Laborationsrapport

Entity Framework Core in ASP.NET Core MVC
DT191G, Webbutveckling med .NET

Författare: Nick Kushkbaghi, niku2001@student.miun.se
Termin, år: VT 2022

Mittuniversitetet
MID SWEDEN UNIVERSITY

1 Sammanfattning

I den här uppgiften besvaras frågor om ASP .NET historia och olika versioner och grundläggande information kunskap om olika begrepp och tekniker inom ASP.Net Core. Frågorna besvaras via kursanvisningar, föreläsningar samt trovärdiga källor från internat.

2 Innehållsförteckning

3 Frågor

3.1 Till vilka olika typer av databaser går det att ansluta från ASP.NET Core?

- SQL Server: som är Microsoft eget server
- SQLLite: är en filbaserade databas.
- MongoDB: är en NoSQL databas
- MySQL/MariaDB: är SQL databser
- Oracle: stödjer SQL/PL och SQL
- PostgerSQL
- Och många fler [1]

3.2 Beskriv vad Entity Framework och Entity Framework Core. Inkludera dess syfte samt histora fram till nu.

Entity Framework: kallas O/RM (Object Relational mapper) är ett ramverk som skapas, utvecklas och publiceras av Microsoft, första version av ramverket slämpts 2008 för Visual Studio Code. "Code First" som var ramverket ny teknik publicerade med i version 3 år 2011, år 2012 kommer "migrations" teknik och år 2016 inkluderades ramverket EF Core 1.0 som i senaste versionen uppgraderades till EF Core 6.0 dessa senaste två versioner är oberoende av tidigare versioner och mer information om dem kommer i nästa punkt. Syftet med användning av ramverket är att underlätta anslutningar mot databaser.

Entity Framework Core: är ren minifierad, modulär ramverkt och skillnad med Entity Entity Framework är att ramverket kan exekveras på alla plattformar. Syftet med ramverket är att med hjälp av OR mappar utvecklare slipper skriva kod för data-access, kunna jobbar med flesta Databas Management Systems, stödjer" Code First", "Database first" (reverse enginering) och NuGet som gör det möjligt att använda diverse paket i projektet. [1]

3.3 Beskriv och förklara begreppen och skillnaderna mellan "model first" och "database first" gällande databas-anslutning med Entity Framework.

Databs first: används för ett projekt som redan har en databas och i vanligt sätt man som webbutvecklare ska omvandlar databasers tabeller till diverse modeller som ska sedan konsumeras av applikationen. Det som databs first gör är att modeller generas av tabeller som redan finns i databasen som man som utvecklare slipper att skapa modeller för var och en tabell. [1] [2]

Model first/Code first: att skriva modeller först och generar databasen från den, till exemepl vi skapa en modell som består av tre element och vid användning av Code First skapas en tabell som har samma information som modellen/class och samma namn och antal kolumner som funktioner i modellen innehåller, hela processen sker automatiskt [1]

3.4 Förklara stegen för att skapa en databas samt ett generera CRUD-interface (CRUD = Create Read Update Delete) med hjälp av "schaffolding" i ASP.NET Core / Entity Framework Core för en befintling modell.

För att skapa databsen från modellen som redan finns i programmet krävs inställningar av diverse paket från Nuget, exempelvis *EntittyFramworkCore- SQLLite, SQLServer* och *Design, VisualStudio.Web.CodeGeneration.Design* samt skapa klassen som motsvarar tabeller i databasen.

Efter dess skapas en databas *context class* som hanterar anropet till databasen. För att registrera databas *context* i program.cs filen lägga till *context* som en *dependency injection* som en *builderservice* kodblock, som visa till programmet vilket typ av databas ska kopplas till programmet och databas informationen inhämtas som en *conectionString* från *appsetting.json* file.

När kopplingen för databasinställningar är färdigt läggs *"Scaffolded"* till programmet som hantera *CURD* metoder. För att skapa databasen används av "migration" och föra tt har tillgång till den för installera verktyg för *dotnet-ef* i terminallen samt ska döpa migrationen via en *dotnet ef add momment* i termainalen samt uppdatera databasen i terminalen via *database update* kommando. Efter detta skapa sen mapp som heter migration och innehåller databastabellens informationer.

3.5 Entity Framework Core har stöd för något som kallas "Raw SQL Queries". Beskriv vad detta är.

För att kunna skriva *SQL* anrop direkt till *en SQL* databasen i stället för *ASP* metoder används av *Raw SQL Queries* exempelvis: *.SqlQuery("SELECT * FROM tabellnamn"),* där all information från *tebellnamn* i kopplade databas hämtas.[3]

3.6 Förklara klassen DbContext i EF Core. Vad är dess syfte och vad innehåller den för funktionalitet?

DbContext kan representeras som en session med databas [4] klassen som innehåller diverse metoder och syftet med metoderna är att hantera koppling till databasen, justera och konfigurera modell och relationer, skicka fråga till databasen, lagrar data i databasen samt uppdatera data i databasen m.m. [5]

3.7 Hur kan vi visa en annan tabell eller kolumnnamn än de som definieras i klassen?

Genom att använda "Data Annotations- Column Attribute in EF 6 & EF Core" kan man visa ett annat namn för kolumn exempelvis genom att skriva [Column("klumnNamn")] ovanför klass metoder/attribut samt skriva infoga [Table("tabellnamn")] ovanför klassen visar ett annat databasnamn. [6]

4 Slutsatser

Det finns många nya begrepp och teknik i ASP .Net som kräver mycket tid för att veta hur och när ska dessa tekniker användas, men det är spännande att jobba och lära mer om ASP.net och diverse ramverk som användas med den.

5 Källförteckning

[1] Mittuniversitetet, " Databas-anslutningar med Entity FrameWork Core"https://elearn20.miun.se/moodle/mod/resource/view.php?id=881775 .Hämtad 2022-02-11.

[2] Microsoft"codeFirst to New Database" https://docs.microsoft.com/en-us/ef/ef6/modeling/code-first/workflows/new-database Hämtad2022-02-13.

[3] Entity Framework, "Raw SQL Queries". https://entityframework.net/raw-sql-queries Hämtad 2022-01-13.

[4] Microsoft, " DbContext Class" https://docs.microsoft.com/en-us/dotnet/api/microsoft.entityframeworkcore.dbcontext?view=efcore-6.0

[5] Entity Framework Tutorial, "Entity Framework Core: DbContext" https://entityframework.net/raw-sql-queries Hämtad 2022-01-13

[6] Entity Framework Tutorial , "Data Annotations - Table Attribute in EF 6 & EF Core", https://www.entityframeworktutorial.net/code-first/table-dataannotations-attribute-in-code-first.aspx Hämtad 2022-02-22.