KNOWLEDGE SPACE

Руководство администратора платформы KNOWLEDGE SPACE

ООО "ИНТЕГРИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ" | im.systems

Оглавление

Введение

Настоящий документ содержит описание порядка подготовки к работе с платформой KNOWLEDGE SPACE (далее – KS) по созданию бизнес-приложений.

Документ содержит следующие разделы:

- Первоначальное развертывание;
- Обновление версии;
- Интеграция в IT-ландшафт;
- Поддержание работоспособности;
- Резервное копирование и восстановление;
- Управление пользователями и ролями;
- Управление информационной безопасностью;
- Приложения. Общая архитектура системы.

1 Первоначальное развертывание

1.1 Используемые технологии и ПО

Backend

1. Тип архитектуры: Микросервисы
2. Основной язык программирования: Golang
3. Хранение данных: PostgreSQL / Postgres Pro 12+
4. Инфраструктурные сервисы:
   - Обнаружение сервисов – Consul 1.9.3 +
   - Очередь сообщений – RabbitMQ 3.8.9 +
   - Кеширование данных – Redis 6.2.1 +
   - Веб-сервер, reverse-proxy – Nginx 1.20 +

Frontend

1. Framework: AngularJS
2. Ключевые библиотеки:
   - Numeral.js
   - Moment.js
   - Apache ECharts
   - Draw.io

1.2 Минимальные системные требования

Backend

Требования к backend в значительной степени определяются назначением приложений, которые разрабатываются и функционируют в периметре платформы KS.

Минимальные требования к развертыванию сервера приложений:

1. Процессор: x64, 16 ядер (поколения 2014 года или новее)
2. Оперативная память: 32 ГБ
3. Жесткий диск: 80 ГБ
4. Операционная система:

- Ubuntu 16.04 / 20.04
  - Debian 4+
  - RHEL / Centos 7+
  - Windows Server 2016 / 2019
  - Red Hat (UBI)
  - Astra Linux
    - Потенциально могут быть использованы и прочие ОС семейства Linux, однако, тестирование работоспособности производилось на вышеуказанных.

Опционально: развертывание в контейнерах, например, Docker, Docker-compose.
Поставляемые образы Docker основаны на дистрибутиве Alpine Linux.

Минимальные требования к развертыванию сервера БД:

Сервер БД разворачивается отдельно:

1. Процессор: x64, 8 ядер
2. Оперативная память: 8 ГБ
3. Жесткий диск: 200 ГБ HDD (желательно SSD)
4. СУБД – PostgreSQL 13+

Frontend

Минимальные требования к клиентскому ПК:

1. Процессор: x86/x64, 2 ядра
2. Оперативная память: 4 ГБ
3. Поддерживаемые браузеры:
   - MS Edge 87 +
   - Apple Safari 12.1.2 +
   - Google Chrome 87 +
   - Yandex Browser 22 +

1.3 Способы развертывания

Платформа Knowledge Space может быть развернута в различных вариантах в зависимости от потребностей проекта и возможностей ИТ-инфраструктуры. Это могут быть варианты контейнеризации (docker и docker-compose) или с оркестрацией контейнеров на Kubernetes или OpenShift. Возможен вариант разворачивания без контейнеров.

При разворачивании через Docker на виртуальных серверах (VDS/VPS) настраиваются лимиты по CPU, памяти (ОЗУ, RAM) и дисковому пространству; на физических серверах используются доступные ресурсы сервера.

В Kubernetes платформа размещается в namespace с установленными квотами на ресурсы — CPU, память (ОЗУ, RAM) и дисковое пространство, выделяемые из общего пула кластера.

Пример развертывания KS на пространстве заказчика представлен в Приложении 2.

1.4 Развертывание в среде контейнеризации с отдельным СУБД

Рассмотрим контейнерный вариант развертывания (docker и docker-compose) для микросервисов KS и отдельный сервер СУБД, управляющий базами данных всех контейнеров.

Подготовительный этап

- **На сервере СУБД:**
  1. Установить операционную систему из списка ОС, поддерживаемых для установки СУБД PostgreSQL.
  2. Установить СУБД PostgreSQL 13+ (https://www.postgresql.org/download).
- **На сервере приложения:**

1. Установить операционную систему из списка п.1.2.1 текущего руководства.
2. Установить Docker 20.10.12+ (https://docs.docker.com/engine/install/debian/).
3. Установить Docker compose 1.29.0+ (https://docs.docker.com/compose/install/).

Необходимые требования к развертыванию

- Образы контейнеров KNOWLEDGE SPACE, поставляемые вендором;
- Файлы конфигурации (.env, docker-compose.yml, docker-compose.infra.yml, db-create.sql), поставляемые вендором;
- Доступ к глобальному или локальному репозиторию Docker для получения образов инфраструктурных сервисов.

Инфраструктурные сервисы могут быть развернуты вне среды контейнеризации.

Шаги по развертыванию

- **На сервере СУБД:**

1. Создать пустые базы данных для сервисов приложения (использовать файл из архива distrib.tar.gz ./db-create/db-create.sql):

```
psql -f ./db-create/db-create.sql <postgresLogin> <postgresPassword>
```

2. Изменить значения max_connections и shared_buffers:

```
psql -c "alter system set max_connections = 700;"
<postgresLogin> <postgresPassword> psql -c "alter system set shared_buffers = '512MB';"
<postgresLogin> <postgresPassword> psql -c "alter system set work_mem = '8MB';"
<postgresLogin> <postgresPassword>
```

Для применения параметров требуется рестарт PostgreSQL.

Предполагается, что СУБД развернута с настройками по умолчанию, это означает, что параметр **standard_conforming_strings** должен быть в значении **on** (начиная с PostgreSQL 9.1).

Если это не так, то необходимо выполнить следующее:

1. Перед развертыванием платформы в БД выполнить запрос:

```
ALTER SYSTEM SET standard_conforming_strings=on
```

2. На сервере БД выполнить команду pg_ctl restart

- **На сервере приложения:**

1. Установить инфраструктурные сервисы согласно рекомендациям вендоров:

Consul https://learn.hashicorp.com/tutorials/consul/reference-architecture

RabbitMQ https://www.rabbitmq.com/production-checklist.html

Redis https://redis.io/docs/manual/admin/

2. Если docker-образы микросервисов KS переданы архивом, потребуется их распаковка перед загрузкой в docker.

3. Загрузить образы контейнеров в Docker. Для загрузки всех образов, размещенных в директории images, можно использовать команду:

```
find ./images -type f -exec docker load -i {} \;
```

4. Скорректировать .env файл конфигурации:

   4.1 Внести актуальные данные для подключения сервисов к СУБД:

```
ИМЯ_СЕРВИСА_DB_HOST=адрес подключения к СУБД
ИМЯ_СЕРВИСА_DB_PORT=порт СУБД
ИМЯ_СЕРВИСА_DB_USER=пользователь СУБД
ИМЯ_СЕРВИСА_DB_PASSWORD=пароль СУБД
ИМЯ_СЕРВИСА_DB_DATABASE=название БД
```

   4.2 Внести актуальные данные для подключения сервисов к RabbitMQ:

```
RABBIT_ADDRESS=amqp://пользователь:пароль@хост:порт/vhost
```

   4.3 Внести актуальные данные для подключения сервисов к Consul:

```
CONSUL_HTTP_ADDR=хост:порт
```

   4.4 Внести актуальные данные для подключения сервисов к Redis:

```
REDIS_ADDRESS=хост:порт
```

5. Внести опциональные данные при развертывании контейнеров на разных сетевых адресах (для обеспечения сетевого взаимодействия):

```
ИМЯ_СЕРВИСА_HOST=адрес для подключения к этому сервису (если не указано – получается адрес сетевого интерфейса по умолчанию, например внутренний адрес в подсети Docker)
ИМЯ_СЕРВИСА_PORT=порт для подключения к этому сервису – должен быть выбран исходя из доступности для контейнера gateway для взаимодействия по сети (по умолчанию выбирается порт 3000)
```

   5.1 Внести параметры для подключения к системам логирования:

   Подключение к syslog:

```
SYSLOG_ADDRESS=адрес:порт сервера syslog
SYSLOG_NET=протокол подключения к syslog (tcp|udp)
```

   Подключение к graylog:

```
LOGS_HOST=адрес и порт сервера Graylog
LOGS_SEND=необходимость подключения Graylog (true|false)
```

6. Сгенерировать сертификат для https.

7. Скопировать файлы в /etc/ks/ssl. Данный каталог будет подключен в контейнер nginx как /ssl.

8. Настроить путь к сертификатам в конфигурационном файле

nginx: ./conf/nginx/default.conf:

```
ssl_certificate /ssl/cert/<СЕРТИФИКАТ>.crt
ssl_certificate_key /ssl/private/<ПРИВАТНЫЙ КЛЮЧ>.key
ssl_dhparam /ssl/cert/dhparam.pem
```

9. Запустить контейнер инфраструктурных сервисов:

   Nginx https://hub.docker.com/_/nginx

   Для загрузки образов должен быть доступ к hub.docker.com. Если доступа нет, то воспользоваться загрузкой образов из файлов (docker load):

```
docker-compose -f ./docker-compose.infra.yml up -d
```

10. Запустить сервисы приложения:

```
docker-compose -f ./docker-compose.yml up -d
```

11. Проверить состояние сервисов в consul. Подключиться из веб-браузера к пользовательскому интерфейсу consul (https://<APP>/ui). Должен отобразиться список из 33 сервисов. У всех должен быть зеленый индикатор.

1.5 Переменные окружения

Настройка микросервисов осуществляется через переменные окружения, указанные в таблице 1.

Таблица 1. Переменные окружения

№	Название	Default	Назначение	Тип	Сервис
1.	HOST	localhost	Сетевое имя хоста. Сервис будет зарегистрирован в Consul с использованием этого значения	строка	Все
2.	IP_ADDRESS	–	IP адрес хоста. Передается в Consul в качестве мета-информации	строка	Все
3.	PORT	3000	Номер порта сервиса	целое число	Все
4.	USE_SSL	false	Переключение схемы (false – http; true – https)	логический	Все
5.	DB_HOST	–	Сетевое имя хоста СУБД	строка	Все кроме aggregator, gateway, ws
6.	DB_PORT	–	Номер порта СУБД	строка	Все кроме aggregator, gateway, ws

7.	DB_USER	–	Логин пользователя для подключения к СУБД	строка	Все кроме aggregator, gateway, ws
8.	DB_PASSWORD	–	Пароль пользователя для подключения к СУБД	строка	Все кроме aggregator, gateway, ws
9.	DB_DATABASE	–	Имя базы данных	строка	Все кроме aggregator, gateway, ws
10.	DB_DEBUG	–	Добавляет в лог sql.query; sql.args; sql.debug	логический	Все кроме aggregator, gateway, ws
11.	DB_DEBUG_LOG	–	Флаг включает логирование всех запросов к БД	логический	Все кроме aggregator, gateway, ws
12.	DB_LOG_LEVEL	1	Уровень логирования работы с базой данных (1 – Silent; 2 – Error; 3 – Warn; 4 – Info)	целое число	Все кроме aggregator, gateway, ws
13.	DB_SLOW_THRESHOLD	30000	Порог длительности выполнения SQL запроса для отнесения его к медленным. Значение в миллисекундах	целое число	Все кроме aggregator, gateway, ws
14.	DB_SSL	disable	Значение будет подставлено в строку подключения к базе данных в параметр sslmode	строка	Все кроме aggregator, gateway, ws
15.	DB_MAX_IDLE_CONNECTIONS	0	Максимальное количество неактивных (idle) соединений в пуле. Это соединения, которые установлены с БД, но в данный момент не используются. Поддержание idle-соединений позволяет быстрее выполнять новые запросы, так как не тратится время на установление соединения	целое число	–
16.	DB_MAX_OPEN_CONNECTIONS	0	Максимальное количество открытых соединений с БД (активных и idle вместе). Этот параметр ограничивает общее число одновременных соединений с базой данных	целое число	–

17.	DebugEmail	false	Включение/отключение отладочного режима для почтового сервиса	логический	–
18.	PPROF	false	Флаг добавления endpoint-ов получения информации для профилировщика	логический	Все
19.	PPROF_INNER	false	Для ребилда модели будет сохранять информацию для профилировщика (CPUProfile) в файл. Путь к файлу /pprof/compute_model__	логический	models_computer
20.	JAEGER_AGENT_HOST	localhost	Сетевой адрес хоста Jaeger агента. К адресу будет дописан порт: 6831	строка	Все
21.	JAEGER_TRACING	false	Флаг включения трассировки запросов в Jaeger	логический	Все
22.	RABBIT_ADDRESS	–	Строка подключения к RabbitMQ. Пример: amqp://admin:admin@rabbitmqhost:5672/vhost	строка	Все кроме aggregator, gateway
23.	REDIS_ADDRESS	–	Строка подключения к redis в формате хост:порт	строка	Все кроме aggregator, gateway
24.	REDIS_PASSWORD	–	Пароль для подключения к Redis	строка	Все кроме aggregator, gateway
25.	AUTH_TOKEN_TTL_MIN	–	Время жизни токена авторизации в минутах. Есть аналогичная настройка в системных параметрах. Значение в БД имеет приоритет	целое число	–
26.	AUTH_SIGNING_KEY	–	Ключ для подписи токена авторизации	строка	–
27.	AUTH_RECOVERY_TOKEN_TTL_HOURS	–	Время жизни токена восстановления пароля учетной записи в часах	целое число	auth
28.	AUTH_ONLY_ONE_ACTIVE_SESSION	false	Если true, то допускается только одна действующая сессия в системе	логический	–
29.	CSRF_SIGNING_KEY	–	Ключ для подписи CSRF токена	строка	–

30.	CSRF_TRUSTED_HOSTS	localhost	Список доверенных доменов для обхода CSRF-защиты при кросс-доменных запросах	массив строк	–
31.	PASSWORD_LIFETIME	25	Продолжительность действия пароля в днях. Есть системная настройка, которая имеет приоритет над значением в ENV	целое число	–
32.	FRONTEND_DOMAIN	http://localhost:3000	Используется для генерации ссылок. В уведомлениях bpms, ссылка восстановления пароля, редирект на логин при неуспешной win-авторизации	строка	–
33.	STORAGE_FILE_SYSTEM_LOCATION	–	Путь для хранения файлов на диске	строка	files
34.	UUID_KEY	–	Постоянная составляющая при генерации uuid-ов. Для различных инстансов сервиса должен отличаться для исключения дублирования uuid- ов	строка	Все
35.	DEBUG_MAIL	false	Флаг отправки отладочного электронного письма, если не найден шаблон	логический	mail_processor
36.	DEBUG_NOTIFY	false	Флаг отправки отладочного сообщения в чат, если не найден шаблон	логический	mail_processor
37.	WS_MAX_CONNECTIONS	–	Количество одновременных подключений клиентов к WebSocket	целое число	–
38.	SMTP_ADDRESS	–	Сетевой адрес SMTP сервера. Ожидает номер порта после двоеточия	строка	mail_processor
39.	SMTP_LOGIN	–	Логин для аутентификации на SMTP/NTLM сервере	строка	mail_processor
40.	SMTP_SENDER	–	Отправитель писем	строка	mail_processor
41.	SMTP_PASSWORD	–	Пароль для аутентификации на SMTP сервере/NTLM сервере	строка	mail_processor

42.	SMTP_USE_SSL	false	Использовать SSL	логический	mail_processor
43.	SMTP_SKIP_VERIFY	true	Значение определяет, проверяет ли клиент цепочку сертификатов сервера и имя хоста	логический	mail_processor
44.	SMTP_CERT	–	Сертификат для подключения TLS	строка	mail_processor
45.	SMTP_KEY	–	Приватный ключ для подключения TLS	строка	mail_processor
46.	SMTP_CA_CERT	–	Корневой сертификат	строка	mail_processor
47.	SMTP_AUTH_TYPE	LOGIN	Возможные значения: «LOGIN» – аутентификация по паре логин/пароль «NTLM» - аутентификация по протоколу NTLM	строка	mail_processor
48.	LOGS_HOST	–	Сетевой адрес агрегатора логов (graylog) в формате хост:порт. Передача по UDP в формате GELF. Пример: graylog:12201	строка	Все
49.	LOGS_SEND	–	Переключатель отправки логов в агрегатор (graylog)	логический	Все
50.	LOGS_LEVEL	0	Уровень логирования в поток вывода. Значения: -1 DebugLevel 0 InfoLevel 1 WarnLevel 2 ErrorLevel	целое число	Все

51.	LOGS_SEND_LEVEL	0	Уровень логирования в агрегатор логов (graylog). Значения: -1 DebugLevel 0 InfoLevel 1 WarnLevel 2 ErrorLevel	целое число	Все
52.	LOGS_CONNECTION_TIMEOUT	2m	Тайм-аут подключения к syslog серверу	строка	–
53.	LOGS_SEND_UNCOMPRESSED	false	Возможность отключить сжатие при отправке логов в syslog сервер	логический	–
54.	MDC_MAX_WORKERS	–	Ограничивает количество параллельных пересчетов моделей/наборов данных	целое число	–
55.	MDC_CALC_SAVER_WORKERS	3	Количество одновременных процессов сохранения данных в базу при пересчете	целое число	–
56.	OPEN_MESH_ENABLED	false	Отключает Consul. Для использования service mesh (OpenShift)	логический	Все
57.	CONSUL_INTERVAL	20s	Частота проверки consul микросервиса (healthCheck)	строка	Все
58.	CONSUL_DEREGISTER	5m	Время до удаления «больного» микросервиса из реестра consul	строка	Все
59.	CONSUL_HTTP_ADDR	–	Адрес и порт для подключения к Consul HTTP API	строка	–
60.	PASSWORD_RULES	–	Regexp выражение для проверки надежности пароля. Будет тестировать пароль, если нет параметров для пароля в настройках платформы	строка	–
61.	SYSLOG_ALL	false	Логирование в SYSLOG. Не включится с пустым SYSLOG_ADDRESS	логический	–
62.	SYSLOG_ADDRESS	–	Сетевой адрес SYSLOG сервера	строка	–

63.	SYSLOG_NET	tcp	The specified network. Пустое значение = запись в локальный SYSLOG сервер	строка	–
64.	EVENT_JOURNAL_NAME	KS	Этим значением будет заполняться поле системного журнала JournalName	строка	system_event_journal
65.	RENAME_POLICIES_FILE_URL	–	URL к файлу CSV переименования маршрутов	строка	users
66.	ENCRYPTOR_SIGNING_KEY	–	Ключ шифрования чувствительных данных. Пароли подключения к внешним источникам сервиса интеграций	строка	integrator
67.	ENABLE_PROMETHEUS_PROFILING	false	Добавляет endpoint /metrics	логический	Все
68.	HIDE_PERSONAL_DATA	false	Замена персональных данных на <скрыто>	логический	–
69.	GRAYLOG_API_ADDRESS	–	Сетевой адрес сервера GrayLog. Ошибки истории пересчетов	строка	models_computer
70.	GRAYLOG_API_USER	–	Логин для доступа к серверу GrayLog. Ошибки истории пересчетов	строка	models_computer
71.	GRAYLOG_API_PASSWORD	–	Пароль для доступа к серверу GrayLog. Ошибки истории пересчетов	строка	models_computer
72.	NATS_URL	–	URL подключения к NATS серверу	строка	–
73.	NATS_CLUSTER	–	Название кластера NATS	строка	–
74.	NATS_ORDERED	false	Значение влияет на способ выборки сообщений из очереди NATS. TRUE = по одному сообщению для обеспечения последовательности обработки сообщений. FALSE = до 1024 сообщений	логический	–
75.	DEFAULT_LANGUAGE	–	Язык по умолчанию	строка	–
76.	MIGRATION_FILE_URL	–	Если доступно подключение к репозиторию IMS, прописать адрес и креды для импорта фигур в	строка	diagrams

			переменную окружения для сервиса Diagrams		
77.	MIGRATION_FILE_URL_LOGIN	–	Если доступно подключение к репозиторию IMS, прописать адрес и креды для импорта фигур в переменную окружения для сервиса Diagrams	строка	diagrams
78.	MIGRATION_FILE_URL_PASSWORD	–	Если доступно подключение к репозиторию IMS, прописать адрес и креды для импорта фигур в переменную окружения для сервиса Diagrams	строка	diagrams
79.	MIGRATION_FILES_FOLDER	–	Если недоступно подключение к репозиторию IMS - скачать файлы вручную из репозитория и положит в папку рядом с исполняемым файлом сервиса, указать путь к папке в переменной окружения сервиса Diagrams	строка	diagrams
80.	WS_PUBLISH_USERS_ONLINE_WORKER_DELAY	10s	Определяет интервал задержки между отправкой запросов в /users/update-online для фиксации факта подключения/отключения пользователя	строка	ws
81.	BACKGROUND_OPERATIONS_TIMEOUT	10m	Таймаут ожидания фоновых операций во время восстановления бэкапа в существующий проект (например ожидание пересчета модели)	строка	Все
82.	BACKUP_ATOMIC_OPERATION_TIMEOUT	10m	Таймаут атомарной операции при восстановлении бэкапа (например восстановление одной пачки данных data). Служит для разблокировки моделей, которые заблокированы для осуществления восстановления бэкапа в сущ. проект	строка	Все
83.	AUTH_EXPIRED_SESSIONS_DELETE_INTERVAL	24ч	Настройка интервала удаления сессий приложений	целое число	auth
84.	AUTH_KERBEROS_KEYTAB_PATH	false	Указывает путь к keytab файлу, для аутентификации по протоколу kerberos	строка	auth

85.	ENABLE_RABBIT_CHECKER	false	Определяет вкл/выкл логирования для RabbitMQ, в случаях различных проблем с БД RabbitMQ. По умолчанию логирование выключено	логический	Все
86.	OPEN_MESH_ENABLED	false	Если установить "true", то сервисы не будут пытаться подключаться к Consul. Попытки разрешить адреса нужных соседних сервисов будут осуществляться с использованием DNS-имён	логический	Все
87.	OPEN_MESH_PROTOCOL	http	Протокол, по которому в данном режиме будут производиться попытки обращения к соседним сервисам	строка	Все
88.	OPEN_MESH_HOST_SUFFIX	_ADDRESS	Суффикс для переменной окружения, значение которой будет использоваться для получения DNS-имён сервисов. Например для сервиса models_data_computer имя переменной окружения при значении по умолчанию будет "MODELS_DATA_COMPUTER_ADDRESS"	строка	Все
89.	OPEN_MESH_PORT_SUFFIX	""	Суффикс для переменной окружения, значение которой будет использоваться для получения портов сервисов. Например для сервиса models_data_computer имя переменной окружения при значении по умолчанию будет "MODELS_DATA_COMPUTER_PORT". Если данный суффикс не задан, то в качестве адреса соседнего сервиса будет использоваться значение, полученное с использованием имени сервиса и переменной OPEN_MESH_HOST_SUFFIX без дополнений. Если данный суффикс задан, то в качестве адреса соседнего сервиса будет использоваться значение, полученное объединением хоста и порта через ":"	строка	Все
90.	MDC_CALC_OUT_OF_RA	false	Ограничивает область действия формулы при наличии лага	логический	mdc

	NGE_PERIODS				

При конфигурации указываются значения переменных окружения, после этого контейнеры запускаются с указанными значениями.

Значения для разных микросервисов могут быть добавлены в один .env файл с использованием префиксов с именем микросервиса. Например, USERS_PORT и CLASSES_PORT.

1.6 Первичная настройка

Для обеспечения работоспособности и безопасности системы необходимо:

1. Выполнить вход в систему под учетной записью администратора (логин и пароль – admin);
2. В профиле пользователя измените стандартный пароль администратора на собственный;
3. В меню «Настройки» заполнить необходимые параметры журналирования, парольную политику, параметры подключения в Active Directory;
4. При необходимости, создайте новых пользователей и заблокируйте стандартных.

1.7 Запуск платформы

- Каждый микросервис - это отдельный бинарный исполняемый файл, не имеющий внешних зависимостей и не требующий специальных разрешений в системе, поэтому может запускаться от имени обычного пользователя;
- Параметры конфигурации передаются микросервисам через переменные окружения;
- Нет необходимости строго определять последовательность запуска микросервисов, т.к. они обнаруживают друг друга с помощью Hashicorp Consul;

Зависимости:

- Hashicorp Consul
- PostgreSQL
- RabbitMQ
- Redis

Запуск платформы:

1. запустить Consul, Postgres, RabbitMQ, Redis;
2. создать базы данных Postgres в соответствии со списком сервисов, при необходимости завести роли и назначить им права db_owner;
3. заполнить значения переменных окружения в соответствующих конфигурационных файлах;
4. запустить микросервисы платформы.

1.7.1. Список сервисов

```
aggregator
approval
auth
backups
batcher
bpms
calendars
chats
classes
```

```
dashboards
data_tables
descriptions
diagrams
dictionaries
entities_hierarchy
files
gantts
gateway
heartbeat
imports_exports
integrator
l10n
mail_processor
models
models_data_computer
objects
projects
publications
restrictions
system_events_journal
system_settings
users
ws

Для сервисов aggregator и ws создание БД не требуется
```

1.7.2. Пример сервиса systemd

```
[Unit]
Description=Knowledge Space approval microservice
After=network-online.target
Wants=network-online.target

[Service]
Type=simple
EnvironmentFile=-/opt/ks/etc/approval.conf
ExecStart=/opt/ks/bin/approval
ExecReload=/bin/kill -HUP $MAINPID
KillMode=process
Restart=always
RestartSec=10
User=ks
Group=ks

[Install]
WantedBy=multi-user.target
```

1.7.3. Пример сервиса docker compose

```
classes:
   container_name: classes
   image: ${REGISTRY_URL}classes
   restart: always
   networks:
     - ks
   environment:
     - SERVICE_NAME=classes
```

```
    - CONSUL_HTTP_ADDR=$CONSUL_HTTP_ADDR
    - RABBIT_ADDRESS=$RABBIT_ADDRESS
    - REDIS_ADDRESS=$REDIS_ADDRESS
    - CLASSES_DB_DATABASE=$CLASSES_DB_DATABASE
    - DB_HOST=$DB_HOST
    - DB_USER=$DB_USER
    - DB_PASSWORD=$DB_PASSWORD
    - DB_PORT=$DB_PORT
    - DB_STATS_INTERVAL=$DB_STATS_INTERVAL
    - DB_MAX_OPEN_CONNECTIONS=$DB_MAX_OPEN_CONNECTIONS
    - DB_MAX_IDLE_CONNECTIONS=$DB_MAX_IDLE_CONNECTIONS
    - AUTH_SIGNING_KEY=$SIGNING_KEY
    - UUID_KEY=$UUID_KEY
```

1.7.4. Пример конфигурации nginx

```
server {
   listen                    :80 default_server;
   server_name                  knowledge.space.host;
   return                     301 https://$host$request_uri;
}

server {
   server_name                  knowledge.space.host;
   listen                    :443 ssl default_server;
   ssl_certificate             /etc/nginx/ssl/ssl.crt;
   ssl_certificate_key          /etc/nginx/ssl/ssl.key;
   include                   /etc/nginx/ssl/options-ssl-nginx.conf;
   ssl_dhparam                  /etc/nginx/ssl/ssl-dhparams.pem;

   client_max_body_size         100m;
   server_tokens               off;

   proxy_connect_timeout         300;
   proxy_send_timeout           300;
   proxy_read_timeout           300;
   send_timeout                300;

   proxy_set_header             X-Real-IP $remote_addr;
   proxy_set_header             X-Forwarded-For $proxy_add_x_forwarded_for;

   location / {
      proxy_pass             http://127.0.0.1:8080/; # Адрес FRONTEND
      proxy_redirect          default;
   }

   location /api/ {
      proxy_pass             http://127.0.0.1:5000/; # Адрес GATEWAY
      proxy_redirect          default;
   }

   location /ws/ {
      proxy_pass             http://127.0.0.1:5001/; # Адрес WS
      proxy_http_version        1.1;
      proxy_set_header          Upgrade $http_upgrade;
      proxy_set_header          Connection "upgrade";
   }

   location /download/ {
      proxy_pass             http://127.0.0.1:5000/files/download/; # Адрес GATEWAY, путь /files/download
      break;
```

```
    }

    location /drawexportpdf {
        proxy_pass                  http://127.0.0.1:8899/; # Адрес сервиса выгрузки диаграмм
    }
}
```

1.7.5. Пример docker-compose.yml для локального запуска контейнеров инфраструктуры

```
services:
  consul:
    container_name: consul
    image: hashicorp/consul:1.20
    network_mode: host
    restart: always
    volumes:
      - "/opt/ks/external/consul/data:/consul/data"
  postgres:
    container_name: postgres
    image: postgres:17
    restart: always
    shm_size: 2g
    network_mode: host
    environment:
      - "POSTGRES_PASSWORD=postgres"
    volumes:
      - "/opt/ks/external/postgres:/var/lib/postgresql/data"
  redis:
    container_name: redis
    image: redis:7-alpine
    network_mode: host
    restart: always
  rabbitmq:
    container_name: rabbitmq
    image: rabbitmq:4.1.0-management-alpine
    network_mode: host
    restart: always
    volumes:
      - "/opt/ks/external/rabbitmq/config:/etc/rabbitmq"
      - "/opt/ks/external/rabbitmq/data:/var/lib/rabbitmq"
      - "/opt/ks/external/rabbitmq/log:/var/log/rabbitmq"
```

1.7.6. Пример файла .env, содержащего значения переменных окружения

```
MASTER_ENCRYPTION_KEY=some-32-bytes-long-encryption-secure-key
CSRF_SIGNING_KEY=some-32-bytes-long-signing-secure-key
UUID_KEY=ALFcDEAA

CONSUL_HTTP_ADDR=127.0.0.1:8500
CONSUL_LOG_LEVEL=INFO

RABBIT_ADDRESS=amqp://login:password@127.0.0.1:5672/vhost

REDIS_ADDRESS=127.0.0.1:6379

FRONTEND_DOMAIN=https://frontend.domain

DB_HOST=127.0.0.1
```

```
DB_PORT=5432
DB_USER=postgres
DB_PASSWORD=postgres
DB_MAX_IDLE_CONNECTIONS=10
DB_MAX_OPEN_CONNECTIONS=10
DB_STATS_INTERVAL=1m

AGGREGATOR_PORT=3001

APPROVAL_PORT=3002
APPROVAL_DB_DATABASE=ks_approval

AUTH_PORT=3003
AUTH_DB_DATABASE=ks_auth
AUTH_TOKEN_TTL_MIN=60
AUTH_RECOVERY_TOKEN_TTL_HOURS=48
AUTH_SIGNING_KEY=KSx63gcpat5VAXauwSPr

BACKUPS_PORT=3004
BACKUPS_DB_DATABASE=ks_backups

BATCHER_PORT=3005
BATCHER_DB_DATABASE=ks_batcher

BPMS_PORT=3006
BPMS_DB_DATABASE=ks_bpms

CALENDARS_PORT=3007
CALENDARS_DB_DATABASE=ks_calendars

CHATS_PORT=3008
CHATS_DB_DATABASE=ks_chats

CLASSES_PORT=3009
CLASSES_DB_DATABASE=ks_classes

DASHBOARDS_PORT=3010
DASHBOARDS_DB_DATABASE=ks_dashboards

DATA_TABLES_PORT=3011
DATA_TABLES_DB_DATABASE=ks_data_tables

MODELS_DATA_COMPUTER_PORT=3012
MODELS_DATA_COMPUTER_DB_DATABASE=ks_datasets

DESCRIPTIONS_PORT=3013
DESCRIPTIONS_DB_DATABASE=ks_descriptions

DIAGRAMS_PORT=3014
DIAGRAMS_DB_DATABASE=ks_diagrams

DICTIONARIES_PORT=3015
DICTIONARIES_DB_DATABASE=ks_dictionaries

FILES_PORT=3016
FILES_DB_DATABASE=ks_files
FILES_STORAGE_FILE_SYSTEM_LOCATION=/data

GANTTS_PORT=3017
GANTTS_DB_DATABASE=ks_gantts

GATEWAY_PORT=5000
GATEWAY_CORS_ALLOW_LOCALHOST_ORIGIN=true
```

```
IMPORTS_EXPORTS_PORT=3018
IMPORTS_EXPORTS_DB_DATABASE=ks_imports_exports

INTEGRATOR_PORT=3019
INTEGRATOR_DB_DATABASE=ks_integrator

L10N_PORT=3020
L10N_DB_DATABASE=ks_l10n

MAIL_PROCESSOR_PORT=3021
MAIL_PROCESSOR_DB_DATABASE=ks_mail_processor
MAIL_PROCESSOR_SMTP_ADDRESS=smtp.domain.com:1025
MAIL_PROCESSOR_SMTP_LOGIN=admin@domain.com
MAIL_PROCESSOR_SMTP_PASSWORD=admin

MODELS_PORT=3022
MODELS_DB_DATABASE=ks_models

OBJECTS_PORT=3023
OBJECTS_DB_DATABASE=ks_objects

PROJECTS_PORT=3024
PROJECTS_DB_DATABASE=ks_projects

PUBLICATIONS_PORT=3025
PUBLICATIONS_DB_DATABASE=ks_publications

RESTRICTIONS_PORT=3026
RESTRICTIONS_DB_DATABASE=ks_restrictions

SYSTEM_EVENTS_JOURNAL_PORT=3027
SYSTEM_EVENTS_JOURNAL_DB_DATABASE=ks_system_events_journal

SYSTEM_SETTINGS_PORT=3028
SYSTEM_SETTINGS_DB_DATABASE=ks_system_settings
SYSTEM_SETTINGS_SETTINGS_AUTOMATICALLY_EXPAND_NOTIFICATIONS=true

USERS_PORT=3029
USERS_DB_DATABASE=ks_users

HEARTBEAT_PORT=3030
HEARTBEAT_DB_DATABASE=ks_heartbeat

ENTITIES_HIERARCHY_PORT=3031
ENTITIES_HIERARCHY_DB_DATABASE=ks_entities_hierarchy

WS_PORT=5001
WS_MAX_CONNECTIONS=1000
WS_PUBLISH_USERS_ONLINE_WORKER_DELAY=10s
```

1.8 Развертывание сервиса документации

Документация в платформе построена на CMS Strapi.

1. Для использования документации потребуется развернуть контейнер strapi согласно инструкции вендора https://docs.strapi.io/developer-docs/latest/setup-deployment-guides/installation/docker.html.
2. Открыть страницу strapi для создания учетной записи администратора.

3. В конструкторе типов контента создать тип коллекций Document с полями:
    3.1. doc – Rich text;
    3.2. title – Text.
4. В конструкторе типов контента создать тип коллекций Node с полями:
    4.1. name – Text;
    4.2. published – Boolean;
    4.3. documentId – Text;
    4.4. parentId – Text.
5. Для созданных типов коллекций нужно настроить права для роли Public (*Рисунок 1*):
    5.1. Для этого в левом меню выбрать пункт «Настройки».
    5.2. На втором уровне выбрать пункт «Роли И Доступы».
    5.3. Найти в списке роль «Public» и перейти к редактированию.
    5.4. В блоке «Доступы» отметить действия count, findone, find для DOCUMENT и NODE.

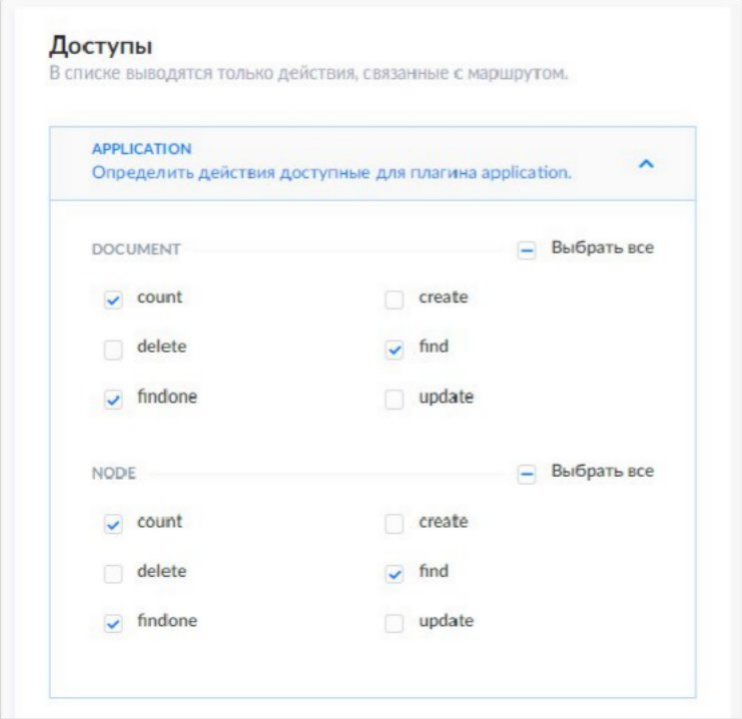


Рисунок 1. Настройка прав роли Public в Strapi

6. Выполнить вход в KS под учетной записью администратора.
7. В левом меню выбрать Настройки.
8. В блоке «Подключение к Документации/Strapi» заполнить значения «Хост» и «Порт» соответствующие Strapi.

1.9 Базовая конфигурация

Для первоначального развертывания может быть использована следующая конфигурация:

1. Сервер приложения:
   - vCPU – 24;
   - RAM –160 Gb;
   - HDD –150 GB;

   - ОС – Ubuntu 16.04 / 20.04, Debian 4+, RHEL / Centos 7+, Windows Server 2016 / 2019, Red Hat (UBI), Astra Linux;
   - Docker 20.10.12+;
   - Docker compose 1.29.0+.

2. Сервер БД:
   - vCPU – 16;
   - RAM – 64 Gb;
   - SSD – 500 Gb;
   - ОС – Ubuntu 16.04 / 20.04, Debian 4+, RHEL / Centos 7+, Windows Server 2016 / 2019, Red Hat (UBI), Astra Linux;
   - СУБД – PostgreSQL 13+.

По результатам опыта работы с конкретными решениями на платформе, при необходимости, можно скорректировать мощности.

2 Обновление версии

Шаги для обновления версии будут различными для разных вариантов развертывания. В этом руководстве рассмотрим последовательность действий для варианта развертывания docker и docker-compose.

Обновление версии в Docker Compose

1. Распаковать архив с образами контейнеров.
2. Загрузить образы контейнеров в Docker:

```
find ./images -type f -exec docker load -i {} \;
```

3. Остановить контейнеры KS:

```
docker-compose down
```

4. Запустить контейнеры KS:

```
docker-compose up -d
```

5. Обновить конфигурацию nginx в соответствующем контейнере:

```
docker exec nginx nginx -s reload
```

3 Интеграция в IT-ландшафт

3.1 Интеграции

Интеграция с внешними системами

Платформа KS имеет интеграции с базами данных. Поддерживаются такие базы данных, как:

- PostgreSQL;
- Microsoft SQL Server;
- MySQL.

Для настройки интеграции с базой данных нужно авторизоваться в ней. Для этого нужно на стороне внешней СУБД создать техническую учетную запись для подключений со стороны платформы KS.

Техническая учетная запись должна обладать правами достаточными для выполнения интеграционных процессов:

- Чтение внешнего источника – права на чтение всех требуемых таблиц и представлений;
- Запуск процедур для извлечения или добавления данных – права на запуск процедуры;
- Добавление записей – права на добавление записей.

Для настройки параметров интеграций техническая учетная запись должна обладать правамина чтение/выполнение следующих системных таблиц /процедур/команд:

PostgreSQL:

- information_schema.columns
- information_schema.constraint_column_usage
- information_schema.routines
- information_schema.parameters

MSSQL:

- information_schema.columns
- information_schema.constraint_column_usage
- information_schema.routines
- information_schema.parameters
- sys.sp_describe_first_result_set

MySQL:

- information_schema.columns
- SHOW PROCEDURE STATUS
- information_schema.parameters

Для создания подключения к внешней БД потребуется информация:

1. Хост СУБД;
2. Порт СУБД;
3. Название БД;
4. Логин технической учетной записи;

5. Пароль технической учетной записи.

Интеграции с внешними сервисами через брокер сообщений

Платформа KS имеет возможность отправлять сообщения во внешний сервис через брокер сообщений RabbitMQ. Через BPMS отправляется сообщение с параметрами. Настройки осуществляются в настройках конкретной задачи.

Логи/отчеты об ошибках интеграции

В случае возникновения ошибки после завершения интеграции в лог выводится информационный код ошибки согласно квалификации.

Пример:

```
{
 "message": "cannot connect to db"
 },
 {
  "message": "cannot establish connection with database",
  "code": 705,
  "data": {
   "database": "postgres"
  },
  "info": {
   "type": "Ошибка подключения к источнику"
  }
}
```

Список возможных ошибок:

Код ошибки	Классификация ошибок	Ошибка	Описание	Возможные варианты решения	Тип ошибки	Обращаться к
700	Во время интеграционного процесса	Ошибка в настройке операции	Ошибка связана с настроенными параметрами операции, которая вызывает некорректное или недоступное поведение операции (например, не указана операция-источник или не указана модель для операций, которые нуждаются в этом)	Требует проверки настроек операции	Ошибка настройки	Интегратору
701		Ошибка генерации временного хранилища данных	Ошибка связана с сохранением межоперационных данных во временное хранилище	Требует обращения в поддержку	Системная ошибка	Вендору
702		Ошибка удаления временного хранилища данных	Ошибка связана с удалением временного хранилища данных	Требует обращения в поддержку	Системная ошибка	Вендору

703		Ошибка получения межоперационных данных	Ошибка связана с получения межоперационных данных для дальнейшего использования	Требует обращения в поддержку	Системная ошибка	Вендору
704		Ошибка при работе с межоперационными данными	Ошибка связана с взаимодействием с уже полученными межоперационными данными, такими как конвертация, трансформация, и др.	Требует обращения в поддержку	Системная ошибка	Вендору
705		Ошибка подключения к источнику	Ошибка при взаимодействии с внешним источником	Требует проверки внешнего источника/данных внешнего источника	Ошибка источника	Источнику данных
706		Ошибка при работе с источником	Ошибка при взаимодействии с внешним источником	Требует проверки внешнего источника/данных внешнего источника	Ошибка источника	Источнику данных
707		Ошибка в логике операции	Ошибка в ходе исполнения действия операции, приводящая к некорректной работе задачи	Требует обращения в поддержку	Системная ошибка	Вендору
710	Во время подготовки запуска интеграции	Циклическая зависимость последовательности операций	Обнаружена циклическая зависимость операций в ходе построения графа операций	Требует проверки предыдущих действий операций	Ошибка настройки	Интегратору
711		Ошибка при инициализации значений по умолчанию переменной	Ошибка при заполнении значений переменных значениями по умолчанию	Требует обращения в поддержку/ Проверка настроек значений по умолчанию переменных	Системная ошибка / Ошибка настройки	Вендору / Интегратору
712		Ошибка получения пользователя, запустившего процесс интеграции	Ошибка связанна с проблемой при получении пользователя, запустившего интеграцию	Требует обращения в поддержку	Системная ошибка	Вендору
720	Во время настройки конструктора	Циклическая зависимость последовательности	Обнаружена циклическая зависимость операций в ходе построения графа операций	Требует проверки предыдущих действий	Ошибка настройки	Интегратору

		операций		операций		
730	Ошибка базы данных системы	Ошибка получения данных	Ошибка при получении данных из хранилища платформы	Требует обращения в поддержку	Системная ошибка	Вендору
731		Ошибка создания данных	Ошибка при создании данных в хранилище платформы	Требует обращения в поддержку	Системная ошибка	Вендору
732		Ошибка обновления данных	Ошибка при обновлении данных в хранилища платформы	Требует обращения в поддержку	Системная ошибка	Вендору
733		Ошибка удаления данных	Ошибка при удалении данных из хранилища платформы	Требует обращения в поддержку	Системная ошибка	Вендору
740	Ошибка системного события	Ошибка отправки системного события	Ошибка при взаимодействии с событиями платформы, связанная с отправкой такого события	Требует обращения в поддержку	Системная ошибка	Вендору
750	Ошибка в логике сервиса	Ошибка в логике сервиса	Описывает ошибку, не предусмотренную выше описанными	Требует обращения в поддержку	Системная ошибка	Вендору
770		Невозможно заблокировать записи в таблице БД	Невозможно заблокировать записи в таблице БД	Требует обращения в поддержку	Системная ошибка	Вендору
705	Ошибка источника (*): Источник АПИ Источник БД Источник Шина данных Источник публикация АПИ	Ошибка подключения	Невозможность подключиться к источнику	Требует проверки учетных данных/сертификатов, доступности источника	Ошибка настройки / Ошибка источника	Интегратору / Источнику данных
706		Ошибка при работе с источником	Некорректное поведение источника при взаимодействии с ним	Требует проверки прав учетной записи, корректности обращения к источнику	Ошибка настройки / Ошибка источника	Интегратору / Источнику данных
7	Ролевая модель	Ошибка доступа	Ошибка при попытке взаимодействия с действиями требуемых доступов на платформе	Проверка политик доступов	Ошибка доступа	Администратору

54	Ошибки работы с сущностями	Ошибка создания сущности	Ошибка при создании сущности (например, объекта)	Требует проверки настройки подключения к источнику, настроек интеграции, при необходимости обращения в тех.поддержку	Ошибка настройки / Системная ошибка	Интегратору / Вендору
53		Ошибка удаления сущности	Ошибка при удалении сущности (например, объекта)	Требует проверки настройки подключения к источнику, настроек интеграции, при необходимости обращения в тех.поддержку	Ошибка настройки / Системная ошибка	Интегратору / Вендору
3/707		Ошибка обновления сущности	Ошибка при обновлении сущности (например, имени объекта)	Требует проверки настройки подключения к источнику, настроек интеграции, при необходимости обращения в тех.поддержку	Ошибка настройки / Системная ошибка	Интегратору / Вендору
3/707	Ошибки при работе с переменными	Ошибка изменения переменной	Ошибка при попытке изменения значения переменной интеграции	Требует проверки соответствия типа переменной, записываемому значению, наличия переменной как таковой, при необходимо	Ошибка настройки / Системная ошибка	Интегратору / Вендору

				сти обращения в тех.поддержку		
3/707	Ошибки при работе с операциями трансформации/ сериализации/нормализации	Ошибка выполнения трансформации	Ошибка выполнения операции трансформации	Требует проверки корректности заполнения параметров действия, при необходимости обращения в тех.поддержку	Ошибка настройки / Системная ошибка	Интегратору / Вендору
3/707		Ошибка выполнения сериализации	Ошибка выполнения операции сериализации	Требует проверки корректности заполнения параметров действия, при необходимости обращения в тех.поддержку	Ошибка настройки / Системная ошибка	Интегратору / Вендору
3/201		Ошибка выполнения нормализации	Ошибка выполнения операции нормализации	Требует проверки корректности заполнения параметров действия, при необходимости обращения в тех.поддержку	Ошибка настройки / Системная ошибка	Интегратору / Вендору

706	Ошибки запуска процедур	Ошибки запуска процедур	Ошибки запуска процедур интеграции	Требует проверки наличия процедур и корректности заполнения их параметров, при необходимости обращения в тех.поддержку	Ошибка источника / Ошибка настройки / Системная ошибка	Администратору / Интегратору / Вендору
706	Ошибки отправки сообщений в шину	Ошибки отправки сообщений в шину данных	Ошибки отправки сообщений в шину данных	Требует проверки наличия шины данных, корректности заполнения параметров, при необходимости обращения в тех.поддержку	Ошибка источника / Ошибка настройки / Системная ошибка	Администратору / Интегратору / Вендору

3.2 Электронная почта

Платформа KS подключается к почтовым серверам с помощью протокола SMTP. Параметры подключения задаются через переменные окружения микросервиса mail_processor.

Значения переменных окружения, которые нужно настроить, представлены в строках 31 – 34 таблицы 1. После этого платформа сможет отправлять сообщения через почтовый сервис.

3.3 Логирование сообщений информационной безопасности

Логирование перечня событий происходит во внутренних журналах платформы KS.

Перечень отслеживаемых событий

В системе отслеживаются следующие типы событий, указанные в таблице 2.

Таблица 2. Типы событий системы

№	Код события	Описание события
1	service_started	Начало работы сервиса
2	service_stopped	Завершение работы сервиса
3	logged_in	Вход пользователя в систему (успешный)
4	login_failed	Неуспешная попытка входа в систему
5	logged_off	Выход пользователя из системы

6	went_online	Подключение пользователя к чатам в системе(опционально)
7	went_offline	Отключение пользователя от чатов в системе
8	created (совместно с типомсущности)	Создание сущности
9	updated (совместно с типомсущности)	Обновление свойств сущности
10	deleted (совместно с типомсущности)	Удаление сущности
11	blocked	Блокировка учётной записи
12	unblocked	Разблокировка учётной записи
13	auth_blocked	Блокировка учётной записи / ip-адреса
14	password_changed	Смена пароля учётной записи
15	password_updated	Смена пароля учётной записи администратором системы
16	password_reset	Сброс пароля пользователем
17	role_set_policies	Назначение роли нового доступа
18	role_unset_policies	Отключение существующего доступа роли
19	role_set	Назначение роли учётной записи
20	role_unset	Отключение существующей роли учётной записи
21	created (тип объекта project-access)	Предоставление доступа для пользователя к проекту
22	created (тип объекта project-role-access)	Предоставление доступа для роли к проекту
23	deleted (тип объекта project-access)	Отключение доступа для пользователя к проекту
24	deleted (тип объекта project-role-access)	Отключение доступа для роли к проекту

Спецификация информации о событии

События в системе содержат следующий перечень атрибутов, указанный в таблице 3.

Таблица 3. Перечень атрибутов системы

№	Название	Описание
system_events		
1	uuid	Идентификатор события
2	time	Время события
3	reference	Тип сущности в системе (если применимо, см. «Типы связанных с событиями сущностей»)
4	reference_uuid	Идентификатор сущности в системе (если применимо)
5	parent_reference	Тип родительской сущности в системе (если применимо, см. «Типы связанных с событиями сущностей»)
6	parent_reference_uuid	Идентификатор родительской сущности в системе (если применимо)
7	action	Тип события / действия
8	actor_user_uuid	Пользователь, инициировавший событие, осуществивший действие
9	owner_user_uuid	Пользователь, являющийся владельцем сущности
10	comment	Дополнительная информация о событии в текстовом виде
11	is_cs_event	Признак события информационной безопасности
extended_data		
1	uuid	Идентификатор записи
2	event_uuid	Идентификатор события
3	event_name	Название события
4	event_success	Признак успешности события
5	event_type	Тип события

6	event_object_name	Название сервиса-источника события
7	journal_name	Название журнала (опционально, константа)
8	author_ip	IP-адрес пользователя, инициировавшего событие
9	author_login	Логин пользователя, инициировавшего событие
10	author_domain	Домен пользователя, инициировавшего событие
11	source_service_ip	IP-адрес сервиса-источника события
12	source_service_mac	MAC-адрес сервиса-источника события (на данный момент не используется)
13	source_service_name	Название экземпляра сервиса-источника события
14	source_service_time_utc	Время экземпляра сервиса-источника события
15	destination_service_hostname	Имя хоста экземпляра сервиса-получателя события
16	destination_service_bd	Название базы данных сервиса-получателя события
17	destination_service_time_utc	Время экземпляра сервиса-получателя события
18	message	Дополнительная информация о событии в текстовом виде
19	changed_values	Измененные значения (для сущностей, значения которых изменяются, например отдельные настройки ИБ)
20	severety_level	Уровень критичности события
21	created_at	Время создания события в БД
22	source_service_version	Версия сервиса-источника события

Типы связанных с системными событиями бизнес-сущностей

Типы сущностей, связанные с системными событиями, указаны в таблице 4.

Таблица 4. Перечень атрибутов системы

№	Тип сущности	Описание
0	Undefined	Не определено
1	Events	События
2	Class	Классы
3	Methods	Формулы
4	Indicators	Показатели
5	Measurements	Измерения
6	ClassesRelations	Связи классов
7	BehaviourRules	Поведенческие правила
8	Interfaces	Интерфейсы данных
9	Objects	Объекты
10	Models	Модели
11	Datasets	Наборы данных
12	Diagrams	Диаграммы
13	Shapes	Фигуры диаграмм
14	Configurations	Конфигурации
15	DataLinks	Ссылки на данные
16	Users	Пользователи
17	Formulas	Формулы
18	Documents	Документы
19	Files	Файлы
20	ObjectsRelations	Связи объектов
21	Tags	Теги
22	TagsRelations	Связи тегов
23	Profiles	Профили пользователей
24	DataTable	Таблицы данных (представления)
25	FormulaElements	Элементы формул
26	Dictionaries	Справочники
27	DictionaryElements	Элементы справочников
28	Chats	Чаты
29	Messages	Сообщения
30	Subscribes	Подписки
31	–	– не используется
32	ProcessCards	Карточки процессов
33	Templates	Шаблоны импорта/экспорта
34	Imports	Экземпляры импорта данных из MS Excel
35	Gantts	Диаграммы Ганта
36	Approval	Экземпляры согласований
37	ApprovalVote	Голоса в рамках согласований
38	DataSlice	Срезы данных
39	Meetings	Собрания
40	Calls	Звонки
41	Dashboards	Дэшборды
42	DashboardCell	Ячейки дэшбордов
43	DashboardEvent	События дэшбордов
44	ResourceRestriction	Ресурсные ограничения
45	MatrixRestriction	Матричные ограничения
46	Project	Проекты
47	Processes	Процессы

48	ClassNode	Элементы дерева классов
49	Exports	Экземпляры экспорта данных в MS Excel
50	Action	Действия (в составе описания политик доступа)
51	ApprovalWorks	Пары работ, помещённые на согласование в соответствиис матричным ограничением
52	SystemSettings	Системные настройки
53	Roles	Роли
54	Integration	Правила интеграции
55	IntegrationSource	Источники данных интеграции
56	ProcessEvents	События в рамках бизнес-процессов
57	Tasks	Задачи в рамках бизнес-процессов
58	Gates	Шлюзы в рамках бизнес-процессов
59	Publication	Публикации
60	Backups	Бэкапы моделей
61	Calendars	Календари
62	DescriptionField	Поля описания сущностей
63	FieldsSettings	Настройки полей описания сущностей
64	Expertises	Чаты экспертизы сущностей
65	–	– не используется
66	DictionaryNode	Элементы дерева справочников
67	Policies	Политики доступа
68	CyberSecuritySettings	Настройки информационной безопасности
69	SystemJournalEvents	События данного журнала
70	DiagramAutosaveConfigs	Настройки автосохранения диаграмм
71	SystemEvents	Системные события
72	IntegrationFieldRule	Правила обработки полей в рамках интеграции
73	IntegrationSamplingCondition	Ограничения на выборку данных в рамках интеграции
74	IntegrationProcedure	Экземпляры процедур интеграции
75	–	– не используется
76	Folder	Папки

Также есть возможно настроить отправку данных сообщений в Syslog.

Syslog – стандарт отправки и регистрации сообщений о происходящих в системе событиях, использующийся в компьютерных сетях, работающих по протоколу IP.

Настройка осуществляется указанием переменной окружения.

3.4 Установка Apache Superset

Apache Superset – это веб-приложение для бизнес-аналитики, позволяющее визуализировать большие объемы данных.

Общая документация по настройке Apache Superset расположена на официальном сайте https://superset.apache.org/docs/intro/

3.4.1 Минимальные системные требования

5. Процессор: 2 ядра
6. Оперативная память (RAM): 8 ГБ
7. Жесткий диск: 2 ГБ
8. Операционная система:

- Ubuntu 16.04 / 20.04
- Debian 4+
- RHEL / Centos 7+

- Red Hat (UBI)
- Astra Linux

В настоящий момент установленный Apache Superset занимает на диске 285 МБ.

3.4.2 Развертывание Apache Superset

Наиболее простое развертывание Apache Superset происходит при помощи **Docker**. Контейнеры, необходимые для развёртывания, можно найти на **Docker Hub** https://hub.docker.com/r/apache/superset/

1. Перед развёртыванием Apache Superset необходимо установить и настроить **Docker Engine**.
2. Скачать последний релиз Apache Superset можно с **GitHub** https://github.com/apache/superset/releases
3. В приведённом ниже описании развертывания используется последняя стабильная **версия Apache Superset 3.1.1**.

Далее приведены шаги по развёртыванию Apache Superset согласно инструкции с официального сайта:

1. Экспортируем версию Apache Superset в переменные окружения:

```
export SUPERSET_VERSION=3.1.1
```

2. Скачиваем докер образ:

```
docker pull apache/superset:$SUPERSET_VERSION
```

3. Генерируем ключ шифрования, при помощи которого, приложение зашифрует свою базу данных:

```
>> openssl rand -base64 42
== nW3NuniZ6QqZH5ipDB+B+2r/613lQ26qA+3g/TVqTeBP3q0408k8B5Fs
```

4. Запускаем докер образ:

```
docker run -d -p 8080:8088 \
-e "SUPERSET_SECRET_KEY=nW3NuniZ6QqZH5ipDB+B+2r/613lQ26qA+3g/TVqTeBP3q0408k8B5Fs" \
-e "TALISMAN_ENABLED=False" \
-v /data/superset:/app \
--name superset \
apache/superset:$SUPERSET_VERSION

1. SUPERSET_SECRET_KEY – был сгенерирован на предыдущем шаге. Чтобы не использовать его при запуске образа, можно прописать непосредственно в конфигурации Apache Superset в файле с константами
2. Чтобы прописать ключ в константы, необходимо смонтировать volume (в предыдущей команде ключ -v)
```

9. Создаем локального администратора:

```
docker exec -it superset superset fab create-admin \
--username admin \
--firstname Admin \
--lastname Admin \
--email admin@localhost \
--password admin
```

10.Инициализируем базу данных, загружаем примеры:

```
docker exec -it superset superset db upgrade && docker exec -it superset superset load_examples && docker exec -it superset superset init
```

Для настройки пользовательского входа в Apache Superset по ссылке (не вводя логин и пароль) в файле конфигурации указывается **переменная окружения**:

```
"PUBLIC_ROLE_LIKE = Gamma_public"
```

3.4.3 Дополнительные настройки

Русификация

Для настройки русификации, в конфигурационном файле Apache Superset, в строке BABEL_DEFAULT_LOCALE, заменить значение по умолчанию с "en" на "ru":

```
Setup default language
BABEL_DEFAULT_LOCALE = "ru"
```

Доступы к базам данных

Также можно использовать любую базу данных, которую поддерживает Apache Superset. Перечень БД представлен на официальном сайте https://superset.apache.org/docs/databases/installing-database-drivers/

По умолчанию Apache Superset использует SQLite.

При выборе другой БД нажмите на ее название в таблице – откроются подробные настройки подключения данной БД (*Рисунок 2*).

Database	PyPI package	Connection String
Amazon Athena	`pip install pyathena[pandas]`, `pip install PyAthenaJDBC`	`awsathena+rest://{aws_access_key_id}:{aws_secret_access_key}@athena.{region_name}.amazonaws.com/{schema_name}?s3_staging_dir={s3_staging`
Apache Doris	`pip install pydoris`	`doris://<User>:<Password>@<Host>:<Port>/<Catalog>.<Database>`
Amazon DynamoDB	`pip install pydynamodb`	`dynamodb://{access_key_id}:{secret_access_key}@dynamodb.{region_name}?connector=superset`
Amazon Redshift	`pip install sqlalchemy-redshift`	`redshift+psycopg2://<userName>:<DBPassword>@<AWS End Point>:5439/<Da`
Apache Drill	`pip install sqlalchemy-drill`	`drill+sadrill:// For JDBC drill+jdbc://`
Apache Druid	`pip install pydruid`	`druid://<User>:<password>@<Host>:<Port-default-9088>/druid/v2/sql`
Apache Hive	`pip install pyhive`	`hive://hive@{hostname}:{port}/{database}`

Рисунок 2. Базы данных, поддерживаемые Apache Superset

4 Поддержание работоспособности

При наличии ошибок в системе необходимо посмотреть логи сервиса, отвечающего за функцию, в которой возникла ошибка.

Категории ошибок в системе

Категории ошибок в системе делятся на 2 типа:

1. Ошибки **frontend**, возникающие на клиентском рабочем месте;
2. Ошибки **backend**, возникающие на сервере.

Frontend – ошибки отслеживаются преимущественно через консоль web-браузера.

Backend – ошибки отслеживаются через всплывающие сообщения в интерфейсах платформы или через консоль web-браузера в ответах на запросы к backend. В ответе на запрос часто фигурирует текст ошибки. Также ошибку можно отслеживать в логах этого микросервиса.

Просмотр ошибки на сервере в потоке вывода контейнера:

```
docker logs –n <кол-во сообщений> –f <контейнер>
```

Система Graylog

Платформа KS позволяет передавать логи в агрегатор логов – Graylog.

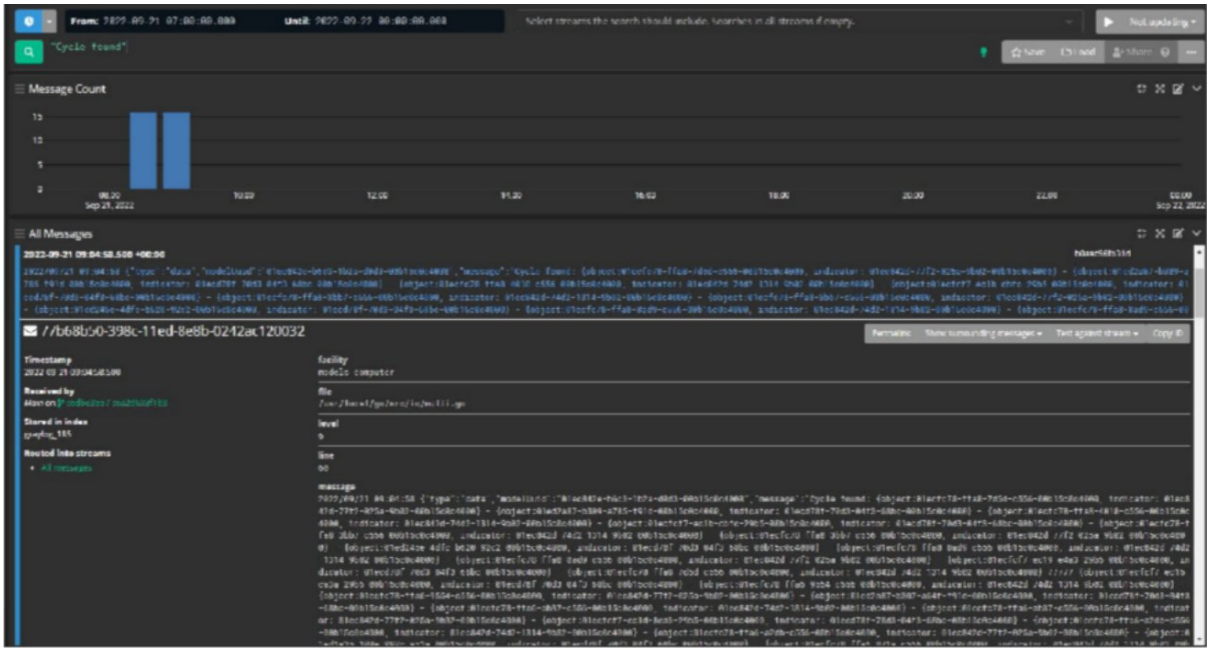
Для микросервиса должны быть настроены значения переменных окружения LOGS_HOST, LOGS_SEND, LOGS_SEND_LEVEL.

Просмотр и поиск сообщений осуществляется пользователем через интерфейс graylog.

Подробное описание инструментов поиска в документации graylog https://docs.graylog.org/v1/docs/queries.

На рисунке 3 пример поиска в логах сообщений о циклических зависимостях в расчетной модели.

Рисунок 3. Пример поиска сообщений в логах


"Cycle found"
Message Count
All Messages
/b68b50-398c-11ed-8e8b-0242ac120032
Timestamp
facility
Received by
file
Stored in index
level
Routed into streams
line
message

5 Резервное копирование и восстановление

5.1 Микросервисы KS

Микросервис KS – это 1 исполняемый файл. При повреждении контейнера или исполняемого файла внутри контейнера его можно восстановить из образа контейнера.

При возникновении ошибки:

1. Остановить контейнер:

```
docker-compose stop <контейнер>
```

2. Удалить контейнер:

```
docker-compose rm <контейнер>
```

3. Создать новый контейнер из образа:

```
docker-compose up -d <контейнер>
```

5.2 Базы данных платформы

Платформа KS состоит из собственных и сторонних микросервисов. Каждому микросервису платформы соответствует своя БД, подключение к которой настраивается через переменные окружения. Микросервисы платформы и тип СУБД представлены в таблице 5.

Таблица 5. Собственные микросервисы

Микросервис	СУБД
Aggregator	БД отсутствует
Approval	PostgreSQL
Auth	PostgreSQL
Backups	PostgreSQL
Batcher	PostgreSQL
BPMS	PostgreSQL
Calendars	PostgreSQL
Chats	PostgreSQL
Classes	PostgreSQL
Dashboards	PostgreSQL
Data_tables	PostgreSQL
Datasets	PostgreSQL
Descriptions	PostgreSQL
Diagrams	PostgreSQL
Dictionaries	PostgreSQL
Files	PostgreSQL
Gantts	PostgreSQL
Gateway	БД отсутствует
Imports_exports	PostgreSQL
Integrator	PostgreSQL
Mail_processor	PostgreSQL
Models	PostgreSQL
Models_computer	PostgreSQL
Objects	PostgreSQL

Projects	PostgreSQL
ublications	PostgreSQL
Restrictions	PostgreSQL
System_events_journal	PostgreSQL
System_settings	PostgreSQL
Users	PostgreSQL
WS	БД отсутствует

Резервное копирование и восстановление баз данных платформы настраивается штатными средствами СУБД или сторонними утилитами. Параметры бэкапирования для каждого внедрения определяются в индивидуальном порядке исходя из допустимого времени потери данных.

Пример плана бэкапирования:

Если потеря данных допустима в пределах 1 дня, то бэкапирование осуществляется 1 раз в сутки. Для экономии места рекомендуется выполнять полное бэкапирование 1 раз в месяц/1 раз в неделю, при этом выполнять инкрементальное бэкапирование по изменениям к полному бэкапированию 1 раз в сутки.

Бэкапирование баз данных PostgreSQL можно осуществлять с помощью стандартных средств pg_dump / pg_restore. Инструкция по стандартным средствам представлена в документации PostgreSQL https://www.postgresql.org/docs/current/app-pgdump.html.

6 Управление пользователями и ролями

В системе реализованы следующие варианты управления пользователями и ролями:

- Управление через пользовательский интерфейс системы;
- Управление при помощи запросов к API;
- Управление при помощи Active Directory;
- Keycloak.

6.1 Управление пользователями и ролями через пользовательский интерфейс системы

После прохождения аутентификации в навигационном меню доступны пункты Пользователи
и Роли.

В меню Пользователи представлен список пользователей системы с возможностью просмотра данных по выбранному пользователю, создания, блокирования или удаления учётной записи пользователя.

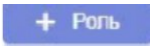
В меню Роли представлен список ролей в системе с возможностью просмотра данных по выбранной роли, создания или удаления роли.

6.1.1 Создание пользователя

1. Для создания пользователя необходимо открыть страницу Пользователи и нажать кнопку 

.
2. Откроется страница создания нового пользователя.
3. При создании нового пользователя необходимо вариант создания профиля пользователя:

- Вручную – данные пользователя (ФИО, логин, e-mail, пароль) можно внести самостоятельно. Пароль пользователя может быть введён вручную, сгенерирован для копирования или отправлен пользователю автоматически с использованием указанного e-mail адреса;
- Получить из AD – при настроенном подключении к Active Directory пользователь сможет проходить аутентификацию используя данные доменной учётной записи. В данном случае необходимо ввести только логин пользователя в Active Directory.

4. Пользователю могут быть сопоставлены календари для определения его рабочего времени и использования в рамках системы управления бизнес-процессами.
5. Если требуется предоставить роли при создании нового пользователя, необходимо нажать кнопку 

и выбрать одну или несколько ролей.

6.1.2 Блокировка/разблокировка пользователя

Блокировка

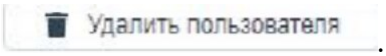
1. Для блокировки учетной записи пользователя необходимо выбрать пользователя в списке.
2. На странице профиля пользователя необходимо нажать кнопку

.

Разблокировка

1. Для разблокировки учетной записи пользователя необходимо выбрать заблокированного пользователя в списке
2. На странице профиля пользователя необходимо нажать кнопку


6.1.3 Удаление пользователя

1. Для удаления учетной записи пользователя необходимо выбрать пользователя в списке.
2. На странице профиля пользователя необходимо нажать кнопку


.

6.1.4 Изменение ролей пользователя

1. Для изменения ролей пользователя необходимо выбрать пользователя в списке.
2. На странице профиля пользователя необходимо выбрать закладку Роли.
3. Откроется список ролей пользователя.
4. Если требуется добавить роль, то необходимо нажать кнопку 

и выбрать списокролей, которые нужно добавить.
5. Если требуется удалить роль, то необходимо нажать кнопку  напротив названия роли в списке ролей пользователя.

6.1.5 Создание роли

1. Для создания роли необходимо открыть страницу Роли и нажать кнопку 

.
2. Откроется страница настройки новой роли.
3. Необходимо ввести данные роли:
   - Название;
   - Роль в AD – если используется подключение к Active Directory, то данная роль будет автоматически сопоставляться всем пользователям, которые проходят доменную аутентификацию и являются участниками выбранной группы;
   - Описание роли;
   - Настройки доступа для роли – используются 3 режима доступа:
     - Выбранные пункты разрешены – необходимо отметить, к каким функциям системы будет иметь доступ данная роль;
     - Выбранные пункты запрещены – необходимо отметить, какие функции системы будут заблокированы для данной роли;
     - Разрешено всё – автоматическое предоставление доступов ко всем существующим и будущим функциям системы.

6.2 Управление пользователями при помощи запросов к API

Для управления пользователями при помощи сторонних инструментов доступен API, которыйпозволяет:

- Создать пользователя;
- Обновить информацию о пользователе;

- Заблокировать / разблокировать пользователя;
- Назначить / отозвать роль пользователя;
- Удалить пользователя.

6.2.1 Аутентификация в системе

Для подключения к API необходимо получить токен аутентификации:

1. **API**

```
настроенный для системы домен/api/auth/login
```

2. **Тело запроса**

```
{"login":"технический логин","password":"пароль"}
```

3. **Тело ответа**

```
{"token":"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJyb2xlcyI6WyLQt9Cw0YDQtdCz0LjRgdGC0YDQuNGA0L7QstC
w0L3QvdGL0LkiLCLQsNC00LzQuNC90LjRgdGC0YDQsNGC0L7RgCJdLCJkb21haW4iOiIiLCJsb2dpbiI8ImFkbWluIiwi
dG1wX3Rva2VuIjpmYWxzZSwiZXhwIjoxNjE5NjgxMjYwLCJqdGkiOiIwMWViMGVlZi0zY2UwLTFlODAtYThiOS0wM
DAwMDAwMDAwMDAifQ.zE QeDxOyKWhGujyhgpI0kYAFyrx0yzjfMuqnLPJD_js",
"user":{"uuid":"01eb0eef-3ce0-1e80-a8b9-000000000000",
"profileUuid":"01eb0eef-3cf2-035e-a8b9-000000000000",
...}
}
```

Токен аутентификации помещается в заголовок последующих запросов:

```
authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJyb2xlcyI6WyLQt9Cw0YDQtdCz0LjRgd
GC0YDQuNGA0L7QstCw0L3QvdGL0LkiLCLQsNC00LzQuNC90LjRgdGC0YDQsNGC0L7RgCJdLCJkb21
haW4iOiIiLCJsb2dpbiI8ImFkbWluIiwidG1wX3Rva2VuIjpmYWxzZSwiZXhwIjoxNjE5NjgxMjYwLCJqdGkiOi
IwMWViMGVlZi0zY2UwLTFlODAtYThiOS0wMDAwMDAwMDAwMDAifQ.zEQeDxOyKWhG ujyhgpI0kY
AFyrx0yzjfMuqnLPJD_js
```

6.2.2 Создание пользователя

1. **API**

```
настроенный для системы домен/api/users/create
```

2. **Тело запроса**

```
{
  "email": "user@domain.com",
  "password": "password",
  "firstname": "firstname",
  "lastname": "lastname",
  "login": "user"
}
```

3. **Тело ответа**

```
{
 "uuid": "уникальный идентификатор учётной записи пользователя", //далее «UUID пользователя»
 "profileUuid": "уникальный идентификатор информации о пользователе"
 "error": {}
}
```

6.2.3 Обновление информации о пользователе

Для обновления информации о пользователе передаются следующие параметры:

1. **API**

```
настроенный для системы домен/api/users/update
```

2. **Тело запроса**

```
{
  "uuid": "UUID пользователя",
  "email": "some@email.com", //обновляемое поле
  "login": "admin", //обновляемое поле
}
```

3. **Тело ответа**

```
{
  "error": {}
}
```

6.2.4 Блокировка/разблокировка пользователя

Для блокировки / разблокировки пользователя передаются следующие параметры:

1. **API**

```
настроенный для системы домен/api/users/block
настроенный для системы домен/api/users/unblock
```

2. **Тело запроса**

```
{
"uuid": "UUID пользователя",
}
```

3. **Тело ответа**

```
{
  "error": {}
}
```

6.2.5 Назначение/отзыв роли пользователя

Для получение идентификатора роли в системе с целью последующего назначения пользователю передаются следующие параметры:

1. **API**

```
настроенный для системы домен/api/access-control/get-roles
```

2. **Тело запроса**

```
{
term": "", //поиск ролей по тексту названия
"limit": 0 //ограничение на количество возвращаемых ролей
}
```

3. **Тело ответа**

```
{
  "data": [
    {
      "uuid": "UUID роли",
      "name": "Название роли",
      "description": "Описание роли",
      "createdAt": "Дата создания роли",
      "settings": {
        "someKey": "someValue" // произвольный набор параметров роли
      },
      "updatedAt": "Дата обновления роли"
    },
    ...
  ]
}
```

Для назначения / отзыва роли передаются следующие параметры:

1. **API**

```
настроенный для системы домен/api/access-control/set-role
настроенный для системы домен/api/access-control/unset-role
```

2. **Тело запроса**

```
{
"userUuid": "UUID пользователя",
"roleUuid": "UUID роли",
}
```

3. **Тело ответа**

```
{
  "error": {}
}
```

6.2.6 Удаление пользователя

Для блокировки / разблокировки пользователя передаются следующие параметры:

1. **API**

```
настроенный для системы домен/api/users/delete
```

2. **Тело запроса**

```
{
"uuid": "UUID пользователя",
}
```

3. **Тело ответа**

```
{
 "error": {}
}
```

6.3 Управление пользователями при помощи групп Active Directory

В рамках интеграции с Active Directory возможно:

- Автоматически добавлять пользователей в систему;
- Управлять ролями пользователей;
- Отключать пользователей от системы.

Настройки Active Directory

Существует несколько параметров для настройки интеграции с Active Directory.

Глобальные параметры ActiveDirectory – доступны в меню «Настройки», закладка «Информационная безопасность» (*Рисунок 4*).

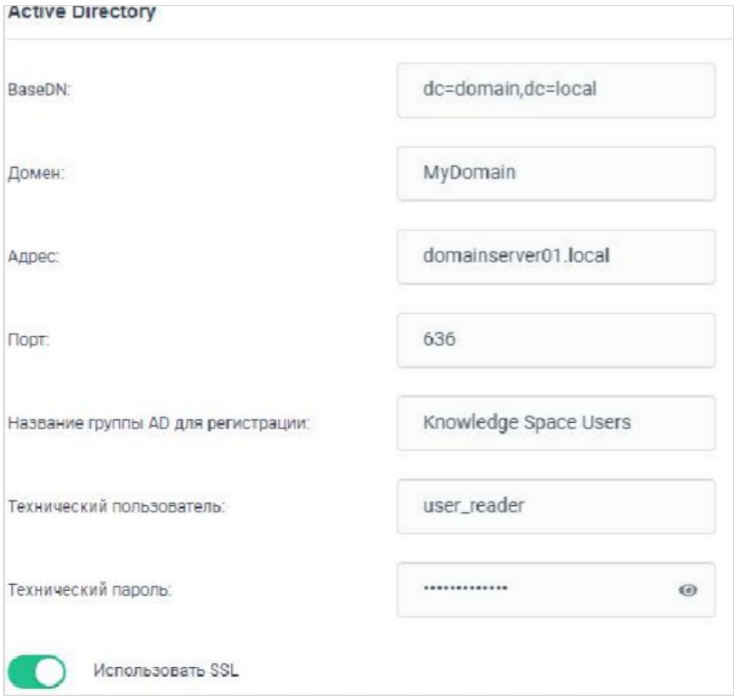


Рисунок 4. Настройка Active Directory

- **BaseDN** – идентификационные параметры доменного пространства имен;
- **Домен** – название домена для проверки учётных записей;
- **Адрес** – адрес сервера доменной аутентификации;
- **Порт** – порт для подключения к серверу доменной аутентификации;

- **Название группы AD для регистрации** – название группы Active Directory, добавление в которую будет означать немедленное получение доступа к системе;
- **Технический пользователь/Технический пароль** – данные для подключения технической учётной записи для последующего сопоставления ролей группами Active Directory.

Сопоставление роли в системе с группами Active Directory – механизм, позволяющий автоматический присваивать роль пользователю на основании его вхождения в группу Active Directory. Для настройки необходимо:

1. Открыть меню Роли и выбрать роль.
2. Для роли заполнить параметр Роль в AD.

При аутентификации пользователя система подключается к Active Directory и синхронизирует его роли с теми группами, которые сопоставлены ему в Active Directory и имеют сопоставление с ролями в системе.

6.3.1 Инструкция по настройке мэппинга ролей KS с ролями ActiveDirectory через БД

1. С использованием pgadmin подключиться к серверу БД KS.

2. Открыть БД KS_system_settings, в ней открыть таблицу cs_settings.

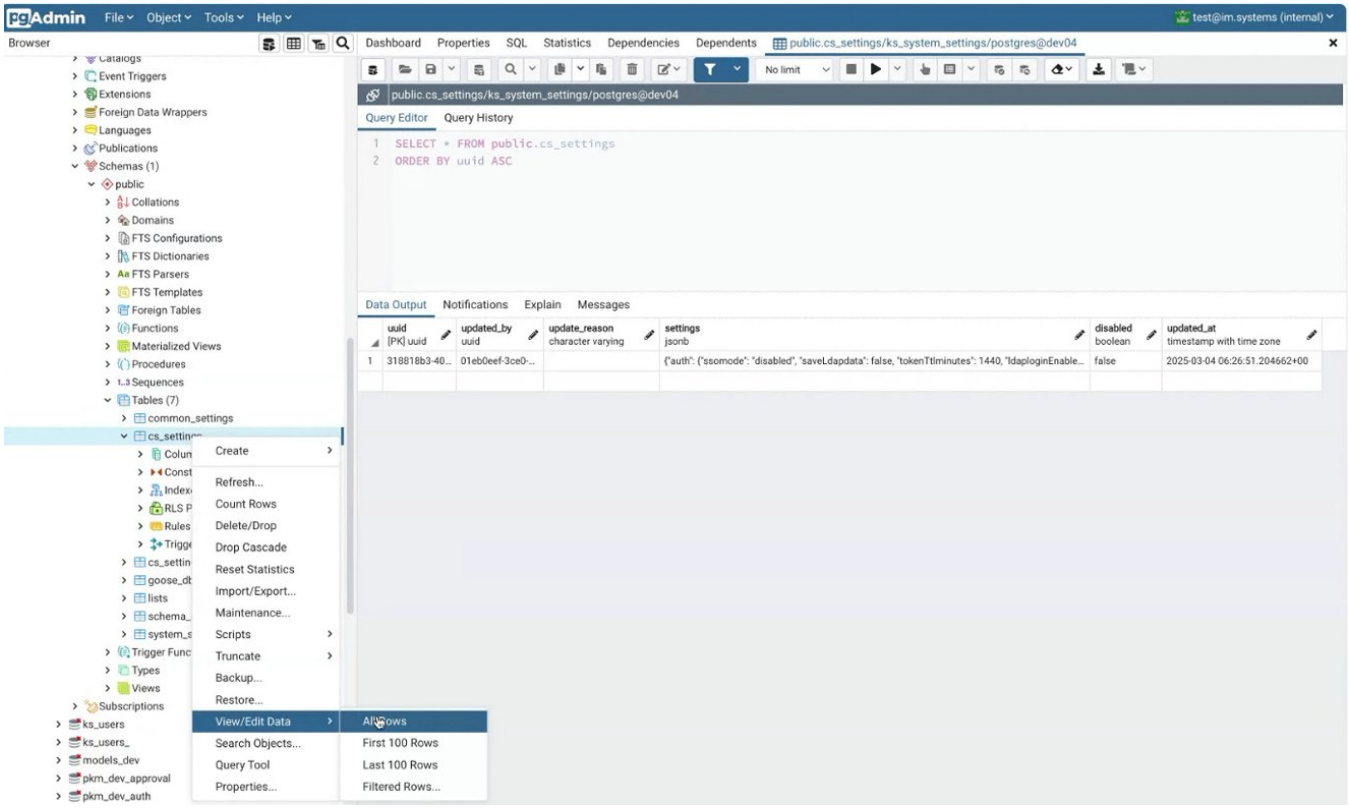


Рисунок 1. Таблица cs_settings в БД KS

3. В таблице cs_settings по умолчанию создана одна запись с полем settings. Открыть поле settings для редактирования.
4. Поле settings содержит набор настроек в формате Json. Необходимо найти раздел LdaprolesMapping. Если данный раздел отсутствует, то его необходимо создать.

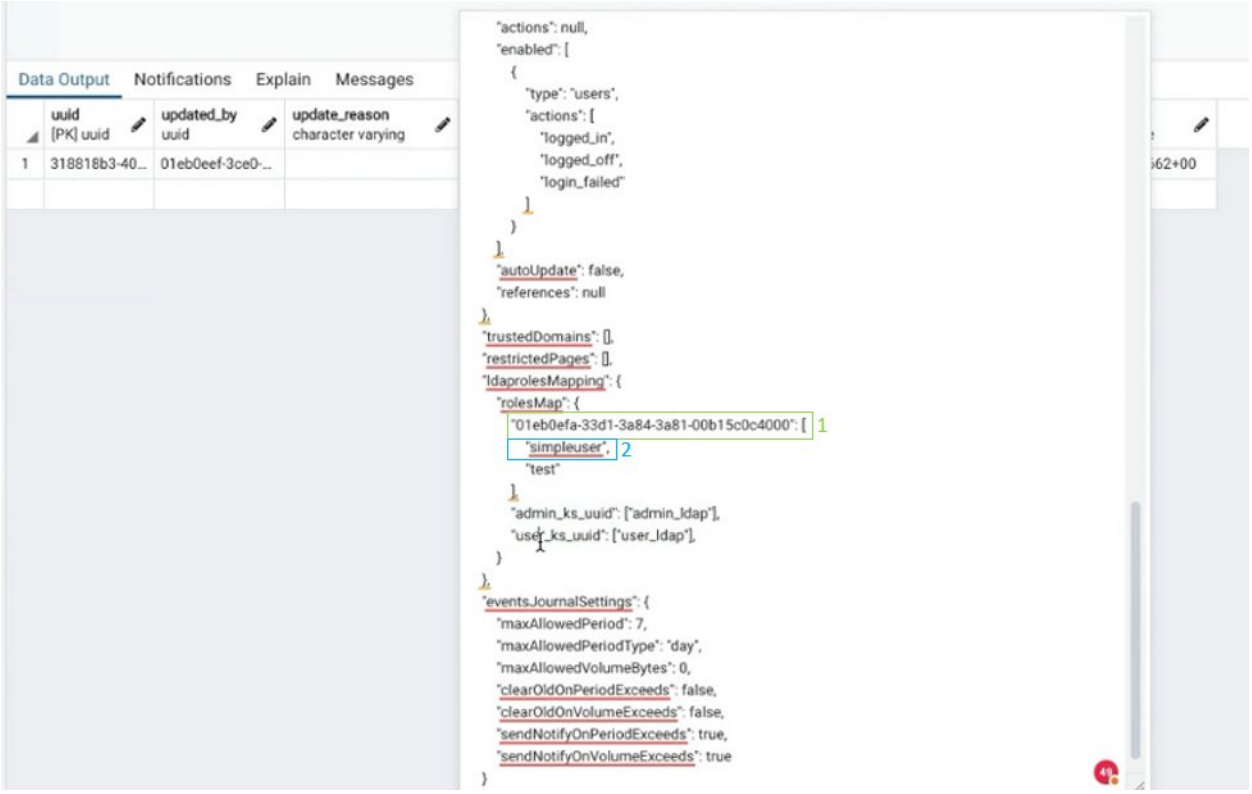


Рисунок 2. Раздел настроек LdaprolesMapping

5. Внутри раздела LdaprolesMapping есть настройка RolesMap, в которой и производится мэппинг ролей KS с ролями Active Directory: идентификатор роли в KS (строка 1 на ***Рисунок 2***) сопоставляется с наименованием роли в Active Directory (строка 2 на на ***Рисунок 2***).

Наименование роли в AD представляет собой массив строк, так как наименования ролей из LDAP поступают строками.

Одной роли в KS может быть сопоставлено несколько ролей в AD (несколько ролей AD перечисляются через запятую).

Внесенные изменения должны быть сохранены в поле БД.

Ниже представлен пример мэппинга нескольких ролей в KS с ролями в AD:

‘RolesMap’: {
‘UUID_KS1’: [‘AD_role1’,‘AD_role2’],
‘UUID_KS_admin’: [‘AD_admin’],
‘UUID_KSN’: [‘AD_roleN’]
}

Примечание:

При авторизации пользователя в системе происходит синхронизация с сервером Active Directory и пользователю предоставляются роли в KS, соответствующие ролям пользователя в AD, если настройки мэппинга ролей были скорректированы во время нахождения пользователя в системе, то новая роль не будет доступна пользователю до момента следующей авторизации в системе (до повторной синхронизации с сервером Active Directory).

6.4. Keycloak

Keycloak — это система управления доступом, обеспечивающая безопасную аутентификацию пользователей. Keycloak позволяет централизованно управлять учетными записями, упрощая процесс входа в систему.

В настройках платформы на вкладке «Информационная безопасность» есть механизм аутентификации с использованием Keycloak. Поля настройки, такие как «Realm», «Auth server url», «SSL Required», «Resource», «Client secret» можно настроить вручную или импортировать их из json-файла по кнопке «Импорт». Для получения такого файла нужно предварительно экспортировать

json из Keycloak.

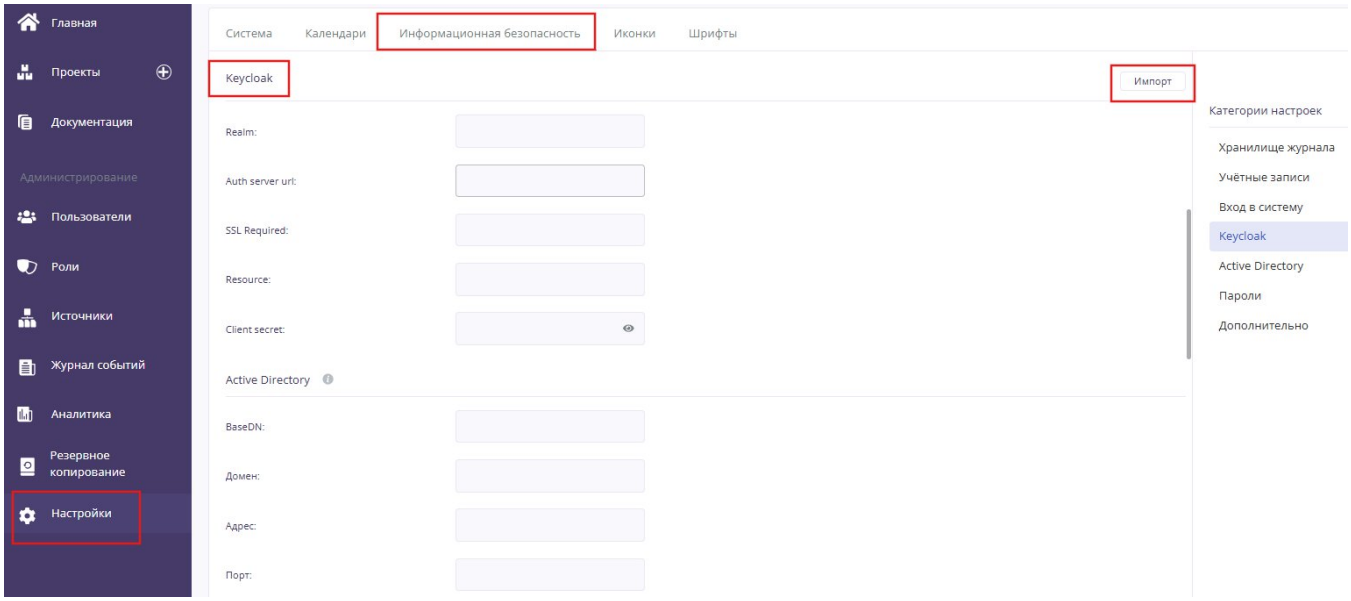


Рисунок 1. Настройки платформы на вкладке «Информационная безопасность»

Если настройки заполнены корректно, то на странице авторизации появится кнопка «Войти через Keycloak». После авторизации на платформе через Keycloak будет создан пользователь с таким же логином, как в соответствующей системе.

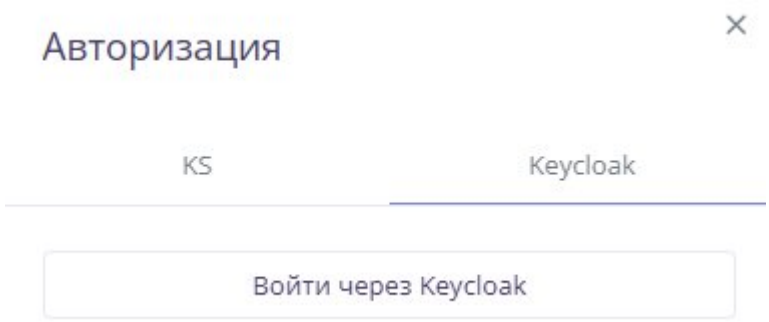


Рисунок 2. Окно авторизации

При открытии KS через Iframe (Мультипортал) можно скрыть возможность локальной авторизации по логину-паролю. То есть будет доступна авторизации только через Keycloak (с кнопкой):

Для этого необходимо заменить в Мультипортале ссылки, ведущие на KS, встроив в них параметр &/?auth=keycloak. При переходе по таким ссылкам окно авторизации будет выглядеть подобным образом:

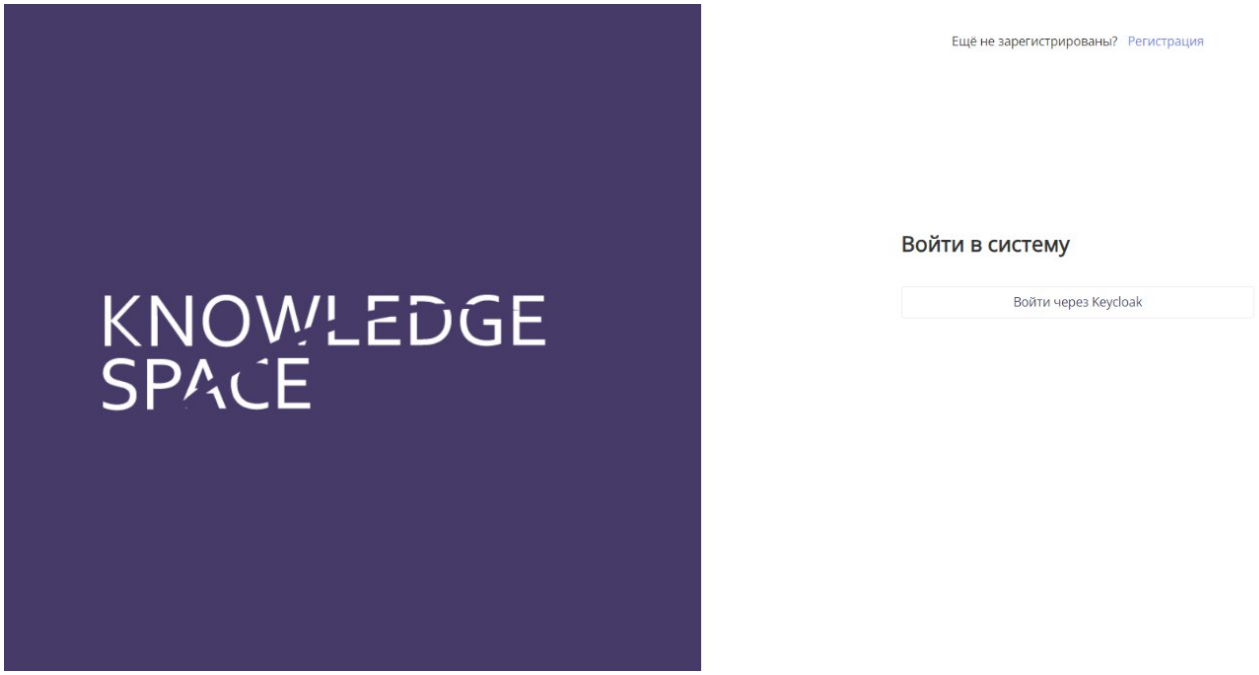


Рисунок 3. Окно авторизации

6.4.1. Используемые способы аутентификации

1. authorization_code

```
{
  code: [authorization_code]
}
```

2. login+password

```
{
  login: [login]
  password: [password]
}
```

3. refresh_token

```
{
  refreshToken: [refresh_token]
}
```

Валидация access token

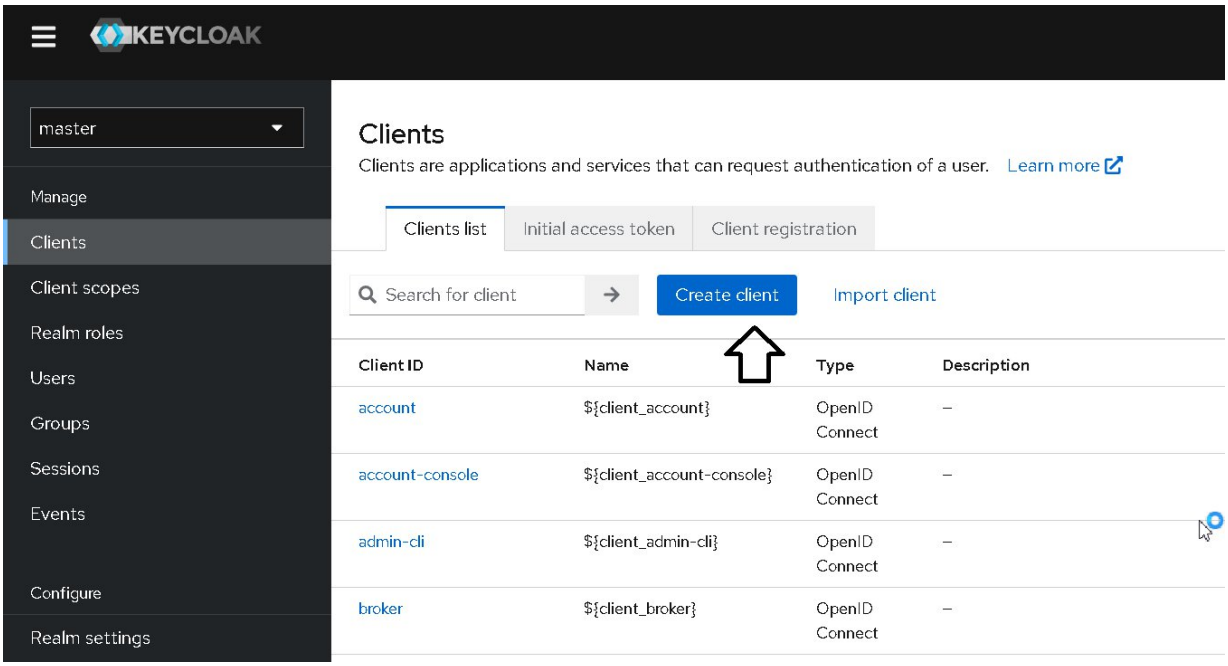
1. Настроена публикация API для внешней системы;
2. Внешняя система идет со своими кредами в Keycloak и получает access token;
3. Внешняя система обращается на наш эндпоинт, передавая в запросе access token Keycloak;

4. Мы идем в Keycloak и проверяем этот токен;
5. Результат:
    a. Если токен валиден, то отдаем данные внешней системе;
    b. Если токен не валиден, то возвращаем 403 ошибку (Неверный токен авторизации)
6. Поддерживает валидацию подписи keycloak token'a для алгоритмов HS256, HS384, HS512

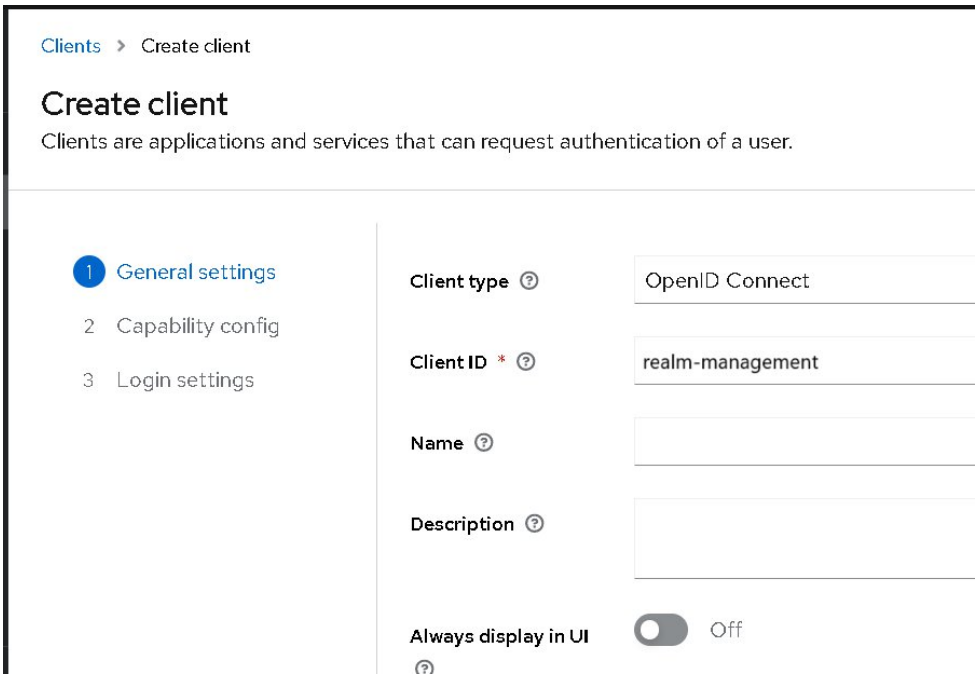
6.4.2. Настройка и маппинг ролей для входа с помощью Keycloak

Настройки keycloak

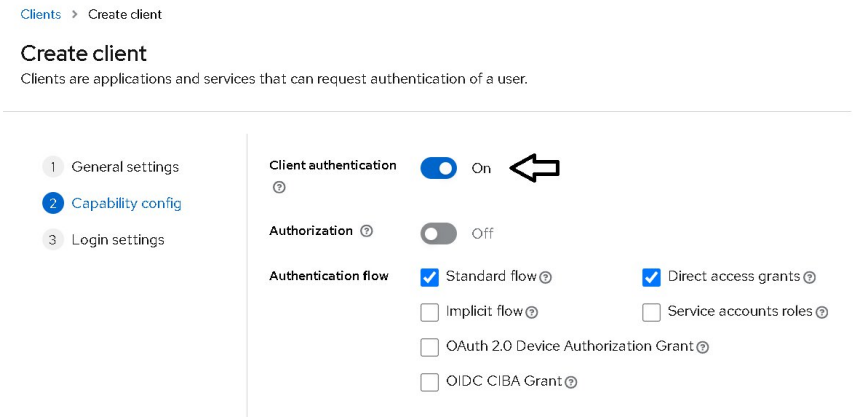
1. Добавление клиента
    1. Нажать Create client



    2. Указать имя клиента

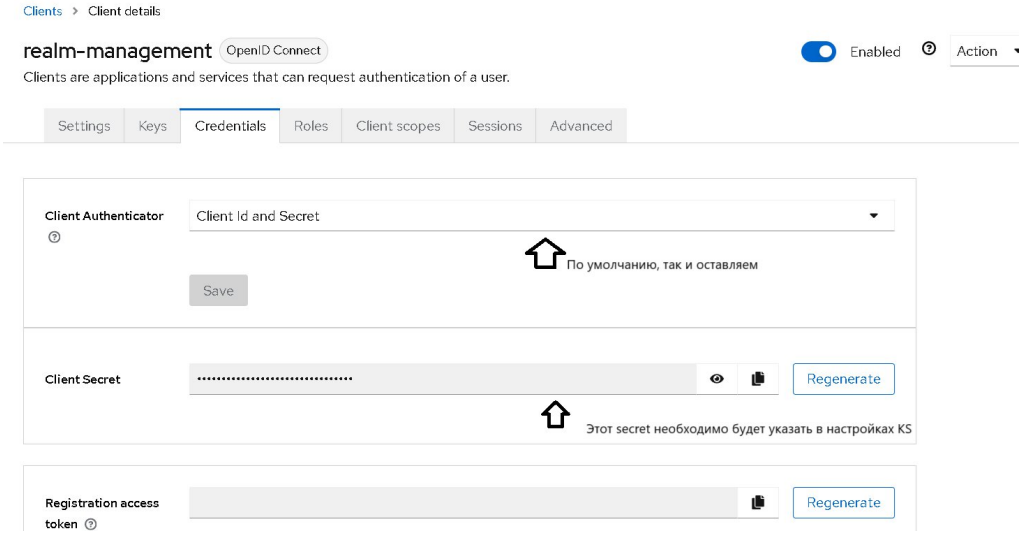


    3. Включить client authentification

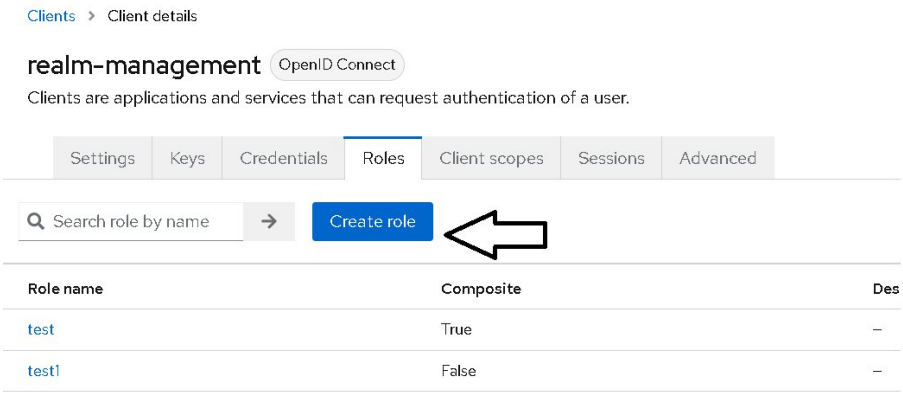


4. Сохранить изменения

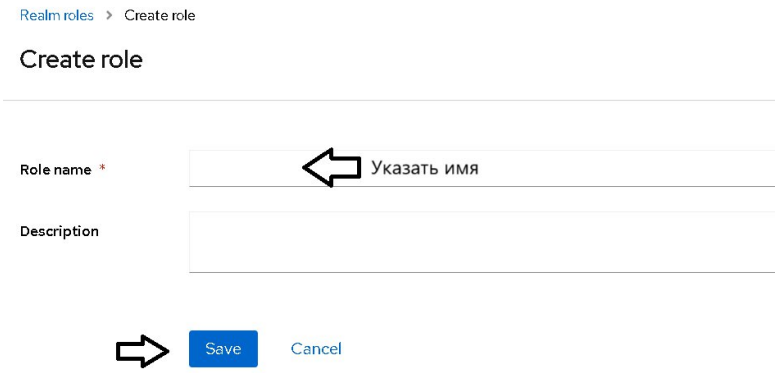
2. Редактирование клиента
   1. Найти в списке созданный клиент и открыть его
   2. Открыть вкладку "credentionals"



3. Добавление роли.
   1. Открыть realm roles и нажать Create role



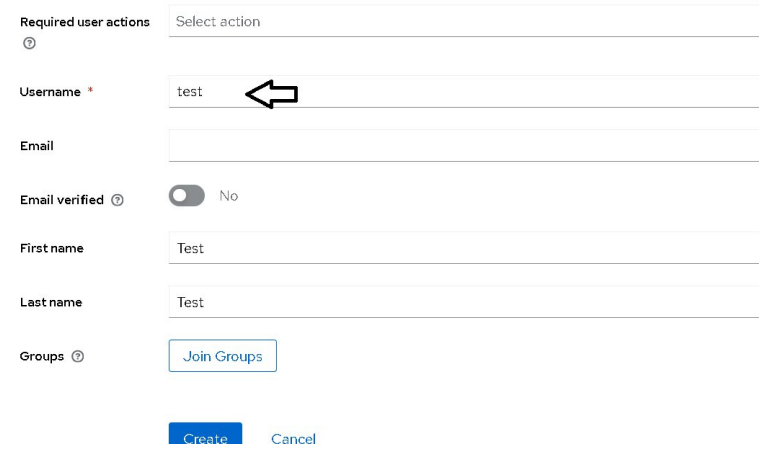
   2. Указать имя и сохранить



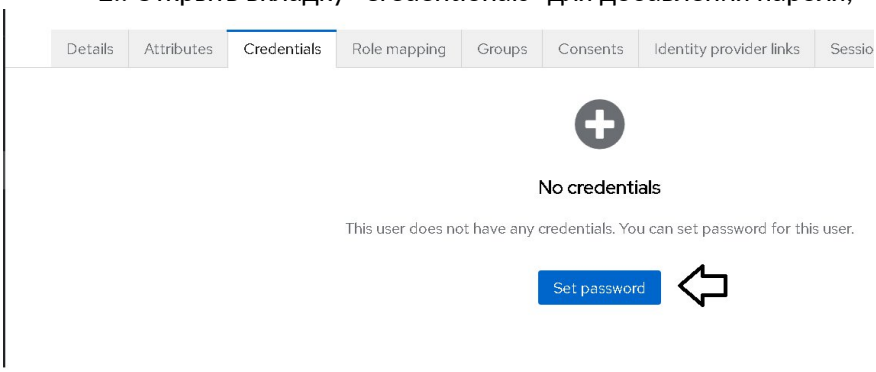
4. Добавление пользователя:

1. Открыть список пользователей и нажать на Add User

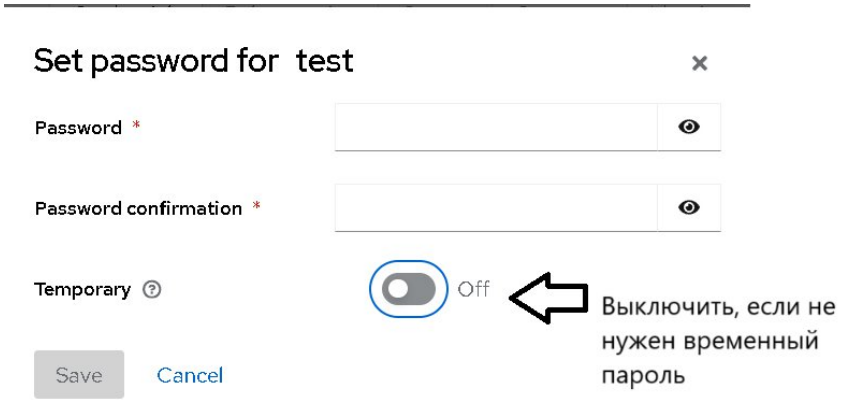
2. Указать username и нажать Create



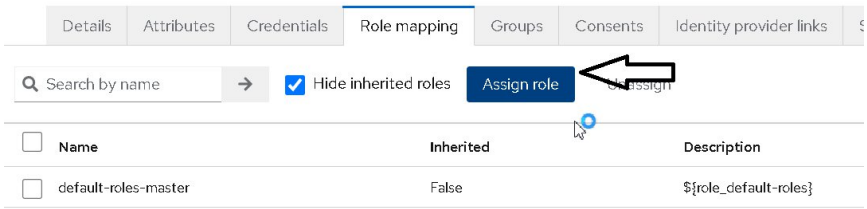
5. Редактирование пользователя:
   1. Найти в списке пользователей и открыть его для редактирования;
   2.. Открыть вкладку "Credentionals" для добавления пароля;



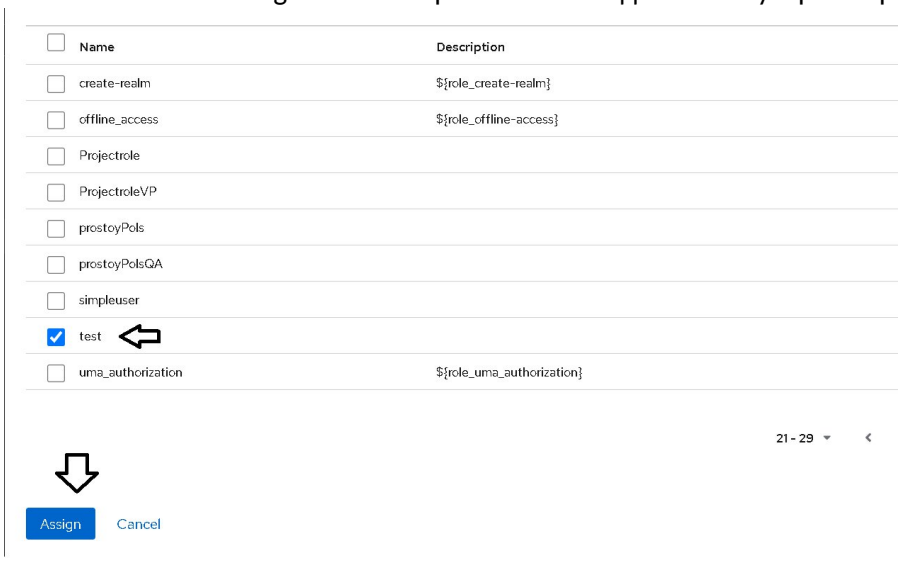
3. Нажать Set password и указать пароль



4. Открыть вкладку "Roles" для установки ролей:

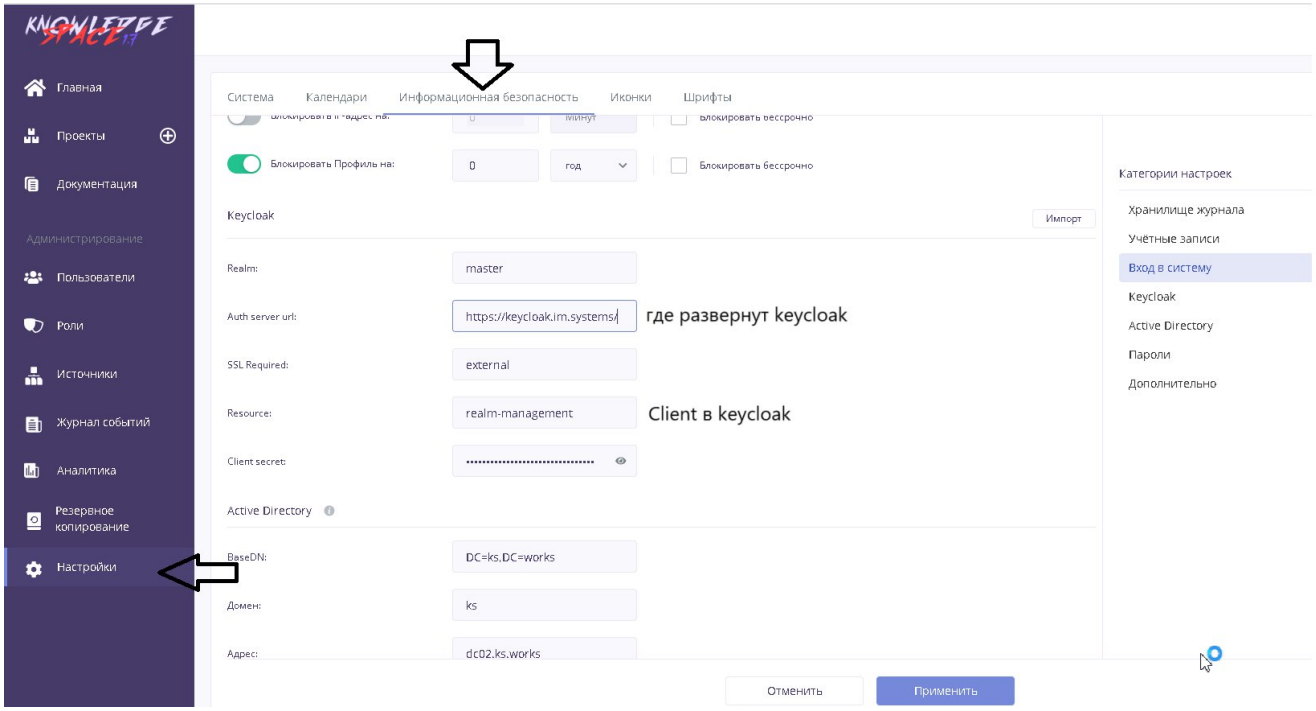


5. Нажать Assign role и выбрать из списка добавленную ранее роль

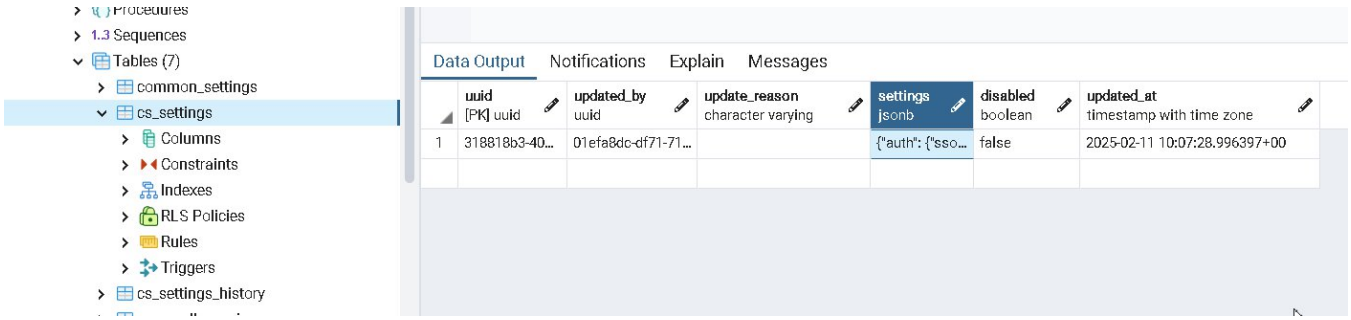


Настройки платформы KS

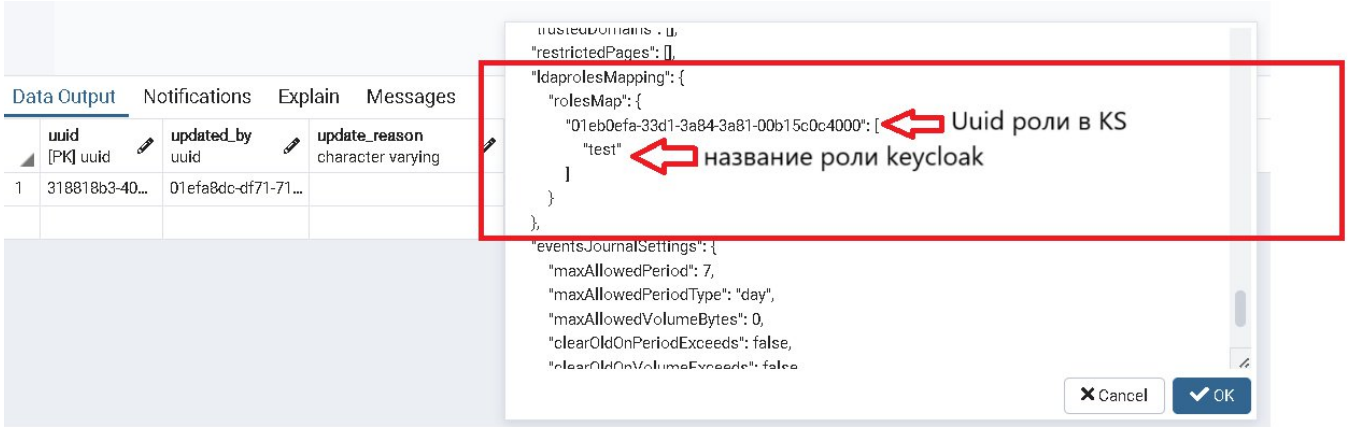
1. Подключение



2. Маппинг ролей keycloak и KS (Пока осуществляется на уровне базы данных KS):
   1. Перейти в базу данных ks_system_settings
   2. Открыть таблицу cs_settings



3. Отредактировать значение поля settings для раздела ldaprolesMapping



4. Если нужно несколько ролей KS сопоставить, то добавляем для каждой роли такое сопоставление типа

```
"ldaprolesMapping": {
  "rolesMap": {
    "01eb0efa-33d1-3a84-3a81-00b15c0c4000": [
      "simpleuser",
      "test"
    ],
    "01eb0efa-88ab-3a82-3a81-00b15c0c4000": [
      "simpleuser",
      "test"
    ]

  }
```


Аналогично настраивается и маппинг ролей KS - LDAP.

6.5. Аутентификация с помощью протокола Kerberos с поддержкой SSO

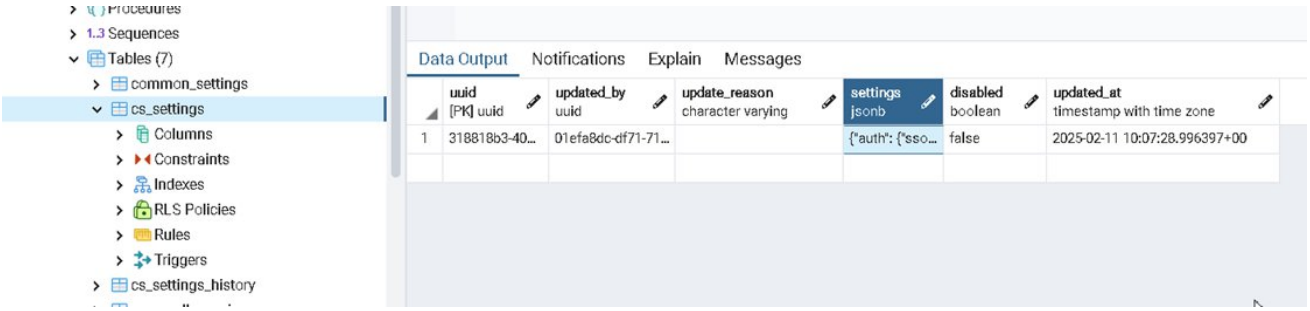
Настройки со стороны DevOps:

- Необходимо установить Kerberos;
- Получить keytab стенда, внести их в keytab-файл;
- Добавить этот файл в VOLUME;
- Добавить адрес файла в переменную окружения.

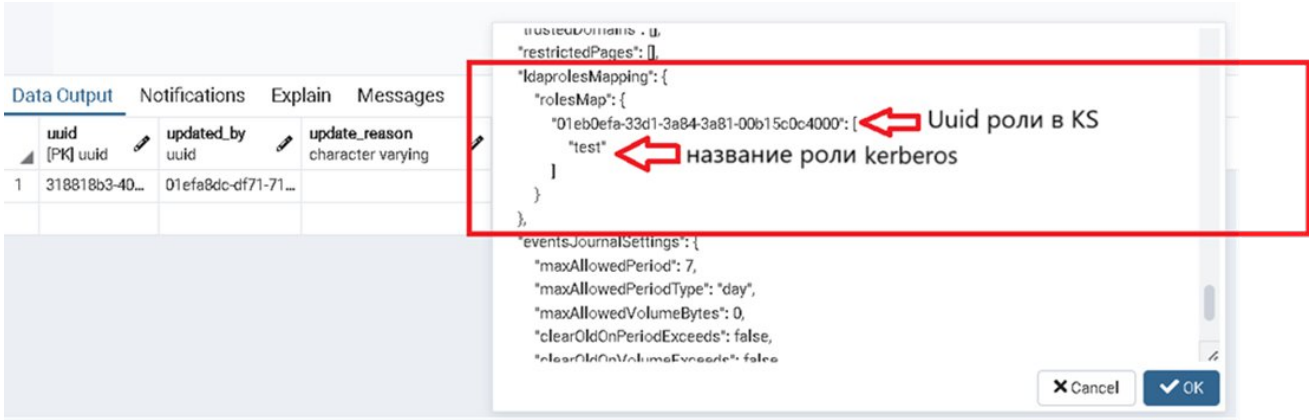
Настройки со стороны KS:

1. В настройках информационной безопасности системы имеется чекбокс «Включить SSO»:
   - По умолчанию он выключен;
   - При включении доступна аутентификация по протоколу Kerberos, для этого:
     - В сервис auth необходимо прокинуть переменную окружения AUTH_KERBEROS_KEYTAB_PATH, которая должна содержать путь к keytab-файлу. Если сервис auth запущен в docker контейнере вероятно так же понадобится смонтировать директорию в контейнер с keytab файлом.
     - По указанному пути уже должен лежать keytab-файл (Keytab (key table) — это файл, содержащий одну или несколько пар принципал + ключ, используемых для аутентификации в системе Kerberos. Он позволяет сервисам или пользователям проходить аутентификацию без ввода пароля вручную).

2. Маппинг ролей. Аналогично ldap или keycloak - Пока осуществляется на уровне базы данных KS
   - Перейти в базу данных ks_system_settings.
   - Открыть таблицу cs_settings:



- Отредактировать значение поля settings для раздела ldaprolesMapping



- Если нужно несколько ролей KS сопоставить, то добавляем для каждой роли такое сопоставление типа:

```
"ldaprolesMapping": {
  "rolesMap": {
    "01eb0efa-33d1-3a84-3a81-00b15c0c4000": [
      "simpleuser",
      "test"
    ],
    "01eb0efa-88ab-3a82-3a81-00b15c0c4000": [
      "simpleuser",
      "test"
    ]

  }
```

3. Как это работает:

- Ks на каждый ответ с 401 статусом будет слать дополнительный header WWW-Authenticate: Negotiate с попыткой инициировать процесс аутентификации по протоколу kerberos
- При получении header'a WWW-Authenticate: Negotiate, браузер (при нахождении в домене и корректной настройке) выполнит попытку по выпуску spnego ticket'a. В случае успеха повторит запрос который завершился с 401 ошибкой, добавив в него дополнительный header Authorization: Negotiate [SPNEGO TICKET].
- KS при получении тикета в заголовке Authorization попробует его дешифровать ключом из загруженного keytab-файла. В случае успеха создаст нового пользователя (по информации, которая содержится в тикете), если такого еще не создано, а так же добавит пользователю роли на основании id групп пользователя group membership из тикета.
- В случае успешного создания пользователя и маппинга ролей, для него создается новая сессия ks, выпускается новый jwt access token и возвращается в ответе в заголовке Set-Cookie.

4. Дополнительные нюансы:

- Если при локальной разработке возникла проблема, при которой браузер (Google Chrome) отказывался выдавать spnego ticket, пока домен web приложения не был добавлен в доверенные через реестр windows:
  - Для добавления необходимо перейти в редактор реестра Windows (ctrl+R -> regedit), перейти в HKEY_LOCAL_MACHINE -> SOFTWARE -> Policies -> Google -> Chrome. Добавить новый строковый параметр AuthNegotiateDelegateAllowlist с содержанием домена, для

которого будут выдаваться тикеты (например dev.ks.works). Добавить новый строковый параметр AuthServerAllowlist так же с доменом.

- Перезапустить браузер, перейти в настройки политик chrome://policy, убедиться, что chrome распознал добавленные правила.
- Проверить, имеется ли spn для аутентификации через kerberos можно командой klist (которую можно выполнить в терминале cmd windows). При выполнении команды в выводе должна быть информация о сервере, для которого разрешена выдача тикетов. Значение должно начинаться с HTTP/, например - Сервер: HTTP/domain, где domain – это домен системы ks (например dev.ks.works).

7 Управление информационной безопасностью

Система управления информационной безопасностью включает следующие элементы:

- Парольная политика;
- Параметры аутентификации;
- Журнал событий.

7.1 Парольная политика

Парольная политика задаётся в меню Настройки (закладка Информационная безопасность, раздел Пароли) и включает следующие параметры:

- Минимальное количество символов;
- Срок действия пароля;
- Параметры сложности пароля:
  - Обязательное наличие цифровых символов;
  - Обязательное наличие строчных букв;
  - Обязательное наличие прописных букв;
  - Обязательное наличие спецсимволов;
- Параметры запрета использования старых паролей:
  - Ограничение по количеству последних паролей, которые могут повторяться;
  - Полный запрет использования старых паролей.

7.2 Параметры аутентификации

Параметры аутентификации устанавливаются в меню **Настройки** (закладка **Информационная безопасность**, разделы **Учётные записи**, **Вход в систему**) и включают следующие компоненты:

- **Время жизни токена аутентификации** – время бездействия пользователя, которое должно пройти после успешной аутентификации, прежде чем токен будет считаться просроченным (следующий запрос к серверу с просроченным токеном приведёт к запросу повторной аутентификации):

  - Бездействием пользователя считается отсутствие запросов клиентской части к серверу. При достижении половины указанного времени жизни токен автоматически обновляется сервером и передаётся в виде заголовка ответа на любой из запросов к серверу.

- **Время жизни токена сброса пароля** – время действия ссылки, которая отправляется пользователю на адрес электронной почты при запросе сброса пароля;
- **Время жизни токена обновления устаревшего пароля** – время, которое начинается с момента первой успешной попытки входа с просроченным паролем. Токен сброса пароля позволяет в течение своего времени жизни сменить пароль при условии успешной аутентификации с просроченным паролем;
- **Количество неуспешных попыток** – количество неуспешных попыток входа под одной и той же учетной записью до осуществления её блокировки;
- **Время между неуспешными попытками** – временное окно, в рамках которого должно быть достигнуто указанное количество неуспешных попыток для осуществления блокировки;
- **Блокировать IP-адрес на** – указание на необходимость блокировки IP-адреса, с которого осуществлялись попытки входа и время такой блокировки;
- **Блокировать профиль на** – указание на необходимость блокировки учетной записи

пользователя, к которой осуществлялись попытки доступа и время такой блокировки.

7.3 Журнал событий

Журнал событий накапливает историю событий в системе в соответствии со своей спецификацией (документ **MS Word Knowledge Space. Журнал событий.docx**) и доступен в меню **Журнал событий**.

Администратору доступны параметры автоматической очистки устаревших событий из журнала в меню **Настройки** (закладка **Информационная безопасность**, раздел **Хранилище журнала**):

- **Срок хранения записей** – предельный срок в течение которого запись гарантированно не будет удаляться системой, в т.ч. с указанием необходимости уведомления о приближении такого срока для имеющихся в журнале событий;
- **Макс. объем хранения записей** – предельный объем, занимаемый журналом на жестком диске, в т.ч. с указанием необходимости уведомления о приближении объема к указанному. При попытке добавления новых записей сверх указанного объема самые старые записи будут удаляться из журнала.

Для просмотра «Журнала событий» необходимо выбрать пункт «Журнал событий» в навигационном меню.

Можно выбрать шаблон для отображения определенных столбцов (параметров события) в журнале событий. Также можно добавить или убрать отдельные столбцы, отметив их в выпадающем списке.

Для фильтрации списка пользователей нажмите на «Фильтр». В левой части экрана появится панель «Настройки фильтрации». Отфильтровать события можно по дате/периоду и по настраиваемым критериям. Для того, чтобы добавить критерий в качестве фильтра, выберите строку в появившемся списке. Настроенный фильтр можно применить или сбросить для отображения полного списка событий.

В нижней части экрана можно осуществить переход по страницам списка, а также настроить количество отображаемых на одной странице строк (событий).

Журнал событий доступен к просмотру через таблицы базы данных ks_system_events_journal:

- system_events – общая информация о системном событии;
- extended_data – дополнительная информация для событий информационной безопасности.

7.3.1 Настройки информационной безопасности

Для того, чтобы перейти к настройкам информационной безопасности, необходимо выбрать пункт «Настройки» в навигационном меню. В открывшемся окне нужно перейти на вкладку «Информационная безопасность».

В разделе **«Хранилище журнала»** можно:

1. Установить срок хранения записей, указав количество выбранных периодов. Включить уведомление о приближении окончания срока.
2. Установить максимальный объем сохраненных записей. Включить уведомление о приближении достижения максимального объема.

Раздел **«Учетные записи»** позволяет установить время жизни токена – время до автоматического выхода пользователя из системы при отсутствии его активности.

В разделе **«Вход в систему»** можно настроить блокировку IP-адреса или профиля пользователя

при заданном количестве неуспешных попыток входа в систему и заданном интервале между ними. Для этого необходимо:

1. Установить количество неуспешных попыток входа в систему.
2. Установить время, которое должно пройти между каждой парой неуспешных попыток.
3. Включить блокировку IP-адреса при выполнении условий 1 и 2. Установить период блокировки либо выбрать опцию «Блокировать бессрочно».
4. Включить блокировку профиля пользователя при выполнении условий 1 и 2. Установить период блокировки либо выбрать опцию «Блокировать бессрочно».

Раздел **«Active Directory»** позволяет:

1. Установить параметры подключения к Active Directory. Стандартный способ управления паролями учетных записей.
2. Для целей создания служебных учетных записей (например, для настройки интеграции) в разделе «Пароли» можно указать минимальное количество символов для пароля и отметить параметры его сложности:

   - должен включать цифры;
   - должен включать строчные буквы;
   - должен включать прописные буквы;
   - не должен содержать спецсимволы.

Также в разделе «Пароли» можно активировать запрет использования старых паролей. В рамках данного запрета можно определить количество последних паролей, которые не должны повторяться, либо активировать опцию «Запрещать использовать любые старые пароли».

Для применения изменений, введенных в настройках информационной безопасности, необходимо нажать «Сохранить».

7.3.2 Просмотр/выгрузка логов

1. Выбрать пункт «Журнал событий» в левом меню.
2. Настроить необходимое отображение журнала событий:

   - Выбрать шаблон из выпадающего списка «Шаблон» для отображения определенных столбцов (параметров события);
   - Добавить или убрать отдельные столбцы, отметив их в выпадающем списке «Столбцы»;
   - Настроить фильтр для отображения событий, нажав на кнопку «Фильтр»; левой части экрана появится панель «Настройки фильтрации», которая позволяет:
     - Установить фильтр по дате;
     - Установить фильтр по периоду;
     - Установить фильтр настраиваемым критериям. Для того, чтобы добавить критерий в качестве фильтра, необходимо нажать «Добавить критерий» и выбрать строку в появившемся списке.

Настроенный фильтр можно либо применить, либо сбросить для отображения полного списка событий.

8 Аналитика

Для просмотра аналитики по проектам, пользователям и их сессиям необходимо выбрать пункт «Аналитика» в навигационном меню платформы.

8.1 Проекты

Вкладка «Проекты» предназначена для просмотра аналитики в разрезе проектов.

В верхней части страницы находятся счетчики проектов и приложений в системе.

Ниже в формате таблицы представлен перечень проектов и их характеристики, которые расположены в столбцах:

- Название проекта
- Тип (dev/prod)
- Владелец
- Приложение
- Количество доступов
- Дата создания
- Последний вход

Отдельные столбцы можно добавить или убрать, отметив их в выпадающем списке поля 

в правом верхнем углу над таблицей.

Для фильтрации списка проектов нажмите на 

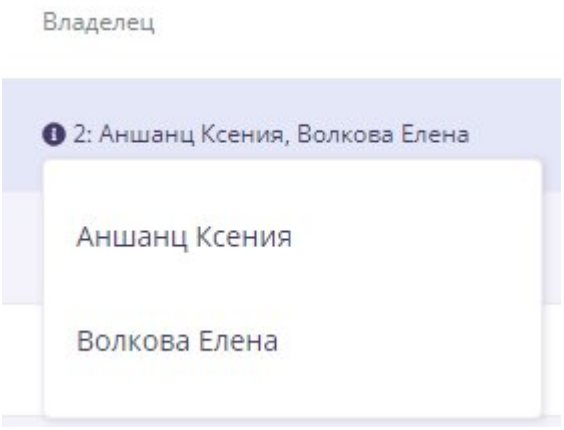
. В правой части экрана появится панель «Настройки фильтрации». Отфильтровать проекты можно по 7ми критериям, которые соответствуют столбцам таблицы. В списке выберете один или несколько критериев и в появившихся полях укажите интересующие значения. Настроенный фильтр можно применить или сбросить для отображения полного списка проектов.

Отфильтрованную и отсортированную таблицу со списком проектов можно скачать по иконке  . Для выгрузки доступны форматы Xlsx и CSV. По умолчанию установлена опция 

, однако при необходимости можно выключить ее и в поле ниже ограничить количество экспортируемых строк до любого значения.

С более подробной информацией можно ознакомиться по нажатию на иконку  . Она доступна для следующих столбцов:

- Название проекта (иконка появляется при наведении на значения данного столбца) - для каждого проекта в списке позволяет открыть подробную аналитику по «Пользователям с доступом» и «Пользователям-владельцам». При выборе одного из вариантов будет автоматически осуществлен переход на вкладку «Пользователи» (подробнее в п.8.2) и отображена таблица, отфильтрованная по соответствующим юзерам.
- Владелец – информационная иконка позволяет раскрыть полный перечень владельцев проекта.
  - При нажатии на имя пользователя в списке будет осуществлен переход в его профиль.
  - Справа от иконки отображается общее число владельцев проекта.



- Приложение - информационная иконка позволяет раскрыть полный перечень приложений, созданных в рамках проекта.
    - При нажатии на приложение в списке публикация откроется для просмотра в новом окне (при наличии у администратора доступа).
    - Справа от иконки отображается общее число приложений проекта.
- Количество доступов - информационная иконка позволяет раскрыть полный перечень пользователей, которым выдан доступ к проекту.
    - При нажатии на имя пользователя в списке будет осуществлен переход в его профиль.
    - Справа от иконки отображается общее число выданных доступов.

В нижней части экрана можно осуществить переход по страницам списка, а также настроить количество отображаемых на одной странице строк (проектов).

8.2 Пользователи

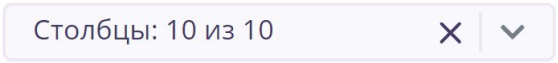
Вкладка «Пользователи» предназначена для просмотра аналитики в разрезе пользователей.

В верхней части страницы находятся счетчики пользователей в системе, в том числе построенные по их общему количеству, количестве активных (незаблокированных), заблокированных и пользователей онлайн.

Ниже в формате таблицы представлен перечень пользователей и их характеристики, которые расположены в столбцах:

- Пользователь
- Логин
- Электронная почта
- Доп. свойства - перечень значений кастомных полей, добавленных администратором для указания дополнительной информации о пользователе, если соответствующие значения заполнены в профиле пользователя (подробнее в п.3.6.1 Руководства no/low-code разработчика платформы)
- Роль (перечень системных ролей)
- Владелец проектов
- Доступ к проектам
- Доступ к приложениям
- Последний вход (дата последнего входа или пометка «Онлайн»)

Отдельные столбцы можно добавить или убрать, отметив их в выпадающем списке поля



в правом верхнем углу над таблицей.

Для фильтрации списка пользователей нажмите на 

. В правой части экрана появится панель «Настройки фильтрации». Доступна фильтрация

- По дате последнего входа (опции «Не учитывать», «Выбранная дата», «Между», «За последние *количество периодов*»)
- По критериям, которые соответствуют столбцам таблицы. В списке выберете один или несколько критериев и в появившихся полях укажите интересующие значения.

Настроенный фильтр можно применить или сбросить для отображения полного списка пользователей.

Отфильтрованную и отсортированную таблицу со списком пользователей можно скачать по иконке  . Для выгрузки доступны форматы Xlsx и CSV. По умолчанию установлена опция 

, однако при необходимости можно выключить ее и в поле ниже ограничить количество экспортируемых строк до любого значения.

С более подробной информацией можно ознакомиться по нажатию на иконку  . Она доступна для следующих столбцов:

- **Пользователь** (иконка появляется при наведении на значения данного столбца) – для каждого пользователя позволяет открыть подробную аналитику по проектам, где он является владельцем/имеет доступ. При выборе одного из вариантов будет автоматически осуществлен переход на вкладку «Проекты» (подробнее в п.8.1) и отображена таблица, отфильтрованная по авторству/наличию доступа у конкретного пользователя.
- **Роль** – информационная иконка позволяет раскрыть полный перечень системных ролей, выданных пользователю. Справа от иконки отображается общее число ролей
- **Доступ к проектам** - информационная иконка позволяет раскрыть полный перечень проектов, к которым есть доступ у данного пользователя.
    - При нажатии на проект в списке в новом окне будет открыт раздел «Проекты» в главном меню платформы и раскрыты детали выбранного проекта (при наличии у администратора доступа).
    - Справа от иконки отображается общее число доступов пользователя к проектам платформы.
- **Доступ к приложениям** - информационная иконка позволяет раскрыть полный перечень приложений, к которым есть доступ у данного пользователя.
    - При нажатии на приложение в списке публикация откроется для просмотра в новом окне (при наличии у администратора доступа).
    - Справа от иконки отображается общее число доступов пользователя к приложениям платформы.

В нижней части экрана можно осуществить переход по страницам списка, а также настроить количество отображаемых на одной странице строк (пользователей).

8.3 Сессии

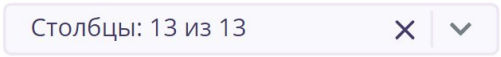
Вкладка «Сессии» предназначена для просмотра аналитики в разрезе сессий. Просмотр аналитики может вестись в рамках сессий на платформе или в приложении. В зависимости от выбора состав страницы с аналитикой будет различным.

Платформа

В верхней части страницы находится счетчик активных пользователей в пределах системы.

Ниже в формате таблицы представлен перечень сессий и их характеристики, которые расположены в столбцах:

- Сессия
- Пользователь
- Логин
- Электронная почта
- Начало
- Окончание
- Длительность сессии
- IP Адрес
- Устройство
- Операционная система
- Браузер
- Версия браузера

Отдельные столбцы можно добавить или убрать, отметив их в выпадающем списке поля 

в правом верхнем углу над таблицей.

Для фильтрации списка сессий нажмите на 

. Фильтрация может быть осуществлена по значениям в любом из столбцов (доступен выбор нескольких).

Отфильтрованную и отсортированную таблицу со списком сессий можно скачать по иконке  . Для выгрузки доступны форматы Xlsx и CSV. По умолчанию установлена опция 

, однако при необходимости можно выключить ее и в поле ниже ограничить количество экспортируемых строк до любого значения.

По нажатию на имя пользователя в столбце «Пользователь» будет осуществлен переход на вкладку «Пользователи» (подробнее в п.8.2) и в автоматически отфильтрованной таблице отобразятся характеристики выбранного пользователя.

В нижней части экрана можно осуществить переход по страницам списка, а также настроить количество отображаемых на одной странице строк (сессий).

9 Приложения

9.1 Приложение 1. Общая архитектура системы

KNOWLEDGE SPACE использует микросервисную архитектуру с применением Open-source инфраструктурных компонентов (*Рисунок 5*). Перечень таких компонентов и их роли представлены ниже:

- СУБД – **PostgreSQL** 12+;
- Обнаружение сервисов – **Consul** 1.9.3+;
- Очередь сообщений – **RabbitMQ** 3.8.9+;
- Хранение полезной нагрузки объемных сообщений – **Redis** 6.2.1+.

Компоненты системы можно разделить на две группы – внешние и внутренние.

Внешние компоненты:

СУБДPostgreSQL

В качестве СУБД используется PostgreSQL. Каждый сервис KS имеет свою отдельную БД. При разворачивании СУБД создается инстанс и около 30 отдельных БД для каждого сервиса KS. Такой подход также может позволить в случае высокой нагрузки на СУБД разнести базы данных сервисов на разные сервера.

Consul

Микросервисная архитектура предполагает, что приложение состоит из набора программ, которые взаимодействуют между собой. Также некоторые микросервисы могут быть запущены в виде нескольких экземпляров, при этом вся нагрузка должна каким-то образом распределяться между ними. Для того, чтобы все сервисы имели возможность взаимодействовать между собой используется служба Service Discovery, которая в KS представлена в виде Consul. Каждый сервис при запуске регистрируется и через Consul может получать адреса требуемых сервисов. Consul также отслеживает состояние зарегистрированных сервисов, если сервис перестает отвечать, он вычеркивает его из списка доступных.

RabbitMQ

Сервисы внутри системы могут взаимодействовать между собой асинхронно для выполнения каких-либо длительных задач. Для организации очереди и асинхронных взаимодействий между микросервисами в KS используется RabbitMQ.

Redis

NoSQL СУБД с открытым исходным кодом, работающая со структурами данных key-value. Позволяет сохранять некоторые сущности и, т.к. БД находится в памяти, получать их с высокой скоростью. В KS используется в качестве кэша для данных, передаваемых между сервисами системы. В случае необходимости передачи между микросервисами больших объемов данных, в RabbitMQ отправляется только идентификатор, ссылающийся на пакет в redis.

Внутренние компоненты:

Веб-сервер

В качестве веб-сервера в KS используется Nginx. Клиент формирует запросы, которые отправляются в backend через две точки:

- **Gateway** – микросервис, который проверяет авторизацию пользователя, права пользователя в проекте, дополняет запрос служебной информацией и передает нужному микросервису.
- **Websocket** – микросервис, позволяющий другим микросервисам обмениваться данными,

а также отправлять данные из серверной части обратно в клиентскую по подписке на некоторые события, обеспечивая интерактивность при выполнении длительных задач.

Микросервисы

Микросервисы содержат непосредственно логику работы KS.

Предпочтительный подход при установке компонент системы – контейнеризация, используется Docker или оркестратор контейнеров OpenShift от RedHat. Каждый компонент системы устанавливается в отдельном контейнере и имеет свою БД. Такой подход позволяет более гибко настроить работу системы, правильно распределять нагрузку, обеспечить отказоустойчивость на уровне контейнера. Также возможна установка системы без использования контейнеров.

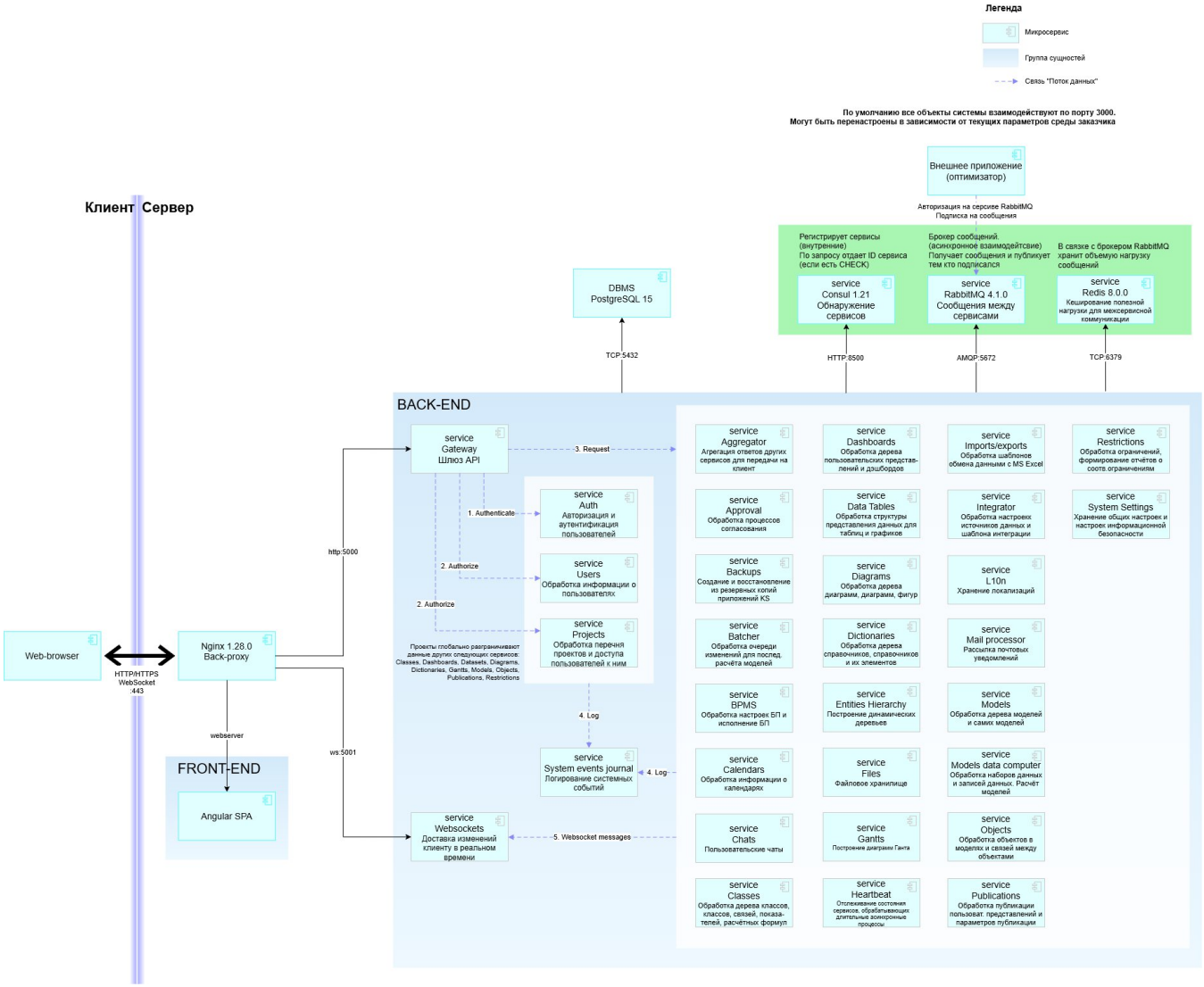


Рисунок 5. Общая архитектура системы

9.2 Приложение 2. Требования к технологической инфраструктуре

1. Требования к конфигурации серверов (при локальном развертывании)

Таблица 1. Требования к характеристикам целевой технологической архитектуре (инфраструктуре) проектного решения

№	Сервер			Количество vCPU, Шт.	Объём памяти RAM, ГБ.	Объём HDD ГБ.	Тип	Объём РК, ГБ.
	Роль сервера	**Назначение**	**Операционная система, Общесистемное прикл. ПО**	**Количество vCPU, Шт.**	**Объём памяти RAM, ГБ.**	**Объём HDD ГБ.**	**Тип**	**Объём РК, ГБ.**
1.	Сервер приложений	Продуктивная среда	RH UBI 8.6-941 / RedOS Consul 1.11.1 RabbitMQ 3.9.5 Knowledge Space 1.3SP1 Nginx 1.18 Redis 6	40	192	200	PV nonssd	–
2.	Сервер БД	Продуктивная среда	Astra Linux SE 1.7 / Windows Server Postgres Pro Std 12 Knowledge Space 1.3SP1	12	48	1000	Silver	2000
3.	Сервер оптимизатора	Продуктивная среда	Astra Linux SE 1.7 / Windows Server	16	32	250	Silver	–
4.	Сервер приложений	Тестовая среда	RH UBI 8.6-941 / RedOS Consul 1.11.1 RabbitMQ 3.9.5 Knowledge Space 1.3SP1 Nginx 1.18 Redis 6	40	96	21	PV nonssd	–
5.	Сервер БД	Тестовая среда	Astra Linux SE 1.7 / Windows Server Postgres Pro Std 12 Knowledge Space 1.3SP1	6	12	500	Silver	500
6.	Сервер оптимизатора	Тестовая среда	Astra Linux SE 1.7 / Windows Server	4	8	250	Silver	–

2. Требования к конфигурации рабочего места

Таблица 2. Требования к рабочим местам пользователей

№	Характеристика	Значение
1.	Процессор	Intel Core i3 и выше AMD Ryzen 3 и выше
2.	Оперативная память	Не менее 4 ГБ
3.	Постоянная память	–
4.	Операционная система	MS Windows 10, Astra Linux 1.7 SE, прочие с поддержкой работы перечисленных ниже браузеров
5.	Системное программное обеспечение	Web-клиент Microsoft Edge 87+, Яндекс. Браузер 20+ для Astra Linux, иные браузеры на базе ядра Chromium
6.	Монитор	24" 1920x1080

9.3 Приложение 3. Развертывание KS на пространстве заказчика