

第 5 组立项报告

(策划)

编 撰 人：全组人员

审 核 人：杨卓

批 准 人：

批准日期：

文档版本：V3

版本历史

日期	版本	说明	作者
2022. 3. 17	V1	第一版立项报告	全体
2022. 3. 24	V2	第二版立项报告	全体
2022. 3. 31	V3	第三版立项报告	全体

目录

一、 项目选题	1
二、 项目背景	1
2.1. 助力教学招生	1
2.2. 提高生源质量	1
三、 项目简介	1
四、 项目思维导图	2
五、 项目主要内容	2
5.1. 活动发布	2
5.1.1 Banner	2
5.1.2 规则设置	3
5.1.3 流程设置	3
5.1.4 指定参与人数设置	3
5.1.5 活动类型设置	3
5.1.6 活动基础信息维护	3
5.1.7 时间设置	3
5.2. 报名审核	4
5.2.1 审核申请	4
5.2.2 筛选查询待审批申请	4
5.3. 活动反馈	4
5.3.1 反馈查看和下载	4
5.3.2 活动内容反馈	4
5.4. 已报名活动	5
5.4.1 已报名活动详情页	5
5.4.2 已报名活动列表	5
5.5. 活动信息	5
5.5.1 活动详情页	5
5.5.2 活动列表展示	6
5.5.3 活动筛选查询	6
5.5.4 活动 banner 轮播图	6
5.6. 用户管理	6
5.6.1 密码找回	6
5.6.2 密码修改	6
5.6.3 用户登录	7
六、 项目所需技术	7
七、 项目分工	7
八、 进度计划	7
九、 工时估计	8
十、 管理需求	8

10.1. 代码规范.....	8
10.1.1 命名	8
10.1.1.1 每一行的字数限制	8
10.1.1.2 命名规范.....	8
10.1.1.2.1 类命名	8
10.1.1.2.2 方法命名	9
10.1.1.2.3 变量命名	9
10.1.1.2.4 图片命名	10
10.1.1.2.5 注释.....	10
10.1.1.2.6 代码优化	10
10.1.1.2.7 系统函数放上面，自定义函数放下面	11
10.1.2 符号	11
10.1.2.1 大括号	11
10.1.2.2 运算符	11
10.1.2.2.1 运算符与变量之间的间隔	11
10.1.2.2.1.1 一元运算符与变量之间没有空格:	11
10.1.2.2.1.2 二元运算符与变量之间必须有空格	11
10.1.2.2.2 多个不同的运算符同时存在时应该使用括号来明确优先级	11
10.1.2.2.3 if 语句	11
10.1.2.2.4 for 语句	12
10.2. 配置管理.....	12
10.2.1 配置管理工具	12
10.2.2 仓库地址	12
10.2.3 环境配置	12
10.3. 缺陷管理.....	13
10.3.1 开发模式	13
10.3.2 管理工具	13
十一、 存在问题及应对措施	13
11.1. 前端	13
11.1.1 Element UI 的引入	13
11.1.2 token 的存储和获取问题	13
11.1.3 网页布局问题	14
11.1.4 Vue	14
11.2. 后端	14
11.2.1 数据库	14

一、项目选题

招生宣传报名系统

二、项目背景

招生管理系统是建设智慧校园的重要支撑部分，主要应用于各大高校的招生管理部门，结合各学校招生的实际情况，招生管理系统拥有自动化的流程、智能的数据分析、科学的管理模式，有效改善了传统招生管理模式产生的管理效率较低，经营管理水平相对滞后等现象。

2.1. 助力教学招生

为适应教育改革的新形势，进一步加大学校招生宣传工作的力度，加强与生源中学的合作，组建稳定高效的招生宣传队伍，构建多渠道、全方位的招生宣传工作格局，提高招生宣传效果，保证招生宣传工作的制度化、系统化、规范化和常态化，不断巩固与提高学校本生源质量已经是教育发展趋势。

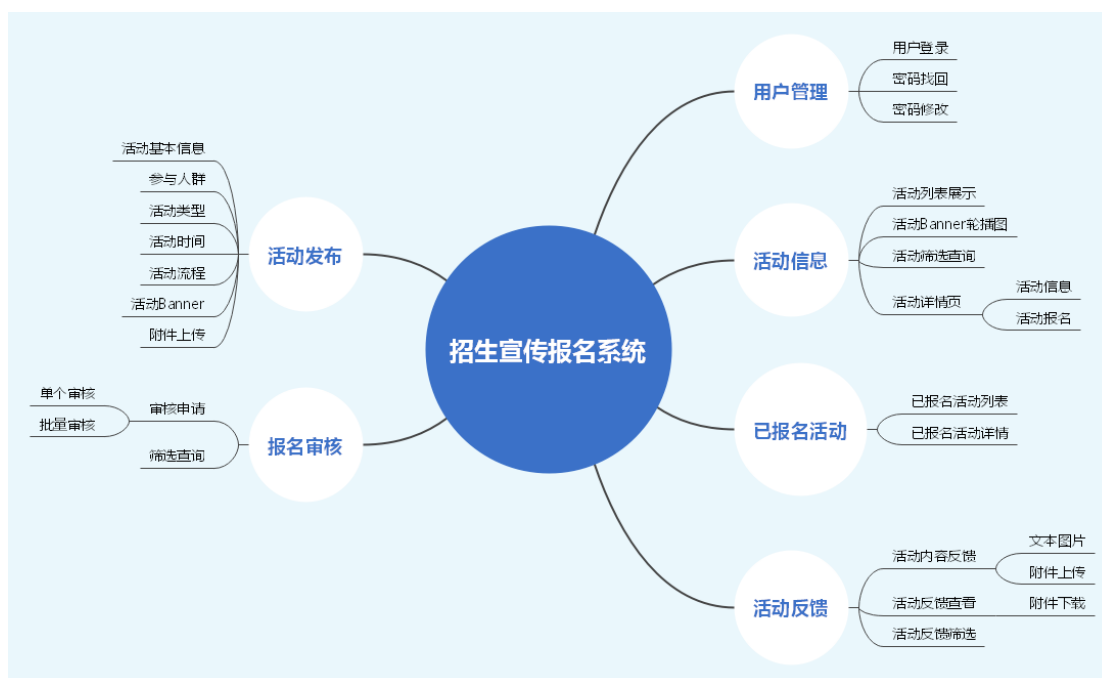
2.2. 提高生源质量

贯彻落实学校教育发展规划纲要，以提高生源质量为核心任务，强化组织领导，挖掘整合资源、拓宽宣传渠道、完善工作机制，按照国家、地方指导的工作方针，着手丰富和创新宣传方式，构建多渠道、多层次、全方位的招生宣传工作格局，助力教育事业更好地发展。

三、项目简介

招生宣传报名系统包含活动发布、报名审核、用户管理、活动信息、已报名活动和活动反馈等六个部分的内容，帮助有效改善了传统招生管理模式产生的管理效率较低，经营管理水平相对滞后等现象。

四、项目思维导图



五、项目主要内容

招生宣传报名系统包含活动发布、报名审核、用户管理、活动信息、已报名活动和活动反馈等六个部分的内容，具体实现包含以下部件及功能：

5.1. 活动发布

学校管理员（招生办）进行活动的创建。主要包含活动主题、时间设置、类型、参与报名者的指定、流程配置、活动介绍及 bannner(轮播图)的设置等

5.1.1 Banner

系统支持在板块中进行轮播图展示, 用户创建时可进行轮播图的维护, 具体包含: 图片上传、URL 配置。Banner 不超过 3 张。当图片无法找到时, 自动删除该 banner。

5.1.2 规则设置

如每一所招生对象指定学生人数和教师人数。指定后，在相关人员进行审批时，系统就会做相对应用的提示

5.1.3 流程设置

活动创建时，创建者自定义审批流程，如审批流程指定为：学院审核→学校审核，当以此设置，发布后，报名者进行报名后，系统自动进行流程审批流转。此处提供选择框。当需要学院审核时，提供学院的选择框供活动发布者选择需要审核的学院。

5.1.4 指定参与人数设置

系统将同步用户中心数据，可按照用户角色、名称、用户组，批量或逐一进行参与人群设置；只有系统指定的参与人群，方可进行报名和参与。

5.1.5 活动类型设置

设定活动类型，包括线下活动、线上活动、校外活动、校内活动等；提供一个下拉选择框，供活动发布者进行选择。

5.1.6 活动基础信息维护

主要包含文字描述，如活动名称、活动简介。支持文字、附件的上传；活动名称的长度不超过 15 字，简介不超过 200 字，图片不超过 3 张，附件大小不超过 5M。

5.1.7 时间设置

创建者可自定义时间设置，如：活动起止时间、报名时间等；
当起始时间晚于结束时间时，提交表单会有错误提示信息。当未设置起始和结束时间时，提交表单也会有错误提示信息。时间的选择包括年-月-日-时-分

5.2. 报名审核

学校用户按照活动的流程设定进行全校师生报名的申请对应节点审批, 其审核节点的审批是根据活动流程设置。进入到审批菜单界面, 进行对申请的查询和审批操作

5.2.1 审核申请

学院和学校招生办对申请进行审批。审批列表可按照指定条件排序（成绩、年级等），审批者可对申请单个审批和批量审批。在审批界面显示剩余名额数，防止通过人数量溢出。添加撤销操作，可撤销上一步审核操作。

5.2.2 筛选查询待审批申请

根据系统提供的筛选查询字段，可进行精准筛选出要进行审核的活动/具体人员的申请。也可以根据学号或者姓名精准查找到待审核用户。

5.3. 活动反馈

工作内容反馈（总结），附件上传和下载

5.3.1 反馈查看和下载

用户可查看相应权限内的反馈条目，并下载附件。

查看反馈可根据时间、发布者身份进行筛选或排序。

学院或者学校管理用户可以删除反馈，不可修改。

5.3.2 活动内容反馈

用户输入活动的反馈内容并提交，字数限制为 200 字以内，若超出则显示提示信息。

用户可添加附件进行上传，附件大小不超过 5M。

用户可以修改和删除自己发布的反馈

5.4. 已报名活动

已报名活动详情查看、已报名活动报名审批进度查看、活动分组查询、活动反馈查看、附件上传和下载

5.4.1 已报名活动详情页

已报名活动详情页展示活动的信息，活动的名字、报名时间、活动时间、活动地点、活动人数、报名按钮、附件下载。

报名按钮显示当前审批状态。

5.4.2 已报名活动列表

已报名活动列表以列表的方式显示已报名活动的信息（活动名、活动图、活动时间、活动人数等）。

活动列表以分页的形式展现，每页可显示 10 条活动信息，底部分页控件显示从页数 1 到总页数的页码、上一页、下一页等。

列表项显示活动的信息之外还会显示一个审批状态。

5.5. 活动信息

Banner 轮播图展示、筛选查询、活动列表、活动详情

5.5.1 活动详情页

活动详情页展示活动的信息，活动的名字、报名时间、活动时间、活动地点、活动人数、报名按钮、附件下载。

若允许的报名时间已过期，报名按钮不可点击并显示“已过期”。若用户无资格报名，报名按钮也不可点击并提示信息。若用户已报名，报名按钮不可点击并显示“已报名”，或显示当前审批状态。

5.5.2 活动列表展示

活动列表以列表的方式显示活动的信息（活动名、活动图、活动时间、活动人数等），用户筛选和搜索的结果也显示在活动列表。

活动列表以分页的形式展现，每页可显示 10 条活动信息，底部分页控件显示从页数 1 到总页数的页码、上一页、下一页等。

5.5.3 活动筛选查询

用户可根据系统提供的筛选查询字段，可进行精准筛选查询活动。用户选择系统提供的关键字段标签来实现感兴趣活动的筛选。用户也可自行输入搜索词搜索活动。如筛选或搜索结果为空，系统应显示提示信息。

5.5.4 活动 banner 轮播图

活动列表页面以 Banner 轮播大图的形式展现部分活动宣传图。

轮播图可包含 5 个活动的宣传图，系统默认选择最新、最热门（报名人数最多）的活动。

轮播图支持循环滚动、定时自动滚动和用户操作滚动。

5.6. 用户管理

用户登录、用户密码修改、密码找回

5.6.1 密码找回

用户忘记密码时可找回密码。在登录界面申请找回密码后，用户的学校邮箱会接受到系统发送的当前密码。

5.6.2 密码修改

用户在账号管理界面中可修改密码。输入当前密码、新密码和确认密码。

若当前密码和确认密码不匹配，会提示输入错误，系统即时显示。而不是等到填写完新密码

后提交表单时才显示错误。

新密码的长度要求 6 位及以上的数字或字母。

5.6.3 用户登录

用户通过填写统一认证的学号或工号登录系统，系统自动识别账号所有者的身份，初始密码都为 123456。若填写的账号不正确，系统显示错误提示信息。

六、项目所需技术

Vue、Spring Boot、MySQL、Element UI

七、项目分工

项目经理：杨卓

成员：翟雨辰 张鲁刚 谢皓宇 阮高立

前端设计及实现：谢皓宇 张鲁刚

后端设计及实现：翟雨辰 阮高立

前后端协作设计及实现，项目代码测试：杨卓

八、进度计划

第 4 周：项目选题，完成立项报告

第 5-6 周：进行需求分析，形成需求规格说明书

第 7 周：开始交互设计

第 8 周：完善交互设计，开始并完成接口定义

第 9-13 周：进行前端页面的编写和后端的搭建以及接口的编写

第 14-15 周：测试并完善项目

第 16 周：项目答辩验收

九、 工时估计

项目选题，完成立项报告：1 个工作周

需求分析，形成需求规格说明书：2 个工作周

交互设计：7 个工作日

接口定义：3 个工作日

前端页面编写：5 个工作周

后端搭建以及接口编写：5 个工作周

项目测试完善：2 个工作周

十、 管理需求

10.1. 代码规范

10.1.1 命名

10.1.1.1 每一行的字数限制

每一行的代码尽量不要超出 80 个字的长度，超出的回车排版，方法名的冒号对齐

10.1.1.2 命名规范

10.1.1.2.1 类命名

首字母大写，每个单词首字母大写（大驼峰命名法），尽量使用能够反映类功能的名词短语。

eg: UserManager , UserData 等。

UIKit 里的 UI 界面部分、控件类型直接使用尾端的驼峰单词。

eg:

UIView -> xxxView

UIViewController -> xxxViewController

UIButton -> xxxButton

UILabel -> xxxLabel

类名过长，取中间的单词作为尾端。

eg:

UIActivityIndicatorView -> xxxActivity

UIPickerView -> xxxPicker

UIProgressView -> xxxPorgress

10.1.1.2.2 方法命名

首字母小写，之后每个单词首字母都大写(小驼峰法命名法)

10.1.1.2.3 变量命名

首字母小写，之后每个单词首字母都大写，具有足够的说明性。

成员变量不需要添加 “_” 前缀。

格式化代码：在 - 、+ 和 返回值 之间留一个空格，方法名和第一个参数之间不留空格

10.1.1.2.4 图片命名

使用英文，全部小写，单词中间“_”隔开

添加 UI 模块名作为前缀，避免冲突

图片存放在 `/static/images` 下

eg: bar_title icon_coin

10.1.1.2.5 注释

以下三种情况需要写注释：

- （1）公共接口（注释要告诉阅读代码的人，当前类能实现什么功能）。
- （2）涉及到比较深层专业知识的代码（注释要体现出实现原理和思想）。
- （3）容易产生歧义的代码（但是严格来说，容易让人产生歧义的代码是不允许存在的）。

10.1.1.2.6 代码优化

避免相同的代码段在多个地方出现相同的代码，必须归纳出来并且用一个类封装起来

语句嵌套层次不得超过 3 层，超出的必须抽取出中间函数

g: for, while 循环 if, do 等

及时删除或注释掉无用的代码

确定不使用的代码应该删除

10.1.1.2.7 系统函数放上面，自定义函数放下面

10.1.2 符号

10.1.2.1 大括号

控制语句(if, for, while, switch)中，大括号开始于行尾
函数中，大括号开始于行首

10.1.2.2 运算符

10.1.2.2.1 运算符与变量之间的间隔

10.1.2.2.1.1 一元运算符与变量之间没有空格：

例如：!bValue、~iValue、++iCount、*strSource、&fSum

10.1.2.2.2.1 二元运算符与变量之间必须有空格

```
fWidth = 5 + 5;  
fLength = fWidth * 2;  
fHeight = fWidth + fLength;  
for(int i = 0; i < 10; i++)
```

10.1.2.2.2 多个不同的运算符同时存在时应该使用括号来明确优先级

在多个不同的运算符同时存在的时候应该合理使用括号，不要盲目依赖操作符优先级。

10.1.2.2.3 if 语句

必须列出所有分支(穷举所有的情况)，而且每个分支都必须给出明确的结果。

不要使用过多的分支，要善于使用 `return` 来提前返回错误的情况

首先判断出各种错误的情况然后提前返回，把最正确的情况放到最后返回。

条件过多，过长的时候应该换行

条件语句的判断应该是变量在左，常量在右

每个分支的实现代码都必须被大括号包围

10.1.2.2.4 for 语句

不可在 `for` 循环内修改循环变量，防止 `for` 循环失去控制。

避免使用 `continue` 和 `break`。

10.2. 配置管理

10.2.1 配置管理工具

使用 Git 进行配置管理：Git 是一个开源的分布式版本控制系统，用于敏捷高效地处理任何或小或大的项目；Git 是 Linus Torvalds 为了帮助管理 Linux 内核开发而开发的一个开放源码的版本控制软件；Git 与常用的版本控制工具 CVS, Subversion 等不同，它采用了分布式版本库的方式，不必服务器端软件支持

10.2.2 仓库地址

Gitee 仓库地址：<https://gitee.com/MiroslavYang/aprsystem>

10.2.3 环境配置

开发、测试环境：Jdk11、MySQL 5.7、Vue2.0

部署环境：阿里云—CentOS7

10.3. 缺陷管理

10.3.1 开发模式

本项目采用敏捷开发模式，以用户的需求进化为核心，采用迭代、循序渐进的方法进行软件开发。在敏捷开发中，软件项目在构建初期被切分成多个子项目，各个子项目的成果都经过测试，具备可视、可集成和可运行使用的特征。换言之，就是把一个大项目分为多个相互联系，但也可独立运行的小项目，并分别完成，在此过程中软件一直处于可使用状态。

10.3.2 管理工具

使用 Leagoo 进行缺陷管理：以看板为核心，通过看板共享和实时同步团队工作来实现高效协同。团队工作体现为卡片，内容可以是需求、任务、问题等，最终完成工作分解结构（简称 WBS）把一个项目，按一定的原则分解，项目分解成任务，任务再分解成一项项工作，再把一项项工作分配到每个人的日常活动中，直到分解不下去为止。即：项目→任务→工作→日常活动。

十一、 存在问题及应对措施

11.1. 前端

11.1.1 Element UI 的引入

问题：引入 Element UI 的样式，但是样式并不符合需求

应对措施：通过自定义修改样式，使得样式符合需求。

11.1.2 token 的存储和获取问题

问题：某些接口的访问需要在 header 中加入 token，但是在进去页面获取 token，然后发送请求的时候，会出现 token 为空的情况

应对措施：查询相关接口规则，解决接口访问问题

11.1.3 网页布局问题

问题：横向排列的几种使用方法常见问题；Element UI 自定义样式的修改问题；
position=fixed 脱离文档流可能导致父元素高度塌陷。

应对措施：通过网络查询排列常见问题和样式自定义等情况的解决方法来解决实际应用中可能出现的问题

11.1.4 Vue

问题：调用组件的操作还不是十分熟练

应对措施：加强组件调用的训练提高实际应用的熟练度

11.2. 后端

11.2.1 数据库

问题：对于 Java 和数据库的操作还不是十分熟练，可能会出现端口占用、数据库连接失败、SQL 语句错误、前端参数收错误、前端请求 JSON 与接口不匹配等错误

应对措施：加强 Java 和数据库的训练提高实际应用的熟练度