БизнесАвтоматика

УТВЕРЖДАЮ
Генеральный директор
ООО «НПЦ «БизнесАвтоматика»

П.С. Петраков
«01» августа 2023 г.

БЕТА-ВЕРСИЯ АНАЛИТИЧЕСКОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ «ВИЗАРИ»

Детализированное техническое задание, включая функциональные требования, требования к структуре и функционированию Решения, требования к показателям надежности, требования к эргономике и технической эстетике, требования к спецификации и унификации, требования к лингвистическому обеспечению, требования к техническим средствам, утвержденное генеральным директором

Проект:
Доработка аналитической информационной системы «Визари» (АИС «Визари»)
(соглашение о предоставлении гранта от 24 декабря 2021 г. № 2021-550-44/)

Этап 3

Москва, 2023

Общество с ограниченной ответственностью «Научно-производственный центр «БизнесАвтоматика»
ОГРН 1107746225366; ИНН 7729652455; КПП 772901001
ОКВЭД 62.01; ОКАТО 45268586000; ОКПО 65342740; ОКФС 16; ОКОПФ 65
Р/с 40702810638180002588; ПАО Сбербанк
БИК 044525225, К/с 30101810400000000225
Адрес 115114, г. Москва, 1-й Дербеневский переулок, д. 5, стр. 2, оф. 401
Тел./факс +7 (800) 100 93 69, +7 (495) 221 29 65
E-mail info@npc.bs
Веб-сайт npc.bs

Содержание

Термины и сокращения....4
1 Общие сведения....7
1.1 О документе....7
1.2 Полное наименование....7
1.3 Перечень документов, на основании которых ведется доработка Решения....7
1.4 Сроки выполнения работы....7
2 Назначение и цели доработки Решения....8
2.1 Назначение бета-версии Решения....8
2.2 Цели создания бета-версии Решения....8
3 Характеристика объекта автоматизации....10
4 Требования к доработке Решения....11
4.1 Требования к бета-версии Решения в целом....11
4.1.1 Требования к структуре и функционированию бета-версии Решения....11
4.1.2 Требования к численности и квалификации персонала бета-версии Решения и режиму его работы....18
4.1.3 Требования к показателям назначения....19
4.1.4 Требования безопасности....19
4.1.5 Требования к эргономике и технической эстетике....20
4.1.6 Требования к показателям надежности....22
4.1.7 Требования по стандартизации и унификации....24
4.1.8 Требования к защите информации....24
4.2 Требования к формированию детализированных функциональных требований к доработке Решения....26
4.3 Функциональные требования к компонентам (модулям) бета-версии Решения 26
4.3.1 Функциональные требования к компоненту нормативно-справочной информации 26
4.3.2 Функциональные требования к компоненту (модулю) аналитической обработки данных....27
4.3.3 Функциональные требования к компоненту (модулю) визуализации аналитической отчетности....40
4.3.4 Функциональные требования к интеграционной шине....54
4.3.5 Функциональные требования к компоненту (модулю) предобработки информации ETL 57
4.3.6 Функциональные требования к компоненту (модулю) автоматизированного проектирования предметной области....59
4.3.7 Функциональные требования к компоненту (модулю) защищенного распределенного хранения данных с встроенными редакторами документов....61
4.3.8 Функциональные требования к компоненту (модулю) управления регламентированными процессами....63
4.3.9 Функциональные требования к компоненту (модулю) управления задачами....66
4.3.10 Функциональные требования к компоненту (модулю) уведомлений....72
4.3.11 Функциональные требования к компоненту (модулю) поиска....72
4.3.12 Функциональные требования к компоненту (модулю) безопасности....74
4.3.13 Функциональные требования к компоненту (модулю) администрирования....75
4.4 Требования к видам обеспечения доработанной бета-версии Решения....76
4.4.1 Требования к математическому обеспечению....76
4.4.2 Требования к информационному обеспечению....76
4.4.3 Требования к программному обеспечению....77
4.4.4 Требования к лингвистическому обеспечению....77
4.4.5 Требования к техническим средствам....78
5 Состав и содержание работ по доработке Решения....79
5.1 Требования к детализированной модели монетизации Решения....79
5.2 Требования расчетным алгоритмам, используемым для Решения....79
5.3 Порядок разработки и этапы работ по доработке Решения....79

5.4 Требования к организациям-соисполнителям работ 82
6 Порядок контроля и приемки результатов работ по доработки Решения 83
6.1 Виды, состав и методы испытаний доработанного Решения и ее составных частей 83
6.1.1 Требования к видам испытаний доработанного Решения 83
7 Требования к документированию.....84
7.1 Перечень документов, предъявляемых по окончании соответствующих этапов работ 84
8 Требования к регистрации ПО бета-версии Решения 85
9 Источники доработки.....86
10 Требования по коммерциализации бета-версии Решения.....87

Термины и сокращения

В настоящем документе применены следующие термины с соответствующими определениями, представленные в таблице 1:

Таблица 1 – Термины и определения

Термин	Определение
1	**2**
Исполнитель	ООО «Научно-производственный центр «БизнесАвтоматика», НПЦ «БизнесАвтоматика»
Проект	Комплекс мероприятий, направленных на разработку (доработку) Решения
Решение	Аналитическая информационная Бета-версия Решения «Визари» (АИС «Визари»)
Визари	Программное решение, разработанное специалистами НПЦ «БизнесАвтоматика, зарегистрированное в Едином реестре российских программ для электронных машин и баз данных (№6653 от 23.04.2020 г.).
BigData	(от англ. Big Data) – обозначение структурированных и неструктурированных данных значительного объема, обработка и анализ которых требует использования современных программных решений
Deep learning	(от англ. Deep Leaning) – совокупность методов машинного обучения, основанных на обучении представлением, а не специализированных алгоритмах под конкретные задачи
WorkFlow	(от англ. Workflow) – графическое представление потока задач в рамках процессов, подпроцессов работ

В настоящем документе применены следующие сокращения с соответствующими определениями, представленные в таблице 2:

Таблица 2 – Сокращения и определения

Сокращение	Определение
1	**2**
АИС	Автоматизированная информационная система
БП	Бизнес-процесс
ВПСНС	Визуальное проектирование сверточных нейронных сетей
ЕСИА	Единая система идентификации и аутентификации
ИИ	Искусственный интеллект
ИМ	Информационная модель
ИНС	Искусственная нейронная сеть
ИТ	Информационные технологии
НДС	Налог на добавленную стоимость
НСИ	Нормативно-справочная информация
ОС	Операционная система
ПИ РНС	Программное изделие разработки нейронных сетей
ПИ СОД	Программное изделие синтезирования обучающих данных
ПО	Программное обеспечение
СМЭВ	Система межведомственного электронного взаимодействия
СОА	Сервисно-ориентированная архитекутра
СОВРСД	Синтез обучающей выборки и разметки синтезированных данных

Сокращение	Определение
1	**2**
СПО	Специальное программное обеспечение
СУБД	Система управления базой данных
ЭВМ	Электронная вычислительная машина
AI	(от англ. Artificial Intelligence) – комплекс технологий, на базе которой происходит разработка интеллектуальных компьютерных систем
API	(от англ. Application Programming Interface) – набор способов и правил, по которым информационные системы производят обмен данными
BI	(от англ. Business Intelligence) – программные решения, основная функциональная задача которых заключается в обработке и визуализации результатов анализа различной информации
BPM	(от англ. Business Process Management) – управление бизнес-процессами, концепция процессного управления организацией, рассматривающая бизнес-процессы как особые ресурсы предприятия
CSV	(от англ. Comma-Separated Values) – текстовый формат, предназначенный для представления табличных данных. Строка таблицы соответствует строке текста, которая содержит одно или несколько полей, разделенных запятыми, или точками с запятой
CSS	(от англ. Cascading Style Sheets) – каскадные таблицы стилей, формальный язык описания внешнего вида веб-страницы, написанного с использованием языка разметки
ETL	(от англ. Extract, Transform, Load) – инструмент управления данными, обеспечивающий возможность извлечения, преобразования, очистки и загрузки данных в хранилище данных
DL	(от англ. Deep Learning) — это вид машинного обучения с использованием многослойных нейронных сетей, которые самообучаются на большом наборе данных.
HTML	(от англ. HyperText Markup Language) – язык гипертекствой разметки, стандартизированный язык гипертекстовой разметки документов для просмотра веб-страниц в браузере
HTTP	(от англ. HyperText Transfer Protocol) – протокол прикладного уровня передачи данных
JSON	(от англ. JavaScript Object Notation) – текстовый формат обмена данными
KPI	(от англ. Key Performance Indicator) – ключевой показатель эффективности, числовой показатель деятельности, позволяющий измерить степень достижения целей и оптимальности процесса
ML	(от англ. Machine Learning) – метод анализа данных, поддерживающий процесс обучения программного аналитического продукта в ходе обработки и анализа данных для решения новых задач.
MVC	(от англ. Model-View-Controller) – схема разделения данных приложения, и управляющей логики на три отдельных блока: модель, представление и контроллер
ODBC	(от англ. Open Database Connectivity) – программный интерфейс (API) доступа к базам данных

Сокращение	Определение
1	**2**
REST	(от англ. Representational State Transfer) – передача репрезентативного состояния, архитектурный стиль взаимодействия распределенного приложения в сети
SOAP	(от англ. Simple Object Access Protocol) – простой протокол доступа к объектам, протокол обмена структурированными сообщениями в распределенной вычислительной сети
SQL	(от англ. Structured Query Language) – язык структурированных запросов, декларативный язык программирования, применяемый для создания, модификации и управления данными реляционной базы данных
SSL	(от англ. Secure Sockets Layer) – уровень защищённых сокетов) – криптографический протокол, который подразумевает более безопасную связь
XML	(от англ. eXtensible Markup Language) – расширяемый язык разметки, предназначенный для хранения и передачи данных
XSLT	(от англ. (Extensible Stylesheet Language Transformations) – язык преобразования XML-документов

1 Общие сведения

1.1 О документе

Документ содержит детализированные функциональные и нефункциональные требования к бета-версии аналитической информационной системы «Визари» (АИС «Визари») (далее также – Решение).

1.2 Полное наименование

Аналитическая информационная система «Визари».

1.3 Перечень документов, на основании которых ведется доработка Решения

- Соглашение о предоставлении гранта № 2021-550-44/_____ от 24.12.2021;
- Техническое задание на доработку аналитической информационной системы «Визари» (АИС «Визари»);
- Описание альфа-версии/прототипа программного обеспечения аналитической системы «Визари» (АИС «Визари»);
- Функциональные требования и спецификация программного обеспечения для бета-версии программного обеспечения аналитической информационной системы «Визари» (АИС «Визари»).

1.4 Сроки выполнения работы

Начало работ: 01.03.2022.

Окончание работ: 31.08.2023.

2 Назначение и цели доработки Решения

2.1 Назначение бета-версии Решения

Бета-версия/прототип Решения предназначена для автоматизации деятельности различных заказчиков, в том числе органов государственной власти, государственных корпораций, коммерческих организаций.

2.2 Цели создания бета-версии Решения

Альфа-версия аналитической информационной системы «Визари» – это высокоэффективная клиент-серверная платформа, позволяющая автоматизировать любые организационные и технологические процессы организации.

Приложения, построенные на платформе «Визари», создаются по модульному принципу, что позволяет добавлять требуемый функционал с помощью выбора и настройки готовых модулей и подсистем. Итоговый набор подсистем и модулей может быть индивидуально подобран под особенности инфраструктуры предприятия любого масштаба. Доработка Решения до бета-версии позволит расширить основной функционал модулей с целью оптимизации работы.

Альфа-версия аналитической информационной системы «Визари» позволяет значительно расширить возможности организаций в части анализа собственных показателей функционирования, определить ключевые векторы развития на основании выявленных по итогам анализа данных тенденций рынка, а также оптимизировать процессы стратегического и операционного управления за счет обеспечения перманентного доступа к ключевым показателям, составляющим необходимую аналитическую базу для принятия управленческих решений.

АИС «Визари» по многомерному анализу и визуализации больших данных должна быть востребована среди федеральных и региональных органов государственной власти, государственных корпораций, крупных и средних коммерческих организаций. Потребителями программного обеспечения также могут выступать международные и российские вендоры, рассматривающие Решение в качестве обособленного инструмента для решения локальных задач, интегрируемого в собственные программные решения. Потенциальными пользователями бета-версии Решения также могут выступать крупные интеграторы, оказывающие услуги по поставке, настройке и внедрению бета-версии Решения для нужд различных заказчиков.

Регистрация бета-версии Решения в Едином реестре российских программ для электронных вычислительных машин и баз данных сделает программное решение особенно востребованным среди государственных заказчиков, в том числе государственных корпораций, которые обязаны соблюдать требования Постановления Правительства Российской Федерации от 16.11.2015 № 1236 «Об установлении запрета на допуск программного обеспечения, происходящего из иностранных государств, для целей осуществления закупок для обеспечения государственных и муниципальных нужд».

Доработка Решения до бета-версии выполняется с целью создания современного, функционального, конкурентоспособного как на российском, так и на мировом рынке программного решения, предусматривающего применение технологий искусственного интеллекта, предоставляющего широкий спектр возможностей по визуальному представлению данных, а также характеризующегося интуитивно понятным и удобным

пользовательским интерфейсом, позволяющим использовать Решение без привлечения специалистов по данным или ИТ-специалистов.

Цели создания бета-версии Решения:

- внедрение простого и понятного решения, не требующего значительных затрат на обучение сотрудников и внедрение;
- предоставление пользователям функционального «коробочного» (готового к эксплуатации без значительного объема доработок) решения, построенного на базе технологичной и гибкой системы;
- расширение функциональных возможностей, предоставляемых пользователям, за счет более эффективного применения механизмов искусственного интеллекта;
- расширение аналитических возможностей информационной системы за счет использования механизмов машинного обучения (ML), в том числе методов глубокого обучения (Deep learning), базирующихся на каскаде нейронных сетей;
- преодоление «геополитических» (формальные или неформальные) ограничений в возможностях использования программных продуктов от вендоров из определенных стран.

Объединение самостоятельных компонентов (модулей) Решения в рамках единой бета-версии Решения обеспечит широкую востребованность на отечественном и международном рынке. Бета-версии Решения позволит заказчикам с минимальными затратами осуществлять переход от широко растиражированных иностранных BI-продуктов (QlikView и QlikSense, Microsoft Power BI, Tableau, SAP Analytics Cloud, Oracle Analytics Cloud), а также составит конкуренцию наиболее распространенным отечественным BI системам (платформа «Loginom», Форсайт. Аналитическая платформа, Платформа «Visiology», Информационно-аналитическая платформа Криста BI, Платформа «Триафлай», Платформа «N3.Аналитика»).

3 Характеристика объекта автоматизации

Объектом автоматизации выступает деятельность российских и иностранных органов государственной власти, корпораций и коммерческих компании по планированию, организации и контролю процессов сбора, многомерного анализа, обработки информации, формирования визуальных форм аналитической отчетности, согласования отчетов, а также их структурированного хранения.

4 Требования к доработке Решения

4.1 Требования к бета-версии Решения в целом

4.1.1 Требования к структуре и функционированию бета-версии Решения

4.1.1.1 Бета-версия Решения должна быть ориентирована на централизованное обслуживание и обеспечение удаленного доступа к данным и возможность полнофункционального использования бета-версии Решения через веб-браузер.

4.1.1.2 Для обеспечения возможностей по масштабируемости и изменению конфигурации бета-версии Решения, а также безопасности и отказоустойчивости при доработке компонентов должна быть использована трехзвенная архитектура, состоящая из клиента, сервера приложений и сервера баз данных (рисунок 1).

Рисунок 1 – Трехзвенная клиент-серверная архитектура

4.1.1.3 Логическая структура бета-версии Решения должна содержать три уровня: хранения данных, бизнес-логики и взаимодействия с пользователями (презентационный).

- Презентационный уровень должен представлять данные пользователю, а также обеспечивать возможность ввода и изменения данных при наличии у пользователя соответствующих прав.
- Уровень бизнес-логики должен содержать программные объекты и программный код, которые реализуют логику работы задач. Компоненты должны выполняться на сервере и разделять общие ресурсы сервера.
- Уровень хранения данных должен обеспечивать долговременное эффективное хранение данных. Уровень хранения должен включать СУБД и компоненты для доступа к данным. Уровень хранения должен предоставлять программный интерфейс для объектов уровня бизнес-логики.

4.1.1.4 Бета-версия Решения должна предусматривать возможности дальнейшей модификации функциональных модулей на основе разработанных решений.

Должен быть использован открытый (open-source) стек-технологии: PostgreSQL, C#.

4.1.1.5 Должна быть предусмотрена возможность масштабирования бета-версии Решения при увеличении нагрузки. При увеличении нагрузки должно обеспечиваться горизонтальное масштабирование системы – увеличение производительности приложения за счёт распределения нагрузки между оборудованием.

4.1.1.6 Бета-версия Решения должна быть спроектирована по модульному принципу (иметь микросервисную архитектуру) для исключения дублирования функциональности и обеспечения возможности обновления компонентов без негативного воздействия друг на друга.

4.1.1.7 Бета-версия Решения должна соответствовать следующим принципам:

- Высокая степень автоматизации – бета-версия Решения должна использовать методы анализа с использованием искусственного интеллекта, для автоматизации типовых этапов технических и организационных процессов.

- Открытость – бета-версия Решения должна использовать общедоступные и специфицированные решения, протоколы и интерфейсы; открытость бета-версии Решения должна позволять выполнить проверку исходных кодов на отсутствие ошибок; архитектура бета-версии Решения не должна ограничивать возможности по разработке новых функций и выбора Исполнителя, а также накладывать ограничения на масштабируемость инфраструктуры потенциального заказчика; архитектура бета-версии Решения не должна привязывать ее к использованию конкретных проприетарных продуктов.
- Модульность – бета-версия Решения должна быть построена с использованием модульной (микросервисной) архитектуры, состоящей из отдельных компонентов, обеспечивающих возможность их независимой модификации; сбой в работе одного из модулей или одного из компонентов не должен приводить к полному прекращению функционирования бета-версия Решения в целом.
- Масштабируемость – архитектура бета-версии Решения должна позволять увеличивать производительность, объемы хранимой и обрабатываемой информации при добавлении ресурсов за счёт распределения нагрузки между оборудованием.
- Управляемость и конфигурируемость – должно обеспечиваться управление элементами бета-версии Решения на всех уровнях ее архитектуры: на уровне инфраструктуры, на функциональном уровне, на уровне представления данных.
- Индивидуализация – должен быть обеспечен удобный индивидуальный доступ к Системе для всех потенциальных групп пользователей с функциональностью, соответствующей их задачам.
- Функциональное соответствие – архитектура бета-версии Решения должна отвечать функциональным задачам бета-версии Решения.
- Системность – все взаимосвязанные элементы бета-версии Решения должны использовать единую методологию и отвечать единым принципам взаимодействия, надежности и управления.
- Единство графического представления – при проектировании и разработке пользовательских интерфейсов должны использоваться общие принципы графического представления информации и организации доступа пользователей к функциональным возможностям и сервисам компонентов.
- Управление содержанием и представлением информации – обеспечение эффективных методов создания, изменения, сохранения и удаления информационного содержания и структуры бета-версии Решения, механизмов управления процессом создания и публикации документов бета-версии Решения; обеспечение средств для создания и изменения представления информации в бета-версии Решения; способность создания и контроля информационного наполнения бета-версии Решения персоналом, не обладающим техническими знаниями, при помощи дружественного и интуитивно понятного интерфейса.
- Навигация и поиск – должны быть обеспечены средства навигации внутри бета-версии Решения, поиска информации в бета-версии Решения с использованием различных типов запросов, а также средства представления результатов поиска в удобной и изменяемой форме.

– Анализ использования информации – обеспечение механизмов эффективного сбора необходимой информации и анализа использования ресурсов бета-версии Решения, данных о доступе к ресурсам бета-версии Решения (например, тип страницы или время суток).

4.1.1.8 Бета-версия Решения должна иметь следующую микросервисную архитектуру:

– Компонент (модуль) нормативно-справочной информации;
– Компонент (модуль) аналитической обработки данных;
– Компонент (модуль) визуализации аналитической отчетности;
– Интеграционная шина;
– Компонент (модуль) предобработки информации ETL;
– Компонент (модуль) автоматизированного проектирования предметной области;
– Компонент (модуль) защищенного распределенного хранения данных с встроенными редакторами документов;
– Компонент (модуль) управления регламентированными процессами;
– Компонент (модуль) управления задачами;
– Компонент (модуль) уведомлений;
– Компонент (модуль) поиска;
– Компонент (модуль) безопасности;
– Компонент (модуль) администрирования.

Таким образом в рамках проекта планируется обеспечить доработку компонентов (модулей) на основе представленных выше подходах для объединения их в базовую версию (бета-версия) Решения (рисунок 2).

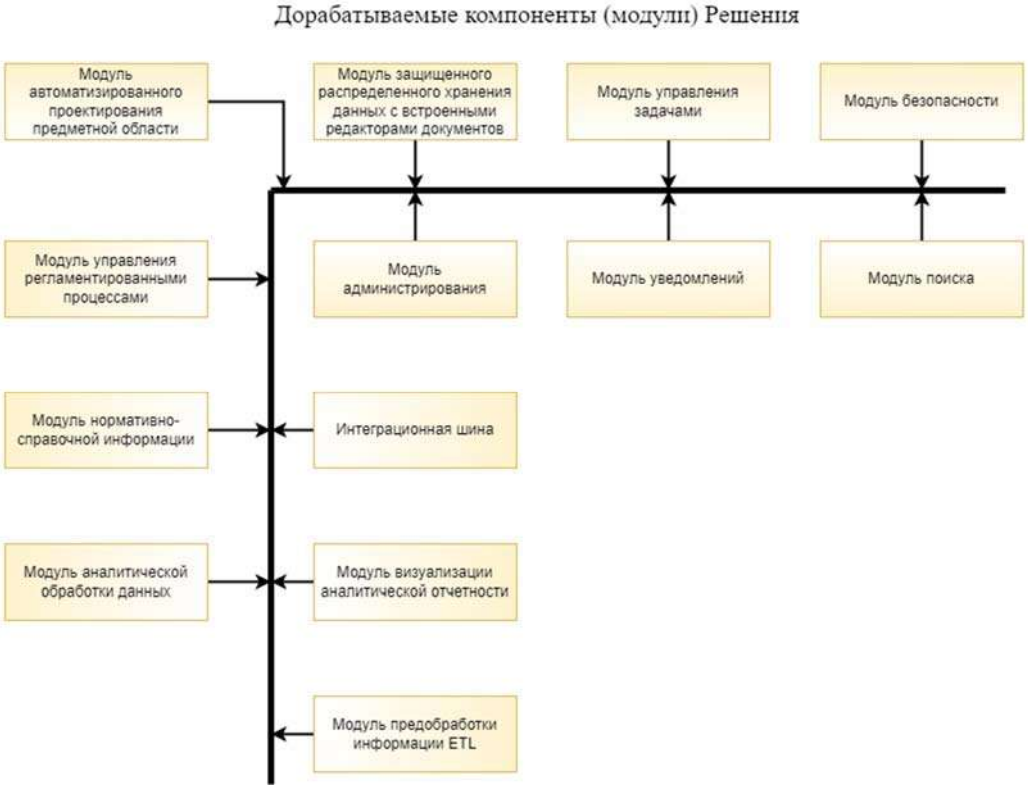

Рисунок 2 – Дорабатываемые компоненты (модули) Решения

4.1.1.9 Все компоненты Решения должны быть спроектированы по шаблону MVC (Model-View-Controller – «Модель-Представление-Контроллер») – схема разделения данных приложения, и управляющей логики на три отдельных блока: модель, представление и контроллер (рисунок 3). Такое разделение позволяет модифицировать каждый уровень независимо.

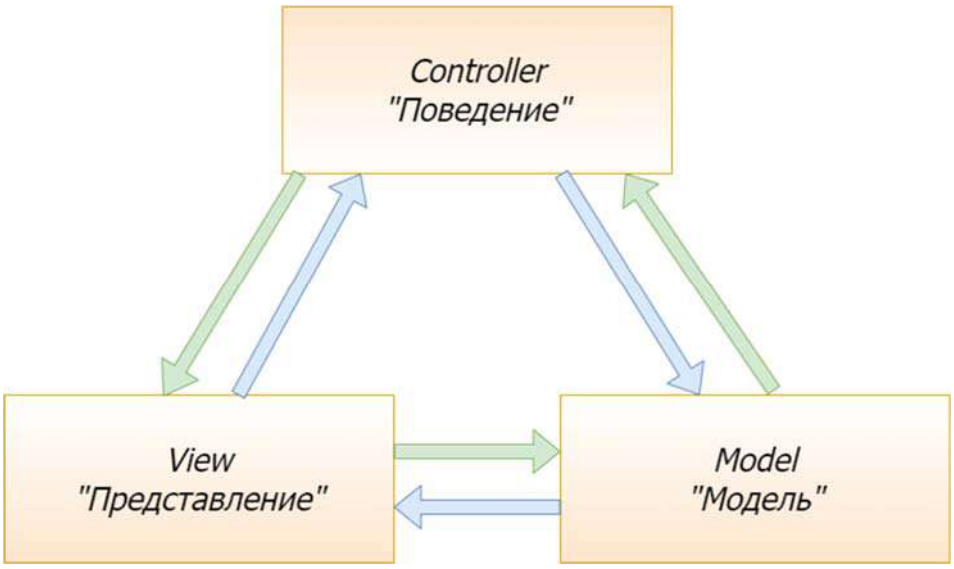

Рисунок 3 – Шаблон проектирования «Модель-Представление-Контроллер»

Блоки в шаблоне MVC выполняют следующие функции:

- «Модель» – реализует бизнес-логику приложения и выполняет обработку данных;
- «Контроллер» – обрабатывает запросы пользователя и выполняет запросы к другим компонентам;
- «Представление» – осуществляет визуализацию полученного результата или упаковывает полученный результат в необходимый формат.

Преимущества MVC:

- сокращение временных и материальных затрат на доработку и отладку сложных приложений корпоративного уровня;
- сокращение временных и материальных затрат на внедрение новых технологий (любой из уровней может быть обновлен без переработки остальных).

4.1.1.10 Платформа «Визари» построена на сервисно-ориентированной архитектуре (СОА).

За счёт высокой эффективности масштабирования СОА большая часть сервисов «Визари» собраны в программные высокопроизводительные и отказоустойчивые кластеры, готовые для использования в высоконагруженных информационных системах, например, программный кластер сервисов СУБД, сервисов файлового хранилища, сервисов анализа данных, интеграционных сервисов, геоинофрмационных сервисов, коммуникационных сервисов и т.д.

СОА обладает следующими преимуществами:

- изолированность: при разворачивании сервисов получаются изолированные функциональные блоки, которые не влияют на работоспособность друг друга;

- производительность: ядро с сильным связыванием позволяет взаимодействовать микросервисам через кэш центрального процессора или оперативную память, минуя накладные сетевые расходы;
- доступность: отсутствие обязательного требования к функционированию всех сервисов системы в любой момент времени;
- гибкость и гетерогенность возможность реализации модулей системы на разных технологиях;
- независимость при развертывании: благодаря слабой связанности сервисов достигается простота разворачивания системы и минимизируется вероятность её отказа, достигается отказоустойчивость и надежность системы в целом;
- масштабируемость: возможность распределения сервисов между разными виртуальными или физическими вычислительными узлами.

4.1.1.11 Сервисно-ориентированная архитектура типового решения, построенного на платформе «Визари» включает следующий список сервисов:

- веб-сервис – сервис для публикации данных для внешних пользователей. Для работы используется сервер nginx с использованием технологии тонкого клиента (web). Может включать в себя сервис для взаимодействия с мобильными приложениями;
- сервисы Веб-приложений, включающие в себя:
  - сервис для обработки информации и взаимодействия между приложениями;
  - сервис авторизации пользователей и администраторов (identity);
  - сервис для хранения пользовательских данных (profile);
  - сервис для взаимодействия со сторонними сервисами (web-api);
  - базовые и прикладные подсистемы и модули платформы «Визари», например:
    - сервис для взаимодействия между пользователями (ikr);
    - геоинформационный сервис;
    - аналитический сервис;
    - сервис поиска;
    - и т.д.
- Сервис Баз Данных позволяет использовать несколько видов баз данных для оптимального доступа к различным типам данных. Например, для хранения и обработки данных пользователей используется реляционная СУБД PostgreSQL, а для данных основной сценарий работы с которыми предполагает частый поиск – NoSQL СУБД Redis и Elasticsearch (хранение, индексация, кеширование).
- Сервис хранения данных – сервис для предоставления доступа к виртуальному хранилищу данных (filestorage).
- Сервис очередей – сервис, который обеспечивает возможность взаимодействия между компонентами и сервисами (rabbitmq).
- Сервис Резервных копий – служебный сервис для периодического резервного копирования компонентов и данных системы. Резервируются файлы, базы данных, индексы Elasticsearch.

- Сервис Мониторинга – служебный сервис обеспечивающий мониторинг сервисов платформы «Визари». В состав сервиса входят компоненты — Elasticsearch, logstash, kibana, а также сервисы предоставления данных (beats).
- Служебные сервисы – служебные сервисы, включающие в себя сервис DNS, NTP, систему централизованного управления (ansible).

4.1.1.12 При повышении нагрузки должны создаваться несколько экземпляров сервисов, которые объединяются в кластеры и данные начинают распределяться между отдельными экземплярами внутри кластера. Пример таких кластеров – кластер Elasticsearch, кластер PostgreSQL.

4.1.1.13 Требования к способам и средствам связи между компонентами бета-версии Решения

Информационный обмен между серверными компонентами бета-версии Решения и клиентскими приложениями должен осуществляться по сети Интернет, посредством протокола HTTP.

Взаимосвязь между компонентами бета-версии Решения должна быть организована на уровне БД при помощи механизма внешних ключей, либо на уровне программного кода.

Взаимодействие подсистем должно обеспечить консистентность данных по одним и тем же объектам вне зависимости от способа их представления.

4.1.1.14 Требования к режимам функционирования бета-версии Решения

Бета-версия Решения должна поддерживать следующие режимы функционирования:

- штатный режим работы;
- режим технического обслуживания (обновления отдельных компонентов программно-технического комплекса);
- режим администрирования;
- режим аварийного завершения работы;
- режим восстановления работы после аварии;
- режим работы с частичной потерей функциональности.

При условии регулярного регламентного обслуживания и мониторинга параметров работы бета-версии Решения должна обеспечить длительно-непрерывное, круглосуточное функционирование в штатном режиме и в режиме системного администрирования.

Основным режимом функционирования должен являться штатный режим, при котором бета-версия Решения должна поддерживать выполнение всех заявленных функций. В этом режиме должна быть обеспечена работа всех зарегистрированных пользователей в круглосуточном режиме.

Режим технического обслуживания предназначен для проведения запланированных работ по обслуживанию бета-версии Решения и может сопровождаться частичной недоступностью функциональности.

Обновления отдельных компонентов бета-версии Решения должны проводиться в фоновом режиме без прекращения работы пользователей.

В режиме администрирования бета-версия Решения должна обеспечивать возможность проведения следующих работ:

- настройка среды функционирования;
- пополнение программных средств новыми компонентами;
- мониторинг, контроль и диагностирование работоспособности;

- копирование и архивирование баз данных;
- копирование и архивирование прикладных программных компонент;
- импорт и экспорт данных для обмена с внешними системами.

Должна быть предоставлена возможность настройки определенного временного периода для автоматического сохранения внесенных в Систему изменений.

Система переходит в аварийный режим при возникновении нештатной ситуации и невозможности штатной работы. Режим аварийного завершения производится с предварительным автоматическим уведомлением пользователей и предложением завершить работу с Системой. Аварийный режим функционирования характеризуется отказом одного или нескольких компонентов бета-версии Решения. В этом режиме должны приниматься меры к обеспечению временной работоспособности бета-версии Решения (возможно с ограничением в выполнении отдельных функций), после чего проводится анализ причин выхода бета-версии Решения в аварийный режим и проводятся мероприятия по восстановлению полной работоспособности.

Режим восстановления работы после аварии выполняется в автоматическом режиме с переключением на резервную копию Системы в случае ее наличия.

Режим работы с частичной потерей функциональности в основном предназначен для:

- проведения реконфигурирования;
- профилактического обслуживания;
- регламентированного обновления и информационного взаимодействия с источниками данных, не предполагающих непрерывную актуализацию.

4.1.1.15 Требования по диагностированию бета-версии Решения

Бета-версия Решения должна быть снабжена средствами диагностики работоспособности. В случае выявления проблем, сведения о них должны сообщаться пользователю и помещаться в протокол сбоев и ошибок.

Бета-версия Решения должна гарантировать отсутствие потерь и искажений информации.

Диагностика работоспособности программного обеспечения, реализующего интерфейсы пользователей, должна осуществляться стандартными средствами клиентских операционных систем, управляющих функционированием рабочих станций.

Диагностирование бета-версии Решения должно осуществляется путем анализа записей в системных журналах СУБД, веб-сервера и операционной системы, а также с помощью встроенных средств диагностирования общего программного обеспечения Системы.

В бета-версии Решения диагностированию подлежат:

- отклонения от заданных параметров быстродействия Решения;
- случаи аварийных остановок и самопроизвольной перезагрузки программного обеспечения;
- случаи нарушений целостности баз данных и файловой системы;
- сбои при выполнении регламентных операций резервного копирования;
- пользовательские действия и запросы;
- доступность технологических интерфейсов интеграции Решения с внешними информационными системами;

- функционирование модулей (компонентов) Решения;
- осуществление резервного копирования и обеспечения безопасности.

4.1.1.16 Перспективы развития, модернизации бета-версии Решения

Бета-версия Решения должна предусматривать возможности дальнейшей модификации функционала, работ и сервисов на основе разработанных решений. Должна быть предусмотрена возможность масштабирования бета-версии Решения при увеличении нагрузки. Бета-версия Решения должна быть спроектирована по модульному принципу для исключения дублирования функционала, а также возможности обновления модулей без негативного воздействия друг на друга.

4.1.2 Требования к численности и квалификации персонала бета-версии Решения и режиму его работы

Численность персонала определяется количеством пользователей бета-версии Решения.

Пользователи должны обладать навыками работы на компьютере.

Администраторы бета-версии Решения должны обладать навыками администрирования информационных систем.

Администраторы модулей (компонентов) бета-версии Решения должны обеспечивать функционирование модулей (компонентов), сохранность данных, их квалификация должна позволять:

- ориентироваться в стандартных возможностях используемых операционных систем и СУБД, протоколах передачи данных;
- владеть средствами мониторинга СУБД;
- владеть средствами резервного копирования и восстановления данных;
- владеть средствами защиты информации;
- работать с архиваторами, дисковыми утилитами, антивирусными программами;
- определять источник сбоя функционирования ПО и грамотно описывать его.

Режим работы персонала должен соответствовать режиму работы предприятия.

Список типовых ролей бета-версии Решения представлен в виде матрицы доступа в таблице 3.

Таблица 3 – Матрица доступа

Роль/группа	**Уровень доступа к данным**	**Разрешенные действия**
Администратор бета-версии Решения	Обладает полной информацией о системном и прикладном программном обеспечении бета-версии Решения. Обладает полной информацией о технических средствах и конфигурации бета-версии Решения. Имеет доступ ко всем техническим средствам обработки информации и данным бета-версии Решения. Имеет доступ к средствам защиты информации и протоколирования и к части	Добавление пользователей. Редактирование личных данных пользователя. Удаление пользователей. Настройка резервного копирования. Рассылка массовых уведомлений пользователям. Включение /отключение функции управления рабочим столом. Включение /отключение автоматической рассылки уведомлений. Настройка портов почтового севера.

Роль/группа	Уровень доступа к данным	Разрешенные действия
	ключевых элементов бета-версии Решения. Обладает правами конфигурирования и административной настройки технических средств бета-версии Решения	Настройка параметров под бета-версию Решения безопасности (в том числе параметров поиска, доступа к разделам бета-версии Решения). Настройка правил интеграции через подсистему интеграции (API). Создание новых и корректировка существующих объектов бета-версии Решения посредством проектирования предметной области. Создание новых и настройка существующих схем регламентированных процессов
Оператор	Обладает ограниченным набором прав, включающим создание, просмотр и редактирование объектов бета-версии Решения в разделах, разрешенных администратором (в соответствии с утвержденной ролевой политикой)	Просмотр информации об объектах. Создание новых экземпляров объектов. Редактирование информации об объектах
Авторизованный пользователь	Обладает правами просмотра, создания контента в разделах, разрешенных администратором.	Создание и просмотр информации об объектах.

4.1.3 Требования к показателям назначения

Время отклика при запросе к бета-версии Решения не должно превышать 5 секунд.

Система должна функционировать 24 часа в сутки 7 дней в неделю за исключением технологических перерывов на обслуживание. Бета-версия Решения должна обеспечивать одновременную работу не менее 50 000 пользователей. Должна быть предусмотрена возможность масштабирования бета-версии Решения при увеличении нагрузки.

4.1.4 Требования безопасности

4.1.4.1 Требования к условиям окружающей среды

Требования к условиям окружающей среды не предъявляются.

4.1.4.2 Требования к кондиционированию

Для обеспечения необходимых климатических условий серверные помещения, в которых располагаются технические средства бета-версии Решения, должны иметь систему кондиционирования.

Используемые системы кондиционирования должны обеспечивать возможность оповещения администраторов системы при невозможности поддержания заданных параметров окружающей среды.

4.1.4.3 Требования по электропитанию и заземлению

4.1.4.3.1 Для защиты от кратковременных перебоев электропитания все средства должны подключаться через источники бесперебойного питания достаточной мощности.

4.1.4.3.2 Защитное заземление должно обеспечивать защиту людей от поражения электрическим током при прикосновении к металлическим нетоковедущим частям, которые доступные для прикосновения человека и могут оказаться под напряжением в результате повреждения изоляции.

Защитное заземление должно выполняться преднамеренным электрическим соединением металлических частей электроустановок с «землей» или ее эквивалентом.

Заземляющие устройства должны выполняться по нормам на напряжение прикосновения или по нормам на сопротивление.

4.1.4.4 Требования к противопожарной безопасности

Требования к противопожарной безопасности не предъявляются.

4.1.5 Требования к эргономике и технической эстетике

4.1.5.1 Дизайн и оформление страниц бета-версии Решения должны предусматривать использование следующих элементов оформления:

- четкие шрифты;
- общепринятые элементы навигации и иконки;
- графические объекты (рисунки, фото).

4.1.5.2 Навигационные средства бета-версии Решения должны соответствовать следующим требованиям:

- пользователям должна предоставляться наглядная информация о структуре бета-версии Решения;
- пункты меню должны называться или изображаться так, чтобы пользователь однозначно понимал их назначение;
- размер шрифта, используемый в интерфейсах, количество элементов на странице, их цвета, размеры и расположение не должно вызывать дискомфорт у пользователей бета-версии Решения.

4.1.5.3 Взаимодействие пользователей бета-версии Решения с прикладным программным обеспечением должно осуществляться посредством визуального графического интерфейса.

Ввод-вывод данных бета-версии Решения, прием управляющих команд и отображение результатов их исполнения должны выполняться в интерактивном режиме.

4.1.5.4 Интерфейс не должен быть перегружен графическими элементами. Навигационные и управляющие элементы интерфейса должны быть выполнены в удобной для пользователя форме с соблюдением следующих условий:

- однозначность интерпретации наименований (наименование элемента должно позволять однозначно определить его назначение);
- унификация наименований (однотипные элементы должны иметь одинаковые наименования);
- унификация обозначений (однотипные элементы должны иметь одинаковые обозначения – графические значки, вид элементов управления);
- унификация использования (однотипные элементы должны иметь одинаковую реакцию на действия пользователя – наведение указателя, переключение фокуса, активации элемента управления).

Интерфейс должен быть рассчитан на преимущественное использование манипулятора типа «мышь», то есть управление должно осуществляться с помощью набора экранных меню, кнопок, значков и т.п. элементов.

Клавиатурный режим ввода должен использоваться главным образом при заполнении и/или редактировании текстовых и числовых полей экранных форм.

Все надписи экранных форм, а также сообщения, выдаваемые пользователю (кроме системных сообщений), должны быть на русском языке.

4.1.5.5 Бета-версия Решения должна обеспечивать корректную обработку аварийных ситуаций, вызванных неверными действиями пользователей, неверным форматом или недопустимыми значениями входных данных. В указанных случаях бета-версия Решения должна выдавать пользователю соответствующие сообщения, после чего возвращаться в рабочее состояние, предшествовавшее неверной (недопустимой) команде или некорректному вводу данных.

4.1.5.6 Экранные формы бета-версии Решения должны проектироваться с учетом требований унификации:

- для обозначения сходных операций должны использоваться сходные графические значки, кнопки и другие управляющие (навигационные) элементы. Термины, используемые для обозначения типовых операций, а также последовательности действий пользователя при их выполнении, должны быть унифицированы;
- внешнее поведение сходных элементов интерфейса должны реализовываться одинаково для однотипных элементов.

4.1.5.7 Должна быть обеспечена возможность масштабирования экранных форм.

4.1.5.8 Интуитивно понятный пользовательский интерфейс должен обеспечивать минимизацию усилий пользователей при работе, что позволит:

- сократить длительность операций чтения и поиска информации;
- уменьшить время навигации и выбора команды;
- увеличить длительность устойчивой работы пользователей.

4.1.5.9 Цветовое решение интерфейса должно быть выдержано в спокойных тонах, не вызывающих утомление зрения.

4.1.5.10 Разработанный дизайн экранных форм должен соответствовать современным направлениям и тенденциям в веб-дизайне, а именно:

- адаптивность – адаптивный дизайн, который обеспечивает правильное отображение системы на различных устройствах, и динамически подстраивающийся под заданные размеры окна;
- простота – интуитивный, минималистический интерфейс;
- свободное пространство – принцип свободного пространства помогает подчеркнуть контент;
- плоский дизайн – принцип плоского дизайна представляет собой четкие визуальные решения, объекты изображаются схематично без передачи объёма;
- блоки – блочное расположение контента позволяет его адаптировать под размер окна браузера;
- большие изображения – использование больших изображений является современным трендом;

– большие кнопки – для удобства навигации используются крупные элементы (кнопки).

Для выполнения ввода типизированных значений по возможности должны использоваться компоненты, предназначенные для ввода значений указанного типа, с помощью которых пользователь может быстро вводить данные. Бета-версия Решения должна поддерживать ввод данных как и вручную, так и с помощью данных компонентов.

4.1.6 Требования к показателям надежности

4.1.6.1 В нормальном режиме бета-версии Решения должна обеспечить круглосуточную и непрерывную работу при условии проведения требуемых технических мероприятий по обслуживанию бета-версии Решения. При возникновении сбоя в программном обеспечении бета-версии Решения после восстановления данные остаются консистентными.

4.1.6.2 Бета-версия Решения должна обеспечивать корректную обработку аварийных ситуаций, в том числе вызванных неверными действиями пользователей. Пользовательский интерфейс бета-версии Решения должен отображать соответствующие сообщения об ошибках для пользователя, после чего Решение должно вернуться в рабочее состояние, предшествовавшее некорректному вводу данных.

4.1.6.3 Бета-версия Решения должна автоматически восстановить свою функциональность, в том числе в случае некорректного перезапуска программных и аппаратных средств.

4.1.6.4 При сбоях в системе электроснабжения технических средств бета-версии Решения, приводящих к перезагрузке операционной системы, восстановление программных средств должно происходить автоматически после перезапуска операционной системы и запуска исполняемого файла бета-версии Решения.

В программном обеспечении бета-версии Решения должно быть предусмотрено:

– контроль целостности данных на уровне СУБД;
– сохранение целостности данных в базе данных при нештатном завершении работы бета-версии Решения;
– автоматизированное восстановление бета-версии Решения при корректном перезапуске технических средств;
– сохранение работоспособности программного обеспечения при некорректных действиях пользователя (ввод неверных по размеру и/или составу данных, нештатное завершение действий, выход из бета-версии Решения до завершения действия);
– резервное копирование базы данных;
– журналирование операций бета-версии Решения

4.1.6.5 Показатели надежности должны определяться прогнозируемой частотой возникновения аварийных ситуаций. Для системы регламентируются показатели надежности для следующих видов аварийных ситуаций:

– общесистемный отказ – выражается в недоступности всех или большинства пользовательских интерфейсов системы вне зависимости от причин, вызвавших этот отказ (отказы средств технического обеспечения, телекоммуникационных средств, общесистемного программного обеспечения, неверная работа специализированных программ, ошибки персонала, сбои электропитания и т.п.), кроме причин катастрофического характера (форс-мажорных обстоятельств);

- частный сбой – выражается в недоступности одного из компонентов, который вызывает недоступность одного или нескольких интерфейсов какого-либо компонента системы, или его некорректной работе. Время восстановления системы: до четырех часов, вызванных ошибками ПО, без учета восстановления дампов из резервных копий.

4.1.6.6 Требования по интенсивности отказов и длительности перерыва действительны при условии надлежащей численности и квалификации обслуживающего персонала, соблюдении ими технологических инструкций, а также при использовании установленных видов технического обеспечения, общесистемного и специализированного программного обеспечения.

4.1.6.7 Требования к надежности бета-версии Решения уточняются по согласованию с заказчиком определенного проекта, реализуемого на базе бета-версии Решения на стадии технического проектирования. При этом технические решения для удовлетворения указанных требований должны быть обоснованы на стадии разработки документации по проекту и отражены в соответствующей документации.

4.1.6.8 Требования к надежности технических средств и программного обеспечения

Надежность технических средств бета-версии Решения должна обеспечиваться за счет резервирования оборудования.

Показатели надежности должны достигаться комплексом организационно-технических мер, обеспечивающих доступность ресурсов, их управляемость и обслуживаемость.

Технические меры по обеспечению надежности должны предусматривать:

- использование средств безопасности вычислительной сети и защиты информации;
- конфигурирование используемых средств с применением программного обеспечения, обеспечивающего высокую доступность;
- строгую регламентацию доступа к функциям подсистем.

Необходимо предусмотреть возможность обеспечения следующих технических мер повышения надежности:

- резервирование критически важных компонентов и данных и отсутствие возможности единовременного отказа дублирующих компонентов;
- организацию многоуровневой системы дублирования и архивирования информации;
- использование технических средств с избыточными компонентами и возможностью их «горячей» замены;
- использование системы комплексного администрирования.

Организационные меры по обеспечению надежности должны быть направлены на минимизацию ошибок персонала службы эксплуатации за счет:

- достаточной квалификации пользователей и обслуживающего персонала;
- регламентации и нормативного обеспечения выполнения работ персоналом;
- своевременного оповещения пользователей и обслуживающего персонала о случаях внештатной работы подсистем;

- минимизации ошибок персонала службы эксплуатации и при проведении работ по обслуживанию комплекса технических средств подсистем;
- ведения журнала отказов;
- минимизации времени ремонта или замены вышедших из строя компонентов за счет:
  - регламентации проведения работ и процедур по обслуживанию и восстановлению подсистем;
  - своевременной диагностики неисправностей;
  - обслуживания и поддержки компонентов комплекса технических средств.

4.1.7 Требования по стандартизации и унификации

4.1.7.1 Должно быть обеспечено соответствие бета-версии Решения требованиям следующих стандартов:

- HTTP 1.1 для обмена данными между уровнем бизнес-логики и презентационным уровнем;
- HTTPS для защищенного обмена данными между уровнем бизнес-логики и презентационным уровнем (при необходимости, в том числе в административном интерфейсе);
- ANSI SQL/92 для обмена данными между уровнем бизнес-логики и уровнем данных.

Унификация программных средств должна быть обеспечена за счет применения унифицированных компонент и средств из состава:

- общего и базового программного обеспечения;
- модуля «защищенное хранилище»;
- модуля «интеграционная шина».

4.1.7.2 При создании бета-версии Решения следует руководствоваться действующими в Российской Федерации национальными стандартами и другими нормативно-техническими документами.

Проводимые работы должны соответствовать следующим требованиям:

- применяемые при создании бета-версии Решения технические (форматы данных, протоколы передачи и прочие) и организационные (регламенты, требования, инструкции и т.п.) решения должны быть доступны и документированы в виде, достаточном для независимой реализации третьими сторонами. Применение недокументированных или недоступных решений не допускается;
- при выборе применяемых решений преимущество должно отдаваться решениям, основанным на стандартизированных технологиях, т.е. прошедшим процедуру стандартизации и утвержденным в качестве стандарта либо рекомендации каким-либо признанным международным, федеральным, отраслевым, промышленным органом по стандартизации;
- экранные формы пользовательского интерфейса должны проектироваться с учетом требований унификации.

4.1.8 Требования к защите информации

Доступ пользователей к персонализированным функциям и данным бета-версии Решения должен предоставляться только после прохождения пользователем процедур аутентификации и авторизации.

Доступ пользователей к функциям и данным бета-версии Решения должен быть ограничен на основе ролевого принципа. Каждому пользователю бета-версии Решения должна быть сопоставлена учетная запись, ассоциированная с одной из нескольких предопределенных пользовательских ролей. Для каждой пользовательской роли должны быть определены конкретные ограничения на доступ к функциям и данным бета-версии Решения. Доступ к функциям чтения, записи, удаления, создания и навигации по объектам должен настраиваться исключительно пользователем, имеющим соответствующие права.

В целях обеспечения защиты бета-версии Решения от несанкционированного доступа к административному интерфейсу и предотвращения размещения несоответствующей его назначению информации должна быть обеспечена возможность защищенного электронного обмена данными.

Параметры бета-версии Решения должны проверяться на соответствие типам, размерам, значениям. Проверка параметров должна осуществляться до их передачи серверным модулям. Проверка параметров на соответствие не должна осуществляться только на клиентской стороне. Проверка должна осуществляться на разрешение только допустимых значений, а не по принципу исключения запрещенных значений. При доступе к закрытым областям проверка должна осуществляться при любой попытке доступа к любому адресу закрытой области. Недопустимо возникновение ситуации, при которой существует возможность получить доступ к объекту закрытой области, минуя проверку.

Конфигурация программного обеспечения должна исключать возможность несанкционированного доступа к бета-версии Решения или отказа в обслуживании при максимальном использовании встроенных механизмов защиты информации.

Программное обеспечение не должно допускать возможности использования его для косвенного доступа к закрытым областям или данным. Не должно быть возможности получения содержимого закрытого раздела путем вызова открытых функций бета-версии Решения с указанием адреса закрытого источника.

Комплекс мер по обеспечению сохранности информации и ее восстановления должен включать в себя:

- проведение регулярного регламентного копирования программного обеспечения и баз данных бета-версии Решения в целом и ее подсистем;
- восстановление данных в непротиворечивое состояние при программно-аппаратных сбоях (отключение электрического питания, сбоях операционной системы и других) вычислительно-операционной среды функционирования;
- восстановление данных в непротиворечивое состояние при сбоях в работе сетевого программного и аппаратного обеспечения;
- проведение внепланового резервного копирования программного обеспечения и баз данных;
- хранение резервных копий на съемных носителях в специально оборудованных помещениях, обеспечивающих условия, исключающие их несанкционированное получение, хищение, подмену или уничтожение;
- восстановление программного обеспечения и баз данных из резервных копий.

4.2 Требования к формированию детализированных функциональных требований к доработке Решения

Должны быть формализованы требования к используемому в компонентах бета-версии Решения математическому аппарату. По итогам данных работ должна быть составлена Алгоритмическая документация, содержащая описание используемых в бета-версии Решении математических методов.

Должна быть разработана презентация концепции реализации Проекта.

4.3 Функциональные требования к компонентам (модулям) бета-версии Решения

4.3.1 Функциональные требования к компоненту нормативно-справочной информации

4.3.1.1 При реализации проекта должна быть полностью сохранена функциональность компонента НСИ альфа-версии Решения.

4.3.1.2 Компонент должен выполнять следующие функции:

- создание группы справочников (классификаторов), в том числе линейных, иерархических, составных;
- изменение группы справочников (классификаторов);
- удаление группы справочников (классификаторов). При удалении группы все входящие справочники (классификаторы) должны переводиться в общую группу;
- обмен данными между справочниками и другими подсистемами (карточки, документы, редакции документов, прочая служебная информация);
- управление справочниками (классификаторами) с возможностью добавления/удаления атрибутов;
- сортировка объектов в справочнике по названию и реквизитам (атрибутам);
- поиск объектов в справочнике по названию и реквизитам (атрибутам);
- добавление в справочники и классификаторы данных из нормативно-правовой информации, требований стандартов, технических условий.

4.3.1.3 Конструктор (программный модуль) справочников и классификаторов должен позволять добавлять новые записи (типы и виды объектов), создавать многоуровневые иерархические структуры реестров, справочников и классификаторов.

4.3.1.4 Должна быть реализована возможность удалять записи по одной или по несколько, выбрав их с помощью соответствующей команды.

4.3.1.5 При просмотре удаленных записей должна быть возможность восстановить конкретную запись.

4.3.1.6 Функция поиска должна позволять выполнять поиск по ключевым словам.

В режиме быстрого поиска пользователь должен иметь возможность задать искомую комбинацию символов. По умолчанию поиск введенной комбинации символов должен осуществляться по всем материалам, открытым для поиска пользователю с учетом связанности информации.

В режиме расширенного поиска система должна позволять пользователю сохранить настроенное логическое выражение в шаблон с целью последующего повторного использования. Расширенный поиск должен иметь возможность включения поиска по любым атрибутам объектов в любой их комбинации, указываемой в виде логических

выражений с применением таких операторов объединения как «И» и «ИЛИ», а также скобок, указывающих очередность и приоритетность применения операторов.

В отношении шаблона поиска должна быть реализована возможность осуществлять следующие операции:

- сохранить;
- изменить;
- удалить;
- применить.

Должна быть реализована возможность осуществлять каскадный поиск (поиск в результатах поиска).

4.3.1.7 При отображении данных справочников должны быть обеспечены следующие возможности:

- сортировка по любому из атрибутов в справочнике;
- установка сортировки по умолчанию;
- фильтр по любому из атрибутов в справочнике;
- выбор из перечня преднастроенных и сохранение собственных фильтров;
- возможность выгружать данные справочника (частично и/или полностью) в форматах xls, xlsx;
- индивидуальная настройка параметров отображения для каждого пользователя.

4.3.1.8 Требуется провести интеграцию компонента в архитектуру бета-версии Решения.

За счет использования микросервисной архитектуры доработка компонента при интеграции в архитектуру бета-версии Решения не требуется.

4.3.2 Функциональные требования к компоненту (модулю) аналитической обработки данных

4.3.2.1 При доработке компонента аналитической обработки данных требуется обеспечить поддержку функциональности компонента аналитической обработки данных альфа-версии Решения.

4.3.2.2 Компонент должен позволять формировать единую базу аналитической информации и рекомендаций.

4.3.2.3 Компонент должен поддерживать обработку массива обращений и входящих запросов, обеспечивать мониторинг тематик и оценку тональности обращений.

4.3.2.4 В рамках реализации проекта должна быть осуществлена интеграция компонента аналитической обработки данных с иными компонентами бета-версии Решения.

4.3.2.5 Интеграция с компонентом визуализации аналитической отчетности должна обеспечить поддержку передачи следующих данных:

- статистики по выделенным сущностям и атрибутам документов;
- поисковой выдачи по автоматически выполненным запросам (мониторинг тематик);
- данных об исправлении пользователем значений;
- поисковые логи.

4.3.2.6 Интеграция компонента аналитической обработки данных с компонентом поиска должна обеспечить механизмы:

- сужения результатов выдачи: релевантность и фильтрацию в соответствии с атрибутами документов, а также в соответствии с выделенными сущностями;
- обработки поискового запроса: учет морфологических форм слова, синонимов распространенных терминов, исправление опечаток;
- обращения к источникам данных: политики обхода различных источников настраиваемы, доступен как обход по расписанию, так и по запросу администратора;
- индексации: поддерживается инкрементальная индексация данных;
- автоматического выполнения запросов: настраивается автоматическое выполнение запросов по расписанию с сохранением поисковой выдачи;
- хранения логов (сохранение логов пользовательских запросов).

4.3.2.7 В ходе реализации Проекта в бета-версии Решения должна быть обеспечена возможность применения механизма автоматического дифференцирования для проведения обучения моделей ИИ. Внедрение ИИ может с высокой точностью производить распознавание документации и конвертации изображений в структурированный набор данных, что необходимо для решения различных поисковых и аналитических задач.

Должен быть реализован отдельный микросервис по распознаванию документации и конвертации изображений в текст, который должен обеспечивать ввод данных, их обработку посредством библиотеки оптического распознавания символов.

4.3.2.8 Должен быть разработан функциональный блок визуального проектирования сверточных нейронных сетей (ВПСНС). Назначение блока ВПСНС:

- визуализация архитектуры, загруженной нейросетевой модели в графическом представлении;
- работа с основными типами слоев (функциональных блоков) нейронных сетей, представленных в таблице 4.

Таблица 4 – Основные типы поддерживаемых слоев, настраиваемые параметры и их наименование в графе

№, п/п	**Название слоя (функционального блока)**	**Обозначение в графе сети**	**Настраиваемые параметры**	**Примечание**
1	**2**	**3**	**4**	**5**
1	Слой ввода изображений	Data	1. Размер входного изображения (channels, width, height) 2. Тип нормализации данных	Слой ввода размеров двухмерных изображений, подаваемых на нейронную сеть. Сочетает в себе совокупность работы слоев image_input и feuter_input.
2	Двухмерный сверточный слой	Conv2d	1. Количество сверточных фильтров 2. Размер сверточных фильтров 3. Размер шага свертки	Основной слой сверточных нейронных сетей. Служит для получения карты признаков из входных данных, содержащей в себе их ключевые особенности. Может

№, п/п	Название слоя (функционального блока)	Обозначение в графе сети	Настраиваемые параметры	Примечание
1	2	3	4	5
			4. Коэффициент расширения окна свертки (dilatation) 5. Величина расширения границ входных данных (padding) 6. Вид функции инициализации начального состояния весов в слое, в том числе поддержка пользовательских функций 7. Вид функции инициализации начального состояния смещений в слое, в том числе поддержка пользовательских функций	уменьшать пространственные размерности карт входных данных. Соответствует слою conv_2d.
3	Транспонированный двухмерный сверточный слой	ConvTranspose 2d	1. Количество фильтров 2. Размер фильтров 3. Размер шага транспонированной свертки 4. Величина уменьшения размера входных данный (обратный padding) 5. Коэффициент расширения окна транспонированной свертки (dilatation) 6. Вид функции инициализации начального состояния весов в слое, в том числе поддержка пользовательских функций 7. Вид функции инициализации начального	Слой, реализующий обратную свертку, применяется для увеличения пространственных размерностей входных данных. Соответствует слою trans_conv_2d.

№, п/п	Название слоя (функционального блока)	Обозначение в графе сети	Настраиваемые параметры	Примечание
1	2	3	4	5
			состояния смещений в слое, в том числе поддержка пользовательских функций	
4	Линейный слой	Linear	Число нейронов в слое	Слой, реализующий преобразование входных данных в выходные с использованием связи «все со всеми». Соответствует слою full_conv.
5	Функция активации – ReLU	ReLU	–	Выполняет пороговую операцию для каждого элемента входных данных, где любое значение меньше нуля устанавливается равным нулю. Соответствует слою relu.
6	Функция активации-LeakyReLU	LeakyReLU	Скалярный множитель для отрицательных входных значений	Выполняет пороговую операцию для каждого элемента входных данных, где любое значение меньше нуля умножается на выбранный скалярный множитель. Соответствует слою lk_relu.
7	Функция активации-гиперболический тангенс	Tanh	–	Применяет функцию гипербалического тангенса ко входным данным. Соответствует слою tanh
8	Пакетная нормализация	BatchNorm2d	Накопление математического ожидания и дисперсии по всем изображениям	Нормализует пакет входных данных для каждого канала независимо. Соответствует слою batch_norm.
9	Пакетная нормализация для каждого экземпляра пакета	InstanceNorm 2d	Накопление математического ожидания и дисперсии по всем изображениям	Нормализует пакет входных данных независимо для каждого канала и изображения в нем. Соответствует слою instance_norm.
10	Слой вероятного зануления входных данных	Dropout	Часть случайного «отключения» входных элементов (в	Случайно обнуляет заданную часть входных элементов (при

№, п/п	Название слоя (функционального блока)	Обозначение в графе сети	Настраиваемые параметры	Примечание
1	2	3	4	5
			интервале от 0 до 0,95)	обучении). Соответствует слою dropout
11	Слой субдискретизации	Pool2d	1.Тип агрегации (max, avg) 2.Размер ядра субдискретизации 3.Размер шага субдискретизации объединения 4.Величина расширения границ входных данных (padding) 5.Коэффициент расширения транспонированной свертки (dilatation)	Выполняет понижающую дискретизацию, разделяя входные данные на прямоугольные области объединения и вычисляет средние или максимальные значения для каждой области. Реализует работу слоев average_pooling и max_pooling.
12	Слой адаптивной субдискретизации	AdaptivePool 2d	1.Тип агрегации (max, avg) 2.Пространственные размерности выхода	Выполняет понижающую дискретизацию, разделяя входные данные на прямоугольные области объединения и вычисляет средние или максимальные значения для каждой области, для получения выходных данных с заданными пространственными размерностями. Реализует работу слоев gloabal_average _pooling и gloabal_max_pooling
13	Слой суммирования	Add	–	Выполняет суммирование данных из двух входов. Соответствует слою add
14	Слой вычитания	Sub	–	Выполняет вычитание данных из двух входов
15	Слой умножения	Mul	–	Выполняет умножение данных из двух входов. Соответствует слою multi.
16	Слой взятия модуля тензора	Abs	–	Выполняет поэлементное взятие модуля из входных данных
17	Слой Softmax	Softmax	Ось, относительно которой применятся функция Softmax	Применяет входные данные к вероятностным значениям (0, 1). Соответствует слою softmax

№, п/п	Название слоя (функционального блока)	Обозначение в графе сети	Настраиваемые параметры	Примечание
1	2	3	4	5
18	Функция активации – сигмоида	Sigmoid	–	Применяет сигмовидную функцию ко входным данным, в результате чего выходные данные ограничиваются интервалом (0, 1). Соответствует слою sigmoid
19	Слой интерполяции	Upsample	1. Выходные пространственные размерности 2. Метод интерполяции	Применяет ко входным данным выбранный алгоритм интерполяции для изменения (увеличения) их пространственных размерностей
20	Объединение по заданной оси	Concat	Номер оси объединения	Выполняет объединение данных по заданной оси. Реализует работу слоя depth_connect
21	Разбиение по заданной оси	Split	1. Номер оси объединения. 2. Размер части	Выполняет разбиение входных данных на части заданного размера по заданной оси
22	Объединение размерности	Flatten	Ось, по которой все размерности будут свернуты	Слой, позволяющий объединить размерности по осям, идущим после указанной оси. На вход принимает любой тензор, на выходе возвращает свернутый тензор
23	Слой преобразования для детектора объектов YOLO	DetectionHead	Число классов	Преобразует вывод последнего сверточного слоя в форму, необходимую для обнаружения объектов. Реализует работу слоя yolo_transform.
24	Выходной слой для детектора объектов YOLO	DetectionYolo Postprocessing	1. Число классов; 2. Порог по confidence; 3. Порог по nms	Обеспечивает конвертацию положения ограничивающей рамки целевых объектов и оценку вероятности класса в человекочитаемый формат. Реализует работу слоя yolo_out.
25	Функция потерь для детектора объектов YOLO	DetectionYolo Loss	Число классов	Реализует функцию потерь, которая используется в процессе обучения нейронной сети для детекции на базе алгоритма Yolo

№, п/п	Название слоя (функционального блока)	Обозначение в графе сети	Настраиваемые параметры	Примечание
1	2	3	4	5
26	Слой преобразования для детектора объектов FasterRcnn	RcnnHead	1. Количество классов; 2. Порого по confidence; 3. Порог по nms; 4. Выходная размерность регионов кандидатов после слоя RoiPolling	Преобразует вывод последнего сверточного слоя в форму, необходимую для обнаружения объектов. В него входят такие части алгоритма FasterRcnn, как: RPN RoiPolling MLP Реализует работу слоев roi_input, roi_predict, roi_max_pooling.
27	Выходной слой для детектора объектов FasterRcnn	DetectionRcnn Postprocessing	1. Порог по score; 2. Порог по nms	Обеспечивает конвертацию положения ограничивающей рамки целевых объектов и оценку вероятности класса в человекочитаемый формат
28	Функция потерь для детектора объектов FasterRcnn	DetectionRcnn Loss	1. Число регионов кандидатов, использующихся для обучения RPN модуля; 2. Число регионов кандидатов, использующихся для обучения модуля предсказаний	Реализует функцию потерь, которая используется в процессе обучения нейронной сети для детекции на базе алгоритма FasterRcnn
29	Выходной слой семантической сегментации	Segmentation Postprocessing	–	Обеспечивает преобразование выхода нейронный сети для семантической сегментации в человекочитаемый формат (на выходе данного слоя формируется итоговая метка класса для каждого пикселя). Реализует работу слоя classif_pix_out.
30	Функция потерь для семантической сегментации	Segmentation Loss	–	Реализует функцию потерь, которая используется в процессе обучения нейронной сети для семантической сегментации

№, п/п	Название слоя (функционального блока)	Обозначение в графе сети	Настраиваемые параметры	Примечание
1	2	3	4	5
31	Выходной слой классификации	Classification Postprocessing	–	Обеспечивает преобразование выхода нейронный сети для классификации в человекочитаемый формат (на выходе данного слоя формируется итоговая метка класса)
32	Функция потерь для классификации	Classification Loss	Функция ошибки классификации (MSE, MAE, CrossEntropy)	Реализует функцию потерь, которая используется в процессе обучения нейронной сети для классификации. Реализует работу слоев classif_out и regress_out (в зависимости от заданной функции ошибки).
33	Выходной слой анализа изменений	Change Detection Postprocessing	–	Обеспечивает преобразование выхода нейронный сети для анализа изменений в человекочитаемый формат (на выходе данного слоя формируется итоговая метка класса для каждого пикселя)
34	Функция потерь для анализа изменений	Change Detection Loss	–	Реализует функцию потерь, которая используется в процессе обучения нейронной сети для анализа изменений.
35	CBL	CBL	1. Количество каналов в выходном тензоре 2. Размер ядра свертки 3. Размер шага свертки 4. Размер дополнения входного тензора по краям 5. Расстояние между точками ядра свертки 6. Число групп 7. Использование биасов (смещений)	Блок состоит из последовательно идущих слоев свертки, батч-нормализации и функции активации LeakyReLU

№, п/п	Название слоя (функционального блока)	Обозначение в графе сети	Настраиваемые параметры	Примечание
1	2	3	4	5
			8. Коэффициент наклона	
36	CBM	CBM	1. Количество каналов в выходном тензоре 2. Размер ядра свертки 3. Размер шага свертки 4. Размер дополнения входного тензора по краям 5. Расстояние между точками ядра свертки 6. Число групп 7. Использование биасов (смещений)	Блок состоит из последовательно идущих слоев свертки, батч-нормализации и функции активации Mish
37	ResnetBasicBlock	ResnetBasic Block	1. Количество каналов в выходном тензоре 2. Размер шага свертки	Базовый блок сетей класса ResNet
38	ResnetBottleneck	Resnet Bottleneck	1. Количество каналов в выходном тензоре 2. Количество каналов в выходном тензоре	Блок сетей класса ResNet
39	ResnetLayer	ResnetLayer	1. Количество каналов в выходном тензоре 2. Количество блоков в слое 3. Размер шага свертки	Конструктор слоев сети класса ResNet на основе **ResnetBasicBlock** или **ResnetBottleneck** с заданным количеством блоков
40	ASPPConv	ASPPConv	1. Количество каналов в выходном тензоре 2. Расстояние между точками ядра свертки	Блок дилатационной свертки нейронной сети DeepLabV3
41	ASPPPooling	ASPPPooling	Количество каналов в выходном тензоре	Блок пирамидального пулинга нейронной сети DeepLabV3
42	ASPP	ASPP	Список значений параметра дилатации для блока ASPPConv	Блок пирамидальной свертки нейронной сети DeepLabV3

№, п/п	Название слоя (функционального блока)	Обозначение в графе сети	Настраиваемые параметры	Примечание
1	2	3	4	5
43	SPP	SPP	–	Блок пространственного пирамидального пулинга.
44	DarknetBlock	DarknetBlock	Количество Shortcut блоков	Блок сетей семейства YOLO, использующий Shortcut и CBL блоки как основу
45	CspDarknetBlock	CspDarknet Block	Количество Shortcut блоков	Блок сетей семейства YOLO, использующий CspShortcut и CBM блоки как основу
46	SPPF	SPPF	1. Количество каналов в выходном тензоре 2. Размер скользящего окна	Блок пирамидального пулинга для нейросети YOLOv5
47	YoloC3	YoloC3	1. Количество каналов в выходном тензоре 2. Количество Bottleneck блоков 3. Использование shortcut	Один из базовых блоков нейросети YOLOv5, состоящий из последовательно идущих конволюций и заданного количества боттлнеков, которые конкатенируются после всех операций
48	CBS	CBS	1. Количество каналов в выходном тензоре 2. Размер ядра свертки 3. Размер шага свертки 4. Размер дополнительного входного тензора по краям 5. Расстояние между точками ядра свертки 6. Число групп 7. Использование биасов (смещений)	Блок, состоящий из последовательно идущих слоев свертки, батч-нормализации и функции активации SiLU.
49	DetectionWithKeypoi ntsHead	DetectionWithKe ypointsHead	Количество классов	Блок, реализующий слой обработки выходов нейросети YOLOv5 для diamond-разметки, включающий в себя свертку. Аналогичен слою YOLOHead.
50	StemBlock	StemBlock	1. Количество каналов в выходном тензоре	

№, п/п	Название слоя (функционального блока)	Обозначение в графе сети	Настраиваемые параметры	Примечание
1	2	3	4	5
			2. Размер ядра свертки 3. Размер шага свертки 4. Размер дополнения входного тензора по краям 5. Число групп	

– формирование (редактирование) архитектуры сверточной нейронной сети в графическом представлении, путем:

1) добавления слоев из таблицы на рабочую область конструктора;
2) редактирование параметров добавленных слоев;
3) удаление добавленных слоев с рабочей области конструктора;
4) соединение связями добавленных на рабочую область конструктора слоев;

– возможность группирования (группового выделения) и изменения порядка слоев при помощи их перемещения или перетаскивания;
– отображение общей информации о формируемой архитектуре нейронной сети, включающей: количество слоев, количество параметров, размер нейросетевой модели (в MiB) и пр.;
– обучение нейросетевых моделей, предназначенных для обнаружения, классификации анализа изменений и семантической сегментации объектов на изображениях с поддержкой следующего функционала и с соответствующим интерфейсом пользователя:

1) определение директории расположения тренировочных и валидационных данных в виде каталога с файлами, содержащими изображения, и файлами разметки в формате json;
2) визуальная оценка корректности валидационных и тренировочных данных;
3) выбор и загрузка разработанной модели нейронной сети;
4) автоматическая оценка коэффициентов сбалансированности классов в обучающем наборе;
5) корректировка функций потерь с учетом коэффициентов сбалансированности классов;
6) установка параметров аугментации тренировочных данных:
   – случайный поворот;
   – случайное масштабирование;
   – случайный сдвиг по оси X и Y;
   – случайное отражение по оси X и Y;
   – случайна вырезка фрагментов заданного размера;
   – случайное изменений яркости (контрастности).
7) установка параметров алгоритма обучения (дообучения):
   – метод оптимизации (Sgd, Adam, RMSProp);

– метод изменения скорости обучения (StepLrSheduler, ExponentialLrSheduler)

– число эпох обучения;

– размер обучающего мини-пакета данных;

– коэффициент снижения скорости обучения;

– частота валидации;

8) сохранение контрольных точек обучения с весами нейросетевой модели, а также метаданными параметров обучения и оценками точности;

9) остановка процесса обучения по заданным пользователем параметрам или критериям остановки с сохранением обученной модели нейронной сети с метаданными параметров обучения и показателями оценки точности;

10) визуализация показателей оценки точности обученной модели:

– точность семантической сегментации по каждому классу;

– индекс Жаккарда (IoU) по каждому классу;

– усредненная точность сематической сегментации по всем классам;

– усредненный индекс Жаккарда (IoU) по всем классам;

– усредненный показатель точности локализации границ объектов (BFscore) по всем классам;

– матрица ошибок;

11) квантизация весовых коэффициентов обученной модели в форматы fp16, int8;

12) тестирование обученной модели на тестовом наборе обучающих данных с визуальной оценкой результата;

13) конвертация обученной модели в формат ONXX или json + pth.

– дообучение существующих нейросетевых моделей на основе новых или синтезированных обучающих данных с возможностью «заморозки» отдельных слоев;

– сохранение (выгрузка) нейросетевых моделей в формате ONNX;

– отображение результатов обучения в графическом и текстовом виде.

14) Входными и выходными данными ВПСНС являются нейросетевые модели в формате ONXX или json + pth.

4.3.2.8 При реализации определенных проектов на базе бета-версии Решения в соответствии с запросами Заказчиков проектов могут быть уточнены:

1) Перечень слоев и типы промежуточных связей уточняются на этапе технического проектирования и согласуется с Заказчиком.

2) Перечень выводимой информации может уточняться в процессе разработки и согласуется с Заказчиком.

3) Перечень выгружаемых форматов нейросетевых моделей может уточняться. Типы графиков и выводимые параметры будут согласоваться с потенциальными Заказчиками. Перечень форматов нейросетевых моделей также может уточняться. Формат обучающих данных выбирается Исполнителем.

4.3.2.9 Должен быть разработан функциональный блок синтеза обучающей выборки и разметки синтезированных данных (СОВРСД). Блок СОВРСД будет обеспечивать:

– загрузку многоканальных изображений в форматах JPEG, TIFF, PNG;

– загрузку трехмерных моделей объектов в форматах:

- в открытом формате FBX – формат, разработанный специально для хранения и обмена данными трёхмерных сцен;
- в открытом формате 3DS – файл содержит 3D-сцену, созданную в программе трехмерного моделирования. Файл хранит данные сетки, свойства материала, ссылки на растровые изображения, параметры освещения и расположения камеры, данные анимации и другую информацию;
- в открытом формате OBJ без поддержки точек привязки, скелетов, анимации, PBR материалов;

- настройку дерева генерируемых объектов;
- настройки параметров генерации обучающей выборки:
  - количества изображений;
  - размера изображений;
  - преобразований изображений;
  - настройки параметров разметки изображений.

Функциональный блок СОВРСД предназначен для синтезирования обучающих данных, применяемых в процессе обучения сверточных нейронных сетей на основе моделирования фотореалистичных изображений объектов на фоне различной подстилающей поверхности в видимом диапазоне спектра.

Функциональный блок СОВРСД должно обеспечивать решение следующих задач:

1) Загрузка панхроматических и многоканальных изображений подстилающей поверхности в форматах JPEG, TIFF, ECW и RSW от 8 до 16 бит на канал размером не менее 2 Гбайт;

2) Загрузка трехмерных моделей объектов в форматах OBJ, FBX, 3DS MAX:
   - выбор цвета модели или выборе текстуры модели;
   - предпросмотр модели;
   - выбор поворотных частей модели, с которыми осуществляется дальнейшая работа и настройка на сцене;

3) Постановка трехмерных объектов загруженных моделей на подстилающей поверхности в виртуальном трехмерном пространстве сцены:
   - настройка правил рендера объекта;
   - настройка положения объекта на сцене и поворота его составных частей;
   - настройка видимости объектов на сцене;

4) Автоматическая расстановка трехмерных моделей на полигоне на подстилающей поверхности в виртуальном трехмерном пространстве сцены

5) Автоматический расчет положения источника света при наличии данных о дате и времени съемки и координатах съемки

6) Настройка угла источника света и интенсивности

7) Настройка позиции камеры, угла поворота и правил, используемых при рендере

8) Динамическое отображение масштаба изображения при разном уровне приближения сцены относительно масштаба подстилающей поверхности

9) Формирование общего обучающего изображения панхроматических и многоканальных изображений объектов в форматах TIFF, PNG, JPEG, BMP с сопутствующим файлом разметки в форматах TXT и XML, содержащих координаты вершин ограничивающих рамок для каждого объекта и идентификатор типа объекта

10) Пакетное формирование изображений объектов в формате PNG по области изображения заданного размера, а также с использованием набора подложек подстилающей поверхности и данных рельефа.

3D-модели для генерации обучающих данных предоставляются заказчиком определенных проектов, разрабатываемых на базе бета-версии Решения.

4.3.2.10 Компонент аналитической обработки данных должен обладать функциональностью, предоставляющей заказчику самостоятельно, без использования средств программирования, настраивать логику работы информационных систем с максимальной степенью автоматизации и принятием решений искусственным интеллектом (AI) без участия человека (в этапах бизнес-процессов, где можно выявить закономерность между входными и выходными данными).

4.3.2.11 Решение должно обеспечивать возможность автоматизации процессов принятия решений по определенной категории задач без участия человека.

4.3.3 Функциональные требования к компоненту (модулю) визуализации аналитической отчетности

4.3.3.1 В рамках реализации Проекта должен быть интегрирован компонент визуализации аналитической отчетности в общую архитектуру бета-версии Решения.

4.3.3.2 Должна быть осуществлена доработка инструментов автоматической генерации отчетов, формируемых с определенной периодичностью (ежедневные, ежемесячные, ежеквартальные, ежегодные и т.д.). Должна быть проведена интеграция с планировщиком задач и файловым хранилищем.

Стандартные таблицы (элементы) инструмента генерации отчетных форм должны содержать следующие поля:

- наименование отчета
- периодичность обновления.

4.3.3.3 Должен быть расширен банк предустановленных дашбордов и отчетов, необходимых для работы специалистов потенциальных заказчиков.

Перечень сфер экономики, для которых предполагается формирование банка предустановленных дашбордов и отчетов, определяется тем, что помимо государственного сектора экономики страны и компаний с государственным участием основными потребителями BI-решений являются:

- банковский и страховой секторы;
- телекоммуникационный сектор;
- ритейл;
- промышленность (в последние годы крупные промышленные предприятия увеличили спрос на BI-системы);
- здравоохранение (демонстрирует быстрый рост спроса на BI-системы).

Принимая во внимание вышесказанное, предустановленные дашборды должны содержать:

- анализ эффективности:
  - эффективность рекламы
  - KPI контрактов
  - Производственный KPI
  - KPI маркетинг
  - KPI обслуживания клиентов

    - KPI продажи
    - KPI техподдержки
    - SaaS метрики
    - ключевые показатели деятельности
- статистика:
    - пользователи социальных сетей
    - отслеживание посещаемости ресторана
    - статистика продаж
    - билетная статистика
    - гоночная серия
    - ОПЕК
    - ланчи в ресторанах быстрого питания
    - счетчики и конверсия сайта
    - статистика сайта
    - статистика заказов.
- Региональные анализ:
    - анализ пользователей
    - использование глобальной сети.
- Продажи:
    - обзор продаж
    - показатели продаж
    - продажи по месяцам в РФ
    - продажи по компаниям
    - новогодние продажи
    - недвижимость.
- Финансовые показатели:
    - биржевые торги
    - основные финансовые показатели.
- Производство:
    - производство
    - производство автомобилей
    - продукты.
- Образование:
    - студенческая статистика
    - разделы
    - финансовая статистика
    - статистика университета
- Медицина:
    - обзор больницы
    - анализ деятельности больницы
    - KPI медицина
    - KPI медицинские финансы
    - внутренний KPI
    - медицина стартовый
    - KPI операций

- здоровье пациентов
- удовлетворенность пациентов
- KPI общественного здоровья
- фитнес и здоровье.

Предустановленные отчеты должны содержать:

- Отчеты:
  - платежная ведомость
  - заказ
  - заявка
  - счет
  - страховой отчет
  - штрих-код.

Каждый раздел должен содержать не менее двух дашбордов/отчетов.

Реализация обозначенной задачи позволит значительно повысить эффективность проведения презентационных мероприятий, проводимых при первичном знакомстве потенциальных Заказчиков с бета-версией Решения, а также значительно оптимизирует процесс внедрения программного решения для нужд Заказчиков, представляющих различные отрасли экономики.

4.3.3.4 В рамках Проекта также будет доработан механизм, обеспечивающий корректный экспорт отчетов в соответствии с верстkой в бета-версии решения.

Допустимые форматы экспортируемых отчетов:

- Report Snapshot (.mdc, .mdz, .mdx)
- Adobe PDF (.pdf)
- Microsoft PowerPoint (.pptx)
- HTML (.html)
- RTF (.rtf)
- Текст (.txt)
- Microsoft Word (.docx)
- OpenDocument Writer (.odt)
- Microsoft Excel (.xls)
- OpenDocument Calc (.ods)
- Данные (.csv)
- Изображение (.jpeg)

4.3.3.5 Должен быть расширен банк обучающих материалов, текущие материалы должны быть обновлены с учетом новых функциональных возможностей. Доработанный банк обучающих материалов будет использован для обеспечения поддержки специалистов Заказчиков, самостоятельно создающих новые виды отчетов и дорабатывающих текущие предустановленные типовые формы.

4.3.3.6 Банк обучающих материалов должен содержать основную информацию для пользователя по работе с основными функциями бета-версии Решения в части BI:

- Создание отчета/дэшборда
- Импорт и экспорт отчета/дэшборда
- Сохранение отчета/дэшборда
- Работа со словарем, а именно работа с источниками данных

- Связывание источников данных
- Добавление компонентов (бэндов) на отчет/дэшборд
- Работа со свойствами элементов
- Написание выражений и функций
- Создание и применение стилей
- Форматирование текста
- Просмотр дэшбордов и отчетов

4.3.3.7 Должна быть обеспечена функциональность по сбору первичных данных из различных источников информации. Загрузка данных должна происходить через конструктор отчетов/дэшбордов во вкладке «Словарь» с помощью инструмента «Источники данных».

Источники данных:

- CSV
- Dbase
- Excel
- GIS
- JSON
- XML
- MS SQL
- MySQL
- Oracle
- PostgreSQL
- ODBC
- OLE DB
- Ms Access
- Firebird
- SQL CE
- SQLite
- DB2
- Informix
- Sybase
- Sybase ADS
- Teradata
- VistaDB
- Universal(dotConnect)
- Elasticsearch
- MongoDB
- Azure Blob Storage
- Azure SQL
- Azure Table Storage
- Cosmos DB
- BigQuery
- Firebase

- Google Analytics
- Google Sheets
- Data.World
- QuickBooks
- OData
- Объекты (данные из Business Objects, DataSet, DataTables, DataViews, User Sources, данные на основе других данных, данные на основе Кросс-таблицы).

4.3.3.8 Инструменты проектирования отчетных форм (конструктор) должны быть доступны с любого рабочего места из АИС без необходимости установки на локальный компьютер (в виде кроссплатформенного web-приложения). Конструктор отчетов должен предоставлять следующий функционал:

- создание, открытие, сохранение отчета;
- создание и загрузка шаблонов отчетов;
- создание компонентов и контейнеров отчета:
  - страница;
  - группа категорий;
  - секции (стандартные бэнды (элементы)):
    - заголовок отчета;
    - итог отчета;
    - заголовок страницы;
    - итог страницы;
    - заголовок группы;
    - итог группы;
    - заголовок данных;
    - итог данных;
    - заголовок колонки;
    - итог колонки;
    - данные;
    - дерево;
    - подчиненный;
    - пустые данные;
    - водяной знак;
  - кросс-бэнды:
    - кросс-заголовок группы;
    - кросс-итог группы;
    - кросс-заголовок данных;
    - кросс-итог данных;
    - кросс-данные;
  - текст;
  - текст в ячейках;
  - изображения;
  - панели;
  - флажок;
  - вложенный отчет;
  - почтовый индекс;

- таблицы;
- кросс-таблицы;
- штрих-коды:
  - линейные штрих-коды:
    - UPC-A;
    - UPC-E;
    - EAN-13;
    - EAN-8;
    - EAN-128;
    - ITF-14;
    - JAN-13;
    - JAN-8;
    - ISBN-10;
    - ISBN-13;
  - двумерные штрихкоды:
    - PDF417;
    - Datamatrix;
    - QR Code;
  - параметры штрих-кодов:
    - код;
    - угол;
    - масштаб;
    - цвет;
    - цвет фона;
    - шрифт;
    - включение показа метки;
- геометрические объекты:
  - стрелка;
  - горизонтальная линия;
  - вертикальная линия;
  - прямоугольник;
  - закругленный прямоугольник;
  - овал;
  - диагональные линии;
  - трапеция;
  - параллелограмм;
  - знаки равно, сложение, умножение;
  - блок-схема: Ручной ввод;
  - блок-схема: Карта;
  - блок-схема: Подготовка
  - блок-схема: Решение;
- диаграммы:
  - гистограммы:
    - гистограмма;

- гистограмма с накоплением;
- нормированная гистограмма;
- Парето;

- линейная диаграмма:
  - линейная диаграмма;
  - линейная диаграмма с накоплением;
  - нормированная линейная диаграмма;
  - плавная линейная диаграмма (сплайн);
  - плавная линейная диаграмма с накоплением
  - нормированная плавная линейная диаграмма;
  - пошаговая линейная диаграмма;
- область:
  - область;
  - область с накоплением;
  - нормированные области с накоплением;
  - сглаженная область;
  - сглаженная область с накоплением;
  - нормированные сглаженные области;
  - пошаговая область;
- диапазоны:
  - диапазон;
  - плавный Диапазон;
  - шаговый Диапазон;
  - столбчатый диапазон;
- линейчатая диаграмма:
  - линейчатая;
  - линейчатая с накоплением;
  - нормированная линейчатая;
- точечная:
  - точечная диаграмма;
  - точечная с линиями;
  - точечная с плавными линиями;
- круговая;
- лепестковая:
  - лепестковая с точками;
  - лепестковая с линиями;
  - лепестковая область;
- воронка:
  - воронка;
  - взвешенная воронка;
- финансовая:
  - японские свечи;
  - биржевая;

- другие:
  - древовидная;
  - Ганта;
  - кольцевая;
  - пузырьковая диаграмма;
  - пиктографическая карта;
  - каскадная диаграмма;
- карты:
  - индивидуальная карта;
  - группа;
  - тепловая карта;
  - тепловая карта с группировкой;
- форма;

- настройка оформления отчета:
  - работа с буфером обмена;
  - оформление отчета с использованием свойств компонента или стилей и коллекций стилей:
    - работа с кистями:
      - заполнение фона;
      - отрисовка текста;
    - работа со шрифтами;
    - выравнивание компонентов;
    - определение границ компонентов;
    - работа с текстом:
      - оформление текста:
        - форматирование чисел:
          - разделение групп разрядов;
          - применение региональных настроек;
          - использование аббревиатур;
          - отображение отрицательных значений красным;
          - задание числа знаков дробной части;
          - задание разделителя дробной части;
          - задание разделителя дробной группы;
          - задание размера группы;
        - форматирование денежных значений:
          - поддержка форматирования чисел;
          - задание символа валюты;
        - форматирование даты и времени;
        - форматирование значений с процентом;
        - форматирование булевских значений;
        - настройка пользовательского форматирования;
      - оформление текста с помощью HTML-тегов:
        - тег <font>, позволяющий определить:
          - цвет;

- шрифт;
- размер;
- теги для изменения стиля шрифта;
- тег <background-color> для задания цвета фона;
- тег <text-align> для задания выравнивания;
- тег <letter-spacing> для задания интервала между символами;
- тег <word-spacing> для задания интервала между словами;
- тег <line-height> для задания интервала между строками;
- форматирование текста;
- настройка отображения текста;
- настройка качества текста;
- условное форматирование компонентов;
- настройка автоматического изменения размера компонентов;
- создание вложенных отчетов;
- использование интерактивных элементов:
  - закладки;
  - гиперссылки;
  - интерактивное выделение;
  - детализация;
  - динамическая сортировка;
  - динамическое сворачивание;
  - редактирование отчета в окне предварительного просмотра;
- настройка выражений:
  - текстовые выражения;
  - выражения с вычислением значения;
  - многострочные выражения;
  - выражения с использованием переменных из словаря;
  - выражения с использованием полей из источников данных;
  - выражения с использованием свойств компонентов;
  - выражения с использованием функций;
  - выражения с условием;
  - выражения с псевдонимами;
- настройка штрих-кодов:
  - выбор типа штрих-кода;
  - настройка штрих-кода:
    - параметры штрих-кода:
      - значение штрих-кода;
      - поворот штрих-кода;
      - автомасштабирование;
      - цвет графических элементов и фона штрих-кода;
      - шрифт;
      - подпись;
      - границы;

        - предварительный просмотр;
        - свойства штрих-кода;
- настройка списков:
    - выбор источника данных;
    - настройка сложных списков:
        - связанные списки (Master-Detail списки);
        - древовидный список;
    - настройка заголовка;
    - настройка итога;
    - выбор стиля оформления списка;
    - настройка нумерации строк;
    - сортировка данных списка:
        - выбор уровня сортировки;
        - выбор порядка сортировки;
    - фильтрация данных списка:
        - выбор типа логической операции:
            - настройка множественной фильтрации;
            - операция логического И;
            - операция логического ИЛИ;
            - операция логического исключающего ИЛИ;
            - операция равно/ не равно;
            - операция между/не между;
            - операция больше чем;
            - операция больше чем или равно;
            - операция меньше чем;
            - операция меньше чем или равно;
            - операция содержит/не содержит;
            - операция начинается с;
            - операция заканчивается на;
        - выбор условия фильтрации:
            - задание условия с помощью мастера;
            - задание условия с помощью выражения;
- настройка группировки данных;
    - выбор источника данных;
    - выбор условия группировки
    - выбор направления группировки;
    - настройка заголовка;
    - настройка итога;
    - настройка сортировки данных в группе и групп;
    - настройка вывода данных в колонках;
- настройка нумерации страниц;
- настройка панелей;
- настройка таблиц:
    - выбор источника данных;

- настройка связей для Master-Detail таблицы;
- настройка элементов таблицы:
  - заголовок;
  - ячейки:
    - перенос текста в ячейке;
    - ширина:
      - автоширина;
      - фиксированная ширина;
    - тип ячейки:
      - текст;
      - изображение;
      - флажок;
  - итог;
- группировка;

- настройка кросс-таблиц:
  - выбор источника данных;
  - настройка связей;
  - настройка элементов кросс-таблицы:
    - колонки;
    - строки;
    - столбцы;
    - итоговые ячейки (ячейки суммирования):
      - включая функции суммирования значений в ячейках:
        - не суммировать значения ячейки;
        - рассчитать сумму значений, содержащихся в ячейке;
        - рассчитать среднее из значений, содержащихся в ячейке
        - рассчитать минимальное из значений, содержащихся в ячейке;
        - рассчитать максимальное из значений, содержащихся в ячейке;
        - рассчитать количество значений, содержащихся в ячейке;
        - рассчитать количество отличающихся значений, содержащихся в ячейке;
        - поддержка математических операций над значениями ячеек;
  - группировка;
  - фильтрация;
  - сортировка;
  - настройка стиля оформления кросс-таблицы;
- настройка диаграмм:
  - основные настройки:
    - поворот;
    - расстояние по горизонтали (отступ);
    - расстояние по вертикали (отступ);
    - источник данных;

        - связь;
        - количество данных;
    - настройка легенды:
        - применение готовых стилей;
        - колонки;
        - направление;
        - горизонтальное выравнивание;
        - расстояние по горизонтали;
        - выравнивание маркера;
        - граница маркера;
        - размер маркера;
        - показывать маркер;
        - размер по ширине и высоте;
        - заголовок;
        - вертикальное выравнивание;
        - расстояние по вертикали;
    - настройка заголовка;
    - задание константных линий:
        - значение оси, на котором выводится линия;
        - стиль линии;
        - толщина линии;
        - ориентация;
        - позиция;
        - показывать сзади;
        - текст;
    - задание полос:
        - максимальное значение;
        - минимальное значение;
        - ориентация;
        - текст;
- настройка карт:
    - выбор типа карты;
    - определение источника данных;
    - настройка отображения наименований;
    - заполнение ключей, наименований, значений;
    - выбор цвета;
- настройка функций:
    - вычисления дат:
        - вычисление интервала между указанными датами;
        - отображение дня из указанной даты;
        - отображение дня недели;
        - отображение дня года;
        - отображение количества дней в месяце;
        - отображение количества дней в году;
        - отображение названия месяца из указанной даты;

- математических:
  - вычисление абсолютного числа;
  - вычисление угла в радианах;
  - вычисление значения cos;
  - вычисление результата деления одного аргумента на другой;
  - вычисление результата возведения в указанную степень экспоненты;
  - вычисление минимального целочисленного значения к указанному числу;
  - вычисление натурального логарифма;
  - сравнение двух значений и отображение максимального;
  - сравнение двух значений и отображение минимального;
  - округление значения до целого числа или до заданного количества знаков дробной части;
  - определение индикатора со знаком числа;
  - вычисление sin;
  - вычисление квадратного корня из числа;
  - вычисление tg
  - вычисление целой части числа без округления;
- работы со строками:
  - конвертирование указанных цифр в арабские цифры;
  - конвертирование даты в текстовое значение;
  - вставка значения после определенного символа в другое значение;
  - вычисление количества символов;
  - конвертирование значения в нижний/верхний регистр;
  - конвертирование числа в текст;
- настройка параметров страницы:
  - настройка полей;
  - настройка ориентации листа;
  - настройка размеров листа;
  - настройка колонок страницы:
    - задание числа колонок;
    - задание ширины колонок;
    - задание расстояния между колонками;
  - настройка водяного знака;
    - текст водяного знака;
    - кисть текста;
    - шрифт;
    - угол;
    - отображение на заднем или переднем фоне;
    - вывод изображения;
    - выравнивание изображения;
    - множитель изображения (масштабирование);
    - соотношение сторон;
    - повтор изображения;
    - прозрачность изображения;

        - режим растягивания изображения;
    - настройка форматов управления размещением компонентов на странице:
        - иерархия;
        - выравнивание;
        - расположение по слоям;
- предпросмотр отчета;
- печать отчета;
- настройка печати бэндов:
    - на всех страницах;
    - кроме первой страницы;
    - кроме последней страницы;
    - кроме первой и последней страницы;
    - только первая страница;
    - только последняя страница;
    - только первая и последняя страница;
    - печатать на четных-нечетных страницах;
    - печатать на всех страницах;
- просмотр свойств каждого компонента;
- просмотр словаря отчета;
- просмотр (подключение) источников данных:
    - MS SQL;
    - MySQL;
    - ODBC;
    - OLE DB;
    - Oracle;
    - MS Access;
    - PostgreSQL;
    - Firebird;
    - SQL CE;
    - SQLite;
    - DB2;
    - Infomix;
    - Sybase;
    - Teradata;
    - VistaDB;
    - Universal(dotConnect);
    - MongoDB;
    - OData;
    - бизнес-объектов;
    - переменных;
    - системных переменных;
    - функций;
    - ресурсов.

4.3.3.9 Конструктор отчетов должен содержать команды настройки компонентов отчета, которые позволяют вставить, скопировать, вырезать, удалить выделенные компоненты шаблона отчета.

Конструктор отчетов должен поддерживать следующие форматы экспорта файлов:

- форматы с фиксированной разметкой страницы:
  - формат PDF (Portable Document Format);
  - формат XPS (XML Paper Specification);
  - формат PPT (Microsoft PowerPoint presentation software);
  - формат HTML (HyperText Markup Language);
  - Text file (TXT);
  - Rich Text формат (RTF);
  - формат Microsoft Word 2007;
  - формат Open Document Text (ODT).
- Электронные таблицы:
  - формат Microsoft Excel;
  - формат Microsoft Excel Xml;
  - формат Microsoft Excel 1997-2003;
  - формат Open Document Spreadsheet (ODS).
- Данные:
  - формат CSV (Comma Separated Values);
  - формат DBF (DataBase File);
  - формат XML (eXtensible Markup Language);
  - формат DIF (Data Interchange Format);
  - формат SYLK (Symbolic Link).
- Изображения:
  - формат BMP (bitmap);
  - формат GIF (Graphics Interchange Format);
  - формат PNG (portable network graphics);
  - формат TIFF (Tagged Image File Format);
  - формат JPEG (Joint Photographic Experts Group);
  - формат PCX;
  - формат EMF (Enhanced Metafile).

4.3.4 Функциональные требования к интеграционной шине

4.3.4.1 Бета-версия Решения должна включать предустановленные интеграции из альфа-версии Решения.

Интеграционная шина поддерживает три основных механизма интеграции:

–через обмен файлами (xlsx, csv и т.д.);

–через общую СУБД;

–через обмен сообщениями (REST, SOAP).

Интеграционная шина включает конструктор метаданных, позволяющий настраивать структуру данных, получаемых из внешних источников. Конструктор позволяет обрабатывать, как простейшие скалярные значения одного типа, так и сложные структуры, включающие несколько типов данных.

4.3.4.2 Должна быть обеспечена поддержка современных инструментов настройки информационного взаимодействия с внешними системами и аппаратно-

программными комплексами для обеспечения бета-версии Решения возможности корректного встраивания в текущую ИТ-инфраструктуру потенциальных Заказчиков, а также для организации корректного информационного обмена с внешними источниками данных для формирования целостной информационной картины.

Разработаны специальные адаптеры для интеграции с:

- ЕСИА;
- СМЭВ;
- Active Directory;
- информационные системы, поддерживающими SOAP/REST;
- почтовыми сервисами SMTP;
- механизмами шифрования и усиленной квалифицированной электронной подписи.

4.3.4.3 Должны быть обеспечена возможность интеграции компонента интеграционной шины бета-версии Решения с распространенными BI-решениями иностранных вендоров с целью обеспечения корректной миграции данных при импортозамещении продуктов на российском рынке.

Ниже представлен список BI-решений иностранных вендоров, с которыми должна быть обеспечена интеграция:

1) Power BI;
2) Qlik Sense.

Интеграция возможна в рамках конкретных проектов, реализуемых на базе бета-версии Решения с учетом требований заказчика, только при условии предоставлении со стороны заказчика реквизитов доступа к API, с входными параметрами для API методов по получению источников данных. Интеграция должна быть обеспечена через BI-конструктор с использованием протокола OData.

4.3.4.4 Требования к функциям (задачам) подсистемы интеграции (API)

Модуль интеграции должен иметь следующую функциональность:

- ведения базы метаданных, включающей описания структуры единой базы данных, источников информации, мест хранения информации, протоколов информационного обмена, форматов представления документов;
- загрузка как структурированных, так и неструктурированных данных с их последующим размещением в подсистеме хранения данных;
- загрузка информации из источников в виде текстовых, табличных файлов или медиа-файлов (изображения, видео, аудио);
- возможность работы со всеми основными типами источников информации (HTTP веб-сервисы, базы данных, xml, плоские файлы, потоковые данные с различных датчиков и устройств, логи поведения пользователей и проч.), а также со всеми основными форматами данных;
- поддержка возможности организации защищенных каналов передачи данных.

Модуль должен обеспечивать выполнение следующих функций:

- обеспечение возможности поддержки синхронного и асинхронного способа вызова служб;
- обеспечение возможности использования защищенного транспорта;

- обеспечение возможности доступа к данным для внешних информационных систем (пользователей системы) с помощью специально разработанных адаптеров;
- обеспечение возможности обработки и преобразования сообщений;
- обеспечение контроля процессов информационного обмена (аудиты, протоколирование).

Для обеспечения юридически значимого документооборота должна быть реализована возможность использования электронной подписи.

Модуль должен обеспечивать:

- автоматический асинхронный (по запросу) и синхронный (по расписанию) сбор, предобработку и классификацию информации от заданных администратором источников (совместно с подсистемой интеграции);
- инкрементальные методы загрузки информации;
- минимизацию объема обрабатываемой информации за счет удаления дублей;
- автоматическую классификацию и фильтрацию данных по таким признакам, как:
  - дата получения информации;
  - объект, с которого была получена информация;
  - группам пользователей, которым доступна информация;
  - другим признакам (должны настраиваться администратором или ответственным лицом);
- получение оценок по передаваемым данным.

Модуль должен позволять:

- выполнять настройку параметров опроса источников/приемников данных;
- выполнять настройку схемы метаданных, загружаемых из источника или передаваемых в приемник данных;
- автоматически формировать метаданные на основе XML/JSON файлов;
- в соответствии с настроенными схемами проверять целостность и корректность данных;
- выполнять автоматическое наполнение информационных объектов системы, в случае настроенного отображения («мэппинга») полей, загружаемых данных на поля внутренних объектов;
- журналировать процессы по загрузке и передаче данных.

Модуль должен включать следующие функциональные блоки:

- блок планировщика для управления расписаниями запуска процедур опроса источников. Данный модуль должен выполнять расчеты времени запуска опроса для каждого источника и осуществлять запуск загрузчиков для каждого источника;
- блок выбора источника для взаимодействия с планировщиком, определения заданных источников, времени их запуска;
- блок загрузки данных для осуществления базовой навигации и выполнения загрузки данных через подсистему интеграции. Должен быть разработан конструктор правил импорта и предобработки данных для каждого источника;
- блок преобразования данных к единому виду для структурирования информации и добавления к ней метаданных;

– блок обнаружения сбоев для выявления случаев некорректной работы загрузчика в связи с изменением структуры источника или API внешних автоматизированных систем. Должна быть реализована двухуровневая проверка корректности загружаемых информационных материалов;

Модуль должен обеспечивать доступ к Системе через протоколы REST и SOAP для обмена данными с внешними информационными платформами.

4.3.5 Функциональные требования к компоненту (модулю) предобработки информации ETL

4.3.5.1 Наличие компонента предобработки информации (ETL) должно обеспечивать гибкие возможности по извлечению данных из внешних источников, их первичной проверки и оценки возможности последующей загрузки в хранилище данных бета-версии Решения, распределению данных на потоки с преобразованием данных к виду, соответствующему эталонной модели данных бета-версии Решения.

4.3.5.2 ETL-сервис должен выполнять весь цикл загрузки, выгрузки и преобразования данных через веб-интерфейс.

С помощью ETL-сервис должны выполняться извлечение, преобразования и загрузку данных силами администратора/оператора Системы.

ETL сервис должен проводить автоматическое выполнение всех типовых задач ETL:

- протоколирование выполнения задачи ETL;
- настройка очередности выполнения задачи ETL;
- извлечение данных из двумерных, реляционных и многомерных источников данных;
- извлечение/загрузка данных из внешних источников: OLE DB; баз данных Microsoft Access, DBase; файлов Microsoft Excel, XML, TXT, CSV, JSON;
- извлечение/загрузка данных из объектов репозитория: таблиц, присоединенных таблиц, представлений, запросов, источников данных ODBC, журналов;
- извлечение данных из REST-источников, доступ к которым осуществляется через Internet;
- преобразование данных: слияние, разделение, удаление дубликатов, группировка, фильтрация, сортировка.
- выгрузка данных в необходимый формат для передачи во внешнюю систему.

ETL сервис может быть использован для:

- управления информационным обменом с внешними системами;
- обработки и преобразования плоских и иерархичных данных для получения и передачи через API;
- настройки разовой миграции большого количества данных.

4.3.5.3 Встроенный конструктор должен позволять мониторинг и контроль процесса исполнения, оповещая о неправильном использовании ресурсов, порядке и статусе обработки данных.

ETL сервис должен обеспечивать извлечение больших данных из различных источников, их дальнейшее прототипирование, анализ, проверку, преобразование и загрузку в любое новое место назначения.

Подраздел ETL-сервиса предоставляет набор готовых сценариев и мастеров и состоит из следующих пунктов меню:

– Реестр для выгрузки;
– Реестр загрузок;
– Xslt редактор (редактор/преобразователь файлов форматов xsl, xml);
– Реестр преобразованных данных (регистры ETL);
– ETL статусы.

4.3.5.4 В ходе развития ETL-компонента должен быть доработан интерфейс XSLT-схем. В верхней части карточки (горизонтально) должны располагаться основные команды для работы с формой («Восстановить», «Пресет», «Импорт/экспорт пресета») и команда для сохранения вносимой информации – «Сохранить».

Форма (карточка) должна содержать поля:

– «Заголовок» – для ввода наименования адаптера загрузки (схемы преобразования);
– «Активация функции» – активация адаптера загрузки в реестре схем;
– «Ввод Xslt» – загрузка схемы преобразования из файлового хранилища или другого репозитория.

В верхнем правом углу должны располагаться кнопки команд «Сохранить» и «Сохранить и закрыть».

4.3.5.5 Должны быть доработаны обучающие материалы, обеспечивающие методологическую поддержку пользователей компонента.

Обучающие материалы должны содержать информацию о создании нового ETL-процесса. А именно:

Шаги формирования ETL-процесса на загрузку:

– подготовка загружаемых данных;
– создание ETL-процесса (бизнес-процесса);
– загрузка данных (запуск бизнес-процесса).

Загружаемые данные в формате XML.

Процесс подготовки данных для Excel-таблиц:

– описать структуру данных в виде XSD-схемы;
– импортировать XSD-схему в Excel;
– сопоставить схему со столбцами;
– экспортировать данные.

Создание ETL-процесса:

– сформировать XSL-схему трансляции данных;
– сформировать БП для загрузки.

Формирование XSL-схемы:

– открыть пункт меню «Конструктор схем сопоставления»;
– загрузить подготовленный XML-файл с исходными данными (не более 1 Мб);
– сформировать схему сопоставления данных.

Формирование бизнес-процесса:

– перейти в модуль бизнес-процессов;
– создать БП с указанием типа объекта «FileEtlProcess» – сущность, отвечающая за загрузку данных из XML в систему;
– в редакторе БП создать схему из двух блоков:
– DownloadCommand – выполняет операцию загрузки;

– DownloadMessage – выполняет операцию ожидания загрузки.

Шаги формирования ETL-процесса на выгрузку данных из системы:

– создание ETL-процесса;
– выгрузка данных.

Создание ETL-процесса:

– сформировать БП для выгрузки;
– сформировать XSL-схему выгрузки данных;
– выгрузить данные из системы.

4.3.6 Функциональные требования к компоненту (модулю) автоматизированного проектирования предметной области

4.3.6.1 Компонент автоматизированного проектирования предметной области построен на базе конструктора объектного проектирования (low-code), который поддерживает возможностью кодогенерации классов, что обеспечивает высокую степень адаптивности программного решения.

Конструктор low-code позволяет создавать новые классы, на основе которых создаются объекты – экземпляры классов (экземпляры объектов заданного вида), из административного интерфейса без привлечения разработчиков.

Конструктор автоматизированного проектирования предметной области должен поддерживать следующий функционал:

– создания/модификации/удаления класса объекта;
– добавления/модификации свойств (полей) к классу;
– управления правами доступа свойств (полей) класса;
– настройки правил создания и уничтожения экземпляра класса.

Механизм кодогенерации должен обеспечивать следующие функции:

– ввод параметров нового класса (представления модели данных):
  – основных данных:
    – имени;
    – наименования;
    – параметров сборки;
    – параметров использования в регламентированных процесса
    – параметров доступа
    – параметров отображения экземпляров класса
  – полей:
    – имени;
    – заголовка на экранных формах;
    – заголовка и порядкового номера в списке;
    – вкладки, на которой осуществляется вывод;
    – ограничений (максимальная длина);
    – параметров доступа (обязательное, только для чтения, скрытое);
  – параметров связей;
– трансляция параметров модели в машинный формат;
– формирование табличного представления объекта в базе данных;
– сборка связей, внешних и внутренних ссылок в базе данных;
– настройка прав доступа и схем движения объектов класса.

Конструктор проектирования предметной области должен обеспечивать добавление в модель класса полей следующих типов:

- int;
- bool;
- date;
- datetime;
- decimal;
- double;
- string;
- text;
- file;
- image;
- enum;
- агрегация;
- ссылка;
- коллекция ассоциаций.

Конструктор конфигураций (проектирования) предметной области должен позволять создавать и отображать новые объекты предметной области (классы, подмножества) наиболее оптимальным для работы пользователя методом.

Конструктор конфигураций (проектирования) предметной области должен обеспечивать:

- унификацию пользовательских интерфейсов;
- специализацию пользовательских интерфейсов под требуемые задачи;
- удаление из интерфейса служебной информации, которая ухудшает восприятие сущности;
- исключение параметров сущности, замедляющих работу пользователей;
- наследование ограничений, понятий и связей одной конфигурации у другой с помощью автоматизации операции наследования;
- создание перегруженных процедур обработки и вывода данных конфигурации.

Процесс создания новых конфигураций состоит из следующих шагов:

- выбор базовой конфигурации;
- задание имени новой конфигурации;
- настройка формы отображения конфигурации;
- настройка отображения списка данных;
- настройка процедур фильтрации отображаемых данных;
- настройка наименования конфигурации, формы и отображения;
- настройка доступа к конфигурации в системе.

4.3.6.2 В рамках реализации проекта должна быть проведена интеграция компонента автоматизированного проектирования предметной области в общую архитектуру бета-версии Решения.

4.3.6.3 Должна быть выполнена доработка функциональности программного компонента в части упрощения интерфейса.

Конструктор автоматизированного проектирования предметной области должен обеспечивать гибкие возможности по оперативному созданию новых сущностей (объектов) программных решений без использования программирования.

Должна быть добавлена настройка параметров типа добавляемого свойства в объект, параметры визуального отображения карточки экземпляра объекта.

Должна быть добавлена настройка карточки объекта, для динамического изменения структуры карточки и упрощения интерфейса.

4.3.6.4 Должен быть доработан комплекс обучающих материалов для обеспечения методической поддержки пользователей, начинающих эксплуатацию программного продукта, включающий в себя следующую информацию:

- типы данных;
- типы связей;
- создание новой модели
- параметры объекта:
  - создание объекта;
  - свойства объекта;
  - создание атрибута;
  - создание зависимости объекта;
- ошибки публикации.

4.3.7 Функциональные требования к компоненту (модулю) защищенного распределенного хранения данных с встроенными редакторами документов

4.3.7.1 Компонент защищенного распределенного хранения данных с встроенными редакторами документов в бета-версии Решения должен обеспечивать возможность безопасного распределенного хранения данных с возможностью ограничения доступа к файлам, их перемещения между дисками и папками, а также должен поддерживать контроль версионности материалов с журналированием событий в системе.

4.3.7.2 Компонент должен обладать единым интерфейсом доступа ко всем информационным объектам, обеспечивать эффективное распределение репозиториев по кластерным узлам, определение доступных пользователю репозиториев.

4.3.7.3 Компонент должен обеспечивать резервное копирование данных и сохранение их копий по физически распределенным кластерным узлам.

4.3.7.4 Компонент защищенного распределенного хранения данных с встроенными редакторами документов в бета-версии Решения должен быть интегрирован в единую архитектуру бета-версии Решения.

4.3.7.5 Компонент должен обеспечивать:

- хранение информации с учетом квотирования предельного размера для хранения, предельного размера загружаемого файла в виде файлов форматов *.rtf, *.doc, *.docx, *.txt, *.xls, *.xlsx, *.png, *.gif, *.jpg, *.jpeg, *.ico, *bmp, *.avi, *.mp4, *.xml, *.pdf и редактируемой метаинформации (тегов) к файлам для поиска и краткого описания;
- просмотр и редактирование файлов форматов *.rtf, *.txt, *.docx, *.xlsx встроенными средствами;
- просмотр и рецензирование файлов формата *.pdf встроенными средствами;
- обеспечение версионности документов с возможностью просмотра и восстановления предыдущих версий;

- хранение удаленных документов в «корзине»;
- шифрование хранимой информации в соответствии с требованиями ГОСТ Р 34.12-2015;
- настройку ограничений на максимальный размер диска/папки/загружаемого файла.

4.3.7.6 При работе с виртуальными дисками должны предусматриваться возможности по:

- созданию диска (стандартного или зашифрованного);
- просмотру предельного размера для диска и размещаемых на дисках файлов;
- загрузке папок и файлов на диск;
- переименованию диска;
- отображению свойств диска в отдельном окне с указанием информации по предельному размеру диска и использованному размеру (как сумма объема расположенных файлов);
- удалению диска;
- настройке отображения реестра дисков в виде дерева или в табличной форме с отображением сведений о диске: наименование, тип, объем, состояние и иные с возможностью фильтрации, сортировки и поиска.

4.3.7.7 Функции работы с папками должны предусматривать:

- создание папки;
- загрузку файлов и папок в папку с выводом сообщения об успешной или неуспешной загрузке, содержащего имя и размер папки или файла, при неуспешной загрузке – системные и иные ограничениях в загрузке;
- архивирование содержимого папки (в формат Zip)/разархивирование;
- скачивание папки в архивированном виде (в формат Zip);
- перемещение папки в другую папку или на другой диск;
- копирование папки в выбранную папку или диск;
- переименование папки;
- отображение свойств папки в отдельном окне (размер (как сумма объема расположенных документов), расположение, владелец, дата и время создания и т.д.);
- добавление папки в «избранное»;
- удаление папки;
- настройка отображения нескольких папок на диске или другой папке в виде галереи отдельных иконок.

4.3.7.8 Функции работы с документами должны предусматривать:

- предварительный просмотр файла (если предварительный просмотр недоступен, должно выдаваться соответствующее сообщение и должна предоставляться возможность скачать файл);
- редактирование/рецензирование встроенными средствами с учетом формата файла;
- перемещение документа в другую папку или диск;
- переименование документа;
- отображение свойств документа в отдельном окне;

– копирование документа в выбранную папку или диск;
– скачивание файла;
– архивирование файла документа (в формат Zip)/разархивирование;
– удаление документа;
– добавление документа в «избранное».

4.3.7.9 Компонент защищенного распределенного хранения данных должен обеспечивать возможность добавления отдельных пользователей или нескольких пользователей в хранилище данных системы. При добавлении пользователя должна быть обеспечена возможность настройки разграничений доступа пользователя к дискам, папкам, файлам.

Компонент защищенного распределенного хранения данных должен обеспечивать возможность извлечения сложных взаимосвязанных комплектов электронных документов из системы с возможностью предварительного формирования архива данных с возможностью массовой выгрузки пакетов данных.

4.3.8 Функциональные требования к компоненту (модулю) управления регламентированными процессами

4.3.8.1 Компонент управления регламентированным процессами (BPM), в основе которого лежит WorkFlow-конструктор, обеспечивает гибкие возможности по оперативной настройке регламентированных процессов, в том числе процессов формирования, согласования и утверждения отчетности и иных сопутствующих материалов. Наличие данного компонента в «Visary BI» обеспечивает значительную гибкость Решения.

Конструктор автоматизации регламентированных процессов (WorkFlow) реализует возможность конструирования регламентированных процессов без знания языков программирования с интуитивно понятным графическим интерфейсом.

Должно быть реализовано формирование и отображение карты движения в соответствии с настроенным регламентированным процессом и соответствующими комментариями ответственных лиц.

Основой Компонента согласования отчетности, определяющей базовый набор сущностей и атрибутов для описания процессов, является ее метамодель, которая должна включать:

– общее описание процессов и типов взаимосвязей;
– множество логических шагов (задач и подпроцессов) процессов, описанных в четырех измерениях: кто, что, когда и как;
– описание участников процессов, отражающее организационную модель их взаимодействия;
– навигацию (взаимодействие) между задачами и подпроцессами в процессах;
– описание приложений, представляющих собой отображения объекта на каждом шаге или этапе процесса в любой момент времени;
– релевантные (актуальные) данные, полученные в результате выполнения шага процесса, связанные с отображением и влияющие на выбор следующего шага.

4.3.8.2 В рамках мероприятий по развитию компонента управления регламентированным процессами (BPM) должна быть проведено обновление интерфейсов конструктора Workflow (рисунок 4).

Панель блоков управления бизнес-процессами должна иметь обновленную структуру, содержащую следующие разделы:

- Основное:
  - Инициализатор
  - Этап
  - Подпроцесс
  - Триггер
  - Ожидание события
  - Действия
  - Условие
  - Условие (контекст)
- Управление объектами:
  - Создание
  - Изменение
  - Удаление
  - Взаимодействие
  - Создание на основании
  - Параллельное выполнение
- Коммуникация с пользователем
- Внутренние действия

Панель инструментов должна быть оптимизирована и содержать следующие элементы:

- Импорт/экспорт схемы
- Печать
- Уменьшить масштаб
- Увеличить масштаб

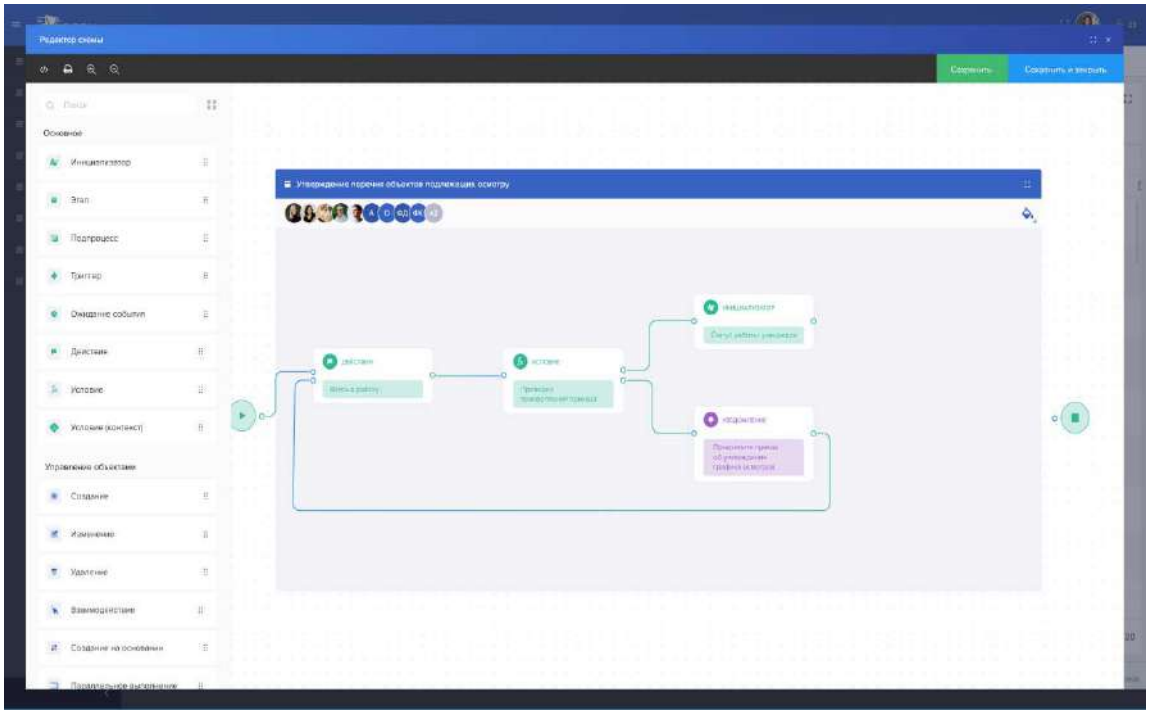

Рисунок 4 – Прототип интерфейса конструктора Workflow

Должна быть добавлена панель настроек, содержащая информацию о пользователях, группах пользователей (рисунок 5).

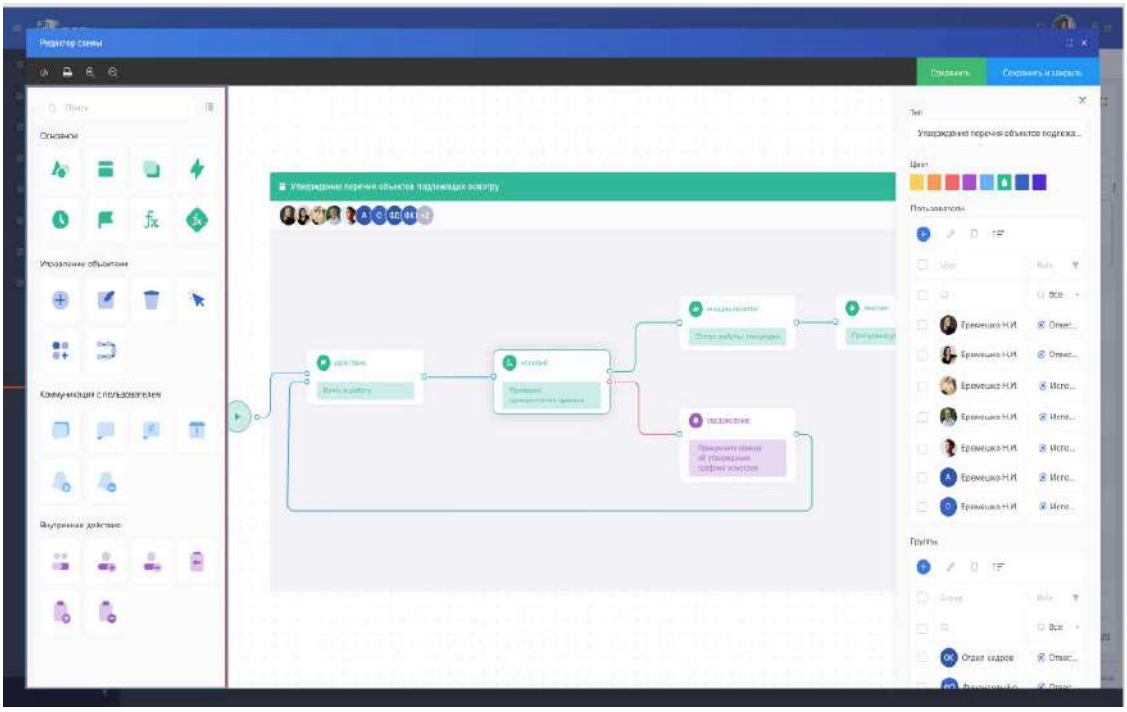

Рисунок 5 – Прототип интерфейса конструктора Workflow с панелью настроек, содержащую информацию о пользователях

4.3.8.3 Должен быть доработан интерфейс форм настройки согласования отчетности и иной сопутствующей документации для обеспечения возможности вывода на печать листов согласования.

Форма настройки согласования отчетности должна содержать следующую информацию:

- тип очереди
- название очереди
- тип согласования
- срок исполнения
- исполнитель
- длительность.

Должна быть предусмотрена возможность согласования непосредственно в листе согласования, где пользователь может при настройке доступа к объекту выбрать следующие кнопки:

- согласовать,
- отклонить,
- делегировать.

Должна быть обеспечена возможность пользователю оставлять комментарии (рисунок 6).

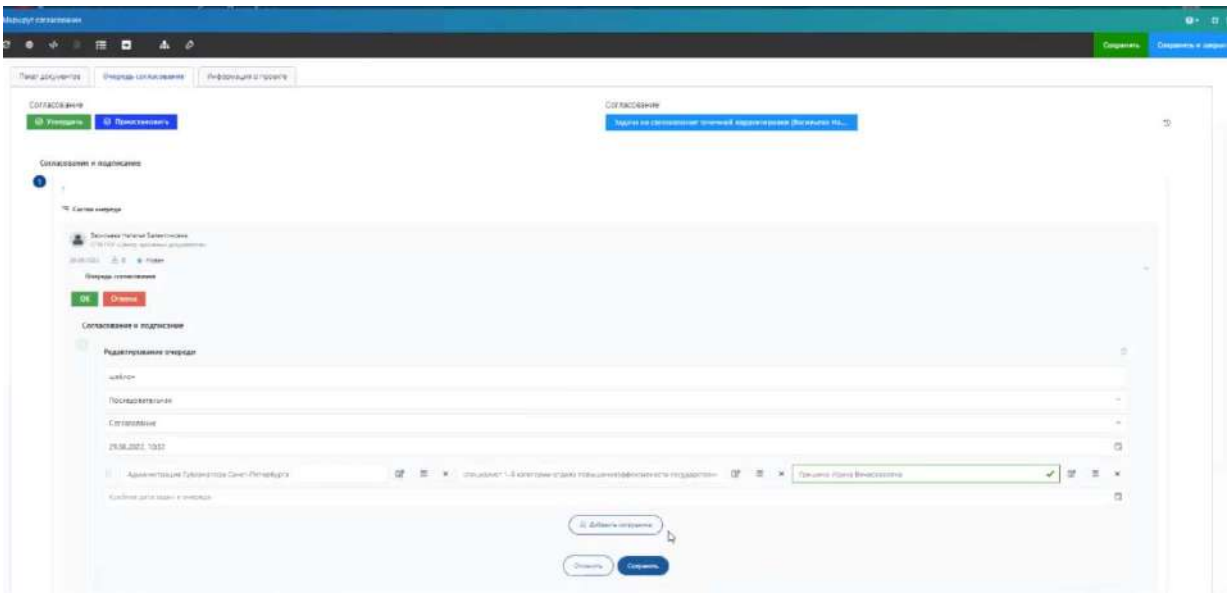

Рисунок 6 – Создание комментария

Должна быть реализована возможность создания шаблонов пользователем. Каждый пользователь может создать шаблон для себя, включая администратора и обычных пользователей. Выбор шаблона должен быть доступен в листе согласования с помощью функционального компонента «Выбор шаблона».

4.3.8.4 Должна быть проведена интеграция компонента с остальными структурными элементами Решения для формирования универсального решения многомерного анализа и визуализации больших данных АИС «Визари» (бета-версии Решения).

4.3.9 Функциональные требования к компоненту (модулю) управления задачами

4.3.9.1 В рамках реализации Проекта должна быть оптимизирована структура карточки задачи. Должен быть сформирован банк карточек задач в зависимости от категории задачи. Задачи должны быть подразделены на следующие категории:

- По статусу задачи:
  - новая
  - назначена
  - в работе
  - на тестировании
  - на согласовании
  - согласована
  - отклонена
  - решена.
- По исполнителю:
  - я исполнитель
  - мои задачи на исполнении.
- По автору.
- В зависимости от роли в команде:
  - куратор
  - наблюдатель
  - соисполнитель
  - соавтор.

- По приоритету:
  - высокий
  - средний
  - низкий.
- По типу задачи:
  - аналитика
  - дизайн
  - разработка
  - BI.

4.3.9.2 При проведении доработки компонента (модуля) управления задачами должны быть сохранены функциональные требования компонента по работе с массивами задач альфа-версии Решения.

4.3.9.3 Модуль должен являться основным инструментом взаимодействия руководителей и подчиненных. Процесс назначения, исполнения и контроля задач и поручений становится эффективным и прозрачным как для управленческого персонала предприятия, так и для исполнителей.

Модуль управления задачами должен обеспечивать автоматизацию процессов постановки задач с указанием ответственных и сроков исполнения, а также обеспечивать контроль их исполнения.

Модуль управления задачами должен автоматизировать:

- распределение задач (например, между сотрудниками Заказчика, подведомственных учреждений и подрядчиками);
- дифференциацию задач по приоритетам и времени исполнения;
- уведомление о критическом состоянии задачи;
- контроль выполнения задач (например, между сотрудниками Заказчика, подведомственных учреждений и подрядчиками).

Модуль управления задачами должен обеспечивать:

- создание задач в ручном режиме;
- автоматическое создание шаблонных задач при выполнении проектов или запуске регламентированных процессов только для пользователей, указанных в них исполнителями;
- разделение задач на категории (входящие, исходящие, выполненные);
- выбор приоритета задачи;
- отображения прогресса выполнения задачи в процентах;
- выбор исполнителя задач;
- отображение пользователя, назначившего (создавшего задачу);
- возможность прикрепления к задаче файлов любого типа;
- возможность связывания задачи с любым информационным объектом системы;
- ведение истории изменения задачи;
- напоминания о просроченных задачах;
- цветовое ранжирование задач по срочности (например, срочность подготовки проведения закупок/поставок).

Модуль управления задачами должен обеспечивать следующие возможности:

- создания задачи с привязкой к конкретному объекту;

- интеграция с компонентом управления регламентными процессами;
- интеграция с компонентом уведомлений. Исполнитель задачи получает уведомление о назначенных на него новых задачах;
- выбора ответственного исполнителя среди пользователей системы. Предусмотрена возможность назначения задачи на группу пользователей (отдел). При этом пользователи, входящие в число исполнителей, обладают возможностью взять задачу на исполнение. В этом случае ответственный исполнитель изменяется на конкретного пользователя системы;
- построения отчетов по завершенным и открытым как входящим, так и исходящим задачам;
- выбора даты начала и плановой даты завершения задачи из календаря;
- автоматического расчета время, оставшееся до завершения задачи. В случае просрочки рассчитывается время просрочки. Просроченные задачи визуально отделены от остальных путем цветовой дифференциации в общем стеке задач исполнителя.

Возможность выбора приоритета задачи для отделения наиболее важных и срочных задач. Возможные приоритеты:

- высокий;
- нормальный;
- низкий.

Возможность цветовой дифференциации любым цветом из матрицы RGBA.

Возможность ведения статусов задач. В базовом варианте предусмотрены следующие статусы:

- Новая;
- В работе;
- Отклонена;
- Закрыта;
- Решена;
- Назначена.

Список статусов может быть дополнен:

- предусмотрена возможность ведения истории задачи с указанием исполнителей, дат совершения действий над задачей, а также комментария пользователя;
- возможность прикрепления файлов любых форматов к задаче.

В компоненте управления задачами должен быть реализован объектно-ориентированный конструктор схем согласования задач. Конструктор позволяет настраивать:

- наименование схемы движения задач;
- подробное описание схемы;
- статус задачи для каждого этапа;
- действие, выполняемое на каждом этапе;
- параметры перехода:
  - начальный этап;
  - конечный этап;
  - прогресс задачи;

    - необходимость ввода комментария;
- параметры уведомления:
    - роль пользователя, для которого создается уведомление;
    - текст уведомления;
- параметры отображения полей задачи.

Конструктор должен включать функционал по созданию и редактированию схем согласования задач в графическом (визуальном) режиме.

В Компоненте должна быть реализована функция динамического распределения нагрузки с учетом текущей занятости и требований официальных регламентов. Распределение нагрузки позволяет распределять работы между исполнителями.

Компонент управления задачами должен интегрироваться с компонентом управления регламентированными процессами и компонентом уведомлений для автоматизации назначения и контроля действий, требуемых от исполнителя, и выполнять следующие функции:

- создание новой задачи в указанной группе;
- просмотр и изменение параметров существующей задачи;
- выбор ответственного исполнителя, соисполнителей и наблюдателей из числа пользователей Системы;
- задание плановой даты начала и завершения задачи;
- автоматический расчет времени, оставшегося до завершения задачи, с уведомлением о просрочке;
- выбор приоритета задачи (высокий, низкий, нормальный);
- отображение всех задач пользователя в общем стеке задач, с возможностью сортировки и фильтрации по простому и составному ключам;
- определение статуса задачи;
- ведение истории задачи (сохранение даты действия, исполнителя и комментариев пользователя);
- прикрепление к задаче файлов любых форматов и текстового описания;
- сопоставление задаче категории новой или существующей;
- сопоставление текущей задаче родительской из числа предшествующих задач с установкой иерархической связи;
- изменение даты завершения задачи и смена исполнителя;
- создание новой схемы движения задачи или привязка уже существующей.

Визуальный объектно-ориентированный конструктор схем движения задач должен позволять:

- создавать схемы, посредством размещения графических примитивов на рабочем поле конструктора;
- назначать схеме уникальное имя и сохранять ее для повторного использования;
- открывать и редактировать ранее созданные схемы;
- сопоставлять статусы задачи для каждого этапа;
- создавать связи между этапами, ассоциированные с выполняемыми действиями по изменению статуса;
- определять параметры перехода (наименование действия, начальный и конечный этап, прогресс задачи);

– настраивать уведомления о совершении действий пользователю в соответствии с его ролью;
– настраивать параметры отображения полей задач;
– выполнять сортировку и фильтрацию действий по заданному критерию.

4.3.9.3 Должна быть проведена интеграция компонента в архитектуру бета-версии Решения.

4.3.9.4 Должна быть доработана модель расчета текущего показателя выполнения задач на основании математических алгоритмов.

Компонент управления задачами предназначен для автоматизации процессов постановки задач и контроля их исполнения. Компонент управления задачами содержит конструктор схемы движения задач, который обеспечивает возможность четкого отображения процесса перемещения задачи между этапами и функциональных механизмов, влияющих на выбор траектории перемещения под воздействием внешних и внутренних факторов. Конструктор обеспечивает визуализацию зависимостей между задачами для чего необходимо обеспечить соответствие схемы следующей математической модели.

Схема движения S задачи представляет собой тройку S =, где M – семейство модификаций иерархических сетей Петри, каждая из которых соответствует множеству траекторий развития задачи, $M = \{C_1, C_2, \ldots, C_i, \ldots, C_q\}$, $z \in [1, q]$ – индекс корневой сети, с которой начинается функционирование схемы, μ – начальная маркировка Cz, определяющая исходный статус задачи. Каждая сеть C_i представляет собой шестерку Ci =, где $P = \{p_1, p_2, \ldots, p_i, \ldots, p_n\}$ – множество позиций сети C_i, $T = \{t_1, t_2, \ldots, t_i, \ldots, t_m\}$ – множество переходов, I – входная функция, O – выходная функция, $A = \{A_1, A_2, \ldots, A_i, \ldots, A_k\}$ – множество активационных функций сети C_i, эквивалентных действию по установке начального состояния задачи, $D = \{D_1, D_2, \ldots, D_i, \ldots, D_l\}$ – множество деактивационных функций, эквивалентных действию по завершению выполнения задачи.

Структурно сеть C_i представляет собой множество позиций, соединенных между собой переходами. Позиция характеризуется числом содержащихся в ней фишек, инкапсулирующих условие перевода задачи на следующий этап, а переходы выполняют функции перемещения фишек по сети. Посредством фишек регламентируется подмножество операций по изменению статуса задачи, доступных в текущей ситуации. Переход может являться или атомарным действием (независимый переход), или сложной последовательностью действий, описываемых иерархически подчиненной сетью (подчиненный переход). Запуск перехода t_i возможен только в том случае, когда в проассоциированных с ним посредством входной функции I позициях (входные позиции) содержится достаточное количество фишек. Результатом успешного запуска перехода является увеличение на определенное число количества фишек в позициях, проассоциированных с переходом t_i посредством выходной функции O (выходные позиции). Выходная функция определена на двумерном множестве, являющемся декартовым произведением пространства переходов на пространство деактивационных функций иерархически подчиненной сети (при ее существовании). В случае неуспешного запуска перехода t_i, возможного только в ситуации использования иерархически подчиненной сети, увеличение количества фишек в соответствующих позициях не производится. При этом в зависимости от значения признака завершения, определяемого деактивационной функцией, количество фишек во входных позициях перехода t_i может оставаться прежним или уменьшаться. Позиции соответствуют статусам задачи, а переходы

– работам, оказывающим влияние на прогресс задачи и изменяющим ее состояние. Активационная функция A_i сопоставляет сети C_i некоторую начальную маркировку μ, определяющую стартовое количество фишек в позициях множества P. Деактивационная функция D_i представляет собой пару D_i = , где μ` – некоторая маркировка, достижение которой является сигналом к завершению работы сети, а двойка r = – определяет итог завершения работы сети (результат s ∈ {0, 1}, где 0 – неуспешное завершение, 1 – успешное завершение, а признак результата f ∈ {0, 1, 2}, где 0 – успешное завершение, 1 – неуспешное завершение без изменения количества фишек во входных позициях, 2 – неуспешное завершение с уменьшением количества фишек во входных позициях). Срабатывание деактивационной функции сети эквивалентно достижению одного из конечных состояний задачи, то есть ее выполнению (в случае подчиненной сети – выполнению этапа).

Таким образом, произвольный переход ti представляет собой тройку ti = , где v ∈ {0, 1} – тип перехода (0 – независимый, 1 – подчиненная сеть), C` – иерархически подчиненная сеть, используемая для определения успешности запуска перехода ti, A` – активационная функция сети C` (A` ∈ A из C`). C` = ∅ и A` = ∅ в случае v = 0. Пример рассматриваемой сети Ci представлен на рисунке 7.

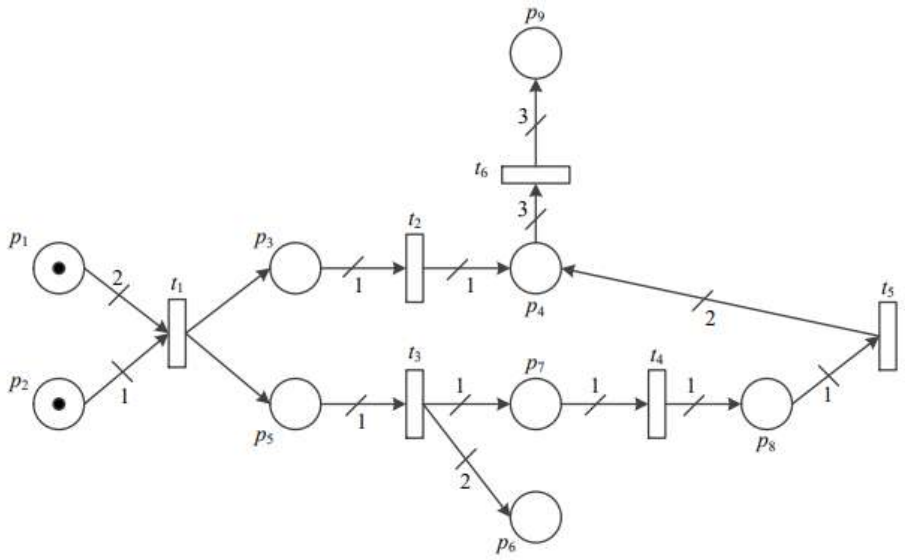

Рисунок 7 – Пример используемой модификации иерархической сети Источник: составлено авторами

Позиции p_1, p_2, …, p_9 соответствуют этапам движения задачи (этап 1, этап 2, ...). Переходы t_1, t_2, …, t_6 эквивалентны работам, оказывающим влияние на прогресс задачи и изменяющим ее статус. Переход t_1 является подчиненным и ему соответствует иерархически вложенная сеть Петри. Переходы t_2 – t_6 независимы. На исходящих дугах указано количество фишек, перемещающееся в выходные позиции в случае активации перехода. Для перехода t_1 количество фишек в выходных позициях будет определяться деактивационной функцией вложенной сети.

4.3.9.5 После интеграции данного компонента с компонентом аналитической отчетности необходимо разработать банк предустановленных отчетных форм, наглядно отображающих степень эффективности работы различных сотрудников и подразделений в целом.

Банк предустановленных отчетных форм должен содержать следующие разделы:

– эффективность работы сотрудников

– эффективность работы подразделений
– статистика по проектным задачам.

4.3.10 Функциональные требования к компоненту (модулю) уведомлений

4.3.10.1 В рамках реализации Проекта должна быть проведена интеграция компонента уведомлений с другими элементами бета-версии Решения.

Модуль уведомлений должен быть интегрирован с модулем бизнес-процессов (регламентированных процессов) и модулем расписаний (планировщиком задач).

Принцип работы модуля уведомлений основан на том, что при возникновении любого события, связанного с пользователем, такого как создание задачи, завершение согласования документа или его возврат на доработку и т.п., то есть при переходе бизнес-процесса от одного участника к другому у исполнителя автоматически формируется уведомление, представляющее графическое, текстовое и звуковое оповещение.

Компонент уведомлений обеспечивает автоматизацию следующих процессов:

– оповещение по электронной почте, в личных кабинетах пользователей системы о событиях, связанных с процессами согласования отчетности;
– рассылка сообщений иного тематического содержания.

При прочтении уведомления, а также при ознакомлении с произошедшим событием (например, при прочтении новой задачи), уведомление автоматически переносится в архив.

Компонент уведомлений обеспечивает рассылку уведомлений по электронной почте тем пользователям, которые выбрали для себя данную функцию. Для предотвращения перехвата информации, содержащейся в уведомлениях, реализована возможность использования e-mail через SSL.

4.3.10.2 Должна быть проведена доработка шаблона уведомлений, генерируемых на почту пользователям бета-версии Решения в виде различных интерактивных отчетов с перечнем задач (поставленных, просроченных и от руководителя) с возможностью перехода в систему при нажатии на соответствующий раздел отчета.

Для рассылки сообщений (уведомлений) пользователям на электронную почту по результатам выполнения шагов схем бизнес-процессов (например, результаты процессов рассмотрения, согласования, голосования, утверждения и т.д.) или о других различных событиях используются настройки почтового сервера.

Для изменения настроек необходимо выбрать поле для изменения.

Для изменения полей с вводом данных необходимо выбрать поле для изменения и ввести желаемое значение.

Для изменения полей «Включено/выключено» необходимо установить переключатель в нужное положение.

4.3.11 Функциональные требования к компоненту (модулю) поиска

4.3.11.1 В бета-версии Решения должен быть обеспечен полный перенос функционала компонента альфа-версии Решения в части переноса функциональности по поиску, сортировке и фильтрации массива данных бета-версии Решения для формирования соответствующей выборки.

Компонент поиска должен позволять выполнять поиск как по отдельным ключевым словам (быстрый поиск), так и составлять сложные запросы с логическими связями (расширенный поиск).

В режиме быстрого поиска пользователь должен иметь возможность задать искомую комбинацию символов. По умолчанию поиск введенной комбинации символов

осуществляется по всем материалам, открытым для поиска пользователю с учетом связанности информации.

В режиме расширенного поиска компонент должен позволять пользователю сохранить настроенное логическое выражение в шаблон с целью последующего повторного использования. Расширенный поиск должен обеспечивать возможность включения поиска по любым атрибутам объектов в любой их комбинации, указываемой в виде логических выражений с применением таких операторов объединения как «И» и «ИЛИ», а также скобок, указывающих очередность и приоритетность применения операторов.

4.3.11.2 В отношении шаблона поиска должна быть реализована возможность осуществлять следующие операции: сохранить, изменить, удалить, применить.

4.3.11.3 Компонент поиска должен позволять осуществлять каскадный поиск (поиск в результатах поиска).

При отображении списков данных обеспечены следующие возможности:

- сортировка по любому из атрибутов в списке;
- установка сортировки по умолчанию;
- фильтр по любому из атрибутов в списке;
- выбор из перечня преднастроенных и сохранение собственных фильтров;
- возможность выгружать список (частично и/или полностью) из Компонента поиска в форматах xls, xlsx;
- индивидуальная настройка параметров отображения для каждого пользователя.

4.3.11.4 Компонент поиска должен позволять оперативно строить сложные запросы, по которым информация, представленная в системе в виде информационных объектов, отбирается в списки информационных объектов, отвечающих заданным критериям поиска. При этом в поиске должна учитываться вся организация модели данных бизнес-процессов и связанность информации в ней.

При открытии реестра объектов в виде таблицы должен быть предусмотрен поиск, фильтрация, сортировка и группировка по полям (колонкам) таблицы, возможность настройки порядка вывода полей и группировки.

В модуле (компоненте) поиска предусмотрена настройка перечня выводимых на экран полей таблицы для каждого пользователя (индивидуальный поиск).

Компонент поиска позволяет настраивать фильтры не только для отображаемых полей, но и для скрытых полей мнемоник, а также связанных объектов.

Разработать экранные формы для эффективной работы с фильтрами. На основной панели инструментов (позволяющей выполнять операции создания, удаления, редактирования записей) каждого окна, содержащего множества объектов, представленных в табличной или списочной формах, представлено поле выбора фильтра.

При создании фильтра пользователем задаются следующие параметры:

- наименование фильтра (краткое описание);
- описание фильтра (расширенное описание);
- критерии фильтрации данных.

Реализованы следующие функции:

- создание фильтров;
- редактирование фильтров;
- сортировка фильтров;

– экспорт фильтров в формат xlsx.

4.3.11.5 Должна быть проведена интеграция компонента поиска в общую архитектуру бета-версии Решения.

4.3.12 Функциональные требования к компоненту (модулю) безопасности

Компонент безопасности (защиты информации) должен быть разработан с учетом требований стандарта ГОСТ Р 51583-2014 «Защита информации. Порядок создания автоматизированных систем в защищенном исполнении. Общие положения».

Компонент безопасности должен представлять единую точку входа Интернет / Интранет (извне и внутри корпоративной сети заказчика) для всех ответственных лиц заказчика, иных пользователей системы, имеющих отношение к процессам подготовки, согласования отчетности.

Компонент безопасности должен представлять инструменты управления учетными данными пользователей, которые позволят настраивать роли, объединять их в группы и разграничивать доступ к функциям для конкретной роли, пользователя и группы пользователей, а также осуществление распределения видимости объектов по ролям, пользователям, группам пользователей, группам ролей.

Компонент безопасности должен представлять механизмы:

– противодействия попыткам несанкционированного доступа к защищаемой информации;
– пресечения попыток несанкционированного считывания, изменения и уничтожения данных;
– контроля и восстановления целостности данных.

Компонент безопасности должен обеспечивать:

– возможность создания групп доступа и групп пользователей с заданными правами доступа, к которым относятся:
  – полный доступ к внутренним и внешним разделам альфа-версии/прототипе Решения;
  – право на создание и редактирование задач, справочников, документов;
  – право на удаление задач, справочников, документов;
  – право на просмотр задач, справочников, документов;
– возможность назначения прав доступа к задачам, справочникам в соответствии с иерархической организационной структурой функциональных заказчиков, включая определение прав доступа на основе принадлежности к определенному подразделению;
– разделение прав доступа по задачам, справочникам, папкам, объектам, по роли пользователя, по принадлежности к подразделению и занимаемой пользователем должности;
– возможность идентификации и проверки подлинности субъектов доступа при входе в Систему на основе альфа-версии/прототипе Решения (обеспечение доступа только зарегистрированных пользователей, прошедших процедуру идентификации и аутентификации);
– регистрацию (журналирование) входа субъектов доступа в Систему на базе альфа-версии/прототипа Решения (при регистрации указываются параметры: дата и время входа, идентификатор субъекта);

- доступ к функциям чтения, записи, удаления, создания объектов и навигации по объектам настраивается исключительно пользователем, имеющим соответствующие права;
- реализацию функции, позволяющей вести мониторинг работы пользователей, выполнять автоматический контроль нормального состояния функционирования системы, задействовать функции отслеживания и фиксации нештатных и аварийных ситуаций, создавать механизмы, обеспечивающие возможность восстановления работоспособности после отказов, сбоев или аварий;
- внесение в журнал таких событий, как добавление, изменение, удаление данных пользователями;
- изменение записей журнала доступно только администратору;
- доступ к функции блокировки учетной записи администратора при управлении учетными записями администраторов без удаления из базы данных;
- интеграцию компонента (модуля) безопасности в архитектуру бета-версии Решения для обеспечения комплексной защиты, включающей в себя защиту от сбоев, ведущих к потере информации, а также защиту от неавторизованного создания или уничтожения данных, обрабатываемой и хранимой в программных продуктах информации, в том числе персональных данных пользователей и иной конфиденциальной информации.

4.3.13 Функциональные требования к компоненту (модулю) администрирования

Компонент администрирования должен позволять администратору настраивать параметры функционирования ПО в целом или отдельных компонентов:

- настройка видимости разделов административной панели;
- настройка компонента безопасности: уровней логирования, аудита и журналирования;
- настройка требований к учетным записям;
- настройка интеграции с внешними системами.

Компонент администрирования должен обеспечивать удаленное журналирование операций, в том числе действий пользователей.

Компонент администрирования должен реализовать возможность анализа журнала на предмет выявления потенциальных нарушений доступа, фильтрации и сортировки журнала.

Компонент администрирования должен обеспечивать организацию доступа пользователей к информации в соответствии с правами доступа, а также реализовывать механизм разграничения доступа к функциям системы в зависимости от роли пользователя в системе. Должна быть реализована возможность создавать и редактировать пользовательские роли. Роль включает в себя перечень доступных пользователю функций и прав доступа к информации. При настройке (создании и редактировании) роли, реализована возможность указывать доступные отчеты, в том числе встроенные отчеты.

4.3.13.1 В ходе реализации проекта должна быть актуализирована рабочая и техническая документация с учетом нового функционала Решения.

4.3.13.2 Интерфейс Решения должен быть локализован для работы на английском языке.

4.3.13.3 На английском языке должны быть разработаны переведены следующие эксплуатационные и технические документы:

- архитектура (дизайн) программного обеспечения (software architecture design document);
- документация по API (API documentation);
- руководство конечного пользователя (end-user guide);
- руководство системного администратора (system administration documentation);
- инструкция по установке (installation instruction).

4.4 Требования к видам обеспечения доработанной бета-версии Решения

4.4.1 Требования к математическому обеспечению

При создании математического обеспечения бета-версии Решения Исполнитель должен использовать расчетные алгоритмы, которые обеспечат реализацию и работу всех реализуемых функций.

При описании расчетных алгоритмов должны быть указаны:

- назначение и характеристика алгоритма;
- математическое описание;
- алгоритм решения.

При аналитической обработке должны использоваться методы статистической обработки и анализа данных:

- фильтрации данных (по задаваемому пользователю набору и комбинациям фильтров);
- суммирования (в том числе по заданной временной и объектной выборке);
- определения среднего (арифметического, геометрического, средневзвешенного и т.п.);
- определения параметров статистических распределений, в том числе индексов динамики процессов и расчета трендов;
- расчета коэффициентов корреляции между различными показателями.

4.4.2 Требования к информационному обеспечению

4.4.2.1 Информационное обеспечение должно представлять собой совокупность сообщений, форм, документов, а также информационных баз данных (массивов информации), включая нормативно-справочную информацию. Состав информационного обеспечения должен быть полным и достаточным для решения задач заказчика.

Информационное обеспечение должно быть достаточным для поддержания всех автоматизируемых функций объекта автоматизации.

4.4.2.2 Информационное обеспечение должно соответствовать следующим требованиям:

- масштабируемость, без ограничения на количество подключаемых пользователей;
- структурированное хранение информации, с учетом поддержки версионности и истории изменения данных, с возможностью хранения истории изменения данных. Состав данных должен быть определен на этапе технического проектирования;
- непротиворечивость и целостность хранимой информации.

4.4.2.3 Состав, структура и способы организации данных должны быть детализированы для определенного проекта, реализуемого на базе бета-версии Решения.

В состав информационного обеспечения Решения должны входить:

- данные, вносимые пользователями;
- данные, загружаемые из внешних информационных систем;
- данные, формируемые в объекте автоматизации в рамках автоматизируемых процессов;
- системные данные, в частности настройки, используемые компонентом.

4.4.2.4 С точки зрения доступа в бета-версии Решения должно быть предусмотрено хранение, ведение и предоставление сведений ограниченного доступа и сведений, доступ к которым не ограничен. Доступ к данным ограниченного доступа должен быть предоставлен только авторизованным пользователям с учетом их полномочий.

4.4.2.5 Для хранения данных в бета-версии Решения должна быть использована открытая СУБД PostgreSQL.

4.4.2.6 Для индексации и кеширования должны быть использованы NoSQL СУБД Redis и поисковая база данных Elasticsearch.

4.4.2.7 Для обмена данными с внешними должны быть использованы следующие адаптеры:

- ЕСИА;
- СМЭВ;
- Active Directory;
- технология SOAP/REST.

4.4.3 Требования к программному обеспечению

Бета-версия Решения должна быть реализована с использованием следующего технологического стека:

- ОС Linux или Astra Linux;
- СУБД PostgreSQL, Redis;
- сервер приложений ASP.NET Core;
- веб-сервер Nginx;
- экранные формы должны быть разработаны в соответствии с условиями предоставления пользователям полноценной работы в следующих браузерах для персональных компьютеров:
  - Google Chrome версии 66.0 и выше;
  - Mozilla Firefox версии 59.0 и выше;
  - Opera 53.0 версии и выше;
  - Яндекс.Браузер версии 10 и выше.

4.4.4 Требования к лингвистическому обеспечению

Должны применяться языки программирования, обеспечивающие работу в используемой СУБД, а также Unix-подобных ОС.

В качестве языка манипулирования данными должен быть использован язык структурированных запросов SQL (Structured Query Language). Выбор конкретного языка манипулирования данными зависит от используемой СУБД.

Для разработки веб-интерфейса должны быть использованы такие средства, как HTML5 и CSS.

Для разработки интерактивных веб-страниц должны быть использованы такие технологии, как ajax и json, средства языка JavaScript и библиотеки JQuery.

Интерфейс пользователя клиентского ПО должен быть на русском и английском языках.

Размещение информации в бета-версии Решения должно осуществляться на русском и английском языках.

Основными языками взаимодействия бета-версии Решения с обслуживающим персоналом (с учетом ограничений, налагаемых системным и базовым программным обеспечением) должны быть русский и английский языки.

Подсистемы бета-версии Решения должны обеспечивать:

- интерфейс взаимодействия с пользователем на русском и английском языках;
- предоставление пользователю отчетов на русском и английском языках;
- формирование текстов сообщений об ошибках и мониторинговых сообщений (кроме сообщений системного и базового программного обеспечения) на русском и английском языках. Сообщения должны быть информативными и понятными пользователю.

При публикации документов должна обеспечиваться возможность чтения документов потребителями информации в кодировке UTF-8.

4.4.5 Требования к техническим средствам

Бета-версия Решения может быть установлена и запущена на физическом или виртуальном сервере, имеющем следующие характеристики:

- количество процессоров: 2;
- количество ядер процессора: 4;
- частота процессора: 1,3 ГГц;
- объем жесткого диска 2500 ГБ;
- объем оперативной памяти: 8 ГБ;
- скорость подключения 100 Мбит/с.

Для эффективной работы приложения на основе платформы «Визари» должен использоваться физический или виртуальный сервер, имеющий следующие характеристики:

- количество процессоров: 4;
- количество ядер процессора: 4;
- частота процессора: 2,4 ГГц;
- объем жесткого диска 21000 ГБ;
- объем SSD 2*256 ГБ;
- объем оперативной памяти: 32 ГБ;
- скорость подключения 1000 Мбит/с.

5 Состав и содержание работ по доработке Решения

5.1 Требования к детализированной модели монетизации Решения

В рамках детализированной модели монетизации бета-версии Решения:

- изучена динамика и тенденции развития мирового и российского рынков BI;
- определен перечень зарубежных и российских BI-решений для анализа;
- выявлены критерии сравнения BI-решений на основе исследования мнения пользователей BI-систем;
- проведена сравнительная характеристика выбранных зарубежных и российских BI-решений по выявленным критериям;
- описана модель монетизации бета-версии Решения с учетом результатов проведенного анализа зарубежных и российских BI-решений.

При проведении последовательного и глубокого анализа рынка существующих российских и иностранных BI-решений для выявления лучших практик реализации функциональных возможностей BI-решений для наиболее оптимального формирования детализированных требований к функциональности бета-версии Решения.

5.2 Требования расчетным алгоритмам, используемым для Решения

При создании математического обеспечения бета-версии Решения Исполнитель должен использовать расчетные алгоритмы, которые обеспечат реализацию и работу всех реализуемых функций.

При описании расчетных алгоритмов должны быть указаны:

- назначение и характеристика алгоритма;
- математическое описание;
- алгоритм решения.

Расчетные алгоритмы должны быть представлены в документе «Алгоритмическая документация, содержащая описание используемых в бета-версии Решения математических методов».

5.3 Порядок разработки и этапы работ по доработке Решения

Этапы и сроки реализации Проекта уточнены в таблице 5.

Таблица 5 – Этапы и сроки реализации Проекта

№	Описание мероприятия	Результат (в т.ч. количественно измеримый)	Подтверждающие документы	Сроки реализации Проекта
Этап 1				
1.1	Мониторинг аналогичных российских и иностранных BI-решений. Анализ решений конкурентов с целью выявления лучших практик для совершенствования	Проведен последовательный и глубокий анализ аналогичных существующих российских и иностранных BI-решений, определена наиболее востребованная со стороны пользователей функциональность (сильные стороны	Детализированное описание модели монетизации Решения по итогам оценки условий, предлагаемых разработчиками других программных продуктов по аналитике и визуализации данных, а также внутренних	С 01.03.2022 по 31.07.2022

№	Описание мероприятия	Результат (в т.ч. количественно измеримый)	Подтверждающие документы	Сроки реализации Проекта
	собственного Решения	существующих программных продуктов).	расчетов себестоимости доработки Решения	
1.2	Формирование детализированных требований к доработке (разработке) программного обеспечения Решения	Формализованы детализированные функциональные требования к доработке (разработке) программного обеспечения Решения. Формализованы требования к используемому математическому аппарату. Разработана презентация концепции реализации Проекта.	– Детализированное техническое задание; – Алгоритмическая документация, содержащая описание используемых в бета-версии Решения математических методов; – Презентация концепции реализации Проекта	
Этап 2				
2.1	Доработка компонентов альфа-версии Решения	Проведена доработка (развитие функциональности) альфа-версии Решения согласно требованиям Детализированного технического задания, а именно доработка компонентов: – нормативно-справочной информации; – аналитической обработки данных; – визуализации аналитической отчетности; – интеграционной шины; – предобработки информации ETL; – автоматизированного проектирования предметной области; – защищенного распределенного хранения данных с встроенными редакторами документов; –управления регламентированными процессами; – управления задачами;	– Ведомость технического проекта; – Пояснительная записка к техническому проекту; – Программа и методика предварительных испытаний.	С 01.08.2022 по 28.02.2023

№	Описание мероприятия	Результат (в т.ч. количественно измеримый)	Подтверждающие документы	Сроки реализации Проекта
		– уведомлений; – поиска; – безопасности; – администрирования.		
Этап 3				
3.1	Тестирование бета-версии Решения	Проведено функциональное и нагрузочное тестирование бета-версии Решения	– Протокол предварительных испытаний	С 01.03.2023 по 31.08.2023
3.2	Доработка программного обеспечения бета-версии Решения по итогам тестирования	Проведена доработка программного обеспечения Решения, устранены замечания, формализованные в Протоколе предварительных испытаний. Разработана пользовательская эксплуатационная документация.	– Акт о доработке Решения по результатам предварительных испытаний; – Актуальное руководство пользователя бета-версии Решения; – Актуальное руководство администратора бета-версии Решения; – Актуальное руководство для интегратора, содержащее рекомендации по развертыванию и интеграции бета-версии Решения, организации мониторинга его работоспособности; – Программа и методика приемочных испытаний; – Акт о завершении опытной эксплуатации и готовности бета-версии Решения к приемочным испытаниям; – Полный актуальный дистрибутив бета-версии Решения, записанный на внешний носитель;	

№	Описание мероприятия	Результат (в т.ч. количественно измеримый)	Подтверждающие документы	Сроки реализации Проекта
			– Протокол приемочных испытаний.	
3.3	Регистрация программного обеспечения	Зарегистрированные в Роспатенте свидетельства на разрабатываемое программное обеспечение. Разработанное программное обеспечение включено в Единый реестр российских программ для электронных вычислительных машин и баз данных	– Свидетельство о государственной регистрации программы для ЭВМ в Роспатенте; – Приказ Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации о включении сведений о программном обеспечении в Единый реестр российских программ для электронных вычислительных машин и баз данных	
3.4	Коммерциализация бета-версии Решения	Объем финансовых средств, полученных от реализации проектов по оказанию услуг по адаптации и внедрению Решения.	Совокупный доход, получаемый от продаж /допродаж лицензий, а также доход от реализации проектов по оказанию услуг по адаптации и внедрению Решения. (Более 50 000 000,00 рублей с НДС в 2023 году, более 110 000 000,00 рублей с НДС в 2024 году)	

5.4 Требования к организациям-соисполнителям работ

Привлечение соисполнителей для целей доработки Решения не предусмотрено.

6 Порядок контроля и приемки результатов работ по доработки Решения

6.1 Виды, состав и методы испытаний доработанного Решения и ее составных частей

6.1.1 Требования к видам испытаний доработанного Решения

Для проверки выполнения функций программного обеспечения бета-версии Решения устанавливаются следующие виды испытаний:

- предварительные испытания;
- опытная эксплуатация;
- приемочные испытания.

Проверка работоспособности программного обеспечения бета-версии Решения, включая проведение предварительных и приемочных испытаний, должна осуществляться на тестовом стенде Исполнителя. Требования к архитектуре, техническому и программному обеспечению тестового стенда представлены в разделе 4.4.3 и 4.4.5.

Предварительные испытания проводят для определения работоспособности программного обеспечения бета-версии Решения и определения возможности ввода бета-версии Решения в опытную эксплуатацию. Испытания должны проводиться согласно документу «Программа и методика предварительных испытаний», разработанному Исполнителем.

По итогам доработки бета-версия Решения оформляется Акт о завершении опытной эксплуатации и готовности бета-версии Решения к приемочным испытаниям.

Приемочные испытания проводятся с целью установления соответствия представленных Исполнителем выполненных работ на соответствие требованиям Технического задания.

7 Требования к документированию

7.1 Перечень документов, предъявляемых по окончании соответствующих этапов работ

На первом этапе Проекта должен быть подготовлен комплект следующих документов:

- Детализированное описание модели монетизации Решения;
- Алгоритмическая документация, содержащая описание используемых в бета-версии Решения математических методов;
- Презентация концепции реализации Проекта.

На втором этапе выполнения Проекта должен быть подготовлен комплект следующих документов:

- ведомость технического проекта;
- пояснительная записка к техническому проекту;
- программа и методика предварительных испытаний.

На третьем этапе выполнения Проекта должен быть подготовлен комплект следующих документов:

- протокол предварительных испытаний;
- акт о Доработке решения по результатам предварительных испытаний;
- руководство пользователя бета-версии Решения;
- руководство администратора бета-версии Решения;
- руководство для интегратора, содержащее рекомендации по развертыванию и интеграции бета-версии Решения, организации мониторинга его работоспособности;
- программа и методика приемочных испытаний;
- акт о завершении опытной эксплуатации и готовности бета-версии Решения к приемочным испытаниям;
- полный актуальный дистрибутив бета-версии Решения, записанный на внешний носитель;
- протокол приемочных испытаний;
- свидетельство о государственной регистрации программы для ЭВМ в Роспатенте;
- приказ Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации о включении сведений о программном обеспечении в единый реестр российских программ для электронных вычислительных машин и баз данных.

На английский язык должны быть разработаны и переведены следующие технические и эксплуатационные документы:

- архитектура (дизайн) программного обеспечения (software architecture design document);
- документация по API (API documentation);
- руководство конечного пользователя (end-user guide);
- руководство системного администратора (system administration documentation);
- инструкция по установке (installation instruction).

8 Требования к регистрации ПО бета-версии Решения

По итогам разработки бета-версии Решения Исполнитель должен обеспечить государственную регистрацию бета-версии Решения в Роспатенте. Также Исполнитель обязан провести регистрацию бета-версии Решения в Едином реестре российских программ для электронных вычислительных машин и баз данных.

По итогам регистрации должны быть получены следующие документы:

- Свидетельство государственной регистрации программы для ЭВМ в Роспатенте;
- Приказ Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации о включении сведений о программном обеспечении в единый реестр российских программ для электронных.

9 Источники доработки

Источниками для разработки настоящего детализированного Технического задания являются:

- Заявка на участие в конкурсном отборе проектов по разработке российских решений в сфере информационных технологий в рамках реализации федерального проекта «Цифровые технологии» национальной программы «Цифровая экономика Российской Федерации».
- Описание альфа-версии/прототипа программного обеспечения аналитической информационной системы «Визари» (АИС «Визари»).
- Детализированное описание модели монетизации Решения.

10 Требования по коммерциализации бета-версии Решения

По итогам разработки бета-версии Решения Исполнитель должен обеспечить коммерциализацию Решения в следующем объеме: более 50 000 000,00 рублей с НДС в 2023 году, более 110 000 000,00 рублей с НДС в 2024 году. Достижение заявленных требований подтверждается соответствующими договорами и государственными контрактами на продажу лицензий, а также на реализацию проектов по оказанию услуг по адаптации и внедрению бета-версии Решения с приложением актов выполненных работ и оказанных услуг.