

华东师范大学软件学院上机实践报告

课程名称: J2EE	年级: 2019 级	上机实践成绩:
指导教师: 毛宏燕	姓名: 王嵛民 苏建锐 林颂家	小组名称: 四面楚鸽 J @ E E
项目实践名称: 项目管 理系统	学号: 10195101563 10195101553 10195101566	上机实践日期: 2020.12

一、目的

综合运用所学 J2EE 规范知识, 包括 JSP, JavaScript、jQuery, Servlet, JavaBean, Database, EJB, MVC 等, 设计并开发一个 Web 应用项目。

二、内容与设计思想

内容: 设计一个项目管理平台, 通过该平台可以进行项目申请, 在线提交报告, 进行项目评审, 查看评审项目的状态, 功能提示如下。

角色与基本功能:

- 学生成员: 注册、登录、个人信息管理、我的项目(组队信息)、提交项目, 修改项目信息、查看项目状态信息。
- 管理员: 项目类别管理, 项目管理, 项目成绩管理; 项目评审, 专家信息管理, 管理中包括基本信息的增、删、改、查。
- 评审导师: 登录, 可查看项目, 下载附件, 对项目评价审核, 打分。
- 项目指导导师: 登录, 确认学生选题, 查看负责项目信息, 对项目提出建议。

流程管理：

- 组队，立项申请，指导导师确认，项目审核与立项确认；
- 中期答辩，项目审核（提交相关材料，每一阶段打分）；
- 终期结题，提交终期成果，项目评审和确定成绩（中期和终期可以执行项目中止/终止）。

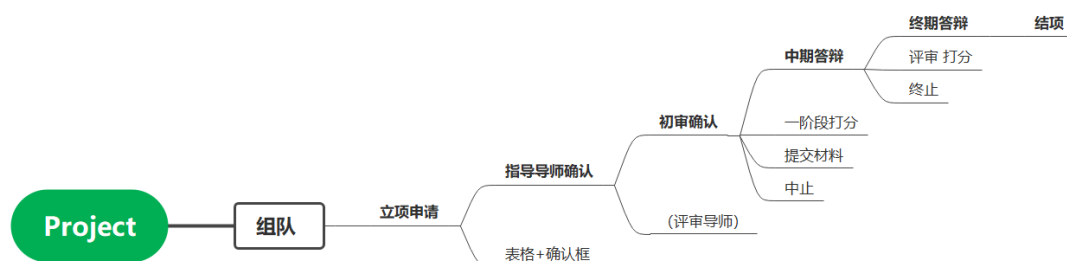
三、使用环境、工具及技术

Windows10, JDK 1.8, IntelliJ IDEA, Tomcat, MySQL, Spring Boot, Thymeleaf, Ubuntu 18.04 Server 等。

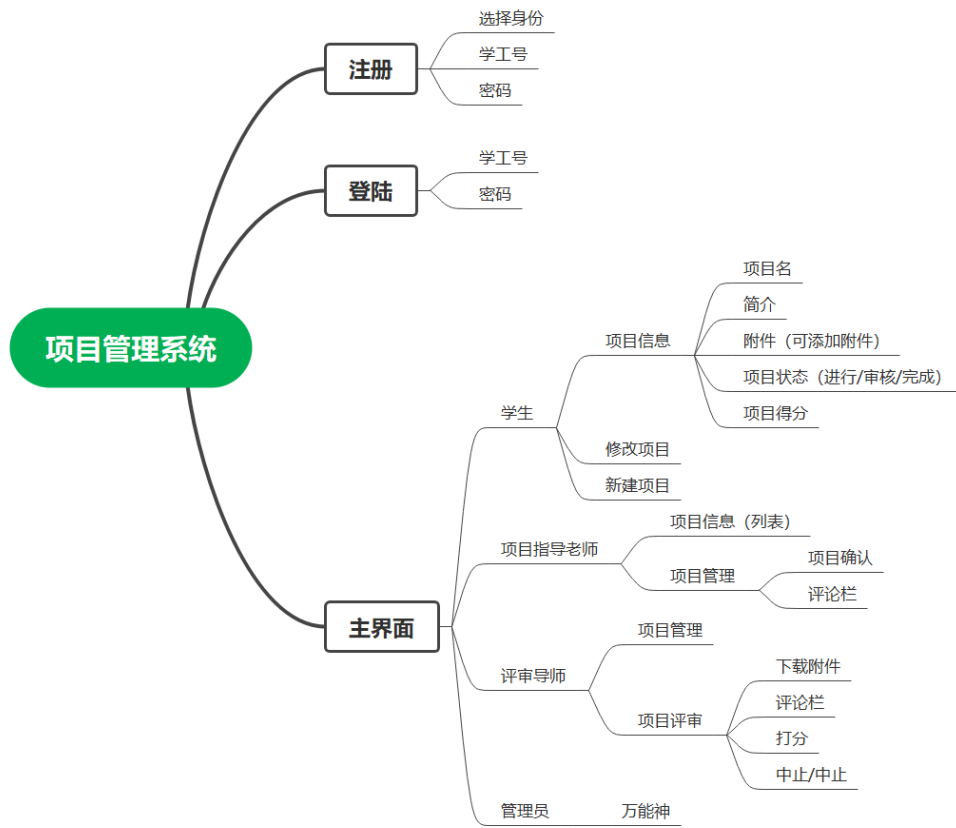
四、分析设计实现过程

（一）需求分析和功能模块

经过对项目需求的分析，得到项目周期结构如图：



具体功能和界面需求如图：



(二) 数据库表的设计

由于主要模块为用户信息及项目信息，因此数据表分为用户（包括学生、项目导师、评审导师、管理员）和项目，其中用户关键信息和个人信息分为两个表。



共创建 3 个数据库表:

- ① **User:** 包括个人信息、身份等;
- ② **Project:** 包括项目名称、得分、成员、项目状态、项目评论、导师等;
- ③ **Profile:** 包括用户具体个人信息, 性别、学校、电话、电子邮箱等;

MySQL 需要注意设置 Charset 为 UTF-8 以解决中文显示问题。

数据库表格设计见下图。











Table Name:

Schema: **project**

Charset/Collation:

Engine:

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
 uid	BIGINT(20)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 ctime	DATETIME	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 mtime	DATETIME	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 name	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 password	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 role	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 status	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL












Table Name:

Schema: **project**

Charset/Collation:

Engine:

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
 uid	BIGINT(20)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 email	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 gender	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 major	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 mobile	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 position	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 school	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 year	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL















Table Name:

Schema: **project**

Charset/Collation:

Engine:

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
 pid	BIGINT(20)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 ctime	DATETIME	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 file_url	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 mtime	DATETIME	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 name	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 profile	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 progress	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 score	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 status	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 teammates	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
 tutors	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

(三) 系统的实现

经过研讨，在 Java Web 项目的基础上，我们主要使用的技术有 Spring Boot、Thymeleaf 模板和 Semantic UI 框架，其中 Semantic UI 框架基于 jQuery。

项目开发使用 IntelliJ IDEA，创建项目流程：

- 选择 Java Enterprise
 - Project SDK: 1.8
 - Project template: Web Application
 - Application server: Tomcat 9.0.39
 - Build tool: Maven
 - Test runner: Junit
 - Languages: Java
- 选择 Libraries and Frameworks: Web Profile
 - Name: ProjectManagement
 - Location: ~/ProjectManagement
 - Group: world.homans
 - Artifact: ProjectManagement
 - Version: 0.0.1-SNAPSHOT

项目结构设置：

- Facets | Web | Add Application specific descriptor...
- Application Server: Tomcat Server

此时 MATA-INF 文件夹会被创建在/src/main/webapp 目录下。

在 Maven 配置中添加相关依赖。

更改 pom.xml 文件。

```
<!-- https://mvnrepository.com/artifact/mysql/mysql-connector-java -->
<dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <version>8.0.22</version>
</dependency>

<!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>javax.servlet-api</artifactId>
    <version>3.1.0</version>
    <scope>provided</scope>
</dependency>

<!-- https://mvnrepository.com/artifact/org.springframework/spring-web -->
<dependency>
    <groupId>org.springframework</groupId>
    <artifactId>spring-web</artifactId>
    <version>5.2.12.RELEASE</version>
    <scope>compile</scope>
</dependency>

<!-- https://mvnrepository.com/artifact/org.springframework/spring-webmvc -->
<dependency>
    <groupId>org.springframework</groupId>
    <artifactId>spring-webmvc</artifactId>
    <version>5.2.12.RELEASE</version>
</dependency>
```

按 Ctrl+Shift+O 刷新让 Maven 自动下载依赖。

配置数据库。

编辑/src/main/resources/application.properties 文件。

```
spring.datasource.url=jdbc:mysql://data.homans.world:8809/project
spring.datasource.username=*****
spring.datasource.password=*****
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver

spring.jpa.database=MYSQL
spring.jpa.show-sql=true
spring.jpa.hibernate.ddl-auto=update
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL5InnoDBDialect
```


对应上传附件过大的问题，另在该文件中添加以下条目。

```
spring.servlet.multipart.max-file-size=200MB
spring.servlet.multipart.max-request-size=200MB
spring.servlet.multipart.enabled=true
```

为了让编程体验更加美好，添加 IDEA 中自带的 Database 连接功能用于 SQL 语法高亮。

- 右击 project name | New | Data Sources and Drivers。
 - Name: Project
 - Host: data.homans.world
 - Port: ****
 - User: *****
 - Password: *****
 - Database: project

具体后台开发过程可详询 GitHub 中的 commit，我们使用 GitHub 做版本管理亦做项目过程记录，链接详询本文档附录。

0f360ab 2 hours ago  59 commits

8 days ago

2 hours ago

7 hours ago


3 hours ago


8 days ago

7 days ago

8 days ago

项目管理平台 - A J2EE project

 [Readme](#)

 [GPL-3.0 License](#)

Releases

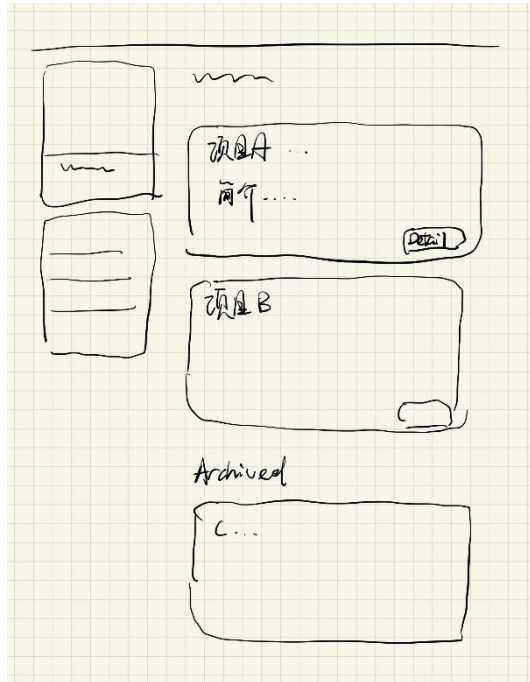
No releases published

[Create a new release](#)

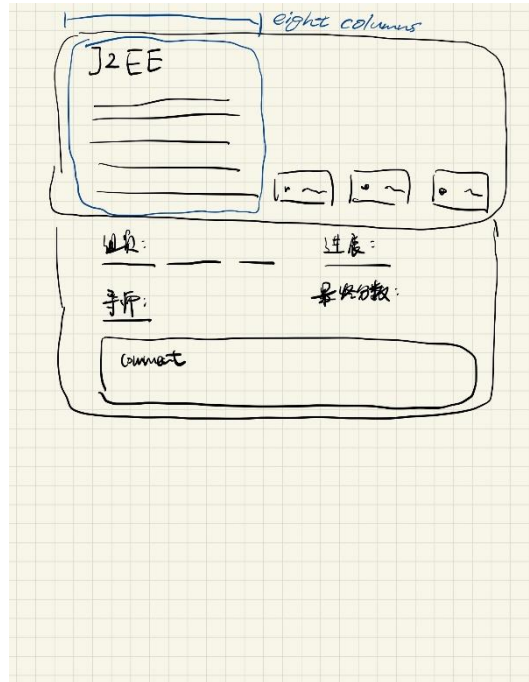
Packages

UI 设计。

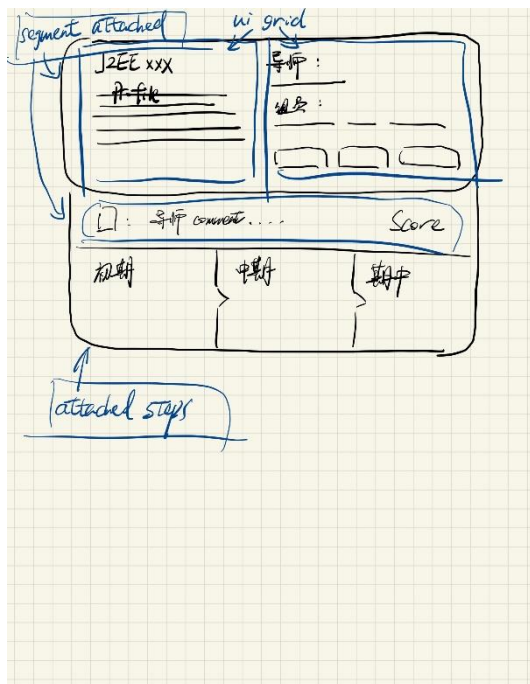
我们使用卡片式设计，让界面更加美观，也让每一个项目信息的显示更加直观！通过点击详情按钮可以直接在管理界面显示详情，或对项目进行删改操作。附设计手稿，最终效果参考本文档成品展示板块。



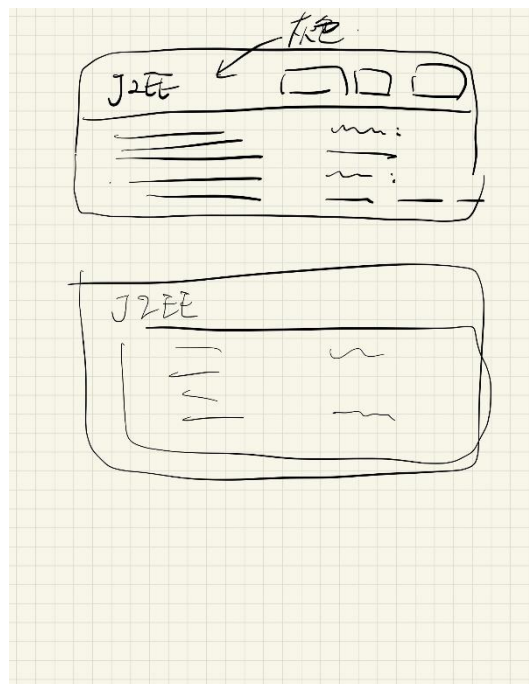
1



2



3



4

生产环境配置。

为了部署项目，我们使用了一台位于香港的谷歌云服务器。服务器系统选用 Ubuntu 18.04，部署时需注意开启 443、80、3306 等相关端口，具体过程此处不再详述。

数据库通过 apt 安装 mysql-server 及 mysql-client，并配置相应用户权限以及开启公网访问。

Java 环境配置，在服务器上通过 wget 下载最新的 JDK 8 压缩文件 jdk-8u271-linux-x64.tar.gz，解压到/usr/local/java/目录，并且配置相关环境变量。

启动和关闭脚本。

为了方便项目部署，我们创建了对应启动和关闭脚本 start.sh 与 stop.sh。需要注意：设置脚本执行权限、脚本需以管理员权限运行、脚本与 project.jar 文件在同一目录下。

```
``start.sh
#!/bin/bash
source /etc/profile;
nohup java -jar project.jar &
...

``stop.sh
#!/bin/bash
PID=$(ps -ef | grep project.jar | grep -v grep | awk '{ print $2 }')
if [ -z "$PID" ]
then
    echo Application stopped.
else
    echo kill $PID
    kill $PID
fi
...
_
```

开启公网访问：配置 nginx。

为了调试到部署阶段的方便，项目端口保留为 8080，通过服务器 nginx 配置 80 和 443 端口转接到 8080 端口实现公网访问项目。

编辑文件/etc/nginx/sites-enabled/project.hoans.world，加入以下内容。

```
server {  
    listen 80;  
    server_name project.hoans.world;  
  
    location / {  
        proxy_pass http://127.0.0.1:8080;  
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;  
        proxy_set_header X-Forwarded-Proto $scheme;  
        proxy_set_header X-Forwarded-Port $server_port;  
    }  
}
```

同时在 [Cloudflare](#) DNS 管理中添加指向服务器 IP 的域名。此后便可以使用 certbot 申请证书，为访问添加 HTTPS 保护，具体步骤详询 [certbot 官方文档](#)。

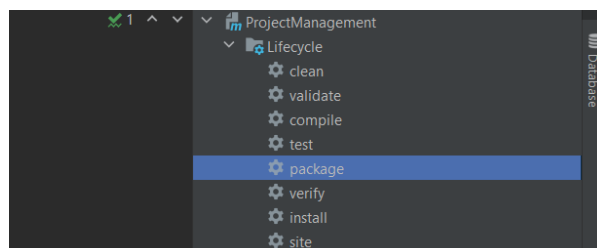
项目部署后遇到客户端请求数据过大的问题。

编辑服务器上的 nginx 配置文件/etc/nginx/nginx.conf，在 http{}块中添加一行配置，使服务器最大可以接受 50MB 的 HTTP 请求。

```
client_max_body_size 50M;
```

打包部署项目。

通过 IDEA 中的 Maven 工具，执行 Lifecycle 中的 package 命令，Maven 会将项目包括 Spring Boot 与其它依赖一同打包至项目的/target 目录。将打包好的 jar 文件上传到服务器后，便可使用配置好的 start.sh 脚本运行。



(四) 成品界面展示

注册

学工号 国 是学生 ☐

姓名

密码

重复密码

注册

什么, 你来过这里? (*^__^*) 登录!

注册界面

王婧民, 欢迎!

王婧民
10195101563
加入自 2021 年 1 月

个人信息
查看和修改个人信息

项目申报
有新的 Idea 吗?

项目管理
管理已经申请的项目

SIMPLE GUIDE

If you want to check Personal Info, click [个人信息]

If you want to create a New Project, click [项目申报]

If you want to update Project Information, click [项目管理]

Create by Homan & Temper.


Enjoy!

主界面

双创项目管理 - 个人信息

https://project.homans.world/profile

双创项目管理 首页 登出



王毓民

10195101563

加入自 2021 年 1 月

个人信息

查看和修改个人信息

项目申报

有新的 Idea 吗?

项目管理

管理已经申请的项目

王毓民的个人信息

注册信息

学工号: 10195101563

姓名: 王毓民

身份: STUDENT

注册时间: 2021 年 1 月

上次修改: 2021 年 1 月

个人信息

性别: MALE

学年: 2019级

学院: 软件工程学院

专业: 软件工程

职称:

电话:

邮箱: 10195101563@stu.ecnu.edu.cn


修改个人信息

个人信息界面

双创项目管理 - 项目申报

https://project.homans.world/application

双创项目管理 首页 登出



王毓民

10195101563

加入自 2021 年 1 月

个人信息

查看和修改个人信息

项目申报

有新的 Idea 吗?

项目管理

管理已经申请的项目

项目申报

项目名称

如何做出一个双创项目

选择你的导师

张四似 20109909

选择你的队友

王毓民 10195101563 我很忙 10215101563 陈俊豪 10185101480

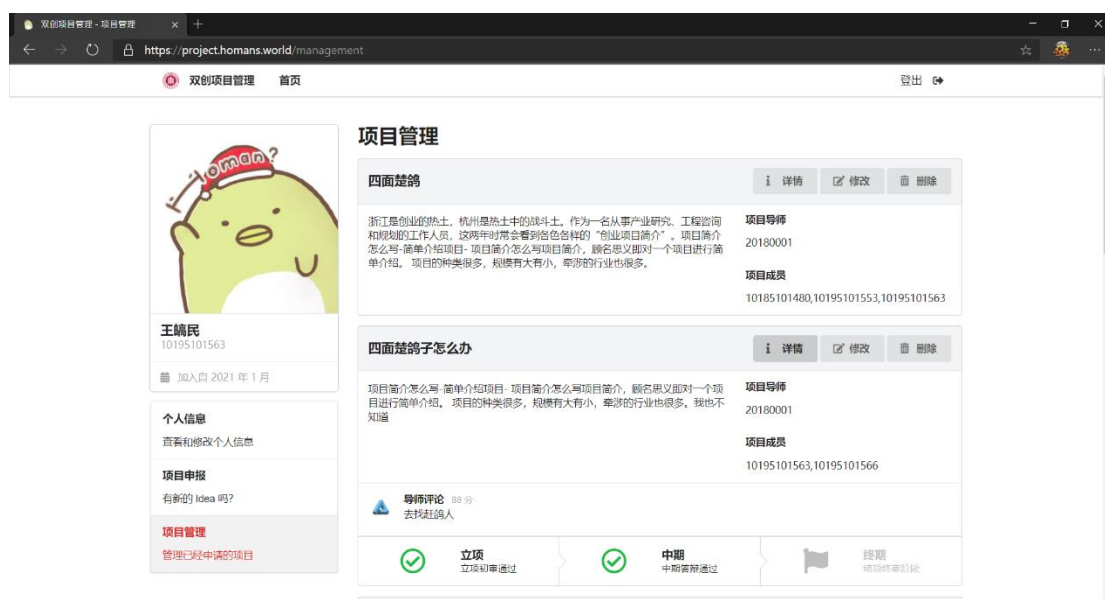
项目简介

上传附件

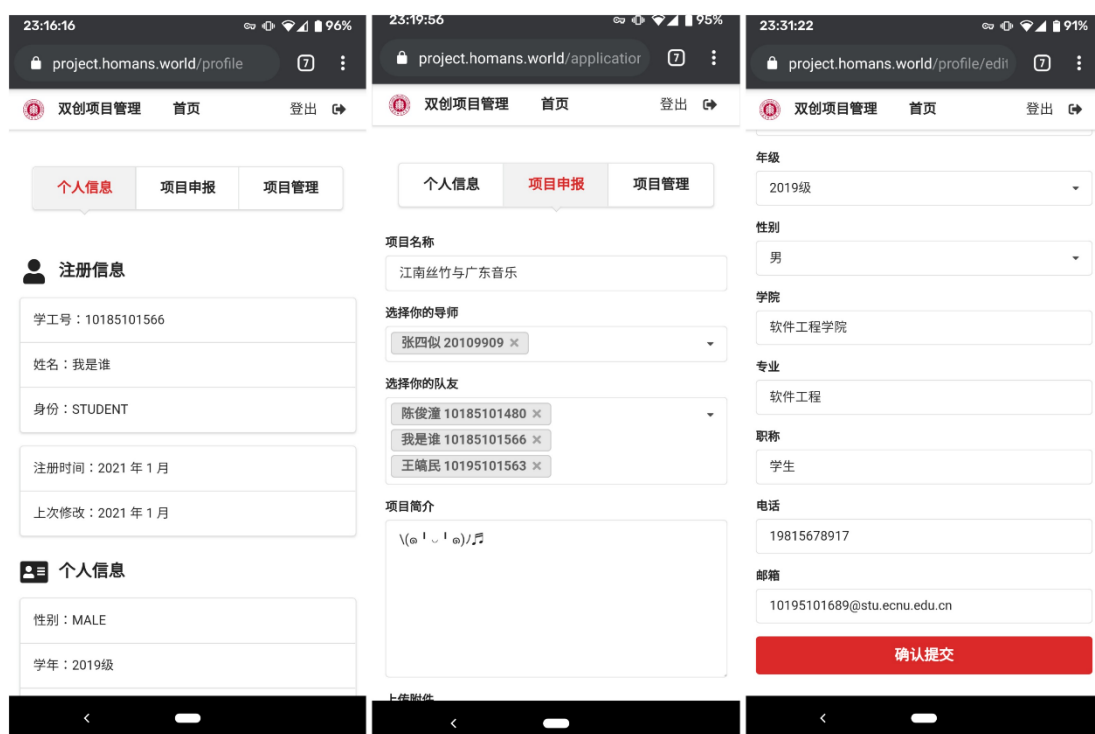
选择文件

确认提交

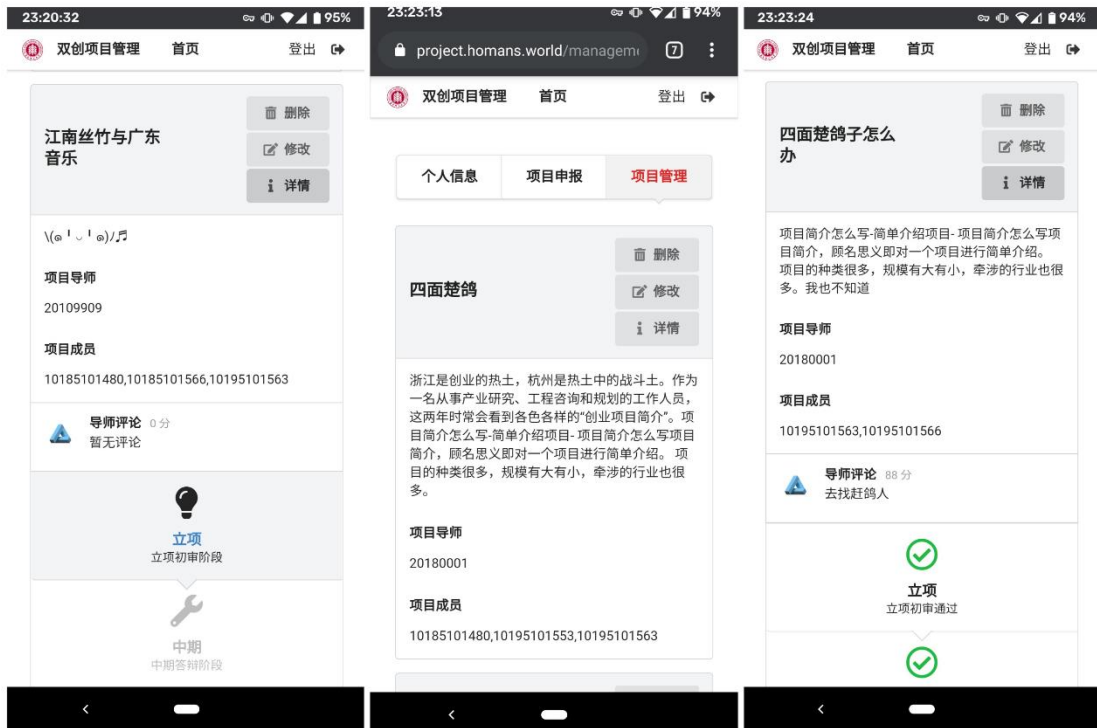
项目申报界面



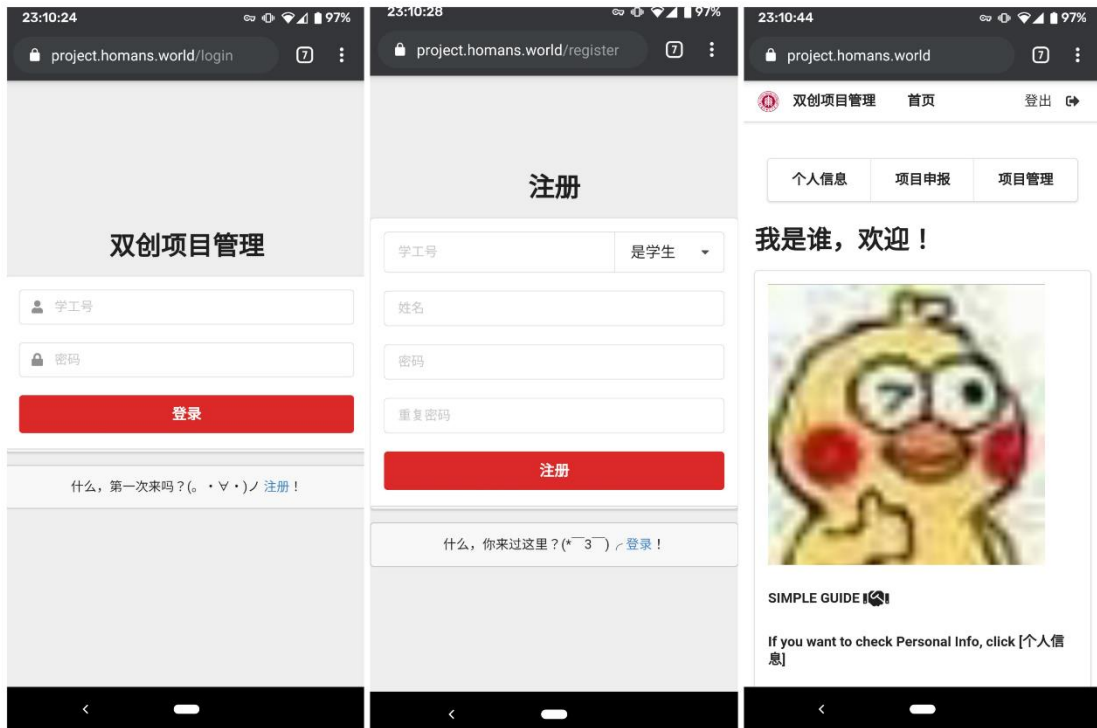
项目管理及详情界面



手机响应式设计优化



手机响应式设计优化



手机响应式设计优化

五、总结

项目开发过程中我们也遇到了许多的小问题，这里列举了一些：

1. 软件安装及插件安装问题，不同组员的环境配置：JDK 需要为 1.8、Maven 设置中国下载镜像等；
2. UI 选用以及 css 文件和 js 文件引用的问题：UI 设计的时候纠结了一段时间，最后选择了组件较丰富，主题可自定义的 Sematic UI，事实证明效果还是很不错的；
3. Thymeleaf 模板的使用方面，由于是新的技术，`{}`、`@{}`选择器的语法需要另外学习，但是实际使用中我们体会到了它的强大：比 JSP 更直观，前端设计可分离，DOM 元素可以通过引用、套用来大量复用 HTML 代码；
4. Spring Boot 框架也是我们初次接触的技术，通过学习 MVC 思想和使用 Spring Boot 注解，我们的项目框架更加的清晰；
5. 文件管理的问题：由于开发环境和生产环境使用两种系统：Windows 10 和 Ubuntu 18.04，文件上传的路径完全不同。最终通过代码实现了自动判断系统以设置相应的文件上传路径；
6. 协作开发的问题。小组中原来只有王竑民同学知道如何使用 GitHub 进行版本管理和协作开发，在他的引导下，小组成员均学会如何使用 GitHub 并顺利进行合作开发。

综上，在本次项目中，我们学习了新的技术，完整的体验了一个 Web 项目的制作过程。经过了必不可少的需求分析、技术栈选用、UI 设计、前后端开发以及上线部署后最终得到我们的成品。

六、附录

项目通过 GitHub 仓库进行小组协同开发，完整的开发过程可见 commit 记录，源码可见。链接：<https://github.com/homanw104/ProjectManagement>。

网站目前部署在 <https://project.homans.world/>，服务器位于 Google Cloud Platform 香港节点，加载速度视网络情况。

类、接口文档详见/doc/index.html 或者代码注释。

Packages

Package

`world.homans.projectmanagement`

`world.homans.projectmanagement.controller`

`world.homans.projectmanagement.entity`

`world.homans.projectmanagement.repository`

`world.homans.projectmanagement.security`

`world.homans.projectmanagement.service`

项目结构树:

```
./src/main
├── java
│   ├── world
│   │   └── homans
│   │       └── projectmanagement
│   │           ├── controller
│   │           │   ├── ApplicationController.java
│   │           │   ├── DownloadController.java
│   │           │   ├── IndexController.java
│   │           │   ├── LoginController.java
│   │           │   ├── LogoutController.java
│   │           │   ├── ManagementController.java
│   │           │   ├── ProfileController.java
│   │           │   ├── ProjectController.java
│   │           │   └── RegisterController.java
│   │           ├── entity
│   │           │   ├── Gender.java           性别 Enum 类
│   │           │   ├── Profile.java         用户具体信息实体类
│   │           │   ├── Progress.java        项目进度 Enum 类
│   │           │   ├── Project.java         项目信息实体类
│   │           │   ├── Role.java           用户角色 Enum 类
│   │           │   ├── Status.java         用户账户情况 Enum 类
│   │           │   └── User.java           用户关键信息实体类
│   │           ├── repository
│   │           │   ├── ProfileRepository.java  用户详细信息仓库
│   │           │   ├── ProjectRepository.java  项目信息仓库
│   │           │   └── UserRepository.java     用户基础信息类
│   │           ├── security
│   │           │   └── SecurityConfig.java     Spring Boot 安全设置
│   │           ├── service
│   │           │   ├── ProfileService.java     用户详细信息增删改查服务
│   │           │   ├── ProjectService.java     项目信息增删改查服务
│   │           │   └── UserService.java        用户基本信息增删改查服务
│   │           └── ProjectManagementApplication.java
└── resources
    ├── application.properties
    ├── static
    │   ├── css
    │   │   ├── common.css
    │   │   ├── login.css
    │   │   └── register.css
    │   └── dist (semantic-ui resources)
```

```
|
|
|   | components ...
|   | semantic.css
|   | semantic.js
|   | semantic.min.css
|   | semantic.min.js
|   | themes
|   |   | basic ...
|   |   | default ...
|   |
|   | img
|   |   | head_chicken.jpg
|   |   | head_tutor.png
|   |   | head_user.jpg
|   |   | icon.png
|   |   | login.jpg
|   |   | logo.jpg
|   |
|   | js
|   |   | common.js
|   |   | jquery-3.5.1.min.js
|   |   | login.js
|   |   | register.js
|
| templates
|   | index
|   |   | index-admin.html
|   |   | index-assessor.html
|   |   | index-student.html
|   |   | index-tutor.html
|   |
|   | management
|   |   | management-admin.html
|   |   | management-assessor.html
|   |   | management-student.html
|   |   | management-tutor.html
|   |
|   | application.html
|   | common.html
|   | login.html
|   | profile-edit.html
|   | profile.html
|   | project-edit.html
|   | register.html
```