   

项目开发总结报告

第08小组

200327040 敬祎丹

200327065 林心代

200327086 吴凌欣

200327114 游璐颖

二〇二一年x月x日

目录

[1 项目地址与开发情况 1](#_Toc11354)

[1.1 相关链接 1](#_Toc21602)

[1.2 进度与任务分配 1](#_Toc19110)

[2 开发技术说明 1](#_Toc29363)

[2.1 app端 1](#_Toc6192)

[2.2 web端 2](#_Toc21568)

[2.3 服务器端 3](#_Toc18983)

[3 项目目录结构 5](#_Toc6940)

[3.1 app端目录结构 5](#_Toc15727)

[3.2 web端目录结构 5](#_Toc20301)

[3.2服务器端目录结构 5](#_Toc17888)

[4 其他内容待定 5](#_Toc22924)

# 1 项目地址与开发情况

## 1.1 相关链接

项目Github地址：<https://github.com/Sonatau/SailorMoon>

## 1.2 进度与任务分配

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **周次** | **产品经理：敬祎丹** | **移动端：吴凌欣** | **web端：游璐颖** | **服务器端：林心代** |
| **一** | 1. <移动端需求分析报告>与<后台管理系统需求分析报告>引言部分撰写 2. <项目开发总结报告>开发技术说明部分撰写 | 1. 各个文档的汇总与整理 2. <移动端需求分析报告>项目概述部分撰写 3. <移动端需求分析报告>与<后台管理系统需求分析报告>结构确定 | 1. 对本次项目进行技术选型 2. 补充学习本次项目web端开发需要用到的相关知识 | 1. 对本次项目进行技术选型 2. 补充学习本次项目服务器端开发需要用到的相关知识 |
| **二** |  |  |  |  |
| **三** |  |  |  |  |

# 2 开发技术说明

## 2.1 app端

**（一）技术选型**

混合式开发（Ionic+angular+cordova）

**（二）技术介绍：**

**（1）Cordova**

Cordova 是Apache的开源框架，是由PhoneGap演化而来的一套核心代码。可以理解为Webkit和Google Chrome的关系。cordova 提供了一系列移动平台的壳，例如 iOS、Android、ubuntu phone os、Blackberry、Windows Phone、Palm WebOS、Bada和Symbian。主要用于嵌套H5，将H5打成应用市场认可的包，并且提供一套基于js的H5和各个移动平台通讯的API。比如获取手机文件、相机、手机信息、拨打电话等等基础功能。Cordova 把这些和手机系统互相通讯的功能，封装成一个个插件，需要用到某一个功能，添加即可调用。如果Cordova提供的插件不能满足你和手机的通讯， Cordova也支持自定义插件，可以自己开发一个插件。

**（2）Ionic**

ionic 是一个强大的 HTML5 应用程序开发框架(HTML5 Hybrid Mobile App Framework )。 可以帮助您使用 Web 技术，比如 HTML、CSS 和 Javascript 构建接近原生体验的移动应用程序。ionic 主要关注外观和体验，以及和你的应用程序的 UI 交互，特别适合用于基于 Hybird 模式的 HTML5 移动应用程序开发。ionic是一个轻量的手机UI库，具有速度快，界面现代化、美观等特点。为了解决其他一些UI库在手机上运行缓慢的问题，它直接放弃了IOS6和Android4.1以下的版本支持，来获取更好的使用体验。

**（3）Angular**

Angular是一款优秀的前端JS框架，已经被用于Google的多款产品当中。AngularJS有着诸多特性，最为核心的是：MVC（Model–view–controller）、模块化、自动化双向数据绑定、语义化标签、依赖注入等等。AngularJS 是一个 [JavaScript](https://baike.baidu.com/item/JavaScript" \t "https://baike.baidu.com/item/AngularJS/_blank)框架。它是一个以 JavaScript 编写的库。它可通过 <script> 标签添加到[HTML](https://baike.baidu.com/item/HTML" \t "https://baike.baidu.com/item/AngularJS/_blank) 页面；通过 指令 扩展了 HTML，且通过 表达式 绑定数据到 HTML。AngularJS 是以一个 JavaScript 文件形式发布的，可通过 script 标签添加到网页中。主要用于，界面的数据绑定，和界面逻辑的处理。

## 2.2 web端

**（一）技术选型**

UI框架：Element

前端框架：Vue

**（二）技术介绍**

1. **Element UI**

Element UI 是一套采用 Vue 2.0 作为基础框架实现的组件库,一套为开发者、设计师和产品经理准备的基于 Vue 2.0 的组件库,提供了配套设计资源,帮助网站快速成型。

1. **Vue**

Vue.js是一套构建用户界面的渐进式框架。与其他重量级框架不同的是，Vue 采用自底向上增量开发的设计。Vue 的核心库只关注视图层，并且非常容易学习，非常容易与其它库或已有项目整合。另一方面，Vue 完全有能力驱动采用单文件组件和Vue生态系统支持的库开发的复杂单页应用。Vue.js 的目标是通过尽可能简单的 API 实现响应的数据绑定和组合的视图组件 。Vue.js 自身不是一个全能框架——它只聚焦于视图层。因此它非常容易学习，非常容易与其它库或已有项目整合。另一方面，在与相关工具和支持库一起使用时 ，Vue.js 也能完美地驱动复杂的单页应用。

## 2.3 服务器端

**（一）技术选型**

服务器端开发技术：JavaEE

框架：Springboot

数据库开发：Mybatis

**（二）技术介绍**

1. **javaEE**

Java2平台企业版（Java 2 Paltform Enterprise Edition）,是一种利用Java2平台来简化企业解决方案的开发，部署和管理相关的复杂问题的体系结构。该体系结构提供中间层集成框架用来满足无需太多费用而又需要高可用性、高可靠性以及可扩展性的应用的需求，通过提供统一的开发平台，为搭建具有可伸缩性、灵活性、易维护性的信息系统提供了良好的机制。主要具有高效的开发、支持异构环境、可伸缩性、稳定的可用性等特点。JavaEE允许公司把一些通用的、很繁琐的服务端任务交给中间件供应商去完成，这样开发工作就可以集中在如何创建业务逻辑上，相应地缩短了开发时间。而且基于JaveEE的应用程序不依赖任何特定操作系统、中间件、硬件。因此设计合理的基于JaveEE的程序只需开发一次就可以部署到各种平台。

1. **Sprintboot**

Spring框架是Java平台上的一种开源应用框架，提供具有控制反转特性的容器。Spring框架为开发提供了一系列的解决方案，比如利用控制反转的核心特性，并通过依赖注入实现控制反转来实现管理对象生命周期容器化，利用面向切面编程进行声明式的事务管理，整合多种持久化技术管理数据访问，提供大量优秀的Web框架方便开发等等。Spring框架具有控制反转（IOC）特性，IOC旨在方便项目维护和测试，它提供了一种通过Java的反射机制对Java对象进行统一的配置和管理的方法。Spring框架利用容器管理对象的生命周期，容器可以通过扫描XML文件或类上特定Java注解来配置对象，开发者可以通过依赖查找或依赖注入来获得对象。Spring框架具有[面向切面编程](https://baike.baidu.com/item/%E9%9D%A2%E5%90%91%E5%88%87%E9%9D%A2%E7%BC%96%E7%A8%8B/6016335" \t "https://baike.baidu.com/item/Spring%20Boot/_blank)（AOP）框架，SpringAOP框架基于代理模式，同时运行时可配置；AOP框架主要针对模块之间的交叉关注点进行模块化。Spring的事务管理框架为Java平台带来了一种抽象机制，使本地和全局事务以及嵌套事务能够与保存点一起工作，并且几乎可以在Java平台的任何环境中工作。Spring集成多种事务模板，系统可以通过事务模板、XML或Java注解进行事务配置，并且事务框架集成了消息传递和缓存等功能。Spring的数据访问框架解决了开发人员在应用程序中使用数据库时遇到的常见困难。它不仅对Java:JDBC、iBATS/MyBATIs、Hibernate、Java数据对象（JDO）、ApacheOJB和ApacheCayne等所有流行的数据访问框架中提供支持，同时还可以与Spring的事务管理一起使用，为数据访问提供了灵活的抽象。

1. **Mybatis**

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 是支持普通 SQL查询，[存储过程](https://baike.baidu.com/item/%E5%AD%98%E5%82%A8%E8%BF%87%E7%A8%8B" \t "https://baike.baidu.com/item/MyBatis/_blank)和高级映射的优秀[持久层](https://baike.baidu.com/item/%E6%8C%81%E4%B9%85%E5%B1%82" \t "https://baike.baidu.com/item/MyBatis/_blank)框架。MyBatis 消除了几乎所有的[JDBC](https://baike.baidu.com/item/JDBC" \t "https://baike.baidu.com/item/MyBatis/_blank)代码和参数的手工设置以及[结果集](https://baike.baidu.com/item/%E7%BB%93%E6%9E%9C%E9%9B%86" \t "https://baike.baidu.com/item/MyBatis/_blank)的检索。MyBatis 使用简单的 XML或注解用于配置和原始映射，将接口和 Java 的POJOs（Plain Ordinary Java Objects，普通的 Java对象）映射成数据库中的记录。每个MyBatis应用程序主要都是使用Sql[Session](https://baike.baidu.com/item/Session" \t "https://baike.baidu.com/item/MyBatis/_blank)Factory实例的，一个SqlSessionFactory实例可以通过SqlSessionFactoryBuilder获得。SqlSessionFactoryBuilder可以从一个xml配置文件或者一个预定义的配置类的实例获得。用xml文件构建SqlSessionFactory实例是非常简单的事情。推荐在这个配置中使用类路径资源（classpath resource)，但你可以使用任何Reader实例，包括用文件路径或file://开头的url创建的实例。MyBatis有一个实用类----Resources，它有很多方法，可以方便地从类路径及其它位置加载资源。

MyBatis应用程序根据XML配置文件创建SqlSessionFactory，SqlSessionFactory在根据配置，配置来源于两个地方，一处是配置文件，一处是Java代码的注解，获取一个SqlSession。SqlSession包含了执行sql所需要的所有方法，可以通过SqlSession实例直接运行映射的sql语句，完成对数据的增删改查和事务提交等，用完之后关闭SqlSession。

MyBatis具有简单易学、灵活的特点，方便同学们学习。并且提供映射标签，支持对象与数据库的orm字段关系映射；提供对象关系映射标签，支持对象关系组件维护；提供xml标签，支持编写动态sql。mybatis的优点同样是mybatis的缺点，正因为mybatis使用简单，数据的可靠性、完整性的瓶颈便更多依赖于程序员对sql的使用水平上了。sql写在xml里，虽然方便了修改、优化和统一浏览，但可读性很低，调试也非常困难，也非常受限。mybatis没有hibernate那么强大，但是mybatis最大的优点就是简单小巧易于上手，方便浏览修改sql语句。

对象关系映射是一种[程序设计](https://baike.baidu.com/item/%E7%A8%8B%E5%BA%8F%E8%AE%BE%E8%AE%A1" \t "https://baike.baidu.com/item/%E5%AF%B9%E8%B1%A1%E5%85%B3%E7%B3%BB%E6%98%A0%E5%B0%84/_blank)技术，简称ORM,用于实现[面向对象](https://baike.baidu.com/item/%E9%9D%A2%E5%90%91%E5%AF%B9%E8%B1%A1" \t "https://baike.baidu.com/item/%E5%AF%B9%E8%B1%A1%E5%85%B3%E7%B3%BB%E6%98%A0%E5%B0%84/_blank)编程语言里不同[类型系统](https://baike.baidu.com/item/%E7%B1%BB%E5%9E%8B%E7%B3%BB%E7%BB%9F" \t "https://baike.baidu.com/item/%E5%AF%B9%E8%B1%A1%E5%85%B3%E7%B3%BB%E6%98%A0%E5%B0%84/_blank)的数据之间的转换。从效果上说，它其实是创建了一个可在编程语言里使用的“虚拟对象数据库”。如今已有很多免费和付费的ORM产品，而有些程序员更倾向于创建自己的ORM工具。面向对象是从[软件工程](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6%E5%B7%A5%E7%A8%8B" \t "https://baike.baidu.com/item/%E5%AF%B9%E8%B1%A1%E5%85%B3%E7%B3%BB%E6%98%A0%E5%B0%84/_blank)基本原则（如耦合、聚合、封装）的基础上发展起来的，而关系数据库则是从[数学](https://baike.baidu.com/item/%E6%95%B0%E5%AD%A6" \t "https://baike.baidu.com/item/%E5%AF%B9%E8%B1%A1%E5%85%B3%E7%B3%BB%E6%98%A0%E5%B0%84/_blank)理论发展而来的，两套理论存在显著的区别。为了解决这个不匹配的现象，对象关系映射技术应运而生。

# 3 项目目录结构

## 3.1 app端目录结构

……

## 3.2 web端目录结构

……

## 3.2服务器端目录结构

……

# 4 其他内容待定

……