

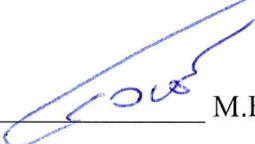
Перв. примен.	
Справ. №	

Подп. и дата	
Инь № дубл.	
Взамен инв. №	
Подп. и дата	

Инь № подл.	
-------------	--

УТВЕРЖДАЮ

Генеральный директор  
ООО «ЗЕН-АЙТИ»

  
М.Б. Буйлов

«    »                      2023 г.



Система для управления нормативно-справочной информацией «ЗЕН-НСИ»

Руководство системного программиста (администратора)

Изм.	Лист	№ докум.	Подп.	Дата				
					Руководство системного программиста (администратора)	Лит	Лист	Листов
Разраб.								54
Пров.						ООО «ЗЕН-НСИ»		
Н. контр.								
Утв.								



## СОДЕРЖАНИЕ

1.	НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ .....	5
1.1	Перечень объектов автоматизации и объектов внедрения, для которых используется ЗЕН-НСИ .....	5
1.2	Функционал ЗЕН-НСИ .....	5
1.3	Сведения о технических и программных средствах, обеспечивающих работу ЗЕН-НСИ .....	6
1.3.1	Требования к техническим и программным характеристикам сервера для установки ЗЕН-НСИ .....	6
1.3.2	Требования к техническим характеристикам АРМ пользователя .....	6
1.3.3	Требования к техническим характеристикам АРМ системного администратора .....	7
2.	СТРУКТУРА ЗЕН-НСИ .....	8
2.1	Сведения о структуре ЗЕН-НСИ .....	8
2.2	Сведения о составных частях программы .....	8
2.3	Сведения о связях между составными частями программы .....	9
2.4	Сведения о связях с другими программами .....	11
3.	НАСТРОЙКА ЗЕН-НСИ .....	12
3.1	Настройка системы на программной платформе Docker .....	12
3.1.1	Установка программного обеспечения на Debian .....	13
3.1.1.1	Установка дополнительных пакетов .....	13
3.1.1.2	Установка Docker .....	13
3.1.1.3	Установка NARoxy .....	13
3.1.1.4	Установка Docker Compose .....	14
3.1.2	Установка программного обеспечения на РЕД ОС .....	14
3.1.2.1	Установка дополнительных пакетов .....	14
3.1.2.2	Установка Docker .....	14
3.1.2.3	Установка NARoxy .....	14
3.1.2.4	Установка Docker Compose .....	15
3.1.3	Проверка установки .....	15
3.1.4	Установка ЗЕН-НСИ .....	16
3.1.5	Настройка программного обеспечения .....	18
3.1.5.1	Настройка сертификатов .....	18
3.1.5.2	Настройка .envnsi .....	19
3.1.5.3	Настройка application.properties .....	19
3.1.5.4	Настройка docker-compose .....	21

Изм.	Лист	№ докум.	Подп.	Дата
------	------	----------	-------	------

3.1.5.5	Настройка haproxy.cfg .....	24
3.1.5.6	Запуск ЗЕН-НСИ .....	26
3.1.5.7	Настройка KeyCloak.....	28
4.	ИСПОЛЬЗОВАНИЕ ЗЕН-НСИ .....	38
4.1	АКТИВАЦИЯ ЛИЦЕНЗИИ .....	38
4.1.1	Сетевая активация лицензии .....	38
4.1.1.1	Сетевая активация лицензии через графический интерфейс .....	38
4.1.1.1	Сетевая активация лицензии через программный интерфейс .....	40
4.1.2	Автономная активация лицензии .....	42
4.1.2.1	Автономная активация лицензии через графический интерфейс .....	42
4.1.2.1	Автономная активация лицензии через программный интерфейс.....	45
4.2	РАБОТА С ЗЕН-НСИ.....	49
5.	ВОЗМОЖНЫЕ ПРОБЛЕМЫ ПРИ УСТАНОВКЕ.....	51
5.1	ОШИБКА ЗАПУСКА KEYCLOAK.....	51
6.	ТЕХНИЧЕСКАЯ ПОДДЕРЖКА.....	52

Инв. № подл.	Подп. и дата	Взамен инв. №	Инв. № дубл.	Подп. и дата					Лист
									4
Изм.	Лист	№ докум.	Подп.	Дата					



8) Ведение истории изменений данных НСИ;

9) Журналирование действий пользователей.

1.3 Сведения о технических и программных средствах, обеспечивающих работу ЗЕН-НСИ

ЗЕН-НСИ функционирует в программно-технической среде информационной системы Заказчика.

1.3.1 Требования к техническим и программным характеристикам сервера для установки ЗЕН-НСИ

Для функционирования ПО «ЗЕН-НСИ» требуется OpenJRE (версии 11) и web-сервер Nginx.

В качестве СУБД для хранения данных используется СУБД, обеспечивающая объектно-реляционное хранение данных – PostgreSQL версии 11.

Для работы с ПО «ЗЕН-НСИ» требуется веб-браузер Yandex (версии 22.11.0 и выше), Mozilla Firefox (версии 107.0.1 и выше), Google Chrome (версии 108.0.5359 и выше) или совместимые.

ПО поддерживает микросервисную архитектуру и может быть поставлено в виде docker – контейнера.

Минимальные требования к аппаратному обеспечению серверной части:

Процессор: Intel Xeon E-2276M BGA1440

Оперативная память: не менее 8 ГБ\*

Жесткий диск: 100 ГБ\*

*\* Ресурсы должны быть увеличены, в случае обработки справочников больших размеров.*

Для настройки и установки специального программного обеспечения необходимо обладать правами системного программиста (администратора).

1.3.2 Требования к техническим характеристикам АРМ пользователя

Требования к необходимым для работы пользователя с ЗЕН-НСИ техническим средствам определяются свойствами ОС и архитектурой вычислительного процесса в системе.

Основываясь на обобщенных оценках, можно считать, что минимальными требованиями к ПЭВМ, функционирующими под управлением ОС Windows или Linux являются:

- процессор не менее Intel Core i3-12100;
- оперативная память от 4 ГБ;

Изм.	Лист	№ докум.	Подп.	Дата

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					6

- дисковая подсистема от 50 ГБайт;
- сетевой контроллер не хуже 100 Мбит/с;
- клавиатура PS/2 104 кл. CYR/LAT Win'95;
- ручной манипулятор типа мышь MOUSE 2 button PS/2.

1.3.3 Требования к техническим характеристикам АРМ системного администратора  
 Требования к необходимым для работы системного администратора с ЗЕН-НСИ техническим средствам равнозначны требованиям к АРМ пользователя (см. пункт 1.3.2).

Инва № подл.	Подп. и дата	Взамен инв. №	Инва № дубл.	Подп. и дата	
Изм.	Лист	№ докум.	Подп.	Дата	Лист 7

## 2. СТРУКТУРА ЗЕН-НСИ

### 2.1 Сведения о структуре ЗЕН-НСИ

ЗЕН-НСИ предназначена для автоматизации процессов управления справочной информацией как автономно, так и в составе других информационных систем.

Обработка данных в ЗЕН-НСИ ведется в многопользовательском режиме с разграничением прав доступа.

### 2.2 Сведения о составных частях программы

В рамках работ по созданию ЗЕН-НСИ были разработаны следующие части:

- 1) Сервис (back-end)
- 2) Графический интерфейс (front-end)

В логической структуре Решения можно выделить модули, изображенные на рисунке 1.

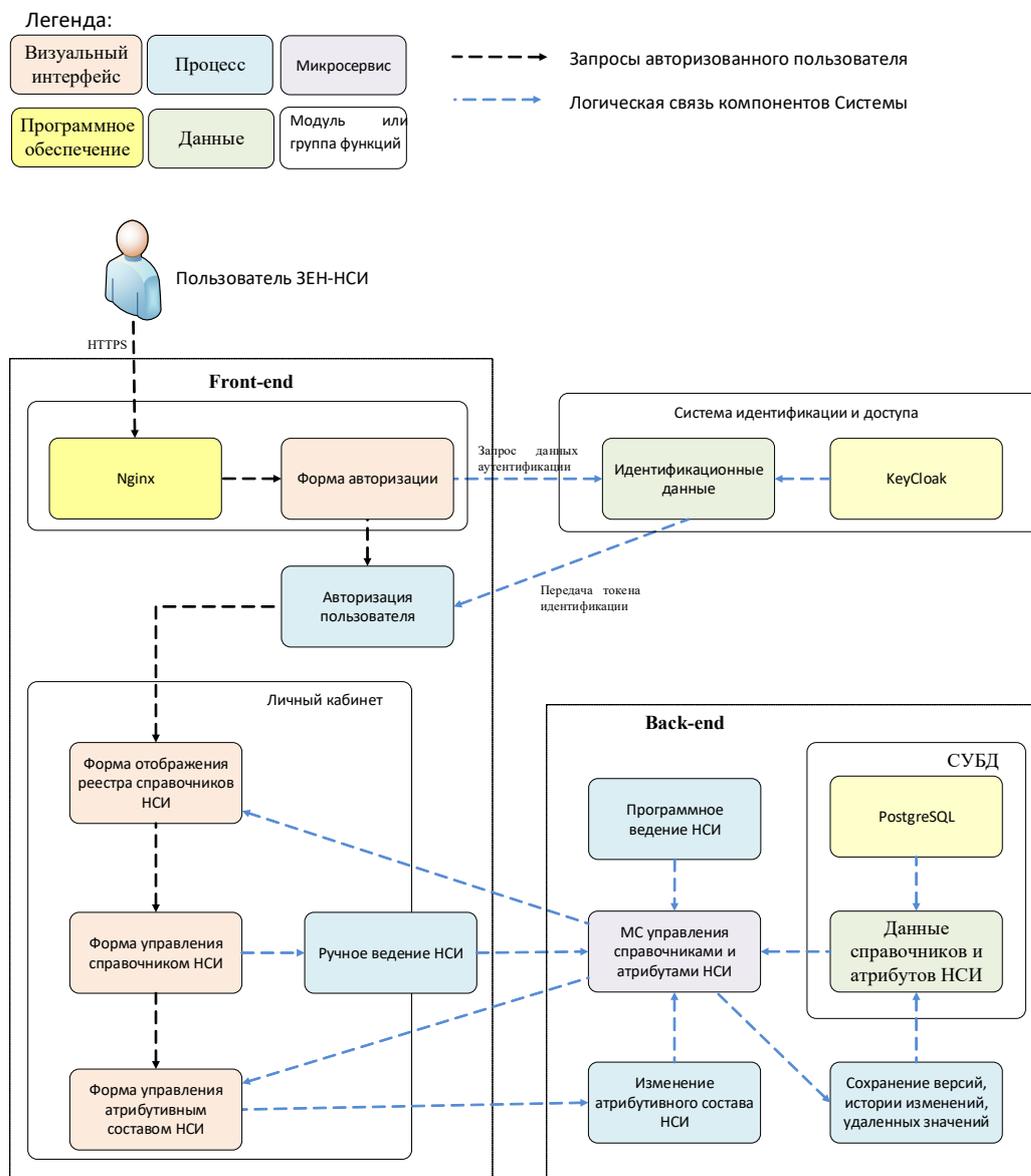


Рисунок 1 – Схема разделения функционала Front-end и Back-end

Изм.	Лист	№ докум.	Подп.	Дата
Изнв. № подл.	Взамен инв. №	Изнв. № дубл.	Подп. и дата	

- 1) Модуль front-end – графический интерфейс реализующий управление функционалом при взаимодействии пользователя с модулем back-end.
- 2) Модуль back-end – приложение обеспечивающее логику работы функционала, управление и обработку данных хранящихся в СУБД.
- 3) СУБД PostgreSQL – реляционная БД обеспечивающая хранение данных.
- 4) Модуль авторизации – внешняя система идентификации и доступа, свободно распространяемое программное обеспечение KeyCloak.

Взаимодействие пользователей с ЗЕН-НСИ осуществляется через web-интерфейс с помощью следующих последних стабильных версий веб-браузеров для настольных и переносных компьютеров или их эквивалентов:

- Yandex Browser;
- Mozilla Firefox;
- Google Chrome;
- Microsoft Edge.

ЗЕН-НСИ обеспечивает возможность автоматического переключения пользователя на версию страницы, адаптированную под размер экрана мобильного устройства.

### 2.3 Сведения о связях между составными частями программы

Принципиальная схема построения ЗЕН-НСИ представлена на Рисунке 2, имеет следующие элементы:

- HAProxy. Прокси-сервер и балансировщик нагрузки, обеспечивает распределение входящих обращений пользователей.
- KeyCloak. Внешняя подсистема авторизации, предоставляет возможности идентификации и доступ (может использоваться любая внешняя система авторизации поддерживающая протокол OAuth 2.0\*)
  - nsifront (модуль front-end) – графический интерфейс реализующий управление функционалом при взаимодействии пользователя с модулем back-end.
  - nsiback (модуль back-end) – приложение обеспечивающее логику работы функционала, управление и обработку данных хранящихся в СУБД.
  - СУБД PostgreSQL. Сервера баз данных для хранения данных приложений.

\*OAuth 2.0 — вторая версия открытого протокола (схемы) авторизации, обеспечивающий предоставление третьей стороне ограниченного доступа к защищённым ресурсам пользователя без передачи ей (третьей стороне) логина и пароля.

Изм.	Лист	№ докум.	Подп.	Дата

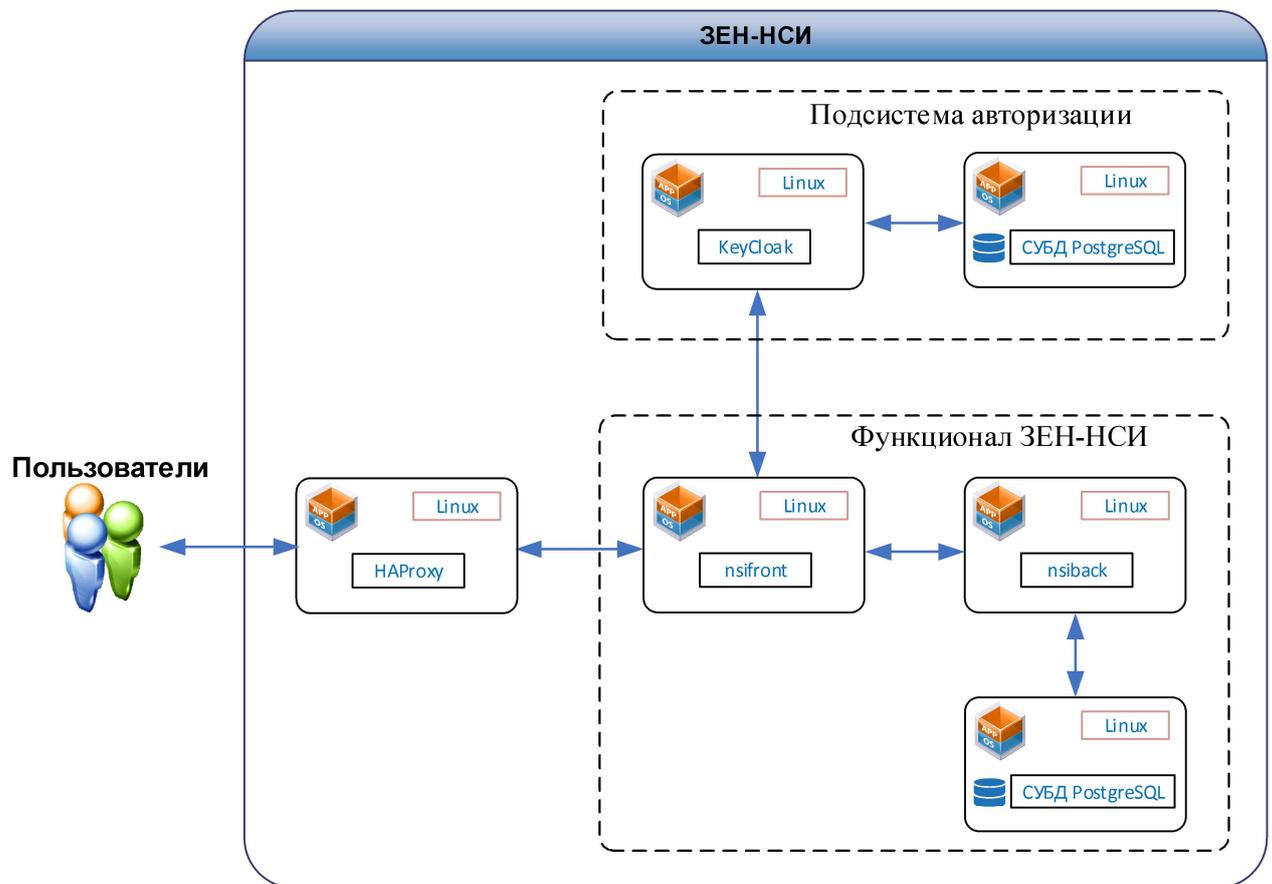
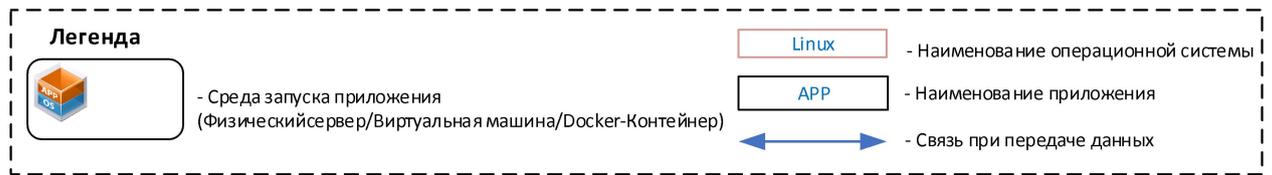


Рисунок 2 – Схема построения ЗЕН-НСИ

ЗЕН-НСИ построена на современной типовой архитектуре как приложение, которое может работать в средах виртуализации и контейнеризации.

Основные архитектурные принципы в процессе разработки:

- горизонтальная масштабируемость компонентов системы, включая автоматическую;
- возможность развертывания любого компонента ЗЕН-НСИ в отказоустойчивом варианте;
- возможность размещения в любом защищенном ЦОД (облаке);
- технологический стек – открытое ПО с запасом развития и поддержки на несколько лет вперед;

Ивл. № подл.	Подп. и дата
Взамен ивл. №	Ивл. № дубл.
Подп. и дата	Подп. и дата
Ивл. № подл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					10



### 3. НАСТРОЙКА ЗЕН-НСИ

Развертывание ЗЕН-НСИ может производиться как на физические сервера, так и в системы виртуализации с использованием программной платформы docker, а также программного обеспечения оркестровки контейнеризированных приложений типа kubernetes\*. Для выполнения установки и настройки ЗЕН-НСИ должно быть предварительно подготовлено программное окружение обеспечивающее работоспособность системы.

В данном руководстве будет описана настройка ЗЕН-НСИ на программной платформе Docker\* на операционной системе Debian свободного ПО с открытым исходным кодом и российской операционной системе общего назначения для серверов и рабочих станций РЕД ОС.

\* Docker — платформа контейнеризации с открытым исходным кодом, с помощью которой можно автоматизировать создание приложений, их доставку и управление.

#### 3.1 Настройка системы на программной платформе Docker

Для установки и работы ЗЕН-НСИ необходимо подготовить следующее программное окружение:

1. Операционная система семейства Linux Debian версия от 11 и выше или операционная система РЕД ОС 7.3 и выше;
2. Платформа Docker версия от 20.10 и выше.;
3. Решение для управления идентификацией и доступом Keycloak версия от 21 и выше (при использовании docker устанавливается и работает из образа ЗЕН-НСИ);
4. СУБД PostgreSQL версия от 11 и выше (при использовании docker устанавливается и работает из образа ЗЕН-НСИ).

Вышеуказанное программное обеспечение устанавливается и настраивается с использованием инструкций производителя программного обеспечения согласно процессам стандартной установки.

Linux Debian - <https://www.debian.org/doc/>

РЕД ОС - <https://redos.red-soft.ru/base/>

Docker - <https://docs.docker.com/get-started/overview/>

Keycloak - [https://www.keycloak.org/docs/latest/server\\_admin/](https://www.keycloak.org/docs/latest/server_admin/)

PostgreSQL - <https://www.postgresql.org/docs/>

Изм.	Лист	№ докум.	Подп.	Дата	Изнв. № подл.	Подп. и дата	Взамен инв. №	Изнв. № дубл.	Подп. и дата	Лист



#### 3.1.1.4 Установка Docker Compose

Установить Docker Compose:

```
sudo apt install docker-compose
```

#### 3.1.2 Установка программного обеспечения на РЕД ОС

Установка программного обеспечения для работы ЗЕН-НСИ производится в данной инструкции на сервер/виртуальную машину с установленной серверной версией РЕД ОС 7.3.

Перед началом установки программного обеспечения необходимо обновить список репозитория, выполните следующую команду с правами суперпользователя:

```
apt update
```

##### 3.1.2.1 Установка дополнительных пакетов

- Установка пакета certbot:

В репозиториях РЕД ОС нет пакета certbot. Для его установки подключим репозиторий EPEL (Extra Packages for Enterprise Linux) — это открытое бесплатное хранилище пакетов от проекта Fedora, содержащее пакеты дополнительного программного обеспечения для Linux.

Выполним установку пакета epel-release для подключения репозитория EPEL:

```
sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
```

Установим certbot:

```
sudo yum install certbot
```

##### 3.1.2.2 Установка Docker

- Установить Docker:

```
sudo yum install docker-ce
```
- Добавить docker в автоматический запуск при загрузке системы:

```
systemctl enable --now docker
```

##### 3.1.2.3 Установка HAProxy

- Установить HAProxy:

```
sudo yum install haproxy
```
- Добавить HAProxy в автоматический запуск при загрузке системы:

```
sudo systemctl enable --now haproxy
```
- Настроить SELinux

Настроим систему принудительного контроля доступа SELinux для работы HAProxy:

Изм.	Лист	№ докум.	Подп.	Дата	Подп. и дата	Изм. № дубл.	Взамен инв. №	Подп. и дата	Изм. № подл.	Подп. и дата	Лист

```
sudo setsebool -P haproxy_connect_any 1
```

- Запустить HAProxy:

```
sudo systemctl start haproxy
```

### 3.1.2.4 Установка Docker Compose

- Установить Docker Compose:

```
sudo yum install docker-compose
```

### 3.1.3 Проверка установки

- Проверить, что Docker установлен и работает:

```
sudo docker run hello-world
```

Пример корректного ответа:

```
[amd@localhost ~]$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
719385e32844: Pull complete
Digest: sha256:926fac19d22aa2d60fla276b66a20eb765fbeeaa2db5dbdaafeb456ad8ce81598
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/
```

Должно отображаться сообщение: This message shows that your installation appears to be working correctly.

- Проверить, что Docker Compose установлен и работает:

```
docker-compose --version
```

Пример корректного ответа:

```
[amd@localhost ~]$ docker-compose --version
Docker Compose version 2.11.2
```

Должен отобразиться номер установленной версии.

Изм.	Лист	№ докум.	Подп.	Дата

- Проверить, что HAProxy запущен и работает:

`sudo systemctl status haproxy`

Пример корректного ответа:

```
[and@localhost ~]$ sudo systemctl status haproxy
● haproxy.service - HAProxy Load Balancer
   Loaded: loaded (/usr/lib/systemd/system/haproxy.service; disabled; vendor preset: disabled)
   Active: active (running) since Fri 2023-08-04 19:13:58 MSK; 1min 50s ago
     Process: 4207 ExecStartPre=/usr/sbin/haproxy -f $CONFIG -f $CFGDIR -c -q $OPTIONS (code=exited, status=0/SUCCESS)
    Main PID: 4209 (haproxy)
       Tasks: 3 (limit: 7078)
      Memory: 4.9M
         CPU: 54ms
    CGroup: /system.slice/haproxy.service
           └─4209 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -f /etc/haproxy/conf.d -p /run/haproxy.pid
             4211 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -f /etc/haproxy/conf.d -p /run/haproxy.pid

авг 04 19:13:58 localhost.localdomain haproxy[4209]: [NOTICE] (4209) : New worker (4211) forked
авг 04 19:13:58 localhost.localdomain haproxy[4209]: [NOTICE] (4209) : Loading success.
авг 04 19:13:58 localhost.localdomain haproxy[4211]: [WARNING] (4211) : Server static/static is DOWN, reason: Layer4 connection pro
авг 04 19:13:58 localhost.localdomain haproxy[4211]: [ALERT] (4211) : backend 'static' has no server available!
авг 04 19:13:58 localhost.localdomain systemd[1]: Started HAProxy Load Balancer.
авг 04 19:13:59 localhost.localdomain haproxy[4211]: [WARNING] (4211) : Server app/app1 is DOWN, reason: Layer4 connection problem,
авг 04 19:13:59 localhost.localdomain haproxy[4211]: [WARNING] (4211) : Server app/app2 is DOWN, reason: Layer4 connection problem,
авг 04 19:14:00 localhost.localdomain haproxy[4211]: [WARNING] (4211) : Server app/app3 is DOWN, reason: Layer4 connection problem,
авг 04 19:14:00 localhost.localdomain haproxy[4211]: [WARNING] (4211) : Server app/app4 is DOWN, reason: Layer4 connection problem,
авг 04 19:14:00 localhost.localdomain haproxy[4211]: [ALERT] (4211) : backend 'app' has no server available!
```

Должно отображаться состояние: active (running)

**Если все пункты проверки выполняются без ошибок, то установка прошла успешно.**

### 3.1.4 Установка ЗЕН-НСИ

- Установить пакет wget:

`sudo apt install wget`

- Получить архив ЗЕН-НСИ

Ссылка на FTP сервер для скачивания с сайта <ftp://zen-it.ru:20201/>

Логин: customer-nsi

Возможно скачать архив напрямую с использованием утилиты wget

`wget --auth-no-challenge --user=customer-nsi --ask-password ftp://zen-it.ru:20201/nsi.tar.gz` где nsi.tar.gz – имя архива ЗЕН-НСИ

Пример работы утилиты:

```
root@deb12-nsi:/home/and# wget --auth-no-challenge --user=customer-nsi --ask-password ftp://zen-it.ru:20201/nsi.tar.gz
Пароль для пользователя «customer-nsi»:
```

Будет запрошен пароль для авторизации.

**Пароль для авторизации и получения архива ЗЕН-НСИ передается пользователю по запросу через контакты организации.**

Если данные логин/пароль были указаны правильно, будет произведено скачивание и сохранение архива.

Пример корректного скачивания архива ЗЕН-НСИ:

Изн. № подл.	Подп. и дата
Взамен инв. №	Подп. и дата
Изн. № дубл.	Подп. и дата

```

root@debl2-nsi:/home/and# wget --auth-no-challenge --user=customer-nsi --ask-password ftp://zen-it.ru:20201/nsi.tar.gz
Пароль для пользователя «customer-nsi»:
--2023-10-06 10:23:41-- ftp://zen-it.ru:20201/nsi.tar.gz
=> «nsi.tar.gz»
Распознаётся zen-it.ru (zen-it.ru)... 185.204.0.129
Подключение к zen-it.ru (zen-it.ru)[185.204.0.129]:20201... соединение установлено.
Выполняется вход под именем customer-nsi ... Выполнен вход в систему!
==> SYST ... готово. ==> PWD ... готово.
==> TYPE I ... готово. ==> CWD не нужен.
==> SIZE nsi.tar.gz ... 246858849
==> PASV ... готово. ==> RETR nsi.tar.gz ... готово.
Размер (байт): 246858849 (235M) (не достоверно)

nsi.tar.gz          100%[=====] 235,42M   845KB/s   за 4м 50с
2023-10-06 10:28:31 (831 KB/s) - «nsi.tar.gz» сохранён [246858849]

```

- Распаковать архив ЗЕН-НСИ

Выполнить команду находясь в каталоге со скачанным архивом:

```
sudo tar -C /opt/ -xvf nsi.tar.gz
```

Где nsi.tar.gz – имя распаковываемого архива

/opt – путь куда распаковывается архив

Пример корректной распаковки:

```

root@ub2204-nsi:/home/sadm# tar -xvf nsi.tar.gz
dist/
dist/init/
dist/init/init.sql
dist/.envnsi
dist/application.properties
dist/docker-compose.yml
dist/images.tar
dist/pgdata/
dist/pgdatansi/
root@ub2204-nsi:/home/sadm#

```

- Загрузить образ ЗЕН-НСИ в docker

`sudo docker load -i /opt/dist/images.tar` где /opt/dist/ - путь к образу images.tar

Пример загрузки образа:

```

root@debl2-nsi:/home/sadm# docker load -i dist/images.tar
d0fe97fa8b8c: Loading layer [=====] 72.49MB/72.49MB
fee20f1b745d: Loading layer [=====] 9.079MB/9.079MB
167efff21776: Loading layer [=====] 3.584kB/3.584kB
30ccd09e6f92: Loading layer [=====] 126.7MB/126.7MB
901602acdcf6: Loading layer [=====] 49.92MB/49.92MB
ea8e3fee67e4: Loading layer [=====] 39.94kB/39.94kB
bca066665b4a: Loading layer [=====] 1.12MB/1.12MB
621bc4c056ba: Loading layer [=====] 7.168kB/7.168kB
Loaded image: back:latest
f1417ff83b31: Loading layer [=====] 7.338MB/7.338MB
f33d2eed4384: Loading layer [=====] 5.326MB/5.326MB
d9c63c119ee8: Loading layer [=====] 3.584kB/3.584kB
b42e5b255ff1: Loading layer [=====] 4.608kB/4.608kB
effffea84b55: Loading layer [=====] 3.584kB/3.584kB
e43c8ee2056a: Loading layer [=====] 7.168kB/7.168kB
24e414d9930b: Loading layer [=====] 30.2MB/30.2MB
9614729d3ef1: Loading layer [=====] 1.087MB/1.087MB
290c116762cc: Loading layer [=====] 4.608kB/4.608kB
Loaded image: front:latest

```

Проверка корректной загрузки образа:

```
sudo docker images
```

Изм.	Лист	№ докум.	Подп.	Дата

```

root@debl2-nsi2:/home/sadm# docker images
REPOSITORY      TAG          IMAGE ID       CREATED        SIZE
front           latest      859332c316dd  5 weeks ago   42.1MB
back           latest      ef264b6b94c4  2 months ago  255MB
hello-world     latest      9c7a54a9a43c  2 months ago  13.3kB

```

Должны присутствовать репозитории **front** и **back**

### 3.1.5 Настройка программного обеспечения

Настройка программного обеспечения для работы ЗЕН-НСИ производится через конфигурационные файлы. **Перед настройкой конфигурационных файлов необходимо получить сертификат SSL для корректной работы ЗЕН-НСИ.**

#### 3.1.5.1 Настройка сертификатов

- Создание DNS записи:

Администратору домена пользователя необходимо создать **A** запись в системе доменных имен (DNS) чтобы она соответствовала внешнему ip адресу сервера, например:

**ваше\_доменное\_имя IN A IP-адрес.**

**пример: nsi-stand.zen-it.ru IN A 8.8.8.8.**

- Генерация запроса на получение сертификата:

`certbot certonly --standalone -d ваше_доменное_имя --non-interactive --agree-tos --email ваша_почта --http-01-port=8888`

Пример корректного получения сертификата:

```

Saving debug log to /var/log/letsencrypt/letsencrypt.log
Requesting a certificate for zennsi.dotbot.ru

Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/ваше_доменное_имя /fullchain.pem
Key is saved at: /etc/letsencrypt/live/ваше_доменное_имя /privkey.pem
This certificate expires on 2023-10-14.
These files will be updated when the certificate renews.
Certbot has set up a scheduled task to automatically renew this certificate in the background.

-----
If you like Certbot, please consider supporting our work by:
 * Donating to ISRG / Let's Encrypt: https://letsencrypt.org/donate
 * Donating to EFF: https://eff.org/donate-le
-----

```

- Устанавливаем сертификат

Создаем папку для хранения сертификата:

`sudo mkdir -p /etc/haproxy/certs`

Комбинируем сертификат (**выполнить команду с правами суперпользователя**):

```

cat /etc/letsencrypt/live/ваше_доменное_имя/fullchain.pem
/etc/letsencrypt/live/ваше_доменное_имя/privkey.pem >
/etc/haproxy/certs/DOMAINNAME.pem

```

Изм.	Лист	№ докум.	Подп.	Дата	Изм.	Лист	№ докум.	Подп.	Дата
------	------	----------	-------	------	------	------	----------	-------	------





*keycloak.cors-allowed-headers=X-Requested-With, Content-Type, Authorization, Origin, Accept, Access-Control-Request-Method, Access-Control-Request-Headers, Cache-Con>*

Где *keycloak.auth-server-url* – путь к сервису авторизации Keycloak, надо подставить доменное имя машины на которой разворачивается приложение. Если сервис авторизации разворачивается по умолчанию из образа ЗЕН-НСИ в докер контейнер или устанавливается на той же локальной системе где и остальные сервисы, оставляем значение по умолчанию ***localhost***.

#### 3.1.5.4 Настройка docker-compose

Внести в конфигурационный файл *docker-compose.yml* необходимые изменения (выделено желтым):

```
sudo nano /opt/dist/docker-compose.yml
```

Пример конфигурации файла *dist/docker-compose.yml*:

```
---
```

```
version: "3.7"
```

```
volumes:
```

```
  pgdata:
```

```
  pgdatansi:
```

```
  init:
```

```
x-op-restart-policy: &restart_policy
```

```
  restart: unless-stopped
```

```
services:
```

```
  dbkc:
```

```
    image: postgres:13
```

```
    <<: *restart_policy
```

```
    stop_grace_period: "3s"
```

```
    volumes:
```

```
      - ./pgdata:/var/lib/postgresql/data
```

```
    environment:
```

```
      POSTGRES_PASSWORD: p4ssw11rd
```

```
      POSTGRES_DB: kc
```

Изм.	Лист	№ докум.	Подп.	Дата

*healthcheck:*

*test: ["CMD-SHELL", "pg\_isready -U postgres"]*

*interval: 10s*

*timeout: 5s*

*retries: 5*

*dbnsi:*

*image: postgres:13*

*<<: \*restart\_policy*

*stop\_grace\_period: "3s"*

*volumes:*

*- ./pgdatansi:/var/lib/postgresql/data*

*- ./init:/docker-entrypoint-initdb.d*

*healthcheck:*

*test: ["CMD-SHELL", "pg\_isready -U postgres"]*

*interval: 10s*

*timeout: 5s*

*retries: 5*

*env\_file: .envnsi*

*kcnsi:*

*image: quay.io/keycloak/keycloak:21.0.0*

*<<: \*restart\_policy*

*command: start-dev*

*environment:*

*KC\_HOSTNAME\_URL: https://ваше\_доменное\_имя/kc*

*KC\_HTTP\_RELATIVE\_PATH: /kc*

*KC\_PROXY: edge*

*KEYCLOAK\_ADMIN: admin*

*KEYCLOAK\_ADMIN\_PASSWORD: 88werwer1*

*KC\_DB: postgres*

*KC\_DB\_URL: "jdbc:postgresql://dbkc:5432/kc"*

*KC\_DB\_USERNAME: postgres*

*KC\_DB\_PASSWORD: p4ssw11rd*

*ports:*

Изм.	Лист	№ докум.	Подп.	Дата



*ports:*

- 8082:8088

*depends\_on:*

- *dbnsi*

- *kcnsi*

*env\_file: .envnsi*

*extra\_hosts:*

- "*ваше\_доменное\_имя:внутренний\_адрес\_сервера\_в\_локальной\_сети*"

### пример записи- "*nsi.zen-it.ru:192.168.0.10*"

\* hairpin NAT – когда устройство, находящееся за NAT'ом, обращается к какому-то порту внешнего IP-адреса этого NAT'а, и шлюз (если на нём настроен форвардинг этого порта) через который проходит запрос, перенаправляет запрос обратно во внутреннюю сеть.

#### 3.1.5.5 Настройка haproxy.cfg

Полностью замените содержимое конфигурационного файла */etc/haproxy/haproxy.cfg* внеся в конфигурационный файл необходимые изменения (выделено желтым). Можно использовать для замены следующую команду:

```
cat > /etc/haproxy/haproxy.cfg << EOF
```

**текст**

**конфигурационного**

**файла**

*EOF*

Пример конфигурации файла *haproxy.cfg*:

*global*

*log /dev/log local0*

*log /dev/log local1 notice*

*stats socket /var/lib/haproxy/stats level admin*

*chroot /var/lib/haproxy*

*user haproxy*

*group haproxy*

*daemon*

*defaults*

Изм.	Лист	№ докум.	Подп.	Дата	Лист

```

log global
maxconn 20000
mode http
balance roundrobin
option splice-auto
option dontlognull
option forwardfor
option http-server-close
retries 3
option redispatch
timeout client 600s
timeout http-keep-alive 2s
timeout http-request 15s
timeout connect 15s
timeout server 600s
timeout queue 5s
timeout check 5s
stats enable
stats refresh 5s
stats hide-version
stats uri /haproxy-status
monitor-uri /prod-monitor

```

```

frontend http-in

```

```

mode http
option httplog
capture request header Host len 50

```

```

bind *:80

```

```

bind *:443 ssl crt /etc/haproxy/certs/ alpn http/1.1,h2

```

```

acl letsencrypt-acl path_beg /.well-known/acme-challenge/
http-request add-header X-Forwarded-Proto https if { ssl_fc }
redirect scheme https code 301 if !{ ssl_fc } !letsencrypt-acl
use_backend letsencrypt-backend if letsencrypt-acl

```

Изм.	Лист	№ докум.	Подп.	Дата	Изм.	Лист	№ докум.	Подп.	Дата



`sudo docker-compose up -d`

Пример корректного ответа:

```
root@debl2-nsi:/home# cd /opt/dist
root@debl2-nsi:/opt/dist# sudo docker-compose up -d
Creating network "dist_default" with the default driver
Creating volume "dist_pgdata" with default driver
Creating volume "dist_pgdatansi" with default driver
Creating volume "dist_init" with default driver
Pulling dbkc (postgres:13)...
13: Pulling from library/postgres
faef57eae888: Pull complete
a33c10a72186: Pull complete
d662a43776d2: Pull complete
a3ba86413420: Pull complete
a627f37e9916: Pull complete
424bade69494: Pull complete
dd8d4fcd466b: Pull complete
03d0efeea592: Pull complete
89495f306b9b: Pull complete
b9ef09cal802: Pull complete
a283f3964686: Pull complete
2b43fcabe8a8: Pull complete
b8dec84b144c: Pull complete
Digest: sha256:0f18de936266e03891e186db16e530e0e4365ef5fb300d4bb27318538b80604
Status: Downloaded newer image for postgres:13
Pulling kncsi (quay.io/keycloak/keycloak:21.0.0)...
21.0.0: Pulling from keycloak/keycloak
ae9e8620cc67: Pull complete
015ac7802109: Pull complete
474624fdc80a: Pull complete
b87ddc5dl856: Pull complete
Digest: sha256:2ff491b346361b84f5923285fa80bd817db211c24c3188bf68d52554a8fce5bf
Status: Downloaded newer image for quay.io/keycloak/keycloak:21.0.0
Creating dist_dbnsi_1 ... done
Creating dist_dbkc_1 ... done
Creating dist_kncsi_1 ... done
Creating dist_nsiback_1 ... done
Creating dist_nsifront_1 ... done
root@debl2-nsi:/opt/dist#
```

- Проверка состояния контейнеров:

`sudo docker-compose ps`

Пример корректного ответа:

```
root@debl2-nsi:/opt/dist# docker compose ps
NAME                IMAGE                COMMAND                SERVICE                CREATED                STATUS
dist-dbkc-1         postgres:13         "docker-entrypoint.s..." dbkc                   4 hours ago          Up 4 hours (healthy)
dist-dbnsi-1        postgres:13         "docker-entrypoint.s..." dbnsi                  4 hours ago          Up 4 hours (healthy)
dist-kncsi-1        quay.io/keycloak/keycloak:21.0.0 "/opt/keycloak/bin/k..." kncsi                  4 hours ago          Up 4 hours
dist-nsiback-1      back:latest         "java -Dfile.encoding..." nsiback                4 hours ago          Up 4 hours
dist-nsifront-1    front:latest        "/docker-entrypoint..." nsifront               4 hours ago          Up 4 hours
```

Должно отображаться в графе Status состояние UP для всех контейнеров.

**Если при проверке вы видите ошибку для контейнера dist-kncsi-1, когда статус контейнера показывает restarting, см. пункт 5.1 данной инструкции.**

- Перезапустить haproxy:  
`sudo systemctl restart haproxy`
- Проверить состояние haproxy:  
`sudo systemctl status haproxy`

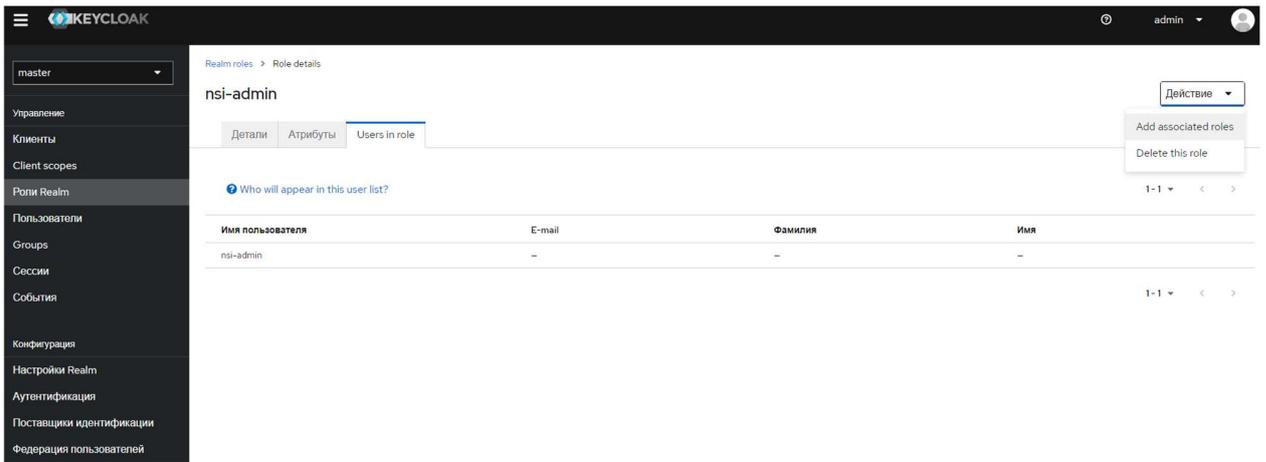
Пример корректного ответа:

Изм.	Лист	№ докум.	Подп.	Дата

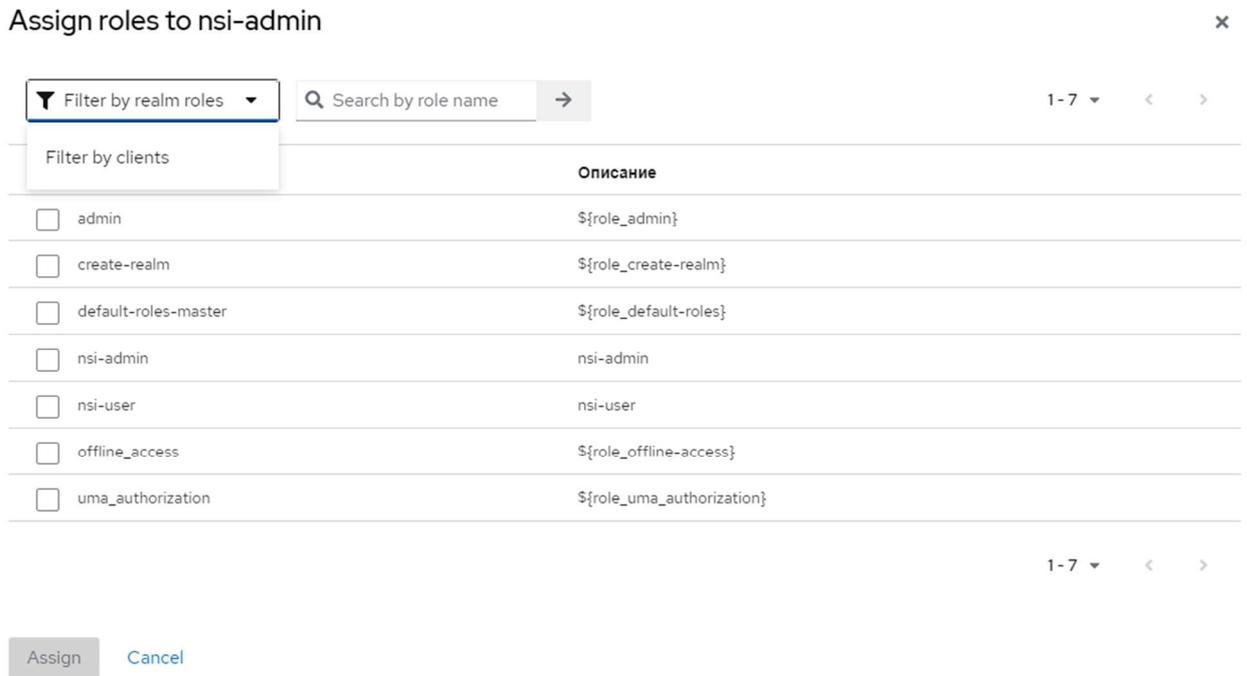




– Находясь в свойствах роли nsi-admin открыть вкладку “Users in role” -> Выбрать “Действие” -> Выбрать “Add associates roles”

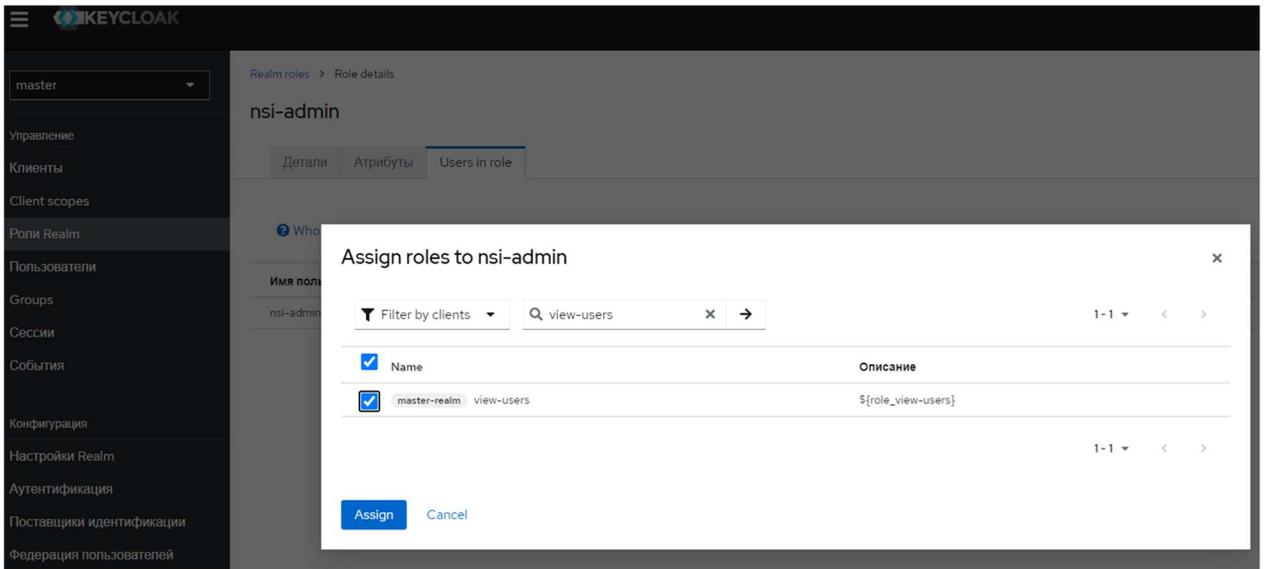


– На странице “Assign roles to nsi-admin” в поле фильтра выбрать “Filter by clients”



– В поле поиска набрать “view-users”, нажать стрелку вправо для поиска и выбрать найденное значение роли view-users

Инд. № подл.	Подп. и дата
Взамен инв. №	Инд. № дубл.
Подп. и дата	
Инд. № подл.	



– Нажать “Assign”

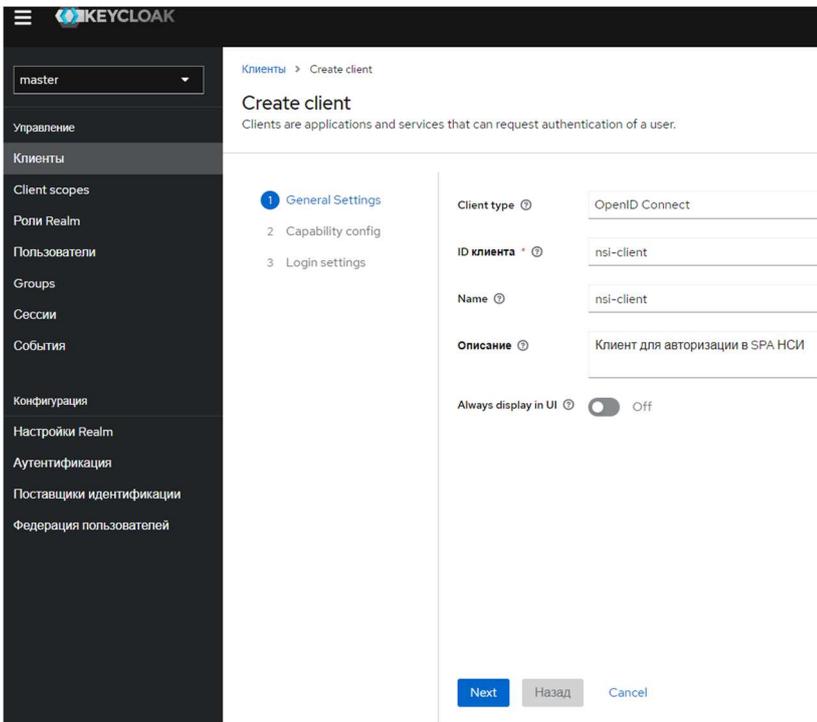
- Создать клиента nsi-client

– Перейти на пункт меню “Клиенты” -> Выбрать “Create Client” -> Заполнить поля:

Поле “ID клиента” = “nsi-client”;

Поле “Name” = “nsi-client”;

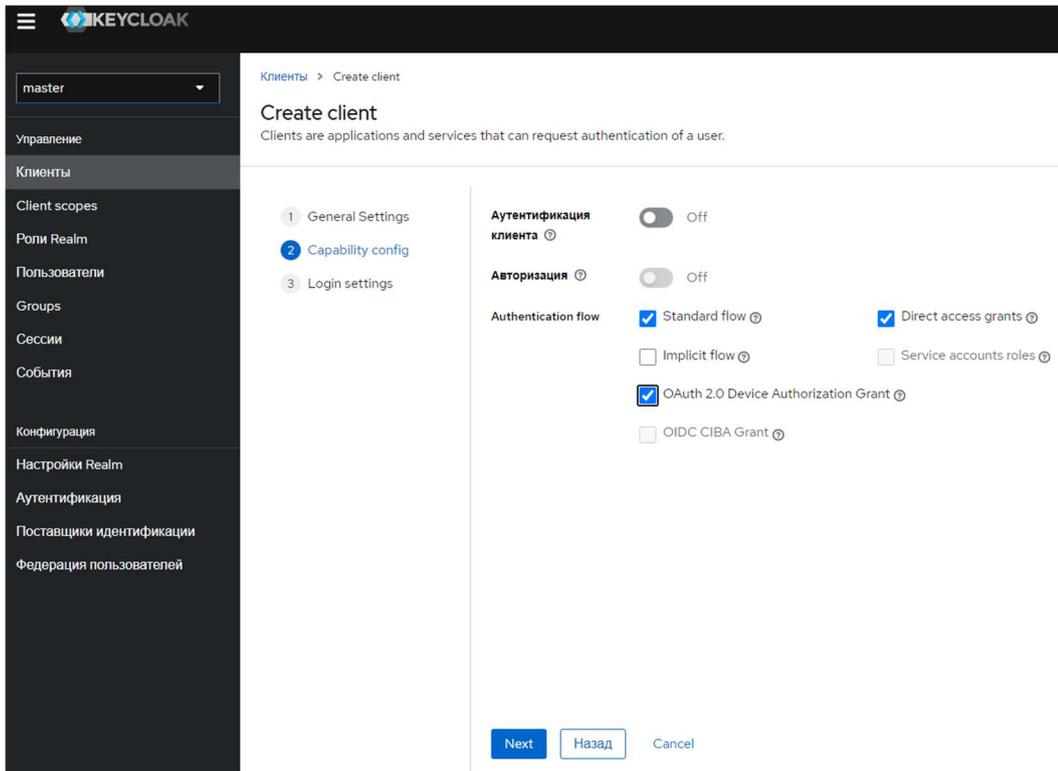
Поле “Описание” = “Клиент для авторизации в SPA НСИ”.



– Нажать “Next”

– Выбрать поля: Standard flow, Direct access grants, OAuth 2.0 Device Authorization Grant, остальное должно быть выключено.

Изм.	Лист	№ докум.	Подп.	Дата
Инд. № подл.	Подп. и дата	Взамен инв. №	Инд. № дубл.	Подп. и дата



– Нажать “Next”

– Нажать “Сохранить”

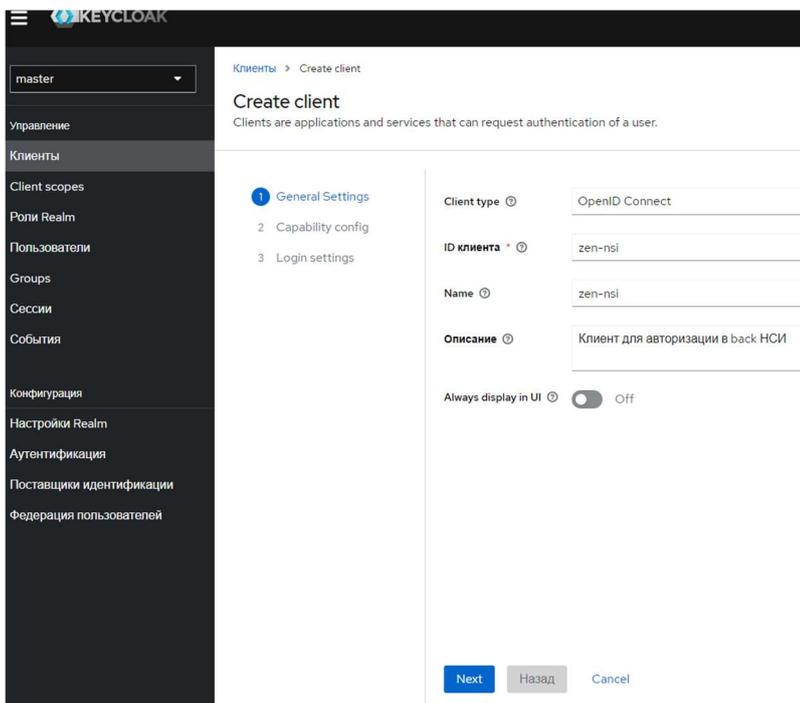
- Создать клиента zen-nsi

– Перейти на пункт меню “Клиенты” -> Выбрать “Create Client” -> Заполнить поля:

Поле “ID клиента” = “zen-nsi”;

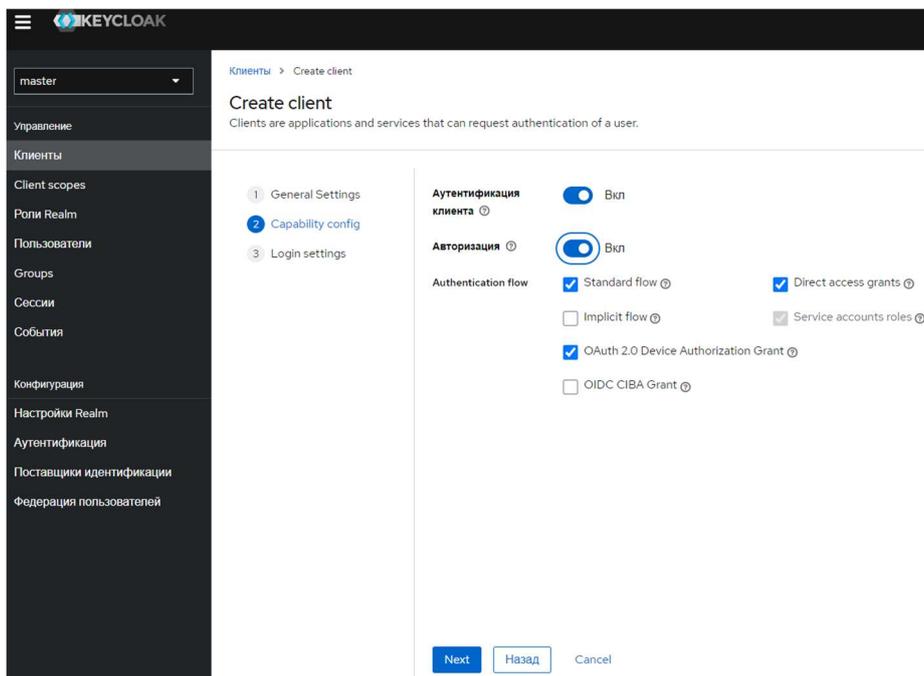
Поле “Name” = “zen-nsi”;

Поле “Описание” = “Клиент для авторизации в back НСИ”.



Изм.	Лист	№ докум.	Подп.	Дата	Инва. № подл.	Взамен инв. №	Инва. № дубл.	Подп. и дата
------	------	----------	-------	------	---------------	---------------	---------------	--------------

- Нажать “Next”
- Выбрать поля: Standard flow, Direct access grants, OAuth 2.0 Device Authorization Grant, Авторизация, Аутентификация клиента. остальное должно быть выключено.



– Нажать “Next”

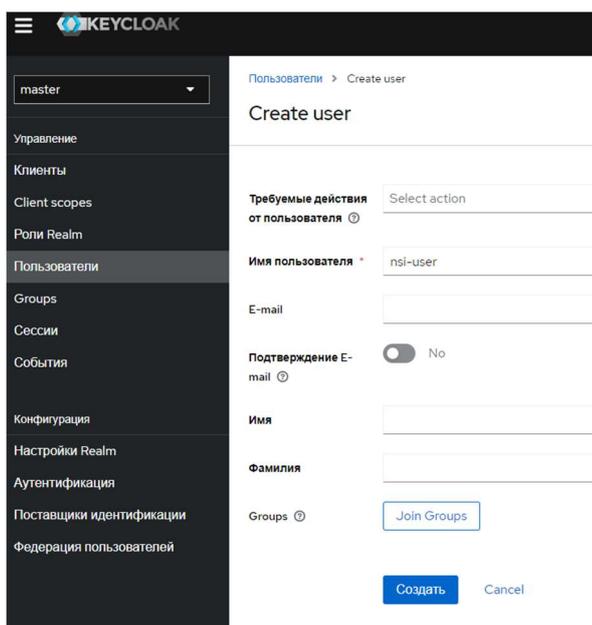
– Нажать “Сохранить”

- Создать пользователя ЗЕН-НСИ с правами пользователя

– Перейти на пункт меню “Пользователи” -> Выбрать “Добавить пользователя” ->

Заполнить поля:

Поле “Имя пользователя” = любое, например “nsi-user”

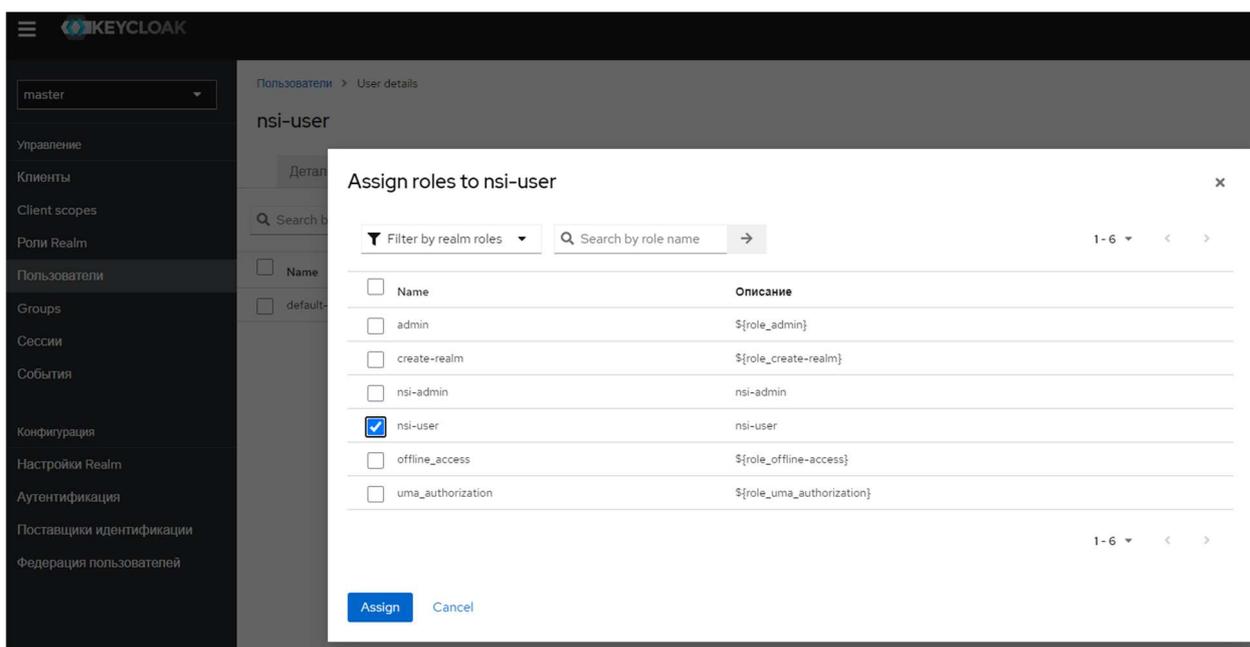


– Нажать “Создать”

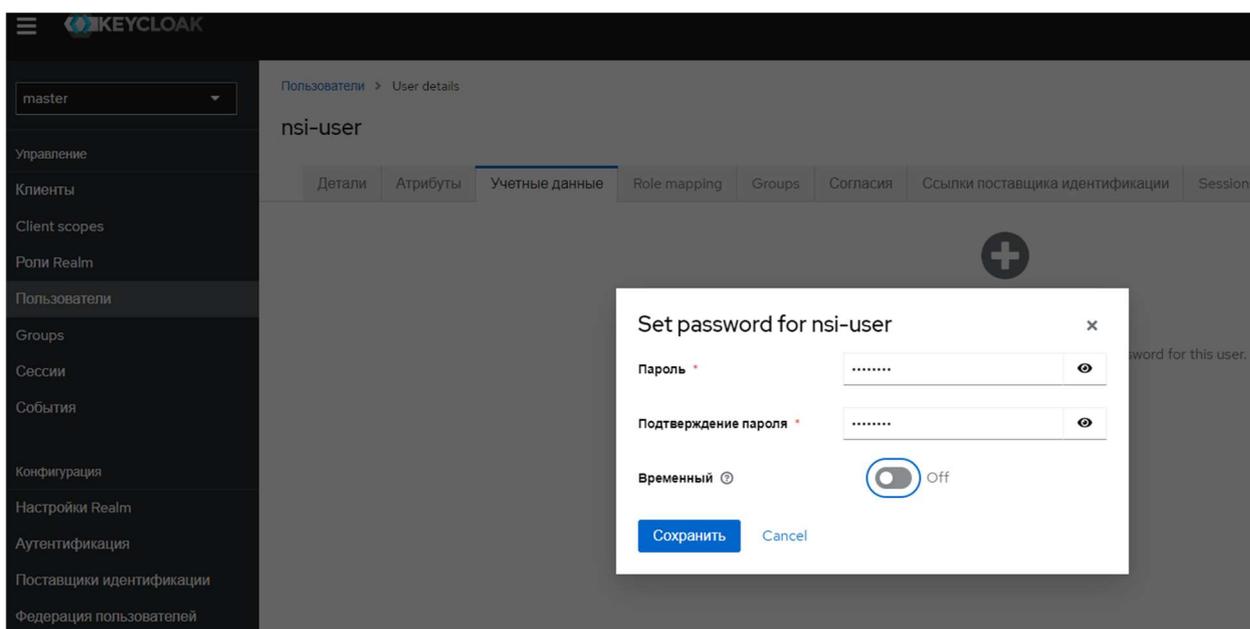
Имя № подл.	Подп. и дата
Взамен инв. №	Имя № дубл.
Подп. и дата	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					33

- В учетной записи пользователя nsi-user открыть вкладку “Role mapping”
- Нажать “Assign role”, выбрать роль nsi-user

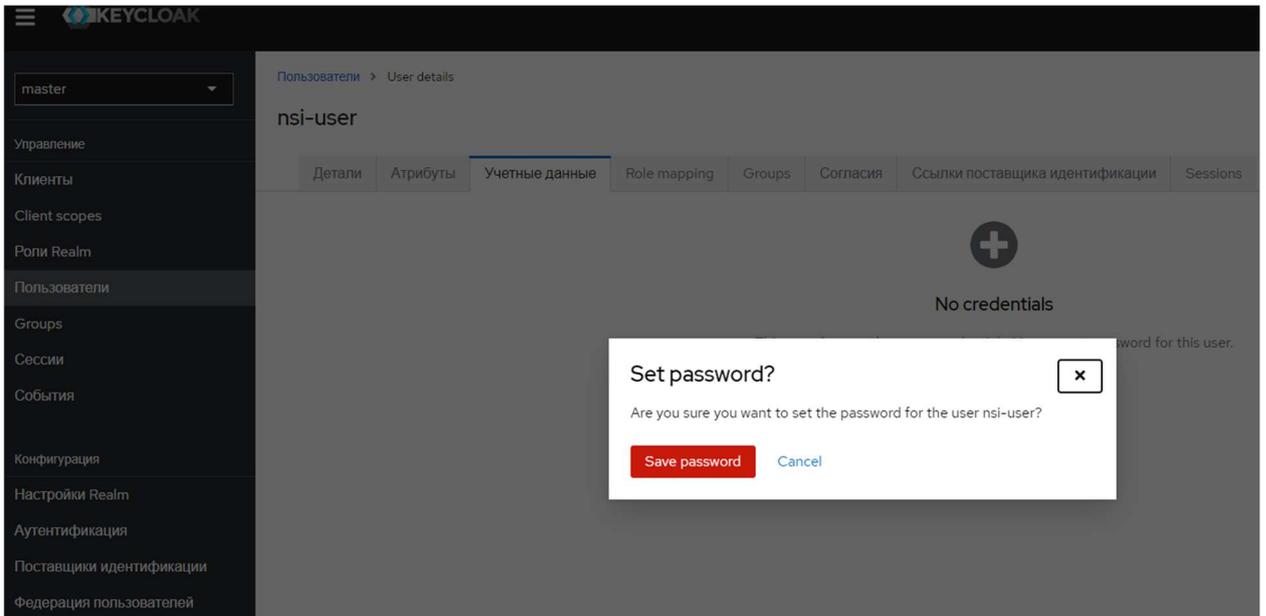


- Нажать “Assign”
- Перейти на вкладку “Учетные данные”
- Нажать “Set password”, придумать и ввести пароль для учетной записи



- Нажать “Сохранить”, будет выведен запрос об установке пароля для данной учетной записи

Изм.	Лист	№ докум.	Подп.	Дата
Инва № подл.	Подп. и дата	Инва № дубл.	Взамен инв. №	Подп. и дата



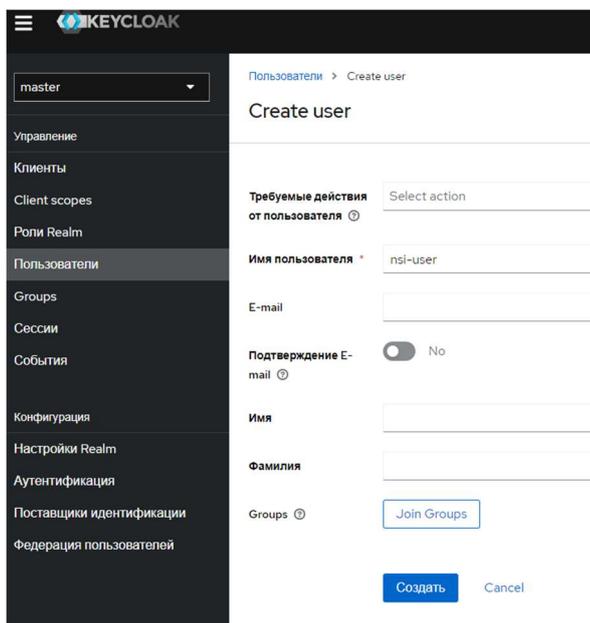
– Нажать “Save password”

- Создать пользователя ЗЕН-НСИ с правами администратора

– Перейти на пункт меню “Пользователи” -> Выбрать “Добавить пользователя” ->

Заполнить поля:

Поле “Имя пользователя” = любое, например “nsi-admin”



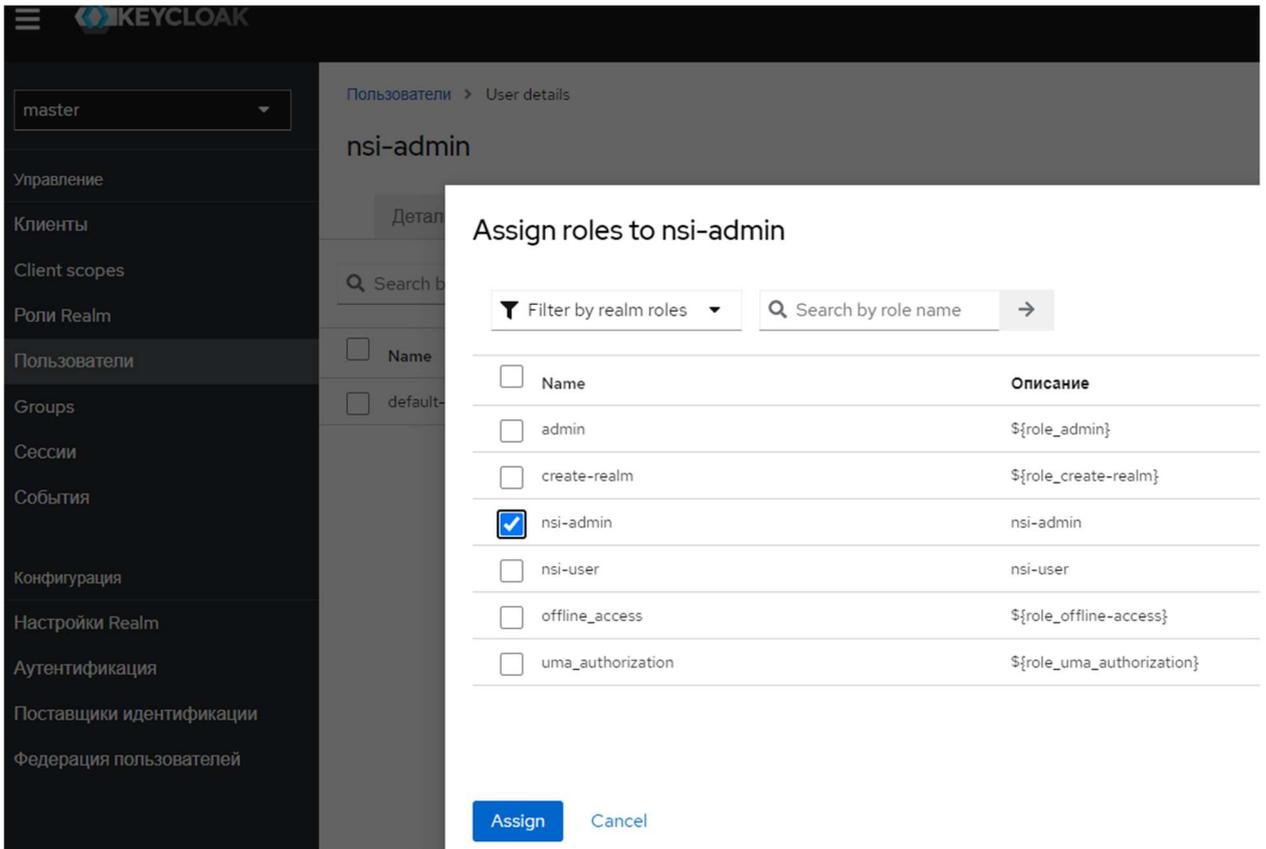
– Нажать “Создать”

– В учетной записи пользователя nsi-user открыть вкладку “Role mapping”

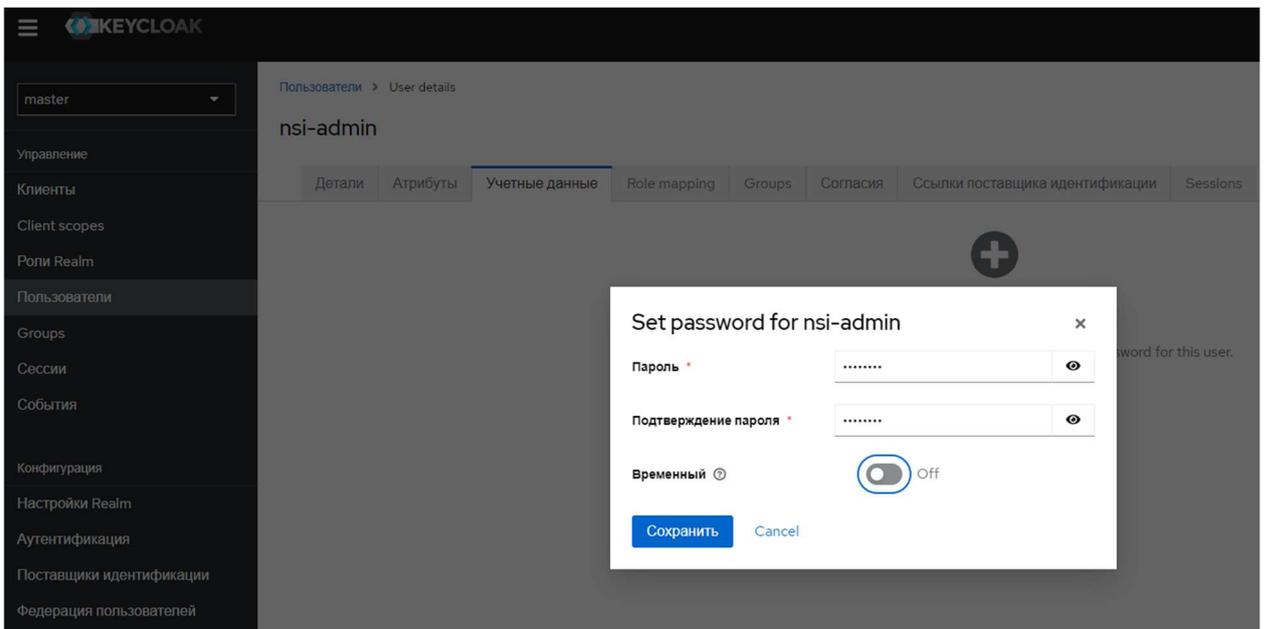
– Нажать “Assign role”, выбрать роль nsi-admin

Имя № подл.	Подп. и дата
Взамен инв. №	Подп. и дата
Инва № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					35

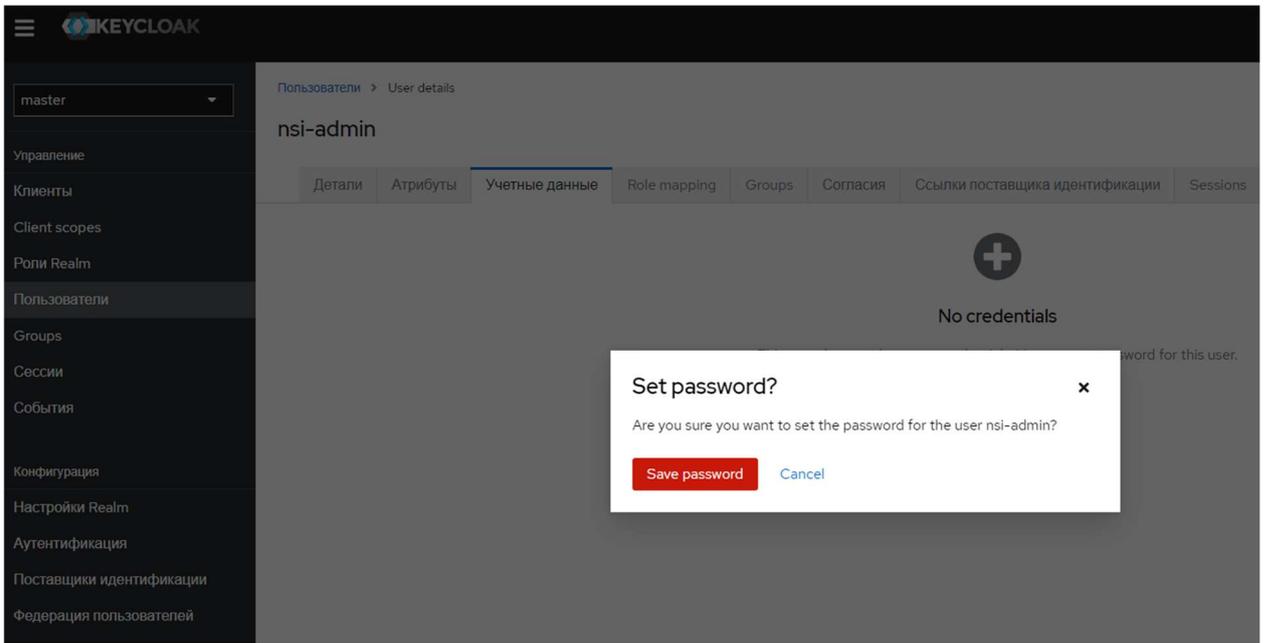


- Нажать “Assign”
- Перейти на вкладку “Учетные данные”
- Нажать “Set password”, придумать и ввести пароль для учетной записи



- Нажать “Сохранить”, будет выведен запрос об установке пароля для данной учетной записи

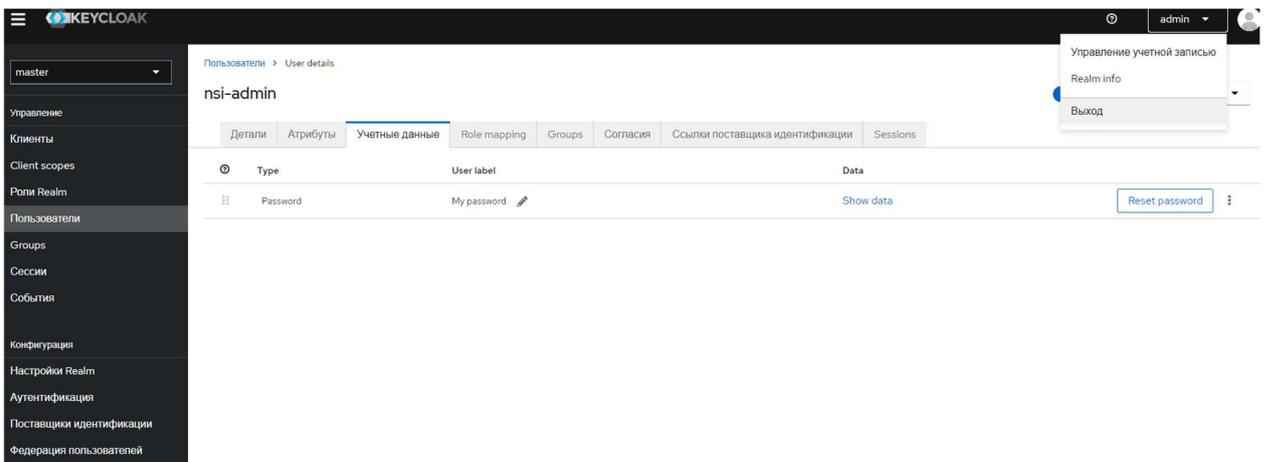
Изм.	Лист	№ докум.	Подп.	Дата	Изм.	Лист	№ докум.	Подп.	Дата



– Нажать “Save password”

- Выйти из панели администрирования KeyCloak

– Нажать на имя учетной записи “admin” в верхней панели интерфейса, будет выведено меню выбора действий.



– Нажать “Выход”

Изн. № подл.	Подп. и дата
Взамен инв. №	Подп. и дата
Изн. № дубл.	Подп. и дата

Изн.	Лист	№ докум.	Подп.	Дата	Лист
					37

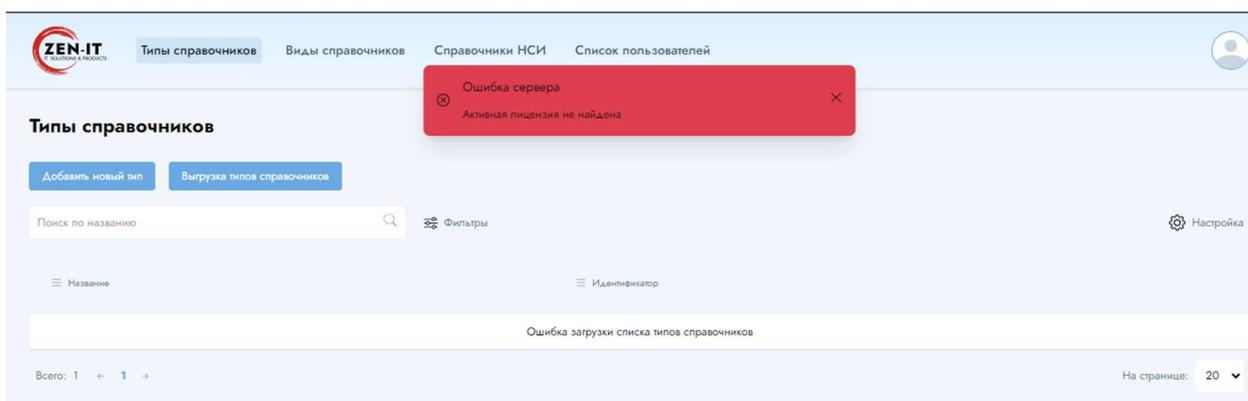
## 4. ИСПОЛЬЗОВАНИЕ ЗЕН-НСИ

Для использования ЗЕН-НСИ следует получить лицензионный ключ на программное обеспечение. Лицензионный ключ выдается службой технической поддержки, куда можно обратиться через электронную почту [support@zen-it.ru](mailto:support@zen-it.ru) или по телефону +7 (929) 975-77-99.

### 4.1 Активация лицензии

Перед началом работы с ЗЕН-НСИ необходимо активировать лицензию для использования программного обеспечения. При отсутствии лицензии показывается информационное сообщение об ошибке “Активная лицензия не найдена” при переходе по вкладкам меню после авторизации.

Пример сообщения об отсутствии лицензии:



#### 4.1.1 Сетевая активация лицензии

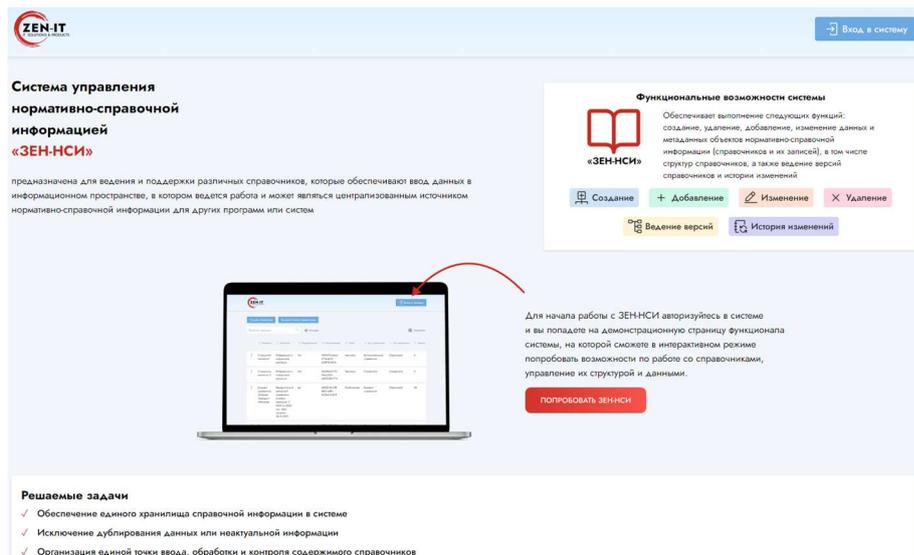
##### 4.1.1.1 Сетевая активация лицензии через графический интерфейс

Для сетевой (online) активации лицензии с использованием Интернет соединения, через графический интерфейс, необходимо:

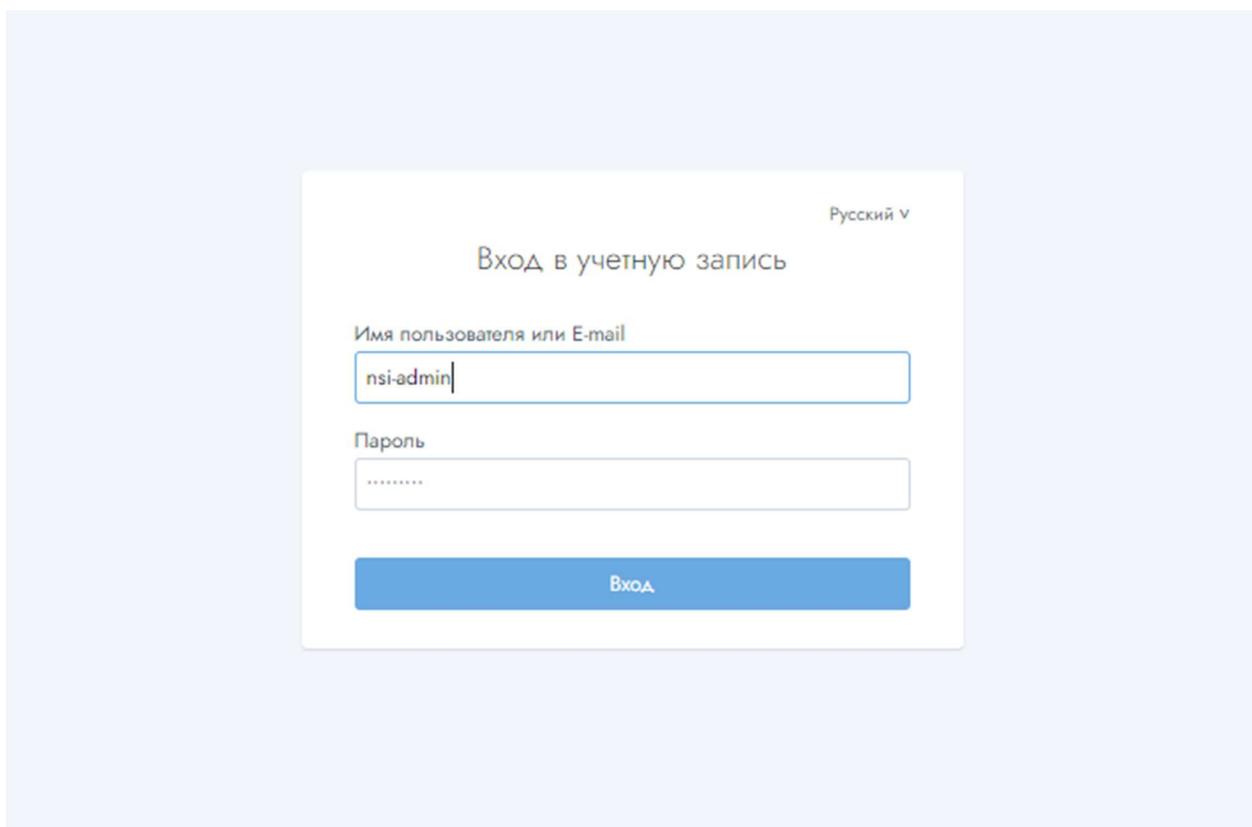
- Для работы с ЗЕН-НСИ в браузере открыть страницу с адресом <https://ваше-доменное-имя> и нажать кнопку “Вход в систему” на странице с описанием ЗЕН-НСИ.

Инва. № подл.	Подп. и дата	Взамен инв. №	Инва. № дубл.	Подп. и дата

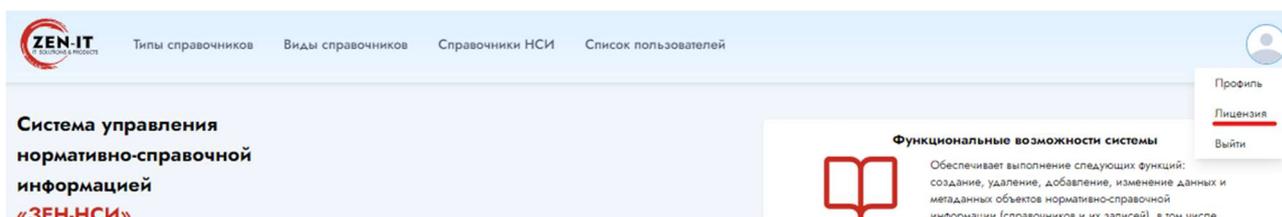
Изм.	Лист	№ докум.	Подп.	Дата	Лист
					38



- Авторизоваться под пользователем `nsi-admin` используя пароль, установленный ранее при создании пользователей в авторизационном сервисе Keycloak



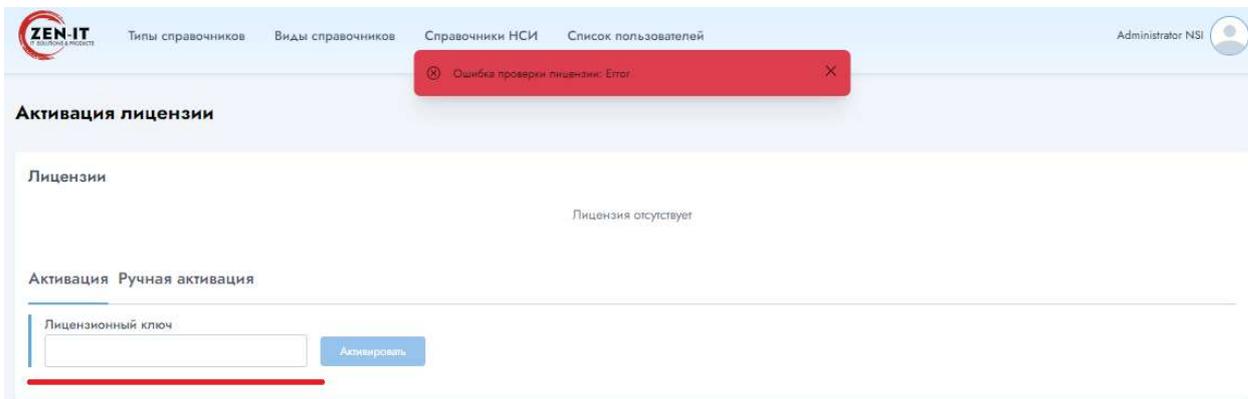
- Под пользователем с правами Администратора ЗЕН-НСИ перейти на страницу «Активация лицензии», для этого нужно нажать на иконку пользователя и выбрать подменю «Лицензия».



Инва № подл.	Подп. и дата	Взамен инв. №	Инва № дубл.	Подп. и дата

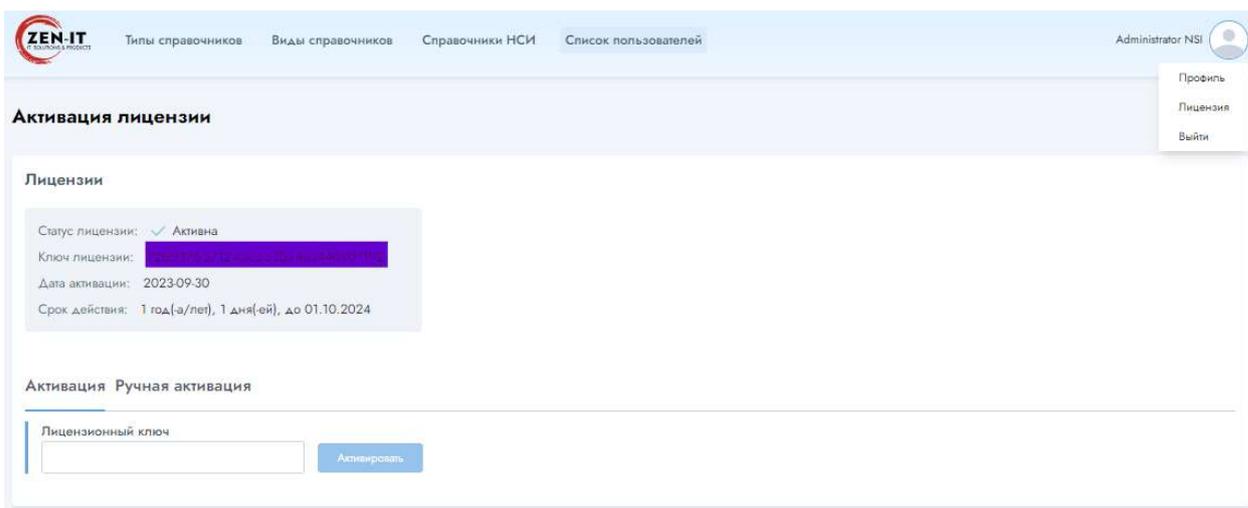
Изм.	Лист	№ докум.	Подп.	Дата	Лист
					39

- На странице «Активация лицензии» в поле «Лицензионный ключ» ввести полученный лицензионный ключ на программное обеспечение и нажать кнопку «Активировать».



- После онлайн проверки лицензии на сервере лицензирования ООО «ЗЕН-АЙТИ» программное обеспечение будет активировано.

Пример сообщения об активации лицензии:



#### 4.1.1.1 Сетевая активация лицензии через программный интерфейс

Для сетевой (online) активации лицензии с использованием Интернет соединения, через API (Application Programming Interface, что значит - программный интерфейс приложения), необходимо:

- В браузере зайти по адресу <https://ваше-доменное-имя/nsi/swagger-ui/index.html>  
Откроется веб-страница с методами API ЗЕН-НСИ

Изн. № подл.	Подп. и дата	Взамен инв. №	Изн. № дубл.	Подп. и дата

Изн.	Лист	№ докум.	Подп.	Дата	Лист
					40

## Сервис НСИ 0.3.0-b11 OAS3

[/insiv3/api-docs](#)

Предоставляет API для управления справочниками

Servers

Authorize

- Виды справочников** ▼  
Предоставляет API для управления видами справочников
- Справочники НСИ** ▼  
Предоставляет API для управления справочниками
- Типы справочников** ▼  
Предоставляет API для управления типами справочников
- Управление статусами справочников** ▼  
Предоставляет API для управления статусами справочников
- Экспорт / импорт данных справочников** ▼  
Предоставляет API для экспорта/импорта данных справочников
- Атрибуты справочника** ▼  
Предоставляет API для управления атрибутами справочников
- Создание / управление записями справочников** ▼  
Предоставляет API для управления записями справочников
- Управление версиями справочников** ▼  
Предоставляет API для управления версиями справочников
- История изменений данных сервиса НСИ** ▼  
Предоставляет API для просмотра истории изменений справочников

**license-management-controller** ▲

- POST [/license/get-activation-request](#) Офлайн активация - этап 1 - получение запроса на активацию ▼
- POST [/license/apply-activation-code](#) Офлайн активация - этап 2 - применение кода активации ▼
- POST [/license/activate](#) Активация лицензии ▼
- GET [/license](#) Получение информации о лицензии ▼

- [Перейти к методу API POST /license/activate](#)

**license-management-controller** ▲

- POST [/license/get-activation-request](#) Офлайн активация - этап 1 - получение запроса на активацию ▼
- POST [/license/apply-activation-code](#) Офлайн активация - этап 2 - применение кода активации ▼
- POST [/license/activate](#) Активация лицензии ▼
- GET [/license](#) Получение информации о лицензии ▼

- [Раскрыть метод и нажать кнопку “Try it out”](#)

POST [/license/activate](#) Активация лицензии ▲

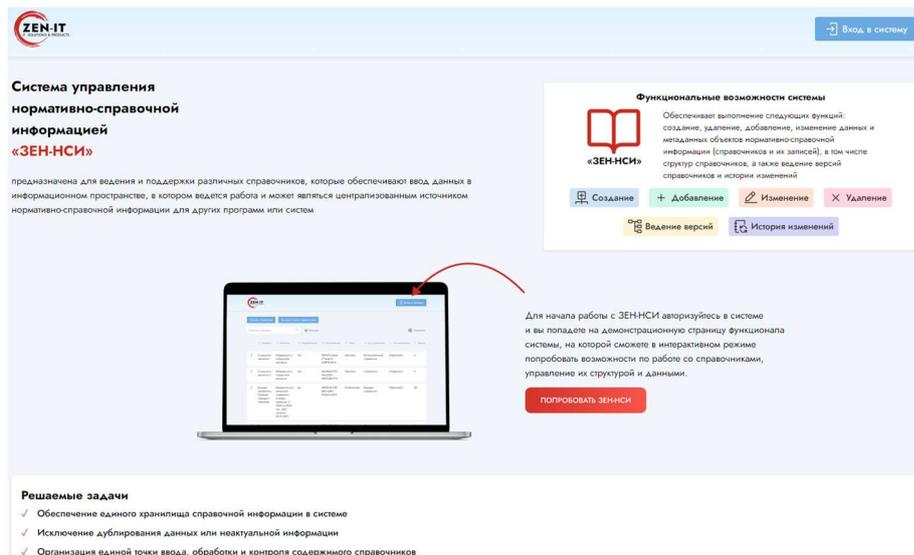
Try it out

**Parameters**

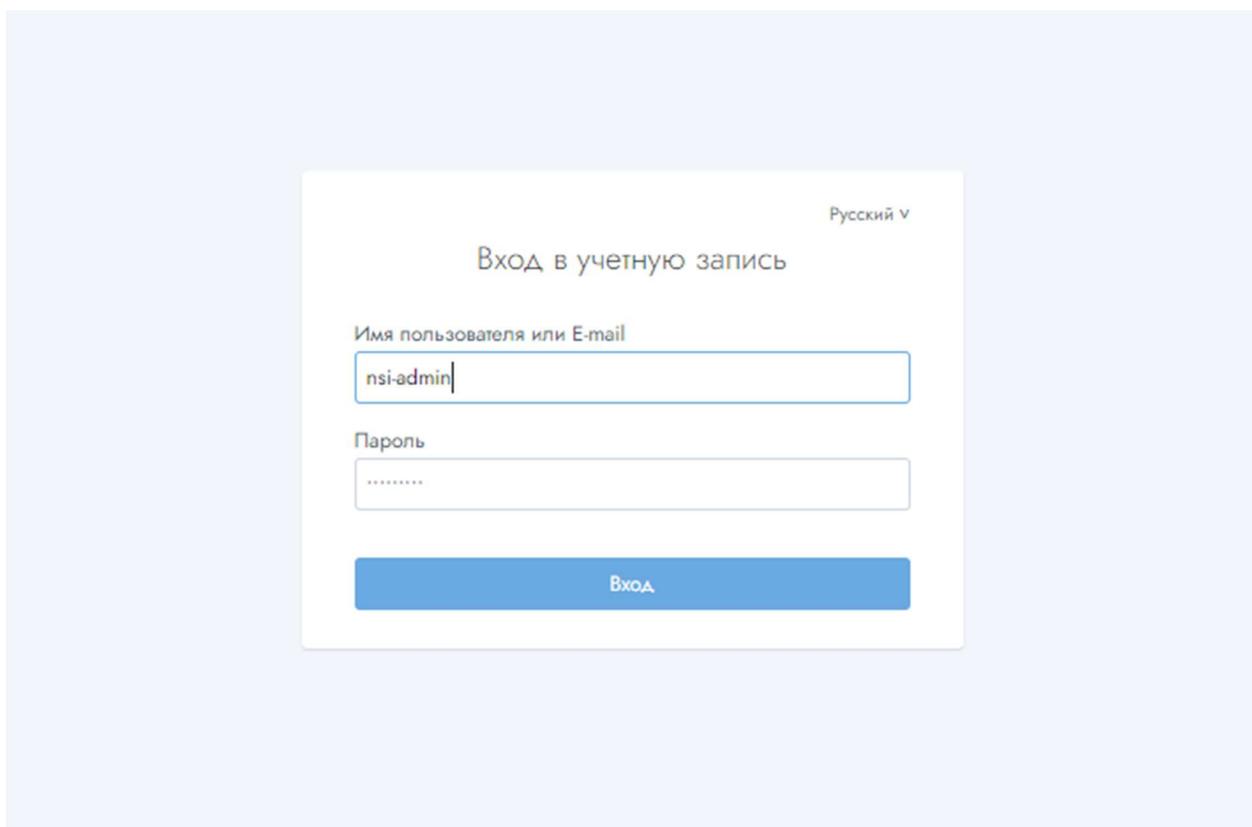
Name	Description
<b>licenseKey</b> <span style="color: red; font-size: small;">* required</span>	Лицензионный ключ
string (query)	

Изм.	Лист	№ докум.	Подп.	Дата
Изнв № подл.	Взамен инв. №	Изнв № дубл.	Подп. и дата	Подп. и дата

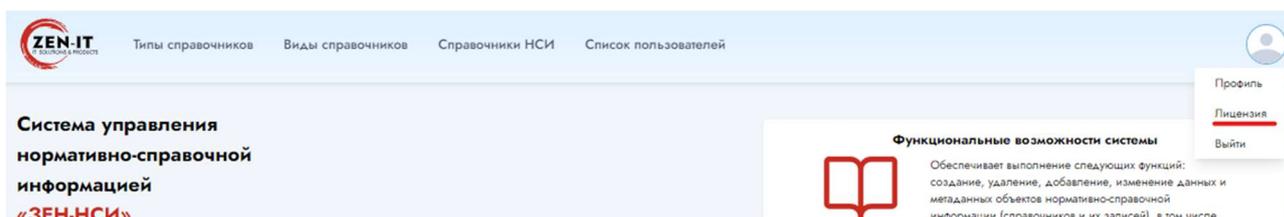




- Авторизоваться под пользователем nsi-admin используя пароль, установленный ранее при создании пользователей в авторизационном сервисе Keycloak



- Под пользователем с правами Администратора ZEN-НСИ перейти на страницу «Активация лицензии», для этого нужно нажать на иконку пользователя и выбрать подменю «Лицензия».



Инва № подл.	Подп. и дата	Взамен инв. №	Инва № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					43





## Сервис НСИ 0.3.0-b11 OAS3

[/insiv3/api-docs](#)

Предоставляет API для управления справочниками

Servers

Authorize

- Виды справочников** ▼  
Предоставляет API для управления видами справочников
- Справочники НСИ** ▼  
Предоставляет API для управления справочниками
- Типы справочников** ▼  
Предоставляет API для управления типами справочников
- Управление статусами справочников** ▼  
Предоставляет API для управления статусами справочников
- Экспорт / импорт данных справочников** ▼  
Предоставляет API для экспорта/импорта данных справочников
- Атрибуты справочника** ▼  
Предоставляет API для управления атрибутами справочников
- Создание / управление записями справочников** ▼  
Предоставляет API для управления записями справочников
- Управление версиями справочников** ▼  
Предоставляет API для управления версиями справочников
- История изменений данных сервиса НСИ** ▼  
Предоставляет API для просмотра истории изменений справочников

**license-management-controller** ▲

- POST [/license/get-activation-request](#) Офлайн активация - этап 1 - получение запроса на активацию ▼
- POST [/license/apply-activation-code](#) Офлайн активация - этап 2 - применение кода активации ▼
- POST [/license/activate](#) Активация лицензии ▼
- GET [/license](#) Получение информации о лицензии ▼

- Перейти к методу API POST [/license/get-activation-request](#)  
Офлайн активация - этап 1 - получение запроса на активацию

**license-management-controller** ▲

- POST [/license/get-activation-request](#) Офлайн активация - этап 1 - получение запроса на активацию ▼
- POST [/license/apply-activation-code](#) Офлайн активация - этап 2 - применение кода активации ▼
- POST [/license/activate](#) Активация лицензии ▼
- GET [/license](#) Получение информации о лицензии ▼

- Раскрыть метод и нажать кнопку “Try it out”

**license-management-controller** ▲

POST [/license/get-activation-request](#) Офлайн активация - этап 1 - получение запроса на активацию ▼

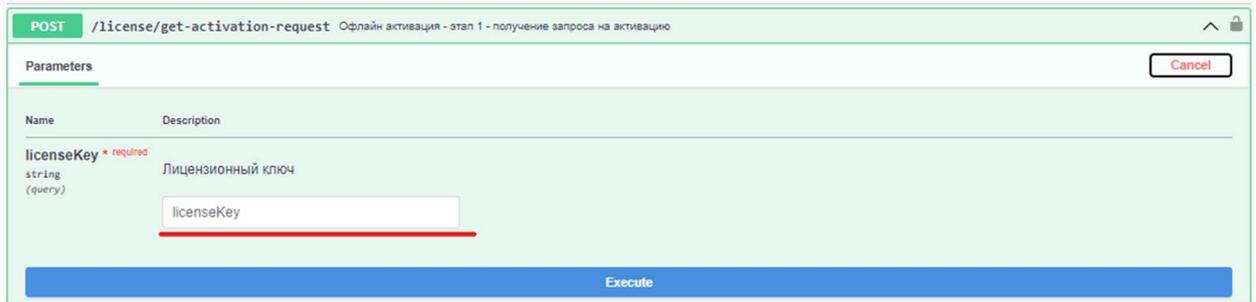
Try it out

**Parameters**

Name	Description
<b>licenseKey</b> * <small>required</small>	Лицензионный ключ
string (query)	
<input type="text" value="licenseKey"/>	

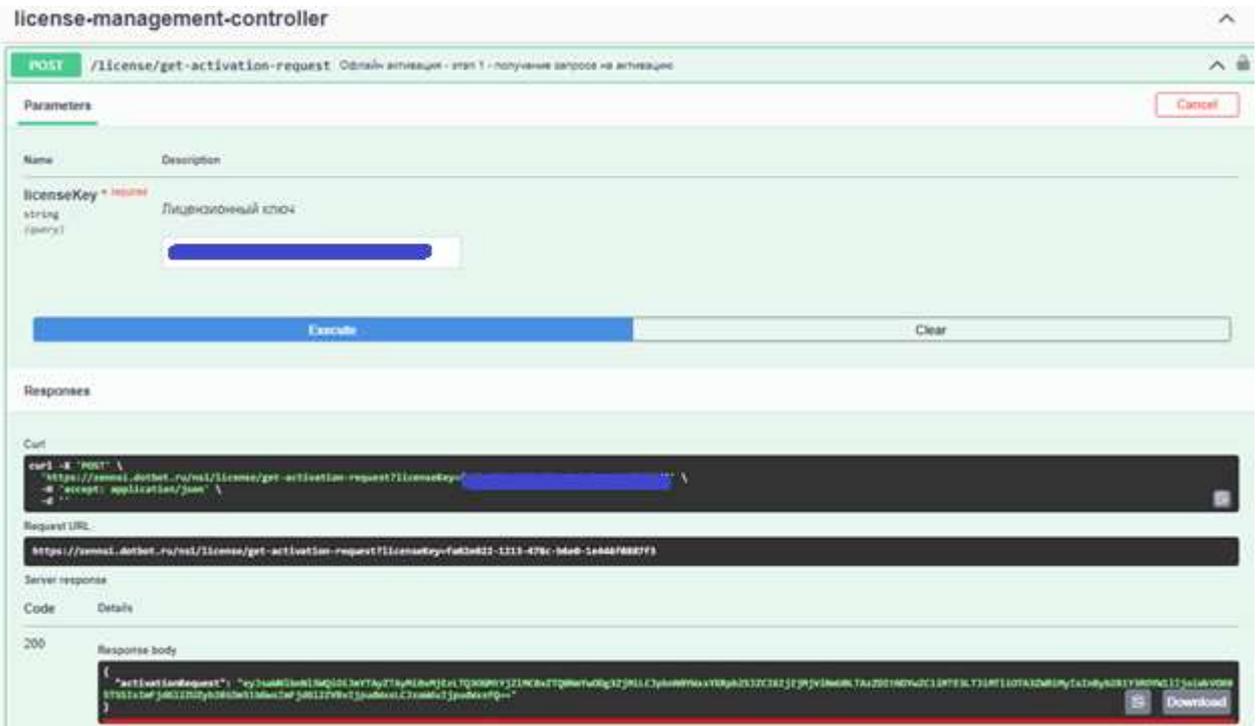
Изм.	Лист	№ докум.	Подп.	Дата
Изнв № подл.	Взамен инв. №	Изнв № дубл.	Подп. и дата	Подп. и дата

- Вставить в поле “Лицензионный ключ” полученный лицензионный ключ

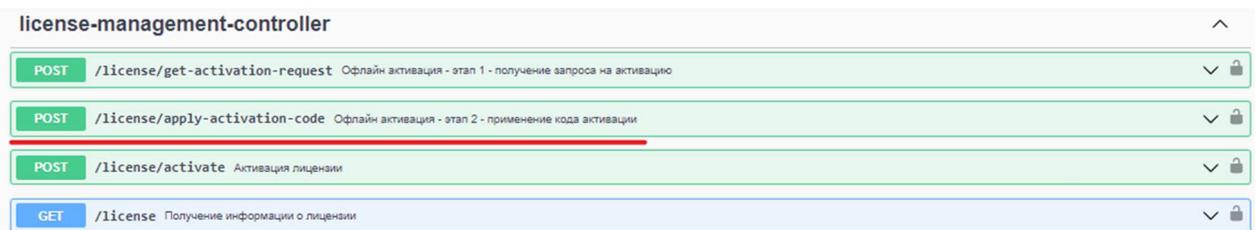


- Нажать кнопку “Execute”

В поле “Responses” будет отображен ответ с кодом 200, а в поле “Response body” будут отображены данные запроса активации для получения кода активации от службы технической поддержки.



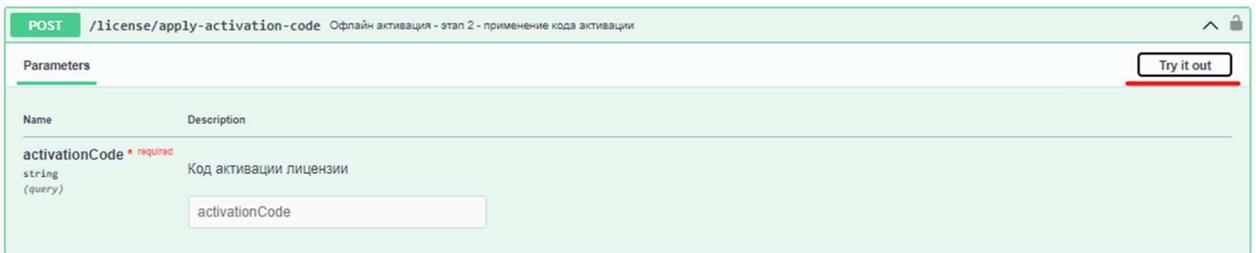
- Передать запрос активации в службу технической поддержки и получить код активации
- Перейти к методу API POST /license/apply-activation-code  
Офлайн активация - этап 2 - применение кода активации



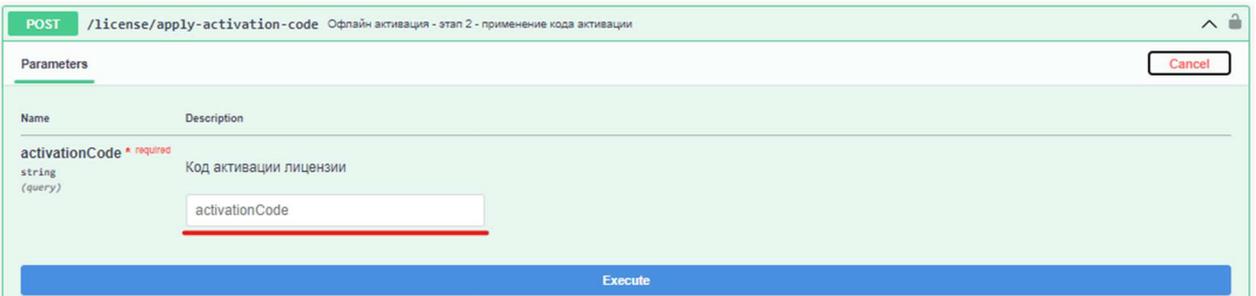
- Раскрыть метод и нажать кнопку “Try it out”

Изн. № подл.	Подп. и дата	Взамен инв. №	Изн. № дубл.	Подп. и дата

Изн. № подл.	Подп. и дата	Взамен инв. №	Изн. № дубл.	Подп. и дата
Изм.	Лист	№ докум.	Подп.	Дата



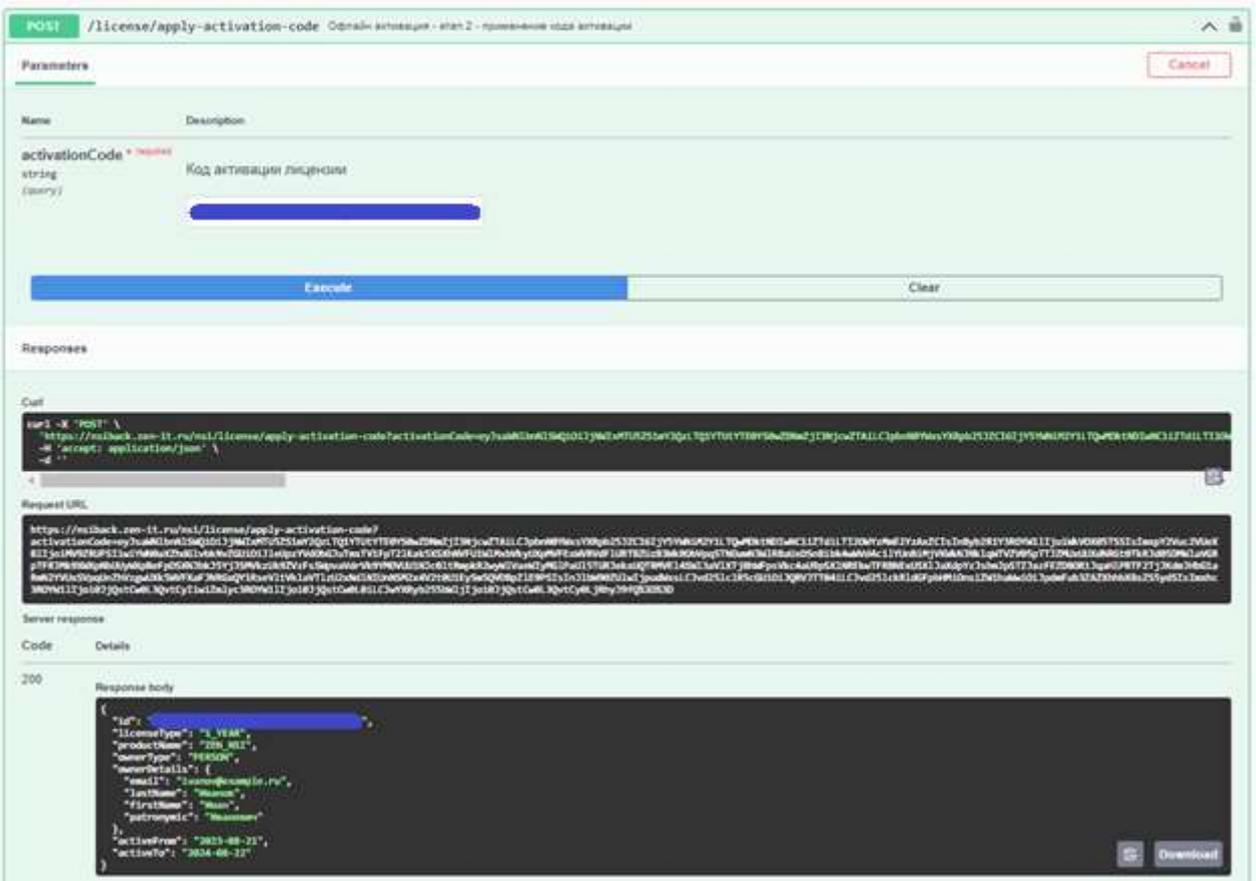
- Вставить в поле “Код активации лицензии” полученный от службы технической поддержки код активации



- Нажать кнопку “Execute”

При успешной активации в поле “Responses” будет отображен ответ с кодом 200, а в поле “Response body” будут отображены данные лицензии.

Пример ответа об успешной активации:

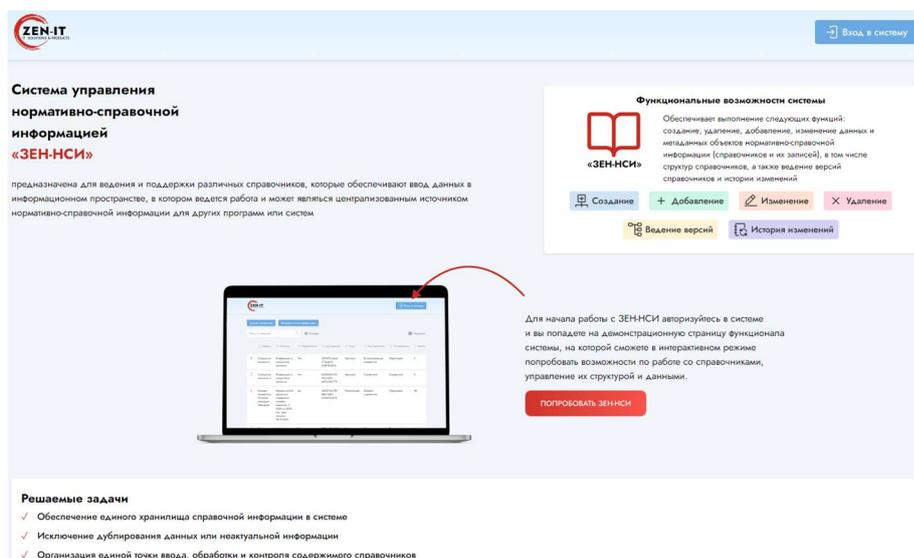


Изм. № подл.	Подп. и дата
Взамен инв. №	Подп. и дата
Изнв. № дубл.	Подп. и дата

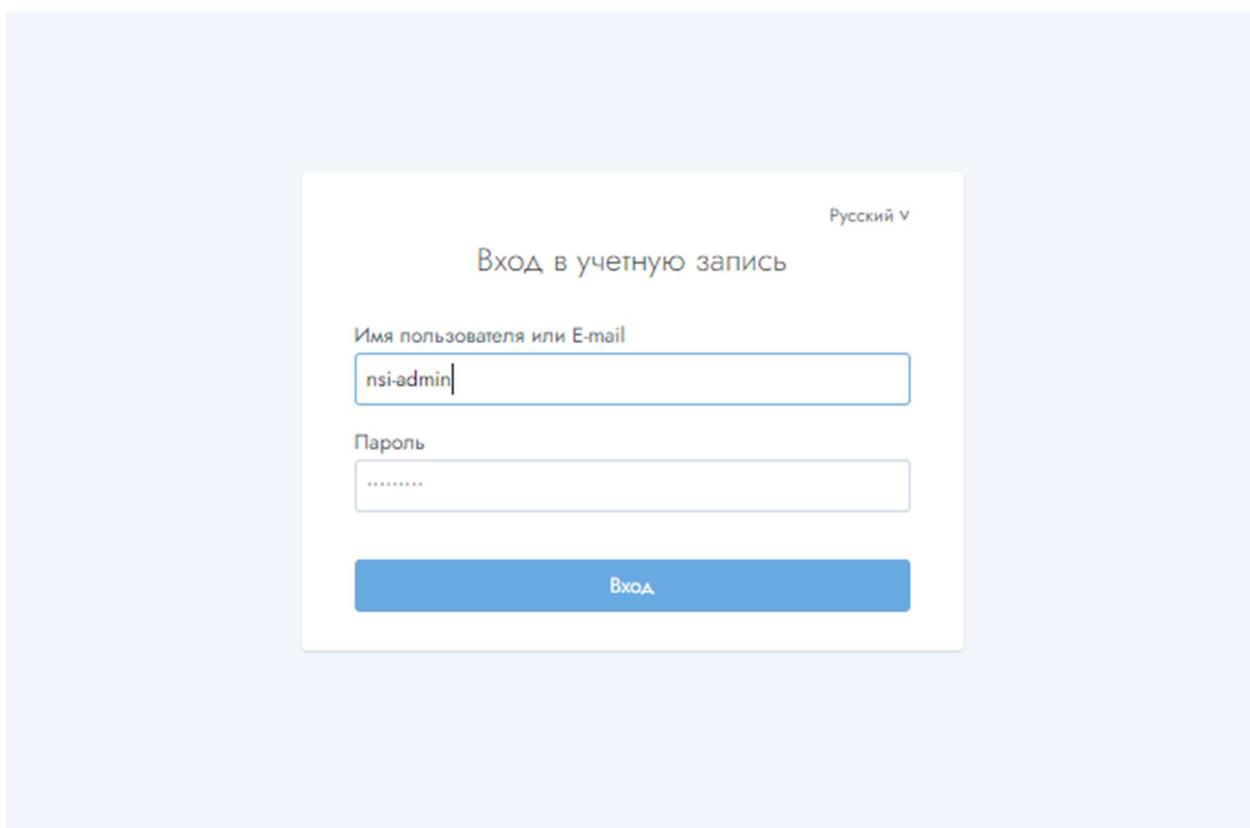
Изм.	Лист	№ докум.	Подп.	Дата	Лист
					48

## 4.2 Работа с ЗЕН-НСИ

- Для работы с ЗЕН-НСИ необходимо в браузере открыть страницу с адресом <https://ваше-доменное-имя> и нажать кнопку “Вход в систему” на странице с описанием ЗЕН-НСИ.



- Авторизоваться под пользователем `nsi-user` или `nsi-admin` используя пароль, установленный ранее при создании пользователей в авторизационном сервисе Keycloak



- После прохождения авторизации будет отображена главная страница ЗЕН-НСИ

Инва № подл.	Подп. и дата	Взамен инв. №	Инва № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					49

**ZEN-IT** Типы справочников Виды справочников Справочники НСИ Список пользователей Администратор НСИ

### Система управления нормативно-справочной информацией «ZEN-НСИ»

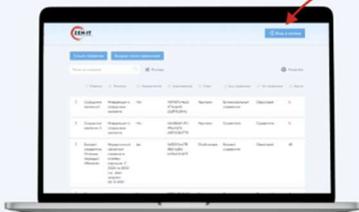
предназначена для ведения и поддержки различных справочников, которые обеспечивают ввод данных в информационном пространстве, в котором ведется работа и может являться централизованным источником нормативно-справочной информации для других программ или систем

**Функциональные возможности системы**

Обеспечивает выполнение следующих функций: создание, удаление, добавление, изменение данных и метаданных объектов нормативно-справочной информации (справочников и их записей), в том числе структур справочников, а также ведение версий справочников и истории изменений

Создание
+ Добавление
Изменение
Удаление

Ведение версий
История изменений



Для начала работы с ZEN-НСИ авторизуйтесь в системе и вы попадете на демонстрационную страницу функционала системы, на которой сможете в интерактивном режиме попробовать возможности по работе со справочниками, управление их структурой и данными.

[ПОПРОБОВАТЬ ZEN-НСИ](#)

**Решаемые задачи**

- ✓ Обеспечение единого хранилища справочной информации в системе
- ✓ Исключение дублирования данных или неактуальной информации
- ✓ Организация единой точки ввода, обработки и контроля содержимого справочников
- ✓ Построение хранилища мастер-данных для других программ или систем

**Работа с интерфейсом и функционалом ZEN-НСИ описана в документе «Руководство пользователя ZEN-НСИ».**

Изн. № подл.	Подп. и дата	Взамен инв. №	Изн. № дубл.	Подп. и дата				Лист
								50
Изм.	Лист	№ докум.	Подп.	Дата				





## СОСТАВИЛИ

Наименование организации, предприятия	Должность исполнителя	Фамилия, имя, отчество	Подпись	Дата
ООО «ЗЕН-АЙТИ»				

## СОГЛАСОВАНО

Наименование организации, предприятия	Должность исполнителя	Фамилия, имя, отчество	Подпись	Дата

Инва № подл.	Подп. и дата	Взамен инв. №	Инва № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	

