****

МИНИСТЕРСТВО НА ОБРАЗОВАНИЕТО И НАУКАТА

**ПРОФЕСИОНАЛНА ГИМНАЗИЯ “ГЕН. ВЛАДИМИР ЗАИМОВ” гр. СОПОТ**

4330 гр. Сопот, ул. ”Иван Вазов” №1, тел./факс: /03134/ 83-31, 83-32, e-mail: [pgzaimov@yahoo.com](mailto:pgzaimov@yahoo.com)

**ДИПЛОМЕН**

**ПРОЕКТ**

**Тема: РАЗРАБОТКА НА УЕБ**

**Магазин за кафе**

*Ученик: Иван Димитров Тенев*

***Професия:*** *код 481030 „Приложен програмист“*

***Специалност:*** *код 4810301 „Приложно програмиране“*

***Консултант:*** *Светлана Георгиева*

Сопот, 2023 г.

**Съдържание**

**Увод………………………………………………………………………………………..4**

**Глава1.ПРОУЧВАНЕ ПО ТЕМАТА…………………………………………………..4**

**1.1.Предпоставка за създаване на продукта………………………………………….5**

**1.2. Съществуващи решения и реализации…………………………………………..6**

**Глава2.**

**2.1. Функционални изисквания към WEB базиран......... /условието на**

**задачата /…………………………………………………………………………………8**

**2.2. Изготвяне на Use Cases Diagrams – описание на отделните на случаи на употреба……………………………………………………………………………………9**

**2.3. Потребителски интерфейс /менюта, екрани, Views/…………………………..13**

**2.4. Описание на избраната технология и софтуерните средства за разработка на приложението /технологии: *ASP.Net, ASP.Net Core, SQL Server, средства: C#, HTML, CSS, JavaScript*, *Bootstrap*/…………………………………………………… 13**

**Глава3.** **ПРОГРАМНА РЕАЛИЗАЦИЯ НА WEB БАЗИРАН…………………… 20**

**3.1. Структурата на базата от данни – Data Models………………………………. 20**

***Освен E/R диаграмата, тук се описват: данните в таблицата, ключовите полета и релациите /по кои полета и за какво се прави дадената релация/***

***3.2. Структура на MVC приложението – Controllers, Views, ViewModels, Routs – описание на стартирането и комуникация между Models – Views – Controllers/***

***3.3. Авторизация и оторизация на потребителите на приложението………… 23***

**Увод**

**Магазинът ми за кафе ,,Coffee La Grande’’ разполага с възможността да поръчате качествено италианско кафе. Това е един от малкото магазини, който предоставя поръчването на различни видове кафе онлайн. Тук ще намерите голямо разнообразие от фино, смляно на зърна, на капсули, филтърни моно дози или инстантно кафе. Независимо с каква машина разполагате във ще намерите точно това което търсите. Предлагаме огромно разнообразие от висококачествени продукти като например: Lavazza, Nesspresso , Vergnano, Kimbo, Dolce Gusto, Covim и Pera.**

**Kратко описание на тема и областта на дипломната работа**

**Гарантирам ви че нашият магазин е точното място за вас поради простата причина че продуктите и кафето са внос от Италия и качеството е гарантирано винаги. Също така поръчките се доставят максимално бързо до съответно локацията която сте ни отбелязали.**

**Цели и задачи**

**Да се проектира и реализира трислойно MVC WEB приложение, организиращо дейността на магазин за кафе „Coffee La Grande“, което трябва да поддържа CRUD операции за артикулите, вида на артикула, наличните им количества и продажбата им. Магазин за кафе „Coffee La Grande“ се нуждае от уеб сайт, чрез който да разшири дейността си, давайки възможност за продажба на своите артикули онлайн. Сайтът трябва да предлага възможност за разглеждане на видовете кафе в няколко категории, начина на купуване - на зърна или смляно и да поддържа информация за наличните количества от всеки артикул. Трябва да се осигури наличието на пазарска кошница, в която регистриран клиент добавя избрани артикули в исканото количество, като по този начин формира поръчка. Сайтът трябва да разполага със следните публично достъпни секции „Начало“, „За нас“, „Кафе“, „Промоции“, „Контакти“. Сайтът трябва да може да се администрира с помощта на администраторско меню, което да позволи добавянето на нови артикули, промяна и изтриване на съществуващи такива, създаване на промоционални оферти. Дизайнът на Сайта трябва да е адаптивен и да изглежда еднакво добре на различни устройства. Изберете аргументирано подходящи технологии.**

**ГЛАВА 1.**

**ПРОУЧВАНЕ ПО TEMATA**

* 1. **Предпоставка за създаване на продукта**

Създаването на магазин за кафе изисква няколко предпоставки, които трябва да бъдат разгледани преди да започнете да планирате бизнеса си. Ето някои от тях: Изследване на пазара - Преди да започнете да създавате магазин за кафе, трябва да проучите пазара. Трябва да разберете какви видове кафе предлагат другите магазини за кафе във вашата област, какви са цените и какви са предпочитанията на потребителите. Местоположение - Местоположението на магазина за кафе е от съществено значение за успеха на бизнеса ви. Трябва да намерите подходящо място, което да е лесно достъпно за потребителите и да има достатъчно пространство за създаване на уютна атмосфера. Финансови ресурси - За да започнете магазин за кафе, ще ви трябват финансови ресурси за наем на място, закупуване на оборудване, наемане на персонал и други разходи. Трябва да разработите бизнес план, който да ви помогне да определите необходимите финансови ресурси и да управлявате финансите на бизнеса си. Меню - Трябва да изберете какво да предлагате в своя магазин за кафе. Това може да включва различни видове кафе, напитки на базата на кафе, храни, десерти и други. Персонал - За да се грижите за своите клиенти, вашият магазин за кафе ще изисква персонал. Трябва да намерите подходящи служители, които да имат опит в областта на приготвянето на кафе и да бъдат приятелски настроени и гостоприемни. Маркетинг - За да привлечете клиенти към своя магазин за кафе, трябва да имате ефективна маркетингова стратегия. Това може да включва рекламиране в социалните медии, директен маркетинг, участие в събития и други. Трябва да се определи какви са целевата аудитория и какви маркетингови дейности ще имат най-голямо въздействие. Лицензи и разрешения - За да стартирате магазин за кафе, може да се изискват определени лицензи и разрешения от местните органи. Това може да включва лиценз за продажба на хранителни продукти, разрешение за работа с храни и напитки и други. Качество на кафето - Качеството на кафето е ключово за успеха на вашия магазин за кафе. Трябва да се уверите, че използвате висококачествено кафе и да работите с доставчици, които могат да ви предоставят качествено кафе. Дизайн на интериора - Дизайнът на интериора е също много важен за успеха на магазина за кафе. Трябва да създадете уютна и приветлива атмосфера, която да привлича клиентите. Конкуренция - Конкуренцията може да бъде много силна в областта на магазините за кафе. Трябва да направите проучване на конкуренцията във вашата област и да намерите начини да се отличите от другите магазини за кафе. Тези предпоставки трябва да бъдат разгледани сериозно, за да гарантирате успеха на вашия магазин за кафе.

1.1.1 Конкуренция

Конкуренцията на магазин за кафе може да бъде много разнообразна и зависи от местоположението и мащаба на магазина. Някои от основните видове конкуренция, които може да срещнете, включват: Други магазини за кафе - Това може да бъдат както вериги от национални или международни марки, така и местни магазини за кафе. Ресторанти и кафенета - Ресторантите и кафенетата, които предлагат кафе, могат да бъдат конкуренция за магазина ви. Барове и клубове - Баровете и клубовете, които предлагат кафе, могат да бъдат конкуренция за магазина ви, особено вечерните часове. Магазини за хранителни стоки - Магазините за хранителни стоки, които предлагат кафе на капсули или мляно кафе, могат да бъдат конкуренция за магазина ви. Онлайн магазини за кафе - Онлайн магазините за кафе могат да предлагат конкурентни цени и удобство за клиентите. Домашно приготвяне на кафе - Много хора предпочитат да приготвят кафето си в домашни условия, което може да бъде конкуренция за магазина ви. Мобилни кафета - Мобилните кафета, които предлагат кафе на пътя, могат да бъдат конкуренция за магазина ви, особено ако се намират в близост до вашия магазин. За да се издигнете над конкуренцията, трябва да се фокусирате върху уникалните предимства на вашия магазин за кафе, като например качеството на кафето, уютната атмосфера или удобното местоположение. Така ще привлечете и задържите повече клиенти и ще се отличите от конкурентите си.

* 1. **Съществуващи решения и реализации**

Има много решения и реализации, които може да се приложат в магазин за кафе, но някои от тях включват:

Използване на специализирана кафемашина - Това може да помогне за подобряване на качеството на кафето, което се сервира в магазина. Машините за кафе са налични в различни модели, които могат да отговарят на нуждите на различни магазини. Използване на качествени кафе зърна - Изборът на висококачествени кафе зърна може да допринесе за по-добро качество на кафето. Можете да изберете марки, които са известни с високо качество на кафето, и да осигурите на клиентите си различни видове кафе. Предлагане на различни видове кафе напитки - Предлагането на разнообразни кафе напитки може да помогне да привлечете клиенти и да увеличите продажбите. Можете да предложите еспресо, капучино, лате, американо и други видове напитки. Изграждане на атмосфера - Магазинът за кафе може да бъде място за срещи с приятели, работа на лаптоп или релаксация. Изграждането на уютна атмосфера може да привлече повече клиенти и да ги задържи за повече време в магазина. Организиране на кафе курсове - Магазинът може да организира кафе курсове, където клиентите могат да научат повече за различните видове кафе, начините на приготвяне и дегустацията. Това може да бъде допълнителен начин за привличане на клиенти и за повишаване на общественото влияние на магазина. Използване на социални медии - Магазинът може да използва социалните медии за рекламиране на своите продукти и услуги, както и за взаимодействие с клиентите. Предлагане на допълнителни продукти - Магазинът може да предлага и други продукти, които може да са свързани с кафето или да го допълват, като например кексове, печива, сандвичи или десерти. Това може да бъде допълнителен начин за увеличаване на продажбите. Изграждане на лоялността на клиентите - Магазинът може да предлага програма за лоялност на клиентите, която да ги мотивира да се връщат и да правят повече покупки. Това може да включва награди или бонус точки за определен брой покупки. Използване на онлайн платформи - Магазинът може да използва онлайн платформи за продажба на своите продукти и услуги, като например доставки на кафе или поръчки за кетъринг. Това може да бъде допълнителен начин за увеличаване на продажбите и за достигане до повече клиенти. Използване на устойчиви продукти - Магазинът може да използва устойчиви продукти, като например чаши и прибори от рециклируем материал, за да намали негативното въздействие върху околната среда. Това може да бъде допълнителен начин за привличане на клиенти, които са насочени към устойчивия начин на живот.

**ГЛАВА 2.**

**ПРОЕКТИРАНЕ НА СТРУКТУРАТА НА WEB БАЗИРАН….**

**2.1. Функционални изисквания към WEB базиран......... /**условието назадачата**/**

Да се реализира уеб приложение за обслужване на магазин за кафе„Coffee break“.

Приложението трябва да поддържа следните функционалности:

o Да се направят роли за работа с приложението:

 АДМИНИСТРАТОР

 Клиент

o Добавяне на нов вид кафе /с Администраторски функции/:

 Име

 Количество

 Категория /смляно, на зърна, нес/

 Вид /пакетирано, на грамаж/

 Снимка

 Цена

 Дата на въвеждане в приложението

o Добавяне на промоции /с Администраторски функции/:

 Име

 Промоционална цена /като % от цената на кафето/

 Начална дата на промоцията

 Крайна дата на промоцията

o Извличане на информация за избран артикул

o Изтриване на артикул

o Извличане на всички артикули по зададен критерий / категория

o Регистрация на клиент:

 Име

 Фамилия

 Електронна поща

 Телефон за контакт

o Пазаруване /записване на артикули в пазарска кошница/:

 Номер на клиент

 Номер на артикул

 Количество стока /брой артикули/

 Единична цена на артикул

Пазарската кошница трябва да може да се визуализира на отделен екран, като избраните от клиента кафе са подредени в добре форматиран списъчен вид заедно с техните количества.

Да се пресмята и общата стойност на поръчката, която може да се финализира с бутон

„Поръчай“. Дизайнът е напълно свободен и авторски. След финализиране на поръчка:

- Пазарската кошница се изпразва;

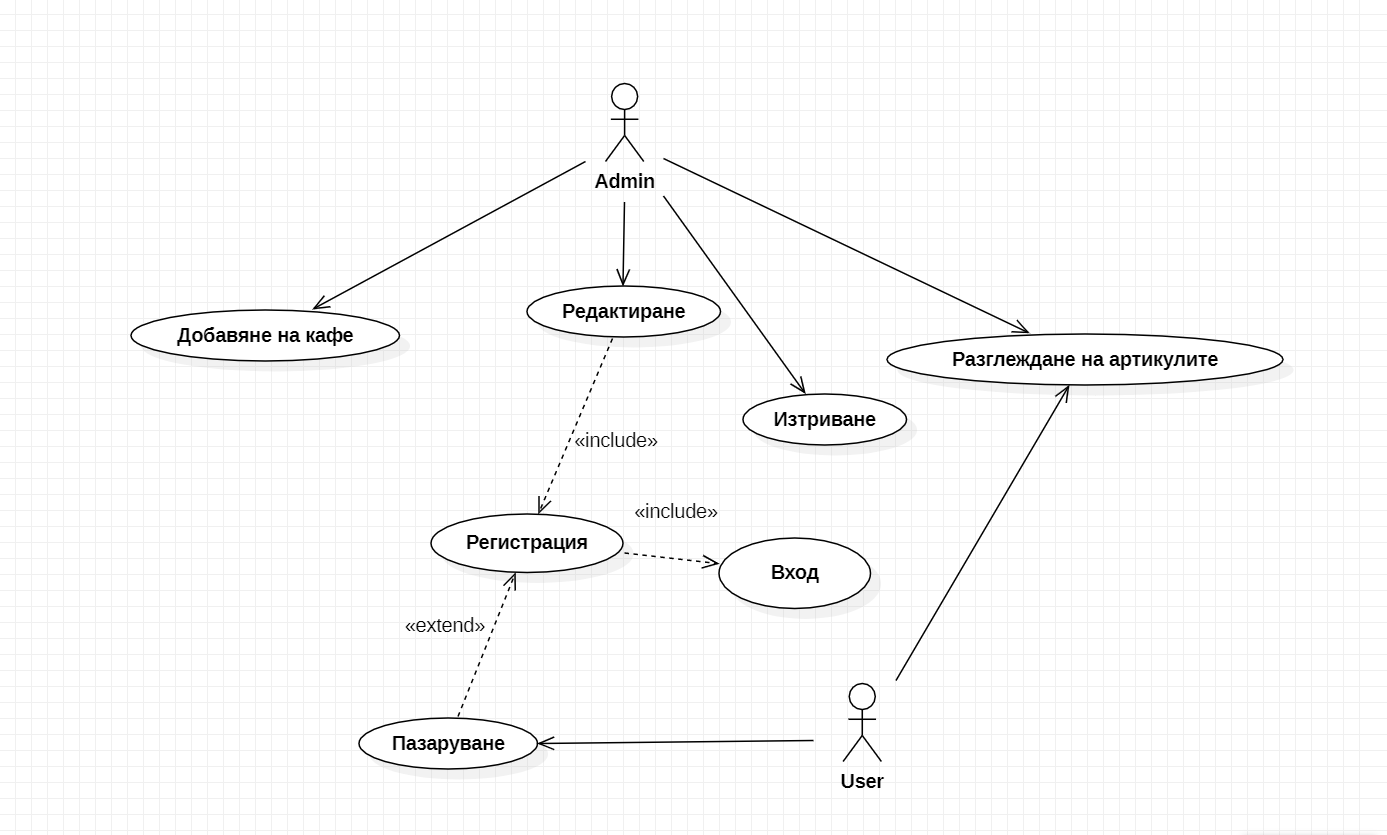
- Поръчката се записва в отделна таблица в БД на приложението /№ поръчка, списък артикули с техните количества и цени, крайна сума за плащане, номер на клиента, който е поръчал/

След продажба количествата в складовата наличност трябва да се актуализират автоматично за всеки артикул. Да се изгради трислойно ASP.NET MVC приложение с БД, което поддържа гореописаната функционалност на различни екрани /страници/.

Страниците да имат структурата на HTML5 и да са форматирани със CSS3, Java Script.

Да се използва HTTP комуникация.

**2.2. Изготвяне на Use Cases Diagrams – описание на отделните на случаи на употреба**

****

Системата се състои от следните функционални звена:

• Регистрационен модул – регистрация на потребители от тип клиент;

• Потребителски профил – попълване на лични данни на клиента;

• Администраторски профил – идентифицира потребители със специфични функции

• Секция „преглед на продукти” – позволява на потребителите да разгледат предлаганите продукти/стоки

• Секция „поръчки на продукт“ - позволява на потребителя да закупи избран от него продукт

• Секция „добавяне на продукти“ - позволява на администраторите да добавят нов артикул

• Секция „отчет“-позволява на администраторите да извлекат информация за направените поръчки на продукти избрани от клиентите

Случаи на употреба

Случай на употреба: добавяне на кафе

Цел: Задейства се, когато възникне нужда от обновяване на налични артикули

Категория: основна

Предусловия: вход в системата

Успешни подусловия: успешно добавяне на нов продукт в системата

Неуспешни подусловия: -

Актьори: администратор

Случай на употреба: Регистрация в системата

Цел: Задейства се, когато потребител желае да използва пълната функционалност на даденото/създаденото от администратора приложение

Категория: основна

Предусловия: отваряне на приложението в неговата пълна форма

Успешни подусловия: успешно регистриране в системата

Неуспешни подусловия: потребителското им ,което може да бъде вече заето

Актьори: клиент

Случай на употреба: Вход в системата

Цел: Задейства се, когато даден потребител направи опит за действие, което е ограничено. Тогава потребителят е подканен да въведе потребителското си име и парола, за да продължи в системата, тоест той като ги въведе доказва че е истинският потребител.

Категория: основна

Предусловия: регистрация в системата

Успешни подусловия: успешно влизане в системата

Неуспешни подусловия: грешно потребителско име или парола или несъществуващ акаунт

Актьори: администратор и клиент

Случай на употреба: Редактиране

Цел: Задейства се, когато се иска да се обнови една или повече информации към по-новата им версия

Категория: основна

Предусловия: редактиране

Успешни подусловия: успешно редактиране на артикулът

Неуспешни подусловия: -

Актьори: администратор

Случай на употреба: Изтриване

Цел: Задейства се, когато се иска един продукт да бъде премахнат от базата данни

Категория: основна

Предусловия: Изтриване

Успешни подусловия: успешно изтриване на продукта от системата

Неуспешни подусловия: -

Актьори: администратор

Случай на употреба: Разглеждане на артикулите

Цел: Да може клиентите да разглеждат продуктите за да знаят какво да си поръчат

Категория: основна

Предусловия: Разглеждане на артикулите в уебсайта

Успешни подусловия: Разглеждането на продуктите е с успех и клиентите са поръчали нещо

Неуспешни подусловия: -

Актьори: администратор и клиент

Случай на употреба: Пазаруване

Цел: Задейства се, когато един клиент иска да поръча и плати продуктите които са харесали от разглеждането им

Категория: основна

Предусловия: плащане на покупките

Успешни подусловия: успешно плащане на поръчания артикули

Неуспешни подусловия: -

Актьори: клиент

2.3. Потребителски интерфейс /менюта, екрани, Views/

**2.4. Описание на избраната технология и софтуерните средства за разработка на приложението /**технологии: *ASP.Net, ASP.Net Core, SQL Server, средства: C#, HTML, CSS, JavaScript*, *Bootstrap***/**

2.4.1 ASP.NET е платформа за разработка на уеб приложения, разработена от Microsoft. Тя използва програмен език C# или VB.NET и работи върху. NET Framework или. NET Core. ASP.NET осигурява много инструменти и библиотеки за уеб разработка, като например богата колекция от контроли за потребителски интерфейс, модели на данни, поддръжка на бази от данни, сигурност, управление на сесии и други. Тези инструменти правят разработката на уеб приложения по-лесна и по-бърза. ASP.NET има много различни компоненти, като например ASP.NET Web Forms, ASP.NET MVC, ASP.NET Web API, SignalR и други. Те позволяват на разработчиците да изберат най-подходящия за тях начин на разработка на уеб приложения. ASP.NET е много популярна платформа за уеб разработка и се използва широко от програмисти и компании по целия свят. Ето още някои важни черти на ASP.NET: Отворен код: ASP.NET е с отворен код и има голяма общност от програмисти, които работят върху него. Това дава възможност за използване на безплатен софтуер и споделяне на знания и опит. Мащабируемост: ASP.NET е мащабируема платформа и може да се използва за създаване на големи уеб приложения, които обслужват много потребители. Сигурност: ASP.NET предлага много възможности за сигурност, като например вградена защита от SQL инжекции, XSS атаки и други. Удобство за разработка: ASP.NET има интегрирана среда за разработка (IDE) - Visual Studio, която предоставя графичен интерфейс за създаване на уеб приложения, както и допълнителни инструменти за ускоряване на разработката. Поддръжка на различни устройства: ASP.NET поддържа различни устройства, като например мобилни телефони и таблети, чрез адаптивен дизайн или отделни мобилни приложения. Поддръжка на мултиезикови уеб сайтове: ASP.NET позволява лесна локализация и поддръжка на мултиезикови уеб сайтове. Интеграция с други продукти на Microsoft: ASP.NET се интегрира лесно с други продукти на Microsoft, като например Microsoft SQL Server, SharePoint, Exchange и други. В общи линии, ASP.NET е мощна и гъвкава платформа за разработка на уеб приложения, която може да отговори на много различни нужди и изисквания на програмистите и бизнеса.

2.4.2 SQL (Structured Query Language) е език за управление на релационни бази от данни. Той се използва за създаване, манипулиране и управление на данни в релационни бази от данни. SQL е стандартизиран език и се използва широко в индустрията и в научните среди. SQL позволява на потребителите да извършват множество действия с данните в базата, като например: Създаване на нови таблиците и бази от данни

Добавяне, промяна и изтриване на данни в таблиците

Търсене на информация в базата от данни чрез използване на заявки (query)

Създаване на индекси за бързо търсене на информация

Дефиниране на права за достъп до информацията в базата от данни

SQL има много различни команди, като SELECT, INSERT, UPDATE, DELETE, CREATE, DROP и други. Тези команди могат да бъдат използвани за извършване на множество операции върху данните в базата. SQL се използва от много приложения, които работят с бази от данни, като например уеб приложения, софтуер за управление на складове, софтуер за управление на персонала и други. Релационните бази от данни, като например MySQL, Oracle, SQL Server и PostgreSQL, са базирани на SQL и са използвани широко в индустрията.Ето някои от основните черти на SQL: Декларативен език: SQL е декларативен език, който описва желаните резултати от заявките вместо да описва как те трябва да бъдат постигнати. Това означава, че потребителят не трябва да се грижи за това как ще бъдат обработени данните, а само да каже какво иска да се направи. Релационна структура: SQL базите от данни се базират на релационна структура, която позволява на потребителите да организират данните в таблиците, свързани помежду си. Мощен език: SQL е мощен език, който позволява на потребителите да извършват сложни операции върху данните в базата от данни.

Стандартизиран език: SQL е стандартизиран език, който означава, че съществува обща синтаксис и структура за всички релационни бази от данни, които използват SQL. Лесна манипулация на данни: SQL позволява на потребителите да манипулират данните в базата от данни по лесен начин, без да се налага да използват сложни програмиране езици. Гъвкавост: SQL е гъвкав и може да бъде използван в много различни контексти, като например в големи корпоративни приложения или в малки проекти. SQL е мощен инструмент за управление на данни и е широко използван в индустрията и научните среди. Знанието на SQL може да бъде много полезно за програмисти, бази данни администратори и всеки, който работи с данни в някакъв вид.

2.4.3 C# е един от най-популярните обектно-ориентирани програмни езици, разработен от Microsoft. Използва се широко за създаване на софтуерни приложения за Windows, уеб приложения и игри. C# е силен език за програмиране, който предлага много функции, включително: Обектно-ориентирана програмна парадигма, която позволява да се работи с обекти и да се изграждат гъвкави и мащабируеми приложения. Мултиплатформен подход, който позволява да се разработват приложения за множество платформи, включително Windows, Mac и Linux. Богата стандартна библиотека, която предлага много полезни функции и класове за работа с данни, мрежи, файлове и много други. Лесен за учене и използван език, който предлага много документация и поддръжка от общността. C# е особено полезен за разработка на уеб приложения и игри, като много популярни игрови двигатели като Unity използват C# като основен език за програмиране. Ето някои от другите характеристики и предимства на C#: Силна типизация - C# е език със силна типизация, което означава, че типовете на данните се определят ясно и прецизно, като това гарантира повече сигурност и устойчивост на програмите. Голяма общност - C# има голяма и активна общност от програмисти, които споделят знания, ресурси и съвети, което прави процеса на разработка по-лесен и бърз. Интеграция с .NET Framework - C# е език, който е интегриран с .NET Framework, който предлага богати библиотеки и инструменти за разработка на софтуер. Голям избор на интегрирани среди за разработка - C# има голям избор на интегрирани среди за разработка (IDE), като Visual Studio, Visual Studio Code, JetBrains Rider и др. Висока производителност - C# е език с висока производителност, който може да обработва големи обеми от данни и да изпълнява сложни операции с голяма скорост. Поддръжка на паралелни процеси - C# има поддръжка на паралелни процеси, което означава, че програмите могат да използват множество процесори и да изпълняват няколко задачи едновременно, като по този начин се повишава ефективността и производителността на програмата.

Това са само някои от основните характеристики и предимства на

2.4.4 HTML е абревиатура от "Hypertext Markup Language" и е език за маркиране на уеб страници. Той се използва за създаване на структура, съдържание и форматиране на уеб страници, като използва различни елементи и тагове за да определи различните компоненти на страницата. HTML е един от основните езици, използвани за уеб разработка, заедно с CSS и JavaScript. Той позволява на разработчиците да създават уеб страници с текст, изображения, видео и аудио съдържание, хипервръзки към други страници, формуляри за вход и много други елементи. HTML е стандартизиран от W3C (World Wide Web Consortium) и се актуализира периодично с нови версии и функционалности. Най-новата версия на HTML е HTML5, която предлага нови функции за аудио и видео вграждане, възможност за локално съхранение на данни и други подобрения.

HTML е синтактичен език, който използва маркиращи елементи, наречени тагове, за да определи структурата и съдържанието на уеб страниците. Таговете са показатели на браузъра, как да покаже съдържанието на страницата.

Ето пример за елемент от HTML:

Картина, която съдържа текст

Описанието е генерирано автоматично

В този пример <h1> е тагът, който определя, че това е заглавието на страницата, а </h1> казва на браузъра да приключи тага и да покаже текста като заглавие. Това е само един пример за HTML елемент, но има много други елементи, като <p> за параграф, <img> за изображение, <a> за хипервръзка, и т.н. HTML може да бъде редактиран в текстов редактор като Notepad, Sublime Text, Visual Studio Code или в специализиран редактор за уеб разработка като Dreamweaver или Adobe Muse. След като е написан, HTML кодът се зарежда в уеб браузъра и се показва на потребителите.

2.4.5 CSS (Cascading Style Sheets) е език за стилизиране на уеб страници. Той позволява на разработчиците да задават външния вид на HTML елементите, като определят свойства като цветове, шрифтове, размери, маргини, падинги и други. CSS позволява на разработчиците да разделят структурата и съдържанието на уеб страниците от визуалното им представяне. Това позволява да се направят промени във външния вид на уеб сайта, без да се налага да се променя HTML кодът. CSS е свързан с HTML чрез свойства на елементите, наречени класове и ID-та. Разработчиците могат да дефинират стилове за определени класове и ID-та, което позволява да се стилизират само определени елементи на страницата. CSS може да бъде редактиран в текстов редактор като Notepad, Sublime Text, Visual Studio Code или в специализиран редактор за уеб разработка като Dreamweaver или Adobe Muse. След като е написан, CSS кодът се зарежда в уеб браузъра и се прилага към HTML елементите на страницата. CSS има много функции и възможности за стилизиране на уеб страници. Някои от тези функции включват: Избор на елементи: CSS позволява на разработчиците да избират определени HTML елементи, за които да приложат стилове. Това се прави чрез класове, ID-та или HTML тагове. Определяне на свойства: CSS позволява на разработчиците да определят различни свойства на стиловете като цветове, шрифтове, размери, падинги, маргини, рамки и други. Каскадност: CSS използва принципа на каскадата, който позволява на разработчиците да определят стилове на различни нива на документа, като приоритизира тези, които са по-специфични. Това позволява на разработчиците да управляват по-ефективно външния вид на уеб страниците. Наследяване: CSS позволява на стиловете да се наследят от родителски елементи на документа, което означава, че разработчиците не са задължени да прилагат стилове за всеки елемент на страницата. Анимация: CSS позволява на разработчиците да създават анимации и транзиции на уеб страниците, като определят свойства като прозрачност, позиция и размер във времето. CSS се използва заедно с HTML и JavaScript за създаване на интерактивни, стилни и атрактивни уеб страници.

2.4.6 JavaScript е програмен език, който се използва за разработване на динамични уеб страници и уеб приложения. Той може да се използва и за разработка на софтуер за мобилни приложения и настолни приложения. JavaScript е език, който се изпълнява върху клиентската страна на уеб браузърите. Това означава, че кодът на JavaScript се изпълнява на компютъра на потребителя, когато той посещава дадена уеб страница. Това позволява на JavaScript да се използва за създаване на динамични ефекти и за интерактивност на уеб страници. JavaScript има много функции и възможности за разработване на уеб приложения. Някои от тези функции включват: Манипулация на HTML: JavaScript позволява на разработчиците да манипулират HTML елементите на страницата, като променят текста, стиловете или добавят и премахват елементи. Валидация на формуляри: JavaScript може да се използва за валидация на входните данни в формуляри на уеб страниците.

Анимация: JavaScript може да се използва за създаване на анимации и транзиции на уеб страниците. Извличане на данни: JavaScript може да се използва за извличане на данни от уеб страници или за комуникация със сървърната страна на приложението.Интерактивност: JavaScript позволява на разработчиците да добавят интерактивност към уеб страниците, като създават елементи като бутони, падащи менюта, скролбарове и други. JavaScript е създаден от Brendan Eich през 1995 г. и е стандартизиран от ECMA International като ECMAScript. Днес се използва широко за създаване на уеб приложения и е един от най-популярните програмни езици в света. JavaScript е език със синтаксис, подобен на C++, което го прави лесен за научаване за програмисти, запознати с този език. Езикът предлага множество вградени функции и методи за работа с различни типове данни, като масиви, обекти, числа, символи и др. JavaScript може да се използва и за създаване на сървърни приложения чрез използване на Node.js - среда за изпълнение на JavaScript в сървърна среда. Това позволява на разработчиците да създават пълноценни уеб приложения, които могат да обработват данни и да извършват логически операции на сървъра. JavaScript също така поддържа библиотеки и фреймуърки, като jQuery, React, Angular, Vue.js и други, които улесняват работата на разработчиците и позволяват бързо и лесно създаване на динамични и интерактивни уеб страници. JavaScript има и свои предизвикателства, като например поддържането на съвместимост с различни браузъри, което може да бъде сложно поради различните интерпретации на езика от различните браузъри. Въпреки това, JavaScript е мощен и гъвкав език, който еднакво добре се справя със сложни и лесни задачи и е необходим при разработването на почти всички типове уеб приложения.

2.4.7 ASP.NET Core е платформа за уеб разработка на Microsoft, която позволява на разработчиците да създават уеб приложения и услуги с висока производителност и мащабируемост. ASP.NET Core е нова версия на платформата ASP.NET, която е написана отново от основи и е пълноценно кросплатформено решение. Това означава, че ASP.NET Core може да бъде използван на Windows, Linux и macOS, както и в Docker контейнери. С ASP.NET Core може да се създават уеб приложения, уеб услуги и RESTful API-та, използвайки различни шаблони за програмиране и изграждане на уеб приложения. ASP.NET Core поддържа различни модели на програмиране, като MVC (Model-View-Controller), Web API и Razor Pages. Основните характеристики на ASP.NET Core включват висока производителност, мащабируемост, бързо изграждане на приложения, добра поддръжка на сигурността и гъвкавост в избора на инструменти за разработка. ASP.NET Core е с отворен код, което означава, че всеки може да го използва, да го подобрява и да го приспособява към своите нужди. Това също така означава, че в даден момент от време съществува голямо количество инструменти, библиотеки и документация, които да помогнат на разработчиците. ASP.NET Core предоставя вградена поддръжка на възможности като внедряване на зависимости, управление на конфигурацията, кеширане, контрол на достъпа и много други. Също така, ASP.NET Core поддържа множество бази данни, включително SQL Server, MySQL, PostgreSQL и MongoDB. С помощта на ASP.NET Core, разработчиците могат да изграждат уеб приложения, които да работят с различни клиентски устройства, като компютри, мобилни телефони и таблети. ASP.NET Core има вградена поддръжка на RESTful API, която позволява на приложенията да комуникират с други приложения и услуги чрез HTTP заявки. В заключение, ASP.NET Core е мощна платформа за уеб разработка, която предоставя много възможности за разработка на висококачествени и мащабируеми уеб приложения и услуги. Благодарение на кросплатформената си поддръжка и отворения си код, ASP.NET Core е предпочитан избор за много разработчици.

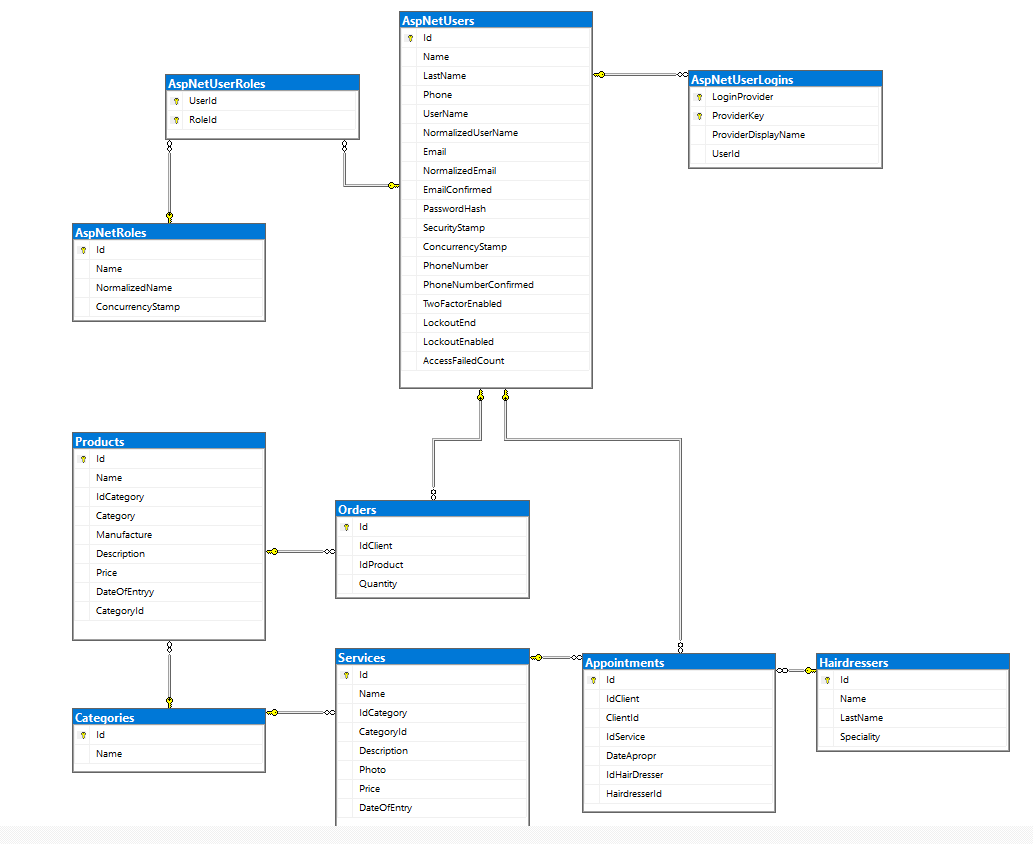
2.4.8 Bootstrap е свободен и отворен източник за фронтенд уеб фреймуърк, който се използва за ускоряване на разработката на уеб сайтове и приложения. Той предоставя готови компоненти за дизайн, като например бутони, форми, таблиц, менюта и други, които могат да бъдат лесно интегрирани в уеб страниците. Bootstrap също така предлага готови CSS стилове, които могат да бъдат прилагани към елементите на уеб страниците, като така се постига по-бързо и лесно стилизиране на уеб сайта. Bootstrap е разработен от Twitter и е използван от милиони разработчици по света.

**ГЛАВА 3.**

**ПРОГРАМНА РЕАЛИЗАЦИЯ НА WEB БАЗИРАН ……………….**

* 1. **Структурата на базата от данни – Data Models**

*Освен E/R диаграмата, тук се описват: данните в таблицата, ключовите полета и релациите /по кои полета и за какво се прави дадената релация/*



* 1. **Структура на MVC приложението – *Controllers*, *Views*, *ViewModels*, *Routs –*** *описание на стартирането и комуникация между Models – Views – Controllers.*

**>>>** *Тук се копира код от конкретния /Вашия/ проект, като се правят обяснения по кода и LINQ изразите. Кодът се копира ( а не скрийншот!!!) и се поставя в каре… като примера по-долу.*

**>>> За какво да пишем:**

* **Как работи схемата Routes Engine**
* **Как работи шаблона на Controller с GET и POST заявки към Actions**
* **Описание на CRUD операциите- особености на отделните ACTIONS, параметри и валидиране на моделите, получени от View, както и зареждане на модела, който се подава към**

*Примерно: обяснете всеки фрагмент от следния код…*

* **Работа с View моделите (където е необходимо). Разлика с използването на ViewBag и/или ViewData, които се използват във Вашия проект (може да се опише разликата между двата обекта)**
* **Описание на Razor Pages и Razor Views (TagHelpers и HtmlHelpers)**
* **Описание и конфигурация на Services (ако е приложимо)**
  1. **Авторизация и оторизация на потребителите на приложението –** *описание на функционалността на отделните потребители според ролите в приложението и как е реализирано това в проекта /според описания дизайн в т. 2.3./ - кой потребител до какви функции има достъп, от какво меню се осъществява този достъп и как това е реализирано в кода /на Controller, от View и т.н. /*

**Авторизация и оторизация на потребителите на приложението**

*Общение на реализирания проект, предимства и недостатъци, теденции за бъдещи разработка и развитие.*

*До каква степен е постигнато планираното за разработване приложение.*

**ИЗПОЛЗВАНА ЛИТЕРАТУРА**

Примери: /трябва да се следва точно следния формат/

1. Наков С., Колев В. и колектив, Въведение в със C#, София, 2015, ISBN 978-954-400-527-6

2. Колисниченко Д., HTML 5 & CSS 3 - практическо програмиране за начинаещи. Трето издание, Асеневци, София, 2017, ISBN 9786197356113

3. Pro ASP.NET Core 3: Develop Cloud-Ready Web Applications Using MVC, Blazor, and Razor Pages 8th ed. Edition, Apress, New York, 2020, ISBN 1484254392

4. Learn ASP.NET Core 3: Develop modern web applications with ASP.NET Core 3, Visual Studio 2019, and Azure, 2nd Edition Paperback, Packt, Birmingham, UK, 2019, ISBN 1789610133

Електронни ресурси

5. https://www.w3schools.com/,

6. https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-3.1; http://www.introprogramming.info/intro-csharp-book/

7. https://to6esko.github.io/ - Български превод на Eloquent JavaScript