**Documentația Proiectului ASP.NET Core**

**Numele Proiectului**

Proiect ASP.NET Core pentru Gestionarea Zborurilor și Biletelor

**Descriere**

Proiectul este o aplicație web dezvoltată în ASP.NET Core care permite gestionarea zborurilor, destinațiilor, pasagerilor și biletelor aferente. Aplicația oferă funcționalități de bază pentru efectuarea operațiunilor CRUD (Create, Read, Update, Delete) pentru fiecare entitate și leagă entitățile între ele pentru a urmări relațiile dinamic.

**Structura Proiectului**

Proiectul este structurat în următoarele componente principale:

1. **Modele (Models):**
   * **Bilet**: Model pentru biletul achiziționat, cu informații precum ID, ZborID, PasagerID, Clasa și Preț.
   * **Destinatie**: Model pentru destinație, cu informații precum ID, Nume, Țară și Aeroport.
   * **Pasager**: Model pentru pasager, cu informații precum ID, Nume, Prenume, Număr de Document, Email și Telefon.
   * **Zbor**: Model pentru zbor, cu informații precum ID, DestinatieID, Data și Ora Plecării, Data și Ora Sosirii și Status.
2. **CRUD**
   * + Se va realiza Crud-ul pentru fiecare model.
3. **Baza de Date (Database):**
   * Aplicația utilizează Entity Framework Core sau Dapper pentru comunicarea cu baza de date. Acesta este un punct de conectare la baza de date pentru stocarea datelor entităților.
4. **Interfața Utilizator (UI):**
   * Interfața utilizator este dezvoltată cu HTML, CSS și JavaScript și oferă pagini pentru vizualizarea și gestionarea datelor entităților. Paginile includ funcționalități de adăugare, editare și ștergere pentru fiecare entitate.
   * Pentru a adauga o imagine o sa adaug un nou folder Images in care o sa adaugam poza de backround .

**Instrucțiuni de Instalare și Utilizare**

Pentru a instala și utiliza aplicația, urmați acești pași:

1. **Configurare Mediu de Dezvoltare:**
   * Asigurați-vă că aveți instalat Visual Studio sau un alt IDE pentru dezvoltarea ASP.NET Core.
2. **Realizare Proiect**
   * Creati un proiect nou si alegeti ASP.net Core .
   * Realizati modele conform specificatiilor intr-un fisier Models .
   * Creati foldere pentru fiecare clasa din Models : Destinatie, Pasageri, Bilete si Zboruri. In fiecare folder click dreapta si apoi New-New Scaffolded Item si selectari modelul dorit . Aceasta comanda va realiza CRUD-ul automat .
   * Exemplu Crud : Pentru folderul Pasageri trebuie sa facem 5 fisiere de tipul New Scaffold , si anume : Index , Delete, Create, Edit si Details.
3. **Configurare Bază de Date:**
   * Configurați conexiunea la baza de date în fișierul **appsettings.json**.
   * Executați migrări pentru a crea baza de date și schemele asociate folosind comanda **Add-Migration InitialCreate Update-Database**
4. **Executare Proiect:**
   * Executați proiectul pentru a porni serverul local ASP.NET Core.
5. **Accesați Aplicația:**
   * Accesați aplicația într-un browser web la adresa **http://localhost:port**, unde **port** este portul specificat în configurarea serverului local.
6. **Utilizare Aplicație:**
   * Accesați paginile de gestionare a entităților (zboruri, destinații, pasageri, biletelor) din meniul principal.
   * Utilizați butoanele de adăugare, editare și ștergere pentru a efectua operațiuni CRUD asupra datelor.

**Tabel de Specificații**

**Entitate: Bilet**

* **Câmpuri:**
  + ID (int): Identificator unic pentru bilet.
  + ZborID (int): ID-ul zborului asociat acestui bilet.
  + PasagerID (int): ID-ul pasagerului asociat acestui bilet.
  + Clasa (string): Clasa biletului (exemplu: Economy, Business).
  + Pret (decimal): Prețul biletului.

**Entitate: Destinatie**

* **Câmpuri:**
  + ID (int): Identificator unic pentru destinație.
  + Nume (string): Numele destinației.
  + Tara (string): Țara în care se află destinația.
  + Aeroport (string): Aeroportul asociat destinației.
* **Relații:**
  + O destinație poate avea mai multe zboruri.

**Entitate: Pasager**

* **Câmpuri:**
  + ID (int): Identificator unic pentru pasager.
  + Nume (string): Numele pasagerului.
  + Prenume (string): Prenumele pasagerului.
  + NumarDocument (string): Numărul de document al pasagerului.
  + Email (string): Adresa de email a pasagerului.
  + Telefon (string): Numărul de telefon al pasagerului.
* **Relații:**
  + Un pasager poate deține mai multe bilete.

**Entitate: Zbor**

* **Câmpuri:**
  + ID (int): Identificator unic pentru zbor.
  + DestinatieID (int): ID-ul destinației asociate acestui zbor.
  + DataOraPlecare (DateTime): Data și ora plecării pentru zbor.
  + DataOraSosire (DateTime): Data și ora sosirii pentru zbor.
  + Status (string): Statusul zborului (exemplu: Programat, Anulat).
* **Relații:**
  + Un zbor are o destinație asociată.
  + Un zbor poate avea mai multe bilete.