

# 面试之工作做了什么

面试的时候有个最经典的问题就是

你在上家公司做了什么

考察业务匹配度 技术能力 架构能力 沟通（管理能力）解决问题能力 快速学习能力等

里面有个 发展时间线的概念 项目迭代的 过程中 你做了什么

写代码是为了解决问题的，要从深度 广度 业务配合去讲

最好提前画好架构图 配合架构图去讲

突出自己的 技术深度 广度 解决问题能力 跨部门合作能力 管理能力 协调能力 学习能力

## 第一阶段 从零到一 验证市场

可以突出开发效率，可以着重突出 当时为了效率 去牺牲的 更加优质的备选方案（可以没做过 但是一定要懂原理）

比如我当时开发的是 机器人监控模块，使用的是 canvas 和 ws 通信机制。其实备选的grpc 和 canvas 离屏渲染是没有做的

难点就是 canvas 实时渲染 监控机器人

（前端可以说canvas 渲染的难度 和优化

后端可以说和硬件通信的实时性）

（可以结合公司的业务去创造场景 但是不要太离谱）

## 第二阶段 适配市场 增改功能 同时优化一阶段的重要紧急方案

比如公司开始增加很多业务模块

前端开始增加 pda 端 机器显示端

身为核心员工怎么去最小成本去更改代码结构

人员扩张 怎么去分配员工工作 跨协调部门（突出管理协调能力）

前端考虑 多端代码的复用问题

比如客户的设备版本不匹配问题 突出广度问题

这个阶段是快和广

后端服务拆分 微服务层面

后端做代码生成工具等

(业务结合遇到的最难的开发问题)

### 第三阶段 项目复用问题

项目越来越多 B端差异化问题也越来越多

如何最小改动标准代码 以及项目版本升级成为最重要事项

开始做数据中台 业务场景拆分服务 同时留好口子适配后面的交付平台

前端 机器人端 和 pda 端 做动态组件库 和后端配合做sdk

单个页面/组件做到所见即所得 让前后端同学更注重本身开发 不用注重全流程

重点可以说一说架构的难点问题 配合业务场景说

还有代码复用 拆分 分包

(架构升级 等问题)

### 第四阶段 交付效率问题

交付项目越来越多 成立 交付部门，但是存在交付 需要熟悉开发环境 打包 配置 多次测试等流程

双方沟通成本 交付学习成本 很高

前后端代码架构继续拆分细化，前端使用微前端管理

前端可以说说微前端改造问题， 流程图开发 有bug 改了哪些源码

后端可以说说 配置化遇到的问题

后端最大化配置能力，运维需要做大量的适配工作

做交付平台 将所有流程细化 能力库 页面库 流程管理等，

交付人员利用能力库 页面库 流程管理等生成交付项目，

交付不用过度关注配置环境 配置文件等

开发不用过度关注项目流程等

# 第五阶段 稳定运维

做监控平台

数据分析平台

监控平台怎么做的 什么解决方案

数据分析怎么做的

前端怎么埋点 数据库怎么优化的