**点菜系统**

**一、简要需求分析：**

1、注册

2、登陆

3-1、菜单录入（管理员）

3-2、点菜模块（普通用户）

**二、项目说明：**

**1、整体架构：**

--data #数据

--db #数据库

--menu.db

--static #静态资源

--css #css文件

--js #js文件

--templates #html模板

--index.html #主页

--login.html #登陆

--register.html #注册

--admin.html #管理员页面

--404.html #404页面

--app.py #flask文件，处理前端请求和获取数据库资源

--db.py #处理有关数据库的增删查改

--cal.py #线性规划计算

--insert\_db.py #将数据添加到数据库

**2、功能描述**

**//按照流程，对部分文件实现的功能描述**

--login.html、register.html #登陆注册，完成了简单检验

#提供两个账号：用来登陆:

管理员:username(admin) password(admin)

普通用户:username(guest1) password(guest)

--登陆：

--前端: 表单不能为空

--后端：用户名是否存在

密码是否正确

--登陆成功：

如果用户类型为管理员，重定向到admin.html

如果用户类型为普通用户，重定向到index.html

--注册：

--前端：表单不能为空

用户名长度要求在5-20

密码长度要求在5-20

两次密码一致

--注册成功：

重定向到login.html

--admin.html #管理员页面，完成单菜品添加，套餐添加

--单菜品添加 #完成了一些检验

--套餐添加 #完成了一些检验，包括套餐价格不能超过单菜品总和

--index.html #主页，完成展示，选购，组合套餐，提交记录等功能

--top #右上角的‘登出’，可以注销登录

#如果登陆成功，会显示出用户名

--left

--单菜品展示 如果想要添加到购物车，点击‘加入点菜单’按钮即可

--套餐展示 如果想要添加到购物车，点击‘加入点菜单’按钮即可

--right #实现三个div的切换（购物车，空购物车，正在支付）

--空购物车 #如果购物车为空，切换为该div

--购物车 #展示选购的所有的单品，可以做添加、减少、移除操作

如果选购的物品再加一个，刚好可以省更多钱，就在下方出现提示（如选择咖啡）和添加按钮

--正在支付 #展示最省钱的组合方式，点击支付按钮，可提交订单

#点击提交的时候，会将时间和用户提交到数据库

--404.html #如果找不到页面会被调用

**3、技术使用：**

前端： --html、css(less)、js

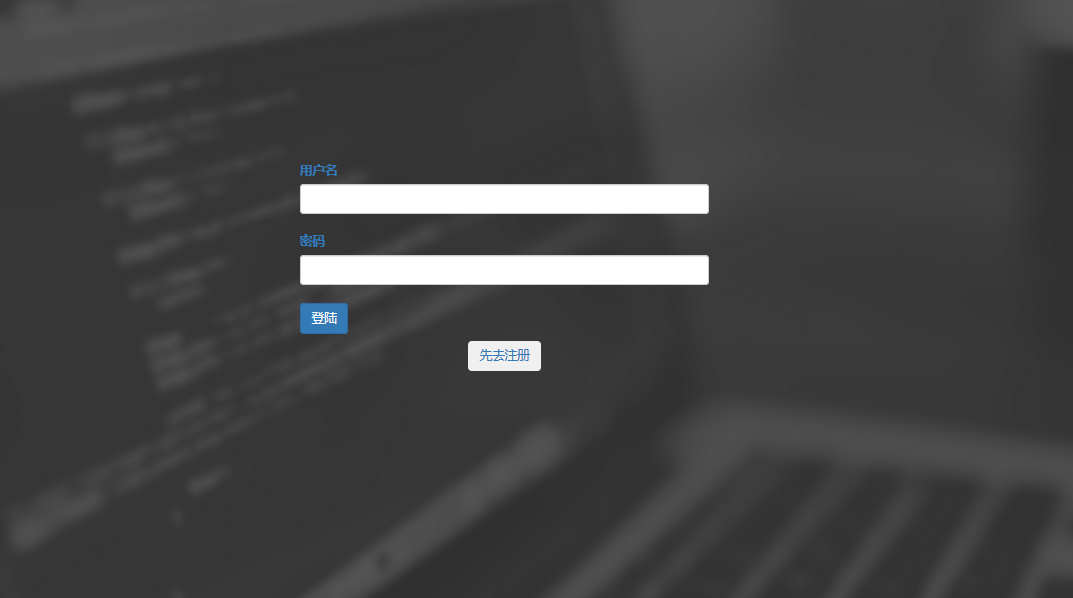
--框架和库：Vue.js、Jquery、Bootstrap

后端： --python

--框架：Flask

**4、效果实现：**

1. 登陆注册



* 1. 表单检验（包括非空，长度，密码和确认密码相同）



1-3后提交到服务器，进行检验（首先查看是否有该用户，再检查密码是否正确）



1. 后台管理

1)可完成单菜品和套餐的添加

2)实现了基本的表单检验



4、主页（index.html）

头部显示用户名，可登出

中间：

左侧可进行单品和套餐的切换，可完成选购功能

右侧共三个组件，空购物车、购物车、菜单提交。可在不同条件下切换

空购物车：



菜单提交组件：



**三、重点难点分析：**

1、数据使用比较频繁，且频繁修改。原生js实现比较难

答：使用Vue.js。Vue.js是MVVM框架，实现了双向数据绑定。而在我的项目中，数据使用比较频繁，且可能频繁修改。使用Vue.js会更加方便一些

2、如何将Vue.js和Flask结合起来

答：

两者本身不存在矛盾，但是Flask使用的jinja2模板的‘{{}}’和Vue.js中的‘{{}}’会出现一些问题。

解决思路：将Vue.js中使用到的数据展示都改写为完整形式 如：原先的<p>{{price}}</p>改写为<p v-text="price"></p>

3、登陆注册的检验

答：分开检验，将可以在前端做的都在前端完成。如非空检验，长度检验，套餐价格和单菜品价格检验

其余需要数据库支持的，放在后端检验

4、“套餐添加只能添加单菜品中有的”

答：用select来实现。直接将所有的单菜品显示出来，由管理员自己选择。而不必填写

5、想要在提交订单信息的同时，将用户名和时间也提交到数据库中

答：利用主页头部的用户名显示，获取到相应信息。时间由Date对象直接得到。