

结对项目之需求分析与原型设计

031402402 曹鑫杰

031402428 鄢继仁

结对项目之需求分析与原型设计

一、需求分析----- (NABCD 模型)

N (Need, 需求)

- **信息收集繁琐:** 系负责人下发导师候选名单, 每个学生填报完提交给年级负责人, 再汇总发给系负责人, 过程繁琐。
- **学生获取信息有限:** 学生了解导师具体信息的渠道有限, 给填报志愿造成了困扰。
- **导师无法选择学生:** 导师往往只能被动分配到学生, 对学生的期望人数也各不相同。
- **分配结果可能不理想:** 系负责人通过一种复杂而说不清道不明的人工排序和安排算法, 统一给每个学生分配导师, 结果可能不理想。

A (Approach, 做法)

我们觉得采用 APP 的方式来实现。

- 同学使用学号注册登录, 查看导师信息, 填写 5 个**梯度**志愿的导师。
- 老师使用相应的账号登陆, 填写自己期望的**学生人数区间**, 查看填报了自己的学生的信息以及完成**选择**。
- 最后再由后台完成分配。

B (Benefit, 好处)

- 省去了人工收集信息的过程。
- 学生更方便了解导师课题选择和研究方向。
- 导师可以了解学生, 设定期望的学生人数。
- 学生和老师能双向选择, 兼顾了双方的意愿。

C (Competitors, 竞争)

- **优势:** app 端方便灵活, 后期可以增加更多功能, 本身还可作为一个版块附在其他 app 上。
- **劣势:** 需要下载, 且使用一次后就没用了。

D (Delivery, 推广)

- 课附在其他 APP 上(福大教务通、福大易班), 作为一小个功能。
-

原型设计

通过 NABCD 方法分析后，我们做出如下原型：

- 1、原型设计工具：Mockplus
- 2、Mackdown 工具：[stack editor](#)

登录界面分为学生和老师。



学生端主页，填报 5 个梯度志愿



学生可查看导师信息

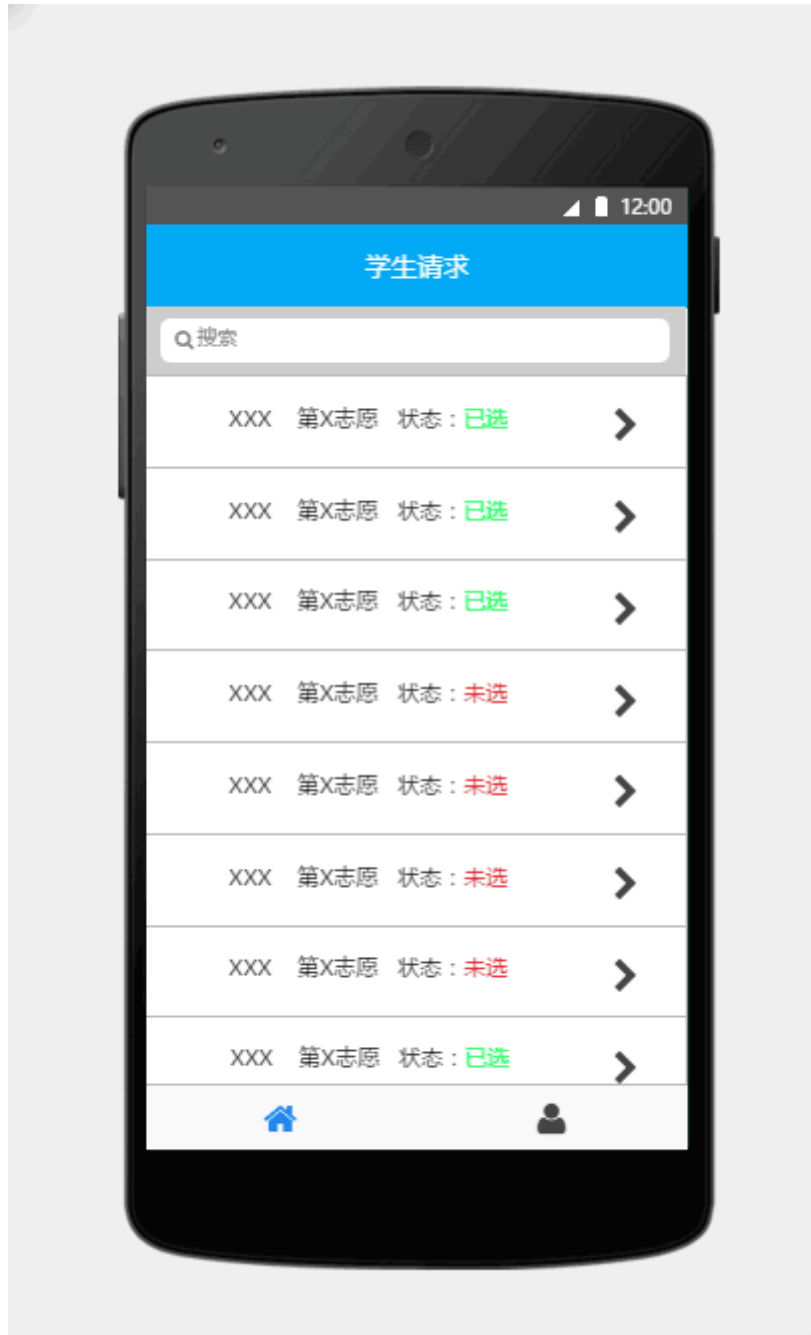




设置界面，消息通知可推送中选，留言等消息



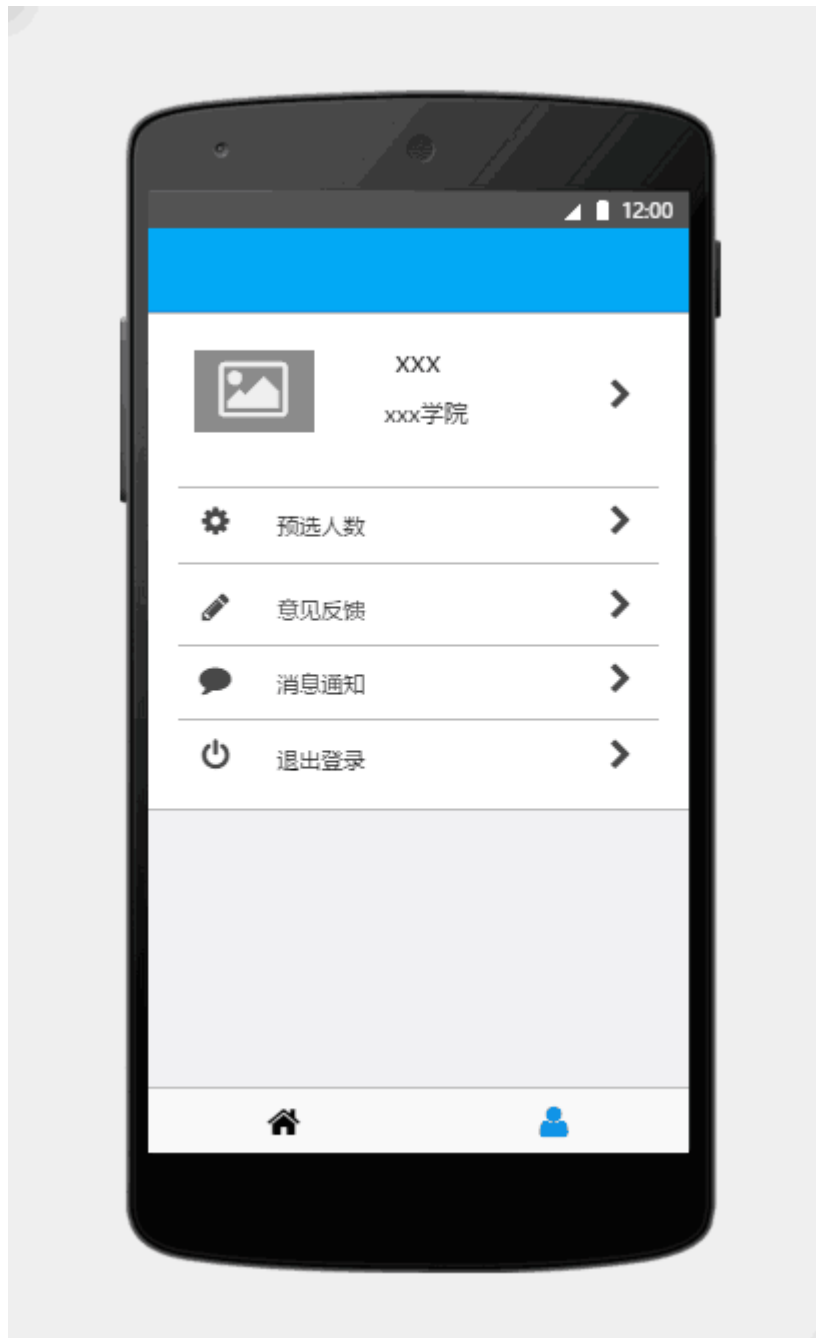
教师端主页，可查看填报自己的学生



教师可查看学生具体信息，可以主动选择学生(但不是最终结果)

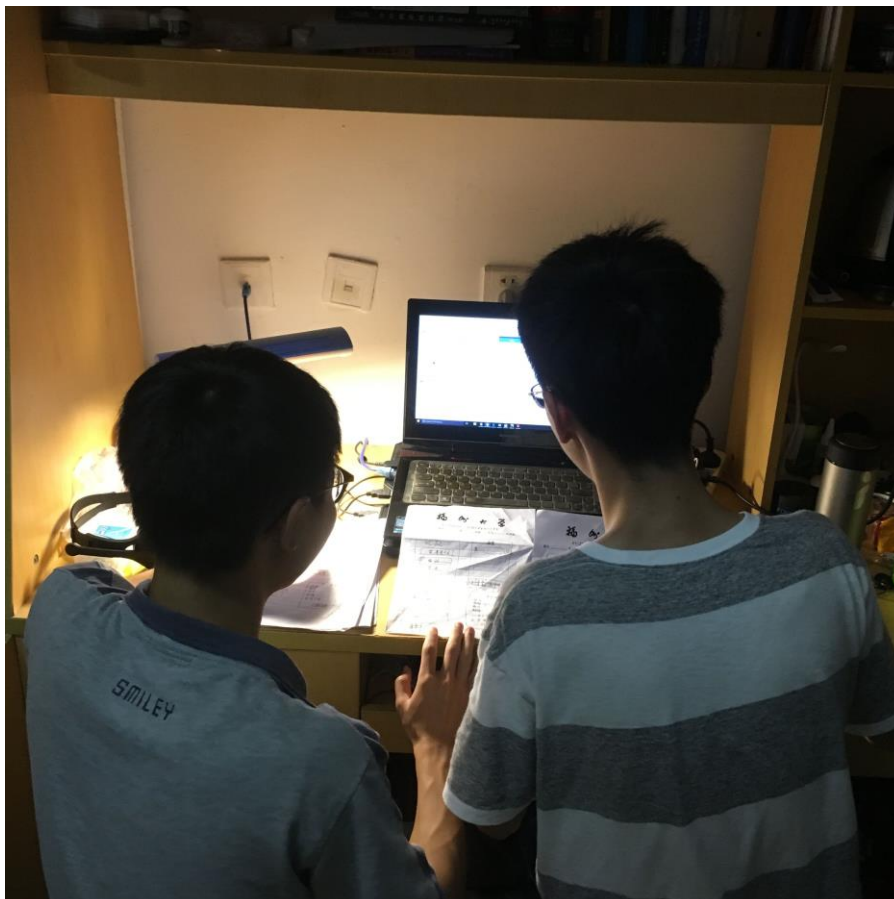
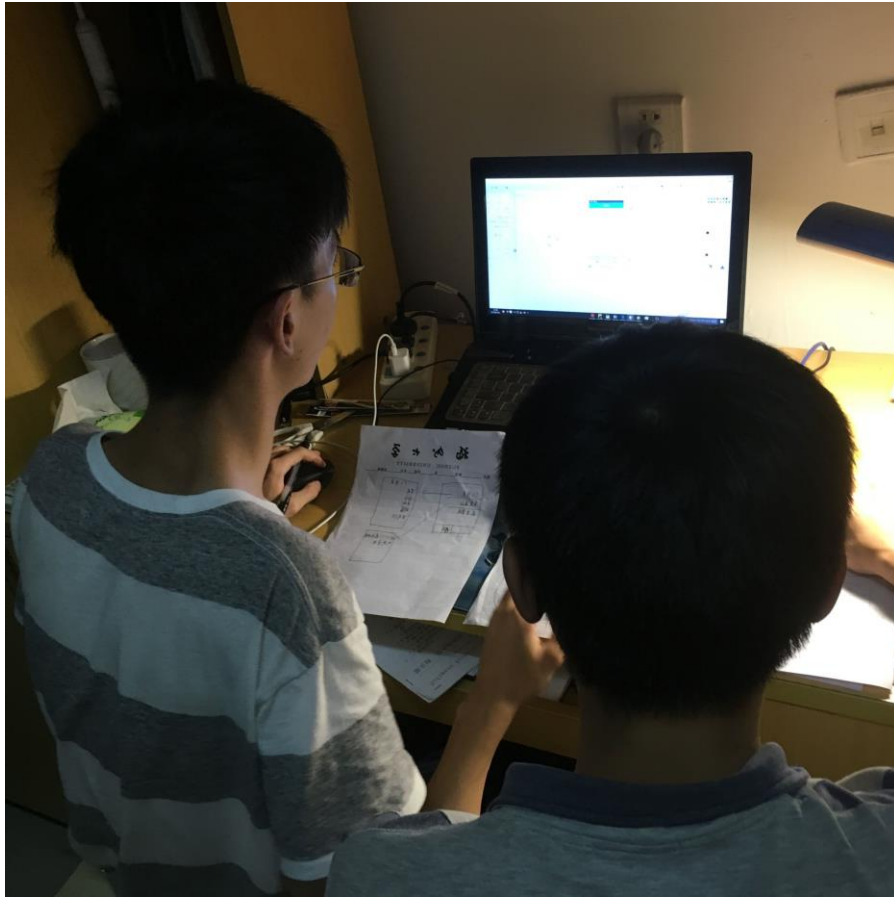


教师可设置自己希望的学生人数区间(默认 0-8)



最后结果由后台分配，具体过程如下

1. 第一次分配，为互相选中的情况。针对多个老师选择同一个学生，按学生梯度志愿优先分配。老师只能选择填报志愿为自己的学生，所以不会产生冲突；同时也因为老师选择学生时有人数限制(自己设定的学生人数)，所以也不会存在老师所收学生超过上限。
2. 第二次分配，剩下学生都为没有老师选择他的情况。先视所有学生的志愿为平行志愿，对于每个人数未滿的老师，将选择他的学生按绩点降序排序，绩点高的优先分配。如果产生一个学生可以分配给多个老师的情况，再按照学生自己的梯度志愿优先分配。
3. 第三次分配，不考虑老师设定的人数上限，尽可能平均的分配剩下的学生给各个老师。



效能分析

需求分析	3h
原型工具的选择和使用	4h
markdown的学习	0.5h
文档	2h

断断续续花费了将近 10 个小时，效率略低。

PSP

PSP	
计划	估计这个任务需要 4 周的时间
开发	
	需求分析：简化信息收集和整理；实现老师学生双向选择
	生成设计文档：博客 pdf
	设计复审：先确定必要功能，再逐步细化讨论各个细节
	代码规范：花括号换行、缩进 4 个空格、变量尽量名词化、条例清楚、写好注释
	具体设计：界面设计，算法设计，数据库设计等
	具体编码：Java
	代码复审：结对编程时，一人编程，另一人复审
	测试：黑白盒
记录用时	课余时间，四周左右
测试报告	边做边测试，根据黑白盒测试写测试报告
计算工作量	两人*4 周
事后总结	边做边总结，将开发过程中所遇问题和解决方法记录下来
过程改进计划	

小结：

通过阅读《构建之法》，明白项目开发过程中，每个人的重点工作不相同，不可随意替换，书中用足球赛形象比喻了这一过程。正是因为这样，软件工程实现困难且常常延期。在开发过程中，需要多沟通，在结对编程中，仅仅两个人就有许多分歧，一定要先解决分

歧再往下，不然常常导致中途返回重新讨论。比如对于后台互选算法，我们当时产生了分歧，导致后来原型设计的时候各个功能不断出现冲突。完成这次作业感触挺多的，发现软件工程不编码的部分要比编码的复杂得多了，今后还要多多努力。