## 法律大模型开发文档

前端基于师爷AI，后端调用通义法睿、通义千问、DeepSeek API接口

服务器：n5.hifiax.com

1. 前端开发

目录：~/feixiong/law-llm-fe ~~，~/baichang/recheck-gpt~~

去掉“百昌”、“师爷”、“再查查”字样

修改router.ts，对应功能入口

export default defineComponent格式，setup函数，要将变量、函数都放进return中

1. 后端开发

目录：~/pwork/fx （~/feixiong/law-llm-be）

编写：main.py，用fastapi封装接口

* chat：法律咨询
* law：法条检索
* case：案例检索
* doc：文书生成
* report：尽调报告
* keypoint：要点抽取

启动命令：

uvicorn main:app --host 0.0.0.0 --port 8000 –reload

1. 界面要素

以views/question为例

* 模板：index.vue
* 代码：index.ts
* 样式：index.scss

用v-for遍历问答序列，用v-html嵌入内容文字，用MarkdownIt作格式转换。用isLoading、loadingStatus控制加载逻辑

前后端访问地址，在vite.config.ts中定义代理服务server.proxy，不然会出现跨域问题。前后端访问方式，改为axios

异步流式访问，通过onDownloadProgress事件回调函数实现。应答结束添加[DONE]标志，前端识别后结束加载状态

1. 法律咨询

后端代码：deepseek.py

通过Stream方式调用siliconflow的DeepSeek-V3模型获取流式响应，再通过yield回调异步向前端发送事件

前端代码包：question

数据定义chat.d.ts，问答序列chatList，核心代码sendFn

1. 法条检索

后端代码：farui.py，函数：law\_query

调用通义法睿run\_search\_law\_query\_with\_options接口，参数为法规名称、关键词列表，返回相关法条。目前返回的是整篇法规，没有根据关键词作进一步筛选。目前限定法规，未作泛化搜索

前端代码包：legal-search

数据定义law.d.ts，法条数据lawList（每次问答重置），核心代码sendFn

1. 案件检索

后端代码：farui.py，函数：case\_query

调用通义法睿RunSearchCaseFullTextRequest接口，参数为提问文字，返回相关案例

前端代码包：case-retrieval

数据定义case.d.ts，案例数据caseList（每次问答重置），核心代码sendFn

1. 文书生成

略

1. 尽调报告

后端代码：report.py

因为没有后台数据库，通过关键词匹配，直接读取word文件，利用docx解析，生成markdown格式。建立Table对象和xml对象间的映射，以定位表格在文档中的先后位置

前端代码包：report

数据定义report.d.ts，案例数据reportList（未按照问答展开），核心代码sendFn

1. 要点抽取

略