《软件工程技术与设计》2018-2019 秋季学期

总结报告

留守儿童

援助系统

日期：2019-1-10

**一 完成情况概述**

1. **项目目标**
   1. **信息显示**

将各个城市，地区的留守儿童信息和发布需求展示出来，提供地理位置坐标，经度纬度等。

，以供网站管理员和志愿者查看。

* 1. **用户登录注册**

留守儿童、志愿者和网站管理员可以登录和注册账户。其中留守儿童和志愿者的账户可以自主注册和登录，而网站管理员有编程人员指定。

* 1. **后台审核**

网站管理员登录系统后台管理页面，对留守儿童发布的需求和志愿者申请的解决需求进行审核。管理员审核留守儿童需求是否合理，审核志愿者是否符合解决需求的条件。

* 1. **地图显示**

将留守儿童所有需求信息展示在地图上，。

* 1. **信息提交**

留守儿童都可以登录以后增加新的需求信息或删除已有的需求信息，志愿者提出申请需求，在审核后在页面上显示。

1. **主要设计技术**

**2.1 VUE.js 2.0**

是一套用于构建用户界面的渐进式框架。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](https://cn.vuejs.org/v2/guide/single-file-components.html)以及各种[支持类库](https://github.com/vuejs/awesome-vue" \l "libraries--plugins" \t "_blank)结合使用时，Vue也完全能够为复杂的单页应用提供驱动。

**2.2 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。

**2.3 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。

**2.4 Spring MVC**

一种基于Java的实现了Web MVC设计模式的请求驱动类型的轻量级Web框架，即使用了MVC架构模式的思想，将web层进行职责解耦，基于请求驱动指的就是使用请求-响应模型，框架的目的就是帮助我们简化开发

**2.5 MyBatis**

MyBatis 是支持定制化 SQL、存储过程以及高级映射的优秀的持久层框架。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。

**3 开发管理过程**

使用Github 进行项目版本迭代管理，

前端项目共进行了10次commits。

1. 初始化项目，完成router初步搭建；
2. 完成需求的更改和审核；
3. 完成后天页面的审核删除机制；
4. 格式化文件；
5. 添加所有主页面；
6. 完成用户需求申请；
7. 完善页面；
8. 添加用户需求帮扶申请；
9. 修复需求申请无效问题；
10. 添加需求筛选；

后端项目一共进行了31次commits，涵盖了从项目搭建到完成的所有过程

4． 项目源码

前端项目： https://github.com/Gahon1995/left-behind-child-web

后端项目： https://github.com/Gahon1995/left-behind-child

**二 系统设计与实现**

1 系统的结构设计

前端

后端

MySql数据库

Ajax

RESTful

Mybatis

前端选型为Vue，主要是由于它是基于组件的，依赖标准ES5或ES6，底层技术为单个文件，包括模板，脚本和样式。在数据绑定上支持单双向绑定，支持原生开发，服务端预渲染nuxt.js，最重要的是它的学习曲线比较简单。

后端选型为Spring ,Spring Boot 和MyBatis。使用Spring Boot 进行项目的快速搭建，采用MVC 架构。

Spring Boot 可以用来简化Spring 应用的搭建和开发过程。数据库采用的是MySql，同时使用MyBatis来完成项目与数据库的数据通信。

2系统的主要功能接口设计

下面主要展示几个比较常用的接口说明，详细的使用方法可以查看swagger文档，以及项目源码的controller目录下的源文件。

公共api：

1. “/points”

该接口用于获取每个服务点的左边以及该服务点的唯一标识，主要是用于将各个服务点的位置展示在地图上

1. “/points/{pid}”

该接口主要用于请求某一个服务点的详细信息，其中{pid}为所查询的服务点的唯一编码，返回的数据均通过脱敏处理，例如会将电话号码进行加密处理

1. “/demands”

该接口主要用于获取当前所有的需求信息，一共有4个请求参数，包括通过城市进行查询，以及使用分页的方式进行查询，返回的结果同样会进行相关的处理，进行隐私保护

1. “/login”

登录接口，采用POST协议进行数据传输，主要发送用户名和登录密码，后端校验通过以后会返回一个Token数据，登录后的操作均需要携带该Token进行登录验证

….

登录后的部分api

1. “/admin/demands”

该api主要是用于管理员进行所有需求的获取，必须要管理员登录以后才能够获取到相应的数据，返回的数据里边的内容主要包含了需求的服务点信息、帮扶信息以及审核状态等等

1. “/user/points”

该接口用于获取用户的服务点信息，使用该接口用户能够获取到自己所申请的服务点列表以及审核状态

…

3 系统的设计模式应用

本项目采用Spring Boot 框架，实现前后端分离，其主要开发思想是使用MVC模式。

MVC 模式代表 Model-View-Controller（模型-视图-控制器） 模式，该模式主要用于应用程序的分层开发， Model（模型）表示模型代表一个存取数据的对象。它也可以带有逻辑，在数据变化时更新控制器。View（视图）表示视图代表模型包含的数据的可视化。Controller（控制器）表示控制器作用于模型和视图上。它控制数据流向模型对象，并在数据变化时更新视图，它使视图与模型分离开。

2.6系统的用户界面设计

本次项目分为7个页面，截图分别是图2-1到图2-7。

1. 如图2-1所示，主页为需求列表，这里使用卡片式显示留守儿童的所在小区，姓名，联系电话，地址，需求详情，发布时间，以及在右上角显示帮扶状态：已有帮扶或者申请帮扶。
2. 如图2-2所示，登录页面，输入框为用户名和密码，登录和注册按钮。注册页面如图2-3所示，需要填写信息为姓名，性别，电话，邮箱，密码和确认密码，在页面设置检测密码和确认密码是否相同。
3. 如图2-4所示，Dashboard页面显示的是用户登录后看到的页面，显示用户个人信息和统计总的服务点数量和需求数量。
4. 如图2-5所示，我的服务点页面显示的是志愿者正在服务或者服务过的服务点信息。
5. 如图2-6所示，我的需求页面显示的是留守儿童发布的需求信息。
6. 如图2-7所示，申请服务点页面需要填写服务的名称，城市选择，详细地址和服务点描述。服务点表示帮助留守儿童的固定站点。
7. 如图2-8所示，后端API接口页面主要显示host地址，接口统计信息等。总共15个GET，7个POST，4个PUT，5个DELETE请求。



图2-1 主页：需求列表

图2-2 登录页面 图2-3 注册页面

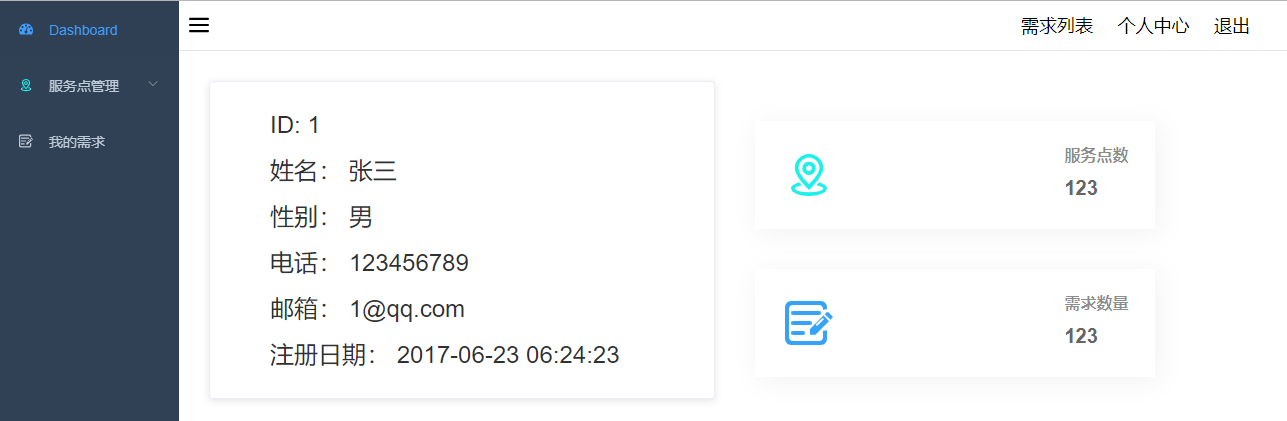


图2-4 dashboard



图2-5 我的服务点



图2-6 我的需求

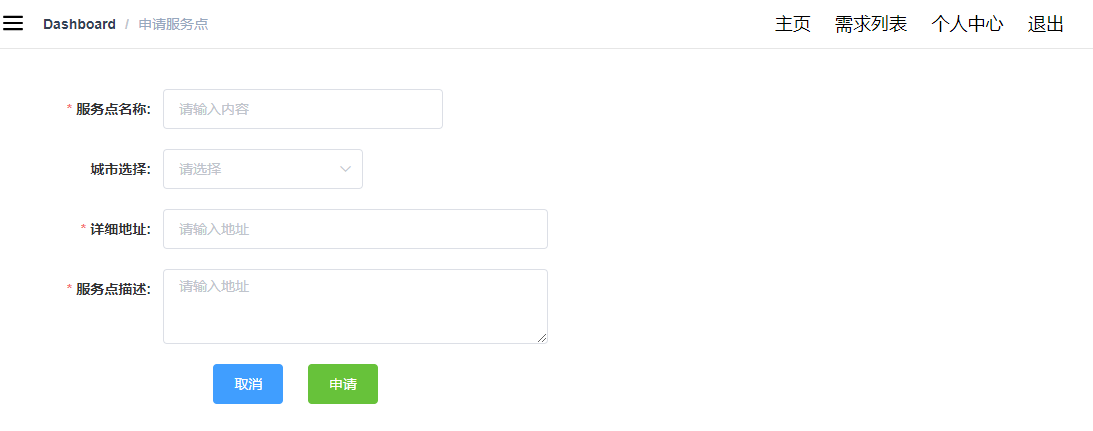


图2-7 申请服务点页面



图2-8 后端API接口页面

2.7系统的部署与运行环境

前端：

前端项目使用的是Vue进行开发的，在编写代码的过程中需要时使用npm来进行编译， 其主要就是静态页面，因此其部署有两种方式， 一是将该项目进行编译，生成静态页面，然后再部署到装有类似nginx等环境的服务器上即可；另外一种方式则是直接使用npm来进行部署

后端：

后端项目使用的开发语言是Java，同时使用Maven来进行依赖控制，其部署方式则是直接将整个项目打包为jar文件，然后在装有java环境的服务器上部署即可。

三 主要技术亮点

本次项目中，实现后端API接口页面中，使用了Swagger框架。

Swagger 是一个规范和完整的框架，用于生成、描述、调用和可视化 RESTful 风格的 Web 服务。总体目标是使客户端和文件系统作为服务器以同样的速度来更新。文件的方法，参数和模型紧密集成到服务器端的代码，允许API来始终保持同步。

swagger-bootstrap-ui是基于swagger接口api实现的一套UI,因swagger原生ui是上下结构的，在浏览接口时不是很清晰,所以，swagger-bootstrap-ui是基于左右菜单风格的方式,适用与我们在开发后台系统左右结构这种风格类似,方便与接口浏览。

在实现的页面中，详细介绍了后端API的定义和获取方法，便于调试，也可以在未来作为一个开源项目，开放接口让外界使用了解和调用接口。

同时该项目对用户的登录验证并没有采用传统的Cookie方式进行验证，而是使用Token的方式进行验证，同时每一个Token中都包含着用户的唯一编码，从而使得后端在进行登录验证的过程中能够获取到当前的登录用户信息。

后端项目的搭建采用了**spring-boot-api-project-seed** ( <https://github.com/Gahon1995/spring-boot-api-project-seed>) 种子项目, 该项目为本人在别人的项目的基础上修改而来的，使用该种子项目能够快速的构建出项目的结构，从而避免在构建项目的过程中会遇到的各种问题

在前端的编写中，使用了开源的后端管理框架，这使得在开发过程中不用去设计每一个页面的具体布局，而着重于去编写具体的功能模块，然后该框架会自动将各个模块组织起来，形成一个完整的页面

四 局限性分析

1.现有的系统没有实现将留守儿童地理坐标和信息显示在地图上，预计实现的功能是地图展示，将所有需求信息展示在地图上，给予浏览者一个直观的展示。

2.在系统中，异常检测做的少，安全性做的不是特别好。例如在注册过程中，无检查密码长度，登录过程中设置了要求密码长度至少为3，在页面端实现异常检查，无法预防在直接构造POST 请求情况下，实现登录。

五 开发过程总结

由于之前没有啥项目基础和编程经验，所以项目中对快速框架和编程技术能力要求比较高，一开始并没有十分适应，到后来实现案例过程中，完善自己的能力过程中渐渐适应。

这次项目经历是一次难得的，将学习理论应用于实践的过程。通过这次实验，也让自己对整个项目的开发过程有了一个比较好的理解，同时也让自己在编程方面的能力得到了进一步的提升。