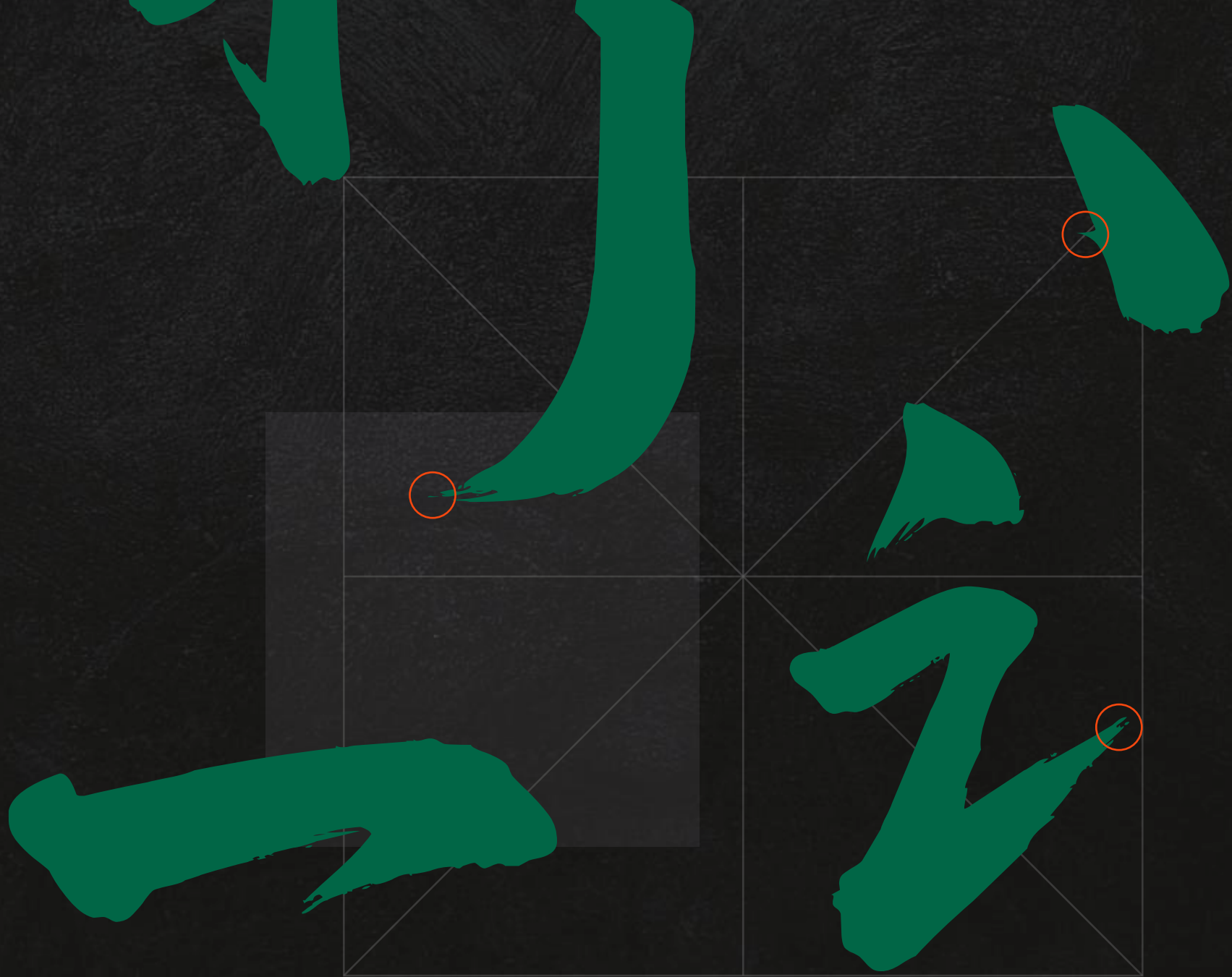


· 八大菜系 · 08小组 · 倾情呈现

老八食谱

08小组—不被优化

赵康明 张弛 傅桐 彭钰钊



/ CATALOGUE

目录

① 团队介绍

② 项目概述

③ 需求分析

④ 技术实现

⑤ 项目开发过程

⑥ 项目演示

⑦ 实现成果

⑧ 技术难点

⑨ 未来展望

⑩ 项目总结



01

团队介绍

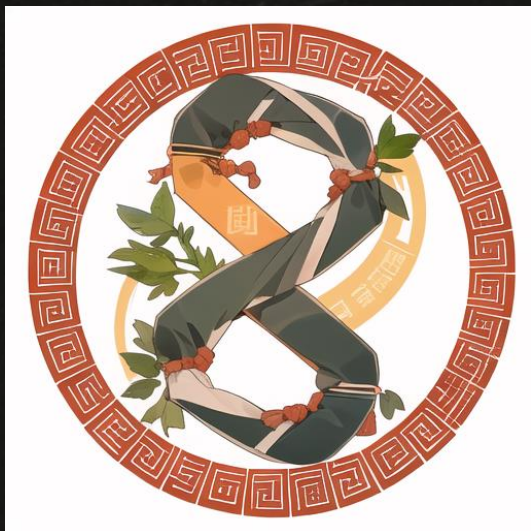
团队名称：不被优化

团队宣言：

落地为兄弟，何必骨肉亲！得欢当作乐，斗酒聚比邻。

团队霸气 logo:

寓意：第八组->八大菜系->老八食谱



团队成员 帅气合影（从左至右）：

开发经理——傅桐

质量保障工程师——彭钰钊

项目经理——赵康明

项目架构师——张弛



成员分工

- 数据爬取与预处理：小组成员均分数据爬取与预处理工作
- 知识图谱可视化界面设计与实现：傅桐、彭钰钊
- 智能问答系统设计与实现：赵康明、张弛
- 演示文稿及项目说明文档撰写：小组成员依照前期分工撰写自己所实现的模块



02

项目概述

选题：中华八大菜系知识图谱

项目背景与目标

南开大学有着光荣的爱国传统。这份爱也体现在对祖国的传统文化的热爱上。

中华文化源远流长，博大精深。在绵延不绝的文化长河中，丰富多彩，令人垂涎的各种中华美食必然是其中的一颗明珠。

为了结合计算机技术，让我们大家都非常喜爱的美食文化生动的展现出来，我们选择了这个课题，旨在通过我们收集和处理数据，为大家清晰的呈现中华美食的丰富多彩，加深大家对于中华美食的理解和喜爱，并且将其相关知识深入脑中。

项目意义

通过传统文化与现代技术的结合，可以使得庞大的菜系菜品文化“跃然显示器上”，为培育文化自信加油助力。

同时诱人的菜品也许能激发大家的食欲，起到鼓励大家好好吃饭、照顾身体的作用。

项目名称：老八食谱

项目内容

项目在呈现时分为两个部分

第一部分是知识图谱可视化呈现。对我们搜集到并整理好的数据进行可视化展示，旨在通过清晰的展现各大菜系之间的菜品和食材的联系，并且辅以精美的网页界面和音乐带给用户以视听刺激来加深用户对于中华传统美食的理解与印象。

第二部分是基于上述数据库的数据，进行智能化的问答系统的构建。目的是通过与系统进行有趣的智能的问答，在获得经由系统筛选挖掘的知识的同时，加深对于中华传统美食的记忆与认知。

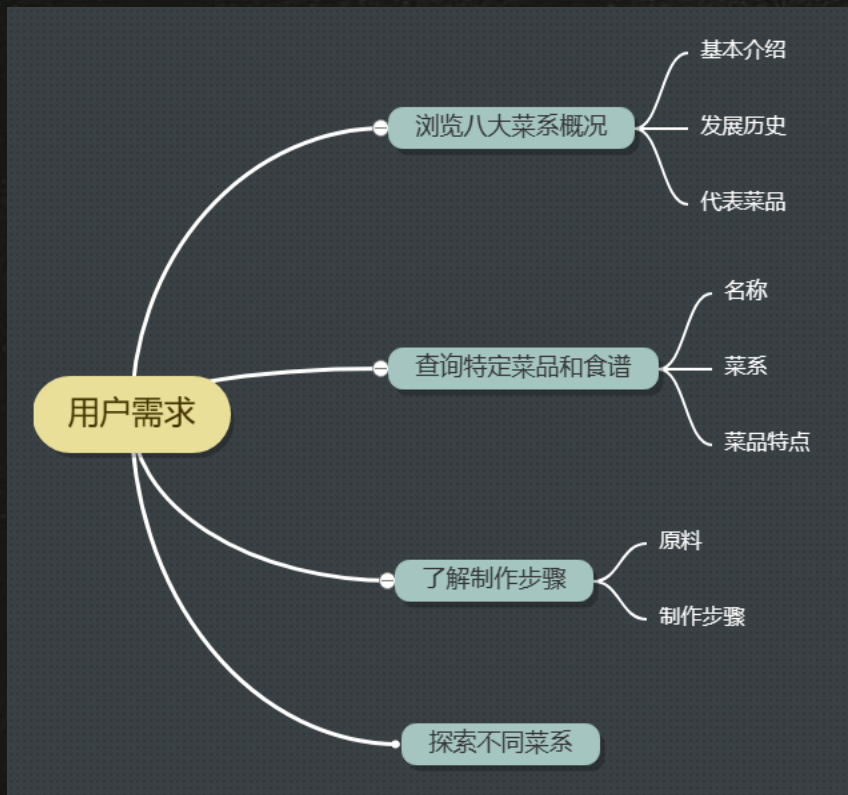


03

需求分析

用户需求

- 通过模拟用户需求（站在用户的角度思考项目功能）我们考虑到如下用户需求



- 对如图的用户需求进行分析我们可以得到如下功能需求：

食谱搜索功能
食谱详情展示
分类浏览功能
菜系概况展示
用户友好界面
知识图谱可视化界面
智能问答系统



04

技术实现

04.初始数据获取与处理

技术选型

这块我们采用的是urllib、requests、bs4、lxml、requests-html、selenium多种库的动静结合的花样繁多的爬取方式，以适应网站不同类型的文本、图像等数据的爬取需求。

根据从网页获取的原数据（已经经过预先的格式设计和处理处理）采用pandas等工具对数据结构进行处理以适应后期导入数据库的需求,包括将csv数据处理成dump文件（neo4j）与nt文件（fuseki）来适配不同的数据库服务。

04.1 可视化网页

技术选型

网页框架采用的是 **Django**

- Django提供了许多内置的功能和模块，可以极大地简化开发过程和提高开发效率。而且因为是python，所以在入门成本上相对便宜。

编辑器采用的是记事本

- 记事本的好处是轻便，而且电脑自带。坏处是它是记事本

可视化数据工具采用的是D3

- 它非常灵活，提供了多组api，支持多种可视化的图像类型，具有极强的兼容性。最重要的是有非常丰富的社区支持，即使是新人也可以在并不精通的情况下速成页面设计。

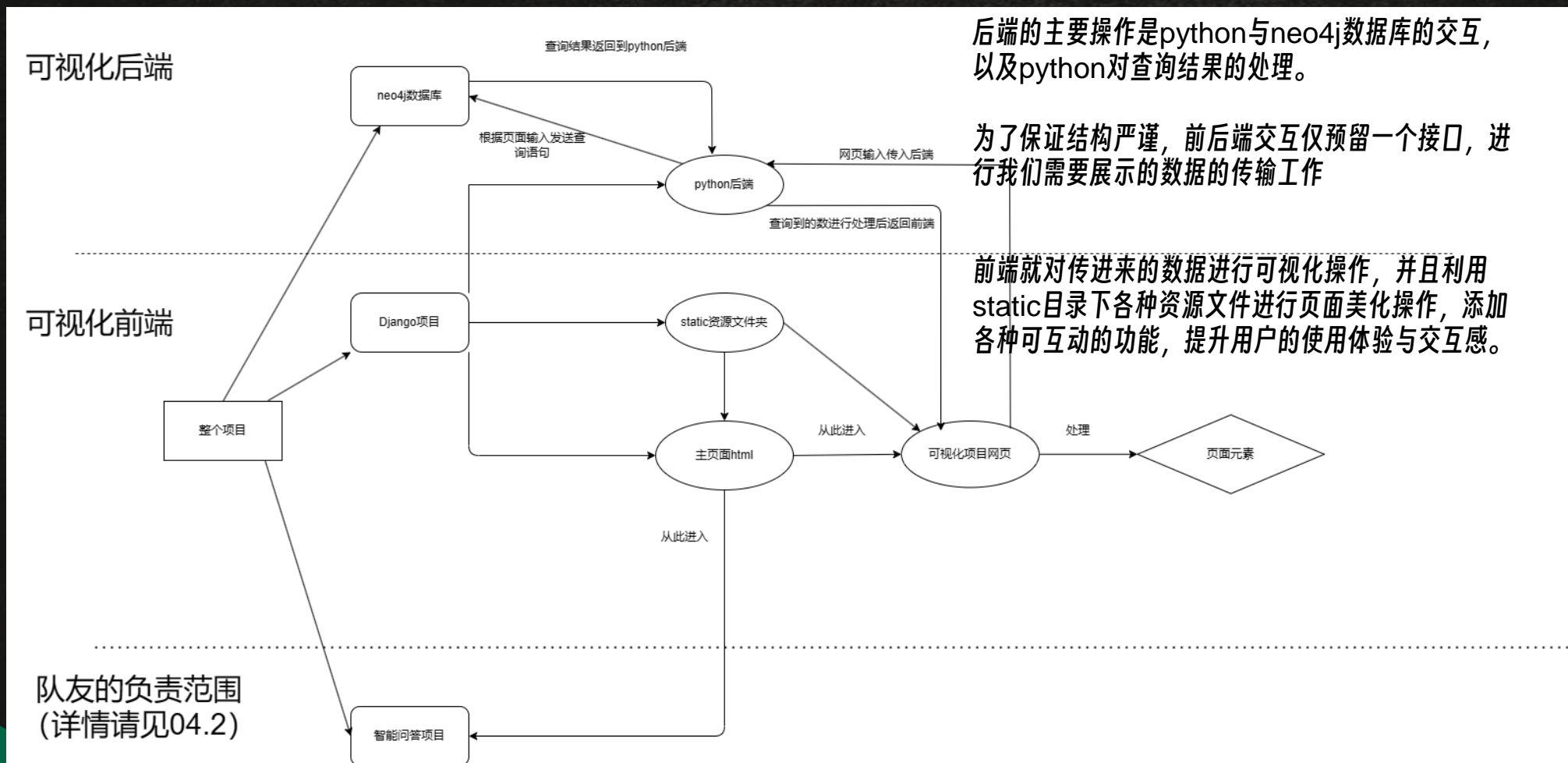
数据库采用的是Neo4j

- 在处理这种类型的数据时，图数据库非常的清晰明了。在与python连接时使用py2neo库。

04.1 可视化网页

系统架构

架构的大体示意图如下：



04.2 智能问答系统

技术选型

- 前端：采用Flask作为网页框架和可视化工具，进行智能问答系统页面的设计，并利用css技术来规范样式、美化网页（加动效、bgm等）

好处：Flask作为一个轻量级的Python Web开发工具，其能够帮助开发者快速进行开发（主要是好学好用）

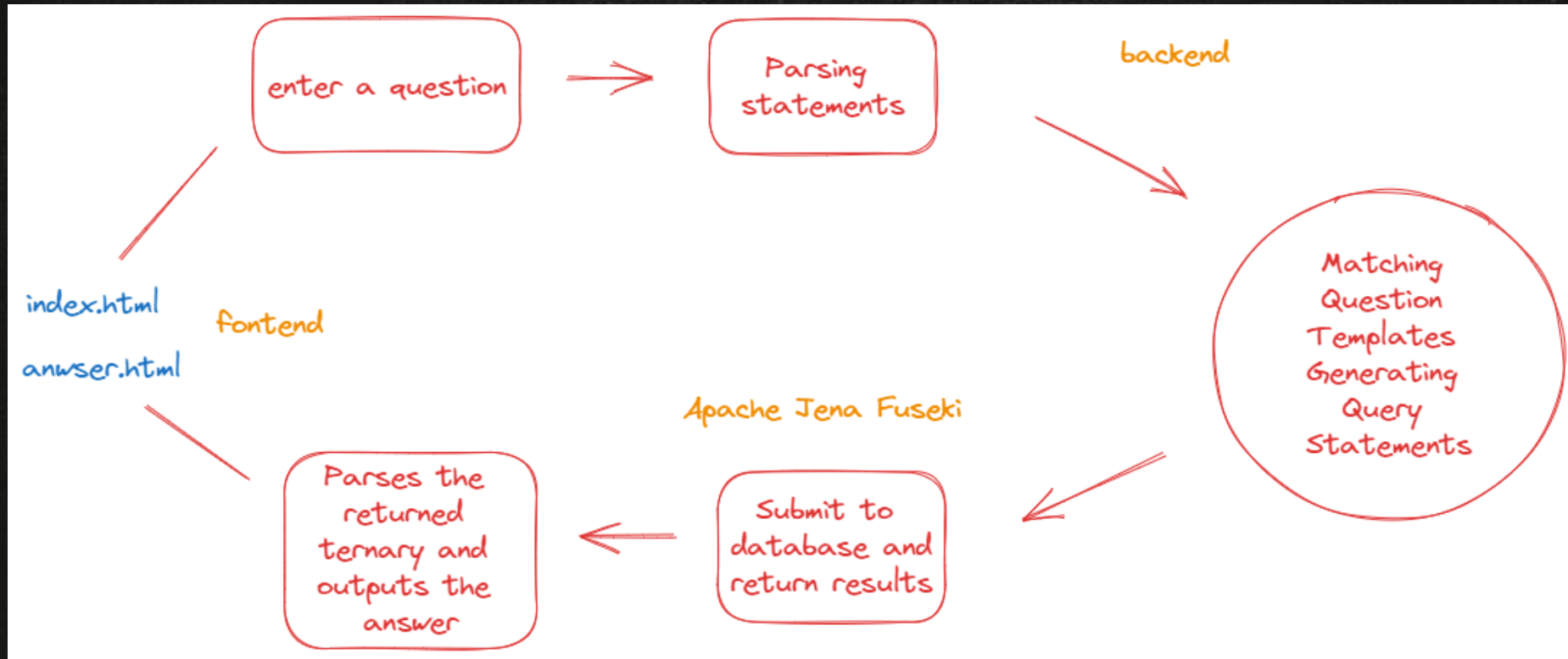
- 后端：数据查询框架采用基于Java的Apache Jena Fuseki 服务器（利用SPARQL查询语句的前缀特性加速查询）

开发环境即选择完美适配前后端的VSCode；语义解析采用的是jieba分词（添加外部词典）+refo 正则匹配（预设提问模板、属性词等）

04.2 智能问答系统

系统架构

智能问答系统的大致 workflow 如下:





05

项目开发过程

05.1 可视化网页

设计

首先是页面结构的设计。我们想要做的是非常有动感的页面，以关系图谱为中心，通过饼状图和柱状图等展现数据分析结果，做到较为完备的数据展现。然后通过对于展现的数据图添加动态效果提高用户的互动感，再辅以背景和页面的各种小元素等来提高用户的视觉体验。而且总的页面所有的元素都时时刻刻在扣紧传统美食的主题。

开发过程

实现

在代码编写上，采用前后端并进的，深度细粒度并行化的联合作战方式。我们首先先把目光放在大体的主要功能上。在这里分两路进行前端的数据接收和最基本的结点的构建以及后端的数据处理和发送。在基本实现结点的展现以后，一同进行页面的设计和细节功能的完善。期间需要细心地进行版本控制与岔路的融合。最终工作效率得到了极大的提升。在实现设计时，也不只是完完全全按照设计来，一些过于理想的设计可能导致程序性能下降。这是后就不得不进行性能和效果的权衡，或者是寻找引入类似缓存、预加载、减小资源文件的大小之类的操作。

05.1 可视化网页

我们的设计总体上得到了很好的完成。基本实现了一个极具动感的，充满交互性的，拥有了相对完备、广度深度兼具的数据展现功能，并且使得网页“声情并茂”--拥有精美的壁纸、细节以及动听的、切合文化主题的背景音乐。(●~▽~●)

完成情况

05.2 智能问答系统

首先自然也是本着**并行处理提高效率**的原则，前端与后端同时开发，并采取一致的代码规范和命名，以保证项目对接时能够顺便实现前后端映射。

从后端来说，从需求出发，要实现的是输入提问语句然后给出相应回答的功能模块，很容易想到要用到NLP的相关方法，而且由于设想中并不需要实现多么智能的人机交互，仅需对预设的几种问题模板作出正确的识别和响应即可，因此没有考虑深度学习等需要海量数据训练模型的方法，经过一番查阅资料，最终敲定了**正则表达式模板匹配**这种相对而言较为轻量化的方法。

设置后好外部词典和问题模板后，经测试分词可以正常工作，接着便考虑该采取怎样的数据查询框架，本着多多益善的探索原则又对**基于 fuseki 的 SPARQL 查询**进行了学习并将其运用到了KBQA中作为**云端数据的来源**（可是又得搞大量的RDF数据，十分痛苦）。

05.2 智能问答系统

三元组数据处理完毕并上传到服务器后，接着便是测试根据问题模板解析提问语句、生成SPARQL查询语句是否成功，反复调试后也能正常运行，然后事情就开朗了起来，仅需要把对应的查询语句提交到服务器中执行返回结果，解析结果三元组，分类讨论答案的情况便可以输出了，然后就是测试样例反复轰炸，查缺补漏并完善反馈错误机制，到此，后端的实现就基本告一段落了。

前端的大致开发流程与知识图谱可视化模块类似，就不赘述了。

两端都开发完毕后，只需在相应要执行的py文件中进行对接即可，这样就完成了前后端的融合，成功建立映射。



06

项目演示

项目演示

* 启动 **fuseki** 框架

```
`java -jar fuseki-server.jar`
```

* 启动 **Django** 项目

```
`python manage.py runserver 0.0.0.0:8080`
```

* 项目主页面网址

```
`http://127.0.0.1:8080/recipe_knowledge_graph/`
```

07

实现成果

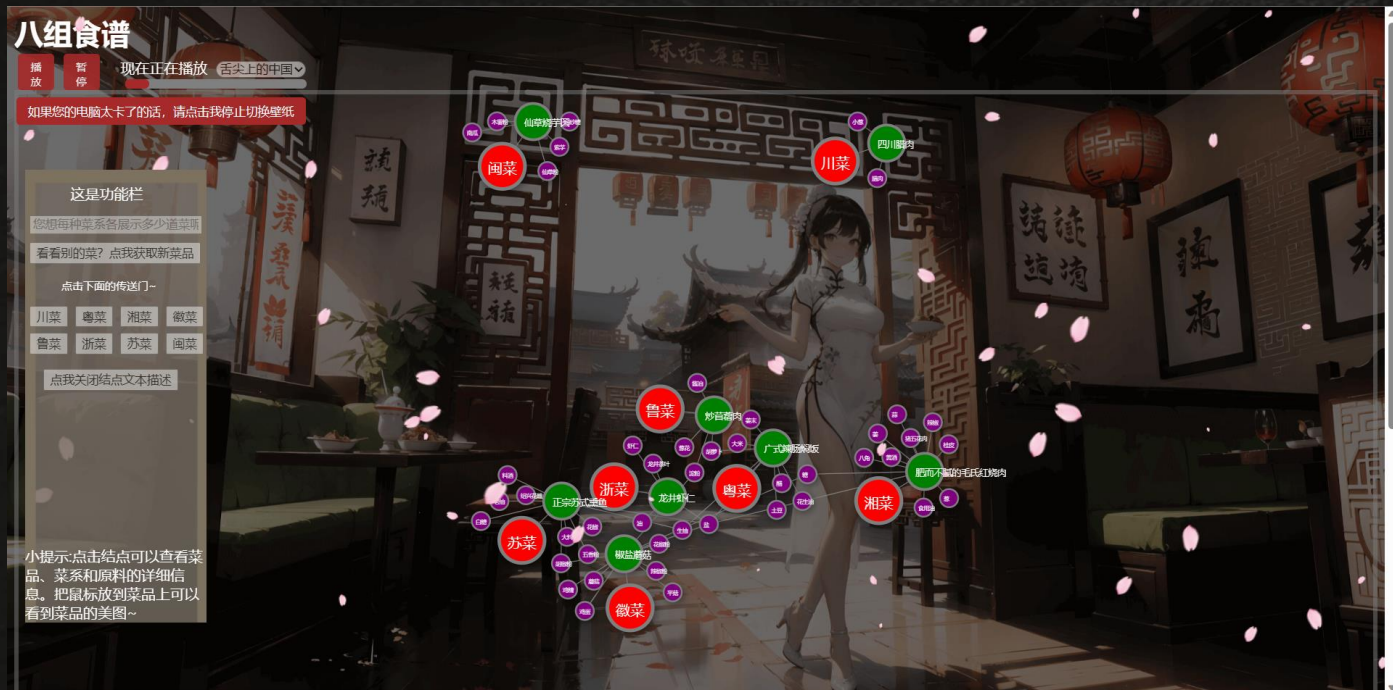


成果展示



主页面+到两个功能模块
的跳转+播放背景视频

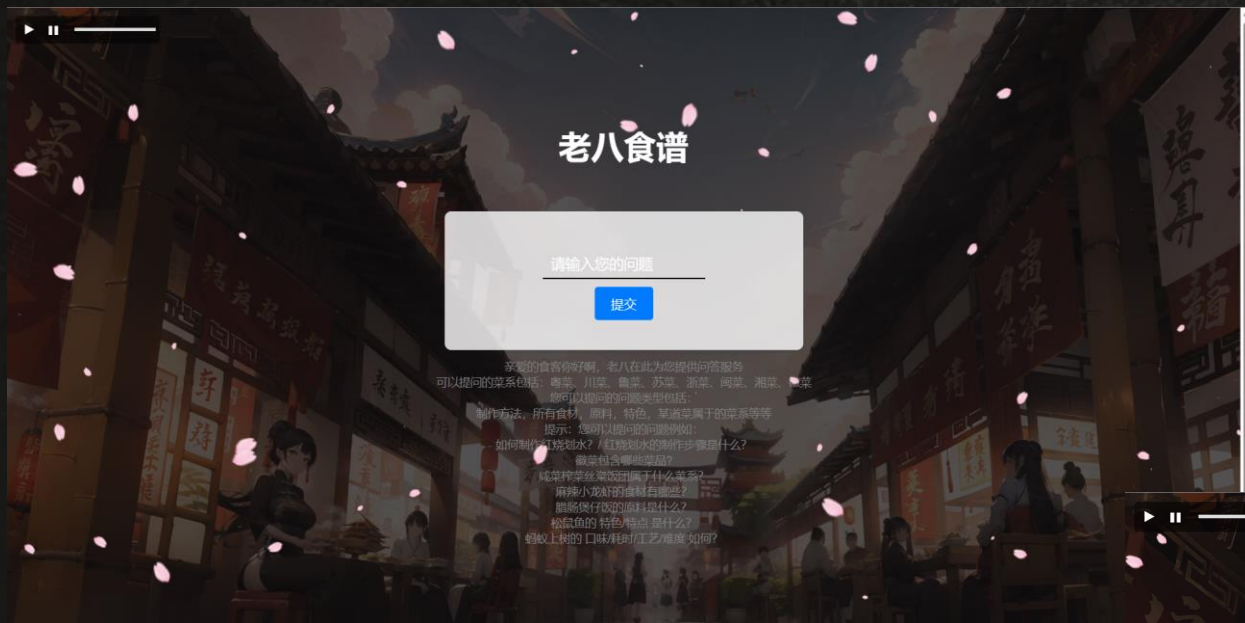
成果展示



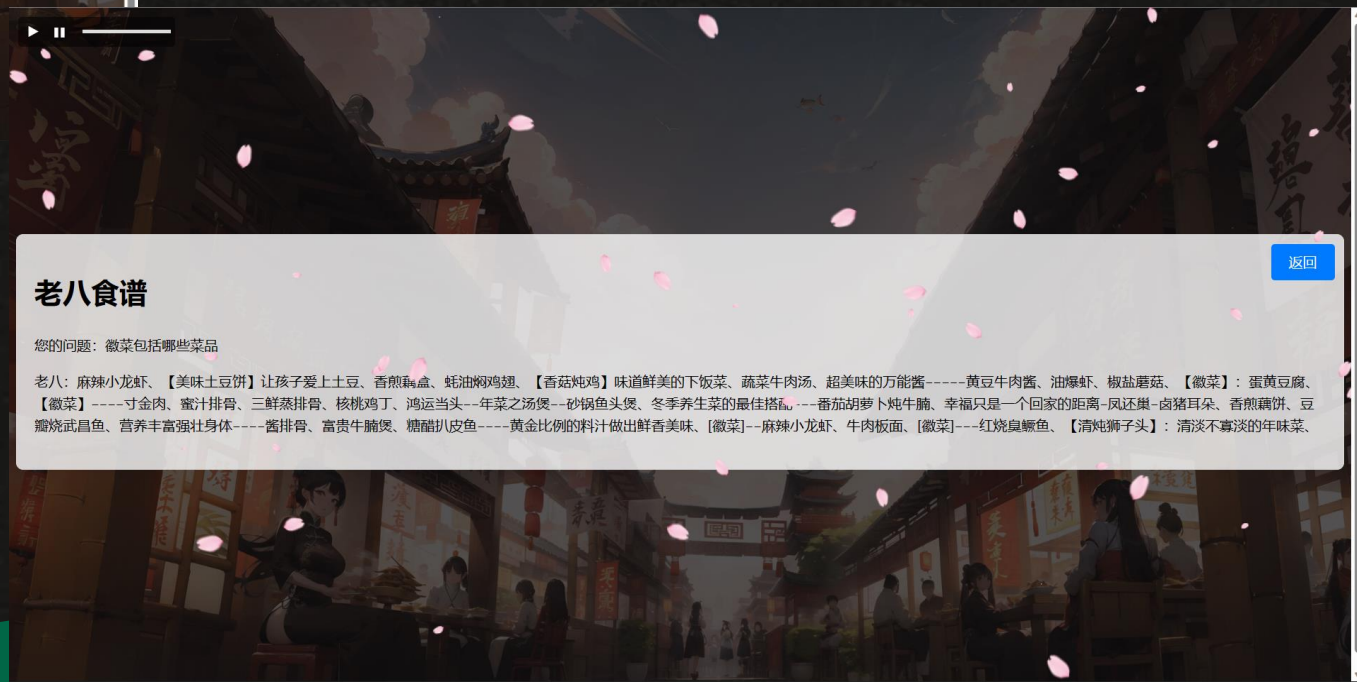
知识图谱可视化模块网页+bgm自动播放+背景图片渐变+原料占比柱形图、扇形图+页面动效+细节tips等等



成果展示



智能问答模块页面+
丰富的提问方式+
回答页面跳转+
视觉动效





08

技术难点

技术难点

- 后端：

- ♥知识图谱可视化：采取了Django作为网页框架

- ♥智能问答系统：采用了Flask作为网页框架，采用Apache Jena Fuseki 处理RDF数据和执行SPARQL 查询，难点在于如何匹配问题和所设置的问题模板，给出正确的答复。

总的来说两部分内容所用的都不太一样，体现了技术种类的多样性。最后将其合并为一个项目，体现其完整性。

- 前端：

- ♥知识图谱可视化：neo4j作为图数据库，d3实现数据可视化，难点主要在于如何将后端的数据发送至前端，并且实现灵活展示。

- ♥智能问答系统：在静态资源加载时候可能需要注意路径问题，美化设计的过程中注意格式上的统一



09

未来展望

未来展望

为了更全面的覆盖到多种菜系的更多菜品，可以考虑扩充数据库的容量，例如增加其他菜系、每个菜系扩充更多菜品等等，并且结合多种数据源**确保数据完整性与准确性**

本项目实现的智能问答是基于预设正则模板的匹配，只能回答较为局限的问题种类，未来可以使用图神经网络（GNN）等表示学习方法来学习更有效的知识图谱表示，以**提高问题与知识图谱之间的匹配效果**。

引入推理与逻辑推断：

- 在知识图谱中添加逻辑规则，以支持逻辑推理，从而更好地回答复杂问题。
- 利用图算法和推理技术，进行跨实体和关系的多跳推理。

交互和用户体验：

- 优化KBQA系统的交互界面和用户体验，使用户可以更自然、更方便地与系统交互。
- 支持自然语言问题的多样化表达和更加灵活的查询方式。



10

项目总结

项目总结

本次“老八食谱”项目主要完成了 **中式菜系知识图谱可视化** 与 **基于前者的智能问答系统** 两部分功能模块的实现，并针对解析提问语句效率、后端到前端的数据映射速度、页面动效与美观度做出了优化，来保证项目整体的可用性
但仍存在不少机制上、数据上等等各方面的不足，有充足日后优化的余地。
同时希望能够在智能问答系统中结合可视化部分使用到的D3功能，将查询结果可视化。



谢谢