РАЗРАБОТКА ВЕДОМСТВЕННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ «ЛЕСОПОЛЬЗОВАНИЕ»

# СОКРАЩЕНИЯ

ПДЛУ

ЕГРН

СЭД ПСО

СЭР РИП

# ВВЕДЕНИЕ

С 2009 года функционирует государственный некоммерческий интернет-портал «Госуслуги». Система обеспечивает доступ юридическим и физическим лицам к получению в электронной форме государственных и муниципальных услуг. Все услуги, размещенные на портале, соотнесены с конкретным регионом России. К концу 2023 года на данном портале зарегистрировано 109 млн россиян.

В Свердловской области существует информационная система для организации мониторинга социально-экономического развития. Система состоит из следующих подсистем:

* Система управления социально-экономическими показателями,
* Подсистема автоматизации технологических процессов ИОГВ,
* Региональная геоинформационная система,
* Система управления информацией здравоохранения,
* Региональная интеграционная платформа.

С системой интегрированы следующие ведомственные информационные системы для оказания услуг со следующими названиями:

* Инвестиционный комплекс
* Недропользование
* Кадастр ООПТ
* Водопользование
* Экология
* Агропромышленный комплекс

12.04.2022 на портале «ЕИС ЗАКУПКИ» был размещен открытый конкурс в электронной форме с номером [№ 0162200011822000790](https://zakupki.gov.ru/epz/order/notice/ok20/view/common-info.html?regNumber=0162200011822000790). Объектом закупки, в рамках конкурса, является выполнение работ по созданию ведомственной информационной системы «Лесопользование Свердловской области». Размещение осуществил департамент государственных закупок Свердловской области. Исполнителем, по результатам конкурса, стала компания ООО «Техноком» в которой я проходил практики, предусмотренные в рамках обучения в ТюмГУ. В рамках дипломной работы будет представлен процесс анализа предметной области, проектирования и реализации программного продукта.

На этом введение в диплом заканчивается, но для отчета по практике я продолжу

ООО «Техноком» предоставляет услуги полного цикла по разработке, внедрению и сопровождению программных решений. Компания имеет ряд программных продуктов собственной разработки: BDSA-GIS, BDSA-REPORTING, NOVAGIS, GeoViewer и еще 7 других.

BDSA-GIS - программный продукт предназначен для обеспечения доступа пользователя к электронным картам информационной системы и позволяет работать с картами через Web-интерфейс, а также формировать тематические рабочие наборы для мобильного приложения.

NOVAGIS - единая централизованная геоинформационная платформа, предназначенная для организации работы с картографическими, пространственными и геоданными Заказчика

GeoViewer - программа GeoViewer предназначена для просмотра в web-интерфейсе геолого-геофизической, пространственной и графической информации, представленной в файлах различных форматов

# ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

В описании объекта закупки, упомянутом ранее, требуется предоставить возможность прохождения цикла процесса комплексного лесопользования, обобщенная схема лесопользования приведена на (Рисунок 1).



Рисунок 1. Обобщенная схема комплексного лесопользования

В рамках работы будет описан этап оформления права пользования лесным участком. При оформлении права пользования лесным участком выполняются следующие административные процедуры:

1. Рассмотрение проектной документации;
2. Прием и регистрация заявления на предварительное согласование лесного участка;
3. Принятие решения о предварительном согласовании либо отказ в предварительном согласовании лесного участка;
4. [Прием и регистрация заявления о предоставлении в пределах земель лесного фонда лесного участка в пользование;](https://www.gosuslugi.ru/)
5. [Рассмотрение заявления о предоставлении в пределах земель лесного фонда лесного участка в пользование;](https://www.gosuslugi.ru/)
6. [Формирование и направление межведомственных запросов в другие органы (организации);](https://www.gosuslugi.ru/)
7. [Направление заявителю копии решения уполномоченного органа о предоставлении лесного участка либо извещения об отказе в предоставлении услуги;](https://www.gosuslugi.ru/)
8. Подготовка правоустанавливающего документа на лесной участок;
9. Подписание правоустанавливающего документа на лесной участок (акт приема-передачи лесного участка в пользование входит в состав правоустанавливающего документа).

Для выполнения этих административных процедур министерство предоставляет следующий список услуг:

* Утверждение проектной документации лесного участка,
* Предварительное согласование лесного участка,
* Предоставление лесного участка в аренду,
* Предоставление лесного участка в безвозмездное пользование,
* Предоставление лесного участка в постоянное пользование

Рассмотрим документооборот при предоставлении услуг, атрибутивный состав заявлений будет рассмотрен во второй главе при проектировании.

Документы необходимые для получения услуги:

* Утверждение проектной документации: заявление; проектная документация.
* Предварительное согласование: заявление; схема расположения участка, если испрашиваемый участок предстоит образовать; ПДЛУ при подаче заявления о предварительном согласовании предоставления лесного участка; Документы, подтверждающие право заявителя на предоставление лесного участка без проведения торгов.
* Предоставление лесного участка в аренду: заявление.
* Предоставление лесного участка в безвозмездное пользование: заявление; Проектная документация; Выписка из ЕГРН.
* Предоставление лесного участка в постоянное пользование: заявление; Проектная документация; Выписка из ЕГРН.

Требования к системе

**Геосервер**

Для получения некоторых услуг используется картографическая информация. В Свердловской области уже реализована система для работы с геоинформацией. Разрабатываемая система будет интегрирована с - региональной геоинформационной системой Свердловской области (РГИС СО. Геосервер нужен нам для чтения границ участка(ов), создания участка, проверки пересечений образованных участков с образуемым участком. Геосервер предоставляет конечные точки для вызова методов на базе архитектурного стиля REST.

Запрос на получение подложки: GET <https://rgis.egov66.ru/geoserver/master/wms?service=WMS&request=GetMap&layers=master%3Abase_cart&styles=&format=image%2Fpng&transparent=true&version=1.3.0&width=256&height=256&crs=EPSG%3A3857&bbox=6887893.492833803,8140237.764258131,7044436.526761844,8296780.798186171> ответ этого запроса – png картинка с задним фоном карты.

Изображение выглядит как карта, текст, атлас

Автоматически созданное описание

Запрос на получение картинки с участком: GET <https://rgis.egov66.ru/geoserver/master/wms?service=WMS&request=GetMap&layers=master%3Abase_cart&styles=&format=image%2Fpng&transparent=true&version=1.3.0&width=256&height=256&crs=EPSG%3A3857&bbox=6887893.492833803,8140237.764258131,7044436.526761844,8296780.798186171> ответ запроса это png картинка с полигоном участка

Изображение выглядит как черный, темнота, снимок экрана, линия

Автоматически созданное описание

Наложением участка на подложку занимается фронтенд.

Запрос на создание участка: POST <https://uslugi.egov66.ru/geoserver/master/wfs> Тело запроса прикреплено в приложении.

Поиск пересечений: POST <http://dkrekb.rusoft.tech:8104/intersector/intersector-controller/findIntersect> Тело запроса прикреплено в приложение.

**Формирование итоговых документов**

На этапе подготовки правоустанавливающего документа (далее Итоговый документ) система должна из шаблона автоматически сформировать итоговый документ в рамках услуги. Для каждой услуги итоговый документ свой. Шаблоны документов прикреплены в приложение. Перечислю их в соответствии с перечислением услуг ранее.

* Приказ об утверждении проектной документации и государственном учете лесного участка; Отказ в утверждении.
* Решение о согласовании лесного участка и Приказ о согласовании лесного участка; Уведомление об отказе в предварительном согласовании лесного участка.
* Проект договора аренды и Приказ о предоставлении в аренду лесного участка; Решение об отказе в предоставлении участка в аренду.
* Проект договора о предоставлении лесного участка в безвозмездное пользование и приказ о предоставлении земельного участка в безвозмездное пользование; Решение об отказе в предоставлении лесного участка в безвозмездное пользование.
* Проект акта приема-передачи лесных участков в постоянное пользование и приказ о предоставлении лесного участка в постоянное пользование; Решение об отказе в предоставлении лесного участка в постоянное пользование.

**СЭД ПСО**

По предоставленному описанию бизнес-процесса необходимо регистрировать заявления и подписывать документы. Регистрация и подписание документов происходит в СЭД ПСО. Интеграция с СЭД осуществится через подсистему интеграционного взаимодействия, которая предоставляет возможность информационного обмена (интеграции) с Федеральными информационными ресурсами, информационными системами Свердловской области, а также взаимодействия внутри подсистем СЭР РИП. В СЭД ПСО необходимо направлять следующий список документов: входящий документ, исходящий документ, служебная записка, приказ. На регистрацию будут отправляться все заявления, упомянутые ранее. На подписание будут отправляться все итоговые документы. Взаимодействие будет описано при описании запросов и ответов к бекенду.

**Меж вед запросы**

Регламент подразумевает межведомственное взаимодействие. Интеграция с ведомствами реализовано через взаимодействие с СЭД ПСО. Рассмотрим какие запросы должны отправляться для каждой из услуг. Перечислю их в соответствии с перечислением услуг ранее.

**Утверждение проектной документации**: не требует межведомственных взаимодействий,

**Предварительное согласование**: не требует межведомственных взаимодействий,

**Предоставление лесного участка в аренду**:

* Запрос сведений в ФНС России - о постановке заявителя на налоговый учет в налоговом органе. **Исходящий документ**: Запрос сведений о постановке заявителя на налоговый учет в налоговом органе. **Входящий документ**: Ответ на запрос сведений о постановке заявителя на налоговый учет в налоговом органе.
* Запрос сведений в ФНС России - о предоставлении выписки из ЕГРН на объекты, подлежащие реконструкции. **Исходящий документ**: Запрос сведений о предоставлении выписки из ЕГРН на объекты, подлежащие реконструкции. **Входящий документ**: Ответ на запрос сведений о предоставлении выписки из ЕГРН на объекты, подлежащие реконструкции.
* Запрос сведений из Министерства природных ресурсов и экологии Свердловской области - о наличии лицензии на пользование недрами. **Исходящий документ**: Запрос сведений о наличии лицензии на пользование недрами. **Входящий документ**: Ответ на запрос сведений о наличии лицензии на пользование недрами.
* Запрос в Федеральную службу государственной регистрации, кадастра и картографии на предоставление выписки о земельном участке из ЕГРН. **Исходящий документ**: Запрос сведений о земельном участке из ЕГРН. **Входящий документ**: Ответ на запрос о предоставлении выписки о земельном участке из ЕГРН.
* Запрос в Федеральную службу государственной регистрации, кадастра и картографии справки о содержании правоустанавливающих документов. **Исходящий документ**: Запрос сведений о содержании правоустанавливающих документов. **Входящий документ**: Ответ на о содержании правоустанавливающих документов.
* Запрос в Министерство промышленности и науки Свердловской области - на получение документов, подтверждающих включение инвестиционного проекта в области освоения лесов в перечень приоритетных инвестиционных проектов. **Исходящий документ**: Запрос на получение документов, подтверждающих включение инвестиционного проекта в области освоения лесов в перечень приоритетных инвестиций. **Входящий документ**: Ответ на запрос на получение документов, подтверждающих включение инвестиционного проекта в области освоения лесов в перечень.
* Запрос в Департамент по охране, контролю и регулированию использования животного мира Свердловской области - на получение документов, подтверждающих заключение охотохозяйственных соглашений. **Исходящий документ**: Запрос на получение документов, подтверждающих заключение охотохозяйственных соглашений. **Входящий документ**: Ответ на запрос на получение документов, подтверждающих заключение охотохозяйственных соглашений.

Курсивом выделю повторяющиеся межведомственные взаимодействия.

Предоставление лесного участка в безвозмездное пользование:

* Запрос в ФНС России - предоставление копии свидетельства о государственной регистрации ЮЛ. **Исходящий документ**: Запрос на предоставление копии свидетельства о государственной регистрации ЮЛ. **Входящий документ**: Ответ на запрос на предоставление копии свидетельства о государственной регистрации ЮЛ.
* Запрос в ФНС России - предоставление копии свидетельства о государственной регистрации ФЛ в качестве ИП. **Исходящий документ**: Запрос на предоставление копии свидетельства о государственной регистрации ФЛ в качестве ИП. **Входящий документ**: Ответ на запрос на предоставление копии свидетельства о государственной регистрации ФЛ в качестве ИП.
* *Запрос сведений в ФНС* *России - о постановке заявителя на налоговый учет в налоговом органе.*
* Запрос в Федеральную службу государственной регистрации, кадастра и картографии - на предоставление выписки из ЕГРН и сделок с ним на испрашиваемый лесной участок. **Исходящий документ**: Запрос на предоставление выписки из ЕГРН и сделок с ним на испрашиваемый лесной участок. **Входящий документ**: Ответ на запрос на предоставление выписки из ЕГРН и сделок с ним на испрашиваемый лесной участок.
* Запрос в Федеральную службу государственной регистрации, кадастра и картографии - на предоставление кадастрового паспорта испрашиваемого лесного участка. **Исходящий документ**: Запрос на предоставление кадастрового паспорта испрашиваемого лесного участка. **Входящий документ**: Ответ на запрос на предоставление кадастрового паспорта испрашиваемого лесного участка.

Предоставление лесного участка в постоянное пользование:

* *Запрос в ФНС России - предоставление копии свидетельства о государственной регистрации ЮЛ.*
* *Запрос сведений в ФНС России - о постановке заявителя на налоговый учет в налоговом органе.*
* *Запрос в Федеральную службу государственной регистрации, кадастра и картографии - на предоставление выписки из ЕГРН и сделок с ним на испрашиваемый лесной участок.*
* *Запрос в Федеральную службу государственной регистрации, кадастра и картографии - на предоставление кадастрового паспорта испрашиваемого лесного участка.*

**Цель работы**: сократить время для работы над заявлением.

**Аналоги**

Аналогов нет, т. к. система разрабатывается для каждой области индивидуально.

**Описание запросов и ответов к бекенду**

В компании есть деление на фронтент и бекенд разработку. В данном случае я разрабатывал бекенд часть. Поэтому дальше описывается взаимодействие с фронтендом, а не с заявителем.

Структуру всех данных реквестов и респонсов определим во второй главе при описании методов контроллера.

Запрос на чтение списка предоставляемых услуг: бекенд формирует список урезанной информации об услуге в виде JSON текста и отправляет в ответе. В запросе возможно передать фильтр по наименованию.

Запрос на чтение одной услуги: бекенд в ответе отправляет подробную информацию об услуге.

Запрос на чтение своих заявлений: бекенд фильтрует заявления по авторизованному пользователю. Формирует ответ в виде JSON текста. В запросе возможно передать параметры фильтрации и сортировки на каждое поле.

Запрос на чтение своего заявления: бекенд проверяет принадлежность заявления к заявителю. Формирует ответ в виде JSON текста.

Запрос для получения услуги (создание заявления): бекенд создаёт заявление, устанавливает статус черновик, пред заполняет информацию о заявителе, передаёт на фронтенд в виде JSON текста.

Запрос на изменение заявления: бекенд валидирует данные, проверяет доступ к редактированию, сохраняет заявление. Если в заявлении прикладывается ГИС информация, то бекенд отправляет запрос для проверки пересечений на геосервер.

Загрузка файла (прикладывание документа): бекенд проверяет доступ для загрузки файла, сохраняет файл на файловое хранилище, сохраняет в базе данных информацию о файле с привязкой к заявлению. Возвращает созданную запись в виде JSON текста.

Скачивание файла: бекенд проверяет доступ на скачивание файла, загружает в тело ответа необходимый файл.

Удаление файла: бекенд проверяет доступ на удаление, удаляет файл из базы данных, после закрытия транзакции удаляет файл с файлового хранилища.

Изменение информации о файле: бекенд проверяет доступ, сохраняет измененные поля в базу данных.

Подача заявления: бекенд проверяет доступ, проверяет заполнение обязательных полей и переводит заявление в статус «не зарегистрировано». Отправляет запрос в «генератор сообщений» для формирования пакета с заявлением и последующей отправкой заявления на регистрацию в СЭД ПСО посредством интеграционного блока. После отправки пакета заявление переводится в статус «На регистрации». После получения ответного пакета в заявление устанавливается регистрационный номер и дата. Статус заявления меняется на «Зарегистрировано». После подачи заявления заявителю блокируется редактировать атрибуты заявления, возможность прикреплять/редактироватьт/удалять приложенные файлы не блокируется.

Чтение зарегистрированных заявлений: бекенд проверяет доступ на чтение всех заявлений, возвращает список заявлений, отфильтрованных по статусу «Зарегистрировано». Автоматически сортируя по возрастанию по количеству дней до истечения срока оказания услуги. Возвращает заявления в виде JSON текста.

Назначение сотрудника на работу над заявлением: бекенд проверяет роль пользователя, связывает заявление и исполнителя.

Вынесение решения по заявлению: бекенд проверяет доступ, сохраняет переданное решение.

Чтение доступных для создания итоговых документов: бекенд формирует список доступных итоговых документов на основе оказываемой услуги. В запросе возможно передать параметры фильтрации и сортировки на каждое поле.

Создание записи итогового документа: бекенд проверяет доступ на создание, сохраняет переданную информацию о документе.

Изменение записи итогового документа: бекенд проверяет доступ, сохраняет изменившиеся данные.

Создание файла итогового документа: бекенд проверяет доступ, на основе переданного итогового документа выбирает шаблон файла docx. Ищет в шаблоне динамические поля, ${fieldName.fieldName…} и подставляет в них соответствующие значения из заявления. Загружает в тело ответа заполненный файл.

Прикрепление файла к записи итогового документа: бекенд проверяет доступ, сохраняет файл на ФХ и прикрепляет его к записи итогового документа.

Создание файла итогового документа и прикрепление к записи: бекенд проверяет доступ, формирует файл документа из шаблона, сохраняет его на файловое хранилище, прикрепляет к заявлению.

Удаление записи итогового документа: бекенд проверяет доступ, удаляет запись из базы данных, после закрытия транзакции удаляет файл с ФХ.

Удаление файла итогового документа: бекенд проверяет доступ, удаляет файл с ФХ.

Формирование (создание) межведомственного запроса: бекенд проверяет досутп, создает запись межведомственного запроса на основе переданного типа запроса.

Изменение межведомственного запроса: бекенд проверяет доступ, сохраняет переданные данные.

Удаление межведомственного запроса: бекенд проверяет доступ, удаляет межведомственный запрос.

Отправка межведомственного запроса: бекенд проверяет досутп, инициирует запрос в «генератор сообщений», генератор формирует пакет с документом – запросом, отправляет, переводит статус запроса на «Отправлен». После получения ответного пакета сохраняется полученный входящий документ, устанавливается регистрационный номер запроса, дата получения ответа. Статус запроса переводится в «Завершен».

*Описание работы генератора сообщений и взаимодействие его с интеграционным блоком будет описано после описания реквестов, респонсов.*

Отправка итогового документа на подписание: бекенд првоеряет доступ, инициирует запрос в «генератор сообщений», генератор формирует пакет с документом – запросом, отправляет, переводит статус заявления на «На подписании». После получения ответного пакета сохраняется полученный подписанный файл документа, устанавливается регистрационный номер записи документа, дата подписания документа. Статус заявления переводится в «Подписан». После смены статуса на подписан заявителю отправляется уведомление о готовности итогового документа.

Завершение работы над заявлением: бекенд проверяет доступ, проверяет возможность сменить статус на предоставленный с текущего. Меняет статус заявления. После установки статуса «Завершено» заявление блокируется для редактирования оператором.

Чтение итоговых документов: бекенд формирует ответ на основе итоговых документов в заявлениях заявителя. В запросе возможно передать параметры фильтрации и сортировки на каждое поле.

Авторизация пользователя: бекенд запрашивает авторизационный токен у ЕСИА. Взаимодействие с ЕСИА описано в пункте 2.9.1.2

Чтение совей учетной записи: бекенд формирует ответ на основе авторизованного пользователя в виде JSON текста

Изменение данных об учетной записи: бекенд проверяет досутп, валидирует данные, сохраняет измененные поля.

**Взаимодействие с интеграционным блоком**

Обмен сообщениями с интеграционным блоком реализуется с помощью файловой системы, с сообщениями в виде ZIP пакетов. Структура сообщений была определена министерством предоставлена в виде XSD файла, в работе представлю её в виде диаграмм классов во второй главе.

Генератор сообщений – отдельное серверное java приложение, которое вызывается из основного, обосновано это тем, что взаимодействие с СЭД ПСО необходимо для других ведомственных систем. Проще запустить копии генераторов с разными конфиг настройками, чем дублировать код во всех системах.

Сейчас рассмотрим логику работы генератора сообщений и его общение с интеграционным блоком.

**Генерация сообщений:**

Поступает реквест на генерацию определенного типа сообщения. На основе переданного идентификатора документа формируется XML текст – описание структуры зип пакета, служебная и предметная информация. Структура XML текста будет рассмотрена в главе проектирования. Рассмотрим правила формирования пакета: XML текст должен быть записан в файл с названием passport.xml, все прилагаемые файлы должны быть загружены в архив. Архив сохраняется в условленную директорию. В базу данных сохраняется идентификатор отправляемого документа, дата и время отправки в специальную сущность для отслеживания пакетов далее «пакет-статус».

**Обработка сообщений:**

Рассмотрим общую логику работы с пакетом. Входящая директория проверяется по расписанию раз в 10 минут. Приложение сканирует директорию и создает список всех файлов, после запоминает их и ставит счетчик «ошибочных обработок» на 0. Если файл уже обрабатывался с ошибкой, то счетчик увеличивается на 1. Если счетчик становится больше 5, то файл перемещается в условленную директорию, для ручной проверки. Если файл успешно обработался, то удаляется запись об этом файле и статус «пакет-статус» меняется на другой в зависимости от обработанного пакета, либо «запрос получен сторонней системой», либо «завершен», файл переносится в директорию «trash» из которой удалится через 4 месяца. Тем самым повторная обработка файлов невозможна.

Всего предусмотрено два блока сообщений: квитанция о успешной доставке пакета в СЭД ПСО и ответ из СЭД ПСО. Рассмотрим логику обработки XML текста для каждого типа сообщения.

Квитанция о получении документа: в сущность «пакет-статус» устанавливает статус сообщения, текст сообщения/ошибка и время получения сторонней системой. Статус

Входящий документ: в заявление устанавливается регистрационный номер и дата, статус заявления устанавливается как зарегистрированный, в «статус-пакета» устанавливается дата обработки полученного пакета, путь до итогового пакета, статус устанавливается как завершен.

Исходящий документ: в документ, привязанный к заявлению, устанавливается регистрационный номер и дата, прикрепляется подписанный файл, прикрепляется весь полученный архив, в «статус-пакета» устанавливается дата обработки полученного пакета, путь до итогового пакета, статус устанавливается как завершен.

Приказ: обрабатывается таким же образом, как и исходящий

Служебная записка: обрабатывается таким же образом, как и исходящий

Обработка отрицательных ответов предусмотрена, но в текущей версии не используется.

Также во входящую директорию может прийди пакет со списком пользователей – адресатов в СЭД ПСО. Если пользователь с логином существует, то запись о нем обновляется, а если не существует, то создается новая запись.

# Проектирование архитектуры ИС

* 1. Выбор технического обеспечения

Выдержка из требований к программному обеспечению: «Система должна базироваться на программном обеспечении СЭР РИП и её подсистем. Исполнитель должен использовать существующее программное обеспечение подсистем СЭР РИП, расширяя их возможности для реализации предметного функционала»

**Требования к ОС**

Стабильное ядро линукс (не ниже 5.3), ОС на базе некоммерческого дистрибутива Denian, обработка системных команд и данных пользователя и данных пользователя: через терминал, с помощью графического интерфейса. ОС поддерживает установку на компьютеры с 64 разрядным процессором Intel с поддержкой EFI, поддержка чтения носителей с файловыми системами Ext 2/3/4, ISO 9660, FAT, NTFS. В качестве ОС выбран дистрибутив Astra Linux версии 2.12.44.

**Веб-сервер**

Веб-сервер: вся серверная инфраструктура СЭР СИП работает на веб-сервере NGINX.

**СУБД**

СУБД должно быть совместимо с открытыми исходными текстами PostgreSQL не ниже 12.2 версии. Поддержка работы в 1С. Встроенные механизмы резервирования и восстановления БД, Управление объектами БД с помощью консольных и графических утилит. В качестве СУБД выбран PostgreSQL версии 16.2.2.

**ЯП**

Используемая при создании Системы среда исполнения кода должна удовлетворять требованиям ГосJava (требование введено для обеспечения совместимости с подсистемами информационной системы для организации мониторинга социально-экономического развития Свердловской области, использующими среду исполнения кода с указанными характеристиками). В качестве ЯП был выбран java 17. В качестве среды исполнения (JRE) кода используется ГосJava 2024.1.

**Веб фреймворк**

Выбор веб фреймворка не регламентирован в описании объекта закупки. Проведем сравнение самых популярных фреймворков по следующим критериям: простота в использовании, качество документации, количество доступных полезных функций, оценка сообщества, простота поддержки написанного кода, возможность использовать вместо стандартных модулей – самописные, реализующие общепринятый стандарт.

Проведем анализ основываясь на топ 3 фремворках по мнению интернет ресурса url: <https://scand.com/ru/company/blog/top-java-frameworks/>

**Spring Boot**

Спринг бут реализует DI контейнер, не обязательно конфигурировать роутинг, зависимость для контейнера в отдельных файлах, можно с помощью аннотаций. У спринг бут подробная и понятная документация. Спринг бут решает проблему совместимости разных библиотек. Так называемые стартеры – содержат в себе необходимые библиотеки для реализации определенного функционала, например для обработки http сообщений, ORM, логирования, безопасности. Спринг предоставляет возможность чистого написания кода из-за совей «магии», поддержки аспектного программирования, и большого количества расширяемых мест. Если не будет ничего лучше, то спринг - вполне хороший вариант.

**Helidon SE**

В этом фреймворке не реализован DI контейнер для внедрения зависимостей, что в свою очередь обязывает использовать другую библиотеку для их внедрения. Также необходимо конфигурировать все серверное приложение самому, в отдельном файле. Нам этот фремворк не подходит.

**Ktor**

В этом фреймворке не реализован DI контейнер для внедрения зависимостей, что в свою очередь обязывает использовать другую библиотеку для их внедрения. Так же требует ручной конфигурации, нам этот фреймворк не подходит.

**ORM**

Выбор ORM не регламентирован. У технокома есть собственная реализация JPA стандарта, которая позволяет конфигурировать бд из приложения. А также поддерживает dynamic object relation mapping – нам необходима эта функция для того, чтобы при добавлении новых заявлений/услуг не требовалось перезапускать серверное приложение. Т.к разработка реализует JPA стандарт, то эта библиотека может быть встроена в экосистему Spring.

**Документирование API**

Для документирования API в экосистеме спринга используется springdoc-openapi-starter-webmvc-ui

**Модульное тестирование**

Выбор фреймворка для модульного тестирования не регламентирован. В экосистеме Spring есть встроенная библиотека для тестирования программного кода spring-boot-starter-test.

<https://uslugi.egov66.ru/forest/workspace/requests/2262912#data>

**Защита от НСД**

Для защиты от НСД будет использован модуль spring-security-bom

## Диаграмма бизнес-процессов после внедрения разработки в нотации BPMN

Построена диаграмма модели бизнес-процесса с использованием разрабатываемой системы в нотации BPMN. На схеме представлен основной сценарий получения услуги.

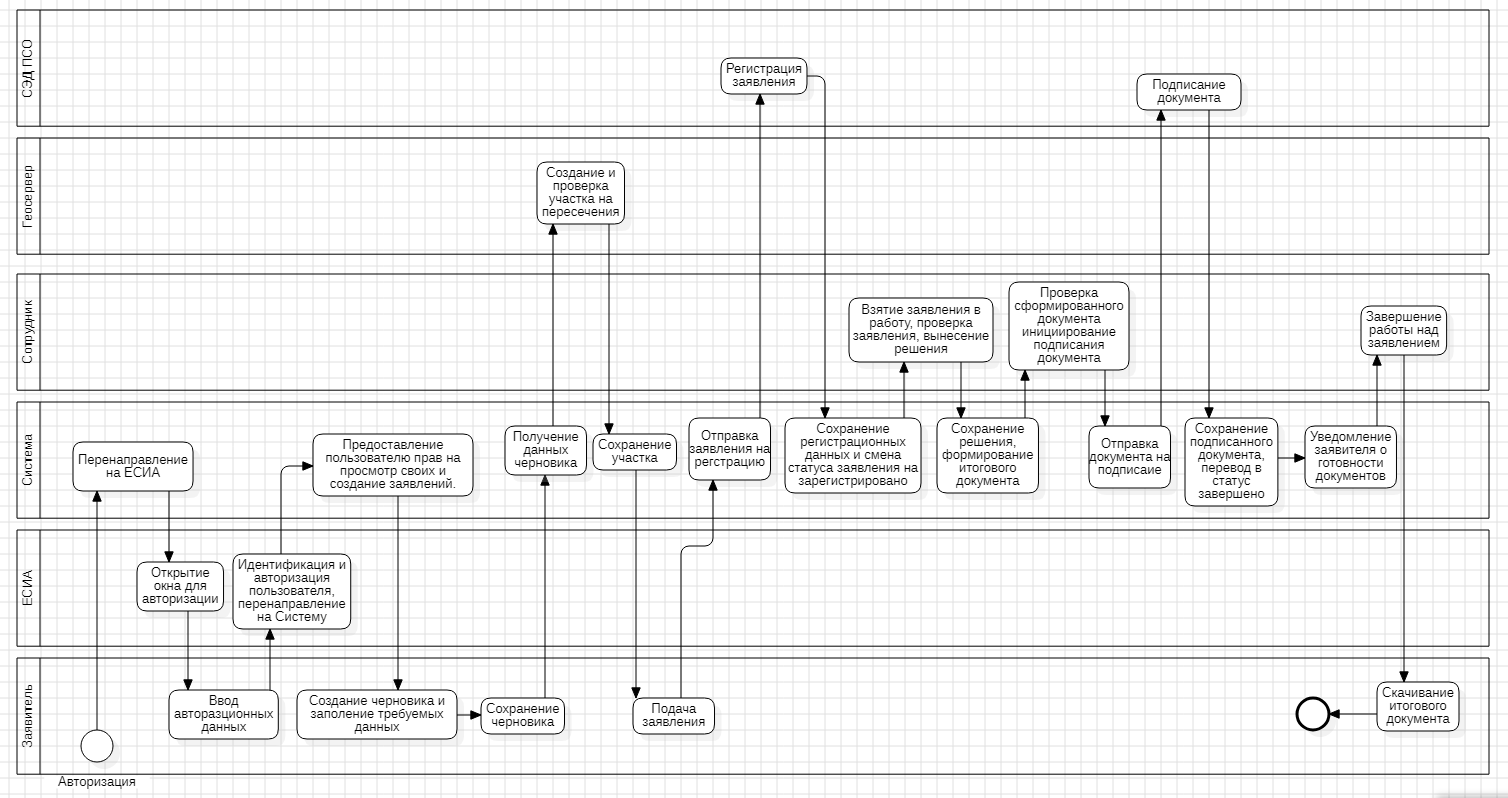


Рисунок 1 - диаграмма модели процесса в нотации BPMN

## Описание диаграммы бизнес-процессов

Участники: Система, сотрудник организации, заявитель, геосервер, ЕСИА, интеграционный блок.

Вход: Заявление

Выход: Итоговый(е) документ(ы)

Задачи:

## Схемы классов предметной области

Т.к предметная область достаточно большая то было принято решение разделить модель на несколько функциональных групп: для описания услуги, для описания работы с заявления, для работы с личным кабинетом, для работы с файлами, для работы со статусом заявления, для интеграции с интеграционным блоком. В рамках каждого блока будет представлено 3 диаграммы: первая описывает модель данных (model), вторая представление модели (view), третья структуру поведенческих классов (Controller, Service, Repository – csr)

Услужная часть:

Изображение выглядит как текст, снимок экрана, Шрифт, Печать

Автоматически созданное описание

Рисунок 2 - услужная часть (model)

Изображение выглядит как текст, снимок экрана, черно-белый, Шрифт

Автоматически созданное описание

Рисунок 3 - услужная часть (view)

Изображение выглядит как текст, снимок экрана, черно-белый, документ

Автоматически созданное описание

Рисунок 4 - услужная часть (controller)

Заявительная часть:

Изображение выглядит как текст, снимок экрана, Шрифт, черно-белый

Автоматически созданное описание

Рисунок 5 - заявительная часть (model)

Изображение выглядит как текст, снимок экрана, Шрифт, черно-белый

Автоматически созданное описание

Рисунок 6 - заявительная часть (view)

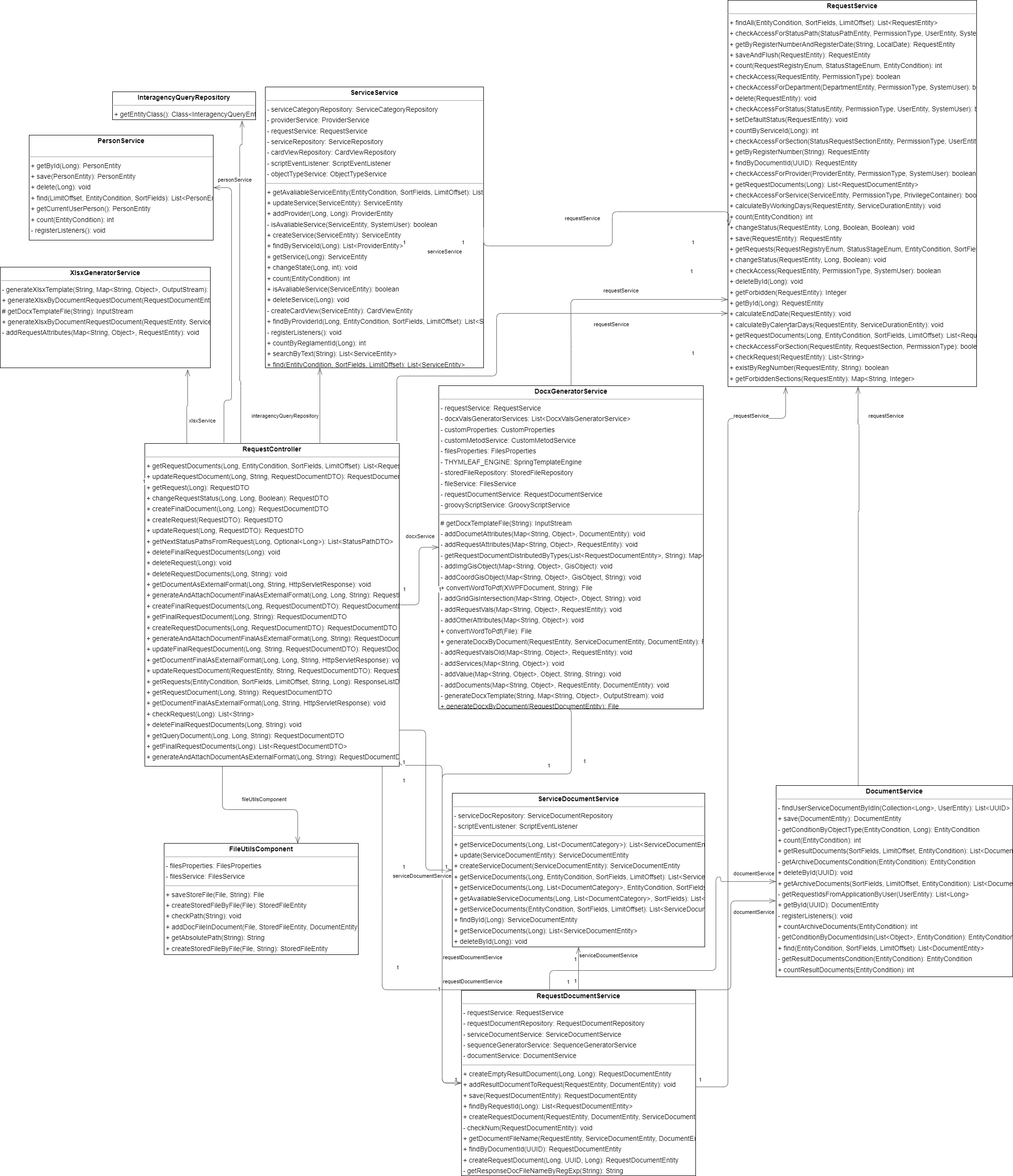


Рисунок 7 - заявительная часть (controller)

Часть для отслеживания статуса заявления:

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Рисунок 8 - статусная часть (model)

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание

Рисунок 9 - статусная часть (view)

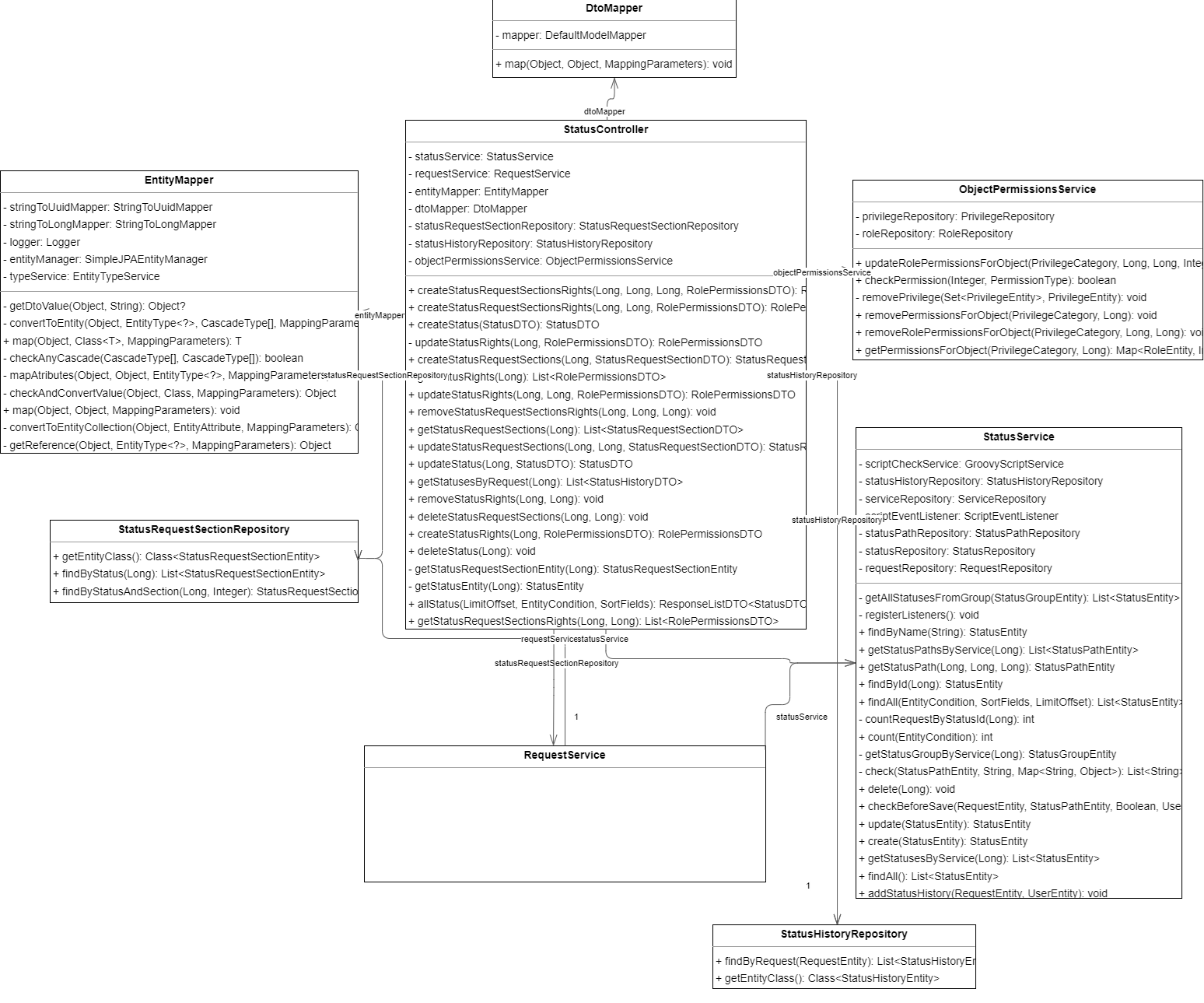


Рисунок 10 - статусная часть (controller)

Часть для работы с файлами:

Часть для интеграции с интеграционным блоком:

Изображение выглядит как текст, снимок экрана, диаграмма, Параллельный

Автоматически созданное описание

Рисунок 5 - схема наследования документов и содержания сообщения (view)

Изображение выглядит как текст, диаграмма, Параллельный, документ

Автоматически созданное описание

Рисунок 6 - содержимое разных типов сообщений (view)

Изображение выглядит как текст, диаграмма, План, Параллельный

Автоматически созданное описание

Рисунок 10 - Диаграмма реализации сервиса (csr)

## Словарь схемы классов предметной области

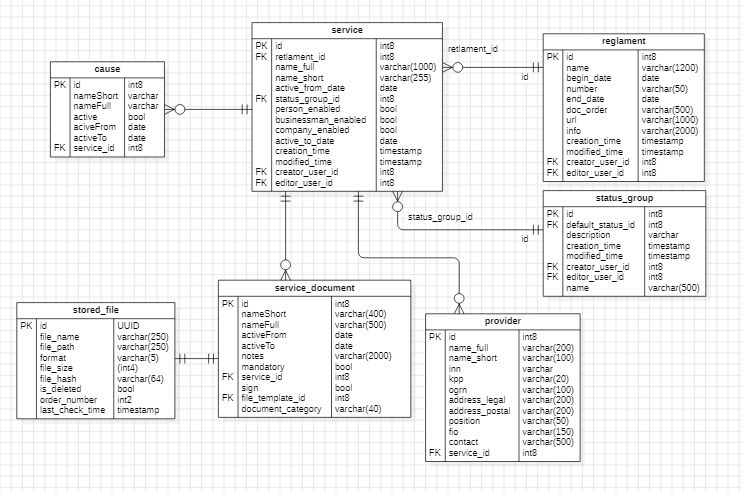
…

## Логическая модель данных

Общий вид схемы данных, сгенерированной из программы для работы с СУБД

Модель логической схемы данных:

Услужная часть: (не дорисовал)



Заявительная часть:

Отслеживание статусов заявления:

## Словарь логической модели данных

…

## Взаимодействие информационной системы другими системами

Разберем подробнее правила обмена данными с каждой из системы.

* + 1. Описание взаимодействия с ЕСИА
       1. **Изучение механизмов ЕСИА**

ЕСИА предоставляет два механизма аутентификации:

* механизм, основанный на стандарте SAML версии 2.0;
* механизм, основанный на модели OpenID Connect 1.0.

**Аутентификация с использованием стандарта SAML**

ЕСИА использует стандарт SAML версии 2.0, который был разработан в 2005 году концерном OASIS. SAML базируется на языке XML и определяет способы обмена информацией об аутентификации пользователей, их полномочиях и идентификационных данных. В соответствии с принятой в этом стандарте терминологией, ЕСИА выступает в роли доверенного поставщика идентификации (Identity Provider), а система выступает в роли поставщика услуг (Service Provider).

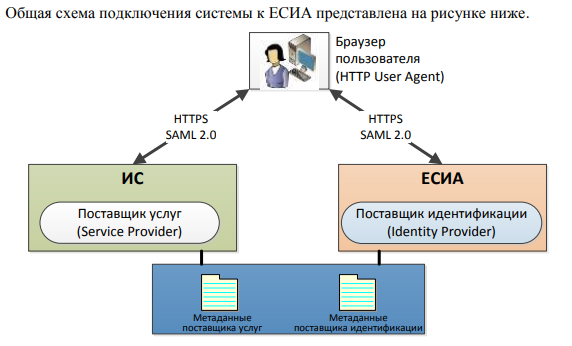


Рисунок 5 - Схема взаимодействия ИС с ЕСИА с целью идентификации и аутентификации с использованием стандарта SAML 2.0

**Аутентификация с использованием модели OpenID Connect**

В ЕСИА создан механизм аутентификации пользователей, основанный на спецификациях OAuth 2.0 и расширении OpenID Connect 1.0.

Протокол определяет взаимодействие следующих сторон:

* владелец ресурса (resource owner) – сущность, которая может предоставить доступ к защищаемому ресурсу (например, физическое лицо, заявитель);
* система-клиент (client) – приложение, которое запрашивает доступ к защищаемому ресурсу от имени его владельца;
* сервис авторизации (authorization server) – сервис, который выпускает для системы клиента маркеры идентификации с разрешениями от владельца ресурса, а также маркеры доступа, позволяющие получать доступ к данным;
* поставщик ресурса (resource server) – сервис, обеспечивающий доступ к защищаемому ресурсу на основе проверки маркеров идентификации и маркеров доступа (например, к идентификационным данным пользователя).

Расширение OpenID Connect 1.0 предполагает использование маркера идентификации (ID Token) в целях проведения идентификации и аутентификации пользователя. Маркер идентификации содержит идентификационные данные пользователя, а также ряд служебных параметров (дата выдачи, время окончания срока действия и пр.).

Для иллюстрации использования OpenID Connect 1.0 в ЕСИА принята следующая терминология:

* владелец ресурса – это пользователь;
* система-клиент – это информационная система, интегрированная с ЕСИА с целью идентификации и аутентификации, например региональный портал услуг;
* сервис авторизации и поставщик ресурса – это ЕСИА.

Общая схема подключения системы к ЕСИА для проведения аутентификации представлена на рисунке ниже.

Изображение выглядит как текст, снимок экрана, диаграмма, линия

Автоматически созданное описание

Рисунок 6 - Схема подключения системы к ЕСИА

Изучив оба механизма, было принято решение использовать второй механизм аутентификации.

* + - 1. **Описание сценария работы с ЕСИА**

Сценарий включает следующие шаги:

1. Пользователь нажимает на веб-странице системы-клиента кнопку «Войти через ЕСИА».

2. Система-клиент формирует и отправляет в ЕСИА запрос на аутентификацию и перенаправляет браузер пользователя на специальную страницу предоставления доступа.

3. ЕСИА осуществляет аутентификацию пользователя одним из доступных способов. Если пользователь ещё не зарегистрирован в ЕСИА, то он может перейти к процессу регистрации.

4. Когда пользователь аутентифицирован, ЕСИА сообщает пользователю, что система-клиент запрашивает данные о нем в целях проведения идентификации и аутентификации, предоставляя перечень запрашиваемых системой-клиентом сведений.

5. Если пользователь дает разрешение на проведение аутентификации системой-клиентом, то ЕСИА выдает системе-клиенту специальный авторизационный код.

6. Система-клиент формирует в адрес ЕСИА запрос на получение маркера идентификации, включая в запрос полученный ранее авторизационный код.

7. ЕСИА проверяет корректность запроса (например, что система-клиент зарегистрирована в ЕСИА) и авторизационного кода и передает системе-клиенту маркер идентификации.

8. Система-клиент извлекает идентификатор пользователя из маркера идентификации. Если идентификатор получен, а маркер проверен, то система-клиент считает пользователя аутентифицированным.

После получения маркера идентификации система-клиент использует REST-сервисы ЕСИА для получения дополнительных данных о пользователе, предварительно получив соответствующий маркер доступа.

Изображение выглядит как текст, снимок экрана, Параллельный, линия

Автоматически созданное описание

* + - 1. **Технологии интеграции**

Интеграция с ЕСИА (получение данных о пользователе) реализуется через REST запросы к сервису единой идентификации и авторизации.

* + - 1. **Используемые конечные точки**

Рассмотрим запросы, отправляемые в ЕСИА и получаемые из ЕСИА.

* + - * 1. **Получение авторизационного кода**

GET <https://esia.gosuslugi.ru/aas/oauth2/ac>

некоторые параметры запроса:

* client\_id - идентификатор системы-клиента (мнемоника системы в ЕСИА);
* client\_secret – подпись запроса в формате PKCS#7 detached signature в кодировке UTF8 от значений четырех параметров HTTP–запроса: scope, timestamp, clientId, state (без разделителей). должен быть закодирован в формате base64 url safe. Используемый для проверки подписи сертификат должен быть предварительно зарегистрирован в ЕСИА и привязан к учетной записи системы-клиента в ЕСИА. ЕСИА поддерживает сертификаты в формате X.509. ЕСИА поддерживает алгоритмы формирования электронной подписи RSA с длиной ключа 2048 и алгоритмом криптографического хэширования SHA-256, а также алгоритм электронной подписи ГОСТ Р 34.10–2001 и алгоритм криптографического хэширования ГОСТ Р 34.11-94.
* redirect\_uri – ссылка, по которой должен быть направлен пользователь после того, как даст разрешение на доступ к ресурсу, в нашем случае: <https://esia.egov66.ru/auth/realms/rgis/broker/esia/endpoint>;
* scope – область доступа, т. е. запрашиваемые права; например, если система-клиент запрашивает доступ к сведениям о сотрудниках организации, то scope должна иметь значение http://esia.gosuslugi.ru/org\_inf (с необходимыми параметрами); если запрашивается scope http://esia.gosuslugi.ru/usr\_inf58 (данные о пользователе), то не нужно в качестве параметра указывать oid этого пользователя;
* state – набор случайных символов, имеющий вид 128-битного идентификатора запроса (необходимо для защиты от перехвата), генерируется по стандарту UUID;
* scope – область доступа, т.е. запрашиваемые права; передаётся наименование полей с личной информацией, пример: fullname, birthdate, snils, email¸ gender, citizenship, birthplace, addresses.

Если в ходе авторизации не возникло ошибок, то ЕСИА осуществляет редирект пользователя по ссылке, указанной в redirect\_uri, а также возвращает обязательные параметры:

code – значение авторизационного кода;

state – значение параметра state, который был получен в запросе на авторизацию; система-клиент должна провести сравнение отправленного и полученного параметра state.

* + - * 1. **Получение маркера доступа в обмен на авторизационный код**

Когда авторизационный код получен, система-клиент может сформировать запрос методом POST: <https://esia.gosuslugi.ru/aas/oauth2/te> для получения маркера доступа. В тело запроса должны быть включены следующие сведения:

client\_id – идентификатор системы-клиента (мнемоника системы в ЕСИА);

code – значение авторизационного кода, который был ранее получен от ЕСИА и который необходимо обменять на маркер доступа;

grant\_type – принимает значение “authorization\_code”, если авторизационный код обменивается на маркер доступа;

client\_secret – подпись запроса в формате PKCS#7 detached signature в кодировке UTF8 от значений четырех параметров HTTP–запроса: scope, timestamp, clientId, state (без разделителей). client\_secret должен быть закодирован в формате base64 url safe.

state – набор случайных символов, имеющий вид 128-битного идентификатора запроса (необходимо для защиты от перехвата), генерируется по стандарту UUID; этот набор символов должен отличаться от того, который использовался при получении авторизационного кода;

redirect\_uri – ссылка, по которой должен быть направлен пользователь после того, как даст разрешение на доступ (то же самое значение, которое было указано в запросе на получение авторизационного кода);

scope – область доступа, т.е. запрашиваемые права (то же самое значение, которое было указано в запросе на получение авторизационного кода);

timestamp – время запроса маркера в формате yyyy.MM.dd HH:mm:ss Z (например, 2013.01.25 14:36:11 +0400), необходимое для фиксации начала временного промежутка, в течение которого будет валиден запрос с данным идентификатором (<state>);

token\_type – тип запрашиваемого маркера, в настоящее время ЕСИА поддерживает только значение “Bearer”.

Если запрос успешно прошел проверку, то ЕСИА возвращает ответ в формате JSON:

access\_token – маркер доступа для данного ресурса;

expires\_in – время, в течение которого истекает срок действия маркера (в секундах);

state – набор случайных символов, имеющий вид 128-битного идентификатора запроса, генерируется по стандарту UUID (совпадает с идентификатором запроса);

token\_type – тип предоставленного маркера, в настоящее время ЕСИА поддерживает только значение “Bearer”;

refresh\_token – маркер обновления для данного ресурса.

Пример ответа:

{

“access\_token” :

“eyJhbGciOiJSUzI1NiIsInNidCI6ImFjY2VzcyIsInR5cCI6IkpXVCIsInZlciI6MX0.eyJleHAiOjEzNTk1NDAxODcsInNjb3BlIj

oiaHR0cDpcL1wvZXNpYS5nb3N1c2x1Z2kucnVcL2VtcF9pbmY\_b3JnX29pZD0xMDAwMDAwMzU3IiwiaXNzIjoiaHR0cDpcL1wvZXNpY

S5nb3N1c2x1Z2kucnUiLCJuYmYiOjEzNTk1MzY1ODcsInVybjplc2lhOnNpZCI6IjE2ZDdmOTNkLTZjZTgtNDE3OS04ZmFmLTdmZDQ2

ZDMyMDhhNiIsInVybjplc2lhOnNial9pZCI6MTAwMDAwMDM4NSwiY2xpZW50X2lkIjoiRVNJQSIsImlhdCI6MTM1OTUzNjU4N30”,

“expires\_in” : 3600,

“state” : “9be638a9-0e05-42e1-b4f8-a3e30457fbdd”,

“token\_type” : “Bearer”,

“refresh\_token” : “54039d1f-9917-43cd-961a-2729c891ef8c”

}

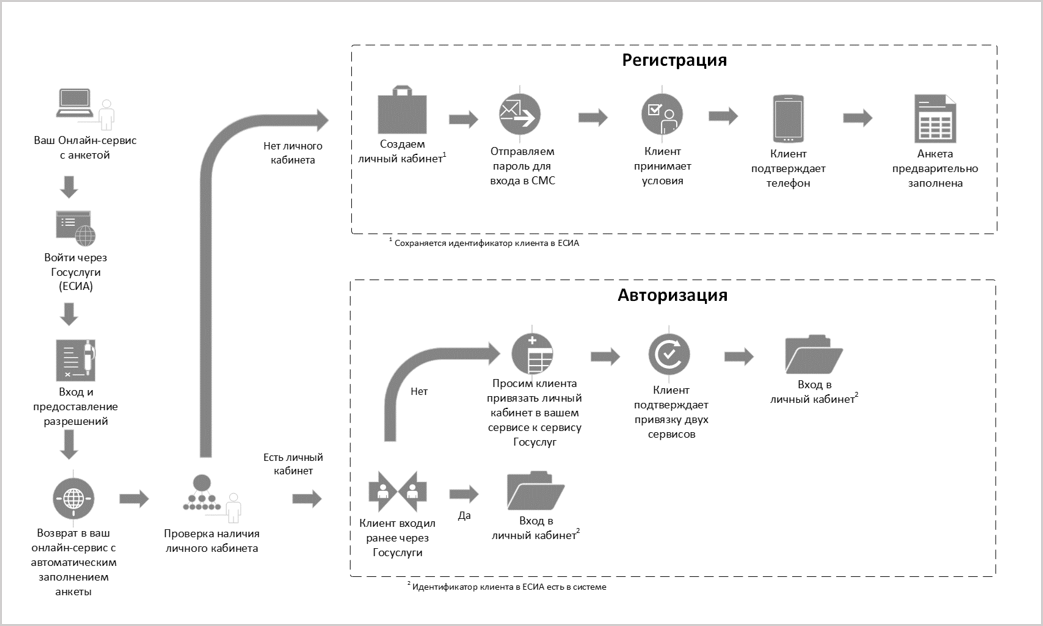


Рисунок 7 - Взаимодействие с ЕСИА

* + - 1. **Диаграмма классов сообщений – нужна ли?**
      2. **Рассмотрим логику обработки ответных пакетов:**
* **Квитанция о получении пакета**: логируется дата и время получения сообщения, меняется статус пакета.
* **Регистрационные данные**: логируется дата и время получения пакета с регистрационными данными, у заявления устанавливаются регистрационный номер и дата, статус заявления меняется на «Зарегистрировано»
* **Подписанный документ**: логируется дата и время получения пакета с подписью. В документ запроса устанавливается регистрационный номер, дата, обновляется наименование документа, статус документа меняется на "Подписан". Статус заявления меняется на Завершено. Так же в пакете присылают файл-документ с прикрепленной и открепленной подписью. Пакет сохраняется на ФХ нашей системы и прикрепляется к заявлению.

# Разработка технического решения

## Атрибутивный состав заявлений

Сведения о заявителе необходимы в каждой услуге, и они одинаковые, вынесем их в отдельный пункт, чтобы не повторяться.

* + 1. Сведения о заявителе

Заявитель – физическое лицо

* фамилия, имя, отчество,
* дата рождения,
* ИНН,
* СНИЛС,
* Адрес проживания
* Контактный телефон
* Электронная почта

Заявитель – юридическое лицо

* Полное наименование организации
* Кратное наименование организации
* ИНН
* КПП
* ОГРН
* Почтовый адрес
* Юридический адрес
* Контактный телефон
* Электронная почта

Заявитель – индивидуальный предприниматель

* Наименование
* ИНН
* ОГРН ИП
* Почтовый адрес
* Юридический адрес
* Контактный телефон
* Электронная почта
  + 1. Описание оформления права пользования лесным участком
* **Лесной участок**
* Вид использования лесов
* Цель, под которую испрашивается лесной участка
* Срок использования лесного участка
* Обоснование цели, вида, срока использования лесного участка
* Банковские реквизиты

Дополнительные документы при получении данной услуги не нужны.

* + 1. проведение государственной экспертизы
* Банковские реквизиты
* Правоустанавливающий документ
* Срок использования (до какой даты)

Для получения услуги необходимо предоставить проект освоения лесов

* + 1. согласование проекта рекультивации нарушенных земель/проекта лесовосстановления
* Исполнитель услуги
* Правоустанавливающий документ
* **Участок, подлежащий восстановлению**

Для получения услуги необходимо предоставить XML-файл проекта лесовосстановления.

* + 1. утверждение акта лесопатологического обследования
* Правоустанавливающий документ
* **Лесопатологический выдел**
* Кадастровый номер
* Способ лесопатологического обследования

Для получения услуги необходимо предоставить проект освоения лесов

* + 1. предоставление выписки из государственного лесного реестра
* Наименование запрашиваемой информации
* Уточнение запрашиваемых сведений
* **Местоположение**

Для подачи заявления обязательных документов нет, но для получения услуги необходимо предоставить документ-подтверждение оплаты

* + 1. прием лесной деклараций и отчетов об использовании лесов
* Номер лесной декларации
* Декларируемый период
* Наименование органа государственной власти, ОМСУ
* Правоустанавливающий документ
* Виды использования лесов
* **Объекты лесной инфраструктуры**
* Отметка о согласии на обработку персональных данных

Для получения услуги необходимо предоставить XML-файл декларации с приложениями

В большей части предоставляемых услуг заявителю необходимо приложить картографическую информацию (выделено жирным ранее), а сотруднику министерства необходимо проверить на пересечение с другими участками, водоохранными зонами, заповедниками и др.

Подтверждение, что система в промышленной эксплуатации: <https://midural.ru/news/vministerstvah/page1/document222264/>

RgisRoleAssignmenter <https://javarush.com/quests/lectures/questspringsecurity.level01.lecture10> – прочитать

Приложение

Создание участка:

Тело запроса

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Ответ на запрос о создании участкаИзображение выглядит как текст, снимок экрана, Параллельный, Шрифт

Автоматически созданное описание

ПРИЛОЖЕНИЕ 2

Запрос и ответ на создание участка

**Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описаниеИзображение выглядит как текст, линия, число, снимок экрана

Автоматически созданное описание**