

Katedra informatiky
Přírodovědecká fakulta
Univerzita Palackého v Olomouci

BAKALÁŘSKÁ PRÁCE

Portál pro výuku webových technologií



2018

Vedoucí práce: Mgr. Petr Krajča,
Ph.D.

Stanislav Čech

Studijní obor: Aplikovaná informatika,
kombinovaná forma

Bibliografické údaje

Autor: Stanislav Čech
Název práce: Portál pro výuku webových technologií
Typ práce: bakalářská práce
Pracoviště: Katedra informatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci
Rok obhajoby: 2018
Studijní obor: Aplikovaná informatika, kombinovaná forma
Vedoucí práce: Mgr. Petr Krajča, Ph.D.
Počet stran: 41
Přílohy: 1 CD/DVD
Jazyk práce: český

Bibliographic info

Author: Stanislav Čech
Title: Website for learning web technologies
Thesis type: bachelor thesis
Department: Department of Computer Science, Faculty of Science, Palacký University Olomouc
Year of defense: 2018
Study field: Applied Computer Science, combined form
Supervisor: Mgr. Petr Krajča, Ph.D.
Page count: 41
Supplements: 1 CD/DVD
Thesis language: Czech

Anotace

Práce popisuje naprogramovanou aplikaci pro podporu výuky webových technologií. Hlavním rysem této aplikace je možnost vytváření interaktivních tutorialů a cvičení. V úvodu práce je proveden průzkum existujících řešení. Dále jsou pak popsány technologie použité při vývoji aplikace. Práce obsahuje také uživatelskou a programátorskou dokumentaci.

Synopsis

This thesis describes a web application that has been programmed to support web technology teaching. The main feature of this application is the ability to create interactive tutorials and exercises. At the beginning of the work, a survey of existing solutions is carried out. Next, the technologies used in the development of the application are described. The work also contains user and programming documentation.

Klíčová slova: HTML; CSS; JavaScript; PHP; Nette; MySQL

Keywords: HTML; CSS; JavaScript; PHP; Nette; MySQL

Děkuji mému vedoucímu Mgr. Petru Krajčovi, Ph.D. za rady a připomínky při vedení této práce. Dále děkuji své přítelkyni a rodině, kteří mě po celou dobu studia podporovali.

Místopřísežně prohlašuji, že jsem celou práci včetně příloh vypracoval/a samostatně a za použití pouze zdrojů citovaných v textu práce a uvedených v seznamu literatury.

datum odevzdání práce

podpis autora

Obsah

1	Úvod	8
1.1	Popis vytvořeného portálu	8
2	Analýza existujících řešení	9
2.1	Jak psát web	9
2.2	Itnetwork	10
2.3	W3schools	10
2.4	Závěrečné shrnutí	10
3	Použité technologie	11
3.1	HTML, CSS a JavaScript	11
3.2	PHP	12
3.3	Framework Nette	12
3.3.1	MVP (MVC) architektura	12
3.3.2	Presenter	13
3.3.3	Latte	13
3.4	MySQL	14
3.5	Ostatní	14
4	Programátorská dokumentace	15
4.1	Struktura projektu	15
4.2	Modely	16
4.3	AdminModule	17
4.4	FrontModule	18
4.5	Databáze	18
4.6	Základní princip interaktivního obsahu	20
4.6.1	Editor	21
4.6.2	Cvičení	21
5	Uživatelská dokumentace	21
5.1	Běžný uživatel	22
5.1.1	Registrace a přihlášení	22
5.1.2	Zapomenuté heslo	22
5.1.3	Správa profilu	22
5.1.4	Změna údajů	22
5.1.5	Hlavní menu	22
5.1.6	Tutorialy	24
5.1.7	Editor	24
5.1.8	Cvičení	25
5.1.9	Testy	25
5.1.10	Fórum	27
5.1.11	Nápověda	27
5.2	Redaktor	27

5.2.1	Přihlášení do administrace	27
5.2.2	Společné rysy vkládání obsahu	27
5.2.3	Vložení tutorialu	28
5.2.4	Vložení cvičení	29
5.2.5	Vložení testu	31
5.2.6	Správa obsahu	31
5.3	Administrátor	32
5.3.1	Instalace	32
5.3.2	Konfigurace	33
5.3.3	Přihlášení do administrace	34
5.3.4	Administrace uživatelů	34
5.3.5	Administrace obsahu	34
5.3.6	Administrace menu	34
5.3.7	Administrace sekcí	34
5.3.8	Administrace textů	35
5.3.9	Administrace fóra	35
	Závěr	36
	Conclusions	37
	A Seznam vlastností	38
	B Obsah přiloženého CD/DVD	39
	Literatura	40

Seznam obrázků

1	Životní cyklus presenteru [11]	13
2	Struktura projektu	15
3	Schéma databáze	19
4	Hlavní stránka portálu	23
5	Uživatelský panel nepřihlášeného a přihlášeného uživatele	24
6	Ukázka editoru	25
7	Ukázka cvičení	26
8	Podmínky u cvičení	29
9	Ukázka cvičení	32

Seznam tabulek

1	Podporované CSS vlastnosti	38
---	--------------------------------------	----

1 Úvod

Při výuce webových technologií se můžeme setkat s různými zdroji učebních materiálů. Jedná se například o klasické tištěné knihy a učebnice. Jejich nevýhodu však spatřuji v nutnosti ukázky zdrojových kódů opisovat, případně stahovat z přiloženého CD nebo internetových stránek. Někdy je jejich nevýhodou i vysoká pořizovací cena či zastaralý obsah.

Dalším zdrojem jsou internetové stránky a portály. Zdrojové kódy jsou obsaženy přímo v daných článcích či tutorialech a není je třeba nikde dohledávat. Obsah může být častěji aktualizován a držet tak krok s rychlým vývojem technologií. Navíc je většina těchto materiálů k dispozici zdarma.

Materiály na některých specializovaných webech jsou doplněny o živé ukázky zdrojových kódů. Lze tak přímo na stránkách vidět, jak se daný kód chová. Případně jej lze upravovat a lépe tak pochopit jak funguje. Přesto že je tato funkce pro učení velice přínosná, velká část stránek s výukovými materiály ji nepodporuje.

Cílem práce je vytvořit internetový portál, který bude uživatelům poskytovat obsah zaměřený na výuku webových technologií. Tento obsah bude interaktivní, uživatelé si tak budou moci aktivně vyzkoušet různé vlastnosti a možnosti jazyků jako jsou HTML, CSS nebo JavaScript. Stěžejní částí portálu tedy bude systém pro vytváření tohoto interaktivního obsahu.

Toto téma jsem si zvolil, protože se o webové technologie zajímám. Tato práce pro mě byla nejen vhodnou příležitostí k prohloubení mých stávajících znalostí ohledně tvorby webových stránek, ale také k naučení se nových technologií, jako je např. PHP framework Nette, se kterým jsem doposud nepracoval.

1.1 Popis vytvořeného portálu

Vytvořený portál obsahuje tři základní typy obsahu. Jedná se o tutoriály, cvičení a testy. Tutoriály slouží jako základní výukové materiály. Cvičení obsahují praktické úkoly, které má uživatel vyřešit. Správnost vytvořeného řešení lze nechat ověřit pomocí automatické kontroly. Pomocí testů si uživatel může ověřit získané znalosti.

Na základě přiřazené role je uživatelům k dispozici rozdílná funkčnost. Ne přihlášený uživatel může pouze prohlížet obsah, zkoušet testy a cvičení a číst příspěvky ve fóru. Přihlášený uživatel může spravovat svůj profil. Za úspěšně vyřešené cvičení nebo test získává body, na základě kterých je sestavován uživatelský žebříček a přiřazovány uživatelské úrovně. Ve fóru může také zakládat nová vlákna a přidávat vlastní příspěvky. Redaktor může vytvářet nové tutoriály cvičení a testy. Může spravovat obsah, který sám vytvořil. Administrátor může spravovat veškerý obsah portálu. Mezi tyto funkce patří například:

- vytváření a úprava kategorií menu a sekcí,
- úprava a mazání obsahu portálu,

- mazání a zamykání uživatelských účtů,
- změna uživatelských rolí,
- správa fóra,
- správa nápovědy.

Kompletní výčet možností a funkcí, které jsou dostupné pro uživatele s jednotlivými rolemi, je obsažen v uživatelské příručce.

V současné době se portál nachází na adrese <https://www.uc-se-web.cz> a pro testovací účely byly vytvořeny následující účty rozdělené dle uživatelských rolí:

- Administrátor:
 - uživatelské jméno: „Administrátor“
 - heslo: „test101“
- Redaktor:
 - uživatelské jméno: „Redaktor“
 - heslo: „test101“
- Uživatel:
 - uživatelské jméno: „Uživatel“
 - heslo: „test101“

2 Analýza existujících řešení

2.1 Jak psát web

Internetová stránka jakpsatweb.cz [1] je zájemcům o tvorbu webových stránek k dispozici již řadu let. Za tu dobu obsahuje velké množství materiálů, které jsou zaměřeny hlavně na HTML, CSS a JavaScript. Okrajově se stránky zabývají také jazykem PHP, databází MySQL nebo SEO¹ optimalizací.

Obsah stránky je různorodý, kromě různých materiálů, zde najdeme třeba seznam HTML značek a atributů rozdělený podle kategorií či seřazený abecedně. Dále je pro uživatele připraven „HTML tahák“, ve kterém mohou najít nejčastěji používané věci, jako je třeba základní struktura HTML dokumentu. Je tu také přehledný seznam CSS vlastností a jejich hodnot. Nechybí ani příklady, které ukazují možnosti jednotlivých jazyků. Dále zde najdeme různé návody a články nebo také encyklopedii pojmů sestavenou z nejvyhledávanějších výrazů na stránce. Ke stránce náleží také poměrně hojně využívané fórum.

¹Search Engine Optimization – optimalizace pro vyhledávače.

2.2 Itnetwork

V případě itnetwork.cz [2] se jedná o rozsáhlou stránku, která se prezentuje jako česká IT sociální síť. Zaměřuje se především na výuku programování v různých programovacích jazycích jako C#, Java, PHP nebo JavaScript. Najde se tu ale i obsah týkající se hardwaru, počítačových sítí či mobilních telefonů a jiných technologií. Uživatelům je umožněna registrace, díky které jim je zpřístupněna nová funkcionalita. Mohou spravovat svůj profil, přidávat komentáře k článkům, diskutovat ve fóru nebo si vyzkoušet své znalosti pomocí testů.

Obsah stránky je v podobě článků řazených v kategoriích. Články se dělí na základní a prémiové. Pro získání přístupu k prémiovým článkům potřebuje uživatel body. Body lze získat buď zakoupením, nebo za rozvoj sítě. Uživatel může požádat o redaktorská práva. Pokud splní podmínky pro jejich udělení, může přidávat na síť nový obsah. Pokud je obsah v požadované kvalitě, získává za něj uživatel body. Jsou zde i úkoly k procvičování, ty jsou ale téměř vždy prémiové.

Články obsahují ukázky zdrojových kódů, které jsou naformátované a se zvýrazněnou syntaxí. K článkům jsou připojeny soubory se zdrojovými kódy, které je možné si stáhnout. Ukázky u jazyků C#, Java a PHP jsou ukázky interaktivní a je možné je spouštět přímo na stránce. U HTML, CSS a JavaScriptu tato funkcionalita chybí.

2.3 W3schools

Stránka w3schools.com [3] je zaměřená opět primárně na výuku jazyků HTML, CSS, JavaScriptu, potom také PHP a SQL. Obsah je řazen do kategorií podle jazyka, v rámci jazyka pak ještě do sekcí. Tvoří ho tutoriály doplněné spoustou ukázek kódů. Ukázky se dají otevřít v novém okně, kde je lze upravovat a vyzkoušet.

Dále jsou zde také interaktivní cvičení. Uživatel dostane zadání s počátečním kódem, jeho úkolem je správně vyřešit problém popsáný v zadání cvičení. Je možné si nechat zobrazit nápovědu pro vyřešení, celé správné řešení, nebo svoje řešení nechat zkontrolovat. Jsou zde také kvízy pro ověření znalostí z tutorialů. Kvíz je tvořen jednotlivými otázkami, na které uživatel odpovídá výběrem jedné z nabízených možností.

2.4 Závěrečné shrnutí

Výše jsem popsal stránky, ze kterých jsem čerpal inspiraci. Nyní popíšu, v čem vidím jejich přednosti, v čem naopak nedostatky, a jak jsem se je snažil ve své práci odstranit.

Za největší slabinu stránky jakpsatweb.cz považuji ukázky zdrojových kódů. Není možné si je živě vyzkoušet, u příkladů nejsou ani uvedeny a uživatel si tak musí zobrazit zdrojový kód stránky v prohlížeči. V mé práci jsou ukázky kódů vykreslovány formátované, se zvýrazněnou syntaxí a lze je otestovat přímo na

stránce v editoru. Naopak se mi líbí fórum, které je zde aktivně využíváno, a proto jsem ho zařadil i do svého řešení.

Ani itnetwork.cz nenabízí živé ukázky pro jazyky HTML, CSS a JavaScript. Další nevýhodu spatřuji v tom, že úkoly k procvičování nejsou interaktivní, navíc jsou i u základních materiálů placené. U testů bych zase uvítal jejich větší počet, existuje zde nejvýše jeden test pro každý jazyk. V mé práci je možné vytvořit libovolný počet testů a rozdělit je například podle různé obtížnosti. U této stránky jsem se inspiroval uživatelskými profily, konceptem uživatelských úrovní a postupu na vyšší úroveň při dosažení určitého počtu bodů.

Stránka w3schools.com se nejvíce podobá mnou vytvořené práci. V případě interaktivních tutorialů a cvičení jsem se nechal inspirovat právě zde. Cvičení jsou však vytvořena jen pro HTML a CSS. V mé práci jsem přidal podporu pro vytváření cvičení i pro JavaScript. Jako v případě předchozí stránky i zde je k dispozici pouze jeden test ke každému jazyku. Dále mi zde chybí možnost vytvoření uživatelského účtu a sledování postupu při učení, například v podobě uživatelských úrovní či žebříku, které jsem proto zařadil do svého řešení.

3 Použité technologie

Následující sekce obsahuje stručný přehled technologií použitých při tvorbě portálu.

3.1 HTML, CSS a JavaScript

Jazyk HTML [4] neboli Hypertext Markup Language se řadí mezi značkovací jazyky a slouží k popisu sémantiky webových stránek. Je definován množinou značek společně s jejich atributy. Zapsáním textu mezi otevírací a ukončovací značku se textu přikládá význam daným typem značky. Takový text se označuje jako element. HTML dokument se skládá z jednotlivých elementů, které utvářejí stromovou strukturu.

Zatímco jazyk HTML definuje strukturu webové stránky, jazyk CSS [5], neboli Cascading Style Sheets, popisuje její vzhled. Vybraným elementům umožňuje nastavovat požadované vlastnosti. Jedná se například o barvu či velikost písma, barvu pozadí nebo okraje elementu. Výběr elementů se provádí pomocí různých typů selektorů např. typu, id, třídy a dalších. Za selektorem následuje blok s množinou vlastností a nastavovaných hodnot.

JavaScript [6] je objektově orientovaný, skriptovací jazyk. Existuje jak jeho klientská, tak serverová varianta. Klientský JavaScript přidává do statických stránek, vytvořených pomocí HTML a CSS, určitou dynamiku. Lze pomocí něj například měnit obsah HTML dokumentu či dynamicky nastavovat CSS vlastnosti. Umožňuje také reagovat na události vyvolané uživatelem, jako je kliknutí myši nebo stisk klávesy.

3.2 PHP

PHP [7] je objektově orientovaný skriptovací jazyk, určený především pro vytváření webových stránek. PHP skripty jsou vykonávány na straně serveru a klientovi je vrácena výsledná odpověď např. ve formátu HTML nebo JSON.

První verze PHP vznikla v roce 1994, od té doby procházelo postupným vývojem. Důležité změny přinesla verze 5, kdy byla výrazně vylepšena podpora objektově orientovaného programování. Přibyla například také podpora jmenových prostorů a anonymních funkcí. Verze 6 byla připravována, ale kvůli problémům při vývoji nikdy nevyšla. Aby se předešlo nejasnostem ve značení, byla nová verze označena číslem 7. Ta přináší zvýšení rychlosti jazyka a byly přidány nové funkcionality jako jsou typová kontrola skalárních datových typů, deklarace návratového typu nebo anonymní třídy. Aktuální verze jazyka je 7.2.[8]

3.3 Framework Nette

Nette [9] je moderní PHP framework vytvořený v roce 2008 českým programátorem Davidem Grudlem. Nabízí funkce pro odstranění bezpečnostních hrozeb jako XSS² nebo CSRF³. Podporuje také moderní technologie a postupy jako jsou MVC⁴, AJAX, Dependency Injection⁵ nebo DRY⁶. Aktuální verze je 2.4, která pochází z roku 2016. Na podzim roku 2018 je plánované vydání verze 3, která má využívat novinek v PHP 7.[10]

3.3.1 MVP (MVC) architektura

Nette framework je postaven na softwarové architektuře MVP⁷. Jedná se o architekturu velice podobnou MVC, v níž je kód aplikace rozdělen do tří vrstev, které oddělují aplikační logiku, zobrazená data a jejich obsluhu:

Model – Obsahuje všechnu logiku aplikace. Poskytuje rozhraní, pomocí kterého se s modelem pracuje. Voláním funkcí tohoto rozhraní mění model svůj vnitřní stav. O existenci controlleru ani view model neví.

View – Představuje vrstvu aplikace, která se stará o vykreslení a zobrazení dat z modelu uživateli. Často používá nějaký šablonovací systém.

Controller – Neboli řadič, slouží jako prostředník mezi modelem a view. Má na starosti zpracování požadavků uživatele, následně volá funkce modelu a nakonec žádá view o vykreslení dat. V Nette je označován jako presenter.

²Cross-site scripting – metoda narušující webové stránky, která spočívá v podstrčení cizího javascriptového kódu.

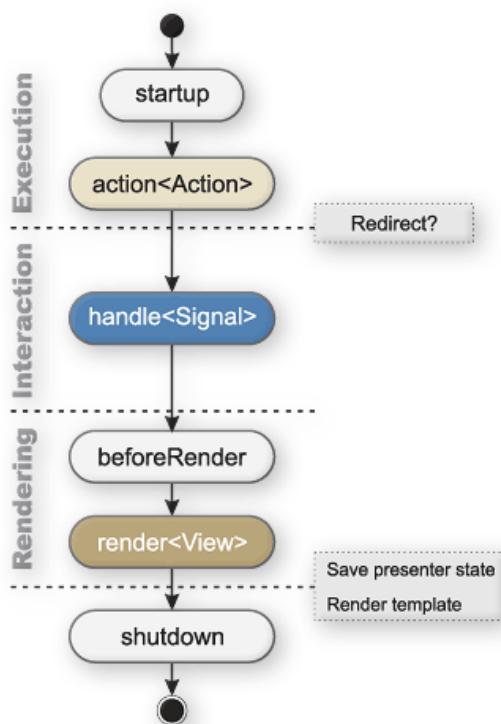
³Cross-site request forgery – metoda narušení webových stránek, která pracuje na principu zneužití nezamýšleného požadavku z neautorizovaného zdroje.

⁴Model view controller

⁵Automatické vkládání potřebných závislostí instancovaným třídám.

⁶Don't repeat yourself – princip založený na neopakování stejného kódu.

⁷Model view presenter



Obrázek 1: Životní cyklus presenteru [11]

3.3.2 Presenter

Presentery jsou v Nette objekty zpracovávající uživatelské požadavky, je proto vhodné si jejich fungování více přiblížit. Uživatelský požadavek na aplikaci se přes soubory `index.php` a `bootstrap.php` dostane do objektu `$application`. Ten s pomocí routeru zjistí, pro který presenter je požadavek určen, a kterou akci má vykonat. Následně objekt `$application` vytvoří instanci daného presenteru a zavolá požadovanou metodu (akci). Životní cyklus presenteru se skládá z několika metod a je znázorněn na obrázku 1.

3.3.3 Latte

Jedná se o šablonovací systém pro jazyk PHP používaný v Nette. Latte obohacuje klasické HTML o množinu maker a filtrů. Pomocí maker lze jednodušeji a přehledněji zapisovat konstrukce jazyka PHP, jako jsou například cykly nebo podmínky. Makra lze zapisovat dvěma způsoby:

- ve složených závorkách například `{foreach}`
- jako `n:makra` – jedná se o makra zapisovaná přímo k HTML elementu jako speciální atributy.

Je také umožněno definovat vlastní makra.

Latte vyniká svojí schopností použít různé escapovací funkce pro zabezpečení výpisu proměnných. Funkce je zvolena podle kontextu (HTML kód, JavaScriptový kód), ve kterém se vypisovaná proměnná nachází a výsledný kód je tak vysoce bezpečný.

3.4 MySQL

MySQL je volně dostupný, multiplatformní systém řízení báze dat, který využívá relační databázový model. Ve spojení s PHP je oblíbené a má dobrou podporu u hostingových serverů. Pro manipulaci s relačními daty se používá jazyk SQL. MySQL disponuje různými typy úložišť (storage engines), mezi které patří například MyISAM, Memory či InnoDB.[12] Pro moji práci bylo vybráno úložiště InnoDB, protože podporuje kontrolu referenční integrity dat pomocí cizích klíčů a transakcí.

3.5 Ostatní

CodeMirror

CodeMirror [13] je JavaScriptový editor, který slouží k formátování kódu a zvýrazňování syntaxe různých programovacích jazyků. V práci je použit pro editor a vytváření cvičení.

TinyMce

TinyMce [14] je JavaScriptový WYSIWYG⁸ editor, který umožňuje jednoduché vytváření HTML dokumentu. V aplikaci je použit pro vytváření tutorialů.

JQuery

jQuery [15] je JavaScriptová knihovna, která usnadňuje práci s JavaScriptem a manipulací s DOM⁹

HTML purifier

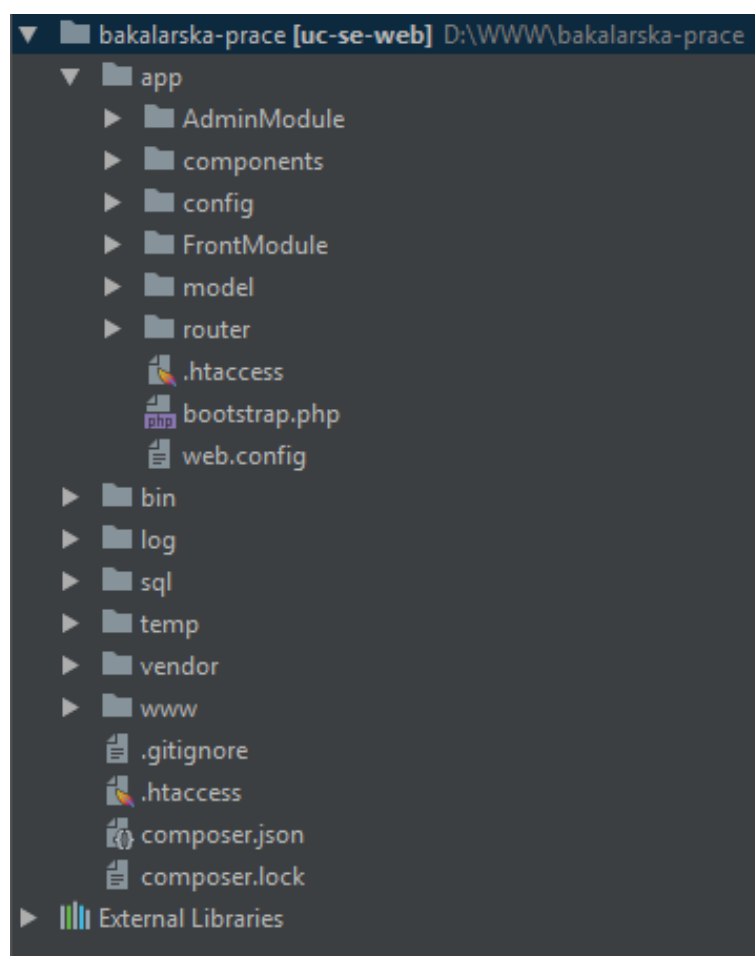
HTML purifier [16] je PHP knihovna, která slouží k odfiltrování potenciálně nebezpečného HTML kódu zadávaného uživatelem například v komentářích.

FontAwesome

FontAwesome [17] je knihovna ikon pro webové stránky a desktopové aplikace.

⁸What you see is what you get.

⁹Domain object model – objektově orientovaná reprezentace HTML dokumentu.



Obrázek 2: Struktura projektu

4 Programátorská dokumentace

Následující sekce slouží jako programátorská dokumentace k vytvořenému portálu. Obsahuje základní popis principu interaktivního obsahu, dále popis adresářové struktury projektu, modulů, ze kterých se skládá a popis databáze.

4.1 Struktura projektu

Adresářová struktura projektu vychází z doporučené adresářové struktury Nette projektu, na kterém je portál postaven. Projekt je rozdělen do dvou modulů se samostatnými presentery a šablonami. Jedná se o moduly AdminModule a FrontModule, detailněji budou popsány dále v textu. Modely a další soubory společné pro oba moduly se nacházejí ve složce app/. Struktura adresářů je znázorněna na obrázku 2 a obsahuje tyto složky:

app – v této složce se nachází jádro aplikace, skládá se z těchto podadresářů:

AdminModule – Obsahuje soubory administrátorské části portálu.

components – Obsahuje komponenty používané na různých částech portálu jako je třeba drobečková navigace nebo stránkování.

config – Obsahuje dva konfigurační soubory. Jsou to `config.neon` a `config.local.neon`. Neon [18] je speciální typ souboru používaný pro konfiguraci v Nette. Soubor `config.neon` obsahuje různé nastavení pro celý projekt, jako například registraci služeb, nastavení autorizace nebo expiraci sessions. Jsou zde také nastavení specifická pro vytvářený portál. V souboru `configlocal.neon` se pak nachází nastavení připojení k databázi.

FrontModule – Obsahuje soubory uživatelské části portálu.

model – Obsahuje modely společné oběma modulům.

router – V této složce se nachází router. Routerem je v Nette objekt, který zajišťuje překlad mezi URL a akcí presenteru.

bin – Obsahuje soubory, které se dají spustit z konzole.

log – Obsahuje soubory, do kterých se zapisují zaznamenané chyby, výjimky apod.

sql – Obsahuje sql skripty, kterými lze vygenerovat základní databázová struktura.

temp – Je to složka pro dočasné soubory, například cache databáze či šablon.

vendor – Je to složka s nainstalovanými závislostmi přes composer, obsahuje i samotné soubory Nette frameworku.

www – Veřejný adresář projektu. Nachází se zde složka s CSS soubory, obrázky nebo také složka s JavaScriptovými soubory potřebnými pro správnou funkčnost portálu. V ní se nachází složky se skripty potřebnými pro administraci a frontend portálu. Důležité jsou zejména soubory:

- `/front/exercise.js` – Obsahuje funkce, které mají za úkol zkontrolovat danou část správnosti řešení. Například funkce `itemHasClass()` kontroluje, jestli má zadaný prvek danou třídu.
- `/front/tutorial.js` – Obsahuje funkci `writteToFrame()`, která zajišťuje funkčnost editoru.

4.2 Modely

V modelech je obsažena logika aplikace. Využívá se zde principu tzv. „manažerů“ – jednotlivé databázové tabulky mají své manažery, které se starají o zápis, čtení, editaci a mazání dat z databáze. Dále bude popsáno několik základních modelů aplikace:

BaseManager – Jedná se o abstraktní manažer, který obstarává připojení k databázi, ostatní modely z něj dědí.

CategoryManager – Slouží ke správě kategorií, které představují menu portálu. Obsahuje metody pro ukládání a načítání stromové struktury menu z databáze.

ExerciseManager – Slouží ke správě cvičení, pro správu kontrolních podmínek využívá *ConditionManager*. Obsahuje funkci *getCheckFunction*, která na základně podmínek vytváří JavaScriptovou funkci pro kontrolu správnosti řešení.

TestManager – Slouží ke správě testů. Pro manipulaci s otázkami využívá služeb *QuestionManager* a pro správu odpovědí *AnswerManageru*.

TutorialManager – Slouží pro správu tutorialů.

UserManager – Slouží ke správě uživatelů, obsahuje metody pro registraci a přihlašování uživatelů nebo třeba zaslání emailu s odkazem pro obnovu zapomenutého hesla.

Ve složce jsou obsaženy i další modely zajišťující vedlejší funkčnost portálu jako je třeba správa uživatelských profilů a úrovní.

4.3 AdminModule

Modul obsahuje presentery a šablony specifické pro administrátorskou část portálu. Hlavní presentery budou vybrány a popsány:

BasePresenter – Jedná se o základní presenter, z kterého dědí další presentery v administrátorské části. Provádí se zde například kontrola administrátorského oprávnění.

ExercisePresenter – Jedná se o presenter, který se stará o administraci cvičení. Vytváří se zde formulář pro přidávání a editaci cvičení. Obsahuje také validaci ukládaných dat.

ManagePresenter – Jedná se o presenter, který obsahuje základní funkcionalitu pro filtrování a řazení spravovaných dat. Dále z něj dědí presentery zajišťující hromadnou správu konkrétních typů dat.

MenuPresenter – Stará se o vytvoření a validaci formuláře, pro správu kategorií menu.

TestPresenter – Má na starosti vytvoření formuláře pro přidávání a editaci testů. Provádí se zde také validace testů, otázek a odpovědí.

TutorialPresenter – Stará se o vytvoření formuláře, pomocí kterého se dají přidávat a editovat tutorialy. Tento formulář je zde následně i validován.

UsersPresenter – Tento presenter má na starosti filtraci a hromadnou správu uživatelů portálu. Obsahuje tedy metody pro hromadné mazání uživatelů, změny uživatelských rolí, zablokování uživatelského účtu apod.

Dále se v modulu nachází ostatní presentery sloužící pro hromadnou správu obsahu. Jedná se o: *ExercisesManager*, *TestsManager* a *TutorialsManager*. Dále presentery správu fóra a ostatních funkcí portálu.

4.4 FrontModule

Modul obsahuje presentery a jejich šablony, se kterými se dostane do styku obyčejný uživatel při běžném užívání. Dále budou opět vybrány a popsány nejdůležitější presentery:

BasePresenter – Slouží jako předek pro ostatní presentery. Vytvářejí se zde například komponenty vykreslující menu a uživatelský panel.

ExercisePresenter – Má na starosti získávání dat pro zobrazení cvičení a obsluhu ukládání výsledků úspěšně vyřešených cvičení.

SignPresenter – Tento presenter má na starosti přihlašování uživatelů. Vytváří se zde registrační formulář, přihlašovací formulář a formulář pro obnovu zapomenutého hesla.

TestPresenter – Obstarává data potřebná k vykreslování testů, obsluhu jejich zpracování a následně i zobrazení výsledků.

TutorialPresenter – Má na starosti získávání dat pro výpis detailů tutorialů, vytváří se zde formulář pro přidávání komentářů, obsahuje také metody pro editaci a mazání komentářů. Dále se zde také vytváří formulář představující editor.

Dále jsou zde obsaženy presentery potřebné pro chod dalších částí portálu např. *ForumPresenter* zabezpečující funkčnost fóra.

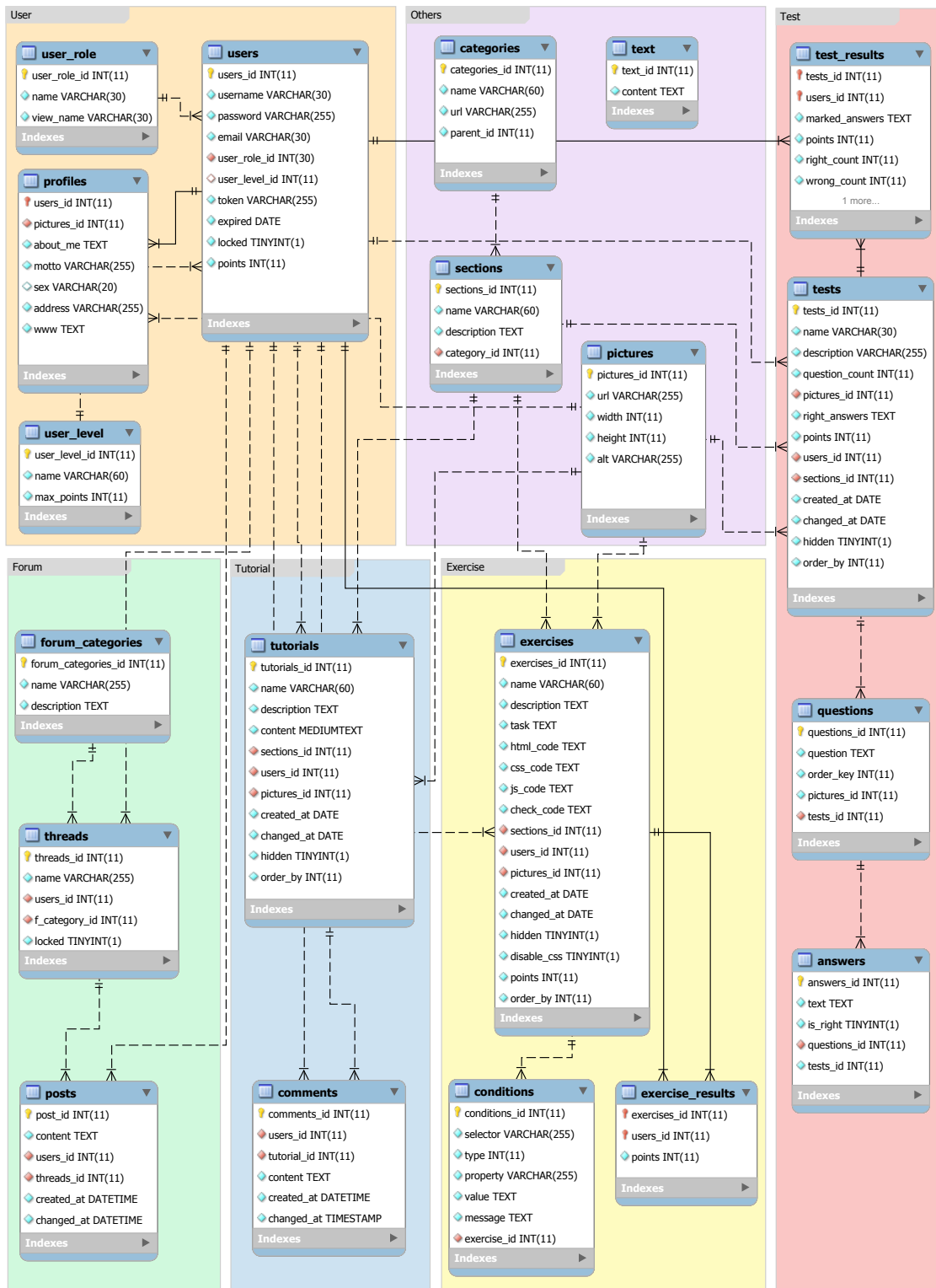
4.5 Databáze

Pro ukládání dat v aplikaci je použita databáze MySQL. Jako typ úložiště je u všech tabulek použit typ InnoDB. Integrita dat je zajištěna pomocí cizích klíčů. Schéma databáze je znázorněno na obrázku 3. Databáze obsahuje následující tabulky:

user_role – slouží pro ukládání rolí, které lze jednotlivým uživatelům přiřazovat,

users – slouží pro ukládání uživatelů,

profiles – slouží pro ukládání dat z uživatelských profilů,



Obrázek 3: Schéma databáze

user_level – slouží pro ukládání uživatelských úrovní,
forum_categories – slouží pro ukládání kategorií fóra,
forum_posts – slouží pro ukládání vláken obsažených v kategoriích fóra,
posts – slouží pro ukládání příspěvků fóra,
tutorials – slouží pro ukládání tutorialů,
comments – slouží pro ukládání komentářů přidaných k tutorialu,
exercises – slouží pro ukládání cvičení,
conditions – slouží pro ukládání podmínek zadaných při vytváření cvičení, ze kterých se sestavují kontrolní funkce,
exercise_results – slouží pro ukládání výsledků cvičení, která byla daným uživatelem úspěšně vyřešena,
tests – slouží pro ukládání vytvořených testů,
questions – slouží pro ukládání jednotlivých otázek přidaných do testu,
answers – slouží pro ukládání odpovědí na otázky,
test_results – slouží pro ukládání výsledků testů pro jednotlivé uživatele,
categories – slouží pro ukládání kategorie, které jsou zobrazeny jako menu portálu,
sections – slouží pro ukládání sekcí, do kterých se přiřazuje vytvářený obsah,
pictures – slouží pro ukládání obrázků,
text – slouží pro ukládání textů nápovědy.

4.6 Základní princip interaktivního obsahu

Hlavní částí vytvořeného portálu je interaktivní obsah v podobě tutorialů a cvičení, v této sekci bude popsán princip, na kterém tento interaktivní obsah funguje.

4.6.1 Editor

Editor slouží k zadávání kódu psaného v jazyce HTML doplněném o CSS a JavaScript. Tento kód se vyhodnotí a uživateli je zobrazena výsledná stránka. Editor se skládá ze tří částí. Levou část tvoří textové pole pro zadávání kódu. Je implementováno pomocí HTML prvku `<textarea>`, na tento prvek je navázán CodeMirror, který umožní lepší formátování textu a zvýraznění syntaxe. Pravá část editoru, ve které se zobrazuje výsledná stránka je tvořena HTML elementem `<iframe>`. Poslední částí je tlačítko pro zobrazení výsledku, je implementováno standardním způsobem pomocí HTML elementu `<input type=button>`.

Pro editor je důležitá vlastnost elementu `iframe` obsahovat v sobě další HTML dokument. Zobrazení výsledné stránky tedy funguje tak, že je pomocí JavaScriptu z textového pole vybrán zdrojový kód. Poté dojde k zapsání kódu do HTML dokumentu elementu `iframe`. Tím dojde k jeho zpracování a je zobrazen výsledek.

4.6.2 Cvičení

Ve cvičeních je opět použit editor s tím rozdílem, že zdrojový kód je do HTML dokumentu vkládán ze tří různých polí. Cvičení navíc obsahují možnost nechat si správnost výsledku ověřit pomocí automatické kontroly. Při vytváření cvičení zadává autor typ kontroly, selektor elementu, na kterém se má kontrola provést a požadované hodnoty. Typem kontroly je vybrána jedna z předdefinovaných JavaScriptových funkcí, která je volána s daným selektorem a hodnotami jako svými argumenty. Uvnitř funkce se dle selektoru vybere určitý prvek stránky. Dále se dle typu funkce provede požadovaná kontrola aktuálních vlastností elementu. Například se ověří, jestli má element požadovanou třídu, nebo vybraný atribut obsahuje požadovanou hodnotu. Při kontrole CSS vlastností hrají důležitou roli funkce `getComputedStyle()` a `getPropertyValue()`. První z nich získá všechny aktuální hodnoty CSS stylů, elementu předaného jako argument. Pomocí druhé lze získat hodnotu konkrétní vlastnosti daného elementu.

5 Uživatelská dokumentace

Tento internetový portál je určen všem, kteří se chtějí naučit základy tvorby webových stránek. Uživatelům je nabízena možnost seznámit se a aktivně si vyzkoušet technologie jako HTML, CSS nebo JavaScript. To vše formou tutorialů, cvičení a testů.

Pro správnou funkci je nutná zapnutá podpora JavaScriptu. Stránky byly optimalizovány pro následující a vyšší verze internetových prohlížečů:

- Google Chrome 65.0,
- Mozilla Firefox 61.0,
- Microsoft Edge 38.14,
- Internet Explorer 11.

5.1 Běžný uživatel

Tato část popisuje funkcionalitu portálu z pohledu běžného uživatele a má sloužit jako uživatelská příručka.

5.1.1 Registrace a přihlášení

Pro používání základních funkcí portálu, jako prohlížení tutorialů a psaní testů, není registrace nutná. Po registraci a přihlášení se však uživateli zpřístupní nové funkce, jako je vkládání komentářů, správa profilu a historie výsledků testů.

V hlavičce stránky, vpravo nahoře, se nachází uživatelský panel, který je znázorněn na obrázku 5. Tento panel obsahuje formulář pro přihlášení uživatele a je zde i odkaz pro registraci. Po kliknutí na tento odkaz se uživateli zobrazí registrační formulář. Je nutné vyplnit uživatelské jméno, email, heslo a heslo znovu pro kontrolu. Po úspěšné registraci je uživatel automaticky přihlášen a požádán o vyplnění profilu.

5.1.2 Zapomenuté heslo

Pokud uživatel zapomene heslo ke svému účtu, může požádat o jeho obnovu. To lze provést na stránce s formulářem pro obnovu hesla, na kterou se uživatel dostane kliknutím na příslušný odkaz v uživatelském panelu. Na této stránce je nutno zadat email vyplněný při registraci uživatele. Po odeslání formuláře je na tento email zaslán odkaz pro obnovu hesla.

5.1.3 Správa profilu

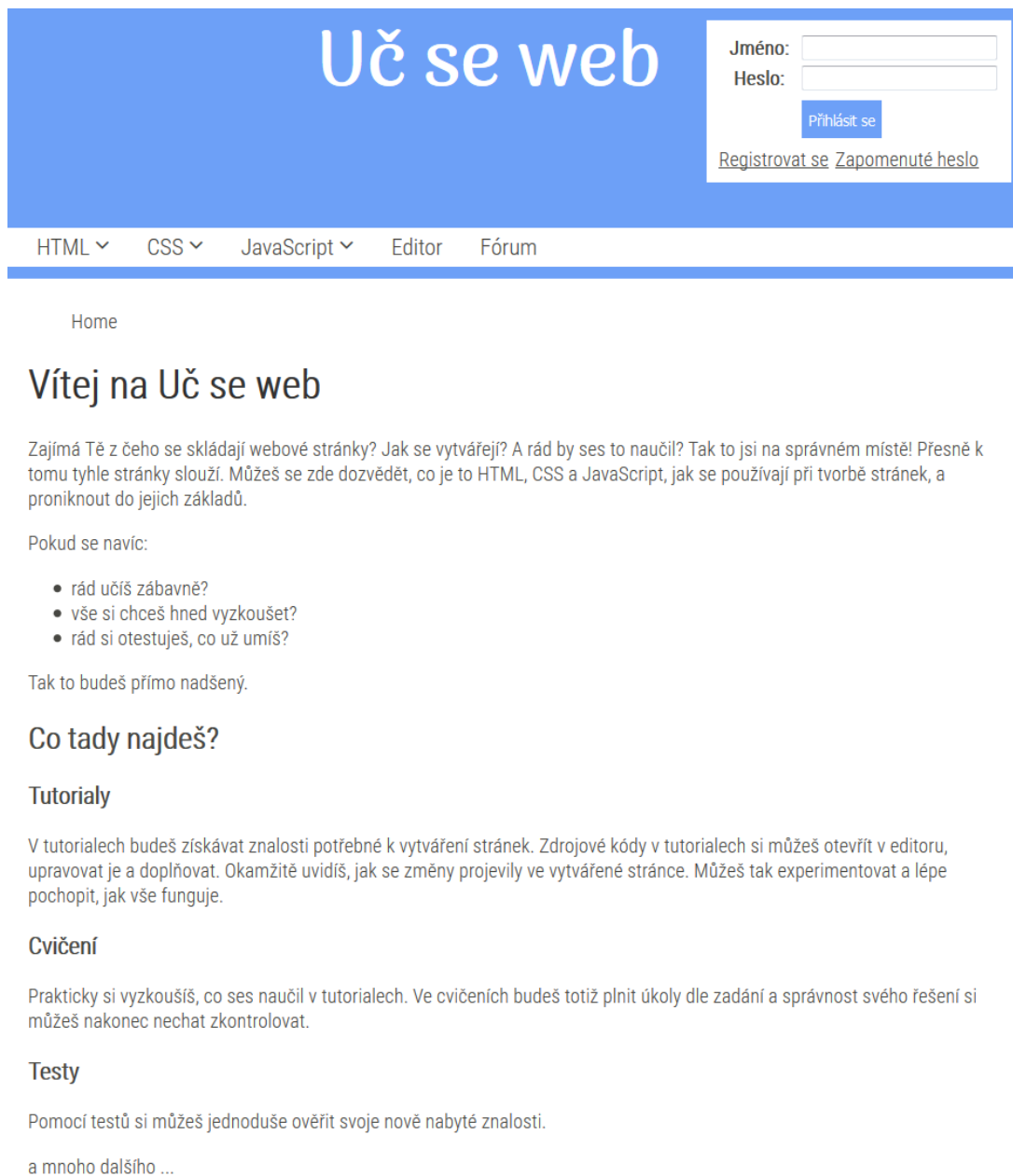
Po přihlášení najdeme v uživatelském panelu odkaz na stránku pro správu profilu. Na této stránce může uživatel vyplnit údaje, které si o sobě přeje zveřejnit. Může zde nastavit například svoje bydliště, oblíbené motto, osobní www stránky a další informace. Zde si může také nahrát vlastní obrázek, zobrazovaný u komentářů a příspěvků v diskuzním fóru.

5.1.4 Změna údajů

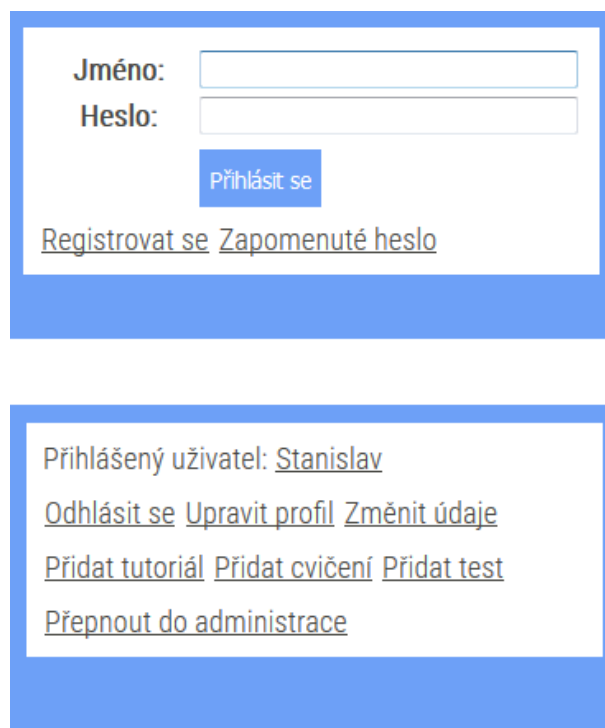
Změnu údajů je možné provést po kliknutí na příslušný odkaz v uživatelském panelu. Na zobrazené stránce je možné měnit uživatelské jméno, email a heslo. Pro změnu uživatelského jména nebo emailu stačí zadat příslušnou hodnotu a odeslat formulář. Pro změnu hesla je nutné zadat původní heslo, nové heslo a nové heslo znovu pro kontrolu.

5.1.5 Hlavní menu

Obsah portálu je členěn do kategorií. Ty mohou tvořit hierarchickou strukturu. To znamená, že každá kategorie může obsahovat další podkategorie. Kategorie a podkategorie se zobrazují jako položky menu v hlavičce stránky. Pokud je



Obrázek 4: Hlavní stránka portálu



Obrázek 5: Uživatelský panel nepřihlášeného a přihlášeného uživatele

u kategorie ikona šipky dolů, znamená to, že kategorie obsahuje podkategorie, které lze kliknutím zobrazit.

Pokud už kategorie neobsahuje žádné podkategorie, může obsahovat sekce. Sekce jsou konečným stupněm členění a nachází se v nich už samotný obsah např. tutoriály nebo testy.

5.1.6 Tutoriály

Tutoriály představují jednu ze tří základních složek obsahu portálu. Z nich může uživatel čerpat základní informace při učení. Kromě textu, obrázků a videí mohou tutoriály obsahovat také ukázky zdrojových kódů.

Pokud uživatel klikne na tlačítko pod ukázkou, otevře se nové okno s editorem, kde je možné si ukázkou vyzkoušet. Přihlášený uživatel může k tutorialům také vkládat komentáře. Může se tak například zeptat na věci, které mu nebyly v tutorialu jasné, nebo naopak někomu pomoci.

5.1.7 Editor

Editor slouží k vyzkoušení zdrojových kódů přímo na stránce. Skládá se ze dvou částí. Levá část je tvořena vstupním textovým polem, do kterého se zapisují příkazy. Lze sem zapisovat společně jak značky jazyka HTML, tak CSS vlastnosti a JavaScriptové příkazy.



```
1 <!DOCTYPE html>
2 <html>
3 <body>
4   <head>
5     <title>Ukázka nadpisů</title>
6   </head>
7   <h1>Nadpis první úrovně</h1>
8   <h2>Nadpis druhé úrovně</h2>
9   <h3>Nadpis třetí úrovně</h3>
10  <h4>Nadpis čtvrté úrovně</h4>
11  <h5>Nadpis páté úrovně</h5>
12  <h6>Nadpis šesté úrovně</h6>
13 </body>
14 </html>
```

Nadpis první úrovně

Nadpis druhé úrovně

Nadpis třetí úrovně

Nadpis čtvrté úrovně

Nadpis páté úrovně

Nadpis šesté úrovně

Vyzkoušet

Obrázek 6: Ukázka editoru

Pro lepší orientaci ve vytvářeném kódu obsahuje vstupní pole číslování jednotlivých řádků a také se zde zvýrazňuje syntaxe daných jazyků. Pod vstupním polem se nachází tlačítko pro vyzkoušení zdrojového kódu. Po kliknutí na něj se v pravé části editoru, v připraveném okně, zobrazí výsledná stránka.

5.1.8 Cvičení

Cvičení tvoří další z hlavních částí portálu. Jejich cílem je praktické procvičení znalostí, získaných v tutorialech, na konkrétních příkladech. Základem cvičení je zadání, které se uživatel snaží splnit. Pod zadáním se nachází opět editor, tentokrát jsou ale vstupy pro HTML, CSS a JavaScript oddělené. Tlačítkem zobrazit je možné podívat se na vytvořené řešení.

Cvičení mohou obsahovat tlačítko pro kontrolu správnosti vytvořeného řešení. Po jeho stisknutí se provede vyhodnocení zadaného řešení. Uživatel je informován buď o správnosti svého řešení, nebo mu je vypsán seznam chyb, kterých se dopustil. Za úspěšně vyřešené cvičení je uživateli připsán určený počet bodů. Podle počtu získaných bodů jsou uživatelům přiřazovány různé uživatelské úrovně. Podle bodů je také sestaven žebříček nejlepších uživatelů.

5.1.9 Testy

Testy slouží k ověření teoretických znalostí nabytých při studiu. Test může být tvořen libovolným počtem otázek. Otázky v testu jsou vždy uzavřené. Každá otázka musí obsahovat dvě a více odpovědí. Na otázku může být libovolný počet

Rozměry a barva pozadí

Zadání:

Divu s id "modry" nastav výšku i šířku na 100px a barvu pozadí na světle modrou(#00FFFF);

Html Css JavaScript

```
1 #modry{
2   height: 100px;
3   width: 100px;
4   background-color: #0000FF;
5 }
```



Zobrazit Zkontrolovat

Obrázek 7: Ukázka cvičení

správných odpovědí.

Napsání testu je intuitivní. Zaškrtnutím příslušného políčka vybere uživatel odpovědi, o kterých si myslí, že jsou správné. Po stisknutí tlačítka „Vyhodnotit test“ jsou zobrazeny správné odpovědi a dosažené skóre. Pokud je uživatel přihlášený, uloží se výsledek testu do jeho profilu. Tam je možné si výsledek znovu projít. Testy je možné psát opakovaně, ale kvůli objektivitě až po uplynutí určité doby, kterou určuje administrátor portálu. Za správně vyřešený test jsou uživateli opět připsány body. Záleží ovšem na procentuální úspěšnosti testu. Plný počet bodů je získán pouze za zcela správný test.

5.1.10 Fórum

Diskuzní fórum slouží k interakci mezi uživateli portálu. Skládá se z kategorií, vláken a příspěvků. Kategorie vytváří, upravuje a odebírá pouze administrátor. Týkají se vždy nějakého většího tematického celku. Do jednotlivých kategorií je možné vkládat vlákna. Vlákno může založit, upravit a případně smazat i uživatel a mělo by souviset s tématem kategorie. Do vláken pak další uživatelé vkládají svoje příspěvky.

5.1.11 Náповěda

Na stránkách kde je to účelné, může být zobrazen symbol otazníku. Tento symbol značí, že pro danou stránku existuje nápověda. Kliknutím na otazník, lze tuto nápovědu zobrazit a případně zase skrýt.

5.2 Redaktor

Tato část popisuje proces vytváření obsahu na portále a jeho následnou správu.

5.2.1 Přihlášení do administrace

Pro vstup do administrace je nutné se nejdříve klasicky přihlásit pomocí formuláře v hlavičce stránky. Po přihlášení se uživateli, s rolí redaktor nebo administrátor, zobrazí v uživatelském panelu odkaz pro vstup do administrace. Na hlavní stránce administrace se nachází administrační menu. Redaktor má zobrazené základní menu, týkající se vkládání a správy obsahu portálu.

5.2.2 Společné rysy vkládání obsahu

Obsah portálu je tvořen třemi základními skupinami. Jsou to tutoriály, cvičení a testy. Při jejich vkládání platí pro všechny skupiny níže uvedené společné rysy. Při popisu jednotlivých skupin už nejsou znovu opakovány.

- Obsah je řazen hierarchicky pomocí kategorií a sekcí. Kategorie mohou obsahovat podkategorie do libovolné úrovně. Nejnížší úroveň hierarchie pak tvoří sekce, do kterých je možné vkládat obsah. Každý tutorial, cvičení nebo

test musí být tedy zařazen v určité kategorii a sekci. Výběr kategorie a sekce se provádí pomocí rozbalovacích nabídek. Nejdříve se vybere kategorie, po jejím výběru se načtou sekce v ní uložené, a pak je možné vybrat sekci, do které má být obsah zařazen.

- Vždy je nutné zadat název. Název se společně s popisem a obrázkem zobrazuje uživateli při výpisu položek obsažených v sekci. Název by měl vystihovat podstatu obsahu. Např. „Tagy v HTML“
- Při vkládání obsahu je možné zaškrtnout políčko „Označit jako rozpracované“. Tím je možné určit, že obsah ještě není hotový. Nebude se tak zobrazovat běžným uživatelům a je možné ho dokončit později. Zároveň je možné takto označený obsah uložit, i když nejsou vyplněna všechna povinná pole. Kategorie, sekce a název však musí být vyplněny.
- Změnou hodnoty v poli „Pořadí v sekci“ je možné určit pozici, na které se bude obsah v sekci zobrazovat. Pokud má více položek stejnou hodnotu, budou seřazeny dle data vložení.
- U každého obsahu je také popis. V popise by měla být uvedena krátká informace o tom, co se zde uživatel dozví a doplňovat tak název. Např. „Zde se dozvíte, co jsou to tagy, jak se zapisují a jaké jsou jejich druhy“.
- U cvičení a testů lze zadat počet bodů, který se přičte přihlášeným uživatelům za správné vyřešení. U testů se získaný počet odvíjí od procentuální úspěšnosti. Za zcela správný test je získán plný počet bodů. Za test s poloviční úspěšností poloviční počet bodů atd. Tutoriály bodované nejsou.
- Ke každému obsahu je možné vložit hlavní obrázek. Ten se bude zobrazovat při výpisu položek obsažených v sekci. Pokud není vložen, zobrazuje se výchozí obrázek.

5.2.3 Vložení tutorialu

Pro vložení tutorialu je na stránce připravený WYSIWYG editor. Ten obsahuje řadu funkcí pro editaci textu, které jsou známé z běžných textových editorů.

Kromě textů lze vkládat i multimediální obsah v podobě obrázků a videí. Po kliknutí na ikonu obrázku se zobrazí dialogové okno s možností vložení obrázku. Je možné zadat buď URL obrázku nebo nahrát obrázek z počítače. Je zde možné provést i další nastavení, jako je změna velikosti obrázku nebo zadání jeho popisu. Další možností přidání obrázku je jeho přetažení myší do prostoru editoru. Videi lze vkládat obdobným způsobem. Po kliknutí na ikonu videa lze v dialogovém okně zadat buď URL adresu videa, nebo HTML kód pro jeho vložení. Lze zde také provést nastavení jako například rozměr videa nebo výběr obrázku pro náhled.

Důležitou funkcí při vytváření tutorialu je možnost vkládání ukázek zdrojových kódů. Vložení ukázky se provede kliknutím na ikonu „{}“ . Poté se v zobrazeném dialogovém okně vybere z rozbalovací nabídky jazyk ukázky a do textu-

Zkontroluj jestli:	Selektor:	Vlastnost:	Požadovaná hodnota vlastnosti:	Chybová zpráva:
Má zadaná Css vlastnost dano	#margin	margin	10px 20px	Margin neodpovídá zadání.
Má zadaná Css vlastnost dano	#padding	padding	10px 5px 8px 20px	Padding neodpovídá zadání.
+ Přidat podmínku				

Obrázek 8: Podmínky u cvičení

vého pole se vloží zdrojový kód. Takto vložená ukázka se při prohlížení tuoriálu přehledně zobrazí. Doplní se čísla řádků a je zvýrazněna syntaxe daného jazyka. Pod každou ukázkou ja navíc automaticky vygenerované tlačítko, pomocí kterého si může uživatel zadaný kód otevřít v editoru a vyzkoušet si ho. Je proto doporučené ukázky kódů v tutorialech používat.

5.2.4 Vložení cvičení

Vytvoření cvičení spočívá ve formulaci zadání, které uživatel plní. Zadání je doplněno o výchozí HTML kód a případně ještě o CSS a Javascriptový kód. Pro formulaci zadání a výchozí zdrojové kódy jsou připravena příslušná textová pole.

Cvičení mohou být doplněna o automatickou kontrolu uživatelova řešení. Automatická kontrola se ke cvičení přidá, pokud je při jeho vytváření vyplněna jedna či více podmínek. Každá podmínka určuje jednu vlastnost, kterou musí uživatelovo řešení splňovat, aby bylo vyhodnoceno jako správné. Přidání podmínky se provede stisknutím tlačítka „Přidat podmínku“. Kliknutím na tlačítko s ikonou odpadkového koše lze podmínku smazat. Podmínek je možné přidat libovolné množství.

Podmínek existuje několik typů, následuje jejich přehled společně se seznamem hodnot, které se zadávají pro konkrétní typ podmínky.

Existence třídy – Kontroluje se, jestli prvek obsahuje zadanou třídu. Vyžaduje zadání těchto hodnot:

- Selektor označující kontrolovaný prvek.
- Název požadované třídy.

Existence atributu – Kontroluje se, jestli prvek obsahuje zadaný atribut. Zadávají se tyto hodnoty:

- Selektor označující kontrolovaný prvek.
- Název požadovaného atributu.

Hodnota CSS vlastnosti – Kontroluje se, jestli je hodnota zadané CSS vlastnosti u prvku shodná s požadovanou hodnotou. Zadávají se tyto hodnoty:

- Selektor označující kontrolovaný prvek.
- Název kontrolované vlastnosti.
- Požadovaná hodnota vlastnosti.

Hodnota atributu – Kontroluje se, jestli je hodnota zadaného atributu u prvku shodná s požadovanou hodnotou. Zadávají se tyto hodnoty:

- Selektor označující kontrolovaný prvek.
- Název kontrolovaného atributu.
- Požadovaná hodnota atributu.

Vnitřní HTML – Kontroluje se, jestli je v prvku obsažena zadaná HTML struktura. Zadávají se tyto hodnoty:

- Selektor označující kontrolovaný prvek.
- Požadovaná HTML struktura.

Text – Kontroluje, jestli prvek obsahuje zadaný text. Zadávají se tyto hodnoty:

- Selektor označující kontrolovaný prvek.
- Požadovaný text.

Obsažení prvku – Kontroluje se, jestli je v prvku obsažen jiný prvek. Zadávají se tyto hodnoty:

- Selektor označující kontrolovaný prvek.
- Selektor označující prvek, který má ležet uvnitř prvního prvku.

Hodnota vrácená funkcí – Kontroluje se, jestli JavaScriptová funkce vrací pro zadané argumenty požadovanou hodnotu. Zadávají se tyto hodnoty:

- Jméno funkce.
- Argumenty funkce.
- Hodnota kterou má funkce vracet.

U všech typů podmínek je jako poslední hodnota zadávána chybová zpráva, která je zobrazena uživateli, pokud daná podmínka není splněna. Hodnoty se zadávají bez uvozovek či apostrofů, až na jednu výjimku u podmínky typu **Hodnota vrácená funkcí**.

Při výběru podmínky typu **Hodnota CSS vlastnosti** nejsou podporovány všechny CSS vlastnosti ale jen jejich část. Jejich seznam naleznete v příloze. Pokud zvolíte vlastnost, která určuje rozměry prvku jako např. *width*, zadávají se hodnoty rozměrů vždy v pixelech. Nelze použít relativní jednotky. Pokud zvolíte vlastnost určující barvu např. *color* zadává se v hexadecimálním tvaru např. *#FFF*, nebo v podobě RGB řetězce takto: *rgb(255,0,0)*. Pokud zvolíte

sduženou CSS vlastnost (např. margin), je možné zadat více hodnot oddělených mezerami např. *10px 20px* viz. obrázek 8.

Při výběru podmínky typu **Hodnota vrácená funkcí** lze zadat více atributů. V tom případě se atributy oddělují čárkou. V případě, že je hodnota atributu nebo kontrolovaná hodnota řetězec, je nutné jej vložit do apostrofů. Typ podmínky **Vnitřní HTML** nelze používat ke kontrole HTML struktur, jejíž prvky obsahují více než jeden atribut. Je to z toho důvodu, že nelze zaručit v jakém pořadí vrátí prohlížeč atributy. Dobře se ale hodí ke kontrole vnořených struktur, jako jsou například tabulky.

5.2.5 Vložení testu

Základem vytvoření testu je přidávání otázek a odpovědí. Text otázek i odpovědí se zadává do příslušných textových polí. Otázek je do testu možné vložit libovolný počet. Otázka se přidá kliknutím na tlačítko „Přidat otázku“. Kliknutím na tlačítko s ikonou odpadkového koše lze otázku smazat. Každá otázka musí povinně obsahovat vlastní text otázky a alespoň 2 možné odpovědi. Volitelně je ke každé otázce možné vložit i obrázek.

Výchozí počet odpovědí u otázky je 3. Další odpověď je možné přidat tlačítkem „+“. Odebrat odpověď lze opět tlačítkem s ikonou odpadkového koše. Maximální počet odpovědí u jedné otázky je 10. Správná odpověď se označí příslušným zaškrtačacím políčkem, které se nachází vedle každé odpovědi. Otázka může obsahovat libovolný počet správných odpovědí. Pokud je před text odpovědi vložena značka „@code“, je při zobrazení odpovědi zachováno odřádkování a odsazení.

5.2.6 Správa obsahu

Spočívá v zobrazování, filtrování, třídění a manipulaci s daty. Pro tento účel jsou data na stránce přehledně zobrazena v tabulce. Koncept tabulky se používá i při správě dalších částí webu a následující popis proto platí i pro administrátorskou část. Záhlaví tabulky je barevně zvýrazněno a obsahuje popis jednotlivých sloupců tabulky. Pokud je text v záhlaví podtržený, je možné na něj kliknout a data v tabulce se podle tohoto sloupce seřadí. Opakovaným kliknutím je možné měnit směr řazení. První řádek pod záhlavím může obsahovat textová pole nebo rozbalovací nabídky. Tato pole slouží pro zadání hodnoty, podle které chceme data v tabulce filtrovat. Pravý sloupec obsahuje tlačítka pro vyvolání akcí, které manipulují s daty v tabulce. První řádek v tomto sloupci obsahuje tlačítko s ikonou filtru pro filtrování dat dle zvolených hodnot. Dále tlačítko s ikonou křížku pro zrušení filtru. Nakonec tlačítko s ikonou odpadkového koše pro hromadné smazání označených položek.

Další řádky obsahují tlačítka akcí specifických pro daný typ dat. Jsou to například tlačítko se symbolem odpadkového koše pro smazání položky, tlačítko se symbolem oka pro prohlédnutí položky a tlačítko se symbolem tužky pro editaci položky.

Administrace tutorialů

	Název	Autor	Kategorie	Sekce	Přidáno	Změněno	Stav	Akce
<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	Všechny	Základy HTML			Všechny	<input type="button" value="🔍"/> <input type="button" value="✖"/> <input type="button" value="🗑"/>
<input type="checkbox"/>	1. Úvod do HTML	Stanislav	Tutorialy	Základy HTML	17-08-2017	05-06-2018	Dokončený	<input type="button" value="👁"/> <input type="button" value="✎"/> <input type="button" value="🗨"/> <input type="button" value="🗑"/>
<input type="checkbox"/>	2. Tagy a elementy	Stanislav	Tutorialy	Základy HTML	17-08-2017	05-06-2018	Dokončený	<input type="button" value="👁"/> <input type="button" value="✎"/> <input type="button" value="🗨"/> <input type="button" value="🗑"/>
<input type="checkbox"/>	3. Struktura dokumentu	Stanislav	Tutorialy	Základy HTML	17-08-2017	10-06-2018	Dokončený	<input type="button" value="👁"/> <input type="button" value="✎"/> <input type="button" value="🗨"/> <input type="button" value="🗑"/>
<input type="checkbox"/>	4. První webová stránka	Stanislav	Tutorialy	Základy HTML	17-08-2017	10-06-2018	Dokončený	<input type="button" value="👁"/> <input type="button" value="✎"/> <input type="button" value="🗨"/> <input type="button" value="🗑"/>
<input type="checkbox"/>	5. Základní tagy	Stanislav	Tutorialy	Základy HTML	18-08-2017	10-06-2018	Dokončený	<input type="button" value="👁"/> <input type="button" value="✎"/> <input type="button" value="🗨"/> <input type="button" value="🗑"/>
<input type="checkbox"/>	6. Atributy	Stanislav	Tutorialy	Základy HTML	18-08-2017	10-06-2018	Dokončený	<input type="button" value="👁"/> <input type="button" value="✎"/> <input type="button" value="🗨"/> <input type="button" value="🗑"/>
<input type="checkbox"/>	7. Seznamy	Stanislav	Tutorialy	Základy HTML	05-11-2017	10-06-2018	Dokončený	<input type="button" value="👁"/> <input type="button" value="✎"/> <input type="button" value="🗨"/> <input type="button" value="🗑"/>
<input type="checkbox"/>	8. Formátování textu	Stanislav	Tutorialy	Základy HTML	05-11-2017	10-06-2018	Dokončený	<input type="button" value="👁"/> <input type="button" value="✎"/> <input type="button" value="🗨"/> <input type="button" value="🗑"/>
<input type="checkbox"/>	9. Odkazy	Stanislav	Tutorialy	Základy HTML	05-11-2017	10-06-2018	Dokončený	<input type="button" value="👁"/> <input type="button" value="✎"/> <input type="button" value="🗨"/> <input type="button" value="🗑"/>
<input type="checkbox"/>	10. Tabulky	Stanislav	Tutorialy	Základy HTML	07-11-2017	10-06-2018	Dokončený	<input type="button" value="👁"/> <input type="button" value="✎"/> <input type="button" value="🗨"/> <input type="button" value="🗑"/>
<input type="checkbox"/>	11. Formuláře	Stanislav	Tutorialy	Základy HTML	11-11-2017	10-06-2018	Dokončený	<input type="button" value="👁"/> <input type="button" value="✎"/> <input type="button" value="🗨"/> <input type="button" value="🗑"/>
<input type="checkbox"/>	12. Obrázky	Stanislav	Tutorialy	Základy HTML	12-11-2017	10-06-2018	Dokončený	<input type="button" value="👁"/> <input type="button" value="✎"/> <input type="button" value="🗨"/> <input type="button" value="🗑"/>
<input type="checkbox"/>	13. Blokované a řádkové elementy	Stanislav	Tutorialy	Základy HTML	16-11-2017	10-06-2018	Dokončený	<input type="button" value="👁"/> <input type="button" value="✎"/> <input type="button" value="🗨"/> <input type="button" value="🗑"/>

◀ 1 ▶

Obrázek 9: Ukázka cvičení

Tutorialy, cvičení i testy lze spravovat stejným způsobem. Je možné je prohlížet, editovat a mazat. V případě editace jsou otevřeny ve stejném formuláři jako při vytváření. Redaktor má právo spravovat pouze obsah, jehož je sám autorem.

5.3 Administrátor

V této části je popsána instalace a konfigurace portálu. Dále pak jeho funkcionality z pohledu administrátora a je vysvětleno, jak používat funkce pro správu samotného portálu.

5.3.1 Instalace

Pro instalaci aplikace je nutné, aby webový server splňoval minimální požadavky pro běh Nette frameworku, jedná se především o PHP 5.6 a vyšší se zapnutým modulem `mode_rewrite`. Kompletní seznam požadavků lze nalézt na <https://doc.nette.org/cs/2.4/requirements>. Dále je pak nutný databázový server MySQL verze 5.6 nebo vyšší. Samotná instalace se skládá z několika následujících kroků:

- Prvním krokem je vytvoření nové prázdné databáze na databázovém serveru. V této databázi je nutné spustit skript `uc-se-web.sql`, který se nachází na přiloženém CD ve složce `app/sql`. Spuštění tohoto skriptu vytvoří potřebné databázové tabulky, dojde také k jejich naplnění výchozími daty a vytvoření administrátorského účtu: uživatelské jméno: „Administrátor“, heslo: „test101“.
- Jako další krok následuje zkopírování veškerého obsahu složky `app/` na při-

loženém CD do kořenového adresáře našeho diskového prostoru na webovém serveru.

- Poté je nutné v aplikaci nakonfigurovat databázové připojení. To se provede úpravou souboru *app/config/config.local.neon*. Úprava se bude týkat části pod klíčem „database“ na řádku č. 4. V této části upravíme následující položky:
 - dns: zde nastavíme položku `mysql:host` na adresu databázového serveru, dále v položce `dbname` nastavíme jméno vytvořené databáze
 - user: zadáme uživatelské jméno účtu pro správu databáze
 - password: zadáme heslo k účtu pro správu databáze
- Naposledy je potřeba přepsat soubor *.htaccess* v kořenové složce webového serveru. Tento soubor je potřeba upravit tak, aby všechny požadavky směřovaly do složky *www*. Tím je samotná instalace hotová, portál by měl být plně funkční a připravený na konfiguraci.

5.3.2 Konfigurace

Veškerou konfiguraci lze provést v souboru *app/config/config.neon*. Důležité jsou zejména první dva body. Zbytek konfigurace lze provést volitelně. Dále je uveden název klíče a popis vlastnosti, kterou lze hodnotou uvedenou u toho klíče nastavit.

email: Nastavení správcovského emailu. Tento email slouží jako kontaktní v případě, že je uživateli zamčen účet. Zobrazuje se také v emailu, který je uživateli odeslán při žádosti o reset hesla.

emailText: Nastavení textu emailu posílaného uživateli při žádosti o reset hesla.

maxFileSize: Maximální velikost nahrávaného obrázku v kB, neplatí pro obrázky nahrávané u otázek testů, ty se nastavují samostatně.

imageSize: Rozměry na které se upraví rozměry obrázku po nahrání na server. Zadávají se ve tvaru [X,Y] kde X je šířka obrázku a Y jeho výška. Obě hodnoty jsou v pixelech. Opět neplatí pro obrázky u otázek testů.

questionImageSize: Rozměry pro obrázky u otázek v testu. Pravidla pro jejich zadávání jsou stejná jako v předchozím případě.

questionMaxFileSize: Maximální velikost obrázku u otázek testů zadávána v kB.

passwordLength: Minimální délka hesla.

testRewrite: Doba za kterou je možné zkusit znovu napsat test. Udává se ve dnech.

tokenValidity: Doba po kterou platí odkaz pro resetování hesla. Ve dnech.

5.3.3 Přihlášení do administrace

Probíhá stejně jako u redaktora v části 5.2.1 Na hlavní administrační stránce je však administrátorovi k dispozici rozšířené menu s možností správy celého portálu. Popis vytváření a správy obsahu, uvedený v části 5.2, platí i pro administrátora. Jediný rozdíl je v tom, že administrátor může, na rozdíl od redaktora, spravovat i obsah, jehož není sám autorem.

5.3.4 Administrace uživatelů

Uživatele je možné v zobrazené tabulce třídit a filtrovat způsobem uvedeným v části 5.2.6. Dále je uživatele možné smazat, změnit mu roli nebo zamčít/odemčít účet. Zamčení/odemčení účtu se provede kliknutím na tlačítko se symbolem zámku. Změna role pak vybráním příslušné role z rozbalovací nabídky.

5.3.5 Administrace obsahu

Vkládání a správa obsahu je stejná jako u redaktora a je popsána v částech 5.2.2 - 5.2.6 . S tím rozdílem, že administrátor má právo spravovat i obsah jiných autorů.

5.3.6 Administrace menu

Administrace menu spočívá v přidávání a odebírání kategorií, které tvoří výsledné menu. Kategorie tvoří hierarchickou strukturu, tzn. že kategorie mohou obsahovat podkategorie a to do libovolné úrovně. Kategorie nejvyšší úrovně jsou zobrazeny jako hlavní položky menu. Jejich podkategorie se pak zobrazují jako rozklikávací podmenu.

Pro editaci menu je připraven editor. Tlačítkem „Přidat kategorii“ je možné vkládat rodičovské kategorie. Po vložení kategorie jsou zobrazeny dvě textová pole. Do prvního z nich je nutné vyplnit název kategorie, do druhého lze zadat URL adresu a vytvořit tak v menu externí odkaz. Místo URL adresy je možné zadat jeden z řetězců „editor“, „forum“ nebo „rank“. Tím je vytvořen odkaz na editor pro vyzkoušení zdrojových kódů, fórum, nebo žebříček uživatelů.

Podkategorii můžeme vložit příslušným tlačítkem, které se nachází u každé kategorie. Její vkládání probíhá stejně jako u kategorie. Podkategorie je vložena pod příslušnou kategorii a je odsazena směrem doprava. K podkategoriím je možné opět vkládat další podkategorie.

5.3.7 Administrace sekcí

Sekce je možné vkládat jen do konečných kategorií nebo podkategorií tzn. takových, které neobsahují žádné podkategorie. Do takové kategorie či podkategorie je pak možné vložit libovolný počet sekcí. Do samotných sekcí se potom vkládá obsah. Nová sekce se přidá příslušným tlačítkem. Nejprve je potřeba vyplnit její

název, potom z rozbalovací nabídky vybrat kategorii, do které má být zařazena a nakonec vyplnit popis.

5.3.8 Administrace textů

Zde je možné měnit různé texty zobrazované na stránkách. Jedná se o nápovědu, úvodní text a text hlavičky a patičky. Po výběru požadovaného druhu textu je možné jej upravit ve WYSIWYG editoru. Text hlavičky a patičky nemá smysl formátovat, do stránky se vkládá pouze prostý text. Délka textu v hlavičce je navíc omezena na 15 znaků.

5.3.9 Administrace fóra

Administrace fóra se skládá z administrace kategorií fóra, administrace vláken a administrace příspěvků. Nejdříve je zobrazena tabulka s kategoriemi fóra. Lze přidat novou kategorii, nebo stávající kategorie prohlížet, upravovat a mazat. U každé kategorie je také tlačítko pro administraci vláken, které jsou v kategorii obsaženy.

Administrace vláken je podobná, také je lze prohlížet, upravovat a mazat. U každého vlákna je také tlačítko pro administraci příspěvků, které obsahuje. Dále je zde tlačítko, kterým lze dané vlákno odemčít nebo zamčít. Do zamčeného vlákna nelze přidávat další příspěvky.

Příspěvky lze buď upravovat nebo je mazat.

Závěr

Zadáním bakalářské práce bylo vytvořit portál, který bude zaměřený na výuku webových technologií a kde si uživatelé budou moci aktivně vyzkoušet možnosti a vlastnosti jazyků HTML, CSS a JavaScript.

Značnou část práce tvoří systém pro vytváření a správu interaktivního obsahu. Výsledný portál obsahuje editor, ve kterém lze otevřít ukázky zdrojových kódů, obsažené v tutorialech. V editoru lze tyto ukázky upravovat, přičemž vliv těchto změn na výslednou podobu stránky lze ihned vidět ve vedlejším okně. Další částí portálu jsou cvičení, kde se uživatel snaží správně vyřešit zadaný úkol. Postup práce lze i zde vidět. Správnost svého řešení si uživatel může ověřit pomocí automatické kontroly. K ověření nabytých znalostí pak slouží testy, které portál také obsahuje. Dle mého názoru byly tedy stanovené cíle práce naplněny.

Výsledný portál může být nadále vylepšován a rozšiřován. Mimo přidávání nového obsahu může být jednou z těchto změn například přepracování vzhledu aplikace nebo vylepšení responzivity, kterým nebylo věnováno tolik pozornosti jako samotné funkčnosti. Co se týká funkčnosti, mohla by být přidána podpora pro některé další jazyky. Vytváření testů a cvičení by mohlo být upraveno tak, aby šlo volitelně přidávat cvičení a testy měřené na čas. Uživatelům, kteří dosáhnou určitého počtu bodů, by mohlo být umožněno podílet se na vytváření obsahu.

Conclusions

The goal of the bachelor thesis was to create a portal, that will be focused on teaching web technologies and where users will be able to actively try HTML, CSS, and JavaScript. In my opinion, the goal of the bachelor thesis was fulfilled. A portal has been created that includes interactive tutorials, exercises and tests. The main part of the portal is a system for creating this interactive content. Tutorials are the main teaching materials. They contain source code samples, that can be tested in the editor. In exercises, the users implement the assigned tasks. Tests are for verifying knowledge.

The portal can still be improved. For example, a graphical and responsive design can be improved. New functionality can also be added, for example support for new programming languages. Next, exercises and tests can be complemented by a time limit. Users who get the specified number of points can be allowed create content.

A Seznam vlastností

Tabulka 1: Podporované CSS vlastnosti

background-color	border-bottom-color	border-bottom-left-radius
border-bottom-right-radius	border-bottom-style	border-bottom-width
border-color	border-left-color	border-left-style
border-left-width	border-radius	border-right-color
border-right-style	border-right-width	border-style
border-top-color	border-top-left-radius	border-top-right-radius
border-top-style	border-top-width	border-width
bottom	color	cursor
display	float	font-family
font-size	font-weight	height
left	line-height	list-style-type
margin	margin-bottom	margin-left
margin-right	margin-top	max-height
max-width	min-height	min-width
opacity	overflow	padding
padding-bottom	padding-left	padding-right
padding-top	position	right
text-align	text-decoration	top
visibility	width	z-index

B Obsah přiloženého CD/DVD

app/

Zde se nachází zdrojové soubory vytvořeného portálu, připravené pro zkopírování do kořenového adresáře webového serveru.

doc/

Zde se nachází text bakalářské práce v souboru bakalarska-prace-cech.pdf, vytvořený s použitím závazného stylu KI PŘF UP v Olomouci pro závěrečné práce. Dále jsou zde všechny zdrojové soubory potřebné pro vygenerování tohoto PDF dokumentu v ZIP archivu s názvem bakalarska-prace-cech.zip.

readme.txt

Instrukce pro nasazení internetového portálu na webový server.

Literatura

- [1] JAKPSATWEB: www.jakpsatweb.cz [online] [cit. 2018-07-01]. Dostupné z: <https://www.jakpsatweb.cz/>
- [2] ITNETWORK: Ajtácká sociální síť. [online] [cit. 2018-07-01]. Dostupné z: <https://www.itnetwork.cz/>
- [3] W3SCHOOLS: Online Web Tutorials. [online] [cit. 2018-07-01]. Dostupné z: <https://www.w3schools.com/>
- [4] W3C HTML: World Wide Web Consortium (W3C). HTML. [online] [cit. 2018-07-01]. Dostupné z: <https://www.w3.org/TR/html52/>
- [5] W3C CSS: World Wide Web Consortium (W3C). CSS. [online] [cit. 2018-07-01]. Dostupné z: <https://www.w3.org/Style/CSS/>
- [6] MDN: Resources for developers, by developers. JavaScript. [online] [cit. 2018-07-01]. Dostupné z: <https://developer.mozilla.org/bm/docs/Web/JavaScript>
- [7] PHP: Hypertext Preprocessor. [online] [cit. 2018-07-01]. Dostupné z: <http://php.net/>
- [8] PHP: Changelog - Manual. PHP: Hypertext Preprocessor [online] [cit. 2018-07-01]. Dostupné z: <http://php.net/manual/en/doc.changelog.php>
- [9] NETTE FRAMEWORK: Rychlý a pohodlný vývoj webových aplikací v PHP.[online] [cit. 2018-07-01]. Dostupné z: <https://nette.org/cs/>
- [10] PHPFASHION: Co se chystá v Nette?. [online] [cit. 2018-07-01]. Dostupné z: <https://phpfashion.com/co-se-chysta-v-nette>
- [11] NETTE FRAMEWORK: MVC aplikace & presentery. [online] [cit. 2018-07-01]. Dostupné z: <https://doc.nette.org/cs/2.4/presenters>
- [12] MYSQL: MySQL 5.6 Reference Manual. [online] [cit. 2018-07-01]. Dostupné z: <https://dev.mysql.com/doc/refman/5.6/en/>
- [13] CODEMIRROR: CodeMirror [online] [cit. 2018-07-01]. Dostupné z: <https://codemirror.net/>
- [14] TINYMCE: The Most Advanced WYSIWYG HTML Editor [online] [cit. 2018-07-01]. Dostupné z: <https://www.tiny.cloud/>
- [15] HTML PURIFIER: Filter your HTML the standards-compliant way! [online] [cit. 2018-07-01]. Dostupné z: <http://htmlpurifier.org/>
- [16] JQUERY: jQuery [online] [cit. 2018-07-01]. Dostupné z: <https://jquery.com/>

- [17] FONT AWESOME: Font and icon toolkit. [online] [cit. 2018-07-01]. Dostupné z: <https://fontawesome.com/>
- [18] NETTE FRAMEWORK: Práce s formátem NEON. [online] [cit. 2018-07-01]. Dostupné z: <https://doc.nette.org/cs/2.4/neon/>