

Отчёт по изменениям на бэкенде — 20 декабря 2024

Раздел 1: Исправления за сегодня

Задача #3: Запрет отмены оплаченной поездки

Проблема:

Арендатор мог отменить уже оплаченную поездку напрямую, без участия поддержки.

Решение:

- При попытке отменить оплаченную поездку через `cancel-by-client` — автоматически создаётся обращение в техподдержку с полной информацией о платеже
- При попытке отменить через обычный PATCH-запрос — возвращается ошибка с просьбой обратиться в поддержку
- Поездка остаётся в статусе `started`, решение принимает менеджер

Файлы:

- `chat/views.py` — логика создания обращения в `cancel_by_client`
- `chat/serializers.py` — блокировка отмены через PATCH

Статус: Задеплоено

Задача #6: Отмена заявки арендодателем

Проблема:

При отмене заявки (`RequestRent` → `denied`) арендодателем:

- Уведомление арендатору приходило
- Но связанная поездка (`Trip`) оставалась со статусом `started`
- Платёж (`Payment`) оставался активным
- Арендатор мог оплатить уже отменённую заявку

Решение:

При установке статуса `denied` на `RequestRent` теперь автоматически:

- Отменяется связанный `Trip` (`status = 'canceled'`)
- Отменяется связанный `Payment` (`status = 'canceled'`)

Файл: `chat/views.py` (метод `perform_update` в `RequestRentViewSet`)

Статус: Задеплоено

Задача #12: Комиссия на стоимость доставки

Проблема:

- При сохранении транспорта `price_delivery` накручивалась комиссия (1000 → 1200)
- Стоимость доставки не добавлялась к платежу в Tinkoff

Решение:

- Убрана накрутка комиссии на `price_delivery` — доставка сохраняется как есть
- В платёж Tinkoff добавлена доставка: `total_amount = commission + delivery`
- В чеке Tinkoff две позиции: "Комиссия платформы" и "Доставка транспорта"

Файлы:

- `vehicle/models.py` — убрана накрутка в `Vehicle.save()`
- `payment/views.py` — добавлена доставка к платежу

Итоговая логика:

Параметр	Значение
<code>price_delivery</code> в Vehicle	Сохраняется как есть (1000 → 1000)
<code>payment.amount</code>	Комиссия от аренды (без доставки)
<code>payment.delivery</code>	Стоимость доставки (без комиссии)
Сумма в Tinkoff	<code>amount + delivery</code>

Статус: Задеплоено и проверено на сервере

Раздел 2: Мультивалютность — анализ и предложение

Текущая ситуация

- В базе есть **44 валюты** (RUB, USD, EUR, THB, AED и др.)
- У пользователя есть поле `currency` — валюта для отображения
- У транспорта нет привязки к валюте** — все цены хранятся "как есть"
- Tinkoff принимает оплату только в рублях**

Проблема

Сервис работает в разных странах (Россия, Таиланд, ОАЭ и др.). Нужно:

- Арендодатель создаёт транспорт в своей валюте (например, 1000 AED/день)
- Цена фиксируется в этой валюте и не меняется при колебаниях курса
- Арендатор видит цену в оригинальной валюте + может посмотреть в своей
- При оплате сумма конвертируется в рубли для Tinkoff

Предлагаемое решение

Логика работы



Пример сценария

1. **Арендодатель** создаёт авто: 1000 AED/день
2. **Арендатор** создаёт заявку на 3 дня
3. **Backend**:
 - Сохраняет: `total_cost = 3000 AED`
 - Получает текущий курс: $1 \text{ AED} = 27.2 \text{ RUB}$
 - Рассчитывает: `total_cost_rub = 81 600 RUB`
 - Комиссия платформы: $81\ 600 \times 20\% = 16\ 320 \text{ RUB}$
4. **Tinkoff** получает: 16 320 RUB

Что видит арендатор

Стоимость аренды: 3 000 AED
 $\approx \$817$ (по текущему курсу)

К оплате (комиссия платформы): 16 320 ₽

Уточняющие вопросы

1. **Валюта по умолчанию для существующих транспортов** — считать все текущие цены рублями?
2. **Выбор валюты при создании транспорта** — арендодатель выбирает вручную или определяется автоматически по городу/стране?
3. **Отображение цен для арендатора** — показывать:
 - Только оригинальную валюту транспорта?
 - Оригинал + конвертация в валюту пользователя?
 - Переключатель между валютами?

4. **Депозит** — тоже конвертировать в рубли или оставить информационным (не проходит через Tinkoff)?
 5. **Частота обновления курсов** — раз в час достаточно? Или нужно чаще/реже?
 6. **Источник курсов** — использовать бесплатный API (exchangerate-api.com) или нужен конкретный источник (ЦБ РФ, банк)?
-

Следующие шаги

После получения ответов на вопросы:

1. Добавить поле `currency` в модель Vehicle
2. Добавить поля для фиксации курса в RequestRent
3. Реализовать сервис обновления курсов
4. Обновить API для отображения цен в разных валютах
5. Доработать Flutter для отображения мультивалютности