⚙ Бэкенд — что нового

Изменения сервера (crm-backend2, ветка manager).

- Деактивация аккаунта клиента
- Интеграция с Tilda: приём лидов без дублей и с привязкой к клиенту
- Фикс гонки бронирования (дубли слота)

Деактивация аккаунта клиента

Деактивация аккаунта клиента

Бэкенд

Что нового

- Удаление собственного аккаунта клиента теперь работает как деактивация: аккаунт закрывается и перестаёт быть активным, а не «обезличивается» с потерей истории.
- Если у клиента есть активные будущие занятия, удаление запрещено — сначала нужно отменить или перенести их. Клиент получает понятное пояснение.
- После удаления сессия уничтожается: пользователь выходит из аккаунта.

Как это работает

1. При удалении система проверяет, нет ли у клиента будущих занятий.
2. Если занятия есть — операция отклоняется с понятной ошибкой.
3. Если занятий нет — аккаунт деактивируется, и текущая сессия завершается.

Выгоды

- Требование магазинов приложений (модерация Apple) выполняется.
- История и данные не разрушаются бесследно.
- Нельзя случайно удалить клиента с активными записями.

Интеграция с Tilda: приём лидов без дублей и с привязкой к клиенту

Интеграция с Tilda: приём лидов без дублей и с привязкой к клиенту

Бэкенд

Что нового

- Заявки с форм Tilda теперь корректно превращаются в лид-задачи.
- Повторные заявки с тем же номером (tranid) не создают дублей — даже при повторной доставке вебхука.
- Если tranid отсутствует, вебхук получает понятный ответ об ошибке вместо внутренней ошибки сервера.
- Задача привязывается к собственному клиенту по телефону, а не к «активной карте».
- Работает для всех клубов.

Как это работает

1. Tilda отправляет заявку на вебхук — система проверяет, нет ли уже лида с таким же номером.
2. При повторной доставке второй лид не создаётся.
3. Клиент определяется по телефону; задача привязывается к нему корректно.

Выгоды

- Нет дублей лидов и повторных задач.
- Заявки не теряются и привязываются к правильному клиенту.
- Понятная диагностика, если вебхук пришёл с ошибкой.

□□ Настройка интеграции (вебхук Тильды)

Адрес вебхука — указывается в форме Tilda (Настройки формы → «Вебхук»):

```
POST https://<домен-клуба>/api/v1/rest/callback/tilda-lead?token=<SERVICE_APP_TOKEN>
```

- Метод — **POST**, тело `application/x-www-form-urlencoded` (UTF-8). Формат запроса Tilda формирует сама.
- Роут: `/api/v1/rest/callback/tilda-lead` (полный адрес = домен клуба + роут + `?token=…`).
- Пример (тест): `https://test.tk-champion.ru/api/v1/rest/callback/tilda-lead?token=<токен>` .

Токен — поле `service_app.token` сервисного приложения клуба. В интерфейсе не выводится. Передаётся в адресе `?token=…` либо заголовком `X-Service-App-Token`. Неверный или отсутствующий токен → `401`.

Требования к запросу

- Тело — так, как отправляет Tilda: поля-ответы идут кириллицей (`Имя`, `Телефон`, `Тип_заявки`), служебные поля — латиницей.
- Обязательное служебное поле `tranid` (формат `<projectid>:<id>`) — ключ дедупликации. Запрос без `tranid` (кроме тестового пинга) → понятный ответ об ошибке, а не 500.
- Тестовый пинг Tilda (`test=test`) → `200` без создания задачи.
- Повторная доставка того же `tranid` → `200`, дубль задачи не создаётся (`duplicate: true`).

- Клиент определяется по нормализованному телефону; если не найден — задача создаётся без привязки.

Фикс гонки бронирования (дубли слота)

Фикс гонки бронирования (дубли слота)

Бэкенд

Что было не так

- При одновременных запросах бронирования (например, двойное нажатие кнопки или параллельные запросы) один и тот же слот мог забронироваться дважды.
- В результате появлялись дубли записей.

Что исправлено

1. Добавлены блокировки при бронировании, чтобы параллельные запросы не создавали два слота.
2. Слот привязывается только если он действительно свободен, с проверкой.
3. Предрасчёт стал идемпотентным: повторный запрос не создаёт второй слот.
4. Проверки добавлены во всех сценариях создания, изменения и переноса занятий.

Выгоды

- Дубли записей и «двойные» брони исчезают.
- Данные о занятости кортов остаются достоверными.
- Повторные нажатия и сбои сети больше не создают лишних записей.