# AKADEMIJA TEHNIČKO-UMETNIČKIH STRUKOVNIH STUDIJA BEOGRAD ODSEK VISOKA ŠKOLA ZA INFORMACIJONE I KOMUNIKACIONE TEHNOLOGIJE

## INTERNET TEHNOLOGIJE

Dokumentacija

CameraShop

Napomena: Projekat neće da se builduje u Visual Studiu 2019. Samo 2022.

link: <a href="https://github.com/vanja-veapi/CameraShop">https://github.com/vanja-veapi/CameraShop</a>

## SADRŽAJ

#### 1.Uvod

- 1.1. Tema projekta
- 1.2. Pokretanje projekta
- 1.3. Kratak opis aplikacije

### 2. Mapa projekta

- 2.1. Organizaciona šema
- 2.2. Kontroleri i opis
  - 2.2.1. PopulateData
  - 2.2.2. Auth
  - 2.2.3. Brands
  - 2.2.4. Camera
  - 2.2.5. Cart
  - 2.2.6. SensorType
  - 2.2.7. Stocks
  - 2.2.8. User
- 2.3. Slika strukture baze podataka
- 2.4. Slika strukture fajlova
  - 2.4.1. Solution
  - 2.4.2. Domain
  - 2.4.3. Data access
  - 2.4.4. Application
  - 2.4.5. Implementation
  - 2.4.6. Api

## 1. Uvod

#### 1.1. Tema projekta

Tema projekta je web shop koji specijalizuje u prodavanju kamera kao proizvod.

#### 1.2. Pokretanje projekta

Za pokretanje projekta potrebno je settovati connection string na dva mesta :

AppSettings->appSetings.Local.json -> field "DBConnectionString"

I u DB Contextu

Kada se pokrene projekat setovano je da se pokrene swagger editor u Browser-u

#### 1.3. Kratak opis aplikacije

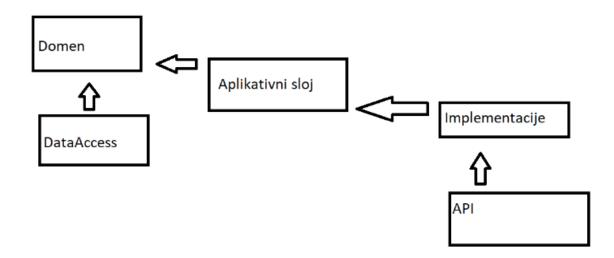
- Aplikacija je podeljena na vise slojeva Domenski sloj, Aplikativi sloj, Implementacioni sloj, DataAccess sloj, I Api sloj
- Baza podataka aplikacije je napravljena Code-first pristupom
- U domenskom sloju nalaze se sve klase koje su potrebne aplikaciji I one predstavljaju jednu tabelu u bazi podataka
- U aplikativnom sloju se nalaze svi interfejsi, koji se moraju implementirati za ispravan rad aplikacije
- Implementacioni sloj je sloj koji implementira interfejse iz aplikativnog sloja
- DataAccess sloj je mesto gde se definise baza podataka, I pravila za bazu podataka
- Api sloj mesto gde se definisu dodirne tacke sa spoljnim svetom, I gde zapravo cela
  aplikacija pocinje da radi, te iz tog razloga ostatak dokumentacije je pisan po redosledu
  kontrolera
- Upotreba odgovarajucih statusnih kodova
- Koriscenje globalnog exception handlera za hendlovanje bilo koje greske
- Kreiranje naloga korisnika, I aktivacija naloga
- Slanje imejla ova funkcionalnost nazalost zbog google-ovih polisa trenutno ne radi
- Upload fajla, prebcivanje fajla na server, I upisivanje naziva fajla
- Svaki endpoint koji menja stanje u bazi je validiran upotrebom FluentValidation pravilima
- Granulacija prava slucajeva koriscenja na nivou korisnika
- Dodeljivanje privilegija drugim korisnicima
- Pretraga sa paginacijom
- Upis serilog gresaka u tekstualni fajl, sa requestom, tekstom greske, I ko je izvrsilac
- Upis uspesnih slucajeva koriscenja u bazu podataka, sa informacijama o korisniku I request-a

• Upotreba swaggera

- Upotreba JWT-a kao nacin autentifikacije na aplikaciju
- Pridrzavanje SOLID principa

## 2. Mapa projekta

#### 2.1. Organizaciona šema



#### 2.2. Kontroleri i opis

#### 2.2.1. PopulateData

Kontroler koji ima jedan endpoint, samo za upis inicijalnih podataka, u aplikaciju.

#### 2.2.2. Auth

Auth kontroler ima jedan endpint takodje, taj endpoint je za autentifikaciju, i kao rezulat ima jwt token.

#### 2.2.3. Brands

Ovaj kontroler je klasican kontroler, CRUD operacija za brendove foto aparata.

#### 2.2.4. Camera

Camera kontroler je mesto gde se kreiraju nove kamere, pregled kamera se pregleda paginacijom, Takodje na ovom endpointu se mogu i kreirati discounti za proizvod, kao i da se brisu discounti.

### 2.2.5. Cart

Cart kontroler je jedan od zanimljivijih kontrolera, tu se moze vrsiti pregled korpe on je paginacijom resen, moze se brisati iz korpe, i za kraj moze se kupiti proizvod, a kada prozivod stigne na adresu postoji api koji bi zavrsio porudzbinu.

#### 2.2.6. SensorType

Jos jedan klasican kontroler CRUD operacija za tipove senzora.

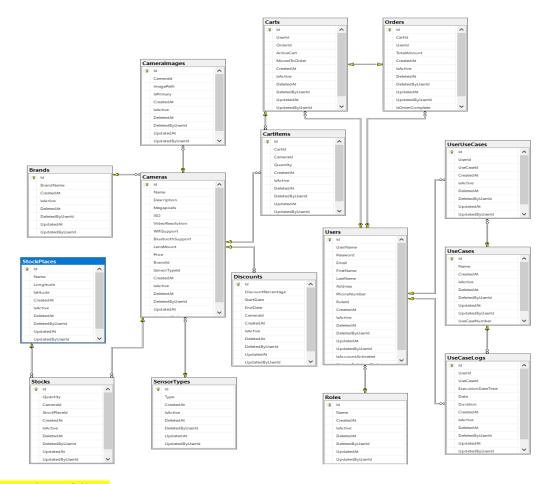
#### 2.2.7. Stocks

Stock kontroler je jos jedan zanimljiv kontroler, koji se bavi kolicinom kamera na lageru, i mestima gde se fizicki nalaze ti lageri.

#### 2.2.8. User

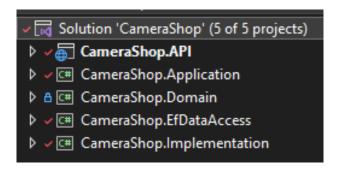
Poslednji ali takodje bitan kontroler, na ovom kontroleru anonimni korisnik moze napraviti novi profil, i aktivirati profil, a ukolio korisnik ima privilegije moze i izvrisiti druge endpointe, endpoint za dodeljivanje ili oduzimanje privilegija drugih korisnika. Takodje postoji i selekcija usera i use case-ova kako bi se na frontu realizovala puna funkcionalnost.

## 2.3. Slika strukture baze podataka



### 2.4. Slika strukture fajlova

## 2.4.1. Solution



#### 2.4.2. Domain

```
▲ 🖺 C# CameraShop.Domain
▶ ₽☐ Dependencies
▶ A C# Brand.cs
> C# Cameralmage.cs
D A C# Role.cs
♦ + C# StockPlace.cs
```

#### 2.4.3. Data access

#### 2.4.4. Application

```
■ ✓ □ CameraShop.Application

  ▶ ₽₽ Dependencies

▲ A ■ Core

    ▶ A  Logging
    ▶ + C# |ApplicationUser.cs
    ▶ + C# IEmailSender.cs
    ▶ + C# IPictureMenager.cs
    ▶ + C# IStringEncryptor.cs

▲ A ■ Dto

    ▶ A  BrandDto
    ▶ A  Logging
    ▶ a  Pagination
    ▶ A  Searches
    ▶ a  SensorTypeDto

▷ △ ■ StockDto

    ▶ + C# DtoBase.cs
    ♦ + C# DtoListReturn.cs
    ▶ + C# EmptyDto.cs
    ♦ + C# IdOnlyDto.cs

▲ A   Exceptions

    ▶ + C# EmailNotSendException.cs
    ▶ A C# EntityNotFoundException.cs
    ▶ ≜ C# ForbiddenUseCaseExecutionException
    ▶ & C# UseCaseConflictException.cs

▲ A ■ UseCases

    ▶ A ■ BrandUseCases
    ▶ △ ■ CameraUseCases
    ▶ △ ■ SensorTypeUseCases

▶ △ ■ StocksUseCases

    ▶ a  UseCaseHandlers
```

#### 2.4.5. Implementation

```
■ ✓ 

CameraShop.Implementation

  ▶ ₽☐ Dependencies

▲ A  Core

    ▶ + C# PictureMenager.cs
    ▶ + C# SMTPEmailSender.cs
    > + C# UseCaseHandler.cs

▲ A ■ UseCases

▲ A  Commands

      ▶ A  Brands
      ▶ A  SensorTypes

▲ A ■ Users

         > + C# ActivateUserAccount.cs
         + C# AddUseCasesToApplication.c
         ▶ + C# CreateNewUser.cs
         ▶ + C# MenageUserPriviledges.cs
         > + C# UpdateUserInfo.cs

▲ A ■ Queries

      ▶ A  Brands
      D △ 🛅 Carts
      ▶ A  SensorTypes

▲ A ■ User

         > + C# SelectUseCases.cs
         ▶ + C# UserSelection.cs
    > + C# EFBaseUseCase.cs

▲ A ■ Validators

    D A ■ Brand
    D △ Camera
    D △ 🛅 Cart
    ▶ a  SensorType
    D A ■ Stock
    D A ■ User
    ▶ + C# ImageValidator.cs
```

#### 2.4.6. Api

