

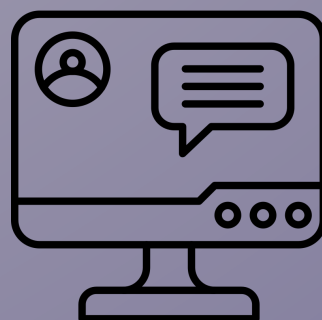
HOTEL RESERVATION MANAGER

Работа по курсов проект

- Математическа гимназия „Академик Кирил Попов“ 4001 Пловдив, ул. „Чемшир“ 11
тел.: + 359 32 643 157,
E – mail: omg@omg-bg.com

Ръководител:

- Росен Вълчев
- Изработен от: Айлин Шабан, Иван Мишевски,
Иван Гавраилов



Документация

Съдържание

1. Цели
2. Разпределение на ролите
3. Ниво на сложност на проекта
 - 3.1 Entity Framework
 - 3.2 Адаптивен и Модерен Потребителски
Интерфейс
 - 3.3 Бизнес логика
4. Инсталация и деинсталация
5. Основни етапи в реализирането на проект
 - 5.1 Потребители
6. Реализация
 - 6.1 Архитектура
 - 6.2 База данни
 - 6.3 Описание на приложението
 - 6.4 Използвани технологии
 - 6.5 Използвани езици за програмиране
 - 6.6 Описание на приложението
7. Развитие и нововъведения
8. Заключение
9. Използвана литература
 - 9.1 Книги
 - 9.2 Линкове, сайтове

1.Цели

Проектът Hotel Reservations Manager има за цел да разработи уеб базирана система за ефективно управление на хотелски резервации. Чрез използване на съвременни технологии като ASP.NET, Entity Framework, SQL Server, C#, HTML и CSS, системата ще предоставя интуитивен интерфейс и стабилна функционалност както за администратори, така и за обикновени потребители (служители).

Основните цели на проекта включват:

- Автоматизация на процесите по резервация на хотелски стаи, като се осигурява лесно създаване, редактиране и премахване на резервации. Системата ще позволява избиране на стая и клиенти от вече налични в базата данни записи.
- Поддръжка и управление на потребители, включително разграничение между администраторски и обикновени акаунти. Администраторът има разширени права за контрол върху системата – създаване, редактиране и премахване на потребители и стаи.
- Ефикасно управление на клиентски данни, с възможност за повторна употреба на информация за вече съществуващи клиенти при нови резервации, с цел ускоряване на обслужването.
- Управление на хотелски стаи – поддръжка на информация за типове стаи, капацитет, цена, наличност и уникален номер. Стаите могат да бъдат редактирани само от администратор.
- Да повиши ефективността и точността на обслужването в хотелската индустрия, като предостави модерен и сигурен инструмент за управление на резервации, с акцент върху удобството на потребителя и качеството на услугата.

1.Цели

- Финансови изчисления и валидни резервации – автоматизирано изчисляване на дължимата сума за всяка резервация на база броя нощувки, броя и възрастта на клиентите, както и допълнителните услуги (закуска или all inclusive).
- Надеждно съхранение на данни – чрез SQL Server база данни, която поддържа всички същности и техните връзки: потребители, клиенти, стаи и резервации.
- Улеснено търсене, филтриране и странициране – във всеки изглед с множество записи (потребители, клиенти, стаи, резервации) се поддържа сортиране, филтрация по ключови полета и визуализиране на данни по страници (10, 25 или 50 записа).
- Сигурност и валидност на данните – всяка форма в системата подлежи на валидация (напр. валиден телефонен номер, коректни дати за резервация, задължителни полета).
- Историчност и отчетност – дори след освобождаване от длъжност, служителите и направените от тях резервации се запазват в системата, което осигурява пълна проследимост на действията.

Системата цели да повиши ефективността и точността на обслужването в хотелската индустрия, като предостави модерен и сигурен инструмент за управление на резервации, с акцент върху удобството на потребителя и качеството на услугата.

2. Разпределение на ролите

Обработката на софтуера е разпределена между екипа, както следва:

Айлин Шабан: Отговорна за разработка на презентационния слой, потребителския интерфейс, базата данни и документация.

Иван Гавраилов: Отговорна за разработка на презентационния слой, потребителския интерфейс и презентацията.

Иван Мишевски: Отговорен за функционалността, тестване, базата данни, документация и управление на версиите чрез Git.

3. Ниво на сложност на проекта

Определяме нивото на сложност на проекта като умерено.

Сблъскахме се с някои предизвикателства, които успешно преодоляхме:

- Entity Framework Едно от първите препятствия беше да се научим как да използваме Entity Framework за управление на базата данни. Сблъскахме се с проблеми със съвместимостта на версиите, но след проучване и тестване успяхме да интегрираме EF успешно.
- HTML за потребителски интерфейс Изборът на подходящ дизайн за потребителския интерфейс беше предизвикателство. Използвахме стандартни HTML елементи и CSS за създаване на прост, но функционален интерфейс. Вместо да използваме готови библиотеки като Bootstrap, решихме да изградим интерфейса ръчно, за да имаме пълен контрол върху дизайна и функционалността.

3.1 Entity Framework

Едно от първите препятствия, които трябваше да преодолеем, беше да се научим как да използваме Entity Framework. Сблъскахме се с проблеми с инсталацията и проблеми със съвместимостта на версиите, което добави сложност и изискваше допълнително отстраняване на неизправности. Въпреки това, след известно проучване и постоянство, успяхме да стартираме Entity Framework.

3.2 Адаптивен и Модерен Потребителски Интерфейс

За да осигурим удобство и интуитивност при използването на уеб приложението, дизайнът се базира на принципите на адаптивния (responsive) и модерен потребителски интерфейс (UI/UX). Това включва:

- **Отзивчив дизайн:** Интерфейсът ще се адаптира динамично към различни устройства и резолюции (настолни компютри, таблети, мобилни устройства), осигурявайки оптимално визуално и функционално изживяване.
- **Модулна структура:** Използване на модулен подход за проектиране, който улеснява бъдещи разширения и поддръжка на нови функционалности.

3.3 Бизнес логика

Бизнес логиката в системата Hotel Reservations Manager обхваща всички основни процеси и правила, които управляват поведението на приложението и взаимодействието между неговите компоненти.

Тя гарантира правилното функциониране на системата, обработката на потребителски действия и поддържането на консистентни и валидни данни. Основните елементи на бизнес логиката включват:

- **Обработка на заявки за търсене и филтриране**

Системата трябва да позволява търсене и филтриране на данни в различни модули – потребители, клиенти, стаи и резервации.

- **Управление на резервации и изчисляване на суми**

Основна бизнес логика е свързана с обработката на резервации. Това включва:

- **Проверка за свободни стаи в зададен период.**

Възможност за добавяне на резервация само ако има поне една свободна стая.

- **Автоматично отбелязване на стаята като свободна, след изтичане на датата на освобождаване.**

- **Управление на потребителски роли и достъп**

Системата поддържа различни нива на достъп:

- Само администратори могат да създават, редактират и изтриват потребители и стаи.
- Обикновени потребители могат да разглеждат стаи, да правят и обработват резервации.
- Освободените служители (неактивни акаунти) нямат достъп до системата, но техните данни и резервации остават запазени в базата.

- **Валидация и обработка на данни**

За поддържане на целостта на данните, системата прилага набор от валидиращи правила:

- Телефонните номера трябва да имат точно определен брой символи (напр. 10).
- Датата на освобождаване не може да е по-ранна от датата на настаняване.
- Задължителни полета за създаване на резервация и потребителски профили.
- Проверка за уникалност на потребителски имена, номера на стаи и имейли.

4. Инсталация и деинсталация

Преди да бъде използван разработения софтуер е необходимо да се премина през последователност от стъпки за неговата инсталация и настройки за експлоатация.

Нека започваме:

- Инсталирайте Visual Studio 2022
- Уверете се, че .NET 9 е инсталиран
- Инсталирайте Microsoft SQL Server
- Инсталирайте SQL Management Studio 1.Клонирайте кода, като използвате следната команда: `git clone` 2.Отворете проекта във Visual Studio. С помощта на SQL Management Studio създайте базата данни. Отидете на Visual Studio 3.Отидете на `appsettings.json` и променете низовете за връзка (connection strings), за да сочат към вашата база данни 4.Отворете Package Manager Console

5. Въведете следната команда, за да приложите миграциите и да създадете базата данни: Update-Database
6. Стартирайте приложението чрез Visual Studio (Ctrl + F5) или чрез командата: dotnet run
7. Приложението ще се стартира на локален сървър (по подразбиране на <http://localhost:5000/>).
8. За деинсталация, изтрийте папката, в която се намира проекта и изтрийте и локалната база от данни.

5. Основни етапи при създаването на проекта

5.1 Потребители

Приложението Hotel Reservations Manager поддържа два основни типа потребители, като достъпът до функционалностите е строго контролиран според техните роли:

Администратори

Администраторите имат пълен контрол върху системата и могат да извършват CRUD операции (създаване, редактиране, изтриване и преглед) върху всички основни данни в приложението, включително:

- Управление на потребителски акаунти (създаване на нови служители, редактиране на данни, деактивиране на профили и др.);
- Добавяне, редактиране и изтриване на хотелски стаи;
- Преглед и управление на клиенти и резервации;
- Пълен достъп до административни изгледи и функционалности.

Обикновени потребители (служители)

Обикновените потребители имат ограничен достъп, предназначен за ежедневната работа в хотела. Те могат:

- Да правят и управляват резервации;
- Да разглеждат списъка със стаи (без възможност за промяна или изтриване);
- Да виждат резервации и техните детайли.

5.1 Модул регистрирани потребители

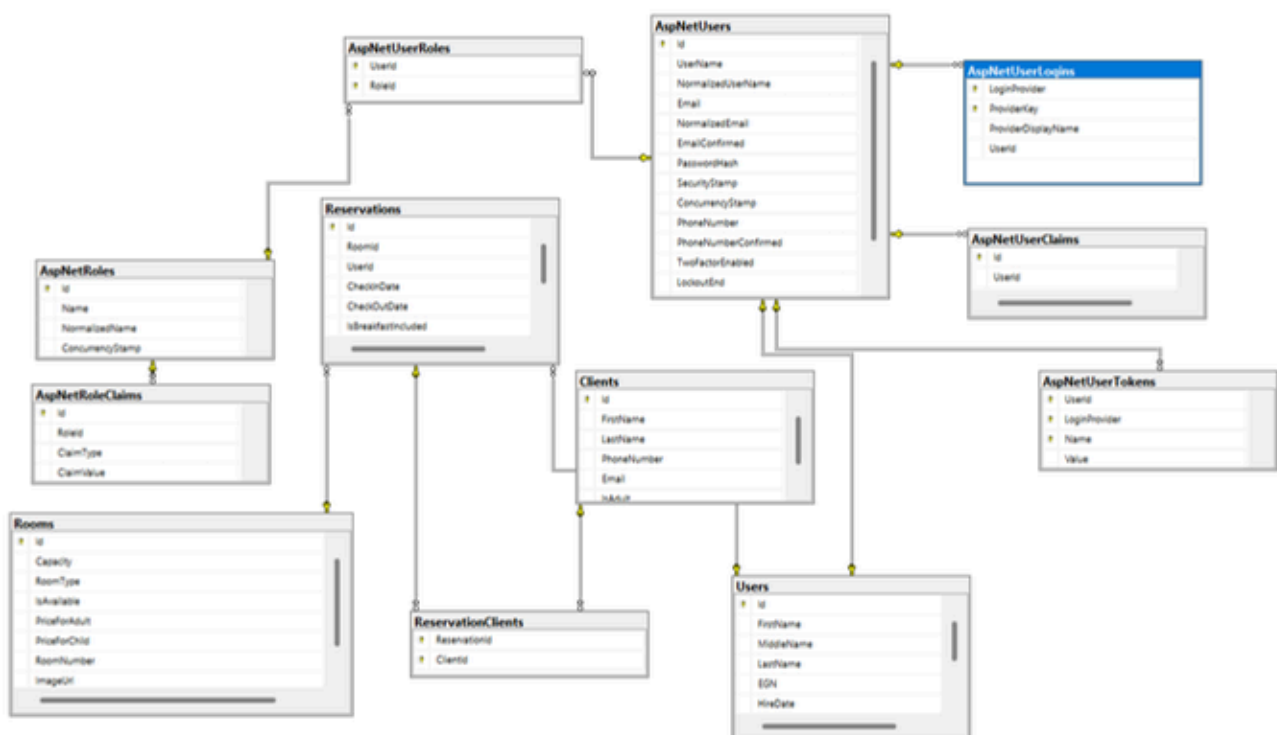
Този модул позволява на всички потребители да се регистрират и да създадат личен профил, чрез който да се идентифицират и използват всички предоставени функционалности на уеб приложението.

6. Реализация

Софтуерът е изграден от един проект – HotelReservationManager.

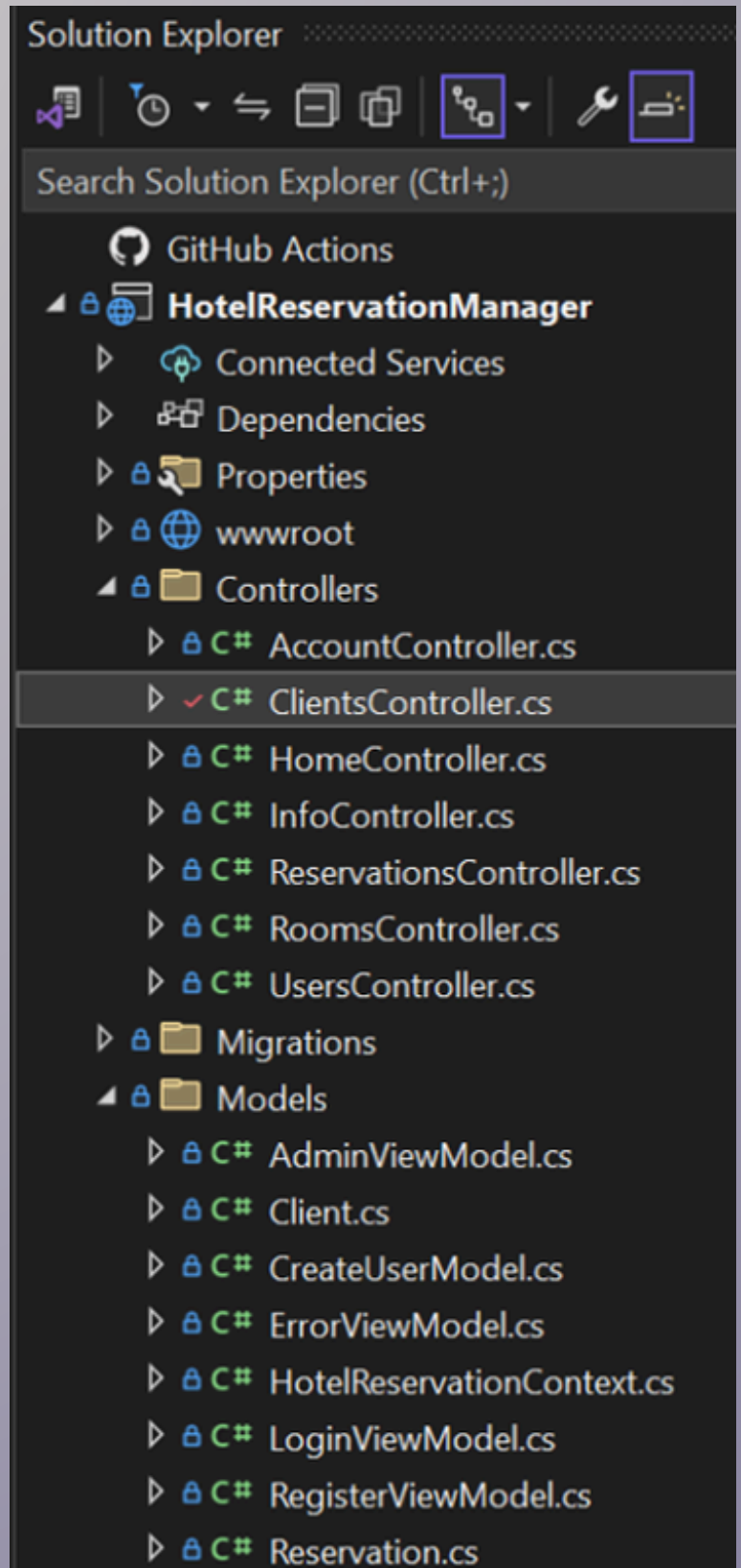
6.1 Архитектура

6.2 Бази данни



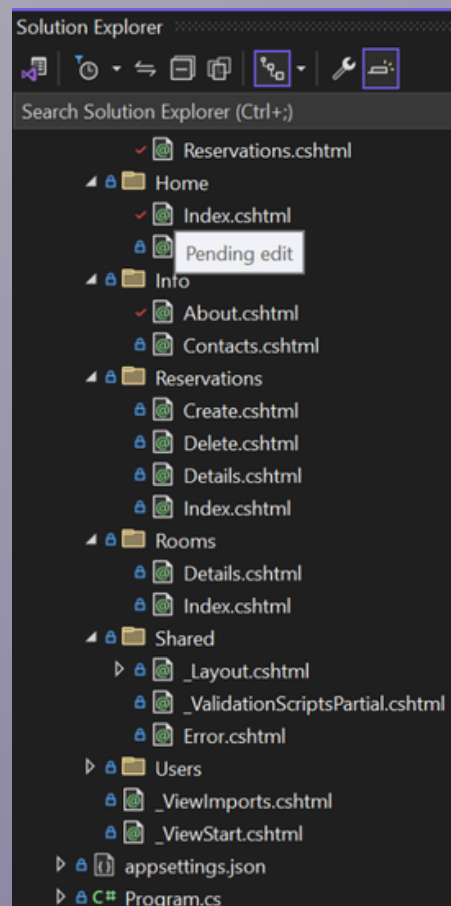
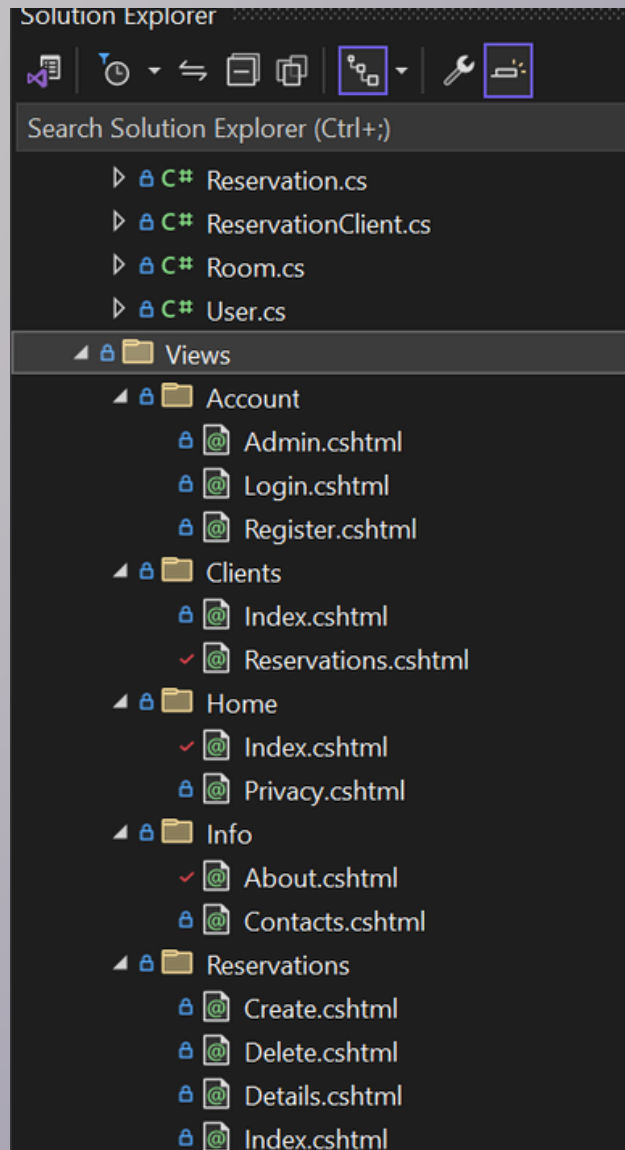
6.3. HotelReservationManager

- **Controllers** – тук се съдържа бизнес логиката. Реализира се връзка с базата данни и модифициране на информация – добавяне, редактиране и други сложни функционалности
- **Migrations** – папка, генерирана от EntityFramework, която служи за съхраняване на миграциите.
- **Models** – основните модели, които представят моделите в базата от данни.
- **Views** – тук се съдържат интерфейсите на приложението



6.3. HotelReservationManager

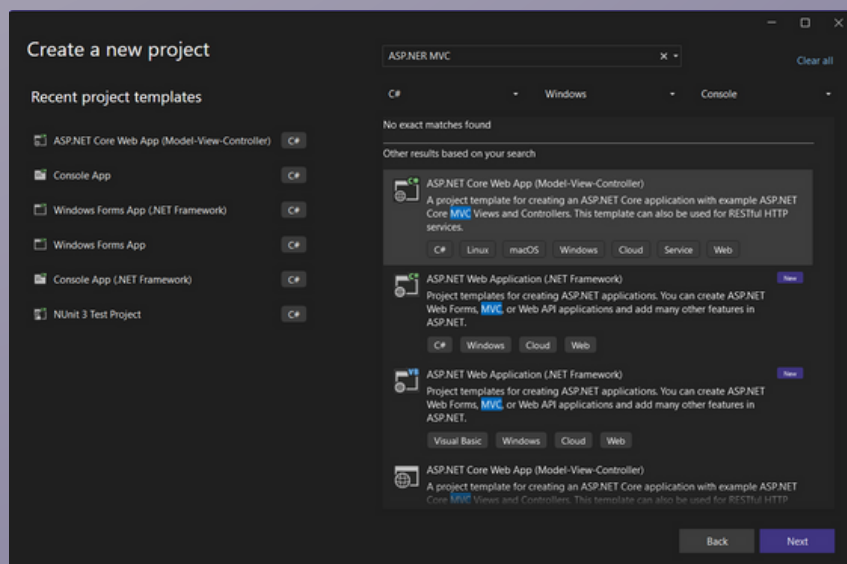
- **Controllers** – тук се съдържа бизнес логиката. Реализира се връзка с базата данни и модифициране на информация – добавяне, редактиране и други сложни функционалности
- **Migrations** – папка, генерирана от EntityFramework, която служи за съхраняване на миграциите.
- **Models** – основните модели, които представят моделите в базата от данни.
- **Views** – тук се съдържат интерфейсите на приложението



6.4 Използвани технологии

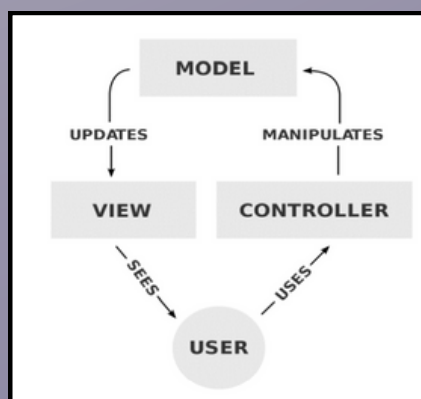
За разработката на уеб приложението се използват следните технологии:

- **.NET Framework 9:** Осигурява стабилна и съвременна платформа за разработка на приложението, като поддържа широк набор от функционалности и инструменти за създаване на мащабируем софтуер.
- **ASP.NET MVC:** Използва се за създаване на динамичен и добре структуриран уеб интерфейс, като разделя логиката на приложението на три основни компонента – Model, View и Controller. Това подобрява поддръжката и развитието на кода.
- **Entity Framework:** Този ORM (Object-Relational Mapping) инструмент улеснява работата с базата данни чрез обектно-ориентиран подход, което намалява сложността при управлението на данните и подобрява производителността.
- **HTML/CSS:** Използват се за разработка и стилизиране на потребителския интерфейс, като гарантират отзивчив дизайн и приятно визуално изживяване за крайните потребители.
- **Git:** Системата за контрол на версиите улеснява управлението на изходния код, позволявайки съвместна работа, проследяване на промените и лесно интегриране на нови функционалности.



- **Visual Studio 2022:** Visual Studio 2022 е мощна интегрирана среда за разработка (на английски: integrated development environment, IDE) на софтуерни приложения за Windows и за платформата .NET Framework. Използва се за разработка на конзолни и графични потребителски интерфейси приложения, както и Windows Forms или WPF приложения, уеб сайтове, уеб приложения и уеб услуги на всички поддържани платформи от Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework и Microsoft Silverlight. Visual Studio предоставя мощна интегрирана среда за писане на код, компилиране, изпълнение, дебъгване (както за високо така и за машинно ниво), тестване на приложения, дизайн на потребителски интерфейс (форми, диалози, уеб страници, визуални контроли и други), моделиране на данни, моделиране на класове, изпълнение на тестове, пакетиране на приложения и стотици други функции
- **Microsoft SQL Server:** Microsoft SQL Server е сървърна система за управление на бази от данни (и по-точно на релационни бази от данни) на компанията Microsoft. Microsoft SQL Server е предназначена за управление на големи сървърно базирани БД, за разлика от MS Access, която е desktop базирана и не е предназначена за управление на големи корпоративни БД.
- **Microsoft.VisualStudio.TestTools.UnitTesting** - е тестов framework, който е част от Visual Studio IDE и се използва за автоматизирано модулно тестване на .NET Framework приложения. Той позволява на разработчиците да пишат и изпълняват тестове във Visual Studio и предоставя набор от функции и инструменти за управление и анализ на резултатите от тестовете.

- Архитектурният модел Model-View-Controller (MVC) разделя уеб приложението на три основни компонента, всеки от които има ясно дефинирана отговорност:
- Модели (Models): Те представляват бизнес логиката и данните на приложението. В контекста на нашия каталог за филми, моделите съдържат класове, които управляват информация за филми, ревюта, потребители, жанрове и други. Моделите са отговорни за комуникацията с базата данни чрез ORM (например Entity Framework) и изпълняват CRUD операции, които гарантират целостта на данните.
- Изгледи (Views): Изгледите отговарят за визуалното представяне на данните. Те използват HTML, CSS и шаблони (например Razor шаблони в ASP.NET MVC), за да генерират потребителски интерфейс, който е интуитивен и лесен за навигиране. Изгледите получават данните от контролерите и ги показват по начин, който улеснява потребителя в търсенето и разглеждането на филмите в каталога.
- Контролери (Controllers): Контролерите действат като посредници между потребителските заявки, моделите и изгледите. Когато потребител направи заявка (например търсене на филм, филтриране по жанр или добавяне на ревю), контролерът я получава и изпълнява необходимите операции чрез комуникация с моделите. След обработката, контролерът избира подходящ изглед и му предоставя всички необходими данни за визуализацията на резултатите.



6.5 Използвани езици за програмиране

1) C# - C# е обектно-ориентиран език за програмиране, разработен от Microsoft, като част от софтуерната платформа .NET. Стремехът още при създаването на C# езика е бил да се създаде един прост, модерен, обектно-ориентиран език с общо предназначение. Основа за C# са C++, Java и донякъде езици като Delphi, VB.NET и C. Той е проектиран да балансира мощност (C++) с възможност за бързо разработване (Visual Basic и Java). Те представляват съвкупност от дефиниции на класове, които съдържат в себе си методи, а в методите е разположена програмната логика – инструкциите, които компютърът изпълнява.

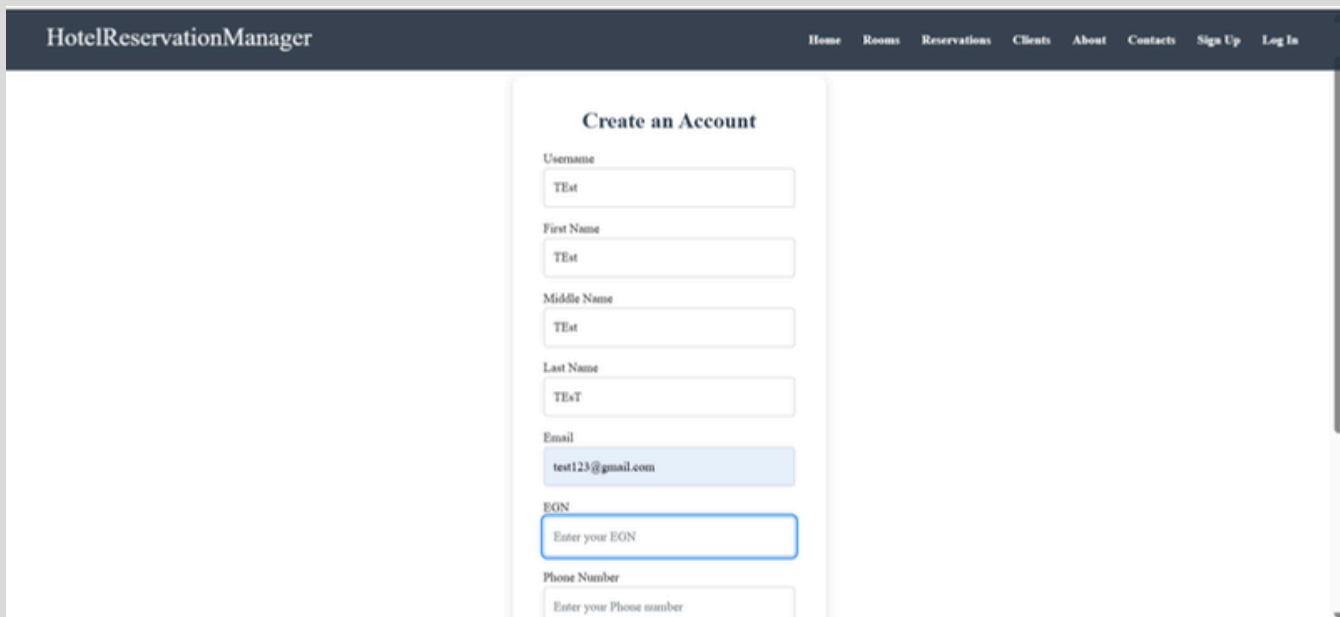
2) T-SQL (Transact-SQL) - е диалект на SQL (Structured Query Language), използван от SQL Server на Microsoft и Azure SQL база данни. T-SQL включва всички стандартни SQL функции, като заявки и манипулиране на данни в релационни бази данни, но включва и допълнителни функции, специфични за SQL Server. Някои от допълнителните функции, предлагани от TSQL, включват съхранени процедури, тригери, дефинирани от потребителя функции и отчети за контрол на транзакциите. Тези функции помагат на разработчиците да пишат сложни и ефективни приложения за бази данни и им позволяват да се възползват напълно от възможностите на SQL Server.

3) HTML/CSS - Използват се за изграждане и стилизиране на потребителския интерфейс, като осигуряват структурирано и визуално привлекателно представяне на съдържанието.

6.6 Описание на приложението

6.6.1 Register

Register.cshtml: Форма за регистрация на нови потребители, където се въвежда основна информация (потребителско име, имейл, парола и т.н.).



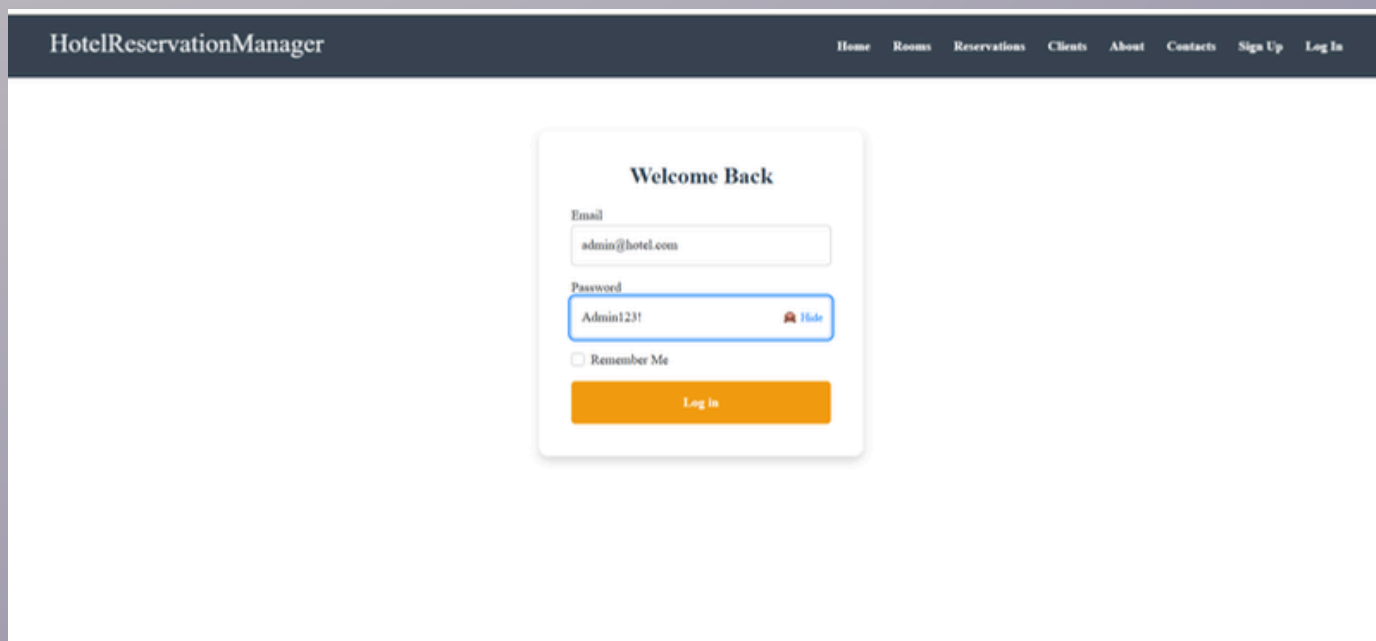
The screenshot shows a web application titled "HotelReservationManager" with a dark navigation bar containing links: Home, Rooms, Reservations, Clients, About, Contacts, Sign Up, and Log In. The main content area features a "Create an Account" form with the following fields: Username (containing "TEst"), First Name (containing "TEst"), Middle Name (containing "TEst"), Last Name (containing "TEsT"), Email (containing "test123@gmail.com"), EGN (with a placeholder "Enter your EGN"), and Phone Number (with a placeholder "Enter your Phone number").

Ако регистрацията е успешна, потребителят може да бъде автоматично вкаран в системата (login) или пренасочен към страницата за вход, в зависимост от настройките на приложението. При неуспешна регистрация (например дублиращо се потребителско име или несъответствие при паролата) се връща съобщение за грешка и формата се показва отново за корекция на данните.

6.6.2 LoginForm

Login.cshtml: Форма за вход в системата, където потребителите въвеждат своите идентификационни данни.

“LoginForm” е форма, използвана за влизане в системата. Тя включва текстови полета за въвеждане на имейл и парола на потребителя. Полето за парола е настроено така, че да приема за валидни само налични в базата потребители. Тази настройка гарантира, че паролата остава скрита и защитена от всякакви неоторизирани зрителни. Въпреки това, за да се даде възможност на потребителите да видят въведената от тях парола, се включва и иконката за показване на парола под формата на изображение на око. Като се щракне върху иконата, потребителят може да превключи видимостта на въведената от тях парола, което му позволява да види текста, който е въвел, вместо само точките. Тази функция е полезна в случай, че потребителят е забравил паролата си или я е написал неправилно, тъй като може визуално да потвърди текста, който е въвел, преди да щракне върху бутона за влизане. Щракването върху бутона „Login“ потвърждава информацията на потребителя.



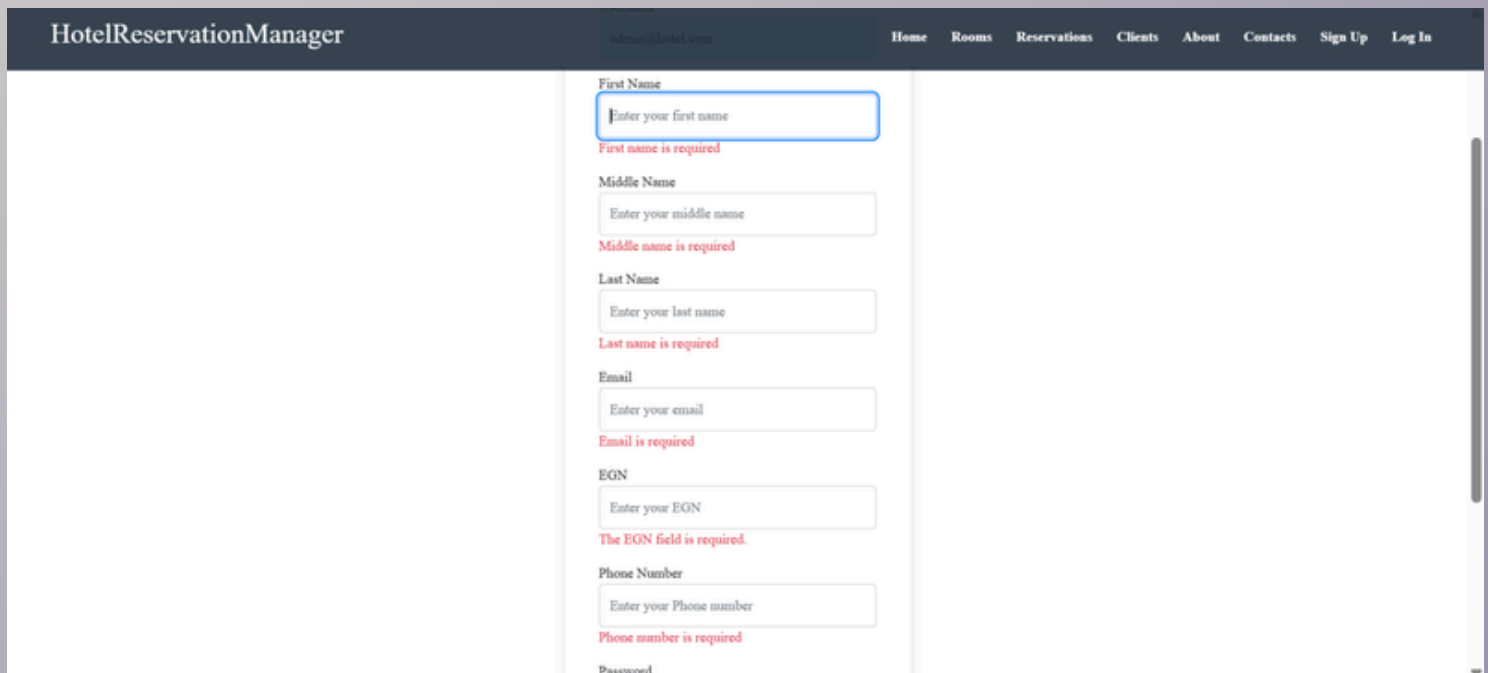
The screenshot shows a web application titled "HotelReservationManager" with a dark navigation bar containing links: Home, Rooms, Reservations, Clients, About, Contacts, Sign Up, and Log In. The main content area features a "Welcome Back" login form. The form includes an "Email" field with the text "admin@hotel.com", a "Password" field with the text "Admin123!" and a "Hide" icon, a "Remember Me" checkbox, and a prominent orange "Log in" button.

Регистрация и Вход: Потребителите могат да създават акаунт и да влизат в системата. След успешен вход имат достъп до допълнителни функции, като могат да видят допълнителни страници (Users, Clients), могат да се резервират стая и т.н.

Тест за опит за влизане с невалидни стойности

При опит за изпращане на формата без да са попълнени коректно полетата, потребителят няма да има възможност да се регистрира. Всички полета са задължителни като:

- Паролата трябва да съдържа 8 или повече символа (главни и малки букви, цифри) — това трябва да се редактира

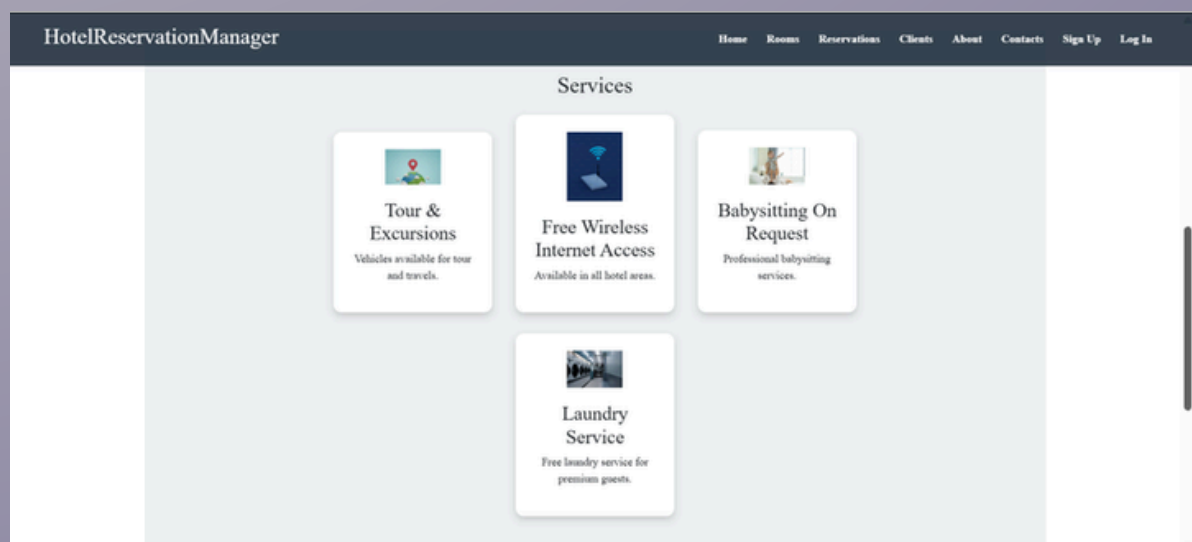


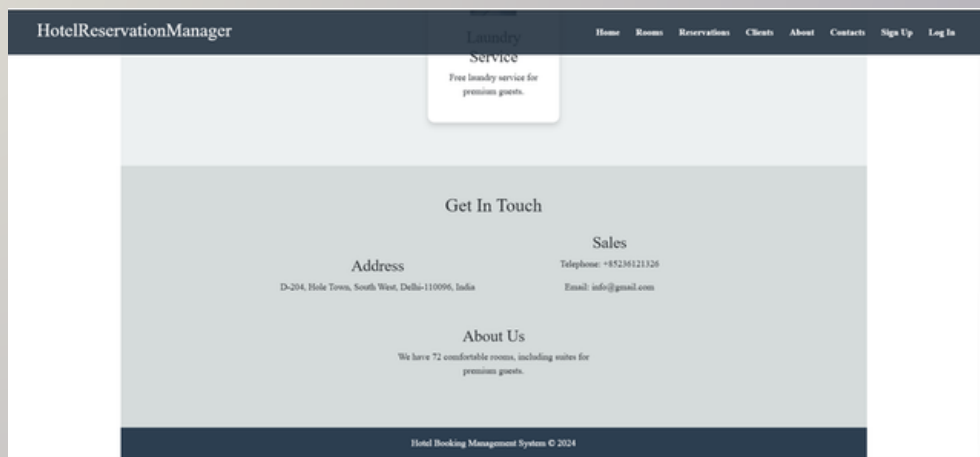
The screenshot shows the registration form for HotelReservationManager. The form includes the following fields and their validation messages:

- First Name:** Enter your first name. Validation: First name is required.
- Middle Name:** Enter your middle name. Validation: Middle name is required.
- Last Name:** Enter your last name. Validation: Last name is required.
- Email:** Enter your email. Validation: Email is required.
- EGN:** Enter your EGN. Validation: The EGN field is required.
- Phone Number:** Enter your Phone number. Validation: Phone number is required.
- Password:** (Field is visible but no validation message is shown in the image).

6.6.3 Home Page

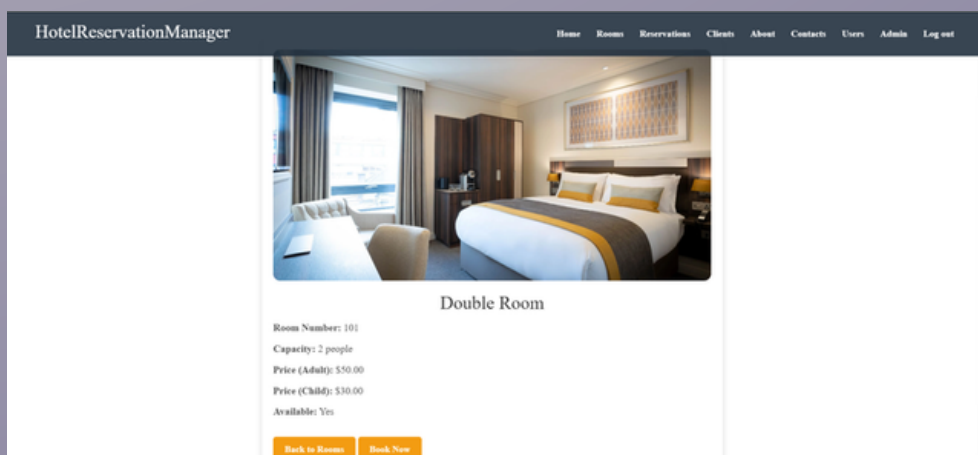
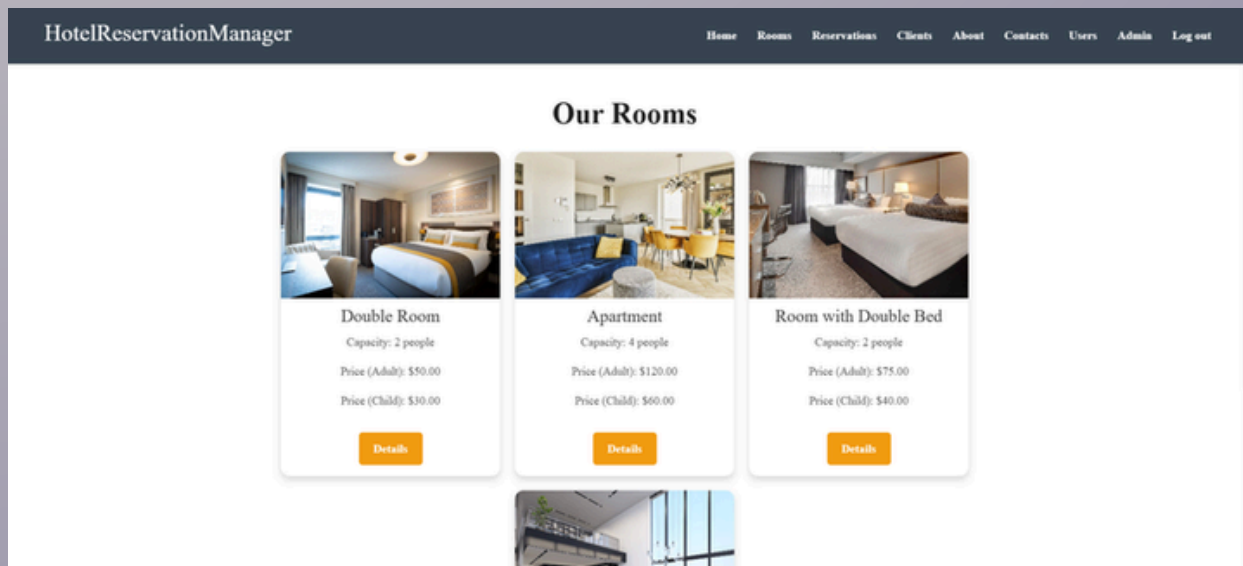
След успешно регистриране потребителят има достъп до “Home page” на приложението, който съдържа информация за услугите на хотела, както допълнителна информация като телефонен номер, адрес и т.н.



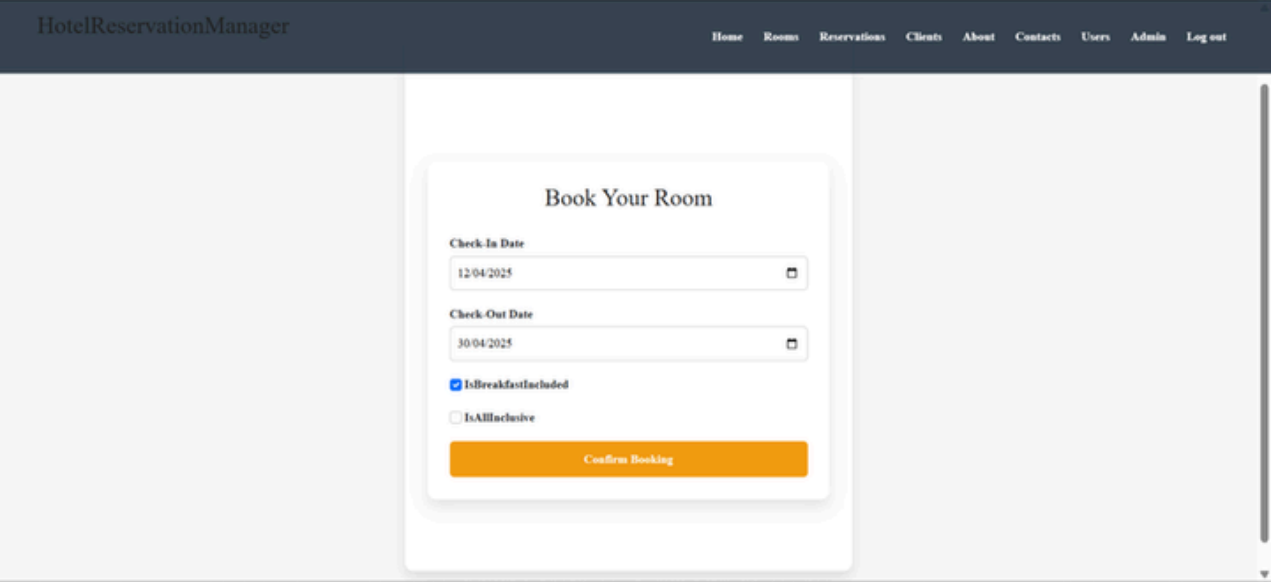


6.6.4 Rooms

Когато изберем от навигационното меню страницата Rooms, ние отиваме в страницата за стаите, където се показват всички стаи от базата. Имаме бутон "Details", като го натиснем ни извежда в страницата, в която може да видим всички детайли само на избраната стая.

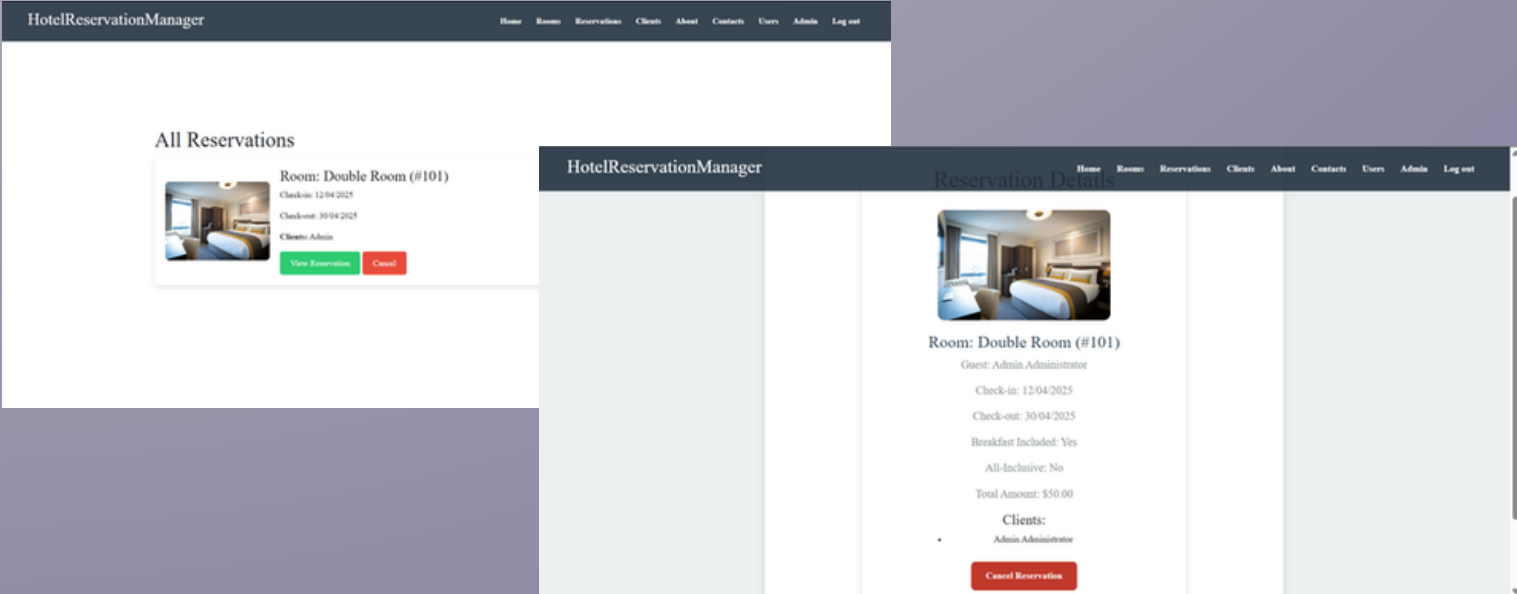


В страницата за детайлите имаме бутон "Book now". Той ни води към нова страница, където потребителят може да избере от коя дата до коя дата да му е резервацията за определената стая, дали да включва закуска или да е all Inclusive и следовател чрез бутона "Confirm Booking" може да си потвърди резервацията.



В страницата за детайлите имаме бутон "Book now". Той ни води към нова страница, където потребителят може да избере от коя дата до коя дата да му е резервацията за определената стая, дали да включва закуска или да е all Inclusive и следовател чрез бутона "Confirm Booking" може да си потвърди резервацията.

6.6.5 Reservations



Страницата Reservations предоставя на логнатите потребители възможност да преглеждат, създават, редактират и изтриват хотелски резервации в системата. Това е една от основните функционалности на приложението, достъпна за всички потребители, независимо от тяхната роля. Потребителите имат достъп до списък с всички съществуващи резервации.

Детайлен изглед на резервация

При преглед на дадена резервация се виждат всички свързани с нея данни, включително:

- списък с клиентите;
- детайли за избраната стая;
- избрани услуги и др.

6.6.6 Users

Страницата „Users” показва всички потребители, които са в системата за хотелските резервации. В табличен вид потребителят вижда всички други активни потребители, а админът може да добави, редактира и изтрива потребителите от базата чрез съответните бутони, които водят към отделните страници. Страницата има търсачка, от която може да търсят потребителите по потребителско име, собствено име, бащино име, фамилно име или имейл. Също така в страницата имаме и функцията за странициране.

HotelReservationManager

HomeRoomsReservationsClientsAboutContactsUsersAdminLog out

Users Management

Users

UserName	FirstName	LastName	Email	HireDate	IsActive	Actions
admin	Admin	Administrator	admin@hotel.com	06.04.2025	Yes	<input type="button" value="Edit"/> <input type="button" value="Delete"/>

Page: 1 of 1

1

Create New User

HotelReservationManager

[Home](#)[Rooms](#)[Reservations](#)[Clients](#)[About](#)[Contacts](#)[Users](#)[Log out](#)

Users Management

Users

Search users...

Search

UserName	FirstName	LastName	Email	HireDate	IsActive
admin	Admin	Administrator	admin@hotel.com	06.04.2025	Yes
ivm	ivm	ivmdev	ivm123@gmail.com	06.04.2025	Yes

Page: 1 of 1

1

HotelReservationManager

[Home](#)[Rooms](#)[Reservations](#)[Clients](#)[About](#)[Contacts](#)[Users](#)[Admin](#)[Log out](#)

UserName

admin@hotel.com

Password

FirstName

MiddleName

LastName

Email

EGN

PhoneNumber

HireDate

dd/mm/yyyy

☐ IsActive

Back to List

Create

HotelReservationManager

[Home](#)[Rooms](#)[Reservations](#)[Clients](#)[About](#)[Contacts](#)[Users](#)[Admin](#)[Log out](#)

FirstName

Admin

MiddleName

System

LastName

Administrator

Email

admin@hotel.com

EGN

0000000000

HireDate

06/04/2025 00:00

☒ IsActive

Back to List

Save

6.6.7 Clients

Страницата „Clients” показва всички клиенти. Системата има изглед, от който могат да се разгледат всички предишни резервации за даден клиент (отново със странициране и филтриране по собствено и фамилно име). Може от бутона "View Reservations" да видите за всеки отделен клиент съответните направени резервации.

HotelReservationManager

HomeRoomsReservationsClientsAboutContactsUsersAdminLog out

Clients Management

Clients

Name	Phone	Email	Adult	Actions
Alexander Atanasov	0876543210	alex.atanasov@example.com	Yes	<input type="button" value="View Reservations"/>
Admin Administrator	0888888888	admin@hotel.com	Yes	<input type="button" value="View Reservations"/>
Stefka Dimitrova	0896543210	stefka.dimitrova@example.com	Yes	<input type="button" value="View Reservations"/>

Page: 1 of 1

1

HotelReservationManager

HomeRoomsReservationsClientsAboutContactsSign UpLog In

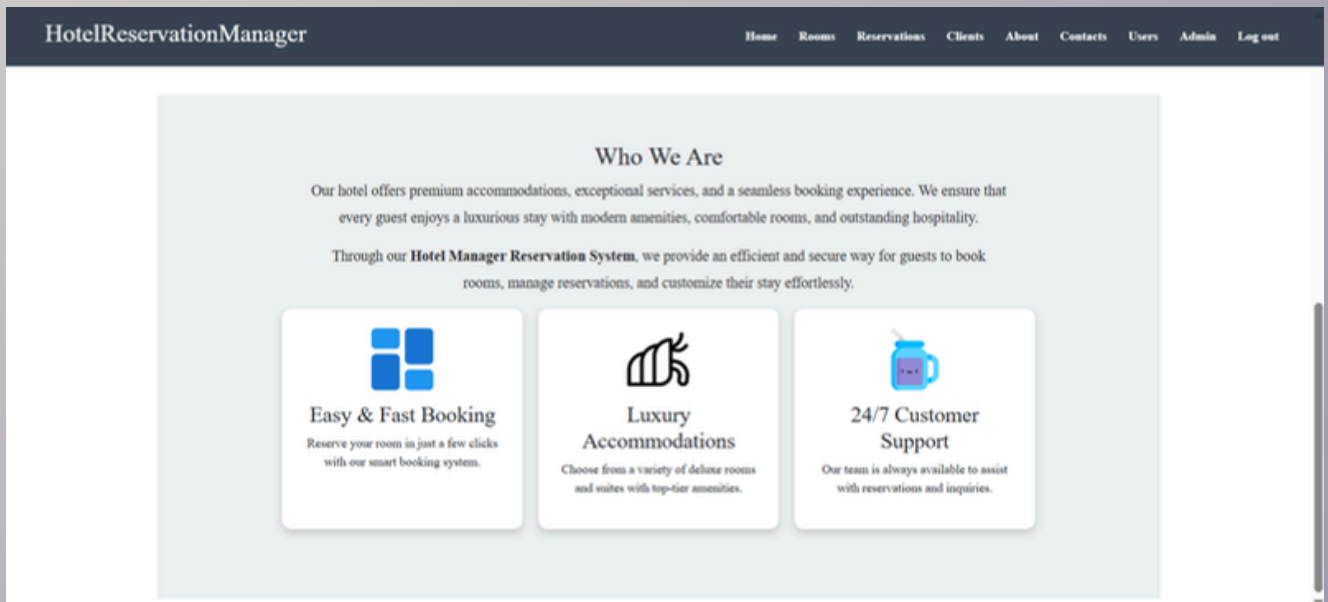
Reservations for ivan ivanov

Reservations

Room	Check In	Check Out
103	13/04/2025	27/04/2025
102	12/04/2025	06/04/2025

6.6.7 About us

Страницата About Us предоставя на потребителите детайлна информация за ценностите, начина на работа и визията на нашето уеб приложение и предимствата на приложението.

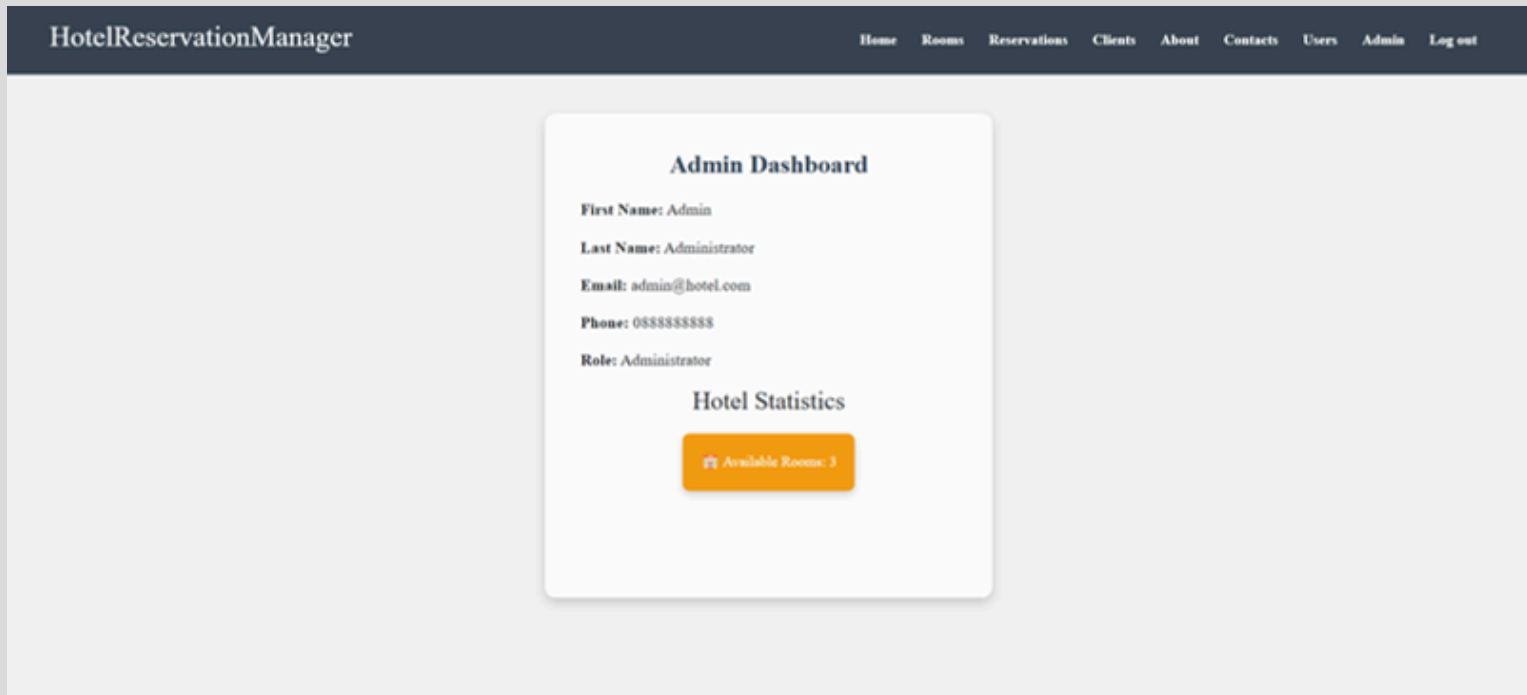


6.6.8 Contacts

Страницата Contact предоставя лесен и бърз начин за свързване с екипа на HotelReservationManager. Тя съдържа ясна и изчистена информация, която позволява на потребителите да намерят всички необходими данни за контакт:



6.6.9 Admin page



6.6.10 Admin account

Администраторският акаунт е специализиран потребителски профил с разширени привилегии, предназначени за управление на съдържанието в каталога на филмите. Този акаунт има пълен контрол върху CRUD операциите (създаване, четене, редактиране и изтриване) на записи.

6.6.9 Admin account

Основни характеристики и функционалности:

1. Управление на потребители: Администраторът има правото да създава нови потребители в системата, както и да ги редактира или изтрива. Това включва възможността да добавя нови служители (потребители), да задава имена, роли, данни за контакт, да определя дали акаунтът е активен или неактивен и т.н. Освен това, администраторът може да търси и да преглежда съществуващите потребители, както и да има пълен контрол върху всички CRUD операции, свързани с тях.
2. Управление на резервации: Администраторът има пълен достъп до резервациите, които са направени в хотела. Той може да преглежда и модифицира резервации, да изтрива стари или невалидни записи и да има пълен контрол върху настройките за дължимите суми и статуси на резервациите.

Този акаунт е от съществено значение за поддържането и обновяването на каталога, като гарантира, че информацията за филмите е винаги актуална и точна, а управлението на съдържанието става по-лесно и ефективно.

7. Развитие и нововъведения

През времето, с което разполагахме, успяхме да реализираме основните функционалности на приложението за управление на хотелски резервации. Въпреки че сме постигнали значителен напредък, нашият екип ще продължи да усъвършенства приложението, за да го направим още по-функционално и удобно за потребителите. Планираме да добавим следните основни подобрения:

7.1. Страница за наличност на стаи по дати:

Ще бъде добавена нова страница, която ще показва актуалната наличност на стаи за различни периоди. Тази страница ще включва:

- Списък на стаите с информация за свободни и заети дати.
- Филтриране по тип стая, капацитет и цени.
- Възможност за бързо резервиране на налична стая от директния изглед.

7.2. Добавяне на "Любими стаи" за клиентите:

Ще добавим функционалност, която ще позволява на клиентите или служителите да добавят предпочитани стаи в списък с "Любими стаи". Това ще позволи:

- Лесен достъп до често предпочитаните стаи при създаване на нови резервации.
- Потребителите да съхраняват стаи, които често резервират или които отговарят на техните специфични изисквания.

7.3. Разширено управление на резервации:

Ще разширим възможностите за управление на резервациите чрез добавяне на нови функции като:

- Модифициране на съществуващи резервации (промяна на дати, тип стая, добавяне на услуги и др.).
- Отказ на резервация и автоматично възстановяване на стаите като свободни.
- Изпращане на напомнания и известия на клиентите за предстоящото настаняване и освобождаване.

8. Заключение

Проектът "Hotel Reservations Manager" беше успешно реализиран и постигна всички поставени цели. Чрез разработката на това уеб приложение успяхме да създадем ефективен и интуитивен инструмент за управление на хотелски резервации, който позволява на потребителите да резервират стаи, управляват клиентски данни и контролират резервациите по лесен и удобен начин. Приложението е изградено с помощта на съвременни технологии като ASP.NET, Entity Framework и SQL Server, което гарантира неговата стабилност, сигурност и мащабируемост. Системата интегрира ключови функционалности, включително управление на потребителите, създаване и управление на резервации, както и филтриране на стаи. Тези възможности правят приложението полезно както за обикновени потребители, така и за администратори, осигурявайки ефективно управление на хотелския бизнес.

Този проект ни позволи да развием уменията си в областта на уеб разработката, работата с бази данни и изграждането на сложни системи за управление на данни. Очакваме, че приложението ще бъде полезно за хотелските мениджъри и служители, като улесни ежедневните им операции и подобри обслужването на клиентите. В бъдеще, планираме да добавим нови функционалности и подобрения, които ще увеличат ефективността и удобството на приложението.

9. Използвана литература

9.1 Книги

1. **ASP.NET Core in Action" – Andrew Lock**

- Подробно ръководство за разработка на уеб приложения с ASP.NET Core, което обяснява основни концепции като MVC архитектурата, автентикация, бази данни и работа с API.

2. **"Entity Framework Core in Action" – Jon P. Smith**

- Практически насочена книга за работа с ORM Entity Framework, използван за управление на базата данни в нашето уеб приложение.

9.2 Линкове


1.Официална документация на Microsoft за ASP.NET Core и MVC

 <https://learn.microsoft.com/en-us/aspnet/core/>

2.Entity Framework Core – официална документация

 <https://learn.microsoft.com/en-us/ef/core/>

3.Видеоурок: ASP.NET Core MVC Tutorial for Beginners (от IAmTimCorey)

 <https://www.youtube.com/watch?v=fmvcAzHpsk8>

4. Entity Framework Core Code First – пълно ръководство (от dotnet)

 <https://www.youtube.com/watch?v=Ofg7HnKgTsg>


5.ASP.NET Core Web API – Как да изградим REST API (от Les Jackson)

 <https://www.youtube.com/watch?v=pS8OID3yk70>

6.Как да работим с ASP.NET Identity (регистрация и логин система) (от Code Maze)

 <https://www.youtube.com/watch?v=GY1fQH4XWM>

7.Изграждане на уеб приложение с .NET Core и React (от Traversy Media)

 <https://www.youtube.com/watch?v=Fdf5aTYRW0E>

8.Видеоурок за работа с Entity Framework Core Migrations (от Coding Droplets)

 <https://www.youtube.com/watch?v=vS7i4zKsEKI>

