

**ОПИСАНИЕ ФУНКЦИОНАЛЬНЫХ ХАРАКТЕРИСТИК
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ RUHUMAN**

**г. Москва
2025**

ОГЛАВЛЕНИЕ:

1. ОБЩИЕ СВЕДЕНИЯ.....	3
1.1. Цели создания ПО.....	3
1.2. Назначение ПО.....	3
2. АРХИТЕКТУРА ПРОЕКТА	3
2.1. Функциональность проекта.....	3
2.2. Безопасность обмена данных.....	3
3. ТРЕБОВАНИЯ К СЕРВЕРНОЙ ИНФРАСТРУКТУРЕ.....	3
3.1. Аппаратные требования к серверу.....	3
3.2. Сетевые настройки.....	4
3.3. Хранение модели, безопасность и контроль выполнения.....	4
3.4. Мониторинг работоспособности.....	4
3.5. Автоматическое развертывание	4
4. ТРЕБОВАНИЯ К КЛИЕНТСКОЙ ИНФРАСТРУКТУРЕ.....	4
4.1. Аппаратные требования.....	4
4.2. Программные требования.....	5
4.3. Сетевые требования.....	5
4.4. Безопасность на клиентской стороне.....	5

1. ОБЩИЕ СВЕДЕНИЯ

Полное наименование ПО: RUHUMAN.

1.1. Цели создания ПО

ПО создано с целью проверки витальности (liveness) — автоматической проверки в режиме реального времени что перед камерой находится реальный, живой человек, а не его дипфейк.

1.2. Назначение ПО

Система предназначена для обеспечения безопасности онлайн телеконференций пользователей за счёт автоматического анализа видеопотока. ПО предоставляет возможность проверки витальности (liveness) и детекции дипфейков.

2. АРХИТЕКТУРА ПРОЕКТА

ПО запускается на пользовательском компьютере с операционной системой Windows и отвечает за локальный захват и предварительную обработку видеоданных.

2.1. Функциональность проекта:

- Обнаружение движущихся лиц с помощью локальных алгоритмов компьютерного зрения
- Автоматическая запись короткого видеофрагмента длительностью до 1 секунды
- Преобразование записанного видеофрагмента в компактный цифровой код (вектор)
- Отправка этого кода на сервер через HTTP-запрос по FastAPI-интерфейсу
- Получение результата классификации (реальное/фейковое лицо) в виде числовой оценки (score), интерпретация и отображение этого результата пользователю

Приложение реализовано с использованием Python и библиотек OpenCV, PyTorch и других.

2.2. Безопасность обмена данных

Обмен между клиентом и сервером осуществляется по FastAPI-интерфейсу. Передаются только обезличенные признаки (векторы) в виде byte-кода, что исключает утечку изображений или видео. Передаваемые данные обрабатываются в оперативной памяти серверной/клиентской части ПО и на диск сервера/клиента не сохраняются.

3. ТРЕБОВАНИЯ К СЕРВЕРНОЙ ИНФРАСТРУКТУРЕ

Серверная часть системы, обрабатывающая запросы от клиентского приложения и выполняющая анализ обезличенного вектора данных, развёртывается на компьютере с ОС Linux, предпочтительно Ubuntu.

3.1. Аппаратные требования к серверу

- Процессор - Intel Core i3 и выше
- Количество ядер процессора – не менее 8 физических или логических ядер

- Частота процессора – от 2600 МГц
- Объем оперативной памяти – от 12 ГБ
- Тип памяти – DDR4 и выше
- Частота оперативной памяти – от 1600 МГц
- Дисковое пространство – от 120 ГБ

3.2. Сетевые настройки

При настройке серверной части ПО в контуре клиента, в конфигурационном файле клиентской части ПО нужно указать IP-адрес сервера.

3.3. Хранение модели, безопасность и контроль выполнения

Обученная нейросетевая модель в хранится локально на сервере. Это позволяет каждому экземпляру сервера работать автономно, без необходимости доступа к внешним хранилищам. Лог-файлы не сохраняются из соображений безопасности. Контроль работоспособности осуществляется штатными механизмами мониторинга в компании клиента - по успешному запуску, отклику на HTTP-запросы и статусу сервиса.

3.4. Мониторинг работоспособности

Для контроля состояния серверной части используется базовый мониторинг на уровне контейнеров. Достаточно проверки успешного запуска и отклика по HTTP-интерфейсу. Интеграция с внешними системами мониторинга настраивается на усмотрение клиента самостоятельно. Централизованное логирование и хранение событий в системе не предусмотрены из соображений безопасности данных.

3.5. Автоматическое развертывание

В целях безопасности автоматическое развертывание не используется. Обновления передаются вручную уполномоченным администратором на сервер по безопасному каналу. Установка и перезапуск сервиса производятся вручную, что исключает риск несанкционированного обновления или внедрения уязвимостей через цепочку поставки.

4. ТРЕБОВАНИЯ К КЛИЕНТСКОЙ ИНФРАСТРУКТУРЕ

Клиентская часть ПО представляет собой настольное приложение, копируемое на рабочую станцию оператора. Для его корректной работы необходимо соблюдение следующих требований к аппаратным и программным компонентам клиентской инфраструктуры.

4.1. Аппаратные требования

Для корректной работы клиентской части ПО необходимо наличие персонального компьютера или ноутбука со следующими минимальными характеристиками: процессор с тактовой частотой не ниже 2.0 ГГц (двухъядерный и выше), оперативная память объемом не менее 4 ГБ, свободное дисковое пространство не менее 10 ГБ для установки программного обеспечения.

Наличие видеокамеры не требуется в рабочем режиме, но может использоваться в отладочном режиме при локальной проверке данных.

4.2. Программные требования

Клиентская часть программного обеспечения должна функционировать под управлением операционных систем Windows 10 / 11 (64-разрядные). Обмен данными с серверной частью осуществляется через API-интерфейс по протоколу HTTP. Программное обеспечение должно быть совместимо с локальными видеоустройствами, если они используются в тестовом или отладочном режиме.

4.3. Сетевые требования

Клиентское приложение должно иметь доступ к серверу по HTTP (порт 8004).

Возможность выхода в интернет, если сервер находится вне контура компании клиента.

В локальной сети необходимо обеспечить доступ к API-серверу по IP.

При наличии межсетевых экранов или прокси — необходимо внести адрес сервера в разрешённые.

4.4. Безопасность на клиентской стороне

Не допускается модификация файлов приложения.

Доступ к данным, генерируемым приложением: данные не сохраняются на диск, т.к. обработка происходит в оперативной памяти.

В случае использования прокси или VPN — необходимо протестировать стабильность соединения с сервером.