# GRAPHICS – GUI

JavaFX Photo Editor

## Přehled

Graphics – GUI je aplikace pro úpravu grafiky založený na JavaFX. Umožňuje uživatelům načítat obrázky, používat různé obrazové filtry a ukládat upravené obrázky.

## Předpoklady

Vhodné IDE

JavaSDK 21

JavaFX

## Instalace

**Instalace řešení**

git install git@github.com:CokoPiskot/Graphics-GUI.git

## Struktura projektu Back-end

**Třída HelloApplication**

* Metoda *start*
  + Načtení FXML souboru:
    - Pomocí třídy **FXMLLoader** je inicializován nový objekt, který slouží k načtení obsahu definovaného ve FXML souboru (**hello-view.fxml**). Tento soubor obsahuje strukturu uživatelského rozhraní v jazyce FXML.
  + Vytvoření Scény:
    - Je vytvořena nová scéna s obsahem načteným z FXML souboru (**fxmlLoader.load()**).
    - Scéna má definovanou velikost 1000x700 pixelů.
    - Přidává se externí styl z CSS souboru (**styles.css**) pomocí **scene.getStylesheets()**.
  + Nastavení Ikony Aplikace:
  + Nastavení Minimální Velikosti Okna:
  + Nastavení Titulku Okna:
  + Zobrazení Scény:
    - Nastavená scéna se přidává do hlavního okna (**stage**).
    - Zavolání **stage.show()** způsobí zobrazení hlavního okna s nastavenou scénou.

Obsah obrázku text, software, Multimediální software, snímek obrazovky

Popis byl vytvořen automaticky

Obrázek 1 - Metoda start

* Metoda *main*
  + Spuštění aplikace pomocí *launch()*

**Třída HelloController**

Tato třída je „mozkem“ celé aplikace *Graphics – GUI*, nachází se zde nejdůležitější metody pro chod celé aplikace *Graphics – GUI.*

Ve třídě se nachází anotace *@FXML*, ta se používá k tomu, aby označila prvky uživatelského rozhraní. Ve třídě *HelloController* s nimi můžeme dále pracovat.

* Deklarace proměnných
  + ***imageView***
    - FXML proměnná díky které můžeme dále pracovat s imageView, což je prostor, ve kterém se nachází případný načtený obrázek
  + ***originalImage***
    - FXML proměnná, která ukládá původní fotku, v případě toho že si uživatel bude chtít původní obrázek prohlédnout přímo v aplikaci
  + ***tempImage***
    - FXML proměnná která ukládá průběžný stav fotky, co se filtrů týče
  + ***menuBar***
    - FXML proměnná díky které můžeme dál pracovat s horním „proužkem“
  + ***mode***
    - FXML proměnná která ukládá Label „Dark Mode“ nebo „Light Mode“, podle toho, co je zrovna zapnuto
  + ***basePane***
    - FXML proměnná která ukládá „pane“ aplikace, dále v kódu se používá k implementaci tmavého režimu
  + ***radio1***
    - FXML proměnná která ukládá, jestli je zrovna vybraná, radio1 označuje „Original Image“
  + ***radio2***
    - FXML proměnná která ukládá, jestli je zrovna vybraná, radio2 označuje „Modified Image“
  + ***textArea***:
    - FXML proměnná pro práci s textArea která zobrazuje historii úprav
  + ***lightMode***
    - Proměnná, která ukládá, jestli je právě zapnutý světlý režim nebo tmavý režim, používá se na různé kontroly
  + ***aboutWindow***
    - Pomocná proměnná, která ukládá stav je-li právě spuštěno podokno „About“
  + ***identity***
    - FXML proměnná která označuje MenuItem v DropBaru „*Filters*“, používá se pro další práci s filtry
  + ***threshold***
    - FXML proměnná která označuje MenuItem v DropBaru „*Filters*“, používá se pro další práci s filtry
  + ***oldStyleFilter***
    - FXML proměnná která označuje MenuItem v DropBaru „*Filters*“, používá se pro další práci s filtry
  + ***bwFilter***
    - FXML proměnná která označuje MenuItem v DropBaru „*Filters*“, používá se pro další práci s filtry
  + ***vignette***
    - FXML proměnná která označuje MenuItem v DropBaru „*Filters*“, používá se pro další práci s filtry
  + ***colorizer***
    - FXML proměnná která označuje MenuItem v DropBaru „*Filters*“, používá se pro další práci s filtry
* Metody
  + ***toBufferedImage***
    - Metoda, která převádí Image na BufferedImage, protože s BufferedImage je jednodušší manipulace

***Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky***

Obrázek - Implementovaná metoda toBufferedImage

* + ***negativeButtonAction***
    - Metoda, která aplikuje filtr *negative* na obrázek
  + ***bwfilterButtonAction***
    - Metoda, která aplikuje filtr *Black & White* na obrázek

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

Obrázek - Implementovaná metoda bwfilterButtonAction

* + ***exitButtonAction***
    - Metoda, která zpracovává požadavek na vypnutí aplikace
  + ***selectImage***
    - Metoda, pomocí které se načítá obrázek do aplikace

Obsah obrázku text, snímek obrazovky, software

Popis byl vytvořen automaticky

Obrázek - Implementovaná metoda saveImage()

* + ***saveImage***
    - Metoda, pomocí které se ukládá modifikovaný obrázek
  + ***aboutButtonAction***
    - Metoda, která otevře nové podokno About
  + ***nightButtonAction***
    - Metoda, která zpracovává požadavek na aktivaci tmavého režimu
  + ***radioButtonController***
    - Metoda pro správu dvou radio tlačítek

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

Obrázek - Implementovaná metoda radioButtonController

* + ***restoreOriginalImage***
    - Metoda, která obnoví původní nemodifikovaný obrázek
  + ***clickSound***
    - Metoda, která přehrává zvuk kliknutí, používá se napříč mnoho metodami

Obsah obrázku text, snímek obrazovky, software, Multimediální software

Popis byl vytvořen automaticky

Obrázek - Implementovaná metoda clickSound

* + ***playMusic***
    - Metoda, která přehrává písničku v pozadí

**Třída Filters**

*David Franc doplní…*

**pom.xml**

V tomto souboru jsou nadefinované potřebné závislosti (externí knihovny, moduly a balíčky) pro běh programu.

Obsah obrázku text, snímek obrazovky

Popis byl vytvořen automaticky

Obrázek - Část souboru pom.xml

## Struktura projektu Front-end

**hello-view.fxml**

V tomto souboru je popsaná struktura hlavní okna tzn. Nachází se se definované elementy jako jsou například: *Label, Button, Pane, TextArea* atd., jejich pozice, velikosti a další různé atributy, je to podobné jako když definujeme *HTML*.

Obsah obrázku text, snímek obrazovky

Popis byl vytvořen automaticky

Obrázek - Část hello-view kde můžeme vidět definované menu "Filters"

**about-view.fxml**

V tomto souboru je popsaná struktura „*About*“ podokna.

**styles.css**

V tomto souboru jsou uložené data pro určení vzhledu Front-Endu aplikace.

Obsah obrázku text, snímek obrazovky, Písmo, software

Popis byl vytvořen automaticky

Obrázek - CSS tlačítka

## Použití aplikace

Obsah obrázku text, snímek obrazovky, software, Počítačová ikona

Popis byl vytvořen automaticky

Obrázek 10 - Rozvržení akčních elementů aplikace

1. DropBar, který se rozvine na tlačítka „**Load**“ a „**Save**“
   1. **Load** - umožňuje načtení obrázku v .jpg, .png, .bmp formátech
   2. **Save** – umožňuje uložení obrázku v .jpg, .png, .bmp formátech
2. DropBar, který se rozvine na tlačítka pro aktivaci filtrů
   1. **Negative**
   2. **Pixelizer**
   3. **Identity**
   4. **Threshold**
   5. **Old – Style filtr**
   6. **Black & White filtr**
   7. **Vignette**
   8. **Colorizer**
3. Tlačítko, které otevře podokno „About“
   1. Informace o aplikaci a autorech
4. Tlačítko, které aktivuje tmavý režim
   1. Po aktivaci tmavého režimu tlačítko změní text na „Light Mode“, po stisknutí navrátí světlý režim

Obsah obrázku text, snímek obrazovky, software, Multimediální software

Popis byl vytvořen automaticky

Obrázek 11 - Aktivovaný tmavý režim

1. Tlačítko, které vypne aplikace
2. TextArea která zobrazuje historii úprav
3. Tlačítko, které vygeneruje unikátní RGB obrázek
4. Tlačítko které, navrátí původní nemodifikovaný obrázek
5. Přepínač, který dokáže zobrazit jak upravenou, tak původní fotografii
   1. „**Original Image**“ zobrazí se původní fotka
   2. „**Modified Image**“ zobrazí se upravená fotka
6. Plátno, na které se umístí načtená fotografie

Obsah obrázku text, voda, snímek obrazovky, Multimediální software

Popis byl vytvořen automaticky

Obrázek 12 - Aplikace po nahrání obrázku

## Autoři

David Křesťan

Petr Čamlík

David Franc

Marek Murín

**Váš tým Microsoft**