

**CRM-система с AI-ассистентом для поддержки продаж и аналитики**  
**Руководство программиста**

Киров, 2025 г.

## **АННОТАЦИЯ**

Настоящий документ содержит основные положения и сведения, необходимые для работы программиста с программным обеспечением для анализа и обработки речевых сигналов.

Документ разработан согласно требованиям ГОСТ Р 59795–2021 и ГОСТ Р 59853–2021.

## СОДЕРЖАНИЕ

1	Общие положения	4
	1.1    Наименование программы .....	4
2	Назначение и условия применения	5
	2.1    Назначение программы .....	5
	2.2    Функции, выполняемые программой .....	5
	2.3    Сведения о технических и программных средствах, обеспечивающих выполнение программы .....	5
	2.3.1    Требования к составу и параметрам комплекса технических средств .....	5
	2.3.2    Требования к общесистемному программному обеспечению .....	6
	2.3.3    Требования к персоналу .....	8
3	Характеристика программных средств программы	11
	3.1    Описание основных характеристик программы .....	11
	3.1.1    Временные характеристики программы .....	11
	3.1.2    Режим работы программы .....	11
	3.1.3    Средства контроля правильности выполнения программы .....	12
4	Обращение к программе	13
5	Входные и выходные данные	14
	5.1    Характер и организация входных данных .....	14
	5.2    Характер и организация выходных данных .....	14
6	Сообщения	16
7	Аварийные ситуации	17
	7.1    Действия при ошибках загрузки аудиофайла.....	17
	7.2    Действия при зависании приложения .....	17
	7.3    Действия при потере данных .....	18
	7.4    Действия при отказе оборудования.....	18
	7.5    Действия в других аварийных ситуациях .....	18

## **1      Общие положения**

Настоящий документ содержит сведения, необходимые для работы программиста с CRM-системой с AI-ассистентом для поддержки продаж и аналитики.

### **1.1    Наименование программы**

Полное наименование системы: CRM-система с AI-ассистентом для поддержки продаж и аналитики.

## **2 Назначение и условия применения**

### **2.1 Назначение программы**

Основным назначением системы является автоматизация процессов управления клиентами и анализа продаж, включая:

- регистрацию и авторизацию пользователей;
- ведение и актуализацию клиентской базы с полной информацией о контактах, сделках и взаимодействиях;
- отслеживание истории покупок и формирование статистики по продажам;
- генерацию аналитических отчётов и AI-рекомендаций для повышения эффективности работы с клиентами;
- интеграцию с внешними приложениями через API и предоставление безопасного доступа к данным.

### **2.2 Функции, выполняемые программой**

Перечень функций, реализуемых системой:

- регистрация и авторизация пользователей;
- генерация и управление персональными API-ключами для интеграции внешних приложений;
- ведение и актуализация клиентской базы с возможностью добавления, редактирования и удаления клиентов;
- учёт и отслеживание истории покупок и взаимодействий клиентов;
- формирование аналитических отчётов по продажам и работе с клиентами;
- анализ тенденций продаж и формирование AI-рекомендаций по работе с клиентами;
- предоставление интерфейса для поиска и фильтрации клиентов и сделок;
- экспорт данных и отчётов в различные форматы (PDF, CSV, XLSX).

### **2.3 Сведения о технических и программных средствах, обеспечивающих выполнение программы**

#### **2.3.1 Требования к составу и параметрам комплекса технических средств**

Средства аппаратного обеспечения, необходимые для функционирования программы, рассчитываются индивидуально под потребность решаемых задач, минимальные требования представлены в таблице.

Таблица 1 – Минимальные системные требования

Вид ресурса	Минимальные требования
Тип ЭВМ	Персональный компьютер или ноутбук
Процессор	Тактовая частота не менее 2.0 ГГц (рекомендуется 4 ядра для корректной работы AI-аналитики)
Оперативная память (ОЗУ)	Не менее 8 ГБ (рекомендуется 16 ГБ при работе с крупными базами клиентов)
Накопитель (HDD/SSD)	Не менее 10 ГБ свободного места (SSD предпочтительно для скорости работы)
Видеокарта	Интегрированная с поддержкой OpenGL 3.0 (для визуализации графиков и отчётов)
Монитор	Разрешение не ниже 1024×768
Устройства ввода	Клавиатура и мышь (для удобной работы с интерфейсом CRM)
Аудиооборудование	Встроенные или внешние динамики/наушники (для уведомлений и AI-сигналов, если предусмотрено)
Операционная система	Windows 11
Сетевое соединение	Локальное подключение (localhost) или доступ к корпоративной сети для синхронизации данных, доступ к сети интернет при развёртывании
Программные компоненты	Python 3.8+, Java 18+, PostgreSQL 12+ (для работы сервера и AI-модулей)
Средства резервного копирования	Носитель ёмкостью не менее 32 ГБ (рекомендуется для регулярного резервного копирования базы клиентов и отчётов и моделей)

Приложение должно использоваться в стабильных условиях окружающей среды без воздействия экстремальных температур и влажности.

### 2.3.2 Требования к общесистемному программному обеспечению

Перечень и описание Системного ПО, обеспечивающего корректную работу приложения, представлены в таблице 2.

Таблица 2 – Сведения о программном обеспечении

Вид ресурса	Минимальные требования
-------------	------------------------

Операционная система	Windows 11 (64-разрядная)
Java Runtime Environment	Версия 18 и выше для работы серверной части (Spring Boot)
Веб-фреймворк (сервер)	Spring Boot 2.7+ для работы CRM и API
Фронтенд	React 18+ для интерфейса пользователя
Интерпретатор Python	Версия 3.8+ для AI-модулей и аналитики
Библиотеки AI и аналитики	SpeechBrain совместимая версия, NumPy, SciPy актуальные стабильные версии
База данных PostgreSQL	Версия 12+ для хранения клиентской базы и истории продаж
OpenGL	Поддержка версии 3.0+ для визуализации графиков и AI-отчётов
Средства архивирования	Любые стандартные утилиты ОС (7-Zip, встроенные архиваторы) для резервного копирования
Браузер	Любой современный браузер (Chrome, Edge, Firefox) для работы с веб-интерфейсом CRM

Средства, обеспечивающие функционирование Системы:

Сервер приложения — локальный или облачный сервис, реализованный на Java Spring, обеспечивающий обработку данных CRM, управление клиентской базой, историей продаж и генерацию AI-отчётов. Сервер предоставляет REST API для взаимодействия с веб-клиентом.

Клиентское приложение — веб-приложение на React, обеспечивающее интерфейс пользователя CRM: просмотр клиентов, сделок, формирование аналитических отчётов и получение AI-рекомендаций. Взаимодействует с сервером через HTTP/HTTPS.

База данных — PostgreSQL, используемая для хранения информации о клиентах, сделках, параметрах аналитики и результатах работы AI-модуля.

Файловое хранилище — локальный или сетевой каталог для хранения отчётов, экспортованных данных и временных файлов системы.

AI-модуль на Python — выполняет анализ данных, формирует прогнозы и рекомендации по продажам. Подключается к серверу через REST API.

Сетевой стек HTTP/HTTPS — обеспечивает безопасное взаимодействие между веб-клиентом, сервером на Java и AI-модулем на Python, а также защиту передачи данных и интеграцию внешних сервисов.

Взаимодействие всех компонентов системы реализуется в локальной клиент–серверной архитектуре, представленной на рисунке 1.

Компоненты системы работают автономно и обеспечивают полную функциональность CRM без постоянного подключения к внешним сервисам, за исключением первичной загрузки и обновления моделей AI для генерации аналитических рекомендаций.

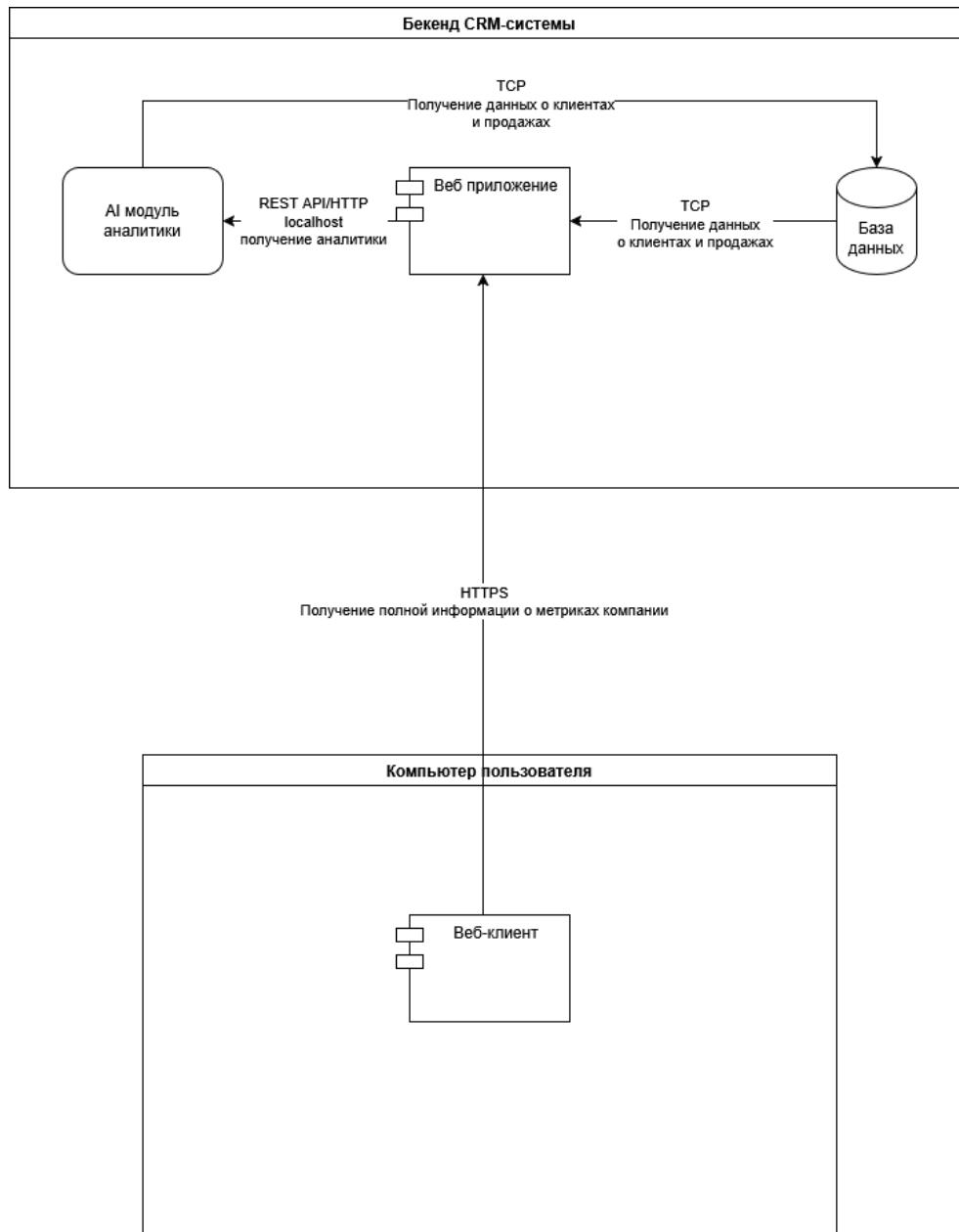


Рисунок 1 – Архитектурная схема функционирования системы

### 2.3.3 Требования к персоналу

Обязательные навыки программиста:

- Владение Windows (администрирование, командная строка, управление процессами);

- Опыт работы с веб-приложениями и клиент-серверной архитектурой;
- Базовые навыки администрирования локальных и сетевых систем.

Специальные навыки:

Python разработка (AI-модули):

- Стандартная библиотека Python;
- Работа с библиотеками для машинного обучения и аналитики (NumPy, Pandas, Scikit-learn, PyTorch, SpeechBrain);
- Основы обработки данных и анализа сигналов;
- Работа с REST API и JSON для интеграции с клиентской частью;
- Асинхронное программирование (asyncio, async/await).

Java разработка (серверная часть и веб-приложение):

- Java 17+ и Spring Framework;
- Разработка REST API и сервисов для клиентских приложений;
- Многопоточность и управление ресурсами сервера;
- Работа с базами данных через JPA/Hibernate.

Frontend разработка (клиентское веб-приложение):

- React 18+, TypeScript;
- Работа с REST API, AJAX-запросами и WebSockets (по необходимости);
- Создание динамических интерфейсов и визуализация данных;
- Опыт интеграции с AI-модулями через API.

Базы данных:

- Знание SQL синтаксиса;
- Проектирование реляционных баз данных;
- Умение работать с PostgreSQL;
- Опыт резервного копирования и восстановления данных.

Смежные знания:

- Основы аналитики данных и визуализации;
- Основы машинного обучения и построения рекомендаций;
- Принципы клиент-серверной архитектуры и REST;
- Навыки работы с кэшированием и оптимизацией производительности.

Желательные навыки:

- Опыт работы с облачными сервисами и API (Hugging Face, облачные модели);
- Знание Docker и контейнеризации приложений;
- Опыт CI/CD (Jenkins, GitLab CI, GitHub Actions);
- Оптимизация производительности веб- и серверных приложений;
- Опыт работы с аналитическими и CRM-системами.

### 3 Характеристика программных средств программы

#### 3.1 Описание основных характеристик программы

Программа предназначена для управления клиентской базой, анализа продаж и генерации отчетов с использованием AI-рекомендаций.

Она реализована как веб-приложение с клиент–серверной архитектурой, состоящее из:

- Клиентское веб-приложение (React/TypeScript) — обеспечивает графический интерфейс пользователя, отображение данных, работу с клиентами, продажами и отчетами, а также получение AI-рекомендаций;
- Серверная часть (Java/Spring) — реализует REST API, управляет бизнес-логикой, обработкой запросов, хранением данных и интеграцией с AI-модулем;
- AI-модуль (Python) — выполняет обработку данных, генерацию прогнозов и рекомендаций, использует предобученные модели (Hugging Face, PyTorch, SpeechBrain при необходимости);
- База данных PostgreSQL — хранит информацию о клиентах, продажах, проектах, результатах анализа и журналах работы;
- Файловое хранилище — локальные и/или сетевые каталоги для хранения отчетов, экспортованных файлов и временных данных.

Компоненты взаимодействуют через REST API по защищённому протоколу HTTP/HTTPS. Первоначальная загрузка моделей AI происходит при первом запуске и доступе к внешнему облачному узлу Hugging Face Hub (или аналогичным хранилищам).

##### 3.1.1 Временные характеристики программы

Среднее время загрузки веб-клиента и отображения стартового экрана — не более 3 секунд.

Формирование отчёта по небольшому набору данных (до 50 клиентов и 100 транзакций) занимает 2–5 секунд.

Генерация AI-рекомендаций для одного клиента или одной продажи занимает 1–3 секунды.

При увеличении объёма данных (база клиентов, история продаж) время обработки растёт пропорционально размеру выборки.

Обмен данными между клиентом и сервером по REST API в локальной сети или через корпоративный сервер занимает не более **0,5–1 секунды** на операцию.

### 3.1.2 Режим работы программы

Программа функционирует в интерактивном режиме: пользователь взаимодействует с веб-клиентом через GUI (React), получает доступ к спискам клиентов и продаж, формирует отчёты и AI-рекомендации, которые отображаются в виде таблиц, графиков и визуализаций ключевых показателей.

Поддерживается также режим взаимодействия через REST API: внешние приложения или пользовательские скрипты могут запрашивать данные о клиентах, сделках и отчетах, а также инициировать генерацию AI-рекомендаций для автоматизации анализа больших объёмов данных

### 3.1.3 Средства контроля правильности выполнения программы

Для обеспечения корректности функционирования системы реализованы следующие механизмы:

- проверка формата и корректности вводимых данных о клиентах и сделках;
- автоматическая валидация параметров генерации отчётов и AI-рекомендаций;
- логирование всех ответов сервера системы в файл server.log, логирование непосредственно из контейнера docker;
- контроль доступности и целостности базы данных PostgreSQL (проверка соединения и таблиц);
- отображение системных сообщений и уведомлений об ошибках в интерфейсе веб-клиента;
- проверка успешного выполнения REST-запросов между клиентом и сервером.

## 4     Обращение к программе

Система запускается через веб-браузер или десктопный клиент (React-клиент для интерфейса и Java Spring-сервер для обработки данных). Для управления системой предусмотрены следующие способы:

- интерактивный запуск пользователем через веб-интерфейс CRM;
- вызов API-сервисов для автоматизированного взаимодействия внешних приложений с CRM;
- загрузка конфигурационных параметров при старте сервера (файл application.properties или .env).

Порядок обращения к системе:

- Пользователь открывает веб-клиент через браузер и проходит авторизацию;
- После авторизации загружается стартовый экран с основными разделами: список клиентов, список продаж, формирование отчётов, получение AI-рекомендаций и просмотр API-ключа;
- Сервер Java Spring и AI-модуль на Python инициализируются, загружаются необходимые модели и устанавливаются соединения с базой данных PostgreSQL;

Пользователь через веб-интерфейс может:

- просматривать и редактировать данные клиентов;
- просматривать и редактировать сделки;
- запускать генерацию отчётов и получать AI-рекомендации;
- управлять API-ключами для интеграции с внешними приложениями;

При необходимости система отображает уведомления о статусах операций, ошибках и успешном завершении действий в интерфейсе CRM.

## **5 Входные и выходные данные**

### **5.1 Характер и организация входных данных**

Входные данные системы представляют собой информацию о клиентах, продажах и параметрах аналитики, а также запросы на генерацию отчетов и AI-рекомендаций.

Основные характеристики входных данных:

- Данные о клиентах: контактные данные, история взаимодействий;
- Данные о продажах: сумма сделки, дата, клиент, товары/услуги;
- Параметры аналитики: временной период, типы отчетов и форматы экспорта (PDF, CSV, XLSX);
- Запросы на AI-аналитику: выбор рекомендаций, прогнозов и сравнений на основе истории покупок и поведения клиентов;
- Формат и структура данных: JSON или XML при передаче через API, записи в реляционной базе данных PostgreSQL;
- Организация данных: все данные структурированы по проектам/учетным записям, обеспечивая целостность и доступность для формирования отчетов и аналитики.

Перед обработкой системой данные проходят проверку на корректность:

- валидация формата и обязательных полей;
- проверка уникальности ключевых идентификаторов (ID клиентов, ID сделок);
- контроль целостности связей между клиентами и их сделками.

Входные данные могут храниться локально в базе данных или передаваться через API для интеграции с внешними приложениями и сервисами.

### **5.2 Характер и организация выходных данных**

Выходные данные системы формируются после обработки информации о клиентах, продажах и аналитических запросов, включая AI-рекомендации и отчёты.

Основные типы выходных данных:

- Отчёты и аналитика:
  - Сводные отчёты по клиентам и сделкам с ключевыми показателями;
  - AI-рекомендации по работе с клиентами, прогнозы продаж и выявление закономерностей;
  - Визуализация данных в виде графиков, диаграмм и таблиц.
- Детализированные данные по клиентам и продажам:

- Списки клиентов с контактами, историей взаимодействий, примечаниями;
- Списки сделок с суммами, статусами, датами и связанными клиентами.
- Форматы экспорта: PDF, CSV, XLSX для отчётов и визуализаций; JSON или XML для интеграции с внешними приложениями через API.
- Логи и системные отчёты: информация о ходе обработки данных, возможных ошибках и выполненных операциях, сохраняется в файл server.log для контроля и отладки.

Все выходные данные структурированы по проектам или учётным записям, что обеспечивает целостность информации и её доступность для последующего анализа, обмена с другими системами и интеграции с внешними приложениями.

## 6 Сообщения

В системе CRM/AI сообщения реализуются как элементы интерфейса, обеспечивающие информирование пользователя о ключевых состояниях, событиях и ошибках. Они помогают пользователю понимать текущий статус операций и принимать корректирующие действия.

Сообщения подразделяются на следующие виды:

Информационные сообщения — уведомляют пользователя о успешном выполнении операций или текущем состоянии.

Примеры:

- «Клиент успешно добавлен»
- «Сделка сохранена»
- «Отчёт успешно сформирован»
- «API-ключ скопирован в буфер обмена»

Предупреждения — информируют о потенциальных проблемах, которые не блокируют работу, но требуют внимания.

Примеры:

- «Некоторые поля клиента не заполнены — данные могут быть неполными»
- «Формат загруженного файла может не поддерживаться полностью»
- «Скорость соединения с сервером низкая, возможны задержки»

Ошибки — отображаются при возникновении критических сбоев или неверных действиях.

Примеры:

- «Невозможно загрузить данные клиента»
- «Ошибка при сохранении сделки»
- «Отчёт не сформирован»
- «Некорректный или отсутствующий API-ключ»
- «Не удалось подключиться к базе данных»

Аварийные сообщения — сопровождают серьёзные проблемы, препятствующие работе системы.

Примеры:

- «Сервер недоступен, завершение операции»
- «Превышено ограничение памяти, операция прервана»
- «Внутренняя ошибка модуля AI — перезапустите сервис»

## 7 Аварийные ситуации

В данном разделе настоящего руководства пользователя содержится информация о действиях пользователя в случаях возникновения аварийных ситуаций.

### 7.1 Действия при ошибках загрузки аудиофайла

Возможные ошибки:

- «Невозможно загрузить данные клиента»;
- «Ошибка при сохранении сделки»;
- «Некорректный или отсутствующий API-ключ»;
- «Отчёт не сформирован».

Действия:

- Убедиться, что все обязательные поля заполнены корректно;
- Проверить соединение с сервером и доступ к базе данных;
- При ошибке сохранения клиента или сделки — повторить операцию после перезапуска приложения;
- При некорректном API-ключе — проверить правильность ключа и при необходимости сгенерировать новый;
- Если отчёт не сформирован, проверить параметры отчёта и фильтры, затем повторить генерацию;
- В случае системных сбоев — обратиться к логам (server.log) и перезапустить сервис.

### 7.2 Действия при зависании приложения

При полном зависании клиента или веб-интерфейса:

- Перезапустить веб-браузер или клиентское приложение;
- При необходимости перезагрузить компьютер.

При медленной работе:

- Закрыть другие ресурсоёмкие приложения;
- Проверить объём свободной оперативной памяти (не менее 8 ГБ рекомендуется);
- Проверить доступное место на диске (не менее 10 ГБ);
- При сохранении проблем с производительностью — обратиться к системному администратору.

### **7.3 Действия при потере данных**

При случайном удалении информации о клиентах, сделках или отчётах:

- Если данные были ранее экспортированы или синхронизированы с сервером, восстановить их из резервной копии или истории версий на сервере;
- Если данные не были сохранены на сервере (например, введённые в веб-форму изменения не были отправлены), восстановление невозможно;
- Для предотвращения потерь регулярно использовать кнопки «Сохранить» или «Сохранить изменения» в веб-интерфейсе;
- Настоятельно рекомендуется настроить автоматическое сохранение и резервное копирование данных на сервере, а также использование функционала истории версий для критичных объектов (клиенты, сделки, отчёты).

### **7.4 Действия при отказе оборудования**

Проблемы с отображением интерфейса веб-приложения:

- Проверить подключение монитора и кабелей;
- Перезагрузить компьютер;
- Убедиться в исправности видеокарты и актуальности драйверов;
- Проверить работу браузера (рекомендуются Chrome, Edge, Firefox последних версий);
- При необходимости очистить кеш браузера или открыть приложение в другом браузере.

### **7.5 Действия в других аварийных ситуациях**

В случае возникновения любых других аварийных ситуаций при работе с веб-приложением необходимо выполнить следующие действия:

- Определить природу и причины проблемы (например, ошибки в данных, недоступность сервера, сбои в браузере или сети);
- Принять меры для минимизации возможного ущерба, например, прекратить редактирование данных, сохранить промежуточные результаты или временно заблокировать доступ к критическим разделам;
- Сообщить о происшествии разработчику, системному администратору или ответственным сотрудникам;
- Проверить состояние серверной части и базы данных, при необходимости перезапустить сервер или обновить соединение с базой;

- После устранения причины возобновить нормальную работу приложения и продолжить выполнение задач.