SPA – (Single Page Application) – to web applikacja, która ładuje pojedyńczą stronę HTML i potem updejtuj ją dynamicznie, zamiast ładować nową stronę.

POCO – (Plain Old CLR Objects – Prosty stary obiekt CLR) – jest to klasa .Net niezależna od żadnej podstawowej klasy specyficznej dla frameworka. Czyli jest to zwykła klasa .net (jaką sobie tworzysz do logiki aplikacji). Jednostki POCO = persistance-ignorant objects-obiekty ignorujące utrwalanie) wspierają większość takich samych LINQ zapytań, jak jednostki wyprowadzone przez Entity Object. **Klasy POCO implementują tylko domenową logikę biznesową aplikacji ( to są te all klasy z Model folder).** Można użyć DTOs z klasami POCO, aby przekazać dane między warstwami, bo przekazywanie POCOs entities jest ciężkie. DTO entitty/object jest lżejszy, bo nie ma metod, tylko publicznych członków.

DTO – Data Transfer Objects- nie zawiera żadnych metod (POCO zawiera, stąd cięższe), tylko publicznych członków.

MyContext :DbContext – zarządza obiektami encji during run time, obejmuje:

* Zapełnianie obiektów danymi z bazy danych (obiektów klas z Model folder),
* Śledzenie zmian,
* Utrwalanie danych w bazie danych

DDD – (Domain-Driven Design) – zasady I wzorce które pomagają w tworzeniu obiektowych systemów. Domain models – tak nazywa się abstrakcje oprogramowania, które zostało zaimplementowane zgodnie z DDD.

* Dobre modele mają wiele atrybutów niezależnych od ich implementacji.
* the Ubiquitous Language – słownictwo z danej specjalistycznej dziedziny klienta zaimplementowane w kodzie ( jako property odpowiednie w klasach modelu)
* Bounded Context – obszary najwyższej odpowiedzialności – pierdolenie to po prostu zestaw danych na odizolowaną jedną klasę modelu.
* Single Responsibility Principle (SRP) – jedna klasa odpowiada za jeden zestaw powiązanych z kluczem danych i one mogą być wykorzystane w różnym kontekście.
* DDD nudges you toward identifying contexts and constraining your modeling effort within particular contexts. – wymusza tworzenie kontekstów I ich modelowanie

Normalizacja db – proces organizowania danych w db. Czyli normalizacją jest tworzenie tabel i ustawianie relacji między nimi wg. Jakiś tam regół. (key words: Nadmiarowe dane i niespójne zależności, postać normalna = reguła wg. Której tworzona jest db).

1. Pierwsza postać normalna – klucz podstawowy
2. Druga – klucz obcy
3. Trzecia – każde pole musi zależeć od klucza = mocno podzielona na osobne tabele db

BUT! Easy with that:

W teorii warto stosować normalizację. Jednak stosowanie wielu małych tabel może pogorszyć wydajność lub spowodować przekroczenie dozwolonej liczby otwartych plików lub pojemności pamięci.

The best way:

Stosowanie trzeciej postaci normalnej tylko do danych często zmienianych – weryfikacja all powiązanych pół po zmianie jednego z nich = bezpiecznie

1. Oraz
2. Piata nie są konieczne aby znać, ale istnieją