УТВЕРЖДЕН
38264925.425000.403.П2-ЛУ

Документ представляет собой пояснительную записку к техническому проекту универсальной платформы подготовки и аннотирования данных «МАРКЕР». Платформа автоматизирует весь цикл сбора, разметки и оценки качества мультимедийных данных для задач искусственного интеллекта. В документе описаны архитектура, интерфейсы, автоматизированные рабочие места, меры информационной безопасности и этапы внедрения. Решение подходит для высоконагруженных проектов с распределённой структурой и требованиями к защищённости данных.
Подробнее: https://marker.im/?ysclid=manotpslos762387984

АВТОМАТИЗИРОВАННАЯ СИСТЕМА
ПЛАТФОРМА СБОРА И РАЗМЕТКИ ДАННЫХ

«МАРКЕР»

Пояснительная записка к техническому проекту
38264925.425000.403.П2

Листов 62

Подпись и дата	
Инв. № дубл.	
Взам. инв. №	
Подпись и дата	
Инв. № подл.	

АННОТАЦИЯ

Настоящий документ содержит пояснительную записку к техническому проекту программной платформы сбора и разметки данных «МАРКЕР» (далее - АС).

Платформа «МАРКЕР» предназначена для организации полного цикла подготовки и аннотирования данных различных типов — текстов, изображений, аудио- и видеозаписей — с целью последующего использования этих данных для обучения моделей машинного обучения и систем искусственного интеллекта. Платформа предоставляет пользователям единое интегрированное программное пространство, включающее в себя модули загрузки, визуализации, шаблонной и экспертной разметки, контроля качества, выгрузки, распределения заданий, оценки исполнителей и генерации обучающих выборок.

Разработка и архитектура платформы ориентированы на высокую масштабируемость, отказоустойчивость и возможность интеграции с внешними системами хранения и анализа данных. Все операции по работе с данными осуществляются внутри системы без необходимости использования сторонних инструментов, что позволяет обеспечивать целостность процесса и соответствие требованиям к защите информации.

СОДЕРЖАНИЕ

Аннотация 2

Содержание 3

1 Общие положения 7

1.1 Наименование проектируемой автоматизированной системы (АС) 7

1.2 Наименования документов, их номера и даты утверждения, на основании которых ведётся проектирование АС 7

1.3 Перечень организаций, участвующих в разработке системы, сроки выполнения стадий 7

1.4 Цели, назначение и области использования АС 8

1.5 Подтверждение соответствия проектных решений действующим нормам и правилам безопасности 9

1.6 Сведения об использованных при проектировании нормативно-технических документах 9

1.7 Сведения о НИР, передовом опыте, изобретениях, использованных при разработке проекта 10

1.8 Очередность создания системы и объём каждой очереди 10

2 Описание процесса деятельности 11

2.1 Виды автоматизированной деятельности 11

3 Состав процедур 12

3.1 Участники автоматизированной деятельности 12

3.2 Описание процедур 13

3.2.1 Настройка проекта 13

3.2.2 Подготовка проекта 13

3.2.3 Разметка данных 14

3.2.4 Оценка рейтингов разметчиков 14

4 Основные технические решения 15

4.1 Решения по структуре системы, подсистем, средствам и способам связи для информационного обмена между компонентами системы 15

4.2 Схема архитектуры АС «МАРКЕР» 16

4.3 Требования и ограничения, накладываемые на структуру 17

4.4 Решения по взаимосвязям АС со смежными системами, обеспечению ее совместимости 17

4.5 Решения по режимам функционирования, диагностированию работы системы 18

4.6 Решения по численности, квалификации и функциям персонала АС, режимам его работы, порядку взаимодействия 18

4.7 Численность персонала 19
4.8 Квалификация персонала 19
4.8.1 Режим работы персонала 19
4.9 Сведения об обеспечении заданных в ТЗ потребительских характеристик 19
4.10 Состав функций, комплексов задач, реализуемых системой (подсистемой) 20
4.10.1 Модуль инструментов разметки данных 20
4.10.2 Модуль исходящих вызовов Marker 20
4.10.3 Модуль рейтингов разметчиков 20
4.10.4 Модуль автоматической генерации ханипотов 21
4.10.5 Модуль загрузки-выгрузки данных 21
4.11 3.10 Решения по комплексу технических средств, его размещению на объекте 21
4.12 Решения по составу информации, объему, способам её организации 21
4.13 Решения по составу программных средств, языкам, методам реализации 22
4.14 Описание автоматизированных рабочих мест (АРМ) 22
4.14.1 Состав автоматизированных рабочих мест 23
4.14.2 АРМ разметчика 23
4.14.3 АРМ администратора контента 24
4.14.4 АРМ системного администратора 24
4.14.5 АРМ куратора проекта *(опционально)* 25
4.14.6 АРМ эксперта по качеству *(опционально)* 25
4.15 Таблица сравнения функционала АРМ-ов 25
4.16 Архитектура АС «МАРКЕР» с АРМ-структурой 27
4.16.1 Шаблон (mockup) интерфейса АРМ разметчика 28
4.16.2 Макет: АРМ администратора контента 28
4.16.3 Макет: АРМ эксперта по качеству 29
4.17 Веб-части системы 29
4.17.1 Концепция веб-частей 29
4.17.2 Преимущества применения веб-частей: 29
4.18 Перечень веб-частей в системе «МАРКЕР» 30
4.18.1 Веб-часть: Информация о сеансе 31
4.18.2 Веб-часть: Рабочее задание 32
4.18.3 Веб-часть: Модуль разметки 32
4.18.4 Веб-часть: Редактор звуковых файлов 33
4.18.5 Веб-часть: Редактор видео файлов 33
4.18.6 Веб-часть: Редактор текста 34
4.18.7 Веб-часть: Изменить запись 34

4.18.8 Веб-часть: Последние изменения записей 35
4.18.9 Веб-часть: Список проектов 36
4.18.10 Веб-часть: Список исполнителей 36
4.18.11 Веб-часть: Аналитика и статистика 37
4.18.12 Веб-часть: Подсказки по интерфейсу 37
4.18.13 Веб-часть: Справочник по правилам разметки 38
4.18.14 Веб-часть: Панель администратора 38
4.18.15 Веб-часть: Глобальное меню 39
4.18.16 Веб-часть: Боковая панель 39
4.18.17 Веб-часть: Панель уведомлений 40
4.18.18 Веб-часть: Авторизация пользователя 41
4.18.19 Веб-часть: Инструкция 41
4.18.20 Веб-часть: Пагинация 42
4.18.21 Веб-часть: Справка о прогрессе 42
4.18.22 Веб-часть: Последние комментарии 43
4.18.23 Веб-часть: Отображение записей 43
4.18.24 Веб-часть: Распределение неразмеченных записей 44
4.18.25 Веб-часть: Журнал событий 44
4.18.26 Веб-часть: Календарь 45
4.18.27 Веб-часть: Облако тегов 46
4.18.28 Веб-часть: Меню "От сборщика" 46
4.18.29 Веб-часть: Субтитры / Аудиокниги 47
4.18.30 Веб-часть: Разметка личных данных 47
4.18.31 Веб-часть: Личный кабинет 48
4.18.32 Веб-часть: Управление командой проекта 48
4.18.33 Веб-часть: Просмотр задания и выполнение разметки 49
4.18.34 Веб-часть: Управление проектом 50
4.18.35 Веб-часть: Выгрузка и фильтрация данных 50
4.18.36 Веб-часть: Анализ качества (экспертная проверка) 51
4.18.37 Веб-часть: Управление пользователями и ролями 52
4.19 Описание структуры хранения и обмена данными 52
4.19.1 Директорная структура и сущности базы данных 52
4.19.2 ER-модель: связи между сущностями 53
4.19.3 Характеристики модели хранения 54
4.1 ER-диаграмма: связи между сущностями 55
4.2 Архитектурная схема хранения данных 56

4.3 Интерфейс АС «МАРКЕР» 56
5 Мероприятия по подготовке объекта автоматизации к вводу системы в действие 58
5.1 Мероприятия по приведению информации к виду, пригодному для обработки на ЭВМ 58
5.2 Мероприятия по обучению и проверке квалификации персонала 58
5.3 Мероприятия по созданию необходимых подразделений и рабочих мест 59
5.4 Мероприятия по изменению объекта автоматизации 59
5.5 Другие мероприятия, исходящие из специфических особенностей создаваемых АС 60
6 Перечень терминов и сокращений 61

1 ОБЩИЕ ПОЛОЖЕНИЯ

1.1 Наименование проектируемой автоматизированной системы (АС)

Полное наименование: Платформа сбора и разметки данных «МАРКЕР».

Сокращённое наименование: АС «МАРКЕР».

Назначение: автоматизация процессов сбора, аннотирования (разметки), обработки и подготовки обучающих выборок, предназначенных для обучения моделей машинного обучения и искусственного интеллекта.

1.2 Наименования документов, их номера и даты утверждения, на основании которых ведётся проектирование АС

Проектирование, реализация и внедрение автоматизированной системы «МАРКЕР» осуществляется на основании следующих документов:

Соглашение о предоставлении гранта № XXXX-XXX-XX;

Техническое задание на создание программной платформы «МАРКЕР» (№ [указать], от [дата утверждения]);

Методические рекомендации уполномоченный орган по проектированию государственных информационных систем;

1. ГОСТ 34.201–89, ГОСТ 34.602–89, ГОСТ 34.603–92, ГОСТ 19.201–78 и др. нормативные документы, регламентирующие структуру и содержание проектной документации;
2. Договор на выполнение НИОКР по теме создания автоматизированной системы подготовки обучающих данных (шифр: MRKR).

1.3 Перечень организаций, участвующих в разработке системы, сроки выполнения стадий

Разработчик: технологический подрядчик — полное выполнение НИОКР, проектирование архитектуры, реализация программных модулей, опытная эксплуатация.

Координатор проекта: фонд поддержки цифровых технологий — оператор государственной поддержки.

Проект реализуется в два этапа:

Этап 1 (2022 г.):

1. Разработка и внедрение инструментов аннотирования;
2. Интеграция с внешними хранилищами (Hadoop);
3. Создание механизмов контроля качества и активного обучения;
4. Реализация интерфейсов и шаблонов разметки;
5. Завершение: 30.09.2022.

Этап 2 (2022–2023 гг.):

1. Проведение испытаний;
2. Расширение интерфейсов;
3. Разработка подсистем рейтингов, выплат и журналирования операций;
4. Тонкая настройка и интерфейсы администрирования;
5. Завершение: 30.06.2023.

Подробный перечень мероприятий — см. таблицу 1 настоящей пояснительной записки.

1.4 Цели, назначение и области использования АС

Цель разработки: создание универсальной программной платформы, обеспечивающей полный цикл подготовки обучающих датасетов для использования в системах искусственного интеллекта, включая аннотирование мультимедийных данных, контроль качества и управление проектами.

Назначение:

1. Поддержка процессов аннотирования текстов, аудио, видео, изображений;
2. Повышение точности, скорости и согласованности разметки;
3. Поддержка коллективной работы и интеграция с ИИ-моделями.

Области применения:

1. Государственные и муниципальные учреждения (Минобороны, Минздрав, Минобрнауки, МВД и др.);
2. Коммерческие компании (банки, логистика, ретейл);
3. Образовательные и научные учреждения;
4. Медицинские организации (разметка DICOM и других данных);
5. СМИ, исследовательские лаборатории, частные лица.

АС «МАРКЕР» ориентирована на использование в высоконагруженных, распределённых проектах с возможностью гибкой настройки и масштабирования.

1.5 Подтверждение соответствия проектных решений действующим нормам и правилам безопасности

Проектирование АС «МАРКЕР» выполнено с соблюдением требований следующих нормативно-технических документов в области безопасности:

1. ГОСТ 12.2.007.0–75 — общие требования безопасности к электротехническим изделиям;
2. ГОСТ Р 12.1.019–2017 — электробезопасность. Общие требования;
3. ГОСТ 12.1.004–91 — пожарная безопасность;
4. Федеральный закон № 152-ФЗ «О персональных данных»;
5. ГОСТ Р ИСО/МЭК 27001–2012 — управление информационной безопасностью;
6. ГОСТ Р 56939–2016 — требования к качеству программного обеспечения.

Дополнительные меры обеспечения безопасности (физической, пожарной, промышленной и ИБ) принимаются при внедрении платформы в инфраструктуре заказчика в соответствии с его внутренними регламентами и действующим законодательством РФ.

1.6 Сведения об использованных при проектировании нормативно-технических документах

При разработке и проектировании использованы следующие нормативные документы:

1. **ГОСТ 34.201–89** — Виды и комплектность документов;
2. **ГОСТ 34.601–90** — Стадии создания АС;
3. **ГОСТ 34.602–89** — Техническое задание на АС;
1. **ГОСТ 34.603–92** — Виды испытаний АС;
2. **ГОСТ 19.201–78, ГОСТ 19.105, ГОСТ 19.106** — Требования к программным документам;
3. **ГОСТ 2.105–95** — Общие требования к текстовым документам;
4. **РД 50-34.698–90** — Стандарты на содержание проектной документации;
5. **ГОСТ Р ИСО/МЭК 12207–2010** — процессы жизненного цикла ПО.

А также отраслевые рекомендации уполномоченный орган и внутренние регламенты ООО «[подрядчик]».

1.7 Сведения о НИР, передовом опыте, изобретениях, использованных при разработке проекта

Проект реализуется на основе результатов научно-исследовательских и опытно-конструкторских работ (НИОКР), проведённых в 2021–2022 гг., с использованием:

1. методов активного обучения и гибридной аннотации;
2. патентованных решений по визуализации разметки и контролю качества;
3. опыта отечественных и зарубежных платформ (Label Studio, Toloka, Amazon MTurk);
4. моделей машинного обучения на базе PyTorch и HuggingFace;
5. лучших практик explainable AI (XAI).

Использованы разработки, полученные при реализации проекта по гранту, в том числе результаты собственных НИР ООО «[подрядчик]».

1.8 Очередность создания системы и объём каждой очереди

Проект разбит на очереди, каждая из которых представляет собой завершённый логический этап с оформлением документации и промежуточных результатов.

Очередь 1 (2022):

1. Разработка ключевых модулей;
2. Создание базы ханипотов и алгоритмов консенсуса;
3. Реализация инструментов визуальной аннотации;
4. Интеграция с системами хранения (Hadoop).

Очередь 2 (2023):

5. Расширение интерфейсов;
6. Валидация модулей;
7. Разработка системы начислений и авторизаций;
8. Обновление шаблонов, функционала и инструкций;
9. Подготовка к опытной эксплуатации.

2 ОПИСАНИЕ ПРОЦЕССА ДЕЯТЕЛЬНОСТИ

2.1 Виды автоматизированной деятельности

Автоматизированная система (АС) «МАРКЕР» обеспечивает выполнение комплекса автоматизированных процессов, связанных с загрузкой, разметкой, проверкой, выгрузкой и управлением данными, а также с администрированием системы и сопровождением пользователей.

К видам автоматизированной деятельности, реализуемым в рамках функционирования АС, относятся:

1. **Управление данными в системе** — автоматизация операций, связанных с загрузкой, обработкой (разметкой) и выгрузкой мультимедийных данных (тексты, аудио, изображения, видео) в рамках заданий на разметку;
2. **Администрирование процессов разметки** — включающее создание проектов, настройку шаблонов, загрузку и распределение заданий, оценку качества и управление результатами;
3. **Управление доступом и пользователями** — предоставление ролей и прав, контроль активности и рейтингов, администрирование учетных записей;
4. **Техническое сопровождение и конфигурирование системы** — поддержка инфраструктуры, обновлений и мониторинга работоспособности компонентов.

3 СОСТАВ ПРОЦЕДУР

Таблица 1 – Состав процедур

Вид деятельности	Процедуры	Участники
Управление данными	- Загрузка файлов (текстовых, аудио, фото, видео)	Разметчик
	- Разметка данных через пользовательский интерфейс	
	- Сохранение результатов разметки	
Администрирование	- Создание и настройка проектов и шаблонов	Администратор контента
	- Загрузка и предварительная подготовка данных (в том числе через панель администратора)	
	- Управление правами и ролями пользователей	
	- Выгрузка размеченных данных (внутренняя БД, Hadoop, локальные хранилища)	
	- Управление ханипотами (создание, назначение, оценка качества)	
Техническая поддержка	- Развёртывание и настройка серверной инфраструктуры, резервное копирование, мониторинг	Системный администратор
	- Управление компонентами системы, логами и обновлениями	

3.1 Участники автоматизированной деятельности

Участниками автоматизированной деятельности в рамках АС «МАРКЕР» являются следующие авторизованные пользователи:

1. **Разметчик** — исполнитель, выполняющий аннотирование мультимедийных данных на основе предоставленных шаблонов и заданий;
2. **Администратор контента** — пользователь, осуществляющий настройку проектов, загрузку и выгрузку данных, управление шаблонами и пользователями, контроль качества выполнения работ;
3. **Системный администратор** — технический специалист, обеспечивающий настройку серверной инфраструктуры, развёртывание компонентов системы, мониторинг и обслуживание.

Дополнительно в АС предусмотрены роли:

1. **Эксперт по контролю качества** — проверяет результаты разметки, проводит верификацию ханипотов и участвует в выставлении рейтингов;
2. **Куратор проекта** — управляет группами разметчиков, отслеживает сроки выполнения и статистику проектов;

3. **Аудитор** — пользователь с доступом к журналам аудита и безопасности.

Платформа поддерживает гибкую ролевую модель доступа. Количество ролей в системе не ограничено — новые роли могут быть добавлены в любой момент в зависимости от требований конкретного проекта или сценария использования. Полномочия каждой роли задаются на этапе её создания, включая доступ к функциональным модулям, типам данных, операциям редактирования и инструментам контроля качества. Такая настройка позволяет точно регулировать доступ пользователей к различным элементам платформы и реализовывать разграничение прав в соответствии с политиками информационной безопасности и организационными регламентами.

3.2 Описание процедур

3.2.1 Настройка проекта

Процедура настройки проекта включает следующие шаги:

3.2.2 Подготовка проекта

1. **Создание проекта**: Администратор контента задаёт параметры проекта, включая наименование, цели, типы данных, шаблоны разметки, роли пользователей и расписание работ.
2. **Загрузка исходных данных**: Через интерфейс администратора контента загружаются исходные данные в допустимых форматах (TXT, PNG, MP3, MP4 и др.), при необходимости проводится предварительная фильтрация или сегментация.
3. **Создание и назначение пользователей**: Администратор формирует список исполнителей (разметчиков), назначает им роли и подключает к проекту.
4. **Настройка шаблонов разметки**: Выбираются или настраиваются шаблоны, включая типы меток, правила выделения сущностей, подсказки и параметры валидации.
5. **Назначение ханипотов и контрольных записей**: В проект включаются заранее размеченные контрольные записи для оценки качества работы разметчиков.
6. **Запуск проекта**: Данные распределяются по пользователям, система запускает цикл обработки.

3.2.3 Разметка данных

1. **Выбор или загрузка шаблона разметки**: Разметчик выбирает шаблон (язык, тип данных, формат), соответствующий заданию.
2. **Загрузка данных**: Разметчик получает назначенные файлы (либо загружает самостоятельно, если предусмотрено политикой проекта).
3. **Разметка объектов**: помощью веб-интерфейса пользователь наносит метки, описания, выделяет сущности, задаёт категории и сохраняет результат.
4. **Передача данных на проверку**: Размеченные данные передаются системе, где возможна валидация (автоматическая или экспертная).
5. **Выгрузка результатов**: Администратор контента выгружает размеченные датасеты (с возможностью фильтрации по рейтингу исполнителей) в Hadoop или на локальные хранилища.

3.2.4 Оценка рейтингов разметчиков

1. **Формирование и загрузка ханипотов**: Администратор контента загружает в проект заранее размеченные контрольные записи (ханипоты).
2. **Разметка ханипотов исполнителями**: Разметчик обрабатывает ханипот наравне с обычными данными, не зная о его контрольном статусе.
3. **Сравнение разметки с эталоном**: АС автоматически оценивает точность аннотаций, выставляет рейтинг на основе совпадений с эталонными ответами.
4. **Корректировка рейтинга**: Администратор контента имеет возможность вручную изменить рейтинг разметчика с учётом экспертной оценки.
5. **Использование рейтинга**: Система использует рейтинги для допуска исполнителей к новым проектам, распределения задач и начисления вознаграждения.

4 ОСНОВНЫЕ ТЕХНИЧЕСКИЕ РЕШЕНИЯ

4.1 Решения по структуре системы, подсистем, средствам и способам связи для информационного обмена между компонентами системы

Автоматизированная система (АС) «МАРКЕР» реализована в виде модульной распределённой архитектуры, построенной по принципам микросервисного подхода и ориентированной на использование web-интерфейсов. Обмен данными между компонентами осуществляется через REST и GraphQL API по протоколу HTTPS с применением механизма авторизации на базе Active Directory.

Основные подсистемы:

1. **Frontend** — пользовательский интерфейс, реализованный на ReactJS + TypeScript;
2. **Backend** — логика приложения, построенная на Python/Django;
3. **СУБД** — PostgreSQL 11+;
4. **Хранилище данных** — интеграция с Hadoop через API;
5. **Модуль аналитики и рейтингов** — формирование оценок по результатам разметки;
6. **Механизмы контроля качества** — система ханипотов и экспертной верификации;
7. **Модуль администрирования** — управление пользователями, проектами и заданиями;
8. **Модули аудита и диагностики** — отслеживание операций и выявление сбоев.

Взаимодействие компонентов обеспечивается:

1. через протокол TCP/IP;
2. режиме "клиент–сервер" с возможностью горизонтального масштабирования;
3. применением механизмов SLA и логирования отказов.

Более подробная детализация модулей — см. п. 4.10 *и* 4.11 *настоящей пояснительной записки.* Визуальная схема архитектуры представлена на Рисунке 1.

4.2 Схема архитектуры АС «МАРКЕР»

На Рисунке 1 отражены основные компоненты клиентской и серверной части АС, а также взаимодействие с внешними системами — Active Directory и Hadoop. Модули внутри серверной части реализованы в рамках микросервисной архитектуры.

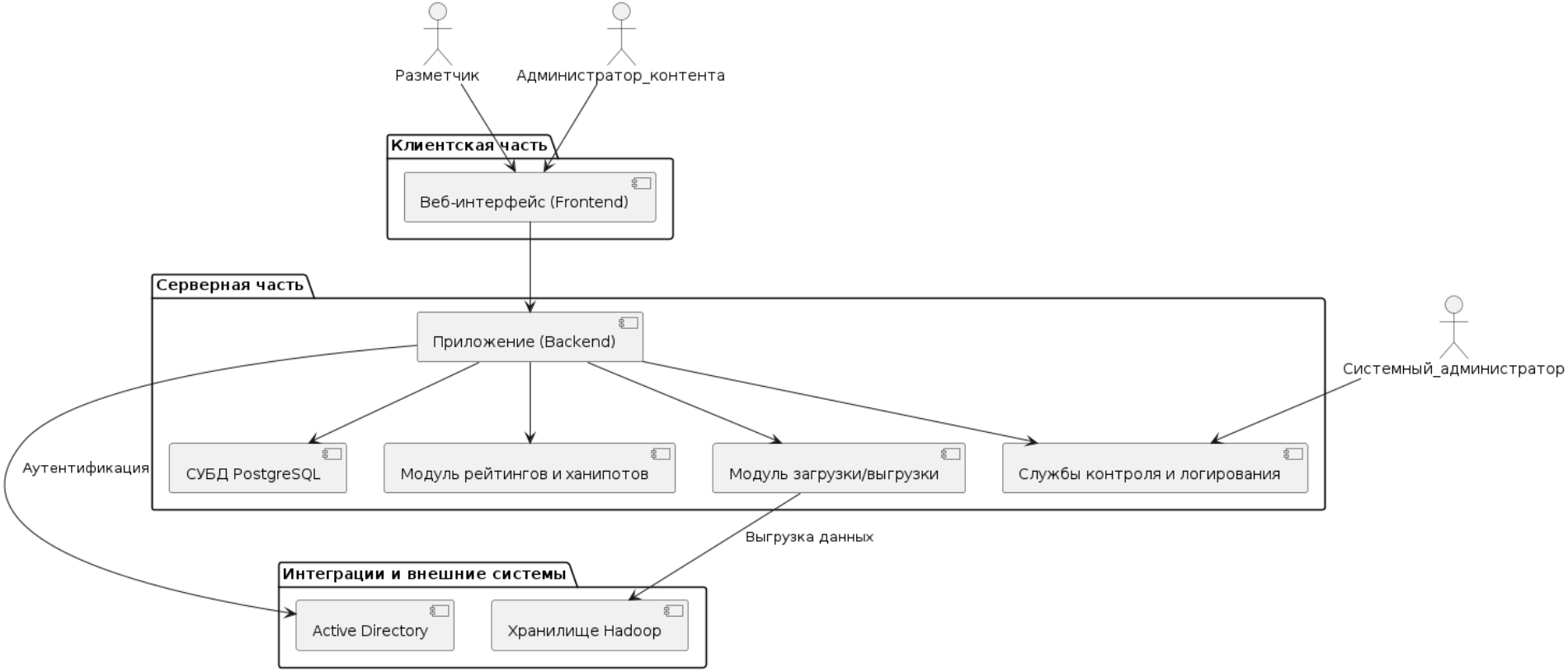

Рисунок 1 – Схема архитектуры и взаимодействия компонентов АС «МАРКЕР»

4.3 Требования и ограничения, накладываемые на структуру

Структура и техническая архитектура системы формируются с учетом следующих требований:

1. Интуитивно понятный пользовательский интерфейс;
2. Поддержка одновременной работы большого количества пользователей;
3. Отсутствие ограничений на количество параллельно выполняемых проектов;
4. Высокая отказоустойчивость и сохранность пользовательских данных;
5. Масштабируемость (как вертикальная, так и горизонтальная);
6. Поддержка резервного копирования и восстановления;
7. Высокая производительность при операциях чтения, записи и поиска;
8. Централизованная аутентификация и разграничение прав доступа;
9. Совместимость с AD, поддержка LDAP-запросов.

4.4 Решения по взаимосвязям АС со смежными системами, обеспечению ее совместимости

АС «МАРКЕР» обеспечивает взаимодействие со следующими внешними информационными системами и компонентами:

1. **Active Directory** — для аутентификации пользователей и управления доступом;
2. **Система хранения Hadoop** — для выгрузки и долговременного хранения размеченных данных;
3. **Внутренние системы заказчика** (по мере внедрения):
   a. CRM-системы;
   b. аналитические хранилища;
   c. ERP и кадровые модули (при необходимости синхронизации данных).
4. **Интеграция реализуется через:**
   a. REST/GraphQL API;
   b. механизм сопоставления ролей и групп из AD;
   c. сериализацию и передачу данных в формате JSON, CSV, MNF, Parquet.

4.5 Решения по режимам функционирования, диагностированию работы системы

АС «МАРКЕР» поддерживает круглосуточный режим функционирования 24/7.

Основные режимы:

1. **Штатный режим** — стандартная эксплуатация системы с доступом к функционалу пользователями всех ролей;
2. **Сервисный режим** — предназначен для выполнения операций технического обслуживания:
   a. обновление компонентов ПО;
   b. восстановление данных из резервных копий;
   c. изменение конфигурации;
   d. анализ журналов и устранение неполадок.
3. **Диагностика:**
   a. Автоматический контроль функционирования с логированием;
   b. Выявление ошибок и нарушений целостности данных;
   c. Предиктивный анализ отказов на основе системных событий;
   d. Поддержка журналов аудита и диагностики (логирование действий пользователей, запросов, ошибок, отказов компонентов).

4.6 Решения по численности, квалификации и функциям персонала АС, режимам его работы, порядку взаимодействия

АС «МАРКЕР» предполагает ролевую модель доступа с разграничением прав пользователей в зависимости от их задач и квалификации.

Основные роли:

1. **Системный администратор** — отвечает за инфраструктуру, серверные компоненты, резервное копирование, обновления и устранение неисправностей;
2. **Администратор контента** — управляет пользователями, проектами, шаблонами разметки, выгрузкой данных и рейтингами;
3. **Разметчик** — осуществляет аннотирование данных согласно заданным правилам.

Возможность выполнения действий в АС определяется ролью, назначенной пользователю. Все взаимодействия зафиксированы в журналах аудита.

4.7 Численность персонала

1. **Системный администратор** — 1 штатная единица;
2. **Администратор контента** — 1 штатная единица;
3. **Разметчики** — количество не ограничено, определяется масштабом проектов.

4.8 Квалификация персонала

1. **Разметчик:**
   a. Уверенное владение ПК;
   b. Знание инструкций по аннотированию данных;
   c. Базовое понимание форматов мультимедийных файлов.
2. **Администратор контента:**
   a. Опыт работы с веб-интерфейсами;
   b. Навыки организации проектов;
   c. Знание логики шаблонов разметки и принципов контроля качества.
3. **Системный администратор:**
   a. Углубленные знания Linux (Ubuntu), сетей, протоколов TCP/IP;
   b. Опыт с PostgreSQL, nginx, Docker;
   c. Навыки работы с мониторинговыми системами и инструментами CI/CD.

4.8.1 Режим работы персонала

Режим работы определяется внутренними регламентами Заказчика. Возможен посменный график, гибкое планирование и удалённый доступ через защищённый веб-интерфейс.

4.9 Сведения об обеспечении заданных в ТЗ потребительских характеристик

1. Обработка большого количества проектов одновременно — **без ограничений**;

2. Среднее время восстановления системы после сбоя — **не более одного рабочего дня**;
3. Поддержка непрерывного функционирования — **круглосуточно**;
4. Выгрузка данных в **Hadoop** или на локальные хранилища — по требованию;
5. Рейтинговая система повышает качество выполнения заданий.

4.10 Состав функций, комплексов задач, реализуемых системой (подсистемой)

АС реализует следующий функционал:

1. Инструменты аннотирования всех типов данных;
2. Модуль исходящих вызовов (интеграция с внешними хранилищами);
3. Система рейтингов исполнителей;
4. Генерация и контроль ханипотов;
5. Модуль импорта/экспорта данных и метаданных.

4.10.1 Модуль инструментов разметки данных

Модуль реализует:

1. Поддержку всех типов данных: текст, изображения, аудио, видео;
2. Конфигурирование шаблонов;
3. Возможность масштабирования и работы с несколькими исполнителями;
4. Контроль качества разметки, ранжирование исполнителей;
5. Интеграцию с системой ханипотов.

4.10.2 Модуль исходящих вызовов Marker

Обеспечивает:

1. Выгрузку разметки через API в хранилище Hadoop;
2. Интеграцию с внутренними системами заказчика при необходимости.

4.10.3 Модуль рейтингов разметчиков

1. Автоматическое присвоение рейтинга по качеству ханипотов;

2. Корректировка вручную;
3. Предпочтение в спорах отдаётся пользователю с более высоким рейтингом;
4. Рейтинг влияет на допуск к проектам.

4.10.4 Модуль автоматической генерации ханипотов

1. Поддержка всех типов данных;
2. Автоматическая вставка ханипотов в задания;
3. Отчёты по статистике прохождения ханипотов.

4.10.5 Модуль загрузки-выгрузки данных

1. Импорт и экспорт мультимедийных файлов и метаданных;
2. Поддержка форматов: `.txt`, `.json`, `.csv`, `.xlsx`, `.jpg`, `.wav`, `.mp4`, `.parquet`;
3. Поддержка пользовательских шаблонов.

4.11 3.10 Решения по комплексу технических средств, его размещению на объекте

АС развёртывается на серверной инфраструктуре заказчика:

1. Сервер приложений и СУБД — размещаются в ЦОД;
2. Клиентская часть — доступ через браузер;
3. **Сеть** — протокол HTTPS, VPN-доступ, TCP/IP;
4. Резервное копирование — по графику;
5. **Мониторинг** — средствами системного администратора.

4.12 Решения по составу информации, объему, способам её организации

1. Входные данные: `.txt`, `.mp3`, `.wav`, `.flac`, `.jpg`, `.png`, `.json`, `.xlsx`, `.parquet`;
2. Выходные данные: `.mnf`, `.json`, `.csv`;
3. Информация структурирована в рамках **датасетов**, **проектов**, **заданий**, **пользователей**;
4. Поддерживается автоматическая генерация служебных сообщений и логов.

4.13 Решения по составу программных средств, языкам, методам реализации

1. **Серверная часть:**
   a. ОС: Linux Ubuntu Server 20.04+;
   b. СУБД: PostgreSQL 11+;
   c. Web-сервер: nginx + gunicorn;
   d. Язык программирования: Python 3.6+;
   e. Фреймворк: Django (DRF, GraphQL).
2. **Клиентская часть:**
   a. ReactJS + TypeScript;
   b. Redux Store для управления состоянием;
   c. GraphQL / REST API для запросов;
   d. Поддержка всех современных браузеров.
3. **Интеграция:**
   a. LDAP/AD, Keycloak;
   b. Система журналирования и аудита действий пользователей;
   c. Мониторинг: Prometheus + Grafana (опционально).

4.14 Описание автоматизированных рабочих мест (АРМ)

В АС «МАРКЕР» применяется концепция автоматизированных рабочих мест (АРМ) — логически обособленных пользовательских интерфейсов, соответствующих ролям пользователей в системе. Под АРМ в настоящем документе понимается совокупность программных, технических и организационных средств, обеспечивающих выполнение пользователем специфического набора функций в рамках своих полномочий.

Внедрение модели АРМ позволяет:

1. формализовать и стандартизировать интерфейсы для разных типов пользователей;
2. разграничить доступ к функциональности в соответствии с политиками безопасности;
3. повысить удобство работы и обучаемость системы;
4. упростить сопровождение и масштабирование платформы.

Каждый АРМ реализуется как веб-интерфейс, доступный пользователю после прохождения авторизации, и содержит набор экранных форм и действий, определяемых ролью пользователя в системе.

4.14.1 Состав автоматизированных рабочих мест

Таблица 2 – Состав автоматизированных рабочих мест

АРМ	Роль пользователя	Назначение
АРМ разметчика	Разметчик	Выполнение заданий по аннотированию данных
АРМ администратора контента	Администратор контента	Управление проектами, заданиями, пользователями, шаблонами
АРМ системного администратора	Системный администратор	Обслуживание серверной части, логирование, резервное копирование
АРМ куратора проектов (опционально)	Куратор (менеджер)	Мониторинг выполнения проектов, аналитика, метрики
АРМ эксперта по качеству (опционально)	Эксперт	Анализ ханипотов, ручная проверка, выставление рейтингов

4.14.2 АРМ разметчика

Назначение: предоставление исполнителю функционала для выполнения заданий по разметке текстов, аудио, изображений и видео с использованием шаблонов, аннотаций и подсказок.

Функции:

1. Просмотр назначенных заданий;
2. Загрузка файлов на разметку (если предусмотрено);
3. Выбор шаблона;
4. Аннотирование объектов с помощью удобного визуального интерфейса;
5. Сохранение и отправка результатов в систему;
6. Просмотр собственной статистики (при наличии).

Разработка: веб-интерфейс на React, с интеграцией к backend через GraphQL. Предусмотрена адаптивная верстка, поддержка горячих клавиш и шаблонов для разных типов данных.

4.14.3 АРМ администратора контента

Назначение: реализация функций управления проектами, заданиями, пользователями, шаблонами, контролем качества и выгрузкой данных.

Функции:

1. Создание и конфигурация проектов;
2. Назначение пользователей и шаблонов;
3. Загрузка исходных данных;
4. Настройка ханипотов;
5. Просмотр и выгрузка результатов (в локальное хранилище или Hadoop);
6. Корректировка рейтингов;
7. Выдача технической информации, формирование отчётов.

Разработка: административная панель на React + Django Admin / кастомизированные компоненты. Разделённый интерфейс по категориям: «Проекты», «Шаблоны», «Пользователи», «Статистика», «Выгрузка».

4.14.4 АРМ системного администратора

Назначение: обслуживание серверной и сетевой инфраструктуры системы, обеспечение непрерывности и безопасности функционирования.

Функции:

1. Мониторинг состояния компонентов (сервер приложений, СУБД, хранилище);
2. Управление логами и аудитом;
3. Обновление версий компонентов;
4. Настройка резервного копирования;
5. Проверка журналов на предмет сбоев;
6. Подключение внешних сервисов (LDAP, хранилища).

Разработка: доступ осуществляется через защищённый канал SSH и/или специализированную веб-консоль администрирования (например, Grafana, Netdata, pgAdmin). Используются инструменты Prometheus, Fail2Ban, rsync, cron, Docker Compose.

4.14.5 АРМ куратора проекта *(опционально)*

Назначение: наблюдение за ходом выполнения проектов, формирование отчётов по качеству, объёмам и активности пользователей.

Функции:

1. Просмотр хода выполнения;
2. Контроль сроков и метрик;
3. Фильтрация заданий по статусам;
4. Статистические отчёты;
5. Интеграция с BI-системами.

Разработка: модули визуализации (на базе Recharts или Chart.js), агрегирующие данные из backend.

4.14.6 АРМ эксперта по качеству *(опционально)*

Назначение: обеспечение независимой верификации результатов разметки, обработка ханипотов, формирование заключений.

Функции:

1. Ручная проверка разметки;
2. Ввод экспертной оценки;
3. Настройка и выбор ханипотов;
4. Создание отчётов по несоответствиям;
5. Управление шкалой рейтинга.

Разработка: интерфейс, аналогичный АРМу разметчика, с расширенным функционалом сравнения и комментирования.

4.15 Таблица сравнения функционала АРМ-ов

Таблица 3 – Сравнения функционала АРМ-ов

Функциональность	АРМ разметчика	АРМ администратора контента	АРМ системного администратора	АРМ куратора	АРМ эксперта по качеству
Просмотр заданий	✓	✓	✕	✓	✓
Выполнение разметки	✓	✕	✕	✕	✓

Загрузка данных	⚠ (если разрешено)	✓✓	✕	✕	✕
Настройка шаблонов	✕	✓✓	✕	✕	✕
Управление пользователями	✕	✓✓	✕	✕	✕
Выгрузка результатов	✕	✓✓	✕	✕	✕
Работа с ханипотами	✕	✓✓	✕	✕	✓✓
Настройка системы, резервное копирование	✕	✕	✓✓	✕	✕
Просмотр логов и диагностика	✕	✕	✓✓	⚠ (аналитика)	✕
Статистика и метрики	⚠ (личная)	✓✓	✓✓	✓✓	⚠
Ручная проверка и оценка качества	✕	⚠ (рейтинг)	✕	✕	✓✓
Доступ к BI-интерфейсу	✕	✕	⚠	✓✓	✕

Обозначения в Таблице 3**:**

✓✓ — функциональность доступна

⚠ — доступна частично / опционально

✕ — не предусмотрено

4.16 Архитектура АС «МАРКЕР» с АРМ-структурой

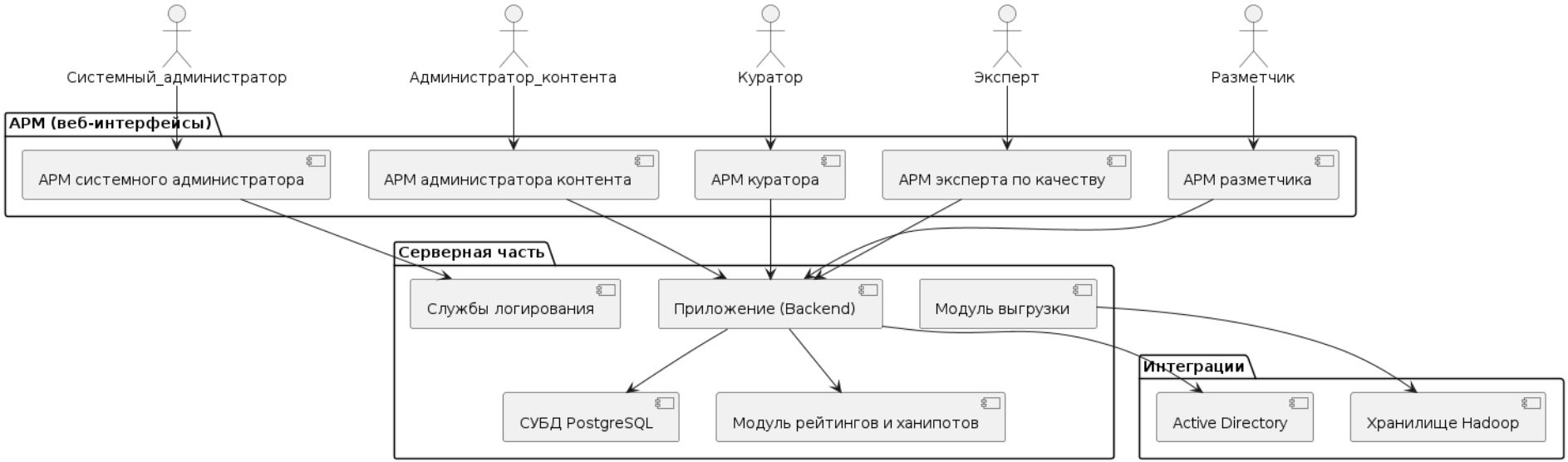

Рисунок 2 – Архитектура АС «МАРКЕР» с АРМ-структурой

4.16.1 Шаблон (mockup) интерфейса АРМ разметчика

```
  Платформа МАРКЕР – АРМ разметчика

[Выбрать проект ▾]
Шаблон: [ ▭Название шаблона ]

Задание: 231/500
Тип данных: Текст

  ▒▒  ОБЛАСТЬ РАЗМЕТКИ  ▒▒

  Текст: "[пример текста]"
  [ ] Географическая сущность
  [ ] Организация

[ Сохранить ] [ Отправить ] [ Пропустить ]
```

4.16.2 Макет: АРМ администратора контента

Ниже представлен макет интерфейса (mockup) для АРМ администратора контента

```
       Платформа МАРКЕР – АРМ администратора контента

[ Проекты ] [ Пользователи ] [ Шаблоны ] [ Ханипоты ] [ Выгрузка ]

>> Проект: [▭Название проекта]      Статус: [Активен ▾]

Задания: [ 500 ]    Разметчики: [ 24 ]   Ханипоты: [Включены]
Дата создания: 01.03.2025       Ответственный: Пользователь А

◉ Панель управления проектом
 - Загрузить датасет    [⍰]
 - Назначить разметчиков [+Добавить]
 - Настроить шаблон      [Редактировать шаблон ▸]
 - Включить ханипоты     [☑]
 - Настроить рейтинги    [⚙ Настройки рейтинга]

◉ Выгрузка данных
 - Формат: [ JSON ▾ ]  Назначение: [ Hadoop ▾ ]  [Выгрузить]
```

4.16.3 Макет: АРМ эксперта по качеству

Ниже представлен макет интерфейса (mockup) для АРМ эксперта по качеству

```
┌──────────────────────────────────────────────────────────────┐
│        Платформа МАРКЕР – АРМ эксперта по качеству           │
├──────────────────────────────────────────────────────────────┤
│ [ Контроль ханипотов ] [ Проверка выборок ] [ Отчёты ]       │
├──────────────────────────────────────────────────────────────┤
│ >> Проект: [▭Название проекта]                               │
│ >> Разметчик: [▭Пользователь А]   Рейтинг: ★★★★☆             │
├──────────────────────────────────────────────────────────────┤
│ ◉ Задание:                                                   │
│  Текст: "[пример текста]"                                    │
│  Разметка исполнителя: [Организация]                         │
│  Эталонная разметка:  [Географическая сущность]              │
│  [⚠ Несоответствие – внести комментарий]                     │
│  [Комментарий: ________________________________ ]            │
├──────────────────────────────────────────────────────────────┤
│ ◉ Действия:  [ Подтвердить оценку ]  [ Корректировать вручную ]
│ ◉ Ханипот: [☑ Включить в выборку для оценки ]                │
└──────────────────────────────────────────────────────────────┘
```

4.17 Веб-части системы

4.17.1 Концепция веб-частей

В рамках архитектуры АС «МАРКЕР» применяется модульный подход к организации клиентского интерфейса, основанный на использовании веб-частей (web components). Каждая веб-часть представляет собой независимый интерфейсный блок, реализующий определённую функциональность в составе АРМ. Это позволяет переиспользовать компоненты в разных ролях, обеспечивать гибкость UI, ускорять доработки и упростить тестирование.

4.17.2 Преимущества применения веб-частей:

1. **Повторное использование компонентов**: одна и та же часть (например, форма разметки текста) может быть встроена как в АРМ разметчика, так и в АРМ эксперта;
2. **Изолированность и тестируемость**: каждая часть реализуется и проверяется как независимый компонент;

3. **Фокус на задаче пользователя**: каждая часть решает конкретную бизнес-функцию — загрузка, аннотирование, оценка;

4. **Гибкость обновления**: легко адаптируется при изменении бизнес-логики или внешнего вида;

5. **Ускоренная разработка**: модули разрабатываются параллельно и независимо.

4.18 Перечень веб-частей в системе «МАРКЕР»

Таблица 4 – Список веб-частей в системе «МАРКЕР»

Название веб-части	Назначение веб-части	Где используется веб-часть
Информация о сеансе	Отображение текущего статуса пользователя и проекта	Все АРМ
Рабочее задание	Отображение текущей задачи и метаданных	АРМ разметчика
Модуль разметки	Интерфейс аннотирования данных	АРМ разметчика
Редактор звуковых файлов	Работа с аудиозаписями	АРМ разметчика, АРМ эксперта
Редактор видео файлов	Просмотр и разметка видеоконтента	АРМ разметчика
Редактор текста	Ввод и корректировка текстовой разметки	АРМ разметчика
Изменить запись	Редактирование записи, настройка ханипотов	АРМ администратора, АРМ эксперта
Последние изменения записей	Журнал правок по заданиям	АРМ эксперта, АРМ администратора
Список проектов	Перечень доступных проектов	АРМ администратора
Список исполнителей	Пользователи, назначенные на проект	АРМ администратора
Аналитика и статистика	Мониторинг активности, метрик	АРМ куратора, АРМ администратора
Подсказки по интерфейсу	Контекстная справка	Все АРМ
Справочник по правилам разметки	Методические рекомендации	АРМ разметчика
Панель администратора	Управление проектами и пользователями	АРМ администратора
Глобальное меню	Основная навигация по системе	Все пользователи
Боковая панель	Навигация по задачам, проектам	Все пользователи
Панель уведомлений	Системные сообщения	Все пользователи
Авторизация пользователя	Вход в систему	Главное меню
Инструкция	Доступ к документации	Главное меню
Пагинация	Разделение длинных списков	Все АРМ
Справка о прогрессе	Оценка выполненного объема	АРМ разметчика
Последние комментарии	Список последних замечаний	АРМ эксперта

Отображение записей	Форма просмотра всех загруженных заданий	АРМ администратора
Распределение неразмеченных записей	Назначение заданий исполнителям	АРМ администратора
Журнал событий	История действий в системе	АРМ администратора, АРМ эксперта
Календарь	Планирование смен и активности	АРМ куратора
Облако тегов	Визуализация часто используемых меток	АРМ разметчика
Меню: От сборщика	Назначенные записи от сборщика	АРМ разметчика
Субтитры/Аудиокниги	Работа с контентом типа субтитров	АРМ разметчика
Разметка личных данных	Аннотирование конфиденциальной информации	АРМ разметчика
Личный кабинет	Профиль пользователя	Все пользователи
Управление командой проекта	Назначение и контроль исполнителей	АРМ администратора

4.18.1 Веб-часть: Информация о сеансе

Назначение веб-части:

Отображает сведения о текущем сеансе пользователя: активный проект, роль, текущее задание, оставшееся время и статус активности.

Где используется веб-часть:

Все АРМ (разметчик, администратор, эксперт, куратор).

Текстовый мокап веб-части:

```
┌──────────────────────────────────┐
│ Пользователь: Пользователь A [Разметчик] │
│ Проект: Анализ отзывов           │
│ Текущий сеанс: 00:18:44          │
│ Роль: Разметчик                  │
│ Статус: В работе                 │
└──────────────────────────────────┘
```

Как разработать веб-часть:

1. React-компонент с `useEffect` для периодического обновления;
2. Вызов REST или GraphQL API для получения статуса сеанса;
3. Стилизация через `styled-components` или `MUI Box`.

4.18.2 Веб-часть: Рабочее задание

Назначение веб-части:

Выводит краткую информацию по текущему заданию, включая его номер, дату и краткое описание.

Где используется веб-часть:

АРМ разметчика.

Текстовый мокап веб-части:

```
Задание № 243 из 500
Тип: Видео
Назначено: 01.04.2025
Статус: В процессе
```

Как разработать веб-часть:

1. Мини-компонент с отображением props из главного экрана;
2. API-запрос на `/api/task/active`;
3. Поддержка смены состояния (в работе / завершено).

4.18.3 Веб-часть: Модуль разметки

Назначение веб-части:

Главный функциональный блок, предоставляющий инструменты аннотирования — чекбоксы, выделение, подписи, ввод текста и др.

Где используется веб-часть:

АРМ разметчика.

Текстовый мокап веб-части:

```
▶ Аудиофайл: feedback_001.mp3
[ ] Содержит эмоциональную речь
[ ] Присутствует фоновой шум
Комментарий: ____________________
[ Сохранить ] [ Пропустить ]
```

Как разработать веб-часть:

1. Конфигурируемый React-компонент с динамической генерацией полей по JSON-шаблону;
2. Связь с API сохранения результата: `/api/annotation/save`;
3. Состояния: новое, редактируется, отправлено.

4.18.4 Веб-часть: Редактор звуковых файлов

Назначение веб-части:

Позволяет прослушивать аудиофайл, отмечать временные интервалы и привязывать к ним аннотации.

Где используется веб-часть:

АРМ разметчика, АРМ эксперта.

Текстовый мокап веб-части:

```
┌──────────────────────────────────────┐
│ ▷ 🔊 00:00 ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒ 03:12 │
│ [ + Добавить аннотацию ]             │
│ Аннотация: [ ] Запись интервью       │
│ Интервал: 00:00 – 00:17              │
└──────────────────────────────────────┘
```

Как разработать веб-часть:

1. Библиотека: `wavesurfer.js` или `howler.js`;
2. Привязка аннотаций к таймкодам;
3. Возможность масштабирования временной шкалы.

4.18.5 Веб-часть: Редактор видео файлов

Назначение веб-части:

Воспроизведение видео и аннотирование его фрагментов по времени или кадрам. Возможность выделения объектов в кадре.

Где используется веб-часть:

АРМ разметчика.

Текстовый мокап веб-части:

```
▶ Воспроизведение: video_03.mp4
□ Отметить объект: [ Лицо ]
Кадр: 1280 × 720, Текущая позиция: 00:13
[ ] Присутствует транспорт
```

Как разработать веб-часть:

1. Воспроизведение через `<video>` + canvas;
2. Отметки по таймкодам или keyframe-интервалам;
3. Использование сторонних JS-библиотек: `fabric.js` или `konva.js`.

4.18.6 Веб-часть: Редактор текста

Назначение веб-части:

Позволяет пользователю просматривать и аннотировать текстовые данные, используя настраиваемые шаблоны (например, сущности, категории, эмоциональные маркеры).

Где используется веб-часть:

АРМ разметчика.

Текстовый мокап веб-части:

```
Текст: "Компания [Организация] открыла новый.."
[[Город]] [География]
[[Организация]] [Организация]
[ + Добавить метку ]
```

Как разработать веб-часть:

1. Использовать `contentEditable` с подсветкой сущностей;
2. Библиотеки: `react-draft-wysiwyg` или `Slate.js`;
3. Поддержка drag-n-drop, быстрых клавиш и автоподстановок.

4.18.7 Веб-часть: Изменить запись

Назначение веб-части:

Позволяет вручную редактировать выбранную запись: изменить метки, комментарии, удалить или назначить повторную разметку.

Где используется веб-часть:

АРМ администратора, АРМ эксперта.

Текстовый мокап веб-части:

```
Задание № 42
Тип: Текст
Метки: [Организация], [География]
Комментарий администратора: ____________
[ Изменить ]  [ Удалить ]  [ Вернуть на разметку ]
```

Как разработать веб-часть:

1. React-форма с модальным окном;
2. Связь с API `/api/record/{id}/update`;
3. Поддержка логирования изменений и прав доступа (RBAC).

4.18.8 Веб-часть: Последние изменения записей

Назначение веб-части:

Отображает журнал недавних изменений в записях: кто и когда отредактировал, какие поля были изменены, кто вернул на доработку.

Где используется веб-часть:

АРМ администратора, АРМ эксперта.

Текстовый мокап веб-части:

```
04.04.2025 – Пользователь А изменил:
→ Метка «[Город]» → «География»
→ Комментарий добавлен
03.04.2025 – Пользователь Б вернула задание
→ Причина: неточность
```

Как разработать веб-часть:

1. Таблица с данными из `/api/audit/record-log`;
2. Отображение в виде timeline (`react-vertical-timeline`);
3. Фильтрация по дате, пользователю и проекту.

4.18.9 Веб-часть: Список проектов

Назначение веб-части:

Предоставляет интерфейс для просмотра, фильтрации и перехода к настройке проектов разметки.

Где используется веб-часть:

АРМ администратора.

Текстовый мокап веб-части:

```
┌──────────────────────────────────────┐
│ 🔍 [Поиск проекта]                   │
│ ▭ Разметка текстов для ОЦР           │
│ ▭ Эмоциональный анализ аудио         │
│ ▭ Видео-анализ для Минобрнауки       │
│ [ + Новый проект ]                   │
└──────────────────────────────────────┘
```

Как разработать веб-часть:

1. Таблица с пагинацией (`MUI Table` или `Antd Table`);
2. Поддержка CRUD через GraphQL: `projectList`, `createProject`;
3. Возможность группировки и фильтрации по статусу и дате.

4.18.10 Веб-часть: Список исполнителей

Назначение веб-части:

Отображает участников проекта, их роли, статусы, рейтинги и активность.

Где используется веб-часть:

АРМ администратора.

Текстовый мокап веб-части:

```
┌──────────────────────────────────────┐
│ ▯ Пользователь А [Разметчик] ★★★★☆ Активен │
│ ▯ Смирнова А.А. [Эксперт]    В отпуске │
│ ▯ Васильев Б.Б. [Разметчик] ★★★☆☆ В работе │
│ [ + Пригласить пользователя ]        │
└──────────────────────────────────────┘
```

Как разработать веб-часть:

1. Список исполнителей проекта (по `project_id`);
2. Интеграция с системой ролей (RBAC);
3. Возможность изменения статуса, роли, удаления.

4.18.11 Веб-часть: Аналитика и статистика

Назначение веб-части:

Предоставляет агрегированные данные по активности пользователей, качеству разметки, загрузке проектов и другим ключевым метрикам.

Где используется веб-часть:

АРМ администратора, АРМ куратора.

Текстовый мокап веб-части:

```
┌──────────────────────────────────────┐
│ 📊 Статистика проекта "Видеоаналитика"
│ ▭ Разметка: 82% завершено
│ ▭ Средний рейтинг: ★★★★☆
│ ▭ Активные разметчики: 12 из 20
│ ▭ Кол-во ханипотов: 320
│ [Скачать отчет] [Открыть график]
└──────────────────────────────────────┘
```

Как разработать веб-часть:

1. Использовать `Recharts` / `Chart.js` для визуализации;
2. Данные через GraphQL-запросы: `statsByProject`, `userActivity`;
3. Поддержка экспорта в `.csv` и `.pdf`.

4.18.12 Веб-часть: Подсказки по интерфейсу

Назначение веб-части:

Выводит пользователю краткие рекомендации и инструкции по работе с интерфейсом, включая контекстные подсказки.

Где используется веб-часть:

Все АРМ.

Текстовый мокап веб-части:

```
┌──────────────────────────────────┐
│ ℹ Подсказка:                     │
│ Вы можете нажать [Tab], чтобы быстро │
│ переключиться на следующее задание. │
│ [X] Больше не показывать         │
└──────────────────────────────────┘
```

Как разработать веб-часть:

1. Плавающее всплывающее окно (`Tooltip`, `Toast`);
2. Хранение состояния показа (`localStorage`);
3. Поддержка многоязычности (i18n).

4.18.13 Веб-часть: Справочник по правилам разметки

Назначение веб-части:

Отображает справочную информацию: инструкции, примеры разметки, схемы и методику оценки качества.

Где используется веб-часть:

АРМ разметчика.

Текстовый мокап веб-части:

```
┌──────────────────────────────────┐
│ ℹ Правила разметки: "Тональность текста" │
│ ▭ [Категории]: позитив / негатив / нейтрал │
│ ▭ [Пример]: "Это отличный товар!" → позитив │
│ [Скачать PDF]                    │
└──────────────────────────────────┘
```

Как разработать веб-часть:

1. Компонент с вкладками или аккордеоном (`Tabs`, `Collapse`);
2. Возможность загрузки markdown или HTML-инструкций;
3. Интеграция с файловым хранилищем или CMS.

4.18.14 Веб-часть: Панель администратора

Назначение веб-части:

Главная административная панель для управления проектами, пользователями, шаблонами, ханипотами и выгрузками.

Где используется веб-часть:

АРМ администратора.

Текстовый мокап веб-части:

```
┌──────────────────────────────────────┐
│ Панель управления                    │
│ [ Проекты ] [ Пользователи ] [ Шаблоны ] │
│ [ Выгрузка ] [ Рейтинг ] [ Настройки ⚙ ] │
└──────────────────────────────────────┘
```

Как разработать веб-часть:

1. Стороннее меню + `Router`-навигация;
2. Иконки (например, `react-icons`), табы, карточки;
3. Разделение прав по ролям и модульное подключение блоков.

4.18.15 Веб-часть: Глобальное меню

Назначение веб-части:

Навигация по основным разделам платформы: профиль, проекты, задания, уведомления, выход.

Где используется веб-часть:

Все пользователи.

Текстовый мокап веб-части:

```
┌──────────── Глобальное меню ────────────┐
│ 👤 Профиль        📁 Мои проекты        │
│ 📊 Статистика     📋 Мои задания        │
│ 🔔 Уведомления     🚪 Выйти из системы  │
└─────────────────────────────────────────┘
```

Как разработать веб-часть:

1. Бургер-меню для мобильной версии;
2. Использовать `Drawer`/`Sidebar` (`MUI`, `Antd`);
3. Навигация через `react-router` или `next/router`.

4.18.16 Веб-часть: Боковая панель

Назначение веб-части:

Обеспечивает постоянный доступ к ключевым разделам текущей роли: задания, шаблоны, справка, журнал, метки и пр.

Где используется веб-часть:

Все пользователи (АРМ разметчика, администратора, эксперта, куратора).

Текстовый мокап веб-части:

```
┌──────────── Боковая панель ────────────┐
│ 📋 Мои задания                         │
│ 📄 Шаблоны                             │
│ 📖 Инструкция                          │
│ 📜 Журнал изменений                    │
│ 🏷 Метки / справочник                  │
└────────────────────────────────────────┘
```

Как разработать веб-часть:

1. Использовать `Drawer`, `Sidebar` или `Collapse menu`;
2. Интеграция с маршрутизацией приложения;
3. Адаптивность и скрытие в мобильной версии.

4.18.17 Веб-часть: Панель уведомлений

Назначение веб-части:

Отображает системные сообщения: новые задания, смена роли, ошибки, оповещения о дедлайнах и отчётах.

Где используется веб-часть:

Все пользователи.

Текстовый мокап веб-части:

```
┌──────────── Уведомления ────────────┐
│ ✅Вы успешно завершили задание №12  │
│ ⏰ Осталось 3 дня до дедлайна       │
│ 📢 Доступен новый шаблон: "OCR 2025" │
└─────────────────────────────────────┘
```

Как разработать веб-часть:

1. `Snackbar` / `Toast`-уведомления (`react-toastify`, `MUI Snackbar`);
2. Подписка на уведомления через WebSocket или polling;
3. Уровни важности (инфо, ошибка, предупреждение).

4.18.18 Веб-часть: Авторизация пользователя

Назначение веб-части:

Форма входа в систему с поддержкой пароля, SSO или LDAP. Может включать CAPTCHA, двухфакторную авторизацию и выбор роли.

Где используется веб-часть:

Главное меню (до входа в систему).

Текстовый мокап веб-части:

```
┌──────────── Вход в систему ────────────┐
│ 🔑 Логин: [__________________]         │
│ 🔑 Пароль: [_______________]           │
│ [ Войти ] [ Восстановить доступ ]       │
└────────────────────────────────────────┘
```

Как разработать веб-часть:

1. Форма с POST-запросом на `/auth/login`;
2. Поддержка редиректов, хранения токена (`JWT`, `cookie`);
3. Защита через HTTPS, антибот-защита.

4.18.19 Веб-часть: Инструкция

Назначение веб-части:

Отображение обучающих материалов по системе, типам разметки, ролям, форматам, процессам. Может быть представлена в виде PDF, HTML или встроенных карточек.

Где используется веб-часть:

Главное меню, АРМ разметчика, эксперта, администратора.

Текстовый мокап веб-части:

```
┌──────────── Инструкция ────────────┐
│ ▭ Введение в разметку              │
│ ▭ Работа с шаблонами               │
│ ▭ Рейтинг и ханипоты               │
│ 📄 [ Скачать PDF ]                 │
└────────────────────────────────────┘
```

Как разработать веб-часть:

1. Подключение markdown-рендерера (`react-markdown`);

2. Хранение в CMS / JSON-ресурсах;
3. Поддержка навигации и закладок внутри инструкции.

4.18.20 Веб-часть: Пагинация

Назначение веб-части:

Разделение длинных списков (записи, задания, отчёты) на страницы, переход между ними, установка лимита отображения.

Где используется веб-часть:

Все АРМ, где есть списки: записи, задания, проекты, отчёты.

Текстовый мокап веб-части:

```
┌──────────── Пагинация ────────────┐
│ Стр. 1 из 24  |  □□□5 6 7 □□□     │
│ □Показать: [ 20 ▾ ] на странице   │
└───────────────────────────────────┘
```

Как разработать веб-часть:

1. Компоненты пагинации (`MUI Pagination`, `Antd Pagination`);
2. Поддержка `page`, `limit` в API-запросах;
3. Состояние сохраняется при переходах между страницами.

4.18.21 Веб-часть: Справка о прогрессе

Назначение веб-части:

Показывает пользователю личную статистику по выполненным заданиям, оставшимся и просроченным. Может служить мотивационным инструментом.

Где используется веб-часть:

АРМ разметчика.

Текстовый мокап веб-части:

```
┌──────────── Прогресс пользователя ────────────┐
│ ✓ Выполнено заданий: 83 из 120                │
│ ⌛ В работе: 15                                │
│ ! Просрочено: 2                               │
│ ★ Средний рейтинг: ★★★★☆                      │
└───────────────────────────────────────────────┘
```

Как разработать веб-часть:

1. Данные из `/api/user/progress`;
2. Визуализация прогресса через `ProgressBar`, `CircularProgress`;
3. Поддержка фильтрации по периоду, типу заданий.

4.18.22 Веб-часть: Последние комментарии

Назначение веб-части:

Отображает ленту последних экспертных или модераторских комментариев к записям с возможностью быстрого перехода к соответствующему заданию.

Где используется веб-часть:

АРМ эксперта, частично — АРМ администратора.

Текстовый мокап веб-части:

```
┌──────── Комментарии к заданиям ────────┐
│ ▯ Задание #129 → "Недостаточно меток"  │
│ ▯ Задание #130 → "Отметить организацию"│
│ ▯ Задание #132 → "Повторно проверить"  │
│ [ Перейти ]                            │
└────────────────────────────────────────┘
```

Как разработать веб-часть:

1. API: `/api/comments/recent`;
2. Компонент `List` с функцией перехода (`onClick → route`);
3. Поддержка сортировки, фильтрации, поиска.

4.18.23 Веб-часть: Отображение записей

Назначение веб-части:

Выводит таблицу всех загруженных в систему или проект записей (файлов), с возможностью фильтрации, поиска, назначения, удаления.

Где используется веб-часть:

АРМ администратора.

Текстовый мокап веб-части:

```
┌──────────── Записи ────────────┐
```

```
№  Тип     Название     Статус
1  Текст   file_001.txt Готово
2  Аудио   audio_23.mp3 В работе
3  Видео   demo_4.mp4   Назначено
[Фильтр] [Поиск] [Сортировать]
```

Как разработать веб-часть:

1. `DataGrid`, `Antd Table` или `MUI Table`;
2. Поддержка загрузки через `lazy loading` / `pagination`;
3. Интеграция с фильтрами и мультиселектом.

4.18.24 Веб-часть: Распределение неразмеченных записей

Назначение веб-части:

Позволяет вручную или автоматически назначать неразмеченные записи конкретным исполнителям по фильтру, очередности, рейтингу и т.п.

Где используется веб-часть:

АРМ администратора.

Текстовый мокап веб-части:

```
Назначение заданий
☐Доступно: 320 записей
Выбрать исполнителей:
☑ Пользователь А  ☑ Пользователь Б
[Автонастройка] [Назначить]
```

Как разработать веб-часть:

1. Подбор кандидатов на основе рейтинга и загрузки;
2. Выбор вручную или с использованием пресетов;
3. API: `/api/distribute/records`.

4.18.25 Веб-часть: Журнал событий

Назначение веб-части:

Фиксирует действия всех пользователей в системе: входы, изменения, удаления, выгрузки, оценки, проверки. Используется для аудита и отладки.

Где используется веб-часть:

АРМ администратора, АРМ эксперта.

Текстовый мокап веб-части:

```
┌──────────── Журнал событий ────────────┐
│ [04.04.25 14:32] Пользователь A → Вошел в систему   │
│ [04.04.25 14:45] Отметка: файл_23.txt размечен      │
│ [04.04.25 15:12] Эксперт: комментарий добавлен      │
│ [Фильтр] [Экспорт CSV]                              │
└────────────────────────────────────────┘
```

Как разработать веб-часть:

1. API: `/api/audit/logs`, `/api/logs/export`;
2. Компонент `LogViewer`, поддержка фильтров по времени, пользователю, действию;
3. Возможность экспорта и маркировки ошибок.

4.18.26 Веб-часть: Календарь

Назначение веб-части:

Предоставляет визуальное представление графика работы пользователей, сроков заданий и событий проекта. Применяется для планирования, координации и уведомлений.

Где используется веб-часть:

АРМ куратора, частично — АРМ администратора.

Текстовый мокап веб-части:

```
┌──────────── Календарь ────────────┐
│ ▯ Апрель 2025                     │
│ ■ 04.04 – дедлайн проекта "Анализ" │
│ ■ 06.04 – отпуск Пользователь Б    │
│ ■ 08.04 – тестирование шаблонов    │
│ [◄] [►] [Фильтр по роли]           │
└───────────────────────────────────┘
```

Как разработать веб-часть:

1. Использовать библиотеку `FullCalendar`, `react-big-calendar`;
2. API для событий: `/api/calendar/events`;
3. Поддержка фильтрации по ролям, проектам, типам событий.

4.18.27 Веб-часть: Облако тегов

Назначение веб-части:

Визуализирует часто используемые метки и сущности. Позволяет анализировать тематику заданий, применять фильтры, отслеживать тренды.

Где используется веб-часть:

АРМ разметчика, возможно — в АРМ администратора для аналитики.

Текстовый мокап веб-части:

```
┌──────────── Облако тегов ────────────────────┐
│ [ ▯ [Город] ] [ ▯ Организация ] [ ▯ Технологии ] │
│ [ ▯ Эмоции ] [ ▯ Закон ] [ ▯ Образование ]      │
└──────────────────────────────────────────────┘
```

Как разработать веб-часть:

1. Визуализация через `wordcloud2.js` или `tag-cloud-react`;
2. Данные из статистики меток по проекту;
3. Клик по тегу применяет фильтр.

4.18.28 Веб-часть: Меню "От сборщика"

Назначение веб-части:

Показывает список заданий, назначенных системой или координатором сборки данных. Позволяет разметчику быстро приступить к работе.

Где используется веб-часть:

АРМ разметчика.

Текстовый мокап веб-части:

```
┌──────── Задания от сборщика ──────────┐
│ ▭Задание #154 – фото_новости.jpg      │
│ ▭Задание #155 – звук_интервью.mp3     │
│ ▭Задание #156 – текст_отзыв.txt       │
│ [ Начать выполнение ]                 │
└───────────────────────────────────────┘
```

Как разработать веб-часть:

1. Источник: `/api/assigned/from-collector`;

2. Таблица или список с кнопкой перехода к разметке;
3. Автообновление и фильтрация по дате.

4.18.29 Веб-часть: Субтитры / Аудиокниги

Назначение веб-части:

Разметка многочасовых аудио- и видеофайлов с временными рамками, разбивкой по репликам, озвучке, смысловым блокам.

Где используется веб-часть:

АРМ разметчика.

Текстовый мокап веб-части:

```
┌──────────── Субтитры ────────────┐
│ 00:00:05–00:00:10: "Добро пожаловать..." │
│ 00:00:11–00:00:18: "Это обучающий курс.." │
│ [ + Новая реплика ] [ ⟳ Прокрутка ]      │
└──────────────────────────────────┘
```

Как разработать веб-часть:

1. Аудио/видео + текст + таймкоды;
2. Использование `react-player` + ручной редактор;
3. Поддержка формата `srt`, `vtt`, `json`.

4.18.30 Веб-часть: Разметка личных данных

Назначение веб-части:

Специализированная форма разметки персональных и конфиденциальных данных: имена, адреса, документы и т.п., в рамках задач по защите информации.

Где используется веб-часть:

АРМ разметчика.

Текстовый мокап веб-части:

```
┌──────────── Разметка ПДн ────────────┐
│ Пользователь А [ ] ФИО                 │
│ 8(800)123-45-67 [ ] Телефон            │
│ passport_77.jpg [ ] Паспортные         │
│ [ Отметить как ПДн ]                   │
```

```
└──────────────────────────────────────┘
```

Как разработать веб-часть:

1. Расширение основного модуля разметки;
2. Предустановленные категории ПДн;
3. Проверка наличия маскировки / нормализации при выгрузке.

4.18.31 Веб-часть: Личный кабинет

Назначение веб-части:

Отображает и позволяет редактировать данные профиля пользователя: имя, роль, смена пароля, языковые настройки, история активности.

Где используется веб-часть:

Все пользователи (АРМ разметчика, администратора, эксперта, куратора).

Текстовый мокап веб-части:

```
┌──────────────── Личный кабинет ────────────────┐
│ Имя: Иванов Иван Иванович                      │
│ Роль: Разметчик                                │
│ Email: ivanov@example.com                      │
│ Язык интерфейса: [ Русский ▾ ]                 │
│ [ ✎ Изменить профиль ]  [ 🔑 Сменить пароль ] │
└────────────────────────────────────────────────┘
```

Как разработать веб-часть:

1. Форма с валидацией (`Formik`, `Yup`);
2. Интеграция с API `/api/user/profile`;
3. Поддержка смены пароля, мультиязычности (i18n).

4.18.32 Веб-часть: Управление командой проекта

Назначение веб-части:

Позволяет администратору проекта управлять составом команды, распределять роли, просматривать статус участников и их активность.

Где используется веб-часть:

АРМ администратора.

Текстовый мокап веб-части:

```
┌──── Управление командой проекта ──────────────┐
│ 🧑 Пользователь А [Разметчик] ★★★★☆              │
│ 🧑 Пользователь Б [Эксперт] В отпуске            │
│ 🧑 Пользователь В [Куратор]                      │
│ [ + Пригласить участника ] [ Назначить роль ]    │
└──────────────────────────────────────────────────┘
```

Как разработать веб-часть:

1. Таблица исполнителей (`Table`, `Dropdown`, `Modal`);
2. API: `/api/project/team`;
3. Поддержка RBAC, подтверждения роли, фильтрации.

4.18.33 Веб-часть: Просмотр задания и выполнение разметки

Назначение веб-части:

Предоставляет интерфейс разметчику для выполнения задания по аннотированию одного объекта (текста, аудио, изображения, видео), с применением шаблонов и встроенных инструкций.

Где используется веб-часть:

АРМ разметчика.

Текстовый мокап веб-части:

```
┌──────────────────────────────────────────┐
│ Название проекта: 📄"Разметка диалогов"  │
│ Тип данных: Аудио                        │
├──────────────────────────────────────────┤
│ Инструкция: [🧑 Открыть инструкцию]      │
├──────────────────────────────────────────┤
│ 🧑 [ Проигрыватель аудиофайла ]          │
│ [ ] Отметить имя говорящего              │
│ [ ] Отметить эмоцию                      │
│ Комментарий: ____________________        │
├──────────────────────────────────────────┤
│ [ Сохранить ]  [ Отправить ]             │
└──────────────────────────────────────────┘
```

Как разработать веб-часть:

1. Компонент на React с настройкой состояния (`useState`, `useEffect`);
2. Проигрыватель аудио/видео — `react-player`, `wavesurfer.js`;
3. Работа с шаблоном через JSON-конфигурацию;

4. API: `/api/task/submit`, `/api/template/load`.

4.18.34 Веб-часть: Управление проектом

Назначение веб-части:

Позволяет администратору контента создавать, конфигурировать, запускать проекты разметки и управлять их параметрами: задания, исполнители, шаблоны, ханипоты, статус.

Где используется веб-часть:

АРМ администратора контента.

Текстовый мокап веб-части:

```
┌──────────────────────────────────────┐
│ Проект: [▭ Создание датасета для СМИ]│
│ Статус: [ Активен ▾ ]                │
├──────────────────────────────────────┤
│ ▷ Задания: 124   ▷ Разметчики: 16    │
│ ▷ Шаблон: [Оценка тональности]       │
├──────────────────────────────────────┤
│ [ Загрузить датасет ] [ Настроить шаблон ]
│ [ Назначить пользователей ] [ Включить ханипоты ]
├──────────────────────────────────────┤
│ ⍰ [ Выгрузить результаты ▾ ]         │
└──────────────────────────────────────┘
```

Как разработать веб-часть:

1. Модульная панель (`Accordion`, `Tabs`);
2. Использование компонентов `Upload`, `Select`, `Checkbox`;
3. API: `/api/project/{id}/config`, `/api/project/upload-data`.

4.18.35 Веб-часть: Выгрузка и фильтрация данных

Назначение веб-части:

Формирует выгрузку размеченных данных по заданным параметрам: формат, период, отбор по рейтингу, способ хранения.

Где используется веб-часть:

АРМ администратора.

Текстовый мокап веб-части:

```
Выгрузка размеченных данных
Формат: [ JSON ▾ ]
Дата: 01.03.2025 – 12.03.2025
Разметчики: [ Только с рейтингом > 4 ★ ]
Место хранения: [ Hadoop ▾ ]
[ ⟳ Подготовить выгрузку ] [ ⬇ Скачать файл ]
```

Как разработать веб-часть:

1. Компоненты: `DateRangePicker`, `Select`, `Checkbox`, `Button`;
2. Backend: сериализация через `Pandas`, `Django REST Framework`;
3. API: `/api/export`, `/api/export/preview`, `/api/export/download`.

4.18.36 Веб-часть: Анализ качества (экспертная проверка)

Назначение веб-части:

Предоставляет эксперту интерфейс для сравнения эталонной разметки (ханипотов) и ответа разметчика с возможностью выставления оценки и комментария.

Где используется веб-часть:

АРМ эксперта по качеству.

Текстовый мокап веб-части:

```
Контроль ханипотов: проект "Синтез речи"
Запись №44
Эталон: [Категория: Информативность]
Разметчик: [Категория: Мусор]
Комментарий эксперта: ____________________
[ ✅Подтвердить ханипот ] [ ⚠Отправить на доразметку ]
```

Как разработать веб-часть:

1. `react-diff-viewer` или собственный diff-компонент;
2. Интерфейс оценки (`ButtonGroup`, `CommentBox`);
3. API: `/api/honeypot/verify`, `/api/honeypot/review`.

4.18.37 Веб-часть: Управление пользователями и ролями

Назначение веб-части:

Позволяет администратору контента управлять составом команды проекта: добавлять новых пользователей, изменять роли, блокировать, просматривать рейтинг.

Где используется веб-часть:

АРМ администратора.

Текстовый мокап веб-части:

```
┌──────────────────────────────────────┐
│ Пользователи проекта "Обучение ИИ"   │
├──────────────────────────────────────┤
│ ☐Пользователь А      [Разметчик] [★4]  │
│ ☐Пользователь Б     [Администратор]   │
│ ☐Пользователь В     [Эксперт]         │
├──────────────────────────────────────┤
│ [ + Добавить пользователя ]           │
│ [ ⛔Заблокировать ] [ Изменить роль ] │
└──────────────────────────────────────┘
```

Как разработать веб-часть:

1. Компоненты: `Table`, `Dropdown`, `Modal`, `InputSearch`;
2. Интеграция с RBAC и AD/Keycloak;
3. API: `/api/users`, `/api/user/update-role`, `/api/user/block`.

4.19 Описание структуры хранения и обмена данными

4.19.1 Директорная структура и сущности базы данных

В целях обеспечения функционирования веб-частей и АРМ-ов в АС «МАРКЕР» реализована структура хранения, построенная по принципу модульной директорной модели, синхронизированной с сущностями реляционной базы данных. Каждая директория и таблица БД предназначена для хранения конкретных видов информации, обеспечивая её централизованное получение, использование и актуализацию.

Таблица 4 – Директорная структура и сущности базы данных				
Наименование поля БД или директории	Назначение	Какие данные хранятся	Источник данных	Использование
projects/	Метаданные проектов	Название, статус, сроки, шаблон	Администратор контента	Управление проектами
datasets/	Исходные/размеченные данные	Файлы: .txt, .json, .mp3 и др.	Администратор, разметчик	Подготовка и обработка данных
templates/	Шаблоны разметки	JSON-схемы аннотаций	Администратор контента	Генерация интерфейсов
assignments/	Назначения заданий	ID задачи, ID пользователя, статус	Система / администратор	Формирование очереди заданий
annotations/	Результаты аннотирования	Метки, таймкоды, координаты	Разметчики	Основной результат проекта
users/	Профили	Логин, роль, рейтинг	При регистрации	Аутентификация и фильтрация
ratings/	Оценки качества	Средний рейтинг, оценка ханипотов	Система / эксперт	Управление доступом, мотивация
honeypots/	Эталонные задания	Предразмеченные записи	Администратор / эксперт	Контроль качества
logs/	Журналы активности	Действия пользователей	Система	Аудит, диагностика
comments/	Комментарии	Текст, автор, объект	Администратор / эксперт	Проверка, обратная связь
exports/	Экспортные файлы	JSON/CSV-выгрузки	Администратор контента	Передача результатов
events/	Календарь	Событие, дата, участники	Куратор / администратор	Планирование работы
notifications/	Уведомления	Сообщения, приоритет	Система / администратор	Информирование пользователей
feedback/	Отзывы пользователей	Оценки, предложения	Пользователь	Обратная связь и улучшения

4.19.2 ER-модель: связи между сущностями

Ниже представлена логическая структура связей между основными сущностями системы:

1. Проект (**`Project`**)
   a. связан с множеством **заданий (`Assignment`)** по `project_id`
   b. использует один **шаблон (`Template`)**
   c. содержит множество **датасетов (`Dataset`)**
   d. назначает **пользователей (`User`)** через таблицу `assignments`

2. Задание (**`Assignment`**)
   a. содержит одну **запись (`DatasetEntry`)**
   b. назначено одному **пользователю (`User`)**
   c. может быть помечено как **ханипот (`Honeypot`)**
3. Аннотация (**`Annotation`**)
   a. принадлежит конкретному **заданию (`Assignment`)**
   b. создаётся конкретным **разметчиком (`User`)**
   c. сравнивается с **эталоном (`Honeypot`)** для оценки
4. Рейтинг (**`Rating`**)
   a. вычисляется по множеству выполненных **ханипотов**
   b. отображается в профиле **пользователя (`User`)**
   c. влияет на допуск к новым **заданиям (`Assignment`)**
5. Комментарий (**`Comment`**)
   a. связан с конкретной **аннотацией или заданием**
   b. оставляется **экспертом** или **администратором**
6. Экспорт (**`Export`**)
   a. связан с конкретным **проектом** и множеством **аннотаций**
   b. инициируется **администратором**
7. Событие (**`Event`**)
   a. связано с **проектом** или **пользователями**
   b. отображается в **АРМ куратора**
8. Уведомление (**`Notification`**)
   a. направляется одному или нескольким **пользователям**
   b. связано с событием или результатом действия

4.19.3 Характеристики модели хранения

1. Все данные логически связаны с проектами и действиями пользователей;
2. Данные о разметке, заданиях, оценках и комментариях обновляются в реальном времени;
3. Система поддерживает отложенную и параллельную выгрузку через очередь экспорта;
4. Все ключевые действия логируются с меткой времени и пользователя;

5. Хранилище данных масштабируемо (в том числе с использованием Hadoop или S3-совместимых сервисов).

4.1 ER-диаграмма: связи между сущностями

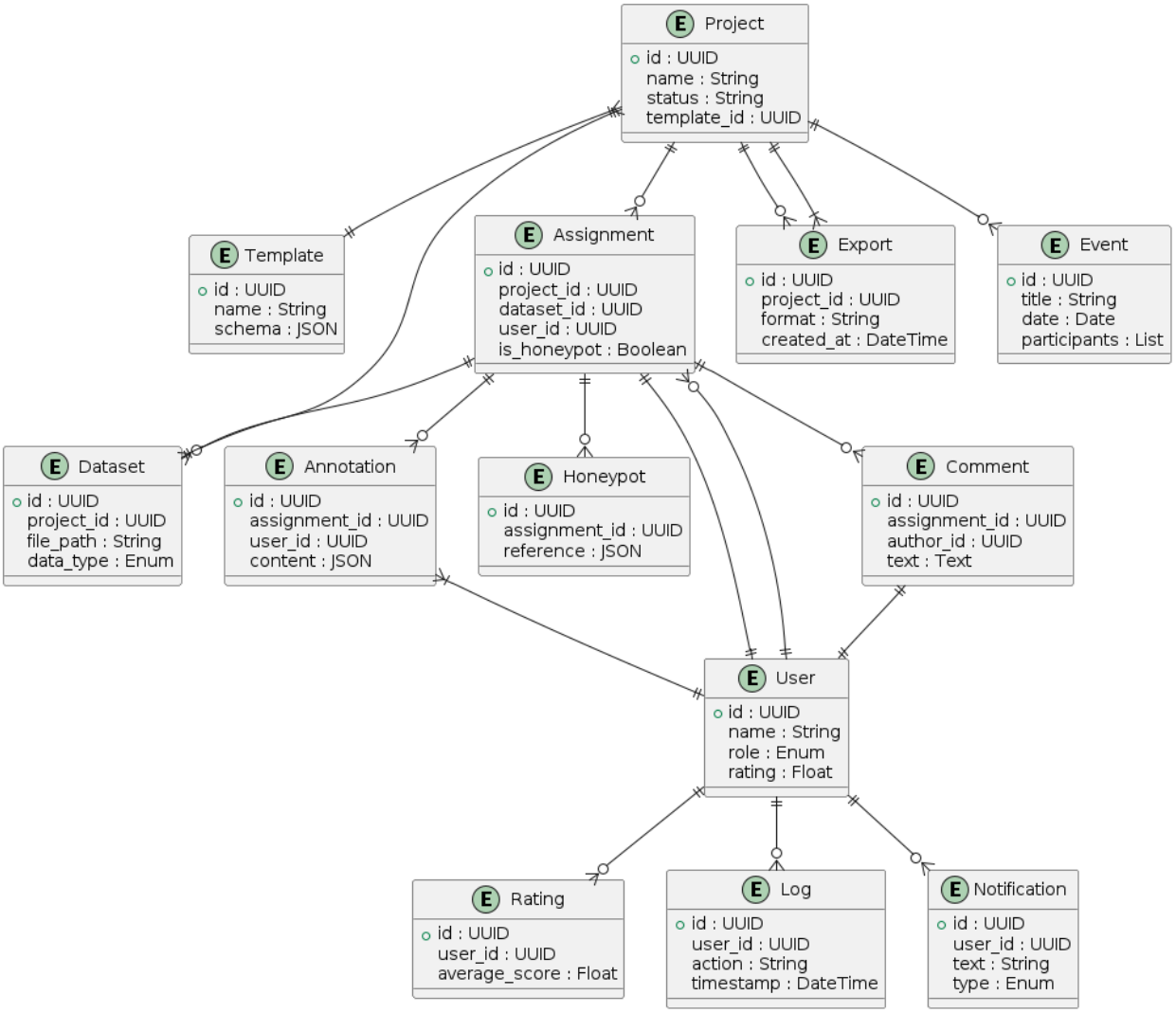

Рисунок 1 – ER-диаграмма сущностей проекта

4.2 Архитектурная схема хранения данных

Раздел 4.2 описывает архитектуру хранения данных в системе «МАРКЕР». На рисунке 4 представлена схема директорной структуры и её связь с базой данных, обеспечивающей обмен данными между веб-частями и АРМ-ами.

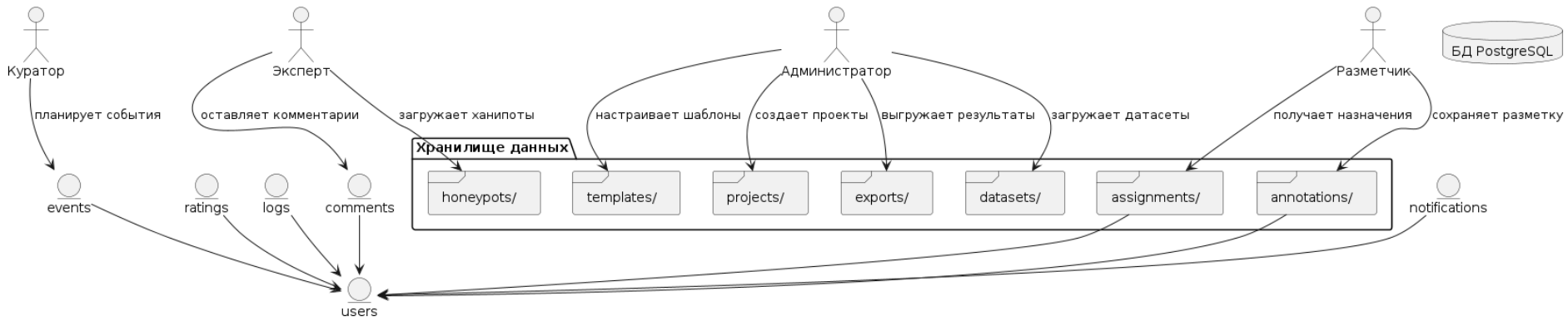

Рисунок 2 – Архитектура директорной структуры хранения данных

4.3 Интерфейс АС «МАРКЕР»

Рисунок 5 демонстрирует распределение веб-частей по областям пользовательского интерфейса. Каждая зона представлена как логическая или визуальная группа компонентов. Макет может быть использован как wireframe-визуализация или как структурное описание интерфейса в составе технического проекта.

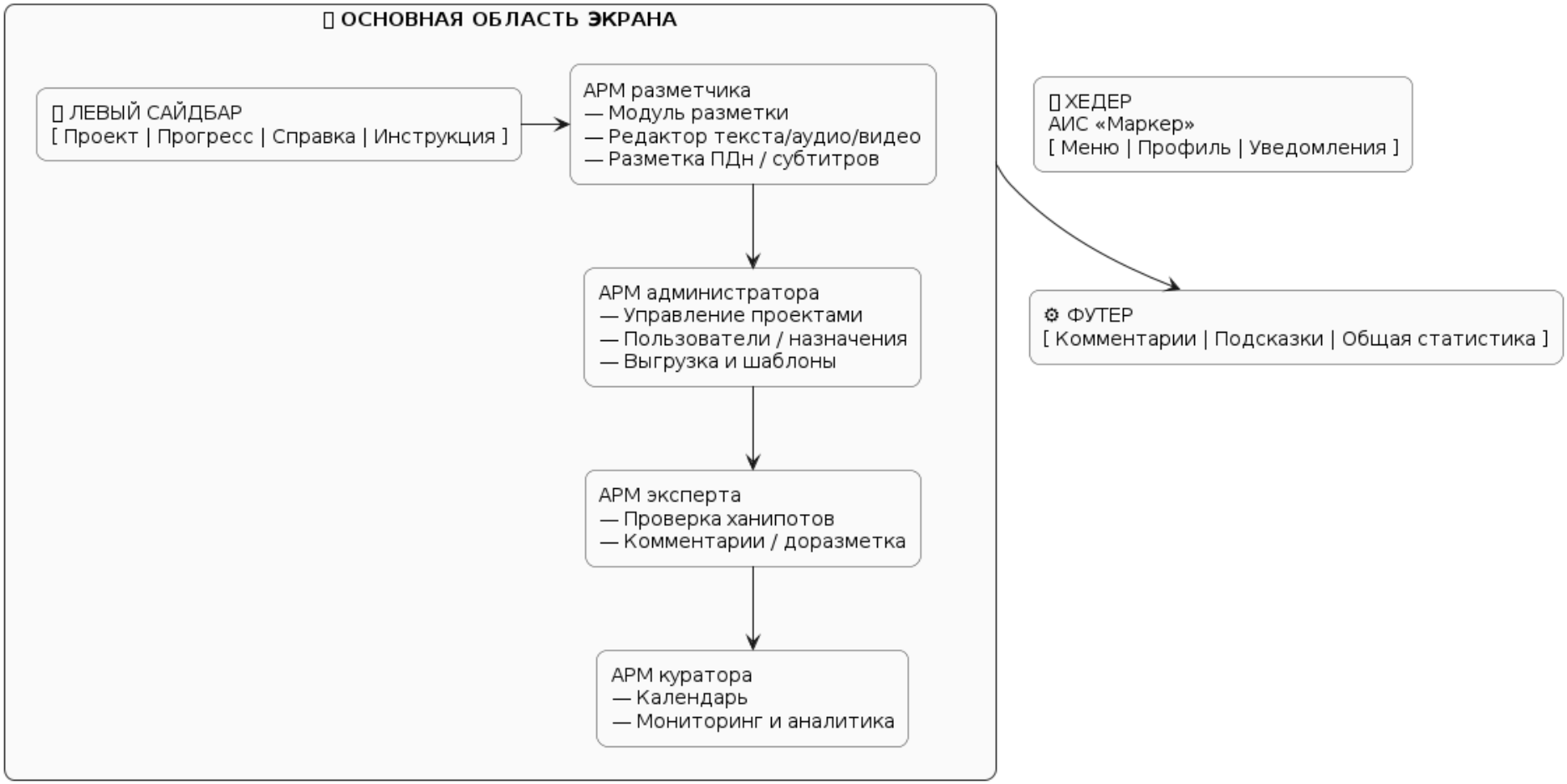

Рисунок 5 – Интерфейс АС «МАРКЕР»

5 МЕРОПРИЯТИЯ ПО ПОДГОТОВКЕ ОБЪЕКТА АВТОМАТИЗАЦИИ К ВВОДУ СИСТЕМЫ В ДЕЙСТВИЕ

5.1 Мероприятия по приведению информации к виду, пригодному для обработки на ЭВМ

В рамках подготовки объекта автоматизации к эксплуатации АС «МАРКЕР» необходимо выполнить следующие мероприятия по подготовке исходных данных:

1. Провести аудит текущих форматов хранения данных (текст, изображения, видео, аудио);
2. Определить форматы файлов, подлежащие загрузке в АС: `.txt`, `.csv`, `.json`, `.jpg`, `.png`, `.mp3`, `.wav`, `.mp4`, `.parquet` и др.;
3. Осуществить предварительную фильтрацию и категоризацию данных;
4. Разработать и согласовать структуру метаданных, сопровождающих исходные файлы;
5. Привести именование и структуру файлов в соответствие с шаблонами, принятыми в АС;
6. Обеспечить организацию входных данных в виде датасетов, пригодных для аннотирования.

В случае необходимости — провести массовую конвертацию данных с использованием предустановленных утилит или средств автоматизированной подготовки (например, скриптов Python для обработки текстов и аудио).

5.2 Мероприятия по обучению и проверке квалификации персонала

До ввода АС «МАРКЕР» в эксплуатацию необходимо организовать мероприятия по подготовке персонала:

1. **Разметчики:**
   a. Проведение вводного инструктажа по интерфейсу и функционалу АРМ;
   b. Обучение работе с шаблонами аннотирования;
   c. Проведение тестовой разметки учебного датасета с последующей оценкой качества.
2. **Администраторы контента:**

a. Обучение процедурам создания проектов, управления пользователями и выгрузки данных;

b. Настройка ханипотов, шаблонов и категорий;

c. Обработка статистических и аудиторских отчётов.

3. **Системные администраторы:**

a. Инструктаж по развёртыванию, сопровождению и резервному копированию АС;

b. Практикумы по обновлению компонентов системы;

c. Настройка мониторинга, логирования, аудита и восстановления данных.

По итогам обучения рекомендуется проведение тестирования и сертификации участников по соответствующим ролям.

5.3 Мероприятия по созданию необходимых подразделений и рабочих мест

В рамках внедрения АС «МАРКЕР» необходимо предусмотреть:

1. Создание центра администрирования АС — не менее 1 рабочего места администратора контента;
2. Назначение системного администратора (может быть совместителем);
3. Организацию рабочих мест для разметчиков (стационарные ПК или ноутбуки с доступом в интернет и браузером Google Chrome / Mozilla Firefox);
4. Обеспечение доступности служебной информации и инструкций по работе с системой;
5. Назначение ответственного лица по каждой роли в проекте (куратор, координатор и т.д.).

5.4 Мероприятия по изменению объекта автоматизации

Если в ходе внедрения АС выявляется необходимость модернизации существующей инфраструктуры, должны быть предприняты следующие действия:

1. Проверка и, при необходимости, расширение пропускной способности каналов связи;
2. Модернизация серверов хранения и обработки данных (при высокой нагрузке);
3. Развёртывание или актуализация Active Directory и системы Single Sign-On;

4. Подключение резервных источников питания и ИБП в ЦОД;

5. Внесение дополнений в существующие регламенты по ИБ, кадровому учёту и документообороту.

5.5 Другие мероприятия, исходящие из специфических особенностей создаваемых АС

Учитывая специфику проекта, в состав подготовительных мероприятий также включаются:

1. Настройка системы учёта производительности и статистики по каждому проекту;
2. Формирование и внедрение шаблонов инструкций для разметчиков;
3. Подключение юридически значимых справочников и справочников идентификаторов;
4. Проведение пилотного тестирования с участием ограниченной группы пользователей;
5. Организация канала обратной связи и службы поддержки (внутренний Helpdesk или через почту/чат);
6. Проверка возможности запуска системы в условиях частичной недоступности компонентов (аварийный режим).

6 ПЕРЕЧЕНЬ ТЕРМИНОВ И СОКРАЩЕНИЙ

Термин / сокращение	Расшифровка / определение
AD (Active Directory)	Служба каталогов для хранения информации об объектах в сети и управления доступом
API	Application Programming Interface — программный интерфейс приложения
Backend	Серверная часть системы, реализующая бизнес-логику и обработку запросов
CI/CD	Непрерывная интеграция и доставка программного обеспечения
CRUD	Создание (Create), чтение (Read), обновление (Update), удаление (Delete) — базовые операции над данными
Frontend	Клиентская часть веб-приложения, реализующая пользовательский интерфейс
GraphQL	Язык запросов к API, позволяющий выбирать только необходимые данные
JSON	JavaScript Object Notation — формат структурированных данных
Keycloak	Система централизованной аутентификации и авторизации с поддержкой SSO
LDAP	Протокол доступа к каталогам, используется для интеграции с AD
MNF	Внутренний формат сериализации разметки, применяемый в АС «МАРКЕР»
NIP / НИОКР	Научно-исследовательская работа / Научно-исследовательские и опытно-конструкторские работы
RBAC	Role-Based Access Control — модель управления доступом по ролям
REST API	Интерфейс взаимодействия по протоколу HTTP с использованием принципов REST
SLA	Service Level Agreement — соглашение об уровне обслуживания
АРМ	Автоматизированное рабочее место
АС	Автоматизированная система
Веб-часть	Логический компонент интерфейса пользователя в системе «МАРКЕР»
Датасет	Совокупность размеченных или исходных данных, объединённых по теме или признаку
ПО	Программное обеспечение
Разметка	Аннотирование, присвоение меток элементам данных (текст, аудио, изображение, видео) с целью подготовки обучающих выборок
РФРИТ	Российский фонд развития информационных технологий — оператор гранта
СУБД	Система управления базами данных
Ханипот (honeypot)	Контрольная запись с заранее известной разметкой, используемая для оценки качества работы разметчиков
Хранилище Hadoop	Распределённая файловая система (HDFS) и вычислительная платформа для хранения и обработки больших данных

Лист регистрации изменений

Изм.	Номера листов (страниц)				Всего листов (страниц) в документе	Номер документа	Входящий номер сопроводительного документа и дата	Подпись	Дата
	измененных	замененных	новых	аннулированных					

Подп. и дата

Инв. № дубл.

Взам. инв. №

Подп. и дата

Инв. № подл.

Изм	Лист	№ докум.	Подп.	Дата	38264925.425000.403.П2	Лист
						62