

# T2Plus Управление Производством PRO

## Инструкция по развертыванию продукта

# Оглавление

1. Общая информация .....	3
2. Предустановленное ПО .....	3
3. Запуск контейнера продукта T2Plus SPM .....	3
4. Запуск проверочного приложения .....	4
5. Конфигурационный файл .....	5

# 1. Общая информация

В инструкции описан порядок развертывания программного продукта **T2Plus Управление Производством PRO** (сокращенное наименование — **T2Plus SPM**) в инфраструктуре Docker.

В поставку программного продукта входят:

- Docker-образ `spm`.
- Файл конфигурации `docker-compose.yml`.
- BackUp-базы `postgres spm.bak`.

## 2. Предустановленное ПО

На рабочей станции должно быть предварительно развернуто ПО Docker Desktop либо его аналоги. Сама рабочая станция может быть как под управлением ОС Linux (например, Astra Linux SE 1.7.5), так и под управлением ОС Windows (например, Windows 10+).

Краткая инструкция по установке Docker Desktop на рабочую станцию с ОС Windows (для ОС Linux выполняемые операции аналогичны):

1. Включите виртуализацию в BIOS.
2. Для Windows включите поддержку Hyper-V.
3. Установите Docker Desktop.
4. Информацию о системных требованиях и инструкцию по установке Docker Desktop см. на сайте (<https://docs.docker.com/desktop/install/windows-install/>).
5. В настройках Docker Desktop (**Settings > File sharing**) определите каталоги (и их подкаталоги), которые можно привязать к контейнерам Docker. Например, укажите каталог, из которого запускается `docker-compose`.

## 3. Запуск контейнера продукта T2Plus SPM

1. Откройте терминал (для ОС Windows — **PowerShell** от имени администратора, для ОС Linux — **terminal**).
2. Установите рабочим каталогом папку `spm`, в которой находится файл `docker-compose.yml` с конфигурацией сервисов приложения продукта **T2Plus SPM**.
3. Запустите контейнеры в фоновом режиме командой:

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

Если контейнер запустится без ошибок, в терминале отобразится примерно следующий текст:

```
d:\Docker_pg\SPM>docker compose up
[+] Running 9/9
  0 spm 8 layers [██████████] 0B/0B Pulled
  0 70ba6f391a98 Pull complete
  0 373a7eec07ac Pull complete
  0 8b82146158e3 Pull complete
  0 75f74dbda27f Pull complete
  0 44a4352dce9a Pull complete
  0 c9d1fd670509 Pull complete
  0 b6086bf7ea59 Pull complete
  0 4f4fb700ef54 Pull complete
[+] Running 4/4
  0 Network spm_default Created
  0 Volume "spm_postgres" Created
  0 Container spm-db-1 Created
  0 Container spm-spm-1 Created
Attaching to db-1, spm-1
```

Рисунок 1.

## 4. Запуск проверочного приложения

Для проверки корректности развертывания рекомендуется произвести запуск приложения. Для этого в веб-обозревателе откройте ссылку <http://localhost:10111/>. Возможно, потребуется какое-то время на запуск приложения (обновите окно по F5).

Приложение можно запустить из GUI Docker Desktop. Для этого в категории **Containers** выберите контейнер **sp** и кликните мышью по ссылке в столбце **Port(s)**.



Рисунок 2.

В диалоговом окне «Вход в систему» нажмите одноименную кнопку [Вход в систему]. В качестве **Имени пользователя** укажите *Admin*, без пароля.

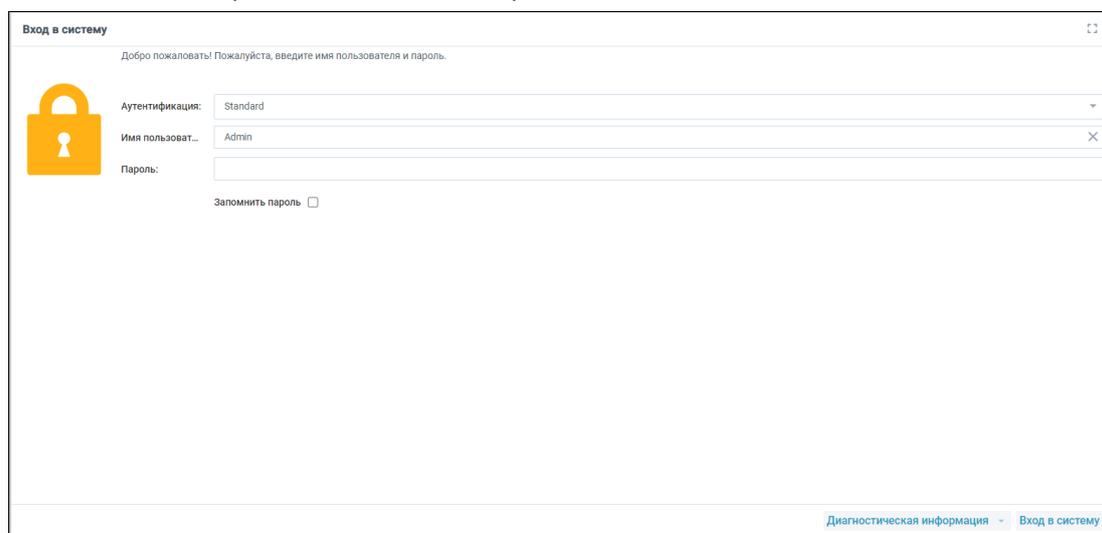


Рисунок 3.

После входа откроется окно приложения.

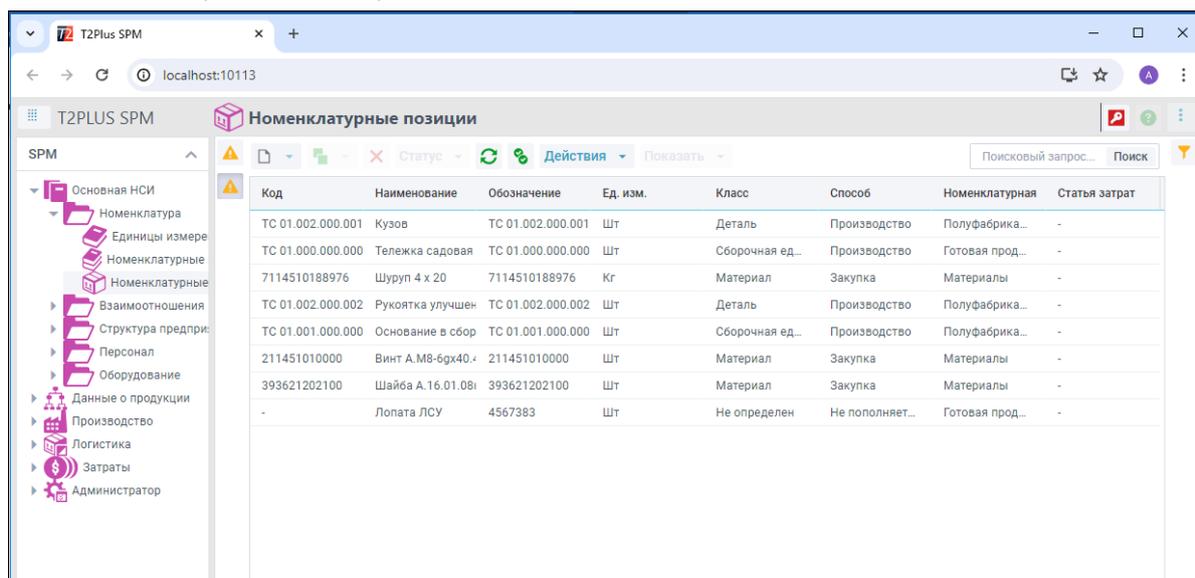


Рисунок 4.

Если все операции выполнены успешно, продукт **T2Plus SPM** развернут корректно и готов к эксплуатации.

## 5. Конфигурационный файл

Конфигурационный файл `docker-compose.yml` определяет сервисы и настройки конфигурации приложения продукта **T2Plus SPM**:

```
services:
  db:
    image: postgres:16.1
    ports:
      - "5432:5432"
    restart: always
    environment:
      TZ: 'Europe/Minsk'
      PGTZ: 'Europe/Minsk'
      POSTGRES_PASSWORD: postgres
      POSTGRES_USER: postgres
      BACKUP: spm.bak
    volumes:
      - postgres:/var/lib/postgresql/data
      - ./postgres.conf:/etc/postgresql/postgresql.conf
      - ./backup/:/backup/
      - ./apmrestore.sh:/docker-entrypoint-initdb.d/dbrestore.sh
    command: -c 'config_file=/etc/postgresql/postgresql.conf'

  spm:
    image: rg.topsoft.local/spm/spm-app:8.0.5
    privileged: true
    depends_on:
      - db
    environment:
      - aspnetcore_environment=production
      - aspnetcore_urls=http://*:80
      - xafari_environment=production
      - aspnetcore_forwardedheaders_enabled=true
      - sezal:connectionstrings:connectionstring=DaperProvider=Postgres;Server=db;
User ID=postgres; Password=postgres; Database=postgres; Persist Security info=true
    ports:
      - 10111:80
      - 1001:443 - ASPNETCORE_URLS=http://*:80
      - Sezal__DatabaseProviderType=PostgreSQL
      - Sezal__ConnectionStrings__Default=Server=db_postgres;User
ID=postgres;Password=postgres;Pooling=false;Persist Security
Info=true;DataBase=Sezal.MainDemo_EF
    volumes:
      postgres:
```