Общество с ограниченной ответственностью
«Интеллектуальные системы разминирования»

Программа для ЭВМ «Dataset Studio»

ОПИСАНИЕ АРХИТЕКТУРЫ

Версия 1.0.0

Операционная система: РЕД ОС 8 (x86_64)

2026

1. Общие сведения

Настоящий документ содержит описание архитектуры программы для ЭВМ «Dataset Studio» версии 1.0.0 (далее — ПО): состав программных компонентов, их взаимодействие, модель данных, основные процессы обработки, внешние интерфейсы и меры защиты информации. Документ предназначен для специалистов, отвечающих за развёртывание, сопровождение и оценку ПО.

ПО представляет собой серверное веб-приложение с клиент-серверной архитектурой. Серверная часть функционирует под управлением РЕД ОС 8, клиентская часть выполняется в веб-браузере пользователя и не требует установки.

При проектировании ПО приняты следующие архитектурные принципы:

- минимум внешних зависимостей: для работы не требуются отдельный сервер СУБД, системы контейнеризации, брокеры сообщений и доступ в сеть Интернет;
- хранение данных на сервере организации-пользователя: изображения и разметка не передаются третьим лицам;
- встраиваемая СУБД SQLite для метаданных и файловая система для изображений;
- поставка в виде одного исполняемого файла в составе RPM-пакета, устанавливаемого штатными средствами РЕД ОС;
- разграничение доступа по ролям и регистрация всех значимых действий пользователей.

2. Структура ПО

ПО состоит из клиентской части, веб-сервера, серверной части и каталога данных (рисунок 1). Клиентская часть обращается к серверной через веб-сервер nginx; любой запрос сначала проверяется модулем аутентификации и только затем передаётся прикладным модулям.



Рисунок 1 — Структура программных компонентов ПО

Прикладные модули обращаются к данным только через слой хранения и модуль обработки изображений. Внешние сервисы, показанные пунктиром, используются только после их настройки администратором.

3. Программные компоненты

Серверная часть ПО реализована на языке Python с использованием фреймворка Flask и выполняется встроенным HTTP-сервером waitress в одном процессе с несколькими потоками обработки запросов. Все модули серверной части собраны в один исполняемый файл.

3.1. Модули серверной части

Модуль	Назначение
Модуль аутентификации и разграничения доступа	учётные записи, вход и сеансы, роли, проверка прав на каждый запрос, приглашения и восстановление пароля, хранение настроек почты и сервиса предварительной разметки в зашифрованном виде
Модуль проектов и партий	проекты и списки классов, загрузка изображений с исключением дубликатов, формирование партий, назначение исполнителей
Модуль разметки и приёмки	сохранение фигур разметки, отправка на проверку, приёмка, возврат, отбраковка и восстановление изображений, формирование набора данных
Модуль анализа баланса и экспорта	расчёт распределения объектов по классам и размерным группам, отбор с ограничениями, формирование архивов COCO, YOLO и DOTA
Модуль аналитики и журнала действий	тепловая карта расположения объектов, отчёт по разметчикам, регистрация действий пользователей и отмена изменений разметки

Модуль	Назначение
Модуль фоновых задач	импорт наборов COCO, формирование архивов экспорта и предварительная разметка в отдельных потоках с отображением хода выполнения
Модуль обработки изображений	определение размеров и формирование уменьшенных копий (320, 1280 и 1600 пикселей) с кэшированием на диске; используется библиотека OpenCV
Слой хранения данных	работа со встраиваемой СУБД SQLite: схема, транзакции, отдельное соединение для каждого потока

3.2. Клиентская часть

Клиентская часть разработана на HTML5, CSS3 и JavaScript без сторонних фреймворков и загружается с сервера ПО.

Страница	Назначение
Страница входа	вход в систему, создание первого администратора, установка пароля по ссылке
Рабочая область	список проектов, справочник классов, управление пользователями и настройками; внутри проекта — разделы загрузки, разметки, набора данных, приёмки, баланса, аналитики, экспорта и журнала
Редактор разметки	рисование и редактирование фигур на основе Canvas 2D, масштабирование, автоматическое сохранение, предварительная загрузка следующего изображения

3.3. Инфраструктурные компоненты

Компонент	Назначение
Веб-сервер nginx	приём запросов пользователей, TLS, защитные заголовки, ограничение размера запроса (2 ГБ), передача запросов серверной части
Служба systemd	запуск серверной части от служебной учётной записи, автозапуск, перезапуск при сбое, изоляция файловой системы
Планировщик cron	ежедневное резервное копирование в 03:30
Средства резервного копирования	согласованный снимок базы данных, архив метаданных, инкрементальные копии изображений, восстановление

4. Модель данных и хранение

Метаданные (проекты, партии, изображения, фигуры разметки, журнал действий) хранятся во встраиваемой СУБД SQLite в режиме журналирования WAL. Изображения, уменьшенные копии и архивы экспорта хранятся файлами в каталоге данных; в базе данных записываются имя файла и его контрольная сумма.

4.1. Основные сущности

Сущность	Содержание	Связи
Проект	идентификатор, наименование, дата и автор создания	содержит классы и партии
Класс проекта	наименование класса и порядок в списке	принадлежит проекту
Партия	наименование, состояние, исполнитель, число отклонённых дубликатов, состояние предварительной разметки	принадлежит проекту, содержит изображения
Изображение	имя файла, размеры, контрольная сумма SHA-256, признак неподтверждённой разметки, автор и время разметки, состояние приёмки и комментарий приёмщика	принадлежит партии, содержит фигуры

Сущность	Содержание	Связи
Фигура разметки	класс, тип (прямоугольная рамка, полигон, повёрнутая рамка), координаты вершин и описанная прямоугольная рамка	принадлежит изображению
Запись журнала	время, пользователь, действие, параметры и предыдущее состояние для отмены	относится к проекту

Удаление проекта или партии каскадно удаляет подчинённые записи. Уникальный индекс по паре «проект, контрольная сумма» исключает повторную загрузку одного и того же изображения в пределах проекта. Счётчики партий, состав набора данных, баланс и отчёты не хранятся, а вычисляются запросом при обращении.

Имя класса записывается в каждую фигуру разметки; список классов проекта определяет только доступные в редакторе варианты. Поэтому переименование и объединение классов выполняются одной транзакцией по всем фигурам проекта.

4.2. Каталог данных

Расположение	Содержимое
`/data/studio.db`	база данных SQLite с метаданными и разметкой
`/data/projects/<проект>/batches/<партия>/images/`	исходные изображения
`/data/projects/<проект>/.thumbs/`	уменьшенные копии изображений
`/data/projects/<проект>/exports/`	сформированные архивы экспорта
`/data/auth/`	учётные записи, сеансы, настройки почты, ключ шифрования; доступ только служебной учётной записи
`/data/global_classes.json`	общий справочник классов
`/data/inference.json`	параметры подключения к сервису предварительной разметки (токен зашифрован)
`/data/backups/`	резервные копии

5. Основные процессы обработки

5.1. Состояния изображения

Каждое изображение проходит последовательность состояний, показанную на рисунке 2. В набор данных, анализ баланса и экспорт попадают только принятые изображения.

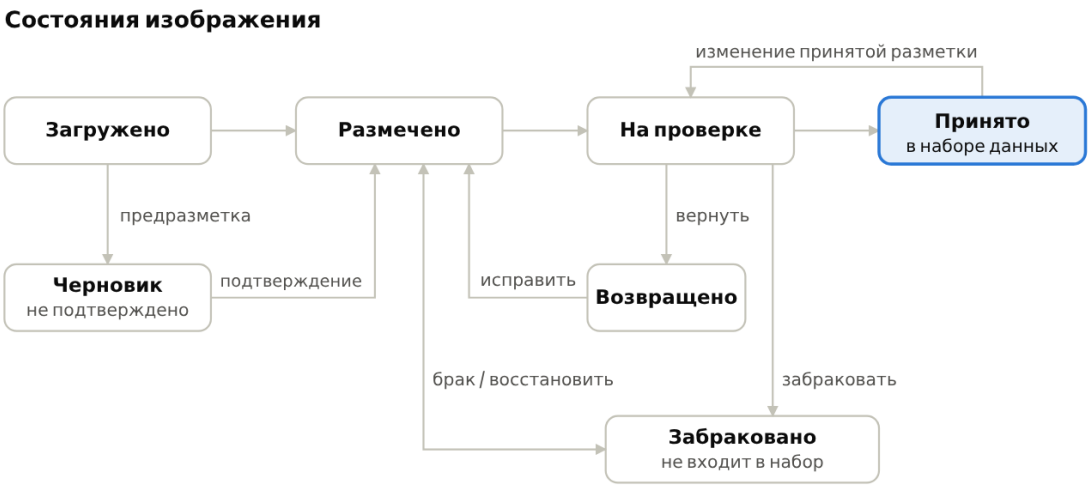

Рисунок 2 — Состояния изображения

Разметка, полученная от сервиса предварительной разметки или импортированная как черновая, считается неподтверждённой до подтверждения разметчиком. Любое изменение принятой разметки переводит изображение на повторную проверку. Отбракованное изображение не удаляется и может быть восстановлено приёмщиком; изменение рамок состояние брака не снимает.

5.2. Состояния партии

Состояние партии не задаётся вручную, а вычисляется по состояниям входящих в неё изображений.

Состояние партии	Условие
Не назначена	исполнитель не назначен
В разметке	исполнитель назначен, есть неразмеченные или возвращённые изображения
На проверке	все изображения размечены, есть ожидающие приёмки
В наборе данных	все изображения приняты или отбракованы

При отправке на проверку части партии размеченные изображения выделяются в отдельную партию, а оставшиеся остаются у разметчика.

5.3. Загрузка и фоновые задачи

При загрузке для каждого файла вычисляется контрольная сумма SHA-256; файл с уже имеющейся в проекте суммой отклоняется как дубликат. Импорт наборов COCO, формирование архивов экспорта и предварительная разметка выполняются фоновыми задачами: пользователь продолжает работу, а ход выполнения отображается в интерфейсе. Прерывание фоновой задачи не приводит к потере уже обработанных данных, а недоступность сервиса предварительной разметки не изменяет состояние партии.

6. Внешние интерфейсы

ПО взаимодействует с пользователями через веб-интерфейс, с другими системами — через файлы обмена и два необязательных исходящих соединения.

Интерфейс	Протокол и формат	Назначение
Веб-интерфейс	HTTP / HTTPS, порты 80 и	работа пользователей в веб-браузере

Интерфейс	Протокол и формат	Назначение
пользователя	443	
Программный интерфейс (HTTP API)	HTTP, данные в формате JSON	взаимодействие клиентской и серверной части: аутентификация, проекты, партии, разметка, приёмка, баланс, экспорт, настройки; доступ только для аутентифицированных пользователей в соответствии с ролью
Проверка работоспособности	HTTP GET `/api/health`, JSON	контроль состояния и версии ПО средствами мониторинга без аутентификации
Импорт данных	изображения JPEG и PNG; разметка в формате COCO JSON	загрузка исходных изображений и ранее размеченных наборов
Экспорт данных	ZIP-архив: COCO JSON; YOLO (текстовые файлы разметки и `data.yaml`, в том числе YOLO-seg); DOTA	передача набора данных в средства обучения нейронных сетей
Почтовый сервер (необязательно)	SMTP с шифрованием (SSL на порту 465 или STARTTLS)	отправка приглашений и ссылок восстановления пароля
Сервис предварительной разметки (необязательно)	HTTP POST, изображение в форме multipart, ответ JSON	получение рамок, найденных моделью распознавания организации-пользователя; адрес, способ авторизации и порог уверенности задаются в настройках

При отсутствии настроек почты и сервиса предварительной разметки ПО не выполняет исходящих сетевых соединений.

7. Безопасность и разграничение доступа

Защита информации обеспечивается на трёх уровнях: операционной системы, веб-сервера и приложения.

Уровень	Меры
Операционная система	служба выполняется от имени непривилегированной учётной записи `datasetstudio` без права входа; изоляция systemd разрешает запись только в каталог данных; SELinux работает в режиме Enforcing
Веб-сервер	серверная часть принимает соединения только по адресу 127.0.0.1; nginx обеспечивает шифрование TLS, передаёт защитные заголовки и не раскрывает версию
Приложение	каждый запрос проверяется модулем разграничения доступа до обработки; пароли хранятся в виде хэшей PBKDF2; сеанс передаётся в cookie с флагами HttpOnly, SameSite и Secure (при работе по HTTPS); пароль почты и токен сервиса предварительной разметки шифруются ключом экземпляра; обращения к файлам вне каталога данных отклоняются

Разграничение доступа основано на трёх иерархических ролях: каждая следующая роль включает полномочия предыдущей.

Роль	Доступные функции
Разметчик	просмотр проектов и партий, разметка, отправка на проверку, отбраковка непригодных изображений
Аналитик	функции разметчика, приёмка, анализ баланса, аналитика, ведение списка классов
Администра	все функции, включая создание и удаление проектов, загрузку, назначение

Роль	Доступные функции
тор	исполнителей, экспорт, управление пользователями и настройками, журнал действий и отчёт по разметчикам

Значимые действия пользователей (загрузка, изменение разметки, приёмка, назначения, изменение классов, экспорт) регистрируются в журнале с указанием исполнителя, времени и предыдущего состояния.

8. Развёртывание, производительность и ограничения

ПО разворачивается на одном сервере под управлением РЕД ОС 8 из RPM-пакета или архива с установочным сценарием. Порядок установки приведён в документе «Инструкция по установке и эксплуатации». Минимальная конфигурация сервера — 2 процессорных ядра и 4 ГБ оперативной памяти; объём диска определяется объёмом хранимых изображений.

Производительность подтверждена на наборе из 8895 изображений общим объёмом 85 ГБ (сервер: 2 процессорных ядра, 8 ГБ оперативной памяти).

Операция	Время
Импорт набора целиком	около 9,5 мин
Расчёт баланса классов	около 4 мс
Загрузка состава набора данных	около 250 мс
Объём базы метаданных	около 6 МБ

Для снижения нагрузки на сеть редактор разметки загружает уменьшенную копию изображения шириной 1600 пикселей; исходное изображение загружается только при увеличении масштаба. Координаты фигур всегда рассчитываются относительно размеров исходного изображения.

Ограничения текущей версии:

- серверная часть работает в одном процессе, состояние фоновых задач хранится в памяти процесса; горизонтальное масштабирование не предусмотрено;
- поддерживаются прямоугольные рамки, полигоны и повёрнутые рамки; разметка ключевых точек и видео не входит в версию 1.0.0;
- аутентификация выполняется собственными средствами ПО, интеграция со службами каталогов (LDAP) не предусмотрена;
- резервные копии создаются на том же носителе; копирование на внешнее хранилище выполняется средствами организации-пользователя.