

抖音-主业务前端面试指南

参考资料

📖 头条研发前端面试标准 面试侧重点与评价标准请重点参考这篇

📖 面试官培训-优秀面试官 要求全文阅读

📖 产品与工程架构部前端校招面试白皮书 请通读，尤其注意里面第四章到第八章的考察大纲

<https://apaas.kunlun.kundou.cn/ui/apps/1710957518900238/appPages/9ppivpl8cubeoa> 公司题库

📖 抖音-主业务前端面试题库 组内题库

面试标准

通用部分

写在最前——大原则

就如前面资料提到的，一面二面中会根据候选人具体情况，针对CS基础/前端基础/前端语言/算法与数据结构/工程化实践/架构与组件设计/技术视野等方面对候选人进行考察。

要求：我们组在面试过程中，会更看重编码能力。编码能力不足可以作为一票否决。务必在面试过程中为编码考察部分充分预留时间。

要求：面试之前请务必提前阅读候选人简历，并提前准备要问的问题。

注意：在给定通过的评价时，请内心询问自己，“我愿意和这位同学一起工作/开发同一个需求吗”。

注意：宁可花费超过1小时的面试时间帮助自己确认判断，也不做无效面试，给出模棱两可的结论

注意：请针对不同同学考察不同的内容，不要同一套题目一直使用。

需要注意，一面二面应该互为补充，而不存在一面只考察基础，二面只考察项目经历等。

请注意，概念题准备好层层递进，候选人如果第一问回答的比较好，可以再追问一些细节，避免停留在浅层。如果候选人能拿到3.0，就再看看能不能拿到3+。

建议：对于之前组内同学面试过的回捞简历，鼓励提前与其沟通。

建议：对于一年以上的社招同学，可以以我们组相同年限同学为参考。

以下列举一些考察的方向，期望大家可以多考察不同的方面，面试过程中留意发现亮点。针对不同阶段的同学，考察重点也有所不同，例如校招的同学不要询问一些工程化/兼容性/框架深度原理之类的问题（除非简历中明确提到）。

准备思路与考察思想

考察方向：项目经历与自我认识

在面试之前，候选人介绍完之后，鼓励大家先结合简历聊一聊项目（尤其是部分候选人可能上来比较紧张，先聊比较熟悉的项目有助于候选人进入状态），可以考察候选人对于自己参与的项目的认识 / 所承担的工作细节等。也可以借着项目进入问答环节。

建议：结合STAR法则

这部分注意控制时长，尤其是部分候选人可能表达欲比较强，由点及面自行扩散，面试官注意把握节奏，可以有礼貌的打断。整体时间建议在5分钟左右，不超过10分钟。

考察方向：计算机/网络基础

可以参考 [国产品与工程架构部前端校招面试白皮书](#) 中第四章考察大纲。

这部分我们不鼓励问过多硬背的知识，很多候选人尤其是校招的通过能明显感受到是在背。如果作为考察点，更鼓励大家灵活的考察候选人对知识的理解。同时一定**注意控制量**（不宜超过5分钟），并且**这部分也不应该作为不通过的唯一理由**。

- 例如，TCP的三次握手过程，可以不需要候选人死记硬背的说出三次握手的详细过程，有的甚至需要回答报文字段SYN/ACK之类的；而是期望在候选人对大概过程有了解的情况下，可以问“有一类网络攻击就是利用三次握手过程进行的，你能描述一下攻击方式以及对应的防御方法吗”，灵活考察知识的应用。

考察方向：前端基础

可以参考 [国产品与工程架构部前端校招面试白皮书](#) 中第七章的考察大纲。

这部分题目鼓励大家尽量通过简单编码的形式，在编码中考察候选人对基础的掌握和运用能力。例如布局题目，可以要求候选人用两种以上方法实现一个左侧导航栏布局，这样大概率可以考察到flex/定位/浮动/BFC的掌握；想考察对于浏览器重绘重排的理解，可以设计一个会频繁触发重排的样式，让候选人指出问题并优化等；想考察候选人对于Promise的掌握，除了询问Promise的基本概念，我们可以设计一些题目同时考察Promise回调/Promise.race/ Promise.all的使用，比如为fetch实现超时功能等。

考察方向：编码能力

编码能力是指候选人将逻辑用代码实现的能力。其实编码能力是贯穿在整个面试中的，前端基础或者算法数据结构都能够考察到。在编码中，我们需要注意候选人的编码速度，对于边界条件的考虑，代码的简洁性等。

这一块的可选方向比较多，除了算法，也可以是常用工具函数的实现，或者常见设计模式的实现。

解题过程中注意引导，鼓励候选人先沟通思路然后再编码实现。引导方法可以见下面FAQ讨论。

编码结果正确之外，编码速度也是能力的一种体现。

建议：编码题不建议太过简单，同时第一道如果偏简单并且候选人回答的比较好，请至少准备另外一道难度更高的题目。

注：代码考察题请至少准备一个test case用于运行，我们要求最终代码是可运行的。

考察方向：算法与数据结构

这方面请保证考察难度，需要候选人对算法或者数据结构有一定的了解才能实现。一方面，我们可以考察候选人对于数据结构的灵活掌握与算法思想的充分理解；另一方面，如果候选人遇到难点，我们可以通过与候选人沟通给予提示，来考察候选人在遇到难题时的思维方式和反应速度。

当然也注意考察范围控制在常见算法与结构中（例如红黑树/图等比较复杂的数据结构就不要考察了），面试过程中这部分时间控制在半小时以内。

这部分的考察过程中，鼓励与候选人充分沟通，并适当给予提示。对于实现方案，很多候选人首先会提供暴力解法，此时可以询问暴力解法的时间和空间复杂度；同时，鼓励给出更优化的思路。一般情况下，只能实现暴力解法是不能算通过的，至少需要有优化思路。

例如，实现链表反转/判断链表中是否存在环，只是算初级的基础概念考察。针对非科班的同学，我们可以考察诸如排序算法实现/简单的dp（一维）/BFS与DFS的简单应用等；针对科班尤其是校招同学，可以适当加强难度，包括二维及以上dp/多种算法结合等。

考察方向：组件设计

设计一个场景，直接让候选人实现一个有具体功能的组件。这个鼓励大家更多从业务中总结场景，或者结合候选人简历中提到的项目设计一些组件题。

也可以是让候选人实现组件装饰器，比如设计一个低侵入式埋点的装饰器，当组件第一次可视时自动上报指定event。

注意：按照经验，组件设计从讨论到实现都比较耗时，如果需要考察请注意把握时间，最终编码实现的时候可以让候选人只实现关键代码。

考察方向：工程化（加分项）

一般情况下，候选人有一定的工作经验之后才会开始比较深入的接触工程化，因此针对校招/实习生不鼓励大家考察这方面。此外，针对非高阶同学，此项可以作为评价参考依据，但不应该作为最终不通过的唯一理由。

考察此项时，一般至少要求候选人在某一个方面有较丰富的实践，如果具备基础架构的开发经验是加分项。包括但不限于，如何对已有项目的性能优化 / 开发效率提升，webpack/rollup/esbuild的最佳实践和原理解析。

考察方向：框架

针对候选人的经历，针对Vue/React/跨端/SSR 方面考察原理方面，比如对React异步更新/Fiber的理解，对Vue 订阅发布模式的理解，对jsb实现方式的理解，SSR同构的实现等。这部分可以从简单的问题问起，一步一步深入，看看候选人对所用框架运行模式的探究程度。

注意，框架原理在单纯支撑业务的时候可能基本不会涉及，但是部分性能优化或者问题排查过程中会要求大家对于底层原理有比较清楚的认识。同时对框架原理的探究也部分体现了候选人对技术的追求和热情。

考察方向：系统设计

主要考察候选人的抽象能力和设计能力，**仅针对高阶同学**。可以结合简历上提到的项目，让候选人详细介绍其中的工程方法，技术选型，系统运转方式包括上下游依赖/运维等；以及使用过的合理的优化；深入考察候选人对产品或者业务的理解。

考察方向：全栈

一般接触过Node的同学会分为两种，一个是只写过简单的BFF或者一些脚本，另一种是写过比较复杂的业务逻辑。

针对第一种同学，我们可能主要会聚焦在NodeJS本身，因为这些同学没有传统后端的开发经验。比如对于node多线程的了解，事件流，文件读取等。

针对第二种同学，可以根据候选人的经历，在后端基础（常见存储MySQL/redis/mongoDB，消息队列Kafka/rabbit），NodeJS（包括koa框架，pm2工作方式），Java（Spring Boot，反射），常见流程控制（限流算法，线程池等）等方向考察。全栈可以考察的点比较多，如果要展开可能是另外一篇文档。

此外，参考服务端标准，全栈同学我们**要求对算法有一定程度的掌握**。

考察方向：跨端能力

也会有候选人已经有跨端开发经验，最常见的是直接写过RN，或者Hybrid开发中与原生进行紧密的交互。针对这部分候选人我们也可以结合项目经历询问相关知识，比如RN的启动过程，常见的RN性能优化项，JSB原理等。

考察方向：通用素质

贯穿在整个面试过程中，综合评价候选人的沟通和表述能力，包括面对难题的态度和思维能力，回答问题是否坦诚清晰等。

注意：面试过程可能存在极少数候选人反复跳出面试界面或长时间跳出界面的情况，这种行为是与字节规范完全相违背的行为，可以作为面试不通过的理由。

考察方向：其他

以上是一些常见的考察方向，除此之外鼓励大家结合候选人经历，问一些相关的其他知识点，包括但不限于SSR / 微前端 / 小程序等。

面试前

请务必提前阅读简历以及安排好自己的时间。

如果被分配到的候选人与自己的实际工作范围或工作经历不匹配，比如偏全栈的面试官被分配了一个主要做跨端的候选人，请直接与HR联系更换面试官。

面试中

时长

一般情况下，面试时长在40-50分钟左右比较合适。但也存在部分情况

1. 候选人表现比较差，在最开始的几个考察方向已经表现的无法通过面试，这个时候继续问下去对于面试官和候选人来说都算煎熬，因此可以提前结束。但也请保障30分钟左右的时间，表现对候选人和本次面试的尊重。

2. 候选人在前面的面试耗时比较长（尤其是编码部分考察的比较高的情况下），导致超时。这种情况下即使面试时间会超过1个小时，也请先考察完所有想考察的方向，给候选人一个全面的评价。面试之后可以总结一下超时原因是否是因为节奏没有把控好导致的。

节奏把控

整个面试环节，请保持提问，而不要陷入聊天或者讨论。抓住候选人回答的关键词，注意追问，挖掘问题的深度。

在问答题中，如果候选人没有回答到重点，并且在自行延伸，注意礼貌打断并将回答方向引导回来。

编码题中，请先让候选人阐述解题思路然后再实现，避免无用功。复杂的编码题请提前准备好几个tips在候选人陷入困境的时候提供思路引导。

候选人自我介绍

面试的第一个环节都会是候选人的自我介绍，一般情况下候选人会简单按简历把自己的经历概述一下。如果有项目或者实习经历，可以根据简历上提到的深入问一问具体的工作内容/技术细节等等，也营造一个稍微轻松的氛围。如果是校招生没有相关经历，可以问一问对前端的认识/学习路线等等。

与下一面的传递

对于判断要通过的候选人，如果HR安排了下一面，并且下一面的时间安排是紧接着的，可以在面试快结束（比如候选人最后一题快做出来或者候提问环节）提前飞书询问下一面面试官是否有时间。如果OK，再跟候选人确认是否可以继续面试，通知候选人线上等候（最好中途休息10分钟左右，比如5点结束面试，就跟下一面和候选人约定好5点10分开始下一面）。利用这个休息时间，跟下一面面试官简单沟通一下表现。否则，面试之后联系HR重新安排面试时间。

候选人提问

候选人的提问如实回答即可。以下为一些常见的问题

Q：你们团队现在的主要业务是什么？主要的技术栈是什么？

A：关于团队整体的三个业务方向和三个技术方向，请参考 [国 抖音主业务前端团队总结\(2020\)与规划\(2021\)](#) [国 新人面对面\(202110武汉\)](#)

Q：我后续可能参与到的业务具体是什么？

A: 一般情况下, 候选人如果最终入职, 优先会参与到二面面试官的业务中。如果一面面试官不清楚, 可以告知候选人在后续面试中再询问。

Q: 我后续整体的晋升/薪资待遇等会是怎样?

A: 具体细节可以在HR面与HR沟通确认

Q: 为什么今天问的问题都这么基础?

A: 我们会从不同的方向考察候选人, 复杂的技术也是从基础中衍生而来的, 不管在哪个阶段我们认为对于基础的掌握都是比较重要的能力。(当然也请注意不同阶段的基础考察比例)

Q: 为什么今天问的编码题这么多?

A: 我们认为作为一名工程师, 编码能力是非常重要的能力, 如何高效/快速/鲁棒的将设计转化为程序实现是衡量工程师的一个重要因素。

Q: 你觉得我今天的表示表现怎么样?

A: 可以取巧反问你觉得自己表现怎么样, 然后顺着候选人的自我评价稍微延伸一点。不要用评价性的语言, 更多的采用鼓励的方式。如果候选人比较真诚的请教, 可以适当给一点建议, 但是不要评价不要评价不要评价!

Q: 我通过了吗?

A: 不要透露最终是否通过, 尤其是在没通过的情况下。可以回复如果后续有面试, HR会再联系你。

注意:

1. 团队规模和业务细节/数据请不要透露
2. 不了解的问题可以告知候选人在后续面试中询问, 不要强行回答

面试结果

重要!!! 如何判断通过不通过

针对不同阶段的同学, 我们的考察重点和权重关系会存在差异, 具体请参考下方“面试场景”部分。

注意: 各面试场景的建议侧重方向是一个参考, 面试官请根据具体候选人情况灵活修改。

特殊case:

1. 简历非常优秀的候选人，一面下来不符合预期。

建议二面check，因为考察可能存在偶然性。

2. 对于某一方向很抢眼，或者都到了3.0的标准，但某些方向不够的同学：比如编码3+，但是基础和工程化都是3-甚至2.5的怎么处理？

a. 首先明确可过可不过的基本要求

- 编码基本达标
- 表达良好，沟通顺畅
- 底层原理不做硬性要求

b. 如果达到了以上标准，拿不准的一面可以先3分通过，2面感受如果一样就2.5 fail，交叉review尽量保证不错失人才

如果对于是否通过比较难以判断，**请务必与前一面/后一面同学沟通**，一来综合考虑不同同学的观点，二来也可以在讨论中统一大家的面试标准。

重要!!! 如何写面评

参考 [面试官培训-优秀面试官](#) 中面试评价部分。

面评编写的过程中，请先用简洁清晰的语言描述该同学的整体表现，包括亮点和缺点。对于通过的同学，需要写清楚其亮点；对于不通过的同学，充分描述不通过理由。

Good Case: 候选人在整个面试过程中沟通顺畅，回答问题切中要害。虽然在网络基础方面对一些知识点了解不深，但在编码过程中表现很好，能够很快提出解决思路并顺利转化为编码实现。予以通过。

Bad Case: 候选人在前端基础/编码能力上都表现一般，能够实现但是存在细节问题，整体中规中矩，先通过，二面再看看吧。

然后请详细描述面试过程中的每一个问题以及候选人的表现，并给出每一问题的打分。不要只记录过程但不给出评价。不然后续面试官需要仔细阅读才能提取出最后的结果。

代码题需要附上候选人的最终代码结果。

注意这部分需要和开头整体表现的描述保持一致，亮点和缺点能够对应上。

For 方向owner

如果自己的方向对候选人有在团队要求上的额外要求，请在评简历的评价中单独说明或者与面试官沟通。

面试场景

校招

校招同学可能在项目实践上不足，一般不考察工程化等需要较多项目经历的方面，侧重基础知识/编码能力与算法等方面。我们看重候选人的**学习能力和潜力**。

建议侧重方向：计算机基础（不超过20%）+ 前端基础（30%左右）+ 编码能力（50%，需要包括算法与数据结构）

特殊场景：毫无前端基础，没有学习过前端知识

这种情况不应该太多，至少候选人在简历上应该有足够多的亮点才能进入面试环节。

面试开始请先确认候选人对前端的兴趣以及了解其是否规划过学习路线。在面试过程中，侧重考察与前端关联不深的计算机基础/网络基础以及算法数据结构。注重考察编码能力和思维能力。

如果候选人对于其他方向有一定的应用（一般情况下都是做过神经网络/做过一些后端项目等），如果面试官自己对这些方面也有了解，可以适当储备这些方向的基础问题，针对简历上提到的内容适当提问。

特殊场景：对前端框架包括Vue/React没有了解

一般情况下，除非有实习经历，校招同学确实不会有特别的框架使用机会。对于校招同学我们不硬性要求熟练掌握任何一种框架，因为在前端基础足够的情况下，目前任何的前端框架都能够很快上手。

在有框架使用经验或者简历上明确提到掌握框架知识的情况下，可以适当询问相关使用知识或简单原理，但是不作为否决条件。

社招1-2年

建议侧重方向：计算机基础（10%以内）+ 前端基础（20%以内）+ 编码能力（50%，需要包括算法与数据结构）+ 组件设计（20%以内）+ 工程化（20%以内）+ 框架（20%以内）

编码能力之外，注重技术视野的考察

这部分同学已经有相当的实践经验，也基本在业务中开始承担重要开发的任务。我们组是立足于业务的团队，因此对业务的支撑能力必须过关，也就是代码能力必须过硬。此外，可以针对简历问一下项目经验，考察比如技术对比和选型/性能优化方案等能够体现候选人自己思考和对技术追求的内容。

社招3年以上

建议侧重方向：编码能力（30%，可以不考察算法）+ 组件设计（20%以内）+ 工程化（20%）+ 框架（20%以内）+ 系统设计（20%）+ 团队/技术视野（20%）

社招三年以上的同学，我们对于算法与数据结构的要求可以适当降低，更多偏重对整体技术方案/工程化能力等方面，也可以加入对业务和团队思考的考察。但是编码能力考察依然需要保留足够的比例。

面试之外

以上的内容更多的是从团队角度对面试官提出的要求或者建议，在保障候选人体验的同时也为团队挑选到符合要求的新力量。但从另一方面讲，面试官在面试中的体验也同样是我们需要保障的。

在面试比较集中的月份（校招/三四月份等），面试官的面试压力会相应增大，如果遇到多个连续面试，或者一天之内安排的面试过多严重影响日常工作的，请直接联系HR调整或者减少面试时间。

对于一些特殊时间的面试（比如上午10点或者活动日/周五晚上），HR可能会提前与面试官沟通，面试官可以视自身情况要求调整时间。如果HR没有提前沟通直接分配，也可以找到对应的同学沟通。

FAQ

面试前

Q: 如果遇到我自己看简历就觉得即使面试也大概率会挂掉的情况，怎么处理？

A:

1. 一般情况下，如果仅仅是觉得比较勉强，那先正常面试，如实填写面评。

a. 若拿到面试简历时，真是觉得无法通过，则考虑直接与HR沟通，然后与筛简历的HR/二面同学沟通。

2. 针对之前面试过但是重新捞的同学，考虑“重复提问”已面过的问题，提前确定该候选人是否有认真复盘，事后学习。

Q: 遇到完全不会前端的候选人，可以从哪些角度考察？

A: 这种场景一般发生在校招/实习的情况，看候选人的情况来设置预期（是否有经验，是否专业方向匹配）我们注重考察学习能力

1. 看在校期间的基础课程（网络，算法等）。

2. 以思路为准，讲清楚设计逻辑流程。

3. 软素质（学习能力、沟通能力）。

4. 针对特别有动力但目前基础能力不够的同学，则礼貌告知建议通过一定学习后再来。

5. 若是CS基础非常好，但对前端确实了解不够，那么就分场景而论：

a. 是否有前端的一些基本认知和动力

b. 考虑与二面同学沟通

Q: 新面试官如何准备自己的面试题？比如针对不同阶段/不同经历的候选人如何选择题目以及难易程度？ A:

1. 和资深面试官**取经**，了解其是如何准备自己面试题的；

2. 完整阅读公司**面试文档和面试案例**，特别是 bad case；

3. 针对候选人经历，**侧重考察点**不同：校招一般项目经验少，可以偏基础，经验丰富的可以根据项目拆解问题，考察项目组织设计；针对已有项目考察

4. 参考部分**统一题库**，题库按照考察方向分类，每道题目有简单、中等、困难等标签，同时面试官需要提前筛选熟悉题目；

5. 挖掘候选人**擅长方向**，寻求候选人与我们业务结合点，考察候选人能否将自己经验应用到我们业务；

6. 面试过程要**注意礼仪**，无论面试结果如何，都给候选人一个良好的面试体验；

面试中

Q: 编码题难度如何控制？解题过程中如何适当的引导？

A:

1. 循序渐进，先从基础简单的题目开始，然后根据表现，再加一道难度大的；或者考察综合应用题，可以按照年限加入一部分算法相关的
 - a. 什么样的算基础？什么题目算难？
 - i. leetCode 中等的难
 - ii. 业务中不太常用的点，比如偏原理的题目
 - b. 如何适当的引导
 - i. 先问思路，如果思路对的话继续写
 - ii. 在写的过程中看到有问题的点会问
2. 如候选人完全没思路；思路错了；有思路写不出来；有思路能写但细节不OK
 - a. 思路错了，建议直接说明，不浪费双方时间
3. 思路正确，但是写的一直很慢，是否还需要给充足时间让其写完？
 - a. 看情况如果同学前面比较ok 会给时间写完，但是在面评中记录
 - b. 否则，直接打断，但是会让其描述整体思路

Q: 面试过程中遇到跳页面（偶尔跳出/反复跳出），或者没有正视镜头的情况，应该怎么处理？

A: 对于第一种情况，请在面评中如实反映；如果基本能够确认面试者是在检索，那可以直接不通过处理。

如果是没有正视镜头，请先提醒面试者。

面试后

Q: 如何写出一份清晰的面试评价？

A: 一份比较好的面试记录能让其他同学通过记录快速了解技术特点和水平

- 详细记录重点问题
 - 给后续面试官作为参考，防止重复考察
 - 每个问题考察写明回答是否准确
- 言简意赅，公平客观
 - 不需要记录每个细节，只需记录核心关键点
 - 语言简短，要让其他人不需过多思考便能理解
- 具有结论性意见
 - 对所有提问做一个总结

- 技术、沟通表达、潜力等
- **注明未考查点及风险点**
 - 让后续面试官参考关注

面试系统手册

飞书面试操作手册：[📄 飞书（含代码考核）技术视频面试操作手册](#)

牛客网常见问题：[📄 牛客网视频面试常见问题](#)