

项目框架设计v0.08.22

- 目前还需要解决的：微信登录、微信支付、服务器域名备案
- 技术栈：
 - 前端：
 - web后台：
 - 框架：vue 2.9.6
 - 前后端交互：axios
 - 微信小程序：
 - 框架：建议用mpvue
 - [快速上手](#) | [mpvue.com](#)
 - [使用手册](#) | [mpvue.com](#)
 - [微信小程序mpvue-部署资源](#) [Vanilla1119的博客-CSDN博客](#)
 - 前后端交互：wx.request
 - [微信小程序https请求wx.request详细教程](#) (360doc.com).
 - [19. 尚硅谷 Mpvue 前后端交互实现](#) - 知乎 (zhihu.com).
 - 后端：
 - 语言：python 3.6.5
 - 框架：flask 1.1.2
 - 数据库：sqlite8
- 项目结构：
 - 设计理念
 - 前后端开发能及时协调修改
 - 组织结构清晰，降低维护难度
 - 具体结构
 - frontend/ 【前端文件存放地址；在指定端口上启动(用于访问页面)】
 - web/ 【vue web后台管理页面项目源码】
 - 【具体结构】
 - wx/ 【mpvue 微信小程序项目】
 - 【具体结构】
 - backend/ 【后端项目存放地址；在指定端口上启动(用于用户端收发信息)】
 - DataBaseFolder/ 【数据库相关模块】
 - Construct/ 【填充初始数据】
 - Interface/ 【用于修改对应数据库的接口文件】
 - DishBaseModify.py

- PyDirectlyAdd
 - PyList
 -
- OrderBaseModify.py
- RestaurantBaseModify.py
- UserBaseModify.py
- InterfaceHelper.py
- Models/ 【数据库各表项的设计文件】
 - CourierModels/
 - CourierBaseInfo.py 【快递信息数据库（没用）】
 - OrderData/
 - OrderBase.py 【订单数据库】
 - RestaurantModels/
 - DishBase.py 【菜品数据库】
 - RestaurantBase.py 【卖家数据库】
 - UserModels/
 - UserBaseInfo.py 【买家数据库】
 - ModelsParameter.py 【定义父类，赋予所有子类to_json方法】
- controllers/ 【用于相应request请求的模块文件】
- config/ 【存储数据库、后端等的配置文件】
- application.py 【后端应用实例化文件】
- manager.py 【后端服务器入口文件】
- readme.md 【说明文件，解释项目特点与简要引导】