

Кейс «Маркетплейс аренды байков»

На острове X есть «Маркетплейс аренды байков», работает через Telegram-бот

На острове Y нет «Маркетплейса аренды байков»

Вы — команда молодых ИТ-специалистов. Временно проживаете на этих островах. Хотите перенести полученный вами классный пользовательский опыт в новые локации

ЦЕЛИ: К чему мы стремимся?

- 1) В течение 1 года привлечь 60% крупных арендодателей территории на нашу платформу
- 2) Через год работы количество сделок, заключённых через платформу > 500
- 3) Через год работы количество уникальных пользователей платформы > 5000
- 4) Количество обращений в службу поддержки не более 2% от количества сделок
- 5) По окончании года CSI не менее 85%
- 6) Обеспечить три альтернативных способа финансовых расчётов за аренду:
 1. Перевод на криптокошелёк платформы
 2. Наличные арендодателю
 3. Перевод на рублёвый счет платформы
- 7) Наше решение должно быть легко переносимым на другие территории

Что является нашим продуктом? Какими услугами мы удовлетворяем потребности клиентов?

Продукт:
Помощь в аренде байков

- Услуги:**
- Дистанционно выбрать байк
 - Арендовать байк
 - Гарантировать честную сделку (роль арбитра)
 - ...

Продукт:
Ещё один канал взаимодействия с арендаторами

- Услуги:**
- Распространять актуальную информацию об арендодателе и его байках
 - Гарантировать честную сделку (роль арбитра)
 - ...

МИССИЯ: Ради чего мы действуем?

Создаём для арендаторов байков удобную и безопасную возможность арендовать большой ассортимент разных байков у честных арендодателей по конкурентным ценам

Общая идея и Happy path

- 1) Потенциальный арендатор через Telegram-бот:
 - Просматривает каталог выставляемых для аренды на маркетплейсе байков
 - Выбирает подходящий ему байк по стоимости аренды, характеристикам, условиям, местоположению, отзывам об арендодателе и т. д.
- 2) Выбрав байк, арендатор оставляет заявку
- 3) Арендодатель и арендатор договариваются обо всех деталях условий сделки через созданный для них персональный чат
- 4) Когда договоренность достигнута:
 - Арендатор оплачивает залог и стоимость аренды
 - Арендатор забирает байк
- 5) По завершению срока аренды:
 - Арендатор возвращает целый и невредимый байк
 - Арендатору возвращается залог

Потребности Арендаторов

- 1) Арендовать подходящий байк по рыночной цене без «надувало́ва»:
 1. Быстро
 2. С удобным способом расчётов
- 2) Получить байк там, где мне удобно
- 3) Сдать байк там, где мне удобно
- 4) Минимизировать общение с арендодателем
- 5) Арендовать байк без залога

Кто наши клиенты?

Какие ещё заинтересованные стороны есть?

- 1) Арендаторы:
 1. Русскоязычные туристы
 2. Русскоязычные, временно проживающие на острове
 3. ...
- 2) Арендодатели:
 1. Крупные арендодатели (от N байков)
 2. ...
- 3) Регуляторы
- 4) Платёжные системы

Кейс «Маркетплейс аренды байков»

Architecture Decision Record. Вариант В

Код	ADR-BIKE-B
Заголовок	Выбор сервиса картографии для приложения по аренде самокатов
Проблема/ контекст	Для приложения по аренде самокатов требуется интеграция с сервисом GIS-карт для отображения маршрутов, точек интереса (POI) и других географических данных.
Цель	Встроить сервис GIS-карт, который будет соответствовать требованиям производительности, надёжности, безопасности и масштабируемости.
Решение	Использовать Google Maps в качестве основного GIS-сервиса.
Отвергнутые варианты	<div>1. Яндекс.Карты</div> <div>2. 2GIS</div> <div>3. Bing Maps</div> <div>4. HERE WeGo</div>
Аргументы/ обоснование	

Кейс «Маркетплейс аренды байков»

Выбор артефактов корпоративной архитектуры

Артефакт	Выбор (да / нет)	Пояснения
1. Архитектурные принципы	+	
2. Архитектурные требования	+	
3. Архитектурные стандарты	+	
4. Архитектурная концепция	+	
5. Референсное архитектурное решение	+	
6. Типовое архитектурное решение	—	
7. Business Capability Map	+	
8. Логическая модель и физическая модель данных	—	
9. Карта приложений	—	
10. Концептуальная архитектура	—	
11. Реестр автоматизированных систем	—	
12. Детальная архитектура сервиса	+	
13. Сетевая архитектура	—	
14. Реестр программных продуктов	—	
15. Реестр архитектурных отклонений	—	
16. Реестр Architecture Decision Record	+	