**1. Configurarea Proiectului**

- Asigurați-vă că ați creat un proiect ASP.NET MVC în Visual Studio

- Asigurați-vă că ați instalat pachetele necesare pentru ASP.NET MVC, inclusiv `Microsoft.AspNetCore.Mvc`.

**2. Definirea Modelului (`Student`)**

Modelul `Student` definește structura datelor pentru un student. Acesta conține atribute care vor fi afișate în formular și reguli de validare. Puteți adăuga și alte atribute, dacă este necesar.

**3. Crearea Serviciului (`StudentService`)**

Clasa `StudentService` servește ca o simulare a unei baze de date în memorie. Într-o aplicație reală, veți folosi o bază de date sau altă sursă de date persistentă.

- Adăugați metode pentru adăugarea, actualizarea, ștergerea și obținerea studenților în funcție de IDNP.

*- Înlocuiți simularea de bază de date cu accesul la o bază de date reală, folosind Entity Framework sau altă tehnologie de acces la date.*

**4. Configurarea Controlerului (`HomeController`)**

Controlerul `HomeController` gestionează cererile și rutează către acțiunile corespunzătoare.

- Creați acțiuni pentru vizualizarea paginilor: `Index`, `Register`, `Edit`, `StudentList`, `Update`, `Delete`.

- Utilizați anotări `[HttpGet]` și `[HttpPost]` pentru a specifica metodele HTTP acceptate.

- Implementați logica pentru gestionarea cererilor, inclusiv validarea datelor și interacțiunea cu serviciul `StudentService`.

- Utilizați returnările `View` pentru a afișa pagini HTML și returnările `RedirectToAction` pentru a redirecționa utilizatorii la alte acțiuni după o operație reușită.

**5. Configurarea Rutei**

În `Program.cs`, configurați rutele pentru aplicație folosind `app.MapControllerRoute`. Aceasta asociază URL-urile cu acțiunile controlerului.

**6. Crearea Șabloanelor de Vizualizare**

Șabloanele de vizualizare (Views) sunt fișiere .cshtml care definesc modul în care datele sunt afișate în interfața de utilizator. Aveți patru șabloane principale:

- `Add.cshtml`: Formularul pentru adăugarea unui student. Utilizează `Html.BeginForm` pentru a crea un formular HTML și `Html.TextBoxFor` pentru a genera câmpurile de intrare.

- `Index.cshtml`: Pagina principală a aplicației. Aici puteți afișa informații generale despre aplicație sau navigația.

- `StudentList.cshtml`: Afișează o listă de studenți. Utilizează o buclă `foreach` pentru a parcurge studenții și afișa datele într-un tabel.

- `Update.cshtml`: Formularul pentru editarea unui student existent. Similar cu `Add.cshtml`, dar cu date pre-încărcate pentru a fi modificate.