

УТВЕРЖДАЮ
Управляющий ООО «ДиСофт»



Шимкова Л.Л.

«16» февраля 2026 г.

ИНФОРМАЦИЯ
необходимая для установки и эксплуатации программного обеспечения
«Табло, дороги, оповещения»

Правообладатель: ООО «ДиСофт» (ИНН 9722044395)

Листов 10

г. Москва
2026

Инструкция по развертыванию приложений для управления дорожными экранами

1. Требования к системе и подготовка.

1.1 Аппаратные требования:

- CPU: 2-4 ядра (рекомендуется 4+)
- RAM: 4-8 ГБ
- SSD: 50-100 ГБ (в зависимости от объема данных)

1.2 Программные требования:

- ОС: Ubuntu 22.04 LTS (с долгосрочной поддержкой)
- Среда выполнения: .NET: ASP.NET Core 8.0 LTS
- Веб-сервер: Nginx: последняя стабильная версия
- База данных: PostgreSQL: 15+

2 Описание структуры приложения

Система управления дорожными экранами состоит из четырёх взаимосвязанных компонентов:

- Disoft.RoadInfo.WebAdminAPI - основной web-сайт для управления контентом и пользователями
- Disoft.RoadInfo.WebClientAPI - web-сайт для обмена информацией с десктопным ПО
- Disoft.RoadInfo.FileManagerAPI - файловое хранилище для созданного контента
- Disoft.RoadInfo.Client.WebAPI - десктопное ПО для отображения контента

Для обеспечения работы клиентского приложения необходимо развернуть на сервере следующие компоненты:

Disoft.RoadInfo.WebAdminAPI, Disoft.RoadInfo.WebClientAPI и Disoft.RoadInfo.FileManagerAPI это позволит десктопному приложению получать контент, управлять данными и обеспечит стабильную работу системы

3 Подготовка к развёртыванию

Развёртывание выполняется через SSH-подключение к серверу. Для передачи файлов рекомендуется использовать WinSCP или аналогичный инструмент.

Подключимся к серверу по SSH, используя его IP-адрес, логин и пароль

3.1 Обновление системы и установка .NET Runtime

Необходимо установить среду выполнения .NET Runtime, для этого в терминале вводим команды

```
# Обновление списка пакетов и установка зависимостей
sudo apt-get update && sudo apt-get upgrade -y
sudo apt-get install -y curl wget gnupg ca-certificates
```

```

# Добавление репозитория Microsoft и установка .NET Runtime
wget https://packages.microsoft.com/config/ubuntu/22.04/packages-microsoft-prod.deb -O
packages-microsoft-prod.deb

sudo dpkg -i packages-microsoft-prod.deb

rm packages-microsoft-prod.deb

# Установка ASP.NET Core Runtime 8.0
sudo apt-get update

sudo apt-get install -y aspnetcore-runtime-8.0

# Проверка установки
dotnet --info

```

3.2 Установка и настройка PostgreSQL

Для приложений Disoft.RoadInfo.WebAdminAPI и Disoft.RoadInfo.WebClientAPI необходимо развернуть СУБД PostgreSQL

```

# Добавление официального репозитория PostgreSQL
sudo sh -c 'echo "deb https://apt.postgresql.org/pub/repos/apt $(lsb_release -cs)-pgdg main" >
/etc/apt/sources.list.d/pgdg.list'

wget -qO- https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo tee
/etc/apt/trusted.gpg.d/pgdg.asc

# Установка PostgreSQL 15
sudo apt-get update

sudo apt-get install -y postgresql-15 postgresql-contrib-15

# Настройка службы
sudo systemctl start postgresql

sudo systemctl enable postgresql

sudo systemctl status postgresqlНастройка базы данных:

# Переключение на пользователя postgres
sudo -i -u postgres

# Запуск psql
psql

-- Изменение пароля пользователя postgres
ALTER USER postgres PASSWORD 'your_new_password';

-- Выход из psql

```

\q

БД будут созданы и инициализированы при запуске приложений

3.3 Установка и настройка Nginx

Установка Nginx

```
sudo apt-get install -y nginx
```

Запуск и добавление в автозагрузку

```
sudo systemctl start nginx
```

```
sudo systemctl enable nginx
```

Пример конфигурации для проксирования запросов к API (файл `/etc/nginx/sites-available/roadinfo`):

```
server {  
    listen 80;  
    server_name your_domain_or_ip;  
  
    location /adminapi/ {  
        proxy_pass http://localhost:5001;  
        proxy_http_version 1.1;  
        proxy_set_header Upgrade $http_upgrade;  
        proxy_set_header Connection keep-alive;  
        proxy_set_header Host $host;  
        proxy_cache_bypass $http_upgrade;  
    }  
  
    location /clientapi/ {  
        proxy_pass http://localhost:5002;  
        proxy_http_version 1.1;  
        proxy_set_header Upgrade $http_upgrade;  
        proxy_set_header Connection keep-alive;  
        proxy_set_header Host $host;  
        proxy_cache_bypass $http_upgrade;  
    }  
}
```

```
location /fileapi/ {
    proxy_pass http://localhost:5003;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection keep-alive;
    proxy_set_header Host $host;
    proxy_cache_bypass $http_upgrade;
}
}
```

proxy_pass <http://localhost:5001>; порты указаны в качестве примера. Для каждого приложения укажите собственный порт из диапазона 1024-49151.

Активируйте конфигурацию и перезапустите Nginx:

```
sudo ln -s /etc/nginx/sites-available/roadinfo /etc/nginx/sites-enabled/
sudo nginx -t
sudo systemctl restart nginx
sudo systemctl enable nginx
```

4 Развёртывание приложений

4.1 Создание рабочих директорий

Определение имени проекта (замените `your_project` на актуальное название)

```
projectName="disoft_roadinfo"
```

Создание корневой директории проекта

```
sudo mkdir -p /var/www/$projectName
```

Создание поддиректорий для каждого компонента

```
sudo mkdir -p /var/www/$projectName/{adminapi,clientapi,fileapi,logs,temp}
```

Создание директорий для файлового хранилища

```
sudo mkdir -p /var/www/$projectName/files/{roadsigns,media,previews,software,uploads}
```

Копирование опубликованных файлов приложений

Предполагается, что файлы находятся в локальной папке `./publish/`

```

# Для каждого компонента:

sudo cp -r ./publish/WebAdminAPI/* /var/www/$projectName/adminapi/
sudo cp -r ./publish/WebClientAPI/* /var/www/$projectName/clientapi/
sudo cp -r ./publish/FileManagerAPI/* /var/www/$projectName/fileapi/

# Настройка прав доступа

sudo chown -R www-data:www-data /var/www/$projectName
sudo chmod -R 755 /var/www/$projectName

# Особые права для директорий с файлами (разрешение записи)

sudo chmod -R 775 /var/www/$projectName/files/
sudo chmod -R 775 /var/www/$projectName/temp/

```

4.2 Конфигурация приложений

Для корректной работы, а также для взаимодействия между собой необходимо настроить следующие параметры в конфигурационном файле appsettings.json.

Сохранить файл и скопировать каталог приложения на сервер в ранее созданную папку

а) Настройка приложения Disoft.RoadInfo.FileManagerAPI

```

"FileManagerSettings": {
  "RoadSignFileFolder": ""<ПУТЬ_К_КАТОЛОГУ_ФАЙЛОВ_ИЗОБРАЖЕНИЙ>"",
  "CustomMediaFileFolder": "<ПУТЬ_К_КАТОЛОГУ_МЕДИАФАЙЛОВ>",
  "ScreenPreviewFileFolder": "<ПУТЬ_К_КАТОЛОГУ_ПРЕВЬЮ_ЭКРАНОВ>",
  "SoftwareFileFolder": "<ПУТЬ_К_КАТОЛОГУ_ФАЙЛОВ_ОБНОВЛЕНИЙ>"
}

```

б) Настройка приложения Disoft.RoadInfo.WebAdminAPI

```

"ConnectionStrings": {
  "DisoftRoadInfo": "<ПАРАМЕТРЫ_ПОДКЛЮЧЕНИЯ_К_ОСНОВНОЙ_БД>",
  "DisoftOperationLog": "<ПАРАМЕТРЫ_ПОДКЛЮЧЕНИЯ_К_БД_ЛОГОВ>"
},
"SmtpSettings": {
  "Server": "<ПОЧТОВЫЙ_СЕРВЕР>",
  "Port": "<ПОРТ_ПОЧТОВОГО_СЕРВЕРА>",
  "SenderName": "<ИМЯ_ОТПРАВИТЕЛЯ>",
  "SenderEmail": "<АДРЕС_ОТПРАВИТЕЛЯ>",
  "Username": "<АДРЕС_ОТПРАВИТЕЛЯ>",

```

```

"Password": "<ПАРОЛЬ_ПОЧТЫ>"
},
"CommunicationSettings": {
  "CommunicationHubURL": "<URL_ДЛЯ_ПОДКЛЮЧЕНИЯ_К_WEBSOCKET>",
  "ClientWebApiURL": "<URL_ОПУБЛИКОВАННОГО_WEB_САЙТА_CLIENT_API>",
  "FileManagerUrl": "<URL_ФАЙЛОВОГО_СЕРВЕРА>",
  "AdminWebApiURL": "<URL_ОПУБЛИКОВАННОГО_WEB_САЙТА_ADMIN_API>"
},
"FileManagerSettings": {
  "RoadSignFileFolder": ""<ПУТЬ_К_КАТОЛОГУ_ФАЙЛОВ_ИЗОБРАЖЕНИЙ>""",
  "CustomMediaFileFolder": "<ПУТЬ_К_КАТОЛОГУ_МЕДИАФАЙЛОВ>",
  "ScreenPreviewFileFolder": "<ПУТЬ_К_КАТОЛОГУ_ПРЕВЬЮ_ЭКРАНОВ>",
  "SoftwareFileFolder": "<ПУТЬ_К_КАТОЛОГУ_ФАЙЛОВ_ОБНОВЛЕНИЙ>"
}

```

с) Настройка приложения Disoft.RoadInfo.WebClientAPI

```

"ConnectionStrings": {
  "DisoftRoadInfo": "<ПАРАМЕТРЫ_ПОДКЛЮЧЕНИЯ_К_ОСНОВНОЙ_БД>",
  "DisoftOperationLog": "<ПАРАМЕТРЫ_ПОДКЛЮЧЕНИЯ_К_БД_ЛОГОВ>"
},
"CommunicationSettings": {
  "ClientWebApiURL": "<URL_ОПУБЛИКОВАННОГО_WEB_САЙТА_CLIENT_API>",
  "FileManagerUrl": "<URL_ФАЙЛОВОГО_СЕРВЕРА>",
  "AdminWebApiURL": "<URL_ОПУБЛИКОВАННОГО_WEB_САЙТА_ADMIN_API>"
},
"FileManagerSettings": {
  "RoadSignFileFolder": ""<ПУТЬ_К_КАТОЛОГУ_ФАЙЛОВ_ИЗОБРАЖЕНИЙ>""",
  "CustomMediaFileFolder": "<ПУТЬ_К_КАТОЛОГУ_МЕДИАФАЙЛОВ>",
  "ScreenPreviewFileFolder": "<ПУТЬ_К_КАТОЛОГУ_ПРЕВЬЮ_ЭКРАНОВ>",
  "SoftwareFileFolder": "<ПУТЬ_К_КАТОЛОГУ_ФАЙЛОВ_ОБНОВЛЕНИЙ>"
},
"JwtSettings": {
  "Issuer": "<ИЗДАТЕЛЬ_КЛЮЧА>",

```

```
"Audience": "<ПОЛЬЗОВАТЕЛИ_КЛЮЧА>",
"SecretKey": "<СЕКРЕТНЫЙ_КЛЮЧ>", (мин. 32 символа)
"TokenLifetimeMinutes": "<ВРЕМЯ_ЖИЗНИ_ТОКЕНА>"
}
```

4.3 Рекомендации

- После настройки всех конфигурационных файлов убедитесь, что пути к директориям существуют и права доступа настроены корректно. Это критически важно для работы файлового менеджера и системы в целом.
- Строка "<ПАРАМЕТРЫ_ПОДКЛЮЧЕНИЯ_К_БД_ЛОГОВ>", в которой будет храниться журнал операций пользователей в системе (отдельная БД! Не допускается использовать одну и ту же БД в обоих строках подключения).
- Строка "<URL_ДЛЯ_ПОДКЛЮЧЕНИЯ_К_WEBSOCKET>". Это URL состоящий из web-адреса сайта Client API "<URL_ОПУБЛИКОВАННОГО_WEB_САЙТА_CLIENT_API>" и служебного пути «user_device_messages». Например, 10.0.1.50:50, то в этом случае, CommunicationHubURL параметр должен быть задан как [http://10.0.1.50:50/user device messages](http://10.0.1.50:50/user_device_messages)

4.4 Создание systemd сервисов для приложений

```
# Обновление демона systemd
```

```
sudo systemctl daemon-reload
```

Для управления запуском и восстановлением приложений установим их в качестве служб

Для каждого приложения создадим отдельную службу

```
# Создание файла сервиса
```

```
sudo nano /etc/systemd/system/kestrel-$(projectName.service)
```

Содержимое файла сервиса:

```
[Unit]
```

```
Description=ASP.NET Core Web Application - $(projectName.service)
```

```
[Service]
```

```
WorkingDirectory=/var/www/$(projectName.service)
```

```
ExecStart=/usr/bin/dotnet /var/www/$(projectName.service)/$(projectName.service).Web.dll
```

```
Restart=always
```

```
RestartSec=10
```

```
KillSignal=SIGINT
```

```
SyslogIdentifier=dotnet-$(projectName.service)
```

```
Environment=ASPNETCORE_ENVIRONMENT=Production
Environment=DOTNET_PRINT_TELEMETRY_MESSAGE=false
Environment=ASPNETCORE_HTTPS_PORT=5001
Environment=ASPNETCORE_URLS=http://*:5090
```

[Install]

```
WantedBy=multi-user.target
```

Где:

WorkingDirectory – путь к каталогу приложения

ExecStart - Команда запуска приложения.

`/var/www/$<projectName.service>/$<projectName.service>.Web.dll` - основная библиотека приложения

SyslogIdentifier - Идентификатор в системных логах

Environment=ASPNETCORE_URLS=http://*:5090 - URL и порты, которые слушает Kestrel

Порт для приложения должен совпадать с портом указанным для приложения в nginx(пункт 3.1)

```
# Проверяем и перезапускаем Nginx
```

```
sudo nginx -t
```

```
sudo systemctl reload nginx
```

Для управления службами есть следующие команды:

```
# Включение автозапуска
```

```
sudo systemctl enable kestrel-$projectName.service
```

```
# Запуск сервиса
```

```
sudo systemctl start kestrel-$projectName.service
```

```
# Проверка статуса
```

```
sudo systemctl status kestrel-$projectName.service
```

```
# Остановка сервиса
```

```
sudo systemctl stop kestrel-$projectName.service
```

```
# Перезапуск сервиса
```

```
sudo systemctl restart kestrel-$projectName.service
```

для каждого приложения нужно включить автозапуск и запустить сервис, проверить статус. Система готова к работе после успешного запуска всех компонентов и проверки их взаимодействия.

5. Установка десктопного приложения Disoft.RoadInfo.Client.WebAPI

5.1 Требования к системе:

ОС: Windows 7/10/11 (x64)

.NET Framework: 4.8 или .NET 6/7/8

Права: Администраторские права для установки

Порты: Разрешенные порты для исходящих подключений (обычно 80, 443)

5.2 Установка приложения

Для установки приложения нужно запустить файл установщик Disoft RoadInfo Setup.exe

Он установит приложение и предложит его запустить.

5.3 Первоначальная настройка

Получение данных для подключения:

1. Адрес сервера API.
2. Ключ устройства.

В качестве адреса указать внешний адрес сервера Disoft.RoadInfo.WebClientAPI и служебный путь «user_device_messages». Например, сервер Disoft.RoadInfo.WebClientAPI имеет адрес: «disoft.webclients.ru», то в этом случае, параметр должен быть задан как [http://disoft.webclients.ru/user device messages](http://disoft.webclients.ru/user_device_messages).

Так же устанавливаем уникальный ключ для устройства. Приложение подключится к серверу под указанным ключом и будет готово к работе.