

转正申请

述职报告:

本人从 2016 年 3 月 11 号入职 55 海淘担任研发工程师一职，到现在近 6 个月时间，在此期间参与了搜索数据服务研发、minshop 业务研发，担当 mall 的 pc 端和运营后台开发，目前接手 app 后台和 app 运营后台的开发和维护，现详细阐述这段时间的工作内容。

3 月份：参与 mall 和搜索数据服务研发。

1. 开发搜索管理系统热词功能。
2. sku 和 spu 数据分析，构建数据库。
3. 实现 sku 和 spu 到 mysql 的数据插入、更新接口。
4. 分析 sku&spu 数据导入业务需求，考虑导入过程中的相关统计业务。
5. 重构当前 sku 和 sp 数据导入模块。
6. 完成 sku&spu 数据处理模块重构，添加数据更新历史记录模块框架。
7. 添加对 json 数据源/data/product 监控模块。
8. 添加 mysql-->scd 的转换模块。

4 月份：参与搜索数据转换体系和 minshop 业务研发。

1. 实现 scd--->mysql 增量更新 (care 字段更新)。
2. 建立操作记录表，添加操作记录模块，便于跟踪数据处理的操作和耗时
3. mysql 导出 scd 数据实现自动导出功能 (操作记录表中记录上次自动导出的时间戳)。
4. 配置信息转移到 config.xml
5. 对数据转换模块进行了分表处理，分表粒度可在 config.xml 中配置，目前 spu-128, sku-256
6. 对数据转换导出进行了测试，并 fix bug。重要 bug: 1. 数据库 wait_timeout 导致数据可能更新失败--- 添加再次提交更新 机制， 2. 一次 insert 事务可能存在相同的 DOCID，导致 insert 失败--- 全部采用 duplicate 插入/更新 3. 部分数据未进入数据库，原因是 status!=0，导致数据不会入库---第一次采用不 care status 的方式更新数据
7. mysql 导出 scd 分 2 个进程执行。
8. 添加数据转换系统的 比对 模块。
9. 测试数据转换系统，解决 bug: (1) 价格不能用 float 表示，换成 decimal; (2) select 操作失败也需要进行回滚。
10. 用 nginx + uwsgi + gevent + flask 搭建基于本地 sku/spu 异步 web 服务; 熟悉了 tornado 和 django 框架。

数据转换调用成果：

1. json--->scd--->mysql 自动监控路径，完成数据转换
- scd-->mysql 提供多种接口：
- (1) 全更新，存在 docid 替换该 item，不存在则插入
 - (2) 增量更新，存在 docid，比较关心 field 不同时，更新 field，不存在则插入

2.mysql-->scd 数据导出功能, 包含:

(1) 自定义导出: sku/spu 按时间范围自定义导出

(2) 增量导出: 根据上次自动导出日志 导出

3. 查询功能: 基于 docid 查询 sku 和 spu, 并按照指定格式给出 success/failed 结果

4. 操作记录: 记录相关操作内容到数据库中 (30%目前计划以较粗的粒度实现)

5. web 服务: 整合 1.2.3 功能, 对外提供 web api

目前已开启 monitor /data/product_data 服务

2016-4-13 操作分析:

scd-->mysql: 2 小时 15 分

sku 1 小时 50 分钟

spu 25 分钟

sql 导出 scd (status==0): 20 分钟

sku-->6 分钟

并行

spu-->20 分钟

5 月份: 参与 minshop 业务研发, 担当 mall 的 pc 端研发。

1. 封装 brand、seller、category 和重庆数据对接, 实现导入和定时同步功能。

2. minishop 热门专题采用 mongodb 管理, 实现添加和修改专题功能。

3. 添加商品 类目纠错功能。

4. scdTranCore 代码和业务熟悉。

5. brand/seller/category 数据同步 添加 手动 active update 功能, 测试和部署。

6. minishop 热门专题 添加上传图片, 返回图片 url 接口, 测试和部署。

7. correctCate 对 scdTransUni 开放接口, 添加批量更新接口。

8. editorManagerProduct (运营商品上下架管理) 需求分析和框架搭建。

9. editorManagerProduct (运营商品上下架管理) 完成编码调试。

10. brand/seller/category 数据同步 添加 验证更新后是否成功功能。

11. 升级 mall.search 中的前后端 html 渲染方式, 使用 tornado 中的模块方式。

12. mall.search 模板升级。

(1.1) search 页面模板渲染的数据接口提供

(1.2) 页面使用 tornado 模板进行 html 渲染

(1.3) url 正则规则定义

13. 协助前端完成 运营上下架 联调。

14. 添加批量更改商品类别功能。

15. mall.search 模板升级

(1.1) search 和 info 页面模板渲染升级

(1.2) url 中的规则更改, 去除?、&、中文等字符

(1.3) 通过 redis 缓存将 seller 参数转换为 id

(1.4) dev1 上部署, 并进行基本测试和压力测试

16. 添加 brand 等数据同步的单项和单条数据更新接口

6 月份: 担当 mall 的 pc 端开发。

1. mall.search 模板升级

(1.1) 前端渲染优化, 去除冗余数据

(1.2) online1 和 online4 部署

(1.3) brand 中文和特殊字符转换

- (1.4) 压力测试
- 2.mall 模板升级：
 - (1.1) 解决跨域 cookie 验证功能，和重庆联调上线 mall 版面页
 - (1.2) mall.search 部署上线，8 tornado 进程 * 2nginx，压测 连续单次 200 并发，响应全部正常
 - (1.3) 维护 mall 页面：首屏 selle/brand 全展示，createCard，price 小数点，列表页 editTitle 判断，card 中商品数量,sup 特殊上标 等
- git 路径： suyuan@121.43.200.103:/data/suyuan/gitsvr/mallTemplateManager.git
- 3.新版 mall 首页：
 - (2.1) 面向运营页的需求分析、框架搭建，数据库设计
 - (2.2) navi 导航栏后端接口实现，-->运营页和面向 homepage 页
 - (2.3) 分类楼层 subject 后端接口实现， -->运营页
 - (2.4) 超值单品 后端接口实现
- git 路径： suyuan@121.43.200.103:/data/suyuan/gitsvr/mallTemplateManager.git
- 4.brand、seller、category 数据同步 bug 修复
- git 路径： suyuan@121.43.200.103:/data/suyuan/gitsvr/cateDataChangeNotify.git
- 5.构建并优化 log 通用模块
- 6.navi 导航栏运营前后端优化测试：
- 7.分类楼层 运营前后端 编码实现；
- 8.专题页后台编码实现；
- 9.navi 导航和分类楼层给出采用 lua 实现的 homepage 接口；
- git 路径： git@121.43.200.103:/home/git/code/cateOptSys.git

mall 对比 JD 性能报告：10t*100c

指标	Mall	JD
Transactions	1000 hits	963 hits
Availability	100.00 %	96.30 %
Elapsed time	18.39 secs	12.58 secs
Data transferred:	76.91 MB	62.77 MB
Response time:	1.60 secs	0.62 secs
Transaction rate:	54.38 trans/sec	76.55 trans/sec
Throughput:	4.18 MB/sec	4.99 MB/sec
Concurrency:	86.80	47.67
Successful transactions:	1000	963
Failed transactions:	0	37
Longest transaction:	4.27	10.16
Shortest transaction:	0.19	0.10

7 月份：担当 mall 首页，二级页以及运营后台研发。

- 1.navi 导航、4 个广告页、超值单品-> 面向 mall 首页 lua 接口编写
- 2.navi 后台前端优化，4 广告页运营前端开发
- 3.mall 首页模板框架搭建，配合前端开发调试首页
- 4.mall 首页 dev 和 online 的部署

遇到的问题处理:

>由于 navi 导航之前采用 mysql 关系型数据库, 导航栏数据具有较强的层次关系, 导致数据库和后台接口设计的略显复杂, 前端处理难度相对较大;

>现在 navi 已经改成使用文档型数据库 mongodb, 前后端进行了重新开发, 首页、列表页、详情页都做了对应修改;

重点: 但是目前存在一个问题, 同一分类下的数据只能一个人修改, 比如同时多人修改服饰分类下的数据, 将存在数据丢失;

5. 整合后台和首页, 添加 mallHome 的发布功能, 修复分布式环境下的发布 bug

6. 添加新品推荐 lua 接口。

7. 修改官网大促后台接口, 开发官网大促运营页, 维护其他运营页。

8. 详情页和列表页添加 navi 导航数据接口。

9. navi 模块重新开发。

10. 分类楼层前端接口实现。

11. 后台优化, 解决多用户操作问题。

12. 提供专题页数据接口, 优化专题页后台。

13. scdTrans 分类优化, 修改为从内存中获取 correctCategory。

14. 新品推荐接口实现, mall 首页开发完善。

15. 解决选品数量匹配问题, 修改首页选品数据缓存机制。

16. 专题页数据接口实现, 添加专题页数据渲染项。

17. 搭建数据聚合模块框架。

18. 新品推荐后台前端实现。

19. 旧版分类楼层 bug 修复。

20. 按照设计稿开发新版专题管理页面。

21. 修复分布式引起的发布 bug。

22. 按照设计稿开发新版分类楼层页面。

23. 新版首页和列表页部署上线, bug 修复。

8 月份: 担当 mall 首页二级页研发, 并进行性能优化, 接手 app 的研发工作。

1. 首页和列表页 实现 redis、mongodb、http 异步调用。

2. 项目重构, 采用 mvc 模式分离逻辑。

3. 优化展示页面, 避免 404 出现。

4. 品牌和商家页面 开发。

5. 开发日志展示功能。

6. 聚合首页数据, 优化展示速度。

7. 使用 fabric 构建智能发布、撤销功能, 纳入的项目包括 mallhome、mallsearch、lua 数据服务、运营后台。

8. 增强页面容错性, 避免列表页商品信息缺失导致的 404。

9. 全部品牌和商家后台 页面开发, 商家单品数量、品牌单品数量 同步。

10. SEO 优化, limit 去除 等。

11. 优化管理后台页面用户体验, 包括官网大促添加顺序切换、各个主题的图片展示等。

12. 修复分类楼层排序 bug。

13. 修改官网大促数据结构, 添加二级页展示。

14. 实现奥运活动页后台, 包括抽奖机制, 奖品后台配置, 名单管理后台, 抽奖接口等。

15. 实现超值单品二级页后台。

16. mall 长文专题后台开发, 添加点赞, 阅览等功能。

17. 熟悉 app 的相关代码，部署和发布，了解服务器的管理情况，熟悉网管、社区管理、用户管理的相关逻辑。

18. 接手 app 后台，添加分享积分接口，修改积分累加逻辑。

未来计划

未来本人继续完善目前手上 mall 和 app 的开发工作，主要是研发 mall 和 app 的活动功能，完善社区的相关体系，根据产品需求研发更多的可运营内容。于此同时还将计划开展 Meetstyle 的研发工作，参与搜索的调优和业务功能研发。

个人感受

我的感受是作为一个研发人员，需要踏踏实实的对待每一个功能需求，为自己的每一行代码负责，于此同时，要善于总结和创新，定期的去 review 自己的代码，将 bug 扼杀在摇篮中，对于性能和用户体验的提升要精益求精。致力于成为国内最顶尖的研发工程师。

个人改进建议

由于工作繁忙，没有更多的参加公司的一些小活动和知识分享会，以后应在保证工作质量和效率的同时，多参与到同事的分享中，自己也经常和同事之间分享前沿技术和优秀实现方案。

姓名：苏远

工号：123

部门：技术部 研发工程师

地址：上海徐汇区古美路 1582 号总部园二期 B 栋 3 楼

手机：13636672480

邮箱：suyuan@55HaiTao.com

55 海淘 55mall 55 闪淘

55海淘
55haitao.com