



Středoškolská technika 2019

Setkání a prezentace prací středoškolských studentů na ČVUT

WEBMAKER

Jan Boháč, Ondřej Matura

Střední škola informatiky a finančních služeb, Plzeň, Klatovská 200 G

Anotace

Práce nabízí vhled do práce s html dokumenty, usnadňuje jejich tvoření pro začátečníky, a to i pro ty, kteří jsou s html seznámeni málo či dokonce vůbec. Aplikace je naprogramována v jazyce C# ve vývojovém prostředí (IDE) Visual Studio. Většina obsahu byla tvořena na bázi našich vlastních zkušeností – co by se nám hodilo, když jsme s html začínali?

Klíčová slova

Tvorba webu; webové stránky; html dokumenty,

Annotation

Our program offers insight into the working process with html documents. It eases the whole process of their creation for those, who are not familiar with creation itself much or even not at all. Application was coded using C# programming language and Visual Studio as our IDE. Majority of the content was based on our own experiences – what would be handy, at the time, we have begun with html?

Keywords

Web creation; web pages, html documents

Obsah

1	Úvod.....	4
2	Teoretická část	5
2.1	C#.....	5
2.1.1	Vývojové prostředí	5
2.2	Databáze.....	6
2.3	OOP.....	6
2.4	Uživatelské rozhraní.....	7
2.4.1	Technologie WPF	7
2.5	Stylování	8
2.6	Technologie MVVM.....	10
3	Aplikační část	11
3.1	Prvky	11
3.1.1	Vlastnosti prvků.....	12
3.1.2	Nahrání/přidání prvku.....	12
3.1.3	Verzování.....	14
3.2	Webmaker – konkrétní tvorba.....	15
4	Závěr	18
5	Shrnutí.....	19
6	Seznam obrázků.....	20
7	Využití zdroje.....	21

1 ÚVOD

Aplikace **WebMaker** je vytvořena pro veřejnost, mířící na jedince se zájmem o problematiku html. Práce cílí k vytvoření aplikace, která bude schopna pomoci úplným začátečníkům v dané oblasti, ale stejně tak i zlepšit tzv. Quality of life – udělat práci více pohodlnější i znalým problematiky.

Byla námi zvolena z několika důvodů. Jedním z nich je stav současných webových stránek na internetu. Některé z nich postrádají spousty prvků, které se dnes považují jako samozřejmost. Využitím tohoto programu bychom chtěli docílit vytvoření možnosti pro vytvoření takových stránek.

Jako kritérium toho, co považujeme za webovou stránku a výstup našeho programu, budeme využívat našich znalostí z programování získané během výuky HTML5 a našich zkušeností při tvorbě webu.

Zpětnou vazbu tvoří kritéria jednoduchosti tvorby v editoru a jeho vybavenosti. Cílem je vytvořit ideální poměr těchto dvou parametrů pro náš program.

Práce zároveň směřuje k vytvoření i částečné online části, ve které budou výtvořeny všech uživatelů nahrány, budou-li je chtít zveřejnit a zpřístupnit.



Obrázek 1: logo aplikace

2 TEORETICKÁ ČÁST

Jedná se o objektově orientovaný program, který využívá databáze. Všechny části tohoto programu byly napsány v jazyce C# za použití vývojového prostředí Visual Studio 2016 verze Community.

2.1 C#

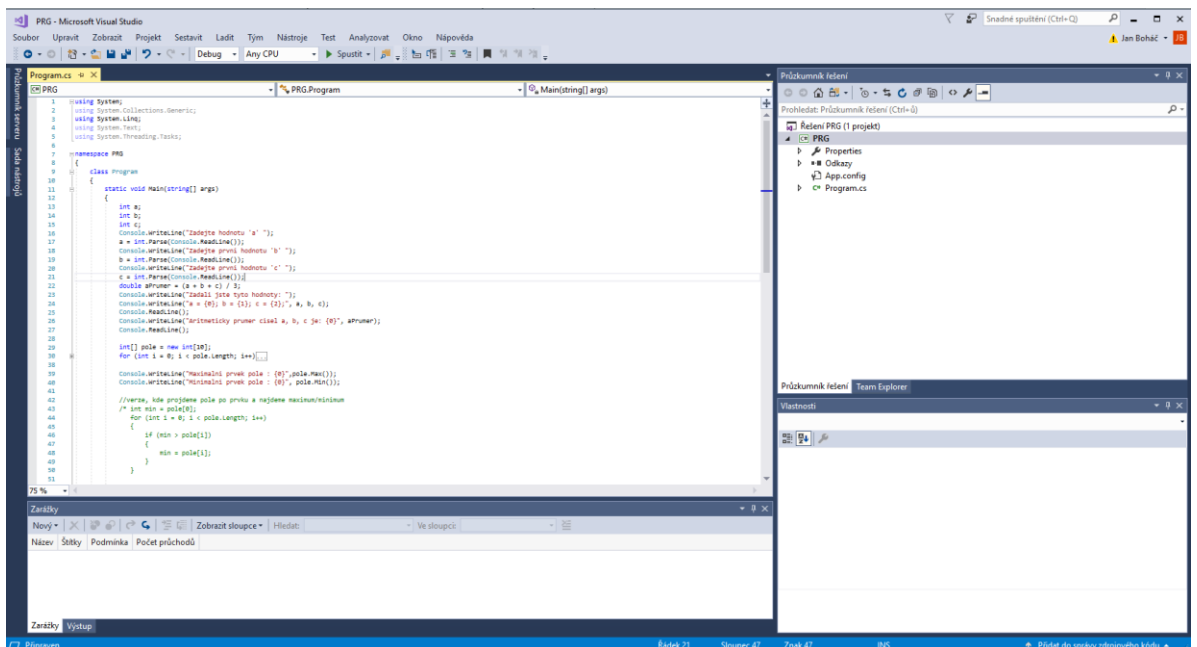
Jedná se o objektově orientovaný programovací jazyk, který původně vychází z jazyka Java a je vyvinut společností Microsoft. Veškerý kód programu je napsán v syntaxi jazyku C#.

2.1.1 Vývojové prostředí

Pro realizaci programu je nutné po zvolení jazyka zvolit i prostředí, do kterého budeme v daném jazyce psát. Takové prostředí pro C# je Visual Studio. Jedná se o vývojové prostředí, do kterého se implementují knihovny jazyka. Stejně jako jazyk samotný i vývojové prostředí Visual Studio bylo vyvinuto firmou Microsoft. [1]

Další jeho podstatnou součástí je umožnění přístupů a práce s .NET frameworkem, který je podstatný z důvodu množství jeho knihoven a nabízených ovladačů.

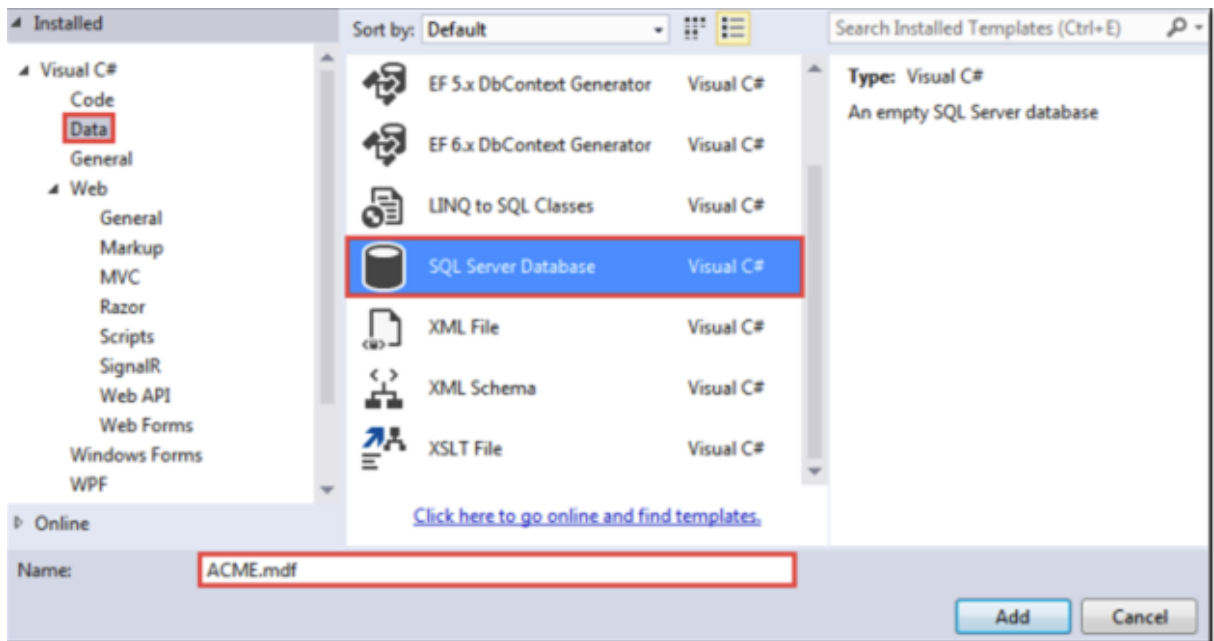
Mimo jiné jsou v nabídce IDE i ovládací prvky pro vytvoření jednoduché databáze a ovladače k nim.



Obrázek 2: ovládání IDE

2.2 Databáze

V tomto projektu jsou data – tedy výstup – uloženy a filtrovány ve vlastní lokální databázi uživatele, která slouží jako základní úložiště pro vytvořené webové stránky. Tato databáze je řízena jazykem SQL a využívá jeho zjednodušenou implementaci ve Visual Studiu



Obrázek 3: tvorba sql databáze ve Visual studiu

2.3 OOP

Objektově orientované programování je dnes prakticky standardem pro vytvoření komplexnější aplikace.

V této práci se dá práce s objekty najít již rovnou v databázi, kde je každý z přidávaných prvků objektem. To znamená, že veškeré šablony, veškeré vytvořené a nahrané dokumenty, které jsou viditelné v aplikaci jsou objekty.

Tyto objekty spojují společné vlastnosti – každá nahraná stránka v databázi má tyto vlastnosti, proměnné:

- jméno – název stránky, dle výběru,
- autor – původní autor stránky, povětšinou ten, který stránku nahrál do databáze,
- thumbnail – obrázek, který je v databázi vidět, když je zobrazen seznam prvků,
- popis – dodatečné informace ke stránce, které si autor přeje sdělit; zobrazené až po rozkliknutí prvku,
- HTML soubor – samotná stránka nebo zazipovaný web,

- dobrovolný prvek: dodatečné obrázky – zobrazené ve chvíli, kdy si uživatel konkrétní prvek rozklikne a přeje si náhled na nahraný dokument

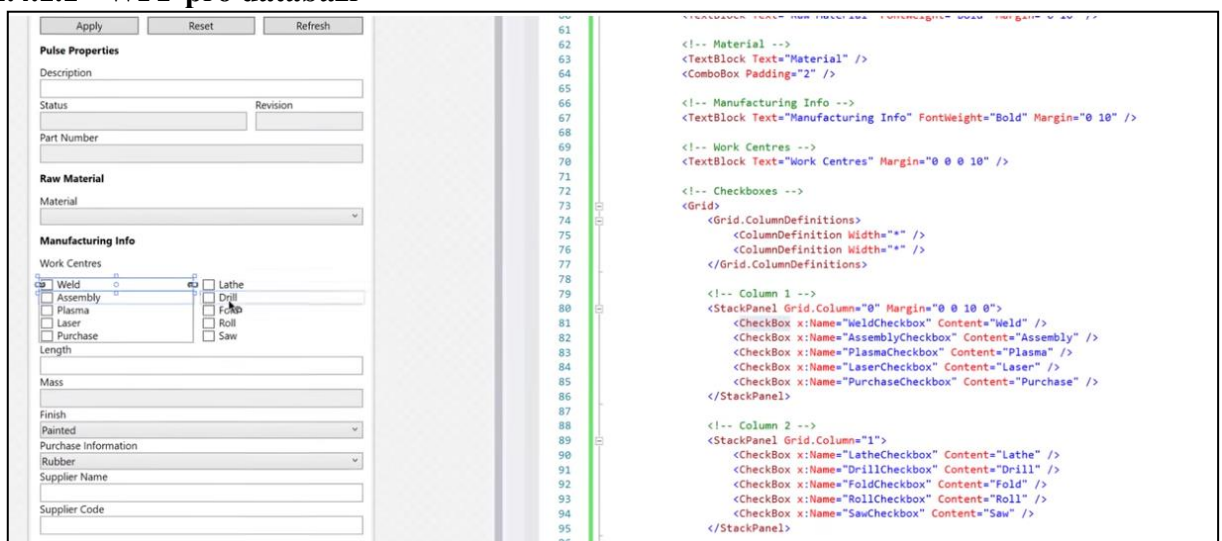
2.4 Uživatelské rozhraní

Veškerý front-end (viditelná část aplikace pro uživatele) je zobrazen na formulářích. V tomto projektu se konkrétně nachází 3 takovéto formuláře, které si vzájemně předávají data, se kterými později pracují.

2.4.1 Technologie WPF

Veškerá aplikační část front-endu, kterou uživatel vidí je zobrazena na speciálním druhu formuláře, který ani nenesou správný význam slova formulář. Jedná se o výstup WPF (Windows Presentation File), což je technologie umožňující vytvářet grafické uživatelské rozhraní. V .NET Frameworku je od verze 3.0 (původní název pro WPF byl Avalon). Ve WPF je připravena rozsáhlá zásoba nezbytných základních ovládacích prvků (tlačítko, textové pole, tabulka apod.) - nic nebrání ve vytvoření si vlastních. WPF používá k vykreslování knihovny Direct3D. Grafika v aplikaci je vykreslována vektorově, což s sebou nese nespočet výhod. Pro vytváření GUI lze využívat značkovací jazyk XAML, díky kterému lze oddělit funkčnost od vzhledu aplikace, a tak i podstatně zefektivnit vývoj aplikací v týmu. [3]

2.4.1.1 WPF pro databázi



Obrázek 4: WPF a jeho XML syntaxe

S již výše zmíněnými funkcemi se jedná o „domovský“ formulář. I když budeme spouštět jakýkoliv ze dvou dalších, vždy je uživatel spustí z tohoto rozcestníku. Funkce, které má tento formulář, jsou nastavené, aby se spustili se stiskem ovládacích prvků – tlačítek.

2.4.1.2 WPF pro WebMaker (tvorba samotného webu)

HTML-maker formulář bude sloužit k tvorbě HTML souborů. Uživatel vůbec nebude muset znát HTML (popřípadě ani Javascript, nebo CSS), aby byl schopný vytvořit jednoduché webové stránky.

Ovládání bude rozděleno do dvou skupin. První bude obsahovat možnosti úprav webu jako celku. Zde bude možnost přidat a odebrat jednotlivé stránky a vygenerovat HTML soubory, které budou propojené hypertextovými odkazy ve sjednoceném navigačním prvku, tato možnost otevře dialog pro výběr složky, kam se všechny soubory uloží. Druhá skupina ovládání bude sloužit k úpravám samostatných stránek. Druhá skupina ovládacích prvků bude obsahovat tlačítka pro přidání a odebrání jednotlivých prvků, jako například odstavec, nadpis, tabulka, seznam apod.

Poslední prvek bude panel záložek, kde každá stránka bude jako samostatná záložka, ve které se zobrazí všechny prvky dané stránky.

2.4.1.3 WPF pro prvky samostatné

Tento formulář není sám o sobě ničím zvláštní, ale to jen za předpokladu, že by byl samostatný. Spustí se ve chvíli, kdy si uživatel vybere nějaký prvek v databázi a klikne na něj. Formulář jako takový slouží pouze jako kostra, ve které jsou připravené obecné parametry každé stránky v databázi.

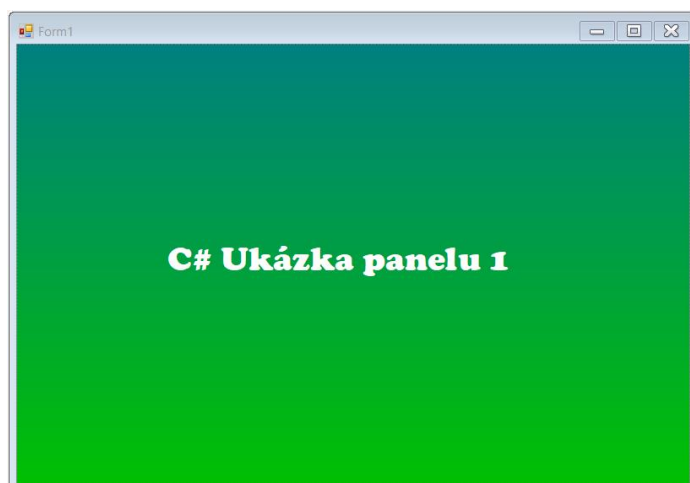
Prakticky ve chvíli, kdy kliknete na prvek X v databázi, tak se spustí formulář, kterému jsou předané parametry objektu. V tu chvíli se objeví autor, popis, název, samotná stránka a možné dodatečné obrázky konkrétního prvku X.

2.4.1.4 WPF pro vložení prvků

Zde se jedná o formulář, který v sobě má nabídku několika nevyplněných textových polí, kde je uživatel vyzván k jejich vyplnění. Tyto textová pole jsou určena pro parametry vloženého objektu. Zde lze využít prvek z nabídky dialogů – OpenFileDialog, Jelikož v rámci obrázku a samotného html dokumentu je k nahrání zapotřebí cesta k souboru. Psát ji ručně je, avšak nepraktické, proto využíváme tohoto dialogu, který uživateli umožňuje si soubor pohodlně najít ve vlastním počítači. Mimo jiné formulář nabízí i pohled na zmenšenou verzi obrázku – tedy na rozměry, které bude mít obrázek v databázi.

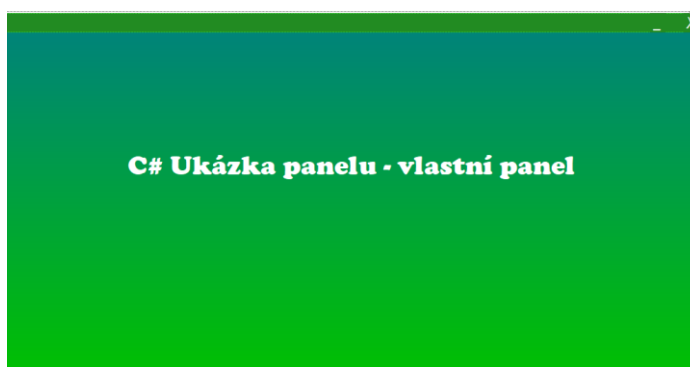
2.5 Stylování

Veškeré formuláře v projektu měly svůj počátek v designeru, který pro jazyk C# poskytuje vývojové prostředí Visual Studio. V tomto designeru je počáteční bod prázdný ohraničený formulář. [2]



Obrázek 5: tvorba vlastního pozadí s panelem v C# (ohraničení)

Toto ohraničení samotné je vcelku omezující a z větší části hlavně nevkusné. S tímto problémem se lze vypořádat odebráním těchto ohraničení (FormBorderStyle). Avšak odstranění ohraničení vytvoří formulář bez typické trojice tlačítek pro minimalizování, zvětšení/zmenšení a pro zavření. Tyto prvky mají pod sebou funkce, ty lze doprogramovat k tzv. panelu, který jsme schopni doprogramovat.



Obrázek 6: panel, který překrývá základní prvek

Principem je vytvoření panelu, který se připevní (anchor) na horní část formuláře a jsou na něm vloženy vlastní prvky – tlačítka, kterým se doprogramuje funkce. Zde se musí formulář samotný hýbat, za předpokladu, že uživatel myši přidržuje panel a pohybuje s ním po obrazovce. U normálního ohraničení to tak funguje, ale ve vlastním případě je nutno doprogramovat. Zde se jedná o jediný stylistický prvek, který za sebou má delší kód k pokrytí jeho funkcí a vlastností.

```

12 {
13     public partial class Form1 : Form
14     {
15
16         public const int WM_NCLBUTTONDOWN = 0xA1;
17         public const int HT_CAPTION = 0x2;
18
19         [System.Runtime.InteropServices.DllImportAttribute("user32.dll")]
20         public static extern int SendMessage(IntPtr hWnd, int Msg, int wParam, int lParam);
21         [System.Runtime.InteropServices.DllImportAttribute("user32.dll")]
22         public static extern bool ReleaseCapture();
23
24
25         public Form1()
26         {
27             InitializeComponent();
28         }
29
30         private void button5_Click(object sender, EventArgs e)
31         {
32             Application.Exit();
33         }
34
35         private void button6_Click(object sender, EventArgs e)
36         {
37             this.WindowState = FormWindowState.Minimized;
38         }
39     }

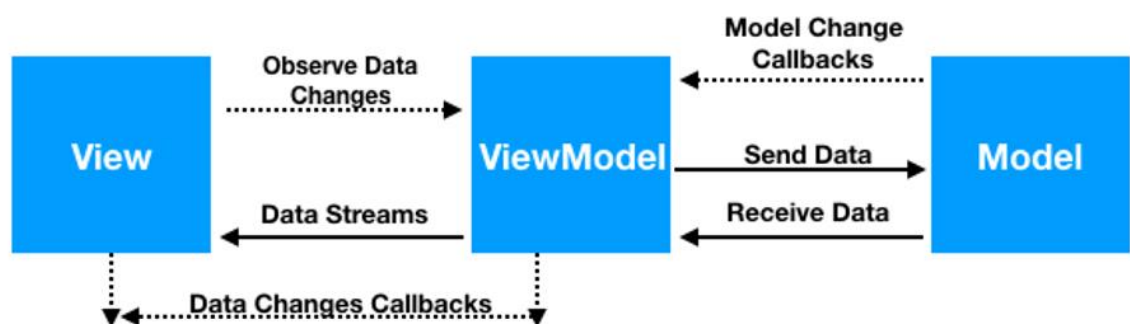
```

Obrázek 7: kód, který zaznamenává pozici kurzoru

Jedná se o využití statických proměnných, které programu sdělují pozici myši a podle ní se okno samotné nastaví na správnou pozici.

2.6 Technologie MVVM

Technologie, která slouží pro zkvalitnění komunikace a separace mezi logikou uživatelského rozhraní a business logiky. Jedná se o kombinaci tří podstatných prvků, kde každý má svůj účel. [3]



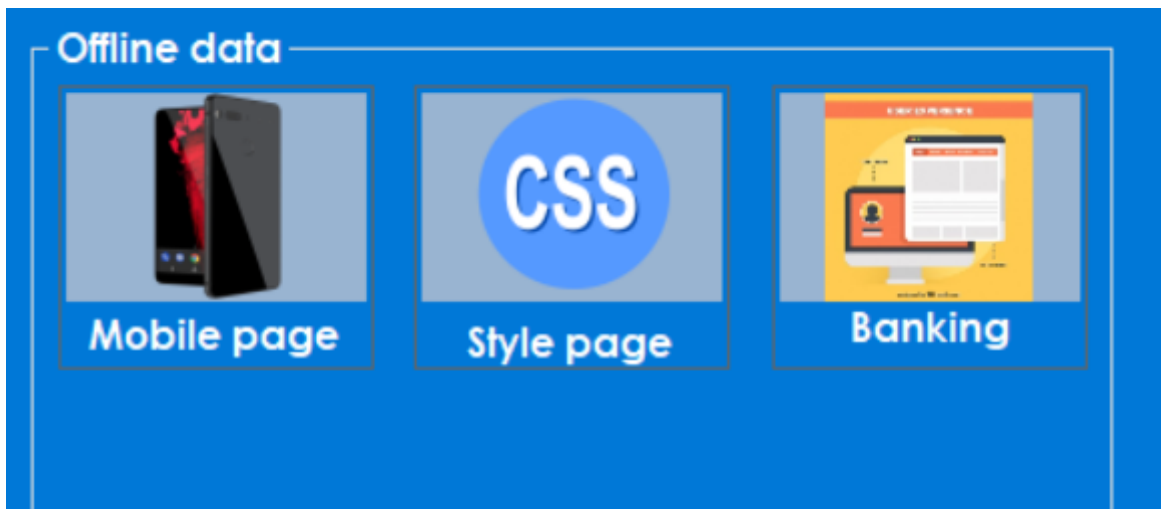
Obrázek 8: schéma MVVM dostupné na

<https://www.journaldev.com/20292/android-mvvm-design-pattern>

3 APLIKAČNÍ ČÁST

3.1 Prvky

V samotné databázi se již zmíněné prvky nacházejí zarovnané ve svých okruzích ale přednastavených vlastností. Uživatel může vložit obrázek jakékoliv velikosti, ale je vždy přizpůsoben tak, aby velikostně pasoval do databáze.



Obrázek 9: první verze offline rozhraní

V kódu programu se tedy nachází opatření, pro úpravu velikosti obrázků tak, aby se vešly tři takové na jednu řádku při daném rozlišení. Dále je i viditelná jejich vlastnost – reakce na přejetí myši (mouseOver), kdy část s prvkem změní barvu, která je na obrázku šedě ohraničena, přejede-li uživatel přes ní kurzorem myši

```
System.Data.SqlClient.SqlConnection con;

private void Form1_Load(object sender, EventArgs e)
{
    con = new System.Data.SqlClient.SqlConnection();

    con.ConnectionString = "Data Source=.\SQLEXPRESS;
AttachDbFilename=C:\\MyWorkers.mdf;Integrated Security=True;Connect
Timeout=30;User Instance=True";

    con.Open();

    MessageBox.Show("Database Open");

    con.Close();

    MessageBox.Show("Database Closed");
}
```

Obrázek 10: rozhraní databáze, navázání spojení

3.1.1 Vlastnosti prvků

Po propojení se do databáze začnou ukládat prvky při každém výtvaru nebo navázání spojení. Po navázání spojení mezi lokální databází programu a jeho dynamickou, cloudovou kopií dojde k aktualizování online databáze s možností stáhnout si poslední release šablon.

	Id	zipped_file	thumbnail	author	description	page_name
	1	C:\Users\saeri\Desktop\path_test\test.zip	C:\Users\saeri\Desktop\path_test\pic.png	User12	Page debating ba...	NULL
	3	ftp://ftp.webmaker.com/path_test/test.zip	ftp://ftp.webmaker.com/path_test/snap.jpg	User25	Web uploaded on...	webTest
**	NULL	NULL	NULL	NULL	NULL	NULL

Obrázek 11: ukázka prvků a porovnání online a offline databáze

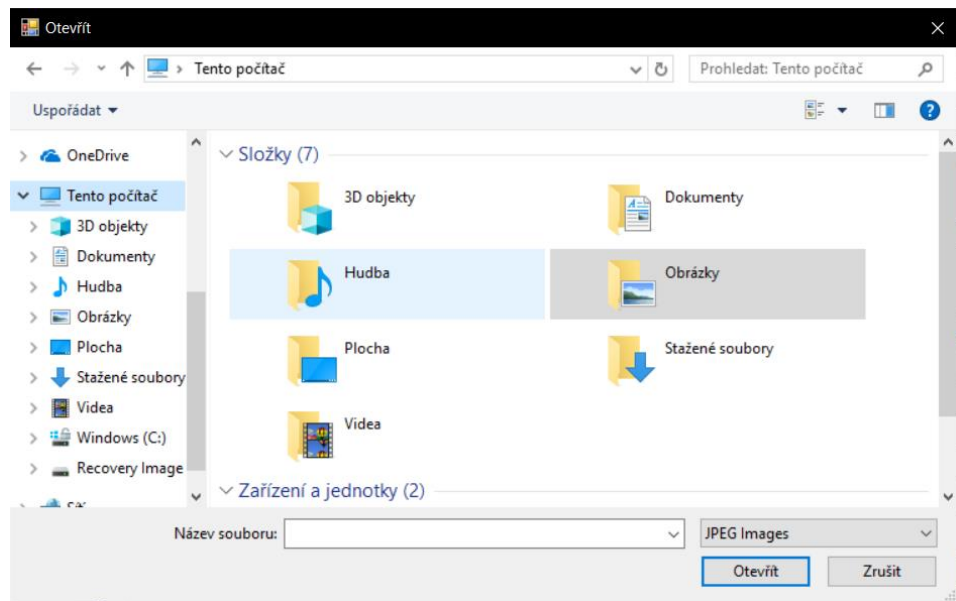
Samotná databáze se pro svém propojení sama přizpůsobí ohledně souborů, které v databázi jsou a zároveň jsou přítomny v počítači, resp. Lokálním uložišti. Zdali nejsou, pomocí protokolu si zapíše jeho adresu online a stáhne jí pouze, pokud k tomu udá žádost uživatel s přístupem k internetu.

3.1.2 Nahrání/přidání prvku

Veškerý tento kódově složitý proces se schovává za příjemné prostředí tlačítek, které veškerou práci udělají za uživatele. Tomu stačí jen vyplnit následující údaje:

- Autor – většinou jím bývá člověk nahrávající soubor,
- stručný název na pojmenování stránky,
- odkaz na samotný dokument (v podobě cestě k souboru),
- odkaz na obrázek, který se uloží (též pomocí cesty k souboru),
- dodatečný popis – zde autor může učinit svá prohlášení o tom, jak chce, aby bylo s jeho stránkou zacházeno dále, zvolí-li si ji nechat veřejnou. Zároveň shrnuje účel a základní prvky stránky.

Jelikož psát cestu k souboru ručně je velmi namáhavá a nepohodlná práce, využíváme dialogové okno, které Visual Studio nabízí. V tomto okně uživatel jednoduše soubor najde a po zvolení možnosti otevřít se cesta napíše do textového pole. V případě obrázku se vytvoří i náhled na obrázek ve zmenšené podobě.



Obrázek 12: okno dialogu

Po nalezení souborů a jeho vypsání cesty stačí pouze kliknout na tlačítka „Upload“ a není-li žádná cesta k souborům chybná, prvek je přidán do databáze pro pozdější využití jako jsou stažení či úprava.

Author
Jan Boháč

Name your web page
User experience on web

Link to html
C:\Users\saeri\Documents\addPage\new

Link to thumbnail
C:\Users\saeri\Pictures\user-experience-e

Describe your page
In this page, i would like to summarize all details I possibly could, to get society info about most typical experience use gets when visiting the web - the world online. I approve this page to be publically used and uploaded on my behalf.

USER EXPERIENCE

Upload

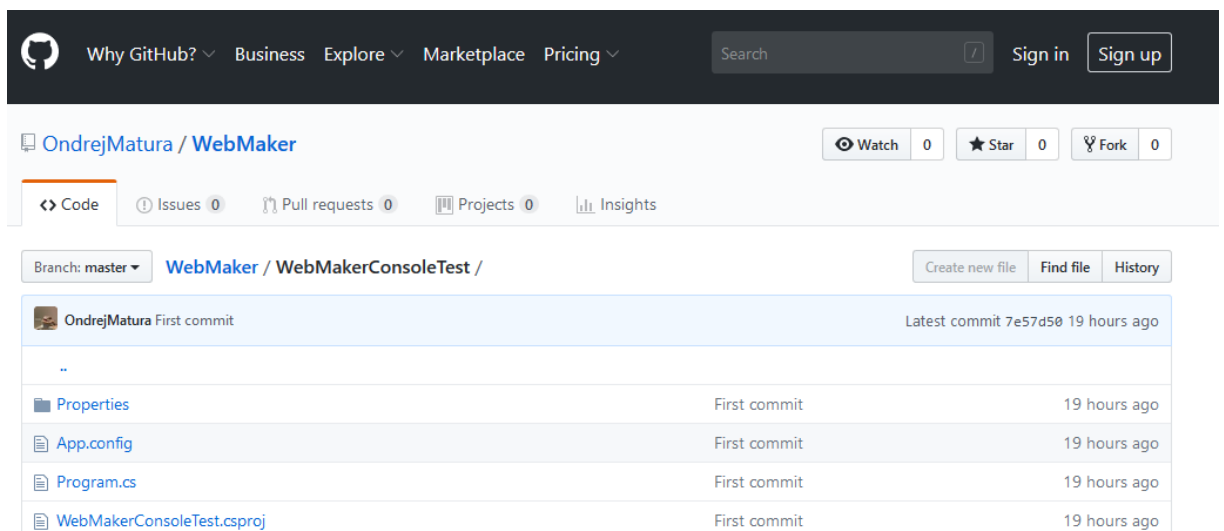
Obrázek 13: formulář pro vkládání prvku do databáze

```
1  #using ...
10
11 namespace BaseEntry
12
13     public partial class Form1 : Form
14     {
15         public Form1()
16         {
17             InitializeComponent();
18         }
19         string chosenFile = "";
20         private void button1_Click(object sender, EventArgs e)
21         {
22             openFileDialog1.InitialDirectory = System.Environment.GetFolderPath(Environment.SpecialFolder.Personal);
23             openFileDialog1.FileName = "";
24             openFileDialog1.Filter = "JPEG Images|.jpg|PNG Images|.png";
25             openFileDialog1.ShowDialog();
26             chosenFile = openFileDialog1.FileName;
27             pictureBox1.Image = Image.FromFile(chosenFile);
28         }
29
30         private void pictureBox1_Click(object sender, EventArgs e)
31         {
32             if (pictureBox1.Width == 200)
33             {
34                 pictureBox1.Width = 400;
35                 pictureBox1.Height = 400;
36             }
37             else {
38                 pictureBox1.Width = 200;
39                 pictureBox1.Height = 200;
40             }
41         }
42     }
```

Obrázek 14: kód pro rozpoznání cesty souboru z okna dialogu

3.1.3 Verzování

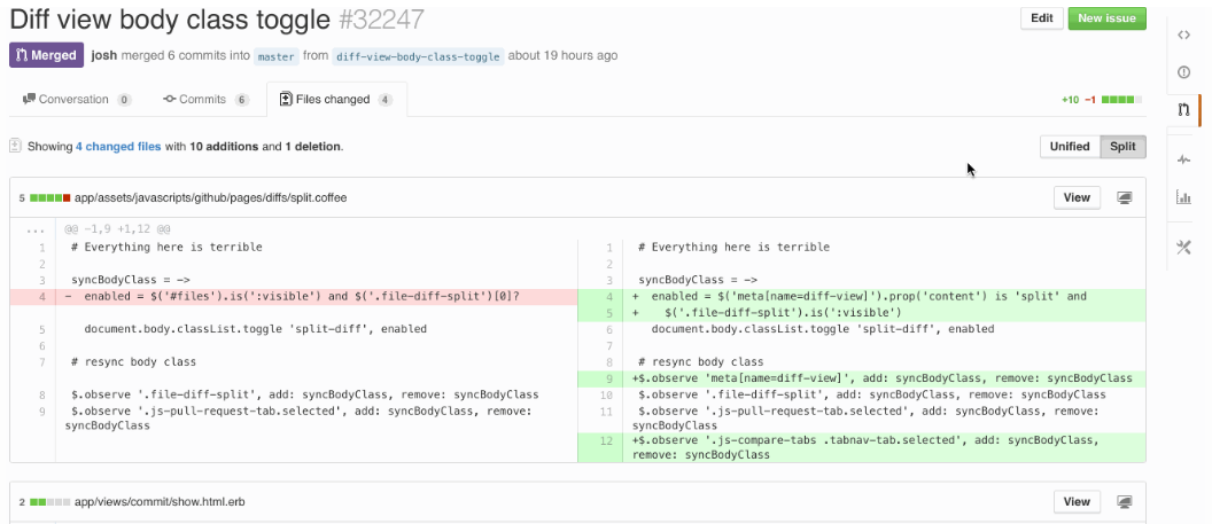
Všechny změny v programu proběhly na bázi sdíleného úložiště GitHub, kde je uložen kód, který je zároveň i v příloze (resp. odkaz na něj).



Obrázek 15: GitHub úložiště projektu

3.1.3.1 GitHub

V tomto uložišti, které slouží jako užitečný nástroj pro kontrolu a práci s kódem po jeho uložení má dvě primární výhody – funkce. Kromě poskytnutí online uložiště řeší problematiku verzování: práce s projektem s tím, že se stále vyvíjí a každá jeho nová verze může přinést komplikace. GitHub nabízí seznam všech verzí, jejich přidávání a ovlivňování jednotlivých



Obrázek 16: verze; změny mezi verzemi

částí projektu napříč časem a zároveň se jedná o skvělý prostředek spolupráce, kdy jako autoři můžeme najednou přispívat a dělat na dvou různých verzích zároveň.

3.2 Webmaker – konkrétní tvorba

WebMaker grafické rozhraní je rozděleno na 3 části:

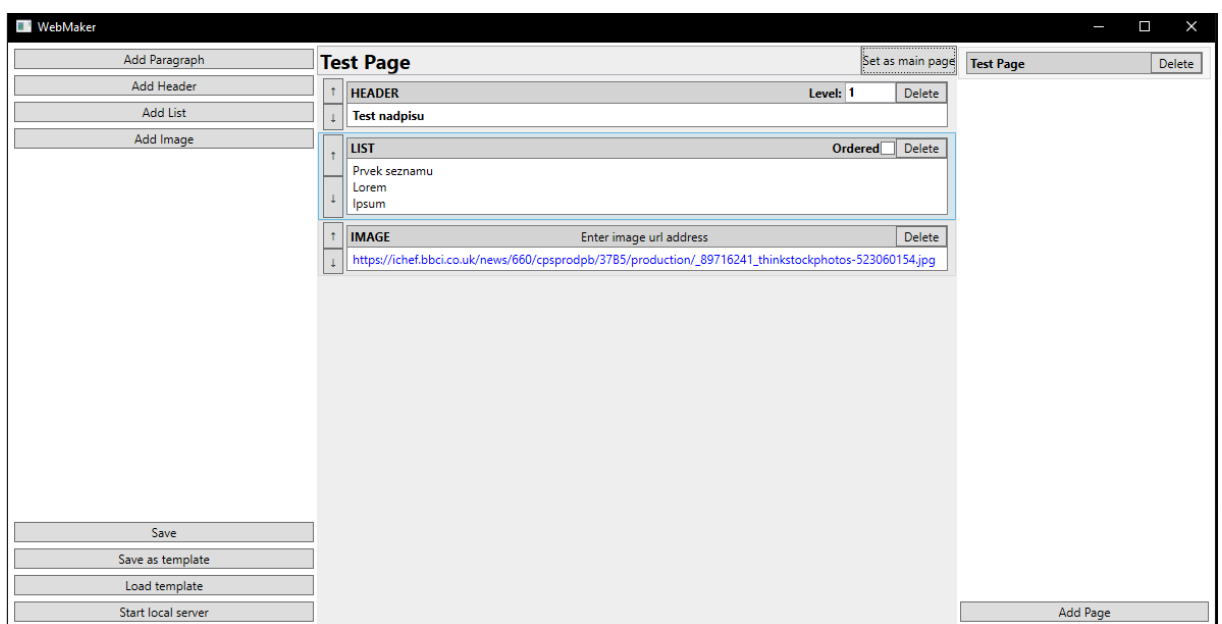
1. Nástrojová část
2. Hlavní část
3. Menu stránek

Menu stránek obsahuje seznam všech stránek tvořeného webu. Pro úpravu jednotlivých stránek se jednoduše klikne na položku v seznam. Každá položka také obsahuje „Delete“ tlačítko pro smazání. Nakonec ve spodní části menu stránek najdeme tlačítko „Add Page“ pro přidání nové stránky.

Hlavní část slouží k úpravě stránek. Nahoře najdeme pole pro úpravu titulku a tlačítko pro nastavení stránky jako hlavní (hlavní stránka má titulek v menu stránek vyznačený tučným textem). Dále zde najdeme seznam ovládacích prvků pro jednotlivé elementy na zvolené stránce. Každý typ elementu má i svůj vlastní typ ovládacího prvku.

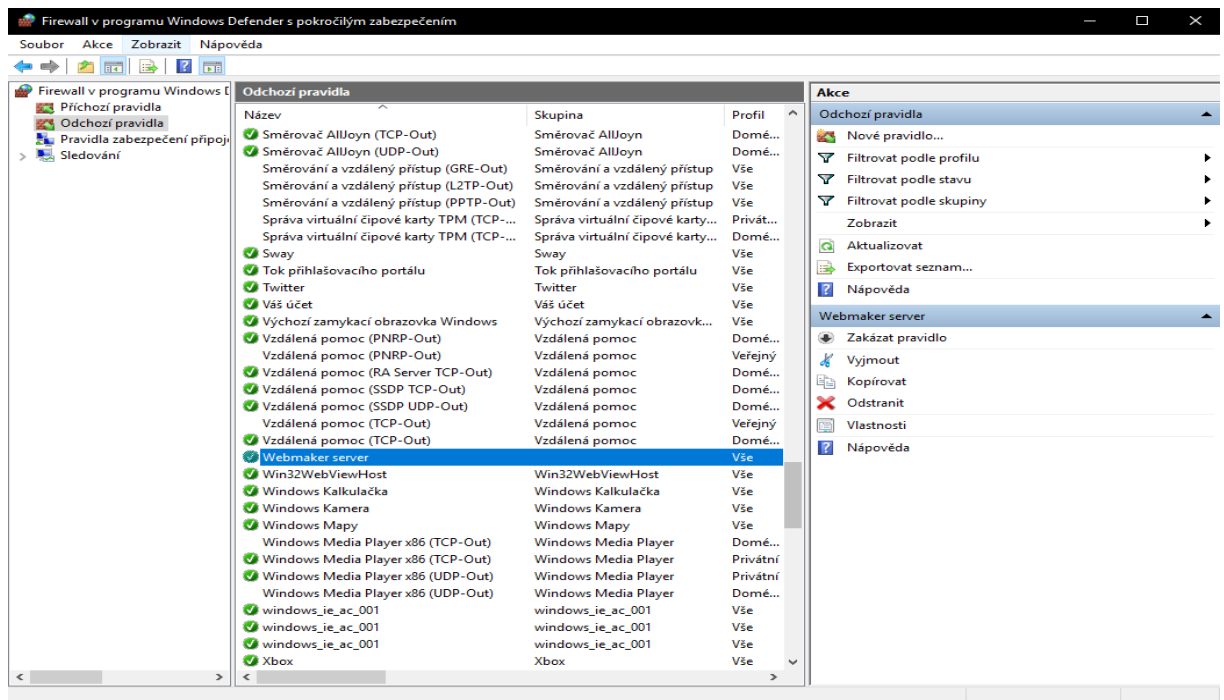
Nástrojová část je rozdělena do dvou dalších oddílů. Horní oddíl je určen k přidávání nových elementů jednotlivých typů. Obsahuje tlačítka „Add Paragraph“, „Add Header“, „Add List“, „Add Image“, sloužící k přidání příslušných typů elementů: odstavec, nadpis, odrážkový seznam, obrázek. Spodní oddíl slouží k ovládání celkového rozhraní. Najdeme zde tyto tlačítka:

- „Save“
 - Otevře dialog pro volbu složku a následně do této složky uloží vytvořený web jako html soubory.
- „Save as template“
 - Uloží web jako šablonu
- „Load template“
 - Načte šablonu s prázdnými elementy
- „Start local server“
 - Spustí lokální server, pomocí kterého si můžeme web prohlédnout bez potřeby ukládání jako html souborů. Po spuštění stačí do prohlížeče napsat „localhost“ a zobrazí se hlavní webová stránka



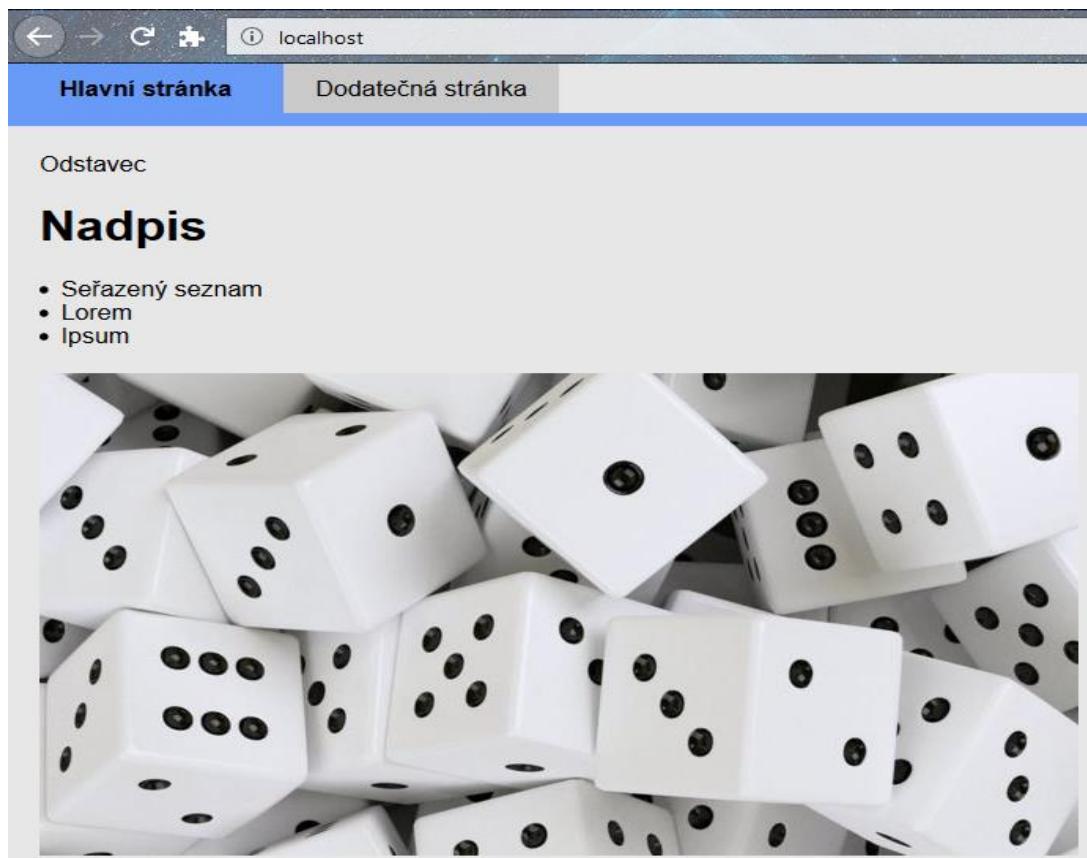
Obrázek 17: WebMaker UI, první verze

Webmaker je program, který byl naprogramován bez schválení standardů a různých bezpečnostních opatření – autoři nejsou samostatná firma, která je autorizovaná k vývoji programů na síti, takže je nutně pro funkci povolit tvorbu webových stránek v programu webmaker na straně administrátora, který bude stránku vytvářet a zveřejňovat jako localhost v síti.



Obrázek 18: Webmaker pravidlo ve Firewallu

Po nahrání a povolení již stačí, aby uživatel v síti k vašemu souboru přistoupil za vložení ip adresy administrátora, který server zapíná.



Obrázek 19: ukázka výtvaru

4 ZÁVĚR

Program jsme začali psát minulý rok v březnu a od té doby jsme navyšovali laťku cílům tak, aby program nebyl dobrý standard nejen pro začátečníka, ale i pro nás, kteří jsme již s problematikou seznámeni, a byl nám nápomocný.

Aplikace stále bude mít svůj vývoj na online uložišti, jelikož technologie HTML jde dopředu, tak stejně i další prvky, které na naše stránky budeme nadále přidávat a vyvíjet, aby aplikace zůstala co nejaktuálnější.

5 SHRnutí

V průběhu práce na programu jsme se setkali s několika komplikacemi, největší z nich byla změna našeho grafického prostředí, a tudíž přesunutí celého kódu na platformu WPF. Toto rozhodnutí jsme učinili již s nějakou prací ve WinForm hotovou, ale nebyli jsme spokojeni s možnostmi nabízených naší první volbou.

Drasticky větší možnosti se nám otevřely po úspěšném zprovoznění online databáze, ve které jsme ověřili zároveň i funkčnost šablon a splnili si i tento cíl.

Po zprovoznění se nám podařilo vytvořit aplikaci, která naplňuje naše cíle, a jsme spokojeni za webové stránky, které jsou z programu vytvořeny, jelikož splňují HTML kritéria, která jsou veřejně uznaná webem a prošli testy syntaxe na stránkách w3schools.

Summary

We began to write this app last March and since that time we made our ambitions higher in way, program wouldn't serve only beginners, but also us, ones, who are acknowledged in matter of writing web.

During our writings we encountered few troubles, largest of them being a change of our UI, therefore we had to take our whole work and move it to the different platform, type the code again. This decision was made on base of being unsatisfied with possibilities offered by WinForm.

Greatest step ahead was making our database online finally working and with that came also our template feature.

After turning it on, we finally made an app, which is fulfilment of our ambitions and we can be satisfied by our web pages, with passing the w3schools tests.

6 SEZNAM OBRÁZKŮ

Obrázek 1: logo aplikace	4
Obrázek 2: ovládání IDE	5
Obrázek 3: tvorba sql databáze ve Visual studiu	6
Obrázek 4: WPF a jeho XML syntaxe.....	7
Obrázek 5: tvorba vlastního pozadí s panelem v C# (ohraničení).....	9
Obrázek 6: panel, který překrývá základní prvek	9
Obrázek 7: kód, který zaznamenává pozici kurzoru.....	10
Obrázek 8: schéma MVVM dostupné na https://www.journaldev.com/20292/android-mvvm-design-pattern	10
Obrázek 9: první verze offline rozhraní.....	11
Obrázek 10: rozhraní databáze, navázání spojení.....	11
Obrázek 11: ukázka prvků a porovnání online a offline databáze.....	12
Obrázek 12: okno dialogu.....	13
Obrázek 13: formulář pro vkládání prvku do databáze	13
Obrázek 14: kód pro rozpoznání cesty souboru z okna dialogu	14
Obrázek 15: GitHub uložení projektu	14
Obrázek 16: verze; změny mezi verzemi.....	15
Obrázek 17: WebMaker UI, první verze	16
Obrázek 18: Webmaker pravidlo ve Firewallu.....	17
Obrázek 19: ukázka výtvaru	17

7 VYUŽITÉ ZDROJE

1. Knihovna C# Microsoft [Online] [Cited: 01 07, 2019.]

<https://msdn.microsoft.com/en-us/library>

2. Informace ohledně stylů pro použití na C# UI Academy [Online] [Cited: 01 07, 2019.]

<https://sharpuiacademy.blogspot.com/>

3. Pro Android Dev, článek MVVM architecture May 31 2017 [Online] [Cited: 01 07, 2019.]

<https://proandroiddev.com/mvvm-architecture-viewmodel-and-livedata-part-1-604f50cda1>