Installation der WebAPI für die Antragsstellung per Amt24

Version: 2024-12-02

Bearbeiter: [pKlopfer@klopfer-software.de](mailto:pKlopfer@klopfer-software.de)

# Bereitstellung eines Dateiordners für die WebAPI:

Hier: c:\inetpub\WebAPIAmt24

# IIS: Hinzufügen einer WebSite:

Hier: WebAPIAmt24

Ein Bild, das Text, Schrift, Screenshot, Reihe enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Screenshot, Zahl, Schrift enthält.

Automatisch generierte Beschreibung

Es sollte anderer http-Port als der Standardport 80 verwendet werden (hier: 8080). Der gewählte Port muss innerhalb der Firewall (eingehende Verbindungen) freigegeben werden.

Als physischer Pfad ist der Dateiordner der WebAPI anzugeben.

# Windows Server: Installation des Frameworks .NET 6.0.1

Quelle: <https://github.com/dotnet/core/blob/main/release-notes/6.0/6.0.1/6.0.1.md> (dotnet-hosting-6.0.1-win.exe)

Bezeichnung: ASP.NET Core Runtime Hosting Bundle

Ein Bild, das Text, Screenshot, Schrift, Software enthält.

Automatisch generierte Beschreibung

# Bereitstellung der Dateien der WebAPI

Der Inhalt des Installationspaketes ist in den vorbereiteten Dateiordner (hier: c:\inetpub\WebAPIAmt24) zu kopieren (WebAPIAmt24\_Test2024-12-01.zip).

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte Beschreibung

In die Datei der Applikationseinstellungen (appsettings.json) sind die Verbindungsparameter für die Datenbank der Fischerei-Gebührenverwaltung einzutragen:

*"ConnectionStrings": {"FISCHE": "Data Source=KS30\\I2019;Initial Catalog=FISCHE;Integrated Security=True;Pooling=True;Min Pool Size=0;Max Pool Size=100;MultipleActiveResultSets=False;Encrypt=False;TrustServerCertificate=False;ApplicationIntent=ReadWrite" }*

(Hier: Datenbank FISCHE; SQL Server: KS30\I2019; Integrated Security)

# Test der installierten WebAPI

Die WebAPI dient als REST-Server für POST-Requests. Dafür stellt sie die Methode create bereit, die entsprechend der Website-Adresse aufzurufen ist: https://<IP>:<Port>/api/ozpuffer/create (hier: localhost:8080/api/ozpuffer/create).

Mit dieser Methode wird ein Datensatz in die Tabelle dbo.ozPuffer der Zieldatenbank geschrieben; dabei wird nur das Datenfeld ozpText nvarchar(max) übergeben, das den Inhalt einer json-Datei mit allen Datenfeldern und zugehörigen Dokumenten zu einem Antrag im Kontext der Gebührenverwaltung beinhaltet.

Des Weiteren wird für Testzwecke die Methode retrieve bereit gestellt. Damit können Datensätze aus der Tabelle dbo.ozPuffer der Zieldatenbank abgefragt werden.

Der Test kann mit einem Internet-Browser oder einem geeigneten Testtool (hier: https://reqbin.com/post-online) erfolgen.

POST-Request:

Ein Bild, das Text, Reihe, Schrift, Zahl enthält.

Automatisch generierte Beschreibung

Im Body des Requests wird der Inhalt einer json-Datei mit Antragsdaten übergeben (Test\_POST\_Beispielbody.txt)

GET-Request:

Ein Bild, das Text, Screenshot, Reihe, Schrift enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Screenshot, Reihe, Schrift enthält.

Automatisch generierte Beschreibung

Der Body der Antwort liefert den Inhalt der Tabelle dbo.ozPuffer der Zieldatenbank.