

作业四

知识点

1. RESTAPI
2. JWT鉴权
3. 参数校验
4. ORM
5. 工程化
6. 异常处理
7. 文件处理

推荐教程

1. RESTAPI
2. JWT鉴权知识 <https://learnku.com/articles/17883>
3. JWT用法 <https://blog.csdn.net/yangbindxj/article/details/125344291> (这里用了Fastapi的例子, 其实jwt库两个都可以用)
4. ORM 用Flask推荐flask-sqlalchemy, 用Fastapi推荐sqlalchemy, flask-sqlalchemy可以通过paginate分页比较方便, sqlalchemy分页需要offset+limit
5. 蓝图 (Flask) /APIRouter (Fastapi) 看各自的文档还有网上资料
6. 通过异常处理器 (@app.errorhandler/@app.exception_handler) 捕获异常 (参数错误/token错误), 并把异常结果格式化为应该返回的格式
7. 文件的上传, 响应文件下载

作业

前端页面 <https://github.com/1379255913/westtwo-winter>

接口文档 <https://console-docs.apipost.cn/preview/40c70e0e04322bed/9a886251aff222d0>

酷我音乐搜索页面 <http://www.kuwo.cn/search/list?key=a>

外链转化工具网站 (通过rid获取歌曲地址) <https://link.hhtjim.com/>

根据前端页面和接口文档, 完成以下内容:

1. 使用爬取酷我音乐上的歌曲或接口, 和本次功能对接
2. 跨域功能
3. 登录注册功能, 对应密码应该进行密码哈希处理
4. 对登录和注册的值进行检验, 在检验有误时按照要求返回结果
5. 登录后后端生成token并返回给前端, 除登录注册外的所有接口都要对token进行检验, 不合法或者过期按照要求返回结果
6. 使用ORM对数据进行操作
7. 可以通过rid来下载音乐文件, 对下载的音乐的历史记录进行保存
8. 可以对历史记录进行收藏和删除
9. 分页搜索, 默认一页为10条记录
10. 有清晰的项目结构, 分为user/search/history三个模块, 通过蓝图 (Flask) 或者APIRouter (Fastapi) 进行分组
11. 符合RESTAPI (不符合你好像也对接不上)

12. 对于有能力修改前端页面的，或者想做更多功能的，可以自行修改或者和我来交流

提示：

- 除了登录、注册、下载 接口不需要token，其他都要做处理
- 搜索时需要添加搜索记录，保存搜索歌曲信息，不然无法根据rid获取歌曲信