

**Платформа интеллектуальной обработки обращений, текста,
документов, картинок, голоса без участия человека с применением
технологий генеративного искусственного интеллекта для
решения отраслевых задач «ГЕН.ИИ»**

ОПИСАНИЕ ПРОЦЕССОВ, ОБЕСПЕЧИВАЮЩИХ ПОДДЕРЖАНИЕ ЖИЗНЕННОГО
ЦИКЛА

АННОТАЦИЯ

В настоящем документе приведено описание процессов, обеспечивающих поддержание жизненного цикла программного обеспечения «ГЕН.ИИ» - интеллектуальной системы поиска и анализа информации в документах различного назначения на базе моделей искусственного интеллекта с применением подхода Retrieval-Augmented Generation (RAG).

Настоящий документ содержит:

- Назначение и функциональные характеристики системы;
- Роли исполнителей, принимающих участие в процессах жизненного цикла программы;
- Модель жизненного цикла системы.

Описание жизненного цикла приведено в соответствии с ГОСТ Р ИСО/МЭК 12207-2010.

Содержание

1. Общие сведения	6
1.1. Назначение системы	6
1.2. Возможности системы	7
1.3. Состав системы	10
2. Роли исполнителей	15
3. Стадии жизненного цикла Системы	18
3.1. Перечень стадий жизненного цикла Системы	18
3.2. Стадия сбора и анализа требований	18
3.3. Стадия проектирования архитектуры и разработки	18
3.4. Стадия передачи	20
3.5. Стадия эксплуатации и сопровождения	22
3.6. Стадия изъятия и утилизации	24
4. Информация о персонале, необходимом для обеспечения поддержки жизненного цикла ...	25
4.1. Роли участников Системы	25
5. Поддержание жизненного цикла программного обеспечения	25
5.1. Контактная информация	26

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

В настоящем документе используются следующие термины:

Термин	Определение
БД	База данных.
Grafana	Платформа для мониторинга, анализа и визуализации данных.
ITSM-система	Система управления сервисом для покрытия бизнес-потребностей, включающая в себя регистрацию инцидентов.
ИТ	Информационные технологии.
ИИ	Искусственный интеллект.
JSON	Текстовый формат обмена данными, основанный на JavaScript.
LLM	Крупная языковая модель, обученная на больших объемах текстовых данных для генерации, анализа и обработки естественного языка с помощью методов искусственного интеллекта.
ML Flow	Платформа, предназначенная для управления жизненным циклом моделей машинного обучения.
NTFS	Стандартная файловая система для семейства операционных систем Windows NT, поддерживающая хранение метаданных.
ПО	Программное обеспечение.
PostgreSQL	Свободная объектно-реляционная система управления базами данных.
Поставщик	Организация или лицо, которое вступает в соглашение с приобретающей стороной на поставку продукта или услуги.
Промпт	Запрос на естественном языке, используемый для взаимодействия с языковой моделью или другой системой искусственного интеллекта с целью получения ответа или выполнения задачи.
Продукт,	Программа для ЭВМ, обеспечивающая автоматизацию

система	деятельности.
Qdrant	Система управления векторной базой данных, предназначенная для хранения, поиска и обработки векторных представлений данных (эмбеддингов), используемых в задачах машинного обучения и искусственного интеллекта.
RAG	Технология генерации текстов с дополнением извлечённой информации (Retrieval-Augmented Generation), объединяющая поиск релевантных данных и генерацию ответов на их основе с помощью моделей искусственного интеллекта.
Разработчик	Организация, которая выполняет разработку задач (в том числе анализ требований, проектирование, приемочные испытания) в процессе жизненного цикла.
СУБД	Система управления базами данных.
ТП	Техническая поддержка.
ЖЦ	Жизненный цикл.

1. Общие сведения

1.1. Назначение системы

Программное обеспечение ГЕН.ИИ представляет собой интеллектуальную систему, предназначенную для автоматизации обработки входящей информации и взаимодействия с корпоративными знаниями.

Система включает в себя два взаимодополняющих модуля, которые могут использоваться как совместно, в рамках интегрированной платформы, так и по отдельности, в зависимости от задач организации:

1. Модуль интеллектуального диалогового поиска по базе знаний
2. Модуль автоматизации обработки обращений на основе машинного обучения

Оба модуля направлены на повышение эффективности обработки входящей информации, но решают разные задачи. Первый специализируется на генерации интеллектуальных ответов на неструктурированные запросы, второй - на автоматической классификации и прогнозировании параметров обращений с помощью обучаемых моделей ИИ.

1.1. Возможности системы

Модуль интеллектуального диалогового поиска по базе знаний

Данный модуль может использоваться как часть интегрированной системы автоматизации обработки обращений, либо автономно - в качестве AI-помощника для работы с внутренней базой знаний компании.

В составе системы модуль выполняет роль интеллектуального обработчика сложных или неструктурированных запросов. В зависимости от бизнес-правил, обращения могут направляться как на обработку классическими ML-моделями, так и в данный модуль. При этом возможна параллельная маршрутизация одного и того же запроса одновременно в ML-сервисы и в LLM-модуль, что позволяет объединять результаты различных подходов и использовать их совместно для принятия итогового бизнес-решения.

Для обработки обращений в модуле используется подход Retrieval-Augmented Generation (RAG), при котором сначала производится поиск релевантной информации в базе знаний с помощью векторного поиска, а затем на основе найденного контекста генерируется ответ с использованием большой языковой модели (LLM).

В автономном режиме модуль функционирует как интеллектуальный помощник, предоставляющий доступ к корпоративной базе знаний через веб-интерфейс. Пользователь

может задавать вопросы в чате, просматривать загруженные документы и получать ответы, сформированные на основе семантического анализа запроса и найденной информации.

Модуль автоматизации обработки обращений на основе машинного обучения

Модуль предназначен для интеллектуальной обработки входящих заявок с использованием методов машинного обучения. Может работать как часть интегрированной платформы, совместно с модулем диалогового поиска, либо автономно - в качестве предиктивного сервиса для анализа и классификации обращений.

Основная задача модуля - автоматическое определение параметров заявки: тематики, приоритета, ответственного исполнителя и других, на основе обученных ML-моделей.

Модуль реализует полный цикл: от приема и подготовки данных до обучения моделей и генерации прогнозов в режиме реального времени. Обращения могут обрабатываться как исключительно внутри модуля, так и в связке с LLM через LLM-коннектор, что позволяет использовать ответы модели в качестве дополнительного источника данных.

Во встроенном интерфейсе доступны инструменты для настройки источников данных, обучения моделей, задания бизнес-правил обработки и мониторинга качества работы.

Роли и права доступа

Программное обеспечение предусматривает гибкую систему управления доступом через три основные роли:

- Администратор с полным доступом ко всем функциям,
- Менеджер с возможностью управления базами знаний и ограниченными правами администрирования,
- Оператор с базовыми функциями работы с существующими документами и чатом.

Такая структура обеспечивает необходимый уровень безопасности и контроля над информационными ресурсами.

1.2. Возможности системы

Функциональные возможности Модуля интеллектуального диалогового поиска по базе знаний

- Создание и управление изолированными базами знаний для различных проектов и доменов
- Множественная загрузка и обработка документов в популярных текстовых форматах (PDF, DOCX, TXT).
- Преобразование содержимого документов в формат, пригодный для семантического анализа.
- Поиск информации на естественном языке с учетом смыслового контекста
- Генерация точных и обоснованных ответов с использованием технологии RAG (Retrieval-Augmented Generation).
- Отображение источников информации, использованных при формировании ответа.
- Возможность перехода от ответа к полному фрагменту или документу-источнику.
- Работа с неструктурированным текстом без необходимости ручной разметки.
- Разграничение прав доступа на основе ролей (администратор, менеджер, оператор).
- Гибкая настройка пользовательских прав и управление доступом к базам знаний.
- Управление пользователями: создание, удаление, назначение ролей.
- Обработка типовых ошибок (отсутствие документов, превышение допустимого размера файла).
- Информирование пользователя о невозможности ответа при отсутствии релевантных данных.
- Централизованное управление данными через интерфейс для администрирования.
- Интеграция с модулем автоматической классификации обращений, где данные из базы знаний могут использоваться для обогащения входного контекста и повышения качества прогнозов ML-моделей

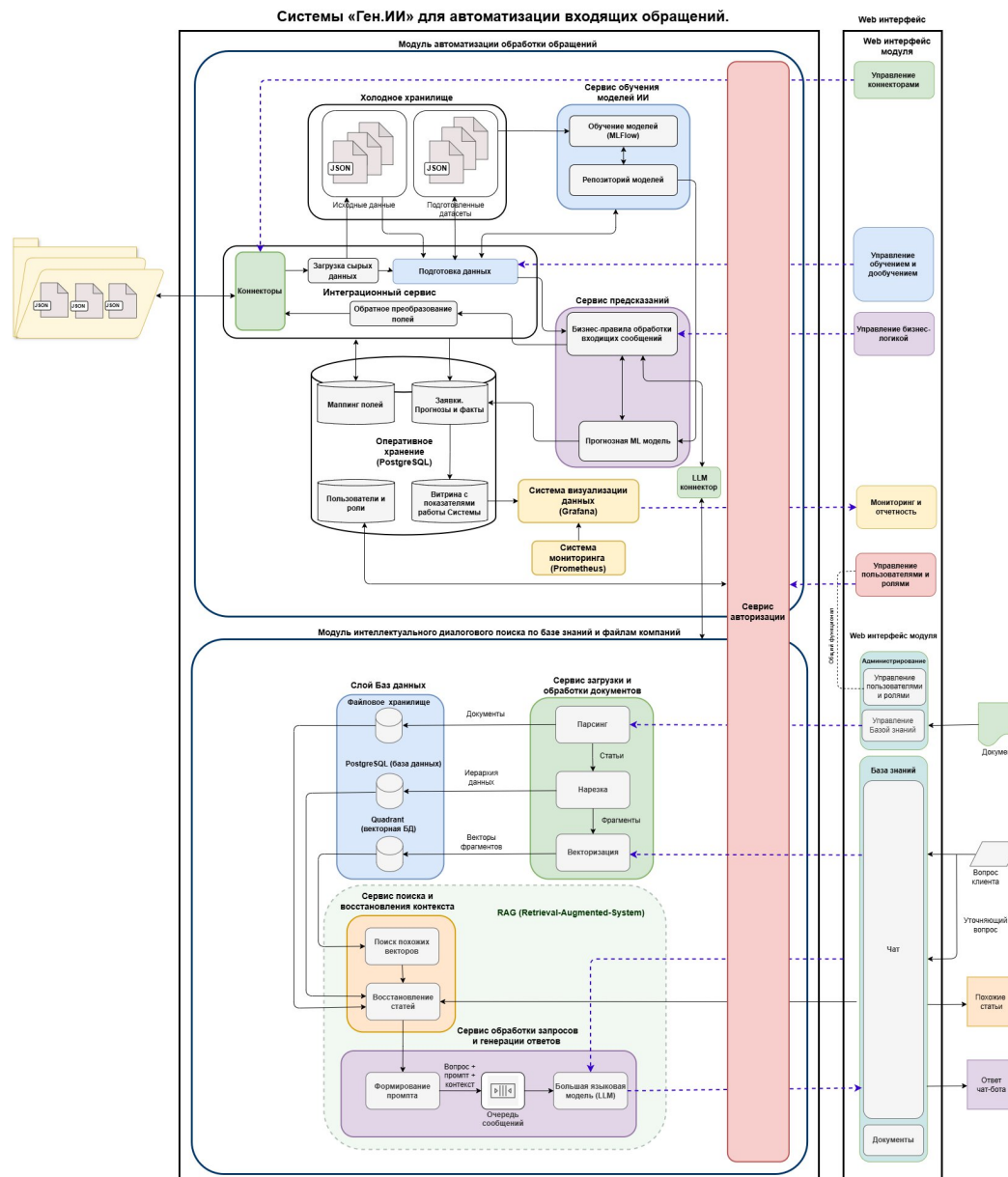
Функциональные возможности Модуля автоматизированной обработки обращений с применением технологий машинного обучения

- Интеграция с внешними ITSM-системами для приема и хранения новых обращений, не имеющих назначенных исполнителей.
- Автоматическое извлечение ключевых параметров обращения, необходимых для последующей обработки.

- Классификация и анализ поступающих заявок на основании алгоритмов машинного обучения для определения наиболее подходящего исполнителя из заданного списка.
- Формирование структурированного ответа о предполагаемом исполнителе с возможностью экспорта в формат, совместимый с используемой ITSM-системой.
- Подключение к хранилищу данных для автоматической загрузки информации об уже обработанных заявках.
- Обработка и сохранение сведений о фактически выполненных обращениях для последующего анализа.
- Обеспечение механизма повторного обучения моделей на основе накопленных данных об исполненных заявках с целью повышения качества прогнозов.
- Поддержка настройки логики назначения исполнителей с использованием заранее определённых правил и условий.
- Ведение мониторинга состояния модуля с предоставлением аналитической отчетности, включающей сравнение прогнозных и фактических результатов назначения.
- Возможность масштабируемой настройки и адаптации под особенности конкретного бизнес-процесса или домена.
- Интеграция с модулем интеллектуального поиска по базе знаний для расширения контекста заявок и повышения точности предсказаний.

1.3. Состав системы

Основные компоненты Системы представлены на рисунке 1.



1.3.1. Модуль интеллектуального диалогового поиска по базе знаний

Основными компонентами Модуля интеллектуального диалогового поиска по базе знаний являются:

Интерфейс управления (Web) - веб-интерфейс доступа к управлению Системой. Состоит из следующих разделов:

- **Управление базой знаний** - позволяет добавлять и удалять информацию в базе знаний: документы, статьи и фрагменты.
- **Пользователи** - используется для управления учетными записями пользователей, их правами доступа и ролями.

- **База знаний** - компонент, обеспечивающий взаимодействие пользователя с системой и доступ к содержимому:
- **Чат** - пользовательский интерфейс, предназначенный для взаимодействия с интеллектуальной системой обработки запросов. Через него пользователи задают вопросы и получают ответы.
- **Документы** - позволяет просматривать список загруженных документов.

Система (основной блок) - совокупность сервисов, выполняющих основную логику обработки данных и генерации ответов. Включает следующие компоненты:

Сервис авторизации - отвечает за проверку подлинности пользователей, обращающихся к Системе. Обеспечивает безопасность и контроль доступа.

Сервис загрузки и обработки документов - сервис, отвечающий за предварительную обработку загружаемых документов. Включает:

- **Парсинг** - извлечение текста и структуры из документов (PDF, DOCX, TXT).
- **Нарезка** - разделение извлеченного текста на статьи и более мелкие фрагменты (абзацы, предложения).
- **Векторизация** - преобразование текстовых фрагментов в числовые векторы (эмбединги), что необходимо для выполнения семантического поиска.

Слой Баз данных - компонент, обеспечивающий хранение и доступ к данным Системы. Состоит из следующих элементов:

- **Файловое хранилище** - используется для хранения исходных документов, загружаемых пользователями.
- **PostgreSQL (реляционная БД)** - предназначена для хранения структурированной информации, включая метаданные документов и иерархию данных.
- **Qdrant (векторная БД)** - специализированная база данных, используемая для хранения и быстрого поиска векторов (эмбедингов).

Сервис поиска и восстановления контекста - модуль, выполняющий поиск информации, релевантной пользовательскому запросу. Включает:

- **Поиск похожих векторов** - осуществляет поиск наиболее близких векторов в Qdrant по вектору запроса.
- **Восстановление статей** - извлекает статьи и текстовые фрагменты, соответствующие найденным векторам, из базы данных PostgreSQL.

Сервис обработки запросов и генерации ответов - основной компонент, обеспечивающий семантическую обработку запросов:

- **Большая языковая модель (LLM)** - отвечает за понимание запроса пользователя и генерацию ответа.
- **Очередь сообщений** - используется для асинхронной обработки запросов, обеспечивая отказоустойчивость и масштабируемость.

- **Формирование промпта** - подготавливает входные данные (промпт) для LLM, объединяя вопрос пользователя, найденный контекст и, при необходимости, дополнительную информацию.

1.3.2 Модуль автоматизации обработки обращений на основе машинного обучения

Основными компонентами Модуля автоматизации обработки обращений на основе машинного обучения являются.

Интерфейс управления (Web) - веб-интерфейс, предоставляющий доступ к настройке, мониторингу и управлению компонентами модуля. Включает следующие разделы:

- **Управление коннекторами** - раздел для настройки соединений с внешними источниками и формирования первичных наборов данных.
- **Управление обучением и дообучением** - позволяет запускать обучение моделей, управлять их версиями и организовывать репозиторий.
- **Управление бизнес-логикой** - интерфейс для создания правил обработки заявок на основе прогнозов моделей и логических условий.
- **Мониторинг и отчетность** - предназначен для анализа производительности системы, визуализации ключевых метрик и контроля работы компонентов.
- **Управление пользователями и ролями** - раздел для создания учетных записей, настройки ролей и доступа. Использует функциональность общего сервиса авторизации.

Система (основной блок) - совокупность сервисов, выполняющих основную логику работы системы.

Сервис авторизации - общий компонент системы, обеспечивающий проверку подлинности пользователей и контроль доступа к функциональности обоих модулей: интеллектуального поиска и автоматизации обработки обращений.

Файловое хранилище заявок - компонент, содержащий исходные клиентские заявки, загружаемые из внешних систем. Эти файлы поступают в систему для последующей обработки и обучения моделей.

Интеграционный сервис - обеспечивает сквозную передачу данных между внешними хранилищами и внутренней архитектурой модуля. Включает следующие логические подсистемы:

- **Коннектор** - отвечает за подключение к файловому хранилищу с клиентскими заявками. Настройка подключения производится через раздел управления в веб-интерфейсе.
- **LLM-коннектор** - компонент, обеспечивающий интеграцию с Модулем интеллектуального диалогового поиска по базе знаний. Позволяет использовать внутренние документы компании и функциональность LLM-модуля для обогащения обработки обращений и усиления интеллектуального анализа.
- **Сервис загрузки сырых данных** - загружает данные из внешних источников в Холодное хранилище модуля. В процессе загрузки формируется таблица соответствия полей, которая сохраняется в Оперативном хранилище.

- **Сервис подготовки данных** - преобразует исходные данные из Холодного хранилища в формат, пригодный для обучения машинных моделей. Преобразование регулируется параметрами, заданными в разделе обучения через Интерфейс управления.
- **Сервис обратного преобразования данных** - отвечает за приведение результатов прогноза к формату, необходимому для корректной записи в исходные заявки.

Холодное хранилище - файловый раздел системы (на базе NTFS), предназначенный для длительного хранения как исходных, так и подготовленных данных. Структурирован на две части:

- **Исходные данные** - сырые заявки, поступающие из внешних систем, сохраняются в формате JSON.
- **Подготовленные датасеты** - наборы данных, созданные для обучения и дообучения моделей. Для каждого запроса на обучение формируется отдельный датасет, соответствующий требованиям модели.

Сервис обучения моделей ИИ - сервис, реализующий процессы обучения и дообучения моделей машинного обучения на основе данных, полученных от заказчика. Состоит из двух компонентов:

- **Обучение моделей ИИ** - построен на базе платформы MLFlow с открытым исходным кодом и обеспечивает весь жизненный цикл моделей: от подготовки данных до вывода в продуктив.
- **Репозиторий моделей** - хранилище всех обученных моделей и связанных с ними метрик, доступное для повторного использования и мониторинга качества.

Сервис предсказаний - компонент, обрабатывающий входящие заявки в режиме реального времени и возвращающий прогнозные значения. Состоит из:

- **Бизнес-правила обработки входящих сообщений** - определяет логику обработки заявок в зависимости от их содержания и типа. Настраивается через интерфейс управления.
- **Прогнозная ML-модель** - обученная модель, развёрнутая в виде сервиса, принимает на вход заявку и возвращает прогноз по заданному параметру. Результаты передаются в бизнес-логику для дальнейшей обработки.

Оперативное хранилище - реляционная база данных PostgreSQL, используемая для хранения ключевых структурированных данных, необходимых для функционирования Модуля. Включает:

- **Маппинг полей** - таблицы преобразования структуры заявок в формат, совместимый с механизмами обучения. Используется как для прямого, так и для обратного преобразования.
- **Пользователи и роли** – хранение информации по учетным записям, их ролям и зашифрованным паролям. Используется общим сервисом авторизации.
- **Заявки. Прогнозы и факты** - содержит сведения о каждой заявке, включая предсказанные значения и фактические результаты, зафиксированные после завершения обработки.

- **Витрина с показателями работы Системы** - представляет собой агрегированные метрики по функционированию модуля, используемые в визуализации и мониторинге.

Система визуализации данных - модуль на базе Grafana, предназначенный для отображения аналитических показателей, состояния компонентов и качества работы ИИ-моделей. Интегрирован в интерфейс управления в разделе мониторинга.

2. Роли исполнителей

1. Функциональный заказчик

В обязанности функционального заказчика входят:

- формирование основных бизнес-требований к Системе;
- контроль соответствия результата заданию.

2. Руководитель проекта

В обязанности руководителя проекта входят:

- управление процессом разработки Системы;
- планирование и согласование сроков и ресурсов;
- контроль соответствия работ согласованному с функциональным заказчиком плану проекта;
- формирование и предоставление функциональному заказчику промежуточных и итоговых отчётов о ходе разработки Системы;
- принятие решения о выпуске релиза Системы.

3. Ведущий программист (архитектор)

В обязанности ведущего программиста входят:

- принятие ключевых решений по архитектуре Системы;
- подготовка состава задач на реализацию изменений в исходных кодах;
- определение требований к исполнителям;
- определение требований к технологиям производства и инструментам разработки;
- разработка и внесение изменений в проектную документацию.

4. Программист

В обязанности программиста входят:

- реализация функциональности подсистем или модулей Системы согласно заданию Ведущего программиста и в соответствии с планом, подготовленным руководителем проекта;
- первичный контроль качества и работоспособности разрабатываемых компонентов Системы;
- подготовка технического описания реализации функциональных возможностей разрабатываемых компонентов Системы;
- устранение ошибок ПО и исправление дефектов, обнаруженных в процессе тестирования;
- предоставление отчётов о состоянии выполнения заданий.

5. Инженер тестирования

В обязанности инженера тестирования входят:

- проверка работоспособности Системы в соответствии с техническими условиями;
- подготовка стендов для проверки работоспособности Системы в соответствии с набором сценариев использования;
- подготовка отчётов о составе и опасности обнаруженных в ПО дефектов;
- воспроизведение на стендах проблем, обнаруженных инженером технической поддержки;
- планирование сроков и ресурсов на выполнение тестирования ПО;
- предоставление отчётов о результатах тестирования Системы.

6. Инженер поддержки процесса и инфраструктуры разработки ПО

В обязанности инженера поддержки процесса и инфраструктуры разработки ПО входят:

- поддержка и развитие сборочной системы (сборка исполняемых модулей и библиотек, запуск различных утилит для подготовки дистрибутива продукта);
- развитие утилит контроля регламента разработки (качество кода, контроль переводов ресурсов, контроль информации о версии, контроль настроек проектов и т.п.);
- стандартизация окружения разработки;
- настройка инфраструктуры с учётом особенностей разрабатываемого ПО;
- поддержка разработки, основанной на отдельных ветках под каждую новую функцию продукта;
- создание шаблонов сборки для веток;
- выполнение регламента работы с итерационными ветками разработки.

7. Аналитик

В обязанности аналитика входят:

- формирование системных требований к продукту;
- контроль соответствия результата системным и бизнес-требованиям;
- разработка технической и пользовательской документации на Систему;
- документирование изменений, производимых в Системе;
- контроль документов проекта на соответствие принятому стандарту документирования;
- принятие участия в решении запросов, переданных во вторую линию поддержки.

8. Инженер технической поддержки

В обязанности инженера технической поддержки входят:

- приём и маршрутизация заявок от пользователей Системы;
- консультация пользователей по возникающим вопросам, связанным с Системой;

- управление критическими инцидентами и оповещение пользователей о статусе заявки;
- сопровождение и участие в обновлении установленного ПО;
- диагностика и устранение неисправностей.

3. Стадии жизненного цикла Системы

3.1. Перечень стадий жизненного цикла Системы

В процессе развития Системы используется спиральная модель жизненного цикла и релизный подход при обновлении. Этапы жизненного цикла Системы проходят несколько итераций. Результат каждой итерации представляет собой очередную версию программного продукта, которая постоянно совершенствуется.

Модель жизненного цикла Системы включает в себя следующие стадии:

- 1) стадия сбора и анализа требований;
- 2) стадия проектирования архитектуры и разработки;
- 3) стадия передачи;
- 4) стадия эксплуатации и сопровождения;
- 5) стадия изъятия и утилизации.

3.2. Стадия сбора и анализа требований

В рамках данной стадии осуществляется взаимодействие с функциональным заказчиком, направленное на выявление требований к Системе, потребностей в изменении существующей Системы, разработке дополнительных подсистем, компонентов.

На данной стадии применяются следующие процессы:

- процесс определения цели и назначения Системы;
- процесс сбора требований;
- процесс систематизации и анализа требований;

Роли исполнителей, действующих на данной стадии:

- функциональный заказчик;
- аналитик;
- руководитель проекта;
- ведущий программист.

Результатом данной стадии является техническое задание на проектирование и разработку Системы.

3.3. Стадия проектирования архитектуры и разработки

В рамках стадии проектирования определяются функциональные возможности Системы и ее место в программной архитектуре предприятия, определяется перечень компонентов Системы и необходимых доработок.

На данной стадии применяются следующие технические процессы:

- процесс проектирования архитектуры;
- процесс реализации;
- процесс комплексирования системы;
- процесс тестирования системы;
- процесс менеджмента документации.

Роли исполнителей, действующих на данной стадии:

- руководитель проекта;
- ведущий программист;
- программист;
- аналитик;
- инженер тестирования;
- инженер поддержки процесса и инфраструктуры разработки ПО.

К моменту старта реализации подготавливается среда разработки и определяется методология управления разработкой.

Комплексирование (сборка) приложений Системы осуществляется из исходного кода. Исходный код хранится в системе управления версиями. Для формирования сборки приложений Системы разработаны специализированные автоматизированные процедуры.

В результате работы инструментов сборки на выходе получаются:

- готовые к установке приложения/компоненты Системы;
- набор скриптов для модификации модели данных Системы.

На данной стадии осуществляется тестирование Системы в различных контурах:

- контур разработки – модульное тестирование;
- контур тестирования – интеграционное и системное тестирование;
- контур регрессионного тестирования – регрессионное тестирование;
- контур нагрузочного тестирования – нагрузочное тестирование.

Применяемые виды тестирования:

- 1) Модульное тестирование – изолированные испытания отдельных программных модулей Системы. Объектом испытания может служить отдельная функция, метод, процедура, модуль или программный объект. Если модуль взаимодействует с внешними системами, для тестирования применяются специализированные программные модули - «заглушки», имитирующие внешние системы.
- 2) Интеграционное тестирование – испытания, при которых программные модули Системы объединяются логически и тестируются как группа. Целью тестирования является выявление проблем взаимодействия отдельных компонентов Системы или взаимодействия Системы с внешними системами.

- 3) Регрессионное тестирование – испытания ранее протестированной Системы, позволяющие убедиться, что внесенные изменения не повлекли за собой появления дефектов в той части программы, которая не менялась.
- 4) Системное тестирование – испытания, направленные на подтверждение корректности поведения Системы в целом, подразумевающее сквозную проверку взаимодействия всех компонентов Системы и взаимодействия Системы с внешними системами.
- 5) Юзабилити-тестирование (проверка эргономичности) – исследование, выполняемое с целью определения, удобен ли пользовательский интерфейс Системы для его предполагаемого применения. Проверка эргономичности – метод оценки удобства продукта в использовании, основанный на привлечении пользователей в качестве испытателей и суммировании полученных от них выводов.

Решения о переводе Системы на следующую фазу разработки или следующую стадию жизненного цикла принимаются в зависимости от результатов тестирования в соответствующих контурах:

- тестирование в контуре разработки – решение о возможности установки сборки в тестовый контур;
- тестирование в контуре тестирования – решение о возможности включения изменения в сборку для регрессионного тестирования и нагрузочного тестирования;
- тестирование в контурах регрессионного и нагрузочного тестирования – решение о начале квалификационного тестирования Системы (стадия приёмки).

Стадия завершается готовностью релиза Системы к предварительным испытаниям, готовностью программной документации в следующем составе:

- 1) Программа и методика испытаний;
- 2) Руководство администратора;
- 3) Руководство менеджера.

Результатом стадии разработки также могут являться дополнительные требования, протокол выполненных работ и прочие документы, предназначенные для использования на следующих стадиях.

3.4. Стадия передачи

В рамках стадии выполняется подготовка экземпляра системы к опытно-промышленной эксплуатации, подготовка конечных пользователей.

Стадия может включать в себя процессы улучшения программного продукта.

В рамках данной стадии:

- передаётся на утверждение функциональному заказчику Программа и методика испытаний (ПМИ);
- производится установка и настройка Системы;
- проводится приёмка Системы, включающая:

1. Предварительные испытания:

В ходе предварительных испытаний проверяется работоспособность и соответствие техническому заданию, устраняются выявленные неисправности и недостатки. Испытания должны включать системное тестирование и нагрузочное тестирование. После проведения предварительных испытаний Система передается в опытную эксплуатацию.

2. Опытную эксплуатацию:

В ходе опытной эксплуатации выполняется работа с программным продуктом по назначению, собираются статистические данные о характеристиках и результатах функционирования, в том числе проводится юзабилити-тестирование. При необходимости осуществляется доработка программного продукта.

3. Приёмочные испытания:

Испытания проводятся после завершения опытной эксплуатации. По результатам приемочных испытаний принимается решение о выпуске релиза Системы (начале серийной эксплуатации Системы).

На данной стадии применяются следующие технические процессы:

- процесс установки и конфигурации Системы;
- подключение технических средств;
- установка прав доступа.

Роли исполнителей, действующих на данной стадии:

- функциональный заказчик;
- руководитель проекта;
- ведущий программист;
- аналитик;
- инженер тестирования;
- инженер поддержки процесса и инфраструктуры разработки ПО.

Результатом стадии является протокол об окончании опытно-промышленной эксплуатации и готовность к промышленной эксплуатации.

3.5. Стадия эксплуатации и сопровождения

Началом стадии применения Системы служит её установка и передача для применения по назначению в соответствии с моделью процесса поставки по ГОСТ Р ИСО/МЭК 12207:2010.

В рамках настоящего документа процесс поставки программного обеспечения для автоматизации процесса интеллектуального поиска и анализа информации на базе алгоритмов искусственного интеллекта «ГЕН.ИИ» не рассматривается.

Сопровождение Системы состоит в обеспечении техническим обслуживанием и сопровождением, и другими видами поддержки функционирования и использования Системы в соответствии с согласованными условиями поставки Системы приобретающей стороне.

В рамках данной стадии:

- осуществляется настройка Системы;
- выполняются консультации пользователей Системы;
- осуществляется контроль работы Системы, и анализ собранных данных об отклонениях и отказах;
- определяются новые требования к Системе и формирование решений, необходимых для реализации данных модификаций;
- производятся модификации Системы;
- по мере необходимости обновляется связанная с изменениями системная и программная документация;
- обновленные компоненты Системы помещаются в среду приобретающей стороны;
- сведения о модификации Системы доводятся до всех затронутых обновлениями сторон.

На данной стадии применяются следующие технические процессы:

- процесс функционирования программных средств.
- процесс сопровождения программных средств.

Роли исполнителей, действующих на данной стадии:

- приобретающая сторона;
- поставщик;
- аналитик;
- ведущий программист;
- программист;
- инженер технической поддержки.

Техническая поддержка пользователей осуществляется через Центр поддержки пользователей. Все обращения в Центр поддержки пользователей регистрируются и классифицируются по типам (Таблица 1).

№	Тип	Формат текста описания заявки
1	Инцидент	Обнаруженное несоответствие функционирования Системы установленным требованиям технического задания и/или эксплуатационной документации, включая случаи отображения сообщений об ошибках.
2	Запрос на изменение	Инициатива по усовершенствованию Системы, включающая предложения по расширению или изменению функциональности, направленные на повышение эффективности или удобства использования.
3	Консультация	Обращение пользователя с целью получения разъяснений по работе Системы, в том числе по вопросам, вызывающим затруднение или требующим дополнительного пояснения.

Устранение ошибок в работе Системы производится путём изменения настроек среды выполнения Системы или модификации программного кода Системы с выпуском новой сборки, которая передается приобретающей стороне. При необходимости корректируется документация на Систему.

На основании анализа обращений пользователей Системы, собранных данных о недостатках и отказах Системы функциональный заказчик может самостоятельно принять решение о модернизации Системы. Модернизация подразумевает развитие функциональных возможностей Системы, повышение удобства использования и администрирования. После модернизации производится выпуск нового релиза Системы. Для выполнения модернизации Системы инициируется переход на стадию проектирования архитектуры и разработки.

В случае заинтересованности в расширении функциональности Системы приобретающая сторона направляет поставщику новые требования к Системе. Требования анализируются поставщиком, далее согласовываются условия оказания услуг по доработке Системы, сроки и стоимость работ. Для выполнения доработки Системы инициируется переход на стадию проектирования архитектуры и разработки.

В случае принятия приобретающей стороной решения о завершении эксплуатации Системы инициируется переход на стадию изъятия и утилизации.

3.6. Стадия изъятия и утилизации

В соответствии с ГОСТ Р ИСО/МЭК 12207-2010 модель процесса прекращения применения программных средств предусматривает прекращение деятельности организации (обладателя исключительного права) по поддержке функционирования и сопровождения или деактивирует, демонтирует и удаляет поврежденные программные продукты, отправляя их в финальное состояние и возвращая окружающую среду в приемлемые условия.

Прекращение применения Системы как серийного программного продукта означает деактивацию и удаление Системы из среды конкретного покупателя (приобретающей стороны).

Деятельность организации (подразделения) по поддержке функционирования и сопровождению может быть прекращена по решению обладателя исключительного права на программное обеспечение для автоматизации процесса интеллектуального поиска и анализа информации на базе алгоритмов искусственного интеллекта «ГЕН.ИИ».

Причиной перевода Системы в данную стадию может служить замещение новой системой, катастрофический отказ, неэффективность дальнейшего применения.

На данной стадии применяются следующие технические процессы:

- процесс прекращения применения программных средств.

Роли исполнителей, действующих на данной стадии:

- приобретающая сторона;
- поставщик;
- функциональный заказчик;
- аналитик;
- инженер технической поддержки.

Все заинтересованные стороны оповещаются о планах и действиях по выводу Системы из эксплуатации.

Вся связанная документация по разработке, журналы и коды помещаются в архивы. Используемые данные или данные, связанные с прекращением применения Системы, должны быть доступны в соответствии с требованиями законодательства, локальных нормативных актов, соглашений и условий поставки.

4. Информация о персонале, необходимом для обеспечения поддержки жизненного цикла

Для успешного развертывания и последующего обслуживания Системы требуются технические специалисты с соответствующей квалификацией и практическим опытом в развертывании и системном/сетевом администрировании ИТ-систем, перечисленных ниже:

- оболочка Bash и система контейнеризации Docker,
- ОС семейства Windows и Linux,
- Системы управления доступом
- Web-сервера
- СУБД
- Системы управления ИТ-инфраструктурой

4.1. Роли участников Системы

Для **непосредственной работы** с Системой выделяются следующие роли:

Роль участника	Описание
инженер технической поддержки	Осуществляет установку приобретённого программного обеспечения, включая начальную настройку системы и конфигурацию прав доступа для конечных пользователей.
поставщик	Отвечает за поставку программного обеспечения, соответствующего заявленным функциональным характеристикам и условиям договора.

5. Поддержание жизненного цикла программного обеспечения

Поддержание жизненного цикла Платформы интеллектуальной обработки обращений, текста, документов, картинок, голоса без участия человека с применением технологий генеративного искусственного интеллекта для решения отраслевых задач «ГЕН.ИИ» обеспечивается за счет его сопровождения и проведения обновлений в соответствии с собственным планом доработки ПО и по заявкам Пользователей, восстановление данных и консультации по вопросам эксплуатации, установке и переустановки ПО.

Поддержание жизненного цикла программного обеспечения обеспечивается за счет следующих процессов:

- Расширение функционала ПО в соответствии с собственным планом доработок и/или на основе запросов пользователей;

- Устранение сбоев и технических проблем выявленных в процессе эксплуатации ПО;
- Внесение изменений в ПО с целью оптимизации его работы (улучшение быстродействия, повышение эффективности использования серверных ресурсов, повышение удобства пользовательского интерфейса и др.).

5.1. Контактная информация

Наименование организации: Общество с ограниченной ответственностью «ДжиДиСи Сервисез»

Адрес: Республика Татарстан, Лаишевский район, с. Усады, ул. Дорожная, д. 42, корпус 2, офис 835.

Телефон: 8-800-333-98-70

Электронная почта: pr@icl-services.com