

**информационная платформа для мониторинга финансовых активов с  
интегрированным ИИ-ассистентом  
Руководство программиста**

### **Аннотация**

Настоящий документ содержит основные положения и сведения, необходимые для работы программиста с информационной платформой для мониторинга финансовых активов с интегрированным ИИ-ассистентом.

Документ разработан согласно требованиям следующих нормативных документов: ГОСТ 19.504-79, СТП ВятГУ 101-2004.

## СОДЕРЖАНИЕ

1	Назначение и условия применения программы.....	2
1.1	Назначение и функции, выполняемые программой .....	2
1.2	Требования к техническому обеспечению .....	2
1.3	Требования к технологическому стеку .....	3
2	Характеристика программы.....	4
2.1	Основные характеристики системы .....	4
2.2	Временные характеристики системы .....	4
2.3	Режимы работы системы .....	5
2.4	Средства контроля правильности выполнения программы .....	5
3	Обращение к программе.....	6
4	Входные и выходные данные .....	7
4.1	Организация входной информации .....	7
4.2	Организация выходной информации .....	7
5	Сообщения.....	8
5.1	Информирующие сообщения.....	8
5.2	Сообщения об ошибках .....	8

## **1 Назначение и условия применения программы**

В данном разделе описано назначение, основная функциональность информационной системы для взаимодействия официантов и кухни ресторана, а также условия для ее применения.

### **1.1 Назначение и функции, выполняемые программой**

Платформа представляет собой веб-приложение, предназначенное для трейдеров, инвесторов и аналитиков, обеспечивая мониторинг и анализ финансовых рынков в реальном времени. Она объединяет данные о активах акциях, криптовалютах, валютах и индексах с инструментами визуализации и искусственным интеллектом для прогнозирования тенденций. Основные модули включают аутентификацию с поддержкой email и Google, персонализированный дашборд с виджетами для активов и новостей, графики с тайм фреймами и индикаторами — скользящими средними, относительной силой и полосами Боллинджера, раздел новостей с поиском и фильтрами, чат с ИИ-ассистентом для анализа и рекомендаций на основе истории запросов, систему комментариев и реакций под новостями, административную панель для управления пользователями, API-ключами и модерацией контента, а также интеграцию внешних источников данных. Функции монетизации предусматривают ограничения для бесплатных пользователей и премиум-подписку для безлимитного доступа к ИИ. Платформа способствует снижению времени на анализ, повышению точности прогнозов и улучшению пользовательского опыта через интеграцию данных, обеспечивая гибкость и масштабируемость для различных типов пользователей. Общий объем функциональности охватывает более десяти ключевых процессов от регистрации до генерации отчетов, что делает систему комплексным инструментом для финансового мониторинга.

### **1.2 Требования к техническому обеспечению**

Платформа предназначена для развертывания на серверах с учетом нагрузки от реального времени обработки данных, обработки ИИ-запросов и хранения финансовой информации:

Серверная часть:

- Процессор: 4-ядерный CPU или выше, с поддержкой многопоточной обработки для ИИ и реал-тайм обновлений;
- Оперативная память: 8 Гб минимум, 16 Гб рекомендуется для кэширования данных и одновременных запросов;
- Дисковое пространство: 64 Гб для системы + место для данных пользователей, включая хранение сессий чата и новостей;
- Операционная система: Linux (Ubuntu 20.04+), Windows Server 2019+;

- Доступ в Интернет: стабильное соединение со скоростью не менее 100 Мбит/с для интеграции внешних API.

Клиентская часть:

- Подключение к сети Интернет;
- Веб-браузер: Chrome 90+, Firefox 88+, Safari 14+, Edge 90+

### **1.3 Требования к технологическому стеку**

Для успешного запуска и функционирования платформы необходимо окружение с установленными технологиями, ориентированными на реал-тайм обработку финансовых данных и ИИ-интеграцию:

Бэкенд:

- Python 3.11+.
- Фреймворк: FastAPI 0.95+.
- База данных: PostgreSQL 14+.
- Веб-сервер: Nginx + Uvicorn.

Фронтенд:

- JavaScript/TypeScript.
- React 18+.
- UI-библиотека: Material-UI.

State management: Redux Toolkit.

Инфраструктура:

- Контроль версий: Git.
- Репозиторий: GitHub.
- CI/CD: GitHub Actions.
- Контейнеризация: Docker + Docker Compose.

## **2 Характеристика программы**

В данном разделе приведено описание основных характеристик и особенностей автоматизированной системы.

### **2.1 Основные характеристики системы**

Платформа построена по модульной архитектуре, где каждый компонент отвечает за конкретные аспекты мониторинга финансовых рынков. Слой представления обеспечивает интуитивный интерфейс для пользователей, включая дашборд с настраиваемыми виджетами для отображения активов и новостей, интерактивные графики с поддержкой различных тайм фреймов и индикаторов, а также чат с ИИ-ассистентом. Это позволяет трейдерам быстро оценивать тенденции без переключения между приложениями.

Бизнес-логика фокусируется на обработке запросов, включая анализ трендов с использованием индикаторов вроде скользящих средних и полос Боллинджера и.т.д, прогнозирование на основе истории чата и персонализацию рекомендаций. Сервисы в FastAPI управляют монетизацией, ограничивая доступ к ИИ для бесплатных пользователей и предоставляя безлимит для премиума. Валидация данных гарантирует безопасность, предотвращая несанкционированные действия и ошибки в вводе.

Слой данных опирается на PostgreSQL для хранения профилей, комментариев и сессий. Redis кэширует данные в реальном времени, такие как цены активов, минимизируя задержки. Репликация базы обеспечивает отказоустойчивость, позволяя системе работать даже при сбоях. Общая архитектура способствует масштабируемости: модули можно обновлять независимо, интегрировать новые API для новостей или улучшать ИИ без влияния на другие части. Это делает платформу надежным инструментом для инвесторов, сочетая скорость, точность и удобство в одном решении.

### **2.2 Временные характеристики системы**

При соблюдении условий эксплуатации система обеспечивает следующие временные характеристики:

- Время отклика API:  $\leq 3000$  мс
- Время загрузки страниц:  $\leq 5$  секунд
- Время выполнения поиска новостей:  $\leq 4$  секунд
- Время анализа ИИ-запроса:  $\leq 1$  мин
- Время обновления графиков через:  $\leq 30$  секунда

## 2.3 Режимы работы системы

Платформа поддерживает три режима работы в зависимости от этапа разработки и эксплуатации:

- режим разработки;
- режим тестирования;
- режим продакшн.

В режиме разработки активированы инструменты отладки, логирование на уровне DEBUG и доступ к тестовым API. В режиме тестирования включены симуляция нагрузки, данные и проверка производительности. В продакшн-режиме система работает с реальными данными, ограничениями по подписке и полной защитой. Переключение режимов осуществляется через переменные окружения в конфигурационном файле, что гарантирует безопасность и контроль над функциональностью.

## 2.4 Средства контроля правильности выполнения программы

Контроль правильности выполнения платформы осуществляется с использованием современных инструментов мониторинга, логирования и тестирования:

- **Мониторинг:**
  - Prometheus для сбора метрик производительности
  - Grafana для визуализации графиков и дашбордов
  - Sentry для отслеживания и анализа ошибок в реальном времени
- **Логирование:**
  - Structured logging в JSON формате
  - Уровни логирования: DEBUG, INFO, WARNING, ERROR
  - Ротация логов по размеру и времени
- **Тестирование:**
  - Pytest для юнит и интеграционных тестов бэкенда
  - Jest для тестирования фронтенда
  - Selenium для end-to-end тестирования пользовательских сценариев
  - GitHub Actions для автоматизированной CI/CD

### 3 Обращение к программе

Запуск серверной части платформы осуществляется как системная служба операционной системы. При старте автоматически загружается конфигурационный файл с параметрами подключения к базе данных, внешним API и настройками безопасности. Далее выполняется инициализация соединения с PostgreSQL и Redis, запуск FastAPI-сервера для обработки REST-запросов и активация WebSocket-сервера для передачи данных в реальном времени. После успешного запуска всех компонентов система переходит в режим ожидания входящих запросов от клиентских приложений, готовая к аутентификации пользователей, обновлению цен активов и генерации аналитики.

Клиентская часть реализована в виде одностраничного веб-приложения, доступного через современные браузеры на компьютерах, планшетах и смартфонах. Подключение к серверу начинается с открытия главной страницы и прохождения процедуры авторизации — через email с паролем или OAuth через Google. После успешного входа загружаются персональные настройки дашборда, устанавливается постоянное WebSocket-соединение для мгновенного обновления графиков и новостей, а также отображается основной интерфейс с настраиваемыми виджетами. Клиентское приложение автоматически поддерживает синхронизацию с сервером, обеспечивая получение актуальных данных о ценах, трендах и рекомендациях ИИ без необходимости перезагрузки страницы.

Управление системой и передача данных между клиентом и сервером организованы через два канала связи: HTTP-запросы к REST API для операций с данными — регистрация, настройка профиля, сохранение дашборда, и постоянное WebSocket-соединение для мгновенной передачи уведомлений, обновлений цен активов, новых комментариев и ответов ИИ-ассистента.



## **4 Входные и выходные данные**

В данном разделе приведено описание организации используемой входной и выходной информации в системе управления личной продуктивностью.

### **4.1 Организация входной информации**

Организация входной информации осуществляется через интерактивные элементы пользовательского интерфейса: поля ввода для регистрации и аутентификации, выпадающие списки для выбора активов и тайм фреймов, текстовые поля для запросов к ИИ-ассистенту, формы добавления комментариев и настройки виджетов дашборда. Пользователь вводит данные о профиле, предпочтениях по активам, ключевых словах для поиска новостей, параметрах технических индикаторов. Все данные передаются на сервер в стандартизированном формате JSON, что обеспечивает единообразие обработки, упрощает валидацию и интеграцию с внешними API для получения рыночных данных.

### **4.2 Организация выходной информации**

Организация выходной информации предусматривает передачу всех обработанных данных с сервера клиенту в формате JSON. На стороне клиентского приложения данные преобразуются в визуальные элементы: интерактивные графики цен с наложением индикаторов, карточки новостей с тегами и реакциями, рекомендации ИИ в чате, персонализированный дашборд с виджетами. Такое представление обеспечивает высокую наглядность, позволяет пользователю быстро анализировать рыночные тренды, отслеживать избранные активы и принимать обоснованные решения на основе актуальной аналитики.

## **5    Сообщения**

В данном разделе приведены тексты сообщений, выдаваемых пользователю в ходе работы системы, с описанием их содержания и рекомендуемых действий.

### **5.1   Информирующие сообщения**

Информирующие сообщения отображаются в виде всплывающих уведомлений на нейтральном фоне. Примеры: «Регистрация прошла успешно», «Дашборд сохранён», «График обновлён», «Комментарий добавлен», «ИИ-анализ завершён», «Актив добавлен в избранное». Эти сообщения подтверждают успешное выполнение операций, не требуют действий от пользователя и автоматически исчезают через несколько секунд.

### **5.2   Сообщения об ошибках**

Сообщения об ошибках выводятся в случае сбоев и отображаются в виде всплывающих уведомлений на красном или жёлтом фоне. Примеры: «Ошибка авторизации — проверьте email и пароль», «Превышен лимит запросов к ИИ», «Не удалось загрузить данные актива», «Ошибка сохранения настроек». При появлении таких сообщений пользователю рекомендуется повторить действие, обновить страницу или проверить интернет-соединение. Если ошибка сохраняется, следует обратиться в техническую поддержку через форму обратной связи в приложении.