总体设计方案

# 设计思想

使用javaee架构mvc模式实现，前后端分离

表现层使用html+css+jquery+ajax

控制层使用springmvc

业务层使用service组件

持久层使用dao组件

# 技术架构

SSM框架：springmvc+spring+mybatis

开发环境：windows10和mac，tomcat、mysql、eclipse、axure（原型）、visio（er图）

采用技术：java+jquery+ajax+springmvc+IOC+AOP+mybatis

Java：核心开发语言

Jquery：简化前端js

Ajax：页面局部刷新，提升用户体验

Springmvc：接收请求、调用业务组件、生成json返回给前端

Spring：管理相关组件

IOC：管理controller、servlet、dao，维护各层关系

AOP：面向切面编程，不修改原有的代码，给系统增加新功能

# 系统架构

系统总体架构包含数据存储层、应用层、web负载层。采用前后端分离架构。

## 前端架构

web 浏览器

数据展示

HTML

CSS

AJAX

JQUERY

GIT

API接口层

JSON

HTTP

## 后端架构

前端应用

控制层

SSM

SpringMVC

Spring

IOC

AOP

GIT

JDBC

JSON

MYSQL

数据持久化

MAVEN

Mybatis

JACKSON

JSON接口服务

# 开发规范

所有请求通过ajax访问

前端页面使用html

请求响应格式为json：{status:xxx,msg:xxx,data:xxx}

# 代码包划分

com.wlfei.mcn.dao 接口层

com.wlfei.mcn.service 服务层

com.wlfei.mcn.controller 控制层

com.wlfei.mcn.entity 实体层

com.wlfei.mcn.util 工具层

com.wlfei.mcn.aspect 切面层

# 功能实现逻辑

## 登录功能

### 发送ajax请求

输入用户名、密码，

点击登录按钮，触发绑定事件

事件获取参数，

事件发送ajax请求

### 服务器处理

请求

-->DispathcerServlet-->HandlerMapping-->Controller.execute()

-->Service(

//检查用户和密码

//错误则返回错误json，否则返回登陆成功json

)

-->Dao-->Mysql(cn\_user)-->返回JSON数据

### ajax回调处理

登陆成功，进入edit.html

登陆失败，提示用户名或密码错误，重新登陆。

## 注册功能

### 发送ajax请求

1. 点击注册按钮，触发绑定事件
2. 事件获取参数
3. 事件发送ajax请求

### 服务器处理

请求

-->DispathcerServlet-->HandlerMapping-->Controller.execute()

-->Service(

//检查用户

//存在重复用户则返回json，否则添加用户

//处理密码

//保存用户密码

)

-->Dao-->Mysql(cn\_user)-->返回JSON数据

### ajax回调处理

成功：注册成功

失败：用户名已存在

## 笔记本列表显示功能

### 发送ajax请求

1. 进入edit.html页面时请求
2. 请求参数：userId
3. 发送json

### 服务器处理

请求

-->DispathcerServlet-->HandlerMapping-->Controller.execute()

-->Service(

//笔记本service接口--查找用户笔记本

//实现类调用dao去查找笔记本

//实现类构建result

//笔记本dao根据userId查找

//mysql表：cn\_notebook

)

-->Dao-->Mysql(cn\_user)-->返回JSON数据

### ajax回调处理

解析json，循环生成笔记本列表项

# 应用部署架构

nginx

tomcat

mysql

web层

应用层

数据层