软件技术架构：

cors

springboot

Hibernate

cors

Echarts

React

Material UI

&

Bootstrap

Sqlalchemy

Flask

Sqlalchemy

sklearn

Frontend

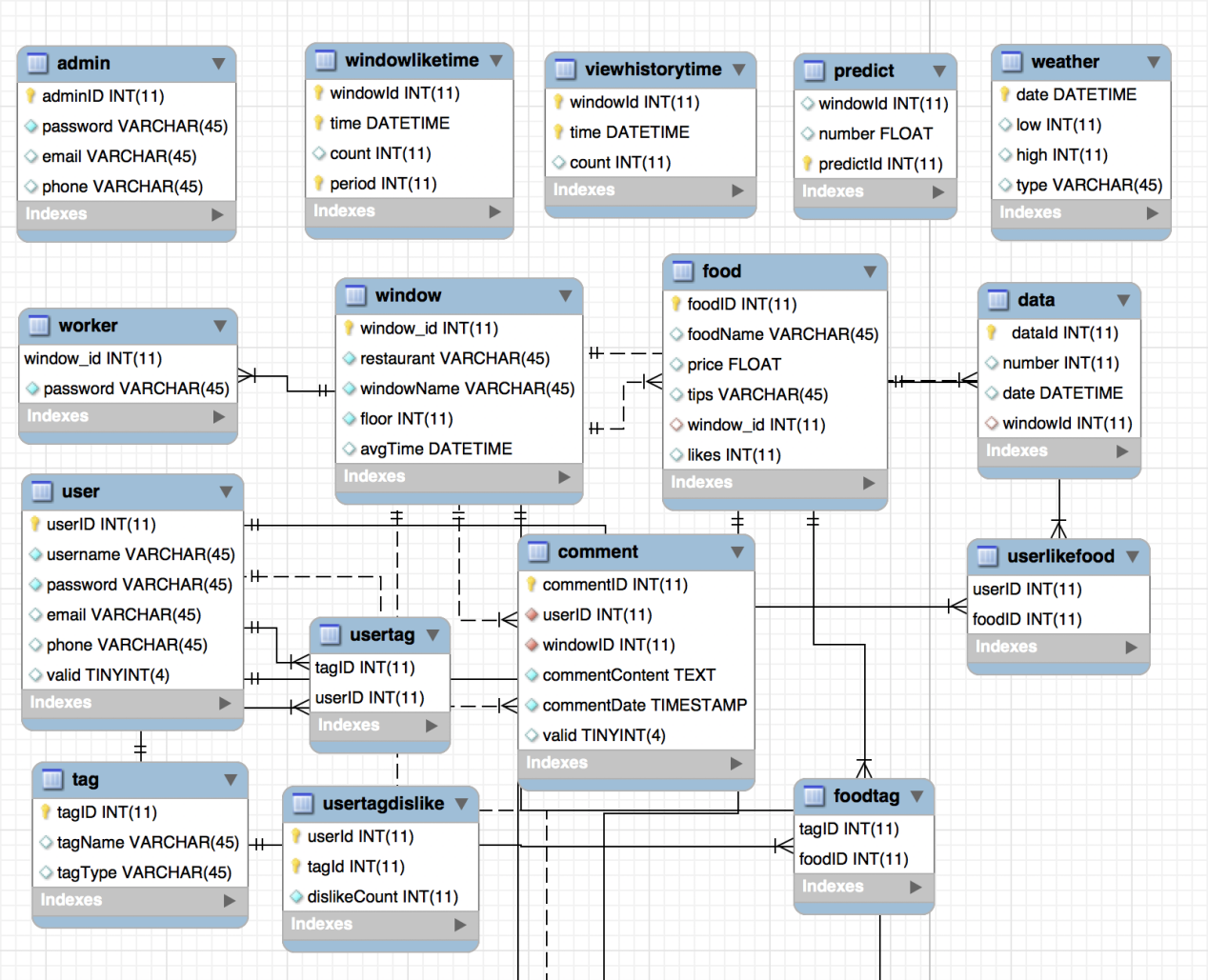
timed task

Backend

MySQL

软件架构

数据库架构：



排队人数模拟、统计、预测与实时可视化设计详情：

* 排队人数模拟：

采用正态分布加随机数并控制上下限的方式生成实时数据。数据间隔为2分钟一个，在软件实际运行时获取数据时即2分钟统计一次餐厅刷卡计数器的计数。

* app.py 用于实时生成模拟数据。
* 排队人数统计：

目前假设早餐时间为7-10点，午餐时间为11-14点，饭餐时间为17-20点， 以2分钟为间隔计数一次，即每个窗口一天会有3 \* 90个数据点， 它们将实时被加入数据库中。

* fetch.py 系统每天在一个时间段，将今日天气情况、气温，今日窗口总点赞数统计，今日窗口总浏览量统计（都为累加值）统一记录在数据库中。
* generate.py 用于生成大量历史排队人数数据。
* generateWeather.py 用于生成大量模拟天气数据。
* generateLike.py 用户生成大量模拟窗口点赞数数据。
* generateHistory.py 用户生成大量模拟窗口浏览数数据。
* 排队人数预测：

对某一时段（早、午、晚）窗口排队人数进行预测。预测算法将前8周对应的该礼拜数、时间段数的90 \* 8个排队人数数据点、天气类型、气温、窗口点赞数、窗口排队人数作为参数，进行岭回归拟合，并考虑到容纳峰值，进行预测。在每个时段结束后对下一时段进行预测。

* ridgePredict.py 用岭回归进行排队人数预测，删除数据库中对于上一时段的预测，并将其计入数据库。
* 时间预测：

给每个窗口设置一个窗口平均出菜时间，结合实时计数算出等待时间。

* 实时可视化
* fetchPredict.py 后端提供3个排队人数数据接口，分别为”data/now”用于获取实时数据，”data/yesterday”用于获取昨日数据，”data/predict”，用于获取预测数据。
* 图表具体实现

1. **窗口图表：**

前端每两分钟向后端请求实时数据刷新一次数据。可视化图表同时提供窗口实时数据的折线以及历史数据的折线。图表分为实时模式与累计模式，历史模式分为昨日模式和预测模式。

1. **总图表：**

提供全部的餐厅、窗口选项，通过手动选择餐厅、窗口获取该窗口的可视化图表。其余交互信息及接口与窗口图表一致。

1. **对比图表：**

可视化图表同时提供两个窗口实时数据的两条对比折线，不提供动态刷新功能。图表分为实时模式和累计模式。

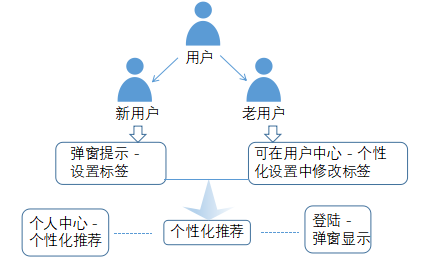
全局（导航栏）菜品搜索设计详情：

对于键入的关键词在实时时段的菜品的名称、提示以及其所有标签中搜索，列出所有符合条件的菜品。对于每条搜索结果，提供菜品的基本信息（名称、价钱、提示、标签等），并分别提供其所属的餐厅、窗口的跳转链接。

用户身份设计详情：

用户身份分为普通用户、管理员与食堂工作人员。为其提供不同的登录入口，仅为普通用户提供注册接口。对存入数据库的用户密码进行加密。普通用户、管理员与食堂工作人员分别有不同的个人中心，并都可进入其余的菜品浏览等页面。

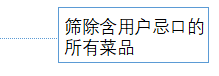
用户个性化推荐及算法实现设计详情：

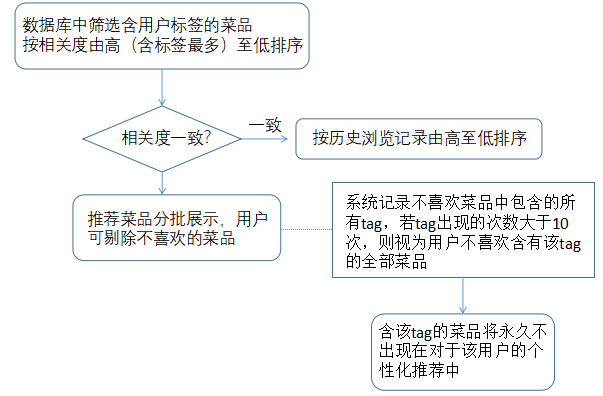


- 模块功能：学校提供的菜品种类丰富，但分布较为零散。个性化推荐旨在以用户设定的标签喜好为主，辅之以菜品的热度等因素有针对性地分批为用户推荐合适的菜品，体现了本软件为用户解决吃饭选择难题这一功能特点。

- 标签设置：设置个人标签是个性化推荐的前提；若没有设置个人标签，则系统不会提供个性化推荐，用户另可通过筛查搜索心仪的菜品。新用户在注册登录之后，会自动弹出设置标签的弹窗，标签按分类分批罗列给用户，用户可自如切换上一批下一批的标签选择，不用担心反悔、错选等问题，用户选择完成后随即弹出今日的菜品推荐；除此之外，新用户还可进入用户中心的个性化设置一栏设置标签并进入用户中心的个性化推荐查看今日推荐；老用户登陆成功之后，会直接弹出今日的菜品推荐，同理也可进入个性化设置修改标签或在个性化推荐中查看推荐。

- 个性化推荐算法：



1. 首先在数据库中筛查含有用户标签的菜品，按相关度由高（含标签最多）至低排序，且排除含有用户忌口的全部菜品。
2. 若两个菜品的相关度一致，则根据该菜品所在窗口的历史浏览记录由高至低排序，浏览记录多的菜品排在前面。
3. 除可查看分批展示的推荐菜品外，用户还可剔除不喜欢的菜品，后台将记录不喜欢菜品所包含的所有标签。
4. 若相同的标签累计产生10次以上，则视为该用户不喜欢任何包含该标签的菜品，在今后的个性化推荐中将不予以推荐。

用户个人信息页面实现设计详情：

- 模块功能：用户可在该界面修改或上传头像，修改昵称、密码、手机及电子邮件。

- 密码修改：用户需正确输入数据库中的原密码，并两次输入满足格式要求的新密码以便确认修改，若原密码输入错误或新密码两次输入不一致都无法修改密码，提高了账户的安全性。

- 电话、昵称的修改注意事项：不允许出现重复的昵称与电话号码，若新电话或新昵称在数据库中出现过，将不予以修改。

用户个性化设置页面实现设计详情：

- 模块功能：用户可在该界面设置个性化标签并自定义新的个性化标签。

- 模块实现：标签按口味、种类、禁忌、食材归类在页面中全部罗列，且将上一次用户设置的标签同时呈现，用户可根据喜好添加或删除专属标签；若用户认为标签有限，也可自定义新的标签，丰富标签种类，同时使得软件设计更显人性化。

我的收藏页面实现设计详情：

- 模块功能：用户可集中查看平时收藏菜品的相关信息、取消对某菜品的收藏。

- 模块实现：该模块显示菜品的地理位子（食堂、窗口信息）、价格、菜名等基本信息，若该菜品今日有售，则会特别标明，方便用户查看喜爱的菜品今日的售出信息。若用户想取消收藏该菜品，点击❤即可，方便快捷。

历史浏览记录实现设计详情：

- 模块功能：用户可查看系统统计的历史浏览记录。

- 模块实现：该模块罗列浏览过的窗口的窗口名称、浏览次数、所属食堂、所属楼层等信息，可以通过浏览次数的升序或降序给浏览记录进行排序。除此之外，还设置删除浏览记录功能，可删除一条或多条历史浏览记录。

食堂工作人员界面设计详情：

- 模块实现：考虑到食堂工作人员普遍年龄偏大、文化水平较低等因素，故食堂工作人员界面设计十分简约，工作人员仅需每天在规定的时间范围内上传一张当日当时的菜品即可。