

Bakalářská práce



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra počítačové grafiky a interakce

Fotogalerie s generátorem webstránek

Petr Cipra

Vedoucí: prof. Ing. Vlastimil Havran, Ph.D.

Obor: Otevřená informatika

Studijní program: Software

Leden 2022

Poděkování

Tímto bych rád poděkoval prof. Ing. Vlastimilu Havranovi, Ph.D. za užitečné a cenné rady a vstřícnost při vypracovávání této práce.

Prohlášení

Prohlašuji, že jsem práci vypracoval samostatně a použil jsem pouze podklady uvedené v příloženém seznamu. Nemám závažný důvod proti užití tohoto školního díla ve smyslu §60 Zákona c. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Praze, 4. ledna 2022

Abstrakt

Tato práce se zabývá návrhem a implementací aplikace v jazyce C++ pro správu obrázků, která umožňuje jejich export do webové galerie. Dále také analýzou podobných již existujících programů pro vytváření webových galerií. Výsledná aplikace umožňuje uživateli obrázky importovat i z adresářů a upravovat je přímo v aplikaci. Vygenerovaná webová galerie je přizpůsobitelná a kompatibilní s HTML 4.01 strict.

Klíčová slova: Webová galerie, Obrázky, Úprava obrázků, C++, HTML

Vedoucí: prof. Ing. Vlastimil Havran,
Ph.D.
Praha 2, Karlovo náměstí 13, E-420

Abstract

This thesis deals with designing and implementation of an application written in C++ for image management, which allows the images to be exported to a web gallery. It also focuses on the analysis of similar existing programs for creating web galleries. The resulting application allows the user to import images from directories and edit them directly in the application. The generated web gallery is customizable and compatible with HTML 4.01 strict.

Keywords: Web gallery, Images, Image editing, C++, HTML

Title translation: Photogallery with webpages generator

Obsah

Zadání práce	1	5.1.1 Obrázky programu	19
1 Úvod	3	5.2 Easy Website Photo Gallery	21
1.1 Cíl této práce	3	5.2.1 Obrázky programu	22
2 Obrázkové formáty	5	5.3 HTTPPhotos	23
2.1 Formát JPEG/EXIF	5	5.3.1 Obrázky programu	24
2.2 Formát TIFF	5	5.4 LightBox Video Web Gallery Creator	25
2.3 Formát PNG	6	5.4.1 Obrázky programu	27
3 EXIF metadata	7	5.5 iWebAlbum	28
3.1 Obrázky typu JPEG/EXIF	9	5.5.1 Obrázky programu	29
3.2 Obrázky typu TIFF	10	5.6 Web Album Generator	30
3.3 Obrázky typu PNG	12	5.6.1 Obrázky programu	32
3.4 Extrahovaná EXIF metadata	13	5.7 Visual LightBox	33
4 Hypertext Markup Language (HTML)	15	5.7.1 Obrázky programu	35
5 Analýza programů	17	6 Shrnutí analýzy programů	39
5.1 jAlbum	17	7 Návrh implementace	43
		7.1 Import obrázků	43
		7.1.1 Možnosti importu	43

7.1.2	Proces importu	44	7.6	Okno pro správu stylů	56
7.1.3	Získávání informací ze struktury složek	45	7.7	Okno pro správu barev	56
7.2	Výpis obrázků	46	7.8	Okno pro správu textů	56
7.3	Editace obrázku	47	7.9	Okno vygenerování galerie	56
7.3.1	Uživatelské rozhraní	47	8	Grafický návrh programu	59
7.3.2	Implementace úprav obrázku a efektů	48	9	Implementace programu	65
7.4	Přizpůsobení výsledné webové galerie	52	9.1	Systém projektů	65
7.4.1	Šablona	52	9.1.1	Formát projektového souboru	66
7.4.2	info.txt	52	9.1.2	Čtení projektového soubor ..	67
7.4.3	Formát textových vzorů pro dynamická data	53	9.1.3	Ukládání projektového souboru	67
7.4.4	Šablonové soubory	53	9.2	Import souborů	68
7.4.5	CSS galerie	54	9.2.1	Výběr souborů	68
7.4.6	Skripty (JS) galerie	54	9.2.2	Výber složky	68
7.4.7	Uživatelské rozhraní	54	9.3	Proces importování	68
7.5	Vytvoření galerie	55	9.4	Výpis položek galerie	70
7.5.1	Proces vytvoření	55	9.5	Editace obrázků	72
			9.6	Generování webové galerie	73

9.6.1 Textové vzory	73	A Literatura	91
9.6.2 Šablony	74		
9.6.3 Téma	74		
9.6.4 info.txt	76		
9.6.5 settings.txt	76		
9.6.6 Proces generování	76		
9.7 Nastavení programu	77		
9.8 Indikátor průběhu	78		
10 Testování programu	79		
10.1 Zpětná vazba	80		
10.2 Navržené úpravy	82		
10.3 Opětovné testování	84		
10.3.1 Zpětná vazba	85		
10.3.2 Navržené úpravy	85		
10.3.3 Další testování	87		
11 Závěr	89		
11.1 Rozšíření práce	90		

Obrázky

5.1 jAlbum, uživatelské rozhraní - výpis obrázků	19	5.12 Web Album Generator, uživatelské rozhraní - výpis obrázků	32
5.2 jAlbum, uživatelské rozhraní - editace obrázku	20	5.13 Web Album Generator, uživatelské rozhraní - generování galerie, sekce pro úpravu barev výsledné galerie	33
5.3 jAlbum, uživatelské rozhraní - výsledná webová galerie	20	5.14 Visual LightBox, uživatelské rozhraní - výpis obrázků	35
5.4 Easy Website Photo Gallery, uživatelské rozhraní - výpis obrázků	22	5.15 Visual LightBox, uživatelské rozhraní - generování galerie, sekce pro úpravu vzhledu náhledových obrázků	36
5.5 Easy Website Photo Gallery, uživatelské rozhraní - výsledná webová galerie	23	5.16 Visual LightBox, uživatelské rozhraní - generování galerie, sekce pro výstup	37
5.6 HTTPPhotos, uživatelské rozhraní - výpis obrázků	24	8.1 Hlavní okno programu s výpisem obrázků	60
5.7 HTTPPhotos, uživatelské rozhraní - editace obrázku	25	8.2 Import obrázků - dialogové okno pro výběr souborů	60
5.8 LightBox Video Web Gallery Creator, uživatelské rozhraní - hlavní okno programu	27	8.3 Import obrázků - dialogové okno pro výběr složky	61
5.9 LightBox Video Web Gallery Creator, uživatelské rozhraní - výsledná webová galerie	27	8.4 Okno pro editaci obrázku	61
5.10 iWebAlbum, uživatelské rozhraní - výpis obrázků	29	8.5 Okno pro editaci obrázku, s otevřeným menu	62
5.11 iWebAlbum, uživatelské rozhraní - generování galerie	30	8.6 Editace obrázku - dialogové okno nastavení efektu	62
		8.7 Okno přizpůsobení galerie	63

8.8 Okno vygenerování galerie 63

Tabulky

6.1 Shrnutí analýzy programů, část 1. 40

6.2 Shrnutí analýzy programů, část 2. 41

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Cipra** Jméno: **Petr** Osobní číslo: **483692**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra počítačů**
Studijní program: **Otevřená informatika**
Specializace: **Software**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Fotogalerie s generátorem webstránek

Název bakalářské práce anglicky:

Photogallery with webpages generator

Pokyny pro vypracování:

Analýzujte dostupné programy pro správu digitálních snímků se zaměřením na takové, které umožňují export do webových stránek ve formě fotoalba. Programy mezi sebou porovnejte z hlediska uživatelského a užitečných vlastností. Dále navrhnete a implementujete multiplatformní aplikaci v jazyku C++ pro správu fotografií s použitím knihovny QT. Musí podle pokynů zadavatele umožňovat import fotografií z adresářů, jednoduchou editaci (rotaci a výmaz, případně zvýšení jasu a kontrastu), anotaci textem a export do formátu HTML 4.01 strict. Případně volitelně do vyšší verze (XHTML). Aplikace bude umožňovat modifikaci vzhledu vytvářených stránek, velikost anotačního textu, velikost fotografie, data, pozadí atd. a vytvářet indexační stránku s orientačními obrázky atd. Aplikaci otestujte s použitím uživatelského testování alespoň tří osob, na základě zpětné vazby aplikaci vylepšete a znovu proveďte testování použitelnosti. Vytvořte ukázkové galerie a program umístěte do repozitáře sourceforge.

Seznam doporučené literatury:

- 1) Aplikace jAlbum, <https://jalbum.net/cs>
- 2) Aplikace Website Photo Gallery, <https://www.webgallerysoftware.com/>
- 3) Aplikace HTTPPhotos, <https://us.digicamsoft.com/softhttpphotos.html>
- 4) Lidwell, Holden, Butler, Universal Principles of Design, Revised and Updated: 115 Ways to Enhance Usability, Influence Perception, Increase Appeal, Make Better Design ... Design Decisions, and Teach through Design , 2010, ISBN-10 : 1592535879.

Jméno a pracoviště vedoucí(ho) bakalářské práce:

prof. Ing. Vlastimil Havran, Ph.D., Katedra počítačové grafiky a interakce

Jméno a pracoviště druhého(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **21.02.2021**

Termín odevzdání bakalářské práce: **04.01.2022**

Platnost zadání bakalářské práce: **19.02.2023**

prof. Ing. Vlastimil Havran, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací.
Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studenta



Kapitola 1

Úvod

Většina programů pro generování webových galerií nenabízí vysokou úroveň přizpůsobení výsledné webové galerie. Spousta z nich nabízí pouze několik šablon a nelze vše upravovat bez následné modifikace statických souborů po vygenerování galerie. Jen málokterý program také nabízí jednoduchou úpravu obrázků integrovanou přímo v programu.

Čtení, ukládání a úprava obrázkových dat s sebou přináší některé komplikace. Je potřeba prostudovat to, jak jednotlivé obrázkové formáty interně vypadají, aby se z nich dala extrahovat metadata. Je nutné postupně číst data ze souboru a logicky v souboru postupovat k požadovaným datům.

U ukládání dat do souborů HTML (Hypertext Markup Language) je nutné dát si pozor, aby výsledný obsah souboru představoval validní HTML 4.01 strict.



1.1 Cíl této práce

Cílem této práce je analyzovat již existující programy pro generování webových galerií, podobný program navrhnout a postupně projít veškeré záležitosti problematiky některých funkcí. Nakonec je nutné program i implementovat v programovacím jazyce C++.

Kapitola 2

Obrázkové formáty

Program musí umět pracovat s různými obrázkovými formáty, konkrétně alespoň s JPEG/EXIF, TIFF a případně i PNG. V této kapitole se nachází základní informace o daném obrázkovém formátu a rozdíly mezi nimi.

2.1 Formát JPEG/EXIF¹

JPEG/EXIF je obrázkový formát hojně využívaný pro kompresi obrázkových dat pomocí metody JPEG, především pro digitální fotografie. Podporuje možnost změnit úroveň komprese, za to ale nepodporuje průhlednost pixelů.

2.2 Formát TIFF²

TIFF je flexibilní a přizpůsobitelný obrázkový formát pro zpracování obrázků a dat v rámci jednoho souboru. Podporuje bezztrátové ukládání obrázkových dat, lze ho použít i pro uchovávání JPEG obrázků a také umožňuje zahrnout tzv. *hlavičkové značky*.

■ 2.3 Formát PNG³

PNG je rozšiřitelný obrázkový formát souboru pro bezztrátové, přenosné a komprimované úložiště rastrových obrázků. Podporuje například truecolor (až 48 bitů na pixel), plný alfa kanál (průhlednost) a detekci korupce souboru.



Kapitola 3

EXIF metadata

Program by měl umět získávat různé informace z obrázků během procesu importu. Konkrétně by měl program získávat metadata EXIF z obrázků těchto typů: *JPEG/EXIF*, *TIFF* a případně i *PNG*. Každý z těchto obrázkových formátů má odlišnou strukturu, nicméně rozebrání (tzv. parsování) EXIF metadat je u všech v zásadě stejné.^{4,5,6} Stačí tedy, aby program uměl pro každý z těchto obrázkových formátů extrahovat blok EXIF, jeho následné rozebrání se pak bude u všech provádět stejným způsobem.

Poznámka: Jednotlivé sekce této kapitoly se zabývají pouze extrakcí EXIF metadata a ne přímo obrázkových dat, ani jejich dekodování apod. Nejlépe by se problém čtení obrázkových dat řešil pomocí nějaké již existující knihovny, která tuto funkcionalitu podporuje. Tím by se dala přecíst jak obrázková data, tak metadata obrázku jedním průchodem datového proudu a zároveň by veškeré případné problémy a komplikace při čtení obrázkových dat byly vyřešeny.

Příprava: Mějme nějaký datový proud (stream) a následující funkce operující nad ním:

```
// Zkontroluje, zdali je v datovém proudu další v pořadí daná
// hodnota daného typu.
<Type> bool stream::hasNext(Type val);
// Přečte v datovém proudu následujících *length* bajtů a převede je
// do daného typu.
<Type> Type stream::readNext(uint32 length);
// Přeskočí v datovém proudu následujících *length* bajtů.
void stream::skip(uint32 length);
// Nastaví limit datového proudu na *length* bajtů. Jakmile je
// přečteno *length* bajtů, stream vyhodí EOF.
void stream::limit(uint32 length);
// Nastaví pořadí bajtů v datovém proudu. Hodnoty přečtené z
// datového proudu budou převáděny s pomocí tohoto pořadí. Např.:
// Datový proud má hodnoty 4D 2E ..., po zavolání
// readNext<uint16>(2), pokud je byteOrder == Byte_Order::LE, se
// vrátí hodnota 0x2E4D, pokud je byteOrder == Byte_Order::BE, se
// vrátí hodnota 0x4D2E.
void stream::setByteOrder(Byte_Order byteOrder);
// Posune se v datovém proudu na pozici *position*.
void stream::seek(uint32 position);
```

Mějme ještě definovány následující typy a funkci, která vezme nějaké argumenty a vytvoří z nich *EXIF_Data*:

```
EXIF_Data_Entry {
    uint16 id, type;
    uint32 count, value;
}

EXIF_Data {
    vector<EXIF_Data_Entry> entries;
}

EXIF_Data create_EXIF_Data(/* argumenty */);
```

Tyto funkce a typy pak budou používány v pseudokódech pro extrakci EXIF metadat u jednotlivých typů obrázků.

3.1 Obrázky typu JPEG/EXIF

Podle specifikace⁴ je soubor strukturován pomocí tzv. markerů tvaru `0xFFxx`, kde `xx` je specifické číslo označující typ markeru. Za ním vždy (až na SOI a EOI markery, viz dále) následují 2 bajty pro délku obsahu (v bajtech) sekce označené daným markerem. Tato délka zahrnuje i tyto 2 bajty, takže je nutné přečíst pouze následujících $N - 2$ bajtů, kde N je přečtená délka obsahu.

Soubor JPEG/EXIF začíná markerem **SOI** (Start Of Image) s hodnotou `0xFFD8` a končí markerem **EOI** (End Of Image) s hodnotou `0xFFD9`. Hledaná EXIF metadata jsou pak v sekci označené markerem **APP1** (Application marker 1) s hodnotou `0xFFE1`.

Pro extrakci EXIF metadat je tedy nutné přeskokovat sekce tak dlouho, dokud nepřečteme marker **APP1** a obsah sekce označené tímto markerem pak přečíst. Následně můžeme tyto data předat proceduře, která data rozebere. Konkrétně se v tomto případě jedná o stejnou proceduru, která čte soubor typu TIFF.

Pseudokód: Zde je ukázka, jak by funkce pro extrakci dat z obrázků typu JPEG/EXIF mohla vypadat (za pomoci předem definovaných funkcí, viz Příprava):

```
EXIF_Data parse_JPEG_EXIF(Stream stream) {
    // SOI (Start Of Image)
    if(!stream.hasNext<uint16>(0xFFD8)) return NULL;
    stream.skip(2); // Přeskočíme SOI
    bool found_app1 = false;
    while(!stream.isEOF())
        // Zdali je další v pořadí nějaký marker
        && stream.hasNext<uint8>(0xFF)) {
        uint8 marker = stream.readNext<uint8>(2);
        // Zdali jsme našli APP1 marker
        if(marker == 0xFFE1) {
            found_app1 = true;
            break; // Nemusíme dále přeskokovat markery
        }
        uint16 size = stream.readNext<uint16>(2);
        stream.skip(size - 2);
    }
    if(found_app1) {
        uint16 size = stream.readNext<uint16>(2);
        // Zdali se doopravdy jedná o EXIF blok
        if(!stream.hasNext<string>("Exif\0\0"))
```

```

        return NULL;
    // Přeskočíme EXIF header
    stream.skip("Exif\0\0".size());
    // Přečteme a rozebereme EXIF data
    return parse_TIFF_EXIF(stream, size);
}
// Jinak EXIF data nebyla nalezena
return NULL;
}

```

3.2 Obrázky typu TIFF

Podle specifikace⁵ začíná soubor typu TIFF tzv. image file header o velikosti 8 bajtů. První 2 bajty říkají v jakém pořadí jsou bajty uspořádány, tedy zdali se jedná o little-endian (označeno jako II = 0x4949), nebo o big-endian (označeno jako MM = 0x4D4D). Dále však budu uvažovat pouze variantu s pořadím bajtů v pořadí little-endian. Následují 2 bajty označující, že se jedná o TIFF soubor, konkrétně hodnota 42, v HEX 0x2A00. Další 4 bajty označují offset IDF (Image file directory), kde se nacházejí informace o obrázku. Offset je vzhledem k začátku souboru, tedy první bajt má offset 0.

Image file directory: Skládá se z tzv. IFD entries, tedy záznamů, kde každý obsahuje nějakou informaci o obrázku. První 2 bajty označují počet takovýchto záznamů, následuje jejich sekvence, která končí buď hodnotou offsetu na další IFD, nebo 4 bajty nul znamenající, že se jednalo o poslední IFD.

IFD záznam (entry): Skládá se z 12 bajtů - první 2 bajty je tag označující záznam, následují 2 bajty reprezentující typ záznamu, pak 2 bajty reprezentující počet hodnot a nakonec 4 bajty označující offset v souboru, kde se hodnota záznamu nachází, nebo rovnou hodnotu záznamu a to pouze tehdy, pokud se hodnota vejde do 4 bajtů. Pokud je hodnota menší než 2 bajty, pak je zarovnaná doleva (například hodnota 0x2A3C je uložena jako 0x2A3C0000).

EXIF IFD záznamy: Podle zdrojů^{4,7,8} se EXIF záznamy ukládají do tzv. Exif SubIFD, na kterou odkazuje TIFF Private tag s označením 0x6987. Právě tyto záznamy chceme z obrázku extrahovat. Formát dat v Exif SubIFD je stejný jako formát IFD.

Pseudokód: Zde je ukázka, jak by funkce pro extrakci dat z obrázků typu TIFF mohla vypadat (za pomoci předem definovaných funkcí, viz Příprava):

```

IFD_Entry {
    uint16 id, type;
    uint32 count, value;
}

IFD_Entry parse_IFD_Entry(Stream stream) {
    IFD_Entry entry = stream.readNext<IFD_Entry>(12);
    return entry;
}

vector<IFD_Entry> parse_IFD(Stream stream) {
    vector<IFD_Entry> entries;
    uint16 num_entries = stream.readNext<uint16>(2);
    // Přečteme všechny záznamy v IFD
    for(uint16 i = 0; i < num_entries; ++i) {
        IFD_Entry entry = parse_IFD_Entry(stream);
        entries.add(entry);
    }
    return entries;
}

EXIF_Data parse_TIFF_EXIF(Stream stream, uint32 limit) {
    if(limit > 0) stream.limit(limit);
    // Přečteme a nastavíme pořadí bajtů
    uint16 byte_order = stream.readNext<uint16>(2);
    if(byte_order == 0x4949) stream.setByteOrder(Byte_Order::LE);
    else stream.setByteOrder(Byte_Order::BE);
    // Offset první IFD
    uint32 offset = stream.readNext<uint32>(4);
    // Pro každou IFD (alespoň 1 musí existovat)
    do {
        stream.seek(offset);
        uint16 num_entries = stream.readNext<uint16>(2);
        // Přečteme všechny záznamy v IFD
        for(uint16 i = 0; i < num_entries; ++i) {
            IFD_Entry entry = parse_IFD_Entry(stream);
            if(entry.id == 0x6987) { // Jedná se o ExifOffset záznam
                stream.seek(entry.value); // Přejdeme na daný offset
                // Rozebereme EXIF záznamy
                vector<IFD_Entry> exif_entries = parse_IFD(stream);
                return create_EXIF_Data(exif_entries);
            }
        }
        // Posuneme se na další IFD
        offset = stream.readNext<uint32>(4);
    } while(offset != 0);
    // EXIF metadata nenalezena
    return NULL;
}

```

3.3 Obrázky typu PNG

Podle specifikace⁹ začíná soubor typu PNG tzv. PNG signature (8 bajtů) s hodnotou 0x89504E470D0A1A0A. Poté následuje sekvence tzv. PNG chunks. Každý chunk se skládá ze 4 částí: jeho velikost (4 bajty; jedná se pouze o velikost jeho dat, tedy bez hodnoty velikosti, typu a CRC), jeho typ (4 bajty), jeho data (o předešle uvedené velikosti) a kontrolní součet CRC (4 bajty).

EXIF Chunk: Jedním ze sekvence PNG chunks může být EXIF chunk,⁶ který je označen typem 0x65584966 ("eXIf"). Tento chunk má stejný formát jako TIFF soubor, a tudíž pro jeho rozebrání lze použít stejnou proceduru jako pro rozebrání EXIF metadat u souboru TIFF.

Pseudokód: Zde je ukázka, jak by funkce pro extrakci dat z obrázků typu PNG mohla vypadat (za pomoci předem definovaných funkcí, viz Příprava):

```

EXIF_Data parse_PNG_EXIF(Stream stream) {
    // Zdali soubor obsahuje PNG signature
    if(!stream.hasNext<uint64>(0x89504E470D0A1A0A)) return NULL;
    // Pro každý PNG chunk (alespoň 1 musí existovat)
    do {
        uint32 chunk_length = stream.readNext<uint32>(4);
        uint32 chunk_type = stream.readNext<uint32>(4);
        if(chunk_type == 0x65584966) { // Jedná se o EXIF Chunk
            // Rozebereme EXIF chunk
            return parse_TIFF_EXIF(stream, chunk_length);
        }
        // Přeskočíme CRC (je možné ho případně využít pro kontrolu
        // dat)
        stream.readNext<uint32>(4);
    } while(!stream.isEOF());
    // EXIF metadata nenalezena
    return NULL;
}

```

■ 3.4 Extrahovaná EXIF metadata

Extrahovaná EXIF metadata (v pseudokódech jako *EXIF_Data*) pak obsahují jednotlivé záznamy. U každého z nich je evidována identifikace záznamu, tzn. o jakou informaci se přesně jedná, a hodnota záznamu. U jednotlivých dat je hodnota reprezentována jinak, a tudíž, pokud by se tyto údaje měly nějak uživateli předávat, je nutné pro každou informaci specifikovat, v jakých jednotkách/jakém formátu by se měla zobrazovat. Například záznam *ExposureTime* je udáván v sekundách (číslo), ale záznam *DateTimeOriginal* je udáván jako text (ASCII). Řešením by mohlo být, že by se předem určená data (pokud v souboru existují) zobrazila správně formátovaná, a zbytek extrahovaných dat by se zobrazoval v tzv. RAW hodnotách, tedy v takových hodnotách, které byly nalezeny v souboru.

Kapitola 4

Hypertext Markup Language (HTML)¹⁰

HTML je standardní značkovací jazyk pro dokumenty určené k zobrazení ve webovém prohlížeči.

Základními bloky HTML dokumentu jsou HTML prvky. Každý dokument je strukturovaný pomocí těchto prvků, které jsou v kódu zapsány tzv. tagy, jež jsou označeny pomocí lomených závorek. Tagy mohou být otevírající (<p>), zavírající (</p>) či samouzavíratelné (). Pro otevírající tagy by se v dokumentu měl nacházet do páru ještě zavírající tag, kde mezi nimi lze mít další HTML prvky. Pro samouzavíratelné tagy se napíše před koncovou lomenou závorkou lomítko. Každý tag také může mít různé atributy, které dále specifikují vlastnosti daného prvku. Každý atribut má své jméno a případně i hodnotu. HTML prvek pak může vypadat například takto: či <div class="..." style="...">...</div>.

Častokrát je doprovázen technologiemi jako je CSS a JavaScript pro doručení lépe graficky vypadajícího a dynamičtějšího obsahu.

Kapitola 5

Analýza programů

V této kapitole se nachází analýzy jednotlivých programů v příslušných sekcích. Konkrétní body analýzy jsou (1) import obrázků do programu, (2) výpis obrázků v rozhraní programu, (3) editace obrázků, (4) přizpůsobení webové galerie, (5) výsledná webová galerie a případně další body specifické pro daný program.

5.1 jAlbum

Verze: 22.1

Oficiální stránky: <https://jalbum.net/cs/>

Operační systém: Windows, Mac, Linux (a další)

Typ programu: Placený (Free trial)

1. Import obrázků

- Pomocí dialogového okna pro výběr souborů.
- Pomocí Drag and Drop.
- Lze specifikovat popis a tagy u obrázku.
- Dodatečné informace (jako například datum pořízení) u obrázku nejsou importovány.

2. Výpis obrázků

- Změna pořadí pomocí Drag and Drop.
- Přejmenování pomocí dvojitého kliku.
- Možnost filtrace obrázků.
- Možnost vytváření hierarchie obrázků pomocí složek.

3. Editace obrázku

- Funkce: překlopení, otočení podle úhlu, ořez, auto-korekce, úprava jasů, kontrastu, sytosti a gamma.
- Efekty: červené oči, zaostření, rozmazání, stupně šedi, mozaika, inverze.
- Přidání textu do obrázku, jeho zarovnání, barva a font (včetně možnosti přidání stínu), náhled v reálném čase.
- Zobrazení histogramu při editaci.
- Jednoduché překlíkávání mezi obrázky v editačním módu pomocí šipek.

4. Přizpůsobení výsledné webové galerie

- Možnost vybrat si z již existujících šablon.
- Možnost změnit barvy pozadí stránky a navigace, odsazení, zaoblení rohů, fonty a barvy Lightboxu.
- Možnost přidat a odebrat sociální sítě u sdílení alba.
- Možnost vložit vlastní HTML do hlavičky a těla stránky, plus přidat i vlastní CSS a JS.
- Dodatečná správa SEO u obrázků a nadpisů ve webové galerii.

5. Výsledná webová galerie

- Zobrazení mřížky s obrázky a jejich popisky na indexové stránce.
- Rozklik obrázku pro vstup do slideshow.
- Možnost vyhledávat obrázky v galerii podle popisu či názvu.

6. Optimalizace obrázků

- Možnost zvolit kvalitu JPEG výsledného obrázku.

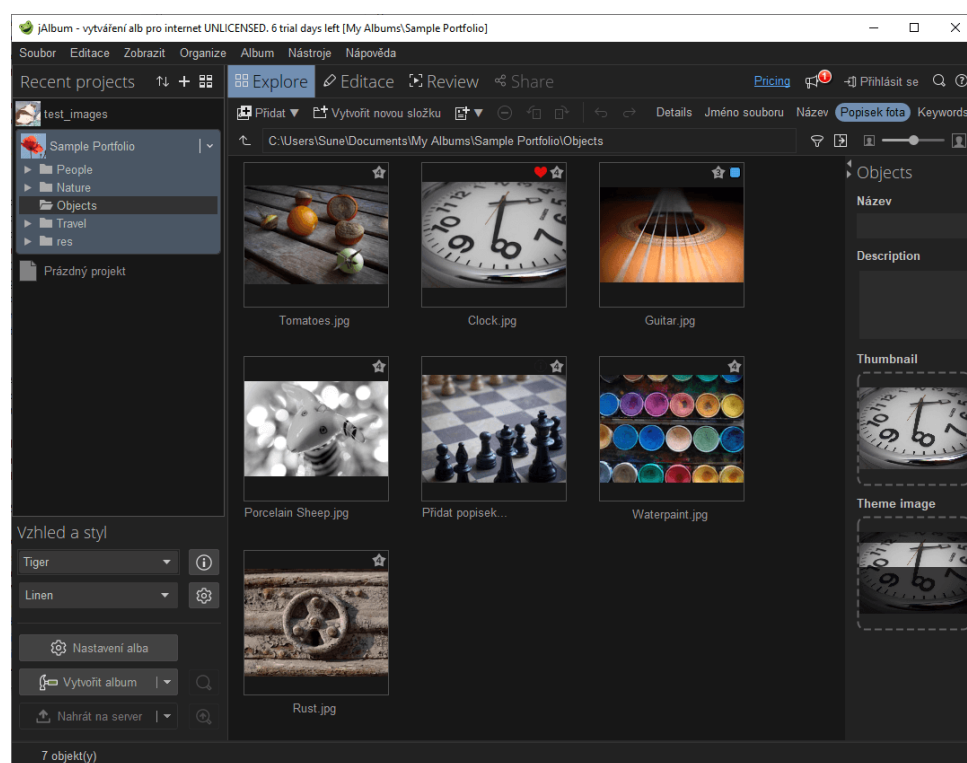
7. Vytvoření galerie

- Jedním kliknutím se album vytvoří.
- Spustí se i lokální server na portu 8080 pro zhlédnutí webové galerie.

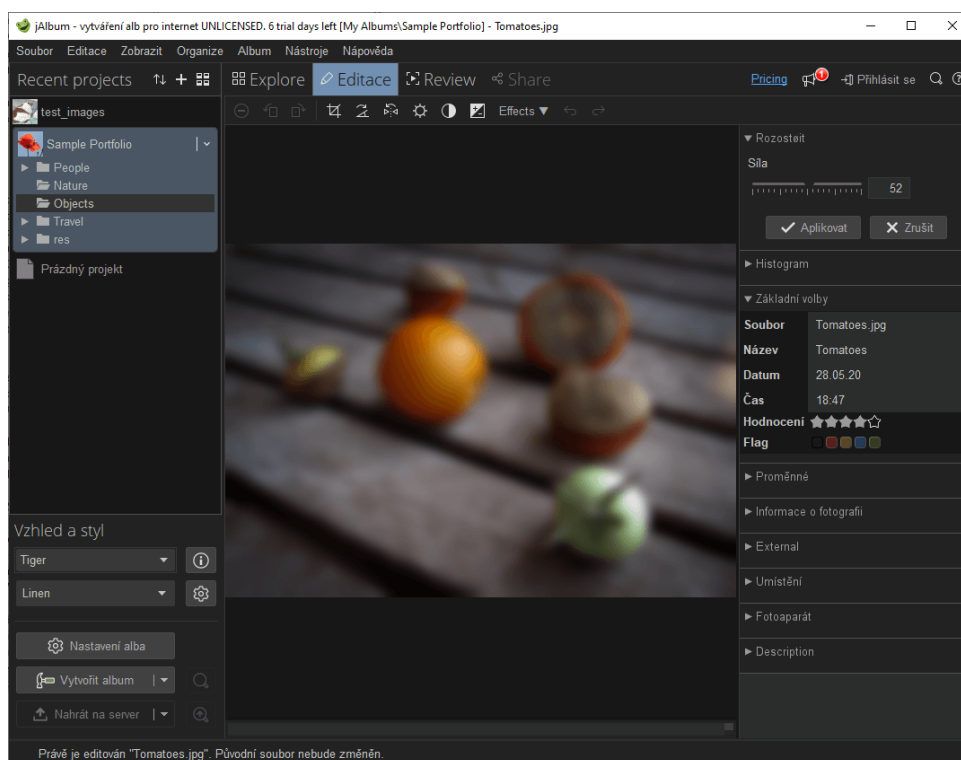
Hodnocení.

1. Co se mi líbí
 - Intuitivnost rozhraní, i když program nabízí spoustu funkcí, není rozhraní příliš přehlcené.
 - Jednoduchost změny barev a dalších nastavení vygenerované galerie.
2. Co se mi nelíbí
 - Složitě vypadající výsledná galerie, více efektů a přechodů než by bylo nutné (nenašel jsem možnost toto vypnout bez použití CSS).

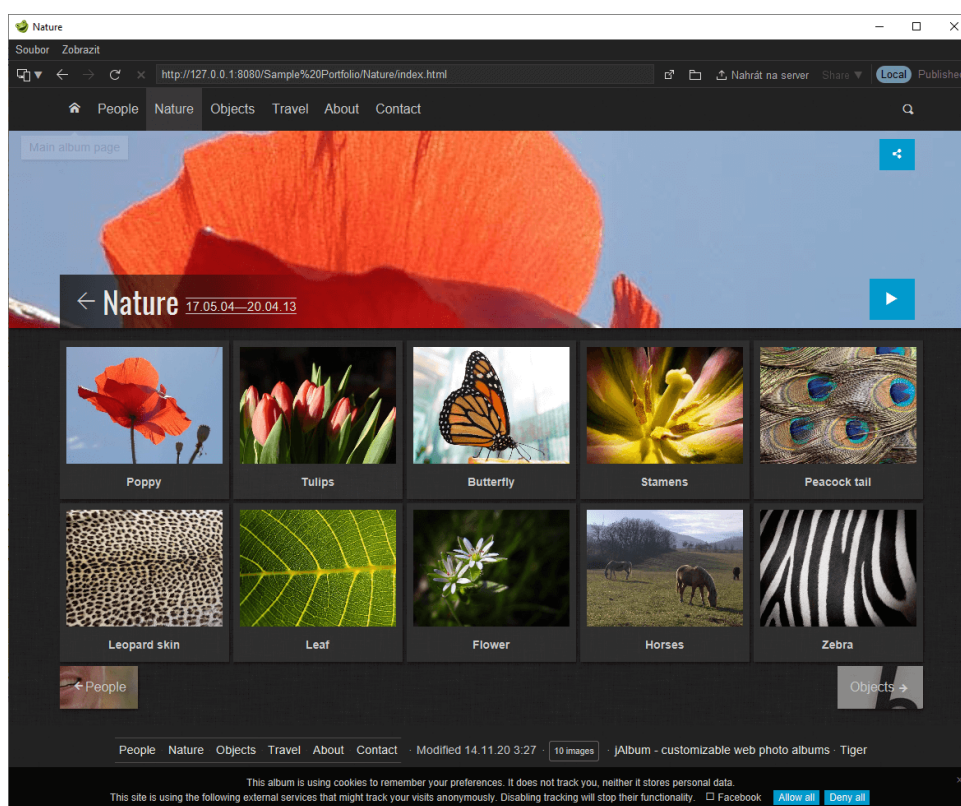
5.1.1 Obrázky programu



Obrázek 5.1: jAlbum, uživatelské rozhraní - výpis obrázků



Obrázek 5.2: jAlbum, uživatelské rozhraní - editace obrázku



Obrázek 5.3: jAlbum, uživatelské rozhraní - výsledná webová galerie

5.2 Easy Website Photo Gallery

Verze: 1.1

Oficiální stránky: <https://www.webgallerysoftware.com/>

Operační systém: Windows

Typ programu: Placený (Free trial)

1. Import obrázků

- Pomocí dialogového okna pro výběr souborů.
- Pomocí Drag and Drop.
- Dodatečné informace (jako například datum pořízení) u obrázku nejsou importovány.

2. Výpis obrázků

- Obrázky jsou zobrazeny v seznamu s jejich názvy.
- Změna pořadí pomocí tlačítek.
- Náhled obrázku v postranním panelu.

3. Editace obrázku

- Funkce: změna jasu, kontrastu, světlost, tmavost, saturace.
- Efekty: rozostření, šum, odstíny šedi, inverze.

4. Přizpůsobení výsledné webové galerie

- Možnost vybrat si z již existujících šablon.
- Úprava stylů a struktury vygenerované webové galerie pomocí HTML a CSS, texty jsou samostatně upravitelné v textových polích.
- Možnost změnit fonty na jednotlivých typech stránek (Index, Detail obrázku atd.)

5. Výsledná webová galerie

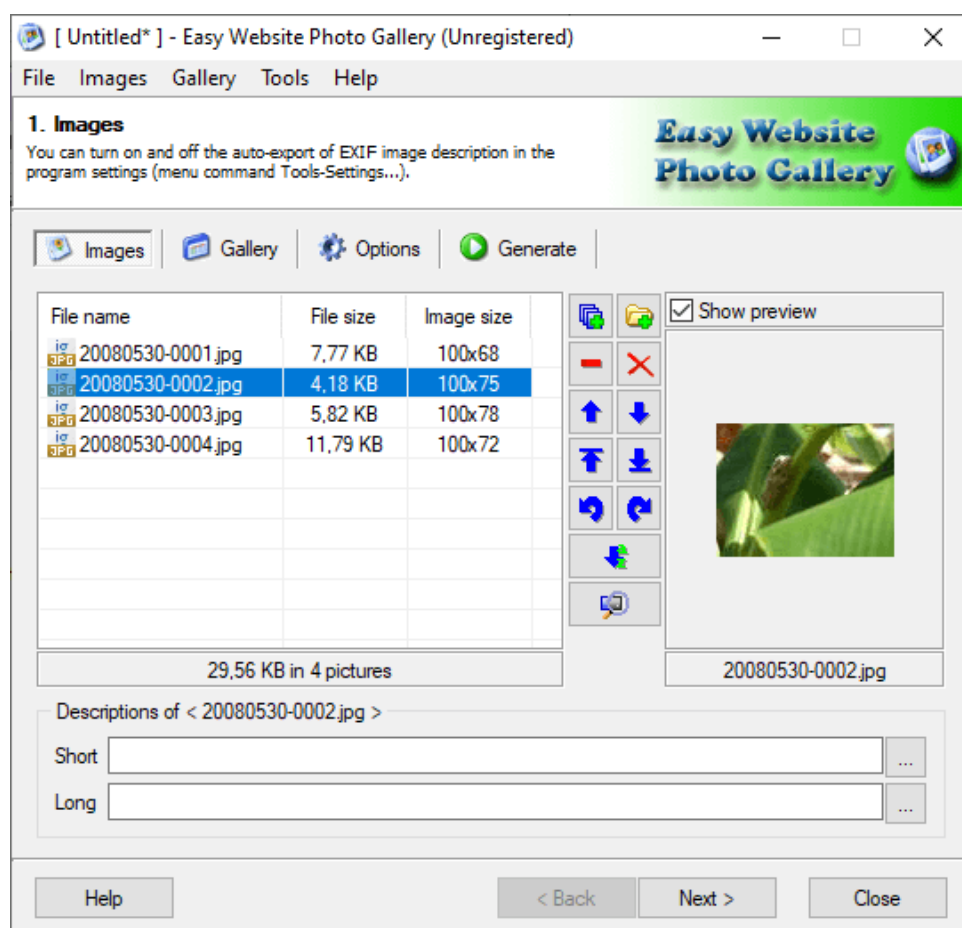
- Zobrazení mřížky s obrázky a jejich popisky na indexové stránce.
- Rozklik obrázku pro vstup na jeho detail.

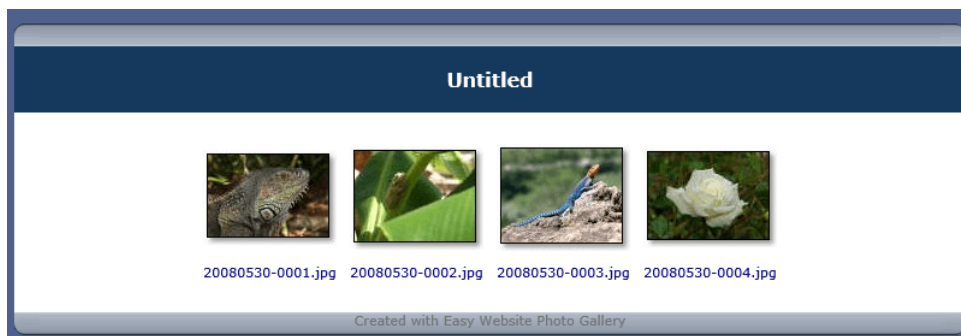
6. Optimalizace obrázků

- Možnost zvolit kvalitu u změny rozměrů obrázku.

Hodnocení.

1. Co se mi líbí
 - Jednoduchost programu a výsledné galerie.
2. Co se mi nelíbí
 - Úprava barev a stylu pouze přes CSS.

5.2.1 Obrázky programu**Obrázek 5.4:** Easy Website Photo Gallery, uživatelské rozhraní - výpis obrázků



Obrázek 5.5: Easy Website Photo Gallery, uživatelské rozhraní - výsledná webová galerie

5.3 HTTPPhotos

Verze: 4.10

Oficiální stránky: <https://us.digicamsoft.com/softhttpphotos.html>

Operační systém: Windows

Typ programu: Free

1. Import obrázků
 - Pomocí dialogového okna pro výběr souborů.
 - Dodatečné informace (jako například datum pořízení) u obrázku nejsou importovány.
2. Výpis obrázků
 - Změna pořadí pomocí Drag and Drop.
3. Editace obrázku
 - Funkce: změna jasu, kontrastu.
 - Efekty: rozostření, zostření, stupně šedi, inverze.
4. Přizpůsobení výsledné webové galerie
 - Možnost změnit rozložení indexové stránky.
 - Možnost změnit font, barvy pozadí a textu.
5. Výsledná webová galerie
 - Zobrazení mřížky s náhledy obrázků na indexové stránce.
 - Rozklik obrázku pro vstup na jeho detail v LightBoxu.

Hodnocení.

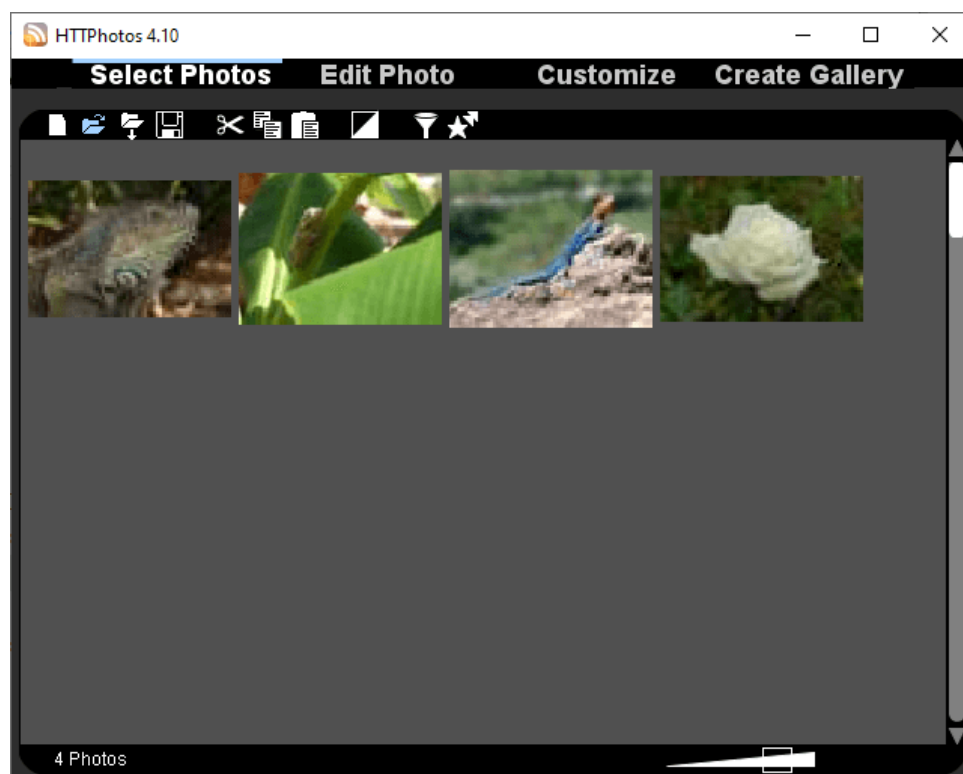
1. Co se mi líbí

- Jednoduchost programu a výsledné galerie.
- Program není nutné instalovat.

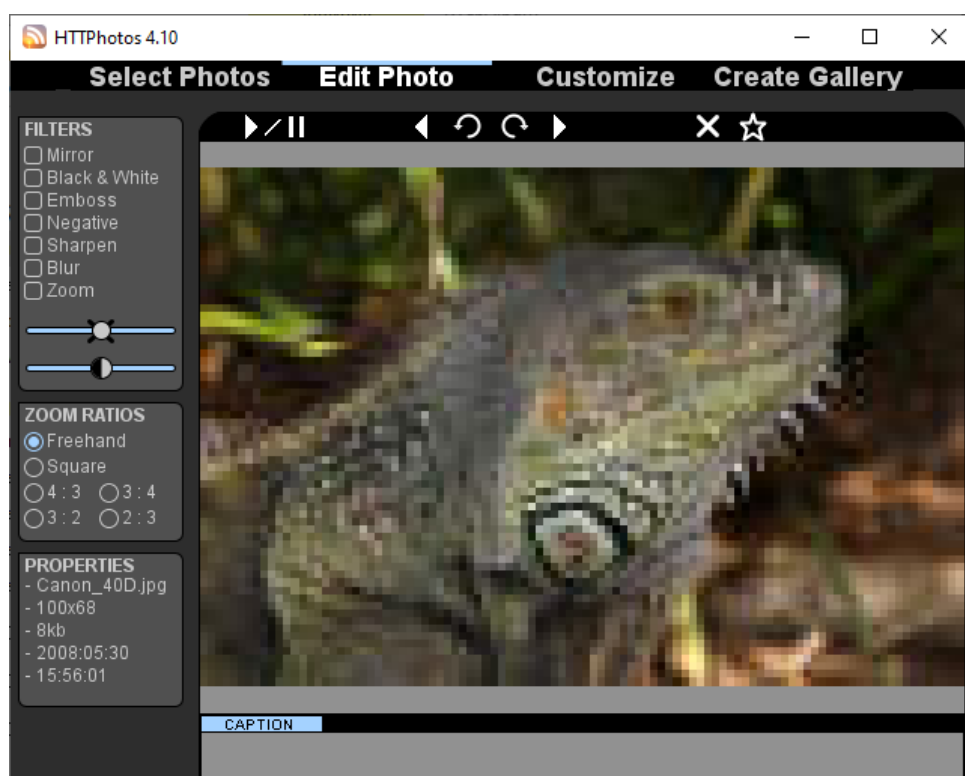
2. Co se mi nelíbí

- Ikona pro editaci obrázku je Lupa (místo například ikony tužky apod.)
- Nelze zvolit sílu efektů u editace obrázku (až na jas a kontrast).
- Překlopení obrázku je ve filtrech, zatímco jeho otočení je ve výpisu obrázků.
- Rozhraní aplikace je méně přehledné.

5.3.1 Obrázky programu



Obrázek 5.6: HTTPhotos, uživatelské rozhraní - výpis obrázků



Obrázek 5.7: HTTPotos, uživatelské rozhraní - editace obrázku

5.4 LightBox Video Web Gallery Creator

Verze: 2.1.7

Oficiální stránky: <http://pranas.net/webgallerycreator/>

Operační systém: Windows

Typ programu: Free

1. Import obrázků
 - Pomocí dialogového okna pro výběr souborů (pouze výběr složky).
 - Dodatečné informace (jako například datum pořízení) u obrázku nejsou importovány.
2. Výpis obrázků - **Není**
3. Editace obrázku - **Není**
4. Přizpůsobení výsledné webové galerie
 - Možnost změnit velikost náhledů.

- Možnost změnit počet řádků a sloupců mřížky na stránce indexu.
- Možnost upravit kvalitu u náhledu obrázku.
- Možnost upravit CSS výsledných stránek.

5. Výsledná webová galerie

- Zobrazení mřížky s náhledy obrázků a jejich popisky na indexové stránce.
- Rozklik obrázku pro vstup na jeho detail v LightBoxu.

Hodnocení.

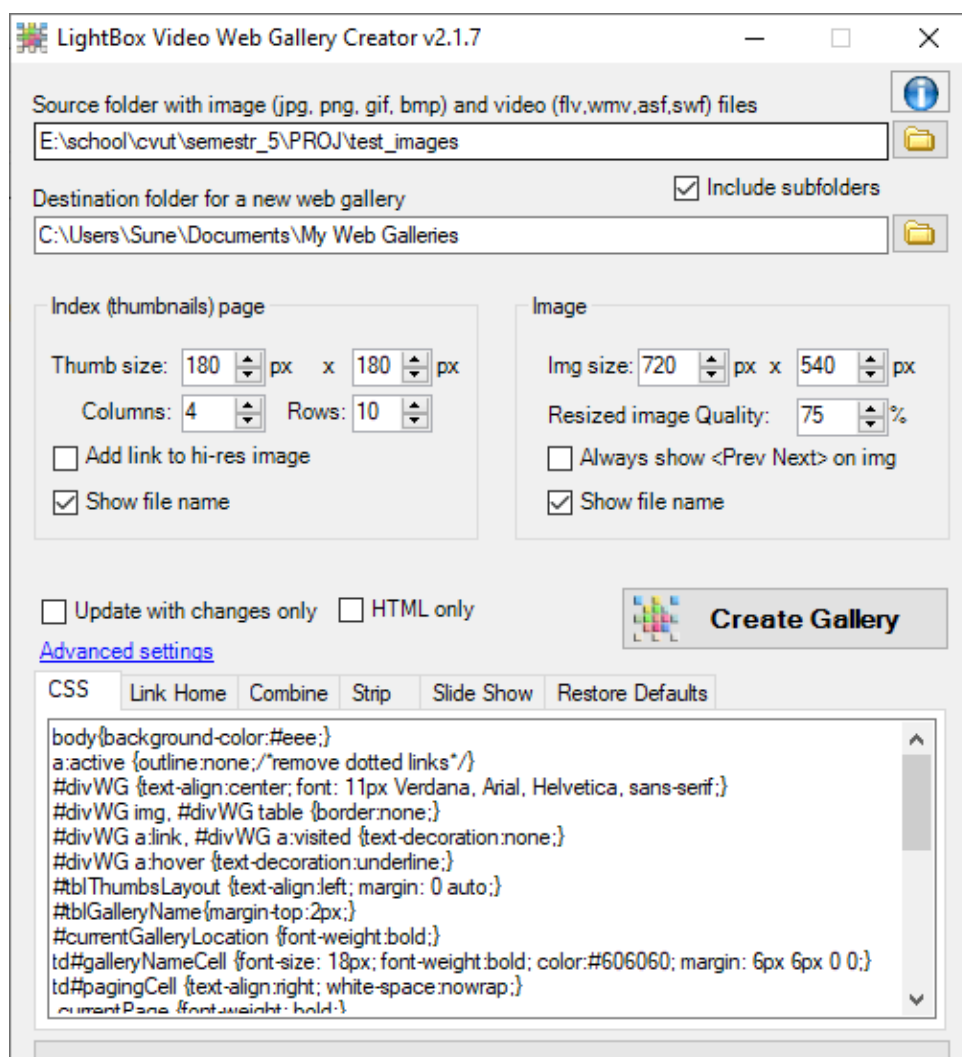
1. Co se mi líbí

- Jednoduchost programu a výsledné galerie.

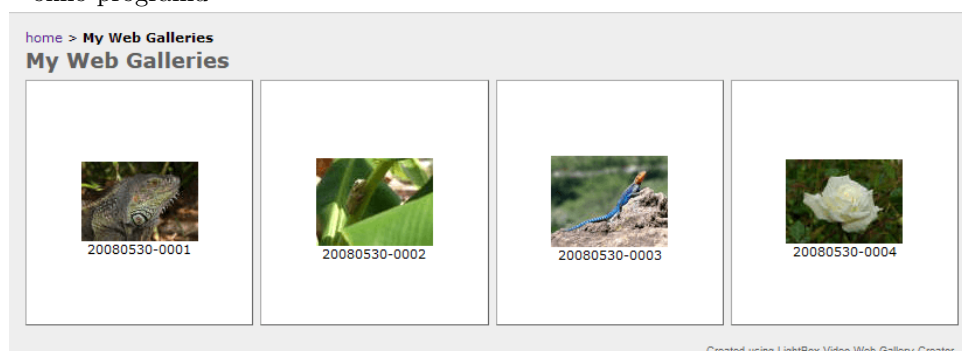
2. Co se mi nelíbí

- Program je nutné instalovat, i přesto, že se pouze extrahuje jeden EXE soubor.
- Nelze změnit barvy, fonty apod. bez znalosti CSS.
- Není možná úprava obrázků před vygenerováním galerie.

5.4.1 Obrázky programu



Obrázek 5.8: LightBox Video Web Gallery Creator, uživatelské rozhraní - hlavní okno programu



Obrázek 5.9: LightBox Video Web Gallery Creator, uživatelské rozhraní - výsledná webová galerie

5.5 iWebAlbum

Verze: 2.02

Oficiální stránky: <http://eunq.com/>

Operační systém: Windows

Typ programu: Free

1. Import obrázků

- Pomocí dialogového okna pro výběr souborů či složky.
- Pomocí Drag and Drop.
- Možnost změnit název a přidat komentář u jednotlivých obrázků.
- Dodatečné informace (jako například datum pořízení) u obrázku nejsou importovány.

2. Výpis obrázků

- Obrázky jsou zobrazeny v seznamu s jejich názvy a informacemi.
- Změna pořadí pomocí tlačítek.
- Náhled obrázku v postranním panelu.

3. Editace obrázku

- Funkce: Otočení a překlopení.

4. Přizpůsobení výsledné webové galerie

- Možnost vybrat si z již existujících šablon a stylů.
- Možnost změnit rozměry obrázků.
- Možnost změnit počet řádků a sloupců v mřížce.
- Možnost změnit název alba a komentáře k albu.

5. Vytvoření galerie

- Dodatečné informace (jako například datum pořízení) u obrázku nejsou zachovány.

Hodnocení.

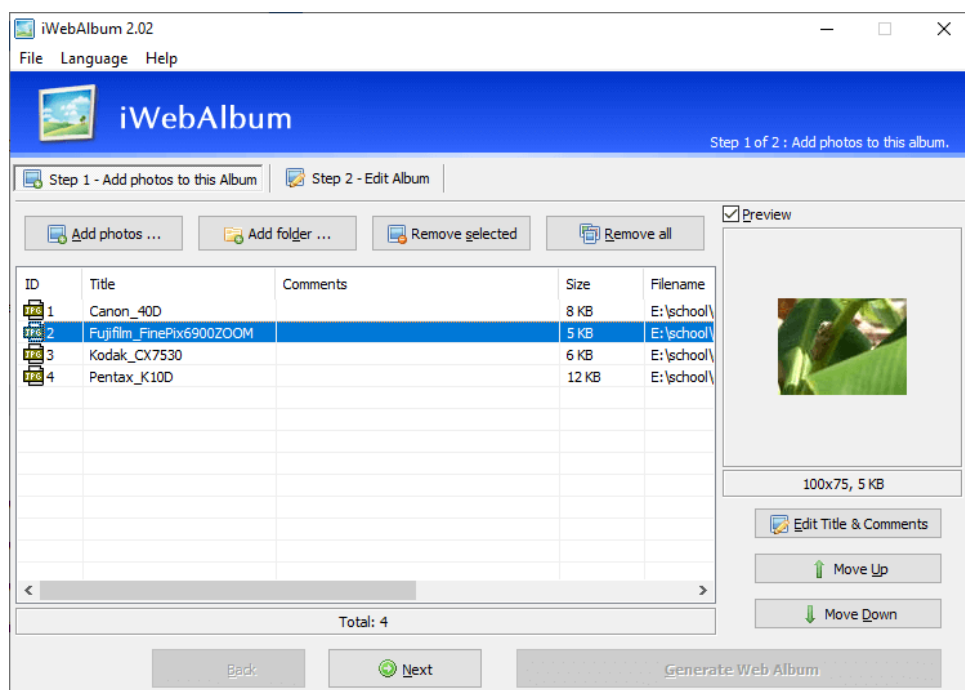
1. Co se mi líbí

- Jednoduchost programu a výsledné galerie.

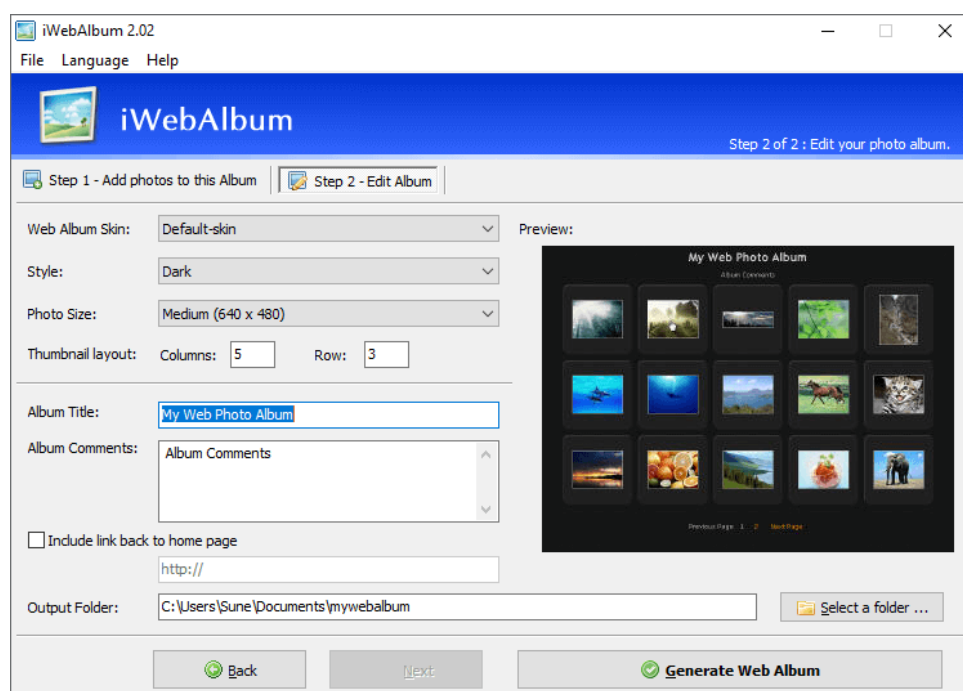
2. Co se mi nelíbí

- Lightbox u výsledné galerie funguje pouze v Internetu Exploreru a starší verzi prohlížeče Edge. Galerie je tedy nepoužitelná v jiných prohlížečích. Otestováno v IE 11.572.19041.0, Edge 44.19041.423.0, Firefox 82.0.2, Chrome 86.0.4240.111 a Opera 72.0.3815.186.
- Obrázky nelze v programu ořezávat či jinak upravovat, vyjma otáčení.

5.5.1 Obrázky programu



Obrázek 5.10: iWebAlbum, uživatelské rozhraní - výpis obrázků



Obrázek 5.11: iWebAlbum, uživatelské rozhraní - generování galerie

5.6 Web Album Generator

Verze: 1.8.2

Oficiální stránky:

<https://www.ornj.net/software/web-album-generator>

Operační systém: Windows

Typ programu: Free

1. Import obrázků

- Pomocí dialogového okna pro výběr souborů.
- Možnost změnit název a přidat komentář u jednotlivých obrázků.
- Dodatečné informace (jako například datum pořízení) u obrázku nejsou importovány.

2. Výpis obrázků

- Obrázky jsou zobrazeny v seznamu s jejich názvy.
- Změna pořadí pomocí tlačítek.
- Náhled obrázku v postranním panelu.

3. Editace obrázku

- Funkce: Otočení.

4. Přizpůsobení výsledné webové galerie

- Možnost vybrat si z již existujících šablon.
- Možnost změnit název alba a komentáře k albu.
- Možnost změnit barvy textu a pozadí.
- Možnost změnit počet řádků a sloupců v mřížce.
- Možnost změnit rozměry obrázků.
- Možnost změnit pozici navigace na stránce.

5. Vytvoření galerie

- Dodatečné informace (jako například datum pořízení) u obrázku nejsou zachovány.

Hodnocení.

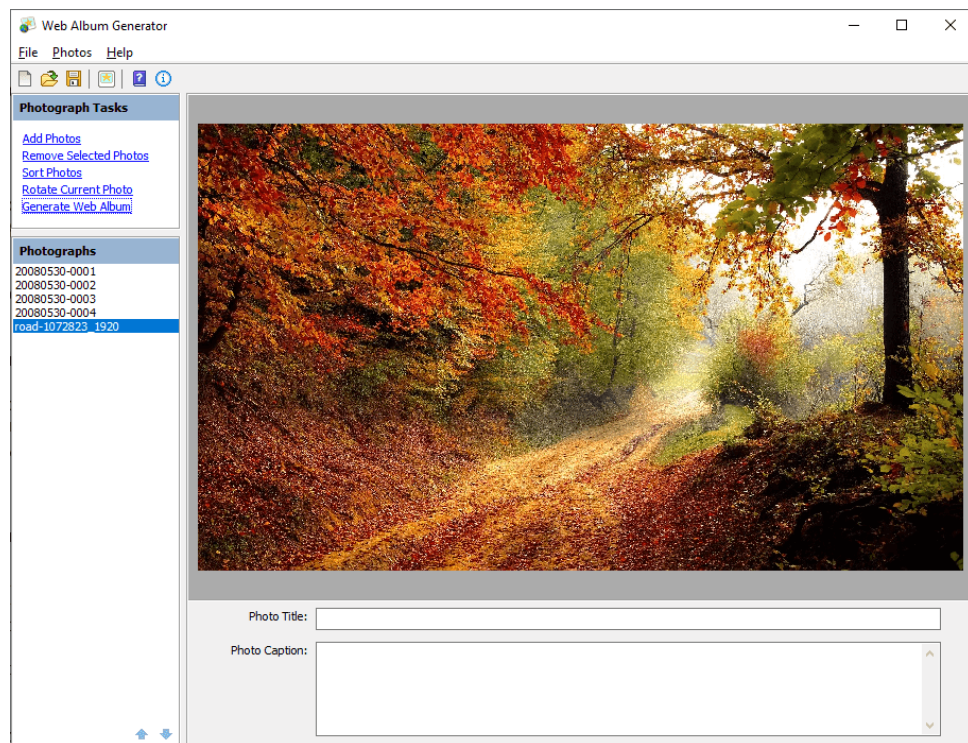
1. Co se mi líbí

- Jednoduchost programu a výsledné galerie.

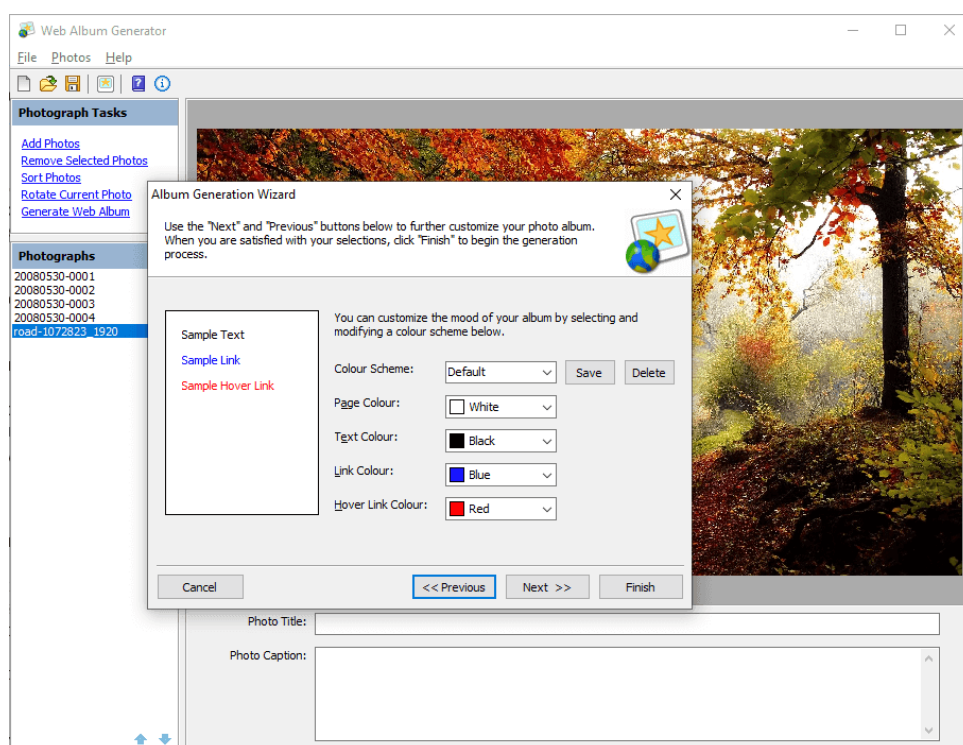
2. Co se mi nelíbí

- Výsledná galerie se nepřizpůsobuje velikosti obrazovky. U velkých obrázků se musí použít posuvníky či oddálit zobrazení stránky.
- Obrázky nelze v programu ořezávat či jinak upravovat, vyjma otáčení.

5.6.1 Obrázky programu



Obrázek 5.12: Web Album Generator, uživatelské rozhraní - výpis obrázků



Obrázek 5.13: Web Album Generator, uživatelské rozhraní - generování galerie, sekce pro úpravu barev výsledné galerie

5.7 Visual LightBox

Verze: 6.1

Oficiální stránky: <http://visuallightbox.com/>

Operační systém: Windows, Mac

Typ programu: Free (pro nekomerční použití)

1. Import obrázků

- Pomocí dialogového okna pro výběr souborů.
- Pomocí Drag and Drop.
- Pomocí importu ze služeb jako Flickr či Picasa.
- Možnost změnit název a přidat komentář u jednotlivých obrázků.
- Dodatečné informace (jako například datum pořízení) u obrázku nejsou importovány.

2. Výpis obrázků

- Obrázky jsou zobrazeny v mřížce s jejich názvy a náhledy.
- Změna pořadí pomocí Drag and Drop.

3. Editace obrázku

- Funkce: Otočení.

4. Přizpůsobení výsledné webové galerie

- Možnost vybrat si z již existujících šablon a stylů, jak mřížky, tak lightboxu jednotlivě.
- Možnost vybrat kvalitu výsledných náhledů obrázků.
- Možnost změnit nastavení slideshow.

5. Vytvoření galerie

- Možnost publikování galerie do CMS jako WordPress a Joomla.
- Dodatečné informace (jako například datum pořízení) u obrázku nejsou zachovány.

Hodnocení.

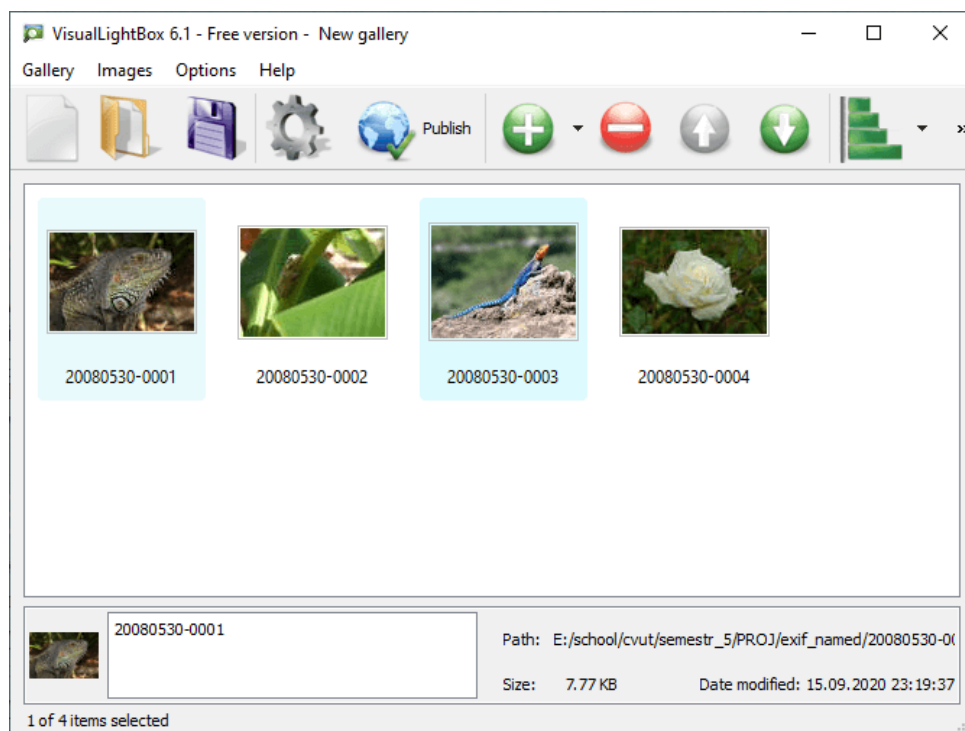
1. Co se mi líbí

- Jednoduchost programu a výsledné galerie.
- Oddělené styly pro Lightbox a mřížku galerie.
- Publikování do různých CMS, jako je WordPress či Joomla.

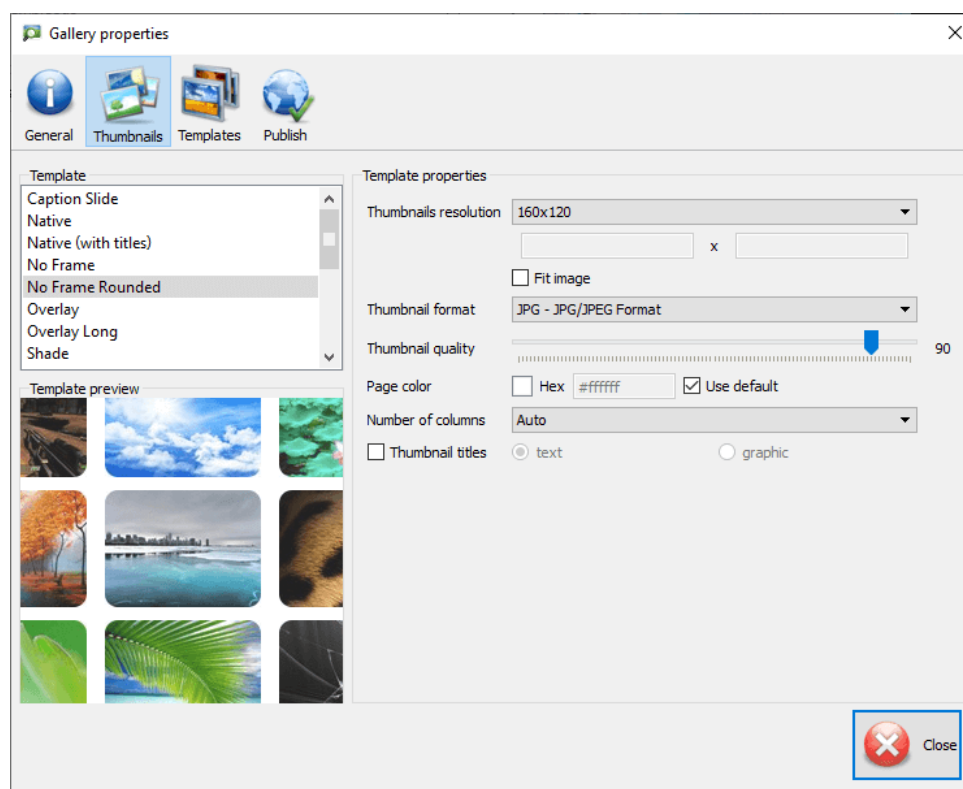
2. Co se mi nelíbí

- Obrázky nelze v programu ořezávat či jinak upravovat, vyjma otáčení.

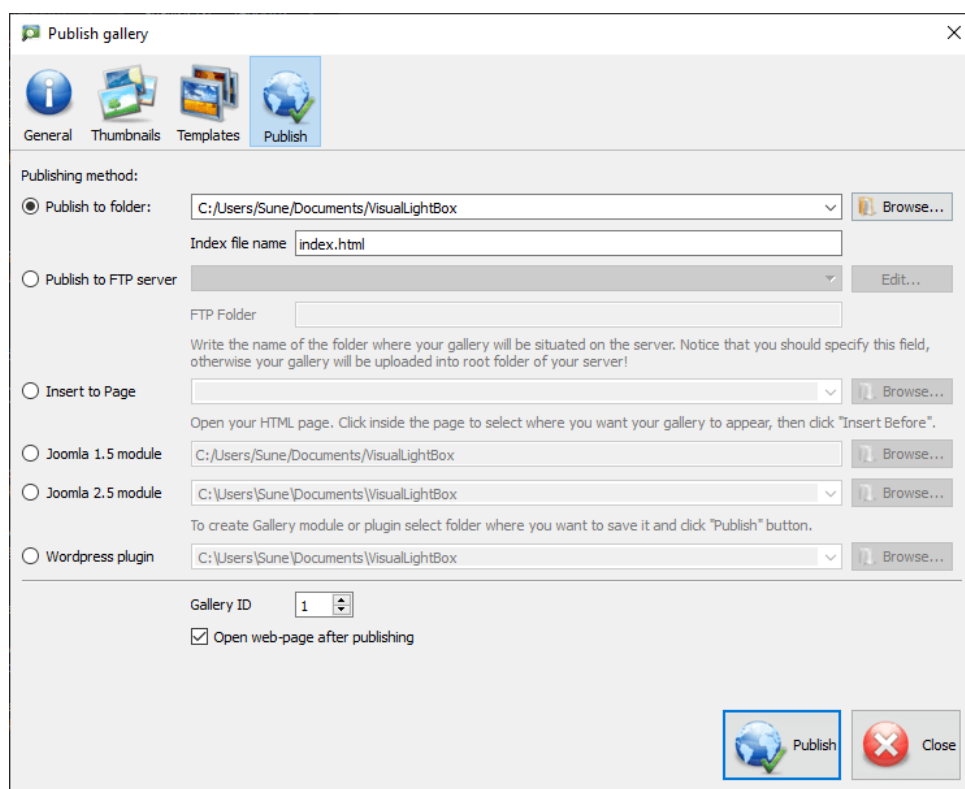
5.7.1 Obrázky programu



Obrázek 5.14: Visual LightBox, uživatelské rozhraní - výpis obrázků



Obrázek 5.15: Visual LightBox, uživatelské rozhraní - generování galerie, sekce pro úpravu vzhledu náhledových obrázků



Obrázek 5.16: Visual LightBox, uživatelské rozhraní - generování galerie, sekce pro výstup



Kapitola 6

Shrnutí analýzy programů

	jAlbum	Easy Website Photo Gallery	HTTPPhotos	LightBox Video Web Gallery Creator
Import obrázků	Pomocí dialogového okna (soubory) a drag and drop. Možnost popisů a tagů u obrázků.	Pomocí dialogového okna (soubory) a drag and drop.	Pomocí dialogového okna (soubory).	Pomocí dialogového okna (složky).
Výpis obrázků	Výpis s náhledy. Změna pořadí pomocí drag and drop. Možnost vytváření hierarchie pomocí složek.	Výpis v seznamu s názvy obrázků. Náhled obrázku v postranním panelu. Změna pořadí pomocí tlačítek.	Výpis s náhledy. Změna pořadí pomocí Drag and Drop.	Není.
Editace obrázků	Funkce: překlopení, otočení, ořez, úprava jasu, kontrastu a další. Efekty: červené oči, zaostření, rozmazání a další.	Funkce: změna jasu, kontrastu, světlost, tmavost, saturace. Efekty: rozostření, šum, odstíny šedi, inverze.	Funkce: změna jasu, kontrastu. Efekty: rozostření, zotření, stupně šedi, inverze.	Není.
Přizpůsobení výsledné galerie	Výběr stylu galerie z existujících šablon. Možnost změnit barvy, odsazení a další prvky Lightboxu. Vlastní HTML, CSS a JS. Správa SEO.	Výběr stylu galerie z existujících šablon. Možnost upravit styl a strukturu galerie pomocí HTML a CSS, texty samostatně upravitelné. Možná změna fontů.	Možnost změnit rozložení indexové stránky. Možnost změnit font, barvy pozadí a textu.	Možnost změnit velikost náhledů. Možnost změnit počet řádků a sloupců mřížky galerie. Možnost upravit kvalitu u náhledu obrázku. Možná úprava CSS.

Tabulka 6.1: Shrnutí analýzy programů, část 1.

	iWebAlbum	Web Album Generator	Visual LightBox
Import obrázků	Pomocí dialogového okna (soubory) a drag and drop. Možnost přidat komentář u obrázků.	Pomocí dialogového okna (soubory). Možnost přidat komentář u obrázků.	Pomocí dialogového okna (soubory), drag and drop a služeb jako Flickr či Picasa. Možnost přidat komentář u obrázků.
Výpis obrázků	Výpis v seznamu s názvy obrázků. Náhled obrázku v postranním panelu. Změna pořadí pomocí tlačítek.	Výpis v seznamu s názvy obrázků. Náhled obrázku v postranním panelu. Změna pořadí pomocí tlačítek.	Výpis s náhledy. Změna pořadí pomocí drag and drop.
Editace obrázků	Funkce: Otočení a překlopení.	Funkce: Otočení.	Funkce: Otočení.
Přizpůsobení výsledné galerie	Výběr stylu galerie z existujících šablon. Možná změna rozměrů obrázků. Možnost změnit počet řádků a sloupců v mřížce galerie.	Výběr stylu galerie z existujících šablon. Možnost změnit barvy textu a pozadí. Možnost změnit počet řádků a sloupců v mřížce galerie. Možná změna rozměrů obrázků.	Výběr stylu zvlášť mřížky galerie a lightboxu z existujících šablon. Možnost vybrat kvalitu výsledných náhledů obrázků. Možná změna nastavení slideshow.

Tabulka 6.2: Shrnutí analýzy programů, část 2.

Kapitola 7

Návrh implementace

Tato kapitola se věnuje návrhu implementace funkcí v budoucím programu tohoto projektu. Každá sekce se zabývá jednou požadovanou funkcí, kterou by měl program disponovat.

7.1 Import obrázků

Tato sekce se zabývá možnostmi importu obrázků do programu a následným procesem importu.

7.1.1 Možnosti importu

Import obrázků do programu by měl být pro uživatele co nejvíce intuitivní a jednoduchý, nicméně každý uživatel může mít jiné preference, proto by se mělo implementovat více možností importu. Nejvíce vhodnými jsou následující:

1. Dialog pro výběr souborů

Uživatel zvolí jeden či více souborů. Soubory pak následně projdou procesem importu. Většina operačních systémů umožňuje do dialogu přidat různé filtry, jako například *Všechny obrázky*, *JPEG/JPG soubory*, *PNG soubory* apod., které by měly být pro uživatele také dostupné.

2. Dialog pro výběr složky

Uživatel zvolí jednu složku, ze které se mají soubory vybírat. Součástí dialogu by měly být i filtry souborů zmíněné v bodě č. 1 a také možnost zvolit, zdali se mají procházet i podsložky. Všechny vybrané (a případně vyfiltrované) soubory pak projdou procesem importu.

3. Drag and Drop

Uživatel přetáhne vybrané soubory či složky do prostoru okna programu. Po přetažení a puštění souborů či složek lze ještě zobrazit okno pro dodatečnou filtraci a možnost zvolit procházení podsložek. Nicméně v analyzovaných programech, které Drag and Drop funkcionalitu podporovaly, takovéto okno nebylo implementováno, takže bych ho spíše označil za volitelnou funkcionalitu. Všechny vybrané (a případně vyfiltrované) soubory pak projdou procesem importu.

7.1.2 Proces importu

Tento proces specifikuje postup importu obrázků pomocí jedné z možností z předchozí sekce. Z té lze vyvodit dva typy importu:

1. Import souboru

Program umí importovat často používané obrázkové formáty, konkrétně minimálně *JPEG/JPG*, *TIFF* a *PNG*. Pro název obrázku v galerii se jako výchozí použije název daného souboru. Pokud je soubor typu *JPEG/JPG* či *TIFF*, pak se z obrázku ještě získají některá EXIF metadata, pokud existují. Případně lze přidat podporu i pro formát *PNG*, pokud obsahuje EXIF chunk. Tyto metadata se pak dají využít v rámci řazení obrázků podle data pořízení apod. Na celý tento postup jsou aplikované případně uživatelem zvolené filtry a pokud soubor není filtrem akceptován, program soubor přeskočí. Nepodporované formáty (v případě čtení metadat i nečitelné obrázky) program odmítne a na konci procesu nahlásí pomocí chybového okna seznam všech souborů, které se nepodařilo importovat.

2. Import složky

Program projde obsah složky a u každého souboru provede postup popsany v predešlém bodě. Pokud uživatel zvolil i procházení podsložek, provede se tento krok i pro každou podsložku v dané složce. Uživatel také může zvolit, zdali se má název složky brát jako datum pořízení všech obrázků v dané složce. Pokud je tato možnost zvolena, je pak u každého obrázku nutné ignorovat datum pořízení z případných metadat. Problémem zde ale může být, že bude ignorován i čas pořízení obrázku, v té situaci lze (a) doopravdy ignorovat čas z metadat a zvolit čas 00:00:00 jako čas pořízení, nebo (b) čas z metadat ponechat a jen změnit

samotné datum bez času. Pokud je možnost zvolení názvu složky jako data pořízení obrázků zvolena, ale název složky není validní datum či nelze rozpoznat formát data, pak se objeví chybová hláška a uživatel musí vybrat, zdali import přerušit, nebo chybu akceptuje a název složky se bude ignorovat.

7.1.3 Získávání informací ze struktury složek

Pokud bude programu při importu pomocí složky předána určitá struktura složek, lze z ní získat informace o datu pořízení jednotlivých obrázků bez toho, aniž by se musela číst jejich metadata. Pokud se v obrázku nenachází informace o datu pořízení, nejde správnost tohoto získaného data nijak ověřit. Uživateli ale lze umožnit, pokud data v obrázku neexistují, využít právě data extrahovaná ze struktury složek.

Struktura složek nicméně může být různá:

■ Plochá struktura

Jedná se o strukturu, kde je programu předána složka, ve které se nachází podsložky, jejichž název obsahuje celé datum v různých formátech, tedy například: *2020-10-14*, *14. 10. 2020* atd. Všem obrázkům z takovýchto podsložek bude přiřazeno právě toto celé datum.

■ Hierarchická struktura

Jedná se o strukturu, kde je programu předána složka, ve které se nachází hierarchie složek. A to taková, že nejdříve jsou složky s názvy roků, v nich složky jednotlivých měsíců a v nich pak složky jednotlivých dní. Je samozřejmě možné uživateli umožnit tuto hierarchii změnit třeba tak, že nejdříve je den, pak měsíc a pak rok.

Řazení obrázků v jednotlivých složkách je nicméně na uživateli. Jelikož program bude pravděpodobně soubory procházet abecedně, k jejich nahrání ve správném pořadí stačí například je očíslovat nebo u nich specifikovat čas pořízení (ve tvaru *hh-mm-ss* apod.)

7.3 Editace obrázku

Uživatel může provést editaci obrázku přímo v programu. Jelikož se nejedná o plnohodnotný obrázkový editor, program bude nabízet pouze nějaké ze základních funkcí pro editaci. Konkrétně minimálně tyto funkce: *překlopení obrázku, rotace obrázku (o 90 stupňů, jako volitelná funkcionalita i libovolný úhel), oříznutí obrázku a úprava jasu a kontrastu obrázku (jako volitelná funkcionalita i úprava saturace)*; a dále konkrétně minimálně tyto efekty: *stupně šedi, inverze, rozostření, doostření a odstranění červených očí*.

7.3.1 Uživatelské rozhraní

Uživatelské rozhraní by mělo být pro uživatele intuitivní a nemělo by být přehlcené. Pro nejjednodušší variantu editace obrázku je potřeba:

- plátno, kde se obrázek v současném stavu zobrazí,
- výběr úpravy obrázku a efektu,
- nastavení jednotlivých úprav obrázku a efektů,
- tlačítka pro potvrzení a zrušení editace.

Hlavní okno: V okně editace se nachází plátno pro zobrazení obrázku v současném stavu, které zabírá značnou část prostoru okna. Umístění výběru pro úpravu obrázku a efektu může být buď (a) lišta menu nahoře v okně s kategoriemi *Úpravy* a *Efekty*, nebo (b) panel vlevo či vpravo vedle plátna s tlačítky jednotlivých úprav a efektů. V poslední řadě se v okně nachází v dolní části dvě tlačítka, jedno pro potvrzení editace a druhé pro zrušení editace.

Okno/panel nastavení úpravy či efektu: V okně/panelu nastavení, pokud to úprava či efekt umožňuje, je nastavení síly dané úpravy či efektu (příkladem může být třeba síla rozmazání). U některých úprav či efektů (například inverze) lze umožnit změnu síly, nicméně v předešle analyzovaných programech takováto funkce není, proto je brána jako volitelná funkcionalita. Dále je nutné mít tlačítko pro potvrzení a zrušení úpravy či efektu. Veškeré změny v rozhraní (jako je změna síly) se projeví v obrázku zobrazeném na plátně. Takto uživatel

může vidět úpravy či efekty ihned a není potřeba nic potvrzovat a následně se pak zbytečně vracet o krok zpět před tím, než byla úprava či efekt na obrázek aplikován.

7.3.2 Implementace úprav obrázku a efektů

Pro úpravy obrázku a efekty lze použít (a) vlastní implementaci, nebo (b) použít již existující knihovnu. Pokud by se zvolila první varianta, bylo by vhodné i přesto použít nějakou již existující knihovnu pro čtení a ukládání různých obrázkových formátů. Obrázkové formáty jsou komplexní, využívají různé algoritmy a mají třeba i určitá rozšíření (například PNG extensions). Vlastní implementací čtení a ukládání obrázků by se mohly objevovat chyby, některé obrázky by ani nemusely být správně přečteny a hlavně by se pouze na této části programu strávila většina času. Většina knihoven je řádně otestovaná, a tudíž je odzkoušeno, že doopravdy funguje.

Po přečtení obrázku je ale stále nutné stanovit, jak se obrázky budou interně v programu reprezentovat, aby se na ně mohly aplikovat algoritmy úprav a efektů. Knihovny většinou mají více možností, jak obrázky reprezentovat - například `BYTE_3_RGB` (3 bajty pro všechny komponenty RGB), `INT_ARGB` (1 celé číslo (integer) pro všechny komponenty ARGB) a podobně. Jelikož se nejedná o plnohodnotný obrázkový editor, stačí zvolit pouze jednu interní reprezentaci. Proto je nutné přečtené obrázky převést do jednotného formátu. Tím, že program umožňuje pracovat s obrázky PNG a ty podporují průhlednost, je nutné umožnit pracovat s průhledností každého pixelu. Proto bych osobně volil `INT_ARGB`, kde je pro každý pixel použito jedno celé číslo (integer), možností je ale třeba také `BYTE_4_ARGB`, kde pro každý pixel jsou použity 4 bajty. Dále by jednotlivé pixely byly v paměti reprezentovány po řádcích.

Po stanovení jednotného formátu v předchozím odstavci, v tomto případně konkrétně `INT_ARGB`, by se jednotlivé úpravy a efekty daly (naivně) implementovat následovně:

■ Překlopení obrázku horizontálně

Pro každý řádek obrázku uděláme následující:

Pro $0 \leq i < (w - 1)/2$ vezmeme dva pixely, jeden na pozici i a druhý na pozici $w - i - 1$, kde w je šířka obrázku, a tyto pixely prohodíme.

■ Překlopení obrázku vertikálně

Pro každý sloupec obrázku uděláme následující:

Pro $0 \leq i < (h - 1)/2$ vezmeme dva pixely, jeden na pozici i a druhý na pozici $h - i - 1$, kde h je výška obrázku, a tyto pixely prohodíme.

■ Otočení obrázku (o 90 stupňů)

Vytvoříme si nové pole o velikosti $w * h$, kde w je šířka obrázku a h je výška obrázku. Pro každý pixel na pozici $[x, y]$ (indexově $y * w + x$) aplikujeme rotaci o 90 stupňů, tedy $R_{\theta=90}([x, y]) = [-y, x]$, a uložíme ho na výslednou pozici (indexově $x * h - y$). Nové pole pak reprezentuje otočený obrázek s šířkou h a výškou w .

■ Oříznutí obrázku (obdélníkové oříznutí)

Vytvoříme si nové pole o velikosti $w_c * h_c$, kde $0 < w_c \leq w$ je šířka ořezu, $0 < h_c \leq h$ je výška ořezu, w je původní šířka obrázku a h je původní výška obrázku. Pak každý pixel $[x, y]$ v oblasti $x_c \leq x < x_c + w_c$, $y_c \leq y < y_c + h_c$, kde bod $[x_c, y_c]$, $0 \leq x_c < w - 1$, $0 \leq y_c < h - 1$, je počáteční bod ořezu, uložíme do nového pole na pozici $[x - x_c, y - y_c]$ (indexově $(y - y_c) * w + (x - x_c)$).

■ Úprava jasu¹¹

Mějme hodnotu $v \in [-255, 255]$ reprezentující sílu úpravy jasu. Pro každý pixel obrázku pak uděláme následující:

```
r := clamp(((p >> 16) & 0xff) + v, 0, 255),
g := clamp(((p >> 8) & 0xff) + v, 0, 255),
b := clamp((p & 0xff) + v, 0, 255),
p' := (p & 0xff000000) | (r << 16) | (g << 8) | b,
```

kde p je hodnota pixelu, r je červená složka pixelu, g je zelená složka pixelu, b je modrá složka pixelu, `clamp` je pomocná funkce pro omezení hodnoty (mezi hodnotu minima a hodnotu maxima), `>>` je binární posun doprava, `<<` je binární posun doleva, `&` je binární AND, `|` je binární OR, hodnoty začínající `0x` jsou celá čísla reprezentovaná v hexadecimální soustavě a p' je výsledná hodnota pixelu.

■ Úprava kontrastu¹²

Mějme hodnotu $c \in [-255, 255]$, pak $v = 259 * (c + 255) / (255 * (259 - c))$ reprezentující sílu úpravy kontrastu. Pro každý pixel obrázku pak uděláme následující:

```
r := clamp(v * (((p >> 16) & 0xff) - 128) + 128, 0, 255),
g := clamp(v * (((p >> 8) & 0xff) - 128) + 128, 0, 255),
b := clamp(v * ((p & 0xff) - 128) + 128, 0, 255),
p' := (p & 0xff000000) | (r << 16) | (g << 8) | b,
```

kde p je hodnota pixelu, r je červená složka pixelu, g je zelená složka pixelu, b je modrá složka pixelu, `clamp` je pomocná funkce pro omezení hodnoty, `>>` je binární posun doprava, `<<` je binární posun doleva, `&` je binární AND, `|` je binární OR, hodnoty začínající `0x` jsou celá čísla reprezentovaná v hexadecimální soustavě a p' je výsledná hodnota pixelu.

■ Stupně šedi¹³

Pro každý pixel obrázku pak uděláme následující:

■ Doostření¹⁶

Poznámka: V této naivní implementaci nejsou ošetřeny okraje obrázku. Mějme hodnotu $v \in \mathbb{N}$ reprezentující sílu doostření. Vytvoříme si nové pole o velikost $w * h$, kde w je šířka obrázku a h je výška obrázku. Pak pro každý pixel provedeme stejné rozostření jako v předchozím bodě akorát s tím rozdílem, že výsledná hodnota pixelu počítána takto:

```
r := (p >> 16) & 0xff),
g := (p >> 8) & 0xff),
b := (p & 0xff),
p' := (p & 0xff000000)
      | ((r + (r - (acc_r / ctr)) * 0.5 * v) << 16)
      | ((g + (g - (acc_g / ctr)) * 0.5 * v) << 8)
      | (b + (b - (acc_b / ctr)) * 0.5 * v)),
```

kde p je hodnota pixelu, r je červená složka pixelu, g je zelená složka pixelu, b je modrá složka pixelu, $>>$ je binární posun doprava, $<<$ je binární posun doleva, $&$ je binární AND, $|$ je binární OR, hodnoty začínající $0x$ jsou celá čísla reprezentovaná v hexadecimální soustavě, p' je výsledná hodnota pixelu a proměnné acc_r , acc_g , acc_b , ctr jsou stejné proměnné jako u rozmazání.

■ Odstranění červených očí

U této funkce je nutné nějakým způsobem nejdříve získat oblast, na kterou se algoritmus pro odstranění červených očí má aplikovat. Jelikož program nemá být plnohodnotný obrázkový editor a nemá být v tomto směru komplexní, zvolíme jednodušší metodu, tj. necháme uživatele provést výběr oblasti, kde se červené oči nacházejí. Uživatel tedy nejdříve vybere v obrázku obdélníkovou oblast $([x, y], w, h)$, kde $[x, y]$ je počáteční bod oblasti, w je šířka oblasti a h je výška oblasti. Jelikož je oční zornice spíše kulatého tvaru, následně se pro takovou oblast zvolí střed $C = [x + w/2, y + h/2]$ a poloměr $r = (max(w, h))/2$, kde max je pomocná funkce, která ze všech svých vstupních hodnot zvolí tu největší. Pro každý pixel v oblasti (C, r) pak vypočítáme jeho výslednou hodnotu takto:¹⁷

```
m := ((p >> 8) & 0xff) + (p & 0xff)) / 2,
p' := (p & 0xff000000) | (m << 16) | (m << 8) | m,
```

kde p je hodnota pixelu, $>>$ je binární posun doprava, $<<$ je binární posun doleva, $&$ je binární AND, $|$ je binární OR, hodnoty začínající $0x$ jsou celá čísla reprezentovaná v hexadecimální soustavě a p' je výsledná hodnota pixelu.

7.4 Přizpůsobení výsledné webové galerie

Uživatel by měl mít možnost upravit všechny barvy a texty ve výsledné webové galerii. Pro pokročilejší uživatele by měl program nabízet i možnost upravit CSS a JS galerie. Samotná struktura galerie - počet řádků a sloupců mřížky, pozice navigace a podobně - by také měla jít upravit, alespoň v nějaké omezené míře. Stejně jako v předešle analyzovaných programech by měly být dostupné nějaké předpřipravené styly, které uživatel může využít nebo z nich při dalších následujících úpravách může vycházet. Všechny tyto požadavky řeší program pomocí šablon.

7.4.1 Šablona

Šablona je složka obsahující veškeré soubory potřebné pro generování galerie. Pro ukládání informací o samotné šabloně, jako je název, verze, autor a podobně, je v šabloně použit soubor `info.txt`. Dále zde existují soubory, které obsahují strukturu stránek galerie: `index.html`, `list.html`, `single.html` atd. V těchto souborech jsou pak určité speciální textové vzory, které do těchto souborů umožňují dynamicky vkládat data, jako je například název obrázku a podobně.

7.4.2 `info.txt`

Tento soubor obsahuje informace (metadata) o samotné šabloně. Program je umí z tohoto souboru přečíst a ty pak zobrazit uživateli (třeba název šablony, který se objeví ve výběru šablony v programu). Na každém řádku je uvedena jedna informace, kde název je první, pak následuje dvojtečka a pak je hodnota. Takový soubor pak může vypadat třeba takto:

Name: Název šablony

Version: 1.0

Author: Jan Novák

Description: Jednoduchá šablona, podporuje Lightbox.

7.4.3 Formát textových vzorů pro dynamická data

V šablonových souborech (template files) lze použít textové vzory, se kterými je možné do takovýchto souborů vkládat dynamicky data. Textový vzor má tento formát: `{{ data_name }}`, kde `data_name` je název informace, která se má na to dané místo vložit. Příkladem pak třeba může být:

```
<p>Tato galerie používá šablonu {{ template.name }}</p>
```

7.4.4 Šablonové soubory

V šabloně mohou existovat určité šablonové soubory, které program využije, pokud jsou dostupné. Pokud dostupné nejsou, využijí se výchozí. Takové soubory jsou:

- **index.html**
Šablona pro index galerie, tedy taková domovská stránka galerie. Na této stránce se většinou vypisují informace o galerii a seznam složek a obrázků v první úrovni hierarchie.
- **directory.html**
Šablona pro složku v galerii. Tento soubor se použije, když uživatel přejde do složky, například kliknutím na odkaz z indexu galerie.
- **single.html**
Šablona pro stránku se samotným obrázkem. Tento soubor se použije, když uživatel klikne na odkaz nějakého obrázku ve výpisu obrázků.
- **header.html**
Šablona pro záhlaví stránek. Zobrazuje se na každé stránce, kde se nachází textový vzor `{{ header }}`.
- **footer.html**
Šablona pro zápatí stránek. Zobrazuje se na každé stránce, kde se nachází textový vzor `{{ footer }}`.
- **navigation.html**
Šablona pro navigaci galerie. Zobrazuje se na každé stránce, kde se nachází textový vzor `{{ navigation }}`.
- **list.html**
Šablona pro výpis obrázků na dané stránce. Zobrazuje se na každé stránce, kde se nachází textový vzor `{{ list }}`. Pro jednotlivé položky v seznamu využívá šablonový soubor `list-item.html`.

- `list-item.html`

Šablona pro položku ve výpisu obrázků a složek.

- `pagination.html`

Šablona pro stránkování galerie. Zobrazuje se na každé stránce, kde se nachází textový vzor `{{ pagination }}`.

■ 7.4.5 CSS galerie

Hlavním souborem CSS galerie je ve výchozím nastavení soubor `style.css`. Toto lze upravit změnou souboru `header.html`. Nachází se zde všechny styly, barvy apod.

■ 7.4.6 Skripty (JS) galerie

Hlavním souborem JS galerie je ve výchozím nastavení soubor `script.js`, pokud existuje. Toto lze upravit změnou souboru `header.html`. Nachází se zde případná JS funkcionality, například pro Lightbox.

■ 7.4.7 Uživatelské rozhraní

Vydeme-li z předešle navržených funkcí, v okně přizpůsobení galerie se nachází:

- pole pro vyplnění názvu galerie,
- výběr stylu galerie a případná správa stylů (jelikož ve správě stylů mají jít přidávat a odebírat styly a nejedná se přímo o přizpůsobení webové galerie, volil bych umístění této správy do samostatného okna),
- pole pro zadání počtu řádků a sloupců mřížky galerie,
- výběr pozice navigace,
- správa barev (jelikož bude většinou obsahovat vícero polí, volil bych umístění do samostatného okna)

- správa textů (jelikož bude většinou obsahovat vícero polí, volil bych umístění do samostatného okna)
- pole pro vlastní CSS,
- pole pro vlastní JavaScript.

Některá pole v tomto okně nemusí nutně u některých stylů být použita, a proto je tedy na dalším zvážení, zdali by se vůbec měla takováto pole zobrazovat v tomto okně, nebo zdali raději v nějakém jiném - například v okně pro nastavení stylu.

7.5 Vytvoření galerie

Program vezme uživatelem vybranou šablonu a veškeré informace o galerii, jako je seznam obrázků atd. a vygeneruje webovou galerii pomocí procesu vytvoření.

7.5.1 Proces vytvoření

Program při vytváření prochází následujícím procesem:

- Vytvoří se soubor `index.html` a vloží se do něj dynamická data, tedy šablonové soubory jako `header.html`, `footer.html` apod. Tím se vygeneruje statický HTML kód pro index galerie.
- Po jednom se pak procházejí obrázky (a složky) ze seznamu, který uživatel importoval do programu, a pro každý z nich se vygeneruje statický HTML soubor pomocí šablonových souborů. Každý obrázek se také zkopíruje do připravené podsložky ve složce `images` odpovídající jeho hierarchii v programu a vygeneruje se i náhledový obrázek, který se pak zobrazuje ve výpisu obrázků.
- Z dynamických dat se vloží barvy do CSS šablony, přidá se uživatelský CSS kód, pokud byl vyplněn, a vygeneruje se soubor `style.css`.
- Přidá se uživatelský JS kód, pokud byl vyplněn, a vygeneruje se soubor `script.js`.

Po tomto procesu je galerie připravená k použití. Jelikož se jedná o statické HTML soubory, lze ji přenášet, archivovat apod.

7.6 Okno pro správu stylů

V okně se nachází výpis všech existujících a programem úspěšně přečtených stylů (případně i oddělený seznam chybných či neúspěšně přečtených stylů s dodatečnou chybovou hláškou), tlačítko pro odebrání stylu a pro přidání stylu, u kterého se po kliknutí na něj zobrazí dialogové okno pro výběr složky (nebo například i pro výběr ZIP souboru) se soubory stylu, který uživatel chce přidat.

7.7 Okno pro správu barev

V okně se nachází výpis všech názvů barev a jejich příslušných hodnot, které lze měnit. V dolní části se pak ještě nachází tlačítka pro zrušení a potvrzení změn.

7.8 Okno pro správu textů

V okně se nachází výpis všech názvů textů a jejich příslušných hodnot, které lze měnit. V dolní části se pak ještě nachází tlačítka pro zrušení a potvrzení změn.

7.9 Okno vygenerování galerie

Pro vygenerování galerie je nutná výstupní cesta, tedy cesta, kam se uloží soubory vygenerované galerie, vybraná šablona galerie a seznam obrázků. Styl galerie a seznam obrázků uživatel zvolil v okně přizpůsobení galerie a okně s výpisem obrázků, a tudíž zbývá zvolit akorát výstupní cestu. V okně

vygenerování galerie by se tedy mělo nacházet pole pro výstupní cestu a pro uživatelskou přívětivost také i tlačítko pro otevření dialogu pro výběr složky, aby uživatel nemusel případně vypisovat celou cestu ručně. Dále také tlačítko pro zrušení a tlačítko pro vygenerování galerie, u kterého se po kliknutí na něj spustí proces vytvoření galerie.



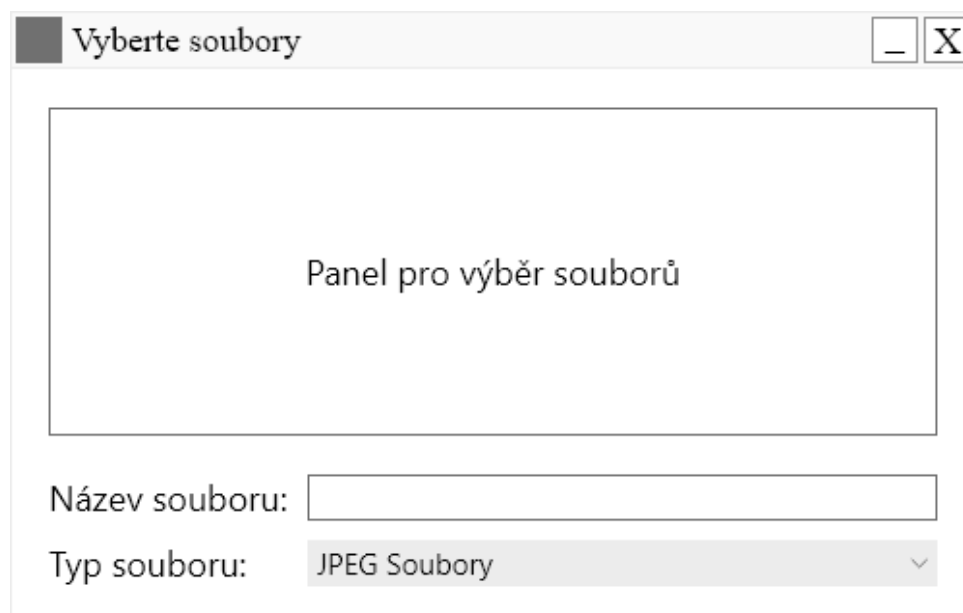
Kapitola 8

Grafický návrh programu

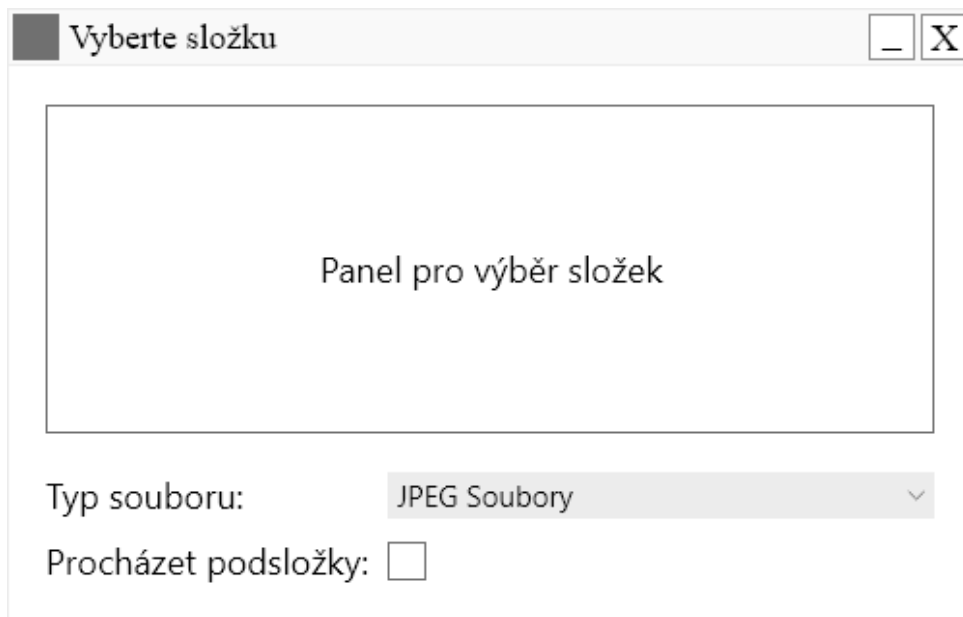
V této kapitole se nacházejí grafické návrhy pro jednotlivá okna programu. Jedná se především o přetvoření textového popisu v předchozí kapitole do nějaké vizuální podoby. Každý z návrhů bere na vědomí jednoduchost, přehlednost a srozumitelnost grafického rozhraní a využívá často již existující prvky nacházející se v různých grafických knihovnách.



Obrázek 8.1: Hlavní okno programu s výpisem obrázků



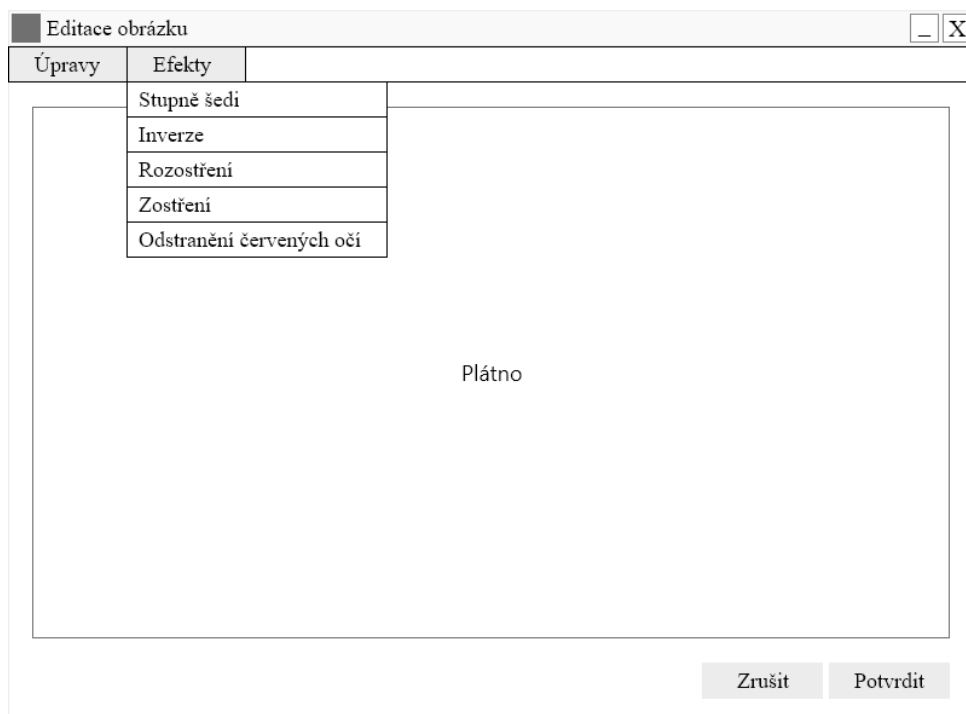
Obrázek 8.2: Import obrázků - dialogové okno pro výběr souborů



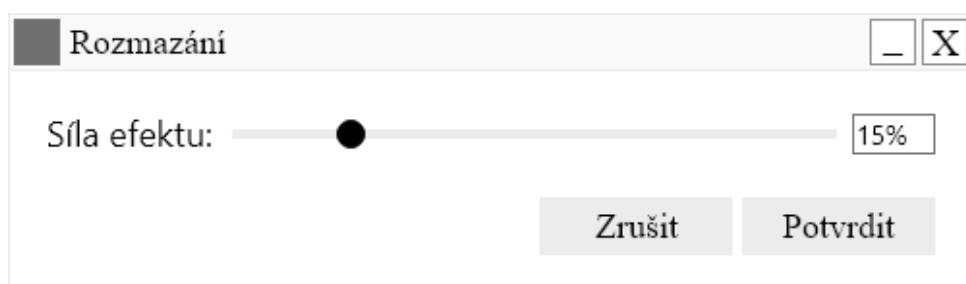
Obrázek 8.3: Import obrázků - dialogové okno pro výběr složky



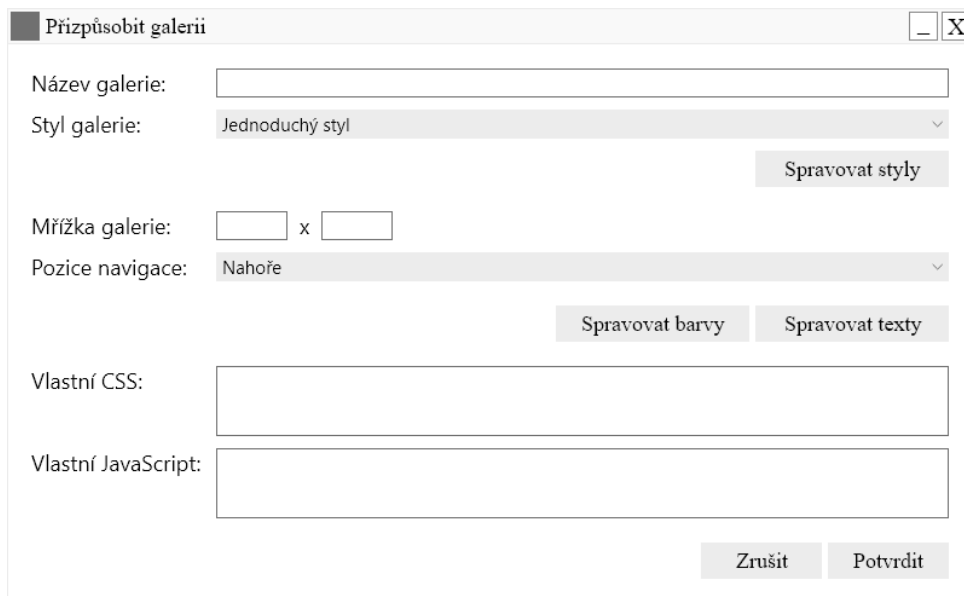
Obrázek 8.4: Okno pro editaci obrázku



Obrázek 8.5: Okno pro editaci obrázku, s otevřeným menu



Obrázek 8.6: Editace obrázku - dialogové okno nastavení efektu



The dialog box titled "Prizpusobit galerii" (Customize gallery) contains the following fields and controls:

- Název galerie:** A text input field.
- Styl galerie:** A dropdown menu currently showing "Jednoduchý styl". A "Spravovat styly" button is located to the right.
- Mřížka galerie:** Two input fields for grid dimensions, separated by an "x" character.
- Pozice navigace:** A dropdown menu currently showing "Nahoře". "Spravovat barvy" and "Spravovat texty" buttons are located to the right.
- Vlastní CSS:** A large text area for custom CSS code.
- Vlastní JavaScript:** A large text area for custom JavaScript code.
- Buttons: "Zrušit" (Cancel) and "Potvrdit" (OK) are located at the bottom right.

Obrázek 8.7: Okno přizpůsobení galerie



The dialog box titled "Vygenerovat galerii" (Generate gallery) contains the following fields and controls:

- Výstupní cesta:** A text input field with a browse button ("...") to its right.
- Buttons: "Zrušit" (Cancel) and "Vygenerovat" (Generate) are located at the bottom right.

Obrázek 8.8: Okno vygenerování galerie

Kapitola 9

Implementace programu

Tato kapitola se zabývá implementací samotného programu. Implementace vychází primárně ze zadání a z návrhu implementace, která byla podrobně popsána v předchozí kapitole. Program je implementován v programovacím jazyce C++ a je využíván standard C++17. Grafické rozhraní programu je pak vytvořeno pomocí populárního aplikačního rámce Qt, konkrétně verze 5 (Qt5). Program nepoužívá žádné další knihovny třetích stran, nicméně využívá implementaci grafického rozložení prvků `FlowLayout`¹⁸, která byla publikována na oficiální doméně firmy Qt.

9.1 Systém projektů

Hlavní částí programu jsou takzvané projekty. Když uživatel poprvé spustí program, zobrazí se mu uvítací okno, kde má na výběr vytvořit nový projekt či již existující otevřít. Pokud neexistuje projekt, nemá program s čím pracovat.

Každý projekt má jen pár atributů - cestu k souboru, kde je projekt uložen a ukazatel na kořenovou složku celého stromu hierarchie položek galerie. V programu je implementováno i automatické ukládání, které probíhá každých 5 minut. Pokud uživatel projekt ještě manuálně a explicitně neuložil do nějakého souboru, vytvoří program dočasný soubor a ukládá data projektu do něj. Jakmile uživatel projekt explicitně uloží do souboru, je projekt ukládán do vybraného souboru. Projekt lze v programu kdykoli uložit, a to buď do stejného souboru nebo lze soubor kdykoli změnit.

9.1.1 Formát projektového souboru

Program využívá pro projektové soubory vlastní formát, pojmenován jednoduše jako Web Gallery Project formát, zkráceně **WGPF**, a soubory s tímto formátem mají příponu **.wgp**. Formát je založen na značení Extensible Markup Language (XML) a data jsou zapsané na jednotlivých řádcích, s případným tzv. escapováním za pomoci zakódování do vlastních znakových entit (např. `&o;`).

První řádka souboru označuje formát (**WGP**) a použitou verzi formátu (např. 1), oddělené mezerou. Tedy první řádek může vypadat třeba takto: **WGP 1**. V současné době existuje pouze první verze tohoto formátu, takže by všechny projektové soubory měly začínat přesně takovouto řádkou.

Následuje pak hierarchie prvků. Prvky jsou označeny tzv. tagy a prvek může být samouzavíratelný (`<NO_EDIT/>`) či uzavřený (`<ITEM>...<ITEM>`). Název prvku je pak právě použit v těchto značkách a každý prvek má v souboru svůj význam. Měl by se tedy vyskytovat jen na určených místech. Prvek může obsahovat další prvky nebo dodatečná data, která se označují jednoduše vypsáním názvu údaje a případně jeho hodnoty, jež je oddělena od názvu dvojtečkou.

Zde je seznam názvů prvků, jejich typ a význam:

- **GALLERY** (uzavřený)
Představuje obsah galerie a její strukturu. Nachází se v něm prvky **ITEM**. Program používá tento prvek jako vstupní bod galerie.
- **ITEMS** (uzavřený)
Představuje obsah dané položky. Používá se u složek k definici jejich obsahu. Nachází se v něm prvky **ITEM**.
- **ITEM** (uzavřený)
Představuje položku galerie. Obsahuje údaj o typu: **FILE** - pro soubory, **DIRECTORY** - pro složky. Také obsahuje prvek **DATA**.
- **DATA** (uzavřený)
Definuje data položky. Používá se v prvku **ITEM**. U složky se používají data **NAME** (název složky v galerii) a **THUMBNAIL** (náhledový obrázek složky; pokud je prázdný, použije se výchozí). U souborů se používají data **PATH** (cesta k původnímu souboru), **NAME** (název souboru v galerii), **DATE** (získané datum pořízení při importu) a **ANNOTATION** (anotace souboru).

- **EDIT** (uzavřený)

Představuje seznam operací, které byly použity při úpravě obrázku v editoru obrázků. Zapisují se v následujícím formátu `název operace|seznam argumentů`, kde v seznamu argumentů jsou argumenty oddělené čárkou a každý z nich má svůj typ (momentálně pouze `int`) a hodnotu, oddělené dvojtečkou (např. `int:10`). Momentálně jsou podporovány tyto názvy operací: `flip_horizontal`, `flip_vertical`, `rotate_right`, `rotate_left`, `brightness` (hodnota - `int`), `contrast` (hodnota - `int`), `grayscale`, `invert`, `blur` (síla - `int`), `sharpen` (síla - `int`), `crop` (`x - int`, `y - int`, šířka - `int`, výška - `int`), `remove_red_eyes` (`x - int`, `y - int`, poloměr - `int`).

- **NO_EDIT** (samouzavíratelný)

Představuje to, že u souboru nebyly provedeny žádné změny v editoru obrázků.

■ 9.1.2 Čtení projektového souboru

Program soubor čte po řádcích, nejdříve přečte hlavičku souboru, zjistí zapsanou verzi a porovná ji, zdali je podporována. Pokud podporována není, vyhodí program chybu. Jinak pokračuje ve čtení souboru, kde program očekává prvek `GALLERY` a v něm prvky `ITEM`. Následně čte jednotlivé položky a zpracovává jejich data a případně další obsah či úpravy. Program soubory rovnou importuje (a upravuje) a načítá je do struktury, to znamená, že pokud některý soubor nelze importovat, program vyhodí chybu. Po přečtení souborů program vygeneruje náhled kořenové složky a ten se nakonec zobrazí uživateli.

■ 9.1.3 Ukládání projektového souboru

Ukládání funguje stejně jako čtení jen z opačného směru. Program nejdříve zapíše hlavičku s poslední verzí, kterou podporuje. Následně vypíše prvek `GALLERY` a začne procházet strukturou galerie a postupně do souboru zapisuje jednotlivé položky, jejich data a případně jejich obsah, či provedené úpravy v editoru obrázků.

■ 9.2 Import souborů

Import obrázků je v programu možný dvěma způsoby - buď lze importovat jednotlivé obrázky výběrem souborů z dialogového okna pro výběr souborů nebo lze importovat celou složku pomocí dialogového okna pro výběr složky. Oba tyto způsoby lze nalézt v podnabídce *Import* v hlavní nabídce, která je umístěna nahoře v okně programu.

■ 9.2.1 Výběr souborů

Zvolením položky *File(s)* v podnabídce *Import* se otevře dialogové okno pro výběr souboru. Uživatel pak potvrzením výběru jednoho či více souborů importuje dané soubory. Pro každý vybraný soubor proběhne proces importování souboru.

■ 9.2.2 Výber složky

Zvolením položky *Directory* v podnabídce *Import* se otevře okno pro import složky. V daném okně se nachází pole pro vložení cesty ke složce a skupina polí pro dodatečnou filtraci. Filtrovat lze soubory podle data a to v rozmezí OD - DO, pro které se nachází v okně příslušná pole. Pro pole cesty ke složce je v okně umístěno tlačítko, po kliknutí na něj se otevře dialogové okno pro výběr složky. Uživatel pak po kliknutí na tlačítko *Import* importuje složku s případným dodatečným filtrem. Pro vybranou složku proběhne proces importování složky.

■ 9.3 Proces importování

Po vybrání souboru/ů či složky uživatelem se tyto dané položky předají třídě *Importer*, která se stará o import souborů a složek. Ta se tyto položky pokusí importovat a extrahovat z nich případná metadata. Takto je odděleno vybírání souborů a složek a samotná logika importování.

■ Proces importování souboru

Importování souboru probíhá v několika fázích:

- V první fázi se program pokusí otevřít soubor. Pokud soubor neexistuje či nastala jiná chyba, program soubor přeskočí a zobrazí chybu. V této fázi se také kontroluje, zdali má soubor správnou příponu, tímto se urychlí proces importování, jelikož se ignorují pravděpodobné neobrázkové soubory.
- V druhé fázi se program pokusí přečíst metadata souboru (toto je dále popsáno v sekci Extrakce metadat ze souboru). Pokud nelze metadata přečíst či extrahovat, pak program podporovaná metadata vyplní jako prázdná. Ještě se pak nastaví název obrázku na stejnou hodnotu jako je název souboru. V této fázi se také případně kontroluje, zdali získané datum a čas jsou v rozmezí specifikované ve filtru souborů. Pokud nejsou, soubor se přeskočí, jinak program pokračuje v procesu importování souboru.
- Ve třetí fázi se program pokusí pomocí knihovny *Qt* přečíst samotná data obrázku. Pokud data nelze přečíst, obrázek je přeskočen a program zobrazí chybu. Pro reprezentaci obrázku napříč programem se používá celočíselný 32-bitový formát, kde každých 8 bitů postupně reprezentuje kanály: alfa (průhlednost), červený, zelený a modrý.
- Ve čtvrté fázi se pouze uzavře daný soubor. Pokud nenastala žádná chyba v předchozích fázích, program úspěšně importoval daný obrázek.

Těmito fázemi se do programu naimportují veškerá potřebná data - metadata obrázku, tedy především datum pořízení obrázku, a samotná obrázková data.

■ Proces importování složky

Pro importování složky program nejprve rekurzivně prochází všechny položky složky, tedy nejdříve prochází soubory a složky, které jsou přímými potomky. Jedná-li se o soubor, je daný soubor importován podle procesu importování souboru. Jedná-li se o složku, program pro tuto složku rekurzivně provede proces importování složky. Tímto způsobem se naimportuje celá struktura vybrané složky včetně všech souborů, které lze importovat. Pro každou složku se ve struktuře galerie vytvoří její reprezentativní složka.

■ Extrakce metadat ze souboru

Program podporuje extrakci metadat ze souboru obrázku typu *JPEG/EXIF* a *TIFF*. Konkrétně se jedná především o datum pořízení obrázku, které pak lze zobrazit i ve výsledné webové galerii. Extrakce pro soubory typu *JPEG/EXIF* a *TIFF* je avšak navzájem odlišná.

Pro *JPEG/EXIF* soubory program nejdříve přečte značku *Start of Image* (SOI) a určí pořadí bajtů (endianitu). Následně prochází sekce označené značkou `0xFFxx`, kde `xx` je specifické číslo označující typ značky, dokud nenarazí na sekci **APP1**, jež obsahuje hledaná EXIF metadata. Obsah této sekce je pak strukturován stejně jako u souboru *TIFF*, proto program následně využije stejné procedury jako pro čtení tohoto typu souborů.

Pro *TIFF* soubory program nejdříve přečte hlavičku souboru - pořadí bajtů (endianitu), značku, že se jedná o soubor typu *TIFF*, a pozici (tzv. offset) prvního adresáře (tzv. Image file directory), kde se nachází záznamy (tzv. entries) dat. Pak prochází záznamy těchto adresářů tak dlouho, dokud buď již žádné další adresáře nejsou, nebo nenarazil na adresář se záznamem s označením `0x8769`, který označuje záznam s pozicí EXIF metadat. Po přesunu na danou pozici, program zpracovává EXIF záznamy. Jakmile narazí na záznam se značkou `0x9003` a typem 2, tj. textový řetězec v kódování ASCII (American Standard Code for Information Interchange), tak tento záznam zpracuje, jelikož se jedná o záznam s datem pořízení obrázku (`dateTimeOriginal`). Jelikož program momentálně neextrahuje žádná další metadata, po extrakci hodnoty tohoto záznamu, čtení metadat zastaví a vrátí nashromážděná data.

■ 9.4 Výpis položek galerie

Výpis položek galerie se nachází v hlavním okně programu, které je otevřeno jako první. Nachází se v něm informace o tom, jaká složka galerie se právě vypisuje, a tlačítko pro vstup do nadřazené složky. To funguje pouze v případě, že se právě nezobrazuje kořenová složka. Nachází se tam také tlačítko pro vytvoření nové složky, které otevře dialogové okno s polem pro název složky a jeho potvrzením se nová složka vytvoří. Pokud uživatel žádný název nezadal, zvolí se výchozí (*New directory*) a pokud již existuje položka s tímto názvem, přidá se k názvu číslo (*New directory (1)* atd.). Dále se tam nachází skupina tlačítek pro práci s označenými položkami:

- Delete - Odebere vybrané položky ze složky.
- Exclude - Vyřadí vybrané položky z generování galerie, toto zanechá vybrané položky v seznamu.
- Include - Opět zařadí vybrané položky do generování galerie.
- Move up - Přesune vybrané položky do nadřazené složky, toto funguje pouze v případě, že se momentálně nevypisuje kořenová složka.
- < - Posune vybrané položky nahoru v seznamu položek, tj. blíže začátku seznamu.
- > - Posune vybrané položky dolů v seznamu položek, tj. blíže konci seznamu.

Pod skupinou tlačítek se nachází samotný výpis položek. Program podporuje dva typy zobrazení:

- Grid (mřížka) - Položky se zobrazují v mřížce, tedy jedna vedle druhé a pokud se již na řádek nevejde, zobrazí se v prvním sloupci na dalším řádku. Pro každou položku se zobrazuje náhledový obrázek a pod ním název položky.
- List (seznam) - Položky se zobrazují pod sebou a každá položka zabírá celou šířku okna. Pro každou položku se zobrazuje náhled a název a speciálně pro soubory (obrázky) se zobrazují také dodatečné informace - extrahované datum a anotace.

S každou položkou ve výpisu lze různě interagovat. Pro soubory se dvojklikem daný obrázek otevře v editoru obrázků a pravým kliknutím se zobrazí okno s informacemi o souboru, kde je možné je upravit a následně potvrdit. Pro složky se dvojklikem složka otevře a ve výpisu se zobrazí její obsah, pravým kliknutím se pak otevře okno s informacemi o složce, kde je možné upravit její název a náhledový obrázek, tyto změny pak lze následně potvrdit či zrušit.

Pod výpisem se nachází spodní panel, který obsahuje tlačítka pro přepínání typu zobrazení a také posuvník pro změnu velikosti náhledových obrázků ve výpisu. K dispozici jsou momentálně čtyři velikosti - malá (small), normální (normal), velká (large), velmi velká (very large). Změna typu zobrazení i změna velikosti náhledových obrázků se dynamicky projevuje ve výpisu.

9.5 Editace obrázků

V programu se nachází i integrovaný obrázkový editor, kde je uživateli umožněno upravit importované obrázky. Nejedná se o plnohodnotný obrázkový editor, a tudíž jsou k dispozici pouze základní funkce pro úpravu obrázků. Obrázkový editor lze otevřít dvojklikem na soubor ve výpisu položek.

V okně editoru obrázků se nachází lišta menu, ve které jsou k dispozici všechny funkce, které uživatel může s obrázkem provést, včetně možnosti úpravy uložit (*File > Save*) a tím změnit obrázek, který se ve vygenerované galerii použije, či editor pouze zavřít (*File > Cancel*) a provedené úpravy tak zahodit. Momentálně podporované funkce pro úpravu obrázku jsou:

- **Operace** - horizontální překlopení, vertikální překlopení, otočení vlevo, otočení vpravo.
- **Úpravy** - změna jasu, změna kontrastu, odstíny šedi, invertování barev.
- **Filtry** - rozmazání, doostření.
- **Nástroje** - obdélníkové oříznutí, odstranění červených očí.

Obdélníkové oříznutí. Obdélníkové oříznutí je jedna ze dvou funkcí, která vyžaduje větší vstup od uživatele než jen jednu hodnotu. V editoru je tato funkce řešena tak, že po výběru nástroje oříznutí se uživatel přepne do módu oříznutí a po kliknutí na obrázek a tažením myši se vytvoří obdélníková oblast, jež má být novým obrázkem. Tato oblast je v okně zobrazena jako červený obdélník.

Odstranění červených očí. Podobně jako je řešeno obdélníkové oříznutí, je řešeno i odstranění červených očí. Nicméně jelikož je tvar zornice oka spíše kruhový, místo obdélníkové oblasti se vezme místo prvního kliknutí myši jako střed kružnice a druhý bod jako poloměr dané kružnice. Tažením myši druhým bodem lze pak kružnici zvětšovat a zmenšovat. Kliknutím na jinou pozici lze změnit střed kružnice a začít tak znovu. Oblast, která bude upravena, je pak zobrazena v okně jako modrá kružnice.

Pod lištou menu se také nachází panel pro zobrazení obrázku, kde jsou tlačítka pro úpravu přiblížení zobrazeného obrázku, konkrétně + a – pro přiblížení a oddálení, *Reset* pro resetování přiblížení na 100%, *Fit to width*

pro přizpůsobení přiblížení na šířku, *Fit to height* pro přizpůsobení přiblížení na výšku a *Fit to size* pro přizpůsobení přiblížení na šířku i výšku, tím se obrázek zobrazí celý.

V poslední řadě se pod lištou menu nachází panel zobrazení obrázku, kde se zobrazuje otevřený obrázek v současném přiblížení. Pokud je potřeba panel zobrazuje posuvníky, takže je možné vždy zobrazit celý obrázek v jakémkoli přiblížení.

9.6 Generování webové galerie

Po importování obrázků a jejich případné úpravě má uživatel možnost vygenerovat si z nich webovou galerii. Má také možnost si galerii přizpůsobit, tj. například změnit barvy a velikost textu, počet obrázků na stránku atd. V implementaci programu je toto řešeno přes témata, šablony a konfigurační soubory tématu. Oproti návrhu programu je ve finální implementaci použito pojmenování Téma namísto Šablony a šablonové soubory jsou pojmenovány jen jako šablony. Dále jsou také některé změny v pojmenování šablon.

Uživatel si veškeré nastavení výsledné webové galerie volí v okně generování galerie, kde jsou vypsané všechny podporované vlastnosti, jako je název galerie, počet položek na stránku apod. Pro každé téma je možné mít jiné vlastnosti.

9.6.1 Textové vzory

Textové vzory jsou textové řetězce formátu `{{ text }}`, kde `text` je obsah daného vzoru. Obsahem může být:

- Název nějaké proměnné (např. šířka právě generovaného obrázku v galerii či třeba jeho název).
- Výraz pro vložení jiného souboru, tj. formát textu: `include("název-souboru")`.
- Podmínkový výraz tj. formát textu: `if(var)`, kde `var` je název proměnné pro kontrolu existence a prázdné hodnoty. Podmínku lze také negovat,

tedy: `if(not var)`. Podmínkový blok pak musí být ukončen textovým výrazem `endif()`. Pokud je podmínka splněna, vše mezi začátkem bloku (`if`) a koncem bloku (`endif`) zůstane ve výsledném souboru, pokud není splněna, vše mezi začátkem a koncem bloku se do výsledného souboru nezapiše.

■ 9.6.2 Šablony

Šablony jsou soubory typu `.html`, `.css` nebo `.js`, které mohou obsahovat textové vzory. Šablony jsou zpracovány programem tak, že vezmou existující proměnné (argumenty) z daného kontextu a pomocí nich nahradí textové vzory obsahem dané proměnné. Pokud proměnná neexistuje, pak je textový vzor nahrazen prázdným textovým řetězcem.

■ 9.6.3 Téma

Hlavním prvkem při generování webové galerie je nutnost tématu. Téma si uživatel vybere před spuštěním generování galerie a následně je pak použito pro vytvoření HTML struktury a spojení všech dat z programu dohromady.

Každé téma obsahuje několik základních souborů:

- **info.txt** - Obsahuje informace o daném tématu, které se načítají do programu do okna generování galerie. Momentálně je v programu použit pouze údaj názvu tématu.
- **__base.html** - Hlavní soubor galerie obsahující základní strukturu HTML dokumentu. Měly by se v něm nacházet základní HTML prvky jako je `<html>`, `<head>`, `<body>`, případně `<link>` pro kaskádové styly a `<script>` pro JavaScript soubory či obsah. Dále může také obsahovat textové vzory `{{ header }}`, `{{ footer }}`, `{{ content }}` a `{{ title }}`, případně `{{ custom_css }}`, `{{ custom_js }}`
- **header.html** - Soubor používaný pro záhlaví galerie. Můžou se v něm nacházet textové vzory pro název galerie (`{{ gallery.name }}`), cestu generované složky (`{{ directory_path }}`) a navigační menu (`{{ navigation }}`).

- **navigation.html** - Soubor používaný pro navigační menu, může obsahovat textový vzor `{{ navigation.items }}`.
- **navigation-item.html** - Soubor používaný pro položku navigačního menu, může obsahovat textový vzor `{{ navigation.item.is_current }}`, `{{ navigation.item.url }}` a `{{ navigation.item.title }}`.
- **directory-path-part.html** - Soubor používaný pro položku navigačního menu, lze v něm použít textové vzory `{{ directory.is_root }}`, `{{ directory.is_current }}`, `{{ directory.url }}` a `{{ directory.name }}`.
- **footer.html** - Soubor používaný pro zápatí galerie.
- **index.html** - Soubor používaný pro hlavní stránku galerie, tj. pro kořenovou složku galerie. Může obsahovat textový vzor `{{ pagination }}`.
- **directory.html** - Soubor používaný pro ostatní složky galerie. Může obsahovat textový vzor `{{ pagination }}`.
- **list.html** - Soubor používaný pro výpis položek v galerii. Může obsahovat textový vzor `{{ items }}`.
- **list-item.html** - Soubor používaný pro jednotlivé položky ve výpisu položek v galerii. Lze použít textové vzory `{{ item.url }}`, `{{ item.path }}`, `{{ item.width }}`, `{{ item.height }}`, `{{ item.name }}`.
- **pagination.html** - Soubor používaný pro stránkování. Může obsahovat textové vzory `{{ pagination.items }}`, případně i `{{ pagination.previous.url }}` a `{{ pagination.next.url }}`.
- **pagination-item.html** - Soubor používaný pro položku ve stránkování. Může obsahovat textové vzory `{{ pagination.item.is_current }}`, `{{ pagination.item.url }}` a `{{ pagination.item.title }}`.
- **pagination-separator.html** - Soubor používaný pro položku značící vícero stránek ve stránkování mezi dvěma stránkami.
- **single.html** - Soubor používaný pro stránku jednotlivých obrázků. Může obsahovat textové vzory `{{ item.path }}`, `{{ item.name }}`, `{{ item.width }}`, `{{ item.height }}`, `{{ item.date }}`, `{{ item.annotation }}`, případně `{{ list.page }}`.
- **settings.txt** - Volitelný soubor. Obsahuje definice doplňujících vlastností, které jsou použité v šablonách tématu. Ty se pak zobrazí i v programu v okně generování galerie.

Program nabízí tři témata - moderní (modern), zaoblené (rounded) a jednoduché (simple). Lze z nich vycházet při tvorbě vlastních témat.

■ 9.6.4 info.txt

Tento soubor obsahuje hlavní informace o tématu, stejně jako bylo v návrhu programu. Program pak uživateli zobrazuje název tématu přímo v programu před spuštěním generování galerie. Na každém řádku je uvedena jedna informace, kde název údaje je první, pak následuje dvojtečka a pak je hodnota údaje. Tento soubor je povinný, jinak se téma nezobrazí v programu v seznamu témat.

Takový soubor pak může vypadat třeba takto:

```
name: A theme
version: 1.0
author: Author
description: A custom theme.
```

■ 9.6.5 settings.txt

Volitelný soubor obsahující doplňující proměnné, které lze následně použít v šablonách pro uživatelem přizpůsobitelná data. Proměnné uvedené v tomto souboru se zobrazí i v programu v okně generování galerie a uživatel jim bude moci změnit hodnotu. Každá proměnná má svůj název, pod kterým ji pak lze vložit do šablon, typ (`string`, `int{min, max}`, `int_string`, `float{min, max}`, `select{item0:title0, ...}`, `color`), nadpis (zobrazuje se v programu) a výchozí hodnotu.

Takový soubor pak může vypadat třeba takto:

```
items_per_row|int{1,10}|Items per row|3
go_back_text|string|Go back button text|Go back
```

Tímto se vytvoří v okně galerie posuvník pro *Items per row* s rozsahem od 1 (včetně) do 10 (včetně) a výchozí hodnotou 3 a pole textového vstupu pro *Go back button text* s výchozí hodnotou `Go back`.

■ 9.6.6 Proces generování

Program nejdříve vezme nastavení zadané uživatelem a s ním pak pokračuje jako s proměnnými při generování galerie.

Program začíná generováním kořenové složky. Pro každou složku, ať je kořenová, nebo ne, je stejný postup. Nejprve se vytvoří složka fyzicky na disku (pro kořenovou složku je použita složka vybraná jako výstupní složka galerie), pak se vezmou všechny podřazené položky složky a rozdělí se do stránek (v úvahu se bere nastavení počtu položek na stránku). Tím se vytvoří stránkování a jednotlivé stránky složky, stránkování však nemusí existovat kvůli malému počtu položek či vybrání nelimitování počtu položek na stránku a výsledkem tak bude jen jediný soubor. Pro každý soubor složky se nejdříve vygeneruje cesta složky a navigace (i když jsou stejné pro každou stránku složky, některé proměnné jsou jiné), poté záhlaví a zápatí a nakonec se vše pomocí základního souboru (`_base.html`) spojí dohromady. Soubor s obsahem stránky se uloží na disk. Pokud má složka náhledový obrázek, vygeneruje se z původního souboru a uloží se do složky náhledových obrázků. Takto se pokračuje rekurzivně pro všechny podřazené složky.

Pro každý soubor ve složce je pak obdobný postup, jen s tou výjimkou, že se používá jiná šablona, tj. namísto `index.html` nebo `directory.html` se použije `single.html`. Mimo náhledového obrázku se ještě zkopíruje původní obrázek do složky obrázků, ten se pak zobrazuje na stránce samotného obrázku.

Po vygenerování program ještě vytváří ve výstupní složce galerie symbolický odkaz na `index.html`, tj. na kořenovou složku galerie. Tímto je uživateli umožněno otevřít galerii jako jakýkoli jiný spustitelný program, pokud má tedy uživatel nastaveno, že se HTML soubory otevírají ve webovém prohlížeči.

Uživatel má také možnost při generování galerii uložit jako projekt. Tímto se pak umožní galerii znovu importovat do programu a pokračovat v její úpravě. Pokud je tato možnost zvolena, vytvoří se ve výstupní složce galerie ještě soubor projektu galerie.

9.7 Nastavení programu

V programu se také nachází jeho nastavení. Lze jej zobrazit v hlavním okně programu pomocí položky v liště menu *Application > Settings*. Po vybrání položky se zobrazí okno s nastavením programu. Momentálně lze pouze změnit velikost písma programu.

V programu je také implementována možnost restartu programu po uložení změn v nastavení. Po kliknutí na tlačítko potvrzení v okně nastavení programu

je možnost program ihned restartovat a tak vidět provedené změny. Program se před ukončením zeptá uživatele, zdali chce uložit otevřený projekt. Následně se program ukončí a poté znovu spustí. Pokud uživatel projekt předtím uložil, automaticky se znovu otevře.

■ 9.8 Indikátor průběhu

Napříč programem existují operace, které mohou trvat delší dobu, např. import souborů či složek, generování galerie apod. Při takovýchto operacích se objeví okno indikátoru průběhu, které má ve svém názvu název probíhající operace a v obsahu text pro detailnější popis a ukazatel průběhu pro lepší vizualizaci.

Každá operace, která využívá okno indikátoru průběhu, je v programu implementována pomocí dvou zásadních komponent - statusu (Status), který drží informace o průběhu operace, a úkolu (Task), který spravuje běh operace v jiném vlákne. Tímto se dokáží relativně jednoduše převést akce, které běží v tzv. GUI vlákne, přepsat tak, aby běžely ve vlákne jiném a za pomocí okna indikátoru průběhu. Úkoly používají *QThread* z aplikačního rámce *Qt* a vlastní struktury pro zacházení s možnými chybami a automatickou správu života vlákna a objektu úkolu. Takto se program na vyšší úrovni nemusí starat o již vykonané operace a zároveň je poskytnuta jednoduchá tvorba takovýchto úkolů.



Kapitola 10

Testování programu

Testování programu probíhalo po celou dobu vývoje programu. Při implementaci se program průběžně testoval na vybraných operačních systémech, konkrétně na Ubuntu 20.04.2, Debian 10.9.0 a Windows 10. Byla nejenom testována kompilace zdrojových souborů a zdali lze program úspěšně sestavit, ale také použitelnost programu na daných operačních systémech. Program se vždy nejdříve muselo podařit sestavit a teprve po úspěšném sestavení se následně zkusilo programem projít, tj. vyzkoušet jeho funkce.

Další testování probíhalo na konci vývoje programu a to pomocí uživatelského testování. Program byl předán čtyřem uživatelům s různou zkušeností s prací na počítači a různou zkušeností s podobnými programy. Tři z nich používají operační systém Windows, jeden uživatel používá operační systém Ubuntu. Uživatelé dostali za úkol programem projít a pokusit se vytvořit si webovou galerii z vlastně vybraných obrázků. Celý průběh testování probíhal pod dohledem a zmíněné nedostatky programu a připomínky uživatelů byly zapsány do jednotlivých seznamů. Získaná zpětná vazba byla zpracována a vytvořil se z ní seznam navržených úprav. Vybrané navržené úpravy se pak následně implementovaly do programu.

10.1 Zpětná vazba

Uživatel 1, operační systém Ubuntu:

- Matoucí umístění panelu pro vytváření nové složky ve výpisu položek.
- Chybějící indikátor stavu při importování, generování a editování obrázků v editoru obrázků.
- Program nekontroluje existenci složek pro zdroje programu a šablony při jeho spuštění.
- Nelze změnit velikost náhledů obrázků a složek ve výpisu položek.
- Nelze jednoduše zobrazit obrázek z výpisu položek a prohlédnout si ho.
- Při otevření editoru obrázků není vidět celý obrázek, pokud je větší.
- Ve výpisu obrázků nejsou vidět dodatečné informace o obrázku.
- Položky ve výpisu nejdou vyřadit z generování galerie, lze je jen zcela odebrat.
- Text obrázků s běžným formátem se ve výpisu položek zobrazuje na dva řádky.
- Nutnost vybrat výstupní složku pro vygenerovanou galerii.
- Nelze změnit název složky ve výpisu položek.
- Nelze zobrazit všechny položky v galerii na jednu stránku, pokud jich je více než maximální limit.
- Nelze změnit rozlišení náhledů ve vygenerované galerii.
- Nelze skrýt názvy obrázků ve vygenerované galerii.
- Nelze změnit velikost mezery mezi položkami ve vygenerované galerii.
- Výchozí hodnota polí pro filtr podle data u importování složky není blízko dnešnímu dni.

Uživatel 2, operační systém Windows:

- Chybějící indikátor stavu při importování, generování a editování obrázků v editoru obrázků.

- Nelze jednoduše zobrazit obrázek z výpisu položek a prohlédnout si ho.
- Při otevření editoru obrázků není vidět celý obrázek, pokud je větší.
- Ve výpisu obrázků nejsou vidět dodatečné informace o obrázku.
- Nelze otevřít vygenerovanou galerii po opětovném spuštění programu bez toho, aniž by uživatel musel manuálně galerii uložit jako projekt.
- Matoucí umístění panelu pro vytváření nové složky ve výpisu položek.
- Chybí výchozí název složky při vytváření nové složky ve výpisu položek.
- Nelze přejmenovat složku ve výpisu položek.
- Nelze vybrat vícero souborů ve výpisu položek.

Uživatel 3, operační systém Windows:

- Matoucí umístění panelu pro vytváření nové složky ve výpisu položek.
- Chybějící indikátor stavu při importování, generování a editování obrázků v editoru obrázků.
- Nelze změnit rozlišení náhledů ve vygenerované galerii.
- Nelze otevřít vygenerovanou galerii po opětovném spuštění programu bez toho, aniž by uživatel musel manuálně galerii uložit jako projekt.
- Chybějící možnost otevřít vygenerovanou galerii dvojklikem, jako například nějaký program.

Uživatel 4, operační systém Windows:

- Chybějící indikátor stavu při importování, generování a editování obrázků v editoru obrázků.
- Při otevření editoru obrázků není vidět celý obrázek, pokud je větší.
- Nelze přejmenovat složku ve výpisu položek.
- Nelze vybrat vícero souborů ve výpisu položek.
- Nelze jednoduše zobrazit obrázek z výpisu položek a prohlédnout si ho.

10.2 Navržené úpravy

Jak je z jednotlivých seznamů možno vidět, všechny z nich obsahují některé opakující se nedostatky v programu. Ze seznamů byly vybrány všechny připomínky a následně implementovány do programu. Implementovaná řešení jsou uvedena dále v textu.

- **Matoucí umístění panelu pro vytváření nové složky ve výpisu položek.**

Řešení: Nahrazení panelu tlačítkem pro vytvoření nové složky, které zobrazí okno se stejným panelem. Tímto se matoucí panel skryje a zobrazí se jen tehdy, pokud jej uživatel vyžaduje.

- **Chybějící indikátor stavu při importování, generování a editování obrázků v editoru obrázků.**

Řešení: Přidání dialogového okna pro zobrazení stavu probíhající akce. Název akce se zobrazuje v názvu okna a okno samotné obsahuje text pro zobrazení detailnějšího popisu stavu a také ukazatel průběhu. Pro některé akce, tj. například importování souborů, je možnost zobrazit přímo konkrétní stav, tedy kolikátý soubor z kolika souborů se právě importuje, u ostatních akcí, tj. například změna kontrastu obrázku, lze zobrazit alespoň informační text a neurčitý ukazatel průběhu, který indikuje, že se program nezasekl a stále pracuje.

- **Program nekontroluje existenci složek pro zdroje programu a šablony při jeho spuštění.**

Řešení: Při spuštění programu kontrolovat, zdali složky pro zdroje programu a šablony existují. Pokud neexistují, zobrazit chybovou hlášku a program následně ukončit.

- **Nelze změnit velikost náhledů obrázků a složek ve výpisu položek.**

Řešení: Přidat posuvník pro změnu velikosti náhledů pod výpis položek. Jeho posunutím se změní velikost náhledů na předem stanovenou velikost a aktualizuje se výpis položek.

- **Nelze jednoduše zobrazit obrázek z výpisu položek a prohlédnout si ho.**

Řešení: Změnit chování kliknutí myši u položky ve výpisu položek. Dvojklik levým tlačítkem otevře obrázek v editoru obrázků a klik pravým tlačítkem zobrazí okno s informacemi o souboru.

- **Při otevření editoru obrázků není vidět celý obrázek, pokud je větší.**

Řešení: Při otevření editoru obrázků změnit přiblížení tak, aby byl obrázek vidět celý.

- **Ve výpisu obrázků nejsou vidět dodatečné informace o obrázku.**
Řešení: Přidat nový typ zobrazení výpisu položek. Položky se zobrazují jako seznam, kde každá položka má v prvním sloupci zobrazen náhled a v druhém sloupci pak její detailnější informace. Přidat také tlačítka pro změnu typu zobrazení, tedy pro zobrazení mřížky (původní zobrazení) a seznamu (nové zobrazení).
- **Položky ve výpisu nejdou vyřadit z generování galerie, lze je jen zcela odebrat.**
Řešení: Přidat funkci vyřazování/zahrnování položek ve vygenerované galerii. Každou položku lze vybrat a následně ji kliknutím na tlačítko vyřadit či kliknutím na jiné tlačítko opět zahrnout. Vyřazená položka se ve vygenerované galerii nezobrazí.
- **Text obrázků s běžným formátem se ve výpisu položek zobrazuje na dva řádky.**
Řešení: Zmenšit písmo názvu položky. Jelikož název může být jakkoli dlouhý, nelze tento problém vyřešit pro všechny délky názvu, ale jen pro některé, tedy hlavně ty běžné.
- **Nutnost vybrat výstupní složku pro vygenerovanou galerii.**
Řešení: Pokud je pole pro výstupní složku prázdné, vygenerovat galerii do výchozí složky nacházející se ve složce s programem.
- **Nelze změnit název složky ve výpisu položek.**
Řešení: Přidat pro složky podobné okno, jako je okno informací o obrázku. V tomto okně se nachází pole pro změnu názvu složky a tlačítko pro změnu náhledového obrázku složky.
- **Nelze zobrazit všechny položky v galerii na jednu stránku, pokud jich je více než maximální limit.**
Řešení: Přidat zaškrtačací políčko pro vypnutí limitu položek na jednu stránku. Pokud je políčko zaškrtnuto, všechny položky ve složce se zobrazí jen na první stránce. Tímto se zcela skryje stránkování.
- **Nelze změnit rozlišení náhledů ve vygenerované galerii.**
Řešení: Přidat pole pro specifikování velikosti náhledu ve vygenerované galerii. Pro všechny náhledy se pak vezme tato hodnota a vygeneruje se obrázek s maximálně touto velikostí, tj. šířkou či výškou.
- **Nelze skrýt názvy obrázků ve vygenerované galerii.**
Řešení: Přidat zaškrtačací políčko pro skrytí názvů obrázků ve vygenerované galerii.
- **Nelze změnit velikost mezery mezi položkami ve vygenerované galerii.**

Řešení: Přidat pole pro zadávání velikosti mezery mezi položkami ve vygenerované galerii.

- **Výchozí hodnota polí pro filtr podle data u importování složky není blízko dnešnímu dni.**

Řešení: Pole pro filtr koncového data nastavit na hodnotu současného času a pole pro filtr začínajícího data nastavit na hodnotu současného času před týdnem.

- **Nelze otevřít vygenerovanou galerii po opětovném spuštění programu bez toho, aniž by uživatel musel manuálně galerii uložit jako projekt.**

Řešení: Při generování galerii také uložit galerii jako projekt programu, a to do složky s vygenerovanou galerií. Uživatel má pak možnost galerii opětovně otevřít v programu po jeho zavření. Přidat také políčko pro možnost neukládat galerii jako projekt programu.

- **Chybí výchozí název složky při vytváření nové složky ve výpisu položek.**

Řešení: Pokud je název nové složky prázdný, zvolit nějaký výchozí. Pokud složka s výchozím názvem již existuje, přidat do názvu ještě koncovku, tj. například '(1)'.
(1)

- **Nelze vybrat vícero souborů ve výpisu položek.**

Řešení: Přidat možnost vybrat vícero souborů pomocí kliknutí na položku ve výpisu položek se stisknutou kontrolní klávesou, tj. například CTRL. Jakmile je vybráno vícero souborů, operace, které se ovládají pomocí tlačítek nad výpisem položek, uzpůsobit tak, aby fungovaly na všechny vybrané soubory, ne jen na jeden z nich.

- **Chybějící možnost otevřít vygenerovanou galerii dvojklikem, jako například nějaký program.**

Řešení: Při generování galerie vytvořit i symbolický odkaz na soubor index.html nacházející se v kořenové složce galerie. Symbolický odkaz je pak ve stejné složce jako je složka vygenerované galerie. Po kliknutí na něj, pokud má tedy uživatel nastavené, že se HTML soubory otvírají ve webovém prohlížeči, se přímo otevře index galerie.

10.3 Opětovné testování

Opětovné testování probíhalo po implementaci navržených úprav a obdobně jako prvotní testování.

■ 10.3.1 Zpětná vazba

Uživatel 1, operační systém Ubuntu:

- Chybí možnost uložit již předtím uložený projekt bez opětovného vybrání cesty.
- Chybí indikátor průběhu u otevírání a ukládání projektu.
- Chybí možnost uložit projekt při zavírání programu.
- Chybí možnost program restartovat po uložení nastavení programu.
- Nelze přemístit položky mezi složkami ve výpisu položek.

Uživatel 2, operační systém Windows:

- Chybí možnost uložit projekt při zavírání programu.
- Chybí indikátor průběhu u otevírání a ukládání projektu.

Uživatel 3, operační systém Windows:

- Nelze přemístit položky mezi složkami ve výpisu položek.

Uživatel 4, operační systém Windows: Uživatel nenahlásil žádné další nedostatky.

■ 10.3.2 Navržené úpravy

Z jednotlivých seznamů se opět vybraly všechny připomínky a byly následně implementovány do programu. Implementovaná řešení jsou uvedena dále v textu.

- **Chybí možnost uložit již předtím uložený projekt bez opětovného vybrání cesty.**

Řešení: Přidat do menu položku **Uložit jako**, která se chová tak, že se otevře dialogové okno pro výběr cesty k souboru, kam se projekt následně uloží. A také změnit chování položky **Uložit** a to tak, že pokud již byl projekt předtím uložen, uloží se do poslední vybrané cesty.

- **Chybí indikátor průběhu u otevírání a ukládání projektu.**

Řešení: Přidat indikátor průběhu u otevírání a ukládání projektu. Indikátor se chová stejně jako indikátor například u aplikování operací v editoru obrázků.

- **Chybí možnost uložit projekt při zavírání programu.**

Řešení: Přidat dialogové okno s otázkou, zdali chce uživatel před zavřením programu uložit projekt. Po kliknutí na tlačítko **Ano** se projekt uloží a následně se program ukončí, po kliknutí na tlačítko **Ne** se program pouze ukončí.

- **Chybí možnost program restartovat po uložení nastavení programu.**

Řešení: Přidat dialogové okno s otázkou, zdali chce uživatel program restartovat nyní. Po kliknutí na tlačítko **Ano** se program ukončí (i zde platí zeptání se uživatele, zdali chce uložit projekt) a poté se opět spustí. Pokud uživatel měl předtím otevřený a uložený projekt, po spuštění programu se projekt znovu otevře. Pokud projekt nebyl uložen, objeví se klasické uvítací okno. Po kliknutí na tlačítko **Ne** v dialogovém okně s otázkou se program nerestartuje a zůstane otevřený, pouze se uzavře okno s nastavením programu.

- **Nelze přemístit položky mezi složkami ve výpisu položek.**

Řešení: Přidat funkci pro přetahování vybraných položek na položky složky tak, že se otevře dialogové okno pro potvrzení, zdali chce uživatel vybrané položky do složky vložit, nebo zdali chce jen změnit pořadí vybraných položek. Po kliknutí na tlačítko **Ano** se položky odeberou ze seznamu otevřené složky a vloží se do složky vybrané. Po kliknutí na tlačítko **Ne** se změní pouze pořadí položek. Jelikož by tímto způsobem nebylo možné položky přesouvat do nadřazené složky, je nutné přidat i tlačítko pro právě tuto funkcionalitu. Po kliknutí na toto tlačítko se vybrané položky přesunou do nadřazené složky, pokud se tedy vybrané položky nenachází v kořenové složce.

■ 10.3.3 Další testování

Po implementaci předchozích úprav probíhalo i další uživatelské testování, uživatelé však již nenašli žádné další závažné nedostatky programu. Tímto bylo uživatelské testování ukončeno.



Kapitola 11

Závěr

Tato práce měla za úkol analyzovat již existující programy pro správu obrázků s možností exportu do webové galerie, navrhnout takovýto program a poté ho následně implementovat a otestovat pomocí uživatelského testování.

Postupně jsem analyzoval programy se stejnými či podobnými funkcemi, inspiroval jsem se jimi a po částech jsem utvářel vizi, jak by program mohl vypadat. Každý z analyzovaných programů implementoval některé funkce trochu jinak, a tudíž bylo potřeba vybírat ty více uživatelsky přívětivé. V některých případech jsou funkce řešeny velmi podobně jako u analyzovaných programů a v jiných případech jsou naopak navrženy úplně od začátku.

V návrhu implementace jsem se věnoval i menším detailům v programu a pokryl jsem všechny požadované funkce. Značnou část návrhu jsem se zabýval postupem pro získávání meta informací z obrázkových souborů a jeho pseudo implementací. Také jsem zahrnul i jednotlivé implementace obrázkových operací, které by v programu měly být zahrnuty. Krátce jsem se věnoval také návržení grafického rozhraní programu.

V samotné implementaci jsem pak většinu navržených funkcí implementoval, nicméně některé byly pozměněny či rozšířeny, podle toho, jak bylo potřeba. Další úpravy funkcí a grafického rozhraní byly provedeny po uživatelském testování. Program ve výsledku obsahuje všechny potřebné a požadované funkce.

Nakonec bylo provedeno uživatelské testování na základě čtyř uživatelů s různou zkušeností s prací na počítači a s různou zkušeností s podobnými programy. Výsledkem testování byla zpětná vazba uživatelů v podobě seznamu, ze kterého jsem vytvořil seznam navržených úprav a ty následně implementoval do programu. Tímto se do programu přidaly některé šikovné a někdy i nutné funkce. Tím se program výrazně doladil.

11.1 Rozšíření práce

Na programu lze pracovat nadále a postupně ho vylepšovat, existuje ještě řada funkcí, které by program vynesly a zlepšily tak jeho používání. Lze také dále pokračovat v analýze podobných programů a získávat více informací o tom, jak jsou některé funkce řešené a zdali by bylo užitečné je implementovat do programu.

Program by bylo vhodné přeložit i do jiných jazyků, aby ho mohlo využívat více lidí. Lze také řešit různé úpravy a doladění grafického rozhraní či také program rozšířit o automatickou kontrolu verzí a následnou automatickou aktualizaci. Výrazným zlepšením by třeba mohla být i extrakce více metadat ze souborů nebo také zlepšení extrakce data pořízení obrázku, aby byl podporován širší záběr formátů z různých fotoaparátů. Tímto vším by se zvedla kvalita a použitelnost programu.

Příloha A

Literatura

- ¹ Formát jpeg. "<https://en.wikipedia.org/wiki/JPEG>".
- ² Formát tiff. "<https://en.wikipedia.org/wiki/TIFF>".
- ³ Formát png.
"<http://www.libpng.org/pub/png/spec/1.2/PNG-Contents.html>".
- ⁴ Exif formát.
"<https://www.media.mit.edu/pia/Research/deepview/exif.html>".
- ⁵ Specifikace formátu tiff. "<https://www.adobe.io/content/dam/udp/en/open/standards/tiff/TIFF6.pdf>".
- ⁶ Specifikace formátu png. "<https://ftp-osl.osuosl.org/pub/libpng/documents/pngext-1.5.0.html#C.eXIf>".
- ⁷ Specifikace tiff exif. "<https://www.awaresystems.be/imaging/tiff/tifftags/exifid.html>".
- ⁸ Informace o exif. "<https://en.wikipedia.org/wiki/Exif#Technical>".
- ⁹ Struktura png. "<https://ftp-osl.osuosl.org/pub/libpng/documents/png-1.2-pdg.html#Structure>".
- ¹⁰ Hypertext markup language (html).
"<https://en.wikipedia.org/wiki/HTML>".
- ¹¹ Úprava jasu. "<https://www.dfstudios.co.uk/articles/programming/image-programming-algorithms/image-processing-algorithms-part-4-brightness-adjustment/>".

- ¹² Úprava kontrastu. "<https://www.dfstudios.co.uk/articles/programming/image-programming-algorithms/image-processing-algorithms-part-5-contrast-adjustment/>".
- ¹³ Odstíny šedi. "<https://www.dfstudios.co.uk/articles/programming/image-programming-algorithms/image-processing-algorithms-part-3-greyscale-conversion/>".
- ¹⁴ Inverzní barvy. "<https://www.dfstudios.co.uk/articles/programming/image-programming-algorithms/image-processing-algorithms-part-7-colour-inversion-solarisation/>".
- ¹⁵ Rozmazání. "https://en.wikipedia.org/wiki/Box_blur".
- ¹⁶ Zostření. "<https://web.stanford.edu/class/cs448f/lectures/2.1/Sharpening.pdf>, strana 8".
- ¹⁷ Zostření. "<https://www.learnopencv.com/automatic-red-eye-remover-using-opencv-cpp-python/>".
- ¹⁸ Flowlayout. "<https://code.qt.io/cgit/qt/qtbase.git/tree/examples/widgets/layouts/flowlayout?h=5.15>".
- ¹⁹ Alan Cooper, Robert Reimann, and David Cronin. *About Face 3: The Essentials of Interaction Design*. Wiley Publishing, 2007.
- ²⁰ Benjamin Baka. *Getting Started with Qt 5: Introduction to programming Qt 5 for cross-platform application development*. Packt Publishing, 2019.
- ²¹ Jesse Liberty and Bradley L. Jones. *Sams Teach Yourself C++ In 21 Days*. Sams Publishing, 2004.