

УТВЕРЖДЕН  
RU.17701729.06.12-01 34 01-1-ЛУ

**СЭД С РАСШИРЕННЫМ ВЕРСИОНИРОВАНИЕМ**  
**Руководство программиста**

**RU.17701729.06.12-01 34 01-1**

**Листов 13**

<i>Подп. и дата</i>	
<i>Инв. № дубл.</i>	
<i>Взам. инв. №</i>	
<i>Подп. и дата</i>	
<i>Инв. № подл</i>	RU.17701729.06.12-01 34 01-1

## ОГЛАВЛЕНИЕ

<b>1</b>	<b>ВВЕДЕНИЕ.....</b>	<b>4</b>
1.1.	НАИМЕНОВАНИЕ ПРОГРАММЫ.....	4
1.2.	КРАТКАЯ ХАРАКТЕРИСТИКА ОБЛАСТИ ПРИМЕНЕНИЯ ПРОГРАММЫ .....	4
<b>2</b>	<b>НАЗНАЧЕНИЕ ПРОГРАММЫ.....</b>	<b>5</b>
2.1.	ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ.....	5
2.2.	ЭКСПЛУАТАЦИОННОЕ НАЗНАЧЕНИЕ.....	5
<b>3</b>	<b>УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ .....</b>	<b>6</b>
3.1.	МИНИМАЛЬНЫЙ СОСТАВ АППАРАТНЫХ СРЕДСТВ .....	6
3.2.	МИНИМАЛЬНЫЙ СОСТАВ ПРОГРАММНЫХ СРЕДСТВ.....	6
3.3.	ТРЕБОВАНИЯ К ПЕРСОНАЛУ (ПОЛЬЗОВАТЕЛЮ).....	6
<b>4</b>	<b>ХАРАКТЕРИСТИКА ПРОГРАММЫ .....</b>	<b>7</b>
4.1.	АРХИТЕКТУРА СИСТЕМЫ .....	7
4.2.	ВЗАИМОДЕЙСТВИЕ КОМПОНЕНТОВ .....	10
4.3.	СХЕМЫ БАЗ ДАННЫХ .....	10
<b>5</b>	<b>ВЫПОЛНЕНИЕ ПРОГРАММЫ .....</b>	<b>11</b>
5.1.	ЗАПУСК ПРОГРАММЫ .....	11
5.2.	РАБОТА С ПРИЛОЖЕНИЕМ .....	11
	<b>СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....</b>	<b>12</b>

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.06.12-01 34 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## 1 ВВЕДЕНИЕ

### 1.1. Наименование программы

**Наименование темы разработки:** «СЭД с расширенным версионированием»

**Наименование темы разработки на английском языке:** «EDM With Advanced Versioning»

### 1.2. Краткая характеристика области применения программы

«СЭД с расширенным версионированием» - веб приложение, система электронного документооборота. Основная область применения программы – ведение документов, что включает: создание, редактирование, согласование, подписание и дальнейший ввод в действие. Включает в себя функционал гибкой настройки всех процессов ведения документов. Отличается расширенной системой версионирования шаблонов, матриц согласования и документов, а также кросс-доменной функциональностью.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.06.12-01 34 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## 2 НАЗНАЧЕНИЕ ПРОГРАММЫ

### 2.1. Функциональное назначение

Программа предоставляет возможность создавать различные шаблоны и гибко настраивать процессы ведения их документов, включая их схему состояний, атрибутивный состав и процессы согласования и подписания. Программа предоставляет функционал хранения, обработки и ведения документов по настроенным в шаблонах процессам.

Функциональное назначение серверной части – обработка запросов от клиентской части и хранение всех данных по документам, и связанными с ними настройками.

### 2.2. Эксплуатационное назначение

Программа является серверной частью приложения, системы для ведения внутреннего документооборота сотрудниками компании в локальной сети. Пользователи могут настраивать процессы документооборота, и, далее, работать с документами по настроенным процессам и конфигурациям. СЭД предполагает возможность интеграции с внутренними системами компании для унификации работы с договорами, заявками, compliance и любыми другими видами документов.

Эксплуатационное назначение серверной части заключается в предоставлении технологии и инфраструктуры, которые необходимы для использования через UI приложения, а также для обеспечения надежности, безопасности и масштабируемости системы.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.06.12-01 34 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

### 3 УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ

#### 3.1. Минимальный состав аппаратных средств

Доступ к интернету.

#### 3.2. Минимальный состав программных средств

- 1) Веб-браузер с поддержкой HTML5,
- 2) Docker desktop или docker engine новейшей версии.

#### 3.3. Требования к персоналу (пользователю)

Для корректной работы программы достаточно одного пользователя. Программист должен иметь базовые знания о веб программировании на любом языке программирования и базовые знания об HTTP и gRPC запросах. Так же необходимы базовые знания о ПО docker для корректного запуска API. Желательно уметь работать в Postman.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.06.12-01 34 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## 4 ХАРАКТЕРИСТИКА ПРОГРАММЫ

### 4.1. Архитектура системы

Система построена на микро-сервисной архитектуре и имеет разделена на три крупных слоя:

1. Клиентская часть (не реализуется в рамках данной ВКР)
2. Api Gateway
3. Api

Всего предусмотрено 2 разные клиентские части:

1. Матрицы согласования
2. Документы

Вся работа с шаблонами и с ведением документов находится на клиентской части по Документам, в свою очередь клиентская часть «Матрицы согласования» отвечает только за настройку правил определения маршрутов согласования по шаблонам.

Каждые клиент (клиентская часть) работает с серверной частью через соответствующие сервисы Api gateway:

1. edm-document-generator-gateway
2. edm-entities-approval-rules-gateway

Взаимодействие между клиентской и серверной частью построено на Rest Api через синхронные запросы по Http протоколу.

Предусмотрены следующие сервисы бэкенд серверной части:

1. edm-document-generator

Основная бизнес-логика приложения – настройка и работа с шаблонами, ведение документов

2. edm-document-searcher

Read модель по документам для быстрого и эффективного поиска. Взаимодействует с edm-document-generator асинхронно для получения синхронизации данных и синхронно с edm-document-generator-gateway

3. edm-document-classifier

Сервис ответственный за классификацию шаблонов и их документы и за все необходимые справочники. Взаимодействует синхронно с edm-document-generator-

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.06.12-01 34 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

gateway и edm-entities-approval-gateway для обогащения данных необходимой информацией.

#### 4. edm-entities-counters

Сервис для настройки счетчиков и генерации регистрационных номеров. Синхронное взаимодействие с edm-document-generator и edm-document-generator-gateway.

#### 5. edm-entities-approval-rules

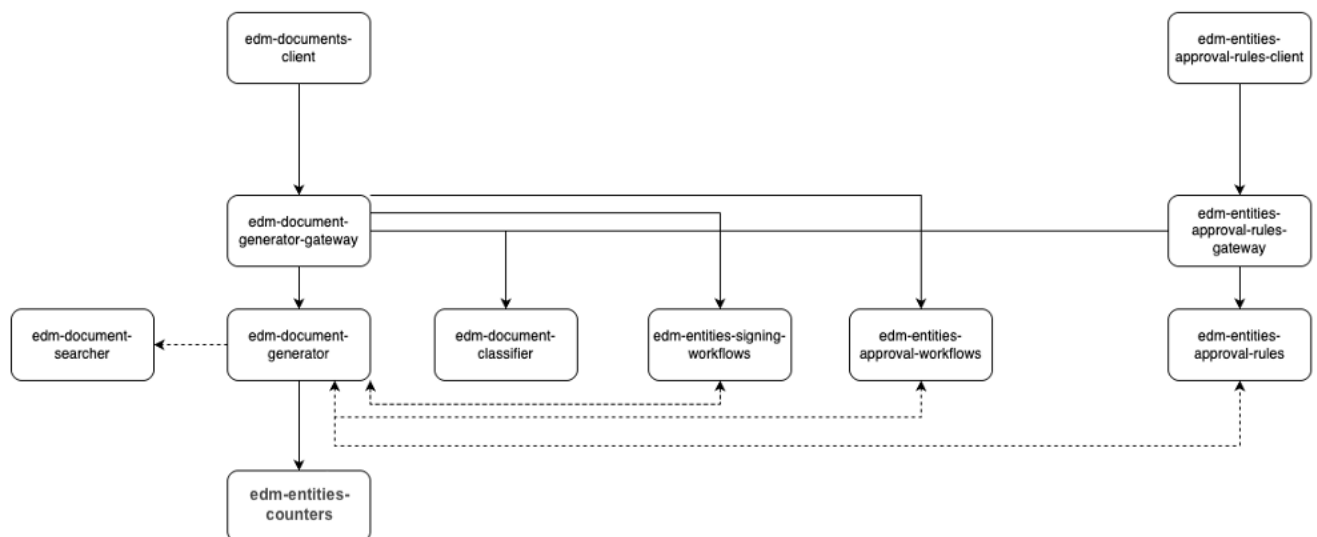
Сервис для работы с матрицами согласования. Работает асинхронно с edm-document-generator и edm-entities-approval-rules-gateway.

#### 6. edm-entities-approval-workflows

Сервис для ведения документа по флоу согласования на основе построенного edm-entities-approval-rules маршрута согласования. Асинхронное взаимодействие с edm-document-generator и синхронное с edm-document-generator-gateway.

#### 7. edm-entities-signing-workflows

Сервис для ведения документа по флоу подписания. Асинхронное взаимодействие с edm-document-generator и синхронное с edm-document-generator-gateway.



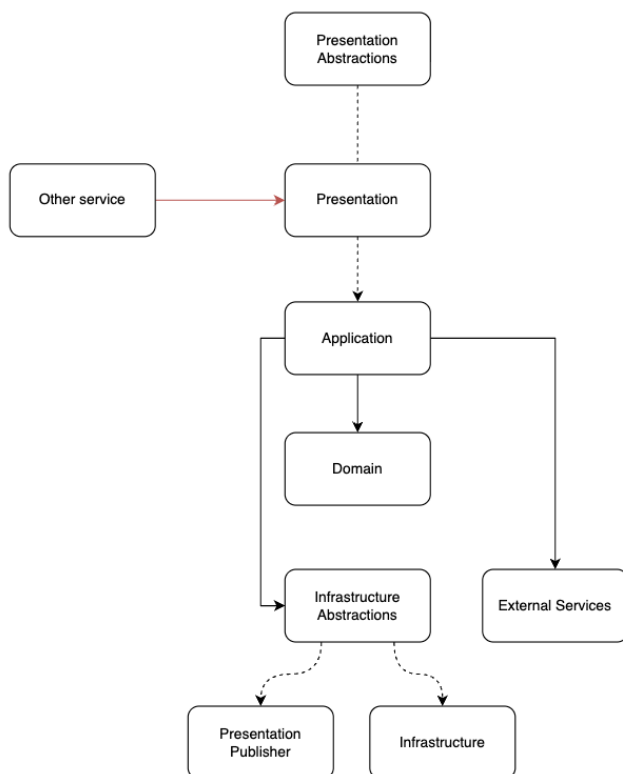
Приложение 1 – UML диаграмма общей архитектуры системы.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.06.12-01 34 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Два сервиса Api gateway реализуют луковую архитектуру и построены на 3 основных слоях: Presentation, Core, ExternalServices для взаимодействия с клиентской частью, бизнес логики взаимодействия с клиентской частью и для взаимодействия с внешними сервисами соответственно.

Каждый из сервисов бэкенд серверной части реализует чистую архитектуру со следующими слоями:

1. Presentation Abstractions – определение интерфейса для внешних взаимодействие;
2. Presentation – реализация внешнего интерфейса;
3. Application – слой по обработке запросов через Mediatr, которые приходят из Presentation слоя;
4. Domain – основной слой, определяющий бизнес сущность сервиса с учетом Domain Driven Design, содержит в себе всю бизнес логику приложения;
5. Infrastructure – слой работы с хранилищами данных;
6. Presentation Publisher – слой для публикации асинхронных событий;
7. ExternalServices – слой для синхронного и асинхронного взаимодействия с внешними сервисами.



Приложение 2 – UML Диаграмма архитектуры микро-сервисов серверной части.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.06.12-01 34 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



## 4.2. Взаимодействие компонентов

Синхронное взаимодействие между сервисами серверной части реализовано через gRPC, что обеспечивает эффективную работу за счет бинарной сериализации и низкой задержки, что критично для микро-сервисной архитектуры.

Асинхронное взаимодействие – наиболее предпочтительный тип межсервисного взаимодействия серверной части системы реализовано через Apache Kafka благодаря высокой пропускной способности и лучшей на рынке отказоустойчивости.

## 4.3. Схемы баз данных

Базы данных есть у каждого сервиса серверной части (Api). Для всех сервисов используется база данных PostgreSQL – является надежной и масштабируемой СУБД с богатым функционалом и наличием уникальных функций, таких как работа с jsonb.

Каждая база имеет 1 таблицу, которая названа по основной бизнес сущности сервиса и имеет следующие обязательные столбцы:

1. Id string,
2. Data bytea,
3. ConcurrencyToken timestamp.

База данных edm-document-searcher имеет схожую схему таблицы, однако тип поля Data вместо bytea – jsonb.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.06.12-01 34 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## 5 ВЫПОЛНЕНИЕ ПРОГРАММЫ

### 5.1. Запуск программы

Для запуска программы необходимо перейти в директорию проекта и открыть ее в терминале. Далее необходимо ввести команду в терминал “docker-compose up -d”. Установка произойдет автоматически

### 5.2. Работа с приложением

При запуске серверная часть начинает принимать запросы. Далее есть 2 варианта взаимодействия с ней:

1. Отправлять запросы по gRPC и сообщения в Kafka серверной части при необходимости интеграции с внешними системами;
2. Использовать клиентскую часть (необходимо реализовать), которая будет работать с api-gateway: edm-document-generator-gateway и edm-entities-approval-rules-gateway.

Для тестирования приложения через свое Api или Postman программист может отправлять запросы к сервисам, используя документацию swagger и контракты слов Presentation.Abstractions.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.06.12-01 34 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] ГОСТ 19.101-77 Виды программ и программных документов. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.;
- [2] ГОСТ 19.102-77 Стадии разработки. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.;
- [3] ГОСТ 19.103-77 Обозначения программ и программных документов. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.;
- [4] ГОСТ 19.104-78 Основные надписи. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.;
- [5] ГОСТ 19.105-78 Общие требования к программным документам. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.;
- [6] ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.;
- [7] ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.;
- [8] ГОСТ 19.603-78 Общие правила внесения изменений. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.;
- [9] ГОСТ 19.604-78 Правила внесения изменений в программные документы, выполненные печатным способом. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- [10] Docker. [Электронный ресурс] – URL: <https://www.docker.com/> режим доступа: свободный (дата обращения: 13.03.25).

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.06.12-01 34 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## Лист регистрации изменений

[illegible]

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.06.12-01 34 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата