

进入大厂的面试经验 (P7)

个人近期面试情况

今年二月以来，我的面试除了一个用友的，基本其他都被毙了，可以说是非常残酷的。其中有很多自己觉得还面的不错的岗位，比如百度、跟谁学、好未来等公司。说实话，打击比较大。

情况基本上是从三月开始好起来的，这个时间点，可能疫情快过去了，国家也开启了接近40万亿的基础建设计划，在这个时间断，面试的几个代表性的公司，包括腾讯、饿了么、宝宝树、旷世科技等等，基本都成了。

最终，就在三月9号，饿了么刚给我确认了p7的职位。也算自己到了另一个奋斗的阶段了。对自己的经历，做一个面试总结。

我的面试准备

面试其实我是属于一边面试，一边准备的过程，但是无论如何，一定要做到基础扎实、比较好的自我介绍，以及描述自己的职业故事。

1、基础知识的准备

这一块，其实很重要，很多问题，必须轻松的理解，就和呼吸一样自然。

因为这一块的掌握程度，就决定了一个前端工程师，能不能达到高级的程度。

抛砖引玉，比如面向对象、原型链、闭包、es6、html5、css3，它们是前端的基础，需要烂熟于心。比如连闭包都解释不明白的，真的该好好下功夫在这些基础上。。

对于前端框架，基本的特性，生命周期、组件、传值交互、路由、redux、vuex等等，基本都没有使用上的疑惑。比如面试官可能问，actions 和 mutations 有什么区别之类的。

这一块，能问的问题几乎是无穷多的，哪怕同一个知识点，问法不一样，不真正理解的话，实在是过不去高级那一关。

2、好的自我介绍

自我介绍，很多人其实做的不好，其实这是面试官对面试者的第一印象。

如果我是一个面试官，我基本能从自我介绍中，得知这么一些信息：

- 1、这个人是不是一个逻辑清晰的人？
- 2、这个人的性格是积极向上的，还是稍显自卑的？
- 3、这个人能不能很好的表达自我？

这些问题很重要，按照这个思路，其实你反过来就可以向面试官表现自我。

- 1、学校、专业
- 2、技术栈
- 3、职业履历
- 4、最近一家公司的贡献

千万不要和面试官说，和产品、后台很好的配合，完成公司的项目之类。因为完全是废话！

自我介绍一般是两分钟左右，你确定你能两分钟，把上面四个问题给面试官讲清楚？并且你还要突出一些重点，比如说，我比较擅长react，做过XXX优化之类，有过XX年的项目经验之类。

自我介绍的本质：用最短的话，向面试官灌输你的优点，并且让面试官在当前大致的范围内考察你。如果你说你熟悉vue，一般面试官都会问vue相关的多一点，应该不会和你死磕react。

3、如何“讲故事”

这是一个比较重要的点。重要的是，这个故事要有看点，表示了你和其他人不一样在哪。

比如我，我会说，我之前在公司，做了我们前端的项目标准化工作，做了我们内部的组件库建设，集成了我们自己的脚手架工具。

这是我觉得，我和其他大多数前端不一样的点，面试官的焦点就会集中在，什么是项目标准化，定义了哪些维度的标准？组件库的建设，解决了什么问题？脚手架的出现，又解决了什么问题？

任何一个人，都是唯一的，任何一个项目，也是不同的，作为面试者的我们，重要的是，要把这些重点要素提取出来，形成自己的“故事”。

比如，我独立承担了公司的xxx项目。遇到了哪些挑战，我如何在不利的环境下成长起来的。

比如，我通过xxx渠道，研究了多少个项目的架构工作。

一定要相信，因为你，什么才发生了变化。其实大多数人是一个被动的状态，不知道自己有什么用，自己在团队中的定位，自己的职责是什么。在工作中，我希望每个人相信自己的是有用的，然后寻找你可以发力的点，去做真正的改变。如果你在找工作，那就好好思考，自己确实在团队中，承担了什么。

面试该注意的点

面试的核心是什么？一定要记住，那就是沟通！而我们做技术的，其实最大的硬伤，多半是沟通，至少对于我自己，就是有这个问题的。

沟通的意义，一方面，是你要让别人听懂你表达的东西；另一方面，是你要听懂别人的话。

1、说话得注意，让别人懂你

直接举个例子，我在面试美团的时候，说话一直结巴，其实我也不知道为什么，平时说话都是正常的，在特殊的场合，就出现这个问题，虽然把话说完了，但是显然，美团第一面，就pass了。当时是在家中视频面试，我的室友，不好意思直接说我，就给我录音，我听了10分钟左右，确实不尽人意。。

我听了10分钟的录音，给我自己的评价就是，说话都费劲，以后一起工作，那还不把我累死。

以后，所有的面试，我都把关注点，从我如何说，变成了，如何让别人很愉快的听懂我说的话。

第一、我向面试官说的所有的话，都是成体系的。

第二、我向面试官说的所有的话，都是逻辑清晰的。

第三、我向面试官说的所有的话，我都先停顿两秒以上，思考清楚了再说话。

当然了，这一块，可能是我个人的特例，但是基本的语句通畅、逻辑清晰，在程序员中，做到的应该不多。

2、学会倾听，让你懂别人

很多情况下，其实面试官在表达的时候，很可能我们自己没听懂面试官的意思，这个时候，就会特别的尴尬。

我一般会这么做，我会说，刚才您说的这个问题，我来描述一遍xxxxxxx。描述完之后，你再问面试官，您表达的是这个意思吗？一般而言，面试官会进行更详细的举例和描述。

我在面试饿了么的时候，二面的面试官，一开始就问了一个业务比较复杂的问题，所以一开始，我其实完全没明白，面试官说的是什么东西，但是我通过自己的方式，引导面试官说的更多，然后到某个程度，达成一个共识，这就是比较愉快的一个交流。

无论是学会说话，还是学会倾听，其实核心的问题，就是让双方的交流，在一个频道上，因为对于领导，面试面试，其实就是找同类人！

必会的面试题目

前端一些核心的经典问题，感觉永远都不会过时，反正我个人遇到的频率非常高，不过个人不会详细解读这些问题，只做一个抛砖引玉。

1、前端浏览器输入URL后发生什么？

普通前端会回答，dns解析，获取html文件，解析DOM，渲染页面这么一个流程。

其实过程复杂的很，比如一个dns解析阶段，它分为哪几步？解析的ip一定是一样的吗？每次都会进行dns解析吗？可能还需要了解cdn托管的一些问题。

比如html文件获取，它是如何传输的？如何建立链接的？三次握手、四次挥手是什么？http协议端口是什么？为什么直接能访问一个html文件？

....

2、前端性能优化？

大多数前端，基本会说，减少http请求、压缩合并js以及css、图片懒加载的技术、防止回流和重绘、css放头部、js放底部。

以上的说法对吗？完全正确！可是在这个过程中，我觉得缺少一种思考。

比如减少http请求，可以从哪些维度上减少呢？比如合并http请求？比如合并资源？比如图片懒加载？http还有其他维度的优化吗？cookie优化？http请求和资源加载的区分优化？在webview中呢？和普通的浏览器的优化技术，又有什么区别？

.....

3、前端如何做性能监控、异常监控？

性能监控，异常监控，基本在小公司，是没有实践基础的，可是在差不多的大厂中，他们会关注这个问题。

首先是性能监控，应该从这么几个维度来说：一个是http的方面，在后端log日志，流入kafka，然后在kafka消费数据，可以准确的监控到哪些接口有异常？异常率是多少？另一个方面，是前端的 Performance 的api，在用户的实时使用的过程中，就会产生数据，这样就能实现页面性能监控。

前端异常监控，首先要明白什么是异常，html、css这些东西，无非就是一个展示的问题，还不至于让页面白屏的事情发生，所谓的异常监控，其实就是js的异常监控。在前端领域，window.onerror是进行js异常的监听事件。并且要知道，它在IE中，是不支持的，所以IE的监控，要使用try catch 的方式进行捕获，比如我们可能还要注意到，遇到异步的时候，这个如何做try catch的异常捕获。

最后一个是前端sdk埋点，直接开发一个js文件，统计用户的UV/PV分析等等，比如用户的转化率之类的，这一块个人没有什么特别的实践，各位可以在网上百度看看。

4、前端安全方面

这是一个高级前端必问的问题，说的是一个前端对整个前端安全的系统认知。

我们必须理解这么几个方面：sql注入、xss、csrf、cookie安全、密码安全等等。

sql注入，要理解sql注入的场景，它的原理是什么，当前的数据库的解决方案是什么？

xss攻击，常见的攻击场景，什么类型的网站容易被xss攻击，整个流程的原理是什么？

csrf攻击，其实就是一个钓鱼网站，要理解为什么会收到攻击，应该采取什么策略进行防御。

cookie安全，要理解为什么用token，优势等。

密码安全，主要是用户登陆，用户数据提交，加密，存入数据库的一整个流程。

其次，其实还有http和https的问题等等。

5、http、https、http1.0、1.1、2.0、3.0的区别

http这一块，其实是一个非常复杂的体系，要深挖的东西特别多。

http进行非对称加密，得到https，这个过程是怎么样的？什么是CA证书？整个网站进行验证的流程是什么？

http各个版本的区别是什么？解决了哪些问题？比如头部缩减的优化，那你了解这个优化的具体策略吗？缩减了什么？又增加了什么？要深挖细节。

http的底层协议？tcp/ip协议的三次握手，四次挥手，具体是怎么通信的？什么叫满启动？甚至延伸到整个网络协议的领域，什么是socket？udp是干什么的？dns解析？ftp？以及不常用的其他协议？

如果再进行扩展，计算机网络的7层结构？每一层做了什么事情？计算机组成原理，如何解析我们的代码等等。

.....

深度解读

这一部分，我大致分了几个模块，源码、可视化、跨平台、工程化、混合app交互、设计模式，其实每一个都值得前端去研究。

作为个人，其实也算是我定了几个方向，自己以后要努力的地方。

1、源码

最好要读一读某个框架的源码，最不济，也看看别人写的源码解读的文章，现在的趋势，基本是以react、vue为主。

以vue为例，你得懂vue框架的整个周期，比如vue的初始化，发生了什么？vue的模板解析，是如何进行的？如何形成AST？render函数的生成？什么是依赖收集？什么是patch？数据更新策略等等。

你也要懂得在这个过程中，混入mixins、\$options、vuex、router他们各自如何通过这些api，实现各自的功能？

源码的维度，可以试着从vue的体系、react体系、lodash、zepto这些库中下手，因为我们就是从这些框架以及js库，走过来的。

2、可视化

地图、echarts、canvas、webgl、d3.js、three.js!

上面是个人想要研究的关于可视化的一个个主题，虽然可视化不是本人的专长，可是因为之前在地图相关的公司待过，也算是沾了可视化的一点光。在面试饿了么的过程中，就提到了一个特殊的使用场景，就是关于地图的线路绘制，大批量数据的推送，以及性能相关的问题。

以后，普通前端的技能，比如vue、react等等，大家都会，没有什么差异化的竞争，前端市场趋于饱和，更需要的是在某些方面专精的人才。

3、跨平台

flutter、react-native、weex、electron

这是目前市场上，针对跨平台的一些解决方案，每一个框架，大致方向上，都解决了一定程度上的多端开发能力。

首先需要明白的是，多端开发并不是万能的，也有一些我们开发中的痛点，无论rn、还是weex，都是给前端提供一些特殊的组件，实现开发能力。但是业务是多样的，需要也是多变的，对于没有提供的组件，或者api，我们前端开发人员，有些功能是没有办法实现的。

对于多端框架，更重要的是理解层面的东西，比如electron，就是要理解它的本质，就是使用node塑造一个桌面应用的容器，然后内部是一个webview。

从面试的角度，面试官重视的不是这个东西怎么用的问题，一般会问一些内部原理，比如小程序和公众号的对比？底层实现方式？比如rn的编译解析过程是什么？如何把js编译为真正的ios、android应用。比如flutter的渲染机制，和rn这些渲染方式有什么不同？和普通的web网页又有什么区别？

跨平台技术，flutter最近比较火，从找工作的角度，其实可以学学。electron这些桌面应用，本身不难，但是学过和没有学过，其实本身就是一种差异。

4、工程化

工程化这一块，是个人比较感兴趣的一个方向。自己也做过一些实践，但是只能说是工程化中的冰山一角。

抛砖引玉，提一些我们可以探讨的点。

1、前端项目标准？

表现为库的选择？文件划分目录的规定？pc、mobile多端实践？ssr方案集成？

2、组件库集成？

组件库建设的目的？npm包的发布？

3、脚手架工具？

webpack编译优化？webpack打包构建优化？自我脚手架的工具使用？

4、git提交规范commit-msg？代码检查规范eslint？

5、前端性能监控？前端异常监控？前端用户埋点sdk？

6、rap？jenkins？

5、混合app交互

jsBridge、性能提升方面、x5内核

算是一个小的模块吧，我们得明白，jsBridge是如何做到h5和原生应用的交互？

ios和h5的交互通信？window.webkit?是否同步？

android和h5的交互？细节和ios有什么不同？

webview在性能提升方面，可以做哪些？什么是离线包？

腾讯x5内核的优势是什么？我们用了x5内核，可以避免什么问题？

6、设计模式

最后谈谈设计模式，算是一个高频的面试题。

我们至少要知道以下这些设计模式的功能、代码实现、使用场景问题。

单例模式、原型模式、工厂模式、观察者模式、策略模式、代理模式等等。

最后祝大家面试愉快。