

Pedagogická
fakulta
Faculty
of Education

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

Jihočeská univerzita v Českých Budějovicích

Pedagogická fakulta

Katedra informatiky

**Systémy webového designu, atomický design
webu**

Web design systems, atomic web design

Bakalářská práce

Vypracoval: Martin Horák

Vedoucí práce: PaedDr. Petr Pexa, Ph.D.

České Budějovice 2018

JIHOČESKÁ UNIVERZITA V ČESKÝCH BUDĚJOVICÍCH

Fakulta pedagogická

Akademický rok: 2016/2017

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Martin HORÁK**

Osobní číslo: **P15480**

Studijní program: **B7507 Specializace v pedagogice**

Studijní obory: **Matematika se zaměřením na vzdělávání**

Informační technologie se zaměřením na vzdělávání

Název tématu: **Systémy webového designu, atomický design webu**

Zadávací katedra: **Katedra informatiky**

Z á s a d y p r o v y p r a c o v á n í :

Cílem bakalářské práce je zpracování problematiky systémů webového designu (tzv. style guides), zejména pak tzv. atomického designu webu. Systémy webového designu usnadňují dodržování jednotnosti front-end designu, zmenšují a zpřehledňují kód a usnadňují testování. Atomický design webu dělí celý design webu na jednotlivé složky od atomů až po stránky, jednotlivá dělení usnadňují práci na webu využitím vývojového prostředí Pattern LAB. Teoretická část práce bude zaměřena na komparaci systémů webového designu obecně a představení jednotlivých prvků atomického designu (atomy, molekuly, organizmy, šablony, stránky). V praktické části bude s využitím vývojové prostředí Pattern LAB vytvořen vzorový webový design na základě principů atomického designu webu, provedeno jeho testování a ověření designérské hypotézy v aktuálních verzích webových prohlížečů.

Rozsah grafických prací: **CD ROM**

Rozsah pracovní zprávy: **40**

Forma zpracování bakalářské práce: **tištěná**

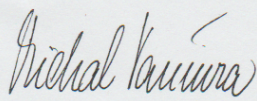
Seznam odborné literatury:

1. **Atomický design a Pattern Lab: návrat do budoucnosti návrhu uživatelských rozhraní. Vzhůru dolů - webový frontend ze všech stran [online]. Dostupné z: <http://www.vzhurudolu.cz/prirucka/pattern-lab>**
2. **Atomic Design by Brad Frost. Atomic Design by Brad Frost [online]. Dostupné z: <http://atomicdesign.bradfrost.com/>**
3. **Website Style Guide Resources. Website Style Guide Resources [online]. Dostupné z: <http://styleguides.io/>**
4. **Pattern Lab — Build Atomic Design Systems. Pattern Lab — Build Atomic Design Systems [online]. Dostupné z: <http://patternlab.io/>**
5. **Making And Maintaining Atomic Design Systems With Pattern Lab 2 - Smashing Magazine. Smashing Magazine - For Professional Web Designers and Developers [online]. Dostupné z: <https://www.smashingmagazine.com/2016/07/building-maintaining-atomic-design-systems-pattern-lab/>**

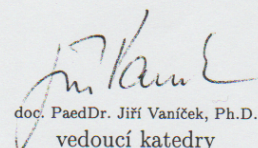
Vedoucí bakalářské práce: **PaedDr. Petr Pexa, Ph.D.**
Katedra informatiky

Datum zadání bakalářské práce: **24. dubna 2017**

Termín odevzdání bakalářské práce: **30. dubna 2018**



Mgr. Michal Vančura, Ph.D.
děkan



doc. PaedDr. Jiří Vaníček, Ph.D.
vedoucí katedry

V Českých Budějovicích dne 24. dubna 2017

Prohlášení

Prohlašuji, že svoji bakalářskou práci 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 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.

V Českých Budějovicích dne 20. dubna 2018.

Martin Horák

Podpis

Abstrakt

Cílem bakalářské práce je zpracování problematiky systémů webového designu (tzv. style guides), zejména pak tzv. atomického designu webu. Systémy webového designu usnadňují dodržování jednotnosti front-end designu, zmenšují a zpřehledňují kód a usnadňují testování. Atomický design webu dělí celý design webu na jednotlivé složky od atomů až po stránky, jednotlivá dělení usnadňují práci na webu využitím vývojového prostředí Pattern LAB. Teoretická část práce bude zaměřena na komparaci systémů webového designu obecně a představení jednotlivých prvků atomického designu (atomy, molekuly, organizmy, šablony, stránky). V praktické části bude s využitím vývojové prostředí Pattern LAB vytvořen vzorový webový design na základě principů atomického designu webu, provedeno jeho testování a ověření designérské hypotézy v aktuálních verzích webových prohlížečů.

Klíčová slova

Web design, systémy webového designu, atomický design

Abstract

The objective of the bachelor thesis will be elaborating of problematics of web design systems (style guides). Web designing systems are simplifying observance of uniformity front-end design, shrinking and facilitating source code and simplifying testing. Atomic web design divides the whole site design into individual folders from atoms to pages, and individual divisions make it easier to work on the web by using the Pattern LAB development environment. The theoretical part will be focused on comparison web designing systems generally and to introduce individual elements of atomic design. In practical part using Pattern LAB will be made web design using atomic design. Then will be tested the hypothesis of design in current versions of web browsers.

Keywords

Web design, web design systems, atomic design

Poděkování

Touto cestou bych rád poděkoval panu PaedDr. Petru Pexovi, Ph.D. za jeho odborná vedení a věcné připomínky k mé práci. Dále bych rád poděkoval mým rodičům, kteří mě podporují v mém studiu. . . .

Obsah

1	Úvod	12
1.1	Východiska práce	12
1.2	Cíle práce	12
1.3	Metody práce	13
2	Systémy webového designu	14
2.1	Výhody systémů webového designu	14
3	Přehled systémů webového designu	16
3.1	BEM	16
3.2	U.S. Web design system	16
3.3	Marrel.css	17
4	Atomický design	18
4.1	Atomy	18
4.2	Molekuly	18
4.3	Organizmy	19
4.4	Šablony	19
4.5	Stránky	20
5	Pattern LAB	21
5.1	Instalace Node.js	21
5.2	Instalace Pattern LAB	22
5.3	Instalace vzorových dat	23
6	Základy práce na návrhu designu	24
6.1	Práce s daty	24
6.1.1	Zapisování dat do souboru	25
6.1.2	Použití dat pro vytvoření součástí	25
6.2	Vytváření souborů pro jednotlivé prvky	26

6.2.1	Soubory .mustache	26
6.2.2	Soubory .md	26
6.3	Generování systému	27
7	Návrh designu webové stránky	29
7.1	Hlavička a patička	29
7.2	Atomy	29
7.2.1	Barvy	30
7.2.2	Pozadí	30
7.2.3	Formulářové prvky	30
7.2.4	Seznamy	30
7.2.5	Horizontální čára	31
7.2.6	Inline elementy	31
7.2.7	Nadpisy	31
7.2.8	Odstavcový text	32
7.2.9	Předformátovaný text	32
7.2.10	Skrytý text	32
7.2.11	Tlačítko	32
7.2.12	Tlačítko menu	33
7.2.13	Tlačítko bočního menu	33
7.3	Molekuly	33
7.3.1	Příklad k tématu	33
7.3.2	Rozcestník tématu	34
7.3.3	široký pruh textu	34
7.3.4	Část vysvětlení teorie	34
7.3.5	Logo link	34
7.3.6	Vyhledávání	35
7.3.7	Pole typu input	35
7.3.8	Textarea	35
7.3.9	Navigace hlavního menu	35

7.3.10	Boční menu	36
7.4	Organizmy	36
7.4.1	Hlavička	36
7.4.2	Patička	37
7.4.3	Formulář	37
7.4.4	Boční panel	38
7.4.5	Text kontakt	38
7.4.6	Text příklady	38
7.4.7	Text teorie	39
7.4.8	Text úvodní stránka	39
7.5	Šablony	40
7.5.1	Úvod	40
7.5.2	Teorie	41
7.5.3	Příklady	42
7.5.4	Kontakt	42
7.6	Stránky	43
7.7	Mobilní verze	43
8	Testování v prohlížečích	44
8.1	Google Chrome	45
8.2	Mozilla Firefox	46
8.3	Internet Explorer	47
8.4	Shrnutí testování	48
9	Závěr	49
	Seznam použité literatury a zdrojů	50
	Seznam obrázků	52
	Seznam příkladů	53

A Příloha	54
B Příloha	55

1 Úvod

Bakalářská práce se zabývá problematikou systémů webového designu, zejména pak atomickému designu a jeho využití při návrhu designu webových stránek. Teoretická část je zaměřena na popis systému webového designu, speciálně pak atomické designu a jeho součástí.

Praktická část zahrnuje popis práce s vývojovým prostředím Pattern LAB. Návrh designu s pomocí tohoto prostředí a následné testování v různých prohlížečích.

1.1 Východiska práce

V současné době je kladen velký důraz na design webu. K vytváření designu se dnes využívá jazyka CSS. Jeho nejnovější verzí je v současné době CSS3. Pro tvorbu designu webu se dnes používají tzv. frameworky. Jedním z nich je např. Bootstrap.

Nový pohled na design webu nabízí systémy webového designu, které se snaží rozložit design na menší části, z kterých se pak skládá celek. Atomický design pak dělí design až do posledních součástí webu a usnadňuje tak tvorbu celého designu.[1]

V České republice existuje v současné době literatura zejména k frameworkům. Literatura k systémům webového designu v současné době v českém jazyce chybí.

1.2 Cíle práce

Cílem bakalářské práce je popsat systémy webového designu, zvláště pak atomického designu a jeho prvků(atomy, molekuly,..). Dále pak seznámení s vývojovým prostředím Pattern LAB.

Teoretická část bude obsahovat informace o systémech webového designu. Zvláštní pozornost bude věnována atomickému designu, který bude popsán

detailně, včetně popisu jeho jednotlivých prvků.

V praktické části provedu instalaci vývojového prostředí Pattern LAB. S jeho pomocí vytvořím za použití atomického designu vzorový design webové stránky. Bude provedeno jeho porovnání v aktuálních verzích prohlížečů.

1.3 Metody práce

V teoretické části popíšu systémy webového designu obecně. Větší prostor pak věnuji samotnému atomickému designu. Podrobně budou popsány všechny jeho součásti.

V praktické části se budu zabývat prací s vývojovým prostředím Pattern LAB. Bude provedena instalace Vytvořím vzorový design s využitím principů atomického designu, který následně porovnáám v aktuálních verzích prohlížečů.

2 Systémy webového designu

S postupným vývojem internetu spatřily světlo světa i kaskádové styly. Spolu s vývojem CSS se začínal vyvíjet i design samotný. Ten je v současnosti velmi důležitou součástí každého webu. I proto zde existuje snaha o co největší zjednodušení práce na designu.

Jistě, že pro tvorbu designu lze v dnešní době využít frameworky jako je například Bootstrap. Do značné míry návrh designu zjednodušují. Nicméně weby, které je používají vypadají do značné míry stejně.[2]

Brad Frost publikoval myšlenku atomického designu, která se dívá na design webu jiným pohledem. Základní myšlenkou je rozdělení webu do co nejmenších součástí - atomů. Celá myšlenka pak míří ještě dál a říká, že si každý může design rozdělit na své vlastní částičky. Tím je pak dán prostor pro vznik různých systémů webového designu.[1]

Tvorba vlastního systému webového designu dává každému designérovi silný nástroj s jehož pomocí si vytvoří systém přesně na míru. Takový systém obsahuje prvky, které jsou pro daný design potřeba. Každý systém webového designu by měl obsahovat jednoznačnou dokumentaci, vzory a příklady. Tato pravidla a postupy umožňují týmům návrhářů věnovat se naplno tvorbě designu a nezabývat se zkoumáním navzájem různých kódů.[2]

2.1 Výhody systémů webového designu

Jak již bylo zmíněno výše systémy webového designu dávají autorovi možnost vytvořit design webu co nejjednodušeji. Nicméně výhod mají systémy celou řadu, například:[2]

- **Jednotnost designu** - nehrozí, že bude každá stránka vypadat jinak, což bývá problém některých webů
- **Jednotné názvy** - na projektu může pracovat kolik lidí chce a všichni znají daná pravidla pro označování prvků. Tím se řeší problém daný tím,

že každý je zvyklý používat své názvy.

- **Rychlost** - vytvoření samotného systému může být sice časově náročnější, nicméně pokud je systém jednou dobře navržený, veškerá další práce na designu se tím značně zrychluje.
- **Testování** - díky rozdělení do jednotlivých malých částek lze každou částku zvlášť testovat. To pomáhá zejména při hledání chyb a jejich odstraňování.

3 Přehled systémů webového designu

V následující části práce budou představeny vybrané systémy webového designu. Nicméně jak je patrné již z předchozí kapitoly, lze systému vytvořit v podstatě nekonečně mnoho. Uvedeny jsou proto pouze některé systémy, ale rozhodně ne všechny. V současné době existuje systémů celá řada.[3]

3.1 BEM

Bem je systém webového designu jehož název vychází ze zkratky (B = Block, E = Element, M = Modifier). Jak je tedy již z názvu patrné jedná se o systém pracující s bloky a elementy a jejich úpravami.[4]

Základní myšlenkou je rozdělení webu do na sobě nezávislých bloků. Podle autorů to usnadňuje práci na designu a napomáhá přehlednosti kódu.[4]

Jako bloky se označuje vše co může být použito opakovaně. V html se poznají tyto elementy, tak že obsahují atributy *class* nebo *id*. Bloky mohou být navzájem různě vnořovány. Element je pak definován jako část bloku, která nelze použít odděleně.[4]

Tento systém je na jednu stranu velmi jednoduchý a dělí web pouze do dvou kategorií. Bohužel však z mého pohledu právě toto dělení je problémem pro větší projekty, kde by takové dělení nemuselo stačit.

3.2 U.S. Web design system

Jedná se o systém designu určený zejména pro weby americké státní správy. Nicméně veškeré informace o něm, stejně tak jako související soubory jsou volně ke stažení na webu tohoto systému.[5]

Tento systém v základu používá výše zmíněný systém BEM pro pojmenování jednotlivých komponent a jejich roztrídění. Lze jej tak použít jako příklad odvození si vlastního systému z jiného již existujícího.[5]

Vzhledem k tomu, že jde o systém primárně pro tvorbu webů americké

státní správy, myslím si, že jinde si moc využití nenajde. Zmínit se o něm však stojí za to minimálně kvůli velmi precizně zpracované dokumentaci. Dále pak jako příklad rozmanité použitelnosti systémů webového designu.

3.3 Marrel.css

Marrel.css je systém webového designu původně určený pro web inktweb.nl. V současné době je však pod licencí open source. Jméno vychází z označení Margin Release, což je tlačítko, jež bývá na psacích strojích a umožňuje psát do okrajů.[6]

Jak je patrné z předchozího odstavce jde o systém, který má za úkol usnadnit práci na designu konkrétního webu. To znamená, že jej rozhodně nelze použít pro každý web. Vychází ze systému sass¹.

¹Syntactically awesome style sheets - Syntakticky úžasné styly, jazyk pro tvorbu stylů

4 Atomický design

Jak napovídá sám název, autor se inspiroval v chemii. Základem je spojování atomů ve větší celky. Díky inspiraci v chemii pak vznikl celý atomický design. Jedná se o hierarchii následujících částic:

1. Atomy
2. Molekuly
3. Organizmy
4. Šablony
5. Stránky

Stejně jako v chemii i zde platí, že z menších částic se skládají ty větší. (Například Molekuly se skládají z atomů atd.)[2]

4.1 Atomy

Atomy jsou nejmenším prvkem webových stránek. Patří sem nejzákladnější elementy HTML. Jedná se například o popisky formulářů, prvky formuláře a další prvky, které již nelze dále rozložit bez toho aby přestaly fungovat[1]

Stejně jako v přírodě má každý atom své unikátní vlastnosti. Jako příklad můžeme uvést třeba velikost písma. Atomy představují všechny základní styly. U designu však atomy nikdy neexistují odděleně, vždy musí být provázány s nějakou aplikací/webem.[2]

4.2 Molekuly

Molekuly jsou jednoduché skupiny elementů, které fungují dohromady jako jednotka. Jako příklad můžeme uvést třeba popisek, textový prvek formuláře a odesílací tlačítko, které dohromady tvoří vyhledávací formulář.[1]

Spojením získávají jednotlivé atomy smysl. Výsledkem pak je jednoduchá opakovaně použitelná součást webu, která může být použita všude tam, kde je potřeba.[2]

Vytváření jednoduchých komponent pomáhá návrhářům snižovat zatížení počítače při načítání. Dále pak usnadňuje testování, podporuje opakované používání a podporuje jednotnost celého rozhraní.[2]

4.3 Organizmy

Organizmy už jsou poměrně komplexní záležitostí. Jsou složené z atomů a molekul nebo jiných organizmů. Tyto organizmy vytváří odlišné části rozhraní. Jako příklad lze použít vložení vyhledávacího formuláře do hlavičky nebo vložení menu a bannerů do bočního panelu. Hlavička či boční panel pak tvoří samostatnou část rozhraní webu. (Organismus)[1]

Organizmy se mohou skládat z podobných nebo odlišných typů molekul. Jako příklad odlišných molekul může posloužit hlavička webu, která se skládá z loga, hlavního menu a vyhledávání. Naopak příkladem použití stejného typu molekul je třeba výpis produktů na e-shopu, kdy každý prvek mřížky obsahuje totéž.[2]

Představují menší a jednodušší komponenty na webu. Lze je použít znovu a znovu. Například mřížka s produkty na eshopu může být použita znovu všude tam kde se vypisují produkty nebo jejich kategorie.[2]

4.4 Šablony

V případě šablon se již nelze bavit o podobnosti s chemií. Šablony jsou objekty na úrovni stránky. Umisťují jednotlivé komponenty do stránky a určují základní strukturu webu. Šablona se může například skládat z organizmů pro hlavičku, jednotlivé obsahy atd.[2]

S pomocí šablon můžeme připravit různé varianty umístění prvků na stránce, tj. různé možné rozložení a následně zkoumat, které je nejvhodnější.[2]

Je nutné na tomto místě zdůraznit, že do šablon se nedává žádný konkrétní obsah. Jeho umístění a vzhled s obsahem je předmětem stránek, viz. další podkapitola.[2]

4.5 Stránky

Stránky jsou v podstatě instance šablon. Na rozdíl od nich však zobrazují rozhraní jak vypadá i s konkrétním obsahem. Jako příklad můžeme použít naši stránku z předchozího textu a doplnit do ní text, obrázky a popřípadě jiná média. Nyní bude vidět jak bude stránka vypadat v reálu.[2]

Stránka je poslední a nejkonkrétnější fáze návrhu designu. Je zde vidět, to co uvidí uživatelé a první možnost vidět fungovat všechny prvky dohromady.[2]

Velmi důležité jsou stránky i pro testování, kromě toho že vidíme web, jak jej uvidí uživatelé, můžeme zde také vidět, zda všechny součásti do sebe zapadají jak mají a zda vše funguje jak má jako celek. Pokud ne, můžeme se kdykoliv vrátit k atomům a molekulám a upravit je tak, aby vypadaly a chovaly se jak mají.[2]

5 Pattern LAB

Pattern LAB je, jak již bylo zmíněno v úvodu, nástroj pro tvorbu atomických designových systémů. Slouží pro generování designového systému.[1]

Je distribuován ve dvou verzích, jedné zakládající se na php a druhé, která je založena na frameworku node.js. V obou případech pak máme dále na výběr ze čtyř verzí. Vždy jednu, která obsahuje pouze prázdné jádro a další, které se liší v použité technologii.[7]

Instalace Pattern LAB na počítač používající windows je možná. Nicméně v obou případech je nutné nejprve nainstalovat potřebné součásti (PHP nebo Node.js). V následujícím textu se budeme věnovat node verzi, protože instalace php do Windows je sice možná, nicméně poměrně pracná. Naproti tomu instalace Node.js je poměrně jednoduchá.[8]

5.1 Instalace Node.js

Instalace Node.js je poměrně jednoduchou záležitostí. Jedinou podmínkou je instalovat jako administrátor. Stačí dodržet následující postup:[8]

1. Stáhneme Node.js z webu nodejs.org. Stahujeme verzi pro ms Windows, v našem případě variantu .msi instalátoru
2. V průvodci instalací si ověříme, že instalujeme všechny součásti.
3. Po dokončení instalace otestujeme, zda je node.js nainstalován pomocí následujícího příkazu zdaného do příkazové řádky

```
1 node -v
```

Příklad 1: Ověření, zda je Node.js nainstalovaný

Pokud se objeví zpráva podobná té na následujícím obázku, je Node.js nainstalován.



```
C:\Users\martin>node -v
v8.9.4
```

Obrázek 1: Ověření, zda je node nainstalován

5.2 Instalace Pattern LAB

Ani samotná instalace Pattern LABu v Node.js není nijak složitá. Nejnutnější je samozřejmě stáhnout si příslušnou verzi Pattern Labu. Ty jsou jak už bylo zmíněno výše pro node čtyři. Liší se tím, jaká je u nich použita utilita pro Node.js. [7]

Ještě před samotnou instalací Pattern Labu je potřeba provést instalaci utility grunt nebo gulp v závislosti na zvolené verzi. Instalace probíhá zadáním jediného příkazu do příkazové řádky.[9]

```
1 npm install -g grunt-cli
```

Příklad 2: Instalace Grunt

```
1 npm install -g gulp
```

Příklad 3: Instalace Gulp

Pokud je Grunt nebo Gulp nainstalovaný, můžeme pokračovat instalací samotného Pattern Labu. Stažené soubory s Pattern Labem umístíme do složky, kde jej chceme mít nainstalovaný. Dále budeme pracovat opět v příkazovém řádku, pomocí příkazu *cd* si zvolíme složku s Pattern Labem a příkazem *npm install* nainstalujeme Pattern Lab.[7]

```
1 cd umisteni_pattern_labu
2 npm install
```

Příklad 4: Instalace Pattern Labu

5.3 Instalace vzorových dat

Pokud uživatel potřebuje zjistit jak Pattern Lab funguje na funkčním příkladu, je zde možnost nahrát si do Pattern Labu tzv. startovací data.[7]

Startovací data obsahují návrh atomického systému pro vzorový web. Lze na nich snadno pochopit strukturu systému, ale i přidávání jednotlivých komponent. Pokud je uživatel chce vyzkoušet z jakéhokoliv důvodu dělá se to způsobem uvedeným v příkladu níže.[7]

```
1 npm install @pattern-lab/starterkit-mustache-demo
2 grunt patternlab:loadstarterkit --kit=@pattern-lab/
   starterkit-mustache-demo
```

Příklad 5: Instalace vzorových dat do Pattern Labu

6 Základy práce na návrhu designu

Veškerá práce s návrhem designu se odehrává ve složce `source`, kde jsou všechny informace a příkazy psány a evidovány v jednotlivých podsložkách a souborech. Pro většinu práce nám postačí se pohybovat právě v této složce.[7]

Tato složka má následující strukturu.[10]

- **annotations** Složka obsahující soubor `annotations.js`, který obsahuje komentáře k různým prvkům designového systému. Ty lze následně zobrazit přímo ve webovém přehledu systému.
- **data** Zde jsou uložena data použitá pro generování náhledů pomocí `Pattern Labu`.
- **meta** Složka obsahující data pro hlavičku a patičku webu. To co je umístěno v `html` v `head` a na úplném konci dokumentu.
- **patterns** Nejdůležitější složka, obsahuje všechny části designu (atomy, molekuly,..).
- **další složky - css, images, js** Složky pro obrázky, `css` styly a `javascripty`, mohou se jmenovat i jinak dle volby uživatele.

V těchto složkách probíhá vytvoření podkladů pro design. Práce se soubory ve složce `patterns` je poměrně jednoduchá.

6.1 Práce s daty

Práce s daty je u `Pattern Labu` propracovaná záležitost. Data se u návrhu berou ze společného souboru `data.json`. U tvorby konkrétních stránek pak ze souboru opět ve formátu `.json` se stejným názvem jako má stránka ve stejné složce.[10]

6.1.1 Zapisování dat do souboru

Veškerá data se zapisují ve stejném formátu. V souboru se zapisují nejdříve názvem v uvozovkách, dvojtečkou a následně obsahem v uvozovkách.[7]

```
1 "text": "text odstave",
```

Příklad 6: Vložení dat do souboru s daty

Dalším typem dat jsou data z pole. Do pole můžeme vložit data pro více částí stejného typu. Vytvoření je následovné:[7]

```
1 "teorielist": {
2   ["nadpis": "Nadpis 1",
3   "text": "<p>Text 1</p>"],
4   ["nadpis": "Nadpis 2",
5   "text": "<p>Text 2</p>"]
6 },
```

Příklad 7: Vložení pole dat do souboru s daty

6.1.2 Použití dat pro vytvoření součástí

Data se samozřejmě používají pro naplnění jednotlivých součástí obsahem. Pro tento účel se používají tři typy odkazů na data.[7]

Prvním typem jsou data, které se po vložení vypíší jako prostý text. Veškeré Html značky v tomto textu se převedou na entity. [7]

```
1 <h2> {{ nadpis }} </h2>
```

Příklad 8: Vložení dat s převodem na entity

Druhým typem je vložení dat i s Html značkami. V tomto případě se Html značky na nic nepřevodou a budou normálně fungovat.[7]

```
1 {{{ nadpis }}}
```

Příklad 9: Vložení dat se zachováním Html značek

Třetí a zároveň poslední případ jsou data z pole. S jejich pomocí se konkrétní prvek vytvoří vícekrát, pokaždé s jinými daty.[7]

```

1 {{# teorielist }}
2   {{> molekuly-text-teorie }}
3 {{/ teorielist }}
```

Příklad 10: Vložení dat z pole

6.2 Vytváření souborů pro jednotlivé prvky

Jednotlivé prvky se vytvářejí pomocí souborů typů *.mustache* a *.md*. Z těchto souborů se generuje celý design a systém designu.[10]

6.2.1 Soubory *.mustache*

Tyto soubory tvoří základ a je nutné je vytvořit pro každou část. Každý takovýto soubor obsahuje zdrojový kód dané součásti a odkazy na potřebná data. Každý takovýto soubor umístěný v jedné z podsložek složky *patterns* se vygeneruje jako jedna položky systému designu.[7]

6.2.2 Soubory *.md*

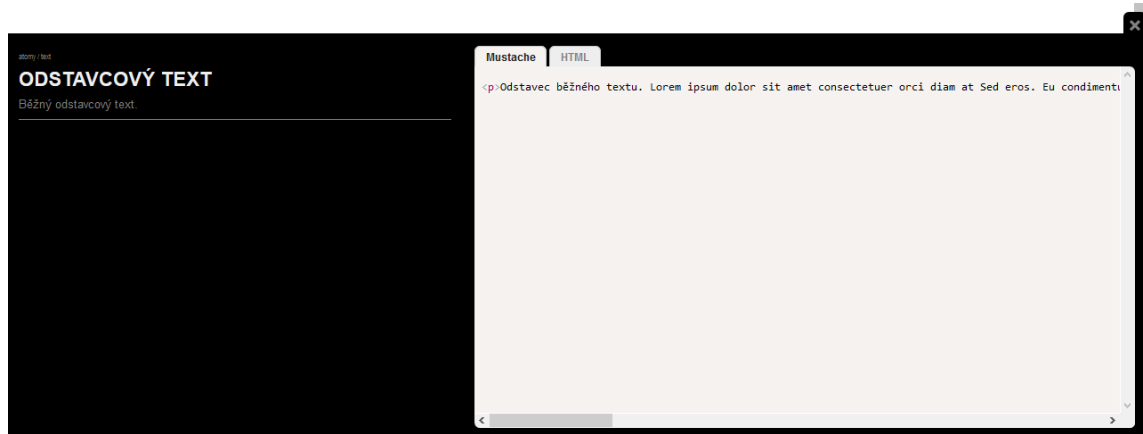
Druhým souborem je soubor ve formátu *.md*. Tento soubor obsahuje název prvku a jeho popis. Tento popis si lze v systému webového designu zobrazit po kliknutí na ikonku matičky a zvolení položky **Show pattern info**. Pomocí tohoto souboru lze prvkům dávat české názvy. Soubor musí mít následující formát.[10]

```

1 _____
2 title: Titulek prvku
3 _____
4 Popis prvku
```

Příklad 11: Obsah souboru *.md*

Po zobrazení popisu pomocí postupu výše se uživateli zobrazí následující dialog s popisem prvku.[10]



Obrázek 2: Dialog s popisem prvku

Pokud není soubor `.md` vytvořen, dojde i tak k vygenerování příslušného prvku. Název se bude zobrazovat stejný jako má soubor `.mustache`. A v případě zobrazení detailu prvku bude dialogové okno vypadat takto.



Obrázek 3: Dialog při neexistenci popisku prvku

6.3 Generování systému

Dalším krokem je vygenerování do verze publikovatelné na webu. Generování probíhá opět pomocí příkazu v příkazové řádce. Je nutné mít opět nastaven

adresář s Pattern Labem jako pracovní. Vygenerovaná verze se objeví ve složce *public*.^[7]

```
1 grunt patternlab:build
```

Příklad 12: Generování systému webového designu

Příklad je uveden pro verzi založenou na Grunt, pokud používáte verzi s Gulp, stačí pouze vyměnit první slovo, jinak je celý postup identický.^[7]

7 Návrh designu webové stránky

Dostáváme se k nejdůležitější části celé práce. Na následujících stránkách si předvedeme tvorbu designu pomocí tohoto nástroje. Jako vzorový budeme tvořit design webu zaměřeného na přípravu k přijímacím zkouškám z matematiky.

7.1 Hlavička a patička

Jak je popsáno v předchozí kapitole, hlavička a patička se upravují ve složce meta. Budeme pracovat se souborem *00-head.mustache*. V první řadě změníme název v hlavičce v tagu `<title>`. Dalším údajem, který mírně změníme jsou odkazy na css styly. Přidáme druhý styl a upravíme volbu stylu pro responzivitu, jako v příkladu.

```
1 <link rel="stylesheet" type="text/css" media="screen and (
  min-width: 1024px)" href="../../css/style.css?{{
  cacheBuster }}" />
2 <link rel="stylesheet" type="text/css" media="screen and (
  max-width: 1023px)" href="../../css/mobile.css?{{
  cacheBuster }}" />
```

Příklad 13: Odkazy na css styly

Další úpravy již nejsou třeba. Do souboru pro patičku webu nebudeme zasahovat vůbec.

7.2 Atomy

V souladu s tím, co bylo napsáno v kapitolách výše začíná návrh designu návrhem atomů.

7.2.1 Barvy

Prvním atomem jsou barvy, které jsou použity pro tvorbu designu. Začneme jimi jako základem. Vytvoření souboru pro barvu je zvláštní v tom, že barvy jako takové se pak nastavují v css a nepoužívají se z atomu. Samotný atom se tak tvoří hlavně pro kompletnost dokumentace designu.

Pro vytvoření použijeme již existující css styl Pattern Labu s jehož pomocí zobrazíme barvy do speciálních boxů.

7.2.2 Pozadí

Dalším atomem je pozadí stránek. Pro pozadí je použit přechod z modré do bílé vytvořený pomocí css3.

Pozadí zobrazíme bez použití speciálních stylů. Použijeme pouze patřičný html prvek s třídou s jehož pomocí se bude následně pozadí zobrazovat na webu.

7.2.3 Formulářové prvky

Jako další vytvoříme kategorii pro formulářové prvky. Jejich vytvoření ve formě atomů je stejné pro všechny typy, proto je i zde uvedu všechny naráz.

Všem formulářovým prvkům je pomocí css nastaven stejný rozměr a rámeček modré barvy. Jiné další úpravy nebyly použity.

Vytvoření všech formulářových prvků je taktéž velmi jednoduchou záležitostí. Jediné co je třeba přidat do jednotlivých souborů *.mustache*, je příslušná html značka.

7.2.4 Seznamy

Dalším atomem, který bude součástí designu, jsou seznamy. Stejně jako v předchozím případě se budeme bavit o všech případech naráz. Zde se jedná o uspořádaný a neuspořádaný seznam.

V css je v tomto případě nastaven pouze vzhled odrážek. Do souboru se seznamem je vložen seznam se vzorovými daty, tím je veškerá práce hotova.

7.2.5 Horizontální čára

Zřejmě nejjednodušší část designového systému, kterou je nutné přidat hlavně kvůli úplnosti celého systému. V souboru je pouze uvedena značka `<hr />`.

7.2.6 Inline elementy

Další částí jsou inline elementy. Pro náš design byly použity následující:

- Tučný text
- Kurzíva
- horní index
- dolní index
- Hypertextový odkaz
- Pojem s definicí

Tyto atomy jsou opět vytvořeny vložením příslušných html značek. U většiny není nic upravováno a stačí pouze značky `<sup>`, `<sub>`, `<i>` nebo ``. U odkazu a pojmu je pak v css nastavena barva a styl podtržení.

7.2.7 Nadpisy

Nadpisy jsou v systému uvedeny pro všechny úrovně. Jejich zobrazení je provedeno pouhým umístěním vzorového textu do příslušných značek. V css stylu je jim pouze přiřazen stín.

7.2.8 Odstavcový text

V tomto případě se bavíme o velmi jednoduchém prvku. Hlavním cílem vypsání odstavce textu je ukázat, jak bude vypadat zvolené písmo. Do souboru pouze vložíme do značek odstavce vzorový text.

7.2.9 Předformátovaný text

Jedná se o podobný případ jako je odstavcový text. Vložíme do značek `<pre>` `</pre>` vzorový text. V tomto případě je ještě nutné nastavit v css stylu neproporcionální písmo.

7.2.10 Skrytý text

Skrytý text je složitější prvek, který zároveň využívá i javascriptu. Tento prvek tvoří nápis zobrazit/skrýt a skrytý odstavec textu.

Pro tento prvek vložíme do zdrojového souboru nápis do prvku s patřičnou třídou, dále pak vložíme odstavec s třídou, která je primárně skrytá. V css souboru je nastavena barva nápisu `zobraz/skryj` a viditelnost skrytého odstavce. Pomocí jednoduchého javascriptu je prvek po kliknutí na nápis měněna hodnota css vlastnosti *display*.

7.2.11 Tlačítko

Dalším prvek designu je tlačítko. Tím je myšleno odesílací tlačítko pro formulář. Dalším tlačítkům se budou věnovat následující podkapitoly.

Tlačítko je ve zdrojovém souboru tvořeno prvkem *input* typu *submit*. Je mu přiřazena třída *tlacitko*. Pomocí css stylu jsou pak nastaveny všechny vlastnosti, jako je barva, písmo nebo okraje. Pomocí odkazu na *data* je mu pak přiřazena hodnota *value*.

7.2.12 Tlačítko menu

Dalším tlačítkem je tlačítko v menu. Tento prvek je tvořen jako položka seznamu.

Do zdrojového souboru je vložen vzorový seznam s obecnou položkou. V css stylu je každé položce nastavena barva odkazu a hover efekt². Všechny položky mají také nastaveno nezobrazování odrážek. Jako další je celému seznamu nastaveno zobrazení v řádce.

7.2.13 Tlačítko bočního menu

I toto tlačítko je tvořeno seznamem. Ve zdrojovém souboru je opět vytvořen vzorový seznam. V css stylech je opět nastavena barva a hover efekt. Stejně jako u hlavního menu jsou zrušeny odrážky. Celý seznam je zobrazen pod sebou.

7.3 Molekuly

Nyní postoupíme v hierarchii výše a začneme se věnovat molekulám. Ty obsahují již konkrétní části webu. V následujících řádcích budou opět stručně popsány jednotlivé části.

7.3.1 Příklad k tématu

Příklad k tématu je součástí stránky příkladů. Jedná se o jeden konkrétní příklad, který bude zvlášť orámovaný.

K vytvoření použijeme příkaz `<article>`. Pomocí css mu nastavíme rámeček. Samotný pak bude obsahovat nadpis, text úlohy a řešení. Prom zobrazení řešení použijeme atom skrytý text. Veškerá data se budou načítat ze souboru s daty.

²Efekt, který se projeví po najetí myši na odkaz

7.3.2 Rozcestník tématu

Rozcestník tématu bude použit na stránkách s teorií. Bude obsahovat odkazy na jednotlivé části textu. Narozdíl od předchozí molekuly nebude mít rámeček.

Postup je podobný jako v předchozím případě. Použijeme opět značku `<article>`. Tato část se skládá ze seznamu částí textu a nadpisu tématu. Seznam bude použit nečíslovaný. Veškerá data se opět načtou ze souboru s daty.

7.3.3 široký pruh textu

Tato molekula jak už její název napovídá zobrazuje text v co nejširší části stránky. Použitá bude na vytváření jednotlivých boxů na úvodní stránce.

Ve zdrojovém souboru vše zapíšeme do značky `<article>`. Do tohoto prvku umístíme nadpis a text. Oba tyto údaje se budou načítat ze souboru s daty. V CSS stylů nastavíme rámeček celé molekuly a její okraje.

7.3.4 Část vysvětlení teorie

Tato složitě pojmenovaná část bude obsahovat jednu část z teorie k danému tématu. Důležité je přiřazení id každému nadpisu, tak aby bylo možné používat odkazy na návěští. Stejně jako rozcestník nebude mít rámeček.

Postup vytvoření je shodný jako u předchozích variant textu. Stejně tak i propojení s daty.

7.3.5 Logo link

Tato molekula obsahuje pouze logo a nadpis 1. úrovně s názvem stránky, které je zároveň i odkazem na titulní stránku. V souboru je tak odkaz v němž je pomocí značky `` vložen obrázek a pomocí `<h1>` název stránky. V verzi pro pc se zobrazí obrázek, v mobilní pak nadpis 1. úrovně.

7.3.6 Vyhledávání

Další částí je vyhledávání. Tato část je tvořena z atomů tlačítka a formulářového pole.

Vytvoření je snadná záležitost. Do zdrojového souboru je vloženo tlačítko a textový prvek. Formulář má přiřazeno vlastní id, které slouží k naformátování celého formuláře, zejména jeho umístění.

7.3.7 Pole typu input

Pole typu input jsou všechny formulářové prvky, které lze vyvolat pomocí značky `<input>`. V tomto případě se jedná již o prvky načítané pomocí dat a základ formulářů ve vyšších úrovních.

Zdrojový soubor je tvořen načítáním dat pro prvky, kterých může být neomezeně mnoho. V základu se pro zobrazení načítá jediný typu *text*. Css stylování není třeba.

7.3.8 Textarea

Jde o prvek, který v principu funguje stejně jako prvky typu input. Pomocí dat se načítají jednotlivé textarey, kterých může být taktéž neomezeně mnoho. Ani v tomto případě nebylo nutné upravovat css styly.

7.3.9 Navigace hlavního menu

Hlavní menu je tvořeno jednotlivými atomy tlačítka menu. Výsledkem by mělo být menu, jehož prvky jsou zobrazeny vedle sebe.

Ve zdrojovém souboru je umístěn seznam, do něhož se ze zdrojového souboru načítají data pro položky menu. Vzhled tlačítek je v CSS nastaven již pro atomy, proto jej není nutné dále upravovat. Naopak umístění prvků vedle se sebe se nyní nastaví pro celý seznam.

7.3.10 Boční menu

V případě bočního menu je situace stejná jako u hlavního. Podstatný rozdíl je v zobrazení položek podle sebou.

Stejně jako v předchozím případě je menu tvořeno seznamem. Načtené si opět položky ze souboru s daty. Dále je práce jednoduchá. Nastavíme zobrazení a mezery a je hotovo.

7.4 Organizmy

Dostáváme se k organizmům, což je poslední fáze před tvorbou šablon a tedy i náhledu vzhledu celého webu. Hlavní část práce bude nyní založena na spojování prvků k sobě. Výsledkem budou prvky, které se v dalších krocích už pouze spojí.

7.4.1 Hlavička

První a zároveň nejdůležitější organizmus je hlavička. Bude se zobrazovat na všech stránkách a bude mít všude stejný obsah.

Vytvoření je poměrně jednoduchá záležitost. Do zdrojového souboru umístíme všechno do značek `<header>` `</header>`. Umístění prvků bude ve skutečnosti spočívat v pouhém odkázání na příslušné molekuly.

```
1 <header >
2   {{> molekuly -logo }}
3   {{> molekuly -hledani }}
4   {{> molekuly -menu }}
5 </header >
```

Příklad 14: Vytvoření hlavičky

7.4.2 Patička

Další společnou částí designu je patička. Také ta bude na všech stránkách stejný obsah.

Vše, co bude zapsané do zdrojového souboru, umístíme do značky `<footer>`. Vložíme sem copyright, rok vytvoření a jméno autora, které bude sloužit jako odkaz na email. Vše se bude načítat ze souboru s daty.

```
1 <footer >
2 (c) {{ rok}} <a href='mailto:{{ email}}'>{{ autor}}</a>
3 </footer >
```

Příklad 15: Vytvoření patičky

7.4.3 Formulář

Formulář je poměrně zvláštní případ. Bude totiž sloužit jako součást dalších organizmů. Zde se pouze spojí jednotlivé části, tak aby vznikl funkční formulář.

Vytvoření je opět poměrně rychlé. Spočívá v načtení molekul s prvky a atomu tlačítka. Celý formulář se pak vytváří přes definici v souboru data.json.

```
1 <form action="{{ action }}" method="post">
2 {{# pole}}
3   {{> molekuly-pole}}
4 {{/ pole}}
5 {{# textarey }}
6   {{> molekuly-textarea }}
7 {{/ textarey }}
8 {{> atomy-tlacitko}}
9 </form>
```

Příklad 16: Vytvoření formuláře

7.4.4 Boční panel

Boční panel bude sloužit k zobrazování bočního menu u stránek s teorií a příklady.

Celý boční panel se bude nacházet ve značce `<aside>`. Do ní načteme molekulu s bočním menu. Vzhledem k tomu, že nic jiného nemá boční panel obsahovat, je tím práce hotova.

```
1 <aside class="menu1">
2   <h2>Menu</h2>
3   {{> molekuly-menu-bocni }}
4 </aside>
```

Příklad 17: Vytvoření bočního panelu

7.4.5 Text kontakt

Tento organizmus obsahuje kontaktní formulář a údaje o autorovi webu. Celý je orámován rámečkem se stínem.

Celý tento organizmus se nachází ve značce `<article>`. Do souboru jsou nahrána data ze souboru s daty a následně je připojen organizmus pro formulář.

```
1 <article class="kontakt">
2   <h2>{{ nadpis }}</h2>
3   {{{ text }}}
4   {{> organizmy-formular}}
5 </article>
```

Příklad 18: Vytvoření textu stránky s kontakty

7.4.6 Text příklady

Text příklady bude blok zobrazující příklady k danému tématu. Použity budou molekuly pro jednotlivé příklady.

Vytvoření bude stejně jako u předchozích organizmů velmi snadné. Do značky *section* pouze nahrajeme molekuly pro jednotlivé příklady.

```
1 <section class="obsah2">
2   <div class="priklady">
3     {{# priklist }}
4     {{> molekuly-text-priklady }}
5     {{/ priklist }}
6   </div>
7 </section>
```

Příklad 19: Vytvoření textu stránky s příklady

7.4.7 Text teorie

Text teorie bude část webu složená z rozcestníku a částí teorie. Celá část bude mít rámeček se stíny.

Vytvoření , které je tak jako u předchozích jednoduché, bude spočívat ve vložení všeho do značky *<section>*. Vložíme sem Úvodní část s rozcestníkem a následně části vysvětlení. Vše bude opět pracovat s daty ze souboru *data.json*

```
1 <section class="obsah">
2   {{> molekuly-text-rozcestnik }}
3   {{# teorielist }}
4   {{> molekuly-text-teorie }}
5   {{/ teorielist }}
6 </section>
```

Příklad 20: Vytvoření textu stránky s teorií

7.4.8 Text úvodní stránka

Posledním organizmem je centrální text pro úvodní stránku. Je to velmi jednoduchá část. Celé vytvoření spočívá pouze v načtení pruhu, popřípadě pruhů textu a dat do nich.

```

1 <section class="obsah4">
2   {{# široky}}
3   {{> molekuly-text-široky}}
4   {{/ široky}}
5 </section>

```

Příklad 21: Vytvoření centrálního textu pro úvodní stránku

7.5 Šablony

Další fází návrhu jsou šablony. Zde se jedná již o celé stránky, které však neobsahují konkrétní data. Vytvářeny budou skládáním z jednotlivých částí nižších kategorií k sobě.

Jak je patrné z předchozího odstavce, šablony obsahují prvky, které jsou již popsány výše. Z tohoto důvodu bude nyní pouze popsáno, které prvky budou vloženy a uveden zdrojový kód každé šablony.

7.5.1 Úvod

První a nejdůležitější stránkou každého webu je Úvodní. Proto i zde bude zmíněna jako první. Tvořena je hlavičkou, pruhy textu a patičkou. Pruhy textu budou zobrazeny tak, že se nejprve zobrazí jeden popřípadě více širokých pruhů s úvodními informacemi a pod nimi pak dva sloupce s posledními přidanými tématy v teorii a příkladech. Všechny prvky s výjimkou hlavičky a patičky jsou umístěny ve značce `<main>`.

```

1 {{> organizmy-hlavicka }}
2 <main class="uvod">
3   {{> organizmy-text-uvod-centralni}}
4   <section class="obsah3">
5     <div class="sloupec">
6       {{# teorieuvod}}

```



```
7  {{> molekuly-text-siroky}}
8  {{/ teorieuvod}}
9  </div>
10 <div class="sloupec2">
11  {{# prikluvod}}
12  {{> molekuly-text-siroky}}
13  {{/ prikluvod}}
14 </div>
15 </section>
16 </main>
17 {{> organizmy-paticka}}
```

Příklad 22: Vytvoření šablony úvodní stránky

7.5.2 Teorie

Další šablonou bude šablona pro stránky s teorií. Její vytvoření bude ještě snažší než v případě úvodní stránky. Tvořit ji budou nahoře hlavička a dole patička jako u úvodní stránky. V části s obsahem pak bude vloženo boční menu a blok s textem pro teorii.

```
1  {{> organizmy-hlavicka }}
2  <main>
3    {{> organizmy-bocni-panel }}
4    {{# teorie }}
5    {{> organizmy-text-teorie }}
6    {{/ teorie }}
7  </main>
8  {{> organizmy-paticka}}
```

Příklad 23: Vytvoření šablony pro stránky s teorií

7.5.3 Příklady

Stránky s příklad budou tvořeny stejným způsobem jako stránky s teorií. Budou obsahovat také boční menu. Také v tomto případě bude použito načtení jednotlivých prvků nižších kategorií. Všechny příklady budou načteny do bloku s vlastní třídou kvůli umístění vedle menu.

```
1 {{> organizmy-hlavicka }}
2 <main>
3   {{> organizmy-bocni-panel }}
4   {{# priklady }}
5   {{> organizmy-text-priklady }}
6   {{/ priklady }}
7 </main>
8 {{> organizmy-paticka}}
```

Příklad 24: Vytvoření šablony pro stránky s příklady

7.5.4 Kontakt

Kontakt je poslední typ šablony, kterou budeme pro návrh designu potřebovat. Bude obsahovat pouze jediný blok s textem. Všechno včetně formuláře je již připraveno v organizmech, a proto bude samotný zdrojový soubor nejkratší.

```
1 {{> organizmy-hlavicka }}
2 <main>
3   {{> organizmy-text-kontakt }}
4 </main>
5 {{> organizmy-paticka}}
```

Příklad 25: Vytvoření šablony pro stránku s kontaktem

7.6 Stránky

Stránky jsou poslední fází návrhu designu a v podstatě už se nejedná o návrh designu. V této fázi je pomocí souborů s daty šablonám přiřazován konkrétní obsah. Na samotném designu se již nic nemění. Tato fáze usnadňuje práci zejména v případě, kdy vytváříme větší množství stránek se stejným vzhledem.

Samotný soubor *.mustache* obsahuje pouze odkaz na patřičnou šablonu, všechno se pak odehrává v souboru s daty pro stránku.

Vzhledem k tomu, že postup je pořád stejný, uvedu níže pouze jeden příklad propojení s šablonou. Všechny ostatní by se pouze opakovaly.

```
1 {{> sablony - uvod }}
```

Příklad 26: Vytvoření stránky ze šablony - obsah souboru se stránkou

7.7 Mobilní verze

Poslední co zbývá je popsat mobilní verzi designu. Ta je z pohledu designu jednodušší. Tvoří ji nadpis a patička s modrým pozadím a obsah s bílým pozadím a modrými okraji. Celý soubor css pro mobilní verzi obsahuje podstatně méně údajů.

8 Testování v prohlížečích

Poslední částí práce je provedení testování designu v prohlížečích. Toto testování bude provedeno s ohledem na správné zobrazení designu. Hlavním cílem je aby se web zobrazil stejně bez ohledu na prohlížeč, který uživatel používá. Hlavní kritéria budou jak vyplývá z tohoto textu následující:

- Správné zobrazení webu v běžné verzi
- Správné zobrazení mobilní verze webu

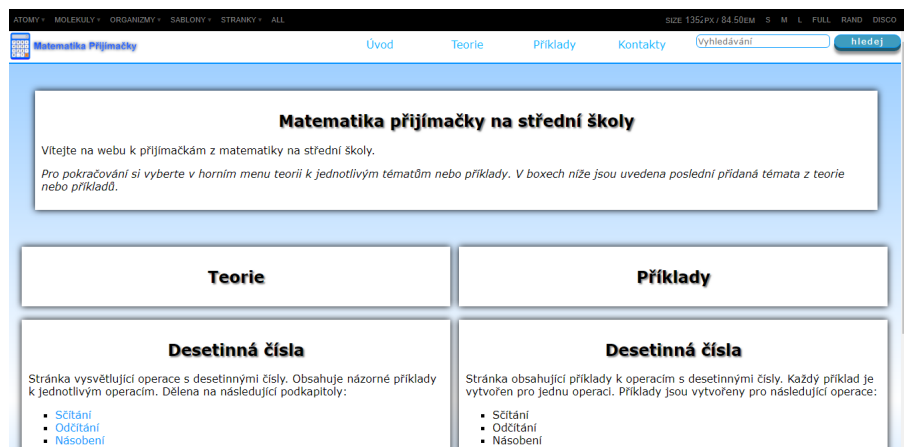
Tato dvě kritéria ukáží, že byl návrh proveden dostatečně pečlivě a nehrozí, že by se mohl najít uživatel, který web uvidí špatně nebo v nejhorším případě vůbec.

Testování je provedeno v pokud možno co nejnovějších verzích prohlížečů. Pro testování sem zvolil tři nejpoužívanější prohlížeče.[11] Jedná se o tyto tři:

- **Google Chrome** - použita byla nejnovější verze 65.0.3325.181
- **Internet Explorer** - použita verze 11.0.9600.18978
- **Mozilla Firefox** - použita nejnovější verze 59.0.2

8.1 Google Chrome

Běžná verze



Obrázek 4: Zobrazení navrženého designu pro pc v Google Chrome

Mobilní verze

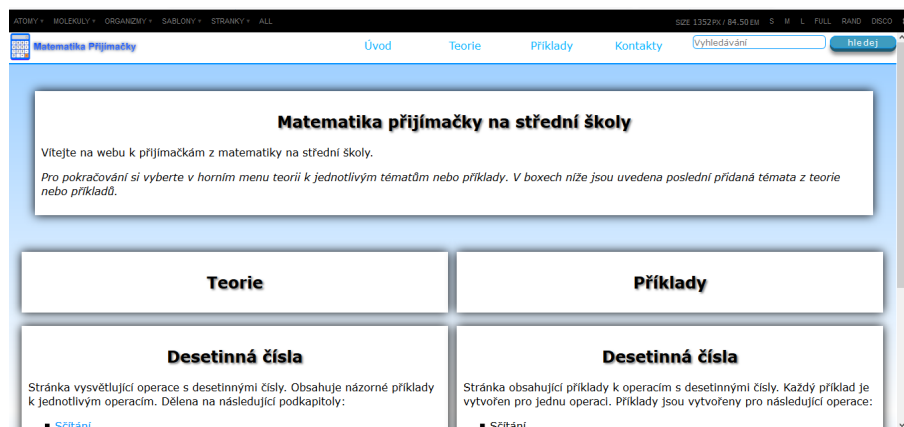


Obrázek 5: Zobrazení navrženého designu pro mobily v Google Chrome

Obě verze se zobrazují správně bez jakýchkoliv viditelných chyb. V Google Chrome je vše v pořádku.

8.2 Mozilla Firefox

Běžná verze



Obrázek 6: Zobrazení navrženého designu pro pc v Mozille Firefox

Mobilní verze

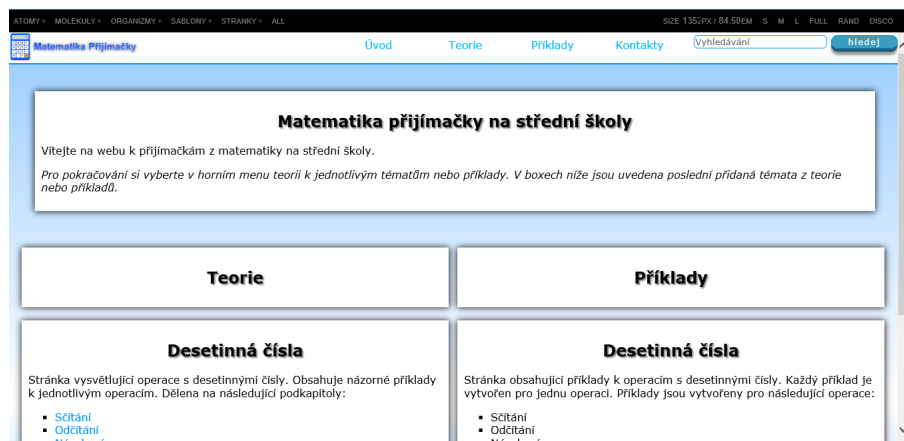


Obrázek 7: Zobrazení navrženého designu pro mobily v Mozille Firefox

Také v Mozille se obě verze zobrazují správně. Neobjevuje se zde žádná viditelná chyba.

8.3 Internet Explorer

Běžná verze



Obrázek 8: Zobrazení navrženého designu pro pc v Internet Exploreru

Mobilní verze



Obrázek 9: Zobrazení navrženého designu pro mobily v Internet Exploreru

I v Internet Exploreru se vše zobrazuje jak má. Ani zde nejsou viditelné žádné chyby.

8.4 Shrnutí testování

Testování bylo provedeno ve třech prohlížečích. Ověřeno bylo jak zobrazení standardní, tak i mobilní verze. Ve všech prohlížečích se navržený design zobrazoval stejně a bez chyb. Lze tedy konstatovat, že návrh je použitelný.

9 Závěr

V této práci jsem se pokusil co nejjednodušeji přiblížit systémy webového designu, zejména pak atomický design. Dále jsem popsal práci s prostředím Pattern Lab.

V praktické části byl vytvořen design pro web s pomocí Pattern Labu. Za použití postupů a principů platných pro atomický design. Celý design se zobrazil ve všech třech prohlížečích stejně.

Použití Pattern Labu a Atomického designu je podle mého názoru vhodné a usnadňuje práci na designu. Pomocí tvoření jednotlivých součástí se v kódu snáze orientuje a dá se lépe nastylovat každá část. Při tvorbě designu mi v usnadnil práci.

Osobně si myslím, že Atomický design je budoucnost tvorby designu webu. U větších projektů jednoznačně usnadňuje práci na designu a pomáhá autorovi se orientovat i v dříve vytvořených částech. Nevhodný je podle mého názoru pouze pro malé projekty, u kterých si vystačíme i bez něj a použití tvorby jednotlivých částí by návrh designu spíše zpomalovalo.

Seznam použité literatury a zdrojů

- [1] MICHÁLEK, Martin. *Atomický design a Pattern Lab: návrat do budoucnosti návrhu uživatelských rozhraní* [online]. 2017 [cit. 2018-04-17]. Dostupný z WWW: <<https://www.vzhurudolu.cz/prirucka/pattern-lab>>. ISSN 1802-2804.
- [2] Atomic Design by Brad Frost. *Atomic Design by Brad Frost* [online]. 2016 [cit. 2018-04-17]. Dostupné z WWW: <<http://atomicdesign.bradfrost.com/>>
- [3] Website Style Guide Resources. *Website Style Guide Resources* [online]. [cit. 2018-04-17]. Dostupné z: <<http://styleguides.io/>>
- [4] BEM — Block Element Modifier. *BEM — Block Element Modifier* [online]. [cit. 2018-04-17]. Dostupné z: <<http://getbem.com>>
- [5] U.S. Web Design System: A design system for the federal government. *U.S. Web Design System: A design system for the federal government* [online]. [cit. 2018-04-17]. Dostupné z: <<https://designsystem.digital.gov/>>
- [6] arrel.css - CSS Toolkit by Inktweb.nl. *marrel.css - CSS Toolkit by Inktweb.nl* [online]. [cit. 2018-04-17]. Dostupné z: <<https://marrel.inktweb.nl/>>
- [7] Pattern Lab | Build Atomic Design Systems. *Pattern Lab | Build Atomic Design Systems* [online]. [cit. 2018-04-17]. Dostupné z: <<http://patternlab.io>>
- [8] How To Make And Maintain Atomic Design Systems With Pattern Lab 2 — Smashing Magazine. *Smashing Magazine — For Web Designers And Developers — Smashing Magazine* [online]. [cit. 2018-04-17]. Dostupné z: <<https://nodesource.com/blog/installing-nodejs-tutorial-windows/>>

- [9] Installing Node.js Tutorial: Windows - NodeSource *For mission-critical Node.js applications - NodeSource* [online]. [cit. 2018-04-17]. Dostupné z: <<https://www.smashingmagazine.com/2016/07/building-maintaining-atomic-design-systems-pattern-lab/>>
- [10] MICHÁLEK, Martin. *Grunt.js* [online]. 2017 [cit. 2018-04-17]. Dostupný z WWW: <<https://www.vzhurudolu.cz/prirucka/grunt>>.
- [11] URBAN, Petr. *Loni Windows 10 rostl, ale Windows 7 příliš neustoupil. Chrome se stal synonymem pro web* [online]. 2018 [cit. 2018-04-16]. Dostupný z WWW: <<https://www.cnews.cz/statistiky-2017-chrome-windows-10-rostly/2/>>.

Seznam obrázků

1	Ověření, zda je node nainstalován	22
2	Dialog s popiskem prvku	27
3	Dialog při neexistenci popisku prvku	27
4	Zobrazení navrženého designu pro pc v Google Chrome	45
5	Zobrazení navrženého designu pro mobily v Google Chrome	45
6	Zobrazení navrženého designu pro pc v Mozille Firefox	46
7	Zobrazení navrženého designu pro mobily v Mozille Firefox	46
8	Zobrazení navrženého designu pro pc v Internet Exploreru	47
9	Zobrazení navrženého designu pro mobily v Internet Exploreru	47

Seznam příkladů

1	Ověření, zda je Node.js nainstalovaný	21
2	Instalace Grunt	22
3	Instalace Gulp	22
4	Instalace Pattern Labu	22
5	Instalace vzorových dat do Pattern Labu	23
6	Vložení dat do souboru s daty	25
7	Vložení pole dat do souboru s daty	25
8	Vložení dat s převodem na entity	25
9	Vložení dat se zachováním Html značek	25
10	Vložení dat z pole	26
11	Obsah souboru .md	26
12	Generování systému webového designu	28
13	Odkazy na css styly	29
14	Vytvoření hlavičky	36
15	Vytvoření patičky	37
16	Vytvoření formuláře	37
17	Vytvoření bočního panelu	38
18	Vytvoření textu stránky s kontakty	38
19	Vytvoření textu stránky s příklady	39
20	Vytvoření textu stránky s teorií	39
21	Vytvoření centrálního textu pro úvodní stránku	40
22	Vytvoření šablony úvodní stránky	40
23	Vytvoření šablony pro stránky s teorií	41
24	Vytvoření šablony pro stránky s příklady	42
25	Vytvoření šablony pro stránku s kontaktem	42
26	Vytvoření stránky ze šablony - obsah souboru se stránkou	43

A Příloha

CD - obsahuje elektronickou verzi textu práce, obrázky použité v práci, kompletní obsah složky s Pattern LABem(zdrojové soubory, soubory s výstupem)

B Příloha

Web: <http://bp.mhdtab.cz/> - kompletní výstup z Pattern Labu. Přehled celého designového systému.