

АРЕХСТАКЕ

Техническое задание на разработку и развитие Web3-платформы на блокчейне Solana

Проект	ApexStake
Направление	Web3 / Blockchain / DeFi
Blockchain	Solana
Тип продукта	Web3-платформа
Версия документа	1.0
Статус	Техническое задание

1. Цель проекта

Разработать и развивать безопасную, масштабируемую и удобную Web3-платформу ApexStake на блокчейне Solana, объединяющую staking, работу со smart contracts, подключение криптокошельков и другие Web3-инструменты в единой экосистеме.

Платформа должна обеспечивать понятное взаимодействие пользователя с цифровыми активами, прозрачное отображение операций и современный пользовательский интерфейс.

2. Описание проблемы

Пользователям Web3 часто приходится использовать несколько разрозненных сервисов для работы с цифровыми активами, staking и DeFi-инструментами. Дополнительными проблемами являются сложность интерфейсов, недостаточная прозрачность операций и отсутствие единого удобного пользовательского пространства.

ApexStake призван решить эту проблему за счёт объединения ключевых функций Web3 в одной платформе на Solana.

3. Целевая аудитория

- Пользователи Web3 и цифровых активов.
- Пользователи, заинтересованные в staking и DeFi.
- Участники blockchain- и crypto-экосистемы.
- Web3-проекты и партнёры, заинтересованные в интеграции с платформой.

4. Основной функционал

4.1. Пользовательская часть

- Регистрация и авторизация пользователя.
- Подключение популярных Solana-совместимых криптокошельков.
- Личный кабинет с отображением баланса, активностей и истории операций.
- Просмотр доступных staking-продуктов и условий участия.
- Отображение статуса staking-позиции, начислений и сроков.
- Уведомления об основных действиях и событиях платформы.

4.2. Smart Contracts

- Разработка и интеграция smart contracts в экосистеме Solana.
- Реализация логики staking согласно утверждённой бизнес-модели.
- Обработка депозитов, начислений, условий блокировки/разблокировки и вывода в соответствии с утверждённой логикой.
- События и транзакции должны быть проверяемыми в блокчейне.
- Защита от некорректных или несанкционированных операций.

4.3. Административная панель

- Управление пользователями и ролями.
- Управление staking-продуктами и параметрами платформы.
- Мониторинг транзакций и ключевых операций.
- Просмотр базовых продуктовых и технических метрик.
- Журнал действий администраторов.

4.4. Интеграции

- Solana RPC / инфраструктура для взаимодействия с сетью.
- Solana-совместимые кошельки.
- Интеграция с blockchain explorers для проверки транзакций.
- API для внутренних модулей и будущих внешних интеграций.

5. Технологические требования

- Blockchain: Solana.
- Smart contracts: Solana-compatible development stack (конкретный стек согласовывается с технической командой).
- Backend: современный серверный стек с REST API и/или GraphQL при необходимости.
- Frontend: современный web framework с адаптивным интерфейсом.
- Database: реляционная или документная БД в зависимости от архитектуры.
- Infrastructure: облачная инфраструктура с возможностью горизонтального масштабирования.
- Source control: Git.
- Разделение dev / staging / production окружений.

6. Требования к безопасности

- Безопасная работа с wallet-подключениями и подписью транзакций.
- Приватные ключи пользователей не должны храниться на сервере.
- Разграничение прав доступа в административной панели.
- Защита API от распространённых атак и злоупотреблений.
- Логирование критических действий.
- Проведение тестирования smart contracts до production-развёртывания.
- Перед запуском production рекомендуется независимый security audit smart contracts.

7. Нефункциональные требования

- Адаптивность: desktop, tablet и mobile.
- Высокая доступность и устойчивость к сбоям.
- Масштабируемая архитектура для роста количества пользователей и транзакций.
- Оптимизированная скорость загрузки основных страниц.
- Понятный UX для пользователей без глубоких технических знаний.
- Мониторинг ошибок, инфраструктуры и blockchain-интеграций.

8. Этапы реализации

- Этап 1 — аналитика, уточнение бизнес-логики и проектирование архитектуры.
- Этап 2 — UX/UI-дизайн и подготовка технических прототипов.
- Этап 3 — разработка smart contracts и их тестирование.
- Этап 4 — разработка frontend, backend и личного кабинета.
- Этап 5 — интеграция Solana и криптокошельков.
- Этап 6 — разработка административной панели.
- Этап 7 — QA, security testing и исправление ошибок.
- Этап 8 — deployment в production и техническая документация.
- Этап 9 — сопровождение и дальнейшее развитие продукта.

9. Ожидаемый результат

- Рабочая Web3-платформа ApexStake на Solana.
- Рабочие и протестированные smart contracts.
- Web-интерфейс пользователя и личный кабинет.
- Подключение Solana-кошельков.

- Staking-модуль согласно утверждённой бизнес-логике.
- Административная панель.
- API и техническая документация.
- Исходный код и инструкции по развёртыванию.
- Тестовые отчёты и результаты проверки безопасности.

10. Критерии приёмки

- Основные пользовательские сценарии работают без критических ошибок.
- Транзакции корректно формируются, подписываются и подтверждаются в сети Solana.
- Данные о blockchain-операциях корректно отображаются в интерфейсе.
- Staking-логика соответствует утверждённой спецификации.
- Критические уязвимости не выявлены на этапе security testing.
- Проект сопровождается исходным кодом и необходимой документацией.

11. Важное примечание

Финальные параметры staking, токеномики, комиссий, reward-механики, ролей пользователей и юридических условий должны быть отдельно согласованы до разработки production-версии smart contracts. Техническая реализация не должна подразумевать гарантированную доходность или финансовый результат для пользователей.

Документ подготовлен как базовая техническая спецификация для размещения и поиска технической команды в Astana Hub. Детальная архитектура, сроки, бюджет и конкретный технологический стек согласовываются с исполнителем после технического обследования проекта.