Основные технические решения

1. Описание общей архитектуры ПО

Представленное программное обеспечение (ПО) представляет собой комплексную систему видеоаналитики, предназначенную для мониторинга соблюдения техники безопасности (ТБ) на предприятии.

Рассмотрим упрощенную схему основных компонентов ПО на рисунке ниже.

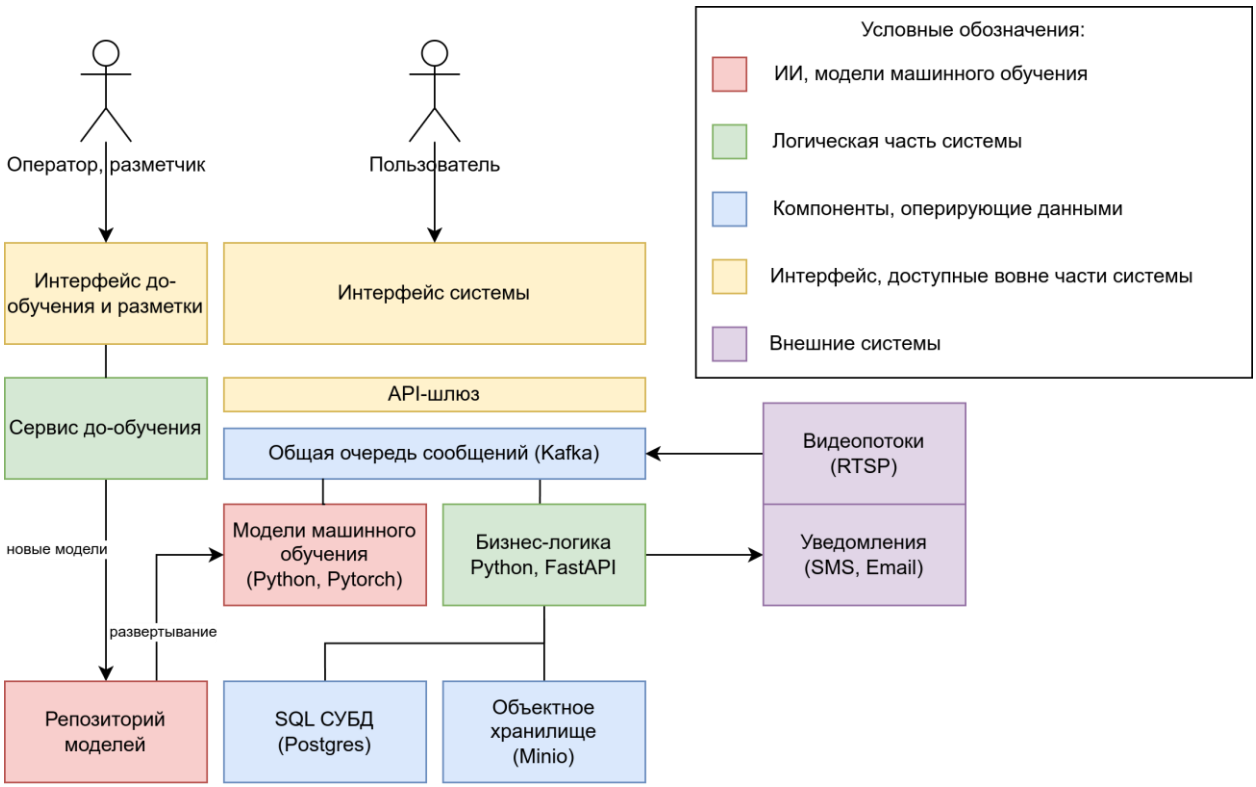

Рисунок 1.1 — Верхне-уровневая архитектура системы

Архитектура системы состоит из нескольких ключевых подсистем:

Интерфейсные элементы:

- Пользовательский интерфейс системы в браузере (веб-интерфейс)
- API-шлюз для приема запросов и управления взаимодействиями.
- Интерфейс для до-обучения и до-разметки

Инфраструктурные элементы

- Общая шина данных для межмодульного взаимодействия (Kafka)
- Модули бизнес-логики и до-обучения

- Модели машинного обучения (ИИ)
- Хранилище данных и моделей

Программные платформы и библиотеки (технологический стек):

- Backend: Python, FastAPI.
- DevOps: Docker, Kubernetes
- ML и CV библиотеки: PyTorch, OpenCV
- Хранилища данных: PostgreSQL (Postgres Pro), хранилище объектов (Minio).
- Системы очередей сообщений: Kafka
- Мониторинг и диагностика (отдельно, опционально):
  ELK-stack (Elasticsearch, Logstash, Kibana).
- Инструменты разметки данных: Label Studio

Принцип работы системы:

- Видео-камеры, находящиеся в одной сети с системой, транслируют RTSP-видеопотоки
- Видеопотоки разбиваются на кадры, кадры помещаются в очереди сообщений в Kafka. Часть кадров может быть пропущен, для достажения нужно частоты (fps)
- Кадры в очередях анализируются многоступенчатой системой с помошью моделей машинного обучения (ИИ), результаты анализа складываются в соответствующие очереди сообщений.
- Модуль бизнес-логики инерпретирует результаты, и производит соответсвующие записи в СУБД системы, в астности, о нарушениях правил ношения СИЗ.
- При необходимости, система может отправлять sms и email-ведомления об этих нарушениях заинтересованным получателям.
- Пользователи взаимодействуют с системой через интерфейс или по API, полчая от система информацию о нарушениях, отчеты, и дугую информацию.

Связь между модулями обеспечивается через общую шину данных (Kafka). Видеопотоки, данные о событиях и результатах анализа передаются асинхронно через очереди, гарантируя отказоустойчивость и масштабируемость.

2.1. Расширенное описание архитектуры системы

На рисунке 2.1 представлена схема целевой архитектуры системы видеоаналитики, которая включает в себя следующие подсистемы (модули):

Сервисы основной функциональности

Данные сервисы обеспечивают выполнение ключевых задач системы, связанных с мониторингом и контролем техники безопасности. В их состав входят следующие модули:

- модуль обработки внешнего API — отвечает за взаимодействие системы с внешними источниками данных (например, получение видеопотоков или других данных);
- модуль детектирования нарушений техники безопасности (ТБ) — использует модели машинного обучения для анализа видеопотоков и обнаружения потенциальных нарушений (например, отсутствие спецодежды или нарушение порядка действий);
- модуль работы с классификатором ML-моделей — обеспечивает выбор и применение наиболее подходящей модели для анализа конкретных видеопотока или ситуации;
- модуль работы с нарушениями — фиксирует, хранит и обрабатывает автоматически выявленные нарушения техники безопасности (на основе настраиваемых порогов уверенности), а также позволяет администратору через интерфейс и по API помечать нарушения как ложноположительные;
- модуль уведомлений — отправляет оповещения ответственным лицам о нарушениях через различные каналы (Email, SMS и т.д.);

Интерфейс системы реализован на русском языке. Система учитывает все требования, предъявляемые к интерфейсу пользователя:

- время отклика не превышает стандартных требований (5 секунд);
- интерфейс пользователя поддерживает русский язык в качестве основного (включая диалоговые и сервисные окна);
- предоставление информации в разделе «Помощь» осуществляется на русском языке.

Сервисы администрирования

Данные сервисы предназначены для управления системой и обеспечения её стабильной работы. Они включают следующие модули:

- модуль администрирования — предоставляет интерфейс для настройки системы, управления пользователями и других административных задач. Здесь же осуществляется включение-выключение дополнительной функциональности и модулей, которые могут потребоваться в дальнейшем;
- модуль внешних интеграций — позволяет интегрировать систему с корпоративными системами (например, с LDAP для аутентификации сотрудников);
- модуль централизованного мониторинга и диагностики системы — обеспечивает постоянный мониторинг состояния всех компонентов системы и отправку уведомлений в случае сбоев или ошибок;
- модуль лицензирования — управляет лицензиями системы, отслеживает активные лицензии и возможности расширения функционала.

Сервис уведомлений

Сервис отправляет уведомления через следующие каналы:

- Email-уведомления — отправка сообщений через email-провайдеров;

- SMS-уведомления — отправка оповещений через SMS;
- системные уведомления — уведомления в интерфейсе пользователя.

Общая шина данных

Для обмена данными между компонентами системы используется общая шина данных, построенная на основе систем очередей сообщений (Kafka, но возможно использование NATS или RabbitMQ). Это обеспечивает асинхронный обмен данными и надежность системы при высоких нагрузках.

Сервис анализа данных

Основной модуль системы для анализа видеопотоков включает:

- модуль компьютерного зрения — отвечает за обработку видеопотоков и выявление объектов или событий, важных для контроля ТБ;
- модуль машинного обучения (ML) — использует предварительно обученные ML-модели для анализа видеоданных и определения нарушений.

Слой данных

Все данные, обработанные системой, хранятся в различных хранилищах в зависимости от их типа:

- архив оперативных данных — хранение временных данных для быстрого доступа к результатам анализа;
- хранилище результатов видеомониторинга — сохраняет данные о событиях и нарушениях, выявленных в ходе видеомониторинга;
- хранилище событий и справочных данных — содержит метаданные и события, связанные с системой;
- хранилище для аналитики и отчетов — длительное хранение аналитических данных и отчетов по системе.

Модуль позволяет напрямую обращаться к базе данных и выгружать данные, а также использовать различные фильтры, которые не включены в исходный интерфейс.

Компонент обучения и дообучения моделей

Данный компонент играет ключевую роль в развитии и повышении эффективности системы, выполняя следующие задачи:

- обработка RTSP-потоков и размеченных данных — позволяет извлекать данные из видеопотоков для дальнейшей их разметки и использования в обучении моделей;
- разметка данных — этап подготовки данных, на котором они размечаются для использования в обучении и дообучении моделей машинного обучения. В качестве данных, в частности, используются нарушения, ранее отмеченные администратором как «ложноположительные»;
- обучение и дообучение ML-моделей — на основе размеченных данных ML-модели обучаются или дообучаются в целях повышения точности их работы.

Каталог моделей включает в себя все доступные версии ML-моделей в зависимости от задачи.

Настоящий модуль позволяет добавлять в систему новые типы объектов (из перечня детектируемых), например, перчатки другого цвета. Для этого необходимо провести разметку новых данных, используя инструменты, перечисленные в пункте 3.3.1.

Взаимодействие компонентов

Пользователи системы взаимодействуют с ней через web-интерфейс, который предоставляет им доступ к функциональности мониторинга и анализа. Администраторы, со своей стороны, управляют настройками системы также посредством использования web-интерфейса. Система получает данные из различных источников, таких как камеры видеонаблюдения или архивы видеозаписей. На основе этих данных, с помощью методов компьютерного зрения и машинного обучения, производится анализ с целью выявления нарушений техники безопасности. Результаты анализа передаются в модуль уведомлений, который отправляет оповещения ответственным лицам.

Таким образом, представленная архитектура системы видеоаналитики обеспечивает отказоустойчивый, масштабируемый и эффективный контроль соблюдения техники безопасности на предприятии с использованием современных технологий обработки видео и машинного обучения.

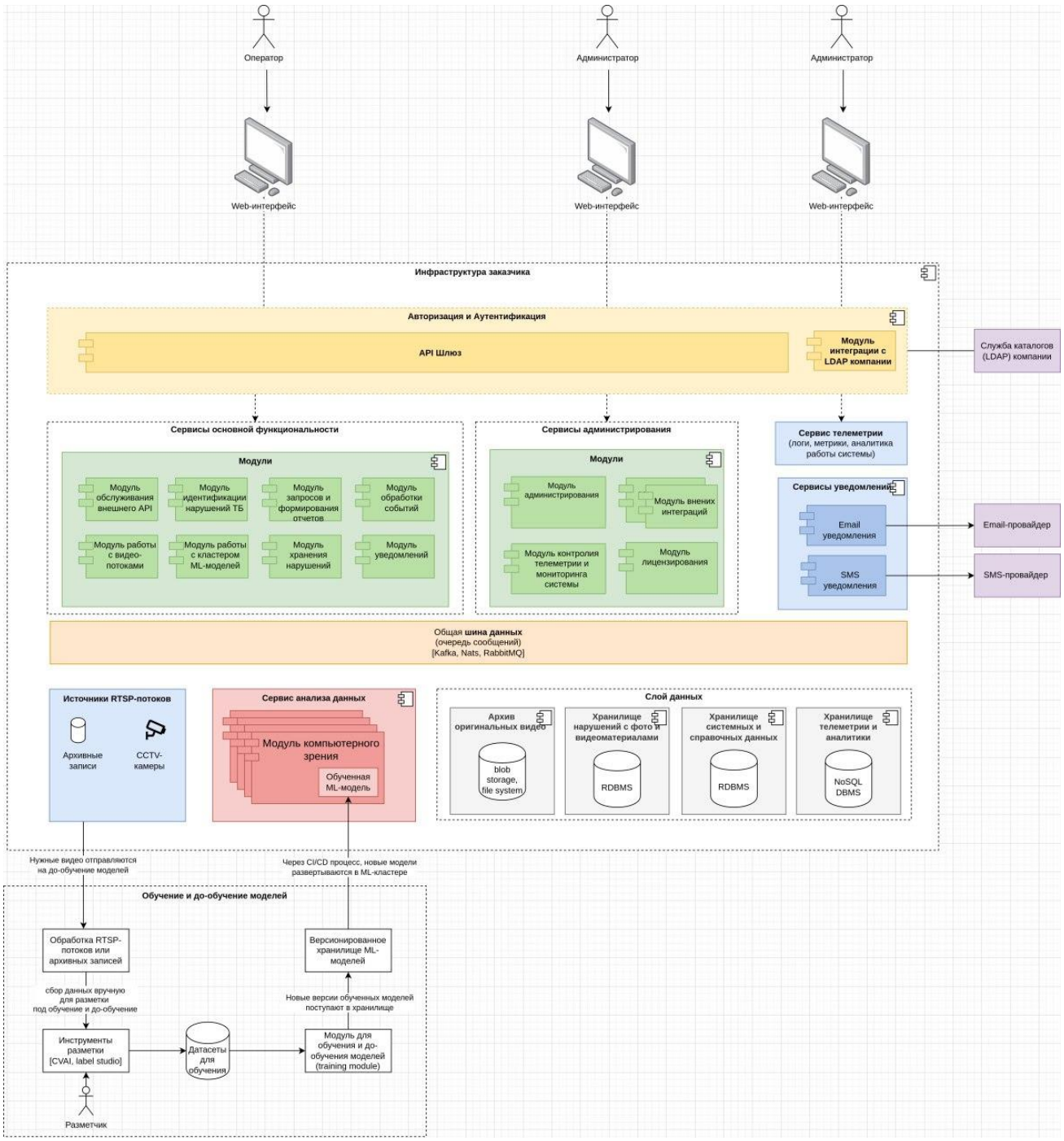

Рисунок 2.2 — Расширенная архитектура системы

2.2. Решение по бизнес – процессам (далее по тексту – «функции»), реализуемым в рамках Проекта.

Перечень функций, реализуемых в рамках проекта, в разрезе блоков представлен в таблице 2.1. и на рисунке 2.2.

Таблица 2.1. Основные функции системы

№ п/п	Функция	Название блока	Краткое описание
1	Получение видеопотоков	Модуль работы с видеопотоками	Считывание видеопотока IP-камер через RTSP-протокол в реальном времени и его передача для анализа и выявления нарушений
2	Идентификация нарушений ТБ	Модуль идентификации	Анализ видеопотока (разделение на отдельные кадры) с целью детекции объектов, событий и выявления нарушений. Данный процесс включает в себя: • обнаружение возможных нарушений в экипировке работников предприятия и работников подрядных организаций; • обнаружение возможного хранения предметов (материалов и оборудования) в зонах, не предназначенных для хранения; • обнаружение возможных нарушений при работе со станками, имеющих защитный кожух (поднят защитный кожух / отсутствует персонал при работающем станке)
3	Обработка событий	Модуль обработки событий	Получение данных идентификации нарушений ТБ и возможность работать с событиями нарушений (просматривать реестр необработанных событий и реестр выявленных нарушений, а также помечать ложные срабатывания)
4	Формирование отчетов	Модуль запросов и формирования отчетов	Формирование отчетов о выявленных событиях и нарушениях. Модуль позволяет просматривать отчеты, сформированные на основе данных, полученных в результате работы интеллектуальных методов. Данный процесс включает в себя: • выбор пользователем параметров отчета; • формирование отчета по данным системы; печать и/или сохранение отчета на доступный пользователю ресурс.
5	Хранение данных	Модуль хранения данных	Подтвержденные инциденты записываются в базу данных Postgres Pro для дальнейшего анализа. Данный процесс включает в себя: • запись уведомлений о зафиксированных нарушениях в базу данных модуля уведомлений в установленном формате.

№ п/п	Функция	Название блока	Краткое описание
6	Уведомления	Модуль уведомлений	• Формирование уведомлений и их рассылка по предопределенным коммуникационным каналам. Сервис рассылки выполняет отправку уведомлений по электронной почте или посредством SMS-рассылки. По факту успешной отправки уведомления помечаются как отправленные.
7	Администрирование	Модуль администрирования	Предоставляет интерфейс для настройки системы, управления пользователями и других административных задач.
8	Ручная проверка и классификация нарушений	Модуль идентификации нарушений	Позволяет администратору в веб-интерфейсе системы пересмотреть автоматически добавленные нарушения и вручную отметить их как ложноположительные по своему экспертному выбору.

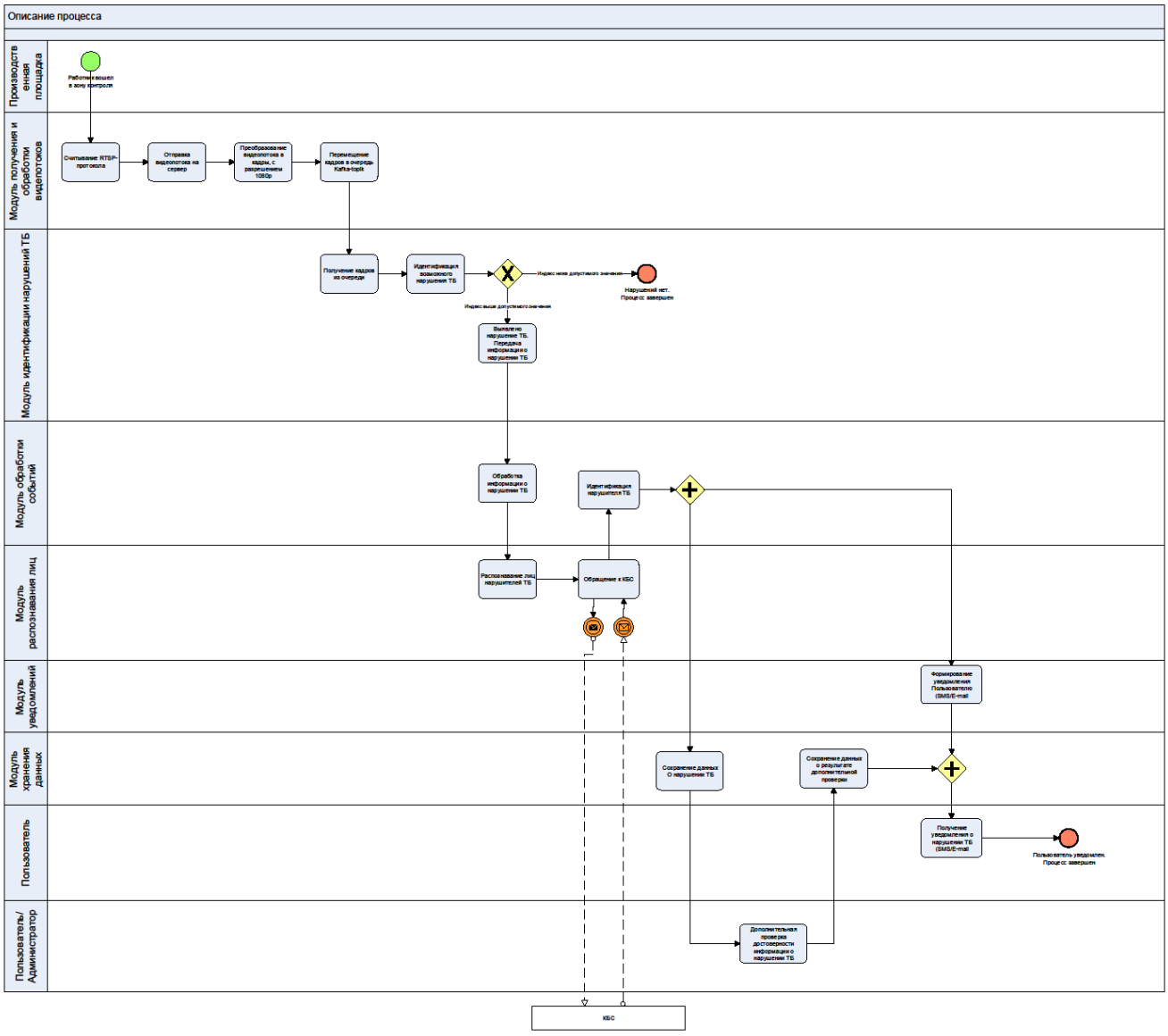

Рисунок 3.2 — Описание сквозного процесса

2.2.1 Получение и обработка видеопотоков

Получение и обработка видеопотоков представляет собой процесс получения стабильного видеопотока с установленных камер видеонаблюдения через RTSP-протокол на серверы, где происходит их обработка. Соответствующий модуль позволяет просматривать трансляции с подключенных к системе камер, в том числе с наложенными на изображение результатами работы интеллектуальных методов.

Основные этапы получения и обработки видеопотоков:

- подключение системы к камерам видеонаблюдения по RTSP-протоколу;
- поступление видеопотоков на сервер;
- преобразование видеопотока в кадры с разрешением 1080p для последующего анализа;
- перемещение кадров в соответствующую очередь (kafka-topic).

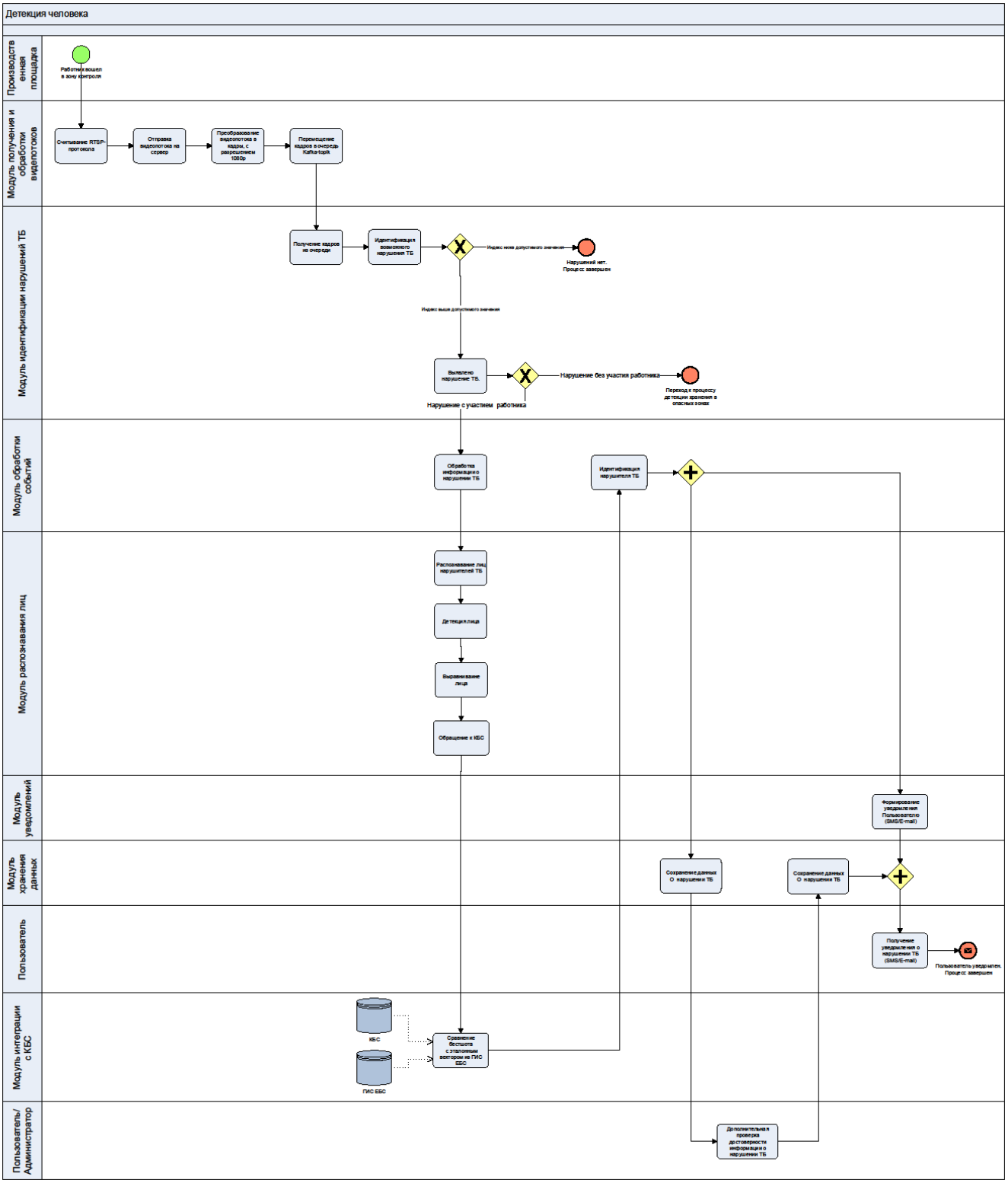

Рисунок 2.3 — Схема процесса (Детекция человека)

2.2.2 Идентификация нарушений ТБ

Идентификация нарушений ТБ представляет собой процесс анализа видеопотока, разделение его на отдельные кадры и помещение в очередь (Kafka) с целью детекции и распознавания объектов, а также выявления нарушений.

Процесс идентификации нарушений ТБ включает в себя:

- обнаружение нарушений в экипировке работников предприятия и работников подрядных организаций;
- обнаружение хранения предметов (материалов и оборудования) в зонах, не предназначенных для хранения;
- обнаружение нарушений при работе со станками, имеющими защитный кожух – система определяет, поднят ли защитный кожух, а также фиксирует отсутствие персонала при работающем станке;
- выявление нарушений работниками периметра опасных зон.

Процесс идентификации ТБ состоит из следующих детекций (схематично отражен на рисунках 2.3. — 2.7.):

1. Выявление нарушений работниками правил использования средства индивидуальной защиты (далее - СИЗ)

Данный процесс включает в себя:

- считывание видеопотока и его разбивку на кадры (интервал пропуска кадров составляет 10 кадров/сек);
- детекцию и трекинг объектов (осуществляется в одном модуле с целью минимизации затрат на передачу кадров). К объектам детекции относятся люди и их перемещения, СИЗ;
- сопоставление местонахождения СИЗ и работников и детекция нарушений. На основании ключевых метрик распознавания человека система определяет, в каком месте кадра располагается СИЗ, далее система производит проверку (посредством метрики оценки IoU — Intersection over Union) предполагаемого местонахождения человека и сопоставляет с местонахождением СИЗ (то есть, если, например, маска висит на шее и не закрывает лицо, система фиксирует нарушение);
- агрегацию нарушений по каждому человеку в кадре в течение n минут;
- анализ общей уверенности агрегированных нарушений, который включает в себя сопоставление с максимальным порогом и определение факта нарушения;
- необходимость дополнительного подтверждения нарушения оператором при пограничном уровне уверенности;
- классификацию нарушений;
- фиксацию факта нарушения и квалификацию его как события с последующей передачей в соответствующий модуль.

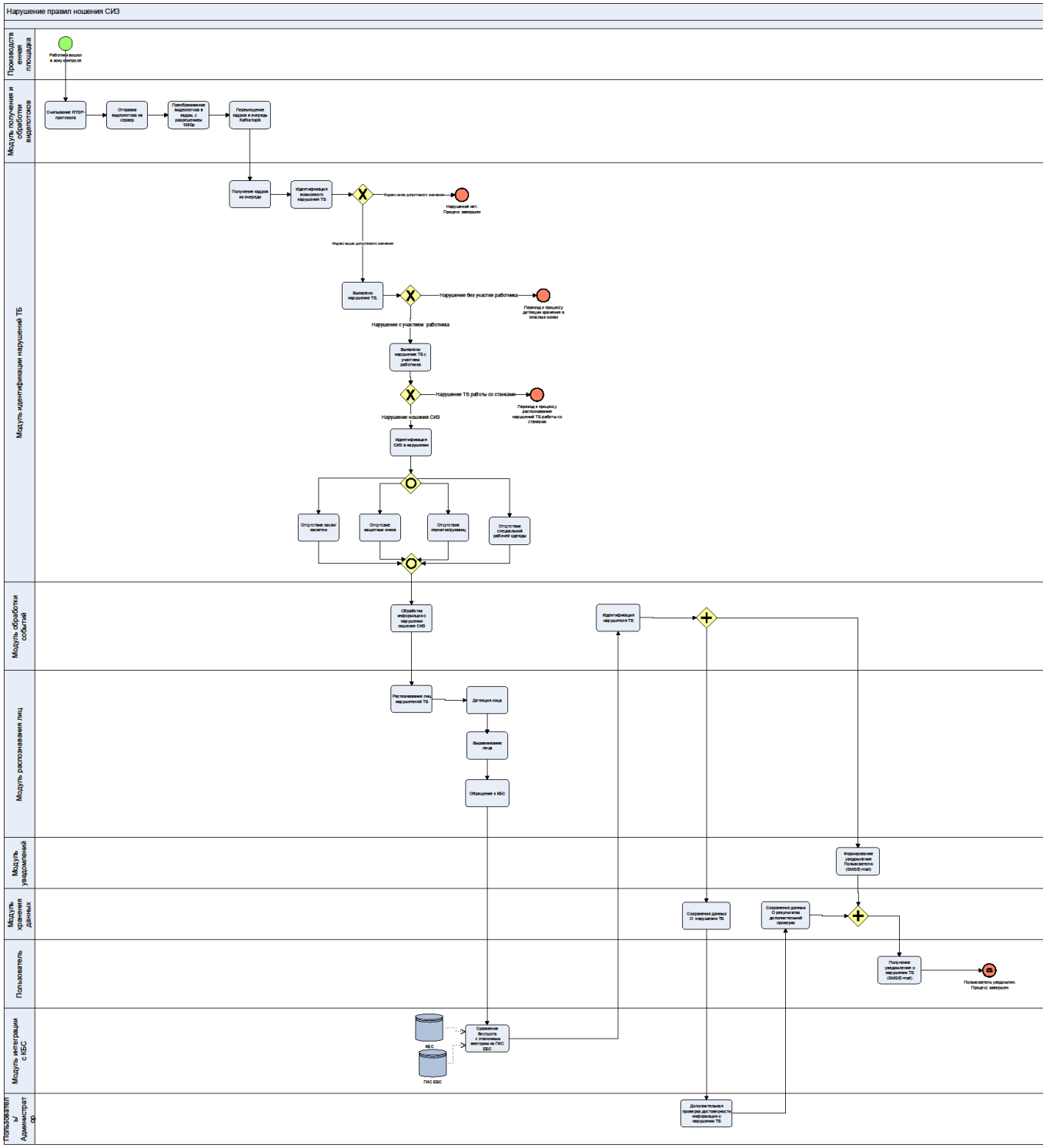

Рисунок 2.4 — Схема процесса (Нарушение правил ношения СИЗ)

2. Выявление нарушений работниками периметра опасных зон

Периметры опасных зон заранее определены для каждой камеры и заданы в интерфейсе системы при настройке.

Данный процесс включает в себя:

- считывание видеопотока и его разбивку на кадры (интервал пропуска кадров составляет 10 кадров/сек);
- детекцию и трекинг человека - определение области, которую занимает человек на кадре;

- определение, находится ли человек внутри периметра опасной зоны;
    - вычисление, пересекается ли область, занимаемая человеком, с опасной зоной, установленной для данной камеры;
    - в случае установки факта пересечения определяется достоверность (уверенность модели) в том, что человек действительно находится в опасной зоне;
- агрегация нарушений по каждому человеку в течение n минут;
- квалификацию нарушения ТБ, при его фиксации, в качестве события и последующая передача в соответствующий модуль.

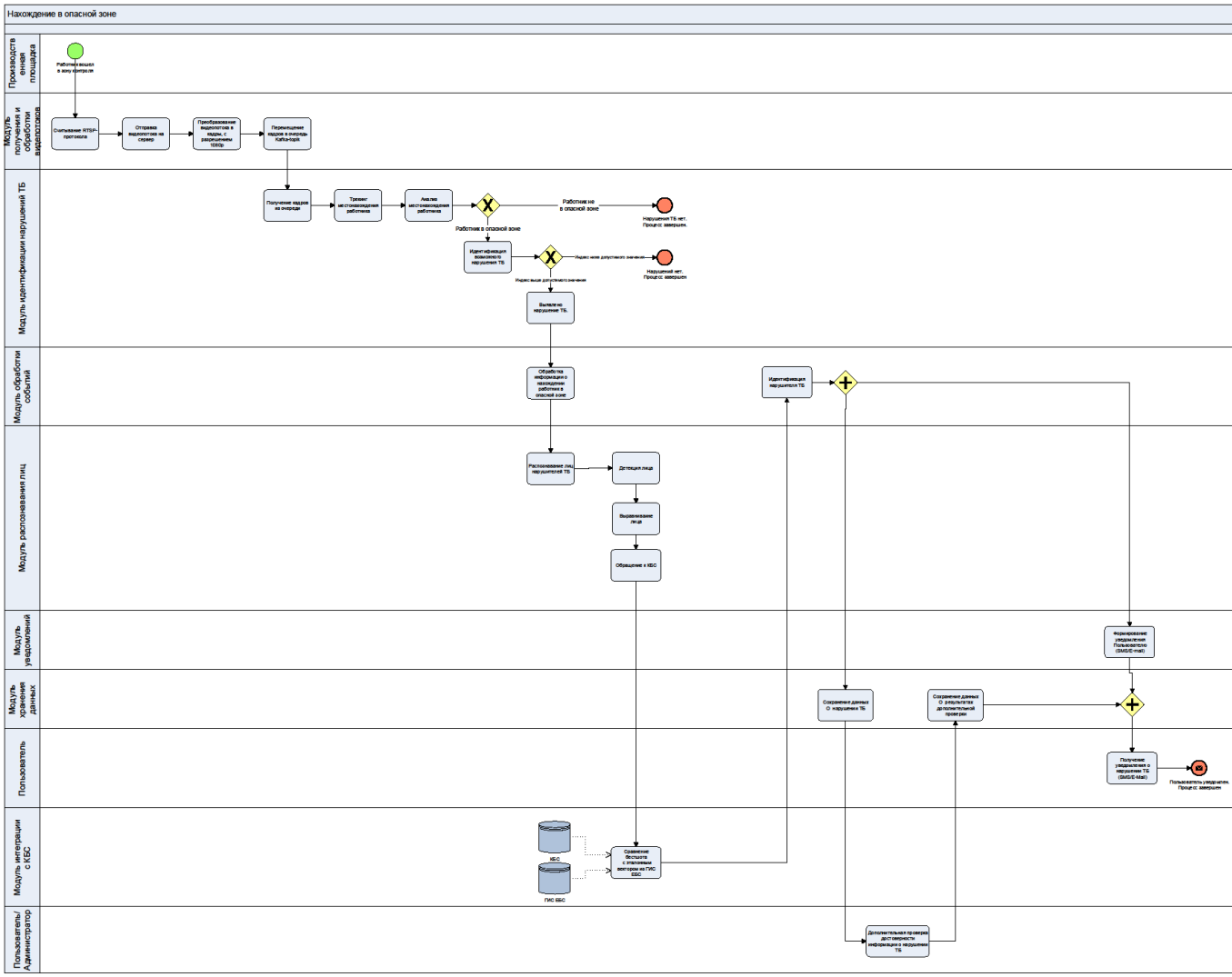

Рисунок 2.5 — Схема процесса (Нахождение в опасной зоне)

3. Выявление факта нарушения работниками правил техники безопасности при работе с токарными станками

Данный процесс включает в себя:

- считывание видеопотока и его разбивку на кадры (интервал пропуска кадров составляет 10 кадров/сек);
- детекцию и трекинг человека;
- сопоставление местоположения человека относительно ограничивающей рамки (в зоне функционирования оборудования);
- анализ положения кожуха;
- анализ состояния работы станка (светофор);
- сопоставление расположения человека относительно зоны функционирования оборудования;
- агрегацию нарушений по каждому человеку в течение интервала в n минут;
- квалификацию нарушения, при его фиксации, в качестве события и последующая передача в соответствующий модуль.

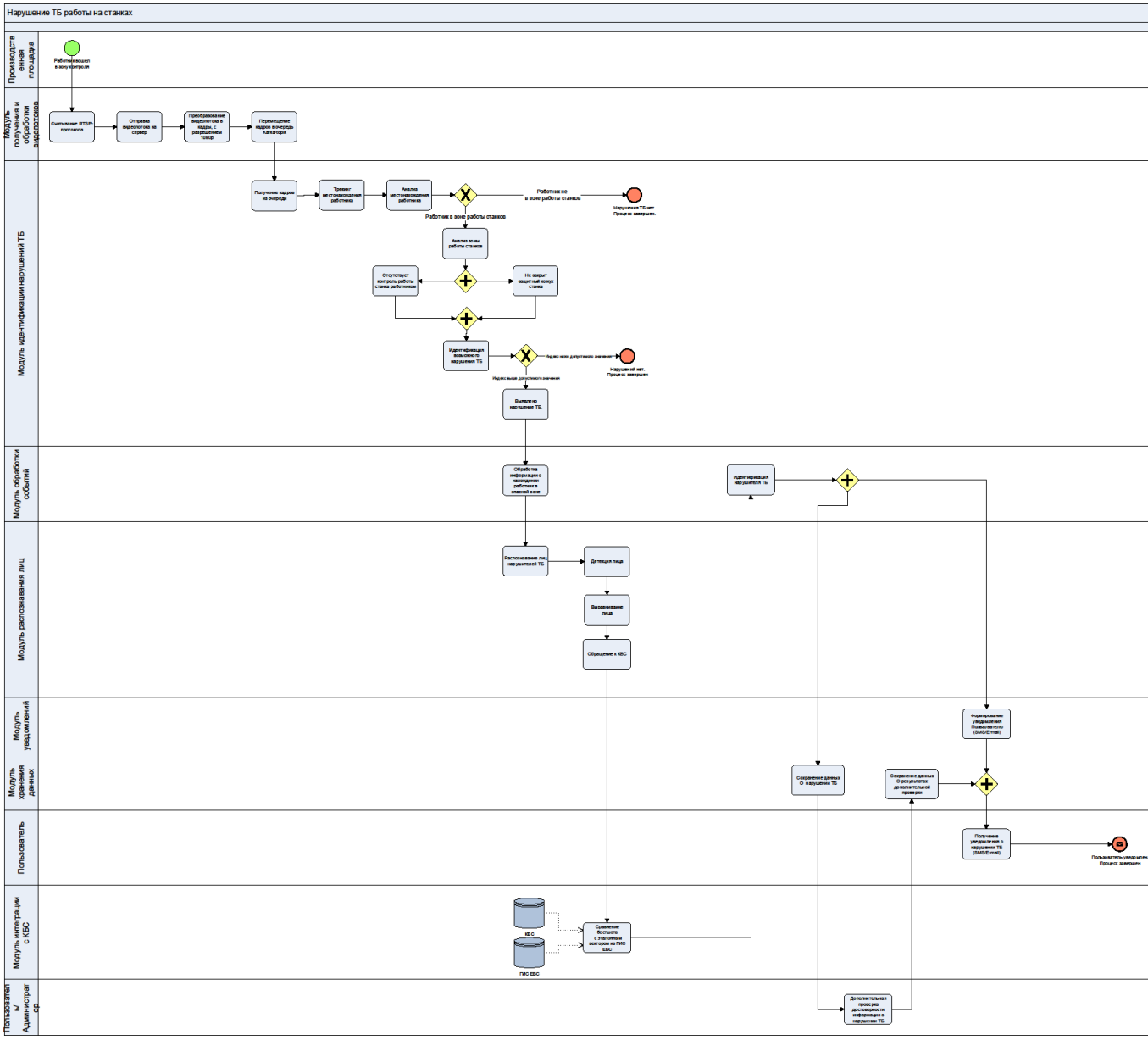

Рисунок 2.6 — Схема процесса (нарушение ТБ работы на станках)

4. Выявление фактов хранения предметов (оборудования, запчастей и материалов) в зонах, не предназначенных для хранения.

Данный процесс включает в себя:

- считывание видеопотока и его разбивку на кадры (интервал пропуска кадров составляет 10 кадров/сек);
- определение факта расположения в зоне хранения постороннего предмета путем сопоставления с эталонным кадром и анализа допустимого времени нахождения в зоне детекции. При этом система позволяет выделить данный предмет как допустимый в режиме редактирования на заданный интервал времени;
- квалификацию нарушения ТБ, при его фиксации, в качестве события и последующая передача в соответствующий модуль.

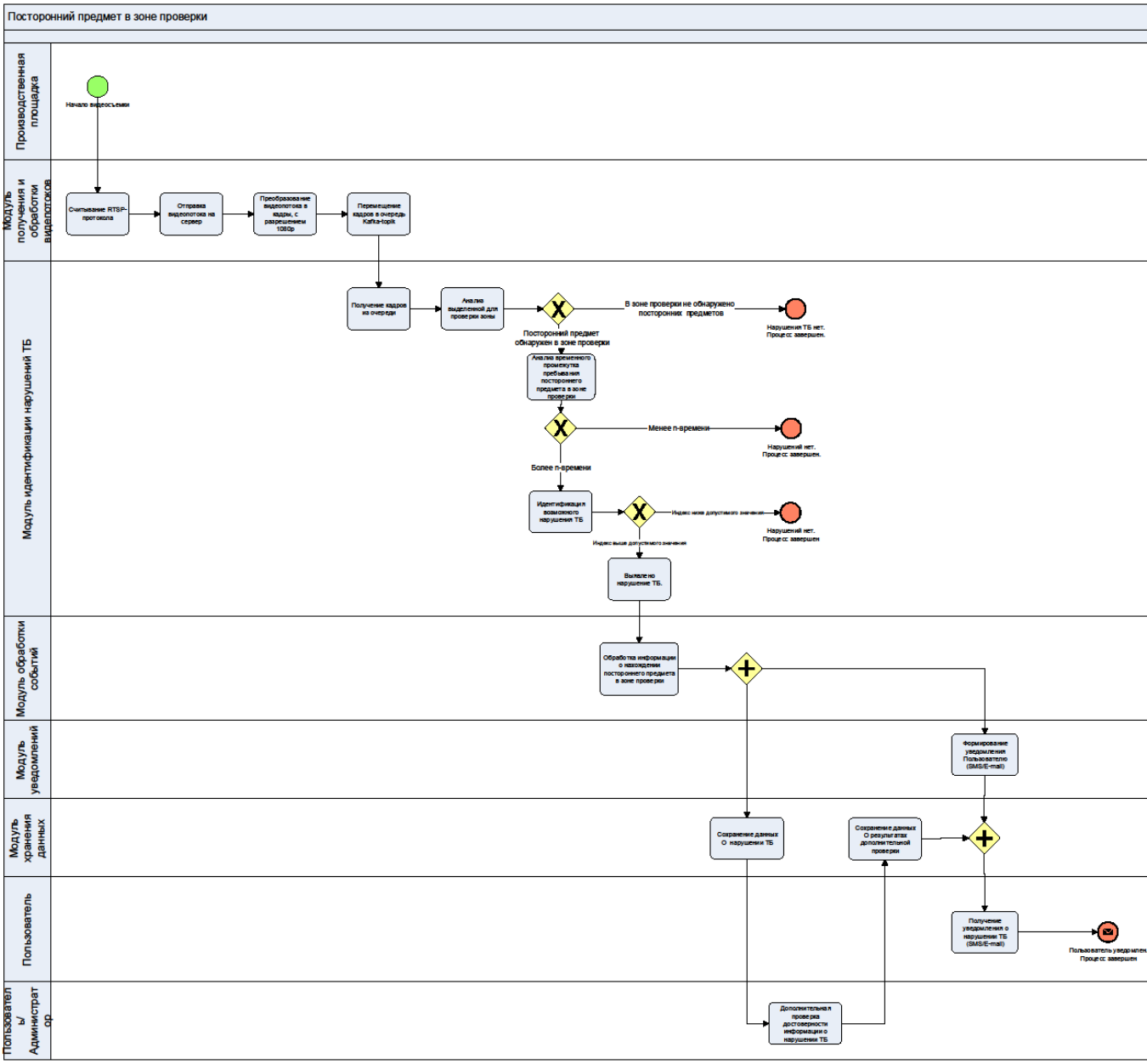

Рисунок 2.7 — Схема процесса (посторонний предмет в зоне проверки)

2.2.3. Обработка событий

Обработка событий представляет собой автоматический процесс фиксации фактов нарушений ТБ.

Модуль анализирует события, получаемые от модуля идентификации нарушения ТБ, и обрабатывает их. Полученное по итогам анализа событие содержит следующую информацию:

- номер камеры, зафиксировавшей событие;
- дата и время события;
- тип события;
- индекс вероятности нарушения;
- фотографию и видеофрагмент события.

Модуль включает в себя пользовательский web-интерфейс, позволяющий просматривать реестр необработанных событий и реестр выявленных нарушений.

Инструментарием web-интерфейса предусмотрена возможность фильтрации и сортировки реестров нарушений и событий. Нарушение содержит следующую информацию:

- дата и время события;
- наименование зоны /номер камеры, зафиксировавшей событие;
- наименование предприятия;
- классификацию нарушения (факт отсутствия каски/каскетки, защитных очков, перчаток, специальной рабочей одежды; факт нарушения периметра опасной зоны; факт нарушения правил использования зоны, не предназначенной для хранения; факт нарушения правил работы с токарным станком);
- индекс вероятности нарушения, %;
- фотографию, соответствующую времени зафиксированного нарушения.

Для каждого события модуль сопоставляет индекс вероятности нарушения с допустимым значением, указанным в настройках системы. При идентификации индекса выше настроенного допустимого значения или определения равным ему, модуль обработки событий фиксирует факт нарушения. Событие сохраняется как нарушение и отображается в соответствующем реестре. При идентификации индекса ниже настроенного допустимого значения, событие помечается как не являющееся нарушением.

2.2.4. Формирование отчетов

Система формирования отчетов обеспечивает формирование аналитических отчетов по различным событиям. Процесс формирования отчетов с использованием web-интерфейса с соответствующими правами доступа позволяет задавать параметры для построения отчета и выгружать сформированный отчет. Содержание формируемого отчета соответствует выбранным параметрам отчета.

Схема процесса формирования и выгрузки отчетности показана на рис. 2.8.

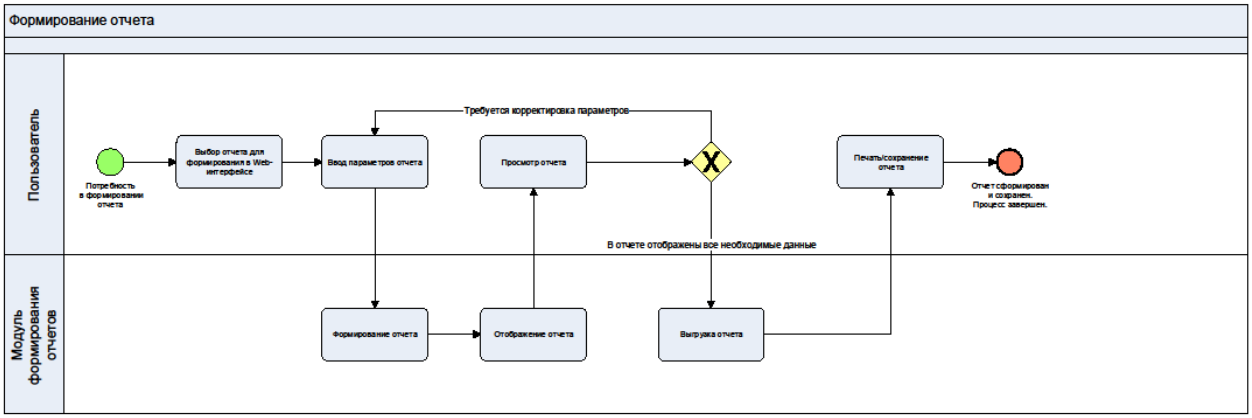

Рисунок 2.8 — Схема процесса (формирование отчета)

Отчет содержит сведения обо всех нарушениях ТБ за выбранный период по всем видеокамерам, доступным пользователю в рамках назначенных ему прав доступа.

Настраиваемые параметры:

- период/дата и время;
- объект (предприятие);
- зона (видеокамера(-ы) в цеху, где нарушение);
- тип нарушения;
- По ID (при наличии) нарушителя.

Сформированный отчет имеет следующие атрибуты:

- период;
- характеристики нарушения:
  - дата и время;
  - наименование объекта;
  - номер камеры/зоны;
  - тип нарушения;
  - индекс вероятности нарушения (при пограничных значениях — факт подтверждения оператором);
  - кадр нарушения;
  - ссылки на видеофрагмент и видеокадр нарушения.

2.2.5. Хранение данных

Система хранения данных обеспечивает хранение фотографий и видеофрагментов нарушений. Процесс хранения данных заключается в получении зафиксированных событий и отчетов и размещении данных о нарушениях в системе. По каждой фотографии и каждому видеофрагменту в модуле хранения фиксируются следующие сведения:

- дата и время;
- наименование зоны /номер камеры, зафиксировавшей событие;
- наименование предприятия;
- классификация нарушения (факт отсутствия каски/каскетки, защитных очков, перчаток, специальной рабочей одежды; факт нарушения периметра опасной зоны; факт нарушения правил использования зоны, не предназначенной для хранения; факт нарушения правил работы с токарным станком);
- индекс вероятности нарушения, %;
- изображение, соответствующее по времени зафиксированному нарушению.

Каждому нарушению в системе соответствуют фотография и видеофрагмент в хранилище.

Система хранения данных позволяет производить просмотр сохраненных фотографий и видеофрагментов.

Срок хранения фото и видеоматериалов реестров нарушений устанавливается длительностью в один год по умолчанию или настраивается для каждой камеры пользователем соответствующей роли. По факту заполнения хранилища наиболее старые события удаляются, и записываются новые.

2.2.6. Уведомления

Процесс функционирования системы уведомлений заключается в формировании и отправке уведомлений в интерфейсе системы, а также посредством сервиса рассылки Email-сообщений и/или сервиса рассылки SMS.

Периодичность отправки, а также получатели (по ролям) задаются настройками системы.

Данные для формирования уведомлений поступают от модуля обработки событий и содержат в себе информацию о событии, индекс вероятности нарушения которого превышает допустимое значение или равен ему, с командой произвести отправку уведомления о факте нарушения ТБ.

Важные события, регистрируемые системой и хранящиеся в БД сервера видеоаналитики, периодически транслируются в БД системы рассылки сообщений.

2.2.7. Администрирование

Администрирование представляет собой процесс управления пользователями и правами доступа пользователей к системе на основе ролей доступа. В системе реализована ролевая модель доступа, которая предполагает наличие не менее двух ролей доступа, а именно: «Администратор», «Пользователь», а также роль «Администратор данных».

Пользователю с ролью «Администратор» предоставлены права доступа по настройке и конфигурированию системы. Пользователю с ролью «Пользователь» недоступны права доступа по настройке и конфигурированию системы. Пользователю с ролью «Администратор» предоставлены возможности загрузки размеченных данных в систему для ее дообучения.

Основные функции модуля администрирования:

- просмотр реестра пользователей, зарегистрированных в системе;
- возможность сортировки и фильтрации реестра пользователей;
- обеспечение регистрации новых пользователей;
- возможность изменения данных зарегистрированных пользователей, а также просмотра списка пользователей их удаления из списка;
- просмотр перечня прав в системе;
- возможность назначения пользователям права доступа и предоставления пользователю доступа к функциям в соответствии с набором назначенных ему прав;
- ограничение доступа пользователя к функциям системы при удалении соответствующего права;
- проверка прав доступа при совершении пользователем действий в системе;
- настройка параметров системы;
- возможность Администратора устанавливать допустимое значение, с которым сравнивается индекс вероятности нарушения, путем назначения допустимого значения в интервале от 0 до 100 %;

- возможность конфигурирования границ сцены (часть кадра, используемая для анализа) в рамках конкретной камеры и интеллектуального метода обработки.

2.2.8. Ручная проверка и классификация нарушений

Процесс ручной проверки нарушений происходит по необходимости и при участии администратора: если величина уверенности, выданная моделью, больше нижнего порогового значения min, но меньше верхнего порогового значения max, то нарушение требует дополнительного подтверждения администратором, как не до конца достоверное. В интерфейсе системы, где отображены нарушения, оператор может ознакомиться с фотографией нарушения, видеофрагментом и степенью уверенности системы в нарушении, а также пометить нарушение как ложноположительное, кликнув на соответствующую кнопку в интерфейсе. Это действие помечает событие нарушения специальным флагом в базе данных, и оно больше не отображается в отчетах по нарушениям, если этого специально не требуется.

Полученные таким образом данные в дальнейшем будут использованы для дообучения соответствующих МЛ-моделей.

2.3. Решения по архитектуре подсистем

Общая схема решения описана в пункте 2.1 (рис. 2.1.).

2.3.1. Группа модулей «бизнес-логики»

В таблице 2.2 представлены входящие требования по модулю.

Таблица 2.2 — Входящие требования по модулю

№ п/п	Раздел	Название модуля	Ссылка на реестр требований	Краткое описание
1	2.3.2.	Модуль обслуживания внешнего API	П.2.8. технического задания	Отвечает за прием и обработку запросов от внешних систем через API-интерфейс
2	2.3.3.	Модуль идентификации нарушений ТБ	П. 2.2. технического задания	Отвечает за анализ видеопотоков и выявление нарушений техники безопасности
3	2.3.4.	Модуль запросов и формирования отчетов	П. 2.6. технического задания	Отвечает за сбор данных о событиях, нарушениях и о состоянии системы, а также за генерацию отчетов по запросам пользователей или по расписанию
4	2.3.5.	Модуль обработки событий	П. 2.3. технического задания	Отвечает за обработку событий в системе

№ п/п	Раздел	Название модуля	Ссылка на реестр требований	Краткое описание
5	2.3.6.	Модуль работы с видеопотоками	П. 2.1. технического задания	Отвечает за прием, обработку и маршрутизацию видеопотоков от камер видеонаблюдения для анализа
6	2.3.7.	Модуль работы с кластером МЛ-моделей	П. 2.7. технического задания	Обеспечивает выбор нужной модели для анализа видеопотоков и занимается обновлением или дообучением моделей
7	2.3.8.	Модуль хранения данных	П. 2.4. технического задания	Обеспечивает хранение всей информации, генерируемой системой, включая результаты анализа видеопотоков, данные о нарушениях, метаданные, статистику и отчеты
8	2.3.9.	Модуль уведомлений	П. 2.5. технического задания	Обеспечивает отправку уведомлений о выявленных нарушениях и других важных событиях системы ответственным лицам через различные каналы (Email, SMS)

На рисунке 2.9 представлена концептуальная схема модулей.

Рисунок 2.9 — Схема сервисов основной функциональности

2.3.2. Модуль обслуживания внешнего API

Модуль обслуживания внешнего API отвечает за прием и обработку запросов от внешних систем через API-интерфейс, который включает в себя запросы на передачу видеопотоков, передачу данных для анализа, а также получение результатов работы системы.

Технологии: FastAPI для реализации RESTful API, асинхронная обработка запросов с использованием Python (AsyncIO).

Архитектура: Модуль поддерживает асинхронные операции для обработки запросов. Kafka используется для передачи видеопотоков и событий в другие модули системы. Данные могут сохраняться и извлекаться из Postgres Pro.

Пример реализации: API для получения RTSP потока от камеры и передачи его в модуль обработки данных через Kafka.

2.3.3. Модуль идентификации нарушений ТБ

Модуль идентификации нарушений ТБ – это основной модуль, который применяет модели машинного обучения для анализа видеопотоков и выявления нарушений техники безопасности (например, обнаружение отсутствия защитной каски или спецодежды).

Технологии: Python с библиотеками для компьютерного зрения и машинного обучения, такими как OpenCV, PyTorch или TensorFlow. FastAPI для интеграции с другими компонентами.

Архитектура: Модуль получает видеопотоки через Kafka, применяет модели машинного обучения для анализа кадров и отправляет результаты (например, метки нарушений) в другие модули или сервисы.

Пример реализации: Использование обученной модели для детектирования объектов на видеопотоке и последующая передача информации о нарушениях в Kafka для уведомления и хранения в базе данных (Postgres Pro).

2.3.4. Модуль запросов и формирования отчетов

Модуль запросов и формирования отчетов отвечает за сбор данных о событиях, нарушениях и о состоянии системы, а также за генерацию отчетов по запросам пользователей или по расписанию.

Технологии: FastAPI для создания конечных точек API, Postgres Pro для хранения и выборки данных, SQLAlchemy для работы с базой данных, Celery для планирования задач.

Архитектура: Модуль собирает данные о нарушениях и событиях из Postgres Pro, комбинирует их в отчеты и отправляет их пользователям по запросу (запрос выполняется в интерфейсе или по API). Этот модуль также позволяет сконфигурировать в интерфейсе

автоматическую рассылку этих же данных и отчетов в том же формате посредством Email или SMS. Конкретный формат, доступные расписания рассылки и правила определяются в частном техническом задании и корректируются на стадии реализации для удобства эксплуатации конечными пользователями. Расписание рассылки будет детально определено в процессе разработки модуля.

Пример реализации: Формирование PDF- или CSV-отчетов по количеству выявленных нарушений за определенный период и отправка отчетов посредством модуля уведомлений.

2.3.5. Модуль обработки событий

Модуль обработки событий отвечает за обработку событий в системе (например, фиксация фактов нарушений или иных инцидентов, связанных с техникой безопасности).

Технологии: FastAPI для управления событиями через API, Kafka для передачи сообщений о событиях в другие модули.

Архитектура: Модуль получает события из Kafka (например, информацию о нарушениях или сбоях), записывает их в базу данных и передает другим компонентам системы для дальнейших действий. Передаваемая таким образом информация включает в себя кадр или кадры с (потенциальным) нарушением, дату и время (timestamp), идентификатор камеры, дополнительную информацию о камере, информацию о детектированных объектах, информацию о предполагаемых нарушениях и порогах уверенности, остальную метаинформацию.

Пример реализации: Принятие сообщения из Kafka о новом нарушении, сохранение его в Postgres Pro, инициация отправки уведомления ответственным лицам.

2.3.6. Модуль работы с видеопотоками

Модуль работы с видеопотоками отвечает за прием, обработку и маршрутизацию видеопотоков от камер видеонаблюдения для анализа.

Технологии: FastAPI для получения и маршрутизации видеопотоков, Kafka для передачи видеопотоков в другие модули, OpenCV для предварительной обработки видеопотоков.

Архитектура: Видеопотоки принимаются через API и передаются через Kafka в модули анализа для обработки. Предусмотрено предварительное преобразование или сжатие видео.

Пример реализации: Прием RTSP потока с камеры, разбивка на кадры и передача в модуль компьютерного зрения для дальнейшего анализа.

2.3.7. Модуль работы с кластером ML-моделей

Модуль работы с кластером ML-моделей управляет кластером моделей машинного обучения, обеспечивает выбор нужной модели для анализа видеопотоков и занимается обновлением или дообучением моделей.

Технологии: FastAPI для управления моделями, PyTorch для работы с ML-моделями.

Архитектура: Модуль получает информацию о новых данных, автоматически выбирает модели на основе новых данных и применяет их для анализа. Kafka используется для передачи результатов от обученных моделей.

Пример реализации: Управление несколькими моделями для разных типов нарушений (например, каска, спецодежда), выбор наиболее подходящей модели для конкретного видеопотока.

2.3.8. Модуль хранения данных

Модуль хранения данных обеспечивает хранение всей информации, генерируемой системой, включая результаты анализа видеопотоков, данные о нарушениях, метаданные, статистику и отчеты.

Технологии: Postgres Pro для хранения структурированных данных, Kafka для передачи данных в модуль хранения. Предоставляемая этим модулем информация включает в себя кадр или кадры с нарушением, дату и время (timestamp), идентификатор камеры, дополнительную информацию о камере, информацию о детектированных объектах, информацию о предполагаемых нарушениях и порогах уверенности, остальную метаинформацию.

Архитектура: Модуль получает результаты от других компонентов системы и сохраняет их в базу данных для последующего использования в отчетах и анализе.

Пример реализации: Сохранение данных о нарушениях в базу Postgres Pro с возможностью их последующей выборки для отчетов.

2.3.9. Модуль уведомлений

Модуль уведомлений обеспечивает отправку уведомлений о выявленных нарушениях и других важных событиях системы ответственным лицам через различные каналы (Email, SMS).

Технологии: FastAPI для получения уведомлений от других модулей, интеграция с внешними сервисами для отправки уведомлений (например, SMTP для Email, SMS API для отправки сообщений).

Архитектура: Модуль получает данные о нарушениях и событиях через Kafka, обрабатывает их и отправляет уведомления через заранее настроенные каналы.

Пример реализации: Отправка Email уведомлений при обнаружении серьезного нарушения с указанием его описания и меток времени.

2.3.10. Сервисы администрирования

Сервис администрирования включает ряд модулей, проиллюстрированных на рисунке 2.10 и в таблице 2.3.

Таблица 2.3 — Входящие требования по модулю

№ п/п	Название процесса	Ссылка на реестр требований	Краткое описание
1	Модуль администрирования	П. 2.7. технического задания	Предоставляет интерфейсы и функциональность для управления конфигурациями системы, управления пользователями и ролями (изменение ролей, удаление пользователей), а также общих административных задач
2	Модуль внешних интеграций	П. 2.8. технического задания	Отвечает за интеграцию системы с внешними корпоративными сервисами, такими как LDAP, для управления аутентификацией пользователей, а также с системами управления уведомлениями и другими сторонними сервисами
3	Модуль контроля телеметрии и мониторинга системы	П. 2.10. технического задания	Отвечает за сбор и мониторинг телеметрии и состояния системы

№ п/п	Название процесса	Ссылка на реестр требований	Краткое описание
4	Модуль Лицензирования	П. 2.9. технического задания	Управляет лицензиями на использование системы или отдельных её компонентов

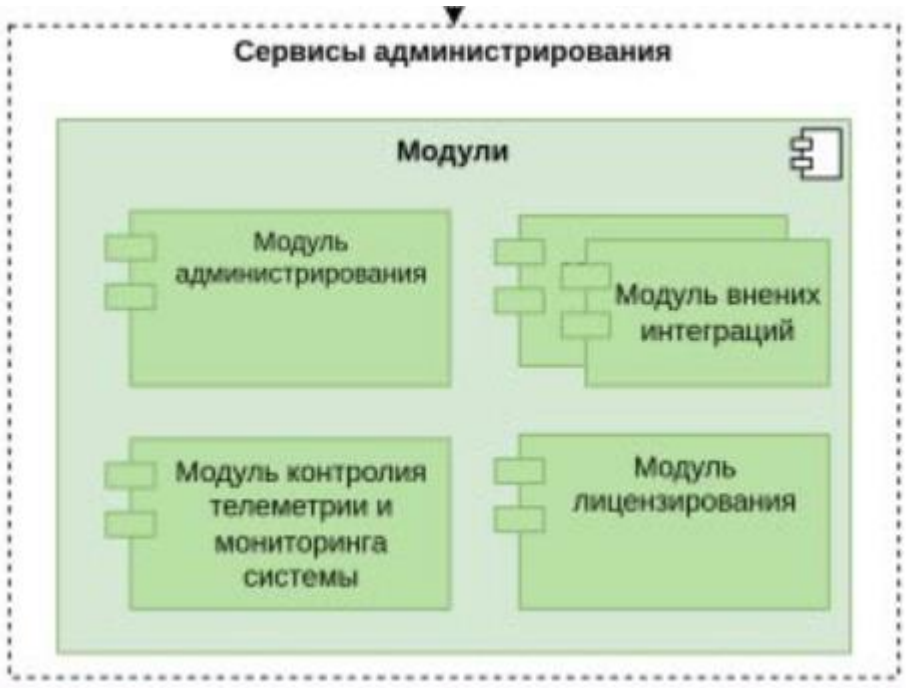

Рисунок 2.10 — Концептуальная схема группы модулей администрирования

2.3.11. Модуль администрирования

Модуль администрирования предоставляет интерфейсы и функциональность для управления конфигурациями системы, управления пользователями и ролями (изменение ролей, удаление пользователей), а также для общих административных задач. Он отвечает за централизованное управление системой. В системе предусмотрена гибкая ролевая модель: базовые роли доступа – «Администратор», «Администратор данных», «Пользователь». Пользователю с ролью «Администратор» предоставлены права доступа по настройке и конфигурированию системы. Пользователю с ролью «Пользователь» недоступны права доступа по настройке и конфигурированию системы. Пользователю с ролью «Администратор данных» доступны возможности загрузки размеченных данных для дообучения моделей. Могут быть добавлены дополнительные роли при необходимости. Набор прав для каждой роли может изменяться администратором: доступ к видеопотоку, доступ к группе видеопотоков (камер), конкретного цеху.

Технологии: FastAPI для построения административных интерфейсов (REST API для управления), SQLAlchemy для взаимодействия с Postgres Pro, Celery для выполнения фоновых задач (например, планирование административных задач).

Архитектура: Модуль получает данные из базы данных Postgres Pro для управления пользователями и ролями. Управление конфигурациями и системными настройками содержится в отдельной таблице конфигураций в базе данных. Kafka используется для отправки событий конфигурации другим модулям для синхронизации настроек.

Пример реализации: В интерфейсе администрирования производится управление доступом операторов и администраторов, обновление конфигурации системы (например, частоты уведомлений или уровня чувствительности моделей ML) и планирование задачи резервного копирования.

2.3.12. Модуль внешних интеграций

Модуль внешних интеграций отвечает за интеграцию системы с внешними корпоративными сервисами, такими как LDAP, для управления аутентификацией пользователей, а также с системами управления уведомлениями и другими сторонними сервисами.

Технологии: FastAPI для создания интерфейсов интеграции, Python библиотеки для работы с LDAP (например, ldap3), интеграция через Kafka для передачи данных о пользователях и ролях из внешних систем.

Архитектура: Модуль подключается к внешним сервисам через API или другие протоколы, такие как LDAP, для получения данных об учетных записях пользователей. Данные передаются через Kafka для синхронизации между модулями, а Postgres Pro используется для хранения локальных копий пользовательских данных.

Пример реализации: Интеграция с корпоративной системой управления доступом на основе LDAP, где новые пользователи автоматически добавляются в систему и получают доступ на основании их ролей в LDAP.

2.3.13. Модуль контроля телеметрии и мониторинга системы

Модуль контроля телеметрии и мониторинга системы отвечает за сбор и мониторинг телеметрии и состояния системы. Он предоставляет данные для диагностики работы всех компонентов системы, включая производительность, ошибки, использование ресурсов, а также общую информацию о системе.

Технологии: FastAPI для создания API для мониторинга и просмотра метрик системы, Kafka для передачи телеметрии и логов, ELK Stack (Elasticsearch, Logstash, Kibana) для хранения и анализа логов и метрик.

Архитектура: Логи и метрики системы передаются через Kafka в Elasticsearch, где они хранятся и анализируются. Kibana используется для визуализации телеметрии и создания дашбордов. FastAPI предоставляет административный интерфейс для запроса метрик, просмотра логов и их фильтрации. В качестве альтернативы можно использовать Prometheus и Grafana для мониторинга метрик производительности.

Пример реализации: Модуль собирает метрики использования CPU, памяти и количества ошибок для каждого модуля. В случае превышения порогов производительности или ошибок система отправляет уведомления администраторам через Kafka.

2.3.14. Модуль лицензирования

Модуль лицензирования управляет лицензиями на использование системы или отдельных её компонентов. Он отслеживает активные лицензии, позволяет администраторам управлять лицензиями и предотвращает использование системы при истечении срока действия лицензий.

Модуль лицензирования позволяет формировать лицензии для нового предприятия.

- лицензия на базовый функционал;
- лицензии на подключаемые модули;
- лицензии на подключаемые камеры;

Технологии: FastAPI для управления лицензиями через интерфейс API, Postgres Pro для хранения данных о лицензиях, Kafka для оповещения других модулей о статусе лицензий.

Архитектура: Лицензии и их метаданные хранятся в базе данных Postgres Pro. Модуль периодически проверяет валидность лицензий и уведомляет другие компоненты через Kafka в случае необходимости ограничения функционала. Также через FastAPI модуль предоставляет интерфейс для обновления или продления лицензий.

Пример реализации: Модуль отслеживает срок действия лицензии и при выявлении факта ее истечения в ближайший срок отправляет предупреждение через Kafka в модуль уведомлений, а также ограничивает доступ к ключевым функциям по факту её просрочки.

2.3.15. Архитектура модуля «хранения данных»

На рисунке 2.11 представлена концептуальная схема группы компонентов хранения данных.

Рисунок 2.11 — Схема группы компонентов хранения данных

2.3.16. Архитектура модуля обучения и дообучения моделей

На рисунке 2.12 показана концептуальная схема реализации обучения моделей.

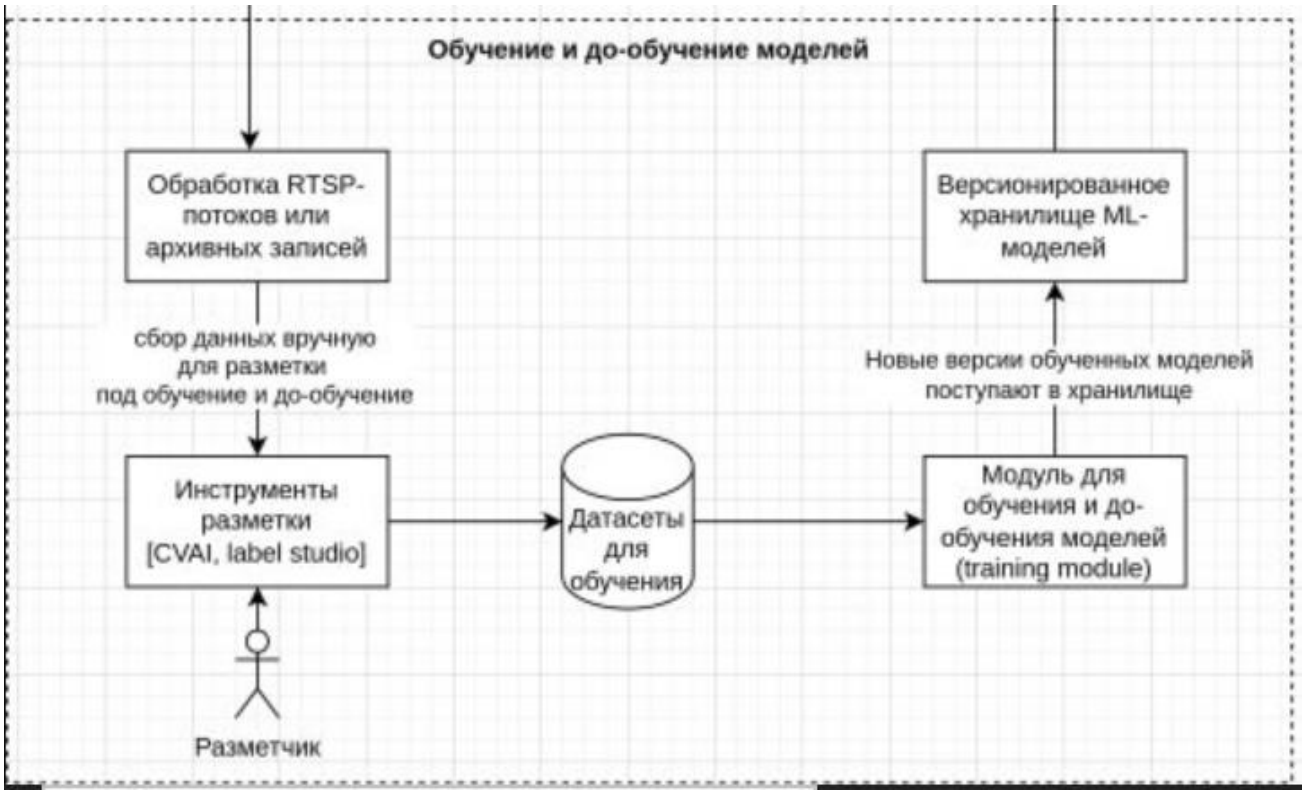

Рисунок 2.12 — Схема реализации обучения моделей

2.3.17. Обработка RTSP-потоков или архивных записей

Модуль обработки RTSP-потоков или архивных записей отвечает за получение и обработку видеопотоков в реальном времени с камер (RTSP-потоки) или архивных видеозаписей для последующей разметки и использования в обучении или дообучении моделей.

Технологии: Использование Python-библиотек для работы с видео, таких как opencv-python, для обработки RTSP потоков или архивных видеофайлов.

Пример: Видеопотоки захватываются и разбиваются на кадры. Данные кадры сохраняются для дальнейшей разметки, например, в формате PNG или JPEG.

2.3.18. Инструменты разметки (CVAT, Label Studio)

Модуль разметки — это инструменты для ручной разметки данных, которая используется для обучения и дообучения моделей.

Технологии: Использование популярных решений для разметки данных, таких как Label Studio (open-source) или CVAИ (платформа для компьютерного зрения). При использовании Label Studio, данные загружаются в платформу, где они размечаются вручную, а затем выгружаются в формате JSON или CSV.

Интеграция с Python: Используя API Label Studio, можно автоматизировать процесс загрузки данных для разметки и их последующей выгрузки для обучения.

2.3.19. Хранилище датасетов для обучения

Датасеты для обучения представляют собой размеченные данные(изображения), которые используются для обучения или дообучения моделей.

Технологии: Хранение датасетов в формате, подходящем для обучения моделей машинного обучения. Сами данные при этом хранятся в бинарном хранилище как minio, в локальной файловой системе или в базе данных (например, Postgres Pro).

Интеграция: после завершения разметки данные автоматически отправляются в хранилище, где они доступны для последующего обучения моделей.

2.3.20. Модуль для обучения и дообучения моделей

Модуль для обучения и дообучения моделей отвечает за обучение моделей машинного обучения на основе размеченных данных, которые хранятся в 2.3.19.

Технологии: Использование PyTorch для создания и обучения моделей машинного обучения (например, для детекции объектов, распознавания действий). Модуль обучения автоматически загружает датасеты для обучения из хранилища, проводит обучение или дообучение модели.

Интеграция: Скрипты на Python могут управлять процессом обучения через torch.utils.data.Dataset и другие инструменты PyTorch, а затем передавать модели в хранилище моделей.

2.3.21. Версионированное хранилище ML-моделей

Версионированное хранилище ML-моделей представляет из себя хранилище, которое поддерживает версионирование обученных моделей. Новые версии моделей сохраняются для последующего развертывания.

Технологии: для версионирования моделей используются такие инструменты, как MLflow, который предоставляет из себя интерфейс для отслеживания версий моделей и метаданных обучения. Хранение моделей организовано через minio-хранилища или локальные файловые системы.

Пример: по факту завершения обучения модель сохраняется в хранилище (например, в формате. pth для PyTorch), где каждая версия модели отслеживается и используется для развертывания или сравнения.

2.3.22. CI/CD процесс для развертывания моделей в ML-кластере

Автоматизированный процесс для развертывания новых версий моделей в продакшн-среде (ML-кластер).

Технологии: CI/CD пайплайны реализованы с использованием инструментов, таких как Jenkins, GitLab CI, или Kubernetes для развертывания моделей. По факту перемещения новой версии модели в хранилище пайплайн автоматически развертывает её на кластере.

Интеграция: Пайплайн использует API для загрузки новой модели из хранилища, а затем деплоит её на рабочие узлы, где она используется для видеоаналитики.

Основные шаги процесса:

- сбор данных: видео с камер в реальном времени или архивные записи поступают в систему для дальнейшей обработки. Модуль обработки RTSP-потоков или архивных записей преобразует эти видеопотоки в данные для разметки;
- разметка данных: разметчики используют инструменты, такие как Label Studio или CVAИ, чтобы вручную пометить нужные объекты или действия на видеокадрах. Эти размеченные данные затем сохраняются в датасетах для обучения;
- обучение моделей: размеченные данные поступают в модуль обучения, где с помощью PyTorch осуществляется процесс обучения или дообучения модели. Полученная модель сохраняется в версионированном хранилище;
- развертывание модели: после успешного обучения и сохранения новая версия модели через CI/CD процесс разворачивается в ML-кластере для использования в рамках промышленной эксплуатации.

Описанный в данном пункте CI/CD процесс развертывания моделей базируется на компонентах архитектуры, описанной в пунктах 2.3.18 (инструменты разметки), 2.3.19 (хранилище обучающих данных), 2.3.20 (скрипты обучения моделей), 2.3.21 (версионированное хранилище обученных моделей).

Все описанное выше является частью системы и частью её архитектуры. Схема процесса, описанного в пункте 2.3.22 содержится на рисунке 2.12. Остальные технические подробности указываются в частном техническом задании при необходимости их добавления.