



# ELEKTRONSKI FAKULTET NIŠ

Predmet – Multimedijalni sistemi

Asistent:

*Dr Petar Rajković*

Student:

*Snežana Spasić*      17408

Niš, 2021.

## Sadržaj

Uvod .....	3
Pokretanje aplikacije .....	4
Funkcionalnosti aplikacije .....	5
Blok 1.....	5
Opis .....	5
Opcije menija .....	6
Blok 2.....	7
Blok 3.....	9
Zaključak .....	10

# Uvod

Tema ovog projekta je kreiranje Windows forms aplikacije za obradu slike po MVC obrascu, koji podržava par različitih View-a. Omogućava korišćenje raznih filtera, učitavanje i čuvanje slike, undo/redo, prikaz grafika, unos vrednosti.

Windows Forms je deo .NET framework-a i pruža alate i kontrole koje olakšavaju pravljenje desktop aplikacija sa vizuelnim interfejsima, kao što su liste, meniji, tekstualna polja i ostalo.

Kod je razvijen pomoću Visual Studio okruženja, koje ima podršku za razvoj Windows Forms aplikacija.

## **Pokretanje aplikacije**

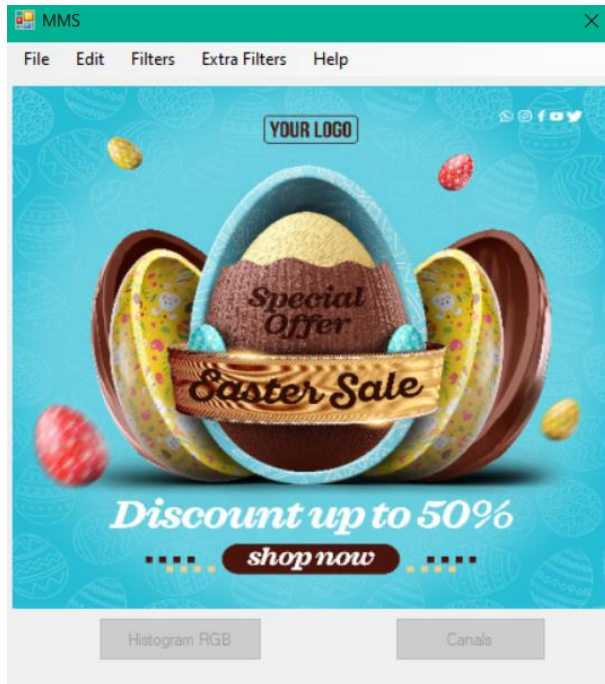
1. Instalirati Visual Studio
2. Otvoriti projekat u Visual Studio okruženju i pokrenuti ga klikom na dugme Start

# Funkcionalnosti aplikacije

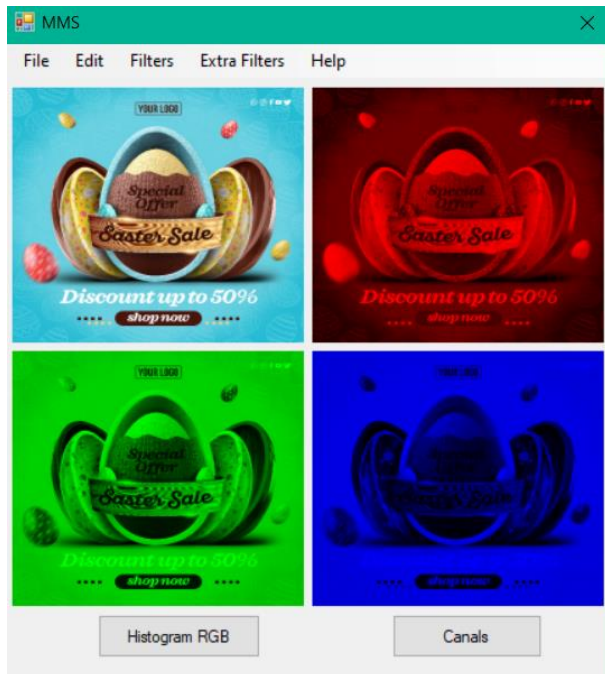
## Blok 1

### Opis View-a

View 1: Treba da prikazuje samo učitano sliku



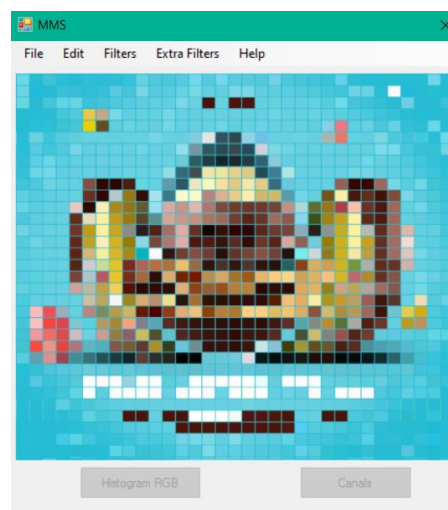
View 2: Treba da prikaže 4 slike, originalnu i slike po red, green i blue kanalima.



## Opcije menija

- File
  - **Load** (podržava učitavanje iz bmp, jpg i png formata)
  - **Save** (podržava snimanje u pomenute formate)
  - **Exit** (izlaz iz programa)
- Filters
  - Prikaz kanalskih slika **RGB Channels** (model boja koji se koristi je RGB)
  - Primena jednog od osnovnih filtera (ovde je implementiran **Gamma filter**). Ako se filter primeni dok je prikazan View2, onda se i kanalske slike ažuriraju.
  - Primena jednog od konvulacionih filtera (ovde je implementiran **Sharpen filter**)
  - Dodatna opcija – **256 Colors**, kreira se BMP slika sa 256 indeksiranih boja. Tih 256 indeksiranih boja treba da budu najčešće boje iz početne slike
- Extra filters
  - Edge detect filter – ovde je implementiran **EdgeEnhance filter**
  - Displacement filter – ovde je implementiran **Pixelate (Grid i NoGrid)**. Posebni parametar PixelSize, default 15.
- Help
  - Informacije o projektu i pomoć oko korišćenja histograma.
- Undo-Redo:
  - **Undo**
  - **Redo**
  - Podešavanje - Ako se bafer napuni, onda se iz njega izbacuju prvi dodati elementi kako bi se napravilo mesta za nove

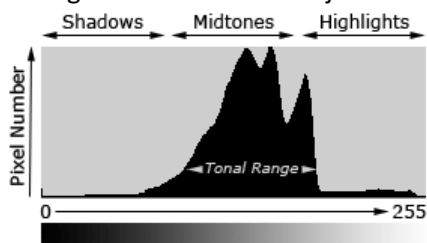
Undo-Redo opcija je implementirana pomoću bafera strukture podataka koja pamti slike. Pre svake operacije se čuva trenutna kopija slike u Undo bafer. Nakon svakog poziva Undo funkcije slika se prebacuje u Redo bafer. Nakon svakog poziva Redo funkcije slika se prebacuje u Undo bafer. Ako se izvrši bilo koja nova operacija Redo bafer se prazni.



*Pixelate Grid filter*

## Blok 2

- Kreirati Channel histogram, za svaki od kanala izdvojenih u prethodnoj vežbi generisati histograme kao na sledećoj slici:

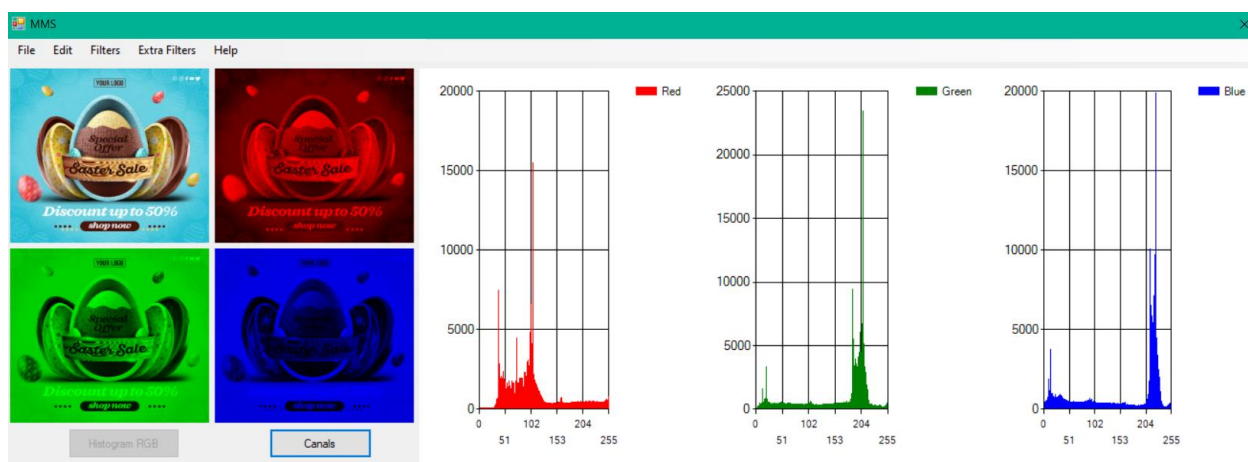


Histogram po x osi treba ima vrednosti koje elementi kanala mogu da uzmu, a po y njihovu učestanost.

Histogram se koristi tako što prvo prikažu slike po svim kanalima **Filters->RGB Channels** a zatim **klikne na dugme Histogram**.

Kada se slika prikaže po kanalima, umesto kanalnih slika se prikazuju histogrami.

Takođe, postoji opcija da se prikaz vrati po kanalskim slikama (promena prikaza).



- Implementiran je **Min and Max Channels filter** nad slikom koji rezultuje promenom histograma na način tako što definiše minimalnu i maksimalnu dozvoljenu vrednost po kanalu (crvene linije na slici). Sve vrednosti manje od minimuma zamenjene su minimalnom dozvoljenom vrednošću, a sve vrednosti veće od maksimuma zamenjene su maksimalnom dozvoljenom vrednošću.
- Implementirani su **tri grayscale algoritma** čiji se rezultati mogu uporediti kroz View2. Prvi od njih je osnovna aritmetička sredina za R, G i B ( $\text{Gray} = (\text{Red} + \text{Green} + \text{Blue}) / 3$ ). Drugi računa  $\text{Gray} = \text{Max}(\text{Red}, \text{Green}, \text{Blue})$ , dok je treći algoritam definisan sa:  
 Podesivi  $\geq 0$  koeficijenti za boje.  
 $\text{Gray} = (\text{Red} * C_r + \text{Green} * C_g + \text{Blue} * C_b)$   
 Default:  $C_r = 0.3, C_g = 0.59, C_b = 0.11$ . I važi:  $C_r + C_b + C_g = 1$ .
- Ordered dithering filter** (po opisu sa [https://www.visgraf.impa.br/Courses/ip00/proj/Dithering1/ordered\\_dithering.html](https://www.visgraf.impa.br/Courses/ip00/proj/Dithering1/ordered_dithering.html) )

- **Cross-domain colorize filter** – Korisnik unosi vrednost za newHue (od -1 do 5) i vrednost za newSaturation (od 0.0 do 1.0). Određuju se širina i visina slike i bajtovi koji predstavljaju piksele iWidth, iHeight, ImageData.

For x = 0 To iWidth - 1

QuickX = x \* 3

For y = 0 To iHeight - 1

R = ImageData(QuickX + 2, y)

G = ImageData(QuickX + 1, y)

B = ImageData(QuickX, y)

Konvertuju se R,G,B vrednosti u H,S,V.

Ako nije prosleđeno newSaturation konvertujte newHue,S,V u R, G, B, u suprotnom konvertujte newHue, newSaturation, V u R, G, B .

Vrednost piksela se postavlja na novoizračunate vrednosti.



### Blok 3

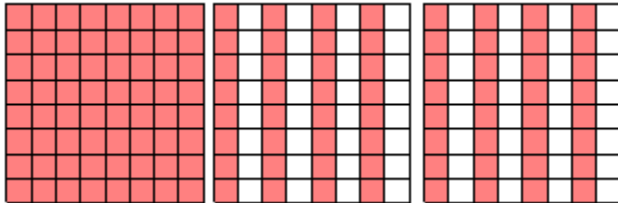
- **Downsampling** – primenjuje se downsampling na učitano sliku i tako se smanjuje količina potrebne memorije za smeštanje slika.
- **Downsampling channels**  
Od tri kanala se na dva primenjuje downsampling (na crveni i plavi kanal da, na zeleni ne), i tako se smanjuje količina potrebne memorije za smeštanje slika.

Iskorišćeni su modeli kanalskih slika implementiranih u Bloku 1 (RGB Channels).

Nakon što se učitava slika i pozove opcija za Downsampling channels, korisniku se otvori View2 i tamo se prikaže originalna slika, kao i tri slike u kojima je urađen downsampling na po 2 kanala a treći kanal ostavljen nepromenjen.

Korisnik sada treba da ima opciju da snimi jednu od tri verzije slike u fajl.

Upotrebljena je downsampling matrica definisana sledećom slikom:



## Zaključak

U ovom radu predstavljen je razvoj WindowsForms aplikacije za obradu fotografija.

Prilikom izrade ove aplikacije stekla sam značajno tehničko znanje i iskustvo u nekoliko ključnih oblasti: rad sa slikama, primena filtera, undo/redo funkcionalnost, upravljanje fajlovima. Uspela sam da se upoznam sa načinima manipulacije slikama koristeći Bitmap klasu. Implementacija različitih filtera (kao što su crno-beli, gamma, sharpen, itd.) pomogla mi je da dublje razumem osnovne principe obrade slike, kao što su manipulacija pikselima, transformacija boja i operacije nad matricama.

Kroz ovaj projekat sam usavršila svoje tehničke veštine u razvoju desktop aplikacija koristeći Windows Forms i .NET kao i proces razvoja softvera.