

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 САЙТ, ОРГАНИЗИРАЩ ДЕЙНОСТТА НА фитнес център „SPORT HOUSE“**

*Ученик:* ***Стилян Николаев Иванов***

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

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

***Консултант:*** *Тоня Белезирева*

Сопот, 2024 г.

**Съдържание**

**Увод**

**Kратко опсание на тема и областта на дипломнната работа**

В последно време много хора работят от дома, а какъв по-добър начин да останат във форма от посещение в близката фитнес зала? Това е една от причините за нарастващата популярност на фитнес сайтовете. Много хора проверяват работното време на залите или търсят личен треньор, в случай че фитнесите са затворени, включително и за онлайн тренировки. Ако сте собственик на фитнес зала или искате да предлагате фитнес тренировки онлайн, Ви трябва уебсайт. Важно е хората да могат да намерят залата лесно или да се свържат, ако имат въпроси, чрез представената навигация в сайта.

**Цели и задачи**

За утвърждаване на имиджа и придобива все по-широка клиентела и популярност Фитнес център „Sport House“, при разработката на Уеб сайт, организиращ дейността на фитнес център си поставям следните цели:

* Да се проектира и реализира трислойно MVC WEB приложение;
* Уеб сайта трябва да поддържа CRUD операции за артикулите, наличните им цени и количества за тяхната продажба;
* Сайтът трябва да разполага със следните публично достъпни секции: „За нас“, „Фитнес инструктори“, „Програми“, „Работно време“, „Галерия“, „Контакти“;
* Сайтът трябва да предлага възможност за разглеждане на заниманията, графика, както и работещите в центъра инструктори.
* Трябва да се осигури наличието на форма за заявка за включване в избрана програма, което може да се извърши от регистриран в системата клиент.
* Сайтът трябва да може да се администрира с помощта на администраторско меню, което да позволи добавянето и редактирането на програми, инструктори, както промяна в съществуващия график.
* Дизайнът на сайта трябва да е адаптивен и да изглежда еднакво добре на различни устройства. Изберете аргументирано подходящи технологии.

За постигане на поставените цели си поставям следните задачи

* Да се направят роли за работа с приложението:
* Добавяне/редактиране на програма: /с Администраторски функции
* Добавяне/редактиране/изтриване на фитнес инструктор: /с Администраторски функции/
* Редактиране на график на заниманията: /с Администраторски функции/
* Извличане на информация за конкретна програма /визуализация на един екран/
* Визуализиране на графика в табличен вид
* Регистрация на клиент: /от меню „Контакти“/
* Предварителна резервация от клиент

**ГЛАВА 1.**

**ПРОУЧВАНЕ ПО TEMATA**

* 1. **Предпоставка за създаване на продукта**
  2. **Съществуващи решения и реализации**

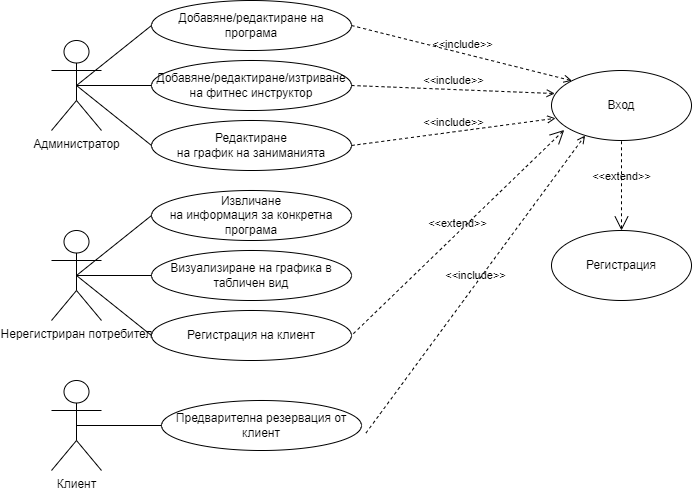
**ГЛАВА 2.**

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

**2.1. Функционални изисквания към WEB базиран** уеб приложение за организиране на дейността на **фитнес център „Sport House“**. Приложението трябва да поддържа следните функционалности:

* Да се направят роли за работа с приложението:
  + АДМИНИСТРАТОР
  + Клиент
  + Нерегистриран потребител
* Добавяне/редактиране на програма: /с Администраторски функции/:
  + Номер на програмата
  + Име на програма /уреди, салса, зумба, степ-фит, каланетика, йога/
  + Категория /зала с уреди, групово/
  + Описание – кратък текст – описание на дейността
  + Снимка/снимки
  + Цена на единично посещение
  + Дата на вписване в системата
* Добавяне/редактиране/изтриване на фитнес инструктор: /с Администраторски функции/:
  + Имена
  + Описание – кратък текст – предназначение и приложение; линкове
  + Снимки
  + Телефон за връзка
  + Дата на вписване в системата
* Редактиране на график на заниманията: /с Администраторски функции/
  + Номер занимание по график
  + Номер на програма
  + Ден от седмицата
  + Час начало
  + Час край
  + ИД на инструктор – име и фамилия на инструктора, който извършва дейността
  + Дата на вписване в системата
* Извличане на информация за конкретна програма /визуализация на един екран/
* Визуализиране на графика в табличен вид
* Регистрация на клиент: /от меню „Контакти“/
  + Име
  + Фамилия
  + Електронна поща
  + Телефон за контакт
  + Дата на регистрация
* Предварителна резервация от клиент:
  + Номер на клиент
  + Номер занимание по график;
  + Дата на регистрация;

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

****

**2.3. Прототип на потребителският интерфейс**

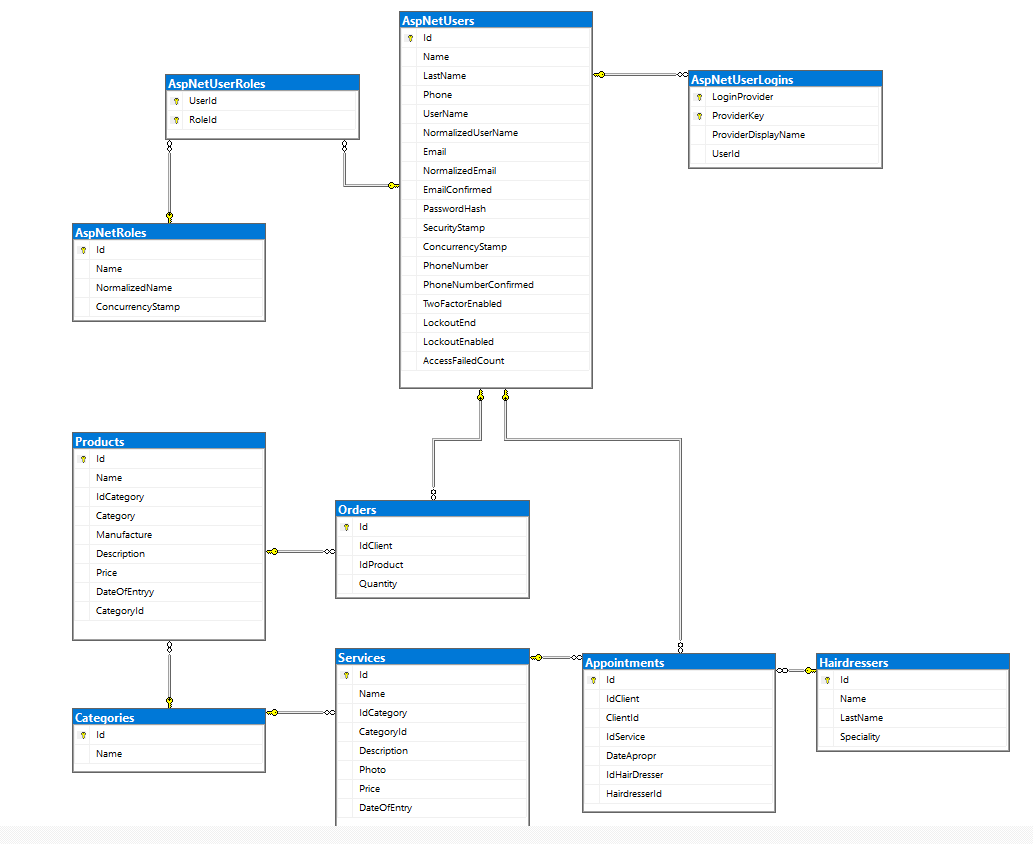
**2.4. Описание на избраната технология и софтуерните средства за разработка на приложението и основните алгоритми/**технологии: *ASP.Net, ASP.Net Core, SQL Server, средства: C#, HTML, CSS, JavaScript*, *Bootstrap***/**

**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…*

// GET: Orders/Details/5

public async Task<IActionResult> Details(int? id)

{

if (id == null)

{

return NotFound();

}

var order = await \_context.Order

.Include(o => o.User)

.FirstOrDefaultAsync(m => m.Id == id);

if (order == null)

{

return NotFound();

}

return View(order);

}

* **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. /*

**ЗАКЛЮЧЕНИЕ**

*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