

1.分案逻辑添加产品期限的维度

1.1分案

步骤: 查看源码,了解分案逻辑以及分案条件配置(需求中一个用户可以指定一个产品期限或分多个产品期限需要可控的),并根据要求在分案逻辑上面添加产品期限维度 耗时: 3, 7h(如果不进行重构,则时长为3小时,进行重构则预计要7小时,同时包括员工上传表格接口以及绩效统计脚本)

PS: 经过初步的讨论, 目前每个国家要实现的分案情况不同,比较好的处理逻辑是 分案和绩效统计都使用一样的代码,以上传员工表格开始区分分案情况,并在该接口做数据校验.好处是需求可以自动变更,而且需求变更并不影响分案和统计的逻辑

1.2员工表上传

1.2.1员工表筛选

步骤: 数据库新增产品期限字段, 并在筛选逻辑和返回结果添加期限展示 耗时: 0.5h

1.2.2上传/导出员工表

步骤: 上传员工表和导出员工表添加产品期限处理逻辑 耗时: 0.5h

1.3绩效--查询/导出绩效

1.3.1web接口

步骤: 添加产品期限的查询和在结果添加产品期限的展示(导出绩效也在同一个逻辑里面) 耗时: 0.5h

1.3.2celery脚本

步骤: celery统计绩效脚本添加产品期限维度 耗时:1h

1.4分案查询

步骤: 添加产品期限的查询和在结果添加产品期限的展示 耗时: 0.5h

2.印度马甲包

2.1 快速查看相关改动

步骤: 全局搜索app/web product相关代码,分析是否需要改动 耗时: 1-2h

2.2 更改相关改动

步骤:

- 根据上面筛选到的需要改动的地方进行更改(经过分析, 改动点只有2-3个点,完成后通过自测代码覆盖率再次检查)
- 首先在数据库product表添加马甲包名称,并在其他产品相关配置进行数据增加(比如逾期等级等)
- 从数据库改动回溯代码相关逻辑部分,并进行检查和更改 耗时: 3h(这里不包括上面的自测代码覆盖率)

2.3 mq_listener兼容马甲包

步骤: mq产品配置更新 耗时: 0.5h

2.4 celery脚本兼容马甲包

步骤: copy印度的脚本并根据需求修改脚本 耗时: 1.5h

2.5 需求列表列出的需求兼容马甲包(可能也会被步骤2.1,2.2覆盖)

步骤:

- 查看上次员工表逻辑是否兼容一个员工不存在两个员工表中
- 绩效表逻辑检查与改动
- 案件表逻辑检查与改动
- app相关逻辑检查与改动(这里跟app要个新旧app包进行自测,速度比较快)

耗时: 2h