Dental Application

Dental Application allows users to Create, Read, Update and Delete a Dentist or Patient. It allows to search a dentist/patient, look at the patients under a dentist and to find out who is the assigned dentist to the patient.

Technologies

ASP.NET Web Api2, MVC 5, C#, Entity Framework with Code First Approach (EF6) and Migrations Enabled, Unity framework and token based authentication via OWIN .

Application Concerns

1. Entity framework creates DentalAppDB database, Dentists, Patients and Users tables at the beginning of the application.
2. To compile the DentalApplication project, please make sure the Entity Framework, Unity framework are installed to avoid compilation errors in the DentalApplication project.

Compile Application - DentalApplication

1. From the project folder, open DentalApplication folder and click on DentalApplication.sln file to open in Visual Studio.
2. Under References, please check if any of the references are missing. If missing, right click on References -> Manage NuGet package manager and install the missing.
3. In web.config file, change the database server name to your server as shown below:

<connectionStrings>

<add name=" DentalAppContext " connectionString="Data Source=[ServerName];Initial Catalog=DentalDB;User ID=[UserID];Password=[Pwd];Integrated Security=false;App=EntityFramework" providerName="System.Data.SqlClient" />

</connectionStrings>

1. Migrations are enabled since Code First approach is used. Please check if the ‘Migrations’ folder with Configuration.cs and InitialCreate.cs files exists under the project.
2. If Migrations Enabled, Open Tools -> NuGet Package Manager -> Package Manager Console window, enter the following command:

PM> Update-database

This will create “DentalAppDB” and tables “dbo.Dentists”, “dbo.Patients”, “dbo.Users” on the mentioned SQL Server with initial seeded data.

1. To enable Migrations, please follow the below steps:

* Open Tools -> NuGet Package Manager -> Package Manager Console window, enter the following commands:

PM> Enable-migrations

PM> Add-migration InitialCreate

* Migrations folder will be created. Open the folder and open Configuration.cs file and add the following method to Configuration class to seed the sample data.
* Run the following command in the Package Manager Console:

PM> Update-database

This will create the database and table in SQL Server.

1. Build the application and check for build errors.

DentalApplication

1. DentalApplication contains 3 controllers: DentistsController, PatientsController and AccountController which extends ApiController to implement WebApi2.
2. DentistsController has all dentist related api methods. PatientsController has all patient related api methods. AccountController has two methods. One is to register the user and the other method is to login. Login method generates the token which is used for authenticating the api calls.