

**Общество с ограниченной ответственностью «Тетракуб»
ОГРН 1107847219480
ИНН 7802719547**

УТВЕРЖДАЮ

Генеральный директор
ООО «Тетракуб»
Лексашов Александр Викторович

**ИНСТРУКЦИЯ ПО УСТАНОВКЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
«ИНДУСТРИАЛЬНАЯ АНАЛИТИЧЕСКАЯ ИНФОРМАЦИОННАЯ СИСТЕМА
«INDANIS»»**

2025

СОДЕРЖАНИЕ

1. Общее положение	3
2. Системные требования	4
3. Описание установочного пакета	5
4. Описание процесса установки	6
4.1. Установка Docker и Docker Compose	6
4.1.1. Для Astra Linux 1.8	6
4.1.2. Для Windows 10/11	6
4.2. Загрузка Docker Images	6
4.3. Запуск и проверка системы	7
4.4. Установка и запуск Task Executor Controller	7
4.4.1. Для Astra Linux 1.8 и Ubuntu 24.04	7
4.4.2. Для Windows 10/11	8
4.5. Проверка доступности сервисов	8
4.6. Ошибки и решения	8
4.7. Полезные команды	9

1. Общее положение

Основной целью программного продукта «Индустриальная аналитическая информационная система «Indanis»» является обеспечение возможности управления данными и расчетами компьютерного моделирования производственных и технологических процессов.

Настоящий документ содержит описание процесса установки программного обеспечения «Индустриальная аналитическая информационная система «Indanis»» (далее – Программа).

2. Системные требования

Для эксплуатации Программы рабочая станция должна соответствовать следующим минимальным системным требованиям:

- Центральный процессор Intel Core i5
- Оперативная память объемом не менее 8 Гб
- Жесткий диск с объемом не менее 40 Гб

Программа может использоваться на следующих платформах:

- настольные ПК, ноутбуки

Зависимости:

- Minio 8.5.17

<https://github.com/minio>

GNU AGPL v3

- Postgis 15-3.4-alpine

<https://git.osgeo.org/gitea/postgis/postgis>

GNU GPLv2

- MailHog 1.0.1

<https://github.com/mailhog>

MIT License (MIT)

Операционные системы:

- Windows 10/11
- Linux (Ubuntu 24.04, Astra Linux 1.8)

3. Описание установочного пакета

Для начала установки и настройки программы необходимы:

1. Установленная операционная система: Ubuntu 24.04, Astra Linux 1.8, Windows 10/11 (64-bit).
2. Интернет-соединение для установки необходимых пакетов.
3. Docker-образы для приложений, файл docker-compose.yml, установочный пакет TaskExecutor Controller:
 - Образ UI: ui.tar
 - Образ App: app.tar
 - Образ TaskExecutor (серверная часть Сервиса обработки вычислений): taskexecutor.tar
 - Образ MailHog: mailhog.tar
 - Образ PostGIS: postgis.tar
 - Образ Minio: minio.tar
 - Установочный пакет TaskExecutor Controller (клиентская часть Сервиса обработки вычислений)
 - Файл docker-compose.yml

4. Описание процесса установки

4.1. Установка Docker и Docker Compose

4.1.1. Для Astra Linux 1.8

Установка выполняется через командную оболочку bash.

1. Установите Docker

```
sudo apt update  
sudo apt install docker.io
```

2. Установите Docker Compose

```
sudo curl -L  
"https://github.com/docker/compose/releases/download/v2.24.5/docker-  
compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose  
sudo chmod +x /usr/local/bin/docker-compose
```

3. Проверьте установку

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

4.1.2. Для Windows 10/11

Установка выполняется через командную оболочку PowerShell. Оператор должен обладать правами администратора.

1. Установите WSL 2:

```
wsl --install
```

Перезагрузите ПК.

2. Установите Docker Desktop с официального сайта, выбрав опцию «Enable WSL 2».

3. Проверьте установку:

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

4.2. Загрузка Docker Images

1. Поместите заранее подготовленные .tar-архивы образов и другие файлы на целевую машину.
2. Перейдите в папку с архивами и загрузите Docker-образы:

```
docker load -i ui.tar
```

```
docker load -i app.tar
docker load -i postgis.tar
docker load -i minio.tar
docker load -i mailhog.tar
docker load -i taskexecutor.tar
```

3. После загрузки проверьте наличие образов:

```
docker images
```

4.3. Запуск и проверка системы

1. Убедитесь, что находитесь в директории, где лежит `docker-compose.yml`, и запустите контейнеры:

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

2. Проверьте, что все контейнеры в состоянии **Up**:

```
docker ps -a
```

3. Проверьте логи каждого сервиса:

```
docker logs app
```

```
docker logs db
```

```
docker logs minio
```

```
docker logs task-executor
```

4.4. Установка и запуск Task Executor Controller

4.4.1. Для Astra Linux 1.8 и Ubuntu 24.04

Установка выполняется через командную оболочку `bash`.

1. Перейдите в директорию с поставляемым `deb`-пакетом:

```
cd <путь-к-deb-пакету>
```

2. Установите приложение:

```
sudo apt update
```

```
sudo apt install ./taskexecutor-controller_1.0.0-1_amd64.deb -y
```

3. Запустите приложение через команду или нажатием на соответствующий ярлык:

```
taskexecutor-controller
```

4.4.2. Для Windows 10/11

Установка выполняется через командную оболочку cmd.

1. Установите приложение при помощи поставляемого мастера установки exe.
2. Перейдите в директорию установки приложения:
`cd <путь-установки>`
3. Запустите приложение через команду или нажатием на соответствующий ярлык:
`.\taskexecutor-controller`

4.5. Проверка доступности сервисов

1. Web-интерфейс:
 - Откройте в браузере `http://localhost:9000`
2. MinIO (файловое хранилище):
 - Откройте в браузере `http://localhost:19001`
 - Логин: `minioadmin`, пароль: `minioadmin`
3. MailHog – SMTP-сервер для обработки почты:
 - Откройте в браузере `http://localhost:8025`

4.6. Ошибки и решения

Проблема	Диагностика	Решение
Контейнер не запускается	<code>docker logs <container_name></code>	Проверить зависимости (<code>depends_on</code>) в файле <code>docker-compose.yml</code>
Ошибка подключения к БД	<code>docker logs db</code>	Проверить логин и пароль в <code>environment</code> (<code>POSTGRES_USER</code> , <code>POSTGRES_PASSWORD</code>) в

		файле docker-compose.yml
TaskExecutor не может подключиться	<code>docker exec -it task-executor curl http://app:8080</code>	Проверить URL в конфиге
MinIO не сохраняет данные	<code>ls -l ./data/storage</code>	Проверить права на папку

4.7. Полезные команды

- Пересоздать контейнеры:

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

- Проверить здоровье сервисов:

```
docker inspect --format='{{.State.Health.Status}}' <container_name>
```