

**Česká zemědělská univerzita v Praze**

**Provozně ekonomická fakulta**

**Katedra informačních technologií**



**Bakalářská práce**

**Porovnání nejznámějších CMS systémů a tvorba  
internetové filmové databáze**

**Richard Markovič**

**© 2019/20 ČZU v Praze**

# ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Richard Markovič

Systemové inženýrství a informatika  
Informatika

Název práce

**Porovnání nejznámějších CMS systémů a tvorba internetové filmové databáze**

Název anglicky

**Comparison of the best known CMS systems and creation of internet film database**

---

### Cíle práce

Tato bakalářská práce si jako hlavní cíl stanovuje analýzu nejpůvodnějších CMS systémů, které jsou Joomla, Drupal a WordPress. Prostřednictvím metody AHP, bude z několika hledisek zjištěno, který systém je pro uživatele nejvhodnější. Na tomto systému pak bude realizována internetová aplikace filmové databáze, která mimo jiné umožní uživatelům hodnotit, přidávat a upravovat filmy. Aplikace bude dále také umožňovat doplnění, úpravu a administraci částí systému dle požadavků zadavatele.

### Metodika

Předkládaná bakalářská práce bude složena ze dvou teoretických a jedné praktické části, přičemž první teoretická část bude založena na sběru informací a informačních zdrojů, které se zabývají definicí CMS a také otázkou, jaké CMS jsou aktuálně nejpoužívanější a nejvyhledávanější uživateli.

Druhá teoretická část bakalářské práce bude zaměřena na otázku hodnocení popularity nejznámějších CMS systémů, jako jsou Joomla, Drupal a WordPress. Na základě zjištěných rešeršních údajů bude prostřednictvím metody AHP zjištěno, který systém je vhodný pro realizaci případové studie, tedy vytvoření filmové databáze.

Třetí, praktická část bakalářské práce se bude věnovat vlastní tvorbě internetové filmové databáze pomocí wireframe návrhů HTML šablon které budou implementovány do CMS systému. V uvedeném systému budou vytvořeny vlastní zdroje dat, které budou sloužit k realizaci uživatelských požadavků. Na základě syntézy teoretických poznatků a výsledků praktické části budou formulovány závěry práce.

### Doporučený rozsah práce

30 – 40 stran

### Klíčová slova

CMS, Plugin, HTML5, CSS3, Bootstrap, JavaScript, jQuery, AHP

---

### Doporučené zdroje informací

- Custom Post Types [online], 2019. <https://developer.wordpress.org>: developer.wordpress. [cit. 2019-08-27]. Dostupné z: <https://developer.wordpress.org/plugins/post-types/>
- Database Description [online], 2019. <https://codex.wordpress.org>: codex.wordpress [cit. 2019-08-27]. Dostupné z: [https://codex.wordpress.org/Database\\_Description](https://codex.wordpress.org/Database_Description)
- KUDLÁČEK, Lubomír, 2010. WordPress: podrobný průvodce tvorbou a správou webů. Brno: Computer Press. ISBN 978-80-251-2734-6.
- Post Types [online], 2019. <https://wordpress.org/support/>: wordpress [cit. 2019-08-27]. Dostupné z: <https://wordpress.org/support/article/post-types/>
- ŠUBRT, T. *Ekonomicko-matematické metody*. Plzeň: Vydavatelství a nakladatelství Aleš Čeněk, s.r.o., 2015. ISBN 978-80-7380-563-0.
- Template Hierarchy [online], 2019. <https://developer.wordpress.org/>: developer.wordpress [cit. 2019-08-27]. Dostupné z: <https://developer.wordpress.org/themes/basics/template-hierarchy/>
- WILLIAMS, Brad, David DAMSTRA a Hal STERN, 2015. Professional WordPress: Design and Development. 3. 10475 Crosspoint Boulevard indianapolis in 46256: John Wiley & Sons, Incorporated. ISBN 9781118987247.
- WILLIAMS, Brad, Ozh RICHARD a Justin TADLOCK, 2011. Professional WordPress Plugin Development. 1. Indianapolis, IN 46256: John Wiley & Sons, Incorporated. ISBN 9780470916223.
- 

### Předběžný termín obhajoby

2019/20 LS – PEF

### Vedoucí práce

Ing. Petr Benda, Ph.D.

### Garantující pracoviště

Katedra informačních technologií

Elektronicky schváleno dne 14. 10. 2019

**Ing. Jiří Vaněk, Ph.D.**

Vedoucí katedry

Elektronicky schváleno dne 14. 10. 2019

**Ing. Martin Pelikán, Ph.D.**

Děkan

V Praze dne 24. 01. 2020

---

### **Čestné prohlášení**

Prohlašuji, že svou bakalářskou práci "Porovnání nejznámějších CMS systémů a tvorba internetové filmové databáze" 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 23.3.2020

---

### **Poděkování**

Rád bych touto cestou poděkoval vedoucímu bakalářské práce Ing. Petr Benda, Ph.D. za odborné rady při psaní bakalářské práce.

# Porovnání nejnámějších CMS systémů a tvorba internetové filmové databáze

## Abstrakt

Záměrem této bakalářské práce je porovnání nejpoužívanějších CMS systémů jako jsou Joomla, Drupal a WordPress. Vyhodnotit, který systém je nejpoužívanější a pomocí něj vytvořit aplikaci, která bude sloužit jako internetová filmová databáze. K sestavení aplikace bude využita knihovna Bootstrap, která byla zvolena především kvůli jednoduchému grid systému. Internetová filmová databáze bude splňovat všechny požadavky plně responzivního webu. Na její tvorbu bude použit značkovací jazyk hypertext markup language ve verzi 5, zkráceně HTML5. Také budou přidány vlastní kaskádové styly ve verzi 3, zkráceně CSS3.

V teoretické části bude vysvětleno, co je to redakční systém CMS, shrnuty nejpoužívanější systémy, jejich výhody a nevýhody. Bude provedeno porovnání CMS systémů z pohledu obchodu, který se snaží minimalizovat náklady na vývoj webové aplikace. Podrobněji bude popsán redakční systém, který má nejširší zastoupení na trhu. Práce se bude soustředit na architekturu systému, způsob ukládání příspěvků, databázový návrh a systém ukládání šablon.

V praktické části bude použita metoda AHP, která vybere kompromisní variantu CMS systému. V systému, který bude zvolen pomocí metody AHP, bude vytvořena internetová filmová databáze. Po výběru CMS se realizují drátové modely, na nichž bude ukázán způsob kódování šablon a implementace html šablon na redakční systém. Dále bude do systému naprogramován vlastní modul, díky němuž bude přihlášený uživatel schopen hodnotit filmy. Na konci praktické části bude vytvořena vazební tabulka, pomocí níž se v systému propojí filmy, herci a režiséři.

**Klíčová slova:** AHP, PHP, SQL, HTML5, CSS3, Bootstrap, Joomla, Drupal, WordPress, Plugin

# Comparison of the best known CMS systems and creation of internet film database

## **Abstract:**

The purpose of this bachelor thesis is to compare the most frequently used CMS systems, such as Joomla, Drupal and WordPress, as well as to assess, which of the systems is the most used and with the use thereof to create an application that would serve as the internet film database. Bootstrap library was chosen mainly due to its simple grid system and will be used for assembly of the application. The internet film database shall fulfil all requirements of the fully responsive website. For the development, markup language hypertext of version 5, abbreviated as HTML5, will be used. Also, particular cascade styles of version 3, abbreviated as CSS3, will be added.

In the theoretical part, explanation of Content Management System (CMS) will be provided, as well as most used systems, their advantages and disadvantages. Also, CMS systems comparison will be offered from the standpoint of commerce subjects that strive to minimize costs on web application development. In more detail the Content Management System that has greatest market share will be described. The thesis will be focused mainly on the architecture of the system, style of input storing, database design and the system of saving templates.

In the practical part, AHP method will be used to choose a compromise variant of CMS system. Upon the system chosen using the AHP method internet film database will be created. Following the CMS selection, wire models will be realized, upon which the method of template coding will be shown, as well as implementation of html templates upon the Content Management System. Furthermore, a particular module will be programmed in the system, due to which the signed-in user will be able to rate movies. In the end of the practical part, the coupling table will be constructed, using which the movies, actors and directors will be interconnected within the system.

**Keywords:** AHP, PHP, SQL, HTML5, CSS3, Bootstrap, Joomla, Drupal, WordPress, Plugin

# Obsah

<b>1 Úvod.....</b>	<b>12</b>
<b>2 Cíl práce a metodika .....</b>	<b>13</b>
2.1 Cíl práce .....	13
2.2 Metodika .....	13
<b>3 Teoretická východiska .....</b>	<b>14</b>
3.1 Pojem redakční systém.....	14
3.1.1 Neznámější redakční systémy.....	14
3.1.2 Redakční systém Joomla.....	15
3.1.2.1 Výhody .....	15
3.1.2.2 Nevýhody .....	15
3.1.3 Redakční systém Drupal .....	15
3.1.3.1 Výhody .....	16
3.1.3.2 Nevýhody .....	16
3.1.4 Redakční systém WordPress.....	16
3.1.4.1 Výhody .....	17
3.1.4.2 Nevýhody .....	17
3.2 Architektura WordPress .....	18
3.2.1 Trojvrstvová architektura.....	18
3.2.1.1 Prezentáční vrstva.....	19
3.2.1.2 Aplikační vrstva.....	19
3.2.1.3 Datová vrstva.....	21
3.3 WordPress – vykreslení příspěvků.....	24
3.3.1 The Loop.....	24
3.3.2 WP_Query .....	24
3.4 WordPress hierarchie šablon.....	25
3.5 Možnosti rozšíření.....	29
3.6 Inicializace systému WordPress.....	29
3.7 Vodopád s iteracemi.....	30
3.7.1 Vodopádový model.....	30
3.7.2 Iterativní přístup.....	31
<b>4 Vlastní práce .....</b>	<b>32</b>
4.1 Porovnání redakčních systémů pomocí AHP.....	32
4.1.1 Saatyho metoda.....	33
4.1.2 Výběr kritérií.....	34



4.1.3	Metoda výpočtu .....	35
4.1.3.1	Výpočet vah.....	35
4.1.3.2	Výpočet hodnot pro varianty .....	39
4.1.4	Vyhodnocení výstupu z metody AHP .....	41
4.2	Vytvoření šablony .....	43
4.2.1	Vytvoření drátového modelu .....	43
4.2.1.1	Drátový model úvodní stránka nepřihlášený uživatel .....	44
4.2.1.2	Drátový model detail filmu .....	46
4.2.2	Nakódování šablony pomocí drátového modelu .....	48
4.2.2.1	Uspořádání prvků .....	49
4.2.2.2	Finální vizuální zobrazení šablon.....	50
4.3	Implementace šablon do CMS systému .....	52
4.4	Vytvoření CP, taxonomie a vazebních tabulek .....	55
<b>5</b>	<b>Závěr.....</b>	<b>62</b>
<b>6</b>	<b>Seznam použitých zdrojů .....</b>	<b>63</b>
	<b>Přílohy.....</b>	<b>66</b>
6.1	Šablona a přídatný modul.....	66

## Seznam obrázků

Obrázek 1	Rozložení adresářů do tří vrstev (vlastní tvorba) .....	19
Obrázek 2	Adresářová struktura WordPress.....	20
Obrázek 3	WordPress tabulka ze vztahem M:N (vlastní tvorba) .....	22
Obrázek 4	WordPress Databázový návrh (vlastní tvorba).....	23
Obrázek 5	WordPress Loop (vlastní tvorba) .....	24
Obrázek 6	Alternativní verze hlavního WP loopu za pomoci wp_query (vlastní tvorba) 25	
Obrázek 7	Schéma šablony (vlastní tvorba) .....	26
Obrázek 8	Šablonovaný systém ve WordPressu (Template Hierarchy, 2019).....	28
Obrázek 9	Vodopádový model (vlastní tvorba).....	31
Obrázek 10	Vodopádový model s iteracemi (vlastní tvorba) .....	32
Obrázek 11:	Hierarchické znázornění úrovní (vlastní výroba).....	33
Obrázek 12	Trend CMS systémů na Slovensku (Grafické znázornění – vlastní tvorba), data získána z (WordPress, Drupal, Joomla – Trendy Google, 2020).....	35
Obrázek 13	Radarový graf zjišťující dominanci kritérií.....	37
Obrázek 14	Koláčový graf vah stanovený Saatyho metodou (vlastní výroba).....	39
Obrázek 15	Stromová struktura výběru CMS systému pomocí metody AHP .....	42
Obrázek 16	Drátový model případ nepřihlášeného uživatele (vlastní tvorba) .....	44
Obrázek 17	Drátový model případ přihlášeného uživatele (vlastní tvorba) .....	45
Obrázek 18	Drátový model detail filmu, případ nepřihlášeného uživatele (vlastní tvorba) .....	46
Obrázek 19	Drátový model detail filmu, případ přihlášeného uživatele (vlastní tvorba). 47	
Obrázek 20	Hierarchie html tagů v šablonách (vlastní tvorba) .....	48

Obrázek 21 Šablona úvodní stránky filmové databáze, nepřihlášený uživatel (vlastní tvorba).....	50
Obrázek 22 Šablona úvodní stránky filmové databáze, přihlášený uživatel (vlastní tvorba) .....	51
Obrázek 23 Šablona detailu filmu na stránce filmové databáze, nepřihlášený uživatel (vlastní tvorba).....	51
Obrázek 24 Šablona detailu filmu na stránce filmové databáze, přihlášený uživatel (vlastní tvorba).....	52
Obrázek 25 Komentář v souboru style.css, nutný k aktivaci témata (vlastní tvorba).....	52
Obrázek 26 Soubor function.php 1/2 (vlastní tvorba) .....	53
Obrázek 27 Formulář pro registraci a přihlášení (vlastní tvorba).....	54
Obrázek 28 Soubor function.php 2/2 (vlastní tvorba) .....	55
Obrázek 29 Registrace přídatného modulu a úprava main query v souboru <i>ticket-movies-must-use.php</i> (vlastní tvorba).....	56
Obrázek 30 Přídatný modul <i>ticket-movies-must-use.php</i> (vlastní tvorba).....	57
Obrázek 31 Vazební tabulka sloužící k propojení filmů, herců a režisérů (vlastní tvorba) .....	58
Obrázek 32 Tabulka obsahující hodnocení filmů (vlastní tvorba) .....	58
Obrázek 33 Registrace vlastního typu příspěvku 1/2 (vlastní tvorba).....	59
Obrázek 34 Registrace vlastního typu příspěvku 2/2 (vlastní tvorba).....	60
Obrázek 35 Registrace vlastní taxonomie 1/2 (vlastní tvorba).....	61
Obrázek 36 Registrace vlastní taxonomie 2/2 (vlastní tvorba).....	61

## Seznam tabulek

Tabulka 1 CMS systémy a zvolená kritéria (WordPress Plugins WordPress.org, 2020), (Joomla! Extensions Directory, 2020), (Module project   Drupal.org, 2019).....	36
Tabulka 2 Změna povahy kritéria "pravidelná doba aktualizace" .....	36
Tabulka 3 Váhy stanovené Saatyho metodou .....	37
Tabulka 4 Saatyho matice pro kritérium <i>trend</i> .....	39
Tabulka 5 Saatyho matice pro kritérium <i>možnosti rozšíření</i> .....	40
Tabulka 6 Saatyho matice pro kritérium <i>podíl na trhu</i> .....	40
Tabulka 7 Saatyho matice pro kritérium <i>pravidelná doba aktualizace</i> .....	41
Tabulka 8 Pořadí CMS systémů s využitím metody AHP.....	41

## Seznam použitých zkratk

**CMS** (Content Management System) – Zkratka pro systém, který se stará o správu webového obsahu. V současnosti se CMS chápe jako webová aplikace

**ITC** (Information and Communication Technologies) – Všechny počítače nebo elektronické zařízení, která jsou připojená do internetové sítě a komunikují mezi sebou

**GPL** (General Public License) – Veřejná licence pro zdarma volně šiřitelný software, který vznikl pro projekt GNU

**GNU** – (*GNU's Not Unix!*) Z anglické věty „GNU není Unix!“ – jde o volně šiřitelný operační systém pro projekt GNU, který neobsahuje žádný zdrojový kód

**IRC** (Internet Realy Chat) – Internetový protokol určený pro textovou komunikaci

**AHP** (Analytic Hierarchy Process) – Analytická metoda párového porovnání využívaná při rozhodovacích úlohách

**PHP** (Personal Home Page) – Skriptovací jazyk určený pro programování dynamických stránek

**HTML** (Hypertext Markup Language) – Značkovací jazyk určený pro tvorbu webových stránek

**CSS** (Cascading Style Sheets) – V informatice se jedná o jazyk, který je zodpovědný za grafické vykreslení značkovacích jazyků

**WP** (WordPress) – CMS napsaný v jazyce PHP a MySQL

**FTP** (File Transfer Protocol) – Webový protokol, který je zodpovědný za přenos souborů na diskové prostory serverů

**UI** (User Interface) – Vizuální zobrazení webové aplikace

**CP** (Custom Post Type) – Vlastní typ příspěvku

# 1 Úvod

Dnes v roce 2020 je internet v technologicky vyspělých zemích již téměř součástí každé domácnosti. Vznik internetu se datuje k začátku padesátých let minulého století. Od té doby se základní myšlenka internetu rozrostla do zcela jiných podob a stalo se z něj prostředí, v němž je možné komunikovat, platit, objednávat produkty, hrát si nebo se věnovat jakékoliv jiné aktivitě.

Neoddělitelnou součástí těchto aktivit jsou webové aplikace, jimiž například mohou být ICT (Information and Communication Technologies), které mají za úkol poskytnout uživateli požadované informace.

Největší podkategorií informačních systémů jsou CMS (Content Management System) – jedná se o systémy, jejichž úlohou je spravovat obsah, jenž mohou využívat online obchody, zpravodajské portály, blogy nebo fóra. Na trhu existuje mnoho takovýchto systémů, největší zastoupení však mají systémy, které jsou pod licencí GPL, a to z důvodu jednoduché dostupnosti. Systémy s největším zastoupením na trhu jsou Joomla, WordPress a Drupal.

Ve své práci budu právě tyto CMS systémy mezi sebou porovnávat pomocí metody AHP. Kritéria systémů, které budou ohodnocené pomocí Saatyho škály jsou – trend, jímž se jednotlivé systémy ubírají, jak velkou disponují databází balíčků, které rozšiřují jejich funkcionalitu, podíl na trhu, neboli jaký podíl internetových aplikací běží na daných řešeních a bezpečnostní díry, neboli jak často jsou systémy aktualizované, aby zabránily útoku hackerů.

Postup zvolený v praktické části bakalářské práce je vodopádová metodika s iteracemi. Každá provedená sekvence bude iterativním způsobem konzultovaná s vedoucím práce.

Systémy Joomla, WordPress a Drupal budou vzájemně porovnány. Na CMS, který vybere metoda AHP, bude postavena internetová filmová databáze. V první části se vytvoří drátový model, který určí rozložení prvků, ve druhém kroku se nakódují všechny šablony. Na vybraný systém budou implementovány šablony a jako poslední krok bude naprogramovaná potřebná funkcionalita pro chod aplikace a budou též vytvořeny vazební tabulky, které obsahují hodnoty, jimiž jsou hodnocení filmů a kardinalitu mezi filmy, herci a režiséry.

## **2 Cíl práce a metodika**

### **2.1 Cíl práce**

Tato bakalářská práce si jako hlavní cíl stanovuje analýzu nejpoužívanějších CMS systémů, jimiž jsou Joomla, Drupal a WordPress. Prostřednictvím metody AHP bude z několika hledisek zjištěno, který systém je pro uživatele nejvhodnější. V tomto systému poté bude realizována internetová aplikace filmové databáze, která mimo jiné umožní uživatelům hodnotit, přidávat a upravovat filmy. Aplikace bude také dále umožňovat doplnění, úpravu a administraci částí systému dle požadavků zadavatele.

### **2.2 Metodika**

Předkládaná bakalářská práce bude složena z jedné teoretické a dvou praktických částí, přičemž první – teoretická – část bude založena na sběru informací a informačních zdrojů, které se zabývají definicí CMS a také otázkou, které CMS jsou aktuálně nejpoužívanější a nejvyhledávanější uživateli.

Druhá – praktická – část bakalářské práce bude zaměřena na problematiku hodnocení popularity nejznámějších CMS systémů, jako jsou Joomla, Drupal a WordPress. Na základě zjištěných rešeršních údajů bude prostřednictvím metody AHP zjištěno, který systém je vhodný pro realizaci případové studie, tedy vytvoření filmové databáze.

Třetí – praktická – část bakalářské práce se bude věnovat vlastní tvorbě internetové filmové databáze pomocí wireframe návrhů HTML šablon, které budou implementovány do CMS systému. V uvedeném systému budou vytvořeny vlastní zdroje dat, které budou sloužit k realizaci uživatelských požadavků. Na základě syntézy teoretických poznatků a výsledků praktické části budou formulovány závěry práce.

## 3 Teoretická východiska

### 3.1 Pojem redakční systém

Redakční systémy patří do kategorie ICT (Information and Communication Technologies). ICT představují všechny informační technologie, které zpracovávají data a pro erudované odborníky reprezentují informace, které slouží k dalšímu zpracování, popřípadě ke komunikaci. Velkou část ICT technologií tvoří webové aplikace, prostřednictvím nichž je uživatel schopen získat požadované informace bez nutné závislosti na operačním systému či instalace aplikace (Basl, 2012).

Nejrozšířenějšími webovými aplikacemi jsou redakční systémy, též známé pod zkratkou CMS (Content Management System) neboli synonymicky pojmenované jako redakční systémy. Hlavní výhodou CMS systémů je rychlá implementace blogů, online obchodů, online fór, wiki a prezentačních webů na produkční server, bez nutné znalosti programování. CMS systém sám o sobě poskytuje vlastníkově různou funkcionalitu, například možnost změny textů a elementů webových prvků bez nutného zásahu do kódu. Obrovskou výhodou redakčních systémů je také možnost připojení se do administračního rozhraní odkudkoliv a z jakéhokoliv zařízení, za předpokladu, že zařízení má přístup k internetu a webovému prohlížeči (Rockley, 2003).

Většina CMS systémů je postavena na architektuře složené ze tří vrstev. Jde o prezenční vrstvu, nacházející se na straně klienta, dále aplikační a datovou vrstvu, které jsou umístěné na straně serveru. Principem fungování trojvrstvé architektury jsou počítače (klienti), které se dotazují na data, většinou metodou GET nebo POST, přičemž dotaz je zpracován nějakým serverovým jazykem, jako například ASP a PHP na aplikační vrstvě, která si vyžádá požadovaná data z datové vrstvy a požadovaný výsledek je znovu přes aplikační část zaslán na prezenční vrstvu (Trojvrstvá architektura – Three-tier architecture, 2016)

#### 3.1.1 Neznámější redakční systémy

Většina aktuálně nejpoužívanějších CMS systémů je naprogramovaná v jazyce PHP s podporou databázového systému MySQL, jež jsou vydávány pod licenci GPL (General Public License), která dovoluje volné šíření a úpravu zdrojového kódu, avšak upravený kód musí být veden jako derivát pod značkou copyleft (What is Copyleft? - GNU Project - Free Software Foundation, 2018).

Výběr systémů, které mezi sebou budou porovnávány, je zvolen na základě statistik z webového portálu W3techs (Word Wide Web Technology Surveys), která vyhodnotila, že jedny z největších podílů na trhu v oblasti CMS systémů mají WordPress, Joomla a Drupal. Tyto systémy se těší přízni velké komunity uživatelů, kteří je používají, zlepšují a vytvářejí novou funkcionalitu (W3Techs.com - Usage Statistics and Market Share of Content Management Systems, February 2020, 2020; Chapman, 2011; Rai, 2019).

### 3.1.2 Redakční systém Joomla

Na systému Joomla běží významné značky, jako jsou MTV, Harvardská univerzita a mnohé jiné. Také má velké zastoupení ve společnostech, jež disponují vlastním intranetem, jako je například systém pro správu materiálu ve firmě. Systém Joomla má aktivní komunitu zahrnující více než 200 000 uživatelů, kteří se o systém určitým způsobem starají, například tvorbou rozšiřitelných balíčků, přispívají na vývoj nebo pomáhají méně zkušeným uživatelům řešit jejich aktuální problémy. Joomla CMS systém uvádí na svém webu 7 895 rozšiřitelných balíčků, které zvyšují funkcionalitu systému (DevriX, 2019; Schäferhoff, 2019).

#### 3.1.2.1 Výhody

- Obsahuje více než 7 000 tisíc balíčků rozšíření
- Velmi přehledně psaná dokumentace aktivní komunitou uživatelů
- Autorizace uživatele pomocí OpenID

#### 3.1.2.2 Nevýhody

- Architektura web template je pro prezenční weby velmi komplikovaná
- Nedostatek témat v porovnání s konkurencí
- Admin UI (User Interface) není na první pohled velmi intuitivní

(Burge, 2015; Kloostra, 2015).

### 3.1.3 Redakční systém Drupal

Na systému Drupal též běží velké množství známých webů, jako například MIT TESLA, nebo Sony Music či mnohé jiné. Samotný systém obsahuje velké množství funkcí,

kteře umožňují jednoduché vytváření webových prezentací a také úpravu/organizaci obsahu v nich.

Velké zastoupení má Drupal v systémech, které podporují otevřený textový protokol, jenž slouží ke komunikaci. Je znám jako IRC (Internet Relay Chat) a na tomto systému je postaveno velké množství diskusních fór. Dokumentace je tvořena komunitou, která se neustále rozšiřuje. Komunita dokumentaci postupně aktualizuje a vylepřuje. Dokumentace obsahuje všechny potřebné informace, například pro instalaci CMS systému, jeho modulů, návrhy na úpravu a hlášení chyb. Na oficiální stránce Drupalu je aktuálně v repozitáři uvedeno 33 011 modulů, z nichž se pravidelně aktualizuje 20 385. Moduly rozšiřují základní funkcionalitu systému bez nutnosti cokoliv programovat. Díky tomu se majitel webové prezentace může zaměřit na obsah a vzhled webové prezentace (Chapman, 2011; Devrix, 2019; Butcher, 2010; Beighley, 2010).

#### 3.1.3.1 Výhody

- Neustále narůstající komunita, její podpora a otevřený komunikační port IRC
- Více než 20 385 udržovaných modulů
- Vysoký počet komerčních společností, které pro systém Drupal poskytují podporu

#### 3.1.3.2 Nevýhody

- Architektura web template pro prezenční weby je velmi komplikovaná
- Nedostatečná počet šablon v porovnání s konkurencí
- Admin UI (User Interface) není na první pohled velmi intuitivní

(Best Content Management System – WordPress Vs Joomla Vs Drupal, b.r.)

#### 3.1.4 Redakční systém WordPress

WordPress vznikl jen za účelem vytvoření plně managovatelného blogového systému, avšak s postupným nárůstem popularity byla do systému zabudovaná taxonomie, díky čemuž se z obyčejného blogového systému stal CMS systém.

Dnes není pochyb o tom, že WordPress CMS systémem je, čehož důkazem je fakt, že na něm běží mnoho neblogových webových aplikací, jako například Whoocommerce – rozšíření, které změnilo systém na elektronický obchod, nebo že je pomocí rozšíření BuddyPress možné vytvořit z WordPressu sociální síť. WordPress sčítá velké množství



rozšiřitelných modulů, díky nimž je možné ze systému vytvořit jakoukoliv webovou aplikaci, kterou je možné vidět na internetu.

WordPress se těší z přízně největší komunity uživatelů, což znamená, že je poměrně jednoduché nalézt potřebné informace a návody. A jak již bylo uvedeno, původně vznikl jako blogovací systém, takže má již v základu integrované funkce pro správu vícero blogů pod jedním rozhraním (Best Content Management System – WordPress Vs Joomla Vs Drupal, b.r.; Kudláček, 2010).

#### 3.1.4.1 Výhody

- Z výše uvedených CMS systémů WP zahrnuje největší komunitu lidí a návodů
- Admin UI (User Interface) je již na první pohled velmi intuitivní, uživatel se rychle zorientuje a dokáže řídit správu obsahu
- Obsahuje nejvíce možností rozšíření a šablon, které je možné pár kliknutími aktivovat
- Na CMS systému WordPress je možné postavit víceméně jakýkoliv typ webu

#### 3.1.4.2 Nevýhody

- Obsahuje nadměrné množství funkcí, které často nebývají využity
- Velmi slabé zabezpečení základní instalace WordPressu, pro lepší bezpečnost je nutné doinstalovat rozšiřitelné moduly
- Aktuální verze textového editoru není příliš přehledná
- Systém nemá oficiální podporu. Absence oficiální podpory má za následek nesprávné postupy při řešení problémů

(Kudláček, 2010; WordPress, Drupal, Joomla – Trendy Google, 2020).

Jak již je uvedeno výše, jelikož WordPress disponuje největším podílem webů, které jsou postaveny na tomto typu architektury, je dobré ještě před výběrem systému, který bude při vytváření internetové filmové databáze nejvíce vyhovovat, podívat se na jeho architekturu. (W3Techs.com - Usage Statistics and Market Share of Content Management Systems, February 2020, 2020; Best Content Management System – WordPress Vs Joomla Vs Drupal, b.r.).

## 3.2 Architektura WordPress

WordPress je open source CMS systém, jenž je napsán v jazyce PHP a MySQL s aktuální verzí 5.3.2. Na oficiálních stránkách dokumentace WordPress se doporučuje mít vždy nainstalovanou nejaktuálnější verzi skriptovacího jazyka PHP a redakčního databázového systému MySQL nebo MariaDB. Aktuálně je doporučeno mít na serveru PHP ve verzi 7.3 a vyšší, MySQL 5.6 nebo MariaDB 10.1 a vyšší.

Doporučenými softwarovými servery, které běží na hardwarovém stroji, jsou Apache nebo Nginx. V případě, že je skriptovací jazyk i databázový systém v požadované verzi, je možné redakční systém nainstalovat (Kudláček, 2010; Leary, 2013; WordPress Web Hosting, 2019).

### 3.2.1 Trojvrstvá architektura

Trojvrstvá architektura bývá využívána v informačních systémech a jejím předchůdcem byla architektura dvojvrstvá, která pracuje na principu klient-server. Jádro WordPress systému je postaveno na třech vrstvách, které jsou:

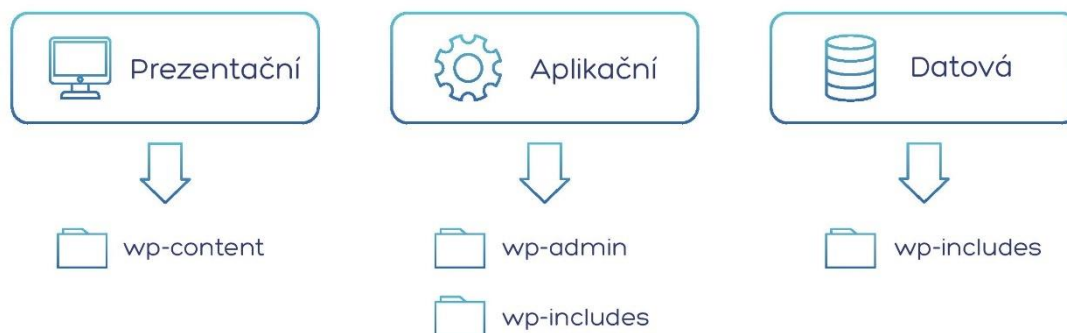
- prezenční
- aplikační
- datová

V systému WordPress je možné si představit jako prezenční vrstvu počítačový program, který slouží k prohlížení „World Wide Webu“ (web browse). Aplikační část je softwarový server, jenž běží na hardwarovém stroji s připojením k internetu, jako je například Apache, nebo Nginx s podporou PHP. Datová vrstva je tvořena databází, o níž se stará redakční systém určený na řízení dat (DBMS), jako je například MySQL či jiné. Hlavní výhodou architektury je nezávislost vrstev jedné na druhé, přičemž ty ve výsledku vytvoří jeden ucelený celek, který vidí uživatel (Trojvrstvá architektura – Three-tier architecture, 2016; Architektura klient-server – Client-server model 2016).

Po instalaci a nastavení PHP souboru wp-config.php nám systém vytvoří tři základní adresáře, jimiž jsou:

- wp-content
- wp-admin
- wp-includes

Skripty jsou rozděleny do výše uvedených tří adresářů, podle toho, s jakým obsahem pracují nebo jaká data zpracovávají. Není možné říci, že každý adresář v systému má na starosti přesně jednu vrstvu, ve zjednodušené formě je však možné si trojvrstvý model ve WordPressu představit tak, jak je zaznamenán níže na obrázku.



Obrázek 1 Rozložení adresářů do tří vrstev (vlastní tvorba)

### 3.2.1.1 Prezentační vrstva

Hlavním úkolem prezenční vrstvy je složení výsledné HTML šablony za pomoci PHP souborů. Tyto soubory se nacházejí ve složce *themes*, která je podadresářem složky *wp-content*. Nachází se tam soubory, jako je *header.php* nebo *footer.php* které jsou zahrnuty do *index.php* a dalších souborů tvořících téma. K výslednému sestavení šablony nám dopomáhá šablonovací systém, jenž je v systému WordPress zabudován. HTML šablony také obsahují kaskádové styly a javaskriptové knihovny, jako například jQuery.

S vykreslením HTML šablony se návštěvníkovi načítá i požadovaný obsah, který je uložen v databázi ve formě textu nebo cesty k obrázku a ten je z datové vrstvy vyžádaný pomocí aplikační vrstvy, která ho předá prezenční vrstvě a ta je schopná požadovaný text nebo obrázek vykreslit (Trojvrstvová architektura – Three-tier architecture; 2016; Leary, 2013).

### 3.2.1.2 Aplikační vrstva

Aplikační část, někdy též nazývaná middleware, zabezpečuje komunikaci mezi prezenční a datovou vrstvou. Ve WordPressu tuto úlohu plní několik set skriptů, které jsou, jak již bylo uvedeno výše, ve skriptovacím jazyce PHP. Skripty jsou rozdělené do tří složek, které byly vytvořeny po instalaci. Kromě složek, které jsou vytvořené při instalaci základního WordPressu, je také vytvořeno dalších sedmnáct skriptů, které zabezpečují funkčnost a plynulý chod systému. Příkladem může být skript *wp-config.php* nebo *wp-*

*config-sample.php*, v závislosti na tom, zda instalace proběhla v pořádku či nikoliv. V případě, že se instalace nezdařila, je nutné prostřednictvím FTP otevřít skript *wp-config-sample.php*, vyplnit jej správnými údaji a přejmenovat na název, který vyžaduje systém *wp-config.php*. Ve skriptu se nachází informace o připojení k databázi, absolutní cesta instalace WordPress nebo bezpečnostní klíče (Kudláček, 2010).

Skripty, které jsou obsahem složky *wp-admin*, zabezpečují správné vykreslení a funkčnost administračního rozhraní. Další složky, jako jsou *plugins* a *themes*, které se nacházejí ve složce *wp-content*, nesou obsah, jenž může být nainstalován přes administrační rozhraní nebo nahrán prostřednictvím FTP. Jedná se především o rozšíření funkcionality CMS systému prostřednictvím rozšiřitelných modulů, nebo změna vzhledu tématu. Složka *wp-includes* obsahuje především skripty, které slouží ke komunikaci s databází WordPress.

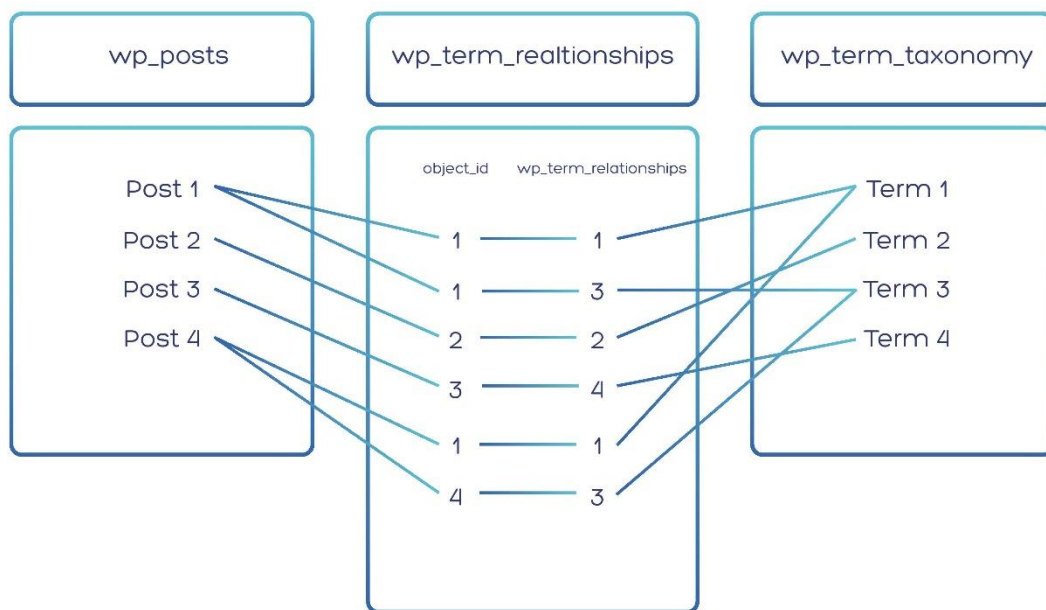


Obrázek 2 Adresářová struktura WordPress

### 3.2.1.3 Datová vrstva

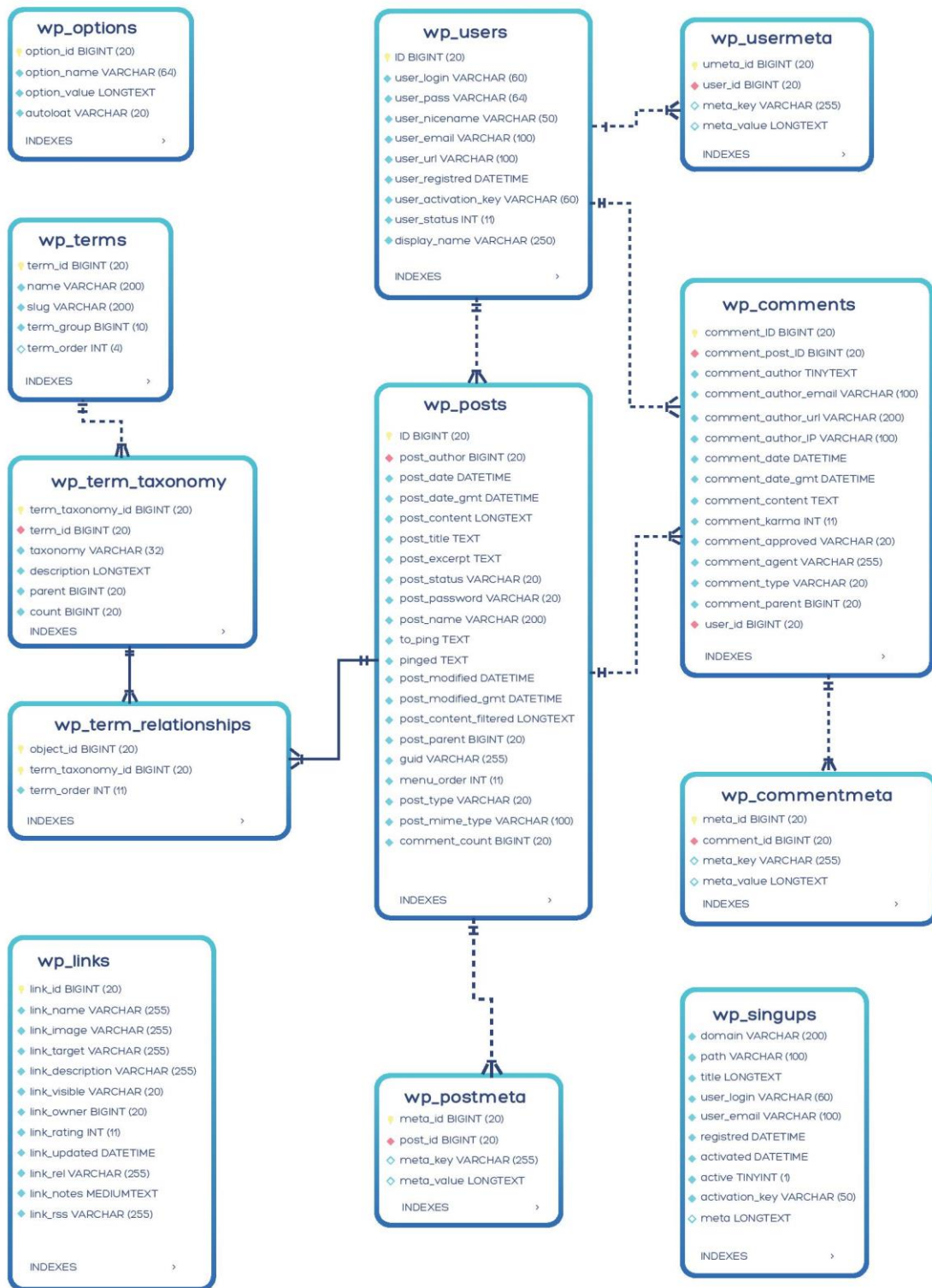
Jedná se o nejnižší vrstvu ve trojvrstevné architektuře. Úkolem této vrstvy je ukládání, čtení, mazání, aktualizace a zabezpečení konzistence dat. Redakční CMS systém WP využívá k řízení dat databázový systém MySQL, jenž byl vytvořen švédskou firmou MySQL AB. V současnosti je však systém vlastněn společností Oracle Corporation (WordPress Web Hosting, 2019).

Při standardní instalaci WordPressu vytvoří systém databázi, která obsahuje dvanáct tabulek s prefixem `wp_`. Tabulka, která obsahuje všechny *příspěvky*, *stránky*, *cesty k obrázkům*, *vlastní post type seznam menu položek*, má název `wp_posts`. Jedná se o tabulku, která je ve vztahu 1:N s tabulkami `wp_comments`, `wp_users`, `wp_term_relationships` a tabulkou `wp_postmeta`, jež obsahuje meta data o datech uložených ve `wp_posts` – jde o často využívanou tabulku k ukládání dat pro přídavné moduly, jakým je například ACF (Advanced Custom Fields). Tabulka `wp_comments` se stará o komentáře, jež bývají zobrazeny pod příspěvkem. Tabulka je propojena s `wp_commentmeta`, která také, stejně jako `wp_postmeta`, ukládá data o datech, tentokrát však o komentářích na stránce. Tabulka `wp_commentmeta` bývá často využívána pro přídavné moduly, které rozšiřují funkcionalitu komentářů o možnost hodnotit příspěvek číselně, hvězdičkou nebo jiným symbolem. Pro ukládání informací o registrovaných uživatelích, jako je například role uživatele, barevné schéma administračního rozhraní, osobní informace a mnohá jiná data, slouží tabulka `wp_users`, která má též propojení s tabulkou `wp_usersmeta`. Databázový vztah M:N je v systému WordPress použit jen jedenkrát, a to z důvodu vytvoření taxonomie, jíž může být například kategorie a štítky. Vztah M:N je možné popsat jako každý příspěvek můžeme mít N štítků/ kategorií a každá kategorie/štítek může mít M příspěvků. Tohoto vztahu se docílí pomocí vazební tabulky s názvem `wp_term_relationships`, která je spojena s tabulkami `wp_term_taxonomy` a `wp_posts`. V minulosti byla ještě propojena s tabulkou `wp_link`, avšak od verze 3.5 (Elvin Jones) byl tento vztah odstraněn. Tabulka `wp_term_relationships` obsahuje `post_id` a `term_id` pod názvy `object_id` a `term_taxonomy_id`, díky čemuž může tabulka obsahovat záznamy o příspěvcích a také záznamy o taxonomiích. (Codex: Database Description, b.r.).



Obrázek 3 WordPress tabulka ze vztahem M:N (vlastní tvorba)

Tabulka, která není propojena s ostatními, se nazývá *wp\_options*. V ní se nachází systémové informace a nastavení WordPressu, které je možné vidět v administračním rozhraní. Níže je uveden databázový návrh WordPress systému (Codex: Database Description, b.r.).



Obrázek 4 WordPress Databázový návrh (vlastní tvorba)  
(Codex: Database Description, b.r.)

## 3.3 WordPress – vykreslení příspěvků

### 3.3.1 The Loop

Pro vykreslení příspěvků nebo vlastních *post\_type* je v systému WordPress vytvořen speciální PHP cyklus, který se nazývá „The Loop“. Před proběhnutím WordPress cyklu je existence jeho obsahu zkontrolována příslušnou podmínkou. *have\_posts()*; – ta zjistí, zda se v databázi nachází nějaký příspěvek nebo vlastní *post\_type* pro nastávající stránku v případě, že je podmínka splněna, dostane se program do samotného těla cyklu, jenž při každé jedné iteraci zjistí, zda se v databázi ještě nachází nějaký *post\_type* a zavolá funkci *the\_post()*;, která nastaví globální proměnnou *wp\_query* na další příspěvek. Příspěvky je možné vykreslit jakýmkoliv HTML nebo PHP kódem při každé jednotlivé iteraci. Uvnitř WordPress cyklu jsou použity funkce, které se nazývají šablonovací tagy – jsou jimi:

- Titulek příspěvku (*the\_title()*; )
- Čas, kdy byl příspěvek zveřejněn (*the\_date()*; *the\_time()*; )
- Kategorie, do níž byl příspěvek zařazen (*the\_category()*; )

```
<?php if (have_posts()) : ?>
    <?php while (have_posts()) : the_post() ?>

    <?php endwhile ?>
<?php else : ?>
```

Obrázek 5 WordPress Loop (vlastní tvorba)

(The Loop | Theme Developer Handbook | WordPress Developer Resources, b.r.; Messenlehner, 2014)

### 3.3.2 WP\_Query

*WP\_Query* je třída, z níž je možné vytvořit instanci a tu používat napříč celým systémem WordPress. *WP\_Query* bývá využívána v případě, že neexistují žádné *conditional tags*. *Conditional tags* jsou ve WordPressu funkce začínající prefixem *is\_\**, jež v závislosti na souboru získají požadovaná data. V případě



tvorby *shortcode* nebo rozšiřitelného modulu není vždy možné použít soubor pro výpis správných dat, v případě využití *wp\_query* je možné požadovaná data získat a vypsat.

Druhým případem, kdy se využívá instance třídy *wp\_query*, je vytváření jiného než hlavního WordPress loopu, popřípadě vypsání jiného typu příspěvku, než je článek. Nejprve se na instanci *wp\_query* zavolá metoda *have\_posts()*, aby se zjistilo, zda je možné požadovaná data z databáze získat a postup cyklu je stejný, jako u hlavního WordPress loopu. V případě využití loopu prostřednictvím *wp\_query* je nutné na závěr zavolat funkci *wp\_reset\_postdata()*, aby bylo možné opět používat hlavní loop.

```
<?php

$the_query = new WP_Query( $args ); >

<?php if ( $the_query->have_posts() ) : >

    <?php while ( $the_query->have_posts() ) : $the_query->the_post(); >

    <?php endwhile; >

    <?php wp_reset_postdata(); >

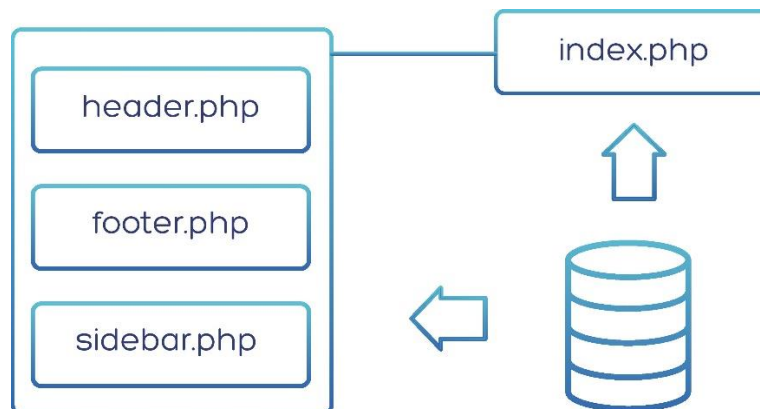
<?php else : >

<?php endif; >
```

Obrázek 6 Alternativní verze hlavního WP loopu za pomoci *wp\_query* (vlastní tvorba) (WP\_Query | Class | WordPress Developer Resources, 2017; Leary, 2013)

### 3.4 WordPress hierarchie šablon

Šablony jsou PHP soubory, které se používají ke generování webových stránek v systému WordPress, soubory jako *header.php*, *footer.php*, nebo *functions.php* se používají napříč všemi šablonami, některé šablony se však použijí jen za splnění určitých podmínek. Jak již bylo uvedeno výše, pro možnost aktivace témata v administraci stačí, aby téma obsahovalo jen dva základní soubory, a to *index.php* a *style.css*. (Kudláček, 2010; Template Hierarchy, 2019).



Obrázek 7 Schéma šablony (vlastní tvorba)

WordPress prohledává hierarchií souborů do té doby, dokud nenajde požadovaný soubor, v případě, že systém nenajde shodu, použije *index.php*. Nejpoužívanější soubory jsou:

- **archive.php** – soubor, který zajišťuje vykreslení starých příspěvků
- **single.php** – soubor, který vykreslí detail článku
- **page.php** – soubor sloužící k vykreslení statických stránek
- **singular.php** – v případě, že program v tématu nenajde soubory *single.php* nebo *page.php*, využije zástupný soubor *singular.php*
- **home.php** – soubor používaný k vykreslení domovské stránky témata, ať již jako statická stránka nebo nejnovější články na blogu
- **404.php** – v případě, že obsah na webu neexistuje, je použit tento soubor
- **search.php** – soubor používaný pro vykreslení výsledku hledání
- **category.php** – soubor používaný k vykreslení kategorií nacházejících se na blogu.

Téměř každý z výše uvedených souborů má ještě své specifitější varianty k zachycení všech možností. Jako příklad je možné uvést soubor *category.php*:

1. Systém se podívá na slug kategorie – v případě, že slug kategorie je „marvel“, WordPress systém hledá soubor, který má název *category-marvel.php*.
2. V případě, že téma neobsahuje požadovaný soubor *category-marvel.php*, zjistí WordPress systém ID, které obsahuje zadaná kategorie – v tomto případě například *category-1.php*.
3. V případě, že téma neobsahuje požadovaný soubor s názvem *category-1.php*, podívá se systém WordPress, zda téma obsahuje *category.php*.

4. V případě, že téma *category.php* neobsahuje, WP systém hledá soubor *archive.php*.
5. Pokud neexistuje ani *archive.php*, systém použije záložní soubor, který je používán vždy, když není nalezen nějaký soubor *index.php*.

Níže je uveden obrázek popisující celou hierarchii souborového systému (Template Hierarchy, 2019).



Obrázek 8 Šablonovaný systém ve WordPressu (Template Hierarchy, 2019)

### 3.5 Možnosti rozšíření

Možnosti rozšíření, nazývané též pluginy, jsou PHP skripty, které umožňují systému WordPress rozšířit jeho funkcionalitu bez nutnosti zasahování do jeho jádra, nebo kompletně změnit funkcionalitu blogového systém na sociální síť (BuddyPress), popřípadě na online obchod (WooCommerce). Hlavními výhodami používání rozšiřitelných modulů jsou:

- **Použitelnost funkcí**, které jsou již vestavěné v základní verzi systému WordPress. Jako například *template conditionals*, jejichž jednou z vlastností je zjišťovat, na které podstránce se uživatel nachází a v případě, že nemá oprávnění, je možné ho přesměrovat na jinou stránku za pomoci WordPress funkce *template\_redirect()*;
- **Opětovná použitelnost** – modul je nezávislý na tématu, takže je možná jeho využitelnost napříč každým WP tématem. Modul se dá využít na jakékoliv stránce s WP systémem.
- **Jednoduchá instalace** – modul je možné nainstalovat třemi způsoby, dva z nich jdou přes administrační rozhraní. Prvním způsobem je stáhnout modul z WordPress serveru nebo manuálně stáhnout .zip archiv z oficiální stránky pro pluginy a nahrát tento soubor přes rozhraní, kde si systém sám soubor rozbalí a nainstaluje. Třetím způsobem je překopírování rozbaleného souboru do složky *wp-content/plugins*.
- **Automatická aktualizace** – v případě, že jsou moduly nainstalované z oficiálního serveru, systém sám rozpozná nejaktuálnější verzi a vyzve administrátora stránky k aktualizaci.
- **Sanbox** – zabezpečí, že v případě jakékoliv chyby se vypne modul, avšak chod stránky se zachová.

(Leary, 2013; Writing a Plugin WordPress Codex, b.r.; WordPress Plugins, 2010)

### 3.6 Inicializace systému WordPress

Při spuštění stránky se program nejdříve podívá do souboru *wp-config.php*, z něhož získá přístup k databázi a ostatním potřebným datům, poté se načtou všechny funkce systému

WordPress, které zabezpečují chod, po inicializaci funkcí se aktivují všechny moduly a na závěr se spustí jazykové mutace a šablona, která je na stránce použita.

### 3.7 Vodopád s iteracemi

Vodopád s iteracemi je poslední kapitola teoretické části bakalářské práce, která popíše postup, jímž je v bakalářské práci vypracovaná praktická část.

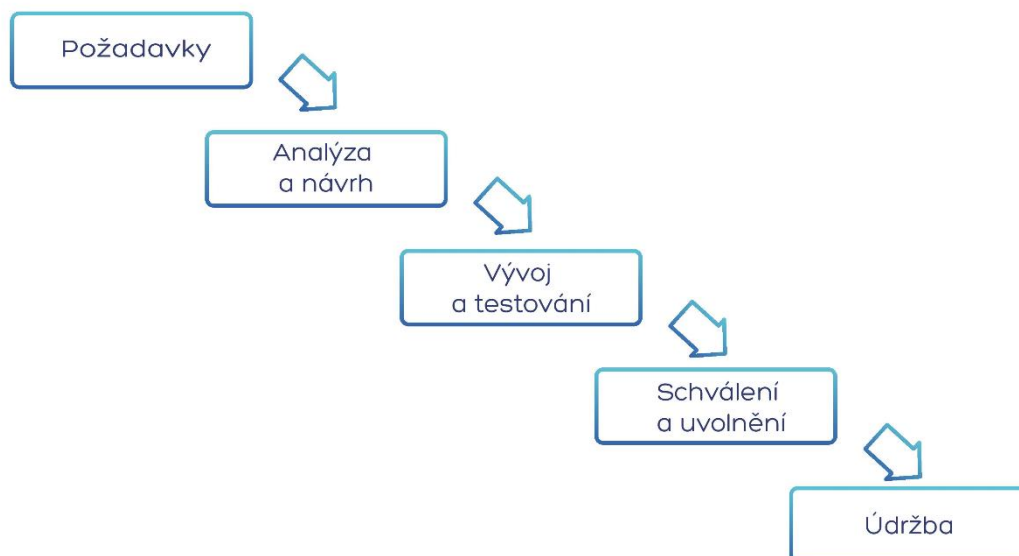
Internetová filmová databáze bude tvořena pomocí vodopádového modelu s iteracemi, též známého pod názvem spirálový přístup. Z důvodu dobře strukturovaného a jasně vymezeného cíle není nutné použít agilní metodiku vývoje.

#### 3.7.1 Vodopádový model

Vodopádový model (anglicky „waterfall“) je založen na sekvencích, které se neustále sbíhají do toku. Původní návrh modelu byl publikován v roce 1970 a skládal se ze sedmi částí, jimiž jsou:

1. Specifikace požadavkem
2. Analýza
3. Návrh
4. Implementace
5. Testování
6. Uvedení do provozu
7. Údržba

Výše uvedených sedm částí se dá zkrátit na pět kroků, které popisuje obrázek níže.



Obrázek 9 Vodopádový model (vlastní tvorba)

Hlavním problémem vodopádového modelu je velmi špatná reakce na změny. Největší problém vzniká, je-li projekt odsouhlasen a výsledek neodpovídá představám zadavatele. Model totiž nedokáže reflektovat problémy, které nebyly předem zjištěné při analýze. Tento zásadní problém modelu se snaží řešit iterativní přístup (Vodopádový model – Waterfall model), 2015).

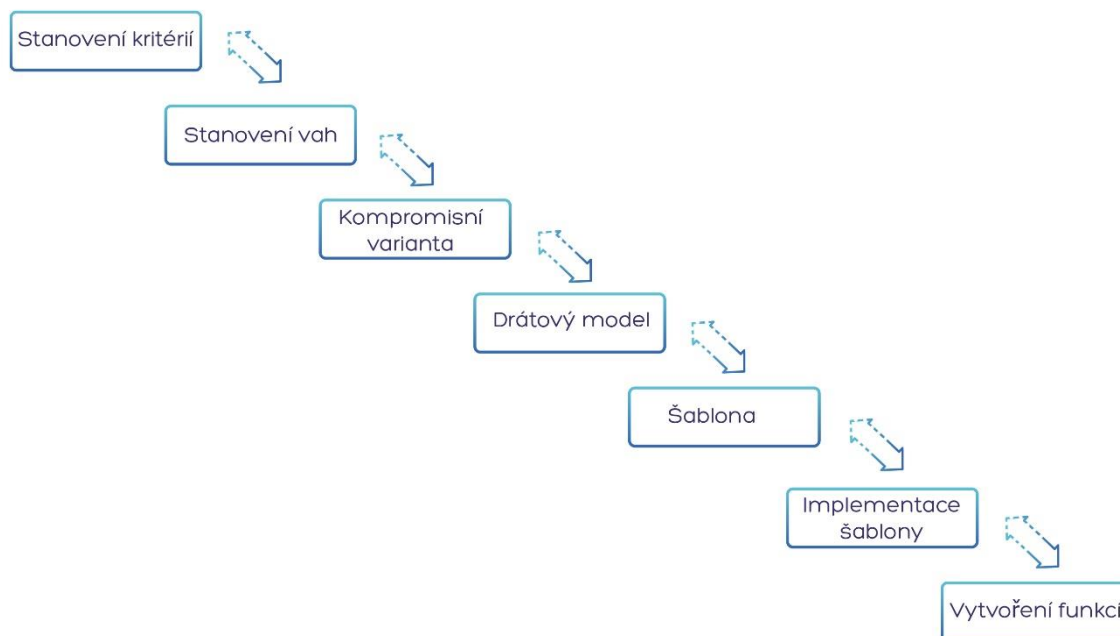
### 3.7.2 Iterativní přístup

Oproti vodopádovému přístupu je hlavním úkolem inkrementálního přístupu rozdělit problém na menší segmenty a zjednodušit je, díky čemuž je v případě nutnosti umožněno zavedení změn během projektu. Základním přístupem je rozložení každého problému do série menších vodopádů, které je nutno ukončit před pokračováním na další sekvenci (Larman, 2003).

Praktická část bakalářské práce bude mít sedm sekvencí, jimiž jsou:

1. Stanovení kritérií pro AHP model
2. Stanovení vah pomocí Saatyho metody
3. Výběr kompromisní varianty
4. Vytvoření drátového modelu
5. Nakódování šablon
6. Implementace šablon do systému
7. Vytvoření funkcí pro správný chod internetové filmové databáze

Vždy před přechodem k další sekvenci bude iterativním způsobem konzultovaná stávající sekvence s vedoucím bakalářské práce. Postup při vývoji je vyobrazen na níže uvedeném obrázku.



Obrázek 10 Vodopádový model s iteracemi (vlastní tvorba)

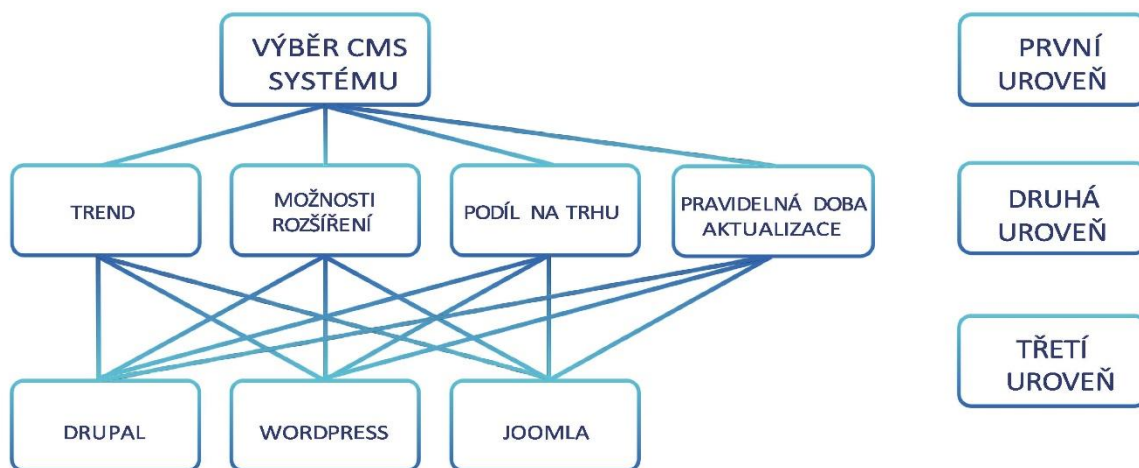
## 4 Vlastní práce

### 4.1 Porovnání redakčních systémů pomocí AHP

Pro začátek je nutno určit, který CMS systém je pro praktickou část bakalářské práce nejvhodnější. K porovnání CMS systémů použijeme metodu AHP (Analytický hierarchický proces). Vznikla v 80. letech minulého století na Wharton School of the University of Pennsylvania a jejím zakladatelem byl prof. Saaty. Účelem AHP metody je urychlit proces rozhodování. Prostřednictvím postupného rozkladu komplikovaného, nedobře strukturovaného problému je možné jej rozložit na dílčí složky, pomocí čehož je možné složení systému odstupňovat. Pod pojmem odstupňované složení je možné si představit stromovou strukturu, většinou o třech úrovních (Šubrt, 2015).



- **První úroveň**, cíl vyhodnocení – v tomto případě výběr CMS systému
- **Druhá úroveň**, vyhodnocovací kritéria – v tomto případě se jedná o kritéria, jimiž jsou trendy, možnosti rozšíření, podíl na trhu a pravidelná aktualizace systému. Důvod pro volbu těchto kritérií je uveden v kapitole Výběr kritérií.
- **Třetí úroveň**, posuzované varianty – v tomto případě CMS systémy.



Obrázek 11: Hierarchické znázornění úrovní (vlastní výroba)

Metoda AHP v základu tvoří hierarchii tří úrovní, složitější úlohy mají ještě mezi cílem analýzy a kritériem úroveň subkritéria, které je složeno ze subjektivních hodnot expertů v dané problematice (Šubrt, 2015).

Jedná se o nadstavbu vícekritériálního rozhodovacího procesu. Základem systému je Saatyho metoda (více v podkapitole níže), která se použije pro každou část stromové struktury. Pomocí kvantitativního párového porovnání jsou subjektivně porovnané všechny varianty mezi sebou. Spojením těchto hodnot se získá hodnota s nejvyšší prioritou porovnávaných variant, na niž je zaměřena pozornost s cílem rozhodovaný problém vyřešit (Fotr, 2010).

#### 4.1.1 Saatyho metoda

Jedná se o jednu z metod, pomocí nichž je možné určit váhy kritérií a vrátit informaci v kardinální podobě – metoda je doporučena v případě, že hodnotí jen jedna osoba na základě subjektivního názoru. Jedná se o kvantitativní párové porovnání, přičemž kritéria se porovnávají pomocí Saatyho škály (Šubrt, 2015; Fotr, 2010).

Počet bodů:

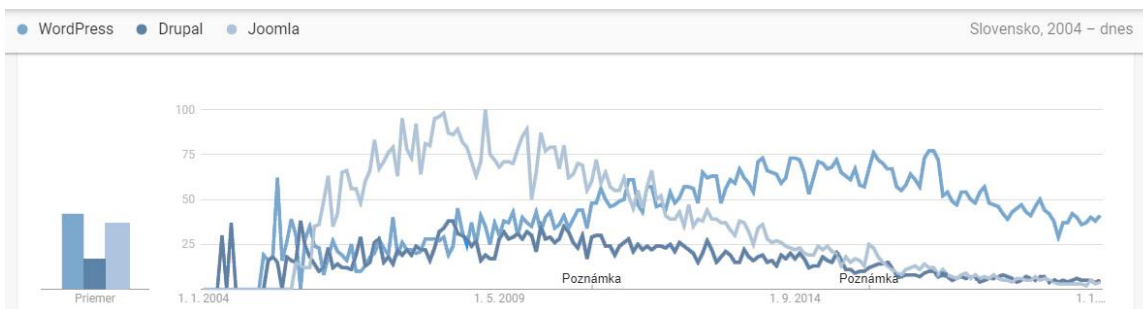
- 1 kritéria mají totožný význam
- 3 kritérium má o něco větší význam než druhé
- 5 kritérium má větší význam než druhé
- 7 kritériu má mnohem větší význam než druhé
- 9 kritérium má absolutně větší význam než druhé

Standardně se používá devítibodová stupnice, je však možné použít i hodnoty mezi (2, 4, 6, 8). Při kvantitativním párovém porovnání všech kritérií vznikne čtvercová matice  $n \times m$ , která vyjadřuje podíl vah  $i$ -tého a  $j$ -tého kritéria, na diagonále je vždy hodnota 1, a to z toho důvodu, že se porovnávají stejná kritéria. Po vytvoření čtvercové matice je vypočítána hodnota každého řádku pomocí geometrického průměru, který se následně znormalizuje, a jsou získané požadované váhy.

#### 4.1.2 Výběr kritérií

Kritéria pro zvolení správného CMS systému jsou volena z pohledu manažera nebo majitele firmy. Hlavní motivací je maximalizovat zisk a minimalizovat náklady. Kritérium cena v případě uvedených CMS systémů nemá žádný význam, jelikož každý systém je vydáván jako *open source* a je zdarma. Vybranými kritérii jsou nejnovější trendy systémů, velikost modulů, které se již v systémech nachází, zastoupení, které má ten který systém na trhu – tj. kolik projektů je na platformě postaveno – a samozřejmě frekvence nových aktualizací, které zajišťují bezpečnost systému.

**Trendy** – pomocí nástroje *google trends* jsou získaná data, která jsou v poměru k hodnotě 100. Subjekt může na základě vybraných údajů vidět, jak se zájem o CMS systémy měnil až do aktuálního data. Na základě tohoto kritéria je možné určit, jak velký je zájem o dané systémy. Obecně je možné říci, že čím větší je o systém zájem, tím dříve je možné najít osoby, které budou umět se systémem pracovat, doplňovat jej, aktualizovat obsah a provádět mnohé jiné úkony, díky čemuž se snižují náklady na zaškolování a v případě potřeby je možná i rychlá zaměnitelnost lidí. Obrázek níže ukazuje vývoj trendu na Slovensku v oblasti nejznámějších CMS systémů od roku 2004 (WordPress, Drupal, Joomla – Trendy Google, 2020).



Obrázek 12 Trend CMS systémů na Slovensku  
(Grafické znázornění – vlastní tvorba), data získána z (WordPress, Drupal, Joomla – Trendy Google, 2020)

**Možnosti rozšíření** – jedná se o naprogramované moduly, pomocí nichž je možné rozšířit základní chování CMS systému o novou funkcionalitu. Kritérium rozšiřitelnosti umožňuje v případě jednoduchých funkcí a opakujících se problémů šetřit náklady na programátorské hodiny (WordPress Plugins WordPress.org, 2020; Joomla! Extensions Directory, 2020; Module project | Drupal.org, 2019).

**Podíl na trh** – kritérium, které poukazuje na to, kolik online webů a aplikací je postaveno na vybraných systémech. Výběr kritéria byl proveden z důvodu nutnosti rychle najít developery, kteří jsou schopni rozšiřovat funkcionalitu daných systémů. Podobně jako u kritéria *trendy* je možno obecně konstatovat, že čím vyšší podíl má daný systém na trhu, tím více developerů jej umí dále vyvíjet (Rai, 2019).

**Aktualizace systému** – velkým problémem populárních systémů je též velký zájem hackerů, kteří hledají bezpečnostní díry a v případě, že je najdou, je aplikace nebo webová stránka v potenciálním ohrožení. Z těchto důvodů je nutná pravidelná aktualizace, která potenciální díry opravuje (Schäferhoff, 2019).

Významem této kombinace kritérií je zvolení systému, který se nejvíce hodí pro vytvoření internetové filmové databáze s tím, aby v případě problémů byla k dispozici dostatečně rozsáhlá komunita uživatelů, kteří budou ochotní a schopní poradit, který má nízkou časovou náročnost na instalaci CMS systému a obsáhlou databázi pomocných modulů, která napomůže budoucí funkcionalitě webového portálu.

### 4.1.3 Metoda výpočtu

#### 4.1.3.1 Výpočet vah

V předchozí kapitole jsou definována relevantní kritéria a je objasněn důvod pro výběr daných kritérií. Níže uvedená tabulka obsahuje data, která jsou vázaná na kritéria, jež jsou zvolena pro další zpracování.

	Trend	Možnosti rozšíření	Podíl na trhu	Pravidelná doba aktualizace systému
DRUPAL	17%	33 011	7%	30
WORDPRESS	44%	54 795	59%	60
JOOMLA	37%	7 895	5%	40
POVAHA	max	max	max	min

Tabulka 1 CMS systémy a zvolená kritéria

(WordPress Plugins WordPress.org, 2020; Joomla! Extensions Directory, 2020; Module project | Drupal.org, 2019).

Před výpočtem vah je nutné zjistit, zda některá z vybraných variant nedominuje nad jinou. V tabulce otištěné výše se v posledním řádku nachází *povaha* kritéria. První tři kritéria mají povahu maximalizačního charakteru, avšak čtvrté (poslední) kritérium *pravidelná aktualizace* je minimalizační. Před zjištěním informace, zda některá z variant dominuje nad jinou, je nutné povahu kritérií sumarizovat (Šubrt, 2015).

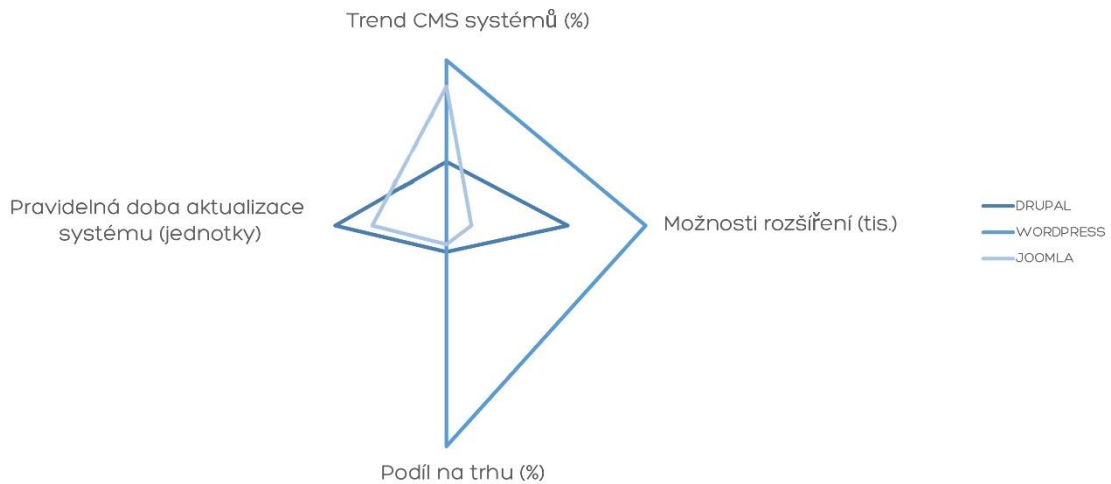
Při změně minimalizačního kritéria ve sloupci *pravidelná aktualizace* tabulky 1 na maximalizační je nutné najít největší číslo, které se nachází v daném sloupci a v každém řádku tohoto sloupce od něj odečíst stávající hodnotu. Ve výsledku jsou získané hodnoty pro řádek *drupal* 30, pro *wordpress* 0 a pro *joomla* 20. Hodnoty je možné vyčíst z tabulky níže.

	Trend (%)	Možnosti rozšíření (tis.)	Podíl na trhu (%)	Pravidelná doba aktualizace systému (jednotky)
DRUPAL	17	33	7	30
WORDPRESS	44	54	59	0
JOOMLA	37	7	5	20
POVAHA	max	max	max	max

Tabulka 2 Změna povahy kritéria "pravidelná doba aktualizace"

V případě, že všechna kritéria mají stejnou povahu, je možné zjistit, zda některá z vybraných variant dominuje nad jinou. Pomocí programu *excel* a dat z tabulky 2 je vykreslen radarový graf, z něhož je možné zjistit, zda některá z variant dominuje nebo zda je dominovaná.

## Dominance podle stavu okolností



Obrázek 13 Radarový graf zjišťující dominanci kritérií

Z výše uvedeného grafu je možné vyčíst, že žádná varianta CMS systémů nad jinou variantou nedominuje. Inverzně též platí, že žádná varianta není dominovaná.

Nadcházejícím krokem na základě subjektivního názoru hodnotitele je vytvoření tabulky, do níž jsou přiřazovány hodnoty ze Saatyho hodnotící škály.

	Trend (%)	Možnosti rozšíření (tis.)	Podíl na trhu (%)	Pravidelná doba aktualizace systému (jednotky)	Geometrický průměr (b)	Váhy (v)
Trend (%)	1	1/7	1/3	1/5	0,31	0,06
Možnosti rozšíření (tis.)	7	1	1/3	1/3	0,94	0,17
Podíl na trhu (%)	3	3	1	9	3,00	0,56
Pravidelná doba aktualizace systému (jednotky)	5	3	1/9	1	1,14	0,21
					5,387717776	1

Tabulka 3 Váhy stanovené Saatyho metodou

Tabulku výše je možné interpretovat následujícími slovy:

- Kritérium *trend*
  - má mnohem menší význam než *možnosti rozšíření*
  - má o něco menší význam než *podíl na trhu*
  - má menší význam než *pravidelná doba aktualizace systému*.
- Kritérium *možnosti rozšíření*
  - má mnohem větší význam než *trend*
  - má o něco menší význam než *podíl na trhu*
  - má o něco menší význam než *pravidelná doba aktualizace systému*.
- Kritérium *podíl na trhu*
  - má o něco větší význam než *trend*

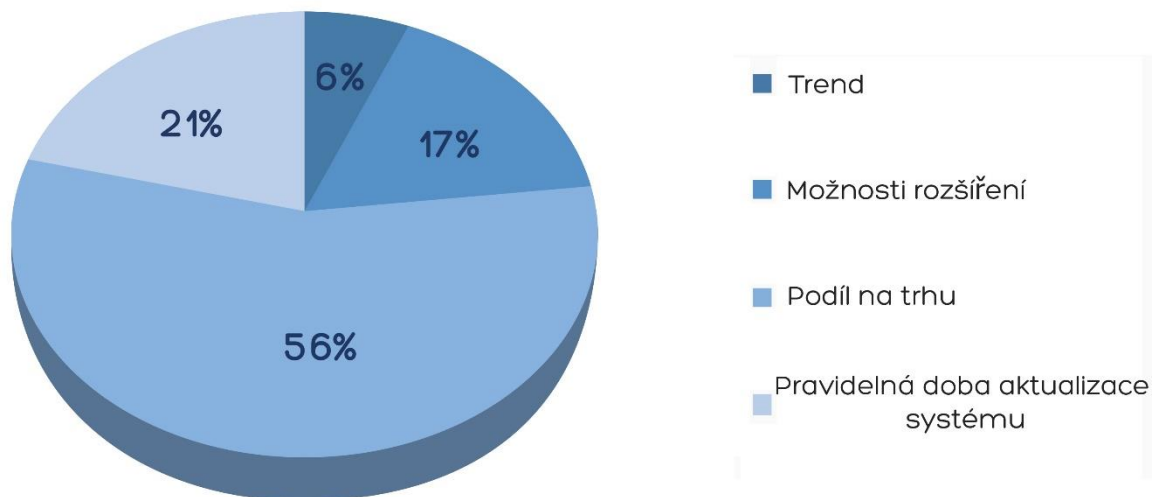
- má o něco větší význam než *možnosti rozšíření*
- má absolutně větší význam než *pravidelná doba aktualizace*.
- Kritérium *pravidelná doba aktualizace*
  - má o něco větší význam než *trend*
  - má o něco větší význam než *možnosti rozšíření*
  - má absolutně menší význam než *podíl na trhu*.

Diagonála ve výše uvedené tabulce obsahuje samé jedničky, a to proto, že tatáž varianta má vůči sobě samé tentýž význam. Po přiřazení hodnot do tabulky je nutné v každém řádku vyjádřit geometrický průměr, s čímž v *excelu* pomůže funkce *GEOMEAN()*. Jakmile je vypočítán geometrický průměr pro každou variantu, provede se v *excelu* jejich součet pomocí funkce *SUM()*. Geometrické výsledky variant se poté vydělí sumou geometrických průměrů. Výsledky jsou zaokrouhlené na dvě desetinná místa a kritéria získala váhy:

- trend: 0,06
- možnosti rozšíření: 0,17
- podíl na trhu: 0,56
- pravidelná doba aktualizace: 0,21

Největší význam je kladen u váhy *podíl na trhu* s hodnotou 0,56, druhý největší význam je kladen pro kritérium *pravidelná doba aktualizace* s hodnotou 0,21 a další kritérium v pořadí je *možnosti rozšíření*, přičemž nejmenší priorita náleží kritériu *trend* s hodnotou 0,06. Obrázek uvedený níže zobrazuje koláčový graf rozložení vah pomocí Saatyho metody.

# Grafické znázornění vah pomocí Saatyho metody



Obrázek 14 Koláčový graf vah stanovený Saatyho metodou (vlastní výroba)

## 4.1.3.2 Výpočet hodnot pro varianty

Po výpočtu druhé úrovně kritérií je nutné skrze subjektivní názor hodnotitele pomocí bodovací Saatyho škály přiřadit hodnoty a porovnat všechny varianty mezi sebou, a to na základě kritérií, které jsou na druhé úrovni stromové struktury.

Trend 0,06	DRUPAL	WORDPRESS	JOOMLA	Geometrický průměr (bi)	Normalizace	Váhy (vi)
DRUPAL	1	1/7	1/3	0,36	0,09	0,01
WORDPRES	7	1	3	2,76	0,67	0,04
JOOMLA	3	1/3	1	1,00	0,24	0,01
				4,121384301	1	0,058

Tabulka 4 Saatyho matice pro kritérium *trend*

Ve výše uvedené tabulce je možno vidět přerozdělení Saatyho hodnot na základě subjektivního názoru hodnotitele pro kritérium *trend*. Tabulku s kritériem *trend* je možné slovně interpretovat jako:

- Kritérium *trend* má pro systém *WordPress* mnohem větší význam než pro *Drupal*.
- Kritérium *trend* má pro systém *WordPress* o něco větší význam než pro *Joomla*.

- Kritérium *trend* má pro systém *Joomla* o něco větší význam než pro *Drupal*.

Možnosti rozšíření 0,17	DRUPAL	WORDPRESS	JOOMLA	Geometrický průměr (bi)	Normalizace	Váhy (vi)
DRUPAL	1	1/3	7	1,33	0,29	0,05
WORDPRES	3	1	9	3,00	0,66	0,11
JOOMLA	1/7	1/9	1	0,25	0,05	0,01
				4,577668216	1	0,1743

Tabulka 5 Saatyho matice pro kritérium *možnosti rozšíření*

Výše je možné v tabulce vidět přerozdělení Saatyho hodnot na základě subjektivního názoru hodnotitele pro kritérium *možnosti rozšíření*. Tabulku s kritériem *možnosti rozšíření* je možné slovně interpretovat jako:

- Kritérium *možnosti rozšíření* má pro systém *WordPress* o něco větší význam než pro *Drupal*.
- Kritérium *možnosti rozšíření* má pro systém *WordPress* absolutně větší význam než pro *Joomla*.
- Kritérium *možnosti rozšíření* má pro systém *Drupal* mnohem větší význam než pro *Joomla*.

Podíl na trhu 0,56	DRUPAL	WORDPRESS	JOOMLA	Geometrický průměr (bi)	Normalizace	Váhy (vi)
DRUPAL	1	1/7	3	0,75	0,15	0,08
WORDPRES	7	1	9	3,98	0,79	0,44
JOOMLA	1/3	1/9	1	0,33	0,07	0,04
				5,066337982	1	0,5568

Tabulka 6 Saatyho matice pro kritérium *podíl na trhu*

V tabulce je výše možné vidět přerozdělení Saatyho hodnot na základě subjektivního názoru hodnotitele pro kritérium *podíl na trhu*. Tabulku s kritériem *podíl na trhu* je možné slovně interpretovat jako:

- Kritérium *podíl na trhu* má pro systém *Drupal* o něco větší význam než pro *Joomla*.
- Kritérium *podíl na trhu* má pro systém *WordPress* mnohem větší význam než pro *Drupal*.
- Kritérium *podíl na trhu* má pro systém *WordPress* absolutně větší význam než pro *Joomla*.



Pravidelná doba aktualizace 0,21	DRUPAL	WORDPRESS	JOOMLA	Geometrický průměr (bi)	Normalizace	Váhy (vi)
DRUPAL	1	7	5	3,27	0,71	0,15
WORDPRES	1/7	1	1/5	0,31	0,07	0,01
JOOMLA	1/5	5	1	1,00	0,22	0,05
				4576777019	1	0,2109

Tabulka 7 Saatyho matice pro kritérium *pravidelná doba aktualizace*

Ve výše uvedené tabulce je možné vidět přerozdělení Saatyho hodnot na základě subjektivního názoru hodnotitele pro kritérium *pravidelná doba aktualizace*. Tabulku s kritériem *pravidelná doba aktualizace* je možné slovně interpretovat jako:

- Kritérium *pravidelná doba aktualizace* má pro systém *Drupal* mnohem větší význam než pro *WordPress*.
- Kritérium *pravidelná doba aktualizace* má pro systém *Drupal* větší význam než pro *Joomlu*.
- Kritérium *pravidelná doba aktualizace* má pro systém *Joomla* větší význam než pro *WordPress*.

Pro každou variantu v Saatyho matici je vypočítán geometrický průměr. Součet všech geometrických průměrů v dané matici vytvoří dělitele, jímž jsou vydělené geometrické průměry. Výsledky po dělení vrátí normalizované tvary, které jsou vynásobeny vahou svého kritéria. Výsledkem jsou váhy pro varianty, které se nacházejí na třetí úrovni stromové struktury v modelu AHP.

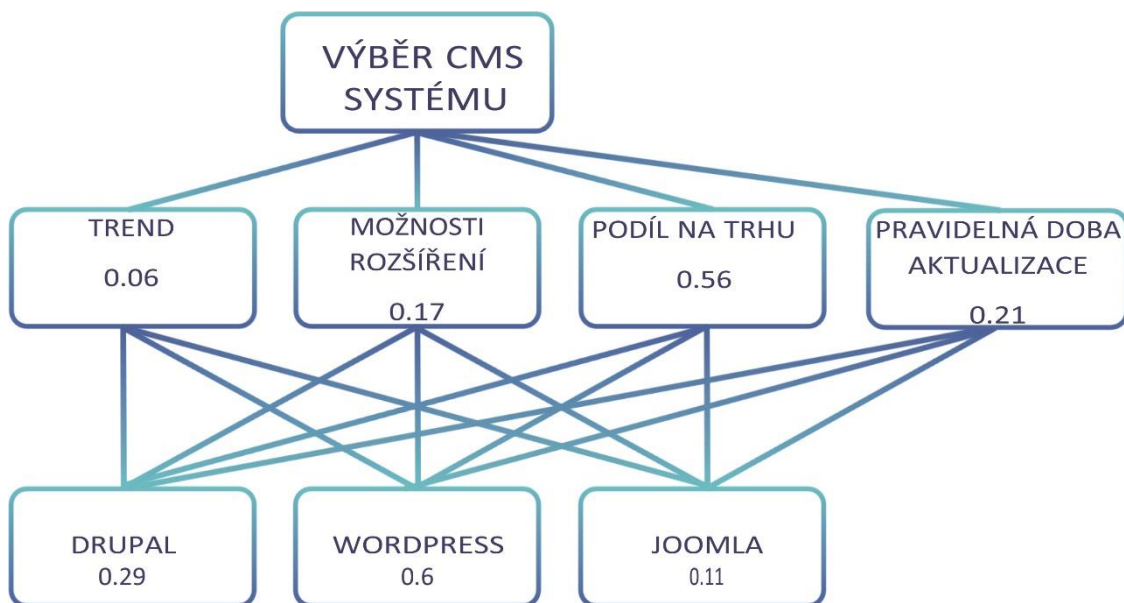
#### 4.1.4 Vyhodnocení výstupu z metody AHP

Tabulka uvedená níže zobrazuje výstup v podobě pořadí, které bylo dosaženo pomocí metody AHP. Z tabulky je možné vyčíst, že kompromisní variantou je systém *WordPress*.

	Trend (%)	Možnosti rozšíření (tis.)	Podíl na trhu (%)	Pravidelná doba aktualizace systému (jednotky)	Váhy	Pořadí
DRUPAL	0,01	0,05	0,08	0,15	0,29	2
WORDPRESS	0,04	0,11	0,44	0,01	0,60	1
JOOMLA	0,01	0,01	0,04	0,05	0,11	3

Tabulka 8 Pořadí CMS systémů s využitím metody AHP

Výsledek z tabulky se dá převést i do grafické podoby, která má hierarchii stromové struktury. Nahlédnutím na stromovou strukturu je též možné zjistit kompromisní variantu.



Obrázek 15 Stromová struktura výběru CMS systému pomocí metody AHP

Metoda AHP vyhodnotila CMS systémy v tomto pořadí – již bylo výše uvedeno, že za kompromisní variantu se dá považovat systém *WordPress* s vahou po zaokrouhlení na dvě desetinná místa **0,60**, druhé místo s hodnotou **0,29** obsadil systém *Drupal* a za nejméně vhodnou variantu se považuje systém *Joomla* s vahou **0,11**. Další praktickou částí bakalářské práce je popsat postup, jímž se vytvoří internetová filmová databáze.

## 4.2 Vytvoření šablony

Jak již bylo uvedeno v kapitole Cíle práce nebo v kapitole Vyhodnocení výstupu z metody AHP, bude v této kapitole rozebrána druhá část praktické práce. V kapitole Výběr CMS systému za pomoci metody AHP byl jako kompromisní varianta vybrán WP. V systému WP bude od úplného počátku vytvořena internetová filmová databáze, jejíž funkcionalitou bude přidávat, upravovat, mazat a hodnotit filmy, herce a režiséry.

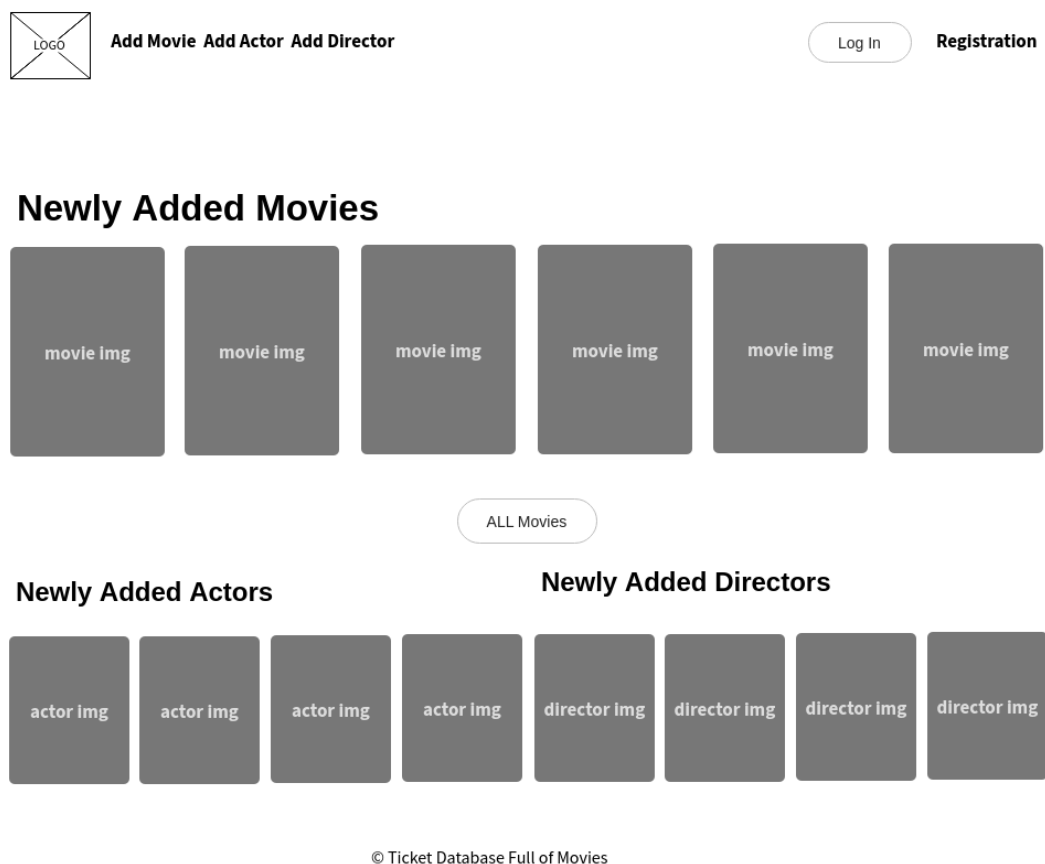
Drátový model vznikl na základě identifikace potřeb komunity uživatelů internetu, která využívá nejznámější filmové databáze. Nejznámější filmové databáze na Slovensku a v Čechách jsou csfd.cz a imdb.com. Z údajů o vyhledávání v těchto dvou databázích vyplývá, že na stránkách primárně hledají filmy, herce a režiséry. Právě v těchto databázích se nachází příliš mnoho informací, které mohou uživatele plést.

Z toho důvodu bude tato internetová filmová databáze na úvodní stránce v detailu filmu, herce a režiséra obsahovat pouze objekty, které jsou na konkurenčních stránkách nejčastěji hledané.

### 4.2.1 Vytvoření drátového modelu

Pro vytvoření drátových modelů byl použit online nástroj, v němž byla navržena rozložení webových elementů a zachycené stránky, jimiž jsou web stránky úvodní, detail filmu, herce a režiséra ve dvou případech – a to jako přihlášený a nepřihlášený uživatel.

#### 4.2.1.1 Drátový model úvodní stránka nepřihlášený uživatel



Obrázek 16 Drátový model případ nepřihlášeného uživatele (vlastní tvorba)

Výše uvedený drátový model ukazuje úvodní stránku ve stavu, kdy není uživatel přihlášen ve filmové databázi, která se skládá ze tří sekcí, jimiž jsou záhlaví, hlavní část a zápatí.

#### Záhlaví v drátovém modelu úvodní stránky

První částí drátového modelu je hlavička, která se skládá ze čtyř webových prvků, jimž jsou logo stránky, navigační menu, které obsahuje tři linky na vytvoření filmu, herce nebo režiséra a dalšími prvky jsou tlačítka, jejichž funkcí má být přihlášení, nebo registrace do filmové databáze.

#### Hlavní sekce v drátovém modelu úvodní stránky

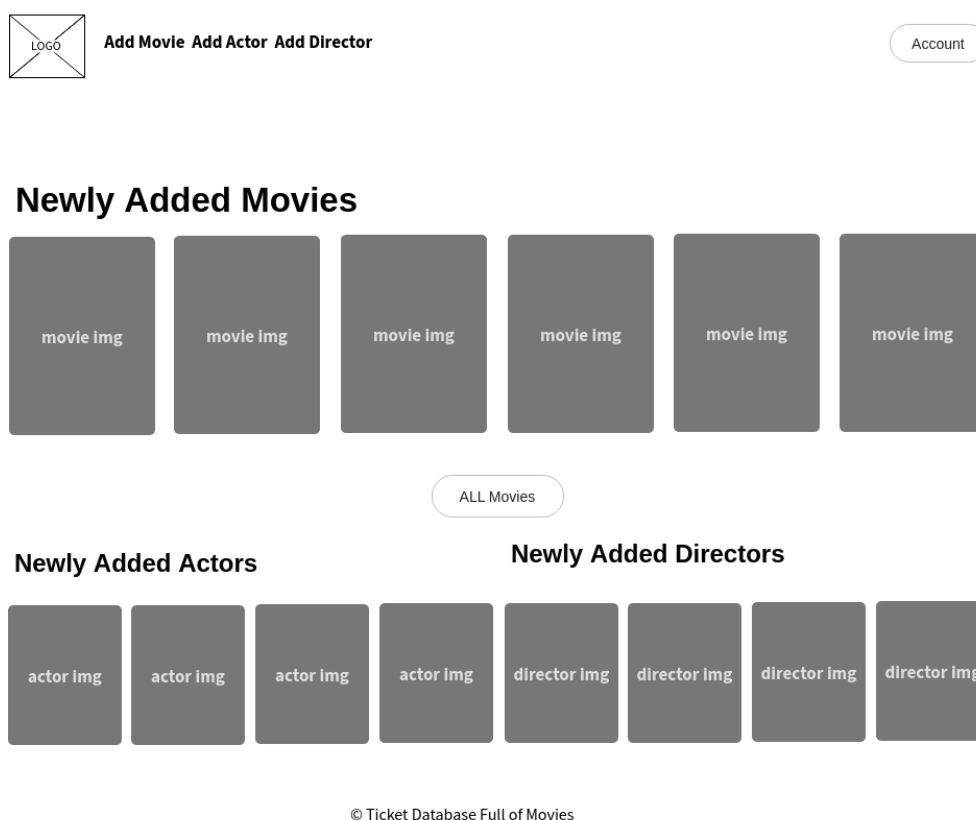
Hlavní sekce úvodní stránky se skládá ze tří webových prvků, jimiž jsou obrázky, titulky a tlačítka. Pod prvním titulkem, který bude obalen v html elementu `<h1>`, se nachází šest plakátů nejaktuálněji přidávaných filmů. Pod objektem seskupujícím plakáty

filmů je tlačítko, které přesměruje uživatele na seznam všech filmů, které se nachází v databázi.

### Zápatí v drátovém modelu úvodní stránky

V zápatí úvodní stránky se bude zobrazovat textový prvek, jenž ponese informaci o názvu internetové filmové databáze a jeho copyright.

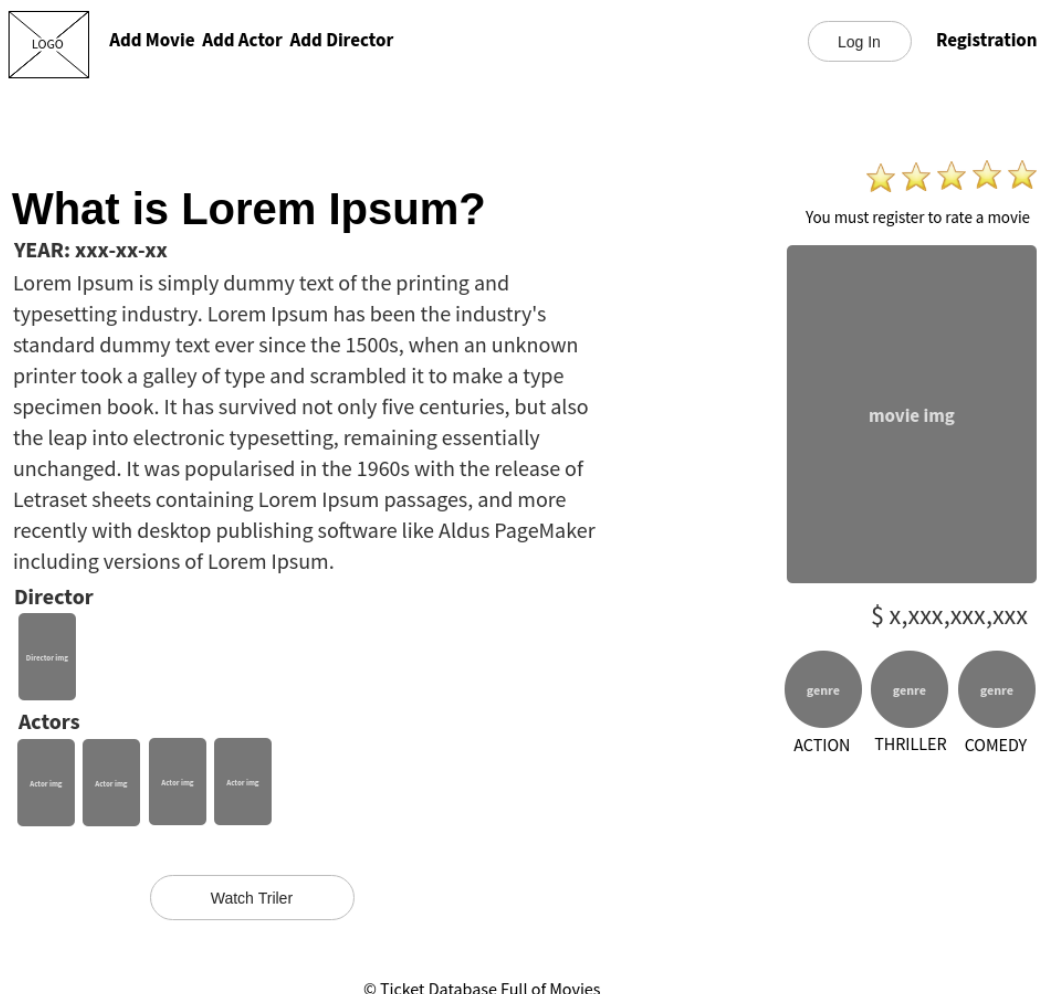
### Drátový model úvodní stránky přihlášeného uživatele



Obrázek 17 Drátový model případ přihlášeného uživatele (vlastní tvorba)

Výše uvedený drátový model ukazuje rozložení prvků v případě, že je uživatel přihlášen. Rozdíl mezi drátovými modely je v hlavičce, kde má přihlášený uživatel odkaz na svůj profil v administraci.

#### 4.2.1.2 Drátový model detail filmu



Obrázek 18 Drátový model detail filmu, případ nepřihlášeného uživatele (vlastní tvorba)

Výše uvedený drátový model ukazuje rozložení webových prvků v detailu filmu. Podobně jako hlavní stránka, i drátový model se skládá ze tří sekcí, z nichž záhlaví a zápatí obsahují totožné webové prvky, které se uvádí v kapitole Drátový model úvodní stránka. Stejně jako detail režisérů a herců, i detail filmů je rozdílný pouze v hlavní sekci.

#### Drátový model hlavní sekce, případ nepřihlášeného uživatele

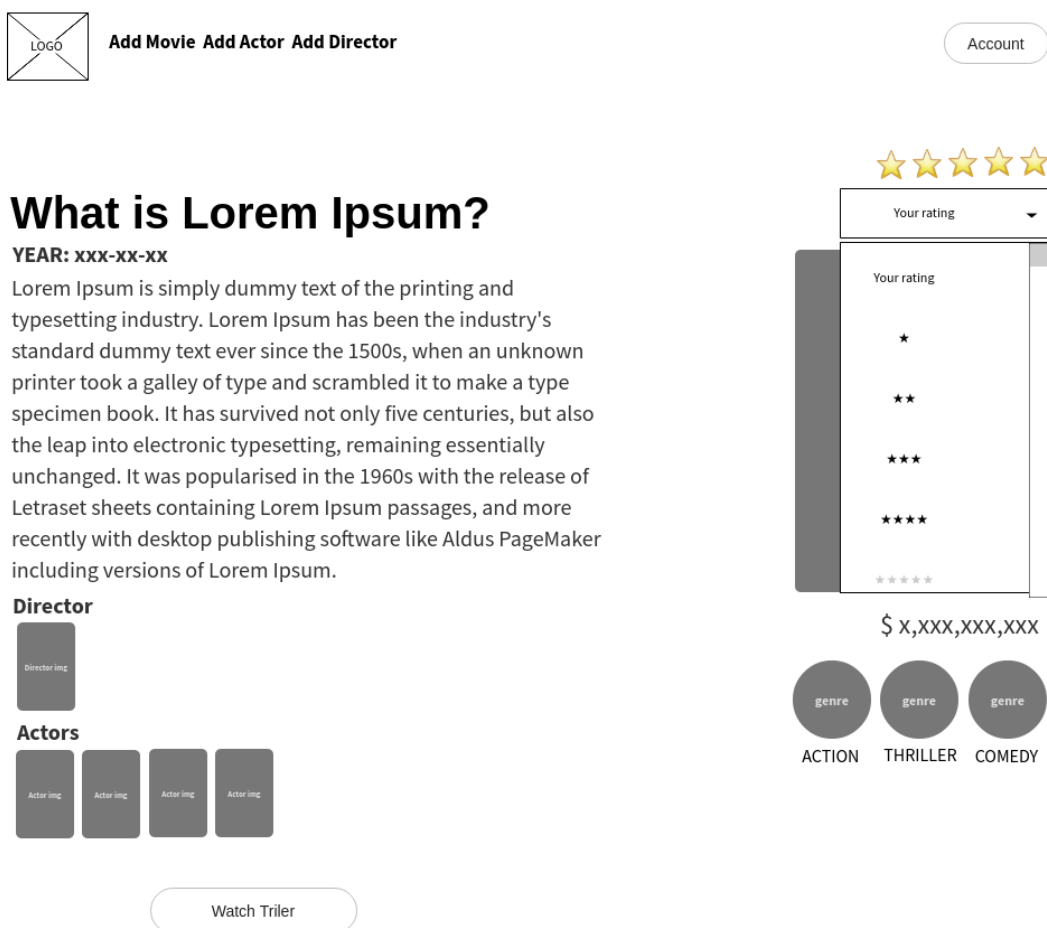
Drátový model hlavní sekce v detailu filmu se skládá ze dvou částí – levé a pravé.

Levá část obsahuje pět webových prvků, jimiž jsou titulky/název filmu, který je obalen v html tagu `<h1>`, datum vydání filmu, které je obaleno v `<time>` html tagu, stručný obsah děje filmu – text bude vložen mezi html tag `<p>`. Čtvrtým webovým prvkem jsou obrázky, které jsou rozloženy do dvou řádků – v prvním řádku se nachází obrázky režisérů, kteří se podíleli na filmu, ve druhém řádku je seznam herců, kteří ve filmu

účinkovali. Dané obrázky mají sloužit jako odkazy na detaily herců nebo režisérů. Posledním webovým prvkem je tlačítko, pomocí něhož si uživatel může prohlédnout upoutávku na film.

Pravá část – stejně jako levá – obsahuje pět webových prvků, prvním, z nichž jsou hvězdičky, které slouží k hodnocení filmu ze strany uživatelů. Druhým prvkem je odkaz pod hvězdičkami, který vyzývá uživatele, aby se zaregistroval, pokud chce uvedený film hodnotit. Třetím prvkem je obrázek, který slouží jako plakát. Čtvrtý a předposlední webový objekt je suma, kterou film vydělal. Posledním webovým prvkem jsou žánry, které jsou obsahem uvedeného filmu.

### Drátový model hlavní sekce, případ přihlášeného uživatele



© Ticket Database Full of Movies

Obrázek 19 Drátový model detail filmu, případ přihlášeného uživatele (vlastní tvorba)

Drátový model nad textem zachycuje detail filmu v případě zaregistrovaného uživatele. Rozdíl mezi případem nepřihlášeného a přihlášeného uživatele je v pravé části. Přihlášený uživatel má místo odkazu na registraci zobrazen objekt v podobě html konstruktů, jímž je *selectbox*, díky čemuž je možné film ohodnotit.

#### 4.2.2 Nakódování šablony pomocí drátového modelu

Po úspěšném návrhu drátových modelů je nutné vytvořit šablony pomocí technologie značkovacího jazyka HTML a za pomoci kaskádových stylů CSS. Vytvořené šablony se později implementují do systému WP. Z důvodu návrhu drátových modelů pouze pro stolní počítače bylo rozhodnuto využít framework bootstrap ve verzi 4.3.1, který v sobě má implementován grid systém, díky němuž je možné stavět šablony do kachliček za pomoci řádků a sloupců. Takto je zabezpečeno správné vykreslení a rozložení prvků na všech typech zařízení.

Níže uvedený obrázek popisuje hierarchii html tagů, jak je použita na všech šablonách, které jsou nakódované pro webovou aplikaci internetové filmové databáze.

```
<!doctype html>
<html lang="en-US" data-livestyle-extension="available">
  <head>...</head>
  <body class="home blog logged-in admin-bar wp-custom-logo customize-support" data-gr-c-s-loaded="true" cz-shortcut-listen="true" == $0
    <header class="container">
      <nav>...</nav>
    </header>
    <div class="clear-fix"></div>
    <main class="container series">...</main>
    <footer>...</footer>
    <script src="http://ticket-movies.6f.sk/wp-includes/js/admin-bar.min.js?ver=5.2.5"></script>
    <script src="https://cdn.3up.dk/jquery@3.3.1/dist/jquery.min.js?ver=5.2.5"></script>
    <!--[if lte IE 8]>
      <script type="text/javascript">
        document.body.className = document.body.className.replace( /(?:\s)(no-)?customize-support(?:\s|$)/, '' ) + ' no-customize-support';
      </script>
    <![endif]-->
    <!--[if gte IE 9]><!-->
    <script type="text/javascript">...</script>
    <!--<![endif]-->
    <div id="wpadminbar" class=">...</div>
    <!-- Begin Query Monitor output -->
    <script type="text/javascript">...</script>
    <div id="query-monitor-main" class="qm-theme-ticket-movies qm-theme-ticket-movies qm-js" dir="ltr" style="height: 119px;">...</div>
    <script type="text/javascript">...</script>
    <!-- End Query Monitor output -->
  </body>
</html>
```

Obrázek 20 Hierarchie html tagů v šablonách (vlastní tvorba)



#### 4.2.2.1 Uspořádání prvků

Základní hierarchie html tagů, které jsou použité v šablonách a obalené v tagu `<body>`, jsou tři sekce a jejich podtagy: `<header class="container">`

```
<nav>
    <ul>
        <li>
            ...
        </li>
    </ul>
</nav>
</header>
<div class="clear-fix"></div>
<main class="container series">
    ...
</main>
<footer>
    ...
</footer>
```

Dvě hlavní sekce obsahují třídu *container*, jejíž primární úlohou je vycentrovat všechny v ní se nacházející objekty, jakož i nastavit pro všechny objekty konstantní šířku, v níž se mají pohybovat.

Úvodní sekci v html kostře je vnořená ve značce **header** o jednu úroveň níže značka **nav**, která slouží jako obal pro neuspořádaným seznamem a všechny prvky nacházející se uvnitř, jako navigační menu. Jeden nebo dva elementy v navigačním menu jsou vždy vytržené ze základního toku dokumentu a jsou floatované na pravou stranu. Jedná se o odkazy, například vstup do administrace, přihlášení či registrace.

Pod první sekci **header** je narušené základné chování blok level elementů z důvodu využití floatů, a proto je nutné opět nastavit předvolené chování, a to pomocí třídy *clear-fix*, která obsahuje v *sytel.css* vlastnost *clear:bouth;*

Druhou a hlavní sekci v této hierarchii je **main**, v níž je naplno využit grid systém. Hlavní část se skládá ze dvou řádků, které jsou označené třídou *.row* – v prvním řádku se

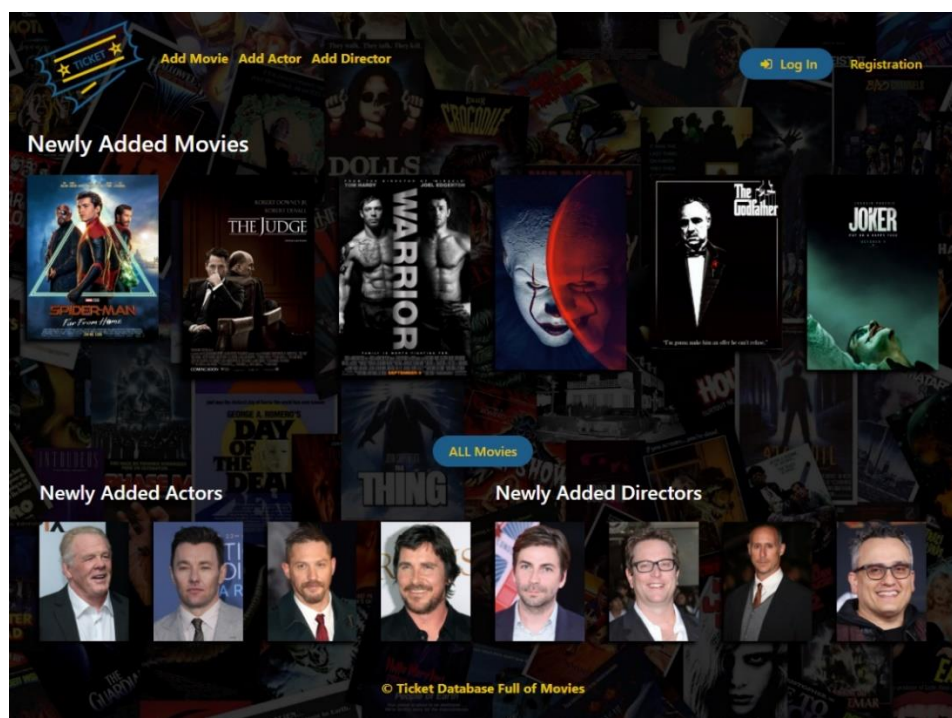
má vykreslit šest obrázků do jednoho řádku. Obrázky jsou rozdělené po párech, přičemž každý pár je obalen do třídy `.col-md-4`, která ve dvanáctisloupcovém grid systému zabere přesně 4 místa. Pod poslední třídou `.col-md-4` je uveden paragraf, který má třídu `.mx-auto`, jejíž vlastností je vycentrovat blok level elementy na střed. Druhý řádek `.row` též vykreslí obrázky v počtu osmi. Obrázky jsou však oproti předcházejícím v menším formátu a v jiném sémantickém významu. Ve druhém řádku bude jedna skupina obsahovat čtyři obrázky, které budou obalené do třídy `.col-md-6`. Třída zabere v grid systému přesně šest míst. Což znamená, že do řádku se vejde přesně osm obrázků.

Poslední – třetí – sekcí v uváděné hierarchii je **footer**, která sama o sobě nenes žádnou třídu, ale má, stejně jako html element, nastavené vlastnosti, jimiž jsou vykreslení jako blok element, nastavení výšky a zarovnání bloku na střed.

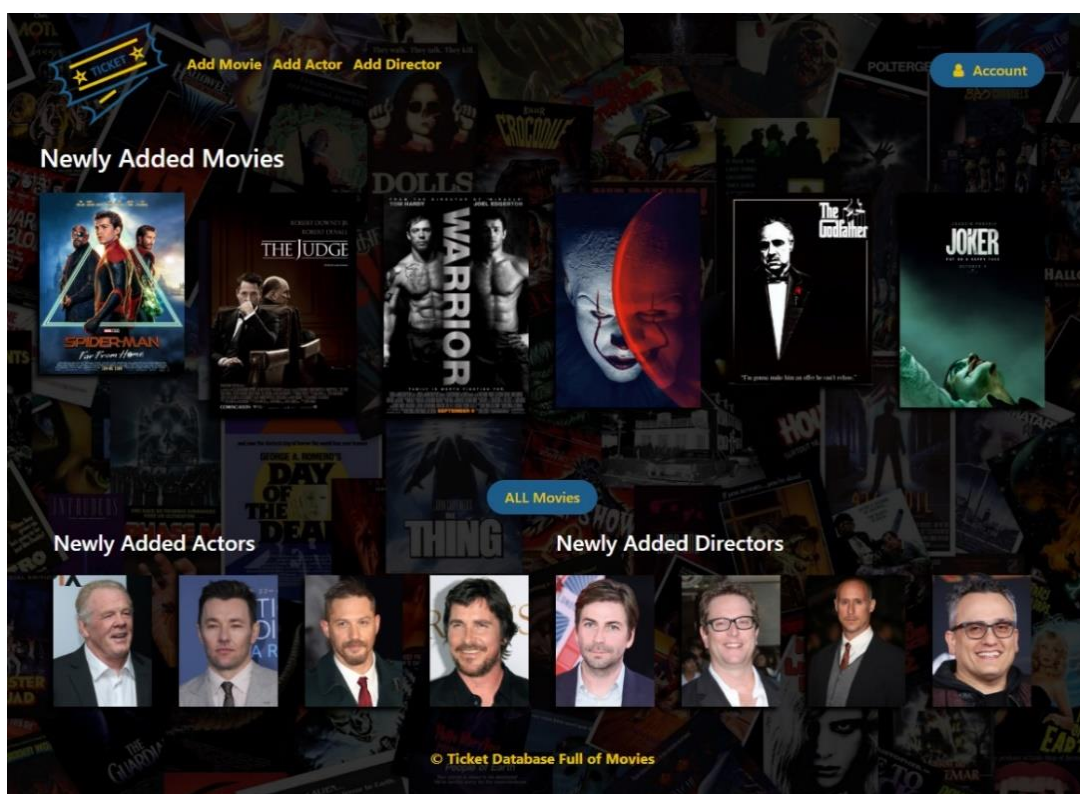
Obdobným způsobem se postupuje napříč všemi šablonami, které pro internetovou filmovou databázi vznikly.

#### 4.2.2.2 Finální vizuální zobrazení šablon

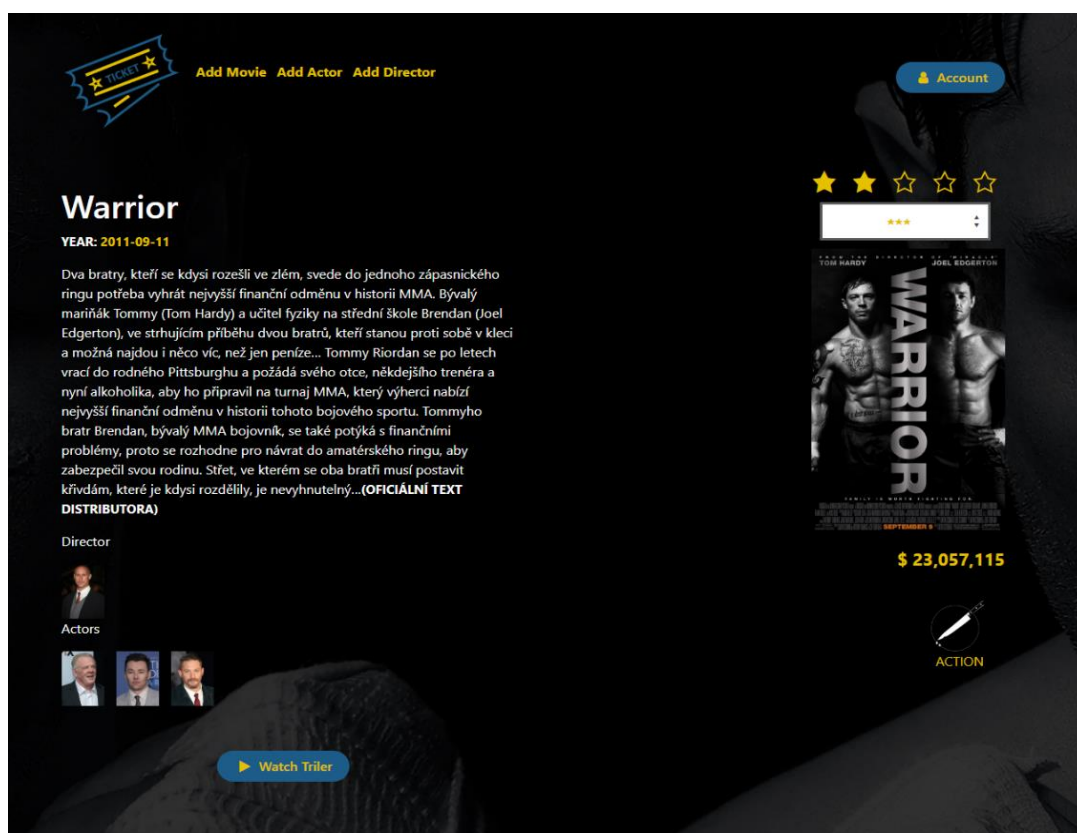
Následující kolekce obrázků zobrazuje soubory, které jsou zakončené koncovkou `.html`. Jedná se o šablony úvodní stránky přihlášeného i nepřihlášeného uživatele a taktéž o detailní náhled na film, herce nebo režiséra ve stavu, kdy je uživatel přihlášen nebo nepřihlášen.



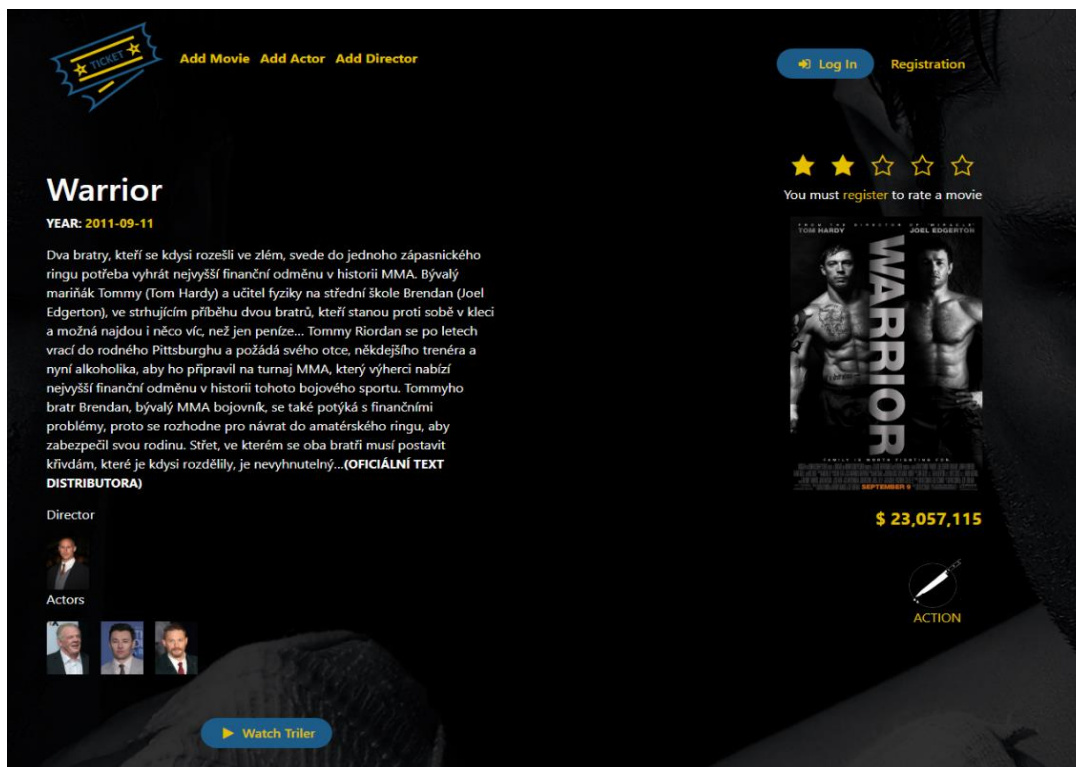
Obrázek 21 Šablona úvodní stránky filmové databáze, nepřihlášený uživatel (vlastní tvorba)



Obrázek 22 Šablona úvodní stránky filmové databáze, přihlášený uživatel (vlastní tvorba)



Obrázek 23 Šablona detailu filmu na stránce filmové databáze, nepřihlášený uživatel (vlastní tvorba)



Obrázek 24 Šablona detailu filmu na stránce filmové databáze, přihlášený uživatel (vlastní tvorba)

### 4.3 Implementace šablon do CMS systému

Po úspěšném nakódování potřebných šablon je nutné uvedené šablony rozdělit do menších souborů končících příponou *.php*. Pro aktivaci témat v redakčním systému musí WordPress obsahovat dva základní soubory *style.css* a *index.php*. V souboru, který obsahuje všechny kaskádové styly *style.css* je nutné na začátku přidat komentář, který je zapsán v přesném formátu a nese o vytvořeném tématu základní informace. Níže uváděný obrázek ukazuje formát, který musí být zvolen pro aktivaci témat.

```

/*
Theme Name: The Ticket Database is Full of Movies
Author: Ing. Petr Benda, Ph.D. & Richard Markovič
Author URI: https://www.facebook.com/markovic.richard
Description: Internetová filmová databáza
Version: 0.1
License: GNU General Public License v2 or Later
License URI: http://www.gnu.org/licenses/gpl-2.0.html
Tags: movies, genre, orange, action, white, thriller, drama, adventure, two-columns, sci-fi,
comedy, title, rating
Text Domain: the-ticket-movies-database

This theme, like WordPress, is licensed under the GPL.
Use it to make something cool, have fun, and share what you've learned with others.
*/

```

Obrázek 25 Komentář v souboru *style.css*, nutný k aktivaci témat (vlastní tvorba)

Dalším souborem, nezbytným pro aktivaci témat, je soubor *index.php*, do něhož jsou vloženy menší *.php* soubory, jak je možné vidět na obrázku 7, přičemž soubor *sidebar.php* nebude pro internetovou filmovou databázi využit a z toho důvodu nebyl vytvořen.

Po vytvoření základních souborů pro aktivaci šablony v systému WP je nutné ještě vytvořit soubor *functions.php*, pomocí něhož je možné doprogramovat funkcionalitu, kterou chceme, aby byla v tématu obsažena.

```
<?php
define( 'THEME_DIRECTORY', get_template_directory() );
define( 'THEME_DIRECTORY_URI', get_template_directory_uri() );

/**
 * Theme Settings
 */
require_once THEME_DIRECTORY . "/inc/theme-settings.php";

/**
 * Scripts and Styles
 */
require_once THEME_DIRECTORY . "/inc/scripts-and-styles.php";

/**
 * Customize Login Form
 */
require_once THEME_DIRECTORY . "/inc/custom-login-form.php";
```

Obrázek 26 Soubor *function.php* 1/2 (vlastní tvorba)

Obrázek 26 ukazuje zdrojový kód, který se nachází v souboru *functions.php*. V prvních dvou řádcích jsou definovány konstanty. Každá z výše uvedených konstant zobrazuje cestu k souborům.

**THEME\_DIRECTORY** – konstanta zobrazuje absolutní cestu k souborům, jako příklad je možné uvést výpis z konstanty:

„/home/users/markovicrichard/ticket-movies.6f.sk/web/wp-content/themes/ticket-movies“

**THEME\_DIRECTORY\_URI** – konstanta zobrazující webovou cestu k souborům, jako příklad je možné uvést výpis z konstanty:

„http://ticket-movies.6f.sk/wp-content/themes/ticket-movies“

Pomocí funkce *require\_once*, která zjistí, zda je soubor již vložen na místě, na němž se funkce nachází. Do souboru jsou vloženy tři menší podsoubory.

- **Theme Settings** – PHP soubor, v němž jsou obsažena základní nastavení témat, příkladem může být uspořádání stránek v administraci, podpora funkcionality – například *thumbnails* – a také vizuální změna textového editoru v administraci.



```

if ( !function_exists( "is_empty" ) )
{
    function is_empty( $val )
    {
        if ( (isset( $val ) && $val != "" ) )
        {
            return $val ;
        }

        $val = "N/A";
        return $val;
    }
}
function my_number_format( $val, $number_of_locations )
{
    if ( $val != "N/A" )
    {
        return number_format_i18n( $val, $number_of_locations );
    }
    else
    {
        return $val ;
    }
}

```

Obrázek 28 Soubor function.php 2/2 (vlastní tvorba)

Výše zařazený obrázek odkazuje na dvě funkce, které se nachází na konci souboru *functions.php*. Úlohou těchto funkcí je správné naformátování číslic, které se vypíší na stránce.

PHP soubory, které byly též využity z důvodu správné implementace témata do systému.

- **taxonomy-genre.php**
- **single-ticket\_movie.php**
- **single-ticket\_directors.php**
- **single-ticket\_actors.php**
- **home.php**
- **archive-movie.php**
- **404.php**

#### 4.4 Vytvoření CP, taxonomie a vazebních tabulek

Po vytvoření témata a jeho implantace do CMS systému je zbylá funkcionálníta přesunuta na přídatný modul. Ten se nachází v adresářové složce *wp-content/mu-plugins*. Souborová složka *mu-plugins* znamená *must use* (musí se použít) – je nahraná na FTP. V případě, že se takto pojmenovaná složka nachází v souborovém systému, je automaticky zaregistrovaná jako přídatný modul a není možné ji deaktivovat v administračním rozhraní.

A také – je-li nutné pro aktivaci tématu mít v souboru *style.css* základní informace v požadovaném formátu, je nutné tak učinit i v případě přídatných modulů. Obrázek, který je uveden níže, ukazuje správný zápis komentáře.

```
<?php

/*
Plugin Name: PLUGIN FOR: The Ticket Database is FULL of Movies
Plugin URI: https://www.facebook.com/markovic.richard
Description: Shitty must use plugins.
Version:      0.01
Author:       Ing. Petr Benda, Ph.D. & Richard Markovič
Author URI:   https://www.facebook.com/markovic.richard
License:      GPL2
License URI:  https://www.gnu.org/licenses/gpl-2.0.html
*/

/*
 * edit MAIN QUERY
 */

add_action( 'pre_get_posts', 'ticket_movies_modify_main_query' );
function ticket_movies_modify_main_query ( $query )
{
    // show movies on homepage
    if ( $query->is_home() && $query->is_main_query() )
    {
        $query->set( 'post_type', 'ticket_movies' );
    }
}
}
```

Obrázek 29 Registrace přídatného modulu a úprava main query v souboru *ticket-movies-must-use.php* (vlastní tvorba)

Druhá polovina výše prezentovaného obrázku ukazuje změnu hlavního WP cyklu *post\_type* z *post* na nově vytvořený post type *ticket\_movies* v případě, že se uživatel nachází na úvodní stránce.



Zbývající funkcionalita modulu je přesunuta do menších souborů, z nichž každý soubor řeší odlišnou problematiku. Obrázek níže v textu ukazuje soubor *ticket-movies-must-use.php*.

```
/**
 * CREATE TABLES for pLugin
 */
require_once WPMU_PLUGIN_DIR ."/inc/sql/create-tables.php";

/**
 * CUSTOM POST TYPES
 */
require_once WPMU_PLUGIN_DIR ."/inc/custom-post-types.php";

/**
 * CUSTOM META BOXES
 */
require_once WPMU_PLUGIN_DIR ."/inc/custom-meta-boxes.php";

/**
 * CLEANUP & REORDER
 */
require_once WPMU_PLUGIN_DIR ."/inc/cleanup-reorder.php";

/**
 * miscellaneous FUNCTIONS
 */
require_once WPMU_PLUGIN_DIR ."/inc/misc-functions.php";
```

Obrázek 30 Přídavný modul *ticket-movies-must-use.php* (vlastní tvorba)

Pomocí funkce *require\_once*, která zjistí, zda je soubor již vložen na místě, na němž se funkce nachází. Do souboru je vloženo dalších pět menších podsouborů, přičemž každý řeší určitou problematiku:

- **CREATE TABLES** – v souboru se nachází kód, který při spuštění WP systému zkontroluje, zda jsou v databázi vytvořené tabulky *wp\_movies\_directors\_actors* a *wp\_pivot\_rate\_table*. V případě, že tabulky vytvořené nejsou, skript tabulky vytvoří.

```

function ticket_movies_pivot_table()
{
    global $wpdb;
    $table_name = $wpdb->prefix."movies_directors_actors";
    if ($wpdb->get_var("SHOW TABLES LIKE '$table_name'") != $table_name)
    {
        $sql = 'CREATE TABLE '.$table_name.'(
            id INTEGER NOT NULL AUTO_INCREMENT,
            movie_id VARCHAR(20),
            director_or_actor_id VARCHAR(20),
            id_typ VARCHAR(20),
            PRIMARY KEY (id))';
        require_once(ABSPATH.'wp-admin/includes/upgrade.php');
        dbDelta($sql);
        echo "TABLE was CREATE";
    }
    else
    {
        echo "already done";
    }
}

```

Obrázek 31 Vazební tabulka sloužící k propojení filmů, herců a režisérů (vlastní tvorba)

```

function create_pivot_rate_table()
{
    global $wpdb;
    $table_name = $wpdb->prefix."pivot_rate_table";
    if ($wpdb->get_var("SHOW TABLES LIKE '$table_name'") != $table_name)
    {
        $sql = 'CREATE TABLE '.$table_name.'(
            id_pivot_table INTEGER NOT NULL AUTO_INCREMENT,
            users_id INTEGER,
            post_id INTEGER,
            rate_values INTEGER,
            PRIMARY KEY (id_pivot_table))';
        require_once(ABSPATH.'wp-admin/includes/upgrade.php');
        dbDelta($sql);
        echo "TABLE was CREATE";
    }
    else
    {
        echo "already done";
    }
}

```

Obrázek 32 Tabulka obsahující hodnocení filmů (vlastní tvorba)

Obrázek 31 popisuje tabulku, která slouží k uchování kardinality vztahů mezi filmy, hercích a režisérech.

Obrázek 32 popisuje tabulku sloužící k uchování hodnoty, kterou přidělili filmům uživatelé.

Tabulky byly vytvořeny na základě získaných poznatků autora z literární rešerše a v rámci práce bylo konzultováno s vývojáři, kteří v systému programují rozšiřitelné moduly, zdali je řešení vyhovující.

- **CUSTOM POST TYPE** – soubor obsahuje funkce, které registrují vlastní typ příspěvku. V případě uvedené bakalářské práce existují tři typy příspěvků, které jsou aktivně používané. Detailní náhled na registraci vlastního typu příspěvku viz. obrázek níže.

```
<?php
/**
 * REGISTER CUSTOM POST TYPES actors
 */

add_action('init', 'ticket_actors_post_types');
function ticket_actors_post_types()
{
    $labels = array(
        'name' => _x('Actors', 'Post type general name', 'the-ticket-Actors-database'),
        'singular_name' => _x('Actor', 'Post type singular name', 'the-ticket-Actors-database'),
        'menu_name' => _x('Actors', 'Admin Menu text', 'the-ticket-Actors-database'),
        'name_admin_bar' => _x('Actor', 'Add New on Toolbar', 'the-ticket-Actors-database'),
        'add_new' => __('Add New', 'the-ticket-Actors-database'),
        'add_new_item' => __('Add New Actor', 'the-ticket-Actors-database'),
        'new_item' => __('New Actor', 'the-ticket-Actors-database'),
        'edit_item' => __('Edit Actor', 'the-ticket-Actors-database'),
        'view_item' => __('View Actor', 'the-ticket-Actors-database'),
        'all_items' => __('All Actors', 'the-ticket-Actors-database'),
        'search_items' => __('Search Actors', 'the-ticket-Actors-database'),
        'parent_item_colon' => __('Parent Actors:', 'the-ticket-Actors-database'),
        'not_found' => __('No Actors found.', 'the-ticket-Actors-database'),
        'not_found_in_trash' => __('No Actors found in Trash.', 'the-ticket-Actors-database'),
    );
};
```

Obrázek 33 Registrace vlastního typu příspěvku 1/2 (vlastní tvorba)

```

$args = array(
    'labels'           => $labels,
    'public'          => true,
    'publicly_queryable' => true,
    'show_ui'         => true,
    'show_in_menu'    => true,
    'query_var'       => true,
    'rewrite'         => array('slug' => 'actor'),
    'capability_type' => 'post',
    'has_archive'     => true,
    'hierarchical'   => false,
    'menu_position'   => null,
    'menu_icon'       => 'dashicons-admin-users',
    // 'taxonomies'    => array( 'post_tag' ),
    'supports'        => array('title', 'thumbnail', 'editor'),
    'register_meta_box_cb' => 'ticket_movies_movie_meta_boxes'
);

register_post_type('ticket_actors', $args);
}

```

Obrázek 34 Registrace vlastního typu příspěvku 2/2 (vlastní tvorba)

Obrázky 33 a 34 popisují registraci vlastního typu příspěvku, jímž je herec. Totožným způsobem jsou zaregistrovány i příspěvky s filmy a režiséry.

- **CUSTOM META BOX** – soubor obsahuje skripty, které jsou zodpovědné za vykreslení meta-boxů v administraci pod příspěvky filmů. V souboru se též nachází vlastní taxonomie. Níže uvedené obrázky ukazují registraci vlastní taxonomie v systému WP.

```

<?php
/**
 * Custom Taxonomy
 */
add_action('init', 'ticket_movies_taxonomies');
function ticket_movies_taxonomies()
{
    $labels = [
        'name' => _x('Genres', 'taxonomy general name', 'the-ticket-movies-database'),
        'singular_name' => _x('Genre', 'taxonomy singular name', 'the-ticket-movies-database'),
        'search_items' => __('Search Genres', 'the-ticket-movies-database'),
        'all_items' => __('All Genres', 'the-ticket-movies-database'),
        'parent_item' => __('Parent Genre', 'the-ticket-movies-database'),
        'parent_item_colon' => __('Parent Genre:', 'the-ticket-movies-database'),
        'edit_item' => __('Edit Genre', 'the-ticket-movies-database'),
        'update_item' => __('Update Genre', 'the-ticket-movies-database'),
        'add_new_item' => __('Add New Genre', 'the-ticket-movies-database'),
        'new_item_name' => __('New Genre Name', 'the-ticket-movies-database'),
        'menu_name' => __('Genre', 'the-ticket-movies-database'),
        'separate_items_with_commas' => __('Separate genres with commas'),
        'add_or_remove_items' => __('Add or remove genres'),
        'choose_from_most_used' => __('Choose from most used genres'),
    ];
}

```

Obrázek 35 Registrace vlastní taxonomie 1/2 (vlastní tvorba)

```

$args = [
    'hierarchical' => false, // make it hierarchical (like categories)
    'labels' => $labels,
    'show_ui' => true,
    'show_admin_column' => true,
    'query_var' => true,
    'rewrite' => ['slug' => 'genre'],
];
register_taxonomy('genre', ['ticket_movies'], $args);
}

```

Obrázek 36 Registrace vlastní taxonomie 2/2 (vlastní tvorba)

- **CLEANUP REORDER** – soubor, který obsahuje všechny funkce odstraňující nadbytečný kód generovaný systémem, příkladem může být generování rss textu nebo emoji smajlíků.
- **MISC FUNCTIONS** – zbývající funkce, které nebyly zařazeny do předchozích souborů.

## 5 Závěr

Účelem této bakalářské práce je porovnat nejznámější CMS systémy z pohledu rychlé implementace, popularity systému a možnosti zaměnit v případě potřeby developerský tým. K výběru systému je použita metoda AHP, jejíž úkolem je identifikovat kompromisní variantu. Po nalezení kompromisní varianty bude na systém implementována webová aplikace, která má sloužit jako internetová filmová databáze.

První část bakalářské práce popisuje a analyzuje pojem CMS systém. Představuje aktuální trendy nejznámějších CMS systémů, jimiž jsou Drupal, Joomla a WordPress a popisuje jejich konkrétní výhody a nevýhody.

Druhá část bakalářské práce – a zároveň poslední část teorie – blíže specifikuje architekturu CMS systému, vysvětluje třívrstvou architekturu WP systému, objasňuje hierarchii šablonového systému, ukazuje možnosti výběru příspěvků z databáze a také ukazuje možnosti rozšíření.

Praktická část bakalářské práce identifikuje pomocí metody AHP kompromisní variantu na základě čtyř zvolených kritérií. Kritérii pro hodnocení kompromisní varianty jsou trend nejpoužívanějších systémů ve vyhledávání, velikost rozšiřitelných modulů, podíl na trhu, který mají jednotlivé systémy a pravidelná doba aktualizace. Jako nejvýhodnější systém byl identifikován WordPress.

Další část bakalářské práce se věnuje vytvoření webové aplikace, jíž je internetová filmová databáze na systému WordPress. Tato část popisuje postup, pomocí něhož je aplikace vytvořena. Jako první je vytvořen drátový model, který určuje rozložení prvků. V další části jsou nakódovány všechny potřebné stránky pomocí jazyka HTML a technologie CSS. Následně tato část práce ukazuje, jak se implementuje šablona do systému WordPress. Poslední část bakalářské práce řeší využitelnost složky *mu-plugins*, do níž jsou naprogramovány funkce, starající se o taxonomii, vazební tabulky a též vlastní typy stránek.

V bakalářské práci jsou uvedeny postupy výpočtu kompromisní varianty pomocí metody AHP a také je ukázáno, jak je možné na vybraném CMS systému vytvořit internetovou filmovou databázi a doprogramovat do ní přídavné moduly. Díky tomu si může uživatel se základní znalostí programování a kódování vytvořit vlastní dynamickou stránku nebo přídavný modul zcela sám.

## 6 Seznam použitých zdrojů

*Architektúra klient-server (Client–server model)*. (2016). Získáno 2020-02-25, z <https://managementmania.com/>: <https://managementmania.com/sk/architektura-klient-server>

Basl, J., & Blažíček, R. (2012). *Podnikové informační systémy: podnik v informační společnosti* (3., aktualiz. a dopl. vyd. vyd.). Praha: Grada.

Beighley, L. (2010). *Drupal for dummies*. Hoboken, NJ: Wiley Pub.  
*Best Content Management System – WordPress Vs Joomla Vs Drupal*. (nedatováno). Získáno 2020-02-25, z How to Make a Website in 2020 - Free Easy Guide to Building a Website: <https://makeawebsitehub.com/compare/content-management-system-cms-comparison/>

Burge, S. (2015). *Joomla! 3 explained: your step-by-step guide* (Second edition. vyd.). Upper Saddle River, NJ: Addison Wesley.

Butcher, M. (2010). *Drupal 7 module development: create your own Drupal 7 modules from scratch* (1st ed. vyd.). Birmingham: Packt Publishing.

*Codex: Database Description*. (nedatováno). Získáno 2020-02-25, z [codex.wordpress.org](https://codex.wordpress.org/): [https://codex.wordpress.org/Database\\_Description](https://codex.wordpress.org/Database_Description)

DevriX, T. (2019). *Top 10 Content Management Systems and Which One We Think is the Best - DevriX*. Získáno 2020-02-25, z WordPress Development Agency - Custom Web Development by DevriX: <https://devrix.com/tutorial/top-10-content-management-systems-one-think-best/>

Fotr, J., & Švecová, L. (2010). *Manažerské rozhodování: postupy, metody a nástroje* (2., přeprac. vyd. vyd.). Praha: Ekopress.

Chapman, C. (2011). *Top 10 content management systems | Webdesigner Depot*. Získáno 2020-02-25, z Web Design Blog - Webdesigner Depot: <https://www.webdesignerdepot.com/2011/10/top-10-content-management-systems/>

*Joomla! Extensions Directory*. (2020). Získáno 2020-02-26, z <https://extensions.joomla.org/>: <https://extensions.joomla.org/>

Kloostra, S. (2015). *Joomla! 3 SEO and performance*. New York: Apress.

Kudláček, L. (2010). *WordPress: podrobný průvodce tvorbou a správou webů* (Vyd. 1. vyd.). Brno: Computer Press.

Larman, C., & Basili, V. (2003). Iterative and incremental developments. a brief history. *Computer*, vol. 36(issue 6), 47-56. doi:10.1109/MC.2003.1204375

Leary, S. (2013). *WordPress for Web Developers*. doi:10.1007/978-1-4302-5867-4\_14  
Messenlehner, B., & Coleman, J. (2014). *Building web apps with WordPress* (First edition. vyd.). Sebastopol, CA: O'Reilly Media, Inc.

*Module project | Drupal.org*. (2019). Získáno 2020-02-26, z Drupal - Open Source CMS | Drupal.org:  
[https://www.drupal.org/project/project\\_module?f%5B0%5D=&f%5B1%5D=&f%5B2%5D=&f%5B3%5D=&f%5B4%5D=sm\\_field\\_project\\_type%3A%5B\\*+TO+\\*%5D&f%5B5%5D=&f%5B6%5D=&text=&solrsort=iss\\_project\\_release\\_usage+desc&op=Search](https://www.drupal.org/project/project_module?f%5B0%5D=&f%5B1%5D=&f%5B2%5D=&f%5B3%5D=&f%5B4%5D=sm_field_project_type%3A%5B*+TO+*%5D&f%5B5%5D=&f%5B6%5D=&text=&solrsort=iss_project_release_usage+desc&op=Search)  
Rai, S. (2019). *Drupal Vs WordPress Vs Joomla: Which is the best CMS Platform in 2020?* Získáno 2020-02-25, z <https://themegrill.com/>: <https://themegrill.com/blog/drupal-vs-wordpress-vs-joomla/>

Rockley, A., Kostur, P., & Manning, S. (2003). *Managing enterprise content: a unified content strategy* (1st ed. vyd.). Indianapolis, Ind.: New Riders.

Schäferhoff, N. (2019). *How to Choose the Right CMS Platform*. Získáno 2020-02-25, z How to Create a Website: <https://websitesetup.org/cms-comparison-wordpress-vs-joomla-drupal/>

Šubrt, T. (2015). *Ekonomicko-matematické metody* (2. upravené vydání. vyd.). Plzeň: Vydavatelství a nakladatelství Aleš Čeněk, s.r.o.

*Template Hierarchy*. (2019). Získáno 2019-08-27, z <https://developer.wordpress.org/themes/basics/template-hierarchy/>

*The Loop | Theme Developer Handbook | WordPress Developer Resources*. (nedatováno). Získáno 2020-02-25, z WordPress Developer Resources | Official WordPress Developer Resources: <https://developer.wordpress.org/themes/basics/the-loop/>

*Trojvrstvá architektura - Three-tier architecture*. (2016.) Získáno 2020-02-25, z <https://managementmania.com/>: <https://managementmania.com/sk/trojvrstvova-architektura-three-tier-architecture>

*Vodopádový model - Waterfall model*. (2015). Získáno 2020-02-26, z Sociální síť pro business - ManagementMania.com: <https://managementmania.com/cs/vodopadovy-model-waterfall-model>

*W3Techs.com - Usage Statistics and Market Share of Content Management Systems, February 2020*. (2020). Získáno 2020-02-25, z Usage statistics of content management systems: [https://w3techs.com/technologies/overview/content\\_management](https://w3techs.com/technologies/overview/content_management)

*What is Copyleft? - GNU Project - Free Software Foundation*. (2018). Získáno 2020-02-25, z What is Copyleft? - GNU Project - Free Software Foundation: <https://www.gnu.org/licenses/copyleft.en.html>

*WordPress Plugins WordPress.org*. (2020). Získáno 2020-02-26, z <https://wordpress.org/>: <https://wordpress.org/plugins/>



*WordPress Plugins.* (2010). Získáno 2020-02-25, z WordPress.org:  
<https://wordpress.org/plugins/>

*WordPress Web Hosting.* (2019). Získáno 2020-02-26, z Blog Tool, Publishing Platform, and CMS &mdash; WordPress.org: <https://wordpress.org/hosting/>

*WordPress, Drupal, Joomla – Trendy Google.* (2020). Získáno 2020-02-25, z WordPress, Drupal, Joomla – Trendy Google:  
<https://trends.google.com/trends/explore?date=all&geo=SK&q=%2Fm%2F02vtpl,%2Fm%2F01641s,%2Fm%2F07qb81>

*WP\_Query | Class | WordPress Developer Resources.* (2017). Získáno 2020-02-26, z WordPress Developer Resources | Official WordPress Developer Resources:  
[https://developer.wordpress.org/reference/classes/wp\\_query/](https://developer.wordpress.org/reference/classes/wp_query/)

*Writing a Plugin WordPress Codex.* (nedatováno). Získáno 2020-02-26, z Main Page WordPress Codex: [https://codex.wordpress.org/Writing\\_a\\_Plugin](https://codex.wordpress.org/Writing_a_Plugin)

## **Přílohy**

### **6.1 Šablona a přídatný modul**

Zdrojové kódy vytvořené šablony i přídatného modulu jsou k dispozici na serveru Katedry informačních technologií PEF CZU v Praze. Konkrétně na adrese:  
<https://kitlab.pef.czu.cz/~bendap/BP/Markovic/Markovic.zip>