WEBBUTIK FRONTEND

Entity Framework For The Win!

MARS 2021

NET20D

Inlämning 2, Marcus Medina

Inlämning 2 – Entity Framework FTW!

Innehåll

Beskrivning av projektet

En kund som säljer begagnade böcker behöver en webbshop. För att snabba upp skapandet av projektet har hon outsourcat projektet i två delar. En back-end och en front-end.

När back-end delen var klar visade det sig att front-end utvecklaren åkt på semester och inte gjort jobbet. Så nu får du äran att göra det jobbet med. Så är det att vara full-stack, man åker på allt jobb.

Du ska alltså göra ett frontend i antingen Consolen, Windows Forms eller ASP.net. Du väljer!

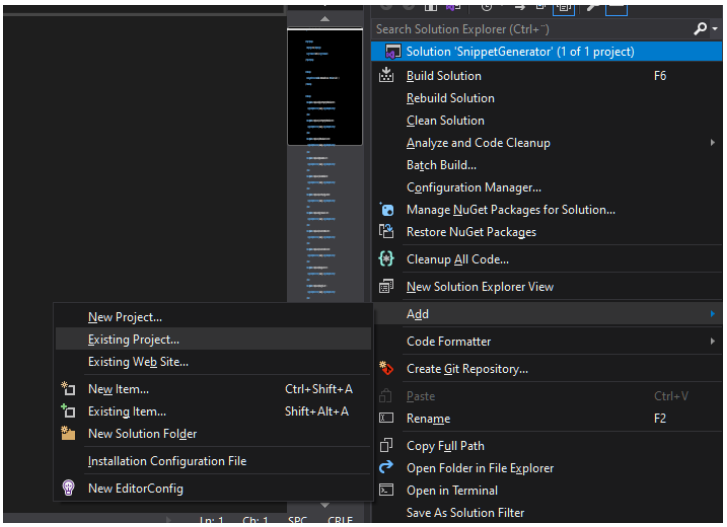
Vad behövs för projektet ska fungera?

Du ska använda dig av din **WebbShopAPI**

Först skapar du ett projekt

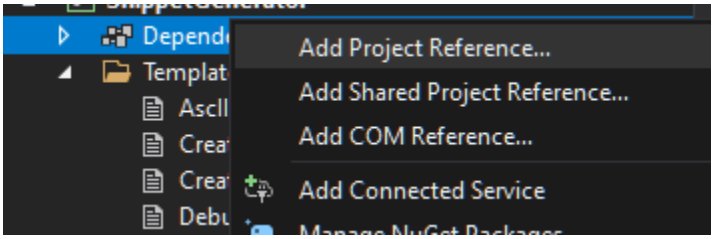
Flytta du ditt förra projekt så att den ligger parallellt med ditt nya projekt. Detta är viktigt för att vid inlämning ska du skicka in båda två i en zipfil.

I ditt nya projekt ska du sedan importera ditt förra projekt genom att klicka på din Solution Explorer och välja Add och sedan Existing project.



Leta upp ditt projekt och öppna den. Den kommer nu att läggas till i din Solution.

Nu ska du välja Add project reference.



Och den kommer att visa ditt tidigare projekt med en liten kryssruta. Kryssa i rutan och tryck OK.

Nu kan du använda namespacet från ditt förra projekt och anropa ditt APIklass.

```
var api = new WebbShopAPI();
```

Varning

Ett varningens ord, om dina models eller ditt API klass är annat än public så kommer de inte att fungera. Då måste du ändra i ditt förra projekt. Annars ska det förra projektet lämnas orört.

Nu ska du bygga ett program som använder sig av enbart APIn för att låta dina kunder handla böcker.

API

API ska förse användarna med följande funktioner

Function	Parameter	Returns	Action
Login	Username, password	User Id if success Nothing if fail	Loggar in användare om IsActive är true. Sätter SessionTimer till DateTime.Now
Logout	User Id		Sätt SessionTimer till DateTime.Default
GetCategories		List of all categories	
GetCategories	Keyword	List of categories matching keyword	
GetCategory	Category Id	List of books in category	
GetAvailableBooks	Category Id	List of books with amount>0	
GetBook	Book Id	Information about book	
GetBooks	Keyword	List of matching books	
GetAuthors	Keyword	List of matching books	
BuyBook	User Id Book Id	True if book purchase is OK	Kolla Session Timer Fail om user inte finns Fail om amount < 1 Kopierar bokdata till soldbooks, fyller på med datum och användarId Minskar bokantal med 1
Ping	User Id	"Pong" if customer is online string.empty if customer is offline,	Ping för att hålla anslutningen vid liv, Sätter SessionTimer till DateTime.Now
Register	Name Password PasswordVerify	True if user is created False is user already exist False is password != verify	Skapar en kund Sätter IsActive = true Sätter IsAdmin = false

Vid varje anrop med UserId/AdminId uppdateras den inloggade användarens SessionTimer till DateTime.Now, men om den inte uppdaterats på över 15 minuter kommer användaren att anses som utloggad och då kommer denne inte längre att kunna handla böcker förrän inloggning har gjorts igen.

Administrator access to API

Inloggad administratör (User.IsAdmin==true) får också tillgång till andra funktioner

AdminID är den inloggade admins User Id egentligen

Function	Parameter	Returns	Explanation
AddBook	Admin Id Title Author Price Amount	True if success False if fail	Öka book.amount om boken redan finns, annars sätt book.amount till antal som skickades in adminId är den inloggade användarens Id Sätter SessionTimer till DateTime.Now
SetAmount	Admin Id Book Id Amount	Amount of books now	Sätter SessionTimer till DateTime.Now Sätter bokens amount till givna värdet
ListUsers	Admin Id	List of users	Sätter SessionTimer till DateTime.Now
FindUser	Admin Id Keyword	List of matching users	Sätter SessionTimer till DateTime.Now
UpdateBook	Admin Id Id Title Author Price	True if success False if fail	Sätter SessionTimer till DateTime.Now
DeleteBook	Admin Id Book Id	True if success False if fail	Minska book.amount, om den blir 0 radera boken Sätter SessionTimer till DateTime.Now
AddCategory	Admin Id Name	True if success False if fail	Sätter SessionTimer till DateTime.Now
AddBookToCategory	Admin Id Book Id Category Id	True if success False if fail	Sätter SessionTimer till DateTime.Now
UpdateCategory	Admin Id CategoryId, Name	True if success False if fail	Sätter SessionTimer till DateTime.Now
DeleteCategory	Admin Id CategoryId	True if success False if fail	Fails om kategorin inte är tom Sätter SessionTimer till DateTime.Now
AddUser	Admin Id Name Password	True if success False if fail	Fails om user redan finns Fails om lösenord är tom Sätter SessionTimer till DateTime.Now

VG Funktionalitet

Function	Parameter	Returns	Explanation
SoldItems	Admin Id	Lists all sold books	Sätter SessionTimer till DateTime.Now
MoneyEarned	Admin Id	How much money was earned by sold books	Sätter SessionTimer till DateTime.Now
BestCustomer	Admin Id	Customer that bought most books	Sätter SessionTimer till DateTime.Now
Promote	Admin Id User Id	True if success False if fail	Fail om user inte finns Sätter SessionTimer till DateTime.Now
Demote	Admin Id User Id	True if success False if fail	Fail om user inte finns Sätter SessionTimer till DateTime.Now
ActivateUser	Admin Id User Id	True if success False if fail	Sätter SessionTimer till DateTime.Now
InactivateUser	Admin Id User Id	True if success False if fail	Sätter SessionTimer till DateTime.Now

Inlämningskrav

Krav för godkänd

1. Clean Code
2. Frontend ska vara baserad på MVC
3. Seeder från förra projektet ska anropas så att Grunddata skapas
4. Detta projekt får inte använda databaskontexten, enbart APIn ska anropas
5. Programmet ska förse användaren med ett kraschfritt menysystem
6. Båda projekten måste finnas med i Zipfilen som lämnas in
7. UML Diagram

Krav för väl godkänd

1. Att alla krav för godkänd uppfylls
2. XML kommentarer i alla klasser
3. VG funktionalitet som är implementerad från förra projektet används

Disclaimer

Även om projektet är tänkt att vara realistisk har jag minskat ner funktionaliteten, så att lösenord inte behöver krypteras, sessionsvariabler skippas, kundvagn och betalning hoppas över och böckerna har bara en kategori.

Dessa funktioner kan ni själva bygga in efter inlämning om ni känner att det ger en bra grund för ett riktigt projekt.

Fokus här ligger på att använda det förra projektet och Linq sökningar.