

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АЕРОКОСМІЧНИЙ УНІВЕРСИТЕТ ім. М. Є.
Жуковського «Харківський авіаційний інститут»

Факультет радіоелектроніки, комп'ютерних систем та інфокомунікацій
Кафедра комп'ютерних систем, мереж і кібербезпеки

Лабораторна робота №7
З дисципліни: «Технології DevOps»

Виконав:

студент 5 курсу групи №555 ім

Напряму підготовки

125 Кібербезпека та захист інформації

ст. Орлов Станіслав Валерійович

Прийняв:

к.т.н., доцент

Узун Дмитро Дмитрович

Харків, 2023

1. Git repo with initial DotNet Project
[stanislavorlov/todo_app_dotnet_back-end \(github.com\)](https://github.com/stanislavorlov/todo_app_dotnet_back-end)
2. Dockerfile for building ASP.Net Core Web App

```
Code Blame 21 lines (17 loc) · 722 Bytes Code 55% faster with GitHub Copilot

1 #See https://aka.ms/customizecontainer to learn how to customize your debug container and
2
3 FROM mcr.microsoft.com/dotnet/aspnet:7.0 AS base
4 WORKDIR /app
5 EXPOSE 80
6
7 FROM mcr.microsoft.com/dotnet/sdk:7.0 AS build
8 WORKDIR /src
9 COPY ["TodoAppWeb/TodoAppWeb.csproj", "TodoAppWeb/"]
10 RUN dotnet restore "TodoAppWeb/TodoAppWeb.csproj"
11 COPY . .
12 WORKDIR "/src/TodoAppWeb"
13 RUN dotnet build "TodoAppWeb.csproj" -c Release -o /app/build
14
15 FROM build AS publish
16 RUN dotnet publish "TodoAppWeb.csproj" -c Release -o /app/publish /p:UseAppHost=false
17
18 FROM base AS final
19 WORKDIR /app
20 COPY --from=publish /app/publish .
21 ENTRYPOINT ["dotnet", "TodoAppWeb.dll"]
```

Рисунок 1 – докер імедж веб аплікації ASP.Net Core

3. Docker-compose file for including Database resources/

[todo_app_dotnet_back-end](#) / [docker-compose.yml](#) 

Stanislav-Orlov_bdgh and Stanislav-Orlov_bdgh Added support of docker via docker compose

Code Blame 23 lines (21 loc) · 538 Bytes Code 55% faster with GitHub Copilot

```
1  version: '3.4'
2
3  services:
4    todoappweb:
5      image: ${DOCKER_REGISTRY-}todoappweb
6      build:
7        context: .
8        dockerfile: TodoAppWeb/Dockerfile
9      ports:
10       - 5000:80
11
12     db:
13       environment:
14         ACCEPT_EULA: "Y"
15         SA_PASSWORD: example_123
16       image: mcr.microsoft.com/mssql/server
17       restart: always
18       healthcheck:
19         test: ["CMD-SHELL", "/opt/mssql-tools/bin/sqlcmd -S localhost -U sa -P example_123 -Q 'SELECT 1' || exit 1"]
20         interval: 10s
21         retries: 10
22         start_period: 10s
23         timeout: 3s
```

Рисунок 2 – docker compose файл для об'єднання імеджу аплікації з базою даних

4. Building docker images

```
PS C:\Projects\Copilot\todo_app_dotnet_back-end> docker-compose build
[+] Building 16.2s (18/18) FINISHED
=> [todoappweb internal] load .dockerignore                                0.1s
=> => transferring context: 382B                                           0.0s
=> [todoappweb internal] load build definition from Dockerfile             0.0s
=> => transferring dockerfile: 781B                                        0.0s
=> [todoappweb internal] load metadata for mcr.microsoft.com/dotnet/sdk:7.0 0.4s
=> [todoappweb internal] load metadata for mcr.microsoft.com/dotnet/aspnet:7.0 0.0s
=> [todoappweb base 1/2] FROM mcr.microsoft.com/dotnet/aspnet:7.0         0.0s
=> [todoappweb build 1/7] FROM mcr.microsoft.com/dotnet/sdk:7.0@sha256:e882cf3bbeadd42cbe42ed5a76b06379564b285cc 0.0s
=> [todoappweb internal] load build context                               0.1s
=> => transferring context: 18.71kB                                        0.0s
=> CACHED [todoappweb build 2/7] WORKDIR /src                             0.0s
=> CACHED [todoappweb build 3/7] COPY [TodoAppWeb.csproj, TodoAppWeb/]     0.0s
=> CACHED [todoappweb build 4/7] RUN dotnet restore "TodoAppWeb/TodoAppWeb.csproj" 0.0s
=> [todoappweb build 5/7] COPY . .                                         0.2s
=> [todoappweb build 6/7] WORKDIR /src/TodoAppWeb                         0.4s
=> [todoappweb build 7/7] RUN dotnet build "TodoAppWeb.csproj" -c Release -o /app/build 11.1s
=> [todoappweb publish 1/1] RUN dotnet publish "TodoAppWeb.csproj" -c Release -o /app/publish /p:UseAppHost=false 3.6s
=> CACHED [todoappweb base 2/2] WORKDIR /app                             0.0s
=> CACHED [todoappweb final 1/2] WORKDIR /app                             0.0s
=> CACHED [todoappweb final 2/2] COPY --from=publish /app/publish .       0.0s
=> [todoappweb] exporting to image                                         0.0s
=> => exporting layers                                                    0.0s
=> => writing image sha256:31ca3eb4ecf45482359fd1803dc8fd594be2eb2b80bf834ab7714dfe0c6ae47 0.0s
=> => naming to docker.io/library/todoappweb                             0.0s
PS C:\Projects\Copilot\todo_app_dotnet_back-end>
```

Рисунок 3 – результат створення імеджів з докер компоуз файлу

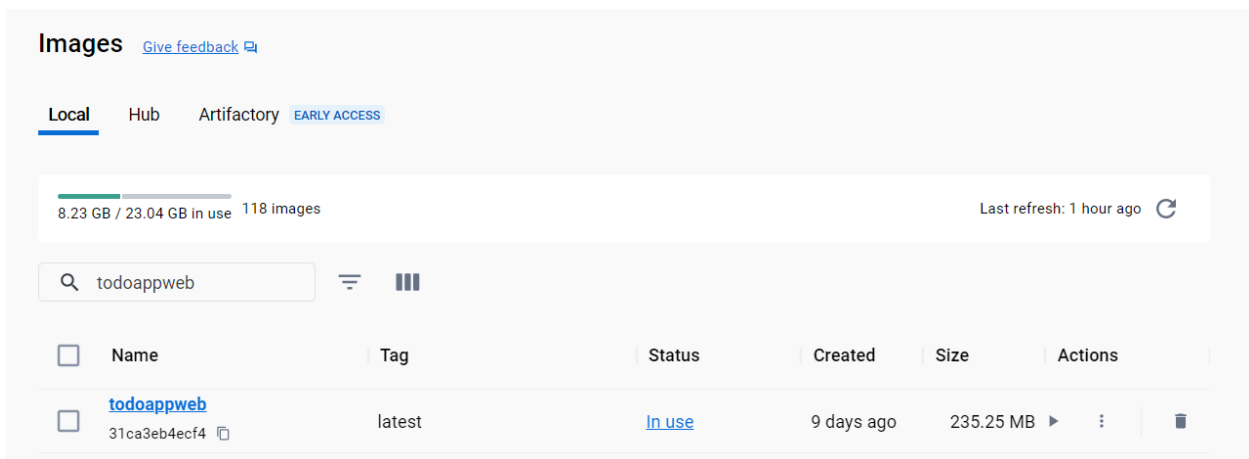


Рисунок 4 – перевірка створених імеджів через через Docker Desktop
аплікацію

```
PS C:\Projects\Copilot\todo_app_dotnet_back-end> docker images
REPOSITORY              IMAGE ID            CREATED             SIZE              TAG
ghcr.io/bd-infusion/ndle-db      98755c7dd97b      4 days ago         555MB             1.45.3
ghcr.io/bd-infusion/uis-mocktale-gateway-service    e653e4fa402c      7 days ago         200MB             1.11.15
ghcr.io/bd-infusion/platform-device-mock            d1211af444c5      7 days ago         200MB             1.11.15
ghcr.io/bd-infusion/platform-idm-mock               4e68532ed1d3      8 days ago         223MB             1.3.5
todoappweb                   31ca3eb4ecf4      8 days ago         235MB             latest
```

Рисунок 5 – перевірка створених імеджів через термінал

5. Running docker containers

```
PS C:\Projects\Copilot\todo_app_dotnet_back-end> docker-compose up
[+] Building 0.0s (0/0)
[+] Running 2/0
  ✓ Container todo_app_dotnet_back-end-todoappweb-1 Created 0.0s
  ✓ Container todo_app_dotnet_back-end-db-1 Running 0.0s
Attaching to todo_app_dotnet_back-end-db-1, todo_app_dotnet_back-end-todoappweb-1
todo_app_dotnet_back-end-todoappweb-1 | warn: Microsoft.AspNetCore.DataProtection.Repositories.FileSystemXmlRepository[
60]
todo_app_dotnet_back-end-todoappweb-1 | Storing keys in a directory '/root/.aspnet/DataProtection-Keys' that may
not be persisted outside of the container. Protected data will be unavailable when container is destroyed.
todo_app_dotnet_back-end-todoappweb-1 | info: Microsoft.Hosting.Lifetime[14]
todo_app_dotnet_back-end-todoappweb-1 | Now listening on: http://[::]:80
todo_app_dotnet_back-end-todoappweb-1 | info: Microsoft.Hosting.Lifetime[0]
todo_app_dotnet_back-end-todoappweb-1 | Application started. Press Ctrl+C to shut down.
todo_app_dotnet_back-end-todoappweb-1 | info: Microsoft.Hosting.Lifetime[0]
todo_app_dotnet_back-end-todoappweb-1 | Hosting environment: Development
todo_app_dotnet_back-end-todoappweb-1 | info: Microsoft.Hosting.Lifetime[0]
todo_app_dotnet_back-end-todoappweb-1 | Content root path: /app
```

Рисунок 6 – запуск контейнерів виконанням команди docker-compose

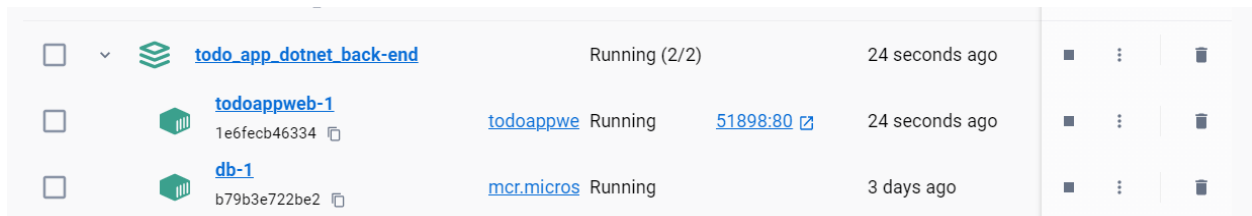


Рисунок 7 – перевірка запущених контейнерів через Docker Desktop
аплікацію

6. Accessing via HTTP and checking if everything is working fine

Navigate to <http://localhost:51898/>

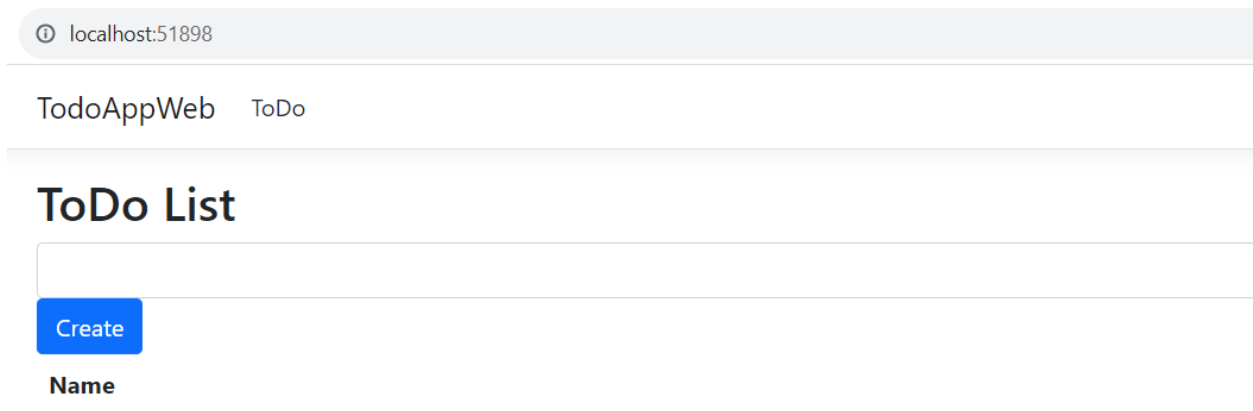


Рисунок 8 – перевірка запущеної аплікації через браузер

7. Checking docker logs

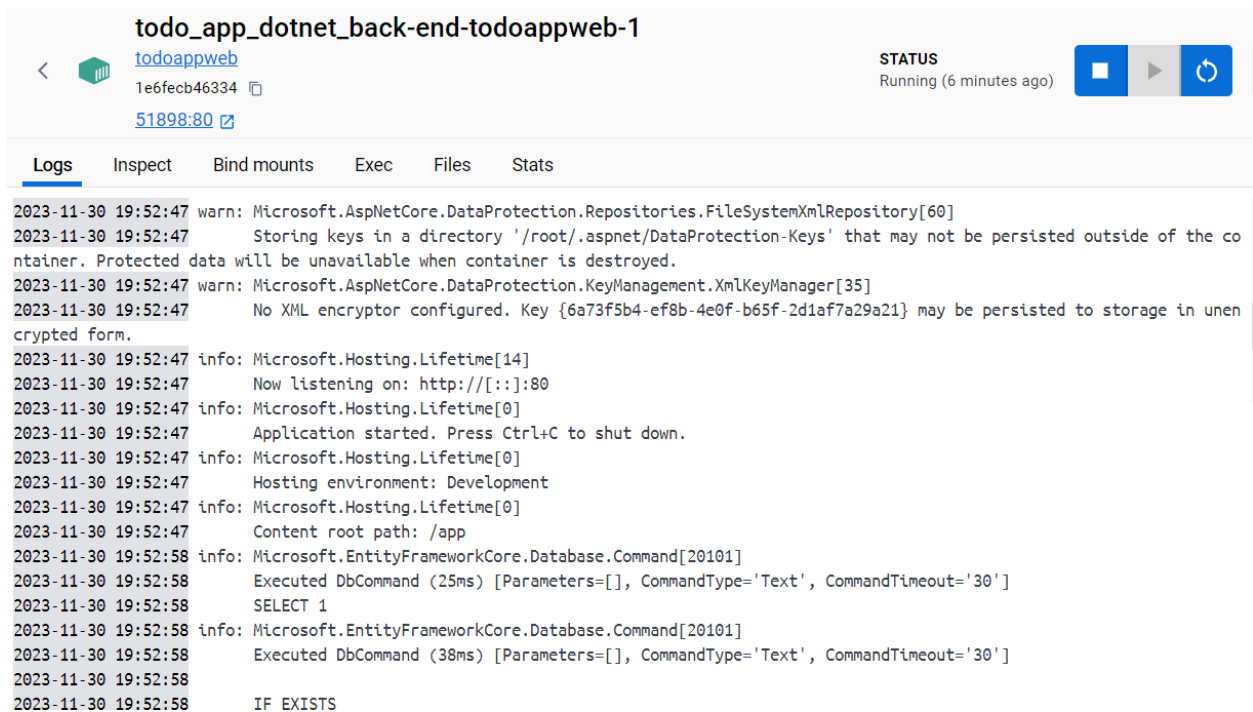


Рисунок 9 – перевірка логів аплікації

```
PS C:\Users\sorlo> docker logs -e 1e6fecb46334
PS C:\Users\sorlo> docker logs 1e6fecb46334
warn: Microsoft.AspNetCore.DataProtection.Repositories.FileSystemXmlRepository[60]
      Storing keys in a directory '/root/.aspnet/DataProtection-Keys' that may not be persisted outside of the container
. Protected data will be unavailable when container is destroyed.
warn: Microsoft.AspNetCore.DataProtection.KeyManagement.XmlKeyManager[35]
      No XML encryptor configured. Key {6a73f5b4-ef8b-4e0f-b65f-2d1af7a29a21} may be persisted to storage in unencrypted
form.
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://[::]:80
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: /app
info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (25ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
      SELECT 1
info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (38ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
```

Рисунок 10 – перевірка логів контейнера

Висновки

У ході виконання лабораторної роботи оволодів основними навичками роботи з системою контейнеризації через створення Dockerfile. Створив базовий веб застосунок та відповідний докерфайл. Побудував докер імедж та запустив його у докер контейнер.