

阿里腾讯面试梳理&个人成长经历分享

前言

好久没有更新了，最近忙着面试，写毕业设计和论文。

不过不想停下记笔记的习惯，所以偷偷的发面经，然后"惊艳"老铁们。

校招面经，面试难度中等，看官老爷们看个热闹就行。

历经一个月战线，投了阿里和腾讯，具体部门这里不展开了，都是核心部门，提供的舞台很大，至于最后选择去哪一家公司，可以关注文末。

接下来复盘一下这一个月来的面试感受吧。

本文不会泄题，只是根据我的简历，来展开谈一谈，可能会被问到的题目。

最后以下都是个人观点，个人能力有限，**「如果有错误，希望能纠正我。」**

简单介绍

2021届毕业生，掘金里面的TianTianUp，双非本科院校(江西师大)，没有听错，那个彩礼"天价"的省份。

21届里面，应该算起步比较晚的了。2020年暑假7月份，才开始认认真真"复习"前端，当时连js作用域，闭包，浏览器缓存都不清楚。因为起步晚，所以后续错过了秋招。

现在我才深刻意识到，视野的重要性。

当时大三学了点vue，做了点看起来牛逼的东西，就觉得很牛了，于是春招投了阿里，结果惨不忍睹。

重点当时没有放心上，还是继续玩，继续打游戏。

还好之后觉悟过来，才有了后面网易的故事。

嗯，也导致后续很多环节跟不上其他同届大佬。

秋招的时候，jack，三元同学，林不渡，单哥，承和等大佬都是offer收割机，各种腾讯，快手，字节ssp，无不冲击着我这个萌新。

暗自下决心：我也要去大厂核心部门，去做有意思的事情。

有了规划，接下来就是看技术博客，动手实践，总结梳理，写博客。

熬了无数的夜，无数个周末在社区徘徊，这个知识点不懂，查文档，看不明白就换一篇文章，争取做到一遍过。

这么坚持了半年，2021年的春招，结果让我满意，但结果还是有些遗憾的。

满意的地方：借面试机会，跟着大佬交流学习了一波。

遗憾的地方：其他大厂没有去尝试，没有趁着面试机会去接触其他领域的大佬。我觉得原因可以归咎于：

- 找不到大厂核心部门的内推，希望去核心部门。
- 需要笔试，觉得流程很慢，没有人跟进。

所以后面提到了面试机会。

整个春招的过程中，聊一聊心态发生的改变吧，迷茫--紧张--坦然--自信，最后到意向书，到拿到offer。

「小tips」 ↓

如果你是在校生，推荐你阅读三元同学博客，干货满满。

面试机会

基本上我是靠内推的，腾讯文档，是一位校招大佬帮忙推的，觉得履历还行，给了个面试机会，才勉强给我通过的。真的是勉强给的机会，因为学校确实跟985比起来，差距很大，还好有网易实习经历buff，加上写了点博客笔记等笔记输出，才有这个面试机会。

内推，内推，内推，这是最快，最有效的方式，可以第一时间知道面试结果，面评，以及后续的面试流程。

至于其他的，美团，百度，shopee走的都是官网流程，需要笔试。腾讯这边到四面总监面的时候，我基本上都放弃了。

最好的方式就是找靠谱的内推吧，比如秋风大佬，字节找ssh（昊神），很靠谱的。

好的履历也很重要：

- 学校985，211。
- 大厂实习经历，众多offer。
- 扎实的基础，读过源码，理解框架设计思想。
- 社区有一定影响力，坚持做博客产出。
- 开源贡献，开源项目维护者，热爱前端。

基本上，我会把握其中的一两个点，争取做好，才会有后续的面试机会。

有了面试机会后，剩下的就是如何去做好一场面试的准备工作了。

如何准备

这个链接里面总结的特别好，我基本上看它复习的。

bitable.feishu.cn/app80k6k9qa...

不过面完，给我感觉，重在平时的积累，基本上都是围绕项目展开的，之前做的项目得好好准备一下，比如项目难点，项目如何做优化的，项目的亮点。

这里推荐一个链接：

面试被问项目经验不用慌，按这个步骤回答绝对惊艳

juejin.cn/post/684490...

至于其他的部分，更多的是向面试官展示自己的优点，闪光点。

面经心得

基本上都是围绕我的简历来的，有兴趣的话，可以联系我要pdf，可以给你参考下。

1. 掌握 JavaScript，熟悉 ES6 语法，对异步编程有一定研究。
2. 熟悉 React + Redux + TypeScript 技术栈，对 React Hooks 有实践，对 React 相关原理有了解。
3. 对浏览器的渲染原理、性能优化有一定研究。
4. 熟悉 TCP、HTTP 网络协议。
5. 扎实的数据结构和算法基础，对基础算法有深入研究。
6. 熟悉 webpack 的基本配置及从 0 搭建完整项目，了解 webpack 常见的打包优化方案。

@稀土掘金技术社区

「简历上技能写的不多，基本上能保证的是我写的都能聊上两句。」

从我简历部分来看，主要涉及几个点：

- React 部分，这里面可以问的内容也很多。
- 浏览器相关的。
- 计算机网络部分。
- 算法与数据结构，基本上这算是我的一个杀手锏吧。
- webpack 相关的，同时会涉及到 babel。

我现在回顾我的面试部分，都是围绕这些点展开的。可能我是应届生校招的原因，很多时候，考察的是我学习能力，理解能力，对前端热爱的程度这些方面。

基于这几项，展开聊聊我面试过程中会注意的点，以及围绕我写的专业技能，会聊到的一些话题。

自我介绍

很重要的一个环节，说不定可以奠定本次面试的节奏，会根据你所说的内容，来展开问你，同时也会给「给面试官一个缓冲的时间来重新熟悉你的简历」。

面试自我介绍，要：

「遵守极简原则，突出重点，忽略细节。」

小参考：

- 我是谁。
- 我的亮点，可以描述一下几个点，最近的经历。
- 我为什么需要这份工作（其实这个我自我介绍都是忽略的）。

围绕上面的点，我会着重突出，我在大学的经历，主要是算法相关的经历，接着就是我在网易实习期间的收获，以及中间会持续更新博客，写总结。

基本上，面试官对我的掘金博客，以及大学搞算法的经历感兴趣，进入了自己擅长的领域，就会轻松很多。

记住一个点，「**面试是跟人打交道**」，学会去展示自己，以及及时跟面试官做合理的沟通很重要。

以上是我会注意的一些点，我也是这么去做的，尽量围绕自己的「**亮点**」去展开，当然了，有些面试官有自己的风格，「**可能一上来就是问你项目**」，那么我们应该如何把握呢。

项目相关

我觉得项目相关的话题，是个挺难的话题，围绕它展开的话，可以延伸出很多话题，我列举了些有意思的：

- 项目难点。（如何发现问题，解决思路，最后结果）
- 项目考虑过优化吗，你是如何优化的，思路是什么。
- 项目的组织架构，你对它的现有架构的理解，哪些优点值得借鉴，哪些缺点需要改进。
- 如果让你从0到1建一个项目，你考虑的点是什么，有哪些流程需要注意的。
- 项目中代码规范，你们项目有方案吗，你了解的代码规范有哪些方案。
- 说一说项目中你们是如何测试的，有哪些单元测试方案，能不能说一说。
- 项目中引入TS的原因，为什么这么做。

不过对于「**应届生**」来说，问项目都是围绕你简历来问的，比如我当时的项目，我觉得没有好的难点，我就「**着重说明**」了下，我对项目是如何优化的，优化了哪些点。

整个面试下来，都会问，你项目写到的优化的地方，能不能具体聊一聊。

这个时候，真的要把握机会了，问到你会的点，你得有清晰的思路，可以参考STAR法则。

重新回顾一下 STAR 法则四要素：

- 「**Situation**」：事情是在什么情况下发生，基于一个怎样的背景；
- 「**Task**」：你是如何明确你的任务的；

- **「Action」**：针对这样的情况分析，你采用了什么行动方式，具体做了哪些工作内容；
- **「Result」**：结果怎样，带来了什么价值，在整个过程中你学到了什么，有什么新的体会。

这里有段话，我觉得说的很不错：

往往大部分同学一上来就直接介绍做了什么以及实现的过程，条理也比较清晰，内容也颇具技术含量。但很多同学很容易忽略了 Situation 和 Result 的部分也就是背景和结果。或者是在面试官进一步了解追问细节的时候容易惊慌失措。这些原因往往都是由于面试前对自己的经历没有将来龙去脉讲清楚以及总结不够全面和深入。

基本上，我对项目这块目前能聊的就是这么多，对了，如果你面**「大厂核心部门」**，可能还需要关注的点是，**「项目的组织架构理解」**，让你实现一个项目，整个流程需要注意考虑哪些点。

这块真的可以着重准备一下，以我这次春招经验来谈，是重点。

聊完项目，接下来就围绕我**「简历写到的专业技能」**，依此来梳理一下。

JS基础

考察的是一些基础问题，这里最好过一遍，切记不要丢大分。

比如我写了对异步编程有理解，所以得看一下，异步编程的解决方案，它的发展过程，每种解决方案的优缺点，得了得，最后我会**「拓展一下async方案存在的缺陷」**，对错误的捕获，聊到**「babel相关」**的，现有的babel社区是否实现了对async语法做错误捕获处理，以及谈了下，社区里面的这个插件存在的缺陷。

总结的话，我会围绕**「面试官问的基础点展开」**，把相关连的部分描述一下，面试官有兴趣的话，会顺着你的思路往下问。

至于面试过程中问到**「JS基础」**相关的题，这里就不列举了。

React相关

第二部分，简历上写了React相关的，那我总结了些可以会问到内容：

- React框架的原理，聊一聊它是如何执行的。
- React更新流程，主要经历了哪些步骤。
- React中setState的理解，需要注意哪些。
- vue和React区别，如何从项目的选型上区分它们。

「React框架的理解，vdom，patch，批量更新，渲染的过程，经历了啥。」

你对React源码很感兴趣，可以关注一下「卡颂大佬」，他的[自顶向下学 React 源码](#)值得推荐学习一下。

当然了，React真的要问，太多可以问你的啦：

- 比如让你手写一个Hoc组件，实现类似怎么样的功能，思路是怎麼样的。
- 比如Hook你对它的理解，当时面试问到了Hook实现原理，还好看平时喜欢折腾，看了下大致流程。
- React-fiber机制引入，解决了什么问题，具体的流程是怎麼样的，数据结构上发生了哪些变化。
- React中的时间切片了解嘛，如何实现的，流程能不能说一说。

这次问到好几个React相关的问题，更关注的点是你对原理的过程，有关注嘛，大致是一个怎么样的过程，最后又从中学到了哪些点。简历上写看过React原理相关的，得慎重啊！！！！

快进到浏览器部分。

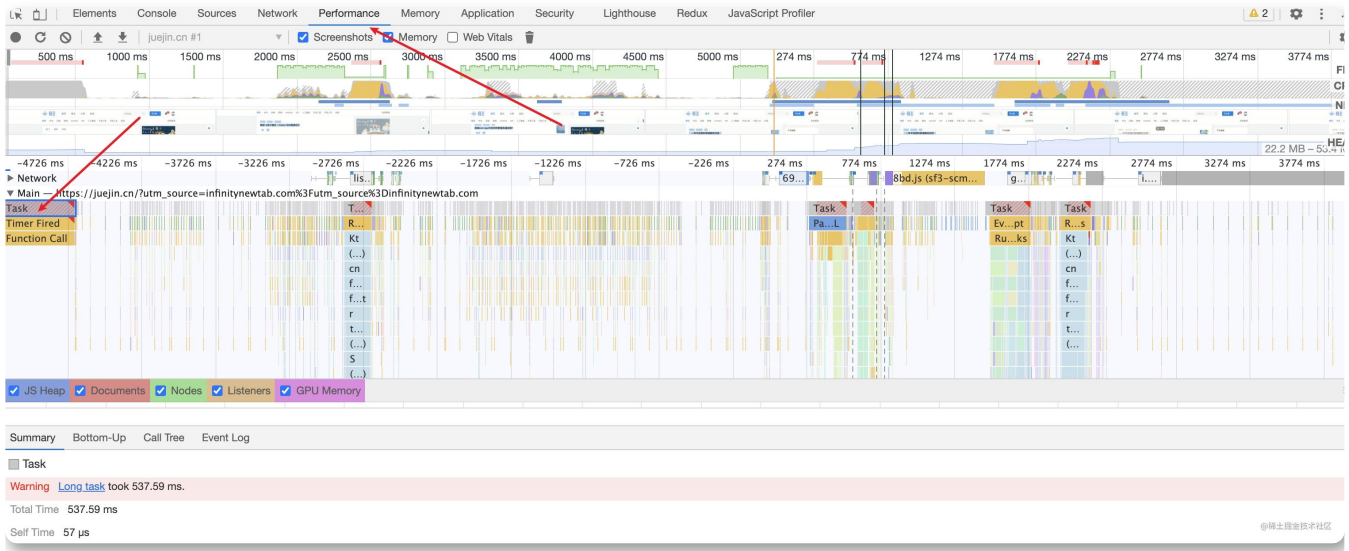
浏览器相关

这也是我简历上专业技能写到的一个点，这里也写了几个点。

- 输入URL到页面渲染。
- DNS解析的端口号是多少，为什么采用UDP协议。
- 你渲染流程知道的这么详细，图层，分层，绘制，栅格化线程池，GPU加速，能不能说一下，对你开发有啥影响呢。（其实这个时候，被无情打脸了，知道这个有什么用，当时反问我）
- 浏览器缓存。（可以拓展一下，结合webpack如何命中缓存，然后就到你熟悉的webpack领域了）
- 浏览器进程架构历史。

- 浏览器内核。（这个可以了解）
- 「浏览器的垃圾回收机制」。

经典八股文了，不过其实我感觉也是可以拓展的，问到浏览器缓存的时候，结合webpack谈一谈如何命中缓存的。比如讲到浏览器的渲染流程时，我们都知道里面的一些具体流程，比如分层，绘制，栅格化操作，然后其实可以结合实际去思考一下，有哪些具体的用途呢，这里我们需要借助浏览器中的performance来看看。



里面有很多关键性指标，比如 Long Task执行的细节，跟面试官交流了下，「排查性能问题」，我是如何通过这个Performance相关的细节去把握的，当然了，里面的内存使用情况，是否出现「内存泄露」问题，都可以借助它。

非要我总结的话，我觉得哪怕是八股文，我们也要尽可能的展示出亮点来。

最近也在学习性能优化相关的部分，发现Performance相关的部分，还是有很多地方可以学习的，「可以持续关注我」噢～

webpack相关

基本上，看我简历上写了这部分的，会问到，部分问题：

- webpack打包原理。
- import最终被webpack编译打包成什么。
- 路由懒加载的原理是啥，能不能结合webpack说一说。
- 写过webpack loader 或者插件吗。
- webpack 如何做性能优化。
- babel的原理了解吗，babel是用来干嘛的。

- 说一说常见的babel库，你有使用过哪些。
- 写过 babel 插件吗？用来干啥的？怎么写的 babel 插件
- 知道怎么转化成 AST 的吗？

基本上考察，你是否实践过，因为自己简历上写了这方面的，可能问的比较多。

比如问到babel原理可以这么答：

babel的转译过程分为三个阶段：**「parsing、transforming、generating」**，以ES6代码转译为ES5代码为例，babel转译的具体过程如下：

- ES6代码输入
- babylon 进行解析得到 AST
- plugin 用 babel-traverse 对 AST 树进行遍历转译,得到新的AST树
- 用 babel-generator 通过 AST 树生成 ES5 代码

基本上至少得知道它大致的原理，平时如果有积累的话，其实问题不大。

比如babel的话，推荐可以看看babel是如何处理async错误捕获转换的，这个有点意思，正好借这个机会跟面试官交流一下这块的知识点，个人觉得还是很有意思。

学的时候，**「多折腾一下」**，应该就问题不大了。

算法与数据结构相关

其实这块，我希望面试官问我多一些，因为大学期间，更多的精力放在算法。有幸碰到过面试官也是打过ICPC，CCPC等比赛的，曾经都是热血青年，最后因为生活，选择了向生活低头。

聊得过程还是很愉快的～ 也是第一次，跟面试官聊到了，线段树，主席树，最小费用流，二分图匹配（匈牙利）算法，差分约束，Graham扫描法，有限状态自动机，AC自动机。

似乎找回了大二的那份对算法的热情。

记得退役的时候，在QQ个签上，写过这么一句话：**「我永远热爱算法，无论它虐我千百遍。」**

可能是真的熬了无数的夜，才会有这样子的感慨吧。有点跑题了，感兴趣的话，后续会更新，我是如何走上算法这条路的，最后给我的成长收获又是哪些。

回到正题，至于这部分的话，我觉得没有一个很明确的范围吧，毕竟算法太广了，过一遍吧。

- 有向图判断是否有环。
- 给定一个数组，如何拆分，保证两个数组的和接近。(想到的是贪心，正解应该是dp)
- `Json.parse()`实现。
- 常见的数据结构。
- 数组和链表区别。
- 时间复杂度和空间复杂度计算。
- **「排序算法」**。
- 你是如何看待算法的，给你的收获是什么。

算法很重要的，对你未来的发展，我觉得还是很有必要去学习的。至于怎么学，我没有能力回答这个问题，要我给建议的话，**「看书是一个选择」**，如果要临时应对面试，刷一刷leetcode来说，只能说短期有效果。

个人建议：

可以结合书的内容，分专题去刷，比如动态规划dp专题很难，我记得我当时看完书后，对它的理解就很清晰了，当然做题是另外一回事，通过题目不断巩固你对一个算法知识点的理解，我觉得是没有问题的，但是至于完完全全刷leetcode，应付面试，不从最基础的开始，这样子“超车”，觉得意义不大。

以上是我个人理解，**「能力有限」**，说的不对，您就当看个笑话。

个人经历

问到这部分的话，可能是总监面你的时候会问，也可能是hr会问，到hr环节的话，基本十拿九稳了。不过也不能大意，说不定一不留神，卡你学历啥的，一个月的面试付诸东流，还是要认真对待的，罗列了几个点：

- 整个经历自我介绍，越详细越好，什么时候接触计算机，什么时候接触前端。
- 整个经历中，你认为最值得骄傲的事情，最难的事情是什么。
- 什么事情让你自豪，什么事情让你有挫败感。
- 未来的发展，自己的规划。
- 为什么现在才来秋招，不选择秋招。

拿我来说，我「**双非本科**」，最担心的就是卡学历。当时hr明确问了我，你对现在的学校满意嘛，师范类的院校，怎么选择走软件工程呀，诸如此类的问题很多，无不困扰着我。

不过我也不搞虚的。实话实话，为什么高中成绩差，又为什么选择软件工程，争取做到坦诚相待的态度，表达出自己最真实的想法吧。

小结

看到这里的话，可能对你来说，没有具体说明白，哪些知识点会考，哪些不会考察。

我面腾讯和阿里的时候，也没有看到我这个部门具体的面试题。

我觉得更重要的是，把握好「**专业技能和项目经历**」的表达，引导面试官去往你所知道的领域交流问题，最后在这个领域拿出自己的「**杀手锏**」，这样子面试官会更有好感，接下来的交流会更加愉快。

对简历感兴趣的话，可以参考下：

XXX

求职意向：前端开发工程师

电话：XXXXXXXXXX

| 邮箱：XXXXXXXXXX@qq.com

学历：XXXXXXXX

掘金等级Lv6，掘金共建者。

掘金主页：<https://juejin.im/user/2348212569517645>

GitHub主页：<https://github.com/daydaylee1227>

个人简介

网名“TianTianUp”，XXXXXXXXXXXXXXXXXX，目前掘金等级Lv6，“掘金优秀作者”，输出多篇JavaScript底层原理、浏览器原理、HTTP、算法与数据结构系列文章，文章阅读量50W,热爱开源，热爱分享，热爱前端技术。

专业技能

1. 掌握 JavaScript，熟悉 ES6 语法，对异步编程有一定研究。
2. 熟悉 React + Redux + TypeScript 技术栈，对 React Hooks 有实践，对React相关原理有了解。
3. 对浏览器的渲染原理、性能优化有一定研究。
4. 熟悉 TCP、HTTP 网络协议。
5. 扎实的数据结构和算法基础，对基础算法有深入研究。
6. 熟悉 webpack 的基本配置及从0搭建完整项目，了解 webpack 常见的打包优化方案。

教育经历

XXXXXXXX 软件工程 2017.09-2021.07

成绩:

- 专业排名前10%、共获得XXXXXXXXXXXXXXXX等数奖项。

竞赛成绩:

- XXXXXXXXXXXXXXXX银奖。
- XXXXXXXXXXXX铜奖。

实习经历 (XXX)


2020.09 - 2021.1 XXXXXXXXXXXXXXXX

- 【项目介绍】XXXXXXXXXXXXXXXX
- 【问题与优化】
 - XXXXXXXXXXXX
 - XXXXXXX
- 【思路与经验】
 - XXXXXXXX
 - XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

@稀土掘金技术社区

作为一个即将毕业的我来说，第一份工作很重要。在我面临选择的时候，慎重考虑以及找前辈交流过后，最后选择了「腾讯」。

在即将毕业的最后阶段，我应该算交上了一份满意的答卷，未来的路还很长，让我们拭目以待吧！

有想交流的问题，欢迎联系我 

微信:DayDay2021

最后做一个简单，自由，有爱的技术人吧。