

Nazwa aplikacji

Karpiński Maciej

Krysa Marcin

Kuczma Łukasz

Mertuszka Adam



Projektowanie i programowanie systemów internetowych II

7 grudnia 2020

Spis treści

1	Opis funkcjonalny systemu	3
2	Opis technologiczny	4
2.1	ASP.NET Core 3.1	4
2.2	AutoMapper	4
2.3	C#	4
2.4	Docker	4
2.5	Entity Framework	5
2.6	MailKit	5
2.7	Microsoft SQL Server	5
2.8	Quartz.NET	6
2.9	Swashbuckle	6
3	Instrukcja lokalnego i zdalnego uruchomienia systemu	7
3.1	Lokalne uruchomienie systemu	7
3.2	Zdalne uruchomienie systemu	7
4	Instrukcja uruchamiania testów oraz opis testowanych funkcjonalności	8
5	Wnioski projektowe	9

1 Opis funkcjonalny systemu

Opis funkcjonalny systemu

- funkcja 1
- funkcja 2

2 Opis technologiczny

Przy tworzeniu projektu „” wykorzystano następujące technologie:

2.1 ASP.NET Core 3.1

ASP.Net Core jest wysokowydajnym frameworkiem, do budowania nowoczesnych aplikacji internetowych wykorzystujących moc obliczeniową chmur. ASP.Net Core jest technologią open - source, wykorzystującą silnik html Razor, dzięki której możliwe jest tworzenie aplikacji multiplatformowych, które mogą być używane na każdym urządzeniu wyposażonym w przeglądarkę internetową.

2.2 AutoMapper

AutoMapper jest biblioteką służącą do mapowania między obiektami, dzięki czemu można automatycznie mapować właściwości dwóch różnych obiektów, przekształcając obiekt wejściowy jednego typu na obiekt wyjściowy innego typu.

2.3 C#

C# jest obiektowym językiem programowania, zaprojektowanym w latach 1998 – 2001 dla firmy Microsoft. Napisany program jest kompilowany do Common Intermediate Language(CLI), który następnie wykonywany jest w środowisku uruchomieniowym takim jak .NET Framework, .NET Core, Mono lub DotGNU. Wykorzystanie CLI sprawia, że kod programu jest wieloplatformowy (dopóki istnieje odpowiednie środowisko uruchomieniowe). C# posiada wiele wspólnych cech z językami Object Pascal, Delphi, C++ i Java a najważniejszymi cechami C# są:

- Obiektowość z hierarchią o jednym elemencie nadrzędnym (podobnie jak w Javie);
- Zarządzaniem pamięcią zajmuje się środowisko uruchomieniowe;
- Właściwości i indeksery;
- Delegaty i zdarzenia – rozwinięcie wskaźników C++;
- Typy ogólne, generyczne, częściowe, Nullable, domniemane, anonimowe;
- Dynamiczne tworzenie kodu;
- Metody anonimowe;
- Wyrażenia lambda.

2.4 Docker

Docker jest otwarcie źródłowym oprogramowaniem służącym do realizacji „konteneryzacji” aplikacji, służąca jako platforma dla programistów i administratorów do tworzenia, wdrażania i uruchamiania aplikacji rozproszonych. Pozwala umieścić program oraz jego zależności (biblioteki, pliki konfiguracyjne, lokalne bazy danych itp.) w lekkim, przenośnym, wirtualnym kontenerze, który można uruchomić na prawie każdym serwerze z systemem

Linux. Kontenery wraz z zawartością działają niezależnie od siebie i nie wiedzą o swoim istnieniu. Mogą się jednak ze sobą komunikować w ramach ściśle zdefiniowanych kanałów wymiany informacji. Dzięki uruchamianiu na jednym wspólnym systemie operacyjnym, konteneryzacja jest lżejszym sposobem wirtualizacji niż pełna wirtualizacja lub parawirtualizacja za pomocą wirtualnych systemów operacyjnych.

2.5 Entity Framework

Entity Framework jest technologią open - source do mapowania obiektowo – relacyjnego (ORM), które wspierają rozwój aplikacji zorientowanych na dane. Entity Framework umożliwia programistom pracę z danymi w postaci obiektów i właściwości specyficznych dla domeny, bez konieczności przejmowania się bazowymi tabelami i kolumnami baz danych, w których dane są przechowywane.

2.6 MailKit

MailKit jest multiplatformową otwarto źródłową biblioteką .NET klienta pocztowego opartą o MimeKit, która została zoptymalizowana pod kątem urządzeń mobilnych. MailKit oferuje następującą funkcjonalność:

- Obsługa proxy HTTP, Socks4, Socks4a i Socks5;
- Uwierzytelnianie SASL;
- Kompletny klient SMTP;
- Kompletny klient POP3;
- Kompletny klient IMAP;
- Sortowanie i wątkowanie wiadomości po stronie klienta;
- Asynchroniczne wersje wszystkich metod sieciowych;
- Obsługa S/MIME, OpenPGP, DKIM i ARC;
- Obsługa Microsoft TNEF.

2.7 Microsoft SQL Server

Microsoft SQL Server jest systemem zarządzania relacyjnymi bazami danych opracowany przez firmę Microsoft. Cechą charakterystyczną jest głównie wykorzystywanie języka zapytań Transact-SQL, który jest rozwinięciem standardu ANSI/ISO. W projekcie wykorzystano wersję 2019 Express, która jest bezpłatną edycją programu Microsoft SQL Server, oferująca podstawowy silnik bazy danych, nieposiadający ograniczenia ilości obsługiwanych baz lub użytkowników. Ograniczenia, występujące w wersji Express to m.in.: korzystanie z jednego procesora, 1 GB pamięci RAM, 10GB plików bazy danych czy brak SQL Agent.

2.8 Quartz.NET

Quartz.NET jest otwartoźródłową biblioteką do planowania zadań. Quartz.NET może być używany do tworzenia prostych lub złożonych harmonogramów wykonywania dziesiątek, setek, a nawet dziesiątek tysięcy zadań. Quartz.NET jest portem biblioteki Quartz dla środowiska Java.

2.9 Swashbuckle

Swashbuckle jest biblioteką, która dodaje zestaw narzędzi „Swagger” generujących automatycznie dokumentację API aplikacji, wyposażoną w przejrzysty interfejs użytkownika. Swashbuckle umożliwia również testowanie API. Dokumentacja jest dostępna pod adresem: „./swagger”

3 Instrukcja lokalnego i zdalnego uruchomienia systemu

3.1 Lokalne uruchomienie systemu

opis lokalnego

3.2 Zdalne uruchomienie systemu

opis zdalnego

4 Instrukcja uruchamiania testów oraz opis testowanych funkcjonalności

5 Wnioski projektowe

wnioski