



# 软件项目管理 立项报告



题 目 基于 QQ 音乐的大数据分析和推荐系统

学 院 软件学院

专 业 软件工程

组 号 第六组

组 长 邱亢迪 2019141461152

成 员 文灿玥 2020141461096

詹维璐 2020141461092

唐靓 2020141461132

和靖东 2020141461181

毛弘宇 2020141461142

指导教师 杨波

二 0 二 三 年 三 月 二 十 日

## 版本历史

日期	版本	说明	作者
2023. 03. 16	V1	第一版立项报告	全体
2023. 03. 24	v2	第二版立项报告	全体

## 目录

一、 项目选题 .....	4
基于 QQ 音乐的大数据分析和音乐推荐系统 .....	4
二、 项目背景 .....	4
2.1 直观展示数据 .....	4
2.2 科学管理歌曲和歌单 .....	4
2.3 实现个性化歌曲推荐 .....	5
三、 项目简介 .....	5
四、 项目思维导图 .....	5
五、 项目主要内容 .....	5
5.1 用户管理 .....	6
5.2 歌曲大数据信息展示 .....	6
5.3 歌曲推荐 .....	7
六、 项目所需技术 .....	9
6.1 框架: Spring Boot .....	9
6.2 数据库管理: Mysql + Navicat Premium .....	9
6.3 前端: vue + axios + router .....	10
6.4 后端: Mybatis-Plus .....	10
6.5 爬虫: Scrapy, selenium .....	10
6.6 系统基础架构: Hadoop .....	10
七、 开发模式 .....	11
八、 项目分工 .....	12
九、 进度计划 .....	13
十、 工时估计 .....	14
十一、 管理需求 .....	14
11.1 代码规范 .....	14
11.2 配置管理 .....	17
11.3 缺陷管理 .....	17
十二、 存在问题及应对措施 .....	18
12.1 前端 .....	18
12.2 后端 .....	18
12.3 数据端 .....	19

## 一、项目选题

基于 QQ 音乐的大数据分析和音乐推荐系统

## 二、项目背景

音乐是最令人愉悦的艺术，音乐是人类的创造，自然与人有着密不可分的联系。音乐在人类文明的进程中，以其特有的方式在社会发展、个体发展以及教育发展中发挥着十分重要的作用。它以人的生存意识为中心，反映人的本质，展示人的心态，塑造人的品格，更是影响着人的长远发展。因此本项目基于 QQ 音乐，进行大数据分析和音乐推荐，让音乐融入日常生活。

现如今，人们的生活已经离不开音乐，但当前市面上主流的音乐平台各种功能让人眼花缭乱，初次接触的用户不可避免会感到繁琐和迷茫，2022 年，QQ 音乐在各在线音乐软件的使用时长占第四位，QQ 音乐作为在线音乐软件的三大巨头之一，月活跃人数已达 23000 万人。因此本项目基于 QQ 音乐，进行大数据分析和音乐推荐，针对不同的用户群体提供相应的歌曲歌单推荐服务，实现用户科学管理歌曲和歌单。

### 2.1 直观展示数据

当前音乐创作层出不穷，在大量的音乐面前，人们无法方便地获取当前音乐的数据，基于 QQ 音乐的大数据分析和音乐推荐系统（以下简称 QMABG），为直观展示当前歌手、歌曲、歌单、音乐风格、歌曲标签、歌曲评论量、最常一起出现的歌曲标签、评论所处的时段的排名以及评论中不同 IP 地址出现的次数，将所有数据采取可视化界面直观展示出来，让用户更加方便地了解当前乐坛数据，给予用户直观详细的信息展示。

### 2.2 科学管理歌曲和歌单

目前用户在使用音乐软件时可管理的信息过多，而 QMABG 仅提供和歌曲歌单管理，将功能简洁化，真正成为一个易于上手，简洁而不繁琐的音乐平台。

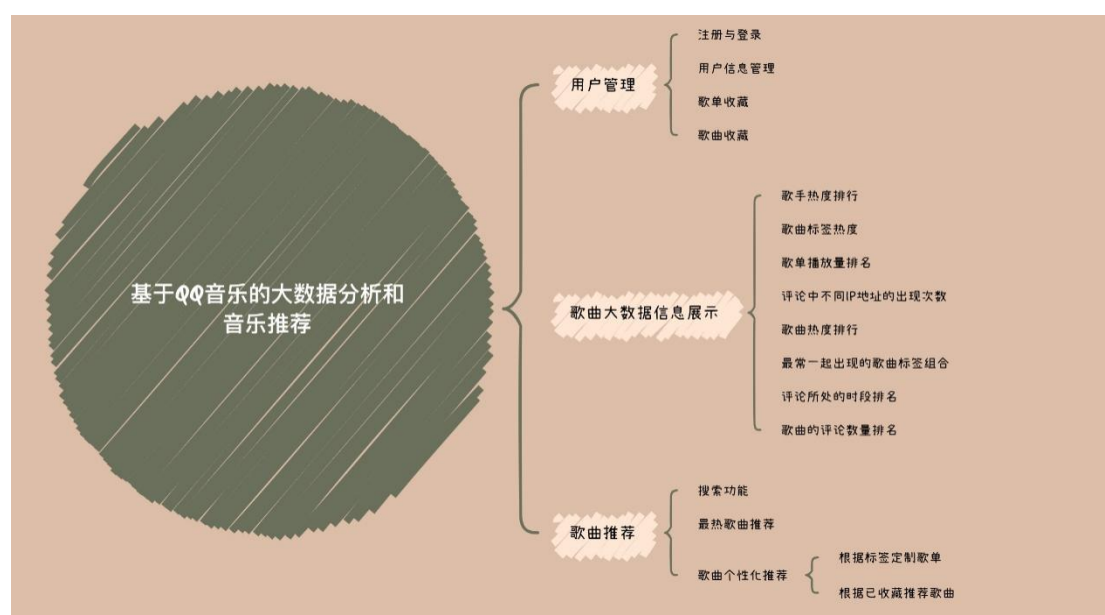
## 2.3 实现个性化歌曲推荐

当前主流的音乐软件因各种因素导致主页所展示的推荐信息无法真正迎合和每个用户，而其又缺少让用户选取自己感兴趣的歌曲类型进行个性化音乐定制的功能，因此 QMABG 将提供给用户个性化歌曲推荐服务，本系统通过两种方式进行个性化歌曲推荐，提升用户的使用体验。

## 三、项目简介

QMABG 从听歌人群、类别、地区、歌手热度等多维度分析对比数据，将最终得到的数据通过可视化展示出来，让数据更加直观可靠，同时，QMABG 将展现用户独特的音乐品味定制用户专属歌单。有效改善了数据过多导致的信息条理性和逻辑性缺乏和功能过多导致初次使用的用户陷入迷茫的现状。

## 四、项目思维导图



## 五、项目主要内容

该项目主要开发工作分为三个模块：用户管理模块、歌曲大数据信息展示模块、歌曲推荐模块，具体如下，也可见表 5-1。

## 5.1 用户管理

注册与登录、用户信息管理、歌单收藏、歌曲收藏。

### 5.1.1 注册与登录

用户通过输入账户名和密码进行账号注册，注册成功后即可解锁该系统所有功能。

用户通过输入账户名和密码，系统进行登录验证，若登录验证成功就可以进入个人页面，若登录验证失败则提示登录失败，需重新登录。

### 5.1.2 用户信息管理

用户登录成功后可以进入个人信息管理界面进行昵称、性别、年龄和地址的修改和重置。

### 5.1.3 歌单收藏

歌单收藏提供给用户收藏歌单、取消收藏歌单、点击跳转链接跳转到相应的QQ 音乐歌单界面、展示歌单相关信息等功能。

### 5.1.4 歌曲收藏

歌曲收藏提供给用户收藏歌曲、取消收藏歌曲、点击跳转链接跳转到QQ 音乐对应的歌曲界面、展示相关歌曲信息等功能

## 5.2 歌曲大数据信息展示

歌手热度排行、歌曲标签热度、歌单播放量排名、评论中不同IP 地址的出现次数、歌曲热度排行、最常一起出现的歌曲标签组合、评论所处的时段排名、歌曲的评论量排名

### 5.2.1 歌手热度排行

歌手热度排行以在30 个榜单top10 中出现次数的排名为根据，进行歌手热度排行。

### 5.2.2 歌曲标签热度

歌曲标签热度排行以指定标签下歌曲播放量之和为根据，进行歌曲标签热度

排行。

### 5.2.3 歌单播放量排名

歌单播放量以各歌单的播放量为根据，进行歌单播放量排名。

### 5.2.4 评论中不同 IP 地址的出现次数

评论中不同 IP 地址的出现次数以省份划分 IP 地址，将 IP 地址出现的次数以 1500 以上、1000-1500、500-1000 进行分类，展示较为活跃的 IP。

### 5.2.5 歌曲热度排行

歌曲热度排行榜以在 30 个榜单 top10 中出现次数的排名为根据，进行歌曲热度排行。

### 5.2.6 最常一起出现的歌曲标签组合

最常一起出现的歌曲标签组合以 Apriori 算法分析指定标签与其他标签之间的置信度，进行关联度排名，展示与指定标签最常一起出现的标签。

### 5.2.7 评论所处的时段排名

评论所处的时段排名以用户发表评论的时间为根据，将时段分为凌晨、上午、中午、下午、傍晚、晚上、深夜七个时段，通过获取不同时段用户发表评论的数量进行评论所处时段的排名。

### 5.2.8 歌曲的评论量排名

歌曲的评论量排名以歌曲的评论量为根据，进行歌曲的评论量排名。

## 5.3 歌曲推荐

搜索功能最热歌曲推荐歌曲个性化推荐根据标签定制歌单根据已收藏推荐歌曲。

### 5.3.1 搜索功能

用户通过对歌曲名称、歌手、专辑名进行搜索，系统会根据用户的输入内容进行相关信息的搜索和展示。

### 5.3.2 最热歌曲推荐

最热歌曲推荐选取当前播放量最高的五首歌进行推荐。

### 5.3.3 歌曲个性化推荐

根据标签定制歌单、根据已收藏推荐歌曲

#### 5.3.3.1 根据标签定制歌单

QMABG 将歌曲标签分为主题、场景、心情、类型四个大类，每个大类又分为数个小类，用户在使用根据标签定制歌单的功能是，可以根据喜好选取各类标签进行个性化歌单定制，需要注意的是，每个大类下仅可选取一个标签，相关歌曲推荐将会展示在“你也许会喜欢”模块。

#### 5.3.3.2 根据已收藏推荐歌曲

QMABG 根据用户收藏的歌单，进行数据分析，获得用户收藏歌单里最常出现的标签和最后收藏的歌曲的标签进行歌曲推荐，相关歌曲推荐放在“根据你的收藏为你推荐”模块。

	功能	描述
通用	登录	用户输入正确的账号和密码进入系统。
	退出登录	用户可以通过点击退出登录按钮来退出系统。
歌曲大数据信息展示模块	大数据信息展示	系统展示 QQ 音乐歌手热度排行、歌曲标签热度、歌单播放量排名、评论中不同 IP 地址的出现次数、歌曲热度排行、最常一起出现的歌曲标签组合、评论所处的时段排名、歌曲的评论量排名。
歌曲推荐模块	热门歌曲推荐	系统默认展示 QQ 音乐上的热门歌曲，用户可以点击歌曲跳转到对应歌曲的 QQ 音乐歌曲详情页面。
	歌曲标签推荐	用户选择多种歌曲标签（最多不超过四种），系统展示根据对应歌曲标签所推荐的歌曲信息，包含歌曲名，歌手名，专辑名，评论数，时长等信息，用户可点击跳转到对应的 QQ 音乐详情页。
	个性化推荐	通过已登录用户的收藏歌曲，系统展示根据对应收藏歌曲所推荐的歌曲信息，包含歌曲名，歌手名，专辑名，评论数，时长等信息，用户可点击跳转到对应的 QQ 音乐详情页。
	歌曲查询	用户通过歌曲名查询对应的歌曲名的歌曲信息。



	歌曲收藏	用户可以点击收藏按钮进行个人歌曲收藏。
用户 管理 模块	个人信息 修改	用户可以修改个人信息，包括姓名、年龄、性别、地址等。
	已收藏歌曲	系统展示用户收藏歌曲信息，包含歌曲名，歌手名，专辑名，评论数，时长等信息，用户可点击跳转到对应的 QQ 音乐详情页。
	取消歌曲 收藏	用户可以点击取消收藏按钮进行个人歌曲取消收藏。

表 5.1

## 六、项目所需技术

Vue, SpringBoot, mybatis plus, MySQL, Element UI, Scrapy, Selenium, Webdriver

### 6.1 框架：Spring Boot

Spring Boot 是由 Pivotal 团队提供的全新框架，其设计目的是用来简化新 Spring 应用的初始搭建以及开发过程。该框架使用了特定的方式来进行配置，从而使开发人员不再需要定义样板化的配置。通过这种方式，Spring Boot 致力于在蓬勃发展的快速应用开发领域(rapid application development)成为领导者。从最根本上来讲，Spring Boot 就是一些库的集合，它能够被任意项目的构建系统所使用。

使用 spring boot 开发项目更加简洁便捷：

- (1) 快速构建独立的 Spring 应用；
- (2) 直接嵌入 Tomcat(默认)、Jetty 和 Undertow，无需外部依赖 Servlet 容；
- (3) 提供依赖启动器简化构建配置；
- (4) 极大程度的自动化配置 Spring 和第三方库； (5) 极少的代码生成和 XML 配置。

### 6.2 数据库管理：Mysql + Navicat Premium

Mysql 是一个关系型数据库管理系统，由瑞典 MySQL AB 公司开发，属于 Oracle 旗下产品。MySQL 是最流行的关系型数据库管理系统之一，在 WEB 应用方面，MySQL 是最好的 RDBMS (Relational Database Management System，关系数据库管理系统) 应用软件之一。

Navicat premium 是一款数据库管理工具，是一个可多重连线资料库的管理工具，它可以让你以单一程式同时连线到 MySQL、SQLite、Oracle、MariaDB、Mssql、及 PostgreSQL 资料库，让管理不同类型的资料库更加的方便。它支持大部分的 MySQL、SQLite、Oracle 及 PostgreSQL 功能，包括预存程序、事件、触发器、函式、检视等。

### 6.3 前端：vue + Element UI + axios + router

Vue 是一套用于构建用户界面的渐进式 JavaScript 框架。与其它大型框架不同的是，Vue 被设计为可以自底向上逐层应用。Vue 的核心库只关注视图层，不仅易于上手，还便于与第三方库或既有项目整合。另一方面，当与现代化的工具链以及各种支持类库结合使用时，Vue 也完全能够为复杂的单页应用（SPA）提供驱动。

Axios 是一个基于 promise 的网络请求库，作用于 node.js 和浏览器中，它是 isomorphic 的(即同一套代码可以运行在浏览器和 node.js 中)。在服务端它使用原生 node.js http 模块，而在客户端（浏览端）则使用 XMLHttpRequest。

### 6.4 后端：Mybatis-Plus

Mybatis 是一款优秀的持久层框架，它支持定制化 SQL、存储过程以及高级映射。MyBatis 避免了几乎所有的 JDBC 代码和手动设置参数以及获取结果集。MyBatis 可以使用简单的 XML 或注解来配置和映射原生信息，将接口和 Java 的 POJOs(Plain Ordinary Java Object, 普通的 Java 对象)映射成数据库中的记录。

MyBatis-Plus (opens new window) (简称 MP) 是一个 MyBatis (opens new window) 的增强工具，在 MyBatis 的基础上只做增强不做改变，为简化开发、提高效率而生。

## 6.5 爬虫：Scrapy, Selenium, Webdriver

Scrapy 是一个适用爬取网站数据、提取结构性数据的应用程序框架，它可以应用在广泛领域：Scrapy 常应用在包括数据挖掘，信息处理或存储历史数据等一系列的程序中。通常我们可以很简单的通过 Scrapy 框架实现一个爬虫，抓取指定网站的内容或图片。

### 使用的库：

Requests：利用 requests 库的 request、get、post 等方法抓取指定 URL 内容，返回期望的结果

JSON：JSON 库提供了解析 JSON 数据的方法，如 Load 方法，利用此库解析并获取界面 JSON 数据

Time：Time 库提供了线程休眠的方法，在爬取动态界面数据过程中，为确保动态数据完全加载成为静态数据，使用 Time.Sleep 方法，确保完全加载后再进行爬取

Selenium & Webdriver：为抓取动态数据，最终选择 Selenium 框架+Webdriver 模拟浏览器，将动态数据转换为静态界面数据后进行抓取。

## 6.6 系统基础架构：Hadoop

Hadoop 是一个由 Apache 基金会所开发的分布式系统基础架构，是一个存储系统+计算框架的软件框架。主要解决海量数据存储与计算的问题，是大数据技术中的基石。Hadoop 以一种可靠、高效、可伸缩的方式进行数据处理，用户可以在不了解分布式底层细节的情况下，开发分布式程序，用户可以轻松地在 Hadoop 上开发和运行处理海量数据的应用程序。

HDFS 有高容错性的特点，并且设计用来部署在低廉的（low-cost）硬件上；而且它提供高吞吐量（High throughput）来访问数据，适合那些有着超大数据集（large data set）的应用程序，它由 n 台运行着 DataNode 的机器组成和 1 台（另外一个 standby）运行 NameNode 进程一起构成。每个 DataNode 管理一部分数据，然后 NameNode 负责管理整个 HDFS 集群的信息（存储元数据）。

Apache Hadoop YARN (Yet Another Resource Negotiator, 另一种资源协调者)是一种新的 Hadoop 资源管理器,它是一个通用资源管理系统和调度平台,可为上层应用提供统一的资源管理和调度,它的引入为集群在利用率、资源统一管理 and 数据共享等方面带来了巨大好处。

## 七、 开发模式

本项目选用瀑布式开发模式,该开发模式重视和强调过程文档,以文档驱动项目,将软件项目开发周期严格划分为几个固定阶段(需求分析、系统设计、软件设计、编码、测试、交付),每个阶段结束都有对应的详细文档作为输出。上一个阶段的输出就是下一个阶段的输入,直至完成整个开发流程。

其优点是每个阶段的目的明确,阶段人员完全专注于该阶段的工作,有助于提高阶段效率,由于存在详细的过程文档,在早期就能明确出项目的范围和概况,能够更有效的组织和调配资源开展项目。

本项目的需求较为明确,并且后期不会频繁变化,适用于瀑布式开发模式。

## 八、 项目分工

- (1) 项目经理: 邱亢迪
- (2) 产品经理: 和靖东
- (3) 运营: 毛弘宇
- (4) 程序员: 文灿玥
- (5) 测试: 詹维璐
- (6) 维护: 唐靓

成员 (六人)	分工	完成情况
邱亢迪 (项目经理)	配置管理 需求分析 (功能性需求) 概要设计 (总体架构设计) 代码编写 (爬虫) 软件测试 (文档)	

	项目部署	
文灿玥	需求分析（数据库设计） 概要设计（数据设计） 详细设计（数据可视化模块） 代码编写（数据可视化模块） 软件测试（文档、可视化模块）	
詹维璐	需求分析（功能性需求） 概要设计（接口设计、个人信息管理模块） 详细设计（界面设计、用户信息管理模块） 代码编写（前后端交互、后端） 软件测试（用户信息管理模块、歌单推荐模块）	
唐靓	需求分析（功能性需求） 概要设计（接口设计、原型设计） 详细设计（歌单推荐模块、界面设计） 代码编写（前端） 软件测试（歌曲推荐模块）	
和靖东	需求分析（项目介绍） 概要设计（数据分析、数据库设计） 详细设计（数据分析模块） 代码编写（大数据分析） 软件测试（数据分析模块）	
毛弘宇	需求分析（功能性需求） 概要设计（用户模块、原型设计） 详细设计（用户模块） 代码编写（用户模块） 软件测试（文档、用户模块）	

## 九、 进度计划

- （1）第 4 周：项目选题，完成立项报告
- （2）第 5~6 周：需求分析，形成需求规格说明书
- （3）第 7 周：完成数据爬取，开始交互设计
- （4）第 8 周：完善交互设计，搭建数据库
- （5）第 9~13 周：完成前端界面编写，后端搭建以及接口编写
- （6）第 14~15 周：测试完善项目
- （7）第 16 周：项目答辩验收

计划	时间
需求分析	5-6 周（32h）
具体设计	7-8 周（36h）
基本界面与功能	9-12 周（64h）
测试与修改	13-15 周（48h）
答辩	16 周
开发人员：六人	总工时：180 小时

## 十、 工时估计

- （1）项目选题，完成立项报告：1 个工作周
- （2）需求分析，形成需求规格说明书：2 个工作周
- （3）交互设计：7 个工作日
- （4）数据爬取：1 个工作周
- （5）搭建数据库：3 个工作日
- （6）前端界面编写：5 个工作周
- （7）后端搭建以及接口编写：5 个工作周
- （8）项目测试完善：2 个工作周

## 十一、 管理需求

### 11.1 代码规范

#### 11.1.1 命名

##### 11.1.1.1 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

##### 11.1.1.2 命名规范

##### 11.1.1.2.1 类命名

（1）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: UserManage ,UserData 等。软件项目管理立项计划 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 、UViewController -> xxxViewController 、UIButton -> xxxButton、UILabel -> xxxLabel

(2) 类名过长，取中间的单词作为尾端。

eg: UIActivityIndicatorView -> xxxActivity 、UIPickerView -> xxxPicker、UIProgressView -> xxxPorgress

#### 11.1.1.2.2 方法命名

(1) 首字母小写，之后每个单词首字母都大写(小驼峰法命名法)

#### 11.1.1.2.3 变量命名

(1) 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。

(2) 成员变量不需要添加“\_”前缀。

(3) 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软件项目管理立项计划

#### 11.1.1.2.4 图片命名

(1) 使用英文，全部小写，单词中间“\_”隔开

(2) 添加 UI 模块名作为前缀，避免冲突

(3) 图片存放在 /static/images 下，eg: bar\_title icon\_coin

#### 11.1.1.2.5 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）。

#### 11.1.1.2.6 代码优化

(1) 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。

eg: for, while 循环 if, do 等

(2) 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软件项目管理立项计划

11.1.1.2.7 系统函数放上面，自定义函数放下面

11.1.2 符号

11.1.2.1 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

11.1.2.2 运算符

11.1.2.2.1 运算符与变量之间的间隔

11.1.2.2.1.1 一元运算符与变量之间没有空格：eg: !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 eg: fWidth = 5 + 5、fLength = fWidth \* 2、fHeight = fWidth + fLength、for(int i = 0; i < 10; i++)

11.1.2.2.2 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。

11.1.2.2.3 if 语句

(1) 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。软件项目管理立项计划

(2) 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

(3) 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。

(4) 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。



(5) 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

(1) 不可在 for 循环内修改循环变量，防止 for 循环失去控制。

(2) 避免使用 continue 和 break。

## 11.2 配置管理

### 11.2.1 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

### 11.2.2 仓库地址

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

### 11.2.3 环境配置

开发、测试环境：Jdk11、MySQL 5.7、Vue2.0

部署环境：阿里云—CentOS7 软件项目管理立项计划

## 11.3 缺陷管理

### 11.3.1 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。

### 11.3.2 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。软件项目管理立项计划。

## 十二、 存在问题及应对措施

### 12.1 前端

#### 12.1.1Element UI 的引入

问题：引入 Element UI 的样式，但是样式不符合需求

应对措施：自定义样式修改

#### 12.1.2 网页布局问题

问题：Element UI 的自定义样式修改问题，父元素高度塌陷

应对措施：用过网络查询排列常见问题和样式自定义等情况的解决办法解决实际应用中出现的问题。

#### 12.1.3Vue

问题：Vue 组件调用易出错

应对措施：通过 Vue 官方文档以及 csdn 等了解组件使用规范，熟悉组件使用

### 12.2 后端

#### 12.2.1Vue

问题：对于 Java 以及数据库的操作还不是很熟练，可能出现端口链接错误，数据库连接失败，SQL 查询语句错误，前端调用错误，请求 JSON 数据失败等问题

应对措施：通过官方文档和 CSDN 等训练数据库能力，提高熟练度

## 12.3 数据端

### 12.3.1 动态数据抓取

问题：对于网站所使用的动态数据，缺少一定的手段进行数据抓取

应对措施：通过 Selenium 框架+Webdriver 将动态数据转换为静态数据，再通过静态数据进行抓取

### 12.3.2 JSON 数据抓取

问题：JSON 类型数据获取困难

应对措施：通过更改请求头，书写 Cookie，利用 JSON 库将 JSON 数据转换为 Python 数据