

Webové stránky konferenčního systému

Semestrální práce z předmětu KIV/WEB

Jan Hinterholzinger

hintik@students.zcu.cz

<https://hintik.cz/KIV-WEB>

7. ledna 2021

Obsah

1	Základní informace	3
1.1	Umístění projektu	3
1.1.1	Webový server	3
1.1.2	Projekt na GitHubu	3
1.1.3	Přístup k databázi	3
1.2	Použité technologie	4
1.2.1	MariaDB 10.3	4
1.2.2	BootStrap 5.0 a JQuery	4
1.2.3	CKEditor	4
1.2.4	FontAwesome	4
1.2.5	Popper	5
1.2.6	Google ReCaptcha v2	5
2	Popis adresářové struktury aplikace	6
2.1	Kořenový adresář	6
2.2	Adresář /app	6
2.3	Adresář /vendor	6
2.4	Adresář /www	7
2.5	Adresář /sql	7
2.6	Adresář /doc	7
3	Popis architektury aplikace	8
3.1	Kontrolery	8
3.2	Modely	8
3.2.1	Třída UserSessionModel	8
3.2.2	Třída UserLoginModel	8
3.2.3	Třída Utilities	8
3.2.4	Třída DatabaseModel	9
3.2.5	Třída Alerts	9
3.2.6	Třídy Roles a States	9
3.3	Šablony	9

4	Struktura databáze	11
5	ReCaptcha na jiném serveru	12

Kapitola 1

Základní informace

1.1 Umístění projektu

1.1.1 Webový server

Celý projekt je uložen na webhostingu provozovaného firmou WEDOS Internet, a. s. Adresa celého projektu je <https://hintik.cz/KIV-WEB> jedná se o kořenový adresář projektu, ve kterém se nenachází indexový soubor *index.php*. Ten je umístěn v podadresáři *www/*, tedy na adrese <https://hintik.cz/KIV-WEB/www>, na které se již zobrazí úvodní stránka práce.

1.1.2 Projekt na GitHubu

Celý projekt je uložen také na službě GitHub na adrese <https://github.com/hintikulus/KIV-WEB-Semestralni-prace>. GitHub byl využíván příležitostně a ne zcela strategicky.

1.1.3 Přístup k databázi

Pro účel prohlížení databáze byl vytvořen uživatel pro dotazování příkazem SELECT. Přihlašovací údaje tohoto uživatele jsou: login: **kivweb-read**, heslo: **heslo**. Databáze je spuštěna na adrese **vpn.hintik.cz** na standardním portu 3306. Název použité databáze (schematu) je **KIVWEB**.

1.2 Použité technologie

Webové stránky využívají následující technologie

- HTML 5
- PHP 7.3
- MariaDB 10.3

Kromě těchto technologií byly využity následující technologie.

1.2.1 MariaDB 10.3

Pro ukládání dat byl použit databázový systém MariaDB ve verzi 10.3 nainstalovaném na VPS (na adrese *vpn.hintik.cz*).

1.2.2 Bootstrap 5.0 a JQuery

Téměř celý web je stylově vytvořen zcela pomocí frameworku Bootstrap ve verzi 5.0 bez vnějších úprav vlastními styly. Tímto byla vytvořena responzivita a umírněný neagresivní vzhled. Pomocí bootstrapu tak byly vytvořeny také upozornění. Také bylo implementováno JQuery pro potřeby technologie Bootstrap například na rozbalovací menu přihlášeného uživatele.

1.2.3 CKEditor

Technologie CKEditor byla použita pro lepší modifikovatelnost určitých vstupů od uživatele. Byla využita například u biografie uživatele na stránce editace uživatele (stránka *user-edit*), pro napsání abstraktu článku při vytváření nového článku (stránka *post*) a při vytváření recenzí u detailu článku (stránka *article*)

1.2.4 FontAwesome

Na určitých místech webu byly použity ikony technologie FontAwesome, hlavně pro zpestření a lepší rozeznatelnost. Ikony jsou použity například na stránce *user-articles* a *article*, kde označují stav příspěvku, *admin-reviews*, kde zobrazují, zda přiřazený recenzent již publikoval svou recenzi.

1.2.5 Popper

Technologie Popper byla využita u popisků tlačítek a ikon, u kterých nemusel jejich vzhled dostatečně odpovídat prováděné akci nebo sdělovací informaci. Bylo použito například u tlačítka stahování PDF souboru článku a na stejné stránce u ikonky označující stav schvalovacího procesu (stránka *article*).

1.2.6 Google ReCaptcha v2

Při registraci nových uživatelů je využita technologie Google ReCaptcha, která se stará o ochranu registrací nových uživatelů před bot útoky (útok automatických registrací uživatelů s použitím robotů).

Kapitola 2

Popis adresářové struktury aplikace

Architektura aplikace je částečně převzata, silně inspirovaná a rozšířena od architektury použité při 7. cvičení předmětu KIV/WEB při ukázce MVC architektury.

2.1 Kořenový adresář

Kořenový adresář obsahuje (kromě skrytých aktuálně nepodstatných adresářů *.git* a *.idea*) adresáře *app*, *vendor*, *www*, *sql* a *doc* a soubory pro konfiguraci projektu pro composer, autoloader, který načítá správné šablony a kontrolery pro aktuální stránku a soubor *settings.php*, obsahující konstaty s nastavením aplikace.

2.2 Adresář /app

Veškerý obsah aplikace je umístěn v tomto souboru a je rozdělen na 3 kategorie, pro dodržení MVC na kontrolery, modely a pohledy (šablony). Kontrolery se starají o obsluhu požadavků šablon, šablony se starají o vzhledovou stránku práce a modely reprezentují a zpracovávají informace se kterými se pracuje.

2.3 Adresář /vendor

Veškeré potřebné technologie, které jsou potřeba k správnému chodu aplikace jsou Composerem nainstalovány do tohoto adresáře.

2.4 Adresář /www

Adresář obsahuje vstupní bod programu (*index.php*), který celou aplikaci spouští.

2.5 Adresář /sql

Obsahuje veškeré SQL skripty pro vytvoření potřebné struktury databáze.

2.6 Adresář /doc

Tento adresář obsahuje pouze tuto dokumentaci, která je přibalena k projektu.

Kapitola 3

Popis architektury aplikace

3.1 Kontrolery

Kontrolery slouží k obsluze požadavků z šablon. Získává tedy data pro šablony a připraví je pro přímé použití v šabloně. Protože existují činnosti, které musí být provedeny v každém kontroleru, tak byla vytvořena třída *PageController*, kterou všechny kontrolery dědí a používají její metody. Následně kontrolery získávají informace z modelů (např. z databáze) a připravují tím data na vykreslení šablonou.

3.2 Modely

3.2.1 Třída `UserSessionModel`

Třída slouží k práci se session. Umožňuje do ní vkládat, odebírat a kontrolovat informace. Využívá se především při práci s přihlašováním.

3.2.2 Třída `UserLoginModel`

Přihlašování uživatele je obsluhováno touto třídou. Třída se stará o správné volání funkcí třídy pro práci se session a poskytuje informace o přihlášení uživatele.

3.2.3 Třída `Utilities`

Tato třída obsahuje funkce, které aplikace pro chod využívá, ale nebylo nutné pro ně vytvářet speciální třídu.

Třída tedy obsahuje funkce pro:

- validaci emailu a hesla
- výpis hvězdiček pro hodnocení článků
- výpis stránkovací navigace
- úprava výstupu z CKEditoru s XSS ošetřením
- přesměrování

3.2.4 Třída DatabaseModel

Jedna z hlavních tříd, která obsluhuje požadavky na přístup k databázi. Třída má několik předpřipravených funkcí pro získání, úpravu nebo vkládání dat aplikace (data uživatelů, příspěvků, recenzí). Kromě toho se také stará o vytvoření a správu spojení s databází. Každá funkce, pokud je to potřeba, si hlídá vstupní data na ochranu proti SQL injection a XSS.

3.2.5 Třída Alerts

Stará se o veškeré výpisy upozornění. Protože upozornění se mohou několikrát opakovat na různých stránkách, tak bylo zvoleno uložení zpráv oznámení jako konstanty, které se poté mohou volat z odkudkoli z projektu. Je tak možné změnit jedno upozornění na více místech současně a neopisovat celou zprávu. Rovněž třída obsahuje funkci pro vytvoření struktury pro uchování upozornění pro jejich následné vypsání šablonou.

3.2.6 Třídy Roles a States

Tyto třídy se starají o sjednocení názvů a pojmenování rolí a stavů v aplikaci.

3.3 Šablony

Šablony jsou použity pro vypsání dat z kontrolerů a modelů do pevně dané HTML šablony. Mohou se zde objevovat podmínky pro odlišné zobrazení například pro přihlášeného uživatele. Šablony jsou rozděleny na dva druhy standardní (*TemplatePlain*) a pro administraci (*TemplateAdministration*). Většina stránek je psána pro standardní šablonu, tedy jako základ využívají předpřipravenou stránku s navigací a patičkou, kde se zároveň nalinkují

všechny potřebné soubory (CSS, JavaScript, atd.) a spustí scripty. *TemplateAdministration* je nadstavbou standardní šablony určenou pro stránky administrace. Oproti standardní šabloně je přidána navigace na levé straně a prostor pro samotný obsah je tedy zúžen.

Kapitola 4

Struktura databáze

Databáze je rozdělena na 3 tabulky *users* (uživatelé), kde jsou uložena základní informace o uživateli jako jméno, příjmení, login, hash hesla a role. Dále se ve struktuře nachází tabulka *articles* (články), kde jsou umístěna data článků jako titulek, abstrakt, PDF soubor článku, informace o publikování a autoru. Poslední tabulka je tabulka *reviews*, která reprezentuje přiřazení recenzenta k ohodnocení článku. Pomocí hodnoty stavu recenze a vyplněnosti určitých sloupců se určuje, zda je recenze již napsána nebo nikoli. Tabulky mají nastavenou kaskádovost při mazání záznamů. Tím je myšleno, že při smazání záznamu z databáze se kaskádově odstraní veškeré záznamy, kde se objevuje cizí klíč odstraňovaného záznamu. To je využito při odstraňování prvků, kdy například při smazání uživatele se odstraní též všechny články a recenze, které uživatel publikoval.

Kapitola 5

ReCaptcha na jiném serveru

Protože je v aplikaci použita služba Google ReCaptcha, je problematické ji zpřístupnit pro localhost nebo pro jiný server. Proto bylo implementována konstanta `RECAPTCHA_VALIDATION` v souboru `settings.inc.php`, podle které se určuje, zda se bude validovat ReCaptcha. Pokud je nastavena na `true`, tak je připravena pro reálné použití a kontroluje vstup. V případě nastavení na `false` se sice okénko pro zadání ReCaptchy ve formuláři objeví, ale jeho validace na serveru nebude provedena a bude ignorováno.