

Sauberfix API - Setup Guide

Voraussetzungen

- .NET SDK 9.0
- PostgreSQL Datenbank

Erste Schritte nach dem Clone

1. Konfiguration erstellen

Kopiere die Template-Datei und passe sie an:

```
cp appsettings.Template.json appsettings.json
```

Bearbeite `appsettings.json` und trage deine Datenbank-Credentials ein:

```
{
  "ConnectionStrings": {
    "DefaultConnection": "Host=localhost;Port=5432;Database=sauberfix;Username=YOUR_USERNAME;Password=YOUR_PASSWORD"
  },
  "JwtSettings": {
    "Key": "CHANGE_THIS_TO_A_SECURE_RANDOM_KEY_MIN_32_CHARS!"
  }
}
```

2. Datenbank erstellen

Erstelle die Datenbank (falls noch nicht vorhanden):

```
CREATE DATABASE sauberfix;
```

3. Anwendung starten

```
dotnet run
```

Die Anwendung wird automatisch: - Alle Datenbank-Migrations ausführen - Einen Admin-User anlegen (Username: `admin`, Password: `admin123`) - Auf Port 5000 starten

Login-Credentials

Nach dem ersten Start kannst du dich mit folgenden Credentials einloggen:

- **Username:** `admin`
- **Password:** `admin123`

Wichtig: Ändere das Admin-Passwort nach dem ersten Login!

API Endpoints

- `POST /login` - Login mit Username/Password, gibt JWT-Token zurück
- `GET /termine` - Liste aller Termine (Auth erforderlich)
- `POST /termine` - Neuen Termin erstellen (Auth erforderlich)
- `GET /kunden` - Liste aller Kunden
- `POST /kunden` - Neuen Kunden erstellen
- `GET /mitarbeiter` - Liste aller Mitarbeiter (Auth erforderlich)

Entwicklung

Für die Entwicklung kannst du auch `appsettings.Development.json` erstellen. Diese Datei wird nicht ins Repository committed.

Troubleshooting

Datenbank-Verbindungsfehler

- Prüfe, ob PostgreSQL läuft
- Prüfe die Connection String in `appsettings.json`
- Prüfe, ob die Datenbank existiert

Port bereits belegt

Falls Port 5000 bereits belegt ist, ändere in `appsettings.json`:

```
{
  "Kestrel": {
    "Endpoints": {
      "Http": {
        "Url": "http://*:ANDERER_PORT"
      }
    }
  }
}
```