Check Inn – Platforma web pentru rezervari online si gestionare hotelului

Pentru a dezvolta aceasta platforma am ales tipul de aplicatie ASP .NET Web Application din cadrul ecosistemului .NET folosind arhitectura MVC. Pentru parte de autentificare si autorizare cu roluri pentru utilizatori, am folosit "conturi individuale" (Individual Accounts), un tip de autentificare a carei functionalitate este in mare parte furnizata de catre ecosistemul .NET insa care poate fi implementata in diferite moduri in functie de nevoile aplicatie dezvoltate.

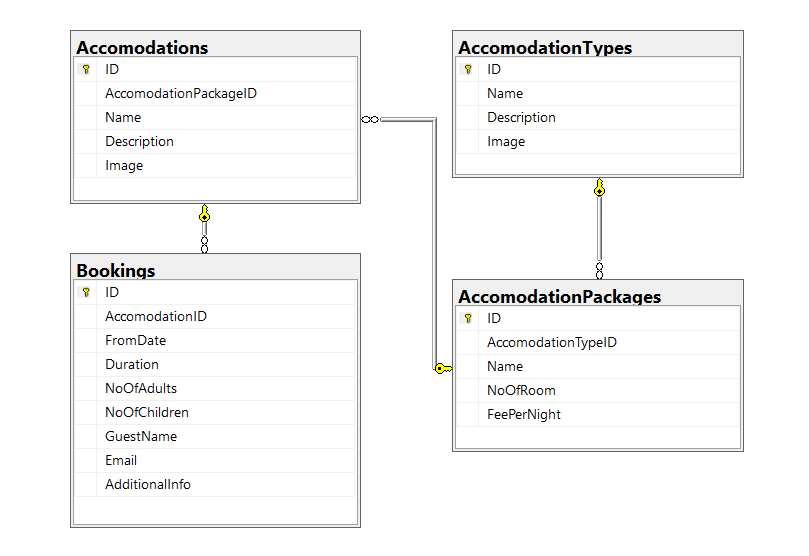
1. Conceperea bazei de date si interactiunea cu aceasta

Am creat o baza de date manual in MS SQL Server pe care mai apoi am adugat-o ca si “DefaultConnection” in fisier Web.Config:

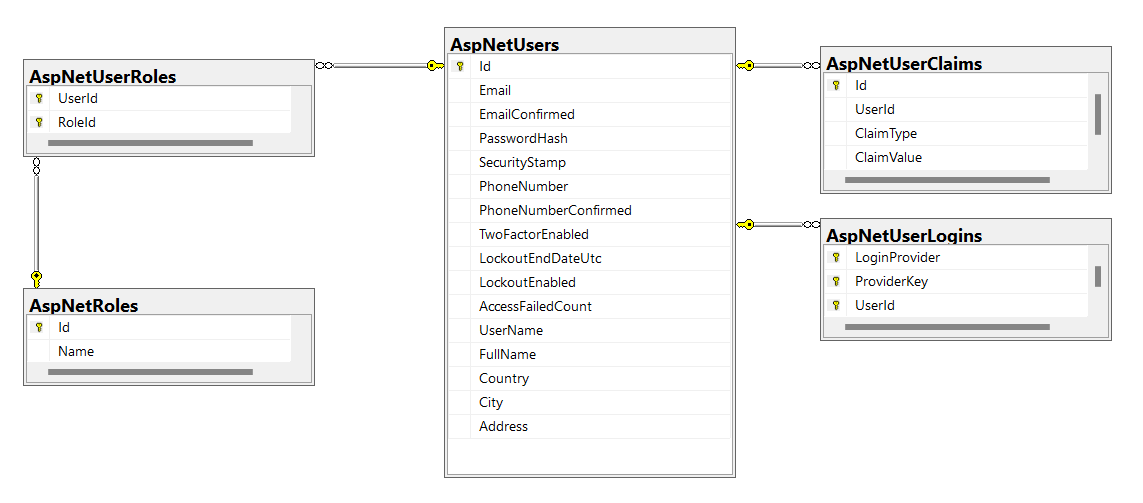
<connectionStrings>  
 <add name="DefaultConnection" connectionString="Data Source=AN515-45\ MSSQLSERVER01;Initial Catalog=Check\_Inn;Integrated Security=True" providerName="System.Data.SqlClient" />  
 </connectionStrings>

Schema bazei de date cu care lucreaza aplicatia este urmatoarea:

* schema tabelelor special concepute pentru aplicatie



* schema tabelelor generate pentru tip de autentificare "Individual Accounts"



Pentu interactiunea cu baza de date am instalat dependinta EntityFramework prin managerul de dependinte NuGet Package Manager.

Install-Package EntityFramework

Am creat un nou folder numit "Data Access Layer" (DAL) in care mi-am definit propria conexiune cu baza de date.

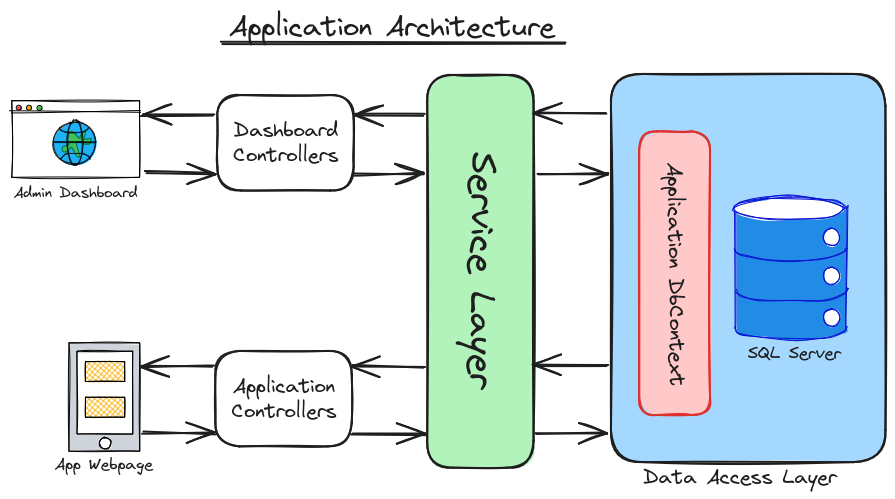
Enable-Migrations -ContextTypeName Check\_Inn.DAL.CheckInnContext

Add-Migration BasicEntities

Am implementat propria entitate “User” pentru a putea stoca informatile legate de utilizatori in baza de date creata manual in SQL Server, am actualizat codul sursa curent pentru a folosi aceasta entitate si nu cea standard. Totodata am creat si un fisier nou de migrari pentru a aseza structura tabelelor ce au legatura cu utilizatorul prin comanda urmatoare:

add-migration identityframeworkreference -verbose

Schema funcţionarii aplicaţiei (User / Admin lifecycle)

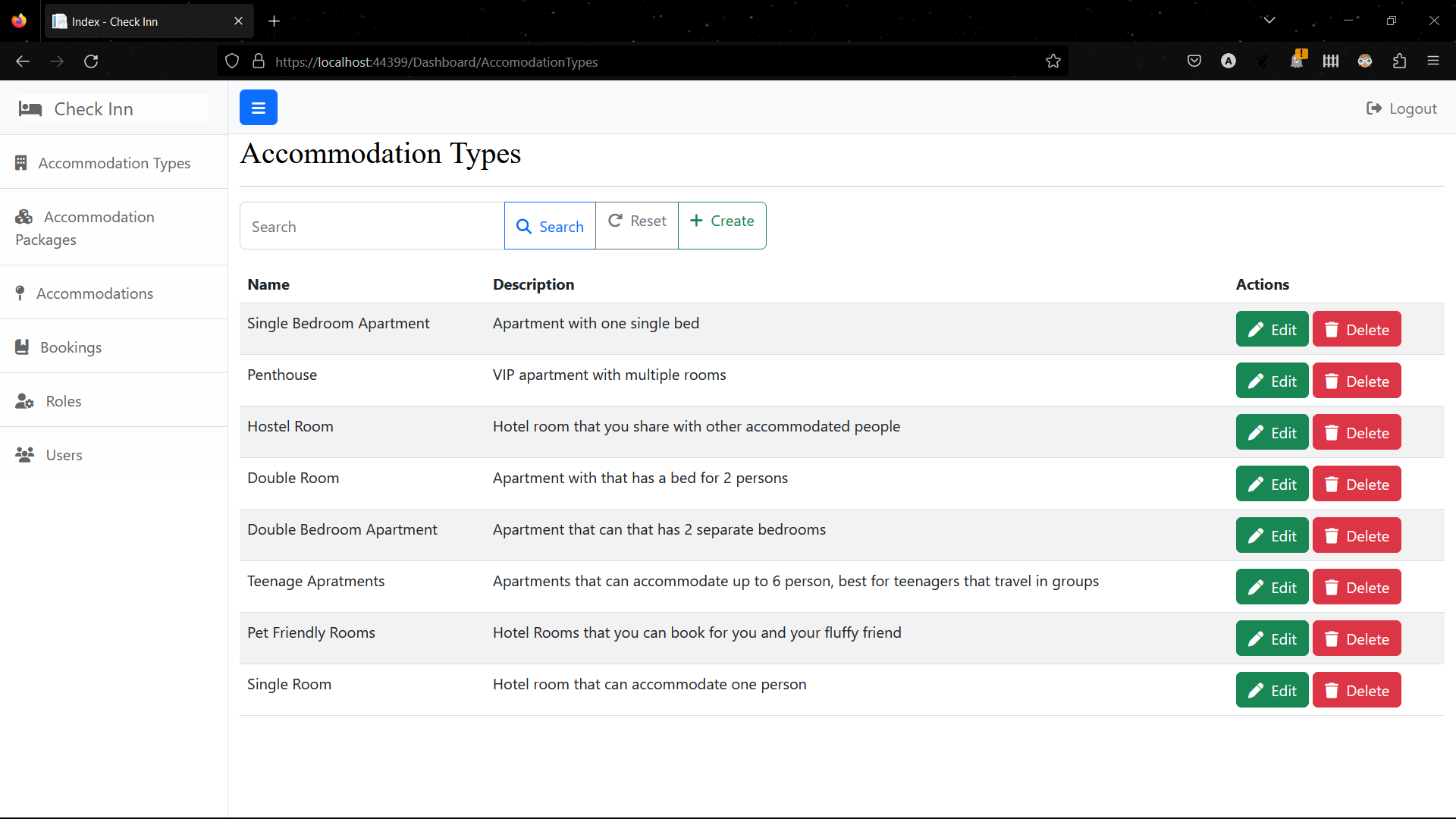


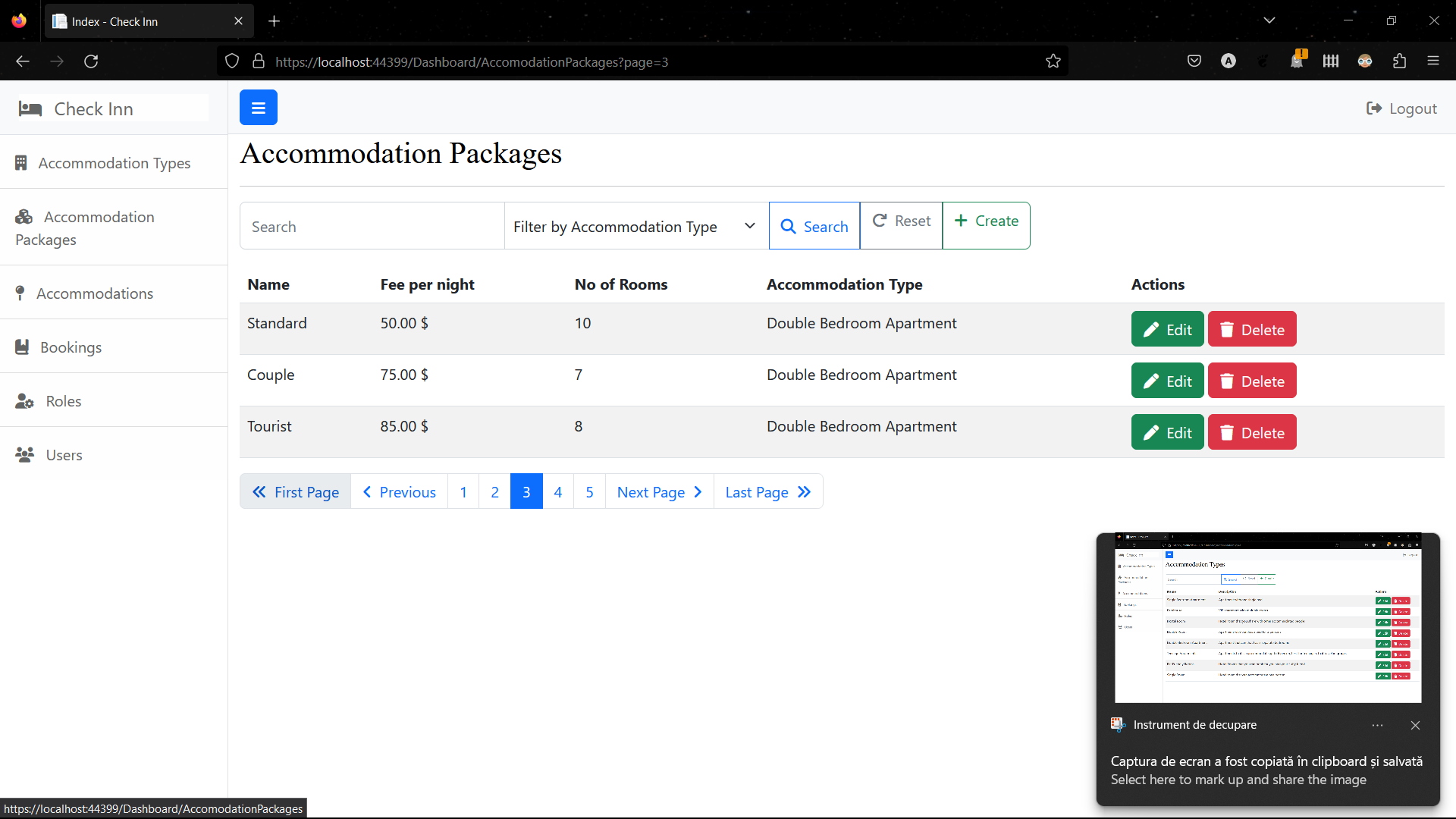
Functionalitatiile de baza ale aplicatiei

Am creat un nou spatiu in cadrul proiectului (Area) pentru dashboard-ul la care au aces administratori. Acest folder din proiect incorporeaza functionalitatea necesara dashboard-ului sparand-o de cea a aplicatiei principale. Acest lucru l-am realiza prin crearea unui nou "Scaffoldaed Item" si selectand "MVC 5 Area".

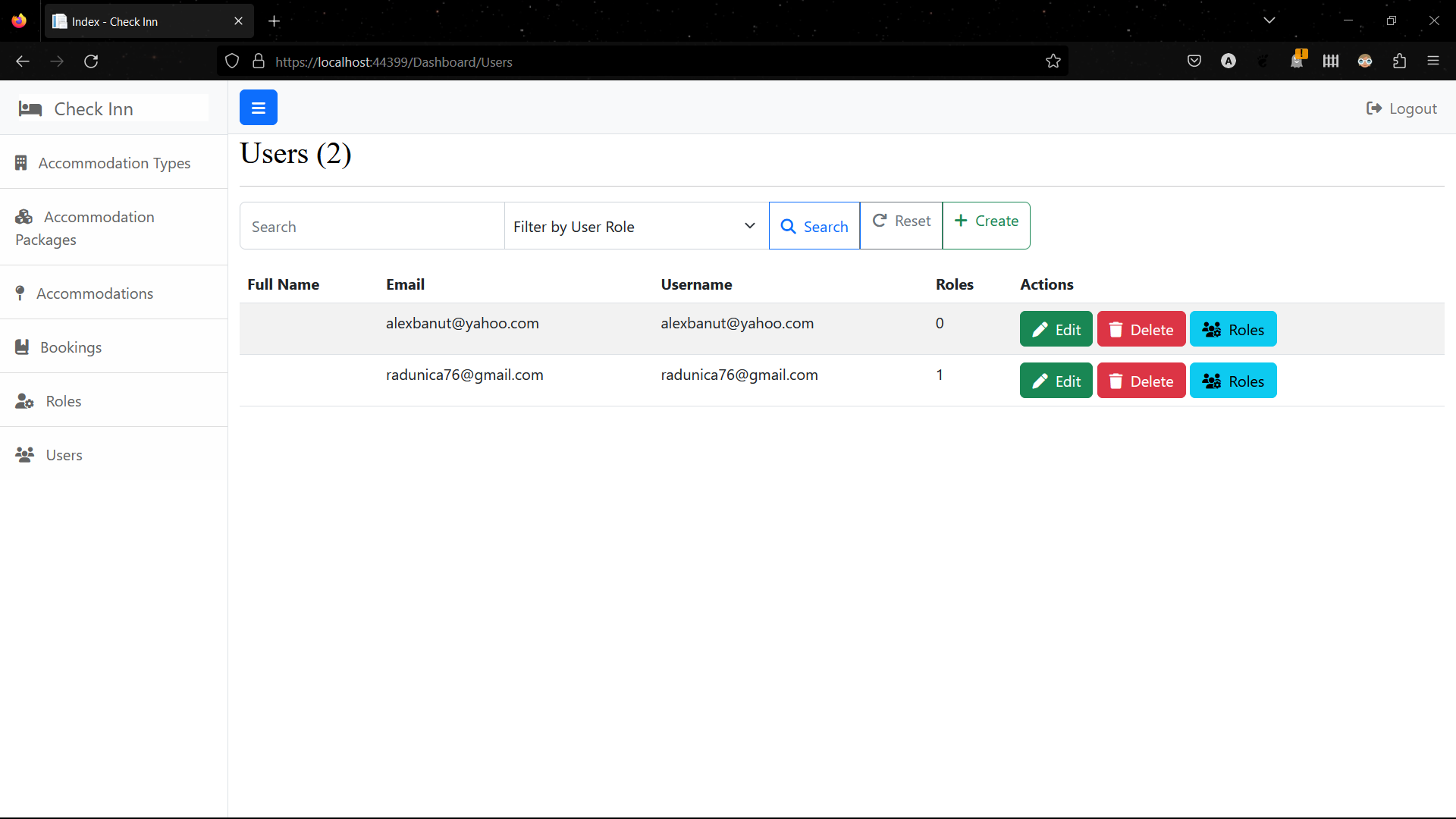
In cadrul dashboard-ului de admin am editat vederea Master a acestuia pentru a incorpora un meniu de navigare printre diferitele pagini ce faciliteaza managementul resurselor din cadrul aplicatiei si una de navigare folosind un template boostrap (<https://startbootstrap.com/template/simple-sidebar>)

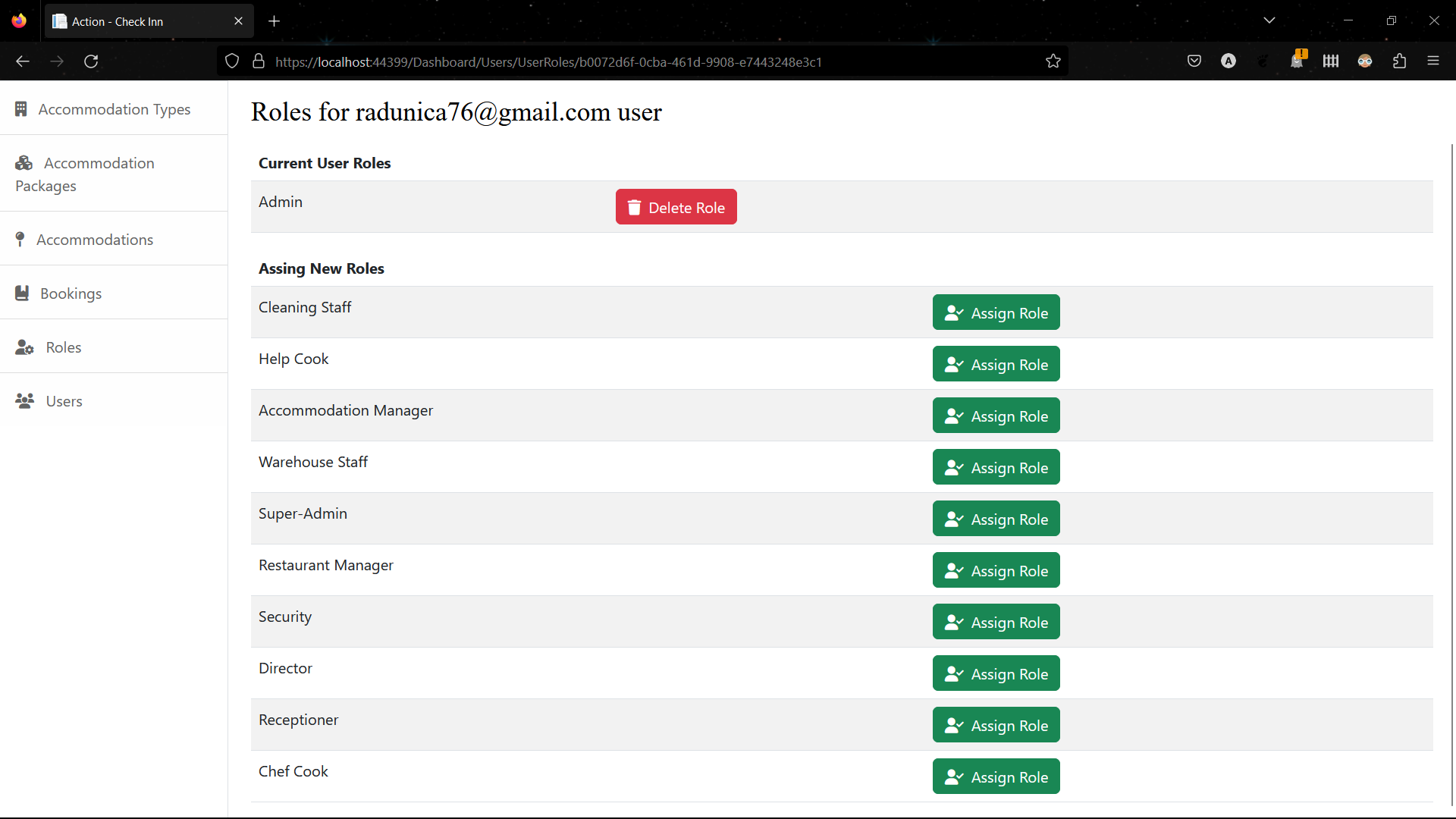
Pentru fiecare entitate am creat o intrare in dashboard care duce catre index-ul controller-ul fiecarei entitati, iar din index-ul repsectiv se pot filtra, cauta, edita sau sterge din inregstrarile entitatii.



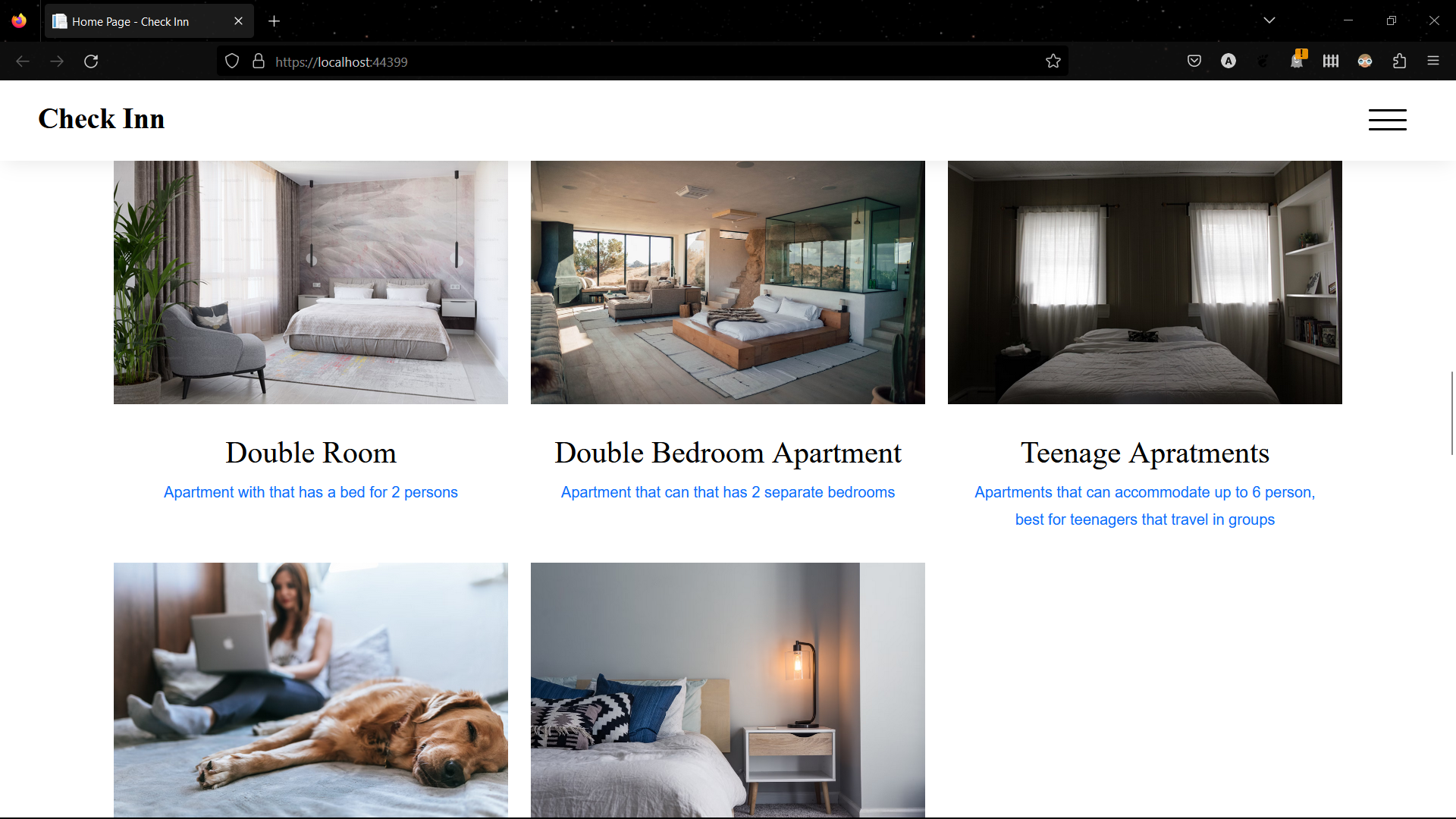


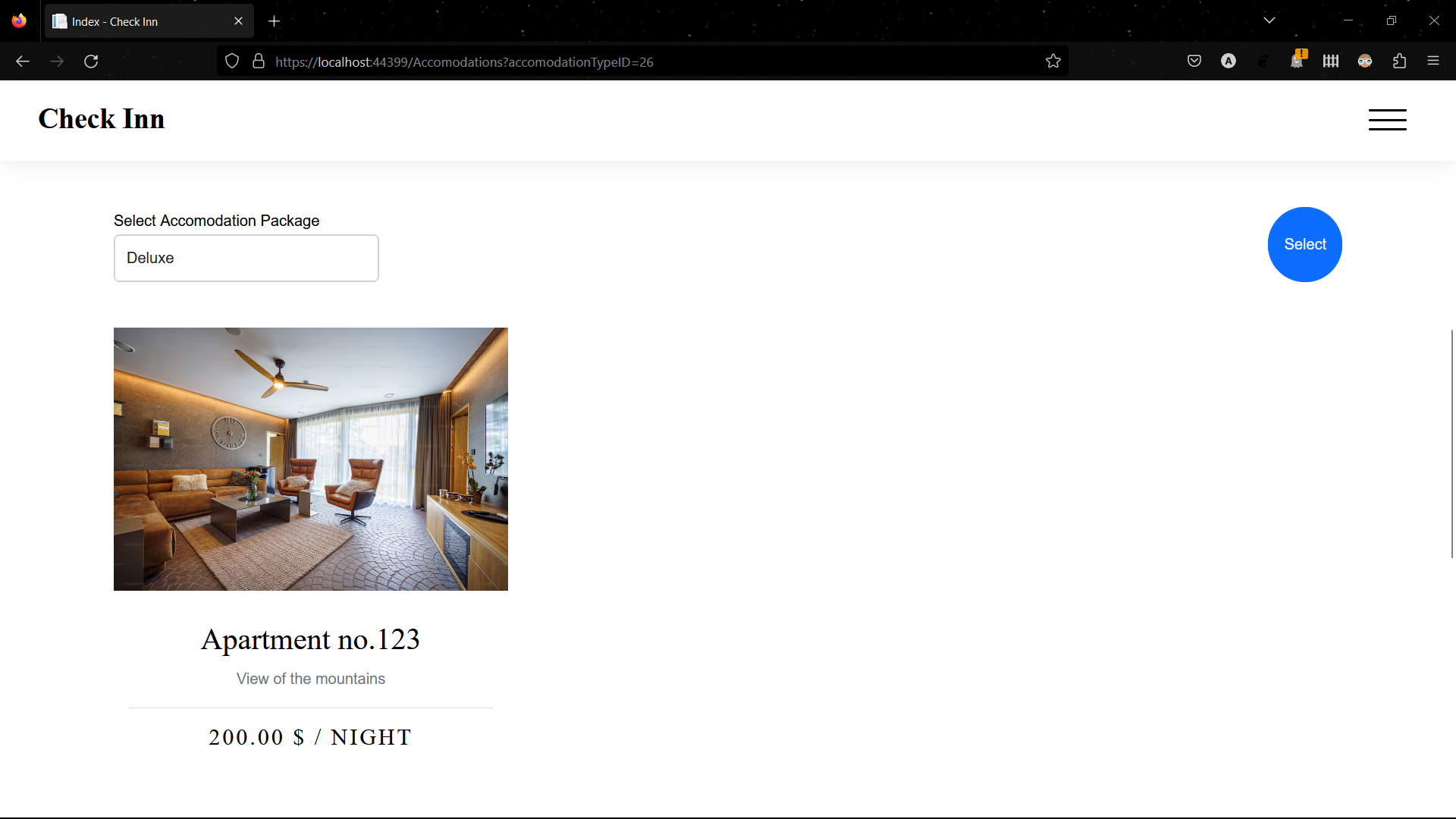
Pentru partea de autentificare si autorizare am folosit Identity Framework, iar pe in baza entitatilor User si Identitilor si claselor de tip service UserManager, SignInManager se pot inregistra si autentifica utilizatori. De asemenea din dashboard-ul admin se pot atribui sau sterge roluri pentru un anumit utilizator. Acestea pot fi folositoare pentru a permite accesul unor utilizator care au un anumit rol doar in unele pagini din dashboard astfel avand acces restricitonat la anumite resure si doar utilizatorul cu rolul de admin poate accesa toate resursele din dashboard.





Utilizatori pot vedea din pagina princiala a platformei toate tipurile si pachetele de cazare care se fac disponibile din dashboard-ul de admin. De asemnea cazarile se pot selecta dupa tipul de cazare si se pot filtra in functie de pachetul de cazare.





Rezervarea unei cazari

