

多设备任务中心 - 任务流转

功能：一个应用进程可以从一个用户的终端流转至另一个用户的终端，举例：游戏打到一半流转到另一个人的手机上

埋一下伏笔：是投屏还是分布式

背景：这个特性是鸿蒙分布式功能的一个体现，核心特性；花费了很多的精力宣传，互联网舆情也很高，有很多大v也进行体验，问题：使用率很低，NPS数据不抢眼

任务：对这个特性进行全面的分析：最主要的目的，找出影响特性的软件层面的体验问题

有个误区：找出软件层面的问题不代表找出所有问题，这是不可能的；我们的目的，以点到面，通过一些离散的问题，我们在进行问题溯源分析后，找出底层的原因，然后以漏斗的形式，用户体验的维度进行问题的梳理，给开发、测试、营销一份有指导性意见的报告

如何去做

- 产品背景，描述一些关注的指标
 - 用户使用流程

设备连接(满足条件) - 发起流转 - 接收到流转 - 调用资源使用app - 断开连接

底层逻辑：调用的是谁的资源
 - 使用率：<1%
 - 在进行横向拆解时，首先关注机型，只有xxx机型可以使用，技术问题(但是不符合用户逻辑)，最重要的原因便找出来了
 - 归根结底，就是研发没有考虑用户使用逻辑及使用的场景，造成了设计特性却没办法让用户大规模使用的情况
 - 同时，我们也总结了互联网热度及舆情量，来说明用户对这个特性是关注的，加强问题性
 - 好评率，使用频次：比较高，说明对于现状还是比较满意
- 用户画像

初步建立用户同理心，站在用户角度感受用户体验的问题

用户特征：Mate/P系列用户的特征吻合，收入很高，一线白领

使用特征：但使用时长没那么高，集中使用在娱乐场景

但通过对平板，我们发现办公场景也很多
- 场景的总结
 - 娱乐：视频、游戏
 - 办公学习：文档编辑、读书
- 分析体验问题
 - 总体分析：梳理主路径，各个部分的转化率，体验五要素(功能分析)
 - 单问题分析：
 - 舆情情况，严重程度
 - 问题发生的场景
 - 发生的原因分析：技术、交互
 - 解决方案：站在用户的角度来看，当然评估技术实现的可能性

输入法、耳机
- 总结所有问题发生的原因：在调用资源时，调用规则不统一，包括硬件上(处理器、摄像头等)以及软件上(分辨率、软件功能-输入法)
- 问题跟踪、闭环