

2022 年, 我终于如愿加入了字节跳动!

21届双非本, 有过三个月实习, 没有正式工作经验, 今年考研真的卷炸了, 只能回归找工作的行列, 通过内推有幸获得面试机会, 并成功夺得offer。

这篇文章是一份4000字的自我总结

- 面试经历 (主要是字节3+1)
- 个人总结
- 未来的一些计划

之后还会专门写一篇文章总结学习资源, 推荐一些宝藏博主和资源网站, 书籍等。

求个关注和赞~



面试经历

社招

字节跳动 三轮技术+一轮HR

我只着重写了面试过程中的一些重点部分, 其实还有很多细节, 大家一定要在短短的面试中, 能够展现出自己对于知识广度和深度的不同理解, 最好的情况当然是能与面试官进行技术分享与交流, 所以关注前沿技术, 工作中持续学习和总结是非常有必要的!

一面 (45min)

- 自我介绍

在自我介绍中要着重项目经验, 突出一下自己掌握熟练的部分

- 首屏渲染优化方式有哪些(懒加载, 预加载)

- 前端性能优化指标有哪些 (FP, FCP, LCP)

这一块感觉答得不是很理想\

- 什么是埋点, 如果进行性能测量, 具体指标等等

- SEO相关优化 (meta, h1, SSR等)

- 项目所使用技术的优劣势, 适用场景

- websocket 性能优化

- 后端性能优化指标

- 用过那些hooks, 自定义实现德hooks, 详细说一下

◦ 这里包括用户登录验证钩子, 表单验证钩子, 预加载逻辑钩子等。

- 登陆状态管理 session cookie token(jwt) 还有分发中心

- 算法题: 金额格式化, 3位一逗号, 没写好, 忽略了负号(蠢死了)

扩展遇到超级大的数字怎么办 (bigint转字符串)

- 内存泄露遇见过吗? 怎么处理 循环引用, 申请空间未释放导致, 可以通过chrome devtool 内存管理查看。

- 循环引用是什么, 产生原因, 怎么检查, 怎么处理

- weakmap, weakset

- css3 动画效果有哪些

- 反问 (学习建议, 发展方向)

第一次面字节, 也算是我第一次面大厂, 难免有点慌慌张张的, 面试官比较平和, 不过算法题没有考虑好多种情况, 我以为自己就要止步于此了, 好在两天后接到了二面的通知。 \

字节的一面考察的非常细致, 淘汰率也是最高的, 就我来说, 主要围绕项目展开, 然后延伸到基础, 一面主要以广度为主。

二面 (90min)

- 自我介绍
- 剩下的内容主要根据项目更更更深的展开
 - 图片上传方式
 - 图片类型的不同(base64 blob)
 - 图片格式不同 (jpg gif png jpeg webp)
 - 项目技术选择, 和优劣势
- 代码实现题: 异步限流器
思路是对上了, 但是有个返回值, 调试了半天没弄出来, 最后只能换了一题
具体代码实现可以参考 [这里](#)
- 算法: 版本号排序 (经典老题)

二面面试官人真的超级和善, 在一面的基础上, 继续对项目深挖, 但引导你深入。

非常感谢二面面试官给了个机会! 感谢不挂之恩!

三面 (60min)

终于到了技术终面, 老实说真的超级紧张!

- 第一题 是数组去重, 但是比想象的不太一样, 需要判断数组内对象键和值来进行去重。

js 复制代码

```
input: [{a:1, b:2},{a:1},{a:1}, {a:1, b:{c:1}}, {b:{c:1}, a:1}]
output: [{a:1, b:2}, {a:1}, {a:1, b:{c:1}}]
```

我的想法和做法是

实现了一个 JSON.stringify 来按键值排序, 然后set去重, 再用JSON.parse转换回来
然而面试官表示能不能不拷贝或者转换, 应该是想考察深比较, 然而当时没反应过来, 不过有提到可以使用循环遍历键值

- 第二题原型链
老生常谈了, 手写es5原型链, 也写了点es6, 没写完, 顺便扩展了一下 new函数, Object.create 还有 symbol等基础知识
- 第三题 React 用 react.createElement 实现 jsx 语法糖

- 第四题 接着深入, 考察 `reactElement`, `reactComponent` 还有 `fiberNode` 的作用
- 第五题 判断事件队列, 然而做错了两个点
 - 连续 `then` 会将后续的 `then` 函数排到微队列尾部
 - 其次 一定要注意, 第二次事件循环, 每处理一个宏任务就要清空一次微任务队列
 - 然后还问了链式调用其中的原理, `then` 的返回值相关的问题, 这里答得不是很好 害

总的来说其实5题都是常规题目, 有难度, 但没有特别难, 考察对于基础知识的掌握吧, 然后正中我的一些薄弱项, 很多概念有所了解但是并不熟练或者没有深入探索。

还是要加强对于基础知识的掌握程度, 焦急等待, 依旧是两天后通知了HR面, 呼, 算是松了一口气。

HR 面 (30min)

- 过去和目前的情况
- 后续安排
- 过去和现在拥有的offer相关
- 对于三次面试的感受和总结
- 对业务了解吗?
- 反问
- 介绍一些字节的业务和相关的工作安排
- 介绍后续安排
- 反问2

到此为止, 算是走完了所有面试的流程了。

依旧是两天后, 收到了oc, 并如愿拿到了offer, 等待入职中。

字节HR的效率真的超级快, 而且有问题回复也很快, 感谢hr的耐心解答和进度查询!

零碎的其他面试经历总结

2022.2 泛微

面试体验, 面试官迟到15分钟, 然后两个人面我, 有的时候还没说完就被另一位面试官打断。

总体来说, 八股文为主, 考察的很广, 面试下来还是修正了不少知识点, 也确实有很多遗忘的地方。

- 行元素和块元素区别 (不可设置宽高, 上下边距)
- CSS 垂直居中
- 为什么要清除浮动 (文档流, IFC, BFC)
- 项目运行中, 全局样式如何进行更替
修改全局CSS变量 修改全局类
- fixed与absolute能否互相定位, 定位基于什么元素?
- 脱离文档流的方式
- CSS 如何触发GPU渲染
- 重绘与回流
- transform 是否触发回流与重绘
- 标准模式和怪异模式的盒子模型
- ES6+ 都有哪些内容, 用过那些
- 箭头函数与普通函数的区别
- 闭包是什么
- React中闭包的运用有哪些(carry, 私有变量, 属性存储)
- 组件通讯的方式
- React router 路由传参的方式
url/params, search/query, browserRouter/state(不消失), hash会消失
hashrouter 基于 window.location, 不支持state 和 key
browserRouter 用的是 h5 history api
- 了解 React router v6的转变吗?
- 封装过那些组件
- react 16和17的区别
- react 生命周期 (细谈) 让你背出来函数名和顺序那种
- react hooks 的生命周期 (如何用 useEffect 模拟)
- 当 url 变更的时候如何触发特定事件 (路由守卫) hash onhashchange
history 监听popstate, 通过 event.state 获取state

2022.3 成都晓多科技

晓多需要到公司面试, 面试官非常和善, 会针对一些业务场景提出问题, 然后看你的思考和所给出的方案。

面试的部门是 AI客户的应答组。

一面

- 手写 (真手写) 数组去重, 防抖节流
数组去重这玩意其实根据业务场景变化特别大\
- 自我介绍
- 前端聊天记录如何存储
- 聊天记录的数据结构如何
- 如何减少前后端交流时间
- 如何缓存在前端
- 只用CSS如何让 H5页面 如何让一个图片始终处于页面特定位置
(minheight 结合 calc(100vh - 100px))
- 循环队列 EventLoop
- 如何学习?
- 学过什么
- 手写二分查找, 二分查找的原理

二面

主要以项目为主

- 介绍一下使用过的hooks
- 为什么要使用hooks
- 介绍 useState 和 useEffect 细谈
- 介绍 react 调度原理
- 介绍一下 fibernode

个人总结

个人经历

可以跳到下一点，没意义内容。

我的出身只是一个普通的民办学校, 但是也很幸运, 计算机学院有着一个非常优秀的实验室, 有着许多志同道合的友人, 许多优秀的前辈。

但其实我在大一面试实验室并没有通过, 但这并不能阻止前进的步伐。

于是开始了大学四年从0到1的征途。有幸在前辈的指导下,从C语言到数据结构,一步步打怪升级。终于在大二进入到了实验室,也由此开始了我的前端之旅。

ps:大家有机会一定要参与学校组织,或者在掘金,github参加社群,闭门造车不可取,很多时候需要交流,碰撞来不断地提升自己。

大二结束,已经有非常优秀的大四前辈进入了字节,去哪儿,58这些知名公司,这也无疑激发了我的斗志。

咸鱼也是要有梦想的,我也约定有朝一日能跟他们做同事。

大三开始一方面,我不断地巩固算法,参加竞赛,这对于代码能力的提升至关重要,老前辈说过:
代码能力就是,能将想法用代码表达出来的能力。只有通过多写多练,才能不断完善你的代码功底。

同时也不能放弃专业知识的学习,从html到js也是一步步积累,大三下开始跟着老师做项目,学习框架。大四秋招的第一份实习,大四春招的第一份offer,大厂简历筛选难以通过,又经历了考研的爆炸,力扣周赛的折磨。

时刻突破自己,跌倒了就站起来,摔了就继续爬,卷啊卷,最终也有幸卷进了字节,也算兑现当年的承诺。

经验总结

能力培养

我所认为的程序员能力包含:

- 代码实现能力
- 专业方向熟练度(语言了解程度,框架熟练度,基础优化)
- 科班基础功:四大基础课,设计模式,代码优化等

在经历了各种面试以后,我了解到,其实作为一名工程师也有很多其他需要提升的:

- 架构能力:技术选型
- 工程化能力:优化开发效率,包含排错,日志,埋点,性能指标,测试等等。
- 交流沟通能力,你会做,并且能教会别人也会做,这也是一种能力。

往后我也会对上述能力不断地进行加强。

学习方法

1. 任务驱动

一定要以任务驱动为主, 你要知道你做这些是为了什么而不是为了做而做。

比如前端的红宝书, JS高程, 并不是你一定要一字一句的去看透, 你也未必能记得。

如果是初学者, 建议在精读 [Javascript DOM编程艺术](#) 之后, 进行一遍略读, 只粗略的了解大概。

然后参与项目的实践开发, 在你不太理解, 或者需要用到时候, 进行翻阅, 这是效率最高的。

当然, 如果时间足够, 整体的过一遍还是非常有必要的, 尤其是框架相关的官方文档!

2. 面向面试

面试不同于看书和做项目, 算是一种变相的应试, 你需要刷面经!

面试内容代表了当下热点技术, 和公司部门的业务要求, 不论是在职还是应届, 我觉得都应该多看看面经, 了解行业主流需求。

3. 表达分享

尝试将自己学到的东西与他人分享, 你能够在这个过程中理顺自己的知识脉络, 提高沟通能力, 当然最主要的是:

- 在交流中发现一些自己的不足
- 获得一些你所不知道的只是盲区

第二点尤为重要, 只要相互交流, 才能发掘那些, 你从来没听说过的知识, 有可能打开新世界的大门。

当然表达的方式不仅限于对话, 也比如说文章视频等等。

4. 总结复盘和沉淀

不论是项目开发, 学习技术, 还是面试, 都必须要做总结, 往往只有总结详细的知识脉络, 才能记住知识。

而只有当你能够将这些知识进行输出, 运用到项目中, 这才是真正掌握了知识。

项目开发中的技术沉淀尤为重要!!!

项目开发

1. 边开发边思考

每次做内容, 一定要深入的思考, 为什么这样做, 选择这个技术甚至是文件格式, 为什么这样选。

而不是为了能够完成功能交付, 随便选择,

最简单的例子, 登陆验证的cookie, session, token 各有什么优缺点, 什么时候该选择什么, 是一个非常值得思考的问题。

2. 抽象迭代

这也是非常重要的能力, 当你意识到你的代码冗余, 或者泛用性低, 难以迭代的时候, 就应该思考了。

函数能否提高泛用性, 逻辑能不能进行封装, 代码能不能复用, 都是值得思考的问题。

3. 紧跟前沿

紧跟最新的热门技术栈, 你可以掌握到很多行业新的解决方案。

这些解决方案都会累积, 成为你往后的一笔财富。

比如 tailwind, graphql 等等。

4. 发现问题并解决问题

复刻是我们学习的第一步, 我们总是不断地复刻中成长, 这当然无可厚非。

当复刻仅仅是一种借鉴, 而真正有能力的人应该是发现问题并解决问题。

但往往大家缺乏的就是发现问题的能力, 热门的主流框架往往都是发现了业务中的缺陷和痛点。

未来的计划表

2022.3.25

- 掘金的技术文章能够有人看
提高文笔
提高技术内涵
- 能够在字节活下来, 毕竟这已经是一个全新的战场。
希望能过试用期
希望结识更多的友人, 前辈
- 能够在工作优化学习方法, 提高开发效率, 规范化
- 不断精进
周赛继续卷, 目前为止场均三题, 1200名左右
看书, 待看书目一直是满载状态
- 参与开源
一方面希望有自己的开源项目
另一方面能够参与到优秀开源项目的开发中
- 提高一下身体素质(锻炼健身)
- 魂系游戏能够冷静思考 (只狼可真《养生》)