

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií



Bakalářská práce

Tvorba responzivní webové aplikace pro oblast e-Learningu

Vladimír Pastorek

© 2018 ČZU v Praze

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Vladimír Pastorek

Informatika

Název práce

Tvorba responzivní webové aplikace pro oblast e-Learningu

Název anglicky

Creating responsive Web Application for e-Learning

Cíle práce

Hlavním cílem práce je kritická analýza možností tvorby responzivní webové aplikace pomocí jedné z nejvíce rozšířených sad nástrojů pro tvorbu responzivního webu – Bootstrap. Dílčím cílem práce je zhodnocení možností tvorby webové aplikace kombinací frameworku Bootstrap a javascriptové knihovny JQuery. Výsledkem praktické části bakalářské práce bude na základě syntézy zjištěných poznatků vytvoření responzivní webové aplikace pro oblast e-Learningu, která bude zaměřena na vytváření a správu online testů.

Metodika

Metodika řešené problematiky je založena na studiu a analýze odborných informačních zdrojů zabývajících se zvolenou problematikou. Na základě získaných informací budou popsány možnosti tvorby responzivních webových aplikací. V praktické části práce bude navržena a vytvořena responzivní webová aplikace pro oblast e-Learningu.

Doporučený rozsah práce

40-60 stran

Klíčová slova

webová aplikace, responzivní design, HTML, CSS, bootstrap, PHP, MySQL, jQuery, eLearning

Doporučené zdroje informací

- Bootstrap Framework [online]. [cit. 2017-06-23]. Dostupné z: <http://getbootstrap.com/>
- CASTRO, E. – HYSLOP, B. *HTML5 a CSS3 : názorný průvodce tvorbou WWW stránek*. Brno: Computer Press, 2012. ISBN 978-80-251-3733-8.
- EFTOS, M. – CASTLEDINE, E. – WHEELER, M. *Vytváříme mobilní web a aplikace pro chytré telefony a tablety*. Brno: Computer Press, 2013. ISBN 978-80-251-3763-5.
- jQuery – Javascript library [online]. [cit. 2017-06-23]. Dostupné z: <http://jquery.com/>
- KADLEC, T. *Responzivní design profesionálně*. Brno: Zoner Press, 2014. ISBN 978-80-7413-280-3.
- MAYER, R E. – CLARK, R C. *E-learning and the science of instruction : proven guidelines for consumers and designers of multimedia learning*. San Francisco: Pfeiffer, 2003. ISBN 0-7879-6051-9.
- NIELSEN, J. *Web.Design*. Praha: SoftPress, 2002. ISBN 80-86497-27-5.
- PHP.net [online]. [cit. 2017-06-23]. Dostupné z: <http://php.net/>
- ROSENBERG, M J. *E-learning : strategies for delivering knowledge in the digital age*. New York: McGraw-Hill, 2001. ISBN 0-07-136268-1.
- SUDICKÝ, P. – ZOUNEK, J. *E-learning : učení (se) s online technologiemi*. Praha: Wolters Kluwer Česká republika, 2012. ISBN 978-80-7357-903-6.
- W3C [online]. 2017 [cit. 2017-06-23]. Dostupné z: <http://www.w3.org/>
-

Předběžný termín obhajoby

2017/18 LS – PEF

Vedoucí práce

Ing. Petr Benda, Ph.D.

Garantující pracoviště

Katedra informačních technologií

Elektronicky schváleno dne 30. 10. 2017

Ing. Jiří Vaněk, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 1. 11. 2017

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 13. 03. 2018

Čestné prohlášení

Prohlašuji, že svou bakalářskou práci "Tvorba responzivní webové aplikace pro oblast e-Learningu" jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené bakalářské práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 14.3.2018

Poděkování

Rád bych touto cestou poděkoval panu Ing. Petru Bendovi Ph.D. za odborné vedení závěrečné práce a za pomoc při práci na projektu.

Tvorba responzivní webové aplikace pro oblast e-Learningu

Souhrn

Práce se zabývá popsáním tvorby webových aplikací pomocí základních technologií a následně její realizací pro oblast e-learningu. První část práce seznamuje s webovými aplikacemi, jejich popisem a vlastnostmi. Poté je představena možnost vývoje webu, či webové aplikace pomocí základních technologií, které jsou reprezentovány klientskou částí. Těmito technologiemi jsou značkovací jazyk HTML, kaskádové styly CSS a skriptovací jazyk Javascript společně s knihovnou JQuery. Je také vysvětlen responzivní design a jeho řešení pomocí Bootstrap, což je jedna z nejrozšířenějších sad nástrojů pro tvorbu právě responzivního designu. Jsou popsány základní technologie, které se zpracovávají na straně serveru, jimiž jsou programovací jazyk PHP a databáze MySQL. Pomocí získaných poznatků je vytvořena responzivní e-learningová aplikace, která je zaměřena na vytváření a správu testů. Tyto testy jsou přístupné běžným uživatelům, kteří si je mohou vyplnit a následně jsou jim testy vyhodnoceny. Praktická část se zabývá popsáním vytvořené aplikace a poté analýzou její tvorby, porovnáním vývoje se sadou Bootstrap a reálným využitím jejich částí. V poslední části práce je zjištěn vliv vývoje na koncového uživatele.

Klíčová slova: webová aplikace, responzivní design, HTML, CSS, bootstrap, PHP, MySQL, JQuery, eLearning

Creating responsive Web Application for e-Learning

Summary

The thesis describes a creation of web applications using basic technologies followed by their implementation for e-learning. The first part of the thesis introduces web applications, their description and properties. After that the possibility of web development or web application development using the basic technologies represented by the client part is presented. These technologies include HTML mark-up language, CSS cascading styles, and JavaScript scripting language along with the JQuery library. Responsive design and its solutions using Bootstrap, one of the most popular sets of tools for creating responsive design, is also explained. Basic server-side technologies, such as PHP programming language and MySQL database are described as well. Using the data gained, a responsive e-learning application is created that focuses on creating and managing tests. These tests are accessible to regular users who can complete them and then the tests are marked for them. The practical part deals with describing the created application and then analysing its creation, comparing the development with the Bootstrap set and the real use of their parts. The final part of the thesis identifies the impact of the development on the end user.

Keywords: web application, responsive design, HTML, CSS, bootstrap, PHP, MySQL, jQuery, eLearning

Obsah

1 Úvod.....	11
2 Cíl práce a metodika	12
2.1 Cíl práce	12
2.2 Metodika	12
3 Teoretická východiska	13
3.1 Webová aplikace	13
3.2 Klientská část	14
3.2.1 Grafika	14
3.2.2 HTML	15
3.2.3 CSS	16
3.2.4 Javascript	19
3.2.5 JQuery	20
3.3 Serverová část	21
3.3.1 PHP	21
3.3.2 Databáze a MySQL.....	22
3.4 Responzivní webdesign.....	22
3.4.1 Media Queries.....	23
3.4.2 Responzivní webdesign z hlediska SEO.....	24
3.4.3 Technické řešení – Bootstrap.....	25
3.5 E-learning	27
3.5.1 Online vzdělávací technologie.....	28
3.5.2 Potencionál.....	28
4 Vlastní práce	29
4.1 Vytvoření základní struktury.....	29
4.1.1 Přístup k databázi a základní proměnné	30
4.2 Uživatel v aplikaci.....	30
4.2.1 Registrace.....	30
4.2.2 Přihlášení	30
4.2.3 Uživatelský profil	31
4.2.4 Správa testů.....	31
4.2.5 Vyplňování testů	32
4.3 Responzivní design	35
4.3.1 Použité komponenty sady Bootstrap.....	36
4.4 Javascriptové funkce	36
4.4.1 Dynamická interakce s prvky pomocí JQuery	37

4.5	Výsledná aplikace a její struktura	38
4.5.1	Souborová struktura	39
4.5.2	Databázová struktura	40
4.5.3	Mapa stránek	41
5	Zhodnocení výsledků	42
5.1	Analýza tvorby	42
5.1.1	Velikost aplikace.....	42
5.1.2	Vliv vývoje na koncového uživatele.....	43
6	Závěr.....	45
7	Seznam použitých zdrojů	47
	Přílohy.....	49

Seznam obrázků

Obrázek 1 -	Vývojářský editor PSPad (Zdroj: autor)	14
Obrázek 2 -	příklad jednoduchého HTML dokumentu (Zdroj: autor).....	16
Obrázek 3 -	zápis CSS (Zdroj: autor)	17
Obrázek 4 -	in-line zápis stylu odstavce (Zdroj: autor)	17
Obrázek 5 -	interní typ zápisu stylu odstavce (Zdroj: autor)	17
Obrázek 6 -	načtení externího souboru se styly (Zdroj: autor)	18
Obrázek 7 -	obsah samostatného CSS souboru (Zdroj: autor).....	18
Obrázek 8 -	in-line zápis Javascriptu do elementu (Zdroj: autor)	20
Obrázek 9 -	interní typ zápisu Javascriptu (Zdroj: autor)	20
Obrázek 10 -	připojení externího Javascriptového souboru (Zdroj: autor)	20
Obrázek 11 -	příklad přidání třídy v JQuery (zdroj: Autor)	21
Obrázek 12 -	ilustrace serverového scriptu (Zdroj: Jak psát web)	21
Obrázek 13 -	zobrazení prvků pomocí responzivního webu na více zařízeních (Zdroj: WebAppRater)	23
Obrázek 14 -	příklad použití media queries (Zdroj: autor).....	24
Obrázek 15 -	příklad použití viewportu (Zdroj: autor)	24
Obrázek 16 -	příklad mobile friendly webu ve vyhledávači Google (Zdroj: Google Webmaster Central Blog)	25
Obrázek 17 -	připojení Bootstrap knihoven do HTML (Zdroj: autor)	25
Obrázek 18 -	seznam důležitých souborů obsahující Bootstrap po stažení (Zdroj: autor) ..	26
Obrázek 19 -	příklad použití grid systému (Zdroj: autor).....	27
Obrázek 20 -	použití Bootstrap glyphicon (Zdroj: autor)	27
Obrázek 21 -	logo aplikace (Zdroj: autor)	29
Obrázek 22 -	základní struktura aplikace (Zdroj: autor).....	30
Obrázek 23 -	výběr registrovaného e-mailu (Zdroj: autor).....	30
Obrázek 24 -	kontrola přihlášení (Zdroj: autor).....	31
Obrázek 25 -	shrnutí profilu v aplikaci (zdroj: autor).....	31
Obrázek 26 -	nastavení vytvářeného testu (zdroj: autor)	32

Obrázek 27 - nastavení otázky při vytváření testu (zdroj: autor)	32
Obrázek 28 - informace o testu (zdroj: autor).....	33
Obrázek 29 - výpis testu a otázek v aplikaci (Zdroj: autor).....	34
Obrázek 30 - vyhodnocení testu (zdroj: autor)	34
Obrázek 31 - vyhodnocení otázky (zdroj: autor)	35
Obrázek 32 - příklad použití grid system v záhlaví aplikace (Zdroj: autor).....	35
Obrázek 33 - použití media queries v aplikaci (Zdroj: autor).....	36
Obrázek 34 - přihlášení v aplikaci (Zdroj: autor)	37
Obrázek 35 - předávání informací při přihlášení: AJAX (zdroj: Autor)	37
Obrázek 36 - zobrazení hlavní strany aplikace na třech zařízeních (zdroj: autor)	38
Obrázek 37 - seznam veřejných testů (zdroj: autor)	39
Obrázek 38 - databázová struktura aplikace	40
Obrázek 39 - Mapa stránek a přístupu uživatele.....	41

Seznam tabulek

Tabulka 1 - velikost aplikace (zdroj: autor).....	42
Tabulka 2 - měření načtení webu na různých zařízeních s různými typy připojení (zdroj: autor).....	43

1 Úvod

Počítače, internet a další technologie změnily svět, změnily způsob myšlení společnosti a lidí. Díky celosvětové síti je přenos informací a dat daleko rychlejší, a to zejména díky webům. Ty dnes již nejsou záležitostí pouze stolních počítačů a notebooků, ale i daleko menších zařízení. Chytrý telefon je dnes již samozřejmostí a jeho používání je na denním pořádku, přičemž svým výpočetním výkonem a funkcionalitou dokáže nahradit stolní počítač, či notebook. Tím mu dopomáhá fakt, že obsahuje operační systém a například internetový prohlížeč. Právě internetový prohlížeč je dějištěm fungování webů a webových aplikací, a to nehledě na platformu, či právě operační systém. Proto existují možnosti, jak uživateli na menším zařízení usnadnit jeho pohyb a orientaci na webu. Jednou z možností, jak mu toto zajistit je, aby web či webová aplikace obsahovala responzivní design. Technologie, díky kterým jsou weby tvořeny standardizovanými postupy, dokáží korektně přizpůsobit obsah na uživatelovo zařízení.

Avšak variabilita těchto zařízení je velice velká, liší se od sebe mnoha parametry, proto zajištění a optimalizace je následně dalším velkým úkolem, při vývoji určitého projektu. Stejně tak rozmístění prvků a práce s obsahem se může na chytrých telefonech lišit. Proto existují sady nástrojů, které vývojářům mají ušetřit práci, čas a obsahují mnoho prvků, které se díky vysokému využití mohou neoficiálně stát standardem.

Člověk se učí již od mala, děti vyrůstají v tomto moderním světě a splynou s dobou, i technologiemi. Výpočetní technika se stále modernějšími technologiemi se dostává také například do škol a stejně tak systémy, či aplikace, které mají za cíl díky vyspělým technologiím nabídnout lidem jiný způsob učení a předávání znalostí.

Tato práce si klade za cíl popsat možnosti tvorby responzivní webové aplikace a následně na základě poznatků responzivní webovou aplikaci vytvořit pro oblast e-learningu.

Teoretická část se zabývá a popisuje základní technologie pro tvorbu webu a webových aplikací. V praktické části je popsán vývoj aplikace a následně se zhodnotí vývoj pomocí jedné z nejvíce rozšířené sady pro tvorbu responzivního designu.

2 Cíl práce a metodika

2.1 Cíl práce

Hlavním cílem práce je kritická analýza možností tvorby responzivní webové aplikace pomocí jedné z nejvíce rozšířených sad nástrojů pro tvorbu responzivního webu - Bootstrap. Dílčím cílem práce je zhodnocení možností tvorby webové aplikace kombinací frameworku Bootstrap a javascriptové knihovny JQuery. Výsledkem praktické části bakalářské práce bude na základě syntézy zjištěných poznatků vytvoření responzivní webové aplikace pro oblast e-Learningu, která bude zaměřena na vytváření a správu online testů.

2.2 Metodika

Metodika řešené problematiky je založena na studiu a analýze odborných informačních zdrojů zabývajících se zvolenou problematikou. Na základě získaných informací budou popsány možnosti tvorby responzivních webových aplikací. V praktické části práce bude navržena a vytvořena responzivní webová aplikace pro oblast e-Learningu.

3 Teoretická východiska

3.1 Webová aplikace

Vzhledem k tomu, že ve světě moderních informačních technologií je možné nalézt velké množství zařízení (chytré telefony, tablety, počítače), které se v zásadě od sebe liší například značkou, velikostí, ale především operačním systémem, tak vývoj aplikací může přinést určitou část obtíží. Právě kvůli různorodosti operačních systémů, či jejich samotných verzí, může vývoj programátorovi ztížit práci, která místo ladění samotné aplikace může přerůst k vývoji a ladění aplikace na všech operačních systémech. Tato práce následně může narůst jak časově, tak samozřejmě finančně. Proto je v některých případech vcelku logické, po přezkoumání všech možností, vzít v potaz i variantu, díky které se dá těmto problémům předejít (ManagementMania.com).

V první řadě je nejdůležitější nalézt aplikaci, kterou nabízí jakýkoliv operační systém a ta umožní výstup informací z druhé strany (serverové), pomocí stejných protokolů a jazyků. Právě takovou aplikací je webový prohlížeč (ManagementMania.com). Ten pomocí HTTP protokolu komunikuje s webovým serverem a následně data, která přijme, zobrazí na výstupu pomocí HTML a grafické formátování pomocí CSS. V zásadě, takto komunikují takřka veškeré webové stránky. Avšak ty, díky použití pokročilejších programovacích jazyků, dokážou sloužit nejen jako obyčejná webová prezentace, ale jako pokročilá webová aplikace. Moderní webové aplikace následně umí prakticky to samé, jako nainstalovaný software v uživatelově zařízení. Příkladem mohou být kancelářské balíčky, e-mailové klienti, či komunikační aplikace (Michl, 2012; Hlaváček, 2015; Michálek, 2017).

Výhody:

- Není potřeba vývoj pro všechny platformy a operační systémy
- Není potřeba nic instalovat, pouze mít webový prohlížeč
- Veškerá aktualizace aplikace se děje pouze na straně serveru (skripty na straně serveru) a grafické změny si stáhne prohlížeč znovunačtením stránky

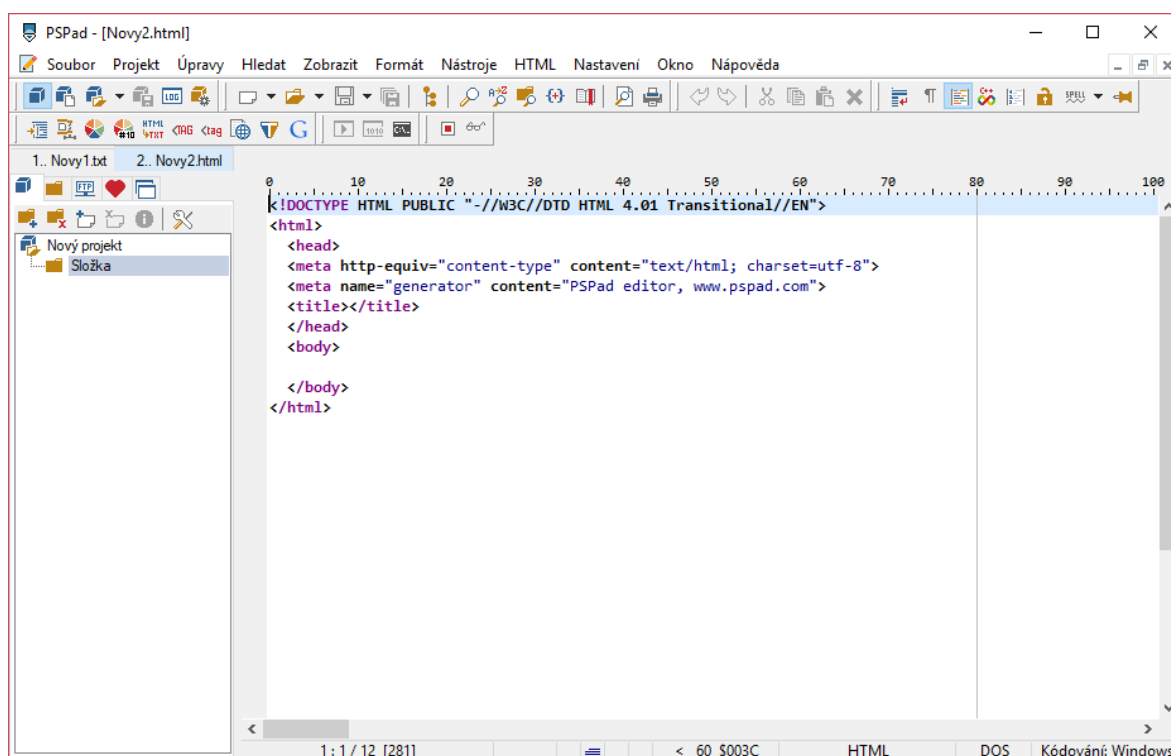
Nevýhody:

- Nižší požitok, než používání nativní aplikace pro daný operační systém (Rychlost aplikace, vzhled, tok dat, použití funkcí, periférií zařízení)
- Alespoň při prvním spuštění požadavek na internet

3.2 Klientská část

Pro výstup dat, informací na klientské části slouží řada jazyků či nástrojů. Nejzákladnější aplikací, kde se celý tento děj odehrává, je zmiňovaný webový prohlížeč.

Pokud by při vývoji webu, či webové aplikace měl být nějaký chronologický postup, tak by se v první řadě vytvářely grafické návrhy. Ty se mohou zpracovávat v běžných grafických editorech, jak rastrových či vektorových. Následně převedení do webové podoby, kterou dokáže přečíst prohlížeč je kódování v HTML a CSS jazycích. Stejně tak skripty, které se provádí na straně serveru. Ty mohou být psány v obyčejných textových editorech, či ve specializovaných vývojářských editorech, které mohou ulehčit autorovi práci a přehled.



Obrázek 1 - Vývojářský editor PSPad (Zdroj: autor)

3.2.1 Grafika

Grafika, která se váže k webovým technologiím, slouží především ke zvýraznění a zefektivnění webových stránek, či aplikací. Pomocí správně zvolených barev, animací a speciálních efektů dokáže následně obsah webu nabídnout návštěvníkovi lepší požitek. V opačném případě může zákazníka takováto stránka uživatele odradit od návštěvy, a k tomu hlavnímu účelu webu se ani nedostane. Obecně taktéž platí, že webová grafika by měla celkově zapadat do kontextu a účelu daného webu (Rose India; Salvat, 2010; Chastian, 2017).

3.2.2 HTML

HTML neboli HyperText Markup Language je název značkovacího jazyka, který se používá převážně při tvorbě webových stránek či webových aplikací. První definici jazyka HTML vymyslel Tim Berners-Lee v roce 1991 jako součást projektu WWW (Kosek, 1998b).

Při vývoji jazyka HTML byl kladen důraz zejména na to, aby bylo možné jeho prostřednictvím prezentovat textový obsah v určité logické struktuře a aby byl tento jazyk zvládnutelný i pro technicky méně zdatnou veřejnost (Škultéty, 2004).

Pomocí HTML se dá snadno popsat struktura webu pomocí značek, či tagů. Ty jsou strukturovány do bloků. Tagy se dělí na dvě skupiny: párové a nepárové. Párové tagy se vyskytují ve dvojicích, kde první tag označí určitý obsah, než najde ten samý ukončující tag, například `<html> obsah </html>`. Nepárový tag vykoná úkol, aniž by musel být uzavřený, například `
`. Tagy také mohou obsahovat atributy, které elementu mohou doplnit jeho funkčnost, či jsou v základu požadovány pro specifickou funkčnost. Prohlížeče následně nezobrazují psané HTML tagy, slouží tak jako reprezentace toho, jak bude výsledný HTML dokument vypadat (W3 Schools).

Dokument v jazyku HTML má předepsanou strukturu:

- První se musí deklarovat typ dokumentu, který sdělí prohlížeči, že otevřel HTML dokument. K tomu slouží nepárový tag `<!DOCTYPE html>`
- Následně se párovým tagem `<html>` reprezentuje celý dokument, kde se nachází HTML značky
- Poté se musí vytvořit HTML hlavička dokumentu pomocí párového tagu `<head>`, která obsahuje metadata pro daný HTML dokument
- Následuje tělo dokumentu, které se označí také pomocí párového tagu a tím je `<body>`. To musí obsahovat vlastní obsah dokumentu.

```

<!DOCTYPE html>
<html>
  <head>
    <title>Titulek</title>
  </head>
  <body>
    <h1>Nadpis stránky</h1>
    <p>Toto je <a href="http:// example.com/">odkaz</a> v odstavci.</p>
    <!-- toto je komentář -->
  </body>
</html>

```

Obrázek 2 - příklad jednoduchého HTML dokumentu (Zdroj: autor)

3.2.2.1 HTML5

HTML5 je aktuálně nejnovější verzí, která přináší řadu změn a novinek. Snaží se především reflektovat potřeby současného i budoucího webu. Samozřejmostí je, že zdědí drtivou většinu funkcí od svých předchozích verzí (Castro, 2012b).

Nejdůležitější HTML5 funkce (W3 Schools):

- Nové sémantické, grafické a multimediální elementy
- Nové atributy pro elementy ve formulářích
- Zjednodušené nastavení jazyku dokumentu a kódování
- Uložení dat do lokálního úložiště – podpora offline aplikací

3.2.3 CSS

Webový obsah, pouze otagovaný a uspořádaný pomocí HTML by sám o sobě graficky moc přívětivý nebyl (také pokud již například je vytvořen grafický návrh stránek). Proto existuje jazyk CSS, který určuje, jak obsah graficky vypadá. CSS je zkratka pro Cascading Style Sheets, což se česky překládá jako kaskádové styly.

První verze jazyka CSS se objevily až několik let po vzniku jazyka HTML, přičemž oficiálně se tento jazyk prosadil v roce 1996 (Castro, 2012a).

Jazyk vytvořila a je pod správou standardizační organizací W3C a autor prvního návrhu byl Håkon Wium Lie.

Syntaxe je taková, že se musí identifikovat prvek (k tomu slouží selektory či in-line zápisy přímo do prvku). Následně se musí specifikovat daná vlastnost, ukončit ji dvojtečkou `:`, poté vlastnost popsat a ukončit středníkem `;`. Výjimky tvoří právě prvky, které jsou identifikovány přímo v HTML dokumentu, či jsou obsahem jiných elementů, nebo určují nějakou akci s prvkem (W3 Schools).

```
color: red;
```

Obrázek 3 - zápis CSS (Zdroj: autor)

3.2.3.1 Typ zápisu

Kaskádové styly mají tři možnosti, jak daný styl přenést do výsledného obsahu (Jak psát web):

- **In-line zápis** – píše se přímo k danému HTML prvku. Může se takto nastylovat jeden konkrétní HTML tag a také má tento styl větší prioritu, než například styl, který bude v externím souboru. To se může hodit, pokud by jeden z mnoha prvků, v daném elementu byl odlišný a autor by chtěl rychle a přímo v dokumentu, tento prvek odlišit.

```
<p style="color: red;">Tento odstavec má zadán inline styl a bude tedy červený</p>
```

Obrázek 4 - in-line zápis stylu odstavce (Zdroj: autor)

- **Interní styl** – používá se jako interní zápis CSS (nejčastěji do hlavičky HTML dokumentu). V tomto zápisu se vloží napsané styly vždy, mezi párový tag `<style>`.

```
<style>
  p {
    color: red;
  }
</style>
```

Obrázek 5 - interní typ zápisu stylu odstavce (Zdroj: autor)

- **Externí CSS styl** – používá se externě vytvořený CSS soubor připojený do hlavičky `<head>` HTML dokumentu. Ten musí být navázaný na každou danou podstránku, kde má být styl aplikovaný. Důležitý je nepárový `<link>` tag, kde musí být

v atributu popsána cesta k souboru. Použitý soubor s příponou .css obsahuje pouze popis vlastností daných prvků.

```
<head>
  <link rel='stylesheet' href='style.css' type='text/css'>
</head>
```

Obrázek 6 - načtení externího souboru se styly (Zdroj: autor)

```
p {
  color: red;
}
.oznaceny {
  Background: yellow;
}
```

Obrázek 7 - obsah samostatného CSS souboru (Zdroj: autor)

Z toho plyne, že u interních a externích stylů se vlastnosti pro dané elementy píší pomocí selektorů.

3.2.3.2 Selektory

Existuje mnoho způsobů, jak označit daný prvek. Vždy je nejlepší volit nejefektivnější selektor, aby nedošlo k redundanci CSS vlastností. Základní selektory jsou (Interval.cz, 2001):

- Typový selektor – Platí pro všechny výskytu daného elementu př.: body
- Selektor třídy – Platí pro všechny elementy, který mají libovolně nazvaný atribut class př.: pro `<p class="test">` platí `.test`
- ID selektor – Platí pro všechny elementy, který mají libovolně nazvaný atribut id př.: pro `<p id="test">` platí `#test`
- Selektor následníka – Platí pro všechny elementy, které jsou uvnitř jiného elementu př.: `body p`
- Pseudotřídy – Speciální výběr, na základě nějakého stavu, či akce př.: `p:hover`, `p:focus`
- Multiselekce – Seskupování více prvků podle čárek, na které má platit konkrétní vlastnost př.: `.test1, #test2, body p`

3.2.3.3 CSS3

CSS3 je aktuálně nejnovější verzí, která se pojí se standardem HTML5. Přináší řadu změn a novinek.

Nejdůležitější CSS3 funkce (W3 Schools):

- Nové selektory
- Nové možnosti pozadí a ohraničení
- Hodnoty obrázků a nahrazený obsah
- Textové efekty
- Animace
- Media Queries

3.2.4 Javascript

Skripty, které fungují na straně klienta, jsou psány ve skriptovacím jazyku Javascriptu. Tyto skripty umí v základu překládat prohlížeč, obvykle až po stažení webových stránek. Jeho účel je web udělat více dynamický a oživit, či urychlit tak zobrazování obsahu, či provádění funkcí (Škultéty, 2004; W3 Schools).

Autorem Javascriptu je společnost Netscape, která jej vyvíjela koncem minulého století. Dříve se jazyk nazýval LiveScript, avšak z marketingového hlediska jej přejmenovali na Javascript, ačkoliv to s Javou nemá téměř nic společného (Škultéty, 2004).

Javascript dokáže ovládat a pracovat s HTML prvky (vytvářet je, mazat je), které mohou mít již aplikované nějaké styly. Stejně tak právě umí pracovat s kaskádovými styly, tudíž v reálném čase je měnit. Reaguje na události, či akce myši nebo klávesnice. Umožňuje také použití objektů, lehce řečeno spolupracovat s webovým prohlížečem. Například dokáže otevírat nová okna v prohlížeči (W3 Schools).

Protože práce se surovým Javascriptem je velice časově náročná, existuje proto velké množství vytvořených knihoven, které ušetří autorovi webu velké množství času a nabídne nové funkce, či jejich snazší použití.

3.2.4.1 Typ zápisu

Javascript má tři možnosti, jak daný skript přenést do výsledného obsahu, či jej aplikovat na daný prvek (Jak psát web):

- **In-line zápis** – píše se přímo k danému HTML prvku. Využívá se atributu u daného elementu, který většinou reaguje na nějakou akci, nebo se používá u párového tagu `<a>`, který v atributu href odkazuje na javascriptovou funkci, místo URL adresy.

```
<a href="http://www.seznam.cz" onmouseover="alert('Alert okno!')">Po  
njetí myši vyskočí alert okno</a>
```

Obrázek 8 - in-line zápis Javascriptu do elementu (Zdroj: autor)

- **Interní styl** – píše se prakticky kamkoliv do HTML dokumentu. Skript se následně vloží mezi párový html tag `<script>` a skript se provede, jakmile jej prohlížeč přečte.

```
<script>  
  alert(Alert okno!');  
</script>
```

Obrázek 9 - interný typ zápisu Javascriptu (Zdroj: autor)

- **Externí Javascriptový styl** – používá se externě vytvořený CSS soubor připojený do HTML dokumentu. Ten musí být navázaný na každou danou stránku, kde má být skript proveden. Důležitý je `<script>` tag, kde musí být v atributu `src` popsána cesta k souboru. Použitý soubor s příponou „.js“ obsahuje pouze javascriptové skripty.

```
<script src="funkce.js"></script>
```

Obrázek 10 - připojení externího Javascriptového souboru (Zdroj: autor)

3.2.5 JQuery

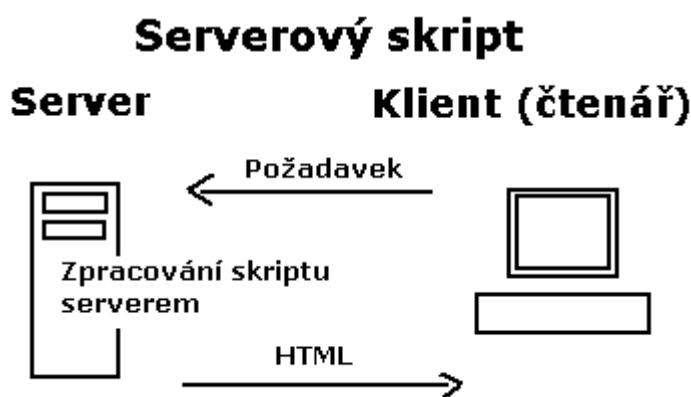
JQuery je rychlá, jednoduchá Javascriptová knihovna, nabízející nepřehledné množství funkcí a vlastností. Díky této knihovně se lépe pracuje a manipuluje s HTML elementy, snadněji se zpracovávají události, lehce vytváření animace, či ajaxové funkce. To co se v Javascriptu musí popsat několika řádky lze v JQuery popsat pouze jedním řádkem. Jednou z nejužitečnějších funkcí je podpora selektorů, tudíž vybrat jakýkoliv identifikovaný element není žádný problém. Základní koncept JQuery je funkce „\$“. Jedná se o jmenný prostor pro JQuery a tato funkce je potřeba například právě při selektování elementů, nebo aplikování JQuery funkcí (jQuery - Javascript library).

```
$(".znaceny_prvek").addClass("nova_trida");
```

Obrázek 11 - příklad přidání třídy v JQuery (zdroj: Autor)

3.3 Serverová část

Druhou stranou webových aplikací, tu kterou uživatel nevidí, jsou právě servery, serverová část, kde se vykonávají dynamické skripty. V praxi fungující tak, že klient zašle požadavek na webu, může se jednat o načtení webu, potvrzení formuláře, či načtení webu s jinými parametry. Následně tento požadavek server zpracuje, vyhodnotí a získaná data zpracuje, dle navržené aplikace a zašle zpátky klientovi. Tomu se opět samozřejmě dle navržené aplikace vyhodnotí data v prohlížeči (PHP.net).



Obrázek 12 - ilustrace serverového skriptu (Zdroj: Jak psát web)

3.3.1 PHP

Nejrozšířenějším skriptovacím jazykem pro webové aplikace, které jsou zpracovávány na straně serveru, je PHP. Ke květnu 2017 má podíl přes 82% v používání (W3 Techs). Jedná se o skriptovací jazyk, který založil Rasmus Lerdorf. Původně zkratka PHP znamenala Personal Home Page, později byl ale tento název změněn na PHP HyperText Preprocessor. PHP také velice často spolupracuje s databázovými servery, čímž jeho funkcionality vzroste (Kosek, 1998a).

Důvody proč je PHP tak oblíbené (Zajíc, 2004):

- Je relativně jednoduché na pochopení.
- Má podobnou syntaxi jazyku C.
- Podporuje širokou řadu souvisejících technologií, formátů a standardů.

- Je to otevřený projekt s rozsáhlou podporou komunity.
- Je multiplatformní a lze jej provozovat s většinou webových serverů.

3.3.2 Databáze a MySQL

Právě databáze dělá jazykem PHP mocný nástroj, protože do ní se mohou ukládat veškeré potřebné informace a následně k nim snadno přistupovat. MySQL je SŘBD (Systém řízení báze dat), který je šířen zdarma jako freeware. Uplatňuje relační databázový model a komunikace s ní probíhá pomocí dotazovacího jazyka SQL (Kosek, 1998a; PHP.net).

Údaje jsou uspořádány do tabulek, které zpravidla shromažďují údaje o jednom druhu objektu. Aby byla možná práce s tabulkami a uloženými daty pracovat, musí být jednoznačně identifikovány. Jednotlivé sloupce v tabulkách musí být unikátně pojmenovány, stejně tak se musí dát jednoznačně identifikovat každý řádek v tabulce. Proto se používají primární klíče – jako jeden ze sloupců v dané tabulce a právě v daném řádku musí být tato buňka jedinečně identifikována. Pro každý sloupec se také musí určit, jaký typ dat mohou sloupce obsahovat. Databáze mohou obsahovat více tabulek, a proto je také možné s nimi kooperovat. Atribut, který slouží jako odkaz na jinou tabulku, se nazývá cizí klíč. Ten se spojí s primárním klíčem jiné tabulky. Tento vztah nemusí být pouze mezi dvěma tabulkami, může být mezi několika dalšími (Kosek, 1998a).

Mezi hlavní výhody MySQL patří:

- Snadná implementovatelnost
- Vysoký výkon, rychlost
- Jedná se o volně šiřitelný software
- Jednoduchý programový systém phpMyAdmin pro snadnou správu MySQL

3.4 Responzivní webdesign

Pojem responzivní webdesign označuje tvorbu webu, která se dokáže přizpůsobit velikosti displeje zařízení (Bittner, 2015).

Pokud se používá pojem mobilní, může to znamenat mnoho různých věcí. Vývoj webů, webových aplikací určených pro mobilní zařízení (toto může být jedna z věcí, co může znamenat pojem mobilní) se na první pohled nijak neliší od vývoje jakéhokoliv webu či webové aplikace. Používají se stejné technologie, možné jsou samozřejmě nadstavby

jednotlivých technologií. Proto se může zdát, že web pro mobilní zařízení se pouze optimalizuje. Avšak není to zas tak jednoduché (Castledine, 2013).



Obrázek 13 - zobrazení prvků pomocí responzivního webu na více zařízeních (Zdroj: WebAppRater)

Není podstatné, jaký operační systém zařízení vlastní, zda je to Android či iOS, jaké funkce aktuálně nabízí. Změny v této oblasti se dějí prakticky neustále. Důležité je však, že vývoj mobilní aplikace není o těchto technických funkcích.

„Mobilní je ve skutečnosti uživatel, a ne jeho zařízení nebo nějaká aplikace.“
(Castledine, 2013)

Alternativou responzivního designu jsou mobilní verze webu. Ty jsou po technické stránce oddělené od „nemobilního webu“ a stahuje ze serveru jiný kód, než ten, který by se stáhnul pro desktopovou verzi, či responzivní verzi. Tento typ mobilního webu se typicky se umísťují na subdoménu *m.domena.cz*. Toto řešení se používá spíše u složitých webů či aplikací, které by bylo velice obtížné vytvořit responzivně (Budiu 2016).

3.4.1 Media Queries

Základním stavebním kamenem pro responzivní design jsou Media Queries, která se stala doporučeným standardem se v červnu 2012. Jedná se o CSS3 modul, který umožňuje používat podmínky, které mohou aplikovat různé styly v různých technických kontextech, například v určitém rozmezí šířky zobrazovacího zařízení (W3 Schools).

```
@media (max-width: 780px) and (min-width: 660px){  
  body{  
    background-color: red;  
  }  
}
```

Obrázek 14 - příklad použití media queries (Zdroj: autor)

Aby bylo vůbec možné web na mobilních zařízeních zobrazit korektně, je potřeba upravit viewport. Ten se dá jednoduše nastavit v meta tagu. Ten ve své podstatě nastaví reálnou šířku displeje, kterou si webové prohlížeče v mobilních zařízeních upravují podle sebe a také může oddalování a přibližování na mobilních telefonech (W3 Schools).

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

Obrázek 15 - příklad použití viewportu (Zdroj: autor)

3.4.2 Responzivní webdesign z hlediska SEO

Responzivní design se stává důležitým i z hlediska optimalizace ve vyhledávačích. Reakce proto například internetového giganta Googlu je určitě při nejmenším k věci a pokud má web cílit na uživatele, respektive má být web snadno dohledatelný, především pokud uživatel používá mobilní zařízení, tak je potřeba i ve vyhledávačích tento obsah rozlišovat.

Google pro indexování webu používá svého Googlebota. Ten může udělit webu štítek „Mobile-friendly“, přičemž jej i upřednostní ve vyhledávači. Aby web tento štítek dostal, je zapotřebí udělat určitá opatření:

- Web nesmí obsahovat software, který není podporován na mobilních zařízeních, například Flash
- Použitý text musí být k přečtení, aniž by se stránka musela přibližovat
- Obsah musí být zobrazen na šíři zařízení, tudíž uživatelé nemusejí posouvat web horizontálně, či jej přibližovat
- Umístěné odkazy jsou od sebe dostatečně daleko vzdáleny, takže jej lze snadno rozlišit poklepním

Google umožňuje také vývojářům použít jejich analytický program, který web prověří a vyhodnotí, zda je „Mobile-friendly“. Pokud ano, je možné odeslat žádost k udělení tohoto štítku ve vyhledávači (Google Webmaster Central Blog, 2014).

Example

<https://www.example.com/>

Mobile-friendly - This is an example of a website that is well-designed for mobile devices.

Obrázek 16 - příklad mobile friendly webu ve vyhledávači Google (Zdroj: Google Webmaster Central Blog)

3.4.3 Technické řešení – Bootstrap

Bootstrap je nejpopulárnější HTML, CSS a Javascriptový Framework pro tvorbu responzivních webových stránek či webových aplikací a je kompletně zdarma. Tato sada nástrojů obsahuje šablony, styly, typografii, které usnadní vývoj responzivních webů či jejich optimalizaci. Bootstrap pro jeho univerzálnost je vhodný jak pro vývoj front-endových tak i back-endových aplikací (Bootstrap).

Použití Bootstrapu je jednoduché. Jelikož jsou jeho součástí CSS soubory a Javascript soubory, stačí je importovat:

```
<script src="http://code.jquery.com/jquery-latest.js"></script>
<link href="css/bootstrap.min.css" rel="stylesheet" media="screen">
<script src="js/bootstrap.min.js"></script>
```

Obrázek 17 - připojení Bootstrap knihoven do HTML (Zdroj: autor)

Důležitou součástí je také přidání knihovny JQuery, bez které Bootstrap ztrácí na funkčnosti.



Obrázek 18 - seznam důležitých souborů obsahující Bootstrap po stažení (Zdroj: autor)

Nejdůležitější je naučit se pracovat s tzv. kontejnery. Bootstrap umožňuje obsah zabalit do dvou kontejnerů. Nejčastěji to bývají elementy označující sekci, nebo blok dané stránky. První možnost je přidání třídy `.container`, která elementu dá fixní šířku, v základním nastavení to je 1160px. Druhou možností je přidání třídy `.container-fluid`, která elementu dá 100% šířku (W3 Schools).

Další klíčovou funkcí je tzv. Bootstrap Grid Systém, který umožňuje na stránku přidat až 12 sloupců, které se podle daných tříd dokáží přizpůsobit, tudíž nastavit a upravit svojí šířku dle velikosti zobrazovacího zařízení dle tzv. „breakpointů“ a vypíná se obtékání sloupce. Ty se aktivují při dosažení daného rozlišení u zařízení. Sloupec se značí třídou `.col-(velikost)-(číslo)`, přičemž velikosti jsou možné čtyři (Bootstrap, W3 Schools):

- **xs** – je určen pro mobilní telefony, kde se neruší obtékání
- **sm** – je určen pro tablety, kde se při rozlišení do 768px ruší obtékání sloupce
- **md** – je určen desktopové zařízení, kde se při rozlišení do 992px ruší obtékání sloupce
- **lg** – je určen větší desktopové zařízení, kde se při rozlišení do 1200px ruší obtékání sloupce

Pro správnou funkčnost grid systému je potřeba se držet základních pravidel:

- Sloupce pro daný řádek by měly být v elementu s třídou `.row`
- Elementy s třídou `.row` by měli být v kontejnerech `.container` nebo `.container-fluid`
- Součet čísel ve sloupcích pro daný řádek by neměl přesáhnout 12

```

<div class="container">
  <div class="row">
    <div class="col-sm-4"></div>
    <div class="col-sm-4"></div>
    <div class="col-sm-4"></div>
  </div>
  <div class="row">
    <div class="col-lg-6"></div>
    <div class="col-lg-6"></div>
  </div>
</div>

```

Obrázek 19 - příklad použití grid systému (Zdroj: autor)

Složka bootstrapu má také adresář s názvem fonts a obsahuje soubory vektorových počítačových písmen s názvem Glyphicons. Ty fungují jako sada vektorových ikon, které lze velice snadno použít a vyhnout se tak složitému formátování velikostně odlišných ikon, které by měli být do webu použity. Sada obsahuje 260 ikon, které se velikostně mohou nastavovat stejně, jako se nastavuje font, tudíž v CSS pomocí vlastnosti `font-size`. V HTML dokumentu se následně musí vytvořit párový tag `` s dvěma třídami `glyphicon glyphicon-název_ikony` (W3 Schools).

```

<span class="glyphicon glyphicon-camera"></span>

```

Obrázek 20 - použití Bootstrap glyphicon (Zdroj: autor)

3.5 E-learning

E-learning je možné definovat jako instrukce, či kurzy dodávané na digitálních zařízeních, jako je například počítač, či mobilní zařízení určené k podpoře učení, získávání nových znalostí. Právě kurzy obsahují dvě důležité části: obsah informací a technické metody, které pomůžou lidem naučit se právě obsah informací. V e-learningu jsou důležité body a funkce pro dané kurzy (Clark, 2003):

- Měl by být dán způsob dodání kurzu k uživateli. Například: kapacitní paměťová média, servery, nebo internet či intranet
- Obsah kurzu by měl být relevantní pro daný cíl učení

- K zobrazení obsahu informací by měly být použity prvky, jako jsou slova a obrázky
- Použití technických metod jako jsou různé formy testů, příklady z praxe a zpětná vazba (jak k daným testům, tak k uživateli samotnému)
- E-learningu může být uživatel veden instruktorem (synchronní e-learning), nebo e-learning navržen pro individuální studium (asynchronní e-learning)

3.5.1 Online vzdělávací technologie

V dnešní moderní době je důležitá typologie online nástrojů. Jako nástroj lze označit webovou aplikaci, pro kterou je klíčové jedno základní použití, nebo několik úzce zaměřených funkcionalit (Zounek, 2012).

Takováto e-learningová webová aplikace může být zaměřená na přívětivý design, snadno použitelná a responzivní. Tematicky zaměřená na vytváření online testů, kvízových otázek, následně by je uživatelé mohli vyplňovat, aplikace by vyhodnocovala jejich úspěšnost, nabízela statistiky a možnost zdokonalovat se. Okruh využití by mohl být široký a spektrum uživatelů by byl od základní po vysoké školy. Ti by si mohli procvičovat testy z mnoha oborů.

3.5.2 Potencional

Ze všech výše uvedených informací bylo zjištěno, že existuje obrovský potenciál v realizaci E-learningové aplikace, nehledě na platformu. Především díky vyspělým webovým technologiím.

4 Vlastní práce

V této části práce bude, na základě syntézy zjištěných poznatků, vytvořena responzivní webová aplikace pro oblast e-learningu, která bude zaměřena na vytváření a správu online testů. Projekt ponese název „RespoELearn“.



Obrázek 21 - logo aplikace (Zdroj: autor)

Část webu, která je přístupná běžnému návštěvníkovi, bude vytvořena pomocí technologií HTML 5 a CSS 3 společně se sadou nástrojů Bootstrap ve verzi 3.3.7. Dynamickou interakci s těmito prvky bude obstarávat javascript, konkrétně knihovna JQuery ve verzi 1.10.1. Skripty, které zpracovává vzdálený server, budou programovány pomocí jazyka PHP ve verzi 5.6 a řízení báze dat bude obstarávat databázový systém MySQL ve verzi 5.5.

Neregistrovaní návštěvníci mohou vyplňovat vytvořené testy jinými uživateli. Testy jim budou vyhodnoceny, avšak jejich dosavadní úspěšnost nebude uložena. Přihlášení uživatelé mohou své testy spravovat, či vytvářet nové. Následně je jejich úspěšnost zaznamenávána a mohou ji kdykoliv zobrazit.

4.1 Vytvoření základní struktury

Pro snazší možnost modifikace každé podstránky a pro výběr jednotlivých modulů, je vytvořena základní struktura, která obsahuje základní komponenty pro funkčnost aplikace. Tato struktura je následně použita v každé podstránce, s rozdílem obsahu podstránky.

```
<?php
    require_once("core/db.php");
    require_once("core/functions.php");
    require_once("core/global.php");
    $title = "Titulek";
    include("include/head.php");
    include("include/header.php");
?>

... obsah ...
```

```
<?php
    include("include/footer.php");
?>
```

Obrázek 22 - základní struktura aplikace (Zdroj: autor)

4.1.1 Přístup k databázi a základní proměnné

Přístup k databázi zajišťuje v jádru aplikace jediný soubor `/core/db.php`, díky kterému se aplikace dokáže spojit s databázovým systémem MySQL pomocí přístupových údajů: Databázový server; Jméno databáze; Uživatelské jméno; Heslo; (PHP.net).

V dalším důležitém souboru `/core/global.php`, se nacházejí důležité informace týkající se aplikace. Nejprve se volá funkce pro ověření přihlášení uživatele a následně jsou uvedeny informace pro cestu aplikace a její aktuální verze.

4.2 Uživatel v aplikaci

Každý uživatel, který navštíví prostředí webové aplikace, bude mít možnost se zaregistrovat, popřípadě přihlásit. Tím se mu zpřístupní veškeré funkce.

4.2.1 Registrace

Při registraci se nejprve ověřuje, zda vůbec byl zaslán požadavek na registraci a odeslán registrační formulář. Informace, které se přenáší k vyhodnocení a kontrolují, jsou:

- E-mail – kontrola validní e-mailové adresy
- Heslo – kontrola minimálně 6 znaků
- Heslo znovu – kontrola shody s řetězcem zadaným v hesle

V poslední řadě se kontroluje, zda již na tuto e-mailovou adresu nebyl účet registrován. V případě splnění všech podmínek je uživatel zaregistrován a automaticky přihlášen.

```
$result = mysqli_query($mysqli,"SELECT id FROM el_users WHERE email LIKE '$email'");
```

Obrázek 23 - výběr registrovaného e-mailu (Zdroj: autor)

4.2.2 Přihlášení

Stejně jako při registraci se nejprve ověřuje, zda byl zaslán požadavek na přihlášení a odeslán přihlašovací formulář. Následně se pouze ověřuje správnost zaslanych informací,

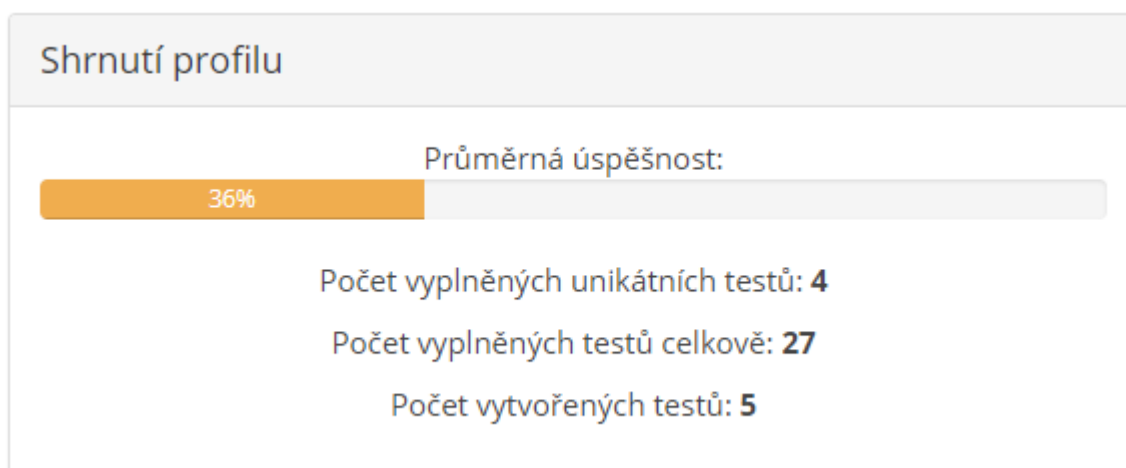
kterými jsou pouze e-mail a heslo. V případě že se k dané e-mailové adrese váže zadané heslo, bude uživatel přihlášen.

```
$result = mysqli_query($mysqli,"SELECT id FROM el_users WHERE email LIKE '$email' AND password = '$password'");
```

Obrázek 24 - kontrola přihlášení (Zdroj: autor)

4.2.3 Uživatelský profil

Uživatelský profil nabídne uživateli vidět jeho úspěšnost v počtu celkově vytvořených testů. Ta se zobrazí pomocí grafického ukazatele aplikovaný Bootstrap CSS třídou `progress-bar`. V uživatelském profilu má uživatel také možnost si změnit své aktuální heslo.



Obrázek 25 - shrnutí profilu v aplikaci (zdroj: autor)

4.2.4 Správa testů

Aplikace přihlášenému uživateli nabídne možnost spravovat testy. Základními třemi akcemi jsou vytvoření, úprava a odstranění. Každý test je unikátní, tudíž k němu uživatel také dostane URL odkaz, který v případě, že test není veřejný, může sdílet a zasílat individuálně. Každý test obsahuje určitá nastavení, stejně tak jednotlivé otázky a odpovědi.

Aplikace umožňuje následující nastavení testu:

- Název testu
- Popis testu
- Veřejnost testu (Veřejný / Neveřejný) – nastaví, zda bude test viditelný v seznamu testů, či jeho přístup bude pouze pomocí odkazu
- Pořadí otázek (Dle otázek / Náhodné)

- Časový limit testu (Bez limitu / Časový limit – v rádech minut)

Název testu:

Popis testu:

Veřejnost testu:

Pořadí otázek:

Časový limit:

Obrázek 26 - nastavení vytvářeného testu (zdroj: autor)

Nastavení otázek a odpovědí:

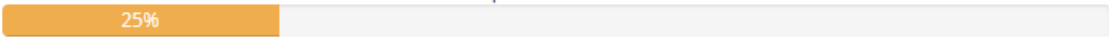
- Znění otázky a odpovědí
- Typ odpovědi (Textová odpověď / Výběr odpovědi z položek / Výběr odpovědi z listu / Výběr více správných odpovědí z položek)
- Možnost posunu pořadí otázek (ikona posuvníku, či klepnutí na šipky)
- Možnost odstranění otázky či odpovědi (ikona koše)

<input checked="" type="checkbox"/>	1991	<input type="button" value="🗑️"/>
<input checked="" type="checkbox"/>	1992	<input type="button" value="🗑️"/>
<input checked="" type="checkbox"/>	1993	<input type="button" value="🗑️"/>
<input checked="" type="checkbox"/>	1994	<input type="button" value="🗑️"/>

Obrázek 27 - nastavení otázky při vytváření testu (zdroj: autor)

4.2.5 Vyplňování testů

Podstránka vytvořeného testu obsahuje informace, než uživatel test spustí. Těmi informacemi je veškeré nastavení testu společně s datem vytvoření či upravení testu a úspěšnosti vyplnění.

Název testu: Test ČR
Popis testu: otestujte Vaše znalosti o České republice
Upraven: 20.02.2018
Časový limit testu: bez limitu
Počet otázek: 4
Úspěšnost testu: 
Test byl vyplněn: 4x
▶ Spustit test

Obrázek 28 - informace o testu (zdroj: autor)

V případě spuštění testu je uživateli vypsán test, společně s otázkami a možností je vyplnit a test odeslat – nechat jej vyhodnotit.

Jaké je hlavní město ČR?

Zde napište odpověď...

V jakém roce vznikla ČR?

1991
 1992
 1993
 1994

Vyberte města která mají více jak 100 000 obyvatel:

Praha
 Mělník
 Brno
 Pardubice
 Plzeň
 České Budějovice

V jakém městě se stéká Labe s Vltavou?

Vyberte odpověď z listu

Obrázek 29 - výpis testu a otázek v aplikaci (Zdroj: autor)

4.2.5.1 Vyhodnocení a revize testů

Po odeslání formuláře při vyplnění testu se data zašlou na server, kde se zpracují a porovnají se správnými odpověďmi zadanými uživatelem, co jej vytvořil (PHP.net). Za každou otázku, která je vyplněna korektně, se získá jeden bod. Uživatel je také informován o správné odpovědi a dle počtu bodů společně s počtem otázek je vygenerována procentuální úspěšnost, která je taktéž uložena v databázi

ZÍSKÁNO **3** z 4 možných bodů

75%

Obrázek 30 - vyhodnocení testu (zdroj: autor)

V jakém roce vznikla ČR?

1991

1992

1993 ✓

1994

✓ Získáno: 1b

Obrázek 31 - vyhodnocení otázky (zdroj: autor)

4.3 Responzivní design

Pro aplikaci responzivního designu na projekt se používá sada nástrojů Bootstrap. Zde jsou nejdůležitější částí kaskádové styly, přičemž doplňující styly pro dané elementy jsou nakódované zvlášť.

Po grafické stránce je aplikace členěna do horizontálních úseků. Šířku těchto úseků určuje třída `.container`, která činí maximálně 1170px. Následně, pokud jsou úseky děleny na více dalších částí, je používán „grid system“. Tyto části jsou odlišované různými typy tříd a jejich velikostí (W3 Schools).

```
<div id="header">
  <div class="container">
    <div class="row">
      <div class="col-sm-5">
      </div>
      <div class="col-sm-7 header-right">
      </div>
    </div>
  </div>
</div>
```

Obrázek 32 - příklad použití grid system v záhlaví aplikace (Zdroj: autor)

O doplnění funkčnosti responzivního designu se starají doplňující styly, které se dokáží měnit dle šířky rozlišení – media queries. Aktuálně v aplikaci je nejvíce využití aplikace stylů pro maximální šířku 768px. Ta je využita a uplatněna především v kombinaci

Bootstrapových tříd `.col-sm`, což je poslední „break point“ (W3 Schools). Dále funkcionalitu responzivního designu nijak nenarušuje ani neupravuje Javascript, ani server nezasílá žádné požadavky při zjištění přístupu například z mobilního typu zařízení.

```
@media (max-width: 768px) {  
    ... obsah ...  
}
```

Obrázek 33 - použití media queries v aplikaci (Zdroj: autor)

4.3.1 Použité komponenty sady Bootstrap

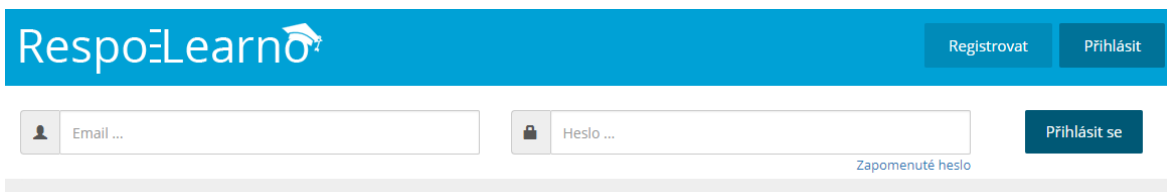
Pokud by se provedla dekompozice aplikace a zjišťovalo by se, jaké konkrétní části ze sady Bootstrap se použily, seznam těch nejdůležitější by byl následující (Bootstrap):

- Grid system – používá se při základní optimalizaci a přizpůsobování na daná zařízení
- Forms – používá se zejména při tvorbě formulářů a jejich úpravě
- Tables – používá se při aplikaci tabulek
- Buttons – používá se při používání účelově odlišných typů tlačítek
- Alerts – používá se při upozorňování uživatele o provedené akci
- Progress bars – používá se při zobrazování statusu úspěšnosti testu či uživatele
- Glyphicons – používá se při aplikaci vektorových ikon
- Dropdowns – používá se při zobrazení rozšiřujících možností

4.4 Javascriptové funkce

Veškeré javascriptové úkony jsou vytvořeny pomocí JQuery. Konkrétněji pokud se popíší skripty, které jsou aplikovány a importovány do všech podstránek tak se jedná o knihovnu `/js/functions.js`. Většina těchto funkcí je aplikovaná stylem provedení nějaké akce uživatelem – například konkrétně `$(document).on('click')` a tím se funkce zavolá. V aplikaci největší počet těchto funkcí obsahuje oblast přihlašování a registrace, která má za efekt provedení dynamické akce, odeslání a vyhodnocení požadavku na server bez znovu načtení stránky – konkrétně AJAX (JQuery - Javascript library).

První funkce ovládá pohyb – zobrazení a skrytí formulářů pro přihlášení či registraci, při klepnutí na jedno z tlačítek. Ověřuje se, zda některé z těchto obsahů již není zobrazeno a korektně uživateli zobrazí požadovaný obsah.



Obrázek 34 - přihlášení v aplikaci (Zdroj: autor)

První funkce ovládá pohyb – zobrazení a skrytí formulářů pro přihlášení či registraci, při klepnutí na jedno z tlačítek. Ověřuje se, zda některé z těchto obsahů již není zobrazeno a korektně uživateli zobrazí požadovaný obsah pomocí funkce `slideDown()` (jQuery - Javascript library).

U odesílání požadavků při registraci či přihlášení se ověřují položky ve formulářích kombinací na serveru a přímo u klienta. Respektive při požadavku na akci přihlášení/registrace se u serveru ověřují korektnost dat a ve formě čísel se vrací zpět k uživateli. Javascriptové podmínky tato čísla odliší, zjistí se, co znamenají, a uživateli se dá vědět, jaká akce proběhla. Pokud zpráva požadavku je číslo 0, akce provedla a uživatel je přihlášen, či registrován. Potencionální chybové hlášky budou uživateli sděleny pomocí vytvořené funkce `doErr()`.

```
$.ajax({
  type: "POST",
  url: "login.php",
  data: 'email='+email+'&password='+password,
  cache: false,
  beforeSend: function(){ $("#logme").val('Přihlašuji...');},
  success: function(data){
```

Obrázek 35 - předávání informací při přihlášení: AJAX (zdroj: Autor)

4.4.1 Dynamická interakce s prvky pomocí JQuery

Největší interakce a spolupráce JQuery s HTML prvky – společně s jejich atributy a obsahem, především u Bootstrapových připravených prvků je u vytváření/úpravy testů – konkrétněji správa otázek a odpovědí. Dynamické akce řídí funkce, které se nachází v souboru `/js/manage.js`. Funkce ověřují platnosti akcí, správnosti provedených akcí či manipulaci s otázkami a odpověďmi – přidání, odstranění, posun, označení správné odpovědi. Připravené html prvky, jedná se o prázdné otázky, odpovědi všech druhů, se

nacházejí v souboru `/include/question_elements.php` a mohou být přeneseny do aplikace pomocí funkce `load()`.

4.5 Výsledná aplikace a její struktura

Výsledkem je validní responzivní webová aplikace za použití sad nástrojů Bootstrap. Hlavní strana zobrazí uživateli především informace o aplikaci. Nabídne mu možnost se ihned registrovat, přihlásit se, či začít s výběrem testu, který je možné vyplnit.



Obrázek 36 - zobrazení hlavní strany aplikace na třech zařízeních (zdroj: autor)

Seznam veřejných testů nabídne návštěvníkovi zobrazit si větší podrobnosti o daném testu. Seznam nadále zobrazí název a průměrnou úspěšnost spolu s možností filtrovat testy v reálném čase bez načtení prohlížeče pomocí javascriptu díky určité části textu v testu. Pokud by tak test společně v názvu obsahoval například kategorii, uživatel by poté mohl snadno vyhledat určitý druh testů. Seznam následně také vypíše kolik celkem testů je v seznamu zobrazeno.

Vyhledat ...		
Název testu	Průměrná úspěšnost	Akce
Test ČR	35%	➤ Zobrazit
Testovací test	31%	➤ Zobrazit
Základy klinických oborů	67%	➤ Zobrazit

Celkem: 3

Obrázek 37 - seznam veřejných testů (zdroj: autor)

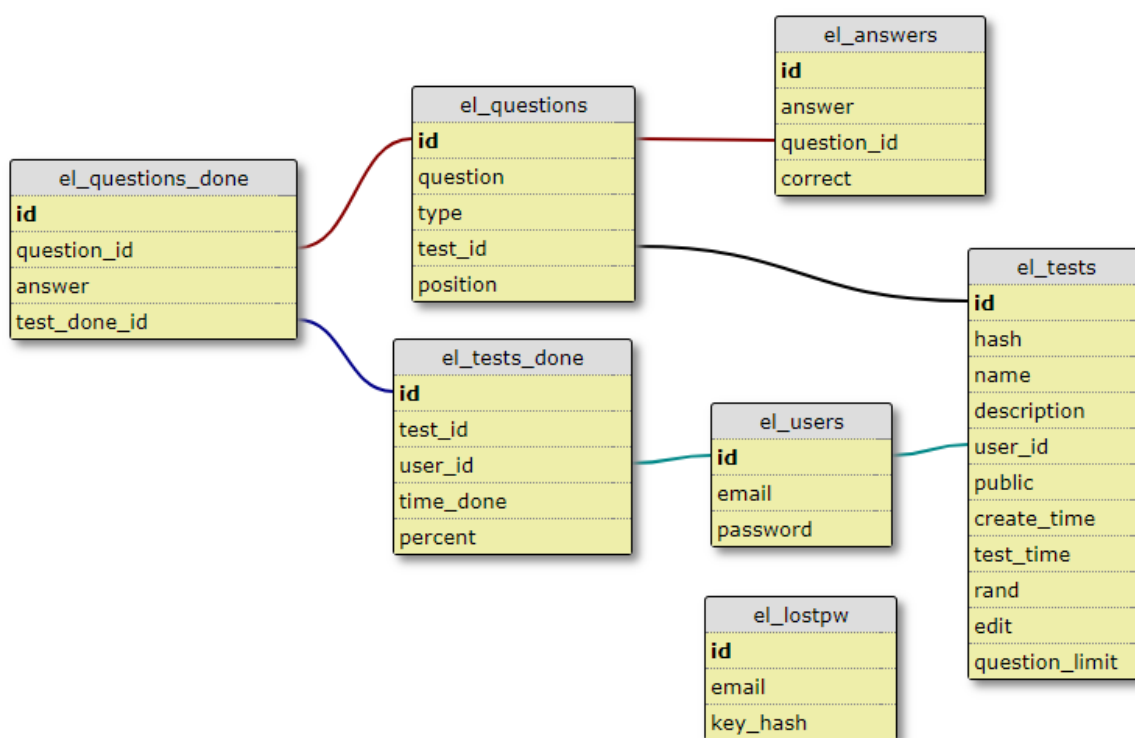
4.5.1 Souborová struktura

Aplikace obsahuje soubory, které jsou uspořádány hierarchicky. V následujícím listu jsou abecedně popsány soubory a adresáře, které aplikaci tvoří:

- `/core/` – adresář obsahující jádro systému, kterými jsou důležité moduly
 - `/core/db.php` – připojení k databázi
 - `/core/functions.php` – vytvořené funkce
 - `/core/global.php` – globální proměnné
 - `/core/manage.php` – vytvořené PHP funkce používající se při správě testů
- `/css/` – adresář obsahující kaskádové styly
 - `/css/style.css` – vytvořené styly
 - `/css/bootstrap.min.css` – knihovna stylů pro sadu Bootstrap
- `/fonts/` – adresář obsahující soubory potřebné pro používání Bootstrap modulu Glyphicons
- `/images/` – adresář obsahující veškeré obrázky, které aplikace používá
- `/include/` – adresář obsahující soubory, které se používají jako určité součásti aplikace
 - `/include/footer.php` – patička pro web, která může obsahovat dodatečné skripty
 - `/include/head.php` – obsahuje hlavičku HTML souboru
 - `/include/header.php` – záhlaví webu
 - `/include/question_elements.php` – všechny typy otázek v HTML formátu
- `/js/` – adresář obsahující javascriptové soubory
 - `/js/bootstrap.min.js` – knihovna javascriptových funkcí pro sadu Bootstrap
 - `/js/functions.js` – vytvořené javascriptové funkce
 - `/js/jquery-1.10.1.min.js` – javascriptová knihovna JQuery
 - `/js/manage.js` – vytvořené javascriptové funkce používající se při správě testů
 - `/js/test_time.js` – javascriptové funkce používající se při testu na čas
- `/add-test.php` – vytváření testů
- `/correct-test.php` – kontrola a revize testů
- `/edit-test.php` – šablona pro úpravu testů
- `/edit-test-content.php` – úprava konkrétního testu

- **/edit-test-list.php** – výpis testů pro úpravu
- **/index.php** – hlavní strana
- **/login.php** – přihlášení uživatele
- **/logout.php** – odhlášení uživatele
- **/lost-pw.php** – ztracené heslo uživatele
- **/profile.php** – profil uživatele
- **/register.php** – registrace uživatele
- **/test.php** – vyplnění konkrétního testu
- **/tests.php** – výpis testů pro vyplnění

4.5.2 Databázová struktura



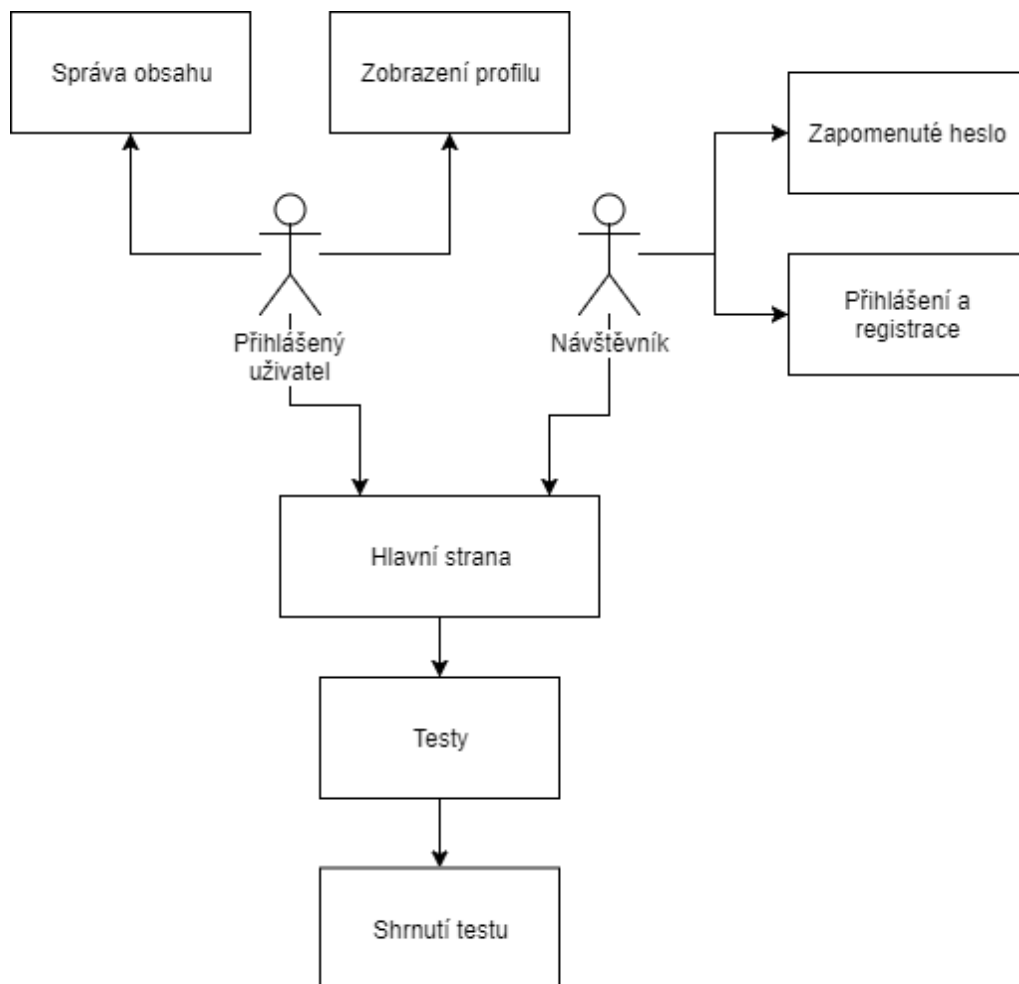
Obrázek 38 - databázová struktura aplikace

Databázová struktura je tvořena šesti tabulkami, které mají určité vazby. Jedná se o relačně databázový model. V následujícím listu se nachází jejich popis:

- **el_tests** – obsahuje informace a nastavení testů
 - každý test je jednoznačně identifikován, na základě tohoto identifikačního klíče je vygenerován unikátní řetězec znaků
- **el_questions** – obsahuje otázky a jejich informace společně s typem otázky a její pozicí
- **el_answers** – obsahuje odpovědi k otázkám a identifikace správné odpovědi
- **el_users** – obsahuje informace o registrovaných uživateli
- **el_lostpw** – obsahuje informace v případě požadavku o obnovení hesla

- **el_tests_done** – obsahuje informace o vyhodnocených testech
- **el_questions_done** – obsahuje informace o vyhodnocených otázkách

4.5.3 Mapa stránek



Obrázek 39 - Mapa stránek a přístupu uživatele

Z mapy stránek a jednotlivých přístupů je patrné, že není kladen rozdíl na návštěvníka a přihlášeného uživatele při vyplňování testů a jejich zpětnou revizi. Přihlášený uživatel má přístup ke správě obsahu – vytváření a úprava vlastních testů a zobrazení profilu.

5 Zhodnocení výsledků

5.1 Analýza tvorby

Při vývoji pomocí Bootstrap není potřeba použíté prvky, především jejich stylizaci popisovat a vést rozsáhlou dokumentaci, protože samotná sada již dokumentaci obsahuje. Sada taktéž nabízí rychlejší a pohodlnější vývoj.

Základní způsob, jakým bude analýza provedena, je porovnání velikostí aplikace a následně také potenciální vliv způsobu vývoje aplikace na koncového uživatele použitím rozdílných vývojářských technologií. Variabilita vývoje je velice vysoká, běžný uživatel neklade důraz a nebere v potaz, jaké technologie jsou použity. Pokud to však může mít vliv na uživatele, jediným pocíitelným a měřitelným faktorem je velikost.

5.1.1 Velikost aplikace

Aplikace byla tvořena v kombinaci se sadou nástrojů Bootstrap a jak již bylo zmíněno, všechny části z této sady nebyly reálně využity. Použití veškerých reálně využitých částí lze v praxi simulovat výběrem použitého obsahu (třídy, funkce, ...) z frameworku, čímž se funkčnost aplikace nezmění. Následně je možné porovnat velikosti aplikace vyvíjené pomocí Bootstrap a vývoje bez frameworku.

	Bootstrap	Bez frameworku
Velikost CSS	9 kB + 118 kB (style.css) + (bootstrap.min.css)	24 kB (style.css)
Velikost JS knihovny jQuery (jquery- 1.10.1.min.js)	90,8 kB	90,8 kB
Velikost JS knihovny pro funkce stránky	36,1 kB + 3,7 kB (bootstrap.min.js) + (functions.js)	10,5 kB (functions.js)
Změny v html kódu	Importování Bootstrap knihoven - zanedbatelné	Žádné
Obrázkové změny	Žádné	Žádné
Celková velikost importovaných částí	257,6 kB	125,3 kB

Tabulka 1 - velikost aplikace (zdroj: autor)

Z výsledné tabulky lze vyčíst, že velikost aplikace vyvíjená pomocí Bootstrap je více než dvojnásobná. Největší podíl na tom má knihovna kaskádových stylů, kde rozdíl velikostí je 103 kB. Následně rozdíl velikosti javascriptových funkcí je 29,3 kB. V ostatních porovnání nejsou žádné změny, nebo jsou zanedbatelné. V souborech nebyla provedena žádná komprese.

5.1.2 Vliv vývoje na koncového uživatele

Odlišné velikosti, které jsou více než dvojnásobné, mohou zavádět k pomyšlení na to, že se web bude načítat dvakrát déle. Avšak při takovéto velikosti a současných rychlostí internetu by tomu naopak být nemělo. Důležitými prvky, které hrají roli při načítání webu, jsou však nejen rychlost internetu, ale také třeba odezva, či výkon zařízení. Pomocí Chrome DevTools, což jsou vývojářské nástroje, které Google implementoval do svého internetového prohlížeče Chrome, bylo provedeno několik testů, díky kterým se dají simulovat a následně vyhodnotit výsledky vlivu vývoje na koncového uživatele, který web navštíví. Testy byly prováděny v prohlížeči Google Chrome ve verzi 63 (Chrome DevTools Overview).

Vzhledem k různorodosti zařízení a internetového připojení, byla pro test použita tři různá zařízení a čtyři různě kvalitní internetová připojení. Druhy testů následně byly dva, první simulovalo běžné načtení uložené stránky – pokud uživatel již web někdy v minulosti navštívil a jeho zařízení si uložilo data. Druhý test simuloval načtení bez právě uložených dat, což v praxi znamená prvotní načtení webu.

	Bootstrap	Bez frameworku	Rozdíl
T1 A	0,33s	0,31s	0,02s
T1 B	0,67s	0,58s	0,09s (15%)
T2 A	0,43s	0,39s	0,04s
T2 B	0,92s	0,77s	0,15s (19%)
T3 A	1,70s	1,65s	0,05s
T3 B	2,96s	2,50s	0,46s (18%)
T4 A	3,17s	3,10s	0,07s
T4 B	11,8s	7,47s	4,33s (58%)

Tabulka 2 - měření načtení webu na různých zařízeních s různými typy připojení (zdroj: autor)

Popis tabulky:

- **T1** = test na stabilním rychlém s ultra nízkou odezvou připojení a výkonném PC (TTFB: 8ms)
- **T2** = test na stabilním rychlém s nízkou odezvou připojení a kancelářském PC (TTFB: 10ms)
- **T3** = test na rychlém 3G připojení a mobilním telefonu střední třídy (TTFB: 600ms)
- **T4** = test na pomalém 3G připojení a mobilním telefonu střední třídy (TTFB: 2000ms)
- **A** = Běžné načtení uložené stránky
- **B** = Načtení bez mezipaměti a cache.

Celkem bylo prováděno 5 testů u každého typu testu. Jejich průměr byl zapsán do tabulky. Vývojářské nástroje umožnily vzít v potaz také několik druhů internetových připojení, s rozdílem rychlosti a TTFB. *Time to First Byte (TTFB) je čas od požadavku internetového prohlížeče, který pošle na server, do doby, než obdrží první bit zpátky* (Seopedia).

Testováním bylo zjištěno, že rozdíl na zařízeních s kvalitním internetovým připojením není tak znatelný a uživatel jej nepocítí. Naproti tomu u datového mobilního připojení může nekvalitní signál zapříčinit prvotní načtení webu se zpožděním jednotek sekund u aplikace vyvíjené pomocí sady Bootstrap. Testy odhalily rozdíl téměř 4,5 sekund u pomalého 3G připojení a téměř půlsekundový rozdíl u rychlého 3G připojení u prvotního načtení. Avšak u načtení s uloženými daty v mezipaměti jsou rozdíly zanedbatelné.

6 Závěr

Na základě syntézy zjištěných poznatků získaných studiem odborných informačních zdrojů a zpracováním literární rešerše týkající se vytvoření responzivní webové aplikace, bylo možné získat nezbytně nutný teoretický základ pro přístup do problematiky a následně byla vytvořena aplikace pro oblast e-learningu nesoucí název „RespoELearn“. Popsán byl její vývoj a poté se provedla analýza tvorby aplikace. Teoretická část popsala webové aplikace a technologie, díky kterým mohou být realizovány. Konkrétně pro klientskou část se práce zabývala technologiemi HTML, CSS a javascript s knihovnou JQuery. Pro serverovou část byly popsány technologie PHP a databáze MySQL. Byl také vysvětlen responzivní design a jeho řešení pomocí Bootstrap, což je jedna z nejrozšířenějších sad nástrojů pro tvorbu právě responzivního designu.

Z vývoje aplikace vyplynulo několik závěrů a vlastností pro danou aplikaci. Sada Bootstrap nabízí použití prvků či funkcí, které zjednoduší a urychlí psaní kódu. Nabídne vývojářům perspektivní možnost vývoje bez nutnosti znát konkrétní řešení vytvářené aplikace. Tyto veškeré vlastnosti jsou na úkor velikosti importovaných částí. Testy ukázaly, že reálné využití veškerých prvků, které sada obsahuje je malá a důsledek toho je navýšení velikosti aplikace o více než dvojnásobek. Pokud by uživatel používal aplikaci na rychlém internetovém připojení, toto navýšení by prakticky nezaznamenal. V opačném případě, při použití nekvalitního internetového připojení, by uživatel mohl čekat na první načtení o více než 50% času déle, což se v testech prokázalo o téměř 4,5 sekundy. Toto nekvalitní internetové připojení může, a jak to testy simulovaly, být 3G připojení, které je používáno především na mobilních zařízeních, kterým je určitá část aplikace věnována – responzivním designem.

Javascriptová knihovna JQuery se v kombinaci se sadou Bootstrap jeví jako vhodné řešení při vývoji webových aplikací. Především pro dynamickou práci s HTML elementy, konkrétně při práci s CSS třídami. Používání a editace či přesouvání atributů mezi prvky je v tomto směru snadnou záležitostí a díky předdefinovaným částem v sadě se takto může jednoduše měnit jejich funkce. Mnoho dalších akcí, které by bez knihovny musely být napsány ručně, je možné použít ve funkci s přidáním určitých parametrů na pár řádků. Pro funkčnost Bootstrap javascriptových funkcí je JQuery knihovna nezbytná. Díky usnadnění práce při tvorbě dynamického obsahu, zejména pro klientskou část, je JQuery dobrým

pomocníkem a při vývoji webových aplikací dokáže usnadnit plno práce, stejně jako sada Bootstrap.

Všechny zmíněné a použité technologie umožňují vytvoření e-learningové aplikace a některé prvky ze sady Bootstrap jsou přímo nabízející se při uplatnění v programu. Jako jednoznačný přínos práce může být samotná webová aplikace, která je velice perspektivní a nabídne uživateli zdokonalovat jeho znalosti. Projekt by mohl v budoucnu obsahovat další prvky a funkce, které by pouze podtrhly uplatnění a umožnily například uživateli rozšířit nastavení testu, či přidat jiné typy otázek. Dále může řešení sloužit jako vzor a být použito v jiných institucích, kde je platné využití e-learningu. Práce také objasnila problematiku při vývoji pomocí sad Bootstrap a poukázala na jeho nevýhody, které byly v praxi otestovány.

7 Seznam použitých zdrojů

BITTNER, Jan, 2015. *Mobile first responzivní design* [online]. (Aktualizováno). [cit. 2017-10-22]. Dostupné z: <https://www.itnetwork.cz/html-css/responzivni-webdesign/mobile-first-responzivni-design>

BUDIU, Raluca, 2016. *Mobile Websites: Mobile-Dedicated, Responsive, Adaptive, or Desktop Site?* [online]. [cit. 2017-10-22]. Dostupné z: <https://www.nngroup.com/articles/mobile-vs-responsive/>

Bootstrap [online], [cit. 2017-10-22]. Dostupné z: <http://getbootstrap.com/>

CASTLEDINE, Earle, Myles EFTOS a Max WHEELER, 2013. *Vytváříme mobilní web a aplikace pro chytré telefony a tablety*. Brno: Computer Press. ISBN 978-80-251-3763-5.

CASTRO, Elizabeth a Bruce HYSLOP, 2012. *HTML5 a CSS3: názorný průvodce tvorbou WWW stránek*. Brno: Computer Press. ISBN 978-80-251-3733-8.

CASTRO, Elizabeth. a Bruce. HYSLOP, c2012. *HTML5 and CSS3: visual quickstart guide*. 7th ed. Berkeley, CA: Peachpit Press. ISBN 978-0-321-71961-4.

CHASTIAN, Sue, 2017. *Advanced Illustration Programs* [online]. (Aktualizováno). [cit. 2017-10-22]. Dostupné z: <https://www.thoughtco.com/best-advance-illustration-programs-1702812>

CLARK, Ruth Colvin. a Richard E. MAYER, c2003. *E-Learning and the science of instruction: proven guidelines for consumers and designers of multimedia learning*. San Francisco, CA: Jossey-Bass/Pfeiffer. ISBN 07-879-6051-9.

Chrome DevTools Overview, Google Chrome [online], [cit. 2018-03-01]. Dostupné z: <https://developer.chrome.com/devtools>

Google Webmaster Central Blog, 2014. *Helping users find mobile-friendly pages* [online]. [cit. 2017-10-22]. Dostupné z: <https://webmasters.googleblog.com/2014/11/helping-users-find-mobile-friendly-pages.html>

HLAVÁČEK, Tomáš, 2015. *Nativní aplikaci, nebo mobilní web?* [online]. [cit. 2017-10-22]. Dostupné z: <http://www.web-integration.info/cs/blog/nativni-aplikaci-nebo-mobilni-web/>

Interval.cz, 2001. *CSS2 – selektory, pseudotřídy a pseudoelementy* [online]. [cit. 2017-10-22]. Dostupné z: <https://www.interval.cz/clanky/css2-selektory-pseudotridy-a-pseudoelementy/>

jQuery - Javascript library [online], [cit. 2017-10-22]. Dostupné z: <http://jquery.com/>

Jak psát web, *Začlenění skriptu do stránky* [online]. [cit. 2017-10-22]. Dostupné z: <https://www.jakpsatweb.cz/javascript/zacleneni.html>

Jak psát web, *Zápisy CSS* [online]. [cit. 2017-10-22]. Dostupné z: <http://polopate.jakpsatweb.cz/index.php?page=zapis-css>

KOSEK, Jiří, 1998. *HTML: tvorba dokonalých WWW stránek : podrobný průvodce*. Praha: Grada. Průvodce (Grada). ISBN 80-716-9608-0.

- KOSEK, Jiří, 1998. *PHP - tvorba interaktivních internetových aplikací: podrobný průvodce*. Praha: Grada. Průvodce (Grada). ISBN 80-716-9373-1.
- MICHL, Petr, 2012. *Je lepší nativní aplikace nebo mobilní web?* [online]. [cit. 2017-10-22]. Dostupné z: http://www.m-journal.cz/cs/internet/je-lepsi-nativni-aplikace-nebo-mobilni-web-__s281x9241.html
- MICHÁLEK, Martin, 2017. *Weby versus aplikace* [online]. (Aktualizováno). [cit. 2017-10-22]. Dostupné z: <https://www.vzhurudolu.cz/prirucka/weby-vs-aplikace>
- ManagementMania.com, *Webová aplikace (Web Application)* [online]. [cit. 2017-10-22]. Dostupné z: <https://managementmania.com/cs/webova-aplikace-web-application>
- PHP.net* [online], [cit. 2017-10-22]. Dostupné z: <http://php.net/>
- Rose India, *What is Web Graphics* [online]. [cit. 2017-10-22]. Dostupné z: <https://www.roseindia.net/services/webdesigning/corporatewebsitedesign/What-is-Web-Graphics.shtml>
- SALVET, Pavel, 2010. *Grafika pro webové stránky: nástroj zkázy?* [online]. [cit. 2017-10-22]. Dostupné z: <https://www.interval.cz/clanky/grafika-pro-webove-stranky-nastroj-zkazy/>
- Seopedia* [online], [cit. 2018-03-01]. Dostupné z: http://seopedia.cz/wiki/Time_to_First_Byte
- ŠKULTÉTY, Rastislav, 2004. *JavaScript: programujeme internetové aplikace*. 2. aktualiz. vyd. Brno: Computer Press. ISBN 80-251-0144-4.
- W3 Schools* [online], [cit. 2017-10-22]. Dostupné z: <https://www.w3schools.com/>
- W3 Schools, Bootstrap 3 Tutorial* [online]. [cit. 2017-10-22]. Dostupné z: <https://www.w3schools.com/bootstrap/>
- W3 Schools, CSS3 Media Queries* [online]. [cit. 2017-10-22]. Dostupné z: https://www.w3schools.com/css/css3_mediaqueries.asp
- W3 Schools, HTML Introduction* [online]. [cit. 2017-10-23]. Dostupné z: https://www.w3schools.com/html/html_intro.asp
- W3 Techs, Historical trends in the usage of server-side programming languages for websites* [online]. [cit. 2017-10-23]. Dostupné z: https://w3techs.com/technologies/history_overview/programming_language
- W3C* [online], [cit. 2017-10-22]. Dostupné z: <http://www.w3.org/>
- W3C, MEDIA QUERIES PUBLICATION HISTORY* [online]. [cit. 2017-10-22]. Dostupné z: <https://www.w3.org/standards/history/css3-mediaqueries>
- ZAJÍC, Petr, 2004. *Historie a budoucnost PHP* [online]. [cit. 2017-10-22]. Dostupné z: http://www.linuxsoft.cz/article.php?id_article=171
- ZOUNEK, Jiří a Petr SUDICKÝ, 2012. *E-learning: učení (se) s online technologiemi*. Praha: Wolters Kluwer Česká republika. ISBN 978-80-7357-903-6.

Přílohy

Obsah příloženého CD:

Na příloženém CD se nachází tyto adresáře a soubory:

- www (adresář se zdrojovým kódem celé aplikace)
- db.sql (SQL soubor obsahující databázovou strukturu a testovací data)