Dokumentacja techniczna ,,*Restaurant Booking*„

by

Damian Kordylasiński

Jakub Kopka

Celem aplikacji ,,*Restaurant Booking*„ jest umożliwienie klientom rezerwacji stolików w restauracjach, poprzez wybór spośród dostępnych restauracji i zarezerwowanie wybranego stolika. Administrator aplikacji ma możliwość dodawania nowych restauracji oraz stolików do już istniejących restauracji. Zakres aplikacji obejmuje więc zarówno możliwość rezerwacji stolika przez klientów, jak i zarządzanie dostępnością stolików przez administratora.

Wykorzystane przez nas technologie to:

* *Microsoft.AspNetCore.Diagnostics.EntityFreamworkCore* - biblioteka umożliwiająca łatwe diagnozowanie i rozwiązywanie problemów związanych z bazą danych w projekcie opartym na ASP.NET Core i Entity Framework Core.
* *Microsoft.AspNetCore.Identity.EntityFreamworkCore* - biblioteka umożliwiająca implementację systemu zarządzania użytkownikami i uprawnieniami w projekcie opartym na ASP.NET Core i Entity Framework Core.
* *Microsoft.AspNetCore.Identity.UI* - biblioteka zawierająca gotowe komponenty interfejsu użytkownika do obsługi systemu zarządzania użytkownikami i uprawnieniami w projekcie opartym na ASP.NET Core.
* *Microsoft.VisualStudio.Web.CodeGeneration.Design* - narzędzie umożliwiające generowanie kodu w projekcie opartym na ASP.NET Core, na podstawie szablonów i modelu danych.
* *Microsoft.EntitFrameworkCore.Tools* - narzędzie umożliwiające wykonywanie różnych operacji na bazie danych, takich jak migracje czy tworzenie tabel na podstawie klas, w projekcie opartym na Entity Framework Core.
* *Microsoft.EntitFrameworkCore.SqlServer* - biblioteka umożliwiająca połączenie projektu opartego na Entity Framework Core z bazą danych MS SQL Server.

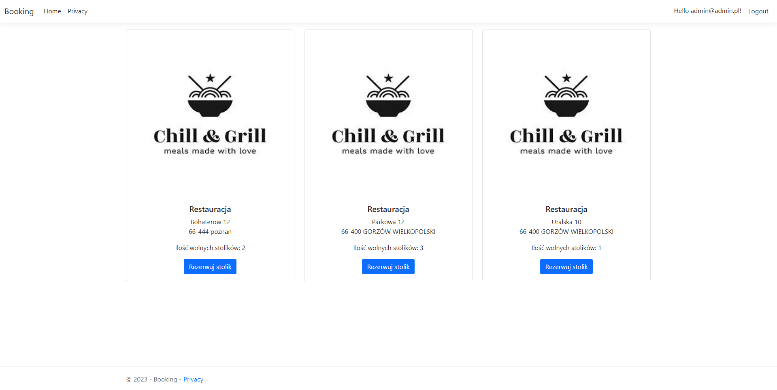
Aplikacja została zbudowana z wykorzystaniem technologii opartych o platformę ASP.NET Core. Wykorzystano Microsoft.AspNetCore.Diagnostics.EntityFreamworkCore do zarządzania bazą danych, co pozwala na łatwe tworzenie, modyfikowanie i usuwanie tabel oraz rekordów. Identity Framework został użyty do obsługi systemu logowania i autoryzacji użytkowników aplikacji (Microsoft.AspNetCore.Identity.EntityFreamworkCore, Microsoft.AspNetCore.Identity.UI) Wykorzystano też Microsoft.VisualStudio.Web.CodeGeneration.Design oraz Microsoft.EntitFrameworkCore.Tools do generowania kodu, co ułatwia tworzenie i zarządzanie modelem bazy danych. Ostatecznie dane przechowywane są w bazie danych SQL Server (Microsoft.EntitFrameworkCore.SqlServer).

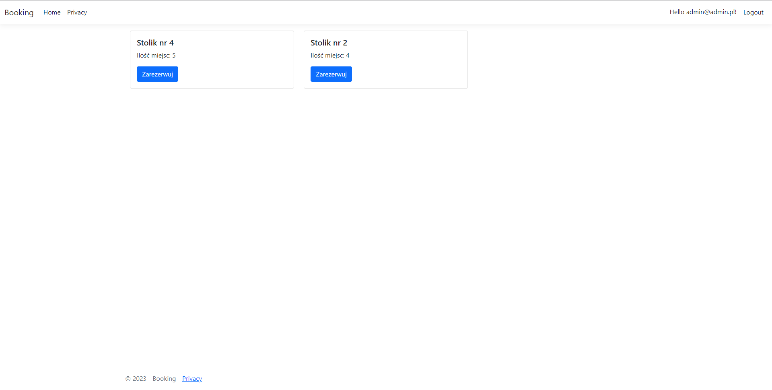
Architektura projektu oparta jest o następujące warstwy:

* *Warstwa prezentacji (Presentation Layer)* - odpowiada za interakcję z użytkownikiem, przetwarzanie danych wejściowych i wyświetlanie wyników.
* *Warstwa biznesowa (Business Layer)* - odpowiada za logikę biznesową aplikacji, taką jak rezerwacja stolików, walidację danych itp.
* *Warstwa dostępu do danych (Data Access Layer)* - odpowiada za komunikację z bazą danych i zarządzanie danymi, wykorzystując Entity Framework.

Takie podejście pozwala na łatwe utrzymanie i rozwój aplikacji, a także ułatwia testowanie i debugowanie poszczególnych warstw.

Opis interfejsu użytkownika:

Widok strony głównej. Na górze widzimy pasek nawigacyjny z menu, dzięki któremu możemy przejść do wszystkich zakładek znajdujących się na naszej stronie. W głównej sekcji wyświetlają się wszystkie dostępne restauracje oraz przycisk z możliwością rezerwacji stolika.

Widok strony rezerwacji stolika. Po klikniecie na stronie głównej w wybrana restauracje pokazują nam się wolne stoliki gotowe do rezerwacji.

Obraz zawierający tekst

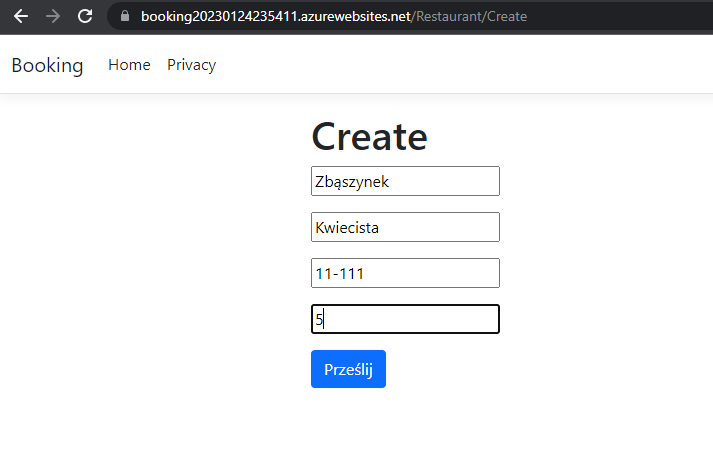
Opis wygenerowany automatycznieWidok potwierdzenia rezerwacji. Po kliknięciu w przycisk zarezerwuj na stronie rezerwacji stolika przenosi nas do strony z potwierdzeniem rezerwacji.

Obraz zawierający stół

Opis wygenerowany automatycznieWidok admina do zarzadzania restauracjami. Aby dostać się do strony admina w pasku wyszukiwania trzeba dopisać do linku strony głównej ,,/RestaurantGetAllAdmin” W tym widoku możemy stworzyć, edytować, usunąć i sprawdzić   
detale restauracji.

Obraz zawierający stół

Opis wygenerowany automatycznieWidok admina w detalach restauracji. Na tej stronie możemy sprawdzić dane restauracji takie jak id czy adres. Możemy również dodać, edytować oraz usunąć stolik lub przypisana do niego rezerwacje.

Widok tworzenia stolika. W tym widoku możemy stworzyć stolik podając miasto, ulice, oraz kod pocztowy.

Nasz autorski opis bazy danych.

Obraz zawierający tekst

Opis wygenerowany automatycznieBaza danych składa się z elementów związanych z użytkownikami oraz z restauracji. Encje odpowiedzialne za restauracje i rezerwacje powiązane są w następujący sposób:

w tabeli tables występuje foreign key tabeli restauracja  
w tabeli reservations wystepuja klucze obce od tabeli tables i user.

W programie na początku wyświetlają się restauracje oraz ilość wolnych stolików ( foreign key tables ). Następnie użytkownikowi wyświetlają się stoliki do wybranej restauracji ( restaurant id ) Później następuje rezerwacja której do tabeli reservation dopisujemy table id i user id aktualnie zalogowanego użytkownika. Użytkownik chcąc wyświetlić rezerwacje odwołuje się do klucza obcego tables, dzięki czemu może zobaczyć zarezerwowany stolik w konkretnej restauracji.

Instrukcja instalacji projektu może różnić się w zależności od konkretnego środowiska, w którym będzie on uruchamiany. Oto przykładowe kroki instalacji dla środowiska programistycznego Visual Studio:

1. Pobierz kod źródłowy projektu z odpowiedniego repozytorium lub rozpakuj plik archiwum z kodem źródłowym.
2. Otwórz Visual Studio i kliknij opcję "Open a Project or Solution" w oknie startowym lub wybierz opcję "Open" w menu "File".
3. Przejdź do katalogu z kodem źródłowym projektu i wybierz plik .sln (rozwiązanie) odpowiadający projektowi.
4. Po załadowaniu projektu, kliknij prawym przyciskiem myszy na projekt i wybierz "Manage NuGet Packages" w celu zainstalowanie potrzebnych paczek.
5. Uruchom kompilację projektu klikając na "Build" -> "Build Solution" w menu lub przyciskiem "Ctrl+Shift+B"
6. Uruchom projekt klikając na "Debug" -> "Start without debugging" lub przyciskiem "Ctrl+F5"

Jeśli projekt zależy od jakiejś bazy danych, należy ją najpierw utworzyć i skonfigurować. Konfiguracja połączenia z bazą danych znajduje się w pliku "appsettings.json"

UWAGA: Jeśli pojawiają się problemy z paczkami, należy zaktualizować Visual Studio do najnowszej wersji oraz zainstalować niezbędne narzędzia do rozwoju .NET (np. .NET Core SDK).

Jeśli masz inne środowisko niż Visual Studio to instrukcja instalacji może się różnić, dlatego zalecam skonsultowanie się z dokumentacją techniczną lub naszym zespołem projektowym.