系统设计报告

1 引言

1.1 编写目的

本系统设计报告以《项目计划书》和《需求规格说明书》为基础，具体讲述系统的总体架构，各个功能的实现方式以及数据库设计方法，明确了各个模块的外部接口、内部接口以及用户接口，为软件系统的开发提供指导，为软件系统的维护提供参照。

预期读者：

- 项目经理
- 系统分析人员
- 系统设计人员
- 系统开发人员
- 系统测试人员
- 系统质量分析员
- 系统维护人员

1.2 软件项目背景

本项目是浙江大学 2024 学年《软件工程管理》课程的课程项目，目标是实现一个就医问诊引擎。

在这个信息爆炸的时代，获取精准、可靠的医疗信息变得尤为重要，但难免仍然存在医疗水平落后、医疗资源受限、就医成本昂贵的地区。为了更好地服务广大群众的便捷看病，就医问诊药学垂直领域搜索引擎，提供了一个专业、高效地信息获取平台。借助于先进的 AI 技术，能够快速解析复杂的不适症状以及医疗数据，推荐最适合患者的药物和治疗方案，为用户提供精准、个性化的健康指导，助力医疗行业的创新发展。就像 2024 年诺贝尔化学奖得主通过 AI 预测蛋白质结构一样，本搜索引擎也将运用类似的智能算法，以获得最前沿、最科学的医疗建议，做出更明智的医疗决策。

1.3 定义

1.3.1 HTML

HTML 即超文本标记语言。HTML 是由 Web 的发明者 Tim Berners-Lee 和同事 Daniel W. Connolly 于 1990 年创立的一种标记语言，它是标准通用化标记语言 SGML 的应用。用 HTML 编写的超文本文档称为 HTML 文档，它能独立于各种操作系统平台（如 UNIX，Windows 等）。使用 HTML，将所需要表达的信息按某种规则写成 HTML 文件，通过专用的浏览器来识别，并将这些 HTML 文件翻译成可以识别的信息，即现在所见到的网页。

1.3.2 CSS

CSS 即层叠样式表，是一种用来表现 HTML 等文件样式的计算机语言，在网络中能够对网页中元素位置的排版进行像素级精确控制。

1.3.3 JavaScript

JavaScript（简称 JS）是一种具有函数优先的轻量级，解释型或即时编译型的编程语言。虽然它是作为开发 Web 页面的脚本语言而出名，但是它也被用到了很多非浏览器环境中，JavaScript 基于原型编程、多范式的动态脚本语言，并且支持面向对象、命令式、声明式、函数式编程范式。

1.3.4 Vue

Vue 是一个用于创建用户界面的开源框架，也是一个创建单页应用的 Web 应用框架，一套用于构建用户界面的渐进式框架。

1.3.5 React

React 是一个用于构建用户界面的开源 JS 库，由 Facebook 开发并维护，于 2013 年首次发布，并且已经成为前端开发中最流行的库之一。

1.3.6 Django

Django 是一个高级 Python Web 框架，鼓励快速开发和干净、实用的设计。它由 Adrian Holovaty 和 Simon Willison 于 2005 年创建。

1.3.7 DRF

DRF（Django REST framework）是一个强大的 Web API 工具包，与 Django 框架一起工作，用于构建 RESTful Web APIs。

1.4 参考文献与资料

- 《软件工程开发国家标准》
- 《软件工程项目开发文档范例》
- [G07] “就医问诊引擎”项目计划书
- [G07] “就医问诊引擎”需求规格说明书

2 需求规定

2.1 用户需求规定

2.1.1 信息检索

用户可以输入关键字进行搜索，系统将基于关键词进行医疗信息的模糊检索，包括医疗用品、常见症状等。

2.1.2 AI 解析

根据用户提供的对话内容进行 AI 解析，进行症状和医疗数据的分析。用户可以通过新建对话来创建一个新话题，从而实现多个主题的分析，也可以通过侧边栏查询已存在的对话。

2.1.3 推荐系统

根据用户查询和 AI 历史记录提供的数据进行针对性的个性化推荐，提供日常健康的维护建议、精准的药物和治疗方案等。

2.2 其他需求规定

2.2.1 性能需求

- 系统应保证运行稳定，避免出现崩溃
- 主流浏览器均能正常访问本系统
- 系统应能保证至少 100 人的并发访问
- 当用户登录以及进行任何操作时，系统应该能及时进行反应，反应的时间在 1s 以内
- 系统应该能及时检测出各种非正常情况，如与设备的通信中断，无法连接数据库服务器等情况，避免用户长时间等待在某个界面。一般情况下应在 1s 内加载完毕，高峰期应在 7s 内加载完毕
- 系统保证在半个月内不超过一次维护与重启

2.2.2 输入需求

- 用户输入数据时，应对数据输入进行数据有效性和安全性检查
- 用户在进行搜索时，应对数据的有效性和合法性进行检查
- 系统应通过程序控制出错几率，减少系统因用户人为的错误引起的破坏，开发者应当尽量周全地考虑到各种可能发生地问题，使出错的可能降至最小

2.2.3 数据传输与并发需求

- 用户输入账号密码点击登录后，对登录的响应时间不能超过 1s
- 系统能支持多个用户同时搜索医疗信息以及查看详细信息、同时进行 AI 查询、并发使用且保证性能不受影响

- 在网页中，系统生成的所有 Web 页面可以在不超过 5s 的时间内可以全部下载下来

2.2.4 数据管理需求

系统既要与其他系统有接口，又必须保证本系统的独立性与完整性。即应防止未经授权的各类人员对本系统进行设置和修改或进行有关统计。

系统服务器软件必须提供可靠的数据备份和恢复手段，在服务器软件或硬件出现严重故障时，能够根据备份的数据和账户信息等必要的配套信息，迅速彻底地恢复正常运行环境。

系统的用户信息管理相关模块，决定了其他众多系统的账户安全性，必须保证统计数据准确、安全，用户信息应当提供完整的备份及恢复措施。账户信息必须提供完备手段由用户自定义和备份保存，软件开发者不得在系统中预留任何特殊账户和密码。

除此之外，系统应具备加密登录、数据加密传输等安全方面的故障，保证数据在不用系统间传输过程中的保密性与安全性。

以下为具体细则：

- 当系统崩溃后，系统应能在 24 小时内恢复运行
- 数据库可支持表得最大行数达到 600 行
- 本系统用于日志等记录的数据增长约为 20MB/月，具体增长速度由用户的使用频率及所发生业务的数据量决定
- 当出现重大事故造成数据丢失后，系统应能在 48 小时内恢复数据
- 系统管理员每两个月应至少维护备份一次数据
- 系统服务器应具备至少 20GB 的存储空间

2.2.5 权限与安全需求

在我们的系统中，对于安全与权限进行了如下设计：

- 所有涉及功能信息或个人信息的网络事务，都应进行加密操作
- 用户无法非法修改数据库
- 系统应该能够记录系统运行时所发生的所有错误，包括本机错误和网络错误，以便于查找错误的原因。系统日志应同时记录下用户的关键性操作信息
- 当流量过大时，优先限制游客流量防止恶意访问
- 系统对可能发生严重后果的操作要有补救措施，通过补救措施用户可以回到原来的正确状态。同时对可能造成等待时间较长的操作应该提供取消功能
- 对错误操作支持可逆性处理，如取消系列操作。在输入有效性字符之前应该阻止用户进行只有输入之后才可进行的操作

2.2.6 防护性需求

- 文件格式错误时，系统提出警告，保持数据库数据不变
- 数据库误删除时，可以使用撤销删除修复
- 系统应该保护未开放下载权限的资料不被下载
- 重复操作导致卡死时，系统提出警告
- 系统应该提供验证码防止恶意使用（如爬虫等）
- 系统应该及时信息备份防止病毒攻击
- 系统应该能检测到恶意操作，提出警告并在一段时间内不允许操作

- 访问无权限时，系统发出提示并禁止用户访问

2.2.7 其他需求

- 软件对用户的使用应该有较为清晰的引导
- 软件对用户的误操作或不合法操作进行检查，并给出提示信息

3 总体设计

3.1 功能设计

功能模块	功能
前端子系统	查询搜索内容
	搜索结果与详情展示
	搜索结果相关信息展示
	对话内容输入
	新建新对话
	查询对话历史
后端子系统	数据导入
	前端搜索接口
	前端 AI 对话接口
	AI 接口
爬虫子系统	信息爬取
	信息筛选
	自动爬取

3.2 用户类型及用户特征

- **普通用户**：需要获取医疗信息和建议，可能缺乏专业知识。
- **医疗专业人士**：需要快速获取最新医疗数据和研究成果，以辅助临床决策。
- **医疗管理者**：需要数据分析支持，以优化医疗资源分配和管理。

3.3 运行环境

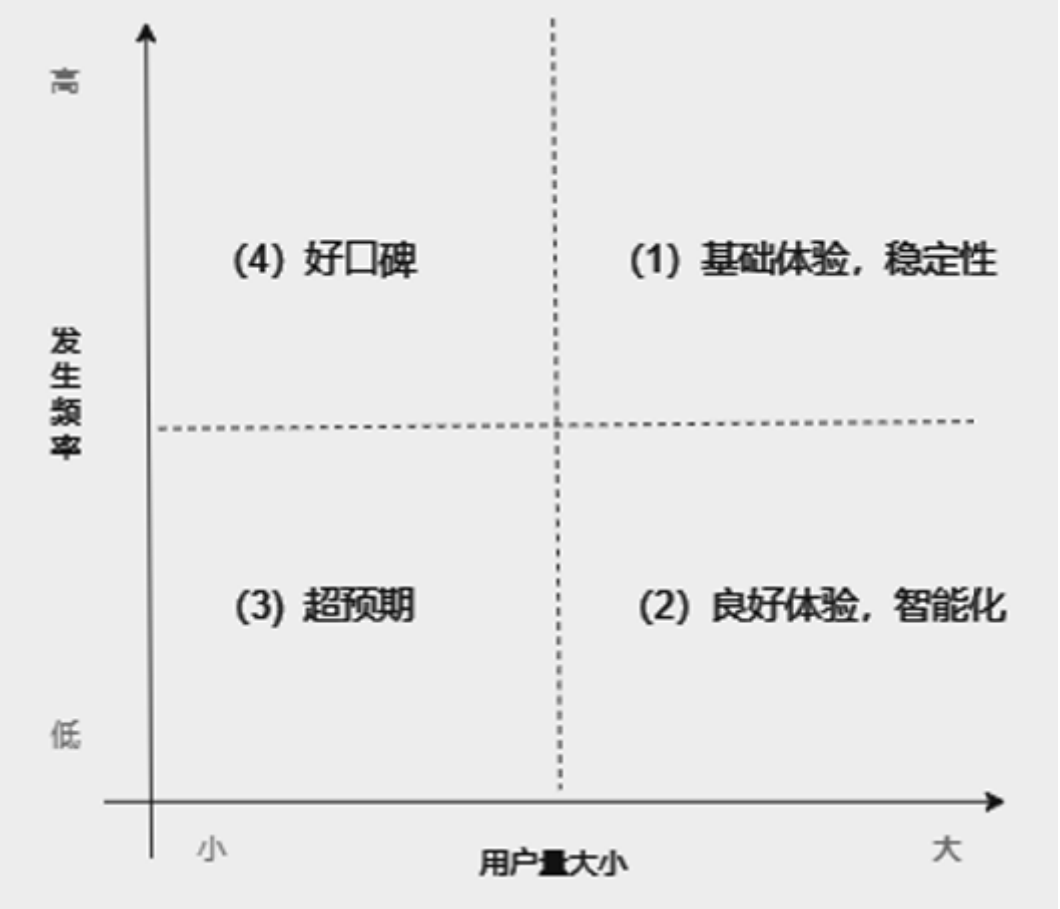
本网站保证至少 100 名网友同时取得服务的需求，包括数据存储能力和网络吞吐能力，保证账户一定的安全性。

- **软件运行环境**：
  - **操作系统**：Windows 7 及以上
  - **网站服务器**：Nginx 1.15.8
  - **数据库服务器类型**：MySQL / PostgreSQL
  - **浏览器**：Chrome / Edge
- **硬件运行环境**：
  - **操作系统**：
    - CPU：CORE i3 及以上
    - 内存：2G 以上

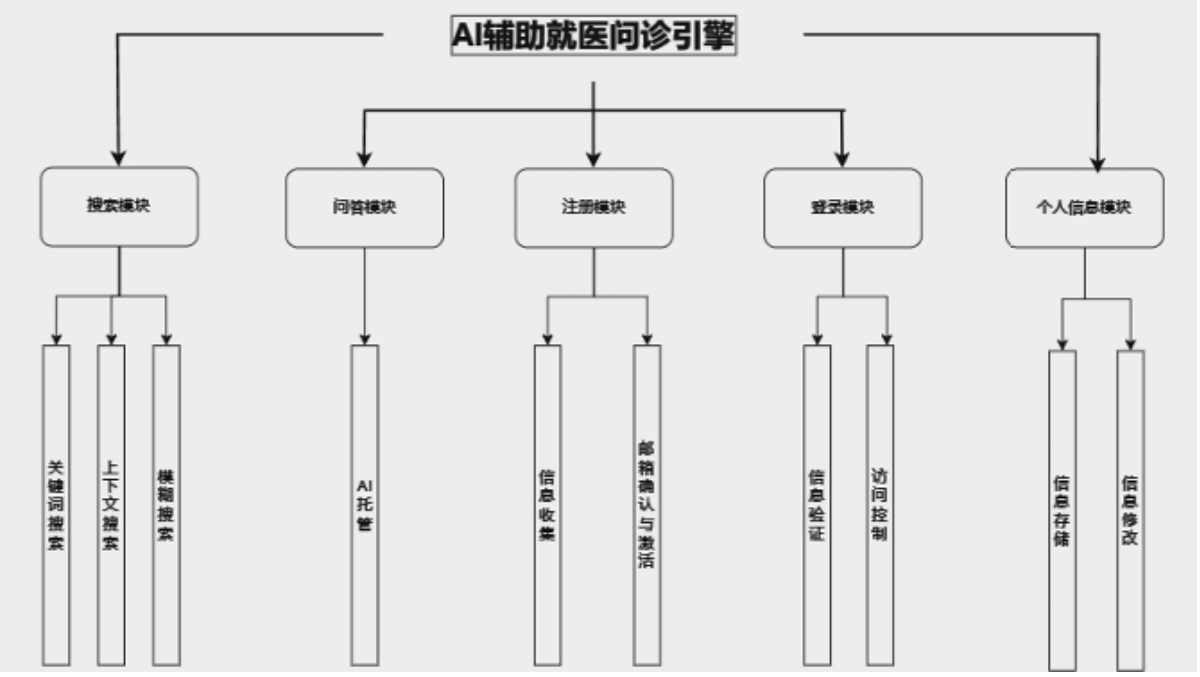
- 硬盘：500G以上
- **数据库服务器**：
  - 内存：512M 及以上
  - 硬盘：50G 及以上

3.4 结构

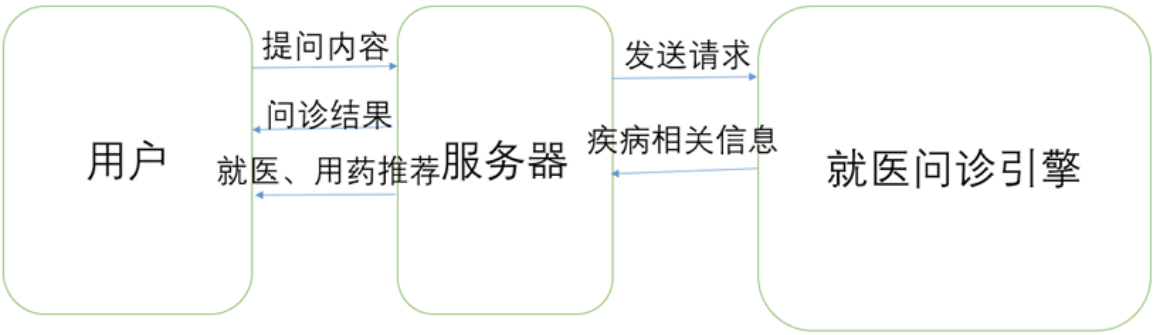
3.4.1 用户需求分析图



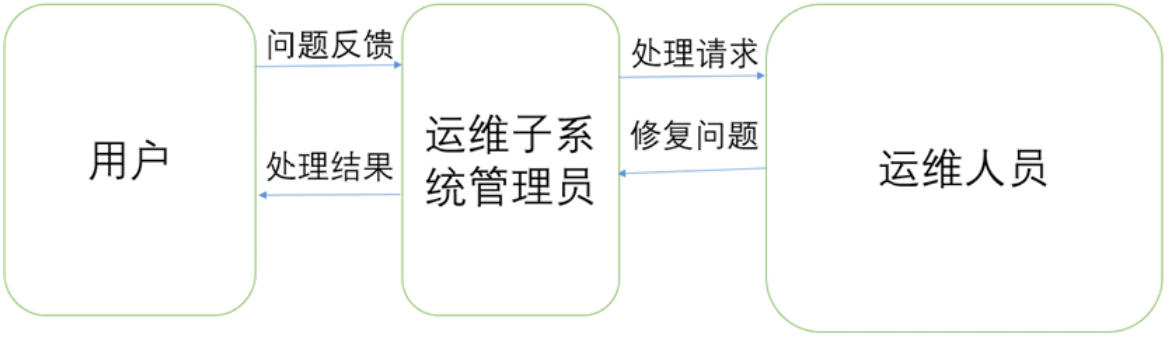
3.4.2 系统模块架构图



3.4.3 数据流图

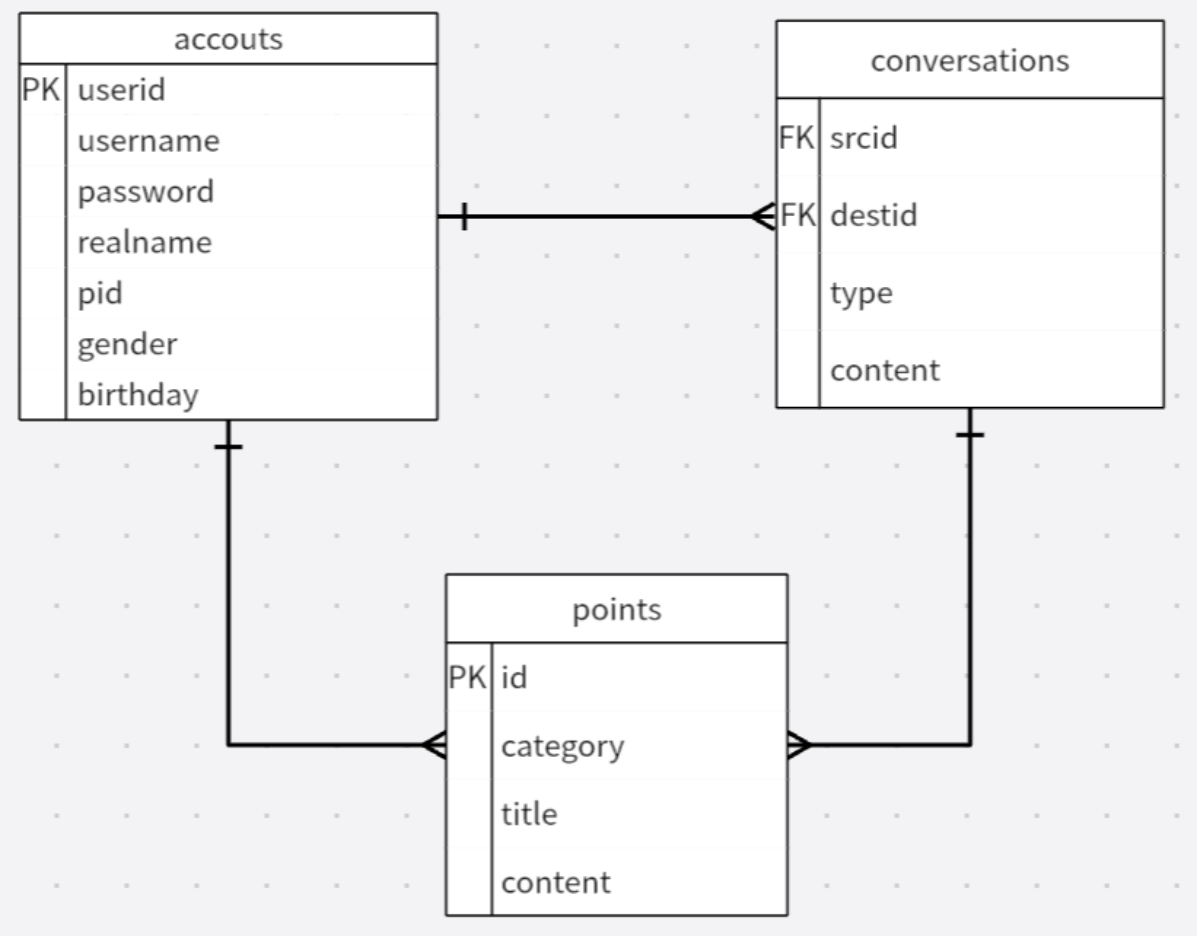


问诊引擎子系统数据流图

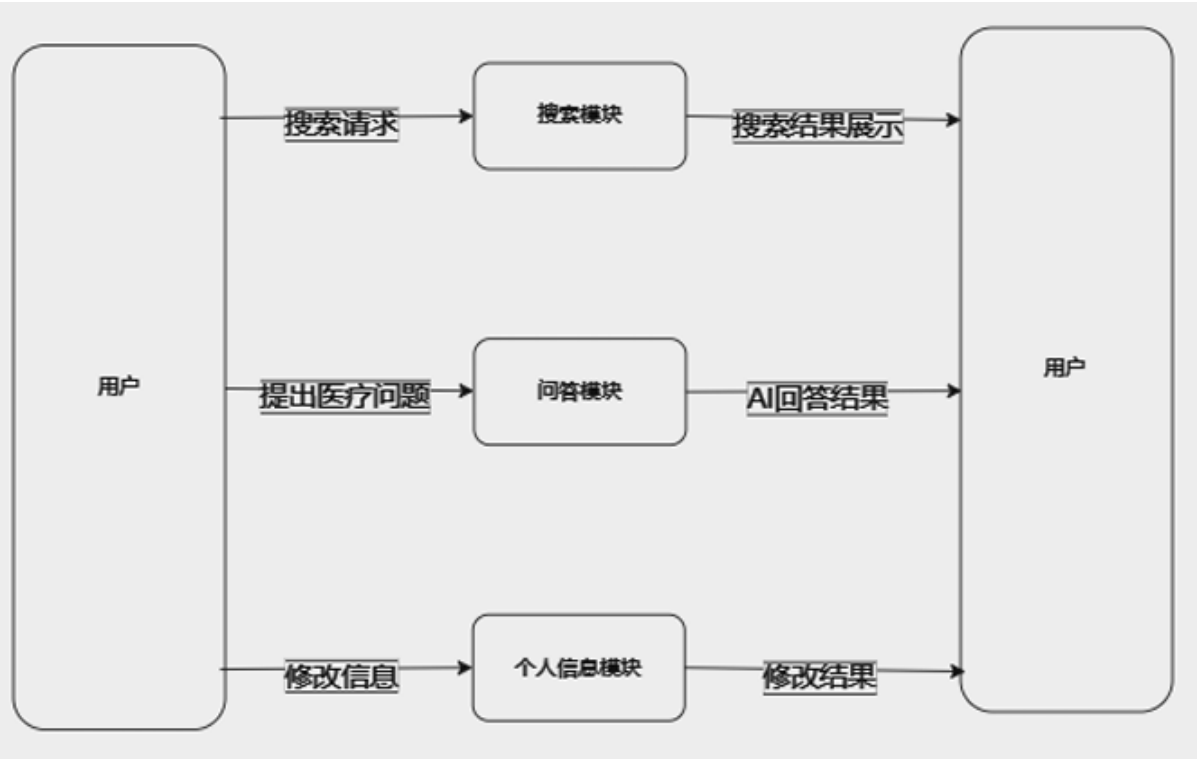


网站维护子系统数据流图

3.4.4 ER图



3.4.5 上下文图



3.4.6 数据字典

编号	数据项名称	类型	取值范围	描述
E1	查询关键词	字符串	长度范围：(0, 256)	用户输入的查询关键词
E2	科目类型	枚举	内科/外科/皮肤科等	用户选择的就诊科目
E3	症状描述	字符串	长度范围：(0, 1024)	用户输入的症状描述
E4	分析结果	JSON		AI 分析后返回的可能病因、药物等
E5	用户身份信息	字符串	长度范围：(0, 256)	用户身份认证信息
E6	操作日志	JSON		记录用户操作行为

4 接口设计

4.1 用户接口

4.1.1 账户管理接口

- **登录**：`POST /usr/login`
- **注册**：`POST /usr/register`
- **获取用户信息**：`GET /usr/info`

4.1.2 信息检索接口

- **搜索医疗信息**：`GET /medical/search`
- **医疗信息详情**：`GET /medical/info_id`

4.1.3 症状解析接口

- **提交症状**：`POST /symptom/submit`
- **获取分析结果**：`GET /symptom/symptom_id/result`

4.1.4 推荐系统接口

- **获取推荐药物**：`GET /recom/drug`
- **获取推荐治疗方案**：`GET /recom/treat`

4.1.5 健康指导接口

- **获取健康建议**：`GET /advice`

4.1.6 用户反馈接口

- **提交反馈**：`POST /feedback/submit`

4.2 外部接口

本系统的部分数据存储在服务器及数据库中，搜索引擎所需数据以文本形式存储。资源文件及不适宜数据库表项存储的超长文本存储在文件中。网页前端获取用户输入后，由网页后端完成与服务器及数据库的交互。利用`Java`、`Vue`与`MySQL`、AI 引擎之间的接口完成网站外部接口设计。

5 运行设计

5.1 运行模块的组合

本系统按照交互逻辑划分模块，每个模块不共享界面，相对独立。每个模块按照流程划分为客户端界面，客户端脚本和后台服务器程序。各个模块之间不会共享界面，但共享数据库数据和搜索引擎，后台程序只共享数据库连接和搜索引擎。

5.2 运行控制

5.2.1 界面

界面是用户直接与系统交互的部分，界面力求简洁而不简陋，能引导用户进行无碍操作。设计时，以在提供用户便捷操作的基础上增加美观度为基准。

5.2.2 运行控制的条件与限制

本项目的开发要求小组成员足够的参与度，能及时保质保量完成任务。且项目开发过程中可能会有技术上的难点和设备、服务器资源等方面的欠缺，需要开发小组合理利用现有设备和资源，积极查找资料解决问题，在完成项目开发的基础上，同时保证项目的可用性、安全性、可维护性等。

5.2.3 前台与后台的关系

前台主要展示搜索结果信息、AI 对话内容等显示信息，后台主要负责业务流程，控制前台显示信息，负责与搜索引擎和数据库交互。

6 总体数据设计

6.1 数据存储

本系统采用 MySQL 进行账户、用户数据、词条等数据的存储。MySQL 作为数据库管理系统的功能较为完善、安全性较为可靠，具有较为完善的故障解决机制，适合本系统的数据存储需要。本系统将各类数据存储在数据库中，并在有需要时令账户管理、信息搜索等服务从中读取。

6.2 数据安全

本系统由于涉及到用户的隐私数据，因此对数据安全有着较高要求。

在数据的机密性上，我们需要采取措施保证用户的数据仅能由这一用户自己访问，而不能被其他用户或恶意攻击者获取。我们需要对用户的关键数据（如密码、身份信息）做加密 / 哈希处理，确保数据只能以预期的形式访问。

在数据的完整性上，我们需要确保数据不因物理因素或外部攻击而篡改。对于前者，我们需要选用完善的存储解决方案，选择可靠的存储设备与数据库软件；对于后者，我们需要确保数据库只能由后端服务器进行访问，并且后端服务器程序在操作数据库前必须采取措施验证输入的有效性，避免未经授权的修改及 SQL 注入攻击。

在数据的可用性上，我们需要建立完善的用户认证机制，使用密码等手段区分不同的用户，并确保有且仅有其被允许访问的对应数据被读取。

6.3 逻辑结构设计要点

本系统采用 MySQL 作为数据存储的解决方案，因此数据的逻辑结构主要以数据表的形式表现。

6.3.1 账户（accounts）

属性名	类型	约束	描述
userid	string	GUID, pk	用户主键
username	string	独特，非空，>3字符，只含英文数字	用户名
password	string	非空	密码的SHA256哈希值
realname	string	>2字符	真实姓名
pid	string	独特，非空	证件号码
gender	int	0/1/2（保密、男、女）	性别
birthday	bigint	大于0	生日（UNIX时间戳）

6.3.2 对话记录（conversations）

属性名	类型	约束	描述
srcid	string	FK: accounts.userid	对话发出者
destid	string	FK: accounts.userid	对话接收者
type	int	0文本，1图片	内容类型
content	string		内容

6.3.3 知识点

属性名	类型	约束	描述
id	string	GUID PK unique	知识点ID
category	int	>0	知识点类型
title	string	unique	知识点标题
content	string		知识点内容

6.4 物理结构设计要点

本系统存储的数据使用 MySQL 的默认数据库格式（InnoDB）存储在内存及磁盘中。

7 系统出错设计

7.1 出错信息

输出信息	含义	解决方法
磁盘空间不足	服务器的剩余磁盘空间不足，导致数据写入失败	清理磁盘空间
数据库连接失败	由于数据库服务器错误或服务器配置错误，导致后端服务器无法连接至数据库服务	重新配置服务器及数据库
AI调用失败	由于服务器配置错误导致无法连接AI大模型接口	重新配置AI接入
用户数量过大	用户数量超出数据库所容许的列表条目数	调整数据库，扩大列表条目数
数据导入失败	管理员导入信息条目时格式错误或大小过大	确定数据的完整性或分次导入

7.2 补救措施

- 定期维护。管理员每天固定时间确认各项服务可用性、确认服务器磁盘、占用率情况，并在必要时升级服务器的软硬件。
- 停机维护。当错误过于严重时，应当终止当前正在进行的数据库操作，关闭服务并进行软硬件修复。

8 系统维护设计

8.1 概述

该系统作为一个搜索引擎，其时效性要求较高，且可预期的运行时长也较长。因此，我们需要尽可能地保证其运行的可靠性，并且应该确保其维护操作容易、风险性低；系统涉及到用户的隐私信息较多，因此需要确保系统的数据安全，避免隐私数据的泄露。具体要求如下：

- 系统应当支持管理员导入医药知识，并确保导入文件过大或格式错误等特殊情况不会导致系统被迫重启；
- 系统应当有完善的用户管理机制，实行会话管理，确保只有用户本人才能管理其自身的隐私数据；
- 系统应当对来自用户的错误输入进行合理的防御操作。当用户输入过长时，系统应该进行必要措施，避免缓冲区溢出等危险情况；当用户输入格式不符合要求时，系统应当拒绝请求；
- 系统应当对用户及管理员的各项重要操作（例如：用户注册，管理员添加信息词条）自动生成日志记录，以供审计人员查阅。

8.2 检测点设计

8.2.1 用户注册

- 用户名：不能为空，不能过长过短，不能重复
- 密码：不能为空，不能过长过短
- 确认密码：必须与密码一致

8.2.2 用户登录

- 用户名：在数据库中存在，否则返回错误
- 密码：能够正确验证

8.2.3 信息查询

- 搜索栏：检查用户输入内容长度
- 搜索结果：相关性应当足够高
- 一般搜索
- 筛选搜索

8.2.4 AI 对话

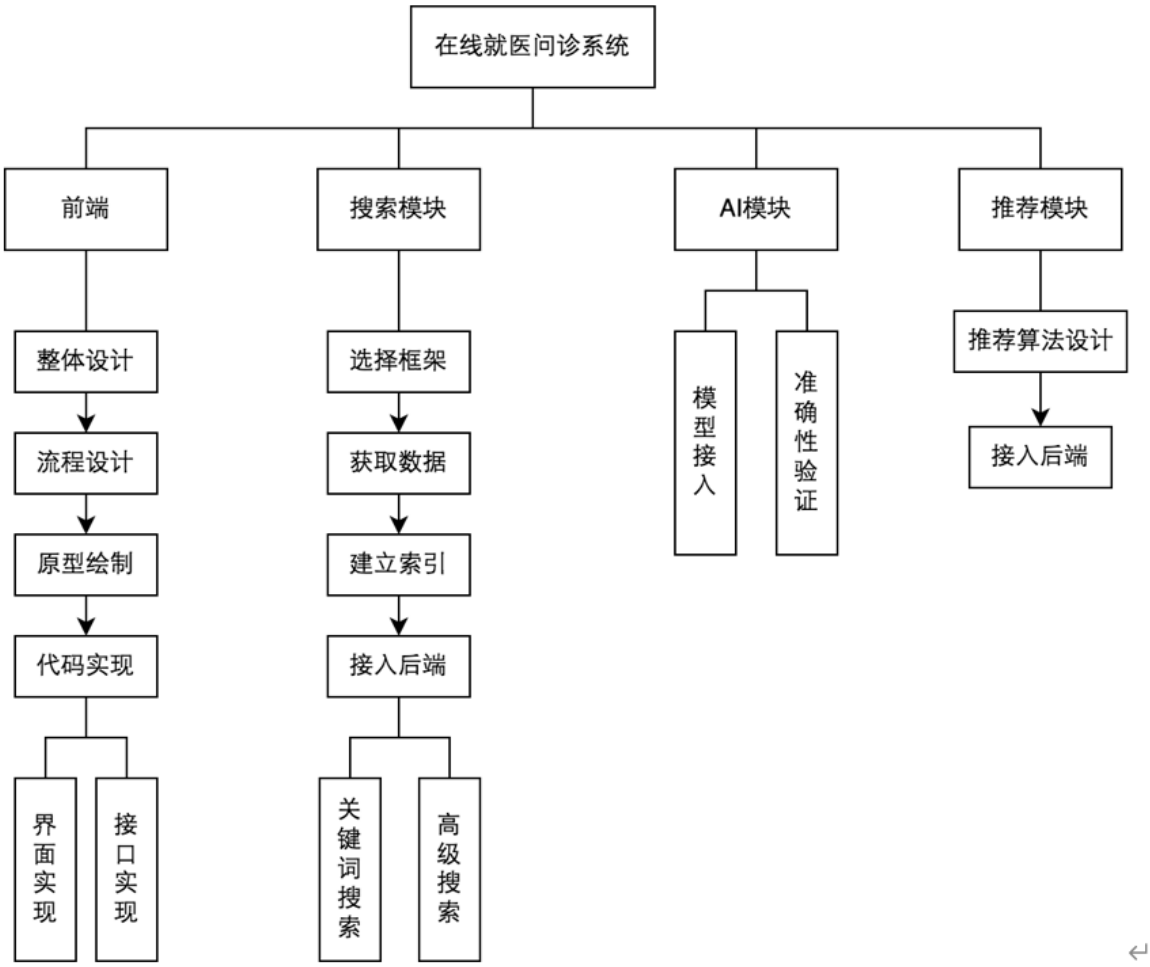
- 输入框：检查用户输入内容长度
- AI 返回结果：是否与医药相关，是否符合医学常识

9 模块设计规划

9.1 项目架构设计

基于 AI 的就医问诊系统采用前后端分离的设计，分为前端模块、信息搜索模块、AI 解析模块、推荐系统模块。前端采用`Vue.js`框架编写，使用`Node.js`运行；后端采用`Django`框架，并采用`MySQL`作为数据存储的数据库解决方案。前后端之间使用`JSON`以`REST API`的形式进行通信，通过`DRF`实现后端的`REST`模型。前端向用户提供一个运行在浏览器中的现代的 UI 界面，并根据用户在界面上的操作通过用户的浏览器向后端发送`JSON`形式的 XHR 请求，后端处理请求、将相应操作与结果记录到数据库中，并向前端返回结果，前端浏览器处理此结果并将用户可理解的信息显示在画面上。

9.2 项目任务分解



9.3 前端

前端采用现代`vue.js`框架，界面简洁高效。本部分展示的界面仅为初步设计。

9.3.1 登陆界面

9.3.2 注册界面

9.3.3 查询与对话界面

9.4 后端

后端采用 `Django` 运行，采用 `DRF` 框架构建前后端 `REST API` 通信模型。

功能编号	功能名称	详细描述
BS-001	用户注册	前端通过 JS 传递新用户的信息，在数据库中添加用户记录、注册用户
BS-002	用户登录	前端传递加密的用户名和密码，后端验证有效性并给予前端登陆令牌
BS-003	信息检索	依据前端传来的搜索条件返回搜索结果
BS-004	AI接入	将前端的对话信息发送给 AI 模型，得出结果经过分析处理后反馈给前端
BS-005	对话管理	将用户与用户、用户与 AI 的对话记录在数据库中，并按需要将当前用户的对话返回给前端

功能编号	功能名称	详细描述
BS-006	反馈与推荐	收集用户的搜索偏好与结果反馈，改进搜索结果或提供更符合用户需要的结果

9.5 数据模块设计

9.5.1 账户管理

本系统要求用户注册账户后再使用，并在系统中记录其用户名、密码、身份信息，以及每个用户的用户记录，以供查询。同时，每个账户只能查看自己的对话记录，以便保障隐私。

用户的密码等关键信息应当哈希或加密处理，以保证信息安全性。

9.5.2 推荐数据

本程序的推荐系统模块记录每个用户所关联的搜索习惯与反馈数据，推测用户的搜索倾向，以便更加精准地向用户推送其所需要的信息。

9.5.3 数据更新

此系统应当能够更新存储的知识数据，以便保持其时效性。可以由管理员导入数据进行更新，或每周、每月定时更新。