



# KIV/WEB

Konference člověk a příroda

**utor:** Jakub Vítek

**Email:** [viteja@students.zcu.cz](mailto:viteja@students.zcu.cz)

**Odkaz:** <http://students.kiv.zcu.cz/~viteja/kiv-web-sp/>

**Datum:** 6.1.2018

**1 OBSAH**

**1 OBSAH**

**2**

## Popis technologií

Aplikace byla vytvořena pro webový server Apache a MySQL databáze a pro svůj běh potřebuje podporu jazyka PHP minimálně ve verzi 5.6.30. Další podporovanou verzí je PHP 7.0.

Pro snazší správu knihoven a jejich závislostí byla použita programová utilita s názvem Composer. Tato utilita také aplikaci poskytuje načítání tříd vyhovující standardu PSR-4.

Pro sestavení a vykreslení HTML výstupu aplikace je použit šablonovací systém Twig. Ten je vyvíjen spolu s frameworkem Symfony. Z tohoto frameworku jsou však použity pouze části nutné k funkčnosti šablonovacího systému. Systém je tedy v aplikaci použit relativně nezávisle na jmenovaném frameworku.

Aplikace dále využívá značkovací jazyk HTML5, stylovací jazyk CSS3 a scriptovací jazyk JavaScript spolu s knihovnou jQuery. Responzivita webové stránky je zajištěna použitím front-end frameworku Bootstrap 4.

## Popis adresářové struktury

- **app** - složka s logikou a řízením aplikace
  - Controllers - Složka s kontrolery aplikace
  - Models - Složka s modely aplikace
  - Routing - Složka se zdrojovým kódem pro routovací systém aplikace
  - Utils - Složka s náhodnými třídami, které se jinem nevešly
  - Views - Složka se třídami zajišťující vykreslení šablon
- **resources** - složka s šablonami, scripty, styly
  - css - kaskádové styly aplikace, CSS "knihovny"
  - json - javascript aplikace, JS knihovny
  - lang - nevyužitá složka
  - vendor - důležitější knihovny, které mají CSS i JS
  - views - Twig šablony
  -
- **routes** - složka se soubory definující použitelné url aplikace
  - web.php - soubor s definicí routes
- **upload** - složka pro data nahrána uživateli
- **vendor** - složka pro composer a jím vyžadované knihovny
- .htaccess - soubor definující přesměrování všech volání na index.php
- composer.json - soubor definující projekt a vyžadované knihovny pro composer
- composer.lock - soubor pro composer
- configuration.ini - soubor, ve kterém lze měnit nastavení aplikace
- index.php - vstupní bod aplikace
- LICENSE - MIT licence projektu
- README.md - Popis repozitáře na github.com
- database.sql - Instalační scripty pro MySQL databázi (struktura + data)
- readme.txt - popis adresářové struktury

# Architektura aplikace

Aplikace se co nejvíce snaží přiblížit obecně uznávanému modelu MVC. Aplikace je rozdělena tak, aby řízení aplikace za pomoci Kontrolerů, získávání a manipulace s daty za pomoci Modelů; a vykreslení výstupu uživateli do prohlížeče, byly navzájem oddělené.

## Kontrolery

V rámci aplikace existuje tzv. routovací systém (třídy App\Routing\Route a App\Routing\RouteValidator). Jedná se o jednoduchý systém, kdy je definována akce na základě volané URL adresy. Routovací systém vyhodnotí, zda existuje definice, která vyhovuje aktuální URL. Pokud daná definice existuje, je aktuálně požadovaná adresa zpracována pro všechny případné proměnné parametry. Následně je vyvolána metoda ve třídě určené samotnou definicí (routou). Routovací systém podporuje pěknou i starší URL, aplikace však byla vytvořena kvůli běhu na školním serveru pro URL starší.

### **příklad definice cesty:**

```
Route::get("index.php?page=about", "App\Controllers\AboutController@show");  
Route::get("/about", "App\Controllers\AboutController@show");
```

### **příklad definice cesty s parametrem:**

```
Route::get("/post/{id}", "App\Controllers\PostController@show", array("int"));  
Route::get("index.php?page=post&id={id}", "App\Controllers\PostController@show",  
array("int","string"));
```

### **příklad zpracování parametru**

Definice: Route::get("/post/{id}", "App\Controllers\PostController@show", array("int"));  
URL: <http://server.cz/post/10>  
Výstup: array("id" => 10);

Prakticky každá zobrazovaná stránka má svůj vlastní kontroler (například App\Controllers\LoginController). Ten přebírá jakékoliv parametry poslané routovacím systémem. Na jejich základě pak rozhoduje o načtení dat z jednotlivých modelů či manipulaci s danými daty, taktéž přes modely.

Načtená data jsou ověřována a nakonec předána šablonovacímu systému Twig (App\Views\Twig), který daná data použije v šabloně.

## Modely

Jakékoliv získávání či manipulace s daty probíhá přes Model třídy. Ty jsou většinou rozděleny podle toho, který objekt představují. Například třída `App\Models\User` řeší pouze věci, které náležejí k uživateli, kdežto třída `App\Models\Post` řeší všechno, co se týká uživateli napsaných příspěvků, atd.

## Pohledy

Jedná se o systém, který načte šablonu, vloží do ní data a na základě příkazů pro šablonovací systém a vložených dat je schopný při jedné definici dosáhnout jiného výstupu. To se obzvlášť hodí v případě, kdy je vytvářen výstup například uživatelských příspěvků. Díky tomuto systému je tedy možné definovat výstup, který mají jednotlivé příspěvky vytvořit pouze jednou a systém se pak postará o vykreslení všech příspěvků.

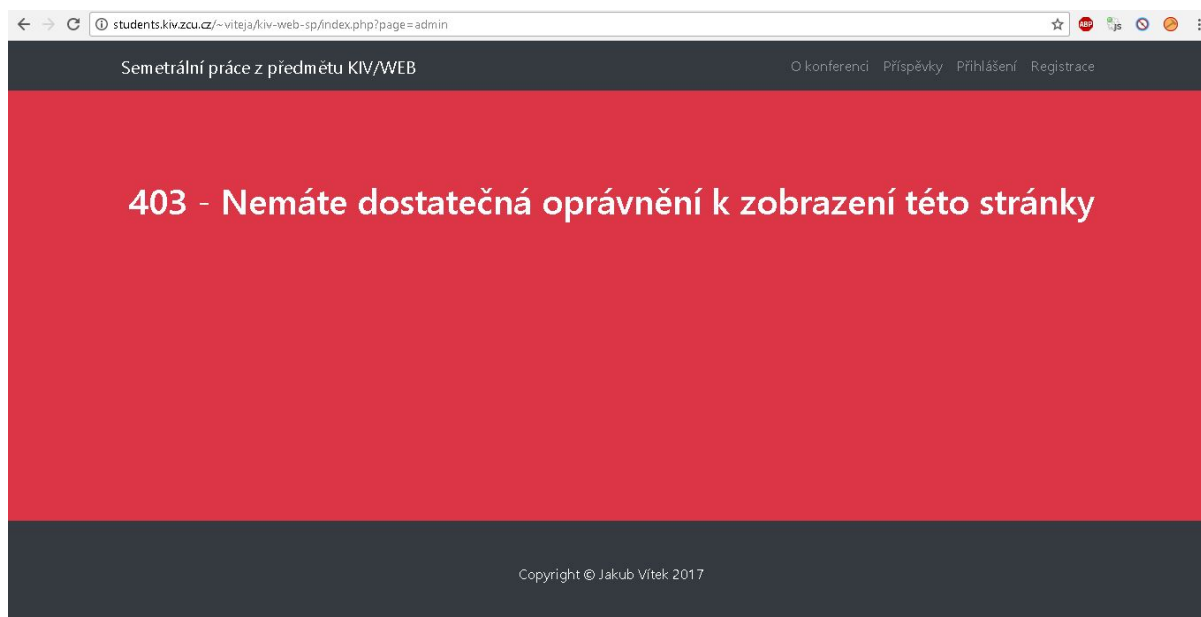
V rámci aplikace bylo implementováno pouze použití šablonovacího systému Twig, struktura aplikace a její dělení na vrstvy však podporuje případnou implementaci dalšího šablonovacího systému (např. Blade).

# Uživatelské rozhraní

Webové stránky jsou plně responzivní, lze je zobrazit jak na počítačích (s různými rozlišeními), tak i na mobilech a tabletech. Nezáleží, na jak obsáhlé stránce se uživatel pohybuje, v horní části stránky se vždy nachází hlavní menu, které se pohybuje zároveň s uživatelem.



Horní menu se také mění na základě přihlášeného uživatele. Administrátor bude mít vždy více odkazů než obyčejný autor či nepřihlášený uživatel. Avšak přístupová práva na zobrazení jsou kontrolována, obyčejné zjištění adresy a její napsání do adresového řádku uživateli nepomůže.



Některé stránky mají i sekundární menu, jedná se hlavně o stránky administrace a další podobné.

Semestrální práce z předmětu KIV/WEB

O konferenci Příspěvky Napsat příspěvek Účet Recenzent Administrace Odhlásit

## Konference - Člověk a příroda

[Administrace](#) [Uživatelé](#) [Příspěvky](#) [Recenze](#)

Přezdívka:	E-mail:	Oprávnění:	Akce:
Sognus	msognus@gmail.com	Administrátor	<a href="#">administrátor</a> <a href="#">recenzent</a> <a href="#">autor</a> <a href="#">vypnout</a> <a href="#">smazat</a>
Novak	marty@server.cz	Recenzent	<a href="#">administrátor</a> <a href="#">recenzent</a> <a href="#">autor</a> <a href="#">vypnout</a> <a href="#">smazat</a>
Zeleny	zeleny@server.cz	Recenzent	<a href="#">administrátor</a> <a href="#">recenzent</a> <a href="#">autor</a> <a href="#">vypnout</a> <a href="#">smazat</a>



## Závěr:

Aplikace by měla splňovat všechny požadavky, které na ni byly kladené. V aplikaci jsou obsaženy všechny uživatelské role. Na jejich základě se uživateli zpřístupňují jednotlivé sekce a s nimi spjaté akce.

Samotná aplikace mohla být navržena lépe, velice dlouho jsem strávil tvorbou routovacího systému a ve výsledku mi chyběly další části. Původní plán totiž byl takový, že jsem si chtěl sám vytvořit můj vlastní jednoduchý framework inspirovaný frameworkem Laravel, se kterým v současné době pracuji.

I tak se ale práce podařila dokončit a veškeré požadavky jsou funkční.

Co se týče zdrojových kódů, jsou k dispozici v mém github repozitáři:  
[https://github.com/Sognus/KIV\\_WEB\\_SP](https://github.com/Sognus/KIV_WEB_SP)