



ZÁPADOČESKÁ UNIVERZITA V PLZNI  
FAKULTA APLIKOVANÝCH VĚD

DOKUMENTACE K  
SEMESTRÁLNÍ PRÁCI  
KIV/WEB  
WEBOVÉ STRÁNKY  
KONFERENČNÍHO SYSTÉMU

Martin Červenka A14B0239P  
cervemar@students.zcu.cz

2. prosince 2016

# 1 Použité technologie

V aplikaci jsem použil mnoho technologií, mezi ty hlavní řadím *HTML5*, *CSS*, *PHP*, *MySQL*, *JavaScript*, *JQuery*, *AJAX*, *Bootstrap*, *CKEditor* a *htaccess*. Následuje popis těchto technologií a místa, kde se tyto technologie v aplikaci vyskytují.

- **HTML5**

Značkovací jazyk HTML5 je základní technologie, kterou jsem použil. Soubory obsahující HTML5 lze nalézt ve složce *view* → *templates*, kde je použit i šablonovací systém Twig.

- **CSS**

Kaskádové styly jsem použil k drobnému vylepšení, kdy většinu stylů jsem použil z knihovny Bootstrap. Příkladem souboru kaskádových stylů je soubor *style.css* ve složce *view*.

- **PHP**

PHP jsem použil pro zpracování požadavků od klienta – jako controller. Tyto soubory se nacházejí ve složce *controller*.

- **MySQL**

Jako dotazovací jazyk jsem zvolil MySQL. Aplikace ale obsahuje PDO, proto je možné dotazovací jazyk poměrně jednoduše změnit. Jako příklad uvádím skript *sql.sql* v kořenovém adresáři, nebo soubor *database.pool.php* ve složce *model*, který s databází pracuje.

- **Twig**

Šablonovací systém Twig jsem použil zejména pro zpřehlednění kódu. Soubory šablon se nacházejí ve složce *templates*, jak jsem již uvedl výše v bodě 1.

- **JavaScript, JQuery a AJAX**

Javascript v kombinaci s JQuery a AJAXem jsem použil hlavně pro asynchronní komunikaci se serverem. Aplikce však funguje i pokud má klient javascript deaktivovaný – v tom případě aplikace komunikuje synchronně. Soubory javascriptu se nacházejí ve složce *controller* → *js*

- **Bootstrap**

Pro vzhled aplikace bylo vhodné použít knihovnu Bootstrap. Obsahuje mnoho předpřipravených stylů a řeší i responzivitu webu. Bootstrap je použit ve většině šablon, které se nacházejí v adresáři uvedeném v prvním bodě.

- **CKEditor**

Pro zjednodušení používání aplikace jsem do aplikace přidal CKEditor. Tato knihovna vytvoří uživatelsky přívětivý editor přímo na webové stránce. CKEditor je přidán pomocí javascriptu do šablony *form.tmpl*.

- **htaccess**

Htaccess jsem použil pro povolení tzv "pěkných URL". Soubor *.htaccess* se nachází přímo v kořenovém adresáři aplikace.

## 2 Adresářová struktura

Adresářový strom 1 popisuje strukturu aplikace. Složky jsou barevně odlišeny od souborů, které jsou pro jednoduchost sdruženy, pokud mají stejnou příponu.

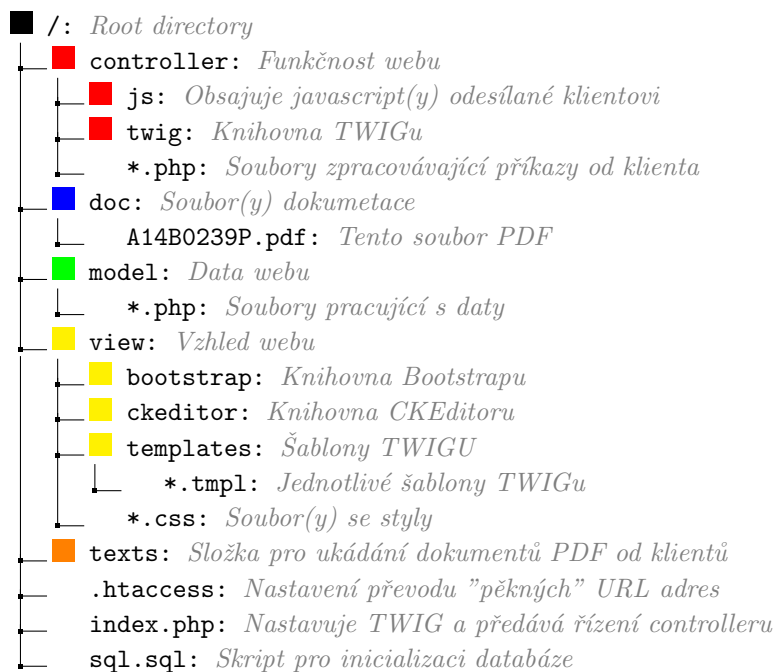


Figure 1: Adresářová struktura

### 3 Instalace aplikace

Instalace aplikace se skládá z několika kroků. Na některých zařízeních již můžou být první tři body hotovy, proto je to potřeba zkontrolovat.

Pro některé platformy je dostupný balík (LAMP, XAMP, WAMP, ...), který spojí první tři body instalace do jednoho bodu.

- **Instalace HTTP serveru**

Prvním krokem je instalace HTTP serveru. Jedním z nich je server Apache. Jak takový server nainstalovat lze pro danou platformu najít na internetu.

- **Instalace PHP**

Dalším krokem je instalace PHP. Návodů na konkrétní platformy lze opět nalézt na internetu.

- **Instalace MySQL**

Dále je třeba nainstalovat databázový server MySQL. Návodů na instalaci jsou opět na internetu.

- **Vytvoření nové databáze**

Poté je potřeba vytvořit na databázovém serveru prázdnou databázi pro tuto aplikaci, jméno je libovolné, stejné se pak nastaví v bodě "Nastavení aplikace".

- **Spuštění MySQL skriptu**

Po vytvoření databáze spustíme v této databázi skript `sql.sql`, který se nachází v kořenovém adresáři projektu.

- **Zkopírování celého projektu**

Poté můžeme celý projekt zkopírovat na webový server.

- **Nastavení aplikace**

Na řádcích 35 - 38 souboru `model` → `database_pool.php` je potřeba nastavit přihlašovací údaje k databázovému serveru a název databáze. Stačí přepsat defaultní nastavení na těchto řádcích.

- **Nastavení práv**

Složka `texts` musí mít nastavené oprávnění, aby do ní mohl webový server zapisovat. Pokud tomu tak není, je potřeba to nastavit.

Při spuštění aplikace neexistuje žádný uživatel, proto první vytvořený dostane nejvyšší oprávnění, další pak běžné oprávnění.

## 4 Architektura aplikace

Základní funkcionalitu popisuje diagram 2. Celá aplikace funguje dvěma způsoby. Při aktivním javascriptu se nahrazují asynchronně pouze části stránky, zatímco když uživatel javascript deaktivuje, stránky se načítají celé. To určuje podmínka *part\_only()*.

První podmínkou v diagramu 2 je získání parametru z URL. Pokud zde žádný parametr není, pak se nastaví parametr *index*.

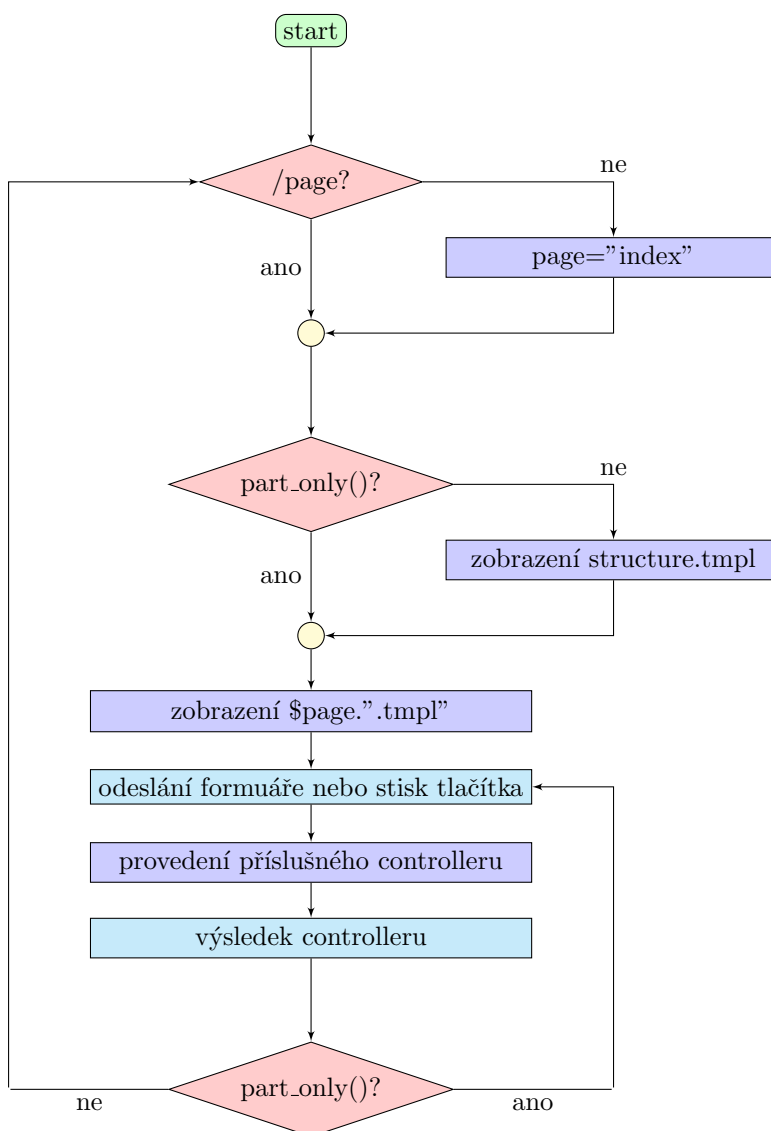


Figure 2: Vývojový diagram