



Ekonomická
fakulta
Faculty
of Economics

Jihočeská univerzita
v Českých Budějovicích
University of South Bohemia
in České Budějovice

Jihočeská univerzita v Českých Budějovicích
Ekonomická fakulta
Katedra aplikované matematiky a informatiky

Bakalářská práce

Vývoj katedrových webových stránek s využitím systému pro správu obsahu

Vypracoval: Petr Knotek
Vedoucí práce: Mgr. Radim Remeš

České Budějovice 2021

JIHOČESKÁ UNIVERZITA V ČESKÝCH BUDĚJOVICÍCH

Ekonomická fakulta
Akademický rok: 2019/2020

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: Petr KNOTEK
Osobní číslo: E18308
Studijní program: B6209 Systémové inženýrství a informatika
Studijní obor: Ekonomická informatika
Téma práce: Vývoj katedrových webových stránek s využitím systému pro správu obsahu
Zadávající katedra: Katedra aplikované matematiky a informatiky

Zásady pro vypracování

Cílem bakalářské práce je – s využitím systému pro správu obsahu – navrhnout a vytvořit webové stránky pro Katedru aplikované matematiky a informatiky. Výsledná website bude navržena a vytvořena v moderním a praktickém stylu. Stránky budou poskytovat v rámci jednotlivých sekcí požadované informace dle potřeb vedení katedry a bude dodržovat zadané specifikace pro použití v rámci Ekonomické fakulty a Jihočeské univerzity.

Metodický postup:

1. Studium odborné literatury.
2. Zjištění funkční specifikace pro výsledné stránky.
3. Návrh a popis vývoje a implementace výsledných webových stránek.
4. Zhodnocení použitelnosti webových stránek pro nasazení v reálném prostředí.
5. Závěr.

Rozsah pracovní zprávy: 40 – 50 stran
Rozsah grafických prací: dle potřeby
Forma zpracování bakalářské práce: tištěná

Seznam doporučené literatury:

1. Eduonix Learning Solutions. (2017). *Learn to Create WordPress Themes by Building 5 Projects*. Birmingham, UK: Packt.
2. Krol, K. (2019). *WordPress 5 Complete*. Seventh Edition. Birmingham, UK: Packt.
3. Kyrmin, J., & Meloni, J. C. (2019). *Sams Teach Yourself: HTML, CSS, and JavaScript: All in One*. Third Edition. Old Tappan, New Jersey (USA): Pearson Education.
4. Nixon, R. (2018). *Learning PHP, MySQL & JavaScript*. 5th Edition. Sebastopol, CA (USA): O'Reilly.
5. Robbins, J. (2018). *Learning Web Design*. 5th Edition. Sebastopol, CA (USA): O'Reilly.
6. Uzayr S. (2016). *Learning WordPress REST API*. Birmingham, UK: Packt.

Vedoucí bakalářské práce: **Mgr. Radim Remeš**
Katedra aplikované matematiky a informatiky

Datum zadání bakalářské práce: **17. ledna 2020**

Termín odevzdání bakalářské práce: **16. dubna 2021**

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

JIHOČESKÁ UNIVERZITA
V ČESKÝCH BUDĚJOVICÍCH
EKONOMICKÁ FAKULTA
Studentská 13 (20)
376 02 České Budějovice


doc. Dr. Ing. Dagmar Škodová Parmová
děkanka


doc. RNDr. Tomáš Mrkvička, Ph.D.
vedoucí katedry

V Českých Budějovicích dne 25. března 2020

Prohlášení

Prohlašuji, že svou bakalářskou práci „Vývoj katedrových webových stránek s využitím systému pro správu obsahu“ jsem vypracoval samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury.

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své bakalářské práce, a to – v nezkrácené podobě/v úpravě vzniklé vypuštěním vyznačených částí archivovaných Ekonomickou fakultou – elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách, a to se zachováním mého autorského práva k odevzdanému textu této kvalifikační práce. Souhlasím dále s tím, aby toutéž elektronickou cestou byly v souladu s uvedeným ustanovením zákona č. 111/1998 Sb. zveřejněny posudky školitele a oponentů práce i záznam o průběhu a výsledku obhajoby kvalifikační práce. Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

Datum

Podpis

Poděkování

Rád bych poděkoval vedoucímu bakalářské práce Mgr. Radimovi Remešovi, CSc. za jeho cenné rady, připomínky a vstřícné jednání při vypracování mé bakalářské práce.

Obsah práce

1	Úvod.....	9
2	Cíle práce	9
3	Web design	10
3.1	Mezi jednotlivé postupy patří:	10
3.1.1	UX design	10
3.1.2	UI design.....	10
3.1.3	Informační architektura.....	10
3.1.4	Interakční design.....	10
3.1.5	Kódování a programování	10
3.1.6	Copywriting	11
3.1.7	SEO.....	11
3.1.8	Content strategy	11
4	Web design pro různá výstupní zařízení (responzivní design).....	12
4.1	Základní pravidla responzivního webdesignu.....	12
4.1.1	Fluid grid – mřížky	13
4.1.2	Fonts reponsive	16
4.1.3	Navigation.....	20
4.1.4	Responsivní obrázky.....	21
4.1.5	Media queries.....	25
4.2	Content Managment systems	31
4.2.1	Historie/ úvod	31
4.2.2	Typy	32
4.2.3	Funkce.....	33
4.2.4	Existující systémy:.....	35
5	Vlastní tvorba webové stránky	37
5.1	Cíl práce	37

5.2	Návrh řešení nových webových stránek	37
5.3	Předpokládaný stav webových stránek	38
5.4	Využití technologie	39
5.4.1	PHP	39
5.4.2	CSS	39
5.4.3	jQuery	40
5.4.4	Wordpress	40
5.4.5	ElementorPro	40
5.4.6	Fontawesome	40
5.4.7	Prostředí vývoje	40
5.4.8	Editor kódu	40
5.4.9	Google chrome nástroj pro vývojáře	41
5.5	Design wireframu	41
5.5.1	Úvodní stránka	41
5.5.2	Uchazeč	42
5.5.3	Aktuality	42
5.5.4	Členové katedry	42
5.5.5	Projekty	43
5.5.6	Kontakt	43
5.6	Vytvoření grafického stylu pro prvky webových stránek	43
5.7	Začátek tvorby webových stránek	43
5.7.1	Příprava a instalace Wordpressu	44
5.7.2	Příprava wordpressu a jeho základní struktura	44
5.7.3	Stažení a příprava šablony	46
5.8	Tvorba jednotlivých komponent a jednotlivých stránek	47
5.8.1	Tvorba headru	47
5.8.2	Tvorba footru	48

5.9	Tvorba jednotlivých stran	48
5.9.1	Úvodní stránka.....	49
5.9.2	Blog.....	49
5.9.3	Členové katedry	50
5.9.4	Projekty.....	50
5.9.5	Kontakty.....	51
5.10	Testování	51
5.10.1	Testování stránek při různých rozlišení	51
5.10.2	Testování různých prohlížečů pro desktopové zařízení.....	52
5.10.3	Testování reálných tabletů a mobilných zařízení	53
5.10.4	Zkouška testování s emulátorem virtuálních zařízení.....	55
5.10.5	Testování rychlosti načtení webu a hodnocení	55
5.10.6	Rekapitulace výsledků z testování	56
6	Závěr	57
	Summary and keywords.....	58
	Seznam literatury	59
	Elektronické zdroje	59
	Seznam obrázků, zdrojových kódů, diagramů, tabulek a zkratk	61
	Přílohy.....	64

1 Úvod

Celosvětová síť jménem internet se stala jednou neopomenutelnou technologií moderní doby, bez které si spousta lidí ani firem nedokáže představit normální život či fungování. V dnešní době se na internetu může nakupovat jakýkoliv druh zboží od potravin, součástek do technického průmyslu až po oblečení. Díky internetu se dá taky celosvětově komunikovat prostřednictvím sociálních sítí nebo prezentovat vlastní obsah či nějaké firemní portfolio. Internet je v neustálém rozvoji nových a inovativních věcí, které nějakým způsobem přispívají do každodenního života a důkazem toho může být neustále přibývající množství webových stránek či serverů. Statistiky říkají, že v dnešní době používá internet něco okolo 4.33 miliardy lidí a celosvětově se blížíme k 2 miliardám webových stránek. [Internet Live Stats 2021]

2 Cíle práce

Cílem této bakalářské práce bude vytvořit webové stránky pro Katedru aplikované matematiky a informatiky s využitím systému pro správu obsahu. Stránky by měli poskytovat všechny potřebné informace dle vedení katedry, a dodržet zadané specifikace pro použití v rámci Ekonomické fakulty a Jihočeského univerzity.

Na začátku práce je popsáno samotné téma webdesign a jeho základní části. Dále se práce věnuje a vysvětluje jednotlivé pravidla a techniky spojené s webdesignem. Druhou důležitou částí je samotné zpracování tématu systému pro správu obsahu neboli jeho historie, typy, funkce a v poslední řadě i existující příklady.

Praktická část bakalářské práce se zabývá samotným návrhem a vývojem webových stránek dle zadání. Výsledné webové stránky jsou spojením grafické návrhu, kódování dle responzivního designu a systému pro správu obsahu. Webové stránky jsou poté testovány zdali splňují responzivní chování pro mobilní zařízení a tablety.

3 Web design

Web design je obor, který zahrnuje spoustu dovedností a disciplín v oblasti navrhování a udržování webových stránek. Web design není jen o psaní kódu a grafické práci, je to blízko k marketingu, obsahové strategii nebo i psychologii. [Sharkie & Fisher 2015]

Cílem web designu je vytvořit webovou stránku, ve které se prolínají jak požadavky zadavatele, tak požadavky zákazníků, kteří stránky budou navštěvovat. Což znamená, že je potřebné vědět, kdo jsou naši potenciální návštěvníci, a z toho se dál pak rozvíjí návrh celého projektu webové stránky.

3.1 Mezi jednotlivé postupy patří:

3.1.1 UX design

User experience design je zaměřený na uživatelský zážitek návštěvníka webové stránky. UX design se zaměřuje na uživatelskou analýzu, interakční design, informační architekturu a další.

3.1.2 UI design

User interface design je zaměřen na vizuální část věci a jak celkový vzhled působí na návštěvníka. UI design řeší např. barvy, rozložení jednotlivých věcí na stránce, typografii a další.

3.1.3 Informační architektura

Informační architektura není zaměřena na rozložení informací na jedné stránce, ale řeší spíše jejich význam a provázání skrze všechny stránky (je dobré při výstavbě IA vzít v potaz jednotlivé informace a napsat si osnovu, co budou jednotlivé stránky sdělovat). Účel IA na webu je, zda uživatel snadno najde informace na webu, a s tím je spojeno i ovlivění SEO, protože dobrá IA zapříčiňuje důležitost stránek a podstránek pro vyhledavače. (Čevelíček 2018)

3.1.4 Interakční design

Interakční design má za úkolu vytvořit prostředí webových stránek, které je jednoduché a efektivní, ale zároveň také stále příjemné. (Niederst Robbins 2018)

3.1.5 Kódování a programování

Kódování a programování je pojem, který by se dal rozdělit na dvě menší podskupiny, což by bylo Front-end a Back-end. Frontendový vývoj se zabývá aspekty webových

stránek, které se zobrazují či mají souvislost s prohlížečem (HTML, JavaScript). Backendový vývoj je spíše zaměřen na tvorbu aplikací a programování serverů s databázemi (C, C++, Java, Python) (Maria DB, My SQL). (Ferguson 2021)

3.1.6 Copywriting

Copywriting je psaní reklamních či prezentačních textů se záměrem zvednout návštěvnost jisté webové stránky či zvýšit prodej nějakého produktu. Lze se setkat i s SEO Copywriting, což je optimalizování textů webové stránky tak, aby je byly pro vyhledavače atraktivnější (vyhledavače nesledují jenom zdrojový kód, nýbrž celkový obsah webové stránky). (Křížek & Crha, 2003)

3.1.7 SEO

SEO neboli Search Engine Optimization je zaměření, které se zabývá vylepšováním struktury webových stránek a kódu tak, aby se potencionálně zvýšila příčka, kde se webová stránka objeví na výsledné stránce vyhledavače. (Křížek & Crha, 2003)

3.1.8 Content strategy

Content strategy neboli internetový marketing se zaměřuje na vizuální část webových stránek. Hlavním cílem je vytvořit obsah webových stránek, který bude mít vymyšlen každý detail od jakéhokoliv textu po styl jednotlivých tlačítek. Soustředí se také na marketingové cíle organizace či společnosti. (Niederst Robbins 2018)

4 Web design pro různá výstupní zařízení (responzivní design)

Před méně než 10 lety bylo docela pochopitelné naprogramovat webové stránky s pevně danou šířkou (fixed width). Očekávání bylo, že všichni uživatelé dostanou docela konzistentní zážitek. Tato pevně daná šířka byla okolo 960px wide, nebylo to moc široké pro laptové obrazovky a uživatelé deskopových monitorů s vyšším rozlišením měli volný prostor po straně.

V roce 2007 společnost Apple Inc. ohlásila úplně první skutečnou mobilní variantu prohlížení webu a v tento okamžik se změnil způsob, jak lidé přistupovali a používali internet do té doby.

Důkazem může být statistika, která nám ukazuje, že na pomezí let 2010 až 2011 vzrostl počet lidí používající mobilní prohlížeč z 2,86 % na 7,02 %. V roce 2014 tato hodnota dosáhla 29,48 %. K nynějšímu roku dosáhla hodnota mobilních uživatelů 52,95 %, desktopový uživatelé jsou na 44,22 % a poslední jsou uživatelé tabletů kteří činí 2,83 %. (Statcounter, 1999 - 2021)

Nesporným faktem je, že počet lidí používající zařízení s menším displejem k prohlížení internetu stále roste. Na druhé straně této problematiky se začali využívat monitory s velkou uhlpříčkou (např. 27 palců či 34 palců prohnutý monitor). V dnešní době se začíná řešit spíše rozdíl mezi nejmenším možným displejem a největším možným displejem či monitorem.

Termín „responsive web design“ poprvé použil Ethan Marcotte v roce 2010 v jeho článku „A List Apart“, ve kterém stojí: “ Fluid grids, flexible images, and media queries are the three technical ingredients for responsive web design, but it also requires a different way of thinking.“. (Marcotte, 2010) Responzivní webdesign je souhrn CSS funkcí tzn, Media Querries, které umí přizpůsobovat rozložení stránek danému zařízení.

4.1 Základní pravidla responzivního webdesignu

Reponzivní design se řídí třemi pravidly, které umožňují svojí kombinací dosáhnout požadovaného výsledku. Téměř vždy při vývoji webových stránek lze narazit na zařízení či displej, na kterém nebudou funkční kaskádové styly a Javascript, což vede k hledání alternativní řešení pro daný typ problému.

- Fluid grid / flexibilní mřížka
- Respozivní obsah
- Media queries

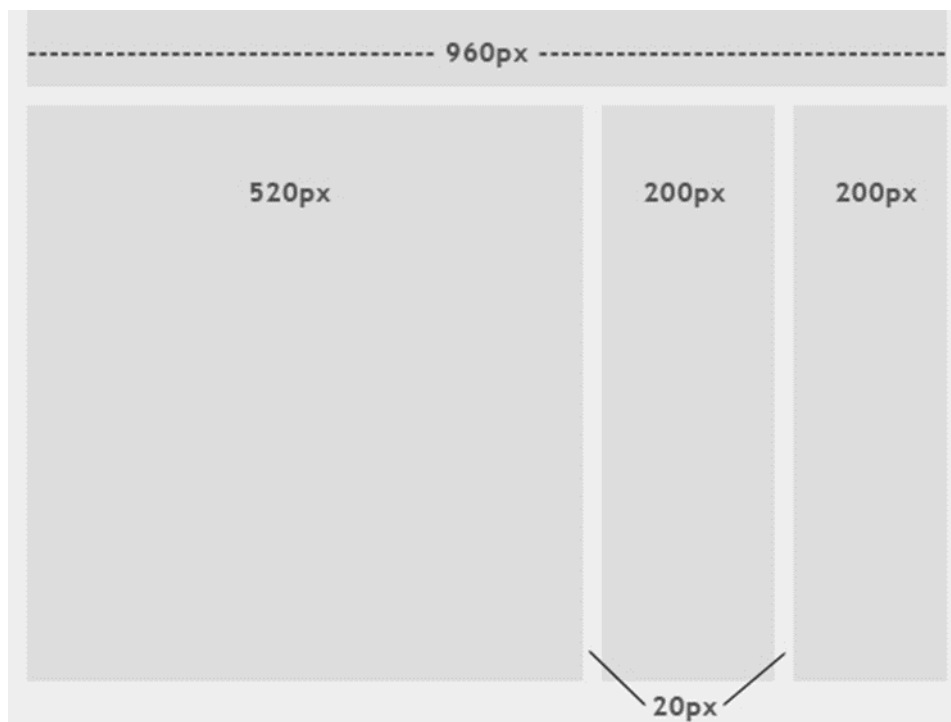
4.1.1 Fluid grid – mřížky

Grid systém je základním prvkem responzivního layoutu webových stránek. Jedná se většinou o dvourozměrnou strukturu tvořenou z protínajících se svislých, vodorovných, zakřivených či pod úhlem vodících linií, které se používají ke strukturování obsahu. Mřížka slouží jako framework, na kterém designer může uspořádat grafický obsah (obrázky, texty). Uvedu zde 4 existující typy layoutů, které pracují právě s mřížkou:

1. Layout with fixed width (s pevnou šířkou)
2. Fluid layout (plovoucí)
3. Elastic layout (elastický)
4. Hybrid layout (hybridní)

4.1.1.1 Layout with Fixed width

Layout s pevnou šířkou má hlavní wrapper (container) s pevnou šířkou, a komponenty v ní mají buď procentuální šířku, nebo pevnou šířku. Důležitou věcí je, že container je nastaven tak, aby se nepohyboval. Návštěvník uvidí stejnou šířku jako ostatní návštěvníci bez ohledu na to, jaké rozlišení obrazovky bude mít.



Obrázek 1: Schéma layoutu s pevnou šířkou

Zdroj: Smashingmagazine 2009

Obrázek 1 výše ukazuje obecný obrys rozvržení webové stránky s pevnou šířkou. Komponenty uvnitř jsou fixovány na 520, 200 a 200 pixelů. Šířka 960 pixelů se stala standardem tehdejšího designu webu, protože se předpokládalo, že většina uživatelů webových stránek bude procházet webové stránky v rozlišení 1024x768 nebo vyšším. [5]

Pozitiva a negativa tohoto layoutu:

Pozitiva

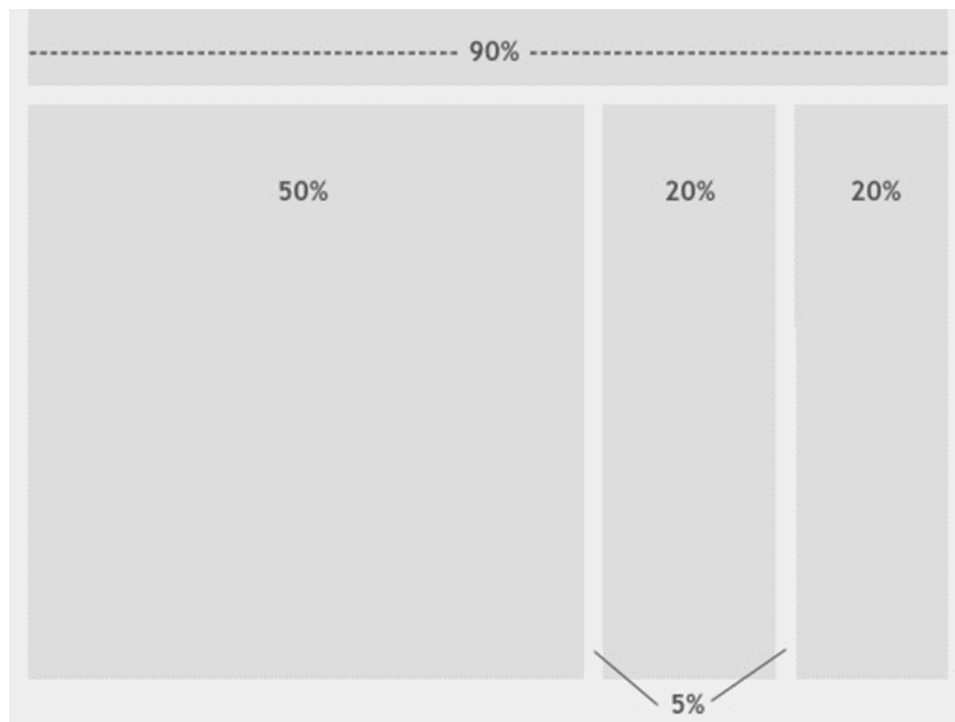
- Rozložení s pevnou šířkou je jednoduché nejen na používání, ale i přizpůsobování.
- S videi, obrázky a dalším obsahem je méně práce, protože šířka je u každého prohlížeče stejná.

Negativa

- Malé obrazovky mohou vyžadovat vodorovný posuvník.
- Rozložení s pevnou šířkou může uživatelům s velkými obrazovkami vytvořit nadměrné prázdné místo.

4.1.1.2 Fluid layout

Fluid layout je označován jako tekuté (plovoucí) rozložení, které pracuje s tím, že většina komponent uvnitř má procentní šířky, a přizpůsobuje se tak rozlišení uživatele.



Obrázek 2: Schéma fluidního layoutu

Zdroj: Smashingmagazine 2009

Obrázek 2 výše ukazuje tekuté rozložení webových stránek. Některé designy mohou určit šířky určitým elementům, jako jsou například okraje, ale obecně layout používá procentní šířky tak, aby byl vzhled webu upraven pro každého uživatele.

4.1.1.3 Elastic layout

Elasticita layoutu funguje v případech, kdy se uživatel rozhodne změnit velikost textu. Někdy se to může zdát jako zbytečná nebo dokonce nežádoucí činnost, ale pokud se vyskytne text, který lze rozšířit nebo zmenšit, může to mít smysl pro layout, který by ho obsahoval a dokázal měnit vlastní velikost dle tohoto textu.

Navíc pokud se uživatel rozhodne zobrazit text ve větší velikosti, velikost mezer se též úměrně zvětší, a tím by měl zachovávat správné proporce pro snadné čtení textu.

“A pixel is an unscalable dot on a computer screen, whereas an em is a square of its font size. Because font sizes vary, the em is a relative unit that responds to users’ text-size preferences.” - Patrick Griffiths, A List Apart (Griffiths, 2010)

Ve výsledku se jedná o mix předchozích dvou layoutů. Šířky prvků jsou vyjádřeny v em, což závisí na velikosti fontu. Proto zde dochází ke změně s různými velikostmi fontu a ne se změnou velikosti okna. Tato funkce je většinou přístupná pod klávesovou zkratkou (Ctrl + Kolečko na myši)

4.1.1.4 Hybrid layout

Hybridní rozložení kombinuje Fixed and Fluid prvky layoutu. Typické rozložení pro tento typ layoutu jsou dva sloupce po stranách s pevnou šířkou a hlavní sloupec ve středu, který má procentuální šířku. Což znamená, že zobrazování hlavního sloupce se bude přizpůsobovat prohlížeči, zatímco se boční sloupce vždy zobrazí dle nastavené šířky.

Tato metoda je hybridní, protože pracuje zároveň s dvěma typy jednotek na určování šířky. Většinou je to procentuální a pixelová. (Frain, 2020)

4.1.2 Fonts reponsive

Jelikož je snahou vybudovat responzivní design webu, tak je nutno přemýšlet nad typografickou změnou textu v rámci velikosti displeje. To vede k jednotkám, které se dělí na dvě skupiny, na absolutní a relativní. Je vhodné zvolit ty nejpoužívanější jednotky pro obě skupiny. Mezi absolutní jednotky patří Px (pixely), Pt (point), atd, a mezi relativní jednotky patří Em, Rem, % (procenta), vh, vw, atd.

4.1.2.1 Jednotka Px

Mezi jednotkami lze pixel označit za základní jednotku, se kterou se za léta práce všichni naučili pracovat. Jednotka je i dnes stále využívána, protože je v celku jednoduchá na přeložení a návrháři obvykle pracující v pixelech mohou snadno vytvářet přímo z Photoshopu.

4.1.2.2 Jednotka Em

Je jednotka, která je založená na velikosti písma rodičovského elementu, což znamená, že 1em není vždy stejná. A tím vzniká problém nepřehlednosti s touto jednotkou. Problematika je znázorněná na níže uvedeném obrázku 3.

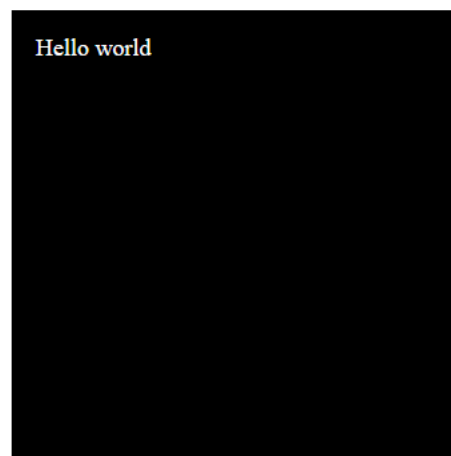

```

div {
  width: 200px;
  height: 200px;
  background: blue;
  font-size: 100%;
  color: white;
  padding: 1em;
}

p {
  width: 200px;
  height: 200px;
  background: black;
  font-size: 75%;
  color: white;
  padding: 1em;
}

<body>
  <div> Hello world </div>
  <p> Hello world </p>
</body>

```



Obrázek 3: Ukázka funkčnosti jednotky Em

Zdroj: Autor

4.1.2.3 Jednotka Rem

Tato jednotka je podobná předchozí jednotce, ale naruší od ní vychází z velikostí písma dokumentu. Jednotka obsahuje výchozí hodnotu velikosti písma nastavenou autorem a případně upravenou v prohlížeči uživatelem. Rem je zajímavé pro svou jednoduchost, protože jde nastavovat např. okraj odstavce dle výšky písma viz výpis 1 níže.

Výpis 1: Příklad použití jednotky Rem na odsazení odstavce

```

p {
  margin: 1rem;
}

```

Při změně výšky písma se mění i výsledná velikost okraje dle potřeby. Zápis ale zůstává stejný.

4.1.2.4 Jednotka Procenta

Zápis procent je tradičním znakem %. Tato jednotka funguje na stejném principu dělení jako em, což znamená že pokud základní velikost je 16px = 100 % tak 200 % = 32px. Negativní část této jednotky je při použití u různých vlastností.

4.1.2.5 Responsive typography

Tato typografie pracuje na základě media queries (mediálních dotazů) k výběru správného CSS setu pro různé widths breakpoint (bod určité šířky displeje).

Lze si vytvořit příklad, který by na mobilních zařízeních zobrazil menší font a na deskopových zařízeních zase větší font viz výpis 2 níže.

Výpis 2: Ukázka responsive typography s využitím media queries

```
html {
  font-size: 100%;
}
h1 {
  font-size: 2.25rem; // 36px
}
@media (min-width: 700px){
  h1 {
    font-size: 3.5rem; // 56px
  }
}
@media (min-width: 1000px){
  h1 {
    font-size: 4.75rem; // 76px
  }
}
```

V tomto příkladu bude text h1 2,25rem (36px) pro zařízení do šířky 700px, 3,5rem (56px) pro zařízení mezi 700px a 1000px a 4,75rem (76px) pro zařízení větší nebo rovna 1000px.

S tímto přístupem se používá základní velikost písma 100 % a ostatní velikosti písma v rem, takže web bude přístupný. Uživatel může zvětšit velikost zobrazovaného obsahu úpravou nastavení velikosti písma prohlížeče nebo přiblížením.

4.1.2.6 Fluid typography

Fluid typography označuje způsob dimenzování fontu tak, aby se zvětšovaly se zvětšující se šířkou vw (viewport width). A nutně nevyžadují dotazy na média.

Hlavní dva komponenty fluid typography:

The `calc()` CSS funkce, která umožňuje provádět matematické operace pomocí kombinace různých CSS jednotek.

Viewport jednotky, specifický viewport width

Jednotky viewport jsou docela obtížné na kontrolu kvůli různému zvětšování a zmenšování. Ale je zde možnost vytvořit pravidlo za pomoci kombinace Vm a Rem jednotek s funkcí `Calc()` viz výpis 3.

Výpis 3: Zápis funkce `Calc()`

```
h1 {  
  font-size: calc(1.3rem + 3.6vw);  
}
```

Tato vypočítaná hodnota bude minimálně 1.3 rem nebo 21 pixelů, pokud by viewport byl 0 (což by se v reálném světě nemělo stát). Hodnota 3,6 vw pak přidá tempo růstu, které lze ovládat a dosáhne se tak větší plynulosti zvětšování než se samotnými Viewport jednotkami.

4.1.2.7 Flexible typography with locks

Tato metoda je podobná předchozímu přístupu, neboť k výpočtu velikosti písma se používá `Calc()`. Oproti předchozímu přístupu je nutno matematicky upravit rovnici pro výpočet tak, aby velikost písma byla taková, která je potřebná při minimálním, ale i maximálním breakpointu. Aby se toho dosáhlo, použije se matematický vzorec napsaný v roce 2012 Timem Brownem v příspěvku na blogu k výpočtu řádku. Mike Riethmuller a další jej poté aplikovali na velikost písma. Tim Brown poté roku 2016 publikoval následné vysvětlení využívající canal locks analogy.

The diagram illustrates the formula for font-size using Calc() with annotations for min font size, min screen size, max font size - min font size, and max screen size - min screen size.

$$\text{font-size: calc(} \underbrace{16\text{px}}_{\text{min font size}} + \underbrace{(24 - 16)}_{\text{max font size - min font size}} * \underbrace{(100\text{vw} - 400\text{px})}_{\text{min screen size}} / \underbrace{(800 - 400)}_{\text{max screen size - min screen size}});$$

Obrázek 4: Dosazení do funkce `Calc`

Zdroj: *Smashingmagazine 2016*

Tato metoda se dá potom propojovat s již zmiňovanými media queries nebo Sass mapping.

4.1.3 Navigation

Navigace je velmi důležitou součástí pro responzivní design webových stránek. Vzhledem k tomu, že navigace na obrazovkách desktopů byla do značné míry překonána, skutečné výzvy přicházejí v novém vytváření možností navigace na malých obrazovkách. Pro malé obrazovky se objevila řada úspěšných vzorů, které jsou dále stručně shrnuty.

4.1.3.1 Top navigace (horní navigace)

Pokud web obsahuje jen několik navigačních odkazů, mohou se vejít do jednoho nebo dvou řádků v horní části obrazovky.

4.1.3.2 Priorita +

V tomto vzoru se nejdůležitější navigační odkazy zobrazují na řádku v horní části obrazovky spolu s odkazem Další, který odhaluje další možnosti. Klady jsou v tom, že primární odkazy jsou v prostém zobrazení a počet zobrazených odkazů se může zvyšovat s rostoucí šířkou zařízení. Nevýhody zahrnují obtížnost určení, které odkazy by měly mít místo na malé obrazovce.

4.1.3.3 Vyberte nabídku

Pro střední seznam odkazů používají některé weby vybraný prvek vstupního formuláře.

Klepnutím na nabídku se otevře seznam možností pomocí uživatelského rozhraní nabídky operačního systému, jako je například rolovací seznam odkazů ve spodní části obrazovky. Výhodou je kompatibilitnost, na druhou stranu se formuláře obvykle nepoužívají pro navigaci a nabídka může být přehlédnuta.

4.1.3.4 Odkaz na zápatí nabídky

Jeden přímý přístup umístí odkaz na nabídku v horní části stránky, který odkazuje na úplnou navigaci umístěnou ve spodní části stránky. Riziko tohoto vzoru spočívá v tom, že může být dezorientující pro uživatele, kteří se najednou ocitnou v dolní části stránky.

4.1.3.5 Accordion sub-navigation

Pokud existuje mnoho možností navigace s pod navigačními nabídkami, řešení pro malou obrazovku se stává náročnějším, zvláště když nelze umístit kurzor myši a získat více možností. Accordions, které se rozbalí, když se klikne na malou ikonu šipky, které se běžně používají k odhalení a skrytí podnavigace. Mohou být dokonce vnořeny do několika úrovní hluboko. Aby se zabránilo vnoření navigace v Sub-menu accordion,

některé weby jednoduše odkazují na samostatné vstupní stránky, které obsahují seznam sub-navigace pro danou sekci.

4.1.3.6 Nabídka stisknutí a překrytí

V přepínací navigaci je navigace skrytá, ale rozšiřuje se dolů, když klepnete na odkaz na nabídku. Může tlačit hlavní obsah dolů pod něj (stisknutím) nebo klouzat dolů před obsah (překrytí).

4.1.3.7 Off-canvas / fly-in

Tento oblíbený vzor umístí navigaci na panelu mimo obrazovku nalevo nebo napravo od hlavního obsahu, který se zobrazí po klepnutí na ikonu nabídky. (Niederst Robbins 2018)

4.1.4 Responsivní obrázky

Když se řekne “responsivní obrázky“, jedná se o poskytnutí obrázku přizpůsobeného prostředí, odkud uživatel prohlíží stránku. Je zde možnost omezení, aby prohlížeče s malými obrazovkami nemuseli stahovat více obrazových dat než je potřeba. Kromě toho poskytují vývojářům způsob jak využít nové efektivnější formáty pro obrázky.

Responsivní obrázky fungují následujícím způsobem: poskytnete více obrázků, které jsou různých velikostí nebo oříznuté dle velikosti obrazovky. Prohlížeč vybere ten nejvhodnější na základě toho, co ví o aktuálním prostředí zobrazení. Rozměry obrazovky jsou jedním z faktorů, ale může být také zahrnuto rozlišení, rychlost sítě, obsah, který již obsahuje mezipaměť, uživatelské preference a další aspekty.

Mezi základní scénáře pro responzivní obrázky platí:

- Poskytování extra velkých obrázků, které vypadají ostře na obrazovkách s vysokým rozlišením.
- Poskytování sady obrázků různých rozměrů pro použití na různých velikosti obrazovky.
- Poskytování verzí obrazu s různým množstvím detailů na základě velikost a orientace zařízení (art direction).
- Poskytování alternativních formátů obrázků, které ukládají stejný obrázek při mnohem menších velikostech souborů. (Niederst Robbins 2018)

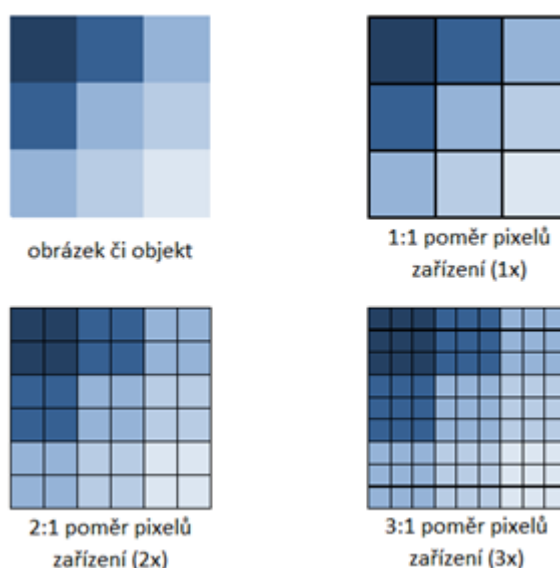
4.1.4.1 Poměr pixelů zařízení

Díky vývoji technologií se vývojáři stále snaží vylepšovat kvalitu rozlišení obrazovky. Výsledkem je, že pixely zařízení jsou menší a menší, což znamená, že by obrázek byl nečitelně malý, pokud by se mapovalo jedna ku jedné.

Pro kompenzaci se využívá takzvaný „referenční pixel“. Referenční pixely jsou také známy jako body (PT) v systému iOS, pixely nezávislé na zařízení (DP nebo DiP) v systému Android nebo CSS pixely, protože jsou měrnou jednotkou, která se používá v šablonách stylů.

Například Iphone 8 má obrazovku, která je tvořena 750 x 1334 pixely zařízení, ale používá rozložení mřížky 375 x 667 bodů nebo pixelů CSS (poměr 2 pixely zařízení k 1 pixelu rozložení neboli 2x). Dalším příkladem může být iPhone X, jehož obrazovka je tvořena z 1125 x 2436 pixely, ale používá rozložení mřížky 375 x 812 bodů (poměr 3 pixely zařízení k 1 pixelu rozložení neboli 3x).

Poměr počtu pixelů zařízení k pixelům CSS se nazývá poměr pixelů zařízení (viz obrázek 5 níže). Běžné poměry pixelů na mobilních zařízeních jsou 1,325x, 1,5x, 1,7x, 2x, 2,4x, 3x a dokonce 4x („x“ je konvence pro označení poměru pixelů zařízení). Dokonce i velké stolní displeje mají poměr 2x, 3x a 4x.



Obrázek 5: Ukázka poměrů pixelů zařízení

Zdroj: Autor

4.1.4.2 Srcset

Srcset je způsob, jak poskytovat větší obrázky pouze prohlížečům na displejích, které z nich mají prospěch. Dělá se to pomocí atributu srcset s elementem img. Srcset umožňuje vývojářům určit seznam možností zdrojů obrázků, ze kterých si prohlížeč může vybrat.

Hodnota srcset je seznam možností oddělených čárkami. Každá položka v tomto seznamu má dvě části: adresu URL obrázku a descriptor x, který určuje cílový poměr pixelů v zařízení. Celý seznam je hodnotou srcset a jde do jedné sady uvozovek.

Prohlížeče, které rozpoznají atribut srcset, zkontrolují rozlišení obrazovky a stáhnou obrázek, který považují za nejvhodnější. Pokud je prohlížeč na počítači Mac s displejem 2x Retina, může si stáhnout obrázek 400px.jpg. Pokud je poměr pixelů zařízení 1,5x, 2,4x nebo jiný, zkontroluje celkové prostředí pro prohlížení a provede nejlepší výběr.

4.1.4.3 Descriptor X

Deskriptory X sdělují prohlížeči, aby provedl výběr pouze na základě rozlišení obrazovky bez ohledu na rozměry obrazovky nebo viewportu. Z tohoto důvodu se selektory x nejlépe používají pro obrázky, které zachovávají stejné rozměry bez ohledu na velikost obrazovky, jako jsou loga, odznaky sociálních médií nebo jiné obrázky s pevnou šířkou.

4.1.4.4 Descriptor W

W deskriptor (deskriptor šířky) poskytuje skutečnou šířku pixelu každého obrázku. Použití srcset s deskriptorem w je vhodné, pokud jsou obrázky identické, kromě jejich rozměrů (jinými slovy, liší se pouze v měřítku). Zde je příklad atributu srcset, který poskytuje čtyři možnosti obrázku a určuje jejich příslušné šířky pixelů pomocí deskriptorů w viz výpis 4.

Výpis 4: Ukázka použití srcset s deskriptorem W

```
<img src="/images/medium_1024.png" alt="alternative image"
      srcset=" /images/small_1024.png 600w,
              /images/medium_1024.png 1024w,
              /images/large_1024.png 1600w,"
```

Při použití descriptoru W se musí také použít atribut size, aby se prohlížeči sdělila přibližná velikost, v jaké se obrázek zobrazí v rozložení stránky (je vyžadován specifikací).

4.1.4.5 Atribut sizes

Když prohlížeč stáhne dokument HTML pro webovou stránku, tak si prvně prohlédne celý dokument a vytvořit jeho obrysovou strukturu (jeho Document Object Model nebo DOM). Pak se téměř okamžitě zapne preloader, který získá všechny obrázky ze serveru, takže jsou připraveny k použití. Nakonec se stáhne CSS a JavaScript. Je pravděpodobné, že šablona stylů obsahuje pokyny pro rozložení a velikosti obrázků, ale v době, kdy prohlížeč uvidí styly, jsou obrázky již staženy. Z tohoto důvodu se musí dát prohlížeči nápověda s atributem size, zda obrázek vyplní celou šířku výřezu nebo jen jeho část. To umožňuje preloaderu vybrat správný obrazový soubor ze seznamu srcset.

Například atribut sizes informuje prohlížeč, že obrázek vyplňuje celý výřez pomocí jednotek šířky viewportu (vw), nejběžnější jednotky pro atribut velikostí, takže prohlížeč může vybrat nejlepší obrázek pro danou úlohu. Například 100vw se promítá do 100% šířky výřezu, 50vw by bylo 50% atd. Lze také použít em, px a několik dalších jednotek CSS, ale nelze použít procenta. Prohlížeče, které nepodporují srcset a velikosti, používají obrázek uvedený v atributu src viz výpis 5.

Výpis 5: Ukázka použití srcset s deskriptorem W

```
<img src="/images/medium_1024.png" alt="alternative image"
srcset="      /images/small_1024.png 600w,
            /images/medium_1024.png 1024w,
            /images/large_1024.png 1600w,"
sizes="      (max-width: 480px) 100vw,
            (max-width: 960px) 70vw,
            240px,"
```

Atribut sizes říká prohlížeči následující:

- Pokud je viewport široký 480 pixelů nebo menší (maximální šířka je 480 pixelů), vyplní obrázek na 100% šířky viewportu.
- Pokud je výřez širší než 480 pixelů, ale ne větší než 960 pixelů (maximální šířka: 960 pixelů), pak se obrázek zobrazí na 70% výřezu (Jsou zde margins na levé a pravé straně po 15%).
- Pokud je výřez větší než 960 pixelů a nesplňuje žádnou z dřívějších podmínek médií, velikost obrázku bude přesně 240 pixelů.

4.1.5 Media queries

Jako další responzivní nástroj může sloužit media queries neboli do češtiny přeloženo: „Dotaz na médium“. Media queries jsou způsob, jak cílit na prohlížeče podle určitých charakteristik, funkcí, uživatelských preferencí a následně použít styly nebo spustit jiný kód založený na těchto vlastnostech viz výpis 6. Snad mezi nejběžnější media queries na světě patří ty, které cílí na konkrétní výřezy a používají vlastní styly, což zrodilo celou myšlenku responzivního designu.

Výpis 6: Ukázka použití @media-queries

```
/* Aktivuje se pokud prohlížeč má na šířku nejméně 600px nebo víc*/  
@media screen and (min-width: 600px) {  
  .element {  
    /*aplikování určitého stylu*/  
  }  
}
```

Kromě šířky výřezů se nabízí možnost cílit na spoustu dalších vlastností, čímž může být rozlišení obrazovky, orientace zařízení, preference operačního systému nebo mnohem více z celé řady možných dotazovaných a používaných vlastností k designování obsahu.

4.1.5.1 Struktura Media Query

Media queries se skládají z typu médií a podmínky obsahující vlastnosti media s hodnotou nebo rozmezím hodnot



Obrázek 6: Struktura media queries

Zdroj:Css-tricks

4.1.5.2 @media

Na prvním místě v kódu na mediální dotaz stojí samotné pravidlo @media, které je z mnoha pravidel CSS. Z hlediska responzivity je tato bakalářská práce zaměřena pouze na toto pravidlo, protože je přizpůsobeno typu média, na kterém je web zobrazen, jaké funkce tento typ media podporuje a operátory, které lze využívat tak, aby kombinovaly jednoduché i složité podmínky.

4.1.5.3 Typy medií

Je nutné předem vědět na jaký typ médií bude zvolen pro prezentaci obsahu. V mnoha ne-li většině případech se zde zobrazí hodnota screen, což dává smysl, protože je hlavní

snahou cílit na zařízení s připojenými obrazovkami. Avšak obrazovky nejsou jediným typem médií, na které můžeme cílit, může se jednat o syntetizátory řeči, projektor a další. Mediální typy vznikly proto, aby upřesnily pravidla pro používání určitých zařízení. Mediální typy sdělují prohlížeči na jakém zařízení je web zrovna prohlížen.

Přehled nepoužívanějších typů

All – odpovídá všem zařízením.

Print – Porovnává dokumenty, které jsou zobrazeny v náhledu tisku nebo jakémkoliv jiném médiu, které rozděluje obsah na stránce určené k tisku.

Screen – Odpovídá zařízením s obrazovkou.

Speech - Odpovídá zařízením, která čtou obsah slyšitelně, například čtečku obrazovky.

4.1.5.4 Vlastnosti Médíí

Jakmile je definován typ média, kterému se vytváří odpovídající vzhled, lze následně definovat k jakým funkcím bude přiřazen. Toto lze vidět na příkladu ve výpisu 6, kde byl použit typ media screen a vlastnost byla min-width 600px, tímto kódem lze zjistit, zda se jedná o počítačovou obrazovku a zda má viewport stejně či více než 600px.

V následující tabulce jsou shrnuty kategorie vlastností:

Viewport / Charakteristika stránky		
<u>Vlastnost</u>	<u>Souhrn</u>	<u>Hodnoty</u>
width	Definuje šířky výřezu. Může to být konkrétní číslo (např. 400 pixelů) nebo rozsah (s použitím minimální a maximální width).	<length>
height	Definuje výšku výřezu. Může to být konkrétní číslo (např. 400 pixelů) nebo rozsah (s použitím minimální a maximální Height).	<length>
aspect-ratio	Definuje poměr stran šířky k výšce výřezu	<ratio>
orientation	Způsob orientace obrazovky, například na výšku nebo na šířku, podle toho, jak se zařízení otáčí.	portrait, landscaped
Kvalita displeje		
<u>Vlastnost</u>	<u>Souhrn</u>	<u>Hodnoty</u>
resolution	Definuje cílovou hustotu pixelů zařízení	<resolution>, infinite
scan	Definuje proces skenování zařízení, což je způsob, jakým zařízení vytváří obraz na obrazovku (kde interlace kreslí střídavě liché a sudé čáry a progressive kreslí všechny postupně).	interlace, progressive
grid	Určuje, zda zařízení používá Grid (1) nebo bitmap (0) obrazovku	0 = Bitmap, 1 = Grid
update	Zkontroluje, jak často může zařízení upravovat vzhled obsahu (pokud vůbec), s hodnotami včetně none, pomalý a rychlý.	slow, fast, none
Color		
<u>Vlastnost</u>	<u>Souhrn</u>	<u>Hodnoty</u>
color	Definuje podporu barev zařízení, vyjádřenou číselně jako bity. Hodnota 12 by tedy byla ekvivalentem zařízení, které podporuje 12bitovou barvu, a hodnota nula označuje žádnou podporu barev.	<integer>
color-index	Definuje hodnoty, které zařízení podporuje. Může to být konkrétní číslo (např. 10 000) nebo rozsah (např. Min. Barevný index: 10 000, max. Barevný index: 15 000), stejně jako šířka.	<integer>
monochrome	Počet bitů na pixel, kde zařízení podporuje monochromatický mód, kde nula reprezentuje žádnou monochromatickou podporu.	<integer>

Tabulka 1: Vlastnosti @media-queries

Zdroj: Autor

Nejpoužívanější vlastnosti jsou ty v kategorii Viewport. Dále v této tabulce nejsou zahrnuty všechny možnosti vlastností, které by se zde daly použít např. User preference, scripting nebo Video prefixed.

4.1.5.5 Operátory

Media queries podporují logické operátory jako mnoho programovacích jazyků, takže je možno za určitých podmínek porovnávat typy médií. Pravidlo @media je samo o sobě

logickým operátorem, který v zásadě uvádí, že „pokud“ jsou následující typy a vlastnosti shodné, použij šablonu stylů.

Mezi nejpoužívanější patří AND, NOT, OR (nebo možnost s ,), ONLY

- AND – Je to operátor, který umožní spojit více výrazů, například lze použít operátor AND když je vyžadováno cílit na obrazovky v určitém rozsahu viz výpis 7. Zde je zajištěno zjištění obrazovky a je hledána vlastnost mezi 320px a 768px.

Výpis 7: Ukázka použití AND operátoru

```
/* Zaměření se na obrazovky mezi 320px a 768px */  
@media screen (min-width: 320px) and (max-width: 768px){  
  .element{  
    /*styl*/  
  }  
}
```

- OR – neboli comma-separated list které reprezentuje (,) v zápisu se chovají jako logický operátor OR při použití v media queries. Pokud při psaní media queries bude využito oddělení čárkami, nastane stav, při kterém některý z media queries vrátí hodnotu true a následně budou použity styly nebo šablony stylů. Každý media queries oddělený čárkou je považován za samostatnou podmínku a jakýkoli operátor aplikovaný na jeden media queries neovlivní ostatní. To znamená, že dotazy na média oddělené čárkami mohou cílit na různé funkce, typy a stavy médií viz výpis 8. Zde je zjišťována existence obrazovky s 700px viewportem nebo otočení zařízení na šířku, neboli také landscape mód obrazovky.

Výpis 8: Ukázka použití OR operátoru

```
@media screen (min-width: 700px), handheld and(orientation: landscape){  
  .element{  
    /*styl*/  
  }  
}
```

- NOT – Může nastat situace, kdy je zapotřebí cílit na zařízení podle toho, co nepodporují nebo neodpovídají. Tento kód odstraní barvu pozadí těla, pokud je zařízením tiskárna, a může zobrazit pouze jednu barvu viz výpis 9 na další stránce.

Výpis 9: Ukázka použití NOT operátoru

```
@media print and ( not(color) ){
  body{
    background-color: none;
  }
}
```

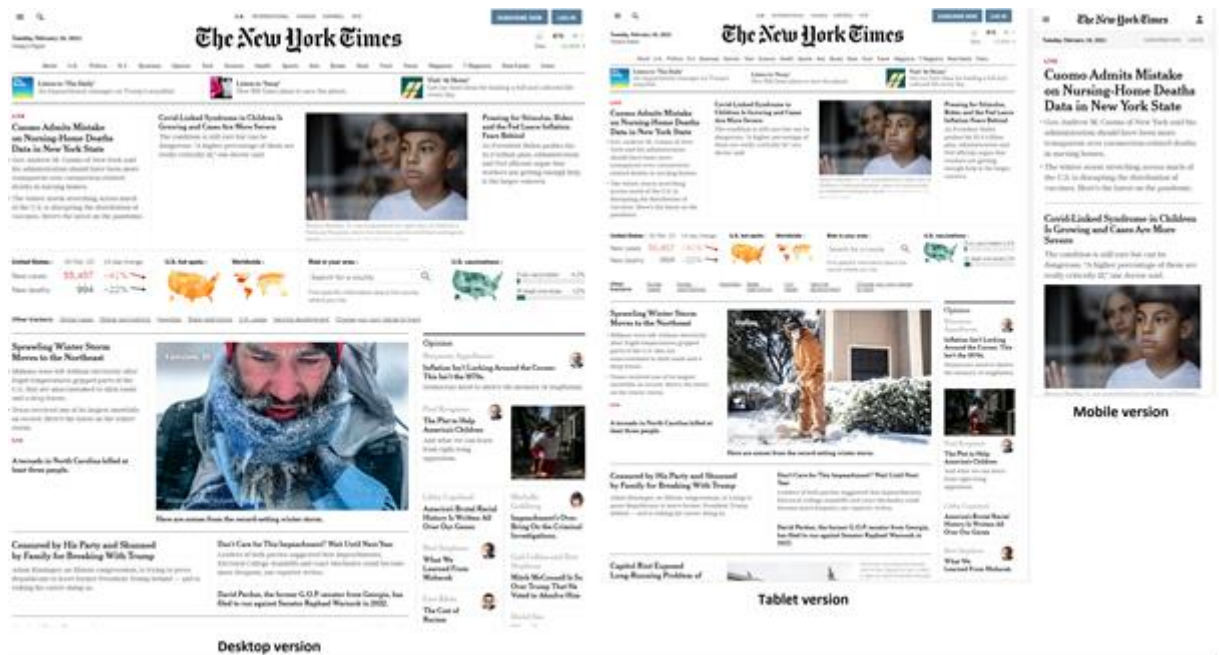
- ONLY – Jediné klíčové slovo, které brání starším prohlížečům, které nepodporují media queries s funkcemi médií v použití daných stylů. Příklad viz výpis 10 zde vidíme zápis, který schová obsah divu na menších obrazovkách. (Frain, 2020; Niederst Robbins, 2018)

Výpis 10: Ukázka použití ONLY operátoru

```
@media screen and (max-width: 600px) {
  div.pokus {
    display: none;
  }
}
```

4.1.5.6 Breakpoints

Breakpoint neboli bod zlomu je bod, ve kterém je použit mediální dotaz k zavedení změny stylu. Když v mediálním dotazu je zadán min-width: 800px, je tak určeno, že 800 px jsou hranice, kde by měl být použit daný styl. Na obrázku 18 lze pozorovat některé příklady breakpointu, ve kterých nytimes.com na své domovské stránce provádí změny v rozložení ale tak i designu.



Obrázek 7: Ukázka funkce breakpoint

Zdroj: The New York Times

Výběr hraničních hodnot může být náročný, ale je možno vycházet ze zkušenosti. V minulosti, kdy bylo poměrně málo zařízení k prohlížení internetu, byla tendence založit breakpointy na ověřených hodnotách (320 pixelů pro smartphony, 768 pixelů pro iPady a 1200 pixelů desktop verze). Postupem času a vývojem technologie netrvalo dlouho a lidé se začali vypořádávat se šířkami zařízení od 240 po 3000 pixelů a proto device-based přístup přestal být využíváný.

4.1.5.7 Module-Based Breakpoints

Tento přístup vytváří breakpointy pro jednotlivé části stránky, nikoliv přepínání celé stránky najednou. Běžnou praxí je nejprve vytvořit design pro úzké obrazovky a poté je pomalu zvětšován a sledují se jednotlivé části, kdy začnou potřebovat upravit.

Ethan Marcotte ve své knize *Responsive Design: Patterns & Principles* (A Book Apart) nazývá tento design „content out“ a vyjadřuje jej takto: For me, that “content out” process begins by looking at the smallest version of a piece of content, then expanding that element until its seams begin to show and it starts to lose its shape. Once that happens, that’s an opportunity to make a change—to introduce a breakpoint that reshapes the element and preserves its integrity. (Marcotte, 2015)

Pokud dojde k zjištění, že je zahrnuto mnoho Breakpointů v rozmezí několika pixelů či Em jednotek od další, jejich seskupení by mohlo zefektivnit používání vaší šablony stylů

a procesů. A není na škodu zjistit velikosti obrazovek u populárních zařízení pro případ, že by se mohl posunout breakpoint právě na zjištěnou hranici a tím vylepšit prostředí pro celou kategorii uživatelů.

Pro vyhledávání velikostí obrazovky lze doporučit webové stránky (screensiz.es), kde se nachází většina populárních zařízení.

4.1.5.8 Em-Based Breakpoints

Předchozí příklady byly založeny na breakpointech s pixely, v této kapitole bude popsána alternativa, která je více populární a využívá jednotky Em namísto pixelů v media queries. Je již známo, že em se rovná aktuální velikosti písma prvku, ale při využití v media queries se rovná velikosti písma dokumentu (16 pixelů ve výchozím nastavení, je zde možnost to změnit jak ze strany autora, tak i ze strany uživatele).

Media queries založené na pixelech se nepřizpůsobí, pokud uživatel změní nastavení velikosti písma, k čemuž dochází poměrně často aby byl text pro uživatele lépe čitelný. Na druhou stranu media queries založené na em reagují na změnu velikosti textu a udržují rozložení stránky.

Například: bude využit layout, který se při dosažení 800 px se změní na dva sloupce. Je počítáno s tím, že hlavní sloupec má optimální délku textového řádku, když je výchozí velikost písma 16 pixelů. Když uživatel změní velikost písma na 32 pixelů, dojde k zvětšení textu do stejného prostoru jako předtím, což je nežádoucí.

Pokud bude použito media queries založené na em a dotaz bude zaměřen na prohlížeče širší než 50em, když je základní velikost písma 16 pixelů, dojde k přepnutí na 800 pixelů (jak je navrženo). Pokud ale dojde k zvětšení písma ze strany uživatele a dosáhne velikost písma na 32 pixelů, tak layout o dvou sloupcích přejde na 1600 pixelový rozložení (50em * 32px = 1600px) a když bude dostatek prostoru, aby se hlavní sloupec vyplnil se stejnou délkou řádku jako původní design. (Niederst Robbins, 2018)

4.2 Content Management systems

4.2.1 Historie/ úvod

První zmínky o CMS neboli content management system se začaly objevovat v takzvaném webu 2.0.

Termín web 2.0 pomohl definovat to, co se také nazývá „participativní“ nebo „participativní a sociální web“, což odkazuje na nárůst obsahu vytvářeného uživateli a snadné použití webových stránek s ostatními produkty a systémy.

Postupně se web přesunul ze statických webů s brožurami na interaktivní web s dynamickým obsahem. S rostoucí potřebou za aktuálním obsahem se dostala do popředí potřeba po správě obsahu (CMS). Úlohou systému pro správu obsahu bylo poskytnout možnost více uživatelům s různými úrovněmi oprávnění spravovat obsah pro web nebo část obsahu.

CMS je obvykle serverový uživatelský software, který interaguje s obsahem na úložišti. Toto úložiště může být umístěno na stejném serveru jako součást stejného softwarového balíčku nebo zcela v samostatném úložišti.

CMS umožňuje editorům vytvářet nový obsah, upravovat stávající obsah, provádět redakční procesy týkající se obsahu a nakonec tento obsah zpřístupnit dalším lidem, aby jej mohli využívat.

Logicky se CMS skládá z mnoha částí. Editační rozhraní, úložiště, publikační mechanismy atd. Pro netechnického editora jsou však všechny tyto části obecně považovány za jediný monolitický celek: „CMS“.

4.2.2 Typy

Jsou 4 hlavní typy CMS:

4.2.2.1 Správa webového obsahu (WCM)

Správa obsahu primárně určeného pro hromadné doručování prostřednictvím webu. WCM vyniká v oddělení obsahu od prezentace a publikování k více kanálů.

4.2.2.2 Správa podnikového obsahu (ECM)

Správa obecného obchodního obsahu, která nemusí být nutně určena k hromadnému doručování nebo spotřebě (např. Životopisy zaměstnanců, hlášení incidentů, poznámky atd.). Tento typ systému byl tradičněji znám jako „správa dokumentů“, ale název se v průběhu let zobecnil. ECM vyniká ve spolupráci, kontrole přístupu a správě souborů.

4.2.2.3 Digitální správa aktiv (DAM)

Správa a manipulace s bohatými digitálními aktivy, jako jsou obrázky, zvuk a video, pro použití v jiných médiích. DAM exceluje v metadatech a ztvárnění.

4.2.2.4 Správa záznamů (RM)

Správa transakčních informací a dalších záznamů, které jsou vytvářeny jako vedlejší produkt obchodních operací (např. Záznamy o prodeji, záznamy o přístupu, smlouvy atd.). RM vyniká retencí a kontrolou přístupu.

DAM je zajímavý tím, že se rozlišuje především na základě toho, co dělá s obsahem. Zatímco téměř jakýkoli systém pro správu obsahu může ukládat video a obrazové soubory, tak ECM v tom ve skutečnosti vyniká. DAM je o krok dále tím, že poskytuje jedinečné nástroje pro vykreslování a transformaci digitálních aktiv. Proto základní funkce správy systému DAM překrývají docela dobře funkce systému ECM, přičemž systém DAM navrhuje úroveň funkčnosti. (Mnoho systémů DAM se skutečně prodává jako doplňky k systémům ECM.)

Existují i jiné příklady:

4.2.2.5 Systémy správy obsahu komponent (CCMS)

Používá se ke správě extrémně jemného obsahu (odstavce, věty a dokonce i jednotlivá slova), často k sestavení dokumentace nebo vysoce technického obsahu.

4.2.2.6 Systémy řízení učení (LMS)

Používá se pro správu učebních zdrojů a interakci studentů; většina vysokých škol a univerzit spravuje osnovy a proces učení prostřednictvím LMS. [12]

4.2.3 Funkce

4.2.3.1 Kontrola obsahu

CMS umožňuje získat kontrolu nad daným obsahem, pokud se obsah vymkne kontrole. CMS sleduje obsah a ví, kde je zvolený obsah, v jakém stavu se nachází, kdo k němu má přístup a jak souvisí s dalším obsahem. Dále se snaží zabránit tomu, aby se s daným obsahem nakládalo špatným způsobem.

CMS konkrétně poskytuje základní řídicí funkce, jako například:

4.2.3.2 Oprávnění

Kdo může tento obsah zobrazit? Kdo to může změnit? Kdo to může smazat?

4.2.3.3 Správa státu a pracovní postup

Je tento obsah publikován? Je to v návrhu? Byl archivován a odstraněn ze stránek?

4.2.3.4 Správa verzí

Kolikrát se tento obsah změnil? Jak to vypadalo před třemi měsíci?

Jak se tato verze liší od aktuální verze? Mohu obnovit nebo znovu publikovat starší verzi?

4.2.3.5 Správa závislostí

Jaký obsah používá jiný obsah? Pokud tento obsah odstráním, jak to ovlivní jiný obsah?

Jaký obsah je aktuálně nepoužívaný?

4.2.3.6 Hledání a organizace

Jak najdu konkrétní část obsahu? Jak najdu veškerý obsah, který odkazuje na X?

Jak mohu seskupit a propojit obsah, aby bylo jeho správu snazší?

Každá z těchto položek zvyšuje úroveň kontroly nad obsahem a snižuje riziko, že obsah bude vydán dřív, nebo že bude část obsahu smazána.

4.2.3.7 Opětovné použití obsahu

Používání obsahu na více než jednom místě a více než jedním způsobem zvyšuje jeho hodnotu. Některé příklady:

Zpravodajský článek se objeví na vlastní stránce, ale také jako ukázka na stránce kategorie a v několika postranních panelech „Související článek“.

Autorova biografie se objeví v dolní části všech článků napsaných touto osobou.

V těchto situacích se tato informace nevytváří pokaždé na každém místě, ale jednoduše se načte a zobrazí ze společného umístění.

Tento problém poněkud zmírnil server Side Includes, který editorům stránek umožňoval vložit úryvek kódu HTML pouhým odkazem na samostatný soubor - soubory byly před doručením na serveru zkombinovány. Pozdější platformy se to pokusily ještě více automatizovat; Například Microsoft FrontPage měl funkci, kterou výslovně nazval „Sdílené hranice“.

4.2.3.8 Automatizace a agregace obsahu

Mít veškerý obsah na jednom místě usnadňuje dotazování a manipulaci s ním. Pokud je požadováno najít všechny zpravodajské články, které byly napsány minulý týden, a zmínit slovo „SPECTRE“, lze to provést, protože existuje jeden systém, který „ví“ vše o daném obsahu.

Pokud je obsah správně strukturován, lze s ním manipulovat tak, aby se zobrazoval v různých formátech, publikovat jej na různých místech a průběžně jej přeskupovat tak, aby lépe sloužil potřebám návštěvníků:

Uživatelům je možné zpřístupnit obsah v jiných formátech, jako je PDF nebo jiné formáty e-knih.

Pro web lze automaticky vytvářet seznamy a navigaci (agregace obsahu)

Lze vytvořit více překladů obsahu, aby bylo zajištěno, že bude poskytnut jazyk nejvhodnější pro aktuálního uživatele.

Obsah, který je publikován, lze změnit v reálném čase na základě konkrétního chování a podmínek nastavených návštěvníky.

CMS to umožňuje strukturováním, ukládáním, zkoumáním a poskytováním kolem obsahu. Stává se jediným zdrojem informací o obsahu.

4.2.4 Existující systémy:

V dnešní době už existují už stovky různých řešení, které jsou schopné uspokojit potřeby a nároky dnešní doby. Jako příklad byly vybrány tři nejznámější CMS řešení, která budou následně popsány: Joomla!, WordPress a Drupal.

4.2.4.1 Drupal

Drupal byl vytvořen v roce 2000 holandským studentem Driesi Buytaertem. Důvodem vzniku systému byla potřeba sdílet informace s kolegy ze své koleje a následně systém dostal jméno Drop (2001). Prvotní veřejnou verzi pojmenoval Drupal a o jeho vývoj se starala skupina hlavních vývojářů a více než 400 uživatelů.

Drupal je založený na programovacím jazyce PHP, disponuje podporou pro různé databázové systémy MariaDB, MySQL, PostgreSQL. Při nutnosti a dalších úpravách funguje i OracleDB a MSSQL .

Hlavní předností Drupal je jeho modularita, bezpečnost, stabilita systému a v neposlední řadě jde o open source systém. (Drupal.cz: Český portál o open source CMS Drupal, 2021)

4.2.4.2 Joomla!

Jméno je převzato ze svahilského termínu jumla, jehož významem je „všichni dohromady“. Vybraný název má symbolizovat svazek mezi vývojáři a komunitou k

tomuto projektu. Prvotní verzi systému Joomla! 1.0.0 byla vypuštěna 16. září 2005. (JOOMLAPORTAL.CZ: Co je Joomla!?, 2012)

Joomla! je založený na programovacím jazyce PHP, primární databáze pro systém je MySQL ale je zde možnost fungování i na Apache (webovém serveru) tak i na Microsoft IIS. Od verze 2.5 byla přidána podpora pro další databázové systémy, PostgreSQL, SQLite, Oracle a další. Od verze 2.5 už zde není nutná závislost na určitém HTTP serveru, to umožňuje využít jakýkoliv server k provozu.

Je zde možnost využít až 64 jazykových mutací, mezi které je zařazen i český překlad. Hlavní přednosti Joomla! jsou ve variabilitě, velké vývojářské komunitě a možnosti správy obsahu. Nevýhodou je však chybějící možnost optimalizace SEO.

4.2.4.3 Wordpress

Wordpress byl vytvořen v roce 2003 se záměrem vytvořit elegantní, však dobře vystavěný personální publikační systém založený na PHP a MySQL s licencí GPLv2. V minulosti se na wordpress nahlíželo jako na blogovací systém, ale dnes je tomu jinak díky tisíci modulům, které může využívat. (Wordpress: Blog Tool, Publishing Platform, and CMS, 2021)

Předností WordPressu je aktivní vývojářská komunita, která je spojena s rozsáhlou technickou podporou. Licence GNU, kterou Wordpress má, umožňuje zdarma využívat a modifikovat kód systému.

Mezi nevýhody systému lze zařadit zranitelnost a dynamické generování stránek. Tím vzniká problém, kdy při velké zátěži dojde k nedostupnosti webu.(McNulty, 2009)

5 Vlastní tvorba webové stránky

Touto kapitolou začíná praktická část této práce a měla by popisovat tvorbu responzivních webových stránek s využitím CMS systému, kterým pro tyto webové stránky bude WordPress. V následujících kapitolách půjde o využití informací z teoretické části v praxi. Finální vzhled webových stránek je dostupný na webové adrese <https://www.vyuka-ict.eu/>. Všechny soubory k webovým stránkám jsou přiloženy v příloze.

5.1 Cíl práce

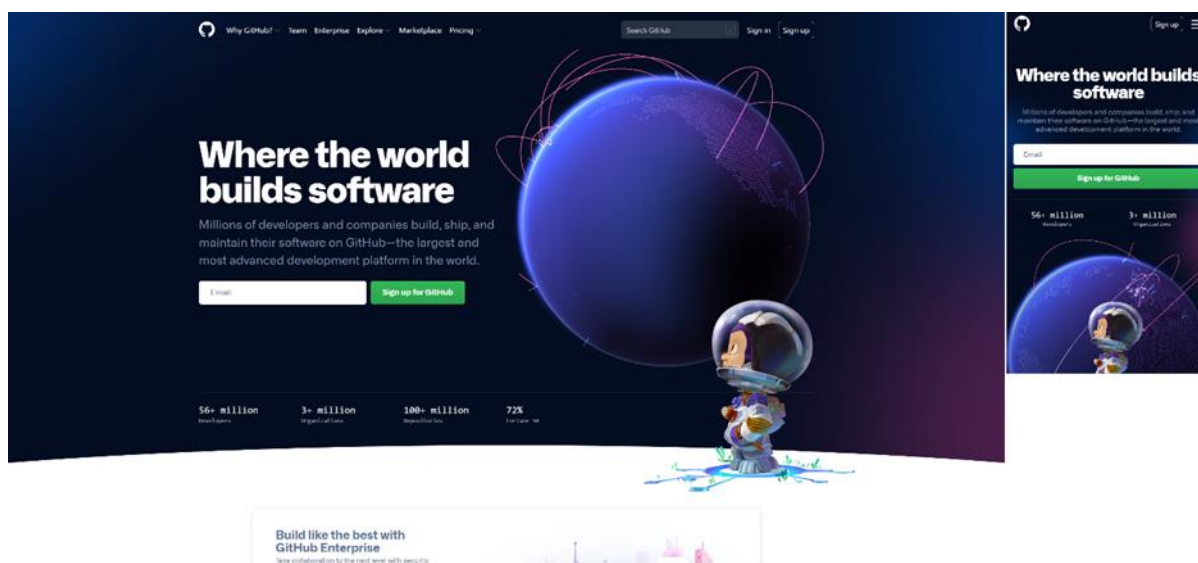
Hlavním cílem této bakalářské práce je vytvořit responzivní webové stránky pro katedru aplikované matematiky a informatiky, jejímž hlavním obsahem jsou informace o katedře a informace pro studenty stávající, tak i nové. Obsah webových stránek byl vytvořen na základě společné domluvy mezi mnou a vedením katedry. Za cíl jsem si vytyčil vytvoření webové stránky, které budou plně responzivní a taky optimalizované pro všechna zařízení, aby uživatel nebyl nijak omezen při prohlížení webových stránek. Další otázkou je design stránek, který bych chtěl udělat jednoduchý, moderní ale při tom elegantní.

5.2 Návrh řešení nových webových stránek

V dnešní době velkých monitorů se 4K (3840 × 2160) rozlišením až po malé displeje mobilů se skoro stejným rozlišením, je nutné hledat alternativu, která by byla schopná najít řešení jak pro velký 27 palcový monitor, tak i pro 6-ti palcový displej telefonu. Samozřejmě je prioritou vyvarovat se chybám jako je fixní layout, který bude fungovat jenom pro jeden typ zařízení, nebo zbytečně nevyužitý prostor po stránkách webové stránky, a to nelze zapomenout taky na font a jeho čitelnost. V poslední řadě zbývá obsah, který by měl být použitelný na všech typech zařízení, měl by být stručně členěn do odstavců a nadpisů, pro obrázky samotné je vždy volen spíše větší formát než menší (a není do nich vkládán text, může dojít ke zmenšení a nečitelnosti).

Pro začátek je snahou se vyvarovat tvorbě dvou verzí webových stránek jako bylo zvykem, kdy se tvořila druhá varianta webových stránek (např. m.jméno stránky.doména), protože docházelo k přidělení práce a vyhledávací algoritmus to považoval za duplicitní obsah. Z toho plyne, že bude zvolena druhá varianta a tou bude responzivní webová stránka, která bude pro všechny typy zařízení. Jako první bude tedy zvolen fluidní layout, který upraví šířku stránky podle velikosti aktuálního okna či typu

zařízení, na kterém je web prohlížen. Tím by byl vyřešen první problém ohledně měnících se velikosti displejů a jejich rozlišení (např. displeje s rozlišením 1920×1080 a 1280×800 mají stanovenou šířku 1170 pixelů, pro menší rozlišení dojde k automatickému zmenšení mřížky). Poté by se správně měl zvolit font, který by byl vhodný pro všechny typy zařízení. Najít takový font by bylo obtížné a časově náročné, tudíž byly pro různé typy zařízení zvoleny odlišné typy fontů, aby bylo dosaženo dobré čitelnosti obsahu. Poté bude vystavován obsah tak, aby ho bylo možné využívat na všech typech zařízení. Tím je myšleno, že pokud se změní velikost stránky na breakpointu, tak aby obsah, který byl v desktopové verzi, byl využit ve zmenšené verzi pro mobilní zařízení (viz obrázek 20).



Obrázek 8: Transformace obsahu pro dva různé typy zařízení

Zdroj:GitHub

5.3 Předpokládaný stav webových stránek

Základní vzhled webových stránek je tvořen úvodní stranou, která je doplněná o navigační menu, které má 7 tlačítek a jsou rozdělena do podskupin podle druhu informací. Úvodní stránka je tvořena úvodním obrázkem s popisem, pod ním jsou rychlo? odkazující tlačítka na nejpoužívanější stránky spojené se studiem (např. Stag, E-Learning, Webmail). Dále se zde nachází slider, který má v sobě určitá témata zajímavá pro návštěvníka s možností rozkliknutí pro více informací. V poslední části úvodu webové stránky jsou vytvořena tři okna, která budou obsahovat motivační text, jako například: proč si vybrat katedru ke studiu a celá stránka je zakončena třemi okny, ve kterých se ukazují nejnovější příspěvky z katedry.

Stránka je doplněna footrem neboli zápatím stránky, kde je logo a informace o katedře. Footer i header jsou tvoří samostatnou část z důvodu úspornosti, protože při každé změně by se musel měnit kód v každé stránce. Header neboli horní část každé stránky je tvořena navigačním menu, ve kterém máme následující odkazy: Uchazeč, Student, Aktuality, Členové katedry, Projekty, Kontakt a logo, které funguje jako odkaz na úvodní stránku. Uchazeč je v desktopové verzi dropdown menu, ve kterém se nachází odkazy na Bakalářské a Magisterské studium. Na druhou stranu v mobilní verzi funguje Uchazeč jako odkaz na stránku uchazeč, kde se nachází dvě okna, která odkazují na stejná témata jako dropdown menu. Student funguje na stejném principu jako uchazeč a jsou pod ním schované odkazy na Studentské praxe, Státní závěrečné zkoušky, Témata bakalářských prací a Témata magisterských prací. Aktuality jsou podstránka, na které jsou vyobrazeny všechny příspěvky, které byly zveřejněny a je zde možnost hledání a filtrování. Členové katedry jsou podstránka obsahující informace o zaměstnancích katedry (jako jméno, kancelář, e-mail či konzultační hodiny). Projekty jsou podstránkou, která obsahuje informace o projektech, ve kterých jsou zainteresovaní členové katedry. Poslední podstránkou je pak Kontakt, ve kterém najdeme adresu, kontaktní údaje jako e-mail, tel a fax a je to doplněno o mapu z google maps.

5.4 Využití technologie

5.4.1 PHP

PHP (rekurzivní zkratka pro PHP: Hypertext Preprocessor) je široce používaný open source univerzální skriptovací jazyk, který je zvláště vhodný pro vývoj webových aplikací a může být vložen do HTML. Výhodou je, že php je vykonáváno na straně serveru a na prohlížeči je vyobrazena už hotová stránka v HTML kódu. (PHP, 2001-2021)

5.4.2 CSS

CSS je jazyk pro popis prezentace webových stránek, včetně barev, rozložení a písma. Umožňuje přizpůsobit prezentaci různým typům zařízení, jako jsou velké obrazovky, malé obrazovky nebo tiskárny. CSS je nezávislý na HTML a lze jej použít s jakýmkoli značkovacím jazykem založeným na XML. CSS v HTML usnadňuje údržbu webů, sdílení šablon stylů mezi stránkami a přizpůsobení stránek různým prostředím. Toto se označuje jako oddělení struktury od prezentace. (W3C, 2021)

5.4.3 jQuery

jQuery je rychlá, malá a na funkce bohatá knihovna JavaScript. Díky snadno použitelnému rozhraní API, které funguje v mnoha prohlížečích, jsou věci jako procházení a manipulace s dokumenty HTML, manipulace s událostmi, animace a Ajax mnohem jednodušší. (jQuery, 2021)

5.4.4 Wordpress

WordPress je bezplatná platforma pro vytváření webových stránek s otevřeným zdrojovým kódem. Na více technické úrovni je WordPress systém pro správu obsahu (CMS) napsaný v PHP, který používá databázi MySQL. (Wordpress: Blog Tool, Publishing Platform, and CMS, 2021)

5.4.5 ElementorPro

Pro úpravy, které by byly potřeba udělat, byl zvolen ElementorPro builder. Jedná se o jednoduchý drag and build website builder, protože nebylo jasné, kdo se bude následně starat o chod stránek, byla tedy zvolena jednoduchá cesta úpravy webové stránky, kterou se naučí každý i bez znalostí a zkušeností v programování webových stránek. Navíc má skvělou kompatibilitu s vybranou šablonou.

5.4.6 Fontawesome

Font Awesome je sada fontů a ikon založená na CSS a LESS, jejím autorem je Dave Gandy. Poskytují okolo 8000 ikon, ale pouze 1609 z nich je možno využívat volně bez placení, zbytek je pod poplatkem 99\$. Icony a fonty jsou jednoduše použitelné a jejich úpravy jsou prováděny skrze CSS jak barva, velikost a další. (FontAwesome, 2021)

5.4.7 Prostředí vývoje

Jako prostředí pro vývoj byl využit hosting skrze firmu Active 24, u kterých jsem zakoupil hosting a doménu jsem již vlastnil, tudíž jsem ji pouze připojil k hostingu. Jsem rád že jsem vybral tuto firmu protože kvality služeb co poskytují jsou na velice dobré úrovni.

5.4.8 Editor kódu

Jako editor kódu byl využit především Sublime text, který mi osobně vyhovuje nejvíce a nabízí spoustu užitečných funkcí jako např. Python API rozhraní, které umožňuje zásuvným modulům rozšířit integrované funkce. Nebo je zde funkce instalace přídatných balíčků pomocí palety příkazů, což poskytuje jednoduchý přístup k tisícům balíčků vytvořených komunitou sublime textu. (SublimeText, 2020)

5.4.9 Google chrome nástroj pro vývojáře

Chrome DevTools je sada nástrojů pro vývojáře webových stránek zabudována přímo do Google Chrome prohlížeče, DevTools pomůže rychle upravovat stránky, ale taky testovat a diagnostikovat problémy, což v konečném důsledku pomůže rychleji vytvářet lepší weby.

5.5 Design wireframu

Wireframe neboli drátěný model stránek. Důvod pro tvorbu wireframe je jednoduchý: každý vývojář by měl mít představu, jak budou stránky vypadat z grafického hlediska, aby si sám dokázal udělat předběžnou strukturu stránek předtím, než je začne programovat. S porovnáním vytváření interaktivní ukázky webu je výroba wireframu mnohem jednodušší a časově je taky méně náročná.

V minulosti by stačil papír a tužka, ale při přidávání a odebírání určitých částí se musí následně celý návrh překreslovat, což není ideální řešení. Proto díky novým softwarovým nástrojům je dostupná možnost vytvářet vše virtuálně, což umožňuje jejich úplnou úpravu. Na internetu lze najít hned několik nástrojů pro tvorbu wireframů, proto si každý uživatel může vybrat to, co mu vyhovuje nejvíce. Po procházení několika webů s nástroji na tvorbu wireframů jsem narazil na vývojáře MockFlow, kteří mají ve svém portfoliu nástroj WireframePro. WireframePro je jednoduchý a přitom obsahuje dostatečné množství prvků pro tvorbu wireframu a kdyby náhodou zde nebyl, dá se jednoduše vyrobit z komponent, které software má. Wireframy byly vytvořeny pro většinu stránek, wireframy jsou vytvořeny pro desktopovou variantu, protože mi přišlo zbytečné je vytvářet pro verzi tablet a verzi mobile phone, responzivitu mezi těmito jednotlivými zařízeními bude řešena později v samotném wordpressu či kódu.

5.5.1 Úvodní stránka

Návrh úvodní stránky je k zobrazení v příloze A.1. V horní části stránky se nachází navbar neboli navigační menu, které je tvořeno logem uprostřed a od něj do stran jdou odkazy na jednotlivé podstránky: tři doprava a tři doleva. Levá strana je ještě doplněna o odkazy na sociální síť. Pod headrem stránky je umístěn úvodní obrázek, který má v sobě úvodní text (text není součástí fotky, ale prvkem, který je tam vložen, aby nedocházelo k problému s responzivitou). Pod našim obrázkem je pět stejných tlačítek, která odkazují na studenty nejvíce naštěvované stránky fakulty či univerzity (např. STAG, E-learning). Další na strance je Carousel. Carousel je jednoduše řečeno posuvník medií, v tomto

případě zde budou umístěny 3 obrázky s textem a odkazujícím tlačítkem. První a druhý obrázek by měl odkazovat na kategorii Uchazeč a třetí by měl odkazovat do kategorie Student. Jednotlivé obrázky by měly nějak zaujmout návštěvníka webu k danému tématu. Další v řadě jsou tři obdelníkové dlaždice, na kterých je umístěna tématická ikona (z knihovny FontAwesome) a pod ní text. Volba textu zaleží na způsobu použití, zde jsou voleny motivační texty a různé benefity plynoucí ze studia uvedené katedry. Předposlední na stránce jsou tři jednotlivá okna, která zobrazují nejnovější příspěvky (články) zveřejněné na webové stránce. Konec stránky je zakončen takzvaným footrem neboli patičkou stránky, který je tvořen dvěma pruhy: jeden je světlejší a širší – ten obsahuje informace o katedře, druhý je tmavší a užší – obsahuje pouze copyrights práva.

5.5.2 Uchazeč

Návrh pro stránku Uchazeč je k zobrazení v příloze A.2. Odkaz Uchazeč v nav-baru je tvořen takzvaným dropdown menu, to je rozbalovací menu, které se po najetí kurzorem rozbalí a nabídne další odkazy. Dalším prvkem je zde page header (záhlaví podstránky), kde lze nalézt nadpis stránky. Pod ním jsou vyobrazeny jednotlivá obdelníková okna, ve kterých jsou vypsané jednotlivé informace k vybranému tématu, a v každém okně je potom tlačítko, které v sobě má odkaz na následující podstránku k tématu. Návrh pro stránku Student má úplně stejný koncept jako stránka uchazeč, díky tomu nemusel být vytvořen wireframe pro stránku student.

5.5.3 Aktuality

Návrh pro stránku Aktuality je k zobrazení v příloze A.3. Tato stránka je tvořena dvěma horizontálními sloupci. Užší sloupec na levé straně tvoří nástroje, jeden z nástrojů slouží k hledání v příspěvcích či článcích, a druhý je facebookovým modulem napojeným na ekonomickou fakultu. Druhý a širší sloupec je pro příspěvky samotné, každý příspěvek je tvořen úvodním obrázkem a poté je nadpis článku. Pod nadpisem se zobrazuje dále krátký úryvek z textu článku a celé je to zakončené tlačítkem pokračovat ve čtení.

5.5.4 Členové katedry

Návrh pro stránku Členové katedry je k zobrazení v příloze A.4. Tato stránka je tvořena page headrem, který říká, na jaké jsme straně a pod ním jsou nadpisy, které symbolizují jednotlivé pozice na katedře (např. Vedoucí katedry, tajemník katedry, Technik) pak jsou i další nadpisy které už symbolizují skupiny lidí podle společného akademického titulu (např. Docenti) nebo i pozice (např. Doktorand). Pro vyobrazení jednotlivých členů jsem

zvolil tvar obdelníku kde vypíšu jméno, kancelář, konzultační hodiny, email a telefon. Obdelník je doplněn o kulatou fotku vybraného člena v horní části obdelníku.

5.5.5 Projekty

Návrh pro stránku Projekty je k zobrazení v příloze A.5. Tato stránka je vytvořena na základě komponenty Accordion. Začátek stránky má page header i s podnadpisem. Při načtení stránky se otevře první položka accordionu kde lze vidět informace k projektu, všechny ostatní sekce v accordionu jsou zavřené, takže pokud by bylo požadavkem zobrazit jiný projekt či téma, tak si stačí nalézt název projektu, kliknout na něj a rozbalí se. Tento typ designu se hodí na stránku, kam je potřeba dostat hodně obsahu, ale není žádoucí, aby obsah byl nepřehledný, a tudíž se nemusí vytvářet pro každý projekt nová stránka.

5.5.6 Kontakt

Návrh pro stránku Kontakt je k zobrazení v příloze A.6. Tato stránka je tvořena page headrem, který nám říká na jaké stránce se návštěvník vyskytuje. Poté je stránka rozdělena do dvou vertikálních sloupců. V levém sloupci se nachází adresa katedry, email, telefon a fax. Každá z informací je doplněna o ikonu z knihovny FontAwesome. Druhý sloupec slouží jako výřez pro Google maps s aktuální adresou katedry.

5.6 Vytvoření grafického stylu pro prvky webových stránek

Po dokončení wireframu celého webu by se teoreticky měl vytvořit grafický návrh jednotlivých komponent, které jsou na wireframech. Pokud by šlo o větší zakázku například pro nějakou firmu, řešilo by se Corporate Identity, což znamená, že je zapotřebí vytvořit styl vizuálně přizpůsobený vzhledu a image firmy. Pro můj projekt jsem tuto část vývoje vynechal, jelikož budu využívat šablonu Astra, která je volně dostupná v CMS Wordpress a základní grafické styly již obsahuje. Tyto styly budou postupně upravovány do vlastní výsledné podoby. Mezi výhody tohoto přístupu patří nízká časová náročnost a taky finanční stránka projektu, protože nemusíme zaplatit vývoj plně grafického návrhu webu, který není důležitý pro tento typ projektu. Jelikož jsou tyto webové stránky cílené spíše už na studující studenty než uchazeče, snažil jsem se vytvořit funkční stránky se zaměřením na obsah pro studenty (např. STAG, E- Learning, konzultační hodiny učitelů).

5.7 Začátek tvorby webových stránek

Po dokončení návrhu drátového modelu a dokončení grafické varianty stránek se pomalu lze posunout k tvorbě zdrojového kódu podle návrhů. Normálně by bylo nutné vytvořit

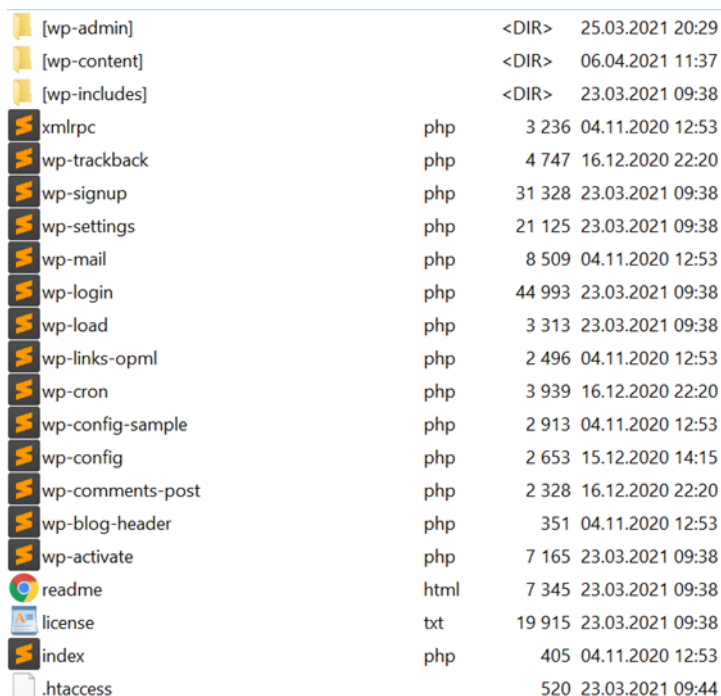
všechny stránky v HTML a tím konstrukci webových stránek, a poté přes CSS by byl dodán grafický design, který byl vytvořen v návrzích. V této bakalářské práci se ale totiž nejprve začne s instalací CMS na lokální server nebo hosting.

5.7.1 Příprava a instalace Wordpressu

Vybraným CMS je Wordpress, instalace Wordpress není nějak náročná, hostingy jí dělají pro své klienty sami. Pokud by se ale instaloval wordpress na lokální server, musí se stáhnout aplikace na tvorbu databáze (např. XAMPP), vytvoří se databáze, do které se nakopírují soubory wordpressu, které byly staženy. Poté se otevře testovací stránka, kde se automaticky spustí instalace wordpressu a následně se s ní udělají jednotlivé kroky jako je volba jazyka, nastavení jména databáze viz obrázek 9 na další straně.

5.7.2 Příprava wordpressu a jeho základní struktura

Před začátkem vytváření webové stránky je třeba určit, jak bude webová stránka strukturována a jak by měla fungovat. Měla by být známa struktura souborů wordpressu a poté i samotné šablony, ve které budou uloženy data k webové stránce, aby se mohly využívat data plnohodnotně a pro třetí stranu, která bude projekt číst poprvé a budou se s ním seznamovat tak, aby měly představu, jak by to mělo vypadat (viz obrázek 8).



[wp-admin]	<DIR>		25.03.2021 20:29
[wp-content]	<DIR>		06.04.2021 11:37
[wp-includes]	<DIR>		23.03.2021 09:38
xmlrpc	php	3 236	04.11.2020 12:53
wp-trackback	php	4 747	16.12.2020 22:20
wp-signup	php	31 328	23.03.2021 09:38
wp-settings	php	21 125	23.03.2021 09:38
wp-mail	php	8 509	04.11.2020 12:53
wp-login	php	44 993	23.03.2021 09:38
wp-load	php	3 313	23.03.2021 09:38
wp-links-opml	php	2 496	04.11.2020 12:53
wp-cron	php	3 939	16.12.2020 22:20
wp-config-sample	php	2 913	04.11.2020 12:53
wp-config	php	2 653	15.12.2020 14:15
wp-comments-post	php	2 328	16.12.2020 22:20
wp-blog-header	php	351	04.11.2020 12:53
wp-activate	php	7 165	23.03.2021 09:38
readme	html	7 345	23.03.2021 09:38
license	txt	19 915	23.03.2021 09:38
index	php	405	04.11.2020 12:53
.htaccess		520	23.03.2021 09:44

Obrázek 9: Struktura wordpress na serveru

Zdroj: autor



Vítejte ve WordPressu. Ještě, než zahájíte jednoduchou instalaci, tak se prosím ujistěte, že máte k dispozici základní informace o databázi. Potřebovat budete následující údaje:

1. Název databáze
2. Uživatelské jméno k databázi
3. Heslo pro přístup k databázi
4. Databázový server
5. Předpona tabulek (pokud chcete provozovat několik různých instalací WordPressu pod jednou databází)

Zmiňované informace jsou potřeba, aby mohl být vytvořen soubor `wp-config.php`. **Pokud by automatické vytvoření konfiguračního souboru z nějakého důvodu selhalo, tak se přesto nemusíte ničeho obávat, protože se instalační skript snaží pouze doplnit informace o databázi do konfiguračního souboru. V případě nečekaných problémů můžete snadno otevřít soubor `wp-config-sample.php` v textovém editoru, potřebné informace tam zapsat ručně a následně soubor uložit pod názvem `wp-config.php`.** Chcete s něčím poradit? Zkuste podrobný [návod](#) (anglicky).

Kompletní informace o databázi byste měli získat od poskytovatele hostingu a pokud je z nějakého důvodu nemáte, tak musíte nejdříve kontaktovat jejich technickou podporu. Až budete mít potřebné údaje připravené...

[Pokračovat!](#)



Zadejte prosím následující informace o připojení k databázi. Pokud uvedené údaje neznáte nebo něčemu nerozumíte, kontaktujte poskytovatele hostingu.

Název databáze	<input type="text" value="test"/>	Název databáze, kam chcete instalovat WordPress.
Uživatelské jméno	<input type="text" value="root"/>	Uživatelské jméno k databázi.
Heslo	<input type="text"/>	Heslo pro přístup k databázi.
Databázový server	<input type="text" value="localhost"/>	Pokud nebude fungovat výchozí hodnota localhost, tak musíte použít informace od poskytovatele hostingu.
Předpona tabulek	<input type="text" value="wp_"/>	Pokud chcete provozovat několik různých instalací WordPressu pod jednou databází, tak musíte zvolit různé předpony tabulek.

[Odeslat](#)

Obrázek 10: Instalace wordpressu

Zdroj:autor

Tento projekt bude vytvářen ze šablony, která je volně dostupná a není nějak omezená právy ke komerčnímu využití a další distribuci, což je jediné plusem. Naše šablona se jmenuje Astra a byla vytvořena tvůrci ze skupiny Brainstorm force. Jedná se o plně přizpůsobitelnou šablonu pro wordpress, je vhodná na tvorbu blogu, osobního portfolia nebo i obchodního webu. Šablona je důmyslně vymyšlená, protože v základní formě pracuje s 50kb na frontendu, což znamená jednu z bezkonkurenčních rychlostí nabízených šablon. Nutno připomenout vyhledávací algoritmy prohlížečů, které je nutné vzít v potaz, když je vytvářena nová stránka. Astra šablona je vytvořena s ohledem na SEO, přichází s integrovaným kódem Schema.org a také je připravena na nativní AMP (Accelerated Mobile Pages), takže pro vyhledávač by měli být stránky nepřehlédnutelné.

5.7.3 Stažení a příprava šablony

Po nainstalování wordpressu a jeho nastavení je možno přejít do záložky vzhled, kde je možno vybrat šablony. Zde bude zobrazena předinstalovaná šablona s názvem twenty twenty-one, kterou lze smazat, aby nebyly drženy zbytečně data na serveru. Poté se v pravém horním rohu nachází vyhledávací okno, do kterého je zadáno jméno dané šablony. Na obrazovce nám vyběhne výsledek vyhledávání, v tomto případě bude vyhledána Astra šablona, a poté bude spuštěna instalace. Po instalaci se šablona aplikuje a aktuální vzhled by měl být stejný jako úvodní obrázek na šabloně. Pokud se tak nestane a stránka si zachová prvky předchozího předinstalovaného designu, nejde o nic nevyřešitelného. Existují dvě varianty, co následně dělat: buď manuálně všechno smazat a poté začít s vlastní tvorbou, což bude případ pro tyto webové stránky a nebo jednodušší variantou je využít některý z nabízených pluginů od wordpressu (např. Starter template), který tuhle práci udělá za vás a stránku upraví přesně podle toho, jak byla na úvodním demu. Na obrázku 10 je znázorněna struktura samotné šablony a co by měla obsahovat.

[assets]	<DIR>	06.04.2021 13:23
[inc]	<DIR>	06.04.2021 13:22
[languages]	<DIR>	06.04.2021 13:23
[template-parts]	<DIR>	06.04.2021 13:23
toolset-config	json	9 022 06.04.2021 13:22
style	css	3 141 06.04.2021 13:22
single	php	740 06.04.2021 13:22
sidebar	php	942 06.04.2021 13:22
search	php	820 06.04.2021 13:22
screenshot	jpg	87 192 06.04.2021 13:22
readme	txt	4 765 06.04.2021 13:22
page	php	918 06.04.2021 13:22
index	php	962 06.04.2021 13:22
changelog	txt	60 800 06.04.2021 13:22
header	php	1 345 06.04.2021 13:22
functions	php	6 295 06.04.2021 13:22
footer	php	655 06.04.2021 13:22
comments	php	3 411 06.04.2021 13:22
archive	php	778 06.04.2021 13:22
404	php	732 06.04.2021 13:22

Obrázek 11: Struktura šablony pro wordpress

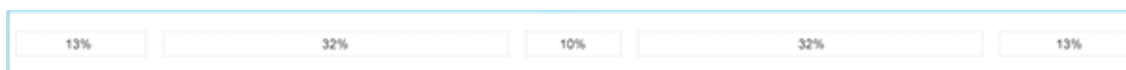
Zdroj: autor

5.8 Tvorba jednotlivých komponent a jednotlivých stránek

Po smazání všech komponent a vytvoření čisté šablony lze začít s tvorbou jednotlivých komponent jako je Header, Footer a potom styly jednotlivých stránek.

5.8.1 Tvorba headru

Header je horní část stránky, která obsahuje logo, navigační menu a odkazy na sociální síť. Pro tuto práci bude vytvořen header vlastní a nebude využit přednastavený. První věcí je, že všechny komponenty, které nejsou ve stránce přímo v jednotlivých stránkách, se spravují zvlášť, jelikož wordpress sám o sobě nedokáže pracovat zvlášť s footrem ani s headrem. Proto může být využit header and footer builder, který je zabudován do šablony astra a nebo si lze stáhnout Plugin ElementorPro, který s tím pomůže. Následně je vytvořena šablona pro header, jelikož už design existuje z wireframu, lze začít s tvorbou. Prvně je vytvořen div box, který v sobě má mřížku o 5 sloupcích s procentuálním rozložením viz obrázek 11.



Obrázek 12: Ukázka mřížky s procentuálním rozdělením

Zdroj: autor

Je známo, že podle wireframu přílohy A.1 header stránky obsahuje dvě vodorovná menu s logem fakulty uprostřed, doplněné o dvě tlačítka sociálních sítí po levé straně. Logo je vytvořeno v grafickém editoru a následně bylo vloženo do stránky. K tvorbě menu je zapotřebí se vrátit zpátky do wordpressu, kde se zvolí odrážka, vzhled a tam se vybere menu. To způsobí přesun na nové okno ve wordpressu, kde se nachází předchozí menu, které může být smazáno a následně může být vytvořeno vlastní menu. Budu vytvářet celkově 3 menu, a to dvě menu pro desktopovou verzi webu a poté jedno mobilní menu, které bude aktivní při překročení break pointu 1025px šířky. Po dokončení se lze vrátit zpátky k div boxu, kde bude všechno zkompletováno a vznikne header, kterému přes css styly budou upraveny barvy a design.

5.8.2 Tvorba footru

Tvorba footru je v podobném stylu jako tvorba headru, takže jsem se zaměřil spíše na rozdíly ohledně tvorby. Design pro footer se skládá ze dvou div boxů, kde první je větší a má tří sloupcovou mřížku, do které bude vytvořeno logo fakulty s nápisem a kontaktní informace dle wireframu, a poté druhý, který má jinou barvu a je 80px vysoký a slouží jako designový prvek, do kterého je vložen odkaz na autorské právo v podobě © Copyright 2021. Design footru bude viz obrázek 12.



Obrázek 13: Ukázka designu a rozložení footru

Zdroj: autor

5.9 Tvorba jednotlivých stran

Základní struktura pro zobrazení všech stránek z šablony je ve výpisu 1, který se nachází na další stránce. Chtěl jsem zde zahrnout kódy k jednotlivým komponentám, ale wordpress data ke stránkám a jejich částem ukládá do databáze jako meta data, které lze

najít v databázi skrze aplikaci MyPHPAdmin, ale k samotnému kódu se není možno dostat, jelikož by byly zapotřebí meta klíče.

Výpis 11: Základní kód templatu pro zobrazení všech stran [Brainstorm force]

```
<?php
if ( ! defined( 'ABSPATH' ) ) {
    exit; }
get_header(); ?>
<?php if ( astra_page_layout() == 'left-sidebar' ) : ?>
    <?php get_sidebar(); ?>
<?php endif ?>
    <div id="primary" <?php astra_primary_class(); ?>>
        <?php astra_primary_content_top(); ?>
<?php astra_content_page_loop(); ?>
        <?php astra_primary_content_bottom(); ?>
</div><!-- #primary -->
<?php if ( astra_page_layout() == 'right-sidebar' ) : ?>
<?php get_sidebar(); ?>
<?php endif ?>
<?php get_footer(); ?>
```

5.9.1 Úvodní stránka

Při tvorbě stylu pro úvodní stránku bylo zapotřebí zajistit efektivnost a jednoduchost pro uživatele stránek, kterými budou převážně již studující studenti než uchazeči. Proto začátek stránky je jednoduchý, úvodní obrázek s popisem je doplněn o 5 tlačítek, které odkazují na nejpoužívanější stránky spojené se studiem. Pro zachování jednoduchého moderního designu jsem zvolil Css carousel, který má tři slide v sobě určené k zaujmutí právě nové studenty, kteří by navštívili stránky katedry. Grafika do carouselu byla vytvořena mnou. Následně jsem zvolil tři dlaždicová okna, do kterých jsem chtěl zakombinovat motivační texty pro nově příchozí studenty. Konec stránky je zakončen komponentou pro zobrazování nejnovějších příspěvků s upraveným CSS stylem tak, aby ladil ke zbytku designu úvodní stránky.

5.9.2 Blog

Je vytvořen z wordpressového základu, jelikož wordpress byl původně hlavně k tvorbě blogů, jedná se o systém, který má jednoduchou funkčnost, takže ohledně této stránky byly provedeny jenom designové úpravy tak, aby zapadaly do designu webu. A byl zde zaimplementován i facebookový modul, který zobrazuje nejnovější příspěvky z fakultního facebookové účtu.

5.9.3 Členové katedry

Členové katedry mají jednoduchý vzdušný design, který je přehledný a při tom elegantní. Pro tvorbu toho designu jsem zvolil 3 sloupcové rozložení mřížky, kde jsem využil obdelníkových dlaždic pro zápis jednotlivých informací pro danou osobu a dlaždice je doplněna o kulatou fotku s mínusovým marginem, aby byl vytvořen efekt overlappingu.

5.9.4 Projekty

Projekty jsou stránkou, kde je poměrně hodně obsahu a aby i tak bylo dosaženo přehlednosti, je zde zvolena komponenta accordion. Kód komponenty bohužel nedokážu vypsat kvůli ukládání dat v databázi, pouze zde uvedu kód accordionu, který je funkční viz výpis 12.

Výpis 12: Kód accordionu s css stylem a skriptem [autor]

```
<body>
<button class="akordion">Kapitola 1</button>
<div class="textpanel">
  <p>Pokus1 pokus2 pokus3.</p>
</div>
<button class="akordion">Kapitola 2</button>
<div class="textpanel">
  <p>Pokus1 pokus2 pokus3.</p>
</div>
<button class="akordion">Kapitola 3</button>
<div class="textpanel">
  <p>Pokus1 pokus2 pokus3.</p>
</div>
<script>
var acc = document.getElementsByClassName("akordion");
var i;

for (i = 0; i < acc.length; i++) {
  acc[i].addEventListener("click", function() {
    this.classList.toggle("active");
    var textpanel = this.nextElementSibling;
    if (textpanel.style.display === "block") {
      textpanel.style.display = "none";
    } else {
      textpanel.style.display = "block";
    }
  });
}
</script>
</body>
```

```
<style>
.akordion {
background-color: #000000;
color: #ffffff;
padding: 20px;
width: 100%;
text-align: left;
font-size: 15px;
}
.active, .akordion:hover {
background-color: #8c8c8c;
}
.textpanel {
padding: 0 20px;
display: none;
background-color: #ffffff;
}
</style>
```

5.9.5 Kontakty

Zde jde pouze o vytvoření kontaktních informací, doplnit je o ikonu z knihovny font awesome a poté zde doplnit modul mapy od google maps. Jsou zde dvě možnosti jak modul google maps vložit do stránek: jeden je pomocí kódu a musí se vytvořit v Googlu Google Maps api key, který se pak využívají přes script, zde je vhodnější volbou druhá možnost a to jsou generátory Iframů, které vygenerují kód, který poté vložíme do naší stránky viz výpis 13.

Výpis 13: Kód Iframu s google maps informacemi [Maps.ie]

```
<div style="width: 100%">
<iframe width="100%" height="600" frameborder="0" scrolling="no" marginheight="0"
marginwidth="0"
src="https://maps.google.com/maps?width=100%25&height=600&hl=en&q=students
k%C3%A11%20787+(My%20Business%20Name)&t=&z=16&ie=UTF8&iwloc=B&a
mp;output=embed">
</iframe>
<a href="https://www.maps.ie/route-planner.htm">Driving Route Planner</a></div>
```

5.10 Testování

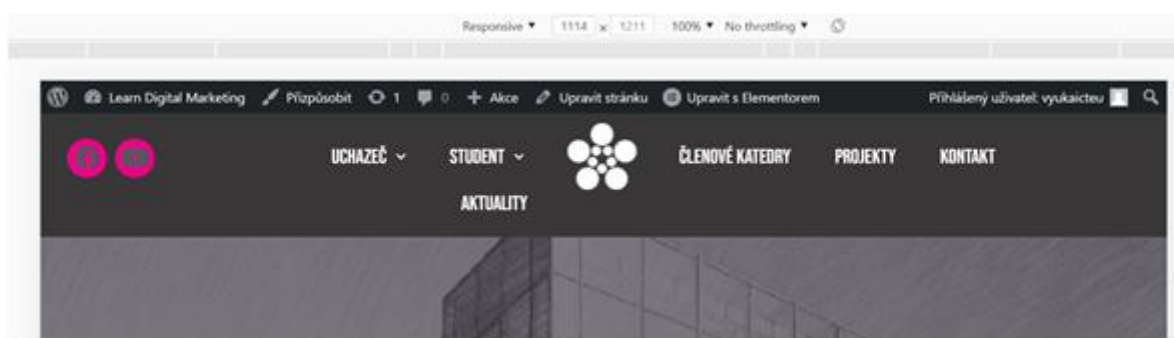
Poslední nedílnou součástí procesu tvorbry responzivních webových stránek je samozřejmě testování. Responzivita stránky samotná bude testována jako první, a to při měnícím se rozlišení, aby byly nalezeny chyby pro jednotlivá zařízení. Druhým typem testování budou desktopové prohlížeče, dále se bude testovat web na reálných mobilních zařízeních a různých prohlížečích. Nabízí se možnost provádět testování skrze emulátor, aby se dokázal vynahradiť nedostatek zařízení, které máme k dispozici. Dále je nutno se zaměřit na typy problémů, zdali je řešitelný nebo ne. Neřešitelným problémem může být prohlížeč nepodporující nové technologie nebo může jít o zastaralou verzi prohlížeče, která už nedostává aktualizace. Naopak řešitelným problémem může být špatně nadeřinované menu, které přetéká kam nemá nebo špatně nadeřinovaný breakpoint pro jiné zařízení.

5.10.1 Testování stránek při různých rozlišení

K testování stránek ohledně responzivity existuje mnoho nástrojů. Zde vybraným nástrojem bude asi ten, který se celkově využívá nejvíce a tím je google devtools. Tento nástroj nabízí hned několik možností, jak testovat stránky, a to formu přednastavených zařízení (např. Google pixel 2, iPhone SE, iPhone X nebo iPad Pro) nebo manuálně nastavit rozměry stránky. Existují i další stránky a softwary, které poskytují podobné

nástroje, příkladem může být Responsive Design Checker, který dává možnost vyzkoušet nejpoužívanější produkty na trhu jako monitory různých velikostí či nejnovějších mobilní zařízení či nejnovějších tabletů.

Při průběhu testování byl objeven problém s levým menu, které se těsně před breakpointem rozhodlo zmizet nebo přeteklo do dvou řad a začalo zakrývat obsah obrázku viz obrázek 26. Problém se vyřešil jednoduchým řešením upravením rozložení sloupce, ve kterém je menu zasazené a mírnou úpravou paddingu textu.



Obrázek 14: Testování responzivity webové stránky

Zdroj: autor

5.10.2 Testování různých prohlížečů pro desktopové zařízení

Pro testování různých webových prohlížečů byly vybrány ty, které jsou řazeny mezi nejpoužívanější prohlížeče. Pro zajímavost jsem zařadil již nepoužívaný Internet explorer a jeden bezpečnostní prohlížeč Tor. Jednotlivé výsledky tohoto testování jsou zapsány v tabulce 2 na další stránce. Hlavními parametry pro testování jsou: Správné chování responzivity webu, správné načtení jednotlivých částí webu, správné zobrazení css stylů a celkového designu. Při testování byl brán zřetel na správné fungování jednotlivých komponent, které jsou na stránkách, jako je například carousel nebo iFrame atd. Hodnocení testování bylo vyhotoveno mnou a obsahuje tedy mé osobní hodnocení k dané problematice. Testování dopadlo výborně s výjimkou zastaralého Internet Explorer, který nám nezobrazoval jednotlivé prvky stránky.

OS	Prohlížeč	Verze	Responzivita	Fonty a ikony	Dodatky	Hodnocení
Win10	Opera	75.0.0	100%	100%	Nejsou	100%
Win10	Chrome	89.0.4	100%	100%	Nejsou	100%
Win10	Firefox	87.0	100%	100%	Nejsou	100%
Win10	IE	11.0	50%	50%	Stránka nenačítá určité části	20%
Win10	Edge	89.0.7	100%	100%	Nejsou	100%
Win10	Tor	9.0	100%	100%	Nejsou	100%
Os Big Sur	Safari	14.0.0	100%	100%	Nejsou	100%
Os Big Sur	Chrome	89.0.4	100%	100%	Nejsou	100%

Tabulka 2: Testování responzivního chování v různých prohlížečích

Zdroj: Autor

5.10.3 Testování reálných tabletů a mobilných zařízení

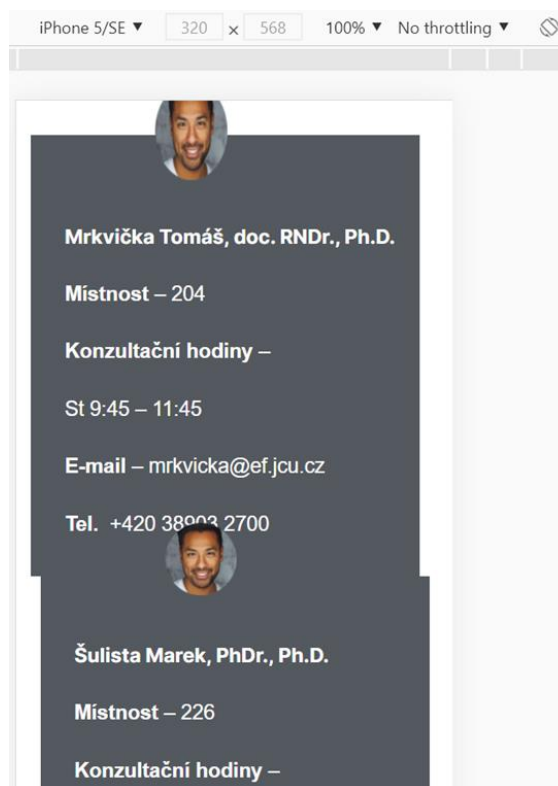
Druhá vlna testování proběhla na reálných zařízeních s různými prohlížeči, které bylo možno získat pro testování viz tabulka 3. Jednotlivá zařízení se lišila parametry hardwaru (např. rozdílná velikost displeje či typ panelu využitého do displeje). Systémy jednotlivých zařízení byly různé, za systémy Android byly využity verze 9.0 oreo, 10.0 a 11.0, která byla na tabletu od značky Samsung. Za značku Apple byla využita nejnovější verze systému kterou je 14.4.2. Pro ostatní systémy jako je Windows, Symbian a Blackberry nebylo získáno zařízení ve fyzické podobě a nebyl nalezen ani emulátor, který by je dokázal nahradit. Samotné testování probíhalo v pořádku. V některých případech se našly chyby, jednalo se například o to, že nové prohlížeče jako je Opera Touch a Xiaomi browser už po nainstalování mají aktivovaný takzvaný dark mode, který by měl ušetřit energie a celkově snižovat zatížení na zařízení. Pro testování to ale znamenalo, že když byla otevřena webová stránka v dark módu, tak se invertovaly barvy pozadí a pár komponent z bílé do černé a většina písma přešla z černé barvy do bílé. Ve výsledku to na responzivní funkci stránky nemělo vliv, pouze to narušovalo styl webu.

Zařízení	Prohlížeč	Verze	Responzivita	Fonty a ikony	Dodatky	Hodnocení
Huawei Mate 20 Pro	Opera Touch	2.9.4-	100%	100%	Nejsou	100%
Huawei Mate 20 Pro	Opera Mini	55.0.0	100%	100%	Nejsou	100%
Huawei Mate 20 Pro	Chrome	89.0.4	100%	100%	Nejsou	100%
Huawei Mate 20 Pro	Firefox	87.0	100%	100%	Nejsou	100%
iPhone 6S	Safari	11.0	50%	50%	Nejsou	100%
iPhone 6S	Chrome	87.0.4	100%	100%	Nejsou	100%
Samsung Galaxy S7	Samsung Internet	14.0	100%	100%	Nejsou	100%
Xiaomi Redmi Note 8	Xiaomi browser	12.9.3-	100%	100%	Nejsou	100%
iPad Pro	Safari	14.0	100%	100%	Nejsou	100%
Galaxy tab s7	Samsung Internet	12.0	100%	100%	Nejsou	100%

Tabulka 3: Testování responzivního chování na mobilních zařízeních

Zdroj: Autor

Další chybou, která se objevila u testování, byla responzivní chyba u podstránky členové katedry, kde se začal hroutit styl stránky do sebe a doslova z toho vznikl „flek“ s přeházenými informacemi viz obrázek 13. Řešení problému bylo v celku jednoduché, šlo pouze o upravení marginu na jednotlivých sloupcích mřížky, které se poté poskládají do požadovaného vzhledu a následně se musel vytvořit nový design pro toto zobrazení, jelikož se původní design smazal při úpravě marginu. Jinak žádné jiné problémy se neobjevily a testování dopadlo v celku dobře.



Obrázek 15: Responzivní problém stránky členové katedry

Zdroj: Autor

5.10.4 Zkouška testování s emulátorem virtuálních zařízení

Jelikož může nastat situace, kdy se při testování nelze dostat ke všem zařízením, na kterých by bylo žádoucí vyzkoušet hotový produkt, nabízí se na internetu emulátory zařízení, které se dají stáhnout jako software do počítače nebo je lze využívat přímo v prostředí prohlížeče. Je zapotřebí brát v potaz, že se nejedná o reálné přístroje, u kterých lze řešit zatížení zařízení při testování, což by mohlo mít jiné výsledky než v emulátoru. Mezi nejvyužívanější patří emulátor Browserstack, který nabízí testování až na 200 různých typech zařízení. Bohužel pro využití této služby je nutné platit měsíční paušál, jinak je povolena jenom jedna minuta simulace, a to na necelé polovině typů z celkově nabízených. Při testování se vyskytl problém pouze u iPad Pro 12, který má někde v systémovém nastavení automatické zvýrazňování telefonních čísel, což působí v našem designu ve výsledku špatně, ale jinak stránky fungují správně. Výsledky testování jsou v tabulce 4.

Zařízení	Prohlížeč	Verze	Responzivita	Fonty a ikony	Dodatky	Hodnocení
iMac	Safari	14.0.0	100%	100%	Nejsou	100%
Windows 10	Chrome	89.0.4	100%	100%	Nejsou	100%
iPhone 12 Pro	Safari	14.0.0	100%	100%	Nejsou	100%
iPad Pro 12	Safari	14.0.0	100%	100%	Zvýraznění telefonních čísel	100%
Pixel 5	Chrome	89.0.4	50%	50%	Nejsou	100%
OnePlus 8	Chrome	89.0.4	100%	100%	Nejsou	100%

Tabulka 4: Testování responzivního chování ve virtuálních zařízeních

Zdroj: Autor

5.10.5 Testování rychlosti načtení webu a hodnocení

Rychlost načítání webové stránky bylo testováno nástrojem jménem Pingdom. Výsledek byl uspokojivý navzdory známce C se skóre 77 bodů a stanoveným průměrným časem na načtení stránky 1.43 sekundy. Problém proč web získal špatné skóre není způsoben jeho strukturou či kódem nebo samotným designem, nýbrž faktem, že většina grafiky byla vytvořena mnou a bohužel není optimalizována pro web, ale je vyexportována v původní velikosti ve formátu .png. Proto soubor, který by měl mít v nejlepším případě 500 Kb má 18 Mb, což způsobuje velký datový skok při načítání největší části obsahu na stránce a snižuje skóre webu.

5.10.6 Rekapitulace výsledků z testování

Záměrem testování bylo vyzkoušet funkčnost webových stránek, a to po stránce doby načítání, struktury kódu, ale také po stránce samotné responzivity a zobrazování na zařízení různých typů. Pro vyzkoušení načítání stránky a zkontrolování struktury kódu a stránky samotné jsem využil nástroj Pingdom, který stránku zhodnotil dobře a vytknul stránce pouze formát a media typ určitých medií na webu. První fáze testování byla zaměřena na responzivitu a chování na desktopových zařízeních, kde si stránky vedly skvěle až na zastaralý internet explorer, který nám nenačítal jednotlivé části webu. Jinak se problémy neobjevily. Při druhé fázi testování, která byla zaměřena na testování mobilních zařízení a tabletů, se vyskytlo více problémů, ale nic nebylo závažným problémem, který by měl řešení nebo by omezil chod stránek. Za zmínku by asi stalo rozhození celého designu stránky a slítí obsahu do jednoho u stránky členové katedry. Z mého osobního hlediska dopadly výsledky testování stránek výborně a jsem příjemně spokojen s výsledkem.

6 Závěr

V závěru bych rád shrnul, cíl této bakalářské práce a její výsledky ke kterým jsem došel.

Cílem této bakalářské práce bylo navrhnout a vytvořit funkční responzivní webové stránky pro katedru aplikované matematiky s využitím redakčního systému Wordpress. Dále jsem se zaměřil na samotnou optimalizaci webových stránek pro mobilní zařízení, jelikož mobilní zařízení je v celosvětových statistikách jako nejpoužívanější zařízení na prohlížení webu.

Teoretická část bakalářské práce byla zaměřena na tematiku responzivní designu a optimalizace pro různá zařízení. Jsou zde vysvětleny základní pravidla a techniky, která jsou spojena se samotným responzivním designem. Mezi hlavní bych zařadil fluid grid layouty, práce s poměrem pixelů v displeji a v poslední řadě breakpoints a media queries. Jednotlivá pravidla jsou popsána a vysvětlena, jsou zde i praktické příklady využití. Další kapitolou které je spojena s našim tématem jsou CMS neboli systémy pro správu obsahu. CMS umožňuje získat kontrolu nad našim obsahem ale nabízí i nesmírné zjednodušení práce.

V praktické části bakalářské práce jsem se zaměřil na návrh designu pro samotné webové stránky a poté i na samotný vývoj a implementaci CMS do webové stránky. Design jsem prvotně vytvořil skrze wireframy a poté jsem si ho doplnil o barvy a vytvořil styl ve kterém stránka bude vytvořena. Samotná tvorba webu proběhla bez větších problémů, vyskytlo se pouze pár minoritních problémů ale nic neřešitelného nenastalo. Tvorba obsahu a vytváření reponsivních částí šlo podle plánu až na pár výjmeck, které způsobily následnou změnu formy obsahu. Implementace CMS do webové stránky proběhla bez problémů, nutné bylo pouze redesignovat styl pro stránku aktuality, která je vytvořena přímo pro redakční systém, kde zobrazuje nejnovější příspěvky a články.

Stanovené cíle na začátku práce byly splněné v plném rozsahu. Hlavní benefitem a přínosem mé práce je rozšíření vědomostí v oblasti vývoje webových stránek s využitím responzivního designu a implementace daného CMS. Dalším přínosem této práce je popis procesu vývoje webových stránek a jednotlivých částí které jsou s tímto procesem spojené, od návrhu až po finální testování stránek.

Summary and keywords

The bachelor's thesis deals with development of website for Department of Applied Mathematics and Informatics. The thesis is divided into theoretical part and practical part. The first part is designed to explore current trends connected to web design. It shows a general overview of the development of programming languages through the time.

The practical part of thesis shows development of modern website for Department of Applied Mathematics and Informatics with use of content management system. It shows a concrete design and whole process of developing with using PHP, CSS, JS. It also show finding a suitable content management system and the actual implementation of the selected content management system. At the end it includes evaluation of usage the website in real environment.

Keywords: Web design, websites, CMS, HTML, development

Seznam literatury

Sharkie, C., & Fisher, A. (2015). Responzivní webdesign: okamžitě. Computer Press.

Niederst Robbins, J. (2018). Learning web design: a beginner's guide to HTML, CSS, Javascript, and web graphics (Fifth edition). O'Reilly.

Křížek, Z., & Crha, I. (2003). Jak psát reklamní text (2., výrazně rozš. a přeprac. vyd). Grada.

Frain, B. (2020). Responsive Web Design with HTML5 and CSS: Develop future-proof responsive websites using the latest HTML5 and CSS techniques (Third edition). Packt Publishing.

Marcotte, E. (2015). Responsive Design: Patterns & Principles: Patterns & Principles. A Book Apart.

McNulty, S. (2009). WordPress: efektivní publikování na webu. Zoner Press.

Elektronické zdroje

Internet Users. (2011). Internet Live Stats. Retrieved February 03, 2021, from <https://www.internetlivestats.com/internet-users/>

Čevelíček, M. (2018). Liquid Design. Návrh informační architektury webu. Jak na to? Retrieved April 08, 2021, from <https://www.lqd.cz/clanek/jak-vytvorit-navrh-informacni-architektury-webu>

Ferguson, N. (2021). What's The Difference Between Frontend And Backend Web Development?. Careerfoundry. Retrieved April 08, 2021, from <https://careerfoundry.com/en/blog/web-development/whats-the-difference-between-frontend-and-backend/>

Statcounter: Desktop vs Mobile vs Tablet Market Share Worldwide. (1999 - 2021). Statcounter. Retrieved April 08, 2021, from <https://gs.statcounter.com/platform-market-share/desktop-mobile-tablet>

Marcotte, E. M. (2010). Responsive web design. A list apart. Retrieved April 08, 2021, from <https://alistapart.com/article/responsive-web-design/>

Griffiths, P. (2010). Elastic Design. A list apart. Retrieved April 08, 2021, from <https://alistapart.com/article/elastic/>

Brown, T. (2011). Flexible typography with CSS locks. Adobe blog typekit. Retrieved February 08, 2021, from <https://blog.typekit.com/2016/08/17/flexible-typography-with-css-locks/>

Wordpress: Blog Tool, Publishing Platform, and CMS. (2021). Retrieved March 08, 2021, from <https://wordpress.org/>

JOOMLAPORTAL.CZ: Co je Joomla!?. (2012). Retrieved April 08, 2021, from <http://www.joomlaportal.cz>

Drupal.cz: Český portál o open source CMS Drupal. Retrieved March 08, 2021, from www.drupal.cz/

PHP. (2001- 2021). Retrieved March 08, 2021, from <https://www.php.net>

W3C. (2021, June 19). Retrieved April 08, 2021, from <https://www.w3.org/>

JQuery. (2021). Retrieved April 08, 2021, from <https://jquery.com/>

SublimeText. (2020). Retrieved April 08, 2021, from <https://www.sublimetext.com/>

FontAwesome. (2021). Retrieved March 08, 2021, from <https://fontawesome.com/>

Seznam obrázků, zdrojových kódů, diagramů, tabulek a zkratek

Seznam obrázků

OBRÁZEK 1: SCHÉMA LAYOUTU S PEVNOU ŠÍŘKOU	14
OBRÁZEK 2: SCHÉMA FLUIDNÍHO LAYOUTU.....	15
OBRÁZEK 3: UKÁZKA FUNKČNOSTI JEDNOTKY EM	17
OBRÁZEK 4: DOSAZENÍ DO FUNKCE CALC.....	19
OBRÁZEK 5: UKÁZKA POMĚRŮ PIXELŮ ZAŘÍZENÍ.....	22
OBRÁZEK 6: STRUKTURA MEDIA QUERIES.....	25
OBRÁZEK 7: UKÁZKA FUNKCE BREAKPOINT	30
OBRÁZEK 8: TRANSFORMACE OBSAHU PRO DVA RŮZNÉ TYPY ZAŘÍZENÍ	38
OBRÁZEK 9: STRUKTURA WORDPRESS NA SERVERU	44
OBRÁZEK 10: INSTALACE WORDPRESSU.....	45
OBRÁZEK 11: STRUKTURA ŠABLONY PRO WORDPRESS.....	47
OBRÁZEK 12: UKÁZKA MŘÍŽKY S PROCENTUÁLNÍM ROZDĚLENÍM.....	48
OBRÁZEK 13: UKÁZKA DESIGNU A ROZLOŽENÍ FOOTRU	48
OBRÁZEK 14: TESTOVÁNÍ RESPONZIVITY WEBOVÉ STRÁNKY.....	52
OBRÁZEK 15: RESPONZIVNÍ PROBLÉM STRÁNKY ČLENOVÉ KATEDRY	54

Seznam výpisů kódu

VÝPIS 1: PŘÍKLAD POUŽITÍ JEDNOTKY REM NA ODSAZENÍ ODSTAVCE	17
VÝPIS 2: UKÁZKA RESPONSIVE TYPOGRAPHY S VYUŽITÍM MEDIA QUERIES.....	18
VÝPIS 3: ZÁPIS FUNKCE CALC().....	19
VÝPIS 4: UKÁZKA POUŽITÍ SRCSET S DESCRIPTOREM W	23
VÝPIS 5: UKÁZKA POUŽITÍ SRCSET S DESCRIPTOREM W	24
VÝPIS 6: UKÁZKA POUŽITÍ @MEDIA-QUERIES	25
VÝPIS 7: UKÁZKA POUŽITÍ AND OPERÁTORU.....	28
VÝPIS 8: UKÁZKA POUŽITÍ OR OPERÁTORU	28
VÝPIS 9: UKÁZKA POUŽITÍ NOT OPERÁTORU	29
VÝPIS 10: UKÁZKA POUŽITÍ ONLY OPERÁTORU	29
VÝPIS 11: ZÁKLADNÍ KÓD TEMPLATU PRO ZOBRAZENÍ VŠECH STRAN [BRAINSTORM FORCE]	49
VÝPIS 12: KÓD ACCORDIONU S CSS STYLEM A SKRIPTEM [AUTOR]	50
VÝPIS 13: KÓD IFRAMU S GOOGLE MAPS INFORMACEMI [MAPS.IE]	51

Seznam tabulek

TABULKA 1: VLASTNOSTI @MEDIA-QUERIES.....	27
TABULKA 2: TESTOVÁNÍ RESPOZIVNÍHO CHOVÁNÍ V RŮZNÝCH PROHLÍŽEČÍCH	53
TABULKA 3: TESTOVÁNÍ RESPOZIVNÍHO CHOVÁNÍ NA MOBILNÍCH ZAŘÍZENÍCH.....	54
TABULKA 4: TESTOVÁNÍ RESPONZIVNÍHO CHOVÁNÍ VE VIRTUÁLNÍCH ZAŘÍZENÍ	55

Seznam zkratk

URL – uniform resource locator neboli jednotný lokátor zdroje

SEO – search engine optimization jedná se o optimalizaci pro vyhledávače

<body> – Značí tělo stránky, kde v kódu přidáváme jednotlivé komponenty

<p> – Značí odstavec v kódu

<div> – Div je element blokový

 - Span je element řádkový

HTML– Hypertext markup language jedná se o značkovací jazyk používaný pro tvorbu webových stránek

HTTP – Hypertext transfer protocol

DOM – Document object model

CMS – Content management system - systém pro správu obsahu

WCM – Web content management system

ECM – Enterprise content management

DAM – Digital access management

RM - Record management – správa záznamů

CCMS - Component content management system

LMS - Learning Management System

PHP - Hypertextový preprocesor a původně Personal home page

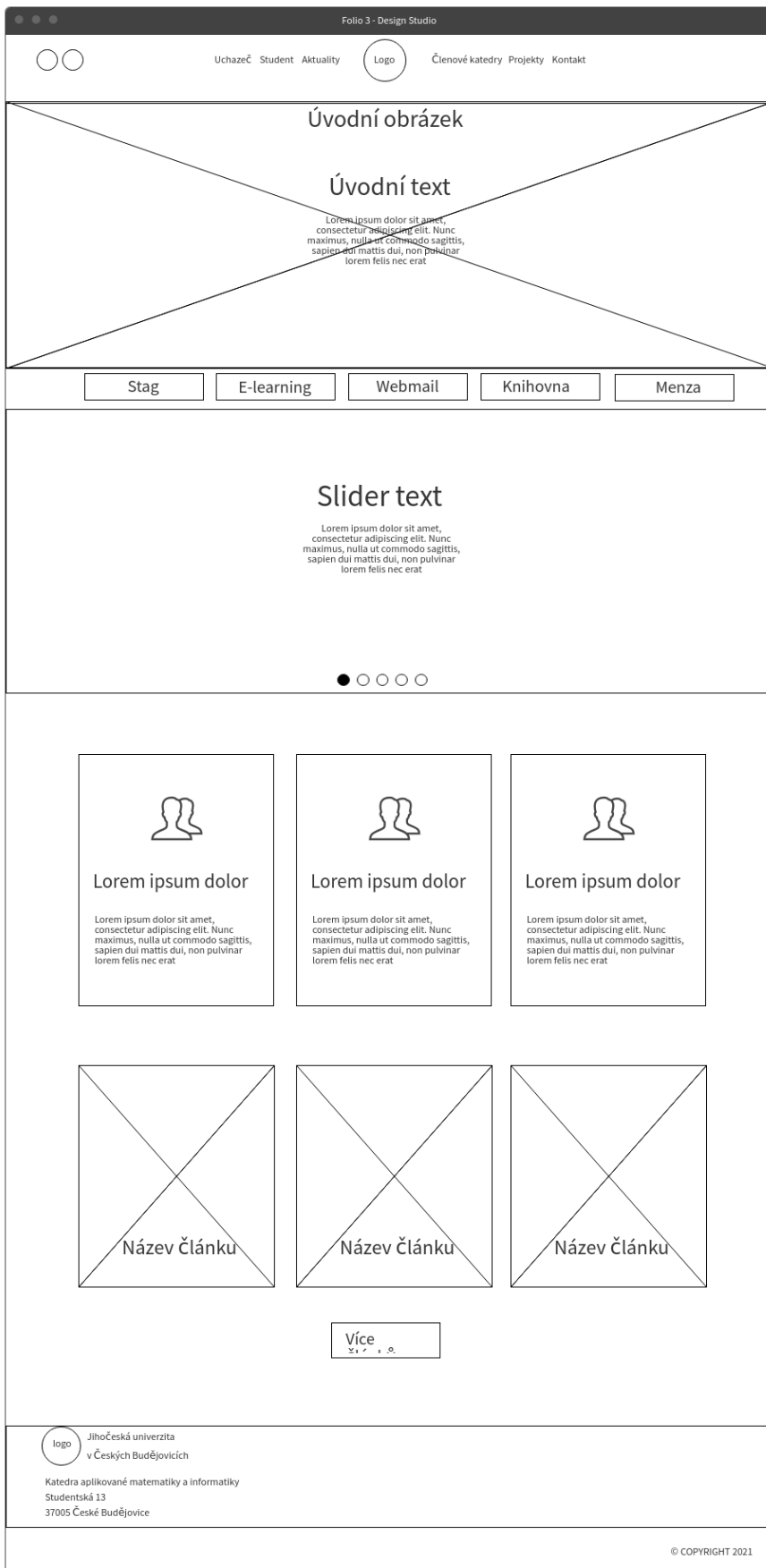
CSS - Cascading Style Sheets neboli kaskádové styly

Přílohy

Příloha A: Wireframy na další straně

Příloha B: CD se všemi zdrojovými kódy webu a plným textem bakalářské práce

Příloha A.1



Příloha A.2

Folio 3 - Design Studio

Uchazeč Student Aktuality Logo Členové katedry Projekty Kontakt

Uchazeč

LOGO

Nadpis

TEXT

TLAČÍTKO

LOGO

Nadpis

TEXT

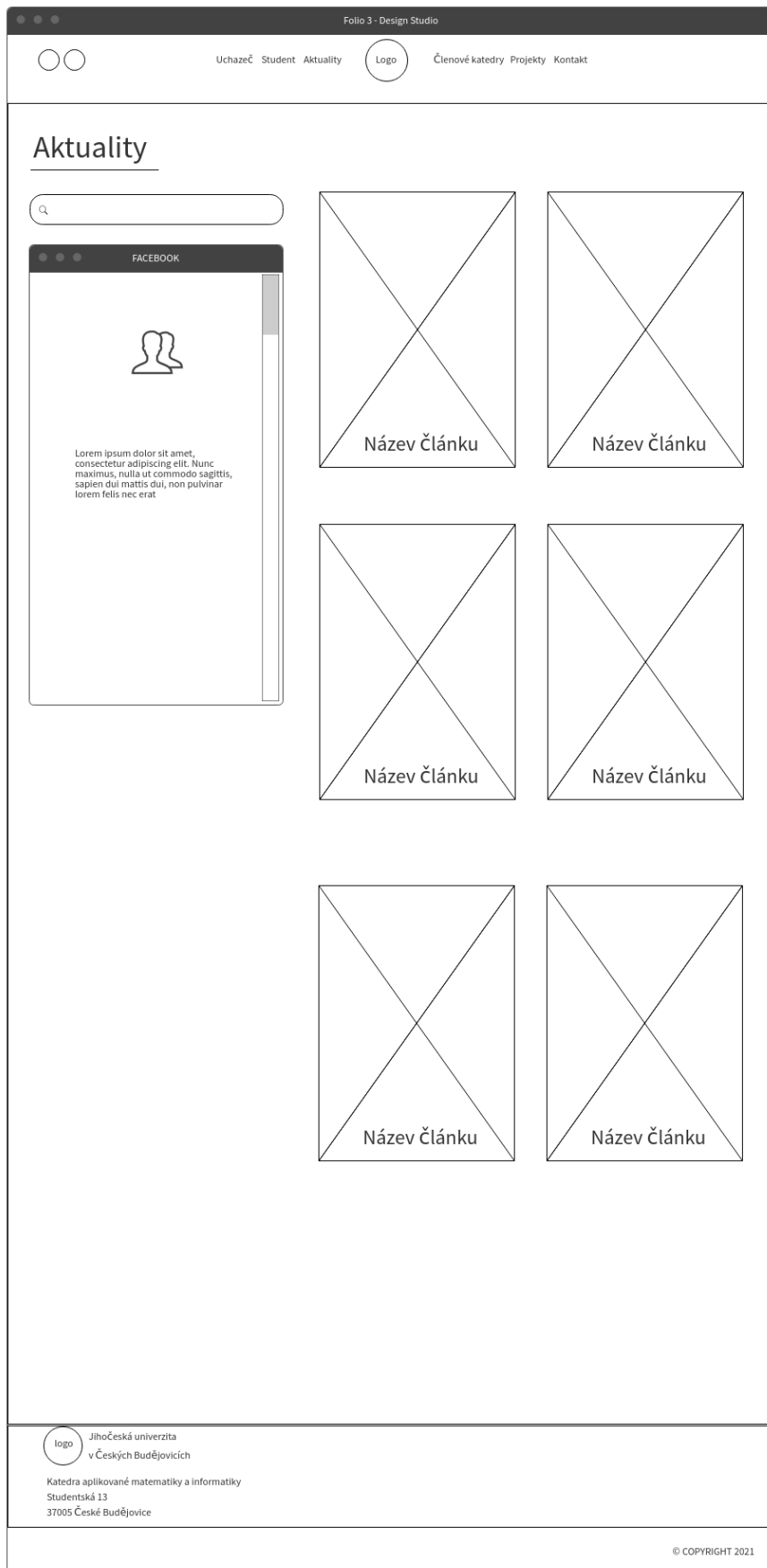
TLAČÍTKO

logo Jihočeská univerzita
v Českých Budějovicích

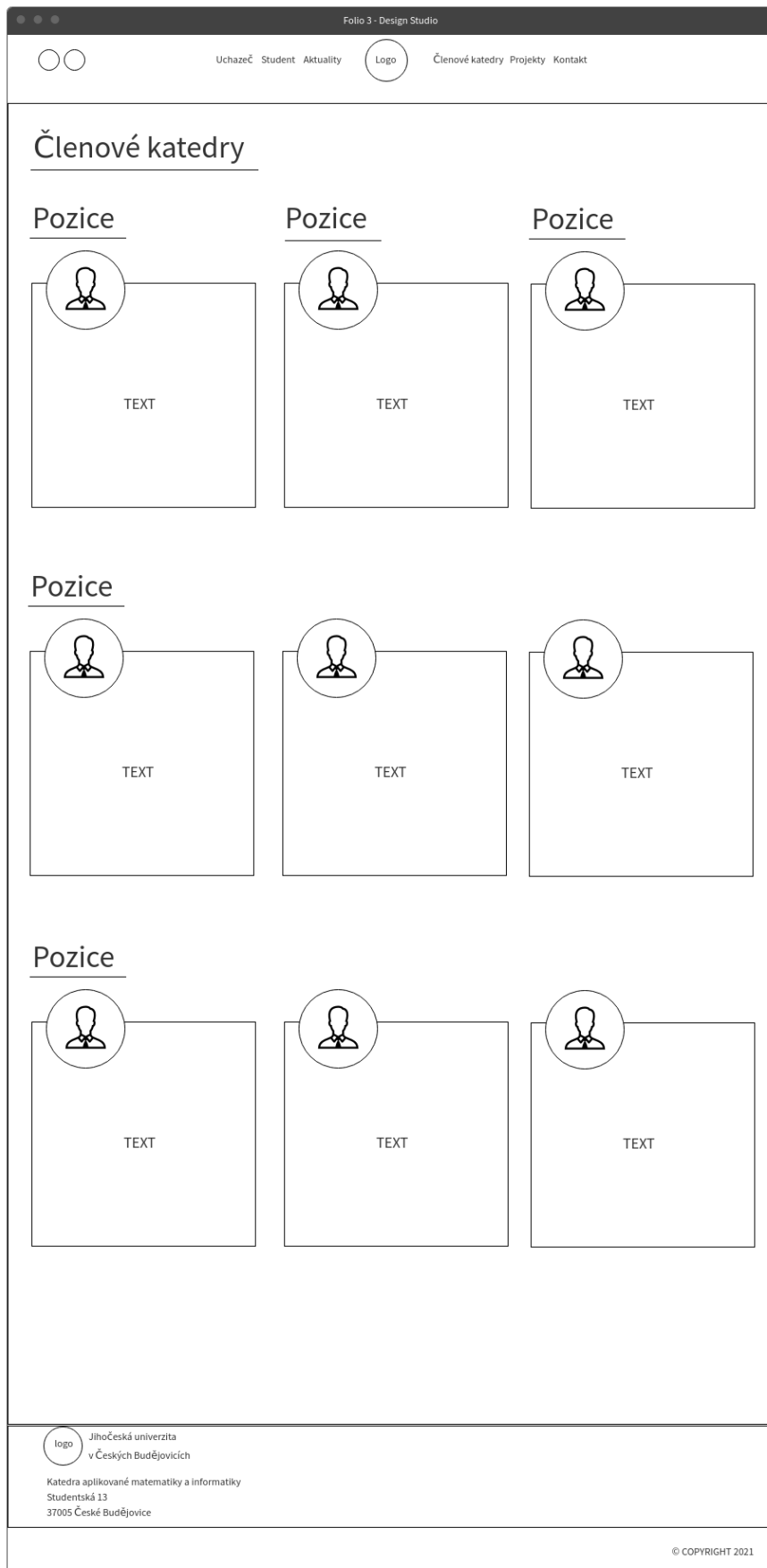
Katedra aplikované matematiky a informatiky
Studentská 13
37005 České Budějovice

© COPYRIGHT 2021

Příloha A.3



Příloha A.4



Příloha A.5

Folio 3 - Design Studio

Uchazeč Student Aktuality Logo Členové katedry Projekty Kontakt

Projekty

Projekt 1
TEXT
Projekt 2
Projekt 3
Projekt 4

Logo Jihočeská univerzita
v Českých Budějovicích

Katedra aplikované matematiky a informatiky
Studentská 13
37005 České Budějovice

© COPYRIGHT 2021

Příloha A.6

