

E-COMMERCE PLATFORM DEVELOPMENT

*Assoc. Prof. PhD Dimitar Minchev, Burgas Free University, mitko@bfu.bg
Konstantin Kirchev, graduate student, konstantin1230@yahoo.com*

Abstract: This publication presents the development of an e-commerce platform on which various organic food producers from all over the country can offer their products. Through it, the production will be available to the end user without the goods going through resellers and becoming more expensive. In this way, the overall e-commerce market in Bulgaria will be improved.

Keywords: e-commerce, Microsoft, Azure, ASP.NET, SignalR.

РАЗРАБОТКА НА ПЛАТФОРМА ЗА ЕЛЕКТРОННА ТЪРГОВИЯ

*доц. д-р Димитър Минчев, Бургаски свободен университет, mitko@bfu.bg
Константин Кирчев, дипломант, konstantin1230@yahoo.com*

Абстракт: Настоящата публикация представя разработване на платформа за електронна търговия на която различни производители на био храни от цялата страна да предлагат продукцията си. Чрез нея продукцията ще бъде достъпна до крайният потребител без стоката да преминава през прекупвачи и да се оскъпява допълнително. По този начин ще се подобри цялостният пазар на електронната търговия в България.

Ключови думи: e-commerce, Microsoft, Azure, ASP.NET, SignalR.

Въведение

Електронната търговия води началото си от 60-те години на миналия век, когато компаниите използват електронния обмен на данни, за да улеснят прехвърлянето на документи.

Първата трансакция за електронна търговия, включваща плащане на средства за дадена стока, обаче е осъществена след появата на интернет на 11 август 1994 г., когато двама приятели извършват онлайн продажба на компактдиск на Sting чрез уебсайта за търговия на дребно NetMarket. Оттогава насам в тази ниша настъпиха драстични промени и това можем ясно да видим чрез създаването и експоненциалния растеж на компании като Amazon, eBay, Etsy и др. Amazon, например, стартира една година след продажбата на компактдиска през 1995 г. и се превърна в нарицателно име. През 1999 г. световната електронна търговия достигна стойност от 150 млрд. долара. Само няколко години по-късно, през 2003 г., Amazon отчита първата си печалба. През 2015 г. на Amazon се пада повече от половината от растежа на нишата, а през 2017 г. глобалните продажби на дребно в областта на електронната търговия достигат 2,304 трилиона долара, което е с 24,8% повече в сравнение с предходната година. През същата година, глобалните трансакции в областта на е-commerce генерират 29,267 трилиона долара, като по-

голямата част от продажбите се дължи на трансакциите между предприятия, следвани от продажбите между предприятия и потребители.

Пандемията от Ковид-19 бе пагубна за развитието на редица бизнеси. Много хора останаха без работа, а някои собственици на търговски обекти и ресторанти бяха принудени да затворят завинаги. Един от отраслите, който не само че не пострада, а преживя своя възход, беше онлайн търговията.

Използвани технологии

1. ASP.NET Core

ASP.NET Core е новата версия на уеб рамката ASP.NET, насочена главно към .NET Core платформа. ASP.NET Core е безплатна, с отворен код и междуплатформена рамка за изграждане на базирани на облак приложения, като уеб приложения, IoT приложения и мобилни бекендове. Той е проектиран да работи в облака, както и на място. Също като .NET Core, той е проектиран модулно с минимални режимни разходи и след това могат да се добавят други по-разширени функции като NuGet пакети според изискванията на приложението. Това води до висока производителност, изисква по-малко памет, по-малък размер за разгръщане и лесна за поддръжка. ASP.NET Core е рамка с отворен код, поддържана от Microsoft и общността.

2. Azure

В основата си Azure е публична платформа за изчисления в облак – с решения, включително инфраструктура като услуга (IaaS), платформа като услуга (PaaS) и софтуер като услуга (SaaS), които могат да се използват за услуги като анализи, виртуални компютри, съхранение, работа в мрежа и много други. Може да се използва за замяна или допълване на вашите локални сървъри.

3. SignalR

SignalR е безплатна софтуерна библиотека с отворен код за Microsoft ASP.NET, която позволява на сървърния код да изпраща асинхронни известия до уеб приложения от страна на клиента. Библиотеката включва JavaScript компоненти от страна на сървъра и от страна на клиента.

ASP.NET SignalR е библиотека за ASP.NET разработчиците, за да добавят уеб функционалност в реално време към своите приложения. Уеб функционалността в реално време е способността да има код от страна на сървъра да изпраща съдържание към свързаните клиенти, както се случва, в реално време.

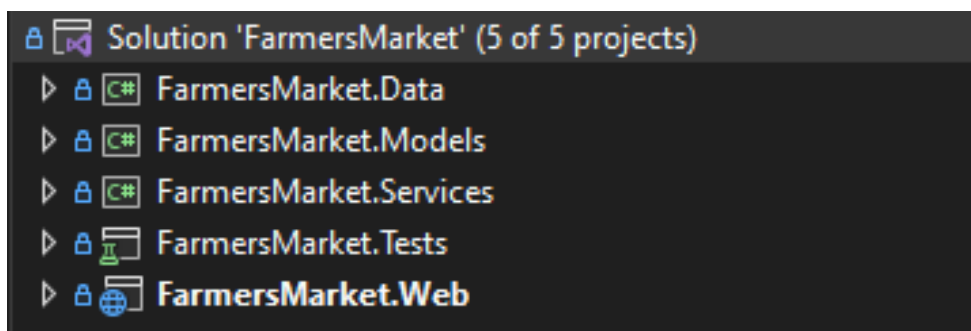
SignalR се възползва от няколко транспорта, като автоматично избира най-добрия наличен транспорт предвид възможностите на клиента и сървъра. SignalR се възползва от WebSocket, HTML5 API, който позволява двупосочна комуникация между браузъра и сървъра. SignalR ще използва WebSockets под кориците, когато е наличен, и грациозно ще се върне към други техники и технологии, когато не е, докато кодът на приложението остава същият.

SignalR също така предоставя прост API на високо ниво за извършване на RPC от сървър до клиент (извикване на JavaScript функции в браузъра на клиента от .NET код от страна на сървъра) в ASP.NET приложение, както и добавяне на полезни куки за управление на връзката, като събития за свързване/изключване, групиране на връзки, оторизация.

Платформа за електронна търговия

Разработена е авторска платформа за електронна търговия на която различни производители на био храни да предлагат продукцията си и тя да бъде достъпна до крайният потребител без стоката да преминава през прекупвачи и да се оскъпява допълнително.

Платформата се нарича Фермерски Пазар (Farmer's Market) и е достъпна на следният адрес <http://farmersmarkets.azurewebsites.net>. Кодът е достъпен в Github на следният адрес <https://github.com/KonstantinKirchev/FarmersMarket-ASP.NET-Core>. Проектът е изработен на платформата ASP.NET Core и следва всички добри практики на архитектурата MVC (Model View Controller) Модел Изглед Контролер. Проектът е разделен на пет слоя:



Фиг.1. Архитектура на приложението

1. FarmersMarket.Data

В този слой се извършва връзката между базата данни и приложението. Има един главен файл FarmersMarketDbContext.cs който държи връзката с базата данни.

В него има списък от DbSet-ове които сочат към Моделите и представляват отделни таблици в базата данни. При добавяне или премахване на DbSet трябва да се изпълни командата Add-Migration последвана от командата Update-Database, за да се актуализира базата данни.

Технологията която се използва се нарича Entity Framework (EF) Core. Тя е лека, разширяема, с отворен код и кросплатформена версия на популярната технология за достъп до данни на Entity Framework. EF Core може да служи като обектно-реляционен картограф (O/RM) които съпоставя (mapping) обекти към таблици в базата данни.

Също така в този файл има и един метод OnModelCreating който можем да пренапишем (override) и в които описваме връзките между отделните таблици в базата данни.

При промяна на Моделите, за да се актуализира базата данни трябва да се изпълни следната команда в терминала Add-Migration последвана от име на миграцията и след нея трябва да се изпълни командата Update-Database.

Всички миграции се записват в папка Migrations и съдържат инструкции описващи отделните таблици с колони и връзките между тях.

В Data слоя също така се намира Repository и Unit of work дизайнерски модел който позволява създаването на абстрактен слой между слоя за достъп до данни и слоя на бизнес логиката на приложението. Внедряването на тези модели може да помогне за

изолирането на приложението от промени в базата данни и може да улесни автоматизираното тестване на модули или разработване, управлявано от тестове. Всички заявки от и към базата минават през него и по този начин не сме зависими от базата данни и лесно и бързо може да се свържем към друга база от данни.

2. FarmersMarket.Models

В този слой се описват всички модели и връзките между тях. Разделен е на три основни групи модели: Entity Models – Модели описващи обектите които се използват за изграждане на базата данни; View Models – Модели които се използват за визуализиране на данни. Подават се от контролера към устройството за визуализация; Binding Models – Модели които се използват за изпращане на данни към базата данни. Подават се от устройството за визуализация към контролера.

В папка Infrastructure се намира Automapper библиотека която се използва за мапване на модели от View Models към Entity Models и Binding Models към Entity Models. В папка Enums се намират енумерации (класове които имат само свойства). Използват се за типизиране на свойствата в моделите които са изброими множества. В файла ModelsConstants.cs са описани всички низови константи които се преизползват в моделите за ограничения които се налагат на отделните свойства.

3. FarmersMarket.Services

В този помощен слой се намират всички сървиси от проекта. Състои се от един абстрактен клас Service.cs в който сетваме в конструктора аутомапера и връзката с базата данни. След това всеки конкретен сървис го наследява и преизползва аутомапера и базата данни. В папка Interfaces се намират всички интерфейси в които са изброени всички методи, но без тяхната имплементация. В папка Implementations се намират всички конкретни сървиси. Наименоването на сървисите е същото като контролерите. Всеки контролер си има сървис. Всеки сървис наследява класа Service и имплементира конкретния интерфейс. Достъпването на сървисите става в контролерите като се използват техните конкретни интерфайси с помощта на Dependency Injection (инжектиране на зависимост е модел на проектиране, при който обект получава други обекти, от които зависи).

4. FarmersMarket.Tests

В този слой се извършва тестването на цялото приложение. В файла MockContainer.cs се подготвят копия на обектите като се използва библиотеката Moq. Копията са нужни, за да се разкачим от базата данни, за да може на се направят тестове на отделни функционалности от проекта (unit tests). В папка Controllers се тестват функционалности на контролерите, а в папка Services се тестват функционалности от сървисите.

5. FarmersMarket.Web

Това е главният слой на приложението фиг.3.7. Тук се намира цялата бизнес логика, ерий (Areas), контролери (Controllers) и устройства за визуализация (Views). Program.cs е главният стартиращ файл. В него се сетват всички библиотеки и методи. Те се закачат към builder.Service. След като се закачат всички сървиси се вика методът builder.Build() който изгражда цялото приложение. След това под формата на мидълуери (middlewares) се закачат отделни методи в конкретна последователност на изпълнение. Най-накрая се вика методът app.Run() който стартира приложението. Мидълуерът е софтуер, който е сглобен в тръбопровода на приложението, за да обработва заявки и отговори.

Заклучение

Като цяло развитието на дигиталните технологии, социалните мрежи и онлайн платформите за електронна търговия предизвикват бум в развитието на онлайн и офлайн пазарите и иновациите в много други сфери като софтуерни продукти и системи (CRM, ERP, и др.), логистика, онлайн трансфери и разплащания, онлайн маркетинг и др. Високата степен на интегриране на различни технологии и иновации в онлайн платформите им позволява да събират, систематизират и анализират огромни масиви от данни – т.нар. Big Data, благодарение на което те анализират поведението на клиентите в интернет относно всички техни предпочитания и формират солидна база за осъществяване на прецизен и таргетиран онлайн маркетинг.

Използвана литература

1. Горанова, К. (2017). Продавай.бг, Капитал. бр. 31 от 4-10 август 2017.
2. Капитал. Регал № 1. Бъдещето на онлайн търговията. март 2017.
3. Капитал. Регал 2. Здравословен обрат. юни 2017.
4. Капитал. Регал № 1. Потребителят Millennial. април 2018.
5. Капитал. Най-важното. бр. 17 от 27 април - 3 май 2018 г.
6. Любенов, Л. (2016). Агромаркетинг. Русе. ISBN 978-619-207-040-3.
7. Любенов, Л. (2018a). Дигитални маркетинг трансформации, Социално-икономически анализи. В. Търново. 2018. кн. 2. ISSN 1313- 6909.
8. Lyubenov, L. (2018b). SWOT analysis of the Bulgarian beekeeping, Journal of Mountain Agriculture on the Balkans, 2018, 21 (4), 50-72, Research Institute of Mountain Stockbreeding and Agriculture, Troyan, ISSN 1311-0489 (Print), ISSN 2367-8364 (Online)
9. Vipin Jain, An Overview of Electronic Commerce (e-Commerce) – май 2021
9. <https://nmgtechnologies.com/blog/advantages-of-asp-net-core.html>
10. <https://tacitcorporation.com/>
11. https://bg.wikipedia.org/wiki/Електронна_търговия
12. <https://www.mypos.com/bg/elektronna-targoviya>