МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА №44

КУРСОВАЯ РАБОТА (ПРОЕКТ)   
ЗАЩИЩЕНА С ОЦЕНКОЙ

РУКОВОДИТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Старший преподаватель |  |  |  | Д. В. Куртяник |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

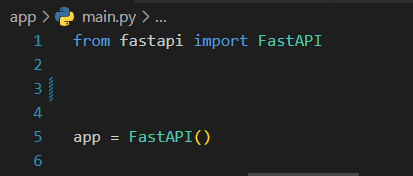
|  |
| --- |
| ПОЯСНИТЕЛЬНАЯ ЗАПИСКА К КУРСОВОЙ РАБОТЕ |
|  |
| по дисциплине: Основы программирования |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ гр. № | 4319 |  |  |  | А. М. Емельянов |
|  |  |  | подпись, дата |  | инициалы, фамилия |

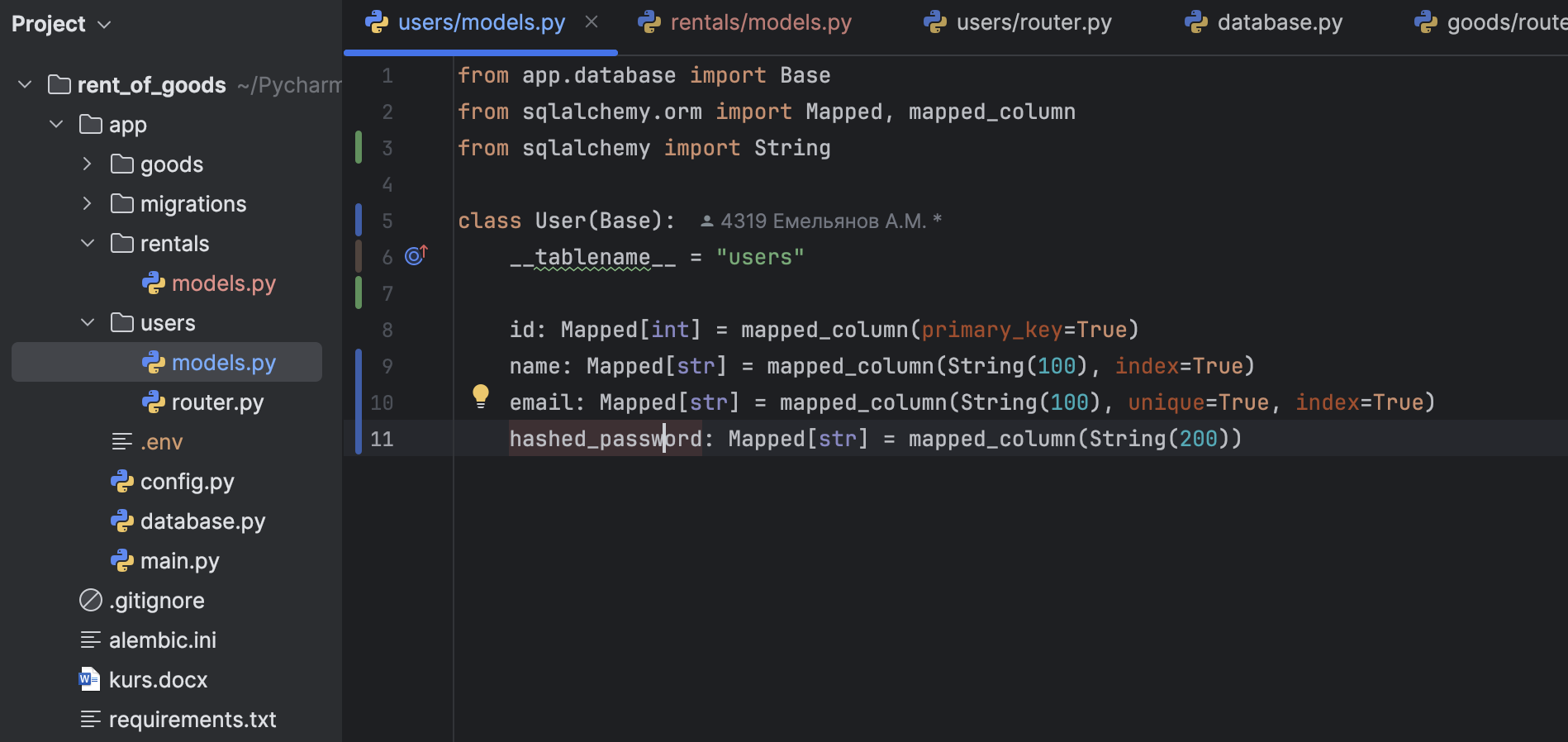
Санкт-Петербург 2024

Создал первый файл (main.py) в нём поместил

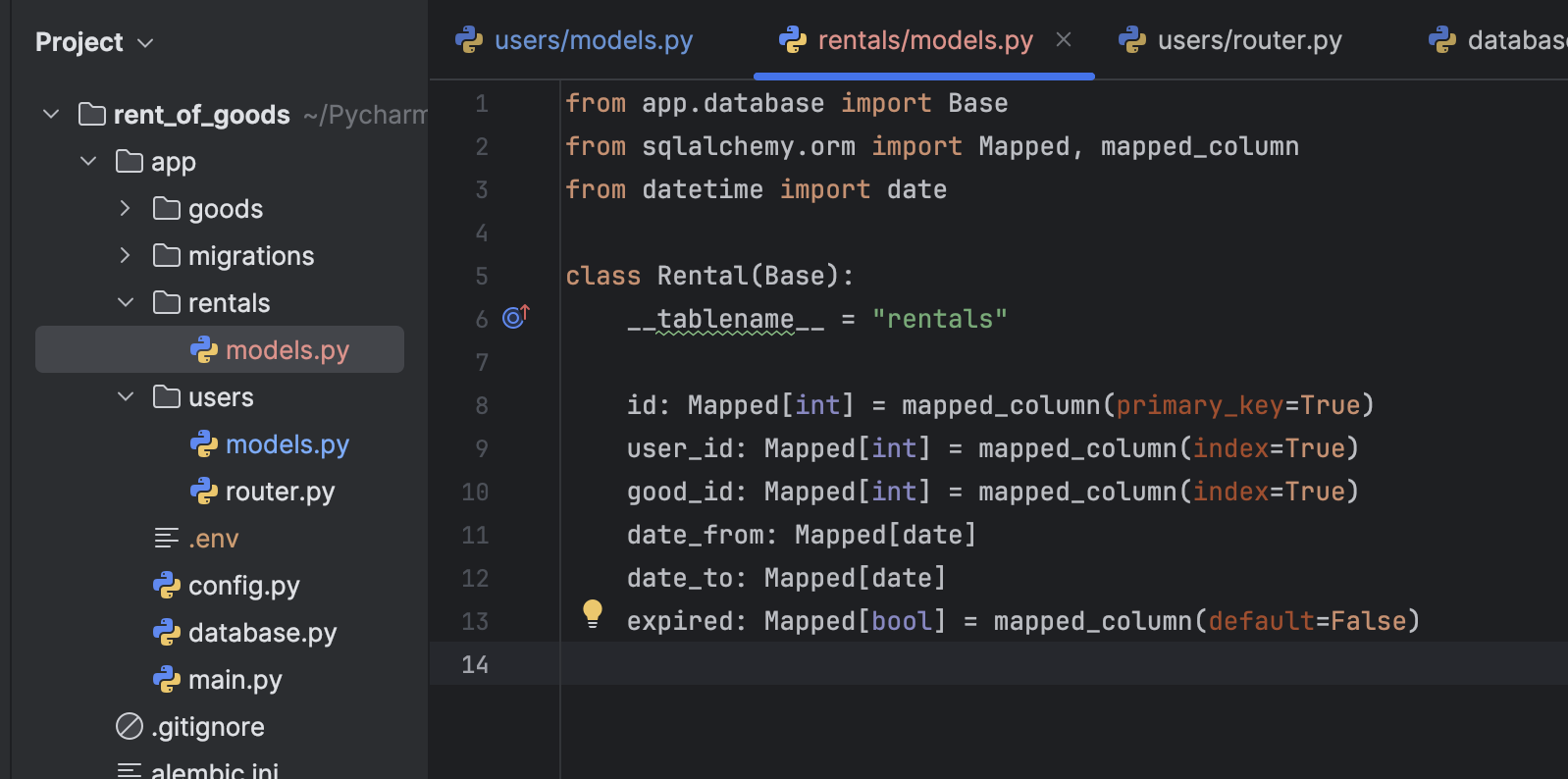
Создал файл requirements.txt для настройки зависимостей



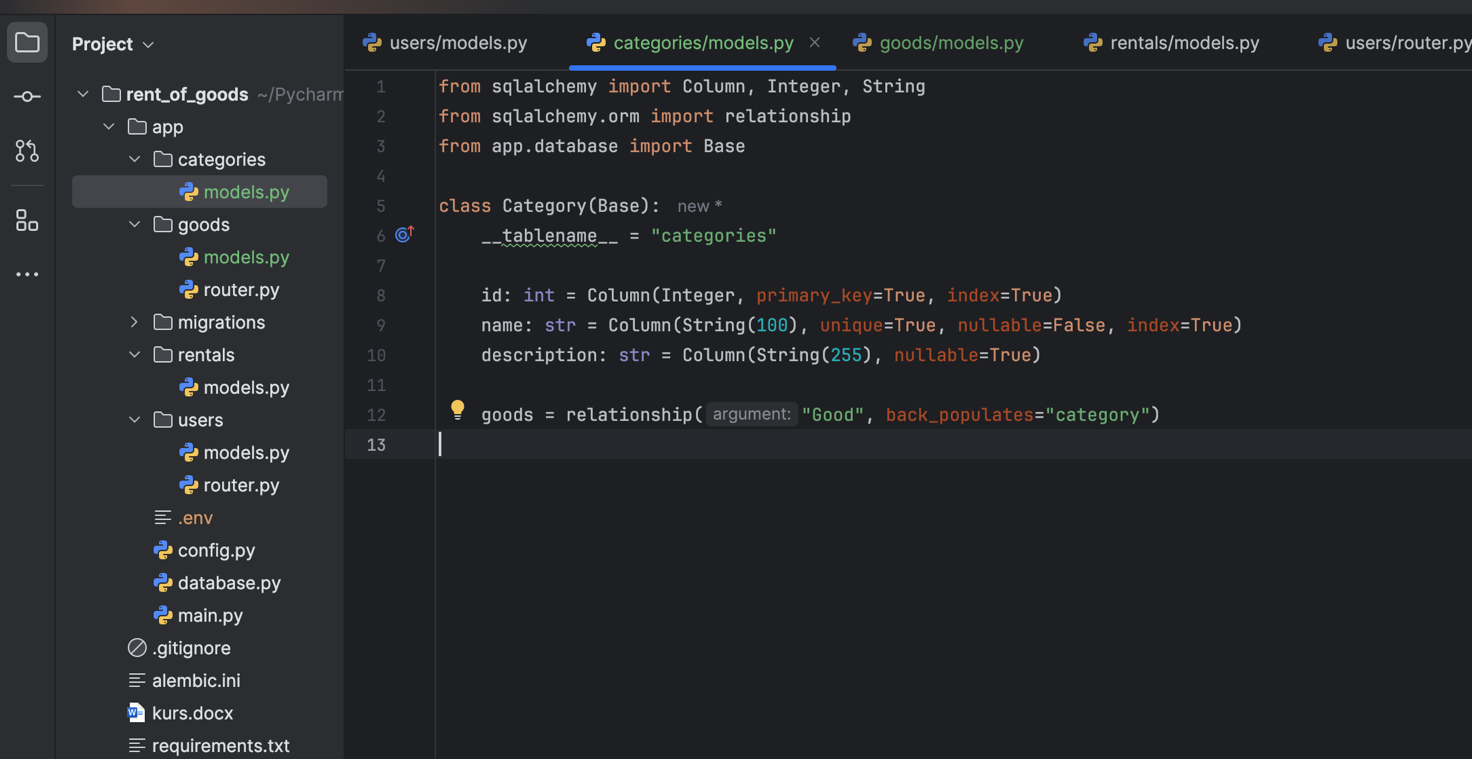
Изменил users/models.py

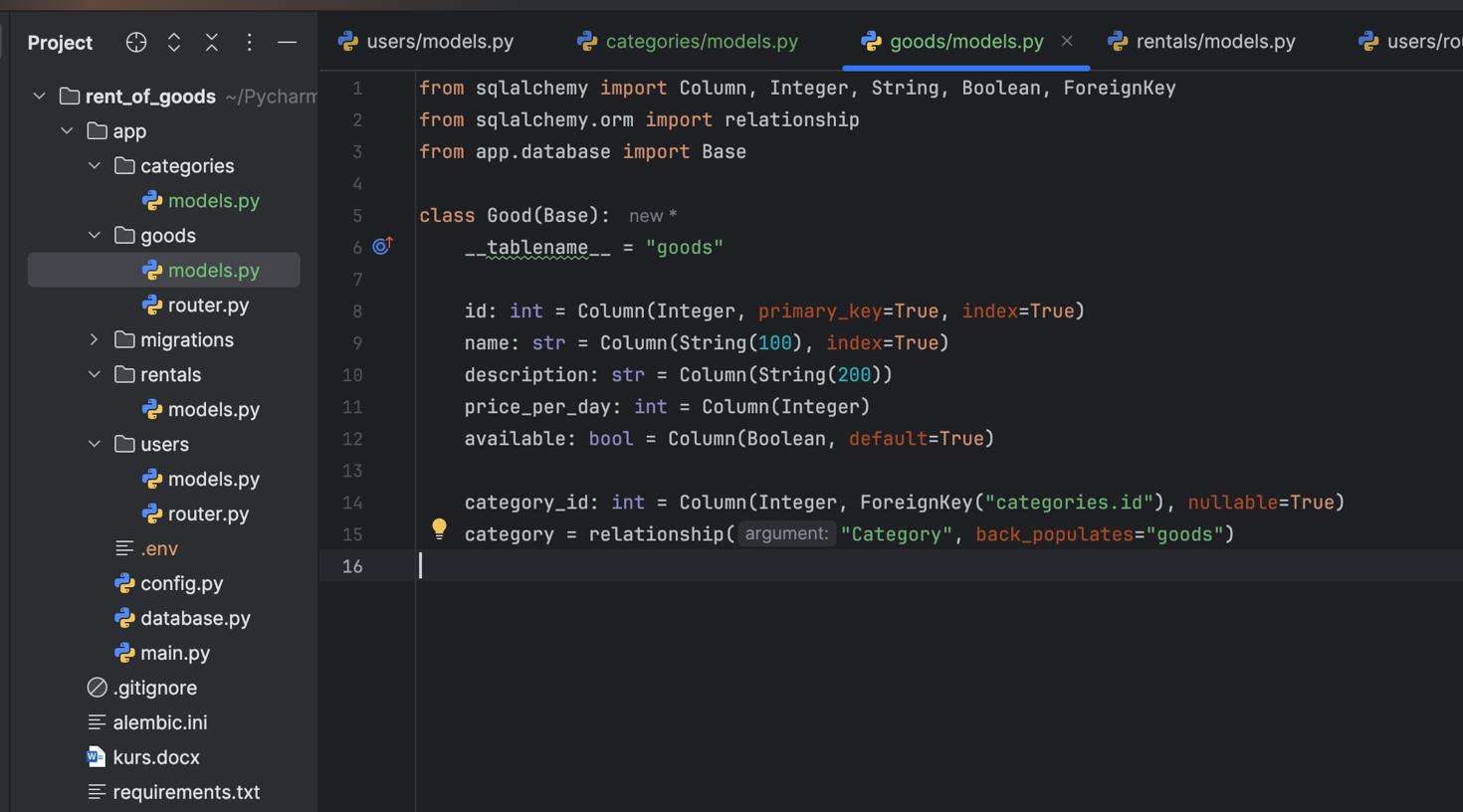


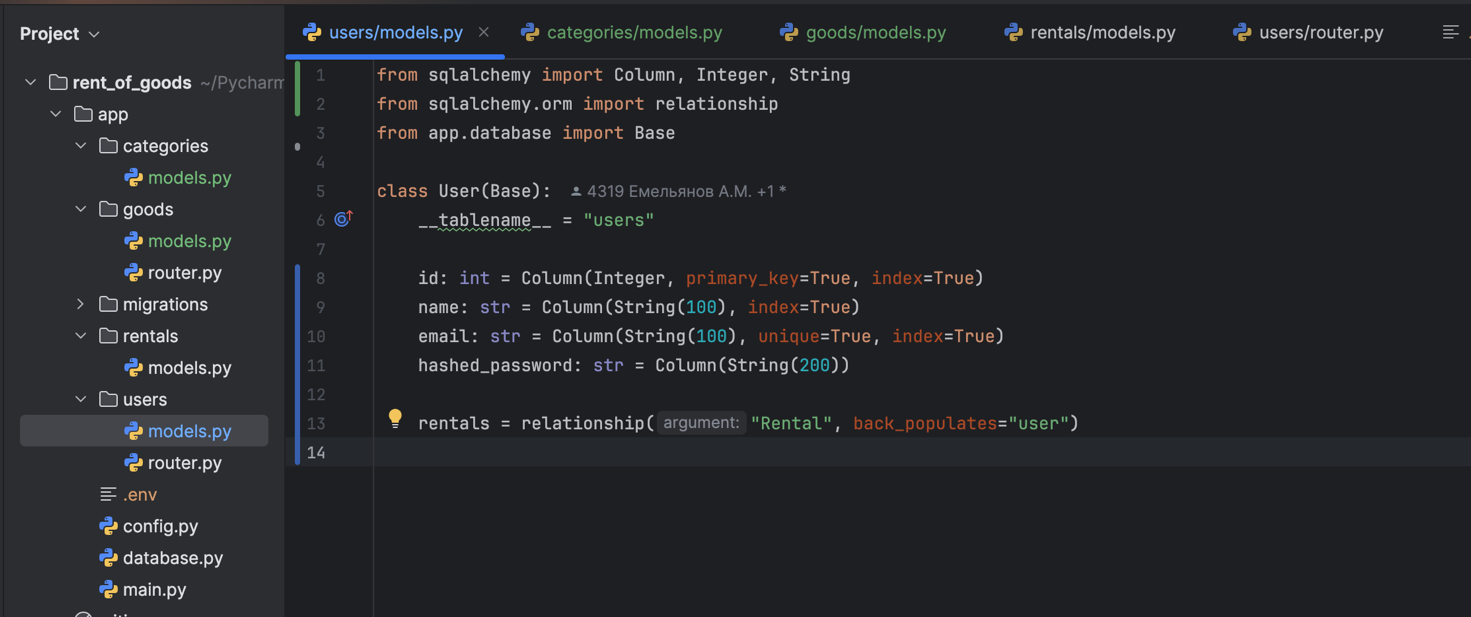
Изменил rental/models.py

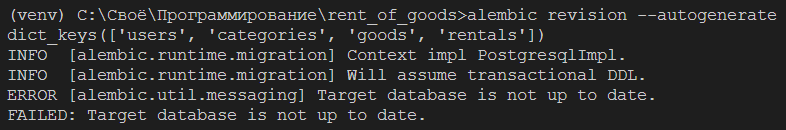
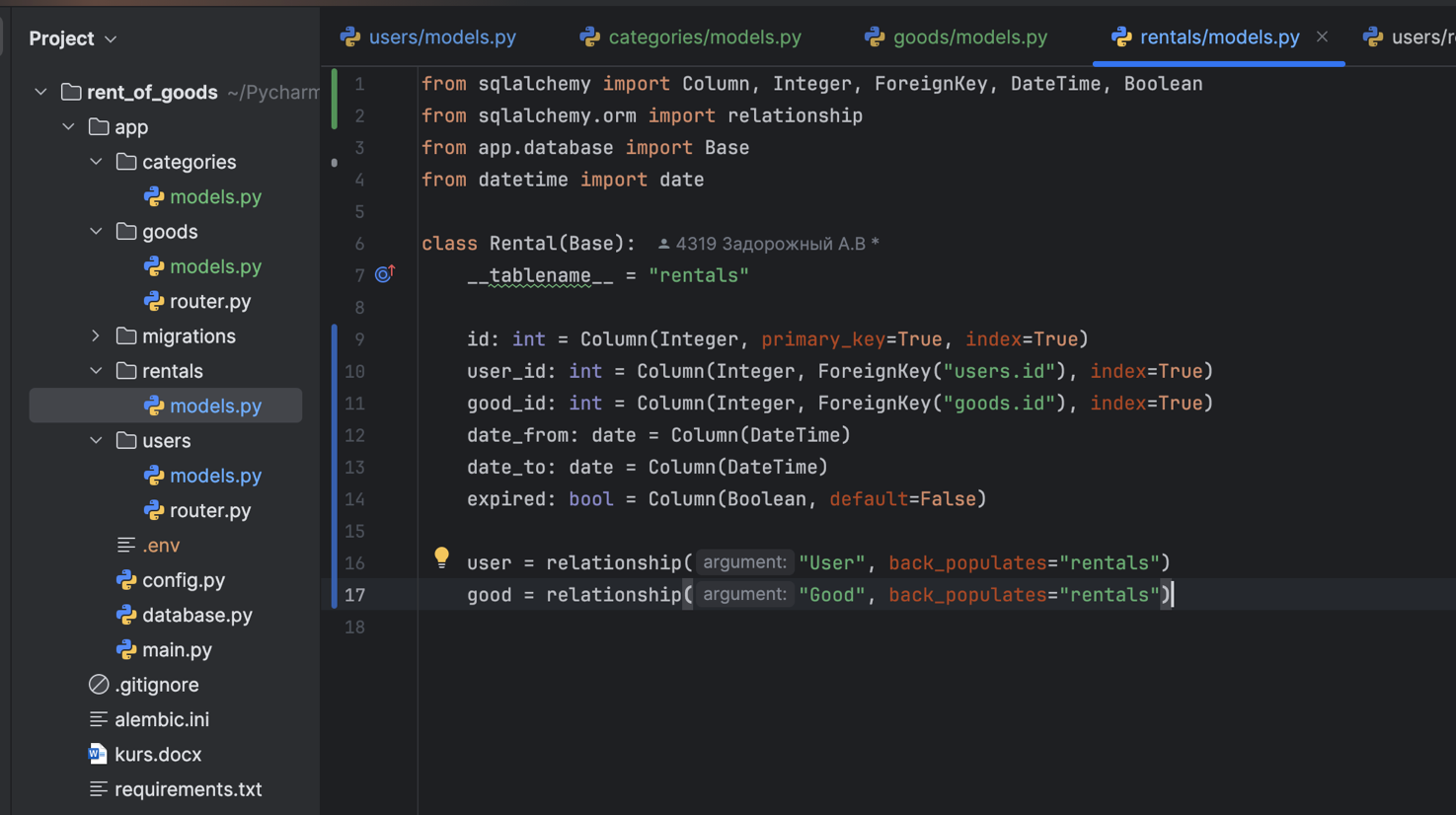


Добавил каталог categories  + models.py

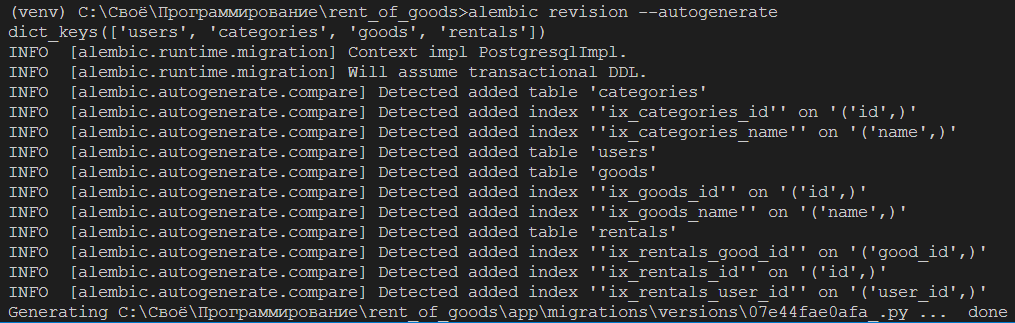


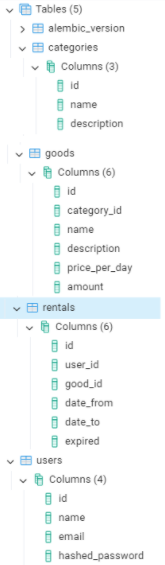
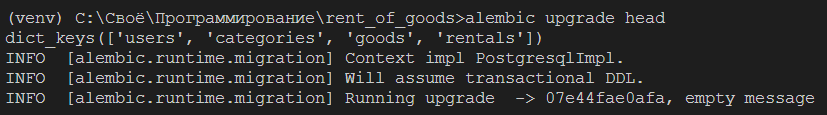
Обновил models.py для /users /rentals /goods (Установил связи между таблицами)

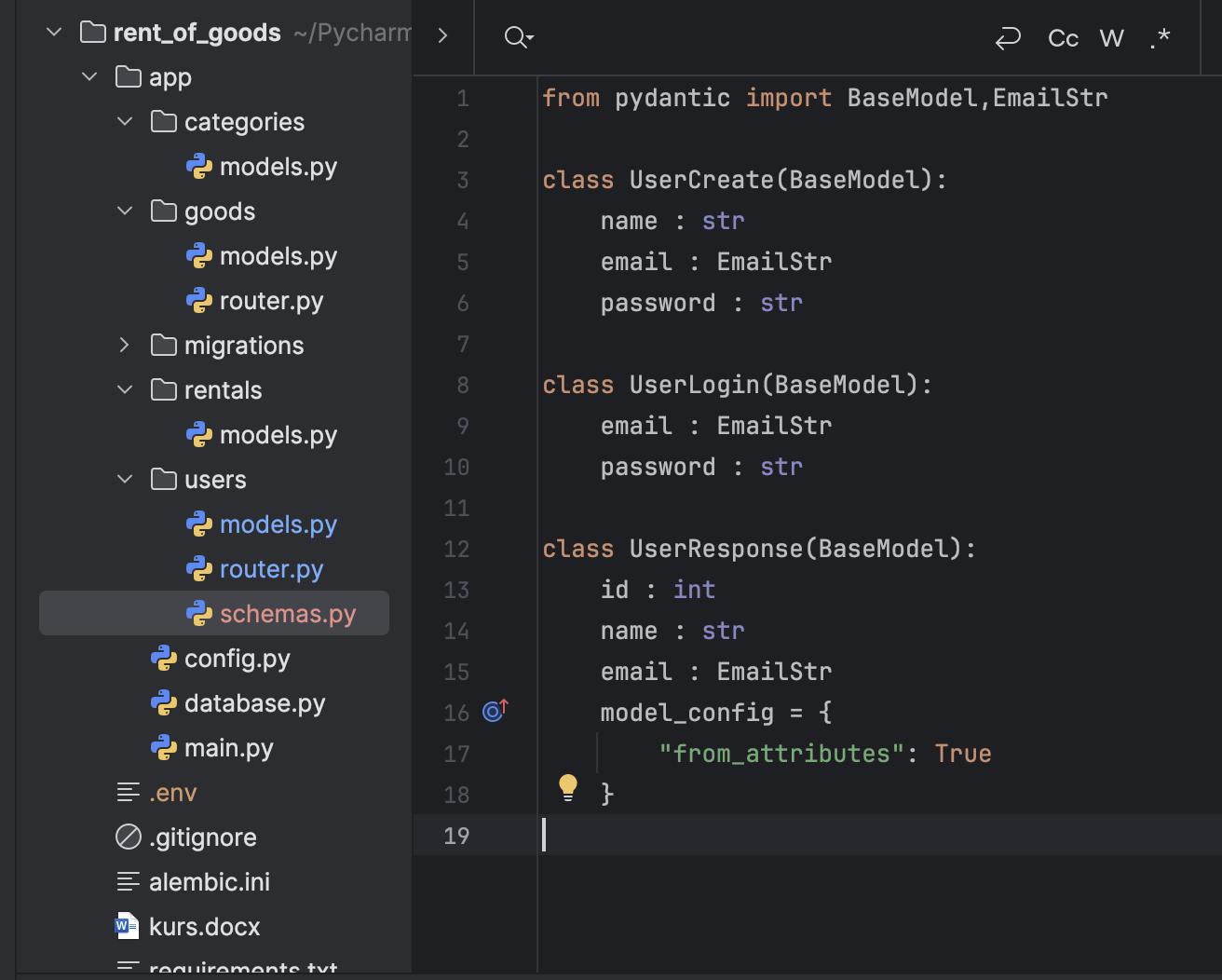


Возникла проблема при миграции. Пытаюсь решить

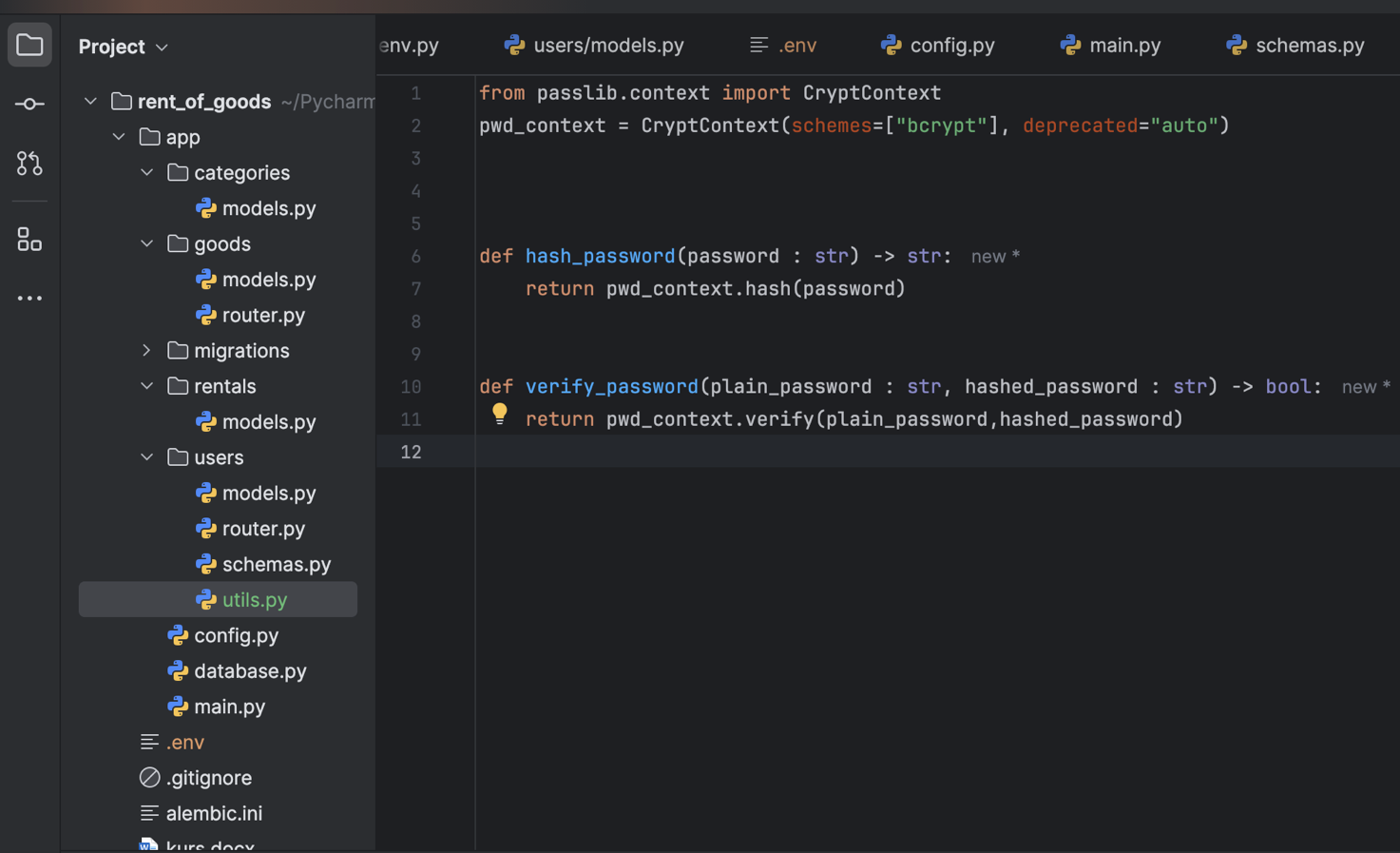
Проблема была в другой миграции. Удалил её и создалась новая миграция

Далее прогоняю миграцию и создаю все необходимые таблицы

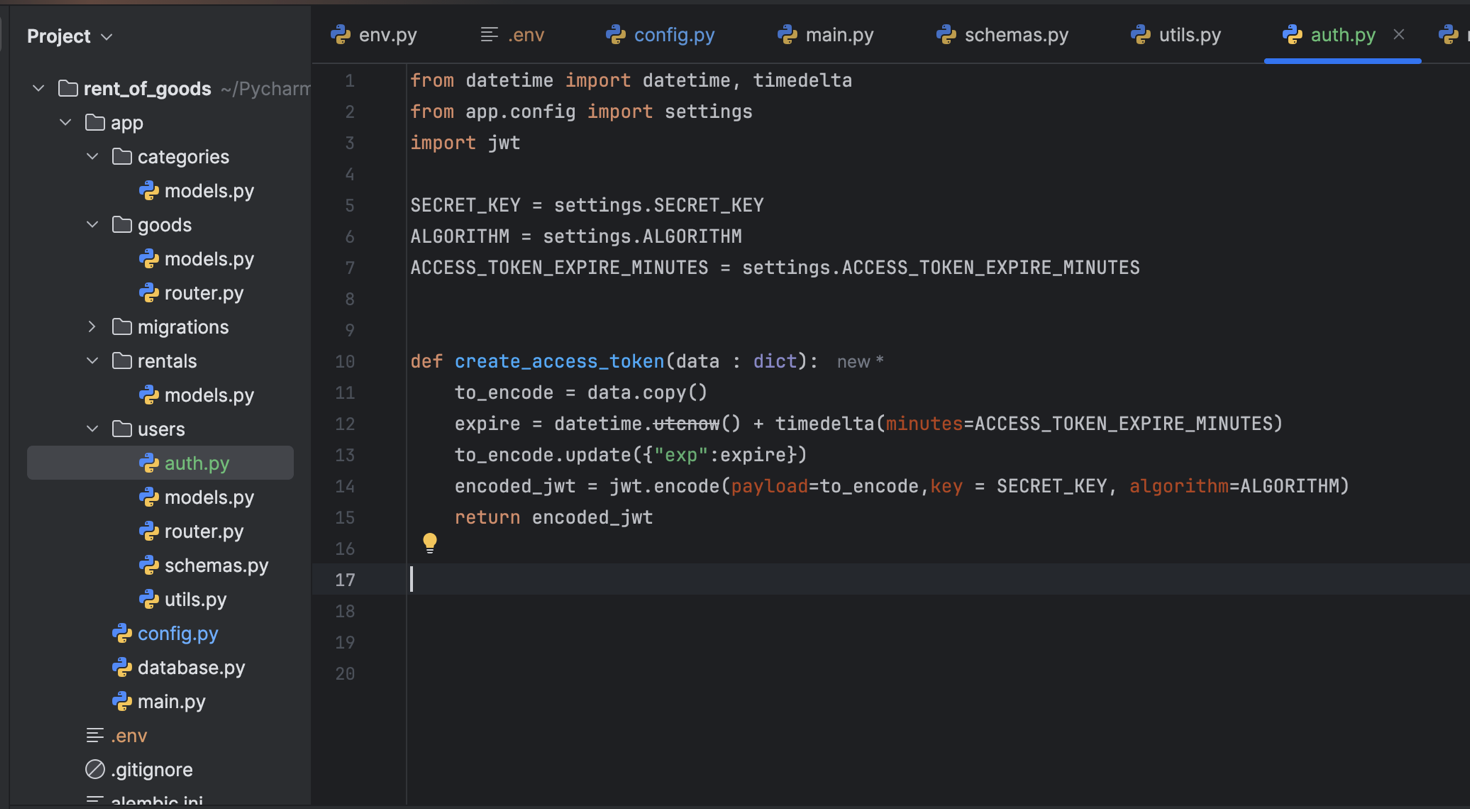
Создал несколько схем для аутентификации



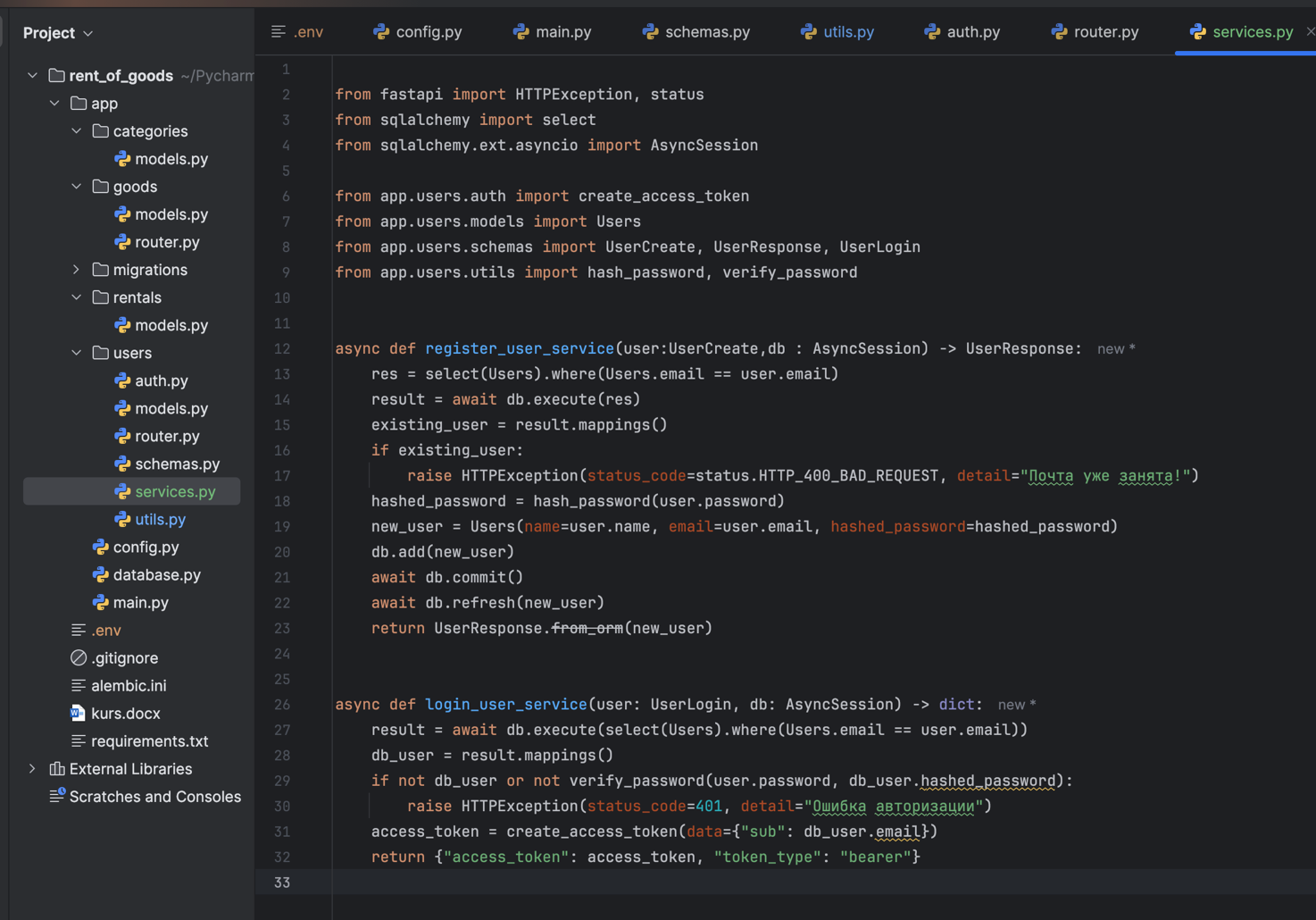
Создал utils.py для хэширования паролей



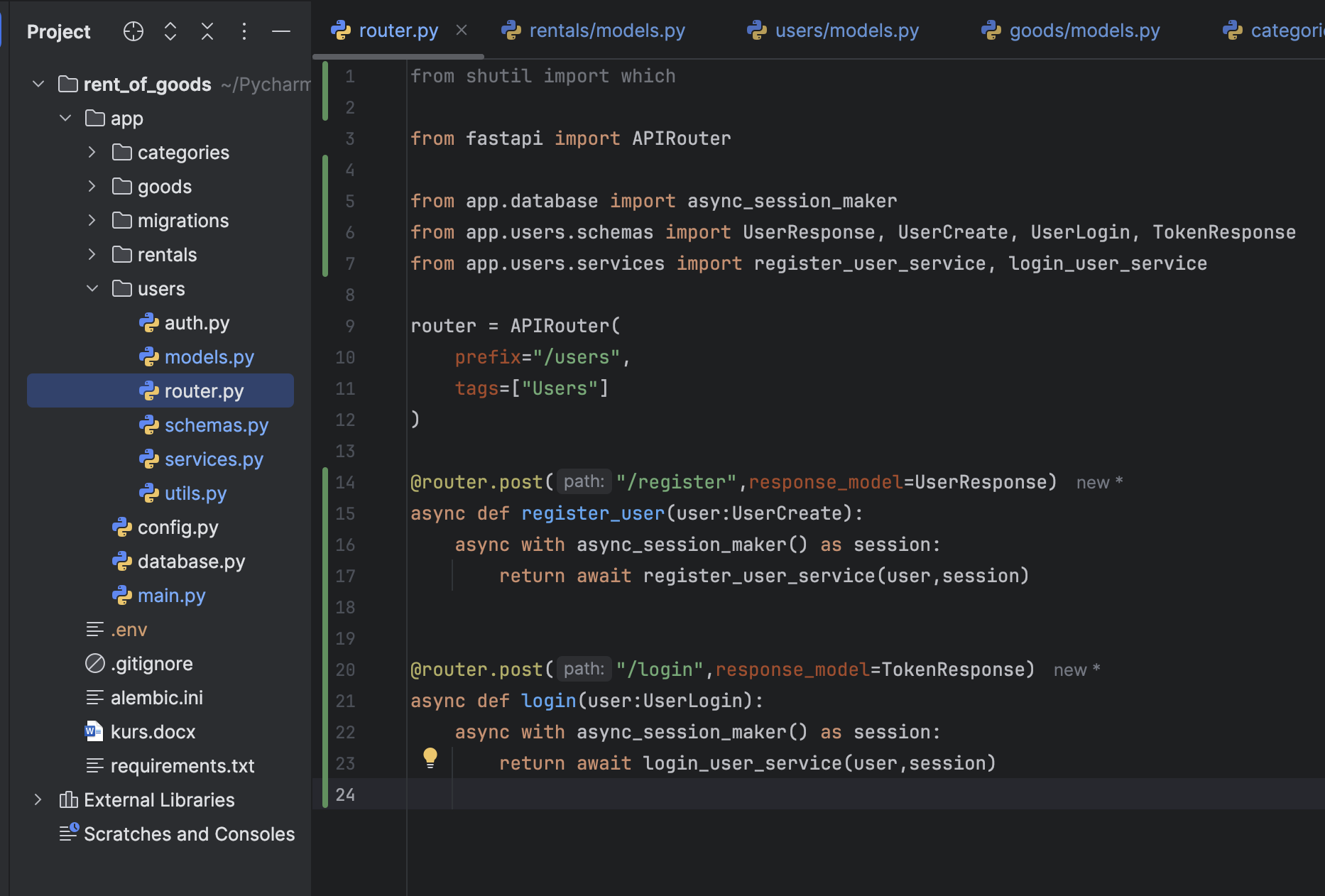
Создал auth.py для получения jwt токена, скрыл важные данные



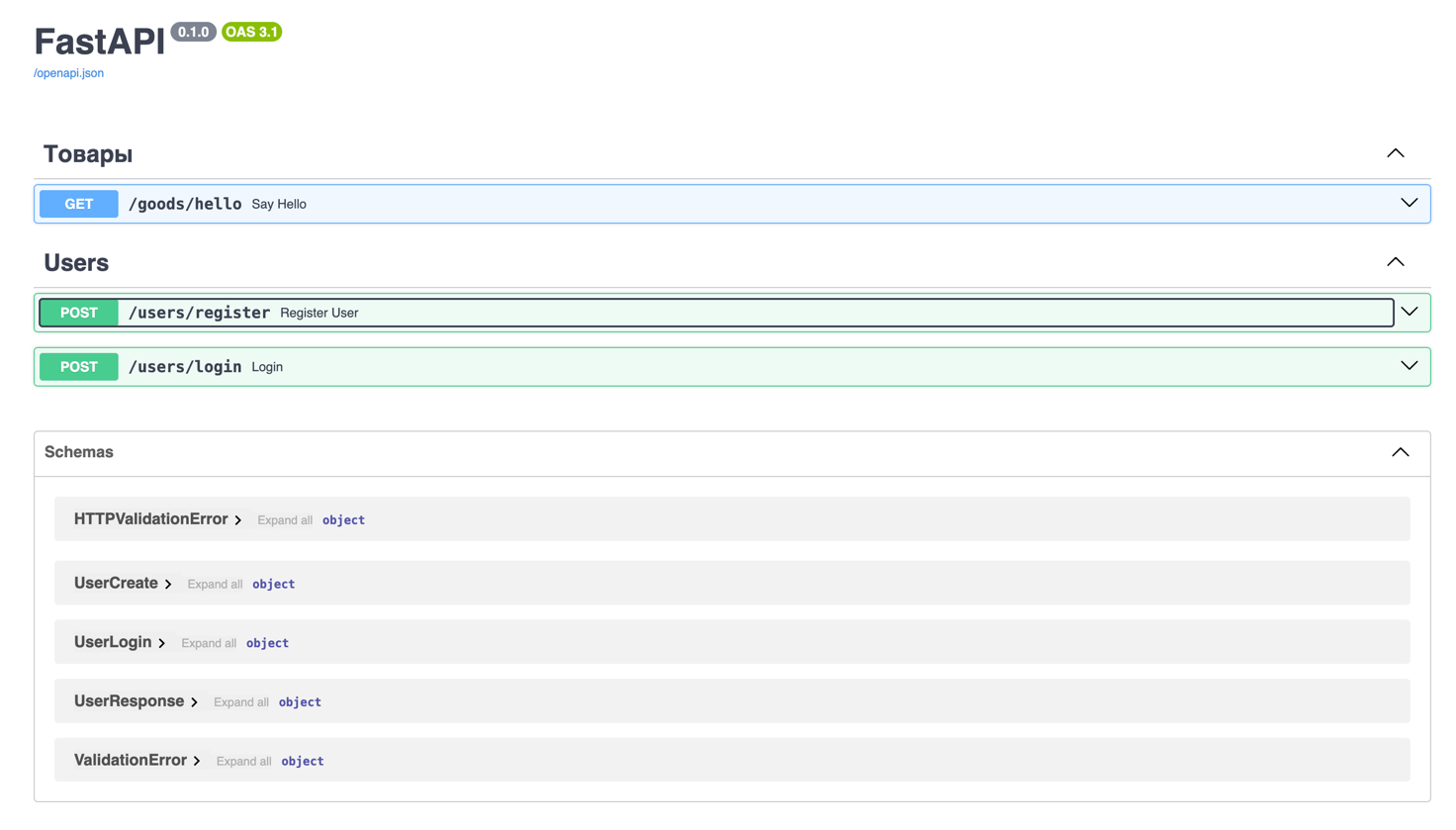
Добавил логику регистрации и входа



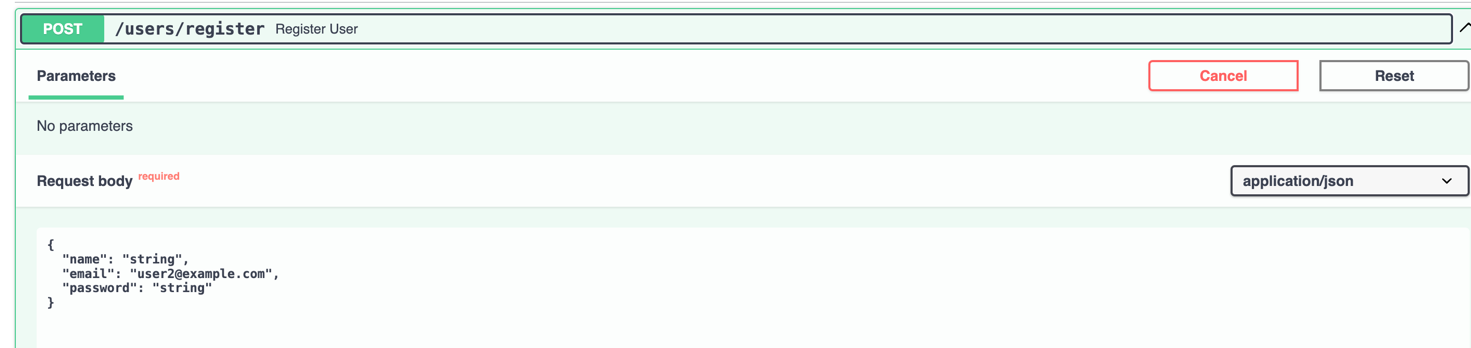
Настроил роутер для регистрации



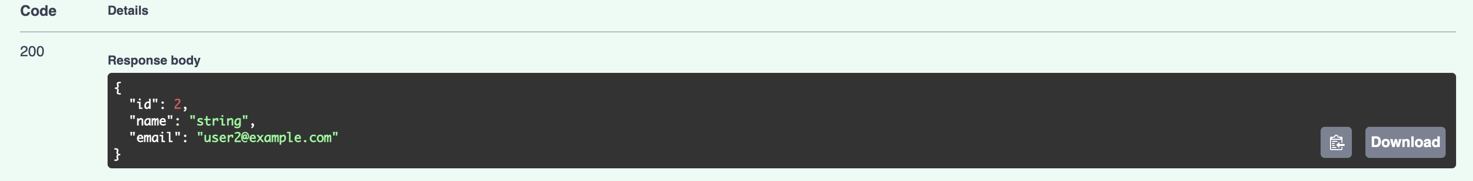
В результате появилось два эндпоинта для аутентификации и несколько схем



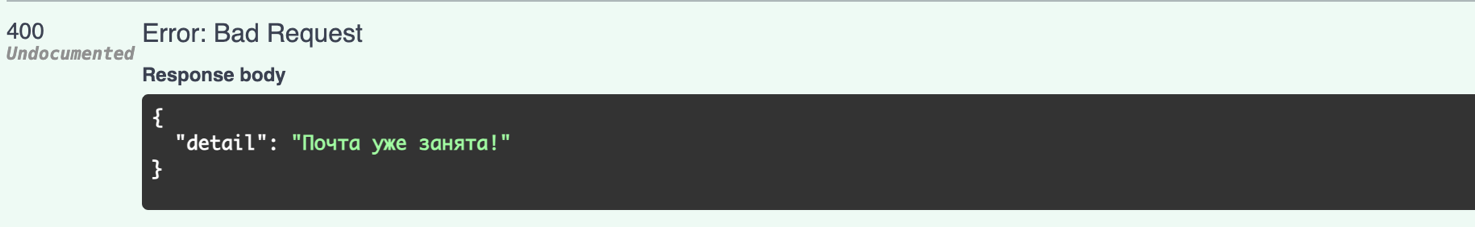
Пример теста энпоинта register



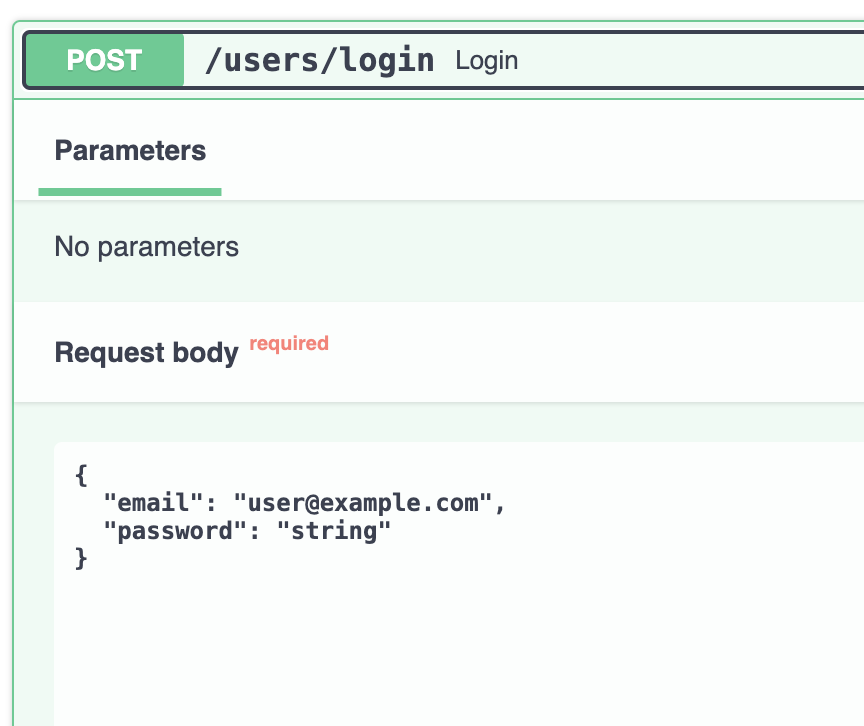
Результат



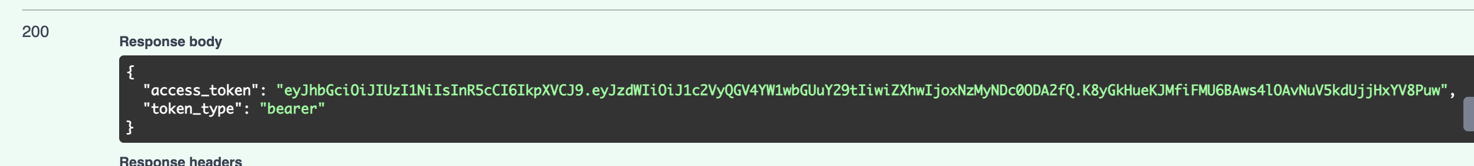
При повторной регистрации с занятой почтой



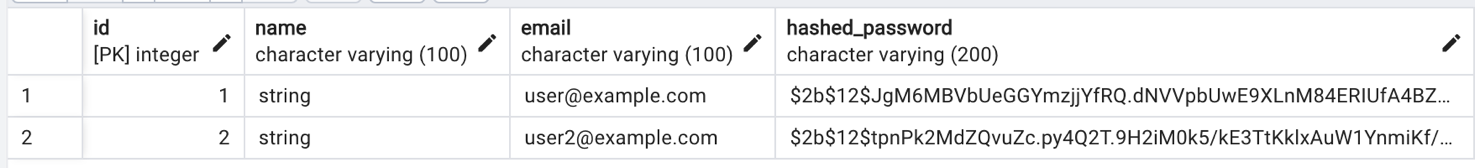
Пример теста эндпоинта login



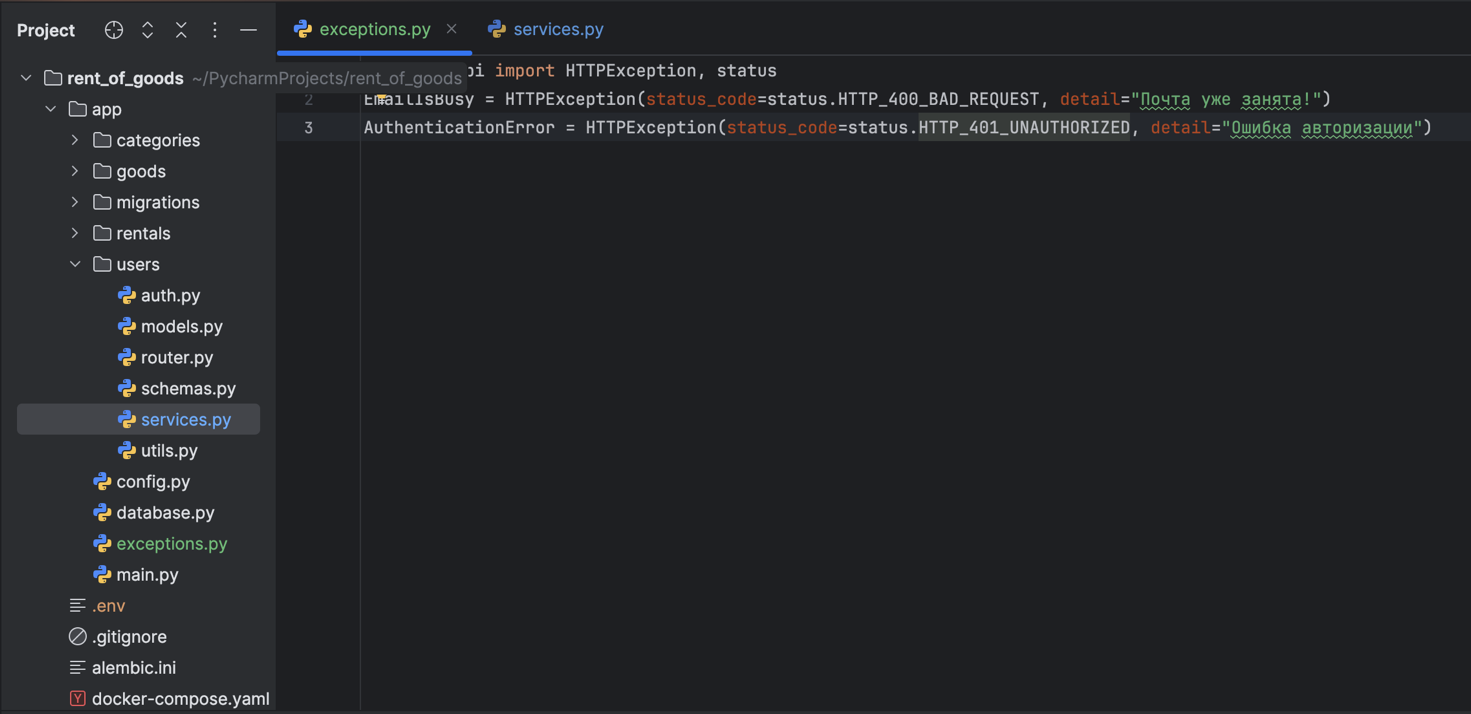
Результат



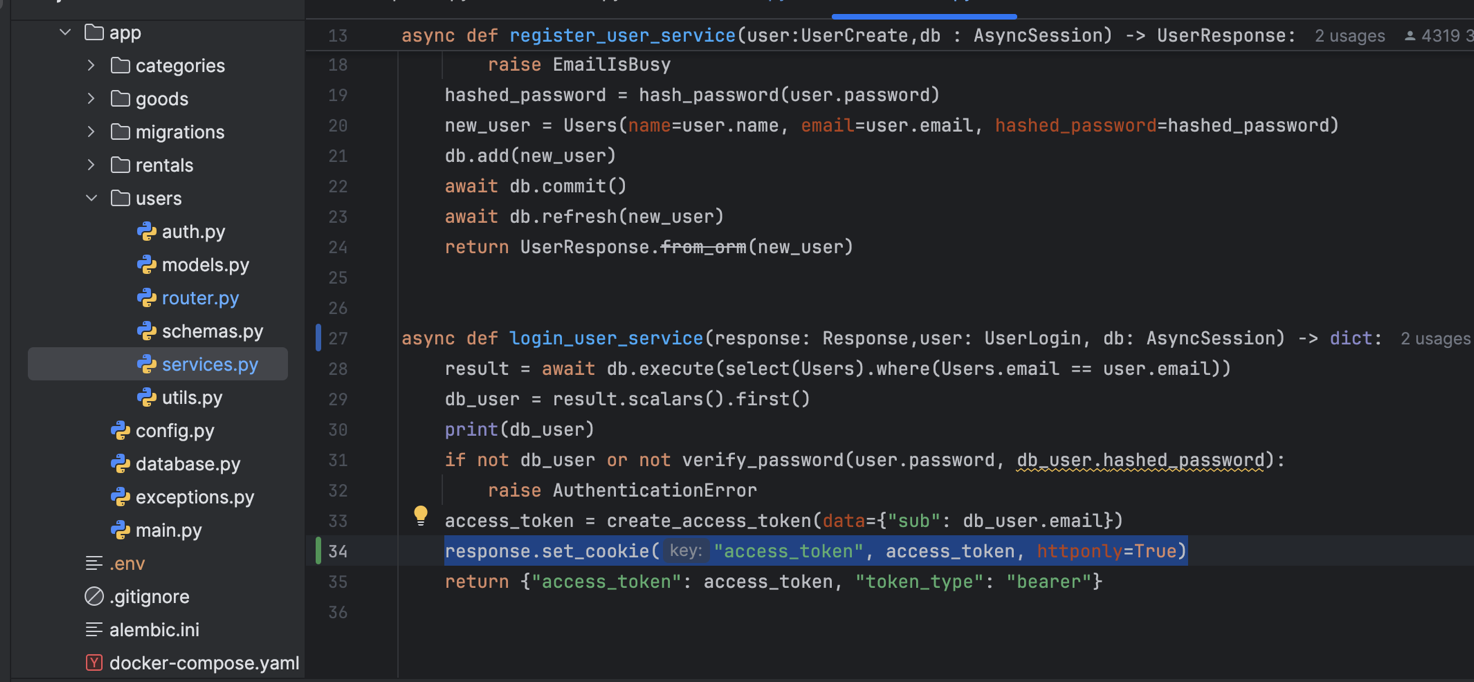
Отображается в бд



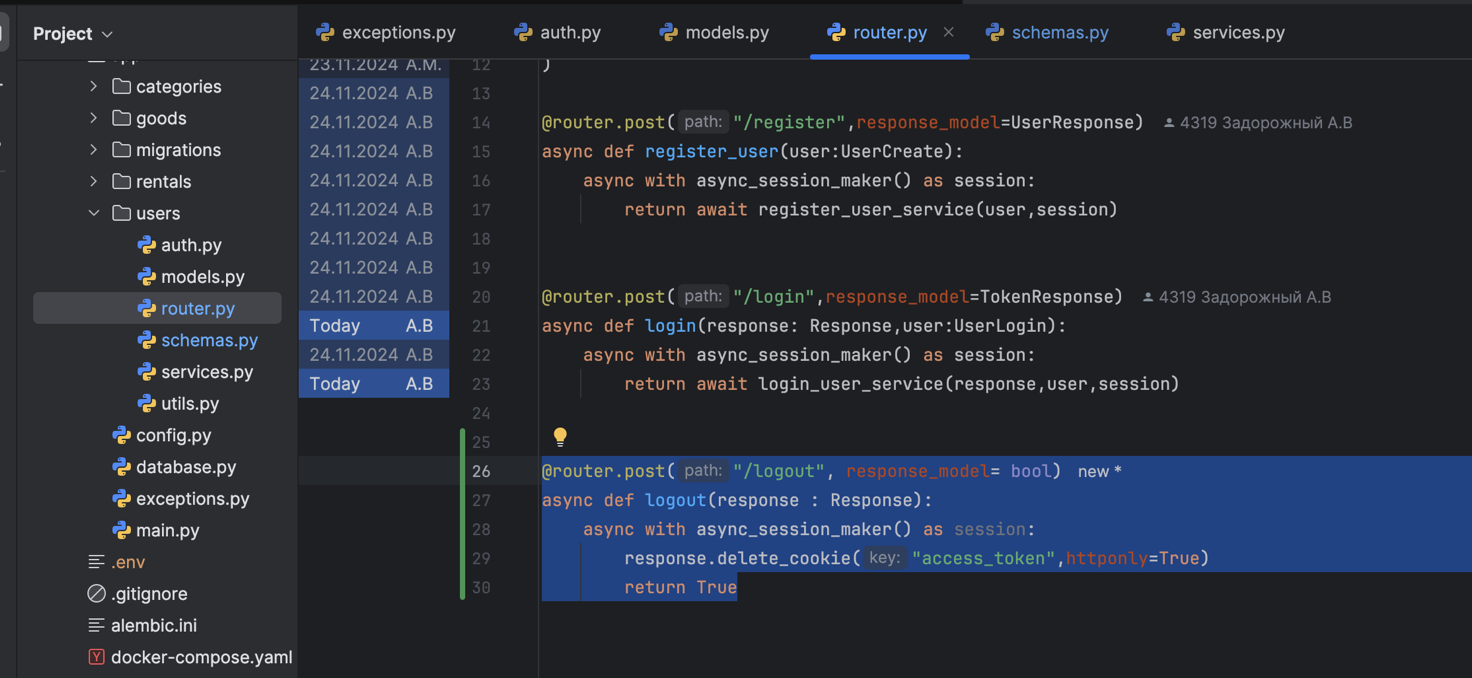
Вынес ошибки в отдельный файл



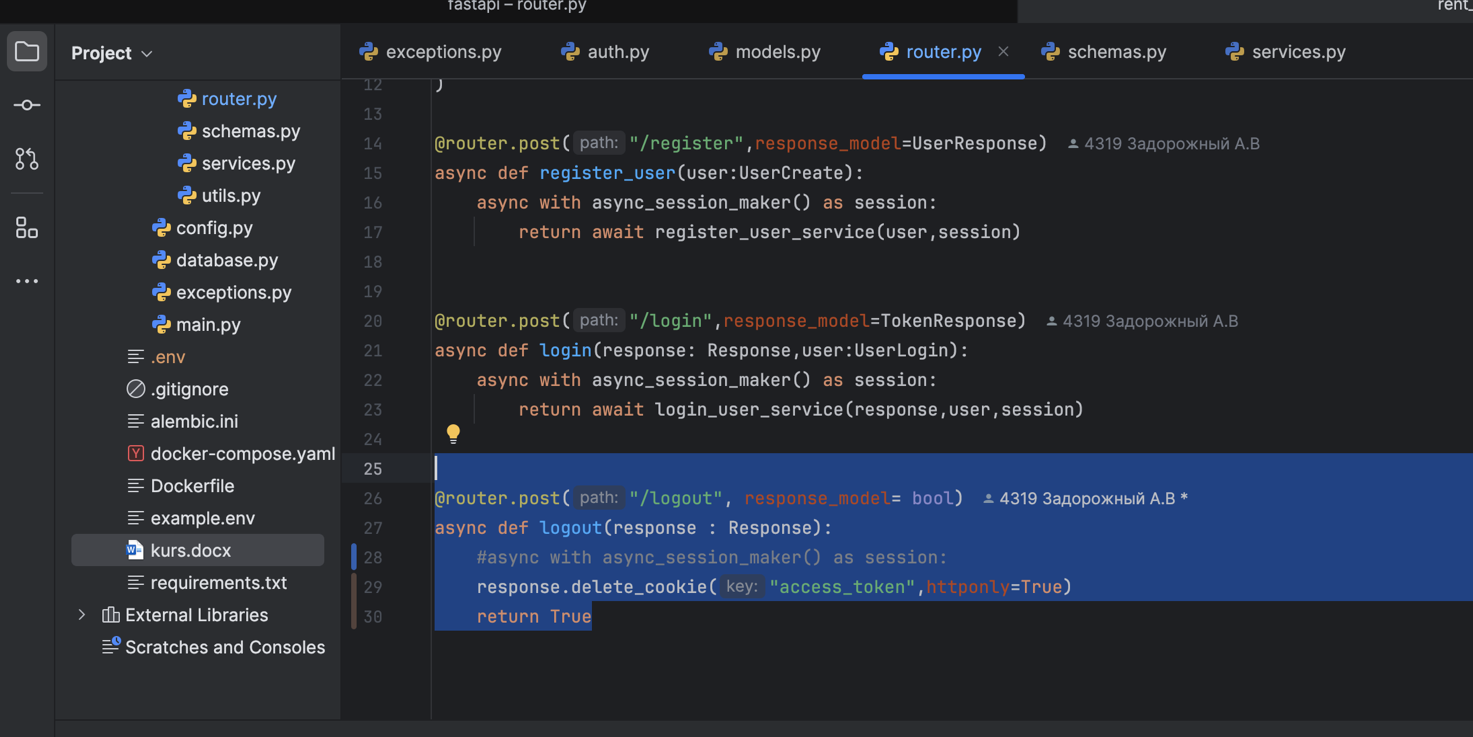
Добавил куки access\_token



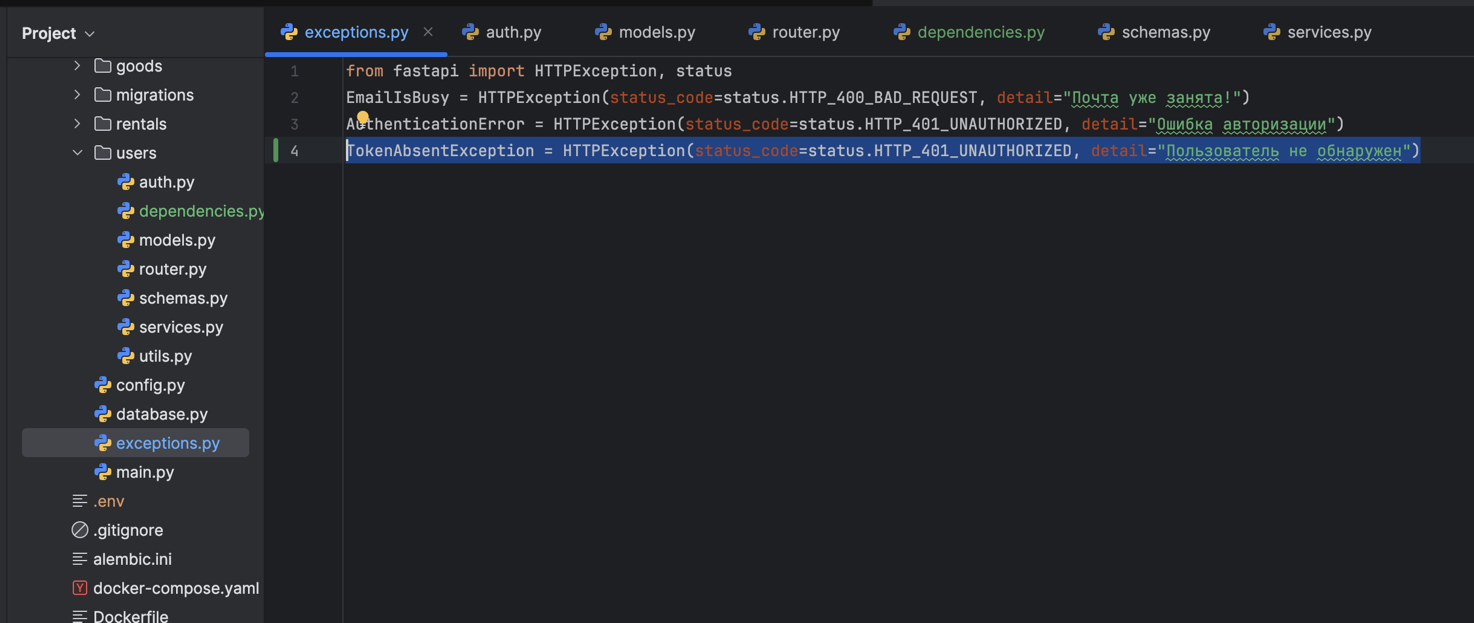
Добавил эндпоинт для логаута



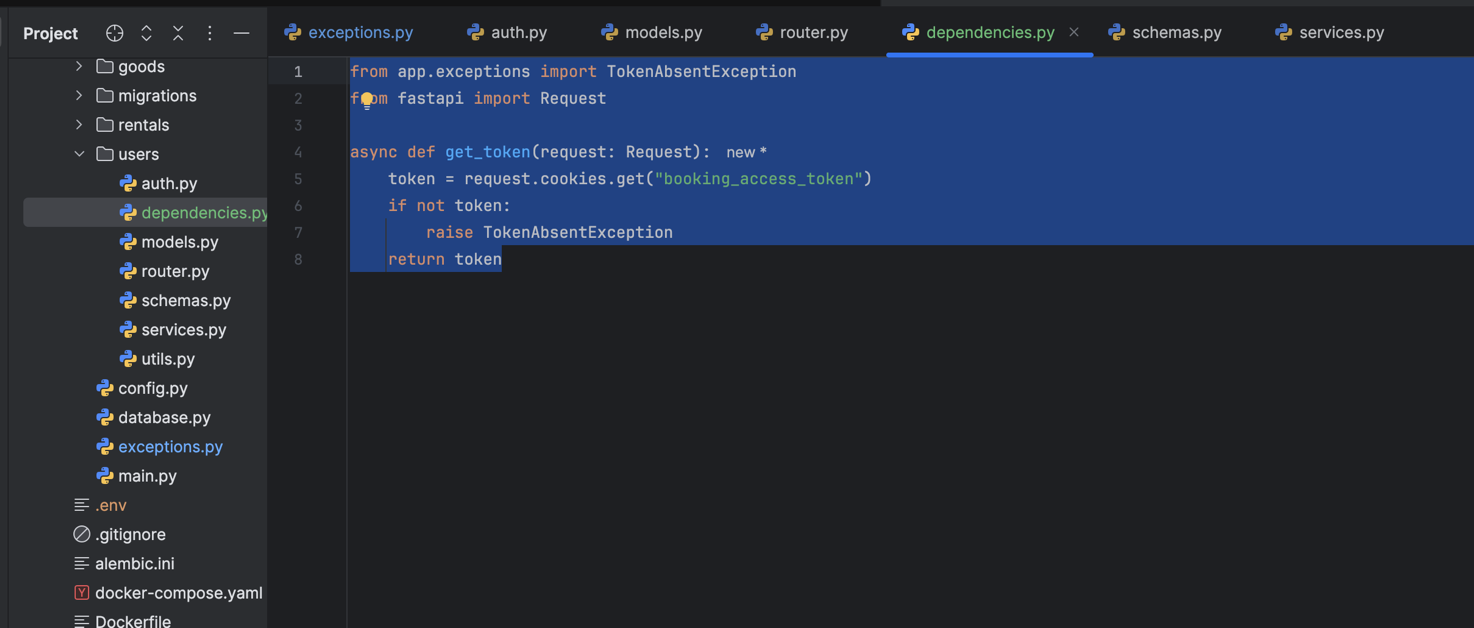
Fix логаута убрал работу с сессиями



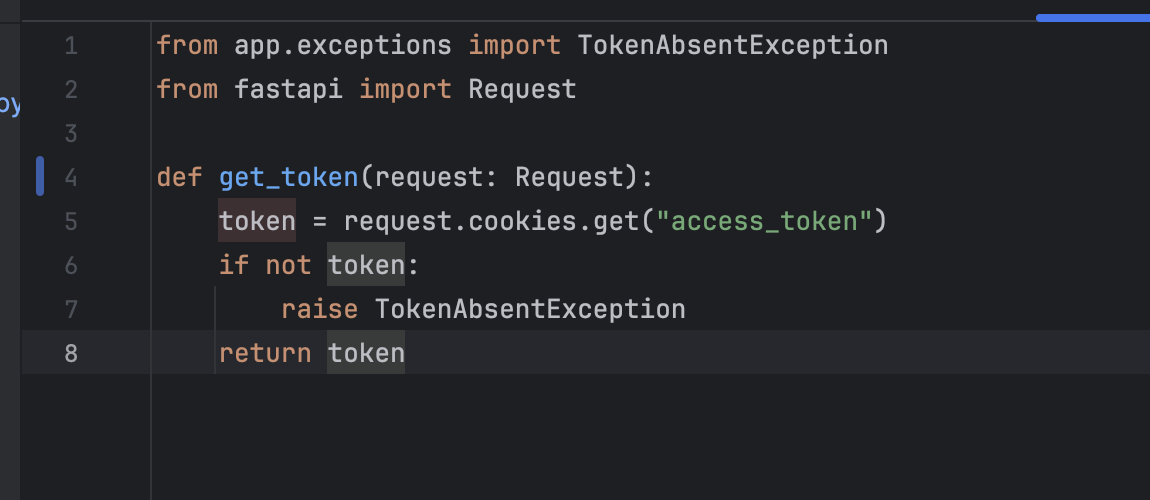
Добавил новую ошибки при поиске токена в куках (для дальнейших авторизации)



Добавил логику для поиска токена

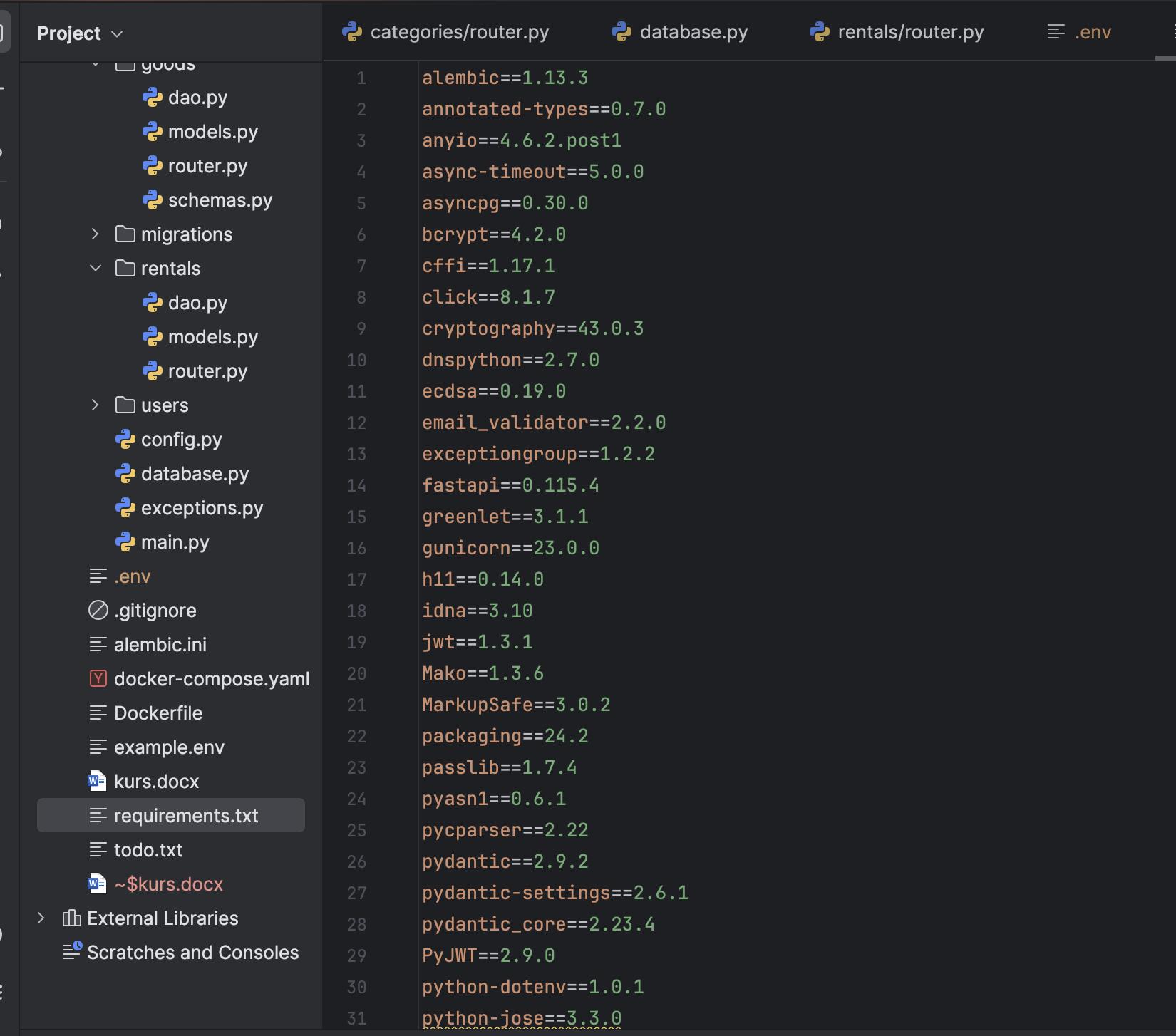


в get\_token() убрал  async

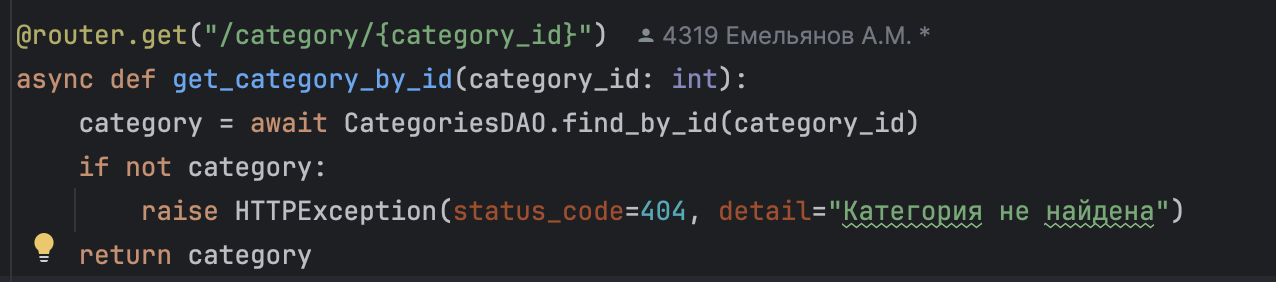




Генерируем список Python-зависимостей



Список всех зависимостей

Если категории нет, вызываем ошибку 404