

WEB 应用开发技术课程项目开发大作业

# 设计 报告

二〇一九年六月



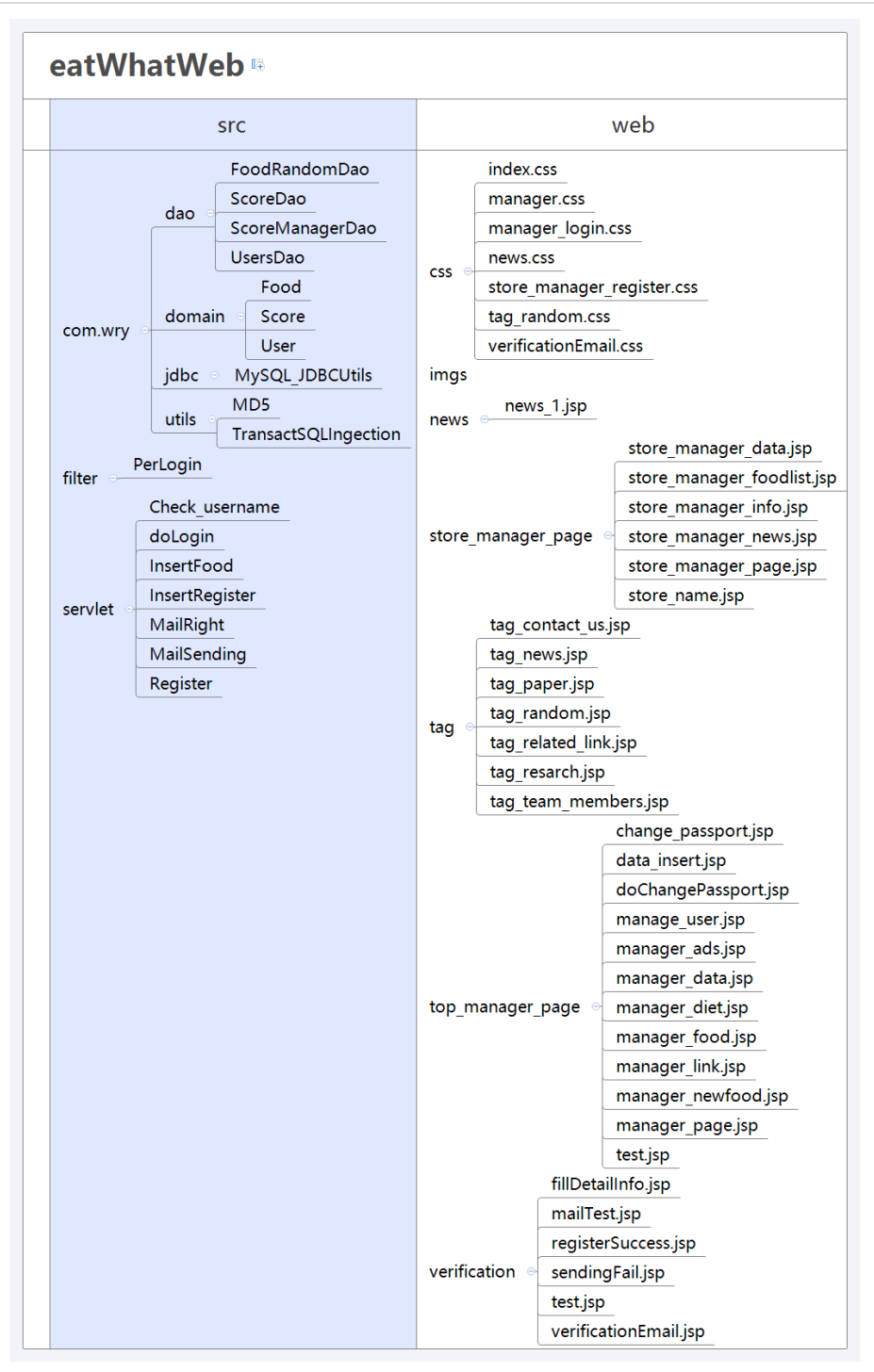
# WEB 应用开发技术课程大作业

学院： 信工      班级： 计工本 1702      学号： 201711010202      姓名： 王汝芸

学院： 信工      班级： 计工本 1701      学号： 201711010111      姓名： 展家慧

分工说明	王汝芸	网站基础框架搭建、数据库设计、安全模块设计
	展家慧	登录功能的实现、前端界面设计、新闻版块设计
主题	“今天中午吃什么”网站设计与开发	
开发环境	框架模式	MVC：JavaBean + JSP + Servlet
	脚本语言	前端：HTML + CSS
		后端：Java Web + JavaScript
	数据库	MySQL 5.5
	部署环境	Tomcat 7.0.93 + 腾讯云 Linux CentOS 7
	其他技术	Ajax
		MD5 数据加密
		SQL 防注入
		JS 级联下拉列表
		SMTP 协议网易服务器邮件发送
文档说明	本文档是今天中午吃什么 Web 项目的指导性文档，该文档确定并描述为完成项目目标所需的各项任务（活动）范围，确定负责制性项目各项任务的全部人员，制定各项任务的时间进度表。	

项目目录结构



## 目录

第 1 章	背景与意义.....	1
1.1	立项背景.....	1
1.1.1	现实环境 .....	1
1.1.2	发展前景 .....	1
1.2	系统的开发意义.....	1
1.2.1	提供决策方案 .....	1
1.2.2	优化方案设计 .....	2
第 2 章	开发环境与开发计划.....	3
2.1	开发环境.....	3
2.2	开发计划.....	3
2.2.1	开发步骤 .....	3
2.2.2	项目角色分工 .....	3
第 3 章	需求分析.....	5
3.1	业务需求.....	5
3.2	用户需求.....	5
3.3	功能需求.....	5
第 4 章	系统设计.....	6
4.1	业务流程.....	6
4.2	数据分析.....	6
4.2.1	实体 .....	6
4.2.2	项目 E-R 模型 .....	7
第 5 章	数据库设计.....	8
5.1	数据表的结构设计.....	8
5.2	字段设计说明.....	8
5.2.1	店铺表 .....	8
5.2.2	店铺管理员表 .....	9
5.3	触发器设计.....	9
第 6 章	技术细节.....	10
6.1	目录结构.....	10
6.1.1	总体目录结构 .....	10
6.1.2	src 目录结构.....	10
6.1.3	Web 目录结构.....	11

6.2	框架模式.....	11
6.2.1	M (Model) .....	12
6.2.2	V (View) .....	12
6.2.3	C (Controller) .....	12
6.3	数据库接入.....	12
6.4	其他技术说明.....	12
6.4.1	随机点餐 .....	12
6.4.2	登录 .....	13
6.4.3	Ajax 技术.....	14
6.4.4	邮箱验证 .....	15
6.4.5	级联下拉框 .....	16
6.4.6	后台数据管理 .....	16
6.4.7	MD5 数据加密.....	18
6.4.8	防 SQL 注入 .....	18
6.4.9	Filter 登录验证.....	18
第 7 章	总结与展望.....	19
7.1	项目开发总结.....	19
7.2	项目后期维护与展望.....	19
7.2.1	Redis .....	19
7.2.2	服务器端邮件发送 .....	19
7.2.3	前端界面 .....	20
7.2.4	移动端界面 .....	20
7.2.5	数据展示 .....	20
7.2.6	完善新闻发布模块 .....	20
7.2.7	用户登录模块 .....	20
7.2.8	分页查询 .....	20
附录	.....	21
	项目 Github 地址.....	21
	项目 URL .....	21



## 第1章 背景与意义

### 1.1 立项背景

#### 1.1.1 现实环境

在当今快节奏、碎片化的时代，人们生活的脚步越来越快，占据人们思维的问题越来越多，越来越繁杂。作为新时代的大学生，学习专业技能只是大学生活的一部分，还有一项很重要的任务便是适应当代社会的节奏，让自己尽快摆脱“学生思维”，成为能够独当一面，自立于社会的成年人。而且大学相较于中学时期按部就班的学习生活，更为丰富，更为“不可确定”。这使得大学结合了学习与工作两方面的生活，极为丰富且灵活。而在这样充实又匆忙的生活中，一直有一个问题困扰着我们——今天中午吃什么。

#### 1.1.2 发展前景

民以食为天。中华饮食文化博大精深、源远流长，在世界上享有很高的声誉。中国人讲吃，不仅仅是一日三餐，解渴充饥，吃的文化已经超越了“吃”本身，获得了更为深刻的社会意义。就其深层内涵来看，可以概括成四个字：精、美、情、礼。这四个字，反映了饮食活动过程中饮食品质、审美体验、情感活动、社会功能等所包含的独特文化意蕴，也反映了饮食文化与中华优秀传统文化的密切联系。

综上，“今天中午吃什么”，是一个困扰我们已久且将继续困扰我们的亘古不变的问题。而我们的系统，就解决这个问题做出了初步的回答。

### 1.2 系统的开发意义

#### 1.2.1 提供决策方案

决策问题是指实际状态与期望状态之间存在的一种需要缩小或排除的差距。每个人在考虑午饭的选择时都会有一个期望，或是一个具体的目标，或是一个大体的方向。而现实中解决这个问题的阻碍有两个：一是有具体目标但是并不可行；二是只有大体方向但是由于对实际情况的掌握情况不好，不能很好地规划出具体目标。而我们的系统就致力于解决这两个问题，通过数据给出一个符合个人期望又符合实际情况的选择。

### 1.2.2 优化方案设计

工程设计中最优化问题的一般提法是要选择一组参数，在满足一系列有关的限制条件下，使设计指标达到最优值。因此，最优化问题通常可以表示为数学规划形式的问题。这就将一个实际问题转化为了一个计算机可以解决的问题，我们的系统可以在现有条件下得到一个实际情况和个人期望下的最优解。



## 第2章 开发环境与开发计划

### 2.1 开发环境

软件开发环境（Software Development Environment，SDE）是指在基本硬件和数字软件的基础上，为支持系统软件和应用软件的工程化开发和维护而使用的一组软件，简称 SDE。它由软件工具和环境集成机制构成，前者用以支持软件开发的相关过程、活动和任务，后者为工具集成和软件的开发、维护及管理提供统一的支持。

本项目开发环境见下表 2-1。

表 2-1 项目开发环境

开发环境	具体内容
脚本	HTML5+CSS+Java+JavaScript
框架模式	MVC
部署环境	Tomcat7+腾讯云 Linux CentOS7
数据库	MySQL
IDE	JetBrains IntelliJ IDEA+eclipse

### 2.2 开发计划

#### 2.2.1 开发步骤

- 1) 调查餐厅菜品安排信息以及同学们的点餐习惯；
- 2) 建立数据库；
- 3) 完善交互页面；
- 4) 系统发布测试。

#### 2.2.2 项目角色分工

对于一个成功的项目来说，角色分工的重要性不言而喻，本小组项目角色分工如下表 2-2 所示。

表 2-2 项目角色分工

负责方向	负责人员
总体设计	王汝芸
数据库设计	展家慧、王汝芸
代码实现	王汝芸、展家慧
美工	王汝芸、展家慧
文档	王汝芸、展家慧
代码整合	王汝芸

## 第3章 需求分析

需求分析是指理解用户需求，就软件功能与客户达成一致，估计软件风险和评估项目代价，最终形成开发计划的一个复杂过程。在这个过程中，用户的确是处在主导地位，需求分析工程师和项目经理要负责整理用户需求，为之后的软件设计打下基础。

### 3.1 业务需求

业务需求反映了组织机构或客户对系统、产品高层次的目标要求。本系统业务需求包括模块化解决点餐问题、培养用户使用习惯以及商家入驻需求。

### 3.2 用户需求

用户需求是产品执行过程中，最有难度的需求挖掘。该跟随用户还是引导用户，该如何深层次的挖掘用户需求，用户的需求该如何转换为产品需求。

本系统项目需求分析如下图 3-1 所示。

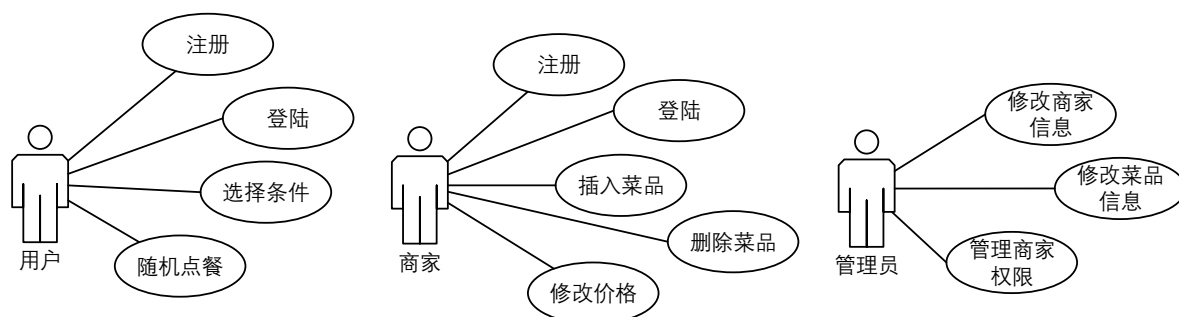


图 3-1 项目用户需求分析

### 3.3 功能需求

功能需求规定开发人员必须在产品中实现的软件功能，用户利用这些功能来完成任任务，满足业务需求。功能需求有时也被称作行为需求。

本项目功能需求包括管理员帐号的登录注册、数据查看，普通用户的使用打分、数据查看，后台数据增删改查及查询优化。

## 第4章 系统设计

### 4.1 业务流程

业务流程，是为达到特定的价值目标而由不同的人分别共同完成的一系列活动。活动之间不仅有严格的先后顺序限定，而且活动的内容、方式、责任等也都必须有明确的安排和界定，以使不同活动在不同岗位角色之间进行转手交接成为可能。活动与活动之间在时间和空间上的转移可以有较大的跨度。

本项目业务流程图如下图 4-1 所示。

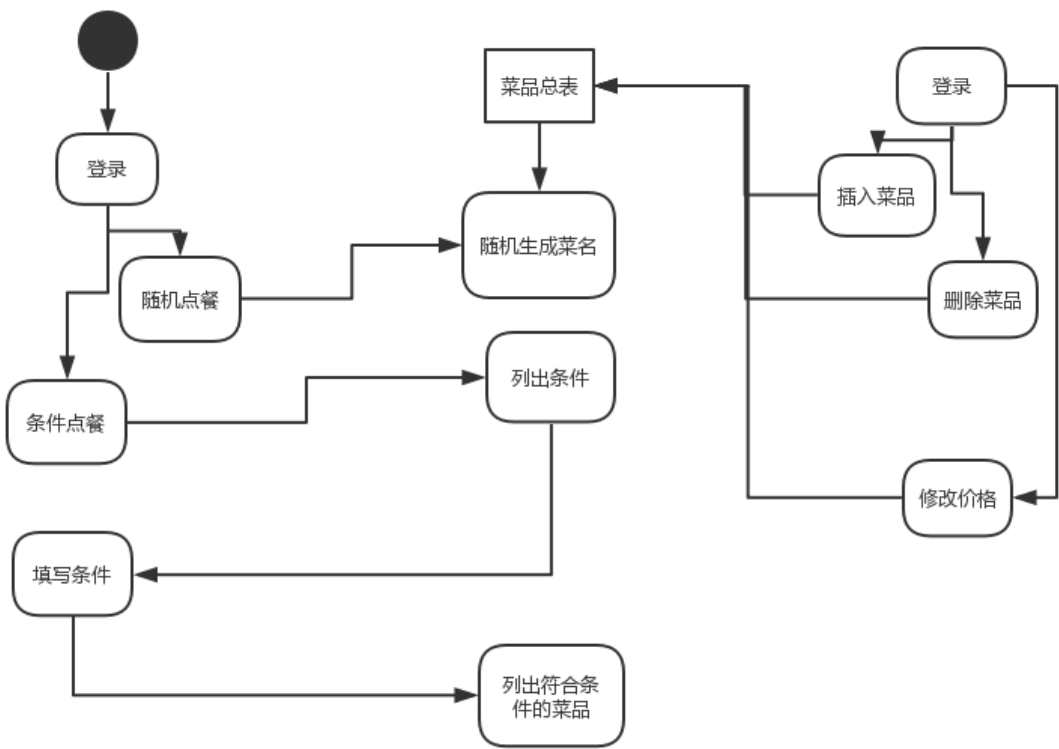


图 4-1 业务流程图

### 4.2 数据分析

#### 4.2.1 实体

实体是客观存在并可相互区别的事物。就数据库而言，实体往往指某类事物的集合。把每一类数据对象的个体称为实体。本项目共设计包括管理员、用户、商家三个实体。

### 4.2.2 项目 E-R 模型

ER 模型，全称为实体联系模型、实体关系模型或实体联系模式图（ERD，Entity-relationship model）由美籍华裔计算机科学家陈品山提出，是概念数据模型的高层描述所使用的数据模型或模式图。

本项目 E-R 模型见下图 4-2 所示。

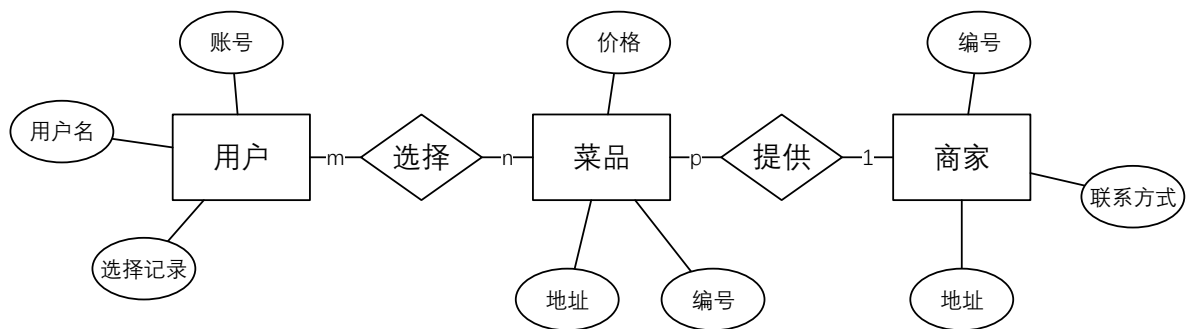


图 4-2 项目 E-R 模型

## 第5章 数据库设计

### 5.1 数据表的结构设计

本项目共创建三张数据表，分别存储实体及其属性。数据表见下表 5-1、5-2 及 5-3 所示。

表 5-1 tb\_store

字段名称	store_id	store_name	store_loc
字段含义	店铺编号	店铺名称	店铺地址
字段类型	VARCHAR(9)	VARCHAR(n)	VARCHAR(n)

表 5-2 tb\_food

字段名称	food_id	food_name	food_price	food_store_id	star_number
字段含义	菜品编号	菜品名称	菜品价格	菜品所在店铺编号	菜品点赞数
字段类型	INT	VARCHAR(n)	INT	CHAR(n)	INT

表 5-3 tb\_store\_manager

字段名称	id	username	password	manager_store_id
字段含义	店主编号	用户名	密码	所管理店铺编号
字段类型	CHAR(9)	VARCHAR(n)	CAHR(n)	CHAR(n)

### 5.2 字段设计说明

#### 5.2.1 店铺表

store\_id 为店铺编码，共 9 位。其中 1—2 位用于表示食堂；3—4 位用于表示楼层；5—6 位用于表示具体食堂（南北）；7—9 位为校验位，用于区别具体窗口号。

如下图 5-1 所示，店铺 ID 010201002 表示一餐二楼北第 2 个注册窗口。

store_id	store_name	store_loc
010101001	炒饭	一餐一楼
010201001	山西刀削面	一餐二楼北
010201002	南京灌汤包	一餐二楼北

图 5-1 店铺表部分数据



### 5.2.2 店铺管理员表

店铺管理员 id 与 manager\_store\_id 数值相等，目前设计为一一对应关系。该设计可进行扩展，以应对日后一店多人情况。

### 5.3 触发器 Trigger 设计

数据表 tb\_store\_manager 中，manager\_store\_id 指向该店铺管理员所管理的店铺，为数据表 tb\_store 外键，管理员与店铺为一一对应关系，当管理员账号注销时，店铺随之注销。

## 第6章 技术细节

该项目包含 110 个文件，总计 6928 行代码，初步完成了一个可行的 Web 项目。

### 6.1 目录结构

#### 6.1.1 总体目录结构

项目总体目录结构如下图 6-1 所示。

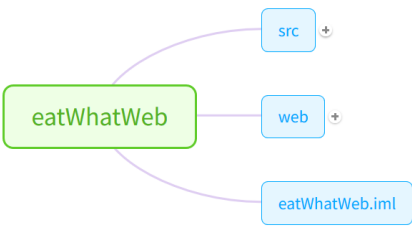


图 6-1 项目总体目录结构

#### 6.1.2 src 目录结构

项目/src 结构如下图 6-2 所示。

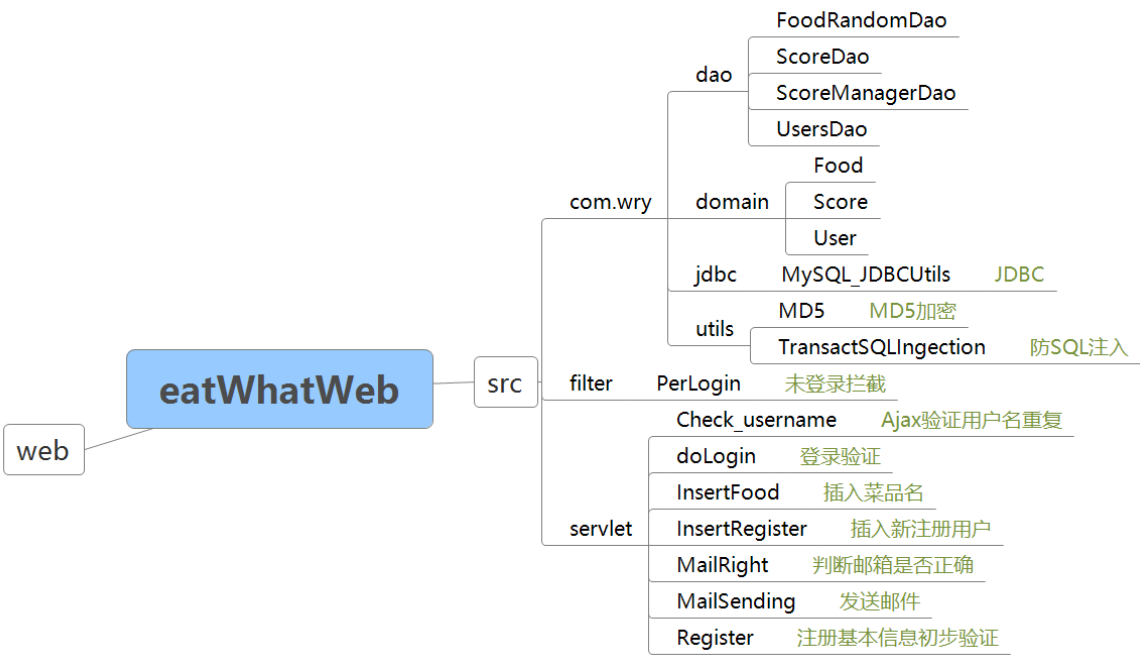


图 6-2 src 目录结构



### 6.1.3 web 目录结构

项目/web 目录结构如下图 6-3 所示。

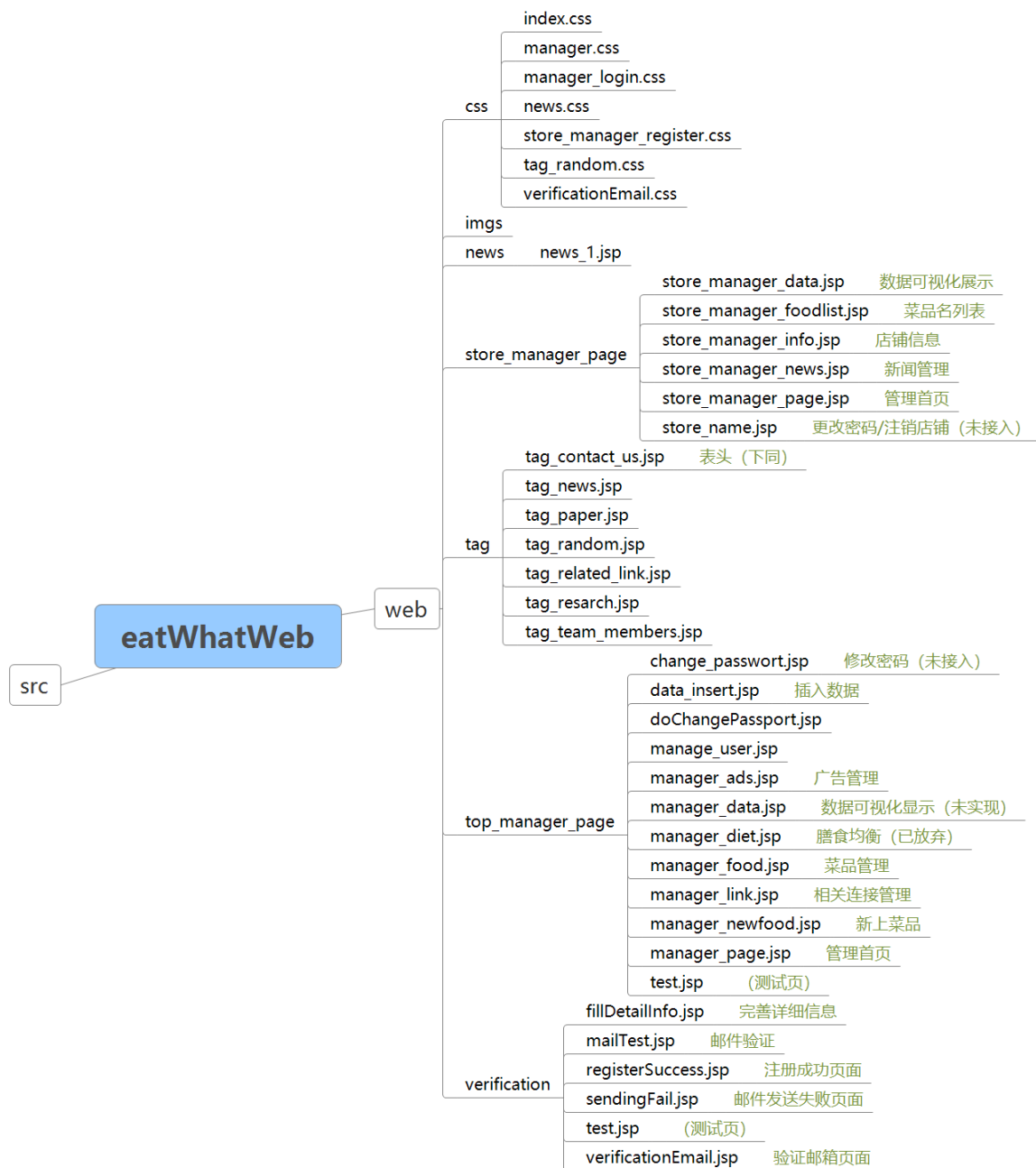


图 6-3 Web 目录结构

## 6.2 框架模式

采用 MVC 框架模式。MVC 全名是 Model View Controller，是模型(model)－视图(view)－控制器(controller)的缩写，一种软件设计典范，用一种业务逻辑、数据、

界面显示分离的方法组织代码，将业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。MVC 被独特的发展起来用于映射传统的输入、处理和输出功能在一个逻辑的图形化用户界面的结构中。

### 6.2.1 M (Model)

模型是应用程序中用于处理应用程序数据逻辑的部分。通常模型对象负责在数据库中存取数据。在 JavaWeb 中，通常应用 JavaBean 及四种作用域作为 M 层。

### 6.2.2 V (View)

视图是应用程序中处理数据显示的部分。通常视图是依据模型数据创建的。在 JavaWeb 中，通常采用 HTML+CSS 的方式进行页面的展示。

### 6.2.3 C (Controller)

控制器是应用程序中处理用户交互的部分。通常控制器负责从视图读取数据，控制用户输入，并向模型发送数据。本项目采用 Servlet 作为控制器层技术。

## 6.3 数据库接入

JDBC 是一种用于执行 SQL 语句的 Java API，可以为多种关系数据库提供统一访问，它由一组用 Java 语言编写的类和接口组成。JDBC 提供了一种基准，据此可以构建更高级的工具和接口，使数据库开发人员能够编写数据库应用程序。采用 Java 与 MySQL 结合的方式，通过中间件 JDBC 进行连接，达到在 java 文件下直接写 SQL 语句即可实现对数据库操作的效果。

## 6.4 其他技术说明

### 6.4.1 随机点餐

随机点餐会从数据库中随机查询一条数据，并返回给前端显示，显示效果如下图 6-4 所示。



图 6-4 随机点餐页面

同时随机点餐会提供无限刷新以及详细信息展示功能

#### 1) 刷新

通过 Java Script 下 `location.reload()` 方法实现网页刷新。

#### 2) 详细信息

通过 Java Script 中 `alert` 弹窗实现详细信息展示，展示效果如下图 6-5 所示。



图 6-5 详细信息展示

### 6.4.2 登录

页面底栏右侧设有管理员入口，登录页提供登录与注册两项业务。

其中，登录时会将表单传入数据与数据库进行比对，匹配则通过登录，同时在 `session` 域中记录用户名。若登录失败，会通过 `Alert` 方法弹出提示，并留在原地。

登录页面如下图 6-6 所示。

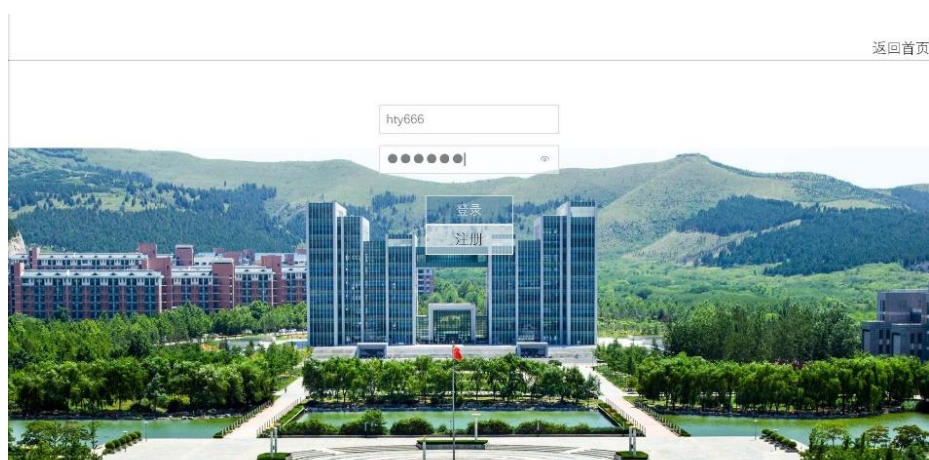


图 6-6 登录界面

#### 6. 4. 3Ajax 技术

店铺可通过注册管理员的方式，实现店铺注册及管理  
注册页面如下图 6-7 所示。



图 6-7 注册界面

其中，用户名验证通过 Ajax 技术，实现无需提交整个网页/表单，后台即可获取输入框值与数据库进行比对。由于整个项目未使用 Spring 框架，故该处 Ajax 通过 JavaScript 与 Servlet 实现对数据的简单处理与反馈。JS 通过判断鼠标离开输入框事件，触发动作向后台传递 input 文本框值，由 Servlet 进行处理，与数据库数据进行匹配，得到判断结果，并将结果显示在前端页面。

具体效果如下图 6-8、6-9 所示。



图 6-8 用户名可以使用



图 6-9 用户名已存在

#### 6.4.4 邮箱验证

考虑到后期修改密码、注销店铺等需求，在注册时便记录验证邮箱。后台通过接入 Java 邮件发送 API，借助 STMP 协议网易 163 服务器实现验证邮件的发送。

后台程序首先随机生成一组四位邮件验证码，发送邮件。验证页面填入验证码并与后台数据进行比对，若成功则进入店铺信息填写页面；若错误则留在原地提示信息；若发送失败则给出信息，返回邮箱填写页。

邮件验证页与邮件内容如下图 6-10、6-11 所示



图 6-10 注册页面



图 6-11 验证邮件

6.4.5 级联下拉框

店铺注册页面通过 JavaScript 级联下拉框实现食堂与具体位置的选择，如下图 6-12、6-13 所示。



图 6-12 级联下拉框 1



图 6-13 级联下拉框 2

同时后台可接收下拉框传递参数，自动补充生成店铺 ID 号，为后续管理奠定基础。

6.4.6 后台数据管理

店铺后台管理界面如下图 6-14、6-15 所示。



图 6-14 网站高级管理员后台管理界面





图 6-15 普通店铺管理员后台管理界面

其中，菜品管理可实现菜品数据的插入；推送管理目前未实现；数据查看如下图 6-16 所示。



图 6-16 数据查看界面

店铺信息如下图所示。



图 6-17 店铺信息查看界面

#### 6.4.7 MD5 数据加密

网站密码通过 MD5 加密的方式存储在数据库中，避免明文存储造成数据泄露的严重事故。后端代码采用 MD5 加密方法，在与 Java Bean 进行交互式进行加密存储。同理在查询时，也通过加密输入文本与后台数据库文本比对进行验证。密码在数据库中存储形式如下图 6-18 所示。

username	password
hty007	e10adc3949ba59abbe56e057f20f883e

图 6-18 MD5 加密后密码在数据库中存储状态（图中明文密码为 123456）

#### 6.4.8 防 SQL 注入

SQL 注入攻击是 Web 安全史上的一个重要里程碑，它从 1999 年首次进入人们的视线，至今已经有十几年的历史了，虽然我们现在已经有了很全面的防范对策，但是它的威力仍然不容小觑，SQL 注入攻击至今仍然是 Web 安全领域中的一个重要组成部分。

页面在所有输入框位置前添加了一层 Filter，通过 java 正则表达式检测、replace 方法过滤异常输入。

后台 DAO 层的所有 SQL 语句采用预编译语句集形式编写。

#### 6.4.9 Filter 登录验证

网站在所有管理页面前都加了一层 Filter，若用户未登录，即 session 中变量值为空，则直接拦截请求，并通过 Alert 弹出提示、跳转至登录页面。





## 第7章 总结与展望

### 7.1 项目开发总结

本项目采用 MVC 开发模式，搭建了一个小型网站，项目基本实现了一个常规网站应有的功能，同时兼顾自身特点，设计了随机点餐模块。本项目实现了基本的前后端分离，某些短小的 Java 语句或 JS 方法选择嵌入在 JSP 中进行实现。项目在安全性方面作出了较为全面的考虑，实现了 MD5 密码加密、SQL 防注入等功能。为后期继续完善提供基础保障。

本项目为解决大学生当下“今天中午吃什么”问题提供了一剂良方。这个困扰我们已久且将继续困扰我们的亘古不变的问题，由于本项目开发的完成，得到了初步的解决。

### 7.2 项目后期维护与展望

#### 7.2.1 Redis

目前项目后台数据库全部采用经典的关系型数据库 MySQL 实现，MySQL 以强大稳定性与多平台适应性的特点为人称道，但在当下大数据的浪潮前，MySQL 显得略微古板。后期拟引入强大的 noSQL 数据库 Redis 进行非重要数据的存储。Redis 是一个高性能的 key-value 数据库。Redis 的出现，很大程度补偿了 memcached 这类 key/value 存储的不足，在部分场合可以对关系数据库起到很好的补充作用。Redis 支持主从同步。数据可以从主服务器向任意数量的从服务器上同步，从服务器可以是关联其他从服务器的主服务器。这使得 Redis 可执行单层树复制。存盘可以有意无意的对数据进行写操作。由于完全实现了发布/订阅机制，使得从数据库在任何地方同步树时，可订阅一个频道并接收主服务器完整的消息发布记录。同步对读取操作的可扩展性和数据冗余很有帮助。

因此，许多经典的网站都采用 Redis+MySQL 进行后台数据的存储。本项目后期亦拟采取该架构，实现数据高速查询与稳定性的兼顾。

#### 7.2.2 服务器端邮件发送

项目目前部署于腾讯云服务器，但由于 STMP/POP3 协议原因，在服务器上无法完成邮件的发送，近期会着手解决此 BUG。

### 7.2.3 前端界面

决定一个项目是否能被用户认可，很大的一方面是界面的美观性与实用性。本项目设计初期，借鉴了某官网布局，导致后来意识到该问题时项目已完成大半，前端的重构耗费将大量精力，因此只能在其后的每个模块尽力调整。未来如果项目反馈良好，将着手前端界面的重构。

### 7.2.4 移动端界面

由于使用场合的限制，“今天中午吃什么”更适合于移动端访问。因此，将尽快着手设计移动端。

### 7.2.5 数据展示

如何直观反映数据的变化，为商家提供简洁明了的决策支持，亦是本项目的努力方向之一。后期拟接入百度 Echarts 端口，为店铺管理员提供数据的查看。同时考虑接入 Kibana 进行数据分析，为项目开发人员提供运维、迭代方向支持。

### 7.2.6 完善新闻发布模块

网站设计初期考虑到宣传需要，后端预留了新闻发布接口。后期会着手完善前端页面，借助富文本编辑器进行 UEditor 实现新闻的发布。

### 7.2.7 用户登录模块

当前用户登录模块较为简易，后期将借助安全轻量的 Shiro 框架，提升登陆体验、提高效率。Shior 验证技术基于 Subject 类，实现了 login() 方法、logout() 方法，如果登录失败会抛出相应异常。

### 7.2.8 分页查询

当前项目菜品名的查询会将查询结果全部显示在页面上，后期会着手设计分页查询，以期达到更好的用户体验。

## 附录

### 项目 Github 地址

<https://github.com/RuYunW/eatWhatWeb>



### 项目 URL

<http://www.ruyunw.club/eatWhatWeb/index.jsp>

