

Требования к проекту «Crypto Currency Bot»

Содержание

Содержание	1
1 Введение	3
1.1 Основные функции	3
2 Общие требования к представлению проекта	4
2.1 Управление версиями и сотрудничество	4
2.2 Целевые артефакты	4
3 Функциональные требования	5
3.1 Аутентификация пользователей	5
3.2 Настройки	5
3.3 Отслеживание валюты	5
3.4 Взаимодействие с портфелем	5
3.5 История отслеживания	6
3.6 Формирование уведомлений	6
3.7 Источник данных для криптовалют	6
4 Общие требования	8
4.1 Производительность	8
4.2 Надёжность	8
4.3 Безопасность	8
4.4 Роли пользователей	9
4.5 Авторизация	9
4.6 Конечные точки	10
5 Требования к технологическому стеку	11
6 Требование к тестированию	12
7 Этапы выполнения проекта	13
7.1 Составление требований	13
7.2 Разработка архитектуры	13
7.3 Полный проект	13
8 Глоссарий	14
8.1 Основные термины	14

8.2 Технические термины	14
8.3 Предметная область	16
8.3.1 Общие термины	16
8.3.2 Фиатные валюты	16
8.3.3 Криптовалюты	16
8.4 Процесс разработки	17
9 Список команд	18

1 Введение

Целью проекта является разработка Telegram-бота для отслеживания текущей цены для некоторой криптовалюты, изменения цены за некоторый период, подсчета текущей цены портфеля пользователя.

1.1 Основные функции

Проект должен реализовывать следующие функции:

- Установка отслеживаемой криптовалюты и фиатной валюты, для отображения цены.
- Просмотр текущей цены криптовалюты
- Просмотр изменения цены криптовалюты за период определяемый пользователем.
- Добавление криптовалют в портфель
- Отслеживание цены портфеля за указанный период
- Вычисление прибыли/убытка полученного за указанный период, для каждой криптовалюты из портфеля
- Установка цены криптовалюты, при превышении которой, будет поступать уведомление
- Формирование уведомлений на основе значительного изменения цены отслеживаемых криптовалют
- Создавать список актуальных уведомлений на данный момент.
- Очищать список уведомлений, переставших быть актуальными.

2 Общие требования к представлению проекта

2.1 Управление версиями и сотрудничество

По отношению управления версиями и сотрудничеству необходимо выполнить следующие требования:

- Весь код должен быть зафиксирован в репозитории GitHub
- Репозиторий должен включать файл README.md с подробной инструкцией по:
 - Настройке проекта локально.
 - Сборке и запуску приложения.
 - Развертыванию приложения с использованием Docker.
 - Взаимодействию с Telegram-ботом.

2.2 Целевые артефакты

- Приложение должно создавать исполняемый fat JAR в качестве основного артефакта.
- Приложение должно быть развернуто в виде Docker-контейнера.
- Для сборки, упаковки и управления зависимостями использовать инструмент сборки Gradle.

3 Функциональные требования

3.1 Аутентификация пользователей

- Аутентификацию пользователя реализовать по ChatID пользователя. Дополнительная регистрация не требуется.

3.2 Настройки

Общая настройка взаимодействия пользователем должна включать:

- Установка фиатной валюты для перевода цены криптовалюты.
 - Представление цены криптовалюты в виде фиатной валюты устанавливается единым образом для всех активов, используемых для отслеживания.
 - Выбор фиатной валюты предоставляется из заранее определенного списка. В качестве доступных валют для представления цены используются резервные валюты (доллар США, евро, японская иена, фунт стерлингов), а также несколько дополнительных (российский рубль, китайский юань).

3.3 Отслеживание валюты

Для отслеживания валюты необходимо реализовать:

- Запрос у пользователя криптовалюты для отслеживания
 - Если пользователь не указал криптовалюту добавлять криптовалюту по умолчанию (BTC) в отслеживание.
 - Выбор криптовалюты предоставляется из заранее определенного списка. В качестве доступных криптовалют представлены: BTC, ETH, SOL, XRP, ADA, DOGE, AVAX, NEAR, LTC.
 - Если добавляемая валюта уже находится в отслеживании, выдавать соответствующее сообщение и возвращаться на главный экран.
- Представлять отчет об изменении цены отслеживаемой валюты за определенный период/за все время отслеживания.

3.4 Взаимодействие с портфелем

Для взаимодействия с портфелем пользователя реализовать:

- Создание пустого портфеля
 - При создании портфеля
 - Добавлять валюту в портфель
 - При добавлении требовать у пользователя количество монет, которое в наличии у пользователя, и криптовалюту помещенную

для отслеживания. В случае, если пользователь не предоставляет данную информацию выдавать сообщение об ошибке и возвращаться на главный экран.

- Вычисление общей стоимости портфеля на данный момент
- Отслеживание цены портфеля за указанный пользователем период
- Предоставление изменения цены портфеля за определенный период(день/три дня/неделя/месяц)/за все время отслеживания.
 - Показывать, насколько цена выросла/убыла с начала периода.

3.5 История отслеживания

Для хранения информации о состоянии рынка необходимо:

- Хранить следующую информацию для вычисления текущей цены:
 - Дата в формате YYYY-MM-DD HH:MM:SS.
 - Название криптовалюты.
 - Цена в долларах США.
- По истечении срока удалять записи о цене валюты каждый день.
 - Срок хранения информации о текущей цене - 30 дней.

3.6 Формирование уведомлений

Пользователю доступна установка трёх разных видов уведомлений на каждую валюту, а именно:

- Превышение цены за единицу криптовалюты установленного пользователем значения.
- Превышение цены за единицу криптовалюты в процентном соотношении, установленного пользователем значения
- Сильное изменение цены криптовалюты (как вверх, так и вниз).
 - Для определение сильного изменения цены используется пересечение экспоненциальной средней скользящей (сверху вниз - понижение цены, снизу вверх - увеличение цены).

Для дополнительного взаимодействия с уведомлениями реализовать:

- Формирование списка актуальных уведомлений для пользователя на данный момент.
- Удаление уведомлений из списка при потере актуальности уведомлений.
 - Уведомления считаются актуальными, если сохраняется условия поступления уведомления (превышения установленного значения, продолжение изменения в соответствующем направлении после сильного изменения цены).

3.7 Источник данных для криптовалют

Для получения данных о курсе криптовалют использовать платформу Bingx

4 Общие требования

4.1 Производительность

4.1.1 Время отклика API:

- 95% запросов должны обрабатываться ≤ 750 мс.
- Критические операции (цена криптовалюты, алерты) – ≤ 400 мс.

4.1.2 Пропускная способность:

- Система должна выдерживать ≥ 20 запросов в секунду в пиковой нагрузке. Допустимая задержка: ≤ 1 сек для 95% запросов.
- Для /healthcheck – ≥ 100 запросов в секунду. Допустимая задержка: ≤ 300 мс.

4.1.2 Задержка Kafka: обработка сообщений в алертах – ≤ 1.5 сек от момента публикации до уведомления.

4.1.3 Лимиты:

- Телеграм-бот: 10 сообщений/секунду на бота, 100 активных алертов/пользователь.
- Внешние API: 10 запросов/минуту
- Mongo DB: 500 операций записи/секунду на инстанс
- Поддержка до 100.000 пользователей одновременно

4.2 Надёжность

Автоматические повторные попытки при сбоях.

- Для внешних API: 3 попытки с экспоненциальной задержкой (1с, 2с, 4с).
- Для операций с БД: 2 попытки с фиксированной задержкой 1с

4.3 Безопасность

- Хранение внешних API-ключей должны храниться в зашифрованном виде в переменных окружения
- Передача паролей, логинов, URL-адресов через защищенные переменные окружения.

- Проверка запросов с криптовалютами и фиатными валютами через регулярные выражения
- Валидация входящих параметров на корректность
- Защита запросов к базе данных от инъекций
- API ключи должны обновляться каждые 30 дней

4.4 Роли пользователей

Роль	Идентификация	Доступные действия
Пользователь (User)	Определяется по chat_id в Telegram	Взаимодействие с ботом через заданные команды
Администратор (Admin)	Авторизация через ключ API	<ul style="list-style-type: none"> - Доступ к эндпоинту /healthcheck - Доступ к эндпоинту /users (просмотр всех пользователей) - Доступ к системным логам

4.5 Авторизация

4.5.1 Авторизация пользователей

1. Пользователь начинает диалог с ботом командой /start
2. Бот получает chat_id пользователя.
3. Система проверяет наличие chat_id в БД:

- Если есть → авторизация успешна.
- Если нет → создается новая запись в БД.

4.5.2 Авторизация администратора

Требования:

- chat_id не должен передаваться в открытом виде в URL или логах.
- Все команды бота (кроме /start) требуют предварительной авторизации.

1. Администратор отправляет запрос с заголовком Authorization: Bearer <API_KEY>
2. Сервер:
 - Расшифровывает ключ из БД.
 - Проверяет его валидность и срок действия.
 - Предоставляет доступ к /admin-эндпоинтам.

Требования:

- Ключи хранятся в БД в зашифрованном виде.
- Срок действия ключа — 30 дней

4.6 Конечные точки

4.6.1 healthcheck

- Не требует авторизации по HTTP
- Возвращает: статус сервера

4.6.2 users

- Доступна только для администраторов
- Возвращает список всех пользователей

5 Требования к технологическому стеку

- **Язык программирования:** Java 23.
- **Фреймворк:** Spring 6.2.3 (WebFlux, JPA, Modulith).
- **Telegram API:** Приложение должно интегрироваться с Telegram API для предоставления функциональности бота.
- **Брокер сообщений:** Kafka 3.7.0.
- **База данных:** MongoDB 8.0.4.
- **Контейнеризация:** Docker, Docker Compose.
- **Логирование:** Log4j, SLF4J
- **Тестирование:** JUnit 4

6 Требование к тестированию

В ходе работы должны быть реализованы unit-тесты и интеграционные тесты для всех модулей.

7 Этапы выполнения проекта

7.1 Составление требований

Требуется представить подробный документ с требованиями, который описывает:

- Назначение и функциональность приложения.
- Ключевые функции Telegram-бота.
- Нефункциональные требования (например, производительность, масштабируемость и т.д.).
- Любые допущения или ограничения.

7.2 Разработка архитектуры

Требуется представить документ архитектуры, который включает:

- Обзор системы на высоком уровне.
- Диаграммы (например, диаграммы компонентов, диаграммы последовательностей или блок-схемы) для иллюстрации архитектуры.
- Объяснение выбранного технологического стека и его обоснование.
- Описание того, как приложение будет собираться, развертываться и запускаться.

7.3 Полный проект

Предоставить полный проект в виде:

- Ссылки на репозиторий GitHub с полной кодовой базой.
- Ссылки на Docker Hub, где опубликован Docker-образ.
- Ссылки или имя пользователя Telegram-бота для тестирования и взаимодействия.

8 Глоссарий

8.1 Основные термины

Термин	Определение
Telegram	Кроссплатформенный мессенджер с поддержкой облачного хранения сообщений
Telegram-бот	Автоматизированная программа, работающая внутри Telegram и обрабатывающая команды пользователей
Chat ID	Уникальный числовой идентификатор чата, группы, канала или диалога с пользователем в Telegram, используемый для аутентификации
Криптовалюта	Цифровая или виртуальная валюта, использующая криптографию для обеспечения безопасности транзакций.
Фиатная валюта	Не обеспеченные драгоценными металлами деньги, номинальная стоимость которых устанавливается и гарантируется государством вне зависимости от стоимости изготовления
Курс криптовалюты	Текущая стоимость криптовалюты, выраженная в фиатной валюте

8.2 Технические термины

Термин	Определение
Git	Распределённая система контроля версий для отслеживания изменений в файлах и координации работы над проектом среди нескольких участников
GitHub	Платформа для хостинга и совместной разработки проектов с использованием системы контроля версий Git.
Репозиторий GitHub	Хранилище кода, где фиксируется весь код проекта и история изменений
README.md	Файл с инструкциями по настройке, сборке, запуску и развёртыванию приложения
Fat JAR	Исполняемый файл Java, включающий все необходимые зависимости для запуска приложения без дополнительных библиотек
Контейнеризация	Технология, позволяющая упаковывать приложения и их зависимости в изолированные контейнеры, обеспечивая их переносимость и согласованность в различных средах
Docker	Платформа для разработки, доставки и запуска приложений в контейнерах, которые позволяют изолировать приложения от их окружения
Docker-контейнер	Изолированный пакет, содержащий приложение и все его зависимости
Docker Compose	Инструмент для определения и запуска многоконтейнерных Docker-приложений, позволяющий управлять конфигурацией и зависимостями между контейнерами с помощью YAML-файлов
Docker Hub	Облачный реестр для хранения и распространения Docker-контейнеров

YAML-файл	Текстовый файл, использующий формат YAML (YAML Ain't Markup Language), предназначенный для удобного описания данных и конфигураций, легко читаемый как человеком, так и машиной
БД (база данных)	Структурированная система хранения данных для эффективных организации, управления и извлечения информации
API	Набор инструментов для взаимодействия между программами, позволяющий обмениваться данными и функциональностью
API-ключ	Уникальный идентификатор, который используется для аутентификации и авторизации при доступе к API
Telegram API	Интерфейс для создания Telegram-ботов и интеграции функций Telegram в приложения
Фреймворк	Комплекс готовых решений и инструментов, который предоставляет разработчикам стандартизированные подходы и библиотеки для ускорения процесса создания программного обеспечения
Spring Framework 6	Платформа для разработки Java-приложений, предоставляющая инструменты для создания веб-приложений и микросервисов
Reactive Spring (WebFlux)	Модуль Spring Framework, предназначенный для создания реактивных, неблокирующих веб-приложений, использующий реактивное программирование
MongoDB	NoSQL база данных, предназначенная для хранения данных в формате документов
Инстанс	Отдельная установка или экземпляр базы данных, который работает на сервере
Брокер сообщений	Программное обеспечение или система, которая управляет обменом сообщений между различными приложениями, сервисами или компонентами в распределенной архитектуре
Kafka	Распределённый программный брокер сообщений
Spring JPA	инструмент Spring Framework для упрощения взаимодействия с реляционными базами данных
ORM	Технология, позволяющая преобразовывать данные между несовместимыми типами систем, например между объектно-ориентированными языками программирования и реляционными базами данных
Spring Modulith	Инструмент для структурирования монолитных приложений, позволяющий организовать код в модули
HTTP	Протокол, определяющий правила и форматы для передачи информации в интернете, позволяя различным устройствам обмениваться данными, обычно в виде текстов, изображений и других ресурсов.
Токен	Уникальный идентификатор или строка данных, используемая для аутентификации и авторизации в системах и приложениях
Конечная точка (endpoint)	Конкретный URL-адрес или точка доступа в веб-сервисе, через которую клиентское приложение взаимодействует с сервером для выполнения запросов или получения данных.

8.3 Предметная область

8.3.1 Общие термины

Портфель	Совокупность купленных инвестором цифровых активов
Оповещение (алерт)	Уведомления, отправляемые пользователю при достижении криптовалютой определенного уровня цены или наступлении важного рыночного события
Тренды криптовалюты	Устойчивые направления движения цены актива), отражающие преобладающую рыночную динамику в определённый период времени.
Новость	Информация о событиях, фактах или изменениях, способных повлиять на цену и спрос на криптовалюты
Объем торгов	Показатель, который отражает количество криптовалют, которые были куплены и проданы за определенный период времени
Капитализация рынка	Общая стоимость всех криптовалют, находящихся в обращении
Экспоненциальная скользящая средняя (EMA)	Технический индикатор, который показывает среднюю цену на актив за определенный период времени и придает больший вес недавним ценовым данным

8.3.2 Фиатные валюты

USD	Доллар США
GBP	Фунт стерлингов
EUR	Евро
CNY	Китайский юань
RUB	Российский рубль
JPY	Японская иена

8.3.3 Криптовалюты








BTC	Bitcoin
ETH	Ethereum
SOL	Solana
MATIC	Polygon (ранее Matic)

XRP	Ripple
ADA	Cardano
DOGE	Dogecoin
AVAX	Avalanche
NEAR	NEAR Protocol
LTC	Litecoin

8.4 Процесс разработки

Пропускная способность	Количество запросов, которые система может обработать за единицу времени.
Роли пользователей	Категории пользователей с различными уровнями доступа и правами в системе.
Авторизация	Процесс проверки прав доступа пользователя к определенным ресурсам или действиям
Эндпоинт Healthcheck	Специальный эндпоинт, который используется для проверки состояния и работоспособности сервера или приложения
Логирование	Процесс систематической записи событий и действий, происходящих в системе или приложении, с целью мониторинга, анализа и устранения проблем
Валидация	Процесс проверки данных на соответствие заданным критериям.
Регулярное выражение	Формальный язык, используемый в компьютерных программах, работающих с текстом, для поиска и осуществления манипуляций с подстроками в тексте, основанный на использовании метасимволов.
Инъекция	Атака, при которой злоумышленник пытается вставить вредоносный код в запросы к базе данных.
Переменная окружения	Значение, которые используются в различных командах и программных сценариях, выполняемых в операционной системе
Экспоненциальная задержка	Метод, используемый для управления повторными попытками выполнения операций, при котором время ожидания между попытками увеличивается экспоненциально
Аутентификация по схеме Bearer	Схема аутентификации HTTP, которая использует токены безопасности, называемые токенами-носителями (bearer tokens).

9 Список команд

№	Команда	Формат	Пример	Описание
1	/start	/start	/start	Инициализация бота
 Добро пожаловать в BitBotX — вашего надежного помощника в мире криптовалют! С моей помощью вы сможете: <ul style="list-style-type: none"> ♦ Узнавать актуальные цены на BTC, ETH и другие криптовалюты. ♦ Анализировать историю цен за выбранный период. ♦ Сравнивать курсы разных активов. ♦ Управлять портфелем: добавлять криптовалюты, отслеживать прибыль и убытки. ♦ Настраивать оповещения о важных изменениях цен.  Нужна помощь? Напишите /help , и я расскажу, как работать с моими функциями!  Давайте начнем! Просто выберите нужную команду или введите интересующий вас запрос.				
2.	/help	/help	/help	Вывод описания функций и соответствующих им команд
 BitBotX Help Center <i>Ваш гид по возможностям крипто-бота</i> <ul style="list-style-type: none"> ♦ Основные команды: <ul style="list-style-type: none"> ▶ /start - Начало работы с ботом ▶ /help - Это справочное меню ▶ /authors - список авторов ▶ /set_fiat <валюта> - Настройка фиатной валюты по умолчанию ▶ /set_crypto <криптовалюта> - Настройка криптовалюты по умолчанию  Мониторинг цен: <ul style="list-style-type: none"> ▶ /show_current_price - Текущий курс валюты по умолчанию ▶ /show_price_history <период> - История цены (3d/7d/1M) ▶ //compare_currency <валюта 1> <валюта 2> <период> - Сравнение криптовалют (3h/12h/24h - часы, 3d/7d/30d - дни)  Портфель: <ul style="list-style-type: none"> ▶ /add <количество> <валюта> - Добавить актив ▶ /portfolio - Просмотр портфеля  Оповещения: <ul style="list-style-type: none"> ▶ /set_alert_EMA <валюта> - По индикатору ▶ /set_alert_VAL <валюта> <валюта> <максимальная цена> <минимальная цена> - По цене ▶ /set_alert_PERC <валюта> <максимальный прирост к изначальному значению в процентах> <максимальный убыток от начального значения в процентах> - По процентам ▶ /my_alerts - Активные оповещения ▶ /delete_alert_ <тип оповещения> <валюта> - Удалить оповещение ▶ /delete_all_alerts - Удаление всех оповещений 				

**Быстрые подсказки:**

1. Все команды можно вводить как с "/"
2. Для точных значений: 0.00000001 BTC

Настройка бота

3

/set_fiat

/set_fiat <валюта>

/set_fiat USD

Выбор фиатной валюты по умолчанию

Успех:**Фиатная валюта обновлена!**Теперь все цены будут отображаться в **USD**.**Неудача:***Валюта «XYZ» недоступна. Пожалуйста, выберите из списка: USD, EUR, JPY, GBP, RUB, CNY*

4

/set_crypto

/set_crypto
<криптовалюта>

/set_crypto BTC

Выбор криптовалюты по умолчанию

Успех:**Криптовалюта по умолчанию обновлена!**Теперь базовой валютой для аналитики и сравнений выбран **BTC**.**Неудача:***Валюта «XYZ» не поддерживается. Доступные варианты: BTC, ETH, SOL, MATIC, XRP, ADA, DOGE, AVAX, NEAR, LTS.***Анализ биржи**

5

/compare_currency

/compare_currency
<валюта 1> <валюта 2>
<период>/compare_currency BTC
SOL 7d

Сравнение цен двух криптовалют за указанный период

Успех:**СРАВНЕНИЕ КУРСОВ (7 дней)****BTC (Bitcoin)**

Цена: \$63,420.50







Капитализация: \$1.24T




Объём (24ч): \$28.5B


**SOL (Solana)**

 Цена: \$142.16
 Капитализация: \$63.8B
 Объём (24ч): \$3.2B
 Изменение за период:
BTC: ▲ +5.4% (был \$60,100)
SOL: ▼ -3.1% (был \$146.80)

 Соотношение BTC/SOL:
1 BTC = 446.2 SOL
(7 дней назад: 1 BTC = 410.5 SOL)
Изменение: ▲ +8.7%





Неудача:

 Валюта «XYZ» не поддерживается. Доступные варианты: BTC, ETH, SOL, MATIC, XRP, ADA, DOGE, AVAX, NEAR, LTC.

 Неверный формат периода. Доступно:

3h, 12h, 24h - часы
3d, 7d, 30d - дни

6	/show_current_price	/show_current_price	/show_current_price	Показать текущую цену криптовалюты по умолчанию
---	---------------------	---------------------	---------------------	---

 Текущий курс криптовалюты
 BTC (Биткоин):
 \$63,420.50 (по курсу USD)
 Обновлено: Сегодня, HH:MM (UTC+3)

7	/show_price_history	/show_price_history <ПЕРИОД>	/show_price_history 3d	Показать историю цен криптовалюты по умолчанию за период
---	---------------------	---------------------------------	------------------------	--

Успех:

 Дата | Цена (USD)


2023-12-01 | \$38,120
2023-12-02 | \$37,890
2023-12-03 | \$38,450

 Изменение за период:
+1.2% (с 38,120 до 38,450)

 Минимум: 37,890 | Максимум: 38,600

 Данные обновлены: Сегодня, 15:30

Неудача:


 Неверный формат периода. Доступно:

3h, 12h, 24h - часы

3d, 7d, 1M - дни

Работа с портфелем

8	/add	/add <количество> <валюта>	/add 0.42 ETH	Добавление указанного количества валюты в портфель
---	------	-------------------------------	---------------	--










 Успешно добавлено:



Количество	0.42 ETH
Текущая цена	\$1,580.72
Сумма	\$663.90

 Портфель обновлен:

- ETH: 0.42 (\$663.90)
- Общий баланс ETH: 1.25 (\$1,975)

9	\set_alert_<тип>	<p>По EMA: \set_alert_EMA <валюта></p> <p>По установленному значению цены: \set_alert_VAL <валюта> <максимальная цена> <минимальная цена></p> <p>По процентному соотношению: \set_alert_PERC <валюта> <максимальный прирост к изначальному значению в процентах></p>	<p>\set_alert_EMA BTC</p> <p>\set_alert_VAL BTC 65000 63000</p> <p>\set_alert_RERC BTC 25 5</p>	Установить оповещение о резком изменении цены валюты по показателю EMA/по установленному значению/ по процентному соотношению
---	------------------	--	---	---

		<Максимальный убыток от начального значения в процентах>		
<p>Успех:</p> <p> Оповещение по ЕМА установлено!</p> <p>Тип: ЕМА (скользящее среднее) Валюта: BTC Триггер: при пересечении ценой 20-дневного ЕМА</p> <p> Я сообщу, когда цена BTC:</p> <ul style="list-style-type: none"> - Прорвёт ЕМА снизу вверх (сигнал к покупке) - Упадёт ниже ЕМА (сигнал к продаже) <hr/> <p> Оповещение создано!</p> <p>Валюта: BTC</p> <ul style="list-style-type: none"> ▫ Верхний лимит: \$65,000 ▫ Нижний лимит: \$63,000 <p>Я пришлю уведомление, если:</p> <ul style="list-style-type: none"> - BTC достигнет \$65,000  - BTC упадёт до \$63,000  <p> Процентное оповещение активировано!</p> <p>Валюта: BTC</p> <ul style="list-style-type: none"> ▫ Макс. рост: +25% от текущей цены ▫ Макс. падение: -5% от текущей цены <p>Уведомлю при:</p> <ul style="list-style-type: none"> - Росте на 25% - Падении на 5% <hr/> <p>Неудача: (пример)</p> <p> Проценты должны быть положительными! Пример: /set_alert_PERC BTC 20 10 (первое число > второго)</p>				
10	/my_alerts	/my_alerts	/my_alerts	Показать активные оповещения
 ВАШИ АКТИВНЫЕ ОПОВЕЩЕНИЯ 				

 Последнее обновление: 15.03.2023 14:30 UTC+3
 Всего активных алертов: 3

1 BTC

— Тип: Процент
— Рост/падение: 25% / 5%
— Установлено: 14.03.2025 12:34:56
— Старт: \$65,000

2 ETH

— Тип: Цена
— Max/min: \$1,500 / \$1,450
— Установлено: 10.03.2025 12:00:00

3 SOL

— Тип: ЕМА
— Старт: \$500,000,000
— Установлено: 12.03.2025 23:59:59


11	<code>\delete_alert_<тип></code>	<code>\delete_alert_<тип></code> <code><валюта></code>	<code>\delete_alert_EMA BTC</code>	Удалить активное оповещение по названию валюты и типу оповещения
----	--	---	------------------------------------	--

Успех:

 Оповещение удалено!

Тип: ЕМА

Валюта: BTC

Статус:  Деактивировано

Неудача:

 Оповещение не найдено

По вашему запросу:













Тип: ЕМА

Валюта: BTC

Возможные причины:

- Оповещение уже удалено
- У вас нет активных ЕМА-алертов для BTC
- Ошибка в типе (доступно: ЕМА, VAL, PERC)

12	<code>/delete_all_alerts</code>	<code>/delete_all_alerts</code>	<code>/delete_all_alerts</code>	Удаление всех
----	---------------------------------	---------------------------------	---------------------------------	---------------

				оповещений
<p>Успех:  Все оповещения удалены!</p> <p>Неудача:  Нет активных оповещений</p> <p>Вы еще не создавали алерты.</p>				
13	/portfolio	/portfolio	/portfolio	Просмотр портфеля
<p> Портфель (3 актива):</p> <ul style="list-style-type: none"> • 0.42 BTC (\$26,544) • 3.1 ETH (\$10,695) <p> Итого: \$37,739</p>				
14	/get_portfolio_price	/get_portfolio_price	/get_portfolio_price	Отслеживание цены портфеля за указанный период
<p> Цена вашего портфеля: \$2,052</p> <p> Динамика изменения:  рост 10% (+\$200)</p>				
15	/get_assets_price	/get_assets_price	/get_assets_price	Показать убыток/прибыль по каждой криптовалюте в портфеле
<p> Цена вашего портфеля: \$37,239</p> <p> Динамика изменения:  рост 10% (+\$200)</p> <p>Активы в портфеле:</p> <ul style="list-style-type: none"> • 0.42 BTC (\$26,544):  рост 0.5% (+\$400) • 3.1 ETH (\$10,695):  спад 10% (-\$200) 				
Информация об авторах				
16	/authors	/authors	/authors	Вывод список авторов
<ul style="list-style-type: none"> • Лобанов Павел (@pmlobanov) • Черепнов Максим (@aofitasotr) • Чурова Софья (@SunnyChur) <p>- Студенты 3 курса СПбПУ Петра Великого направления “Математика и компьютерные науки”</p>				