## Technologie po stronie Back-End.

Do stworzenia serwerowej części aplikacji została wykorzystana technologia oparta na języku programowania C# oraz bibliotekach zawartych w .Net Framework którego twórcą jest Microsoft. Za pomocą tej technologii zostały stworzone wszystkie moduły składające na aplikację serwerową. Za serwis REST odpowiadają biblioteki WebApi2, dostęp oraz obsługa zrealizowany jest przy pomocy ORMa Entity Framework, wysyłanie maili, komunikacja z Docusign, tworzenie logów oraz modele domeny również powstały w oparciu o język C# oraz technologię .Net. ????????????

Język C# jest to obiektowy język programowania, stworzony przez pracowników Microsoft. Do jego głównych cech należą obiektowość hierarchiczna, automatyczna zarządzanie pamięcią, deletegaty oraz zdarzenia, system refleksji oraz atrybutów klas. W kolejnych wydaniach tego języka, C# doczekał się szeregu rozszerzeń, takich jak typy generyczne, zarządzanie wielowątkowością za pomocą „Tasków”, ułatwienie programowania asynchronicznego przy pomocy słów kluczowych async i await. Ponadto język ten posiada wsparcie w postaci technologii LINQ. LINQ umożliwia programiście tworzenie zapytań przypominających w składni SQL bezpośrednio na obiektach oraz ich kolekcjach. Posiada on szereg własnych zapytań umożliwiających w szybki i wygodny sposób zarządzać kolekcjami obiektów. W połączeniu z Entity Framework daje on potężne narzędzie, za pomocą którego obsługa bazy różnorakich źródeł danych jest niezwykle proste i intuicyjne.

Entity Framework jest to biblioteka ORM (Object Relational Mapping) umożliwiająca odwzorowanie relacyjnej bazy danych przy pomocy modelu obiektowego. Jedną z głównych zalet tej biblioteki jest możliwość rozdzielenia warstw logiki biznesowej oraz warstwy dostępu do danych. Operowanie na danych w Entity Framework wykorzystuje składnie LINQ przy pomocy specjalnych bibliotek LINQ to Entities oraz LINQ to SQL. Wyróżniamy trzy różne podejścia do tworzenia bazy danych przy pomocy Entity Framework. Pierwszym z nich jest „Database First”. Najczęściej korzystamy z niego, kiedy baza, z której chcemy skorzystać już istnieje. Przy pomocy odpowiedniego narzędzia zawartego w Visual Studio jesteśmy odwzorować w obiektowym kodzie strukturę tabel, widoków oraz procedur stworzonych w podłączonej bazie danych. Kolejnym podejściem jest „Model First”. Jego główną zaletą jest uniknięcie przez programistę konieczności pisania jakiegokolwiek kodu SQL. Za pomocą narzędzia ADO.NET Entity Data Model Designer definiujemy odpowiedni model danych, który będzie posiadał zarówno odwzorowanie obiektowe w naszej aplikacji, jaki i utworzy odpowiednią strukturę tabel oraz relacji w bazie danych. Ostatnim podejściem jest tak zwane podejście „Code First”. Programista najpierw tworzy klasy oraz powiązania między nimi. Następnie przy mocy odpowiedniego silnika mapującego Entity Framework tworzy odpowiednie tabele oraz klucze główne i prywatne. Podejście to jest uważane za najlepsze, wpływa na to szereg czynników. Twórca takiego kodu modyfikując strukturę klas, ma praktycznie bezpośredni wpływ na strukturę bazy danych, z której korzysta. Ponadto zwolniony jest z pisania kodu SQL, który by odzwierciedlał zmiany jakie zaszły by w czasie modyfikacji modelu domenowego w takiej aplikacji. Aby utrzymać synchronizację i jednoznaczność modelu danych, programista zmuszony jest tworzyć tak zwane „migracje” po każdej zmianie, jaka zaszła w modelu obiektowym. Migracja taka, generowana automatycznie, aktualizuje strukturę tabel. Ponadto programista ma możliwość dodawania własnej logiki, która wygeneruje dodatkowy kod SQL mający wpływ na bazę danych. …………………….

WebApi jest to framework stworzony na potrzeby platformy .Net umożliwiający sprawne i wydajne tworzenia serwisów REST. Zanim nastąpi prezentacja możliwości samego WebApi, przybliżona zostanie sama idea serwisów REST. REST, czyli z angielskiego Representational State Transfer jest zarówno pewnym wzorcem dobrych praktyk w tworzeniu architektury aplikacji rozproszonych, jak implementacją usług sieciowych, bazującą na protokole http. Oczywiście taka usługa musi spełniać założenia, jakie REST sam w sobie niesie. Główną ideą serwisów takiego typu, jest podzielenie komunikacji sieciowej, na cztery główne metody pełniące różne funkcje. I tak do pobierania danych z serwera korzysta się z metody GET, dodawanie danych następuje przy pomocy metody POST, za edycję danych odpowiada metoda PUT, a usuwaniem zajmuje się metoda DELETE. Taka klasyfikacja metod wiąże niepodzielnie metody http z typem operacji, jaką chcemy przeprowadzić na naszych danych po stronie serwera. Podejście takie posiada taką zaletę, że programista wywołujący którąś z wybranych metod http, może bezproblemowo określić jaka operacja zostanie wykonana.