

背景：

Wxx是H厂公司级移动办公平台，融合了邮件，IM，通讯录，视频会议，直播，审批等办公场景，全球200多个国家的员工每天都在使用。研发团队共有100多位工程师，由H厂员工和外包人员组成。超过40个独立业务团队，10多个BU参与贡献代码。

目前困难：

- 1. 项目业务多，需要跨团队协作，代码容易冲突。
- 2. 测试效率低，目前用手动打包，手动测试。
- 3. 发布效率低，手动打包手动上传，全部rd需要等待包上传才能下班。
- 4. 运维效率低，用户反馈问题无法定位，rd团队之间互相推诿。

方案分析

方案分析

- 1. H产品功能界限明确，有邮件、IM、通讯录、视频会议、直播、审批等各种场景。有100多个rd和40个团队同时贡献代码。可以采用组件式方案。App平台化。
- 2. 目前QA都手动测试，因为有明确业务可以梳理每个业务的核心回归点保障App在大部分场景是正常的。可以采用Appium自动化测试框架
- 3. 目前是手动打包效率低容易出错，可以采用FastLane，自动打包上传方案。业务和部门之间梳理负责人，负责业务的稳定构建。
- 4. 运维效率低和用户反馈没法定位，可以采用底层用户监控框架，分析用户行为和设备各种状态。

组件试方案

1.CocoaPods管理整个工程，梳理业务模块

业务线	负责人	涉及到的库	工作量排期
IM	A-rd	IMBusiness、IMSDK、IMFoundation	XX
通讯录	B-rd	ContactListSDK	XX
视频会议	C-rd	VideoMeetingSDK、AVChatSDK、AVChatFoudation	XX
邮件	E-rd	EmailSDK	XX
审批	F-rd	ApproveSDK	XX
直播	G-rd	LivingBusiness、AVChatSDK、AVChatFoudation	XX
平台化	H-rd	WXFoundation、WXUIKit、WXRocket	XX

- IMSDK、IMFoundation作为底层IM模块设计，IMBusiness用IMSDK实现业务逻辑，IMSDK、IMFoundation不能编写业务代码。
- AVChatSDK、AVChatFoudation作为底层视频模块设计，VideoMeetingSDK和LivingBusiness用这两个SDK实现业务代码，AVChatSDK、AVChatFoudation不能编写业务代码。
- WXFoundation是原工程梳理出的代码工具库、WXUIKit是梳理出来的通用UI组件库、WXRocket为底层监控SDK。

2.根据排期各业务方把跟各自业务代码都先沉库到Development Pods，多仓库管理避免代码冲突。

3.各个业务方业务的调用尽量使用Route的解耦方式跳转业务。

4.各个业务方如果有依赖点可以先梳理出来，如果是公用模块可以给平台化rd处理。

App平台化

1. 和产品负责人梳理每个版本要合入那些需求。
2. 每个版本有固定的合码时间比如下午6点前。
3. 每个版本都有轮值RD对这个版本所有问题负责跟进，分发问题，跟进问题修复情况。
4. 把控每次版本合入的代码质量，codereiew、代码静态检查、包大小等。

自动化测试

1. QA和业务方负责人梳理每个业务的回归路径。
2. 采用Appium自动化测试框架，每次打包发布之前都跑一遍用例保证，保证关键功能稳定。

自动化打包平台

1. QA采用Jenkins打包并且自动上传到TestFlight or Appstore.
2. QA和产品确定产品的发布周期，确定每周固定那天打包。
3. QA打包过程中出现问题及时反馈值班RD，值班RD跟进打包问题直到出包上传成功。

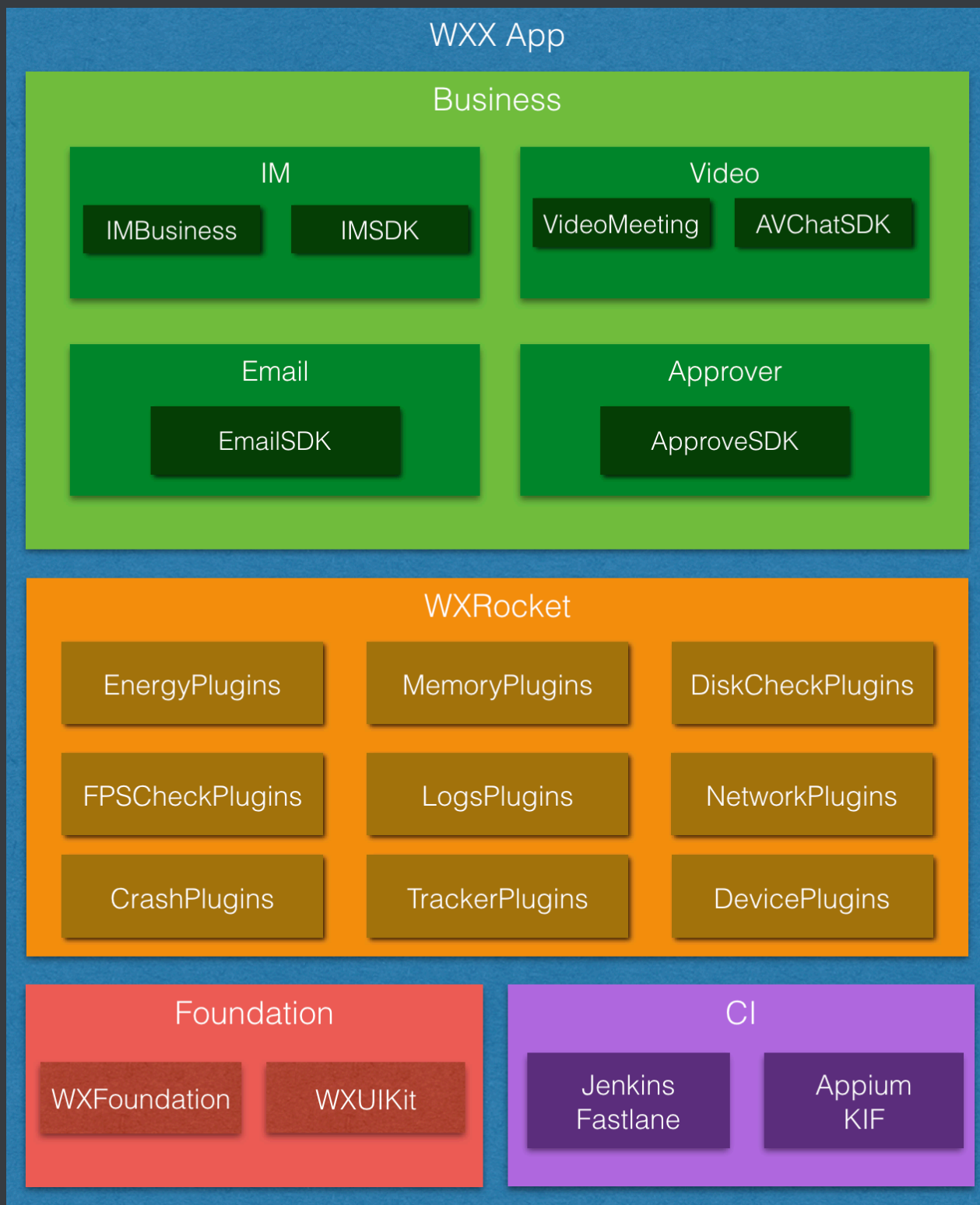
用户反馈

1. 每台设备都生成DeviceID和UserID，方便跟进个案。
2. 重构用户反馈页面，问题分类和常见问题方便用户分类问题反馈处理更精准。
3. 常见问题默认有返回文案解决方案，添加有无用按钮给用户反馈是否有用。
4. 如果有必要可以接入客户机器人和在线客服（比如网易七鱼）
5. 运维实时观察用户反馈数，如果有同一问题反馈剧增要及时同步到值班RD去跟进问题。
6. 业务负责人梳理每个模块的关键指标，监控数据是否异常。如果有异常要及时跟进处理。
7. 遇到很难定位的问题，可以让运维连续用户，请求用户协同解决，同时可以赠送配合用户些周边或者购物卡等。
8. 用户反馈时上传日志等追查信息。

监控平台

1. 电量监控
2. Crash日志监控
3. Log日志收集
4. 埋点数据收集
5. 电量监控
6. 磁盘监控
7. FPS监控
8. 设备信息
9. 网络监控

整体架构图



扩展思考

深入沟通了解研发流程中是否会遇到其他问题，一个需求的合入会经过、PM产出需求文档、需求评审、UI和UE设计、开发评审、开发研发、需求方验收、UI和UE验收、QA验收、需求合码、线上跟踪、数据分析等。

比如目前rd经常在UI和UE验收过程中花费很多时间，经常会导致需求delay。我们可以分析这个过程的问题，是否这个环节可以优化。比如UI和UE出方案时就跟RD同步，RD可以在一些疑问点上抛出来，UI和RD评估设计和工作量，最后取得一个平衡点。如果尽量做到信息同步的话在UI和UE验收上可以更顺畅，从而优化整个研发速度。