# Инструкция по установке дистрибутива

# ПО «Сервис интеграции с ФРЛЛО»

На 3 листах

## 1. Требования к системе

Для функционирования ПО «Сервис интеграции с ФРЛЛО» необходимо следующее обеспечение:

- ОС AlterOS 9.7 (x86_64) или совместимая Linux-система.
- Docker Engine и Docker Compose v2.
- Веб-браузер: Yandex Browser, Google Chrome или Mozilla Firefox.
- ОЗУ: не менее 8 ГБ. Свободное место: не менее 10 ГБ плюс место под Docker-образы и базы данных.
- Доступ в интернет для загрузки Docker-образов (postgres, .NET, nginx) при первом запуске.
- PostgreSQL и .NET отдельно устанавливать не требуется: они запускаются в контейнерах.

## 2. Порядок установки дистрибутива ПО «Сервис интеграции с ФРЛЛО»

1. Установите Docker и Docker Compose v2:

```
sudo dnf install -y docker docker-compose2
sudo systemctl enable --now docker
docker --version
docker compose version
```

2. Распакуйте дистрибутив и перейдите в каталог проекта:

```
unzip 'Сервис интеграции с ФРЛЛО.zip'
cd frllo-compose
```

3. Загрузите Docker-образы, используемые сервисами:

```
docker compose pull
```

4. Проверьте конфигурацию Docker Compose:

```
docker compose config >/dev/null && echo "Compose config: OK"
```

5. Запустите сервисы:

```
docker compose up -d
docker compose ps -a
```

Будут запущены контейнеры postgres_db, nsi_api и nsi_frontend. Они должны иметь статус Up или Up (healthy). При первом запуске выполняются миграции базы данных и инициализация сервиса, это может занять некоторое время.

## 3. Открытие программы

Если браузер запущен на том же компьютере, где установлен дистрибутив, откройте:

```
http://127.0.0.1:4200
```

Если браузер запущен на другом компьютере в сети, используйте IP-адрес сервера:

```
http://<ip-address>:4200
```

Адреса компонентов сервиса:

| Компонент | Адрес |
|---|---|
| Веб-интерфейс сервиса (NSI UI) | http://127.0.0.1:4200 |
| API сервиса (Swagger) | http://127.0.0.1:5000/swagger |
| База данных PostgreSQL | 127.0.0.1:5432 |

Убедитесь, что порты 4200, 5000 и 5432 доступны (открыты в межсетевом экране при необходимости).

## 4. Быстрая проверка

```
docker compose ps -a
docker compose exec -T postgres pg_isready -U nsi_user -d nsi_db
```

Если контейнер завершился с ошибкой, посмотрите его журнал:

```
docker compose logs --tail=100 <service>
```

## 5. Остановка и повторный запуск

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

Обычная команда docker compose down не удаляет данные PostgreSQL.

## 6. Удаление

Для удаления контейнеров и сети:

```
docker compose down
```

Для полного удаления вместе с Docker volumes:

```
docker compose down -v
```

**ВНИМАНИЕ:** docker compose down -v удаляет данные PostgreSQL.

## 7. Важно

Не удаляйте файлы docker-compose.yml, init.sql, nsi-ui-nginx.conf и каталоги api/ и frontend/: они используются сервисами при запуске.

При первом запуске автоматически срабатывают скрипты инициализации базы данных (init.sql) и миграции.