

腾讯Next专项工程师训练营 移动App专项导论

腾讯 victorhuang

自我介绍

- 从前： 搜索我



担心大家因为不了解我， 而耽误了学习知识

自我介绍

- 职场
 - 微软、腾讯
 - 总监
 - T4



1. 2015-2016明星讲师
2. 2017-2018优秀出品人，百度，阿里，硅谷创业公司，优酷等
3. 《android移动性能实战》作者

自我介绍

- 职场
 - 微软、腾讯
 - 总监
 - T4
- 职场之外
 - QCon（全球开发者大会）
 - 作者
 - 两个娃的爹



1. 2015-2016明星讲师
2. 2017-2018优秀出品人，百度，阿里，硅谷创业公司，优酷等
3. 《android移动性能实战》作者

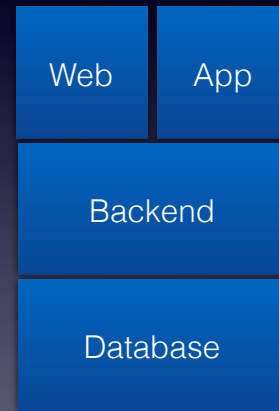
开发工程师的物种分类？

开发工程师的物种分类



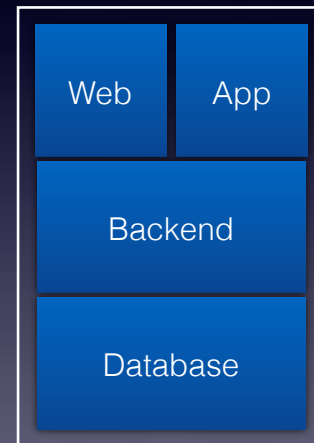
Web, App, 后台, 数据库的工程师, 还有那个大包大揽的全栈工程师

开发工程师的物种分类



Web, App, 后台, 数据库的工程师, 还有那个大包大揽的全栈工程师

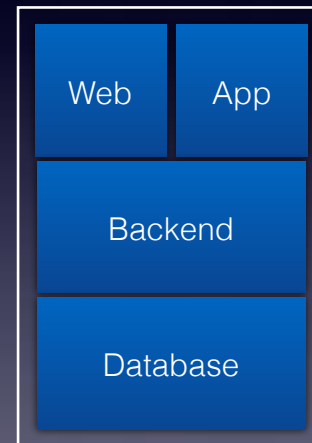
开发工程师的物种分类



全栈

Web, App, 后台, 数据库的工程师, 还有那个大包大揽的全栈工程师

开发工程师的物种分类



全栈

他们要面对什么相同的问题？

Web, App, 后台, 数据库的工程师, 还有那个大包大揽的全栈工程师

两个问题

- 工程效率
- 专项深度

简单说下工程效率

两个问题

- 工程效率
- 专项深度

简单说下工程效率

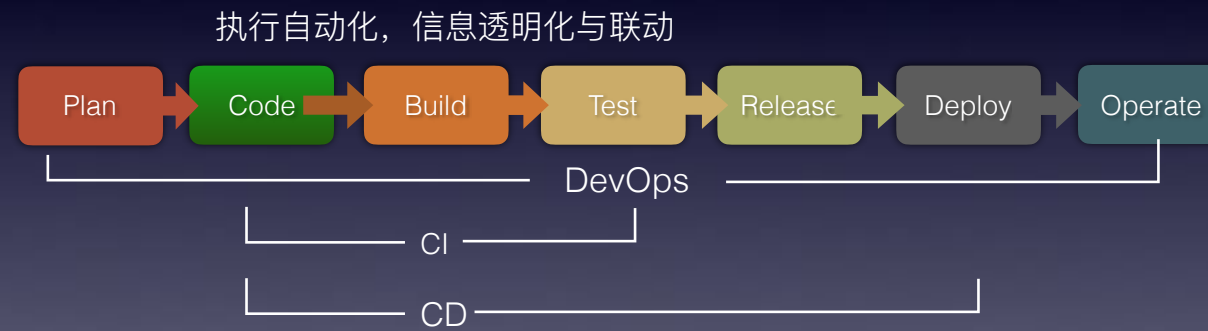
工程效率



谷仓效应

- 角色之间
- 部门之间
- BG之间
- 系统之间

工程效率



我们叫测试开发、工程效率部门，他们会通过Devops，CI，CD去解决实际上工程实践的效率问题。

Plan，需求规划 Code，编码（写代码） Build，代码集成（集成成sdk或者app）

Test，测试 Release，发布 Deploy，部署 Operate，运维

CI，持续集成， CD，持续部署

相关链接：<http://insights.thoughtworks.cn/instantiate-the-principles-of-devops/>

两个问题

- 工程效率
- 专项深度

简单说下工程效率

两个问题

- 工程效率
- 专项深度

简单说下工程效率

专项深度

- 安全
- 性能
- 稳定\兼容
- 音视频



1. 安全：WEB XSS，数据库 sql注入，移动app 二次打包，后台 隐私安全，哪怕是底层的虚拟化技术还有Intel幽灵和熔断
2. 性能：无论哪种语言都会有内存泄漏，常驻、哪种前端都会有卡顿，无论后台前台都要考虑能耗
3. 音视频：采集、解码、编码缺一不可，同时也贯穿了后台，前端，移动App

相关链接：

<https://www.kernel.org/doc/html/latest/admin-guide/l1tf.html>

专项深度

- 安全
 - 性能
 - 稳定\兼容
 - 音视频
- XSS攻击、SQL注入、二次打包、隐私等



1. 安全：WEB XSS， 数据库 sql注入， 移动app 二次打包， 后台 隐私安全， 哪怕是底层的虚拟化技术还有Intel幽灵和熔断
2. 性能：无论哪种语言都会有内存泄漏， 常驻、哪种前端都会有卡顿， 无论后台前台都要考虑能耗
3. 音视频：采集、解码、编码缺一不可， 同时也贯穿了后台， 前端， 移动App

相关链接：

<https://www.kernel.org/doc/html/latest/admin-guide/l1tf.html>

专项深度

- 安全
 - 性能
 - 稳定\兼容
 - 音视频
- XSS攻击、SQL注入、二次打包、隐私等
 - 内存泄漏、内存常驻、卡顿、耗电等



1. 安全：WEB XSS， 数据库 sql注入， 移动app 二次打包， 后台 隐私安全， 哪怕是底层的虚拟化技术还有Intel幽灵和熔断
2. 性能：无论哪种语言都会有内存泄漏， 常驻、哪种前端都会有卡顿， 无论后台前台都要考虑能耗
3. 音视频：采集、解码、编码缺一不可， 同时也贯穿了后台， 前端， 移动App

相关链接：

<https://www.kernel.org/doc/html/latest/admin-guide/l1tf.html>

专项深度

- 安全
- XSS攻击、SQL注入、二次打包、隐私等
- 性能
- 内存泄漏、内存常驻、卡顿、耗电等
- 稳定\兼容
- crash、闪退、花屏
- 音视频



1. 安全：WEB XSS， 数据库 sql注入， 移动app 二次打包， 后台 隐私安全， 哪怕是底层的虚拟化技术还有Intel幽灵和熔断
2. 性能：无论哪种语言都会有内存泄漏， 常驻、哪种前端都会有卡顿， 无论后台前台都要考虑能耗
3. 音视频：采集、解码、编码缺一不可， 同时也贯穿了后台， 前端， 移动App

相关链接：

<https://www.kernel.org/doc/html/latest/admin-guide/l1tf.html>

专项深度

- 安全
- XSS攻击、SQL注入、二次打包、隐私等
- 性能
- 内存泄漏、内存常驻、卡顿、耗电等
- 稳定\兼容
- crash、闪退、花屏
- 音视频
- 视频卡、模糊，音频响度不足等



1. 安全：WEB XSS，数据库 sql注入，移动app 二次打包，后台 隐私安全，哪怕是底层的虚拟化技术还有Intel幽灵和熔断
2. 性能：无论哪种语言都会有内存泄漏，常驻、哪种前端都会有卡顿，无论后台前台都要考虑能耗
3. 音视频：采集、解码、编码缺一不可，同时也贯穿了后台，前端，移动App

相关链接：

<https://www.kernel.org/doc/html/latest/admin-guide/l1tf.html>

专项深度

- 安全
- XSS攻击、SQL注入、二次打包、隐私等
- 性能
- 内存泄漏、内存常驻、卡顿、耗电等
- 稳定\兼容
- crash、闪退、花屏
- 音视频
- 视频卡、模糊，音频响度不足等

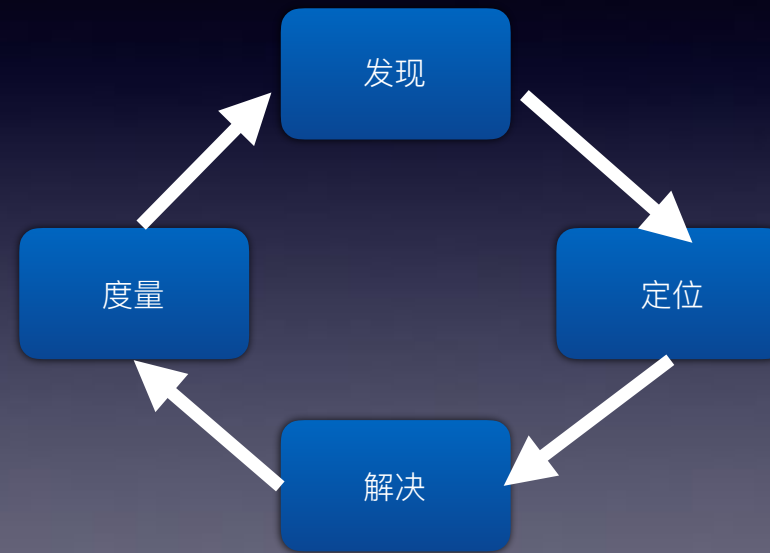


1. 安全：WEB XSS，数据库 sql注入，移动app 二次打包，后台 隐私安全，哪怕是底层的虚拟化技术还有Intel幽灵和熔断
2. 性能：无论哪种语言都会有内存泄漏，常驻、哪种前端都会有卡顿，无论后台前台都要考虑能耗
3. 音视频：采集、解码、编码缺一不可，同时也贯穿了后台，前端，移动App

相关链接：

<https://www.kernel.org/doc/html/latest/admin-guide/l1tf.html>

课程结构与基本方法论



发现：一个缺陷被发现需要什么条件？

案例：发现静态头像挂件I/O Buffer设置太小的条件



触发I/O加载

1. 打开聊天窗口
2. 有朋友带静态头像
3. 没有缓存

发现：有量化的数据

	调用方法	耗时(ms)
MI2S (android 5.0)	decodeFile	26-49

能发现是问题么?
😞

发现：有量化的数据

	调用方法	耗时(ms)	读磁盘次数
MI2S (android 5.0)	decodeFile	26-49	108

发现：有量化的数据

完整

	调用方法	耗时(ms)	读磁盘次数
MI2S (android 5.0)	decodeFile	26-49	108

一个缺陷被发现需要有什么条件

- 核心难点
 - 让它出现，如何覆盖各种用户场景？
 - 看得见它，如何采集可量化且完整的数据？

定位

- 至上而下：解Bug，交互类性能定界，定位
- 至下而上：用于性能优化
- 核心难点：变化大



1. 前面的I/O问题就已经进行了定界，一般来说解单点Bug，后面说的交互类性能（GC、CPU）问题会这样
2. 至下而上，用于挖掘性能优点的点

解决

WebP优化

- Tips, 思路, 坑
- 核心难点: 性价比

压缩率:

耗时、清晰度、流量、CPU

↑

不同机型

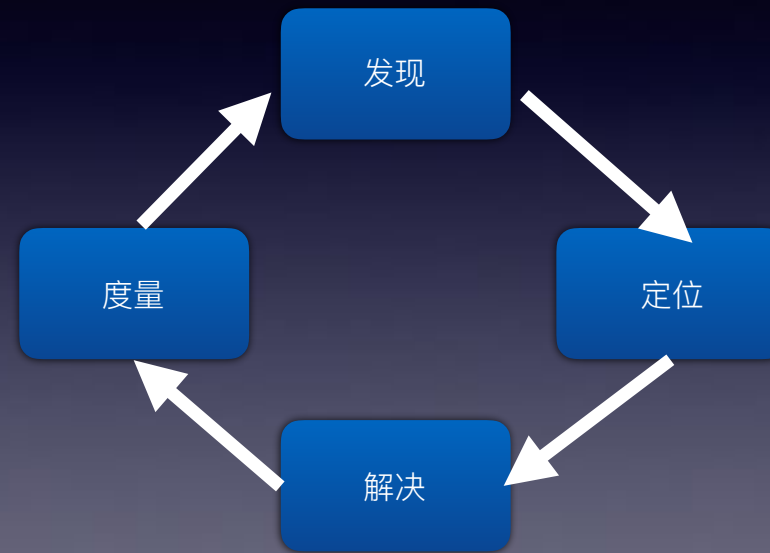


度量

- 跟发现相似，完整&量化
- 核心难点：指标如何贴近用户的真实体验

例如到了稳定性，我们会看一下，真假crash率，看下我们是怎么逼近用户真实遇到的crash的

课程结构与基本方法论



最后我们来看看有什么作业

作业：移动App专项攻防战

- 5人一组
- 防御：埋入缺陷，输出Demo源码与缺陷列表
- 攻击：挖掘缺陷，输出缺陷报告，修复后的Demo
- 总结：输出ppt

作业：移动App专项攻防战

	评核标准	评分占比
移动App埋入缺陷	8个类别的缺陷每个类别至少埋入一个	25%
移动App定位并解决缺陷	能正确分析并解决至少5个缺陷	15%
移动App演示文档	展现出从埋入缺陷的巧思，分析或者解决缺陷的思路和技巧	20%
后台App定位并解决缺陷	能正确分析并解决全部缺陷	25%
其他	代码与相关文档的整洁、逻辑清晰	15%

相关链接

- [QConBeijing 2015 手Q性能专项最佳实践](#)
- [Google Android Performance Patterns](#)