EF: PA01

Creating a simple Multi-Layered application

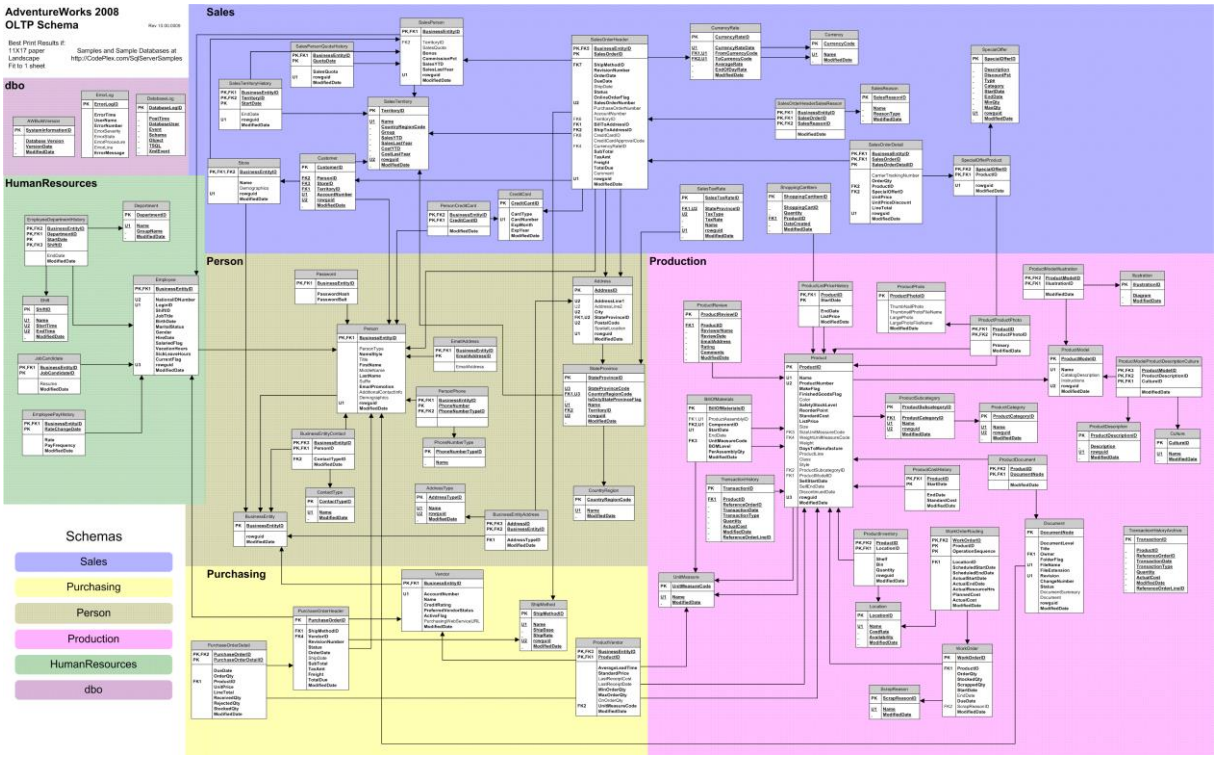
With WEB API & Angular

Starting Point

You need the AdventureWorks database. In order to install it, I reference to PA0 from the SQL Server SQL01 course.

We are going to make a simple feature based on the AdventureWorks database. In future excercises, we can extend and re-arrange the features in order to become really good full-stack developers.

The AdventureWorks Model



Functional Requirements

Display a grid with customers

I want a grid of the customers with the following information:

- First Name
- Last Name
- AccountNumber
- Sum of TotalDue

The grid should be searchable on all fields.

You should also be able to display the customers with a sales higher or lower then X Sum of TotalDue

Make sure only the right type of users can edit information

There are 2 types of users: Administrators and Users.

An administrator can change (edit) the information displayed in the grid. An administrator can also create users & administrators, change the roles, etc.

A user can only view the information displayed in the grid.

Non-Functional Requirements

This application should – in collaboration with your mentor – be multi-layered.

You need to include the possibility to login and implement proper authorization and authentication-mechanisms which work through the layers. This means both within Angular and Web API.

This means we expect the following application-layers:

- User Interface Layer – UI – ASP.NET MVC (core) + Angular
- Service Layer (WEB API)
- Business Layer - BL
- Data Access Layer – DAL

The application should be written according to the best practices as explained earlier.

UI

You should use a combination of ASP.NET MVC and Angular for the UI.