect on your experiences and see how things like the problem solving process, communication, or group dynamics had an impact
- **Technical Question:** The next 30 minutes will consist of 1 technical question with multiple layers.
 - The goal of this question is to see how you solve analytical problems, come up with edge cases, and consider business context
 - The question will often have multiple layers. It will be straightforward when initially presented. As you progress, your interviewer will add further constraints or twists
 - You will not be asked to write code, nor are you expected to have a deep understanding of any coding languages

Applying for an internship? Usually within a few days, but always within two weeks, we will be in touch about whether you will be getting an offer.

Applying for a full-time role? Usually within a few days, but always within two weeks, we will be in touch about whether you will progress to the final round interviews

Final round interview

For full-time roles, there will be a final round consisting of two one-hour back-to-back interviews, each with two interviewers. Unlike the first round, each interview will have a different format:

- **Mostly-technical interview**
 - **Technical Questions:** This interview will have 2 separate technical questions, each taking 25 minutes. It will be similar to the first round interviews.
- **Mostly-background interview**



- **Extended background discussion:** The next 25 minutes will be a conversation about your experiences including classes you have taken, internships, extracurriculars, side projects, etc.
 - The goal of this discussion is to evaluate your technical communication skills – e.g. how clearly can you explain a technical problem encountered at an internship?
 - This conversation will be based on the content of your resume, so come prepared to chat about any line item on your resume.
- **Technical Question:** This part will be identical to the first round interview, a single 25-minute technical question.

Usually within a few days, but always within two weeks, we will be in touch about whether you will be getting an offer.

How to prepare

Since more than half of your interview duration will be spent on the technical questions, it makes sense to focus your preparation on knocking these out of the park.

- **Practice problem solving instead of memorizing gotchas.** We try to avoid asking "gotcha" questions, which require knowledge of an obscure algorithm or a very specific algorithm. Your time will be best-spent practicing soft skills like talking through your thought process and general problem solving with code through doing practice questions.
- **Practice aloud.** While practicing technical questions is important, it is a good investment of time to practice the soft skills of interviewing too – identifying when to ask clarifying questions, walking through your ideas and explaining your reasoning out loud, etc.

Tips for the interview

Background

- **Review your own resume.** Take the time to review your own resume. You should be able to talk about any line item on your resume (e.g. class assignments, internship experience, a hackathon project, etc.) and provide specific examples about what you took away from the experience.
- **Share concrete examples or anecdotes.** Your answers should be backed up with concrete examples of what you learned / took away from an experience.
- **Familiarize yourself with the SQE role.** What about software excites you? What excites you about the SQE role in particular? Why Mastercard?

Communication

- **Think out loud.** We care about how you get to your solution, sometimes more than whether your solution is optimal or fully exhaustive. If you voice your thoughts, your interviewer can better follow your thought process and better provide hints if needed. For example, how are you planning to break apart the problem? Are you going to go into detail about a certain aspect later? This is also good to do since the interviewer may tell you not to worry about a specific concern, and may want to prompt you to dig further into something you glossed over.
- **Ask for clarification.** Remember we are not only evaluating your technical ability, but also how you approach open-ended problems. Be sure to voice your assumptions and ask clarifying questions.
- **Ask questions!** The interview is a chance for you to learn more about the company, position, and team as well!



Navigating a remote interview

Both rounds of interviews are being conducted remotely, so it is good to prepare for that eventuality.

- **Pick a good interview spot.** Choose a place that will be quiet and have a consistent internet connection.
- **Speak clearly.** Despite advances in video calls, it can still be difficult to hear one another. Ensure you are speaking clearly and likewise, if you cannot hear the interviewer clearly, let them know so they can accommodate! Use headphones if at all possible, and set them up and test ahead of time.
- **Do not panic!** We have all been in your shoes. Technical issues happen in our personal and work lives, but your interviewer is there to help sort any of them out.



Practice problems

These are adaptations of **real** questions we asked during previous interviews.

Practice interview #1: Testing weather.com

Imagine the team is building weather.com. How would you test the website?

Narrow down the scope of the question

Weather.com has a lot of content! Narrow it down to just a few parts to start with, such as the hour-by-hour forecast page.

Ask clarifying questions

For example:

- Is this the desktop site or mobile app?
- Should I test the source of the data being shown on the page, or just that the page itself is working?

State your assumptions

Make your assumptions clear! That will demonstrate that you realize the problem can always be bigger, and the interviewer can ask you to go into more detail if needed. For example, you can state that you are assuming things like:

- The website only works for the US
- We're only going to focus on imperial units
- Search box X and search box Y are probably implemented in the same way, so you only need to test one of them

Break the problem into components

Focus on one aspect at a time. This can include how you would test:

- The search functionality
- UI elements, such as buttons, fields, dropdowns, etc.
- Navigation
- Confidence in the temperature values on the page

Practice interview #2: Clock Angle

We have a function that takes the hour and minute as two separate inputs and outputs the angle between the hour and minute hand. How would you test it?

Ask clarifying questions

For example:

- Is this a 12 hour or 24 hour clock?



State your assumptions

For example:

- It's a 12 hour clock
- We always return the smallest angle between the hour and minute hand

Break the problem into components

Focus on one aspect at a time. For example:

- "Normal" cases (e.g. 1:00, 2:00, 3:00)
- Edge cases (e.g. what is the angle for 9:00? 90 degrees or 270 degrees?)
- Invalid cases (e.g. 25:00)

