**Руководство по установке**

Дата: 2025-08-13

- WebAPI — серверная часть (ASP.NET Core Web API).

- ClassLibrary — общие модели.

- BlazorApp.sln — решение для сборки.

- BlazorApp — клиентская часть(ASP.NET Core Blazor WebAssembly).

**1. Предварительные требования**

1) .NET 9 SDK (Linux/macOS/Windows) — нужен для сборки и запуска проектов.

Ссылка: https://dotnet.microsoft.com/download/dotnet/9.0

2) PostgreSQL 16/17 (или совместимая версия).

Ссылка: https://www.postgresql.org/download/

3) Git (для клонирования репозитория).

Ссылка: https://github.com/Maksim9056/WareHouse.git

Проверка версий:

dotnet --info

psql --version

**2. Клонирование проекта**

git clone https://github.com/Maksim9056/WareHouse.git

cd WareHouse

**3. Настройка конфигурации**

**3.1 Клиент (Blazor WebAssembly)**

Файл: BlazorApp/BlazorApp/BlazorApp/appsettings.json

Пример:

{

"Urls": {

"Google": "https://www.google.com",

"Api": "https://localhost:7166/api/"

}

}

Файл: BlazorApp/BlazorApp/BlazorApp.Client/wwwroot/appsettings.json

Пример:

{

"Logging": {

"LogLevel": {

"Default": "Information",

"Microsoft.AspNetCore": "Warning"

}

},

"Urls": {

"Api": "https://localhost:7166/api/"

}

}

Где Urls:Api — это базовый адрес API. При локальном запуске WebAPI по умолчанию используется https://localhost:7166/api/. При необходимости измените порт на фактический порт вашего WebAPI.

**3.2 Сервер (WebAPI, PostgreSQL)**

Файл: BlazorApp/WebAPI/appsettings.json

Пример:

{

"Logging": {

"LogLevel": {

"Default": "Information",

"Microsoft.AspNetCore": "Warning"

}

},

"ConnectionStrings": {

"Postgres": "Host=localhost;Database=WareHouse1;Username=finance;Password=finance\_pass",

"Client": "https://localhost:7104"

},

"AllowedHosts": "\*"

}

Параметры:

- Host — адрес сервера PostgreSQL (например, localhost или IP сервера).

- Database — имя базы (пример: WareHouse1).

- Username — пользователь PostgreSQL (пример: finance).

- Password — пароль пользователя.

- Client -адрес клиента

**4. Подготовка инструментов**

Если требуется ручной прогон миграций, установите инструменты EF:

dotnet tool install --global dotnet-ef

и выполните внутри проекта WebAPI :

dotnet ef database update

**5. Сборка и запуск**

В корне решения:

dotnet restore

dotnet build

**5.1 Запуск приложения**Запуск с помощью Visual Studio 2022.  
Нажать на файл BlazorApp.sln  
и проект откроется, выберите два проекта: BlazorApp , WebAPI .  
Нажми кнопку пуск.