Введение

Современный мир постоянно развивается. Частью этого процесса является веб-технологии, отрицание влияния которых невозможно. Все сферы жизни человека, будь то образование, медицина или политика используют современные наработки программистов. Бизнес так же не остался в стороне.

Одно из множества его проявлений – страхование. Страховой рынок содержит большое количество профессионально работающих компаний и перестраховочных обществ, в том числе иностранных и с участием иностранного капитала. Многие из данных организаций для оптимизации своей деятельности и экономии средств, стараются взять новые инструменты на вооружение.

Одной из самых затратных, как с денежной, так и временной точки зрения, является импорт данных в систему для дальнейшего хранения и использования ее. При большой количестве запросов на изменение данных, сервер может не выдержать и зависнуть.

Для того, чтобы не произошел во время задачи импорта, необходимо разработать специальный модуль, который будет оптимизировать добавление, обработки, а так же удаление данных из хранилища.

1 Анализ предметной области

* 1. Описание предметной области

Страхование – особый вид экономических отношений, призванный обеспечить страховую защиту людей и их интересов от различного рода опасностей. Суть страхования заключается в том, что человек имеет возможность получить перспективы своеобразной защиты от негативных факторов финансового плана. Услуги страхования предоставляют страховые компании.

Страховая компания — это [юридическое лицо](http://discovered.com.ua/glossary/yuridicheskoe-lico/) любой организационно-правовой формы, предусмотренной законодательством, созданное для осуществления страховой деятельности (страховая организация и общество [взаимного страхования](http://discovered.com.ua/glossary/vzaimnoe-straxovanie/)) и получившее в установленном законом порядке лицензию на осуществление страховой деятельности.

Законодательство допускает создание страховых компаний в виде [акционерных обществ](http://discovered.com.ua/glossary/akcionernoe-obshhestvo/), [обществ с ограниченной ответственностью](http://discovered.com.ua/glossary/obshhestvo-s-ogranichennoj-otvetstvennostyu/), государственных предприятий. В качестве специфической формы организации страхового дела предусматривается создание обществ [взаимного страхования](http://discovered.com.ua/glossary/vzaimnoe-straxovanie/).

Процесс сотрудничества со страховой компанией заключается в том, что вы подписываете договор, в рамках которого будете осуществлять определенные выплаты денежных средств за страховку. Если же случится неприятный инцидент, оговоренный в страховом договоре, то компания обязана произвести необходимые финансовые выплаты.

Таким образом, становится понятно, что страхование по своей сути предполагает определенный элемент защиты человека, его интересов и его гражданской ответственности. В настоящий момент появилось существенное количество компаний, которые способны предоставить вашему вниманию услуги страхования.

Проводя анализ предметной области, я выяснил, что множество компаний сталкиваются с проблемой быстрого импорта документов в базы данных. К сожалению, в настоящее время, даже с учетом технического прогресса, серверы не могут обрабатывать большое количество договоров бесперебойно. Серверное оборудование может не выдержать и зависнуть, что, по итогу, может привезти к огромным финансовым потерям. Для устранения данной проблемы было принято решение о создании модуля загрузки и обработки реестров договоров.

* 1. Основные понятия

Страхование — отношения по защите [имущественных интересов](https://ru.wikipedia.org/wiki/%D0%98%D0%BC%D1%83%D1%89%D0%B5%D1%81%D1%82%D0%B2%D0%B5%D0%BD%D0%BD%D1%8B%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D0%B5%D1%81_(%D1%81%D1%82%D1%80%D0%B0%D1%85%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) [физических](https://ru.wikipedia.org/wiki/%D0%A4%D0%B8%D0%B7%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B5_%D0%BB%D0%B8%D1%86%D0%BE) и [юридических лиц](https://ru.wikipedia.org/wiki/%D0%AE%D1%80%D0%B8%D0%B4%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B5_%D0%BB%D0%B8%D1%86%D0%BE) при наступлении определённых событий ([страховых случаев](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D1%80%D0%B0%D1%85%D0%BE%D0%B2%D0%BE%D0%B9_%D1%81%D0%BB%D1%83%D1%87%D0%B0%D0%B9)) за счет [страховых фондов](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D1%80%D0%B0%D1%85%D0%BE%D0%B2%D0%BE%D0%B9_%D1%84%D0%BE%D0%BD%D0%B4), формируемых из уплачиваемых ими страховых взносов.

Страховщик – понятие, которое определяет компанию частного или же государственного формата, которая осуществляет процесс страхования на основании четко проработанных и предусмотренных законодательных норм;

Страхователь – лицо, которое страхует свою ответственность посредством подписания договора. Данное же лицо осуществляет соответствующие выплаты;

Объект страхования - это именно тот элемент, который страхуется. По своей сути, это может быть движимое или же недвижимое имущество, денежные средства, здоровье или же жизнь. В зависимости от вида страхования формируется расчет стоимости взносов;

Договор – документ, который является основным элементом подтверждения наличия страхования;

Случаи страхового характера – возникновение действий, которые провоцируют возможность получения выплат от страховой компании; Возмещение – сумма, которую уплачивает компания в том случае, если возникает страховой случай.

1. Описание проблемы

Загрузка реестров, обработка, удаление реестров договоров – это сложный процесс, требующий мощного технического обеспечения. К сожалению, даже при наличии дорогой техники, нагрузка может быть настолько велика, что серверы могут не выдержать и повиснуть. Таким образом может остановиться процесс документооборота внутри компании.

Для того, чтобы избавиться от возможности остановить делопроизводство страховой компании необходимо использовать некоторые программные средства, например как брокер сообщений RabbitMq. Взаимодействие с модулем показано на рис.1.

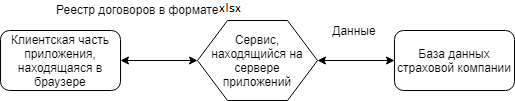


Рис 2.1 - Взаимодействие между элементами модуля импорта реестров договоров страховой компании.

1. Обзор аналогов

При анализе целесообразности создания данного модуля было проведено исследование, результатом которого стало известно, что ни один существующий аналоги не может быть использован. Это обуславливается тем, что не существует универсального функционала для работы с данными. Использование уже готовых программных решений приведет к тому, что многие части придется доделывать. Таким образом, вероятность того, что все будет работать успешно, уменьшается.

По этим причинам проще и выгоднее разработать собственное программное обеспечение для реализации потребностей конкретной системы. Таким образом, аналогов, способных реализовать аналогичный функционал, не существует.

4 Программная документация

4.1 Техническое задание на программное обеспечение

4.1.1 Общие сведения

4.1.1.1 Полное название системы и её условные обозначения

Полное наименование системы: Модуль импорта и корректировки реестров договоров страховой компании.

Краткое название системы: Модуль загрузки реестров.

4.1.1.2 Перечень документов, на основании которых создается системы, кем и когда утверждены эти документы

Основанием для разработки интеграционного шлюза являются следующие документы и нормативные акты:

1. Заказ для ООО «БиАйВи» на создание модуля и корректировки реестров договоров от 10 июня 2018 года. Заказчик: ООО СК «Сбербанк страхование»

2. Функциональные требования, присланные заказчиком

#### 4.1.1.3 Порядок оформления и предъявления заказчику результатов работы по созданию системы

Передается в виде функционирующего комплекса на базе средств вычислительной техники заказчика ООО СК «Сбербанк страхование» в сроки, установленные договором.

#### 4.1.1.4 Перечень нормативно-технических документов, методических материалов, использованных при разработке ТЗ

* ГОСТ 19.201-78. Техническое задание. Требования к содержанию и оформлению
* РД 50-34.698-90. Методические указания. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов

4.1.2 Назначение и цели создания системы

4.1.2.1 Назначение системы

Модуль импорта и корректировки реестров договоров страховой компании предназначен для автоматизации процесса работы с реестрами договоров, в части исполнения процессов ведения учета, обработки и добавления договоров, а так же нахождения ошибок, содержащихся в договоре

Модуль импорта и корректировки РД страховой компании предполагается использовать для упрощения процесса интеграции договоров.

4.1.2.2 Цели создания системы

Основными целями создания программного продукта являются упрощение ведения учета договоров, автоматизация в процессе обработки и добавления договоров, а так же повышение качества облуживания клиентов.

4.1.3 Требования к системе

4.1.3.1 Требования к структуре и функционированию системы

В состав модуль импорта и корректировки реестров договоров должны входить следующие компоненты:

* Модуль аутентификации
* Модуль загрузки данных
* Модуль обработки данных
* Модуль удаления данных

Структура частей модуля загрузки реестров показана на рисунках 2,3,4.

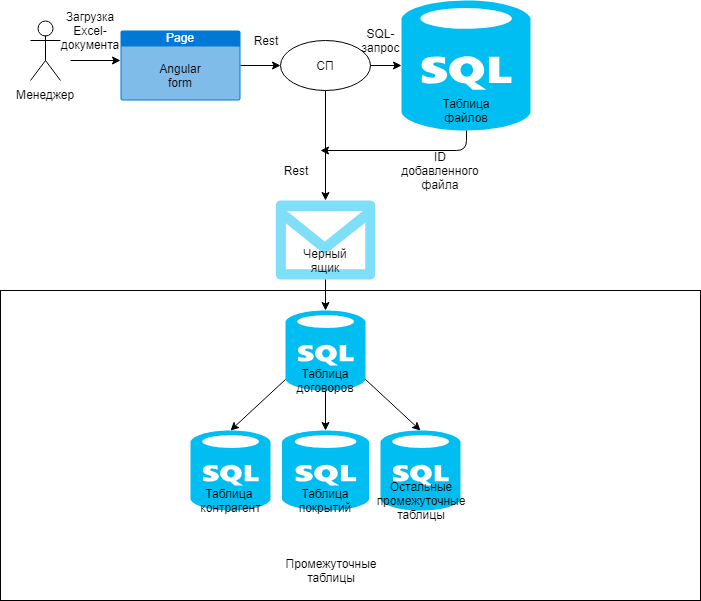


Рисунок 2. Архитектура подмодуля загрузки реестров договоров

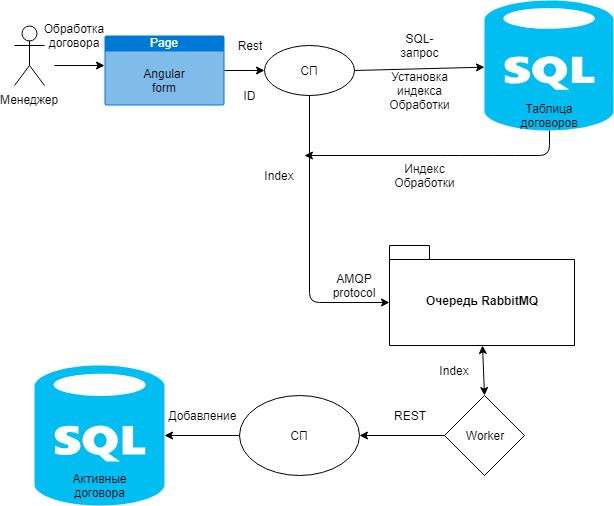


Рисунок 3. Архитектура подмодуля обработки реестров договоров / договора

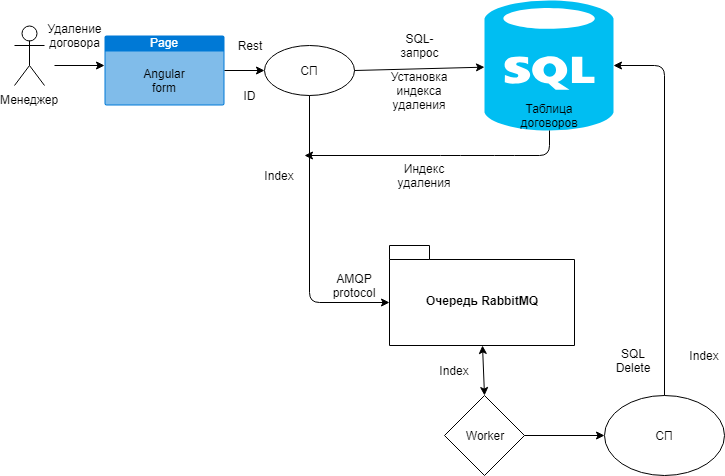


Рисунок 3. Архитектура подмодуля удаления реестра договоров / договора

Обмен данными между компонентами системы происходит с помощью REST-сообщений. Для работы с базой данных используется язык запросов SQL.

4.1.3.2 Общие требования

Ниже представлены требования, присланные заказчиком:

1. Разработка функционала должна соответствовать установленной архитектуре системы
2. Функционал должен использовать готовые микросервисы и окружение системы
3. Функционал должен использовать существующую структуру БД в части хранения договоров, продуктов и справочников

При вызове методов сервисов выполняется проверка:

* Авторизация пользователя
* Прав доступа к сервису для авторизованного пользователя
* Прав доступа к методам сервиса для авторизованного пользователя

4.1.3.3 Правила вызова веб-сервиса

Вызов методов Web-сервиса построен на двухуровневой модели запроса:

* Обращение – заголовок запроса
* Данные для обработки – содержимое запроса

Процесс разбора запроса к Web-сервису состоит из двух этапов:

1. Анализ обращения – прием данных и выполнение проверок
   1. Доступ авторизованного пользователя к методу сервиса
   2. Собственные проверки метода сервиса
2. Исполнение метода сервиса – анализ переданных данных, форматно-логический контроль и непосредственно обработка данных

Ответ Web-сервиса представляет собой двухуровневую структуру:

1. Отчет по обращению к сервису
2. Результат обработки переданных данных

Ключевым блоком для каждого уровня является «статус», в котором указывается:

1. Код
2. Текст
3. Детализация – список подробной информации по состоянию:
   1. Код
   2. Текст

4.1.3.4 Установленный формат ответа

…

Остальные пункты …

### 4.1.4 Перспективы развития, модернизации системы

Система должна реализовывать возможность дальнейшей модернизации как программного обеспечения, так комплекса технических средств.

Также необходимо предусмотреть возможность увеличения производительности системы путем её масштабирования.

4.1.5 Требования к информационному обеспечению системы

Уровень хранения данных в системе должен быть построен на основе современных реляционных или объектно-реляционных СУБД. Для обеспечения целостности данных должны использоваться встроенные механизмы СУБД. Средства СУБД, а также средства используемых операционных систем должны обеспечивать документирование и протоколирование обрабатываемой в системе информации.

Структура базы данных должна поддерживать кодирование хранимой и обрабатываемой информации в соответствии с общероссийскими классификаторами (там, где они применимы).

Доступ к данным должен быть предоставлен только авторизованным пользователям с учетом их служебных полномочий, а также с учетом категории запрашиваемой информации.

Структура базы данных должна быть организована рациональным способом, исключающим единовременную полную выгрузку информации, содержащейся в базе данных системы.

Технические должны средства, использовать обеспечивающие современные технологии, хранение информации, позволяющие обеспечить повышенную надежность хранения данных и оперативную замену оборудования (распределенная избыточная запись/считывание данных; зеркалирование; независимые дисковые массивы; кластеризация).

4.1.6 Требования к программному обеспечению системы

При проектировании и разработке системы необходимо максимально эффективным образом использовать ранее закупленное программное обеспечение, как серверное, так и для рабочих станций.

Используемое при разработке программное обеспечение и библиотеки программных кодов должны иметь широкое распространение, быть общедоступными и использоваться в промышленных масштабах.

4.1.7 Требования к техническому обеспечению

В состав технического обеспечения должны входить следующие технические средства:

- Сервер БД

- Сервер приложений

- Веб-сервер

- RabbitMq- сервер

- ПК пользователей

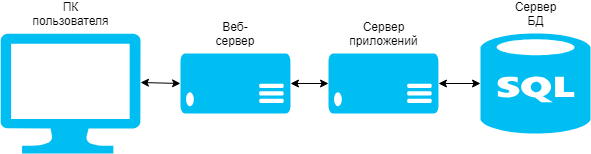


Рисунок 6. Схема взаимодействия технических средств

Серверы БД должны быть объединены в отказоустойчивый кластер. Серверы приложений должны образовывать кластер с балансировкой нагрузки.

Серверы БД, серверы приложений и сервер системы формирования отчетности должны быть объединены одной локальной сетью, с пропускной способностью не менее 100 Мбит.

Требования к техническим характеристикам серверов БД:

* Процессор – Intel Xeon Gold 6254 3.2 ГГц
* Объем оперативной памяти – 128 Гб
* Дисковая подсистема – 60 Тб
* Сетевой адаптер – 100 Мбит

Требования к техническим характеристикам системы хранения данных:

- Дисковая подсистема 0,5 Тб Raid Array 5

Требования к техническим характеристикам серверов приложений:

- Процессор – Intel Xeon 5120 2.2 ГГц;

- Объем оперативной памяти – 32 Гб;

- Дисковая подсистема – 1 Тб;

- Устройство чтения компакт-дисков (DVD-ROM);

- Сетевой адаптер – 100 Мбит.

Требования к техническим характеристикам веб - сервера:

- Процессор– [AMD Ryzen **Threadripper** 2950X](https://market.yandex.ru/product--protsessor-amd-ryzen-threadripper-2950x-colfax-str4-l3-32768kb/214768441?show-uid=15576723491050176095016004&nid=55330&context=search)3.5 ГГц;

- Объем оперативной памяти – 32 Гб;

- Дисковая подсистема – 4 Тб;

- Сетевой адаптер – 100 Мбит.

Требования к техническим характеристикам ПК пользователя

- Процессор – Intel Pentium 4560 3.5 ГГц;

- Объем оперативной памяти – 4 Гб;

- Дисковая подсистема – 40 Гб;

- Сетевой адаптер – 100 Мбит.

4.1.8.1 Требования к обеспечению надёжного (устойчивого) функционирования системы

Программа должна работать с надежностью 95 %.

Надёжное (устойчивое) функционирование программы должно быть обеспечено выполнением совокупности организационно-технических мероприятий, перечень которых приведён ниже:

* Организация бесперебойного питания сервера, на котором установлена система
* Организация технического обслуживания сервера, обеспечивающего бесперебойную работу всех его составляющих (как программных, так и аппаратных)
* Организация технического обслуживания сети в связи с необходимостью обеспечения бесперебойного доступа к системе через сеть.

4.1.8.2 Время восстановления после отказа

Время восстановления после отказа, вызванного сбоем электропитания технических средств или другими внешними факторами не должно превышать времени запуска всех служб системы-сервера, на котором установлена настоящая система.

4.1.9 Требования к безопасности

4.1.9.1 Защита от несанкционированного доступа

Для защиты от несанкционированного доступа система должна быть снабжена подсистемой авторизации. Для каждой системы необходимо создать учётную запись. Авторизация должна производиться по логину и паролю.

### 4.1.10 Требования к метрологическому обеспечению

Требования к метрологическому обеспечению не предъявляются.

## Пояснительная записка к программному обеспечению

### Анализ предметной области

Анализ предметной области приведен в разделе 1.

### Формулировка проблемы

Формулировка проблемы приведена в разделе 2.

### Обзор аналогов

Обзор аналогов выполнен в разделе 3.

### Назначение и область применения

Цели, назначение и области использования системы приведены в подразделах 4.1.2.1 и 4.1.2.2 технического задания соответственно.

Модуль импорта и корректировки реестров договоров – это программное решение, обеспечивающее оптимизация выполнению работы по внедрению, редактированию и учету договоров страхования, предоставляя разработчикам и администраторам унифицированные и централизованные средства разработки, тестирования и контроля протекания всех интеграционных сценариев.

4.2.5 Технические характеристики

4.2.5.1 Постановка задачи на разработку программы

Необходимо разработать программный продукт который будет:

1. Выполнять аутентификацию пользователя
2. Загружать документы в формате XML в клиентской части приложения в формате REST-сообщений
3. Отображать информацию о содержании БД в клиентской части приложения
4. Передавать и сохранять реестры договоров в БД с помощью SQL запросов
5. Позволять редактировать данные из реестров
6. Позволять удалять данные из реестров

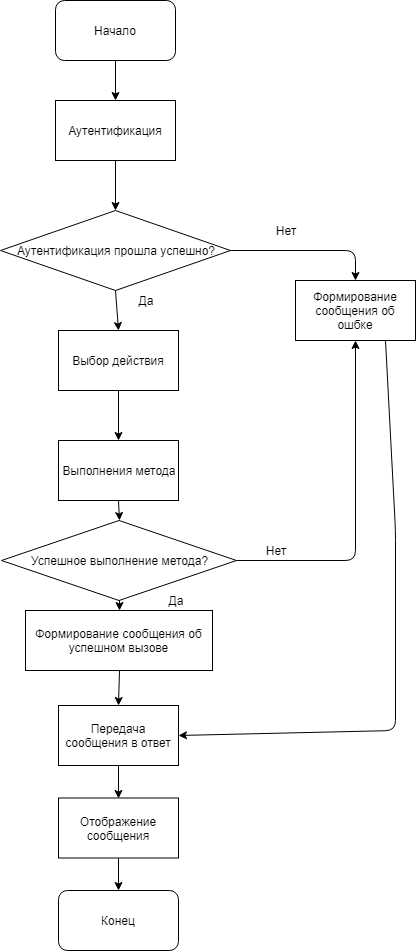
4.2.5.2 Описание алгоритма и функционирования программы

Общий алгоритм функционирования программы состоит из следующих этапов:

1. Аутентификацию
2. Выполнение возможных действий с реестрами: добавление, удаление, обработка
3. Отображение результатов действия

Также общий алгоритм программы представлен на рисунке 7.

Также общий алгоритм работы программы представлен на рисунке 7.



В качестве входных и выходных данных выступают REST-сообщения, в качестве протокола выступает HTTP.

4.2.5.3 Описание и обоснование выбора состава технических и программных средств

**Java** является одним из самых распространенных и популярных языков программирования. Первая версия языка появилась еще в 1996 году в недрах компании Sun Microsystems, впоследствии поглощенной компанией Oracle. Java задумывался как универсальный язык программирования, который можно применять для различного рода задач.

Ключевой особенностью языка Java является то, что его код сначала транслируется в специальный байт-код, независимый от платформы. А затем этот байт-код выполняется виртуальной машиной JVM (Java Virtual Machine). В этом плане Java отличается от стандартных интерпретируемых языков как PHP или Perl, код которых сразу же выполняется интерпретатором. В то же время Java не является и чисто компилируемым языком, как С или С++.   
  
 Подобная архитектура обеспечивает кроссплатформенность и аппаратную переносимость программ на Java, благодаря чему подобные программы без перекомпиляции могут выполняться на различных платформах - Windows, Linux, Mac OS и т.д. Для каждой из платформ может быть своя реализация виртуальной машины JVM, но каждая из них может выполнять один и тот же код.

Java является объектно-ориентированным языком. Он поддерживает полиморфизм, наследование, статическую типизацию. Объектно-ориентированный подход позволяет решить задачи по построению крупных, но в тоже время гибких, масштабируемых и расширяемых приложений.

Язык Java был выбран из за того, что это универсальный язык программировании, который используют сотни тысяч компаний разного масштаба в своем корпоративном серверном ПО, а так же то, что он работает на всех платформах.

**WildFly** (JBoss Application Server) – сервер приложений Java EE, полностью разработанным на [Java](https://ru.bmstu.wiki/Java" \o "Java), и, следовательно, может работать в любой операционной системе, как 32-битной, так и 64-битной. Основными характеристиками, на которых акцентируется внимание, являются возможности подключения, скорость отклика и масштабируемость, а основным фактором, стоящим за улучшениями в этих областях, является новый веб-сервер Undertow, который является более мощным, чем его конкуренты, такие как Jetty Web Server.

Управление памятью очень строгое, чтобы минимизировать максимальное выделение памяти кучи. Все сервисы, используемые WildFly, используют общие индексированные метаданные, которые кэшируются, чтобы избежать дублирования. Они также имеют модульную загрузку, которая предотвращает загрузку дублирующихся классов и должна загружаться поверх системы, необходимой для загрузки.

**Apache Maven** — фреймворк для автоматизации сборки проектов на основе описания их структуры в файлах на языке POM, являющемся подмножеством XML. Проект Maven издаётся сообществом Apache Software Foundation, где формально является частью Jakarta Project.   
 Maven обеспечивает декларативную сборку проекта. В файлах описания проекта содержится его спецификация, а не отдельные команды выполнения. Все задачи по обработке файлов, описанные в спецификации, Maven выполняет посредством их обработки последовательностью встроенных и внешних плагинов.

**Apache Cayenne** является общедоступным проектом под лицензией Apache, обеспечивающим объектно-реляционной отображение (ORM) и удаленное взаимодействие сервисов. Другими словами, это инструмент для разработчиков Java, которым необходимо обращаться к базе данных (или нескольким базам данных). Сayenne имеет множество уникальных и мощных возможностей, может решать широкий круг потребностей разработчиков. Он легко связывает одну или несколько схем баз данных непосредственно через Java объекты, управляет автоматическими фиксациями и откатами, генерирует SQL , объединения, последовательности, и многое другое.

**RabbitMQ** — программный брокер сообщений на основе стандарта AMQP — тиражируемое связующее программное обеспечение, ориентированное на обработку сообщений.  
 Состоит из сервера, библиотек поддержки протоколов HTTP, XMPP и STOMP, клиентских библиотек AMQP для Java и .NET Framework и различных плагинов (таких как плагины для мониторинга и управления через HTTP или веб-интерфейс или плагин «Shovel» для передачи сообщений между брокерами).

**Angular** представляет собой фреймворк от компании Google для создания клиентских приложений. Прежде всего он нацелен на разработку SPA-решений (Single Page Application), то есть одностраничных приложений. В этом плане Angular является наследником другого фреймворка AngularJS. В то же время Angular это не новая версия AngularJS, а принципиально новый фреймворк.   
  
 Angular предоставляет такую функциональность, как двустороннее связывание, позволяющее динамически изменять данные в одном месте интерфейса при изменении данных модели в другом, шаблоны, маршрутизация и так далее. Одной из ключевых особенностей Angular является то, что он использует в качестве языка программирования TypeScript. Одной из ключевых особенностей Angular является то, что он использует в качестве языка программирования TypeScript.

4.2.5.4 Ожидаемые технико-экономические показатели

Значительное повышение прибыли страховой компании, вследствие оптимизации документооборота внутри предприятия и уменьшения длительности обслуживания клиентов.

* 1. Описание программы
     1. Общие сведения

Модуль импорта и корректировки реестров договоров страховой компании имеет следующие атрибуты.

* Наименование файла веб-архива: sberimpdog.war
* Исходное имя файла: sberimpdog.war
* Внутреннее имя файла: sberimpdog
* Название продукта: sberimpdog
* Идентификатор группы Maven: com.bivgroup.ws
* Идентификатор артефакта Maven
* Версия 1.01.05

4.3.1.2 Программное обеспечение, необходимое для функционирования программы

Системные программные средства, используемые модулем загрузки реестров должны быть, представлены одной из следующих операционных систем: Windows 7 или выше, Ubuntu 16.04 LTS или выше, UbuntuServer 16.04 LTS или выше.

Так же для функционирования программного продукта необходимо следующее предустановленное программное обеспечение стороннего разработчика:

* Сервер приложений Wildfly версии 10 и выше
* Oracle JDK версии 8 и выше
* Сервер RabbitMQ

4.3.1.3 Языки программирования, на которых написана программа

Исходным языком программирования серверной части приложения является Java. В качестве языка используемого в клиентской части выступает TypeScript.

* + 1. Функционально назначение
       1. Классы решаемых задач

Модуль импорта и корректировки реестров договоров страховой компании - программное решение, обеспечивающее увеличение документооборота, предоставляя работника предприятия удобное средство для ведения учета договоров.

* + - 1. Назначение программы

Разрабатываемый программный продукт позволит снизить нагрузку на системы страховой компании, увеличит скорость учета договоров, а так же увеличит число обслуживаемых клиентов в единицу времени.

* + - 1. Сведения о функциональных ограничениях на применение

Модуль загрузки реестров предназначен для работы с протоколом HTTP для общения между серверной и клиентской частью, а так же с протоколом AMQP для взаимодействия с брокером сообщений RabbitMQ.

Разрабатываемое веб-приложение может запускаться только внутри сервера приложений.