

VŠB – Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky  
Katedra informatiky

**Absolvování individuální odborné praxe**  
**Individual Professional Practice in the**  
**Company**

## Zadání bakalářské práce

Student: **Adam Albert**

Studijní program: B2647 Informační a komunikační technologie

Studijní obor: 2612R025 Informatika a výpočetní technika

Téma: **Absolvování individuální odborné praxe**  
**Individual Professional Practice in the Company**

Jazyk vypracování: čeština

Zásady pro vypracování:

1. Student vykoná individuální praxi ve firmě: Digital Wizards Group s.r.o.
2. Struktura závěrečné zprávy:
  - a) Popis odborného zaměření firmy, u které student vykonal odbornou praxi a popis pracovního zařazení studenta.
  - b) Seznam úkolů zadaných studentovi v průběhu odborné praxe s vyjádřením jejich časové náročnosti.
  - c) Zvolený postup řešení zadaných úkolů.
  - d) Teoretické a praktické znalosti a dovednosti získané v průběhu studia uplatněné studentem v průběhu odborné praxe.
  - e) Znalosti či dovednosti scházející studentovi v průběhu odborné praxe.
  - f) Dosažené výsledky v průběhu odborné praxe a její celkové zhodnocení.

Seznam doporučené odborné literatury:

Podle pokynů konzultanta, který vede odbornou praxi studenta.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Michal Radecký, Ph.D.**

Konzultant bakalářské práce: Bc. Aleš Rosa

Datum zadání: 01.09.2015

Datum odevzdání: 28.04.2017



doc. Dr. Ing. Eduard Sojka  
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.  
děkan fakulty

Prohlašuji, že jsem tuto bakalářskou/diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě, 30.3. 2017

*Albert*  
.....

Souhlasím se zveřejněním této bakalářské/diplomové práce dle požadavků čl. 26, odst. 9 Studijního a zkušebního řádu pro studium v bakalářských/magisterských programech VŠB-TU Ostrava.

V Ostravě, 30.3. 2017



Digital Wizards Group s.r.o.  
708 00 Ostrava - Poruba, Spolu 835/2  
+420 775 940 635 • www.dwgroup.cz  
IČ: 03496996 DIČ: CZ03496996

*Rozm*

Rád bych poděkoval společnosti Digital Wizards Group s.r.o., která mi umožnila tuto praxi, a všem, se kterými jsem během praxe spolupracoval.

## **Abstrakt**

Cílem této bakalářské praxe je popis absolvování odborné praxe ve firmě Digital Wizards Group s.r.o., ve které jsem působil jako frontend vývojář webových stránek. Náplní této praxe byl vývoj webových stránek v základních jazycích HTML a CSS a poté vývoj pomocí moderních nástrojů jako October CMS nebo frontend framework Materialize.

Nejdříve jsem se zaměřil na firmu samotnou a na pozici, kterou jsem zastával, dále jsem popsal jednotlivé projekty, na jejichž vývoji jsem se podílel. V závěru práce jsem praxi zhodnotil a shrnul znalosti a dovednosti, které jsem získal, a které mi scházely.

**Klíčová slova:** Odborná praxe, Webové stránky, Frontend, Framework, Digital Wizards Group s.r.o.

## **Abstract**

The object of this bachelor thesis is a description of professional practice I performed in company Digital Wizards Group s.r.o., where I worked as a frontend website developer. Main content of my practice was website development in basic languages HTML and CSS and later development with help of modern tool such as October CMS or Materialize frontend framework.

First, I focused on the company itself and the position that I held then I described individual projects, whose development I participated in. In conclusion, I reviewed the practice summed up the skills and knowledge that I gained and skills and knowledge that I missed.

**Key Words:** Professional practice, Websites, Frontend, Framework, Digital Wizards Group s.r.o.

# Obsah

<b>Seznam použitých zkratk a symbolů</b>	<b>9</b>
<b>Seznam obrázků</b>	<b>10</b>
<b>1 Úvod</b>	<b>12</b>
<b>2 Představení firmy a popis pracovního zařazení</b>	<b>13</b>
2.1 Popis odborného zaměření firmy . . . . .	13
2.2 Pracovní zařazení . . . . .	13
<b>3 Zadané úkoly v průběhu praxe</b>	<b>14</b>
3.1 Cestovní kancelář Fischer . . . . .	14
3.2 Vývoj webu Zajdisi . . . . .	14
3.3 Revitalizace webu pro střelnici Ostrava - Stará Bělá . . . . .	14
<b>4 Cestovní kancelář Fischer</b>	<b>15</b>
4.1 Zadání pro práci . . . . .	15
4.2 Vývojové prostředí . . . . .	15
4.3 Chrome Developer Tool . . . . .	16
4.4 Shrnutí vývoje . . . . .	16
<b>5 Vývoj webu Zajdisi</b>	<b>17</b>
5.1 Zadání pro práci . . . . .	18
5.2 Verzovací systém . . . . .	18
5.3 October CMS . . . . .	19
5.4 CSS preprocesor . . . . .	21
5.5 jQuery plugin pro rezervaci . . . . .	22
5.6 Shrnutí vývoje . . . . .	24
<b>6 Revitalizace webu pro střelnici Ostrava - Stará Bělá</b>	<b>27</b>
6.1 Zadání pro práci . . . . .	27
6.2 Framework Materialize . . . . .	28
6.3 Shrnutí vývoje . . . . .	29
<b>7 Závěr</b>	<b>30</b>
7.1 Teoretické a praktické znalosti a dovednosti získané v průběhu průběhu studia uplatněné v průběhu odborné praxe . . . . .	30
7.2 Znalosti či dovednosti scházející studentovi v průběhu praxe . . . . .	30
7.3 Dosažené výsledky v průběhu praxe a zhodnocení . . . . .	30





## Seznam použitých zkratk a symbolů

DWG	– Digital Wizards Group s.r.o.
SEO	– Search Engine Optimization
JS	– JavaScript
HTML	– Hyper Text Markup Language
CMS	– Content Management System
CDN	– Content Distribution Network
AJAX	– Asynchronous JavaScript and XML
DOM	– Document Object Model
PHP	– Hypertext Preprocessor

## Seznam obrázků

1	Headrer stránky cestovní kanceláře Fischer . . . . .	16
2	Ukázka stránek Zajdisi . . . . .	17
3	Aplikace pro verzování projektu SourceTree . . . . .	18
4	Ukázka pluginu v režimu pro návštěvníky . . . . .	23
5	Ukázka konzole Gulpu s kompilací SCSS souborů . . . . .	26
6	Ukázka práce s nástrojem Adobe Assets . . . . .	26
7	Ukázka vzhledu stránky . . . . .	27
8	Ukázka responzivní mřížky . . . . .	28

## Seznam výpisů zdrojového kódu

1	Využití rozšíření Emmet . . . . .	15
2	Layout stránky vytvořený pomocí October CMS . . . . .	19
3	Část pluginu v PHP pro snadné vkládání obrázků do HTML a CSS . . . . .	20
4	Vkládání obrázků pomocí pluginu . . . . .	21
5	Příklad proměnných v jazyce SASS . . . . .	21
6	Syntaxe zanoření v SASS . . . . .	21
7	Rozdíl mezi JS a jquery . . . . .	22
8	Kód responzivní mřížky ukázky . . . . .	29

# 1 Úvod

Ve své Bakalářské práci se budu zabývat individuální praxí, kterou jsem absolvoval ve společnosti Digital Wizards Group s.r.o. Příležitost k této praxi jsem získal prostřednictvím svého známého, který pracoval ve firmě na pozici senior programátora. Dostal jsem nabídku zapojit se do nového týmu, který by testoval nejnovější technologie pro tvorbu webových stránek. Zkušenosti s kódováním webových stránek jsem měl na začátku jen malou, proto jsem praxi chtěl využít k získání informací vhodných pro uplatnění v tomto oboru. Požadavek firmy na mne byla jen zodpovědnost a ochota se učit se novým věcem. Tato bakalářská práce popisuje získávání znalostí pro vývoj moderních webových stránek.

## 2 Představení firmy a popis pracovního zařazení

### 2.1 Popis odborného zaměření firmy

Firma Digital Wizards Group s.r.o. (dále jen DWG) se sídlem v Ostravě má tři základní zaměření. Vývoj webových stránek a aplikací, vytváření e-shopů na míru a marketing.

DWG pracuje s nejmodernějšími technologiemi ve svém oboru, ale také si vyvíjí své vlastní, jako redakční systém Woodie<sup>2</sup>, který je založen na open-source CMS platformě October nebo Platforma Rocketoo pro e-shopy, která usnadňuje tvorbu pomocí direct marketingu, napojování na platební brány, jazykové verze a další funkce.

Pomocí marketingu firma spojuje svůj vývoj s další spoluprací DWG a klientů. Nabízí analýzu webů, návrhy pro nové nebo úpravy stávajících a po vytvoření webu nebo e-shopu jeho správu tvoření obsahu, SEO a další prvky nezbytné pro úspěšný chod produktu.

### 2.2 Pracovní zařazení

Po domluvě s kolegou, který mi praxi nabídl, a po krátkém informačním pohovoru jsem byl na praxi oficiálně přijat. Kolega, se kterým jsem sdílel kancelář, byl tedy nejen můj konzultant, ale i vedoucí naší skupiny a mohl mi tedy rychle poskytnout odbornou radu nebo mi dát zpětnou vazbu.

V týmu, ve kterém jsem zastával pozici frontend vývojáře, byl kromě mého dohlížejícího kolegy zkušeného frontend/backend vývojáře také další student, který se zaměřil na backend. S ním jsem poté úzce spolupracoval, při vývoji webových stránek.

Cílem mé praxe bylo získání co nejvíce zkušeností pro pozdější spolupráci v týmu i po absolvování praxe za použití moderních technologií a efektivních pracovních postupů pod dohledem odborného vývojáře.

## **3 Zadané úkoly v průběhu praxe**

### **3.1 Cestovní kancelář Fischer**

Mé zkušenosti s vývojem webových stránek končily na střední škole, kde jsem dokázal vytvořit jednoduché webové HTML stránky, kterým jsem upravil vzhled pomocí jednoduchého CSS. Zapojením osoby s těmito schopnostmi do práce na moderních technologiích by vedla ke značnému zdržení projektu a k potencionálním chybám. Pro osvěžení znalostí HTML a CSS jsem využil projekt pro cestovní kancelář Fischer, na kterém v té době v DWG pracovali, kde jsem vytvořil hlavičku a patičku stránek.

### **3.2 Vývoj webu Zajdisi**

Po úspěšném splnění prvního úkolu jsme zahájili práci na našem prvním týmovém úkolu pro partnerskou firmu zabývající se hlavně marketingem, ale i vývojem jednoduchých webů.

Na svůj projekt Zajdisi však nestačili, proto ho předali na vývoj nám. Zajdisi byl projekt s nízkým rozpočtem a nebylo třeba s ním spěchat, protože zadavatelé sami ještě nebyli plně rozhodnutí, co všechno web bude obsahovat. Projekt byl pro náš tým ideální, protože jsme se na něm mohli naučit nové věci a vyzkoušet nové technologie.

### **3.3 Revitalizace webu pro střelnici Ostrava - Stará Bělá**

Třetí úkol mé praxe byl vývoj webových stránek, pomocí frontend frameworku Materialize. V tomto projektu jsem dostal za úkol nejen vytvořit frontend stránek, ale také navrhnout jejich design.

V tomto projektu jsem chtěl využít funkcionalitu frameworku na maximum a omezit vlastní CSS kód na nezbytné minimum.

## 4 Cestovní kancelář Fischer

### 4.1 Zadání pro práci

Prvním úkolem mé bakalářské praxe bylo vytvoření headru a footeru stránek cestovní kanceláře Fischer. Jednalo se o jednoduchý HTML kód s připojeným CSS bez žádného využití JS ani žádného frameworku. Tento úkol byl pro mne vhodný, protože jsem si na něm mohl osvěžit své znalosti HTML a CSS kódu.

### 4.2 Vývojové prostředí

Na radu svého kolegy jsem si stáhl a nainstaloval open source editor Brackets, který podporuje širokou škálu rozšíření, ulehčující práci a tedy i proces učení se. [3]

Základní pluginy, které jsem využil byly:

- Beutify - rozšíření, které formátuje námi napsaný HTML, CSS nebo JS kód. Při procesu učení jsem totiž velice často různě přepisoval kód a neustálé stylování do přehledného kódu mou práci jen zdržovalo.
- File Icons - toto rozšíření, podporující velké množství typů souborů, přidává ikony k souborům ve stromu projektu, a tím ulehčuje orientaci v projektu
- Quick multicursor selection - rozšíření umožňující nám ovládat kurzor a výběr textu v editoru na více místech a upravovat najednou více částí kódu. Velice užitečné například při upravování všech odrážek ve velice dlouhém HTML seznamu
- Emmet - rozšíření Emmet mi velice ulehčovalo práci díky používání zkratk HTML tagů, které přetransformoval v plnohodnotný HTML kód, např.:

---

```
1 ul#nav>li.item$*4>a{Item $}
2
3 <ul id="nav">
4     <li class="item1"><a href="">Item 1</a></li>
5     <li class="item2"><a href="">Item 2</a></li>
6     <li class="item3"><a href="">Item 3</a></li>
7     <li class="item4"><a href="">Item 4</a></li>
8 </ul>
```

---

Výpis 1: Využití rozšíření Emmet

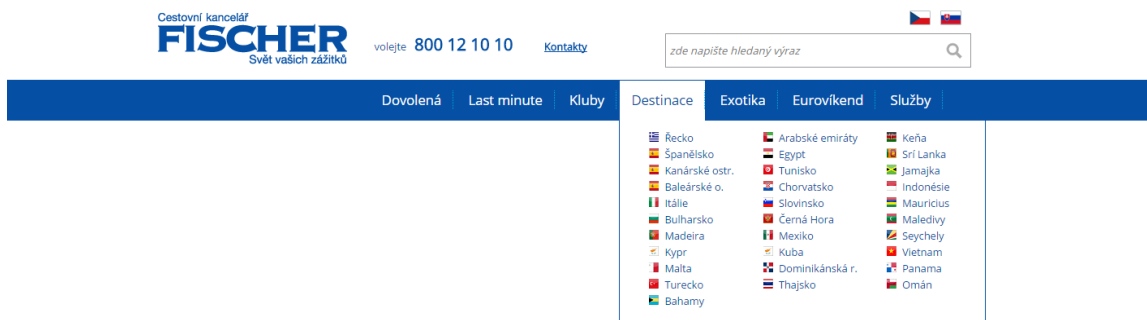
### 4.3 Chrome Developer Tool

K práci na headru a footeru stránky jsem nedostal žádnou grafickou předlohu. Vzhled jsem měl napodobit pomocí nástroje pro vývojáře v prohlížeči Google Chrome (dále jen DevTool). Během mé praxe se DevTool stal velice užitečným nástrojem, protože jsem díky němu efektivně debugoval a do hloubky zkoumal kód, na kterém pracoval.

Při práci na tomto úkolu jsem pomocí DevToolu inspektoval jednotlivé elementy na oficiální stránce Fischeru, prozkoumal jejich CSS a napodobil prvek ve svém HTML a CSS kódu.

### 4.4 Shrnutí vývoje

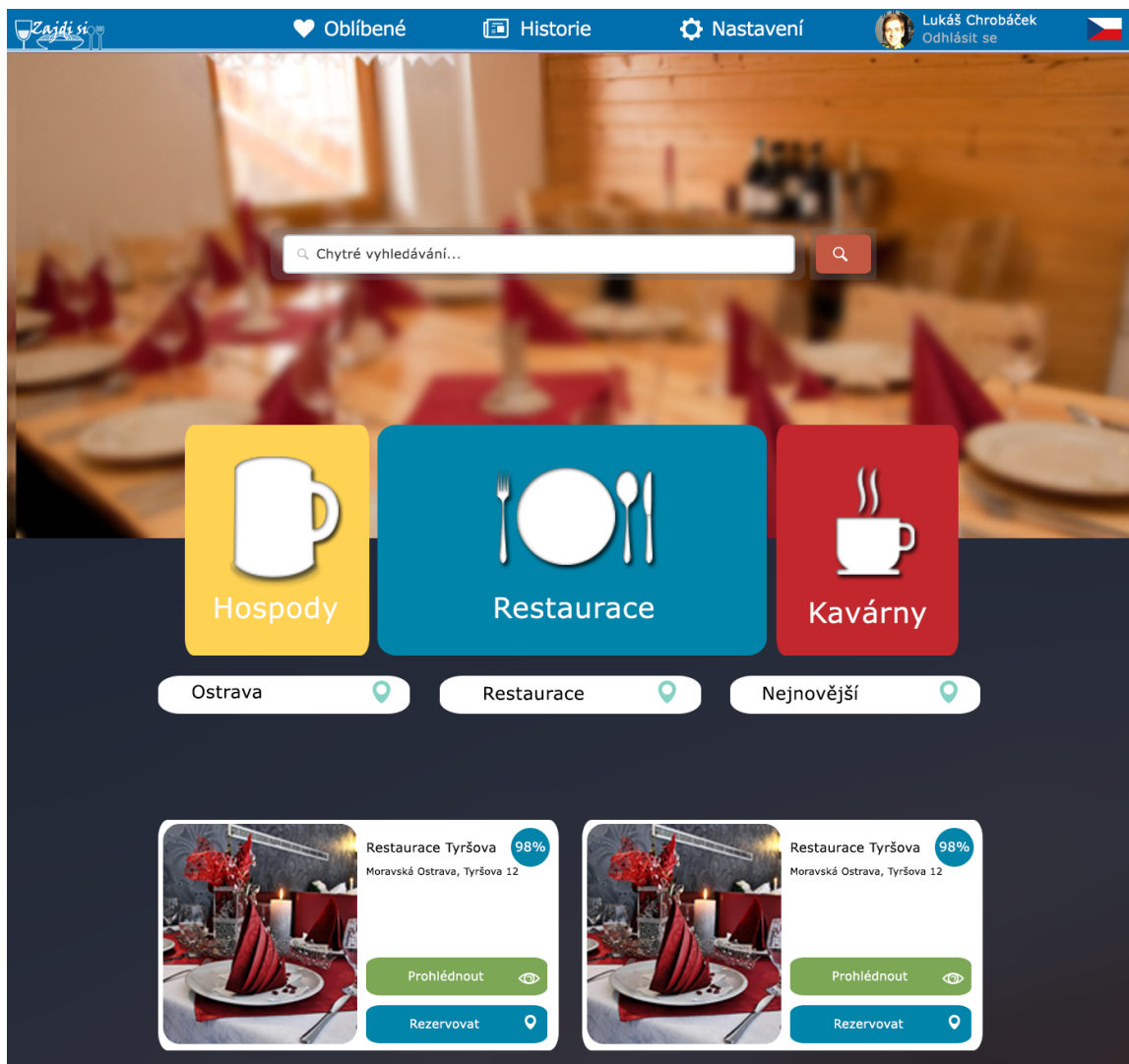
S optimálně nastaveným vývojovým prostředím a po seznámení se s Chrome DevTool jsem začal kódování headru a footeru. HTML kód obsahoval standardní tagy, skládal se z <ul> seznamů a jednoduchých elementů selektovaných pomocí tříd a id. U CSS jsem poprvé narazil na selektor :hover a na klauzuli @fontface pro nastavování fontů písma.



Obrázek 1: Header stránky cestovní kanceláře Fischer



## 5 Vývoj webu Zajdisi



Obrázek 2: Ukázka stránek Zajdisi

Cílem mého druhého projektu byl vývoj webových stránek pro partnerskou firmu podle grafického návrhu. Jednalo se o netriviální webové stránky, na kterých jsem si vyzkoušel práci s technologiemi a postupy pro tvorbu moderních webových stránek.

Zajdisi je webová stránka zobrazující gastronomické podniky. Podniky mohly být filtrovány podle pozice uživatele. Každý podnik má svůj profil, kde se nachází jeho prezentace, nabídka, hodnocení uživatelů, informace o podniku a také možnost rezervace stolu v podniku.

## 5.1 Zadání pro práci

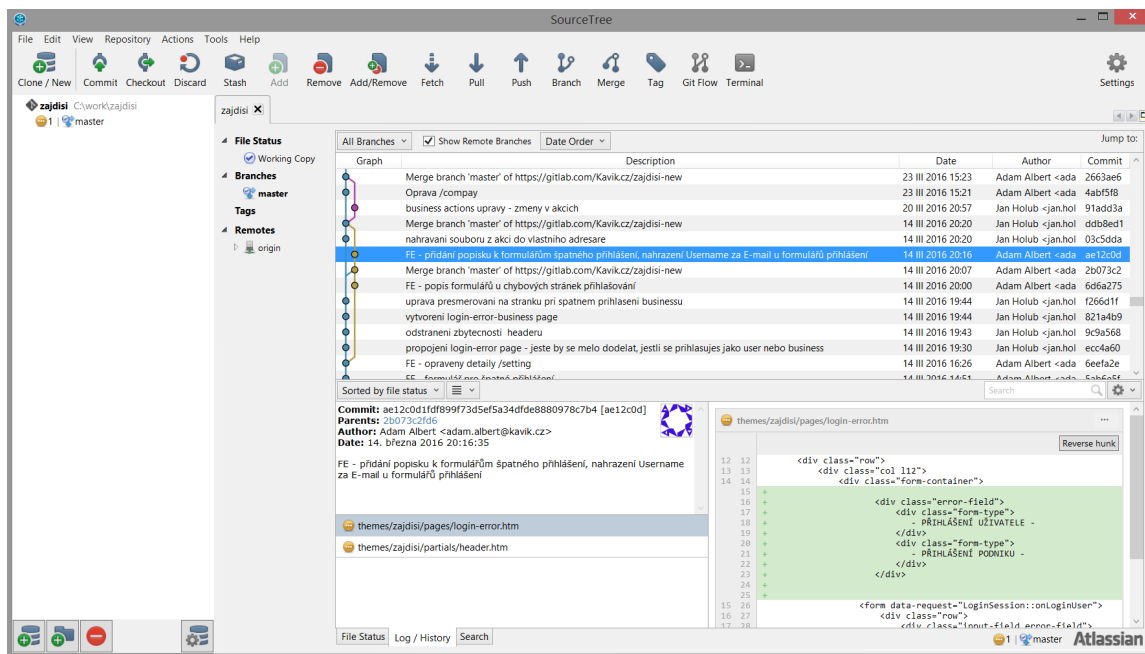
Při mém druhém projektu praxe jsem dostal za úkol vytvořit frontend pro webové stránky Zajdisi a dále spolupracovat s kolegou, který bude vyvíjet backend stránek na úpravách potřebných pro jeho práci. K dispozici jsem dostal .psd soubory s grafickým náhledem, který si firma sama vytvořila podle svých představ, dále jsem navázal kontakt s osobou odpovídající za návrh stránek, kterou jsem kontaktoval v případě dotazu na funkčnost nebo při připomínce či návrhu na úpravu stránek.

Jednou z chyb, kterých jsme se při vývoji dopustili, bylo, že jsme se s firmou nenapsali pevně stanovené zadání. To bylo způsobeno tím, že sama firma si před vývojem neujasnila, co všechno od stránek očekává, protože myšlenka projektu se stále vyvíjela.

## 5.2 Verzovací systém

Předtím, než jsem začal pracovat na samotných stránkách, jsem si stáhl klientskou aplikaci pro práci s webovým repozitářem Gitlab[4], se kterým jsem verzoval projekt. Při práci na netriviálních aplikacích nebo při práci v týmu zabraňuje nechtěnému přepsání částí kódu někoho jiného, dělí nám projekt na elementární části, které mohou být pojaty jako jednotlivé úkoly, a umožňuje vrátit projekt do některé předchozích verzí.

Během své praxe jsem používal aplikaci SourceTree, která mi umožnila v uživatelsky přívětivém prostředí přehlednou orientaci v projektu. Před prací jsem si stáhl z Gitlabu aktuální verzi projektu a během práce jsem po dokončení hotových částí kódu provedl *commit* (vytvoření její aktuální verze projektu,) a ten poté odeslal na GitLab server.



Obrázek 3: Aplikace pro verzování projektu SourceTree

## 5.3 October CMS

Asi nejdůležitější technologie, kterou jsme chtěli vyzkoušet na našem projektu, byl v té době docela nový systém pro správu obsahu October (dále jen October CMS). October CMS je open source platforma založená na Laravel PHP frameworku.

### 5.3.1 Šablony v October CMS

Pro mě, jako frontend vývojáře, byla asi největší výhoda October CMS vytváření *template* (šablon), díky kterým jsem si mohl jednoduše rozvrhnout stránky webu do menších částí, nacházející se ve samostatných souborech.

Nejzákladnější práce se šablonami spočívala ve vytvoření layoutu stránek, do kterých jsem potom vkládal jednotlivé *partials* (části kódu). V praxi to vypadalo tak, že jsem si vytvořil základní stránku, která obsahovala header a body HTML stránky. V headru byly všechny věci náležící do této části stránky, od meta dat přes CDN linky fontů, až po připojení CSS souborů.

---

```
1 <html lang="cs">
2 <head>
3
4   <meta name="Adam Albert">
5   <title>Ukazkovy prikklad layoutu stranky v October CMS</title>
6   <link href="https://fonts.googleapis.com/css?family=Comfortaa" rel="stylesheet">
7   <link href="{ ['assets/css/style.css']|theme }" rel="stylesheet">
8   <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/2.1.4/jquery.min.js"></script>
9 </head>
10 <body>
11
12   {% partial "header" %}
13   {% page %}
14   {% partial "footer" %}
15
16 </body>
17 </html>
```

---

Výpis 2: Layout stránky vytvořený pomocí October CMS

V příkladu vidíme vkládání headru stránky a footeru stránky pomocí *partial*. Jelikož tento kód je na všech stránkách stejný, nemusíme jej tedy ručně vkládat na každou stránku. V *page* se pak bude dosazovat jen obsah stránek, který se nachází v samostatných HTML souborech.

### 5.3.2 Pluginy v October CMS

Pluginy jsou jedna z největších výhod October CMS. Díky jeho modularitě jej můžeme rozšířit o novou funkcionalitu pomocí pluginu, který si sami napíšeme nebo můžeme využít pluginy, které naprogramovali lidé pracující s tímto frameworkem a dali je jako open source program na stránky Octoberu. Může se jednat o jednoduché pluginy zdarma, například pro automatické mazání cache paměti, kterou si October ukládá, a občas dělala problémy nebo komplexní, zpoplatněný plugin, který například nahradí celý backend stránek.[2]

Já jsem využíval plugin, který mi nakódoval můj kolega starající se o backend stránek. Plugin nám tvořil filtr pro pohodlné vkládání obrázků.

---

```
1 public function registerMarkupTags()
2     {
3         return [
4             'filters' => [
5                 // A local method, i.e $this->images()
6                 'images' => [$this, 'images']
7             ],
8         ];
9     }
10 public function images($imageName)
11     {
12         $theme = Theme::getActiveTheme();
13         $themeDirName = $theme->getDirName();
14         $imageUrl = '/themes/' . $themeDirName . '/assets/css/images/' . $imageName;
15         return $imageUrl;
16     }
17 }
```

---

Výpis 3: Část pluginu v PHP pro snadné vkládání obrázků do HTML a CSS

Tento plugin nám zajišťoval, že jsme nemuseli pokaždé psát složitou cestu k umístění obrázků, ale pomocí Octoberu jsme přidali k názvu souboru filtr obsažený v pluginu.

V HTML kódu potom vypadalo vkládání obrázků takto.

```
1 <!-- Pridavani obrazku do scss -->
2
3 background: url("#{images}/nazev_obrazku.jpg");
4
5 <!-- Pridavani obrazku do html -->
6
7 
```

Výpis 4: Vkládání obrázků pomocí pluginu

## 5.4 CSS preprocessor

LESS, SASS a další preprocessory frontend vývojářům zjednodušují práci s CSS. Jedná se o jazyky postavené nad CSS. Přidávají do něj nové vlastnosti nebo zjednodušují organizaci kódu. Už na vývojářově stanici jsou pak kompilovány do CSS, aby kódu rozuměly prohlížeče.

Ač je nováčkům v oblasti kódování pomocí preprocessorů doporučeno začít s LESS preprocesorem, na doporučení mého kolegy jsem použil preprocessor SASS, který zdánlivě připomíná syntaxi Javascriptu, jenž také využívám při tvorbě webových stránek.[1]

Během kódování v SASS jsem nejvíce využíval těchto vlastností:

- Proměnné - Vlastnost známá z imperativních jazyků je velice užitečná například při hromadné změně určité hodnoty v celém CSS, například při změně barvy všech prvků, které využívají stejnou proměnnou. Při definici proměnnou změním a její hodnota se změní všude, kde ji využívám.

```
1 $font-stack: Helvetica, sans-serif;
2 $primary-color: #333;
3
4 body {
5   font: 100% $font-stack;
6   color: $primary-color;
7 }
```

Výpis 5: Příklad proměnných v jazyce SASS

- Zanořování - slouží k lehčí orientaci v kódu a pro jednodušší zápis například dědičnosti elementů

```
1 nav {
2   ul {
3     margin: 0;
4     padding: 0;
5     list-style: none;
6   }
7 }
```

```
8   li { display: inline-block; }
9
10  a {
11    display: block;
12    padding: 6px 12px;
13    text-decoration: none;
14  }
15 }
```

---

Výpis 6: Syntaxe zanoření v SASS

- Import - preprocesorový soubor můžeme pro přehlednost rozdělit na nezávislé části, například na CSS pro jednotlivé stránky, a poté je importovat do jednoho souboru, ze kterého se vykompiluje výsledné CSS všech stránek.

## 5.5 jQuery plugin pro rezervaci

jQuery je rychlá, malá a na funkčnost bohatá knihovna Javascriptu. Úkolem jQuery je ulehčit práci s Javascriptem na našich webových stránkách. jQuery nabízí mnoho funkcí, které v Javascriptu zabírají mnoho řádků, nebo jsou složité, zaobalí je do metod, které potom voláme jediným řádkem. Ulehčuje také práci při práci s AJAX nebo při manipulaci s DOM.[7]

---

```
1  /* -jQuery- */
2  $( 'body' ) .css ( 'background', '#ccc' );
3
4  /* -JavaScript- */
5  Function changeBackground(color) {
6
7      Document.body.style.background = color;
8
9  }
10
11 Onload="changeBackground ('red');"
```

---

Výpis 7: Rozdíl mezi JS a jQuery

Úkolem bylo vytvořit jQuery plugin, který by obsahoval mapu podniku tvořenou rastrem čtverců (buněk). Počet těchto buněk byl na výšku a šířku stejný, a proto tvořily také čtverec, mapa mohla obsahovat více rastrů, představující podlaží v podniku. Plugin měl pracovat ve dvou režimech rozdílných ve funkcionalitě.

V Administrátorském režimu si klient vytvořil mapu svého podniku nastavováním počtu pater a půdorysu pater. Půdorys vytvořil nastavováním buňky na hodnotu stolu a hodnoty stěny pro obvod místnosti nebo prázdného místa. Stoly mohly zabírat více buněk. Buňky stolu ve svém nejbližším okolí sdílely stejné číslo stolu, které si nastavil klient podle sebe, stejně jako počet míst u daného stolu.

V režimu pro návštěvníky je vidět půdorys vytvořený majitelem podniku. Zákazník vyplnil formulář nacházející se na stránce s pluginem, který s pomocí backendu zjistil podle údajů obsazenost stolů a zobrazil na půdorysu, které stoly odpovídají požadavkům návštěvníka (počet míst u stolu) a zbarvil stoly na červeně, pokud by byly v době návštěvy už obsazené nebo zeleně, pokud byly stoly k dispozici.

## Rezervace

Datum rezervace  
23. 03. 2017

---

Od  
12:00

---

Do  
14:00

---

Počet osob  
2

---

Zadejte své jméno a příjmení  
Admin Admin

---

Zadejte své telefonní číslo

---

Zadejte svůj email  
admin

---

VYHLEDAT

**Patro**

1

2

	1	1	1	1					
	2	2	2	2					

Číslo stolu je: 2  
 Kapacita stolu je: 4lidí

Toto je vybraný stůl

Toto je obsazený stůl

Toto je volný stůl

ZAREZEROVAT

Obrázek 4: Ukázka pluginu v režimu pro návštěvníky

## 5.6 Shrnutí vývoje

Po nastavení všech technologií jsem se začal s kódováním svého prvního plnohodnotného webu. Jako začátečník mé pracovní tempo nebylo tak rychlé jako u zkušeného vývojáře. To však nebyl problém, protože na projekt jsme neměli předem vymezený čas a určený deadline, jak již jsem výše zmiňoval, kvůli nedokonalému zadání projektu. S relativně "neomezeným" časem jsem mohl řešit problémy, které se během kódování ukázaly, například z důvodu nezvyku na nové technologie.

### 5.6.1 HTML kód stránek

Kódování HTML stránek byla asi nejjednodušší část při vývoji tohoto webu. Nejdříve jsem se svým kolegou konzultoval, jaký by byl nevhodnější layout pro stránky, prošli jsme a prodiskutovali jednotlivé grafické návrhy stránek a načrtl jsem si jejich rozdělení na menší a jednodušší části.

Nakódoval jsem základní *layout* stránek v October CMS obsahující hlavu stránek s meta informacemi, *stylesheety* a CDN linky pro jQuery atd. V těle stránek jsem vytvořil tag pro *partials* hlavičky, *patičky* stránek a také pro samotný obsah stránek, abych mohl dále kódovat obsah v samostatných souborech.

U samotného kódování jsem použil na doporučení framework Materialize, který budu popisovat níže. To byla chyba, kterou jsem se dopustil z neznalosti. Jelikož jsem neměl dostatek zkušeností s kódováním a prací s frameworkem, nedokázal jsem na začátku posoudit jeho využitelnost na stránkách a dopad, jaký měl při použití. Grafický návrh stránek Zajdisi byl silně zakázkově navržený. To znamená, že u každého elementu jsem musel ručně kódovat jeho vzhled v CSS a generické prvky frameworku jsem takřka nevyužil. Druhá věc, kterou jsem neodhadl, byla využití *gridu* pro responzivní design stránek frameworku. Tento *grid* jsem neměl vůbec použít, protože stránky měly být samostatně rozděleny na mobilní a desktopovou verzi. Na tyto problémy jsem však přišel, až když jsem měl značnou část stránek hotových a začít znovu mi nepřišlo jako dobrý nápad, proto jsem kód stránek později musel po sobě hodně upravovat, a to vedlo ke značnému zdržení při práci.

### 5.6.2 CSS kód stránek

Grafický návrh stránek Zajdisi byl velmi uživatelsky navržený a jeho stránky obsahovaly mnoho graficky rozličných prvků. Proto jsem si pomocí SASS preprocesoru rozdělil kódování vzhledu tak, aby každý soubor obsahoval specifické elementy jednotlivých stránek. Všechny soubory jsem potom naimportoval do jednoho hlavního SCSS souboru a ten poté kompiloval. Kompilaci stránek mi zajišťoval nástroj gulp.js.[5]

Gulp.js je nástroj, který zastává stále opakující se práci, která programátorovi zabírá zbytečně čas jako:

- generování HTML z šablon



- komprese obrázků
- kompilování SASS do CSS kódu
- minifikace CSS a Javascript kódu

Samotný kód k jednotlivým elementům na stránce jsem poté bral z grafického návrhu v podobě .psd souboru. Jelikož nejsem grafik a s programem Photoshop jsem neměl žádné zkušenosti, používal jsem jednoduchý nástroj Adobe Assets. Do tohoto cloudového uložiště jsem si nahrál všechny grafické návrhy. Assets poté umožňuje prohlížení těchto souborů podrobně po vrstvách a zobrazuje CSS vlastnosti elementů v nich obsažených. Nástroj je velice jednoduchý a efektivní. Po kliknutí na některý element, například tlačítko, Assets zobrazil všechny důležité informace, které jsem potřeboval k nakódování elementu do svého SCSS.[6]

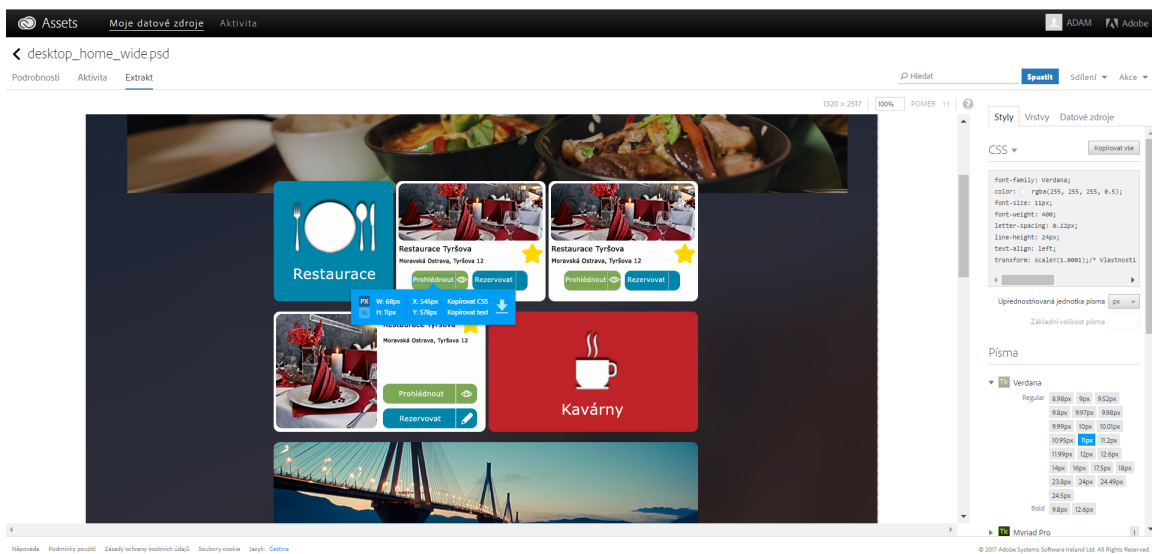
### 5.6.3 Ukončení vývoje

Během práce na tomto projektu jsme vytvořili samotné stránky Zajdisi, administrátorské stránky pro správu webu, registrační stránky pro majitele podniků a administrátorské stránky pro správu podniků. Web se stále rozšiřoval a rozšiřoval a kvůli absenci plnohodnotného zadání nebylo jasné, v jaké fázi bude projekt ukončen. Většinu funkcionalit stránek jsem domýšlel se svým kolegou, který stránce vyvíjel backend a firma kladla další požadavky na rozšíření. To se našemu vedoucímu týmu nelíbilo. Proto jsme dodělali projekt do fáze, o které jsme se zhruba domlouvali na začátku projektu, a projekt jim odevzdali. Projekt Zajdisi byl pozastaven, protože ani zadávající firma nepokračovala ve vývoji, jelikož její zaměření se v té době ubíralo jiným směrem.

Na projektu Zajdisi jsme se však všichni naučili nové věci, a to byl hlavní úkol tohoto projektu. Projektanti se naučili, že pořádné zadání s jasně stanovenými úkoly je pro úspěšný projekt naprostou nezbytností, backendista se naučil pracovat s frameworkem na vývoj backendu v October CMS a já jsem získal bohaté zkušenosti pro vývoj frontendu moderní webové stránky. Na projektu jsem celkem pracoval 38 dní.

```
gulp
INFO: Ports 8600 (database) and 35729 (live-reload) have to be free.
INFO: Press anytime Ctrl+C two times and then any key to restart.
INFO: Please, refresh browser tab after loading.
[15:41:03] Using gulpfile C:\work\zajdisi\gulpfile.js
[15:41:03] Starting 'sass'...
[15:41:03] Finished 'sass' after 20 ms
[15:41:03] Starting 'watch'...
[15:41:06] Finished 'watch' after 2.8 s
[15:41:06] Starting 'default'...
[15:41:06] Finished 'default' after 9.41 us
[15:41:07] C:\work\zajdisi\themes\zajdisi\assets\css\app.css reloaded.
[15:41:07] C:\work\zajdisi\themes\zajdisi\assets\css\admin_app.css reloaded.
[15:41:07] C:\work\zajdisi\themes\zajdisi\assets\css\company_app.css reloaded.
[15:41:08] C:\work\zajdisi\themes\zajdisi\assets\css\company_registration_app.css reloaded.
```

Obrázek 5: Ukázka konzole Gulpu s kompilací SCSS souborů



Obrázek 6: Ukázka práce s nástrojem Adobe Assets

## 6 Revitalizace webu pro střelnici Ostrava - Stará Bělá



Obrázek 7: Ukázka vzhledu stránky

### 6.1 Zadání pro práci

Poslední úkol, který mi byl zadán během mé praxe, bylo vytvořit nové responzivní webové stránky pro známého kolegy ve firmě. Jednalo se o jednodušší projekt, na kterém jsem si mohl vyzkoušet práci s moderním frontendovým frameworkem a práci s responzivitou webu.

Existuje mnoho responzivních CSS frameworků, které nabízí ulehčení práce jak kodéru stránek, tak i designera. Asi nejznámějším takovým frameworkem je Bootstrap, protože kromě HTML a CSS v sobě obsahuje i prvky Javascriptu. Jako doporučení od kolegy jsem však dostal framework Materialize, který je celkem novinkou, a také obsahuje prvky Javascriptu a podporuje využití CSS preprocesoru SASS.

Klient neměl neměl vlastní grafický návrh a v designu nám nechal volnou ruku. Díky tomu jsem mohl využít plný potenciál frameworku.

## 6.2 Framework Materialize

### 6.2.1 Material design

Metafora Material popisuje vztah mezi pohybem a prostorem. Myšlenkou je, že technologie byla inspirována papírem a inkoustem, které umožňují průchod kreativitě a inovaci.

Prvky a komponenty jakou jsou například mřížky, barvy, typografie, zobrazení nejsou pouze příjemné na pohled, ale zároveň poskytují dojem hierarchie, smyslu a zaměření. Důraz na různé akce a komponenty poskytuje vizuální vodítko.

Pohyb umožňuje spojit si to, co je na obrazovce s něčím ve světě. Poskytnutím jak zpětné vazby, tak obeznámenosti, umožňuje uživatelům se plně ponořit do doposud neznámé technologie.[9]

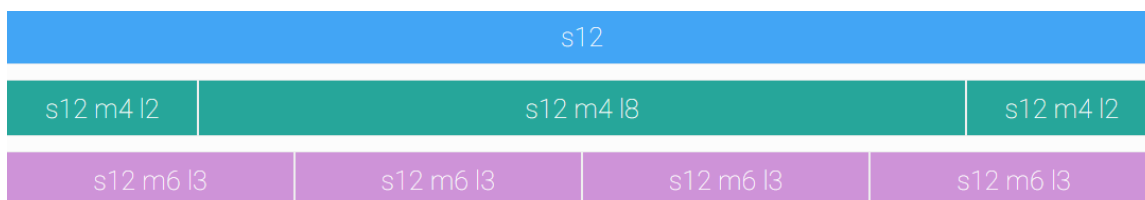
### 6.2.2 Grid

Responzivní design je způsob stylování HTML dokumentu, které zaručí, že zobrazení stránky bude optimalizováno pro všechny druhy nejrůznějších zařízení (mobily, notebooky, netbooky, tablety atd.). Především díky vlastnosti Media Queries, která je zahrnuta ve specifikaci CSS3, lze rozpoznat vlastnosti zařízení, na kterém je stránka prohlížena, a přizpůsobit tak samotnou stránku a její obsah.[10]

Responzivní mřížka (dále jen grid) je nástroj pro tvoření responzivity. HTML stránka je nejdříve rozdělena na 12 sloupců. Tyto sloupce mají proměnnou velikost v závislosti na šířce obrazovky zařízení, na kterém je stránka zobrazena. Podle těchto sloupců je potom určována šířka prvků, které jsou v nich obsaženy. Šířka prvků se mění se šířkou sloupců.

Šířky sloupců mají 3 základní kategorie.

- small - grid pro malé obrazovky (většinou mobilní telefony), které mají šířku displeje do 600px, prefix sloupce *s*
- medium - grid pro střední obrazovky (většinou tablety), které mají šířku displeje mezi 600px a 992px, prefix sloupce *m*
- large - grid pro střední obrazovky (obrazovky osobních počítačů), které mají šířku displeje větší než 992px, prefix sloupce *l*



Obrázek 8: Ukázka responzivní mřížky

---

```
1 <div class="row">
2   <div class="col s12"><p>s12</p></div>
3   <div class="col s12 m4 l2"><p>s12 m4</p></div>
4   <div class="col s12 m4 l8"><p>s12 m4</p></div>
5   <div class="col s12 m4 l2"><p>s12 m4</p></div>
6 </div>
7 <div class="row">
8   <div class="col s12 m6 l3"><p>s12 m6 l3</p></div>
9   <div class="col s12 m6 l3"><p>s12 m6 l3</p></div>
10  <div class="col s12 m6 l3"><p>s12 m6 l3</p></div>
11  <div class="col s12 m6 l3"><p>s12 m6 l3</p></div>
12 </div>
```

---

Výpis 8: Kód responzivní mřížky ukázky

### 6.3 Shrnutí vývoje

U tohoto projektu jsem zastával pozici jak frontend vývojáře, tak grafického návrháře. Jelikož jsem chtěl využít co nejvíce možnosti frameworku a pokud možno nepsat vlastní CSS kód, nezdržoval jsem se grafickým návrhem v nějakém editoru (UXpin, Photoshop, atd.), ale vystačil jsem si s obyčejnou tužkou a papírem. Po načrtnutí umístění základních komponent jsem začal s kódováním.

Zkušenosti s Materialize jsem měl už z minulého projektu, nevyužil jsem však jeho responzivní možnosti. Nejdříve jsem vytvořil desktopový vzhled, ze kterého jsem vycházel. Poté jsem odhadem nastavil hodnoty prvků pro zobrazení na malých obrazovkách a otevřel stránku v mobilním režimu Chrome DevToolu. Nedostatky jsem upravil přímo v tomto zobrazení. Zobrazení na středních obrazovkách bylo jen o ladění poměru velikosti v porovnání s malými a velkými obrazovkami.

Práce s frameworkem Materialize byla velice zábavná a intuitivní. Práce s elementy je jednoduchá a kód je čistý a přehledný. Práci na projektu jsem strávil 7 dní a během toho jsem vytvořil responzivní webové stránky a stránky pro administraci obsahu.

## 7 Závěr

### 7.1 Teoretické a praktické znalosti a dovednosti získané v průběhu průběhu studia uplatněné v průběhu odborné praxe

Během absolvování praxe jsem nejvíce využil obecné znalosti informačních technologií, které jsem získal během studia. Mimo to jsem nejvíce ocenil znalosti získané v předmětech, které se zabývali programovacími jazyky, například předměty Programovací jazyky I a II, kde jsem studoval C# a Javu, dále Skriptovací programovací jazyky a jejich aplikace, kde jsem se seznámil s Pythonem. Při návrhu webových stránek jsem využil znalosti získané z předmětu Uživatelská rozhraní.

### 7.2 Znalosti či dovednosti scházející studentovi v průběhu praxe

Znalosti, které mi během praxe chyběly asi nejvíce, nebyly zkušenosti s jQuery nebo s frameworky, ale takzvané *best practices*. Jedná se o soubor metod a pravidel, které se během vývoje webových stránek osvědčily jako nejefektivnější nebo nejsprávnější pro použití. Tyto znalosti se dají nastudovat díky mnoha článkům napsaných na toto téma, ale nejlépe se získají léty vývoje webových stránek během kterých zjistíme, jaké postupy nebo pravidla pro vývoj jsou ta nejlepší.

*Best practices* obsahují technologie, které jsou v dané době nejspolehlivější a nejlépe se s nimi pracuje, ale také jak správně psát kód, jakým věcem se máme vyhýbat při vývoji a které naopak využít maximálně. Jelikož jsem jako nováček neovládal tyto znalosti, dopouštěl jsem se zbytečných chyb, které jsem později po sobě opravoval.

### 7.3 Dosažené výsledky v průběhu praxe a zhodnocení

Absolvovanou praxi ve firmě Wizards Group s.r.o. musím hodnotit jedine kladně. Během dvou semestrů jsem získal tolik znalostí a naučil se tolik nových věcí, které bych samostudiem nikdy nezískal. Na začátku praxe jsem se naučil, jak funguje pracovní proces v týmu obsahující více lidí, jak vypadá reálný projekt moderních webových stránek a postupem času jsem ovládl prostředky a technologie, se kterými se tyto stránky vyvíjí. Dosažené výsledky tedy hodnotím kladně, protože si myslím, že jsem svědomitě splnil veškeré úkoly, které mi byly přiděleny.

Mezi hlavní dovednosti, které jsem získal, patří znalost struktury HTML a CSS stránek, ovládání verzovacích systémů a práce s frameworky. Naučil jsem se pracovat s CSS preprocesory, seznámil jsem se s Javascriptem a jQuery, ve kterém jsem napsal plnohodnotný plugin. Dále jsem získal zkušenosti s moderními frameworky pro vývoj webových stránek.

## Literatura

- [1] Martin Michálek. Vzhůru do CSS3 [online]. 2015[cit. 2017-02-14], ISBN:978-80-260-8440-2. Dostupné z: [http:// www.vzhurudolu.cz/ebook](http://www.vzhurudolu.cz/ebook)
- [2] October CMS: Features [online]. 2016 [cit. 2017-02-14]. Dostupné z: <https://octobercms.com/features>
- [3] Wikipedie: Otevřená encyklopedie: Brackets [online]. c2016 [cit. 2017-02-14]. Dostupný z: <https://cs.wikipedia.org/w/index.php?title=Brackets&oldid=13336552>
- [4] Karen Carias. Simple words for a GitLab Newbie [online]. 2015[cit. 2017-02-14]. Dostupné z: <https://about.gitlab.com/2015/05/18/simple-words-for-a-gitlab-newbie/>
- [5] Craig Buckler. An Introduction to Gulp.js [online]. 2017[cit. 2017-02-14]. Dostupné z: <https://www.sitepoint.com/introduction-gulp-js/>
- [6] Maile Valentine. Access your assets anywhere [online]. 2014[cit. 2017-02-14]. Dostupné z: <https://helpx.adobe.com/creative-cloud/how-to/assets-get-started.html>
- [7] w3schools.com. jQuery Introduction [online]. 2017[cit. 2017-02-14]. Dostupné z: [https://www.w3schools.com/jquery/jquery\\_intro.asp](https://www.w3schools.com/jquery/jquery_intro.asp)
- [8] Materialize [online]. 2017[cit. 2017-02-14]. Dostupné z: <http://materializecss.com/>
- [9] Google. Material Design Introduction [online]. 2017[cit. 2017-02-14]. Dostupné z: <https://material.io/guidelines/#>
- [10] Wikipedie: Otevřená encyklopedie: Responzivní web design [online]. c2016 [cit. 2017-02-14]. Dostupný z: [https://cs.wikipedia.org/wiki/Responzivn%C3%AD\\_web\\_design](https://cs.wikipedia.org/wiki/Responzivn%C3%AD_web_design)