Flask, Flask WTF, Flask-SQLAlchemy

1. Создайте модели:
   1. Пользователь:

- имя

- роль (админ, обычный)

- баланс

- ставка комиссии

- URL webhook

* 1. Транзакция

- пользователь

- сумма

- комиссия

- статус: ожидание, подтверждена, отменена, истекла

1. Создайте команду flask create-admin для создания дефолтного админа
2. Создайте админку с тремя вкладками: Дашборд(кол-во пользователей, кол-во транзакций, сумма транзакций за текущий день и список последних транзакций), Пользователи(создание, редактирование, удаление), Транзакции(список, детальный просмотр транзакции с возможностью смены статуса на Подтверждена или Отменена, если текущий статус Ожидание)
3. Учтите в админке роль Пользователя (админ может все, пользователь видеть только свои транзакции
4. Создайте API метод /create\_transaction - POST, JSON **(необязательный ID**, сумма), с автоматическим расчетом комиссии
5. Создайте API метод /cancel\_transaction - POST, JSON
6. Создайте API метод /check\_transaction – GET
7. Создайте Swagger для описания API
8. Создайте Celery задачу, которая каждую минуту проверяет транзакции со статусом Ожидание и если прошло более 15 минут с момента создания - поменяйте статус на Истекла и отправьте вебхук на URL webhook (POST, JSON: ид транзакции, статус)

1. Добавьте в админке настраиваемое авто-обновление списков (0 сек, 10 сек, 15 сек, 30 сек, 1 минута)
2. Добавьте в админку фильтр для транзакций (отбор по пользователю, отбор по статусу)

“Сложное задание” (**не обязательно**):

1. Добавьте в модель пользователя поле - номер USDT кошелька
2. Сгенерируйте кошелек в тестовой сети, если он у пользователя не заполнен