

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Bakalářská práce

Wordpress plugin pro vyhledávání předků

Místo této strany bude
zadání práce.

Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 23. dubna 2019

Jakub Šantora

Abstract

Goal of this bachelor's thesis is to develop a WordPress plugin which will assist american users of Czech-American TV's website with czech roots to find information about their ancestors. The theoretical part of this thesis describes the content management system WordPress, with increased focus on plugin development, additional technologies used for developing the plugin and problems USA citizens have to face when researching their Czech ancestry. The practical part analyzes available data that can be used for finding one's ancestors and the needs of Czech-American TV for the developed plugin. It also describes the development and testing of said plugin.

Abstrakt

Cílem této bakalářské práce je vytvoření WordPress pluginu pro Czech-American TV, který bude pomáhat americkým návštěvníkům webu Czech-American TV s českými kořeny ve vyhledávání informací o jejich předcích. Teoretická část práce popisuje redakční systém WordPress se zaměřením na tvorbu pluginů, další technologie které budou při tvorbě pluginu využity a na problémy vyhledávání obyvatel USA při vyhledávání jejich českých předků. Praktická část práce analyzuje dostupná data k vyhledávání, potřeby Czech-American TV pro vytvářený plugin, a dále popisuje implementaci a následné testování vytvářeného pluginu.

Obsah

1	Úvod	8
2	WordPress	9
2.1	Základní Informace	9
2.2	PHP	10
2.3	MySQL	10
3	Pluginy ve WordPressu	12
3.1	Plugin API	12
3.2	Uživatelské rozhraní	13
3.2.1	Shortcodes	13
3.2.2	Administrační menu	14
3.2.3	Připojení skriptů a stylů	15
3.3	AJAX	15
3.4	Databáze pluginu	16
3.4.1	Třída wpdb	16
3.4.2	Vytváření tabulek	17
3.5	Bezpečnost	17
3.5.1	Kontrola uživatelských práv	17
3.5.2	Nonce	18
3.5.3	Validace a sanitizace dat	18
4	Další použité technologie	19
4.1	Bootstrap	19
4.2	jQuery	20
4.3	Google Charts	20
4.4	Google Maps	21
5	Emigrace do USA	22
5.1	Amerikanizace českých jmen a příjmení	22
5.1.1	Křestní jména	22
5.1.2	Příjmení	22
5.1.3	Přechýlená příjmení	23
5.2	Německé názvy a písma	23

6	Analýza	25
6.1	Původní stav	25
6.2	Data pro plugin	26
6.2.1	Česká příjmení a jejich četnosti	26
6.2.2	Historické názvy měst	27
6.3	Požadavky na nový plugin	28
7	Návrh	30
7.1	Databáze	30
7.2	Uživatelské rozhraní	31
7.2.1	Překlad anglických jmen a příjmení	31
7.2.2	Práce s německými dokumenty	33
7.2.3	Mapa četnosti příjmení	34
7.2.4	Video návod k obsluze pluginu	35
7.2.5	Administrátorské rozhraní	35
8	Implementace	36
8.1	Struktura pluginu	36
8.2	Databáze pluginu	36
8.2.1	Vytvoření tabulek a import dat	37
8.2.2	Komunikace s databází	38
8.3	Rozhraní pluginu	38
8.4	Komunikace se serverem	39
8.5	Funkční požadavky	40
8.5.1	Překlad a přepis jmen	40
8.5.2	Práce s německými dokumenty	45
8.5.3	Zobrazení na mapě	47
8.5.4	Požadavky společné pro všechny funkce	51
8.5.5	Video návod	53
8.5.6	Administrace pluginu	54
8.6	Mimofunkční požadavky	54
8.6.1	Snadná instalace	55
8.6.2	Responzivní design	55
8.6.3	Rychlé provádění dotazů	55
8.6.4	Uživatelské rozhraní v anglickém jazyce	55
8.6.5	Zadávání pouze znaků anglické abecedy	55
8.6.6	Bezpečnost	56

9 Testování	57
9.1 Systémové testy	57
9.2 Podpora prohlížečů a OS	58
9.3 Testování zabezpečení	59
10 Závěr	61
Přehled Zkratk	63
Literatura	64
Přílohy	66

1 Úvod

Czech-American TV (zkráceně CATV) je americká nezisková televizní stanice věnující se České republice. Poměrně velká skupina Američanů má předky z Evropy a tedy i České republiky a Vysílání CATV jim poskytuje informace o rodné zemi jejich předků. Mnozí z nich by se také rádi podívali přímo do rodných měst svých předků a případně kontaktovali své vzdálené příbuzné. V současnosti bohužel neexistuje mnoho zdrojů, ze kterých by mohli čerpat relevantní informace. Původně Česká příjmení se navíc často z praktických důvodů transformují do anglické podoby a poměrně velká část emigrantů z České republiky má ručně psané dokumenty psané v německém jazyce, např. z dob Rakousko-Uherska. Tyto dokumenty jsou navíc často psány německými písmi, například Kurrentem nebo Švabachem.

Webové stránky CATV, založené na redakčnímu publikačnímu systému WordPress, poskytují základní informace které mohou ve vyhledávání předků jejich návštěvníkům pomoci - popisují transformování jmen do angličtiny, odkazují na článek o překládání německých názvů měst a vesnic do češtiny. Pomocí pluginu (rozšíření) pro WordPress umožňují stránky také vyhledat četnost příjmení v jednotlivých městech ČR na interaktivní mapě a tím určit jeho původ. Tyto informace však nejsou dostačující k spolehlivému vyhledání rodinné historie, navíc se s nimi obtížně pracuje uživatelům, kteří již neovládají češtinu.

Proto bude v rámci této práce vytvořen nový plugin pro WordPress, který by měl návštěvníkům webu vyhledání informací o jejich kořenech značně zjednodušit. Teoretická část práce se věnuje systému WordPress (se zaměřením na vývoj pluginů), dalším technologiím využitým k tvorbě pluginu (například frameworkům k zobrazení map) a problémům spojeným s emigrací do USA (transformací jmen a čtením německy psaných dokumentů). Praktická část práce analyzuje původní plugin používaný Czech-American TV, požadavky na nový plugin a dostupné datové zdroje pro vyhledávání informací. Dále pak praktická část popisuje návrh databáze a rozhraní jednotlivých částí pluginu a jejich následnou implementaci. Poslední kapitola praktické části se věnuje testování pluginu na splnění požadavků pomocí frameworku Selenium, testování kompatibility na různých zařízeních a webových prohlížečích a testování zabezpečení pluginu.

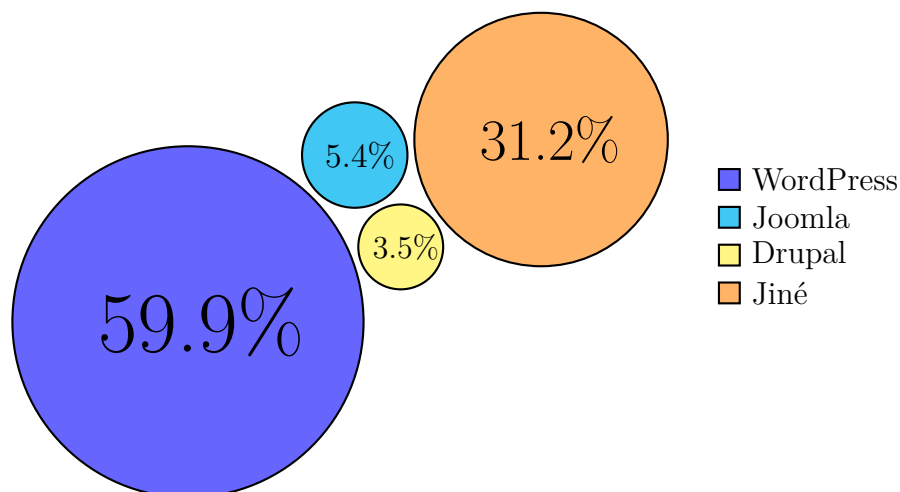
2 WordPress

Tato kapitola se věnuje redakčnímu systému pro správu obsahu (CMS, z anglického content management system) WordPress (WP), který je zaveden na webu CATV a pro který je určeno vytvářené rozšíření (plugin) - předmět této práce. Kapitola popisuje základní informace o systému WordPress a jeho jazyky a technologie.

2.1 Základní Informace

WordPress je populární open source CMS napsaný v PHP a MySQL pod licencí GNU GPL. Mezi jeho funkce patří například podpora pluginů, vzhledových témat, funkčních bloků (widgetů), zabudovaná obrázková galerie, hierarchické uskupení stránek, pokročilé formátování textu, podpora uživatelských účtů s různými úrovněmi přístupu, kategorizace a tagování příspěvků a mnoho dalších.

WordPress je používán na téměř 60% webových stránek s CMS, což činí 32.7% ze všech veřejných webových stránek [2]. Oproti druhému systému Joomla s 5.4% procenty CMS (cca. 3% webu) je tedy WordPress jasně nejpopulárnější, viz obrázek 2.1. Využívají ho například i americký časopis The New Yorker¹ nebo oficiální webové stránky Švédska².



Obrázek 2.1: Podíl CMS frameworků

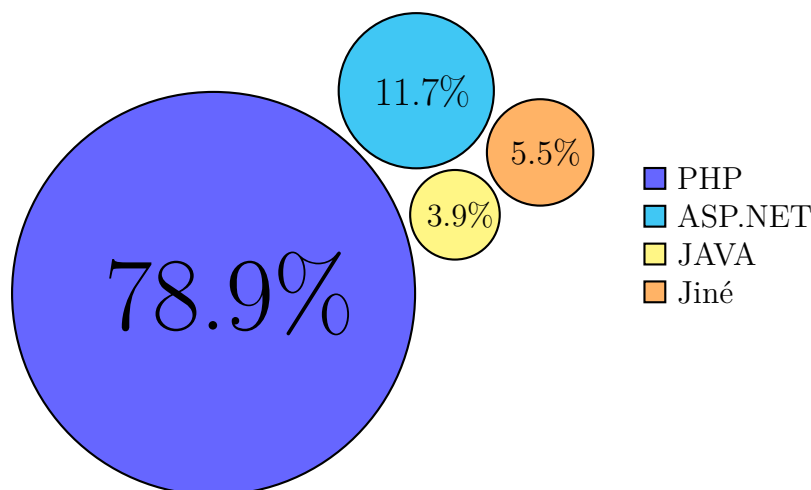
¹<https://www.newyorker.com/>

²<https://sweden.se/>

Základem WordPressu jsou příspěvky (*posts*) a stránky (*pages*), které tvoří obsah webové stránky. Statické weby budou zpravidla tvořeny převážně stránkami, zatímco dynamické weby (blogy, eshopy atd.) budou obsahovat spíše příspěvky. Při načtení webové stránky se WordPress podle URL a použitých šablon rozhodne jakou stránku má uživateli zobrazit a jaké příspěvky bude stránka obsahovat [21].

2.2 PHP

PHP, rekurzivní zkratka z PHP: Hypertext Preprocessor, původně Personal Home Page [5] je dynamicky typovaný skriptovací programovací jazyk, používaný primárně k tvorbě dynamických webových stránek. Jak již bylo zmíněno, PHP tvoří základ systému WordPress, jehož jádro je v něm napsané. Zároveň se PHP používá k rozšíření jeho funkcionality, např. pomocí pluginů. I díky tomu se PHP těší podobné popularitě - téměř 80% webů v roce 2018 používalo PHP [6], viz obrázek 2.2.



Obrázek 2.2: Využití serverových programovacích jazyků

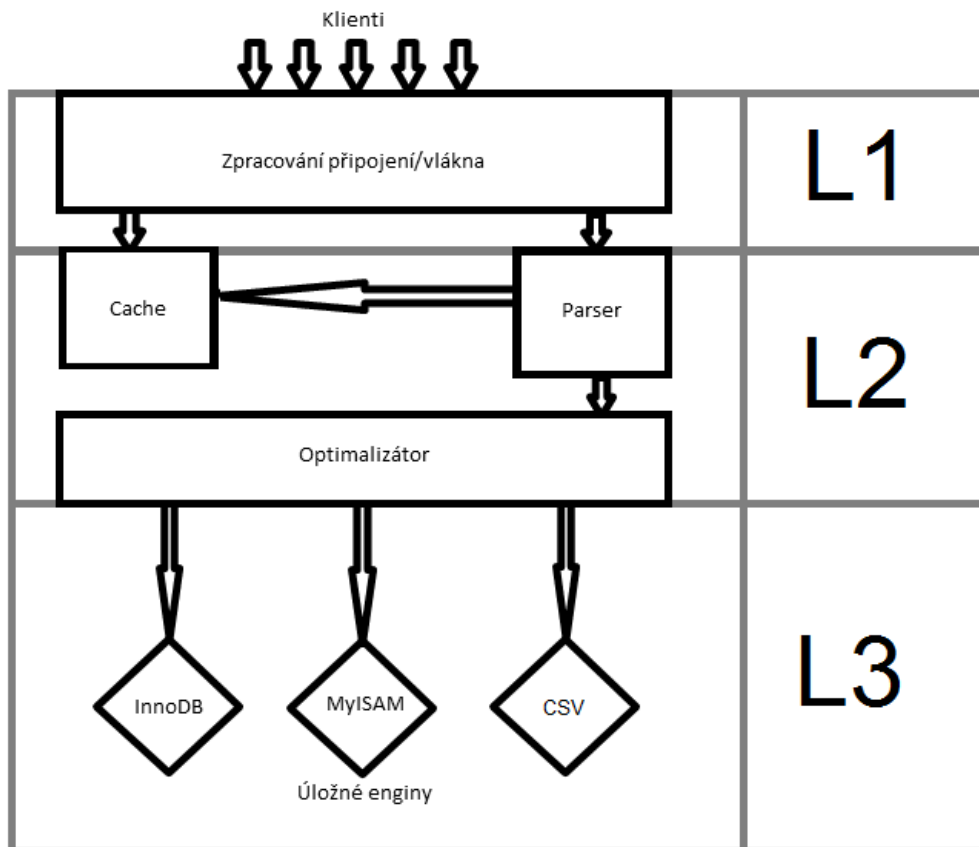
Webové stránky se v PHP vytváří na serveru - interpret ze skriptu vytvoří klasický HTML kód, který poté zašle uživateli. Dynamičnosti stránek je zajištěno úpravou výstupu, například pomocí parametrů z GET / POST požadavku nebo cookies.

2.3 MySQL

Druhou částí WordPressu je systém řízení báze dat MySQL. MySQL se stará o samotná data webu spravovaná WordPressem - například uživatele, pří-

spěvky, média, ale i nastavení samotného WordPressu a nainstalovaných rozšíření. Systém MySQL byl také od začátku vyvíjen se zaměřením na rychlost (cachování a optimalizace dotazů) a je tedy pro správu databáze WordPressu velmi vhodný.

MySQL servery používají třívrstvou architekturu [24]. První vrstva slouží k propojení serveru s klienty (autentikace, správa spojení, zabezpečení atd.). Druhá vrstva slouží k samotné funkci systému MySQL - na této vrstvě se parsují, optimalizují a cachují dotazy a provádí další interní funkce (například matematické výpočty nebo šifrování). Třetí vrstva obsahuje úložné enginy, například InnoDB, MyISAM nebo obyčejné CSV soubory (Comma Separated Values - hodnoty oddělené čárkami). Rozdělení vrstev architektury je k vidění na obrázku 2.3.



Obrázek 2.3: Architektura systému MySQL

3 Pluginy ve WordPressu

Tato kapitola popisuje vývoj pluginů pro WordPress. Pluginy (česky rozšíření) umožňují rozšířit nebo upravit základní funkcionalitu WordPressu pomocí funkcí v PHP [10], bez nutnosti přepisovat jádro systému WordPress. Kapitola popisuje základy Plugin API (Application Programming Interface): vytvoření uživatelského rozhraní pluginu, ukládání dat a zabezpečení pluginu před různými útoky.

3.1 Plugin API

Integrace do webu probíhá pomocí Plugin API [8]. Plugin API je rozhraní pro připojení pluginu k jádru WordPressu. Toho dosahuje použitím *Hooks*, které, umožňují spuštění jednotlivých funkcí pluginu ve správný čas. Hooks dělíme na *Akce* a *Filtry*.

Akce jsou spuštěny událostí na stránce, například načtením konkrétního příspěvku, publikováním nového příspěvku, nebo třeba změnou vzhledu. V reakci na tuto událost se spustí funkce k ní navázaná pomocí WP funkce `add_action()`.

Filtry jsou oproti tomu spouštěny procházejícími daty mezi databází a WordPressem, která upraví a pošlou dále k zobrazení. Tím umožňují například udržovat v databázi kompletní data pro příspěvek, ale nepřihlášenému uživateli jich zobrazit pouze část. Podobně jako akce i filtry se na WordPress připojí pomocí registrační funkce `add_filter()`.

Funkce `add_action()` a `add_filter()` přijímají téměř shodné parametry, kterými jsou:

- Jméno hooku - určení na co má akce reagovat, například hook `publish_post` spustí akci při publikování příspěvku, a hook `the_content` spustí filtr při zobrazování obsahu příspěvku uživateli.
- Jméno funkce - jméno funkce pro obsluhu akce nebo filtru, může to být vestavěná funkce PHP, funkce WordPressu, nebo nejčastěji uživatelská funkce definovaná přímo v pluginu.
- Priorita - nepovinná celočíselná hodnota, určující prioritu spuštění akce / filtru

- Parametry - nepovinná celočíselná hodnota, určující kolik parametrů obslužná funkce akceptuje, jelikož některé hooky vyžadují více než výchozí jeden parametr.

Při vytváření akcí a filtrů je důležité zohlednit pojmenování obslužných funkcí - toto jméno musí být unikátní napříč všemi aktivními pluginy i jádru WordPressu. Nejjednodušším způsobem jak zajistit unikátnost je prefixovat všechny názvy funkcí pomocí unikátního prefixu specifického pro vytvářený plugin. Alternativou je zapsání funkcí do tříd, čímž přesuneme nutnost unikátního jména pouze na třídu, ale funkce v ní už mohou mít jméno libovolné.

3.2 Uživatelské rozhraní

Tato kapitola se věnuje vytvoření grafického rozhraní pluginu na uživatelské straně pomocí shortcodes a na administrační zařazením do menu WordPressu. Dále také popisuje možnosti využití CSS skriptů a JS skriptů v uživatelském rozhraní.

3.2.1 Shortcodes

Často je v pluginu potřeba zobrazení nějakého výstupu na volitelném místě, čehož by se pouze pomocí filtrů a akcí dosahovalo obtížně. V takovém případě lze použít takzvané Shortcodes [14] - krátké kódy, které zobrazí výstup funkce připojené pomocí `add_shortcode()`. Tato funkce vyžaduje dva parametry - unikátní jméno pro shortcode a jméno obslužné funkce. V případě pluginů musí být tato funkce volána až po inicializaci jádra WordPressu, a vkládá se proto do uživatelské funkce pluginu, která se naváže na hook `init` pomocí `add_action('init', 'shortcodes_init')`, kde `shortcodes_init` je jméno funkce, jejíž výstup se zobrazí místo daného shortcode.

Vytvořený Shortcode poté může být použit v libovolném příspěvku nebo stránce, kde se koncovému uživateli webu zobrazí výstup jako kdyby se na místo použitého shortcode vložila obslužná funkce. Shortcode dělíme na dva druhy - jednoduché:

```
[shortcode]
```

nebo uzavírající obsah (*enclosing*):

```
[shortcode] --obsah-- [/shortcode]
```

V druhém případě má funkce přístup k obsahu uvnitř a může ho libovolně editovat a znovu zobrazit. Oba druhy shortcode také akceptují atributy zapsané následovně:

```
[shortcode attr="value"]
```

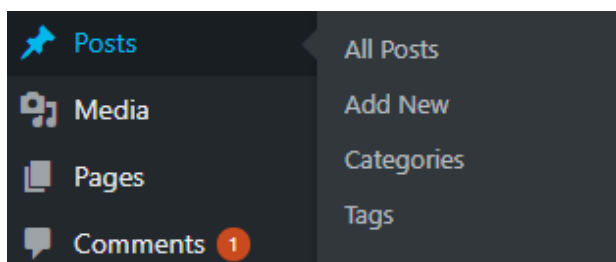
Obsluhující funkce má potom k dispozici pole všech atributů a může podle nich upravit svoje chování.

3.2.2 Administrační menu

Kromě grafického výstupu pro návštěvníky stránek je často u pluginů potřeba menu pro různá nastavení. Pro takové administrační rozhraní pluginu se místo Shortcode API používá funkce `add_menu_page()`, která vytvoří menu přímo v rozhraní WordPressu (Top Level Menu), případně funkce `add_submenu_page()`, která přidá menu pod již existující položku v menu, viz obrázek 3.1. Funkce akceptují následující parametry:

- Název stránky použitý jako titulek při zobrazení položky menu
- Název stránky zobrazený v menu / submenu
- Uživatelská práva potřebná k zobrazení položky menu
- Unikátní ID pro položku menu
- Funkce s výpisem obsahu položky - podobně jako u shortcodes

Položky v hlavním menu mohou mít navíc nastavenou vlastní ikonu ve formátu SVG, nebo jednu ze zabudovaných ikon WordPressu a položky v submenu musí mít navíc jako první parametr ID menu rodičovského.



Obrázek 3.1: Příklad Top Level Menu a jeho položek (Sub Menu)

3.2.3 Připojení skriptů a stylů

Důležitou částí uživatelského rozhraní jsou skripty pro interaktivitu a styly pro vzhled stránky. Pokud nechceme CSS styly a JS skripty zapisovat přímo do obslužných funkcí v PHP, poskytuje WordPress možnost použít oddělené soubory, ať již na stejném serveru nebo externí. Připojení souborů není vhodné psát přímo do PHP funkcí pomocí tagů `<script>` a `<link>`, místo toho existují funkce `wp_register_script()` a `wp_register_style()` [10].

Tyto funkce vyžadují jako první parametr unikátní ID skriptu / stylu, pomocí kterého se na něj bude později odkazovat a jako druhý cestu k jeho souboru. Třetím volitelným parametrem je pole jiných skriptů / stylů, které se musí načíst před spuštěním. Po registrování můžeme skripty a styly načítat selektivně pomocí `wp_enqueue_script` a `wp_enqueue_style` pouze tam, kde je jejich potřeba.

3.3 AJAX

Po interpretování skriptu na serveru a odeslání HTML kódu se musí pro další komunikaci (například přidání nových dat do databáze pomocí administračního menu) odeslat nový dotaz na server a znovu načíst stránka. Elegantnějším řešením je využití AJAX (Asynchronous JavaScript And XML), díky kterému komunikace se serverem může probíhat na pozadí bez nutnosti znovu načítat celou stránku.

AJAX request (vytvořený například pomocí frameworku JQuery, který už je zabudovaný přímo ve WordPressu, viz kapitola 4.2) poté na specifikovanou adresu odešle požadavek s libovolnými parametry a asynchronně čeká na odpověď. AJAX se běžně používá například na validaci formulářů na serveru a zvýraznění polí s chybnými údaji bez nutnosti zadávat všechny údaje znovu.

Při komunikaci s WordPressem se veškeré AJAX požadavky z důvodu bezpečnosti posílají přes jediný soubor - `admin_ajax.php`. Veškeré požadavky je tedy nutné posílat přes tento soubor a na serveru je obsluhovat pomocí připojené akce `wp_ajax_jmenoakce` pro přihlášené uživatele a `wp_ajax_nopriv_jmenoakce` pro uživatele bez přihlášení [7].

Problémem při použití AJAX v pluginu je neznámá adresa souboru `admin_ajax.php` - na každé instalaci WordPressu je odlišná a nelze ji proto psát napevno do JavaScript kódu, který bude vykonávat požadavky. V administračním rozhraní WordPressu je adresa dostupná pod globální JavaScript proměnnou `ajaxurl`. Na uživatelské části k ní ovšem nemá JavaScript přístup a je proto nutné vyřešit přístup k proměnné použitím funkce

wp_localize_script. Tato funkce byla původně navržena k lokalizování řetězců v JavaScriptu do jiných jazyků, ale dá se použít k předání libovolných informací ze serveru do JS souboru, tedy i adresy souboru **admin_ajax.php**, kterou lze na serveru snadno zjistit pomocí funkce **admin_url('admin_ajax.php')**. Do **wp_localize_script** se poté pouze jako parametr zadá ID registrovaného JavaScript souboru vytvářející AJAX requesty a zjištěné URL souboru **admin_ajax.php**.

3.4 Databáze pluginu

Velká část pluginů ke svojí funkci potřebuje pouze editovat či rozšiřovat stávající data. Toho lze snadno docílit různými funkcemi pro přístup k například příspěvkům a jejich tagům nebo metadatům.

Některé pluginy však kromě editace standardních záznamů v databázi WordPressu (příspěvky, stránky atd.) mohou též vyžadovat práci s vlastní databází (například databáze zboží a zákazníků pro e-shop) nebo tabulkou. K ukládání vlastních dat existuje několik přístupů:

1. Využití Options API¹ - jednoduchého mechanismu pro ukládání párů Klíč:Hodnota do databáze. Options API je vhodné například pro ukládání nastavení pluginu nebo dalších dat malé velikosti.
2. Vlastní vytvoření a správa databáze - použití PHP k vytvoření vlastních tabulek a jejich správě. Tato metoda není příliš vhodná, jelikož uživatel pluginu musí nastavit připojení k databázi.
3. Využití třídy wpdb ve WordPressu, která umožní zápis do standardní databáze WordPressu.

3.4.1 Třída wpdb

Třída wpdb a její stejně pojmenovaná globální instance \$wpdb obsahuje sadu funkcí ke komunikaci s databází WordPressu [12] - umožňuje pluginu použít základní SQL operace **SELECT**, **INSERT**, **REPLACE**, **UPDATE** a **DELETE**, případně spustit libovolný další dotaz (například **CREATE** / **DROP** na vytvoření a smazání tabulek). Výhodou použití těchto funkcí oproti ručnímu psaní je kompatibilita v libovolné instalaci WP bez nutnosti zadávání přihlašovacích údajů k databázi - vytvoření spojení už je zařízeno ve třídě. Třída též poskytuje snadné ošetření parametrů v SQL dotazech pro

¹https://codex.wordpress.org/Options_API

ochranu před SQL Injection (spouštění podvrženého SQL dotazu potenciálně nebezpečného pro databázi).

3.4.2 Vytváření tabulek

Počáteční vytvoření tabulek je možné, jak již bylo zmíněno, použitím vlastního SQL dotazu pomocí třídy `wpdb`, ale podle WordPress Codexu [11] je preferované vytvářet tabulky s pomocí funkce WordPressu `dbDelta(sql)`. Tato funkce bere jako parametr stejný SQL dotaz jaký by byl použit při tvorbě pomocí `wpdb`, ale má tu výhodu, že automaticky porovná požadovanou strukturu databáze s již existující a pouze ji upraví - pokud tedy v nové verzi pluginu do tabulky přibude nový sloupec, není nutné psát celou novou funkci, ale pouze přidat sloupec do existujícího dotazu a znovu spustit `dbDelta`.

3.5 Bezpečnost

Poslední, ale jeden z nejdůležitějších kroků vývoje pluginu je jeho zabezpečení. Kromě již zmíněného SQL injection je tu nebezpečí Cross-site scripting (XSS), kdy útočník do stránky může vložit vlastní skript, nebo nedostatečně ošetřená kontrola uživatelských práv. K zabezpečení pluginu pro návštěvníky i provozovatele webu existuje několik kroků, popsanych dále.

3.5.1 Kontrola uživatelských práv

Základním mechanismem je zabezpečení přístupu k částem pluginu pouze pro uživatele, kteří k ní mají přístup - každý uživatel patří pod roli, například Administrator nebo Editor a všechny role mají vlastní způsobilosti, například editace příspěvků nebo mazání uživatelů [9].

Ke kontrole práv uživatele lze použít funkci `current_user_can`, která jako parametr vyžaduje id způsobilosti, například pro publikování příspěvků je to `publish_posts`. Role jsou hierarchicky uspořádané, takže vyšší role, jako například Administrator přebírají i práva nižších rolí, například Editor.

Vhodným prostředkem k zamezení přístupu k souboru je také použití funkce `defined('ABSPATH')`, která zkontroluje, jestli není k souboru přístupováno přímo přes jeho adresu, ale pouze načtením v rámci pluginu. Tím se zamezí nechtěnému přístupu i u neregistrovaných uživatelů.

3.5.2 Nonce

Zabezpečení práv uživatele stále obsahuje jedno bezpečnostní riziko - uživatel s patřičným oprávněním se může stát obětí útoku a využít svá práva k vykonání útočnickova záměru. Útočník mu například může poslat odkaz ke smazání příspěvku, k němuž by sám neměl práva, ale uživatel, který je trvale přihlášen tyto práva má.

Proto je důležité kromě práv kontrolovat i záměr uživatele - že uživatel provádí akci opravdu z místa, odkud se to očekává (například smazání příspěvku přímo ze stránky, na které se nachází. K tomu se ve WordPressu využívají tzv. *nonce* [13].

Nonce je unikátní číslo vytvořené pomocí funkce `wp_create_nonce()`, která zaručuje že akce byla prováděna přímo z prostředí WordPressu porovnáním vytvořené nonce. Každá nonce se dá použít pouze jednou a vytváří se při načítání stránky s odkazem na akci.

3.5.3 Validace a sanitizace dat

Dalším důležitým krokem pro zabezpečení je validace a sanitizace dat. Validaci (kontrolu správnosti dat) je vhodné provádět nejen pomocí JavaScriptu u klienta, ale i na serveru, aby se zajistilo že validace skutečně proběhla [25].

Sanitizace (ošetření dat, odstranění neplatných znaků) se provádí vždy na serveru a zamezuje útokům pomocí neplatných vstupů, například SQL Injection. Lze k ní využít například vestavěné funkce WordPressu `sanitize_*`.

Kromě vstupů je vhodné ošetřovat i výstupy, minimálně pomocí PHP funkce `htmlspecialchars`, která přepíše vybrané znaky na html entity (například znak `<` na `<`). Tím se zabezpečí proti XSS útoku, i pokud plugin používá data z jiného zdroje (například příspěvky z databáze WordPressu).

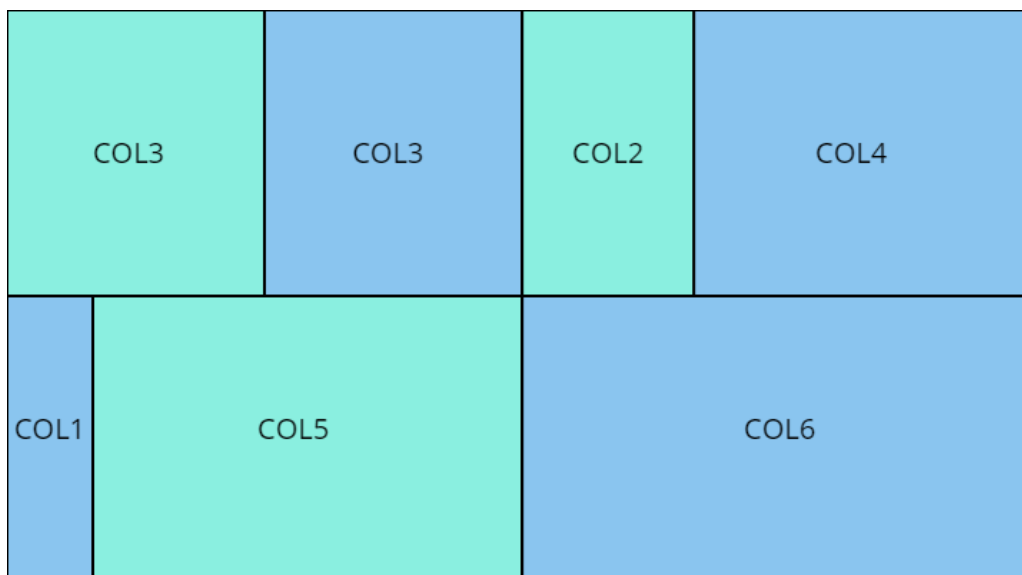
4 Další použité technologie

Následující kapitola popisuje další technologie použité k tvorbě pluginu - frameworky Bootstrap a JQuery k usnadnění implementace rozhraní a zajištění kompatibility mezi různými prohlížeči a frameworky Google Charts API a Google Maps API k zobrazení map.

4.1 Bootstrap

Framework Bootstrap byl původně vytvořen designéry Twitteru v roce 2010 ke snadnější tvorbě interních nástrojů pro Twitter. V roce 2011 byl poté vydán veřejně jako open-source projekt a do dneška zůstává jedním z nejpopulárnějších front-end frameworků [1].

Bootstrap slouží k snadnému vytváření webových stránek pomocí jeho komponent, například různých formulářů, štítků, vyskakovacích upozornění a dalších. Další důležitou funkcí je grid systém, pomocí kterého se každá stránka rozdělí na dvanáct sloupců a obsahu se poté pomocí předdefinovaných tříd **col-x** nastaví počet sloupců, které má zabírat, viz obrázek 4.1. Sloupce lze navíc vnořovat do sebe a upravovat jejich velikosti, uspořádání a další vlastnosti v závislosti na velikosti zobrazení.



Obrázek 4.1: Sloupce v Grid systému frameworku Bootstrap

Bootstrap zajišťuje responzivní design a konzistenci napříč všemi pod-

poroványi prohlížeči, což jsou v současné verzi 4.2.1 všechny populární prohlížeče (Chrome, Firefox, Safari, Edge a Android Browser).

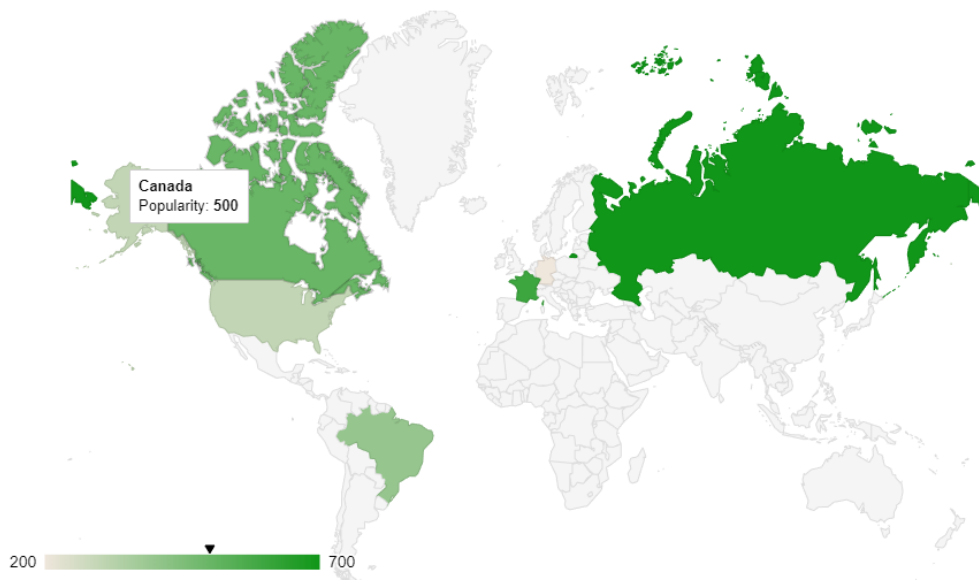
4.2 jQuery

Knihovna jQuery byla poprvé zveřejněna v roce 2006, kdy bylo jejím hlavním úkolem usnadnění přístupu k elementům na stránce (odtud také pochází název, query = dotaz, tedy dotaz na element) [16].

V dnešní době se knihovna rozrostla o mnoho dalších funkcí, jako je například snadné použití AJAX requestů, reakce na události pomocí listenerů, animace, nebo widgety pro uživatelské rozhraní v knihovně jQuery UI. Velkou výhodou jQuery při používání WordPressu je přímá podpora v jeho jádře - knihovnu není nutné importovat z vlastního souboru nebo externího zdroje, ale je už zabudovaná v každé instalaci WordPressu.

4.3 Google Charts

Google Charts je knihovna od společnosti Google, sloužící k vizualizaci dat. Kromě tradičních grafů (například sloupcových, čárových, koláčových, nebo korelačních diagramů) podporují také graf typu *GeoChart* - grafy vztahující data k místům na mapě státu, kontinentu nebo města [3].



Obrázek 4.2: Příklad vizualizace GeoChart

Data lze zobrazit buď jako hustotní mapu pro daný kraj, viz obrázek 4.2, nebo jako značky na daných pozicích, podobně jako u klasického Map API. Grafy GeoChart jsou ideální k zobrazení porovnání jednoduchých dat na mapě, ale pokud je vyžadována interaktivita mapy (například přibližování, trasy atp.), je nutné použít komplexnější framework, například Google Maps.

4.4 Google Maps

Webová mapová služba Maps od Google, česky Mapy Google je pravděpodobně nejpopulárnější služba svého typu, s pokrytím 99% světa a bilionem uživatelů měsíčně [4].

Zabudovat Google Maps do stránek lze pomocí několik API podle využití na stránkách - například pomocí JavaScript API lze vytvářet komplikovanější interaktivní mapy s vlastními vrstvami a značkami, zatímco Static API vytváří pouze statický obrázek konkrétní adresy. Oproti GeoChart z frameworku Google Charts jsou mapy plně interaktivní - umožňují přibližování a pohyb po mapě, nebo třeba přidání značek do přesných zeměpisných souřadnic. Mají však podstatně větší paměťové nároky a jejich použití je zpoplatněno podle počtu vyžádaných map za měsíc.

5 Emigrace do USA

Tato kapitola popisuje úskalí vyhledávání předků či vzdálených příbuzných způsobené emigrací do USA. Nejčastějším problémem je transformace původně českého příjmení do podoby srozumitelnější pro anglicky mluvící, druhým nejčastějším problémem potom bývají německy psané dokumenty, které ještě více znemožňují vyhledávání původně českých obyvatel.

5.1 Amerikanizace českých jmen a příjmení

První část se zabývá popisem transformací českých jmen a příjmení při emigraci českých občanů do Spojených Států, která probíhá buď hned při vstupu do země, nebo přirozeně časem ať už dobrovolně českými přistěhovalci pro lepší začlenění do společnosti, nebo nedopatřením ze strany amerických obyvatel, například špatným zápisem jména kvůli odlišné výslovnosti [18]. V závěru se kapitola věnuje zvláštnosti českých příjmení - přechylování.

5.1.1 Křestní jména

Křestní jména v České republice jsou, jak již název napovídá, obvykle křesťanského původu [23]. Většina jmen je tedy původem z Bible (např. Adam, Eva, Pavel), případně mají původ přímo slovanský - obvykle složenina slovesa a podstatného jména či zájmena, například Jaroslav = oslavující jaro. Tradiční anglická křestní jména jsou též křesťanského původu a proto je podle článku *The Americanization of Given Surnames* [17] k mnoha českým jménům možné najít snadnou anglickou variantu, například Anthony = Antonín, Jana = Jane. Článek [17] dále popisuje překlad jmen slovanského původu a to buď překladem do významově podobného jména (např. Bohuslav = Theodore; obojí znamenající oslavující Boha), nebo podle vlastní volby nositele jméno, pokud je přeložené anglické jméno nezvyklé nebo vůbec neexistuje (např. tradiční české jméno Václav bývá nejčastěji překládáno jako James).

5.1.2 Příjmení

Podle knihy *Naše a cizí příjmení v současné češtině* [20] můžeme příjmení dělit na Příjmení *deonymická*, tj. utvořená z vlastních jmen - jmen původně křesťanských, jmen místních nebo jmen zemí, národů a krajů, nebo *deapelativní*

označující či vyjadřující - mající za základ polohu, vlastnosti, řemesla a zaměstnání, živočichy, rostliny, přírodní jevy, věci, označení dějů a neskloňná slova jako příslovce nebo citoslovce.

Na rozdíl od jmen křestních, která často čeští přistěhovalci přijímají ihned při příjezdu jsou česká příjmení podle článku *The Americanization of Czech Surnames* [18] obvykle ponechána v původní podobě, případně s odstraněnou diakritikou při zápisu. Výjimkou jsou příjmení s přímým anglickým překladem (například Černý = Black, Kovář = Smith), ale i u těchto příjmení jich část zůstává v češtině, protože anglický překlad je příliš netradiční (například Vomáčka = Womatchka, nikoliv Sauce).

Druhá nebo třetí generace narozená ve Spojených Státech už ale obvykle na původním příjmení tolik nelpí a většinou přijme amerikanizovanou verzi. Díky této opožděné transformaci jsou ale příjmení často zkomolená a to i příjmení, která by mohla být přeložena přímo - například český Jestřáb podle článku *The Americanization of Czech Surnames* [18] po několika generacích přijal jméno Chestrob, ačkoliv by existoval přímý a použitelný překlad Crane.

Příjmení tvořená z vlastních jmen většinou přeložit nelze vůbec a proto zůstávají v české podobě, případně s přepisem pro usnadnění správné výslovnosti v angličtině (například Vašíček = Vashicheck). Spousta původně českých příjmení též místo přepisu nebo překladu do angličtiny byla místo toho germanizována, čímž vzniklo ve Státech již zavedené příjmení.

5.1.3 Přechýlená příjmení

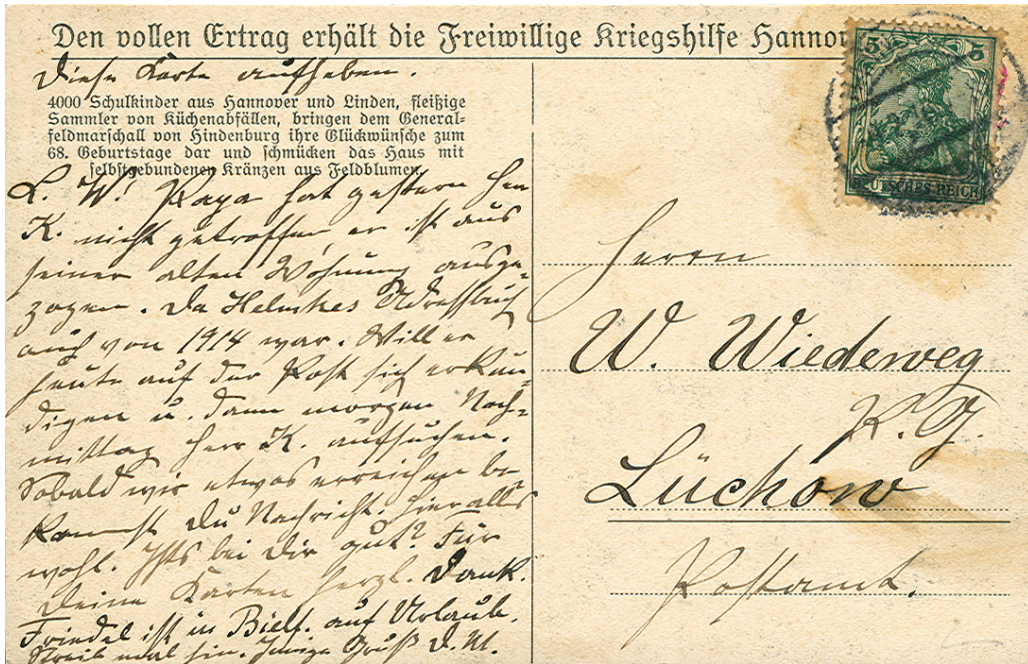
Zvláštností českých příjmení oproti angličtině je přechylování ženských příjmení. Vytvoření přechýleného příjmení se pro česká příjmení podle článku *Přechylování příjmení v češtině* [19] řídí přesnými dvanácti pravidly v závislosti na koncovce a slovním základu mužského příjmení.

Ačkoliv pro rodilého mluvčího je tvorba přechýleného příjmení většinou triviální, pro cizince může být přechylování příjmení, u nichž není pouze mechanicky dodána koncovka -ová komplikované, obzvláště pro příjmení přechylovaná podle podstatných jmen nebo příjmení, u kterých se vypouštějí samohlásky před koncovkou.

5.2 Německé názvy a písma

Druhá část kapitoly se zabývá dokumenty emigrantů z Česka. Dokumenty psané před rokem 1918 (za dob Rakousko-Uherska) nebo mezi roky 1939 a 1945 (během okupace Německem - Protektorát Čechy a Morava) byly často

psány v německém jazyce s použitím německých ručních písem, nejčastěji Kurrent nebo Švabach (viz obrázek 5.1), které je obtížné nejen přečíst, ale pro potomky českých emigrantů žijící celý život v USA může být komplikací i určit o jaké písmo se jedná. Místo českých názvů měst byly také většinou použity názvy německé, což vyhledání rodného města předků ještě více komplikuje, jelikož většina menších měst a vesnic se již pod těmito názvy nedá nalézt.



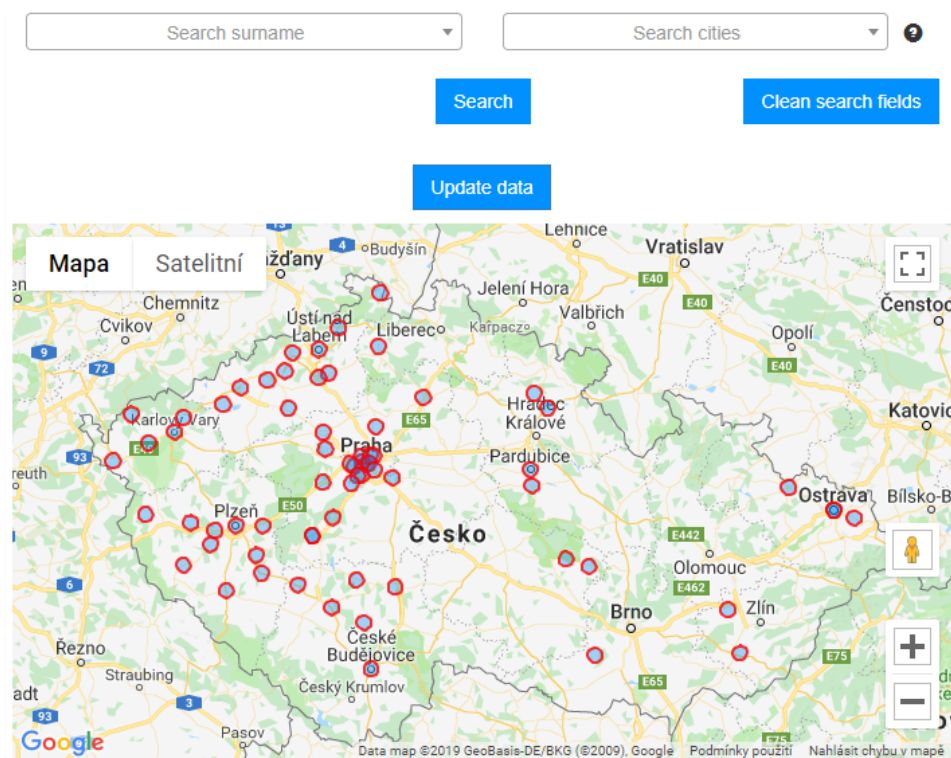
Obrázek 5.1: Příklad německého ručně psaného textu [15]

6 Analýza

Tato kapitola se věnuje analýze původních stránek na webu CATV věnujících se problematice genealogie. Bude zde letmo popsán původní pluginu pomocí kterého byla implementovaná interaktivní mapa pro vyhledávání četností a datové zdroje, které používá. V závěru se kapitola věnuje požadavkům na nový plugin.

6.1 Původní stav

Před úpravami se na webových stránkách Czech-American TV¹ v sekci věnující genealogii (věda o rodové historii) nacházelo několik stránek: stránka věnující se v teorii českým jménům a převody mezi češtinou a angličtinou, stránka s odkazy na externí nástroje (např. český online telefonní seznam), a stránka s interaktivní mapou četností příjmení (viz obrázek 6.1).



Obrázek 6.1: Původní interaktivní mapa

¹<https://www.catvusa.com>

Četnostní mapa zároveň sloužila jako překladač německých názvů měst a zobrazení výsledného překladu na mapě. Interaktivní mapa byla implementována pomocí původního pluginu, ostatní stránky byly statické a byly tedy zapsány přímo v databázi WordPressu.

6.2 Data pro plugin

Databáze původního pluginu obsahuje tři tabulky - **cities_translate** s překlady německých názvů do češtiny, tabulku **people** s četnostmi příjmení v obcích s rozšířenou působností (dále jen obce) a tabulku **cities**, obsahující indexované obce (s Prahou rozdělenou na městské části 1-22) a kraje.

6.2.1 Česká příjmení a jejich četnosti

Tabulky **people** a **cities** používané k vyhledávání četnosti v jednotlivých obcích jsou převzaty z oficiálních stránek Ministerstva Vnitra², kde byla data umístěna ve formátu XLS.

Tyto tabulky mají pro databázi velmi nevhodnou strukturu - v původním XLS dokumentu četností má každý řádek (odpovídající jednomu příjmení) 244 sloupců, kde názvy jednotlivých sloupců odpovídají čtyřmístným kódům obcí. Některé sloupce nejsou navíc kódem obce, ale “součtovým” sloupcem - kódy 1-14 odpovídají součtům pro kraje a kód 3000 pro celou republiku.

Kódy obcí, krajů a součtový kód 3000 a odpovídající názvy jsou poté umístěny v číselníku v samostatném souboru XLS. Obě tyto XLS tabulky byly uloženy do databáze původního pluginu, viz obrázky 6.2 a 6.3.

id_people	int	PK
surname	varchar	
surname_no_dia	varchar	
surname_female	varchar	
3101	int	
⋮	⋮	
3000	int	

Obrázek 6.2: Schéma původní tabulky People

Do databáze jmen byly dále doplněny verze příjmení bez diakritiky a ženská verze příjmení. Ženská příjmení byla bohužel doplněna strojově - buď

²<https://www.mvcr.cz/>

code_city	int	PK
city	varchar	
lat	float	
lng	float	

Obrázek 6.3: Schéma původní tabulky Cities

doplněním koncovky -ová nebo zaměněním koncovky -ý za -á (u příjmení tvořených z přídavných jmen). Jelikož původní databáze nebyla rozdělena na mužská a ženská příjmení, vzniká v databázi zdvojení koncovek (například Novákováová) a mužská příjmení u kterých se ženské verze netvoří mechanicky, ale je nutno například vypustit samohlásku mají v databázi zapsané neplatné ženské verze.

6.2.2 Historické názvy měst

K překladu německých názvů měst a vesnic na české slouží poslední tabulka `cities_translate`. Data v ní jsou čerpána z článku Seznam německých názvů obcí a osad v Česku³ na Wikipedii, který čerpá z lexikonu historických míst Čech a Moravy, který byl sepsán v roce 1998 dvaceti německými a dvaceti českými historiky [22]. Podobně jako u tabulek pro četnosti jsou informace pouze připraveny z článku a použity v databázi, viz obrázek 6.4.

id	int	PK
city	varchar	
region	varchar	
superior_city	varchar	
city_german	varchar	
note	varchar	

Obrázek 6.4: Schéma původní tabulky Cities Translate

Kromě překladů (kde sloupec `city` je český název a `city_german` německý) obsahuje tabulka sloupec `region`, který značí spadá-li vesnice / město do Čech, Moravy nebo Slezska, sloupec `superior_city`, značící větší město

³https://cs.wikipedia.org/wiki/Seznam_německých_názvů_obcí_a_osad_v_Česku

u kterého se nachází a sloupec **note** s poznámkou (například pokud jde o již zaniklou osadu).

6.3 Požadavky na nový plugin

Požadovány jsou následující funkce nově vytvořeného pluginu:

1. Přepis anglického příjmení na české, např. Vashicheck na Vašíček.
2. Vytvoření ženské (přechýlené) varianty příjmení k mužskému příjmení.
3. Překlad anglického jména z databáze překladů, např. překlad George na Jiří.
4. Překlad historického německého názvu města na současný český název, např. Pilsen na Plzeň.
5. Zobrazení textu v německém písmu (např. Švabach) pro ruční porovnání s ručně psanými dokumenty.
6. Integrace četnostní mapy původně používané na stránkách CATV do nového pluginu.
7. Možnost tisku výsledků ostatních částí pluginu pomocí tlačítka, bez nutnosti tisknout celou webovou stránku.
8. Všechny funkce pluginu budou obsahovat uživatelskou nápovědu k jejich používání.
9. Vložení video návodu k vytvořenému pluginu do původně statické stránky obsahující informace o Genealogii a odkazy na externí zdroje. U návodu bude seznam témat s tlačítky, která v něm umožní přetočení na dané téma.
10. Uživatelské rozhraní na administrační straně, umožňující přidání nových překladů jmen a příjmení.

Kromě funkčních požadavků existují také následující mimofunkční požadavky:

1. Snadná instalace a nastavení pluginu, kterou zvládne i člověk s minimální nebo žádnou znalostí programování.
2. Responzivní design podporující mobilní zařízení i počítače se zachováním stejné funkcionality.

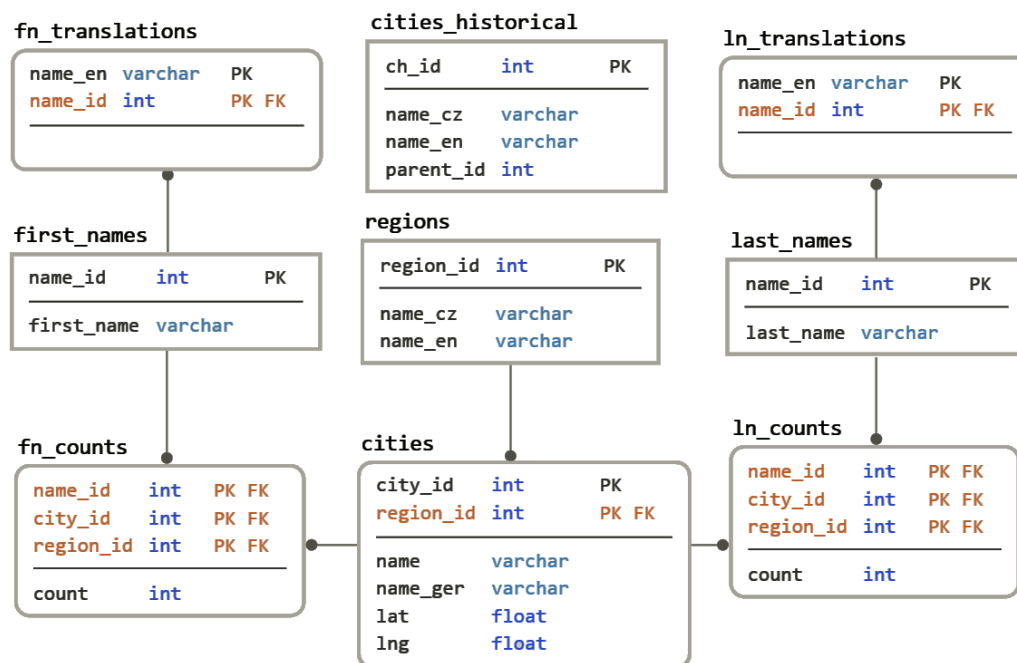
3. Rychlé provádění dotazů - žádný dotaz by neměl trvat déle než pár sekund.
4. Uživatelské rozhraní pro návštěvníky webu i administrátory v anglickém jazyce.
5. Zadávání vstupů musí počítat s nepřítomností českých a německých znaků na klávesnici uživatele.
6. Plugin musí být zabezpečen proti útokům jak na návštěvníky webu, tak na web samotný.

7 Návrh

Následující kapitola se věnuje návrhu nového pluginu. V první části popisuje novou strukturu databáze jmen, příjmení a měst. Dále se pak věnuje uživatelskému rozhraní - návrhu funkcí pluginu a jejich rozdělení na jednotlivé stránky. V závěru kapitoly je popsáno rozhraní pluginu pro jeho administraci.

7.1 Databáze

Ačkoliv vytvořený plugin používá stejné zdroje dat jako původní plugin, struktura databáze byla značně pozměněna. Přibyly tabulky s četnostmi křestních jmen a záznamy krajů byly přesunuty z tabulky **cities** do vlastní tabulky **regions**, čímž byly mimo jiné odstraněny “součtové” sloupce krajů, které se dají řešit součtem v MySQL dotazu a přiřazením měst do krajů.



Obrázek 7.1: Nové schéma databáze pluginu

Nejdůležitější změnou je však vytvoření pomocných tabulek **ln_counts** a **fn_counts**, pomocí kterých se rozděluje původní tabulka četností příjmení **people** a přidaná tabulka četností křestních jmen - města jsou se jmény

a příjmeními v relaci N:M, která je v databázi tvořena dvěma relacemi 1:N (viz obrázek 7.1).

Posledními přidanými tabulkami jsou tabulky obsahující překlady křestních jmen a příjmení **fn_translations** a **ln_translations**. Z tabulek byly také odstraněny nepotřebné sloupce, jako například poznámky u překladů německých měst nebo příjmení bez diakritiky a v ženské podobě.

7.2 Uživatelské rozhraní

Uživatelské rozhraní pluginu bude rozděleno do čtyř samostatných celků, z nichž každý bude umístěn na vlastní stránce a dynamicky zobrazen pomocí vlastního shortcode. Všechny stránky budou umístěny pod kategorii Genealogy v hlavním menu webu CATV.

- První stránka bude obsahovat funkce pro převod anglických jmen a příjmení na české - přepis a překlad jmen a příjmení a utvoření ženské varianty z mužského příjmení.
- Druhá stránka bude určena pro práci s německými dokumenty, obsahující překladač německých názvů na české a utilitu pro převod německých písem do latinky.
- Třetí stránka bude vyhrazena pro interaktivní mapu pro četnosti příjmení z původního pluginu.
- Na poslední stránce se bude nacházet video návod k obsluze pluginu a seznam odkazů na další zdroje, převzatý z jedné z původních stránek

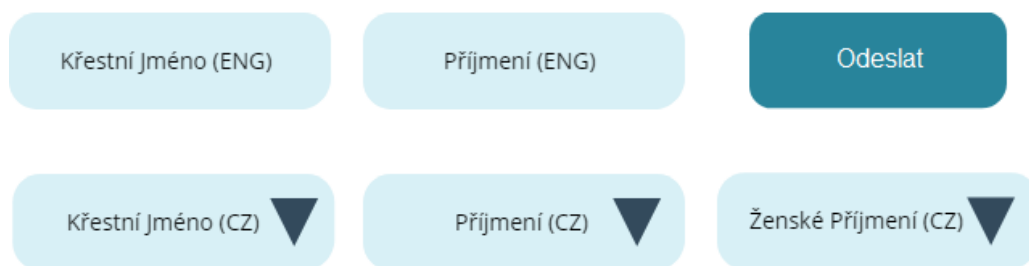
7.2.1 Překlad anglických jmen a příjmení

Stránka pro převod jmen a příjmení na české bude vyžadovat dvě vstupní pole - anglické křestní jméno a anglické příjmení. K funkci nebude potřeba zadávat hodnoty do obou polí a výpočty výsledků obou budou probíhat samostatně.

K přepisu z angličtiny na původní českou variantu je potřeba vrátit zpět části nahrazené pro snadnější anglický zápis a výslovnost. Kromě vrácení odstraněné diakritiky (např. Novak = Novák) se může jednat také o nahrazení skupin znaků jinými s ekvivalentní anglickou výslovností (např. Cheyka = Čejka). Pravidla pro přepis těchto znaků budou umístěna ve vlastním textovém souboru pro jejich snadnou úpravu bez nutnosti zasahovat do kódu

pluginu. Křestní jména se budou hledat přímo v databázi překladů. U příjmení bude navíc třeba vytvořit ženskou variantu - změnou koncovky podle pravidel popsaných v článku Přechylování příjmení v češtině [19]. Všechny výsledky bude nutno kontrolovat v databázi, aby se zajistilo že výsledek skutečně odpovídá používanému českému jménu / příjmení.

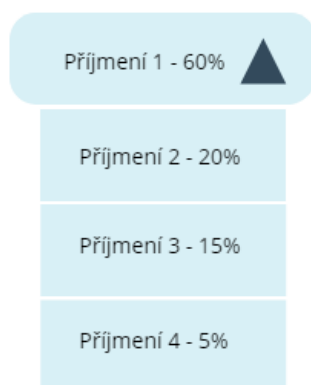
Výstupem pak budou tři rozbalovací seznamy - jeden pro zobrazení všech možných variant křestního jména, jeden pro varianty příjmení a jeden pro možné ženské varianty vybraného mužského příjmení, viz obrázek 7.2.



The image shows a user interface for translating names. It consists of two rows of three rounded rectangular buttons each. The top row contains three light blue buttons with the text 'Křestní Jméno (ENG)', 'Příjmení (ENG)', and a dark blue button with 'Odeslat'. The bottom row contains three light blue buttons with dropdown arrows, labeled 'Křestní Jméno (CZ)', 'Příjmení (CZ)', and 'Ženské Příjmení (CZ)'.

Obrázek 7.2: Návrh rozhraní pro překlad anglických jmen a příjmení

Rozbalovací pole místo jednoduchých textových polí jsou nutná z důvodu několika možných správných výsledků - pro jedno jméno/příjmení se může vytvořit několik českých verzí. Stejně tak pro mužské příjmení není pevně daná ženská alternativa. Pro snadnější výběr výsledku budou výsledky seřazeny podle relativní četnosti (počet výskytů jména dělený součtem výskytů všech jmen ve výsledcích) a tato četnost bude ve výsledcích vypsána jako procentuální pravděpodobnost výsledku, viz obrázek 7.3.



The image shows a vertical list of four light blue rounded rectangular items, representing a dropdown menu. Each item contains text and a small triangle icon. The items are: 'Příjmení 1 - 60%' with an upward-pointing triangle, 'Příjmení 2 - 20%', 'Příjmení 3 - 15%', and 'Příjmení 4 - 5%'.

Obrázek 7.3: Detail rozbalovacího menu

7.2.2 Práce s německými dokumenty

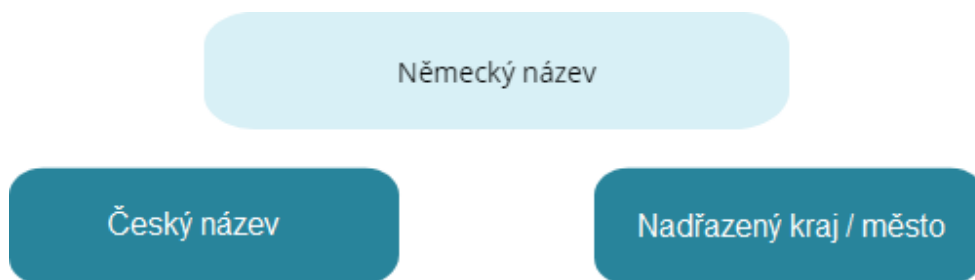
Druhá stránka se věnuje práci s německými dokumenty - překládání historických německých názvů a porovnávání psaných textů s německými písmi. Překládání historických názvů obcí je snadno implementovatelné - vyhledání v databázi a zobrazení výsledků a potřebuje tedy pouze jedno vstupní pole, které bude podporovat automatické doplňování stejně jako v původním pluginu.

Výstupem budou dostupné informace o městě vybraném z automatického doplnění, viz obrázek 7.4. Jelikož je vybráním výsledek pevně daný, může se výsledek zobrazit v textovém poli a není nutné použít rozbalovací seznam jako u jmen a příjmení.



Obrázek 7.4: Návrh rozhraní pro překlad německých názvů

Porovnávání ručně psaných textů by se ideálně řešilo pomocí rozpoznání textu, ale vzhledem k netradičním písmům a různým rukopisům je toto řešení velmi obtížně realizovatelné. Snadnějším řešením je zobrazení libovolného textu ve vybraném fontu, které umožňuje zobrazit zadaný text v různých fontech simulující ručně psaná písma a následné porovnání s reálným dokumentem uživatelem samotným. K porovnání textu bude tedy potřeba jedno vstupní pole pro zadaný text, seznam písem pro výběr a tabulka znaků vybraného písma, aby byl uživatel schopen určit které z nich má použít. Výstupem této části bude textové pole se zadaným textem zobrazeným ve vybraném fontu, s dostatečně velkou velikostí textu aby se písmo dalo snadno porovnat s reálnými dokumenty.

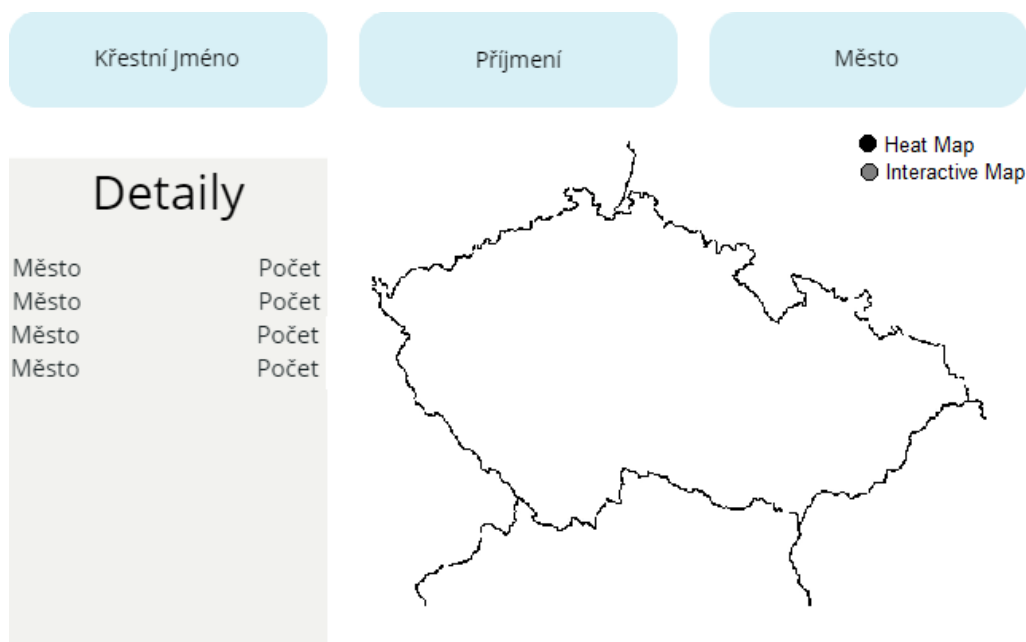


Obrázek 7.5: Návrh rozhraní pro porovnání německých písem

7.2.3 Mapa četnosti příjmení

Předposlední stránkou v uživatelské části pluginu bude stránka s interaktivní mapou četnosti příjmení. Překlad německých názvů měst bude přesunut na vlastní stránku a nahrazen vyhledávačem pouze měst které mají v databázi data o své poloze (obce s pověřeným obecním úřadem) a lze je proto na mapě snadno zobrazit.

Tato města bude možné hledat také pod německým názvem, jako tomu bylo u původního pluginu. Dále přibude možnost vyhledávání křestních jmen a s ní odpovídající vstupní pole.



Obrázek 7.6: Návrh rozhraní mapy distribuce příjmení

Mapa samotná bude implementována pomocí dvou frameworků: Google

Maps API, který používala původní mapa a Google Charts, který umožňuje zobrazení přehledné vektorové mapy hustoty obyvatelstva ve formátu SVG. Jako výchozí se bude zobrazovat SVG mapa, interaktivní mapa pomocí původních Google Maps bude stále k dispozici pomocí přepínače.

7.2.4 Video návod k obsluze pluginu

Poslední sekci pluginu bude stránka s videonávodem k ovládání jednotlivých funkcí pluginu a odkazy na další užitečné stránky mimo web CATV, například digitální archivy matrik a online verzi českého telefonního seznamu. Video návod bude umístěn na serveru YouTube.com a bude do stránky vložen pomocí tagu iframe. Pod návodem se bude nacházet seznam funkcí pluginu s časy, kdy se jim návod věnuje. Pod video návodem bude umístěn seznam dalších užitečných odkazů, editovatelný přímo jako součást WordPressové stránky.

7.2.5 Administrátorské rozhraní

Pro správu pluginu bude existovat jedna stránka v administrátorském rozhraní WordPressu, zařazená pod výchozí menu nastavení (settings). Administrátorské rozhraní bude umožňovat přidávání překladů jmen a příjmení, kontrolu stavu používané databáze a její případný import, viz obrázek 7.7.



Obrázek 7.7: Návrh administrátorského rozhraní pluginu

8 Implementace

Tato kapitola se zabývá adresářovou strukturou pluginu, prací s databází a implementací uživatelského rozhraní a všech funkčních i mimofunkčních požadavků uvedených v kapitole 6.3.

8.1 Struktura pluginu

Plugin je strukturován podle best practices doporučených plugin developer handbookem [13]. V hlavní složce pluginu je umístěn hlavičkový soubor odpovídající jménem složce pluginu, `catv_genealogy_tools.php`. Dále kořenový adresář obsahuje složku `includes`, které obsahují php soubory pro serverové výpočty a složky `public/admin`, které obsahují prezentační vrstvu uživatelského a administrátorského rozhraní. Další složkou je složka `common`, obsahující soubory které jsou využity v uživatelském i administrátorském rozhraní pluginu. Posledními složkami v adresáři pluginu jsou složky `data`, obsahující CSV soubory pro naplnění databáze pluginu a složka `images`, obsahující obrázky potřebné v uživatelském rozhraní.

PHP soubory (kromě hlavičkového) tvoří třídy, do kterých jsou umístěny jednotlivé funkce. Tím je odstraněna nutnost unikátních jmen funkcí a stačí unikátní jména tříd, čehož se dosáhne snadno použitím prefixu specifického pro plugin - v případě tohoto pluginu je to prefix `Gt`, odvozený z názvu pluginu - Genealogy Tools.

8.2 Databáze pluginu

Prvním krokem vývoje bylo zpracování dostupných dat a vytvoření databáze, z které bude plugin čerpat. Tato kapitola se věnuje postupu zpracování dat, vytvoření databáze a jejímu následujícímu používání.

Zpracování dat

Původní Excelové soubory četností jmen a příjmení byly děleny po 65000 záznamech do několika listů a byly proto ručně zkopírovány do nových tabulek ve formátu CSV. K vytvoření dat pro naplnění databáze byla vytvořena samostatná utilita v programovacím jazyce Python - parser (viz kód 1), který z CSV souborů původních dat vytváří datové soubory ve struktuře odpoví-

dající databázi, vhodné k přímému importu. Python skript parser.py i data potřebná k jeho funkci jsou k dispozici na příloženém DVD ve složce **parser**.

Kód 1 Psuedokód parseru

```
# id jmena
nameId = 0;
# csv tabulka z MVCR
table = load(origTable.csv);
# vytvorena tabulka
newTable = create(newTable.csv);

for row in table:
    # preskakovani slozenych jmen a jmen s malym vyskytem
    if " " in row.name or row.totalCount < 10:
        continue;
    nameId += 1;
    for city in row.cities:
        # preskakovani mest s nulovymi vyskyty
        if city.count == 0:
            continue;
        # preskakovani kraju a souctoveho sloupce
        if city.id <= 3000:
            continue;
        # zapsani do nove tabulky
        newTable.write(nameId+" "+city.id+" "+city.count);
```

Utilita při svém běhu ignoruje záznamy s mezerou (například příjmení složená ze dvou) a jména a příjmení s celkovým počtem menším než 10 záznamů, u kterých je velmi nízká pravděpodobnost vyhledávání americkými uživateli. Dále program sčítá četnosti jednotlivých městských obvodů Prahy (Praha 1 - Praha 22) do celkového součtu pro celou Prahu, jelikož rozdělení na obvody není pro cizince důležité.

8.2.1 Vytvoření tabulek a import dat

K práci s databází v pluginu slouží třída **Gt_Database**, která vytváří databázovou strukturu při aktivaci pluginu. Díky zredukované velikosti dat (soubory všech tabulek mají dohromady méně než 30MB, záznamy v databázi pak celkem 150MB oproti původním 350MB) není problém při aktivaci pluginu též databázi rovnou naplnit daty z vytvořených souborů v podsložce data pomocí MySQL příkazu **LOAD INFILE**, viz kód 2.

Kód 2 Příklad importu dat pomocí MySQL příkazu LOAD DATA

```
LOAD DATA LOCAL INFILE '/data/' . $table . ".csv" IGNORE
INTO TABLE " . $table . "
FIELDS TERMINATED BY ','
ENCLOSED BY '\"'
LINES TERMINATED BY '\r\n';
```

8.2.2 Komunikace s databází

Další funkcí třídy pro práci s databází je uchovávání odkazu na databázové spojení, které ostatním třídám pluginu umožňuje snadné provádění dotazů. Třída má proto jako atribut vytvořen objekt třídy **wpdb**, ve kterém jsou uloženy přihlašovací údaje k databázi - plugin využívá vlastní databázi oddělenou od hlavní databáze WordPressu, jelikož hlavní databáze je každý týden zálohovaná, což je u velké statické databáze pluginu nežádoucí. Třída si také uchovává pole jmen jednotlivých tabulek a umožňuje tak jejich přejmenování bez nutnosti úprav ve všech třídách komunikujících s databází.

8.3 Rozhraní pluginu

Dalším krokem byla implementace navrženého uživatelského rozhraní pluginu. Uživatelské rozhraní na klientské straně je implementováno třídou **Gt_Html_Public**, jejíž soubor se nachází společně se soubory skriptů a stylů používanými k zobrazení uživatelské části rozhraní pluginu ve složce **public**. Rozhraní každé uživatelské funkce pluginu lze zobrazit pomocí vlastního *enclosing* shortcode - shortcode, do kterého se dá vepsat vlastní text (například popis dané funkce), se kterým plugin může dále pracovat.

Použitím vlastního shortcode pro každou funkci oproti původnímu návrhu kdy měl být použit jeden shortcode pro celou stránku umožňuje rozdělení funkcí nejen mezi více stránek ale i změnu pořadí ve kterém jsou zobrazeny na jednotlivých stránkách. Použitím enclosing shortcode se pak zajišťuje že administrátoři webu budou moci přidat popisy k jednotlivým funkcím bez nutnosti zasahovat do kódu pluginu. Popis implementace rozhraní funkcí je umístěn do podkapitol jednotlivých funkcí v kapitole 8.5.

K zobrazení uživatelského rozhraní pro administraci pluginu slouží třída **Gt_Html_Admin**, nacházející se ve složce **admin** společně s odpovídajícím JavaScript souborem. Na rozdíl od rozhraní na klientské straně se administráční strana nezobrazuje pomocí shortcode, ale přidává se přímo do administráčního rozhraní WordPressu pomocí funkce **add_submenu_page()**.

Stejně jako u klientské strany je i implementace administrační strany detailněji popsána v samostatné kapitole 8.5.6.

8.4 Komunikace se serverem

Další částí kapitoly je implementace mezi webovým serverem a klienty. Veškerá komunikace se serverem probíhá dynamicky pomocí AJAX, který je na serverové části obsluhován třídou **Gt_Ajax**. Třída kromě samotného zpracování AJAX požadavků (volání funkcí jiných tříd a odeslání výsledků, případně chybových hlášení) také zajišťuje validaci a verifikaci příchozích zpráv. V klientské části pluginu jsou požadavky implementovány pomocí knihovny jQuery, která je součástí jádra WordPressu.

Odesílané požadavky obsahují jméno akce (prefixované pro zamezení kolizí s jinými částmi WordPressu), nonce vytvořenou při načtení stránky a vstupy potřebné pro danou operaci (například anglické příjmení u přepisu). Pro předání url souboru **admin-ajax.php** a platné nonce bez nutnosti používání inline JavaScript kódu vepsaného přímo v uživatelském rozhraní jsou tyto údaje předávány pomocí funkce **wp_localize_script()**, dle doporučení WP codexu [7].

Request		
Parametr	Hodnota	Popis
<code>_ajax_nonce</code>	81e2edb546	nonce předaná ze serveru
<code>action</code>	<code>gt_transcribe</code>	jméno fce na serveru
<code>name</code>	novak	zadané příjmení
<code>type</code>	lname	typ vstupu (příjmení)
Response		
Property	Hodnota	Popis
<code>status</code>	success	info o úspěchu fce
<code>results</code>	NOVAK: - id: 49117 - percentage: 0.16 NOVÁK: - id: 49126 - percentage: 99.84	pole objektů výsledků, indexované jmény

Tabulka 8.1: Příklad komunikace se serverem - přepis příjmení

Odpovědi serveru jsou ve formátu JavaScript Object Notation (JSON). Všechny zprávy mají společnou property **status** - řetězec určující byla-li

operace na serveru vykonána úspěšně. Pokud se operace nezdařila, je řetězec nastaven na “error” a zpráva obsahuje další property **errorMsg**, obsahující řetězec popisující chybu. Pokud se operace zdařila, je Status nastaven na “success” a druhou property zprávy je pole **results**, které obsahuje výsledky provádění funkce. Příklad zprávy a odpovědi je k vidění v tabulce 8.1.

8.5 Funkční požadavky

Hlavní částí implementace pluginu je vývoj funkcí samotných, popsany v následující kapitole, rozdělené na podkapitoly podle jednotlivých funkčních celků, které jsou dále děleny dle požadavků z kapitoly 6.3.

8.5.1 Překlad a přepis jmen

První funkční celek se věnuje vyhledání českých alternativ k anglickým jménům a příjmením a tvoří ho funkční požadavky 1-3. Kapitola také popisuje implementaci funkce automatického doplňování, která se ukázala během vývoje téměř nezbytnou.

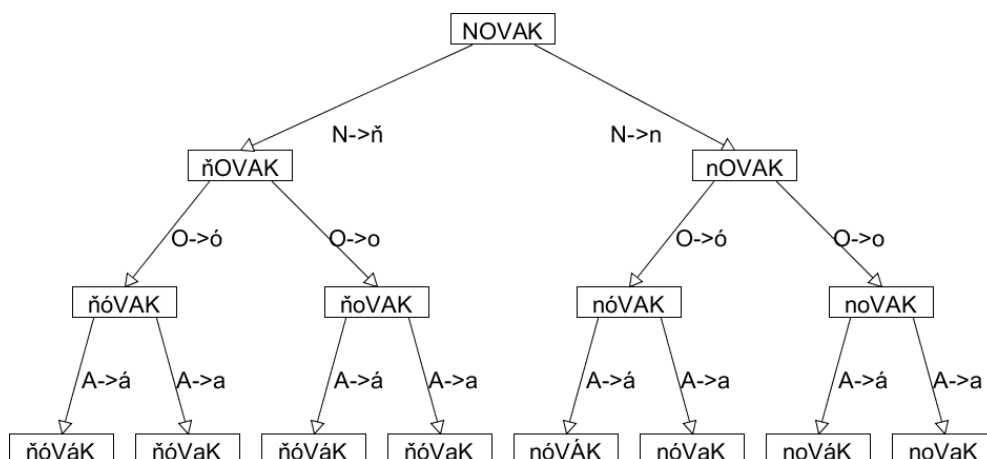
Přepis příjmení

Prvním funkčním požadavkem je přepis anglického příjmení na české. Přepis probíhá postupným aplikováním přepisovacích pravidel na vstupní řetězec, a je zajištěn funkcemi třídy **Gt_Transcriber**.

Přepisovací pravidla jsou uložena ve vlastním souboru ve tvaru $X>y$, kde X je jeden nebo více nahrazovaných znaků a y libovolný počet znaků (včetně žádného znaku), kterými jsou nahrazeny. Původní řetězec je před startem algoritmu převeden na velká písmena a nahrazené znaky jsou psány malými písmeny, čímž se na ně nebudou další pravidla aplikovat a zamezí se tím smyčkám.

Pravidla jsou aplikovaná zleva doprava a je nutné počítat s možností že pravidlo nebude použito - například nechceme přidávat interpunkční znaménko nad všechna možná písmena - z příjmení Novak by tak například vznikla jediná výsledná varianta Ňóvák. Při nalezení nahrazovaných znaků v řetězci jsou proto vytvořeny dvě varianty nahrazeného řetězce - jednu s aplikováním pravidla a druhou s přepsáním vybraných znaků na stejné v malých písmenech. Na oba tyto řetězce se poté rekurzivně zavolá stejná funkce, čímž se vytváří strom s výsledky, viz obrázek 8.1.

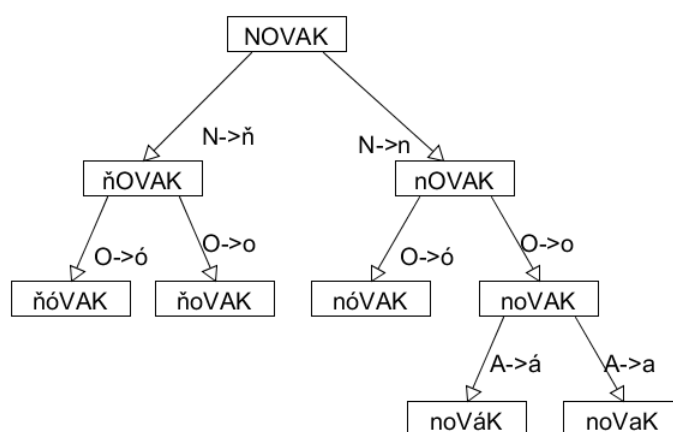
Při velkém počtu přepisovacích pravidel a s delším vstupním řetězcem se tato úloha stává velmi výpočetně náročnou - nahrazení se musí provést



Obrázek 8.1: Graf variant pro příjmení Novak; počáteční pravidlo N->ň

pro všechny možné kombinace a poté výsledky zkontrolovat oproti databázi příjmení. I pro zdánlivě jednoduchá příjmení tak vzniká mnoho výsledků, například pro příjmení Novak to je celkem 15 variant, viz obrázek 8.1, a to pouze s pravidly pro přidání diakritiky. Všechny varianty by se navíc procházely několikrát - pro každé pravidlo jednou. Pro příjmení Novak by tak reálný strom byl třikrát větší než strom na obrázku 8.1 - z kořene by vedly další 2 větve pro každé ze zbylých dvou pravidel.

Delší příjmení jsou proto výpočtově náročná a je nutné výpočet výsledku optimalizovat. Toho je dosaženo ořezáváním vyhledávacího stromu - příjmení až po právě nahrazené znaky se porovnává s databází a pokud není nalezeno žádné příjmení začínající těmito znaky, je větev ukončena, viz obrázek 8.2.



Obrázek 8.2: Graf variant pro příjmení Novak s ořezáváním

Pro všechny nalezené varianty se po aplikování všech možných prepisovacích pravidel (nebo jejich přeskočení) v databázi vyhledá celkový počet výskytů daného příjmení v České Republice a pokud je nenulový, uloží se společně s příjmením do pole s výsledky, které se po skončení prepisovacího algoritmu odešle zpět.

Kromě ořezávání stromu si algoritmus též udržuje pole již navštívených variant, pro které již výpočet neopakuje a ušetří tak ještě více výpočtového času - všechny varianty se naleznou již při prvním rekurzivním průchodu a reálný strom je tedy shodný se stromem na obrázku 8.2. Pseudokód výsledného algoritmu je k vidění v kódu 3.

Kód 3 Pseudokód algoritmu přepisu

```
visited = []; # seznam navstivenych retezcu
found = []; # seznam nalezenych jmen
db; # databaze jmen v~CR
rules = []; # pole prepisovacich pravidel

# rekurzivni funkce k~vyhledavani prepisu
function transcribe(name):
  if name in db:
    found.add(name); # pridavani do seznamu nalezenych
  if name in visited:
    return; # preskakovani navstivenych retezcu
  else:
    visited.add(name); # pridavani do seznamu navstivenych
  for rule in rules:
    # aplikovani prepisovaciho pravidla
    newName = apply rule to name
    # kontrola jmen se stejnym zacatkem v~databazi
    if (any name in db starts with newName.substring):
      transcribe(newName);
```

Uživatelské rozhraní přepisu příjmení bylo po prvním testování pluginu koncovými uživateli oproti původnímu návrhu pozměněno, viz obrázek 8.3. Funkce pro přepis příjmení a překlad jmen byly rozděleny do vlastních formulářů, aby se zdůraznil rozdíl mezi častějším přepisem u příjmení a častějším překladem křestních jmen.

Ačkoliv se touto změnou zdůrazňuje že u příjmení půjde ve většině případů spíše o přepis zkomolené anglické verze, mohou se vyskytnout případy kdy příjmení bylo přeloženo - hlavně příjmení jejichž přímý překlad odpovídá běžnému příjmení v angličtině. Vstupní příjmení je proto také přeloženo pomocí databáze překladů a nalezené překlady jsou přidány k výsledkům přepisu.

Last Name Changes ?

The tool below can add removed diacritical marks and revert spelling changes.

Last Name in English:

Czech Last Name:

No name entered.
▼

[Map](#)
[Copy](#)
[Print](#)

Send

First Name Translation ?

The tool below can translate many English first names to their Czech equivalent.

First Name in English:

Czech First Name:

No name entered.
▼

[Copy](#)
[Print](#)

Send

Obrázek 8.3: UI přepisu příjmení a překladu jmen

Ženská varianta příjmení

Druhou požadovanou funkcí pluginu je vytvoření ženské varianty k mužskému příjmení, která je též zajištěna třídou **Gt_Transcriber**. Ačkoliv je utvoření přechýleného příjmení dáno přesnými pravidly, některá z nich jsou příliš komplikovaná pro algoritmické řešení - například rozlišení příjmení utvořených z přídavných a podstatných jmen.

Vzhledem k relativně nízkému počtu pravidel však není problém aplikovat všechna použitelná a výsledky porovnat s databází a vybrat pouze existující příjmení. Vytváření možných alternativ probíhá pomocí regulárních výrazů - funkcí **preg_replace**, která vyhledá koncovku a podle daného pravidla ji nahradí přechýlenou variantou, viz tabulka 8.2.

Regulární výraz	Náhrada	Příklad příjmení
/EC\$/u	COVÁ	CHOVANEK - CHOVANCOVÁ
/NĚK\$/u	ŇKOVÁ	VANĚK - VAŇKOVÁ
/Ě\$/u	ĚTOVÁ	HRABĚ - HRABĚTOVÁ
/(Ů ŮV ŮJ ÝCH)\$/u	OVÁ	TOMŠŮ - TOMŠOVÁ

Tabulka 8.2: Příklad regulárních výrazů k určení přechýleného příjmení

U vyhledávání přechýleného příjmení také došlo k zratelnější změně v uživatelském rozhraní, viz obrázek 8.4. Na základě zpětné vazby bylo vyhledávání přesunuto do vlastní sekce, aby uživatel nemusel k vyhledání ženské varianty napřed hledat přepis mužského příjmení. Možnost vyhledání ženské varianty z výsledků přepisu zůstala k dispozici i nadále pomocí odkazu umístěného pod vyhledávačem, viz obrázek 8.3.

Oddělením této funkce do samostatného formuláře však nastal nový problém - zadávané mužské příjmení bude často muset obsahovat české znaky,

Obrázek 8.4: UI vyhledávání přechýlených příjmení

kteří uživatel nebude mít na klávesnici k dispozici. Vstupní pole je proto rozšířeno o funkci automatického doplňování.

Automatické doplňování

Automatické doplňování nebylo sice součástí funkčních požadavků pluginu, ale mnohonásobně ulehčuje obsluhu, a tím i vývoj, většiny ostatních funkcí. Navíc je díky implementaci automatického doplňování pokryt mimofunkční požadavek na zadávání pouze ve znacích anglické abecedy. Funkce je implementována ve třídě **Gt_Autocomplete** a používá jednoduchý SQL dotaz, který vyhledává záznamy začínající vstupním textem, viz kód 4.

Kód 4 SQL dotaz pro automatické doplnění příjmení

```
SELECT name_id AS id , name FROM 'last_names'
WHERE name LIKE _utf8 $name
COLLATE utf8_general_ci
ORDER BY CHAR_LENGTH(name) ASC LIMIT 10
```

Pomocí nastaveného collation (pravidel pro porovnávání a řazení) dotaz také ignoruje diakritiku, takže například pro zadání příjmení Novák stačí napsat Novak a vybrat ho z nabídnutého seznamu. Tím se značně usnadňuje práce uživatelů kteří nemají k dispozici klávesnici s českými nebo německými znaky. Pro zamezení vizuálního zahlcení navrhanými možnostmi se zobrazuje pouze prvních 10 výsledků seřazených podle délky, viz obrázek 8.5.

Klientská část automatického doplňování je zajištěna pomocí widgetu knihovny jQuery UI, která má za úkol ulehčit implementaci uživatelského rozhraní a animovaných efektů při zachování kompatibility napříč prohlížeči. Jako alternativa bylo testováno použití tagu `<datalist>`, který disponuje stejnou funkcionalitou, ale není kompatibilní se současnou verzí prohlížeče Safari a některými mobilními prohlížeči.

Last Name:

- NOVÁ
- NOVÁK
- NOVAK
- NOVÁČEK
- NOVÁČIK
- NOVANSKÁ
- NOVAKOVÁ
- NOVÁKOVÁ
- NOVÁČKOVÁ
- NOVAKOVSKÝ

Obrázek 8.5: Příklad automatického doplňování

Překlad křestních jmen

Třetím funkčním požadavkem je překlad anglických křestních jmen na české. Překlad je implementován vyhledáním v databázi a vrácením možných překladů seřazených podle počtu výskytů v ČR.

Podobně jako u překladů příjmení se i v případě křestních jmen může využít přepis - u jmen, která mají podobnou anglickou a českou verzi, například Thomas = Tomáš. Aplikováním přepisu na křestní jména tak lze najít i výsledky, které nejsou přímo uloženy v databázi překladů. Z tohoto důvodu je funkce překladu umístěna také ve třídě přepisů **Gt_Transcriber**, kde mají obě funkce snadný přístup k výsledkům překladů i přepisů. Uživatelské rozhraní funkce je jak již bylo zmíněno u příjmení oproti návrhu umístěno v samostatném formuláři, viz obrázek 8.3.

8.5.2 Práce s německými dokumenty

Druhý funkční celek se věnuje práci s dokumenty psanými v němčině. Obsahuje funkční požadavky číslo 4 (překlad historických názvů) a 5 (porovnávání ručně psaného textu).

Překlad historického německého názvu

Pro překlad německých názvů, který byl v původním pluginu součástí interaktivní mapy stačí stejně jako u křestních jmen opět vyhledání výsledků v databázi a jejich zobrazení.

Problémem je ale zadání správného vstupu od uživatele - nezanedbatelná část uživatelů nebude vědět přesný název města, který by byl nutný pro přímý překlad. Proto je zde stejně jako u přechýlených příjmení využito funkce automatického doplňování, která kromě názvu města zobrazuje i nejbližší větší město nebo kraj k rozlišení více měst se stejnými německými názvy, viz obrázek 8.6.



Obrázek 8.6: Příklad automatického doplňování německých názvů

Vzhledem k tomu že všechny požadované informace jsou již obsaženy ve výsledcích automatického doplňování, nemá tato funkce vlastní třídu na serveru a všechny dotazy jsou již pokryté automatickým doplňováním třídy **Gt_Autocomplete**.

Porovnávání s německými texty

Pátý požadavek navazuje na čtvrtý - jedná se o porovnávání německy psaného textu, které umožní uživatelům snadnější přečtení německých názvů z ručně psaných německých dokladů.

Zobrazování textu je realizováno pomocí sady volně dostupných fontů vytvořených Peterem Wiegalem¹. Fonty byly normalizovány na stejnou velikost a převedeny do více formátů pro větší kompatibilitu online utilitou Font Squirrel². Výstupnímu textu se poté pomocí CSS nastaví vlastnost **font-family** odpovídající vybranému písmu.

¹<http://www.peter-wiegel.de/>

²<https://www.fontsquirrel.com/>

Fontů je k dispozici celkem devět, kromě běžně používaného Šwabachu, Kurrentu a běžného německého psacího písma (kurzívy) je to například font Amptmann (profesionálnější obdoba Kurrentu, používaná pro technické dokumentace) nebo font Leipzig, který vznikl skenováním německých tištěných knih.

German Handwriting ?

This tool can help with reading older handwritten documents by inputting the text you believe is written on them in English and comparing the original with the text generated in German handwriting.

Text:

Special characters:

The quick brown fox jumps over the lazy dog

Font: Case:

Not sure which font to use?
Click this button to show the entire alphabet in selected font and letter case.

a ä b c d e

Obrázek 8.7: UI pro porovnávání německy psaných textů

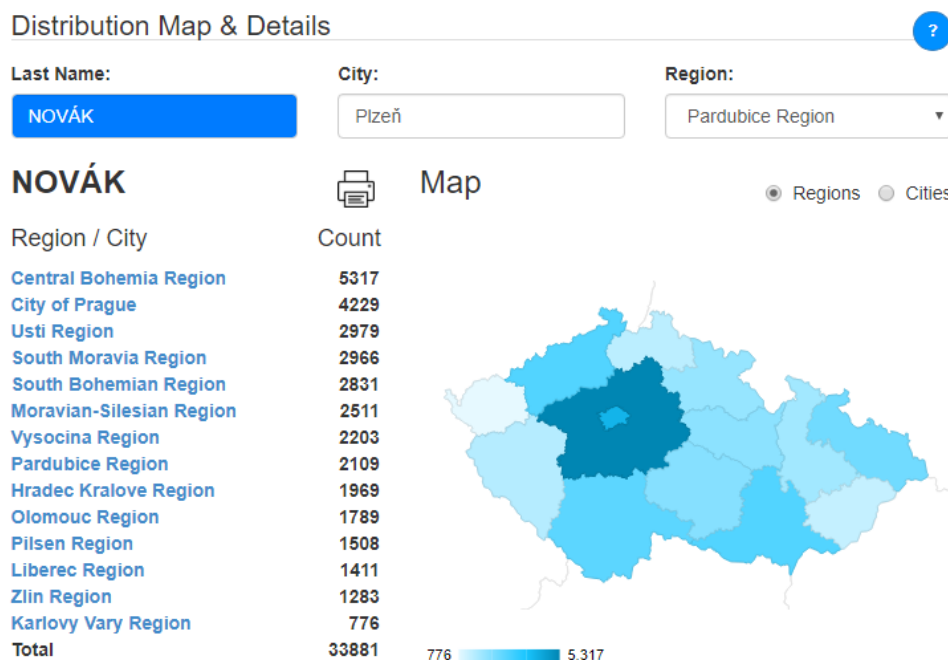
V uživatelském rozhraní proběhly dvě změny, viz obrázek 8.7 - u vstupního pole oproti návrhu přibyla sada tlačítek pro zadání znaků specifických pro německý jazyk (ä, ö, ü, ß), které běžný uživatel nemusí mít k dispozici na klávesnici a v pomocné abecedě znaků bylo ponecháno pouze jedno písmeno ve vybraném fontu pro lepší přehlednost, hlavně při menším rozlišení.

8.5.3 Zobrazení na mapě

Dalším požadavkem na uživatelskou část pluginu je integrace původní četnostní mapy do nového pluginu. Mapa byla oproti původní předělána a rozšířena o další funkce a možnosti zobrazení, a proto je i tato kapitola dále rozdělena na popisy jednotlivých funkcí. Mapa má dvě možnosti zobrazení - *Regions*, implementované pomocí frameworku Google Charts, a *Cities* implementované pomocí Google Maps. O serverovou funkcionalitu vyhledávání v mapě a zobrazení detailů měst a příjmení se stará třída **Gt_Details**. Z rozhraní mapy bylo oproti návrhu odstraněno vyhledávání křestních jmen,

kteřé bylo při testování matoucí pro uživatele - naznačovalo že je možné vyhledávání kombinací křestního jména a příjmení, k čemuž bohužel v databázi nejsou k dispozici data. Pole bylo nahrazeno rozbalovacím seznamem s kraji České republiky, který umožňuje zobrazit informace o jednotlivých krajích.

Vyhledávač četnosti příjmení

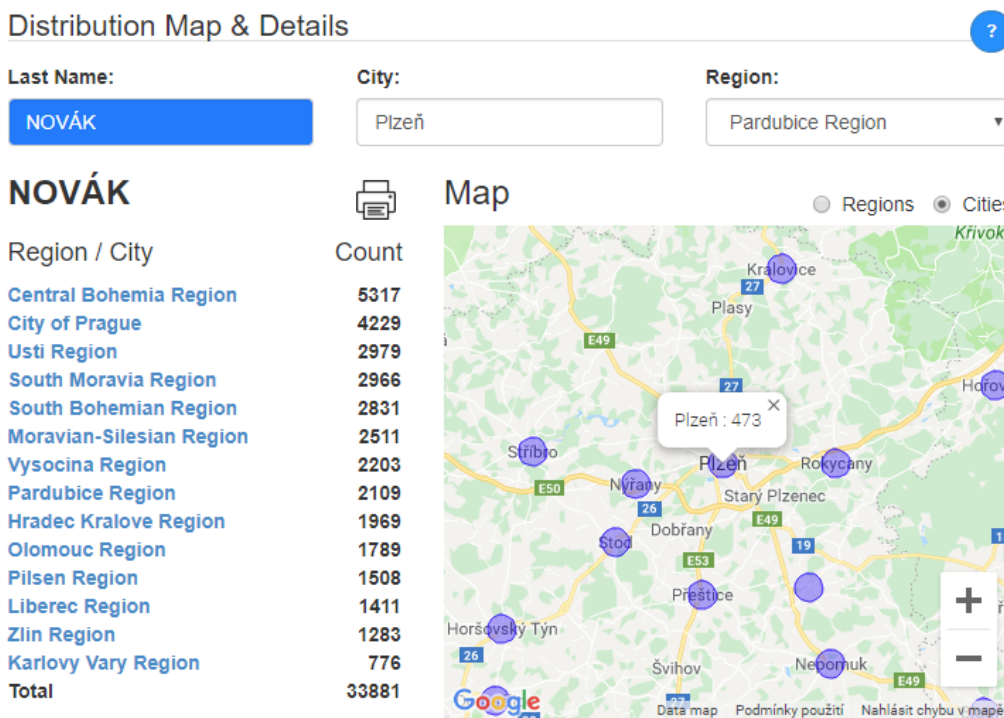


Obrázek 8.8: UI četnostní mapy příjmení pomocí Google Charts

Vedle mapy se po vyhledání příjmení nachází seznam krajů České republiky seřazených podle počtu obyvatel se zvoleným příjmením, který po kliknutí na libovolný kraj zobrazí seznam obcí v daném kraji a počty v jednotlivých obcích, viz obrázek 8.5.3. Rozkliknutí kraje lze provést také kliknutím na daný kraj v mapě.

Četnostní mapu lze též přepnout do druhého režimu cities, kde se v klasické Google Maps mapě zobrazí na místě každého města s alespoň jedním výskytem interaktivní kruh, který po kliknutí ukáže název města a počet obyvatel s vybraným příjmením, podobně jako na mapě v původním pluginu, viz obrázek 8.9.

Vstupní pole pro vyhledávání opět disponuje funkcí automatického doplňování, čímž se zaručí přesné a snadné zadání i pro uživatele, kteří nemají na klávesnici české znaky.



Obrázek 8.9: UI četnostní mapy příjmení pomocí Google Maps

Vyhledávač českých měst

Druhou částí interaktivní mapy je zobrazení měst na mapě. Tato funkce využívá zobrazení v režimu cities pomocí Google Maps a stejně jako vyhledávání příjmení má vstupní pole funkci automatického doplňování. Automatické doplňování navíc podporuje české i německé varianty názvů měst. Vybrané město se na mapě zobrazí jako tradiční značka (marker) Google Maps, viz obrázek 8.10.

V informacích vlevo od mapy se u vyhledávače měst zobrazuje kraj ve kterém se město nachází, jeho německý název, odkazy na další informace na wikipedii a seznam pěti nejpopulárnějších křestních jmen a příjmení přiřazených pod vybrané město.

Informace o krajích ČR

Poslední funkcí interaktivní mapy je zobrazení informací o vybraném kraji České Republiky, které nahradilo původně plánované vyhledávání četností křestních jmen. Informace o kraji využívají zobrazení v režimu regions, které pouze vyznačí kraj na mapě podobně jako u četnosti příjmení, viz obrázek 8.11.

Distribution Map & Details



Last Name:

NOVÁK

City:

Plzeň

Region:

Pardubice Region

Plzeň

Region:

Pilsen Region

German:

Pilsen

Wikipedia:

[Link](#)

Popular Last Names

	Count
ČERNÁ	561
ČERNÝ	516
NOVÁK	473
NOVÁKOVÁ	462
SVOBODOVÁ	389

Popular First Names

	Count
JAN	5538
JIŘÍ	5251
JANA	5189
PETR	4700
MARIE	4105

Map

Regions Cities



Obrázek 8.10: UI vyhledávače měst

Distribution Map & Details



Last Name:

NOVÁK

City:

Plzeň

Region:

Pardubice Region

Pardubice Region

Wikipedia:

[Link](#)

Unique First Names:

2123

Unique Last Names:

31369

Cities

Pardubice	Details
Chrudim	Details
Vysoké Mýto	Details
Svitavy	Details
Žamberk	Details
Litomyšl	Details
Moravská Třebová	Details
Ústí nad Orlicí	Details

Map

Regions Cities



Obrázek 8.11: UI informací o krajích

Informační část zobrazení krajů obsahuje stejně jako u vyhledávače měst odkazy na další informace na wikipedii a dále pak počet unikátních křestních jmen a příjmení ve vybraném kraji a seznam měst kraje s odkazy na zobrazení jejich detailů na mapě.

Propojení s ostatními částmi pluginu

K výsledkům vyhledávání jmen a příjmení se lze odkázat též z ostatních stránek pomocí tří parametrů požadavku GET:

- gt-id: id jména, příjmení, města nebo kraje - pro vyhledání v databázi
- gt-name: text vyhledávání - pro zobrazení ve vstupním poli
- gt-type: typ vyhledávání - fname, lname, city nebo region

Pokud jsou parametry přítomné, automaticky se po načtení stránky načtou odpovídající výsledky. Tohoto odkazování se využívá v ostatních částech pluginu, například u přepisu příjmení lze tak snadno rovnou vyhledat jeho četnosti pomocí tlačítka Map vedle výsledků, viz obrázek 8.12.



Obrázek 8.12: Příklad tlačítka pro zobrazení příjmení na mapě

8.5.4 Požadavky společné pro všechny funkce

Požadavky číslo 7 a 8 (tisk výsledků a nápověda pro uživatele) jsou součástí rozhraní všech zmíněných funkčních celků a jsou proto zařazeny do této samostatné kapitoly, popisující jejich implementaci.

Tisk výsledků

Prvním funkčním požadavkem společným pro všechny funkce je tisk výsledků. Tisk je spustitelný pomocí kliknutí na nápis “Print” u jednotlivých výsledků, případně tlačítkem s ikonou tiskárny u vyhledávače jmen na mapě. Tiskové zobrazení je poté tvořeno skrytými tagy `<div>` v uživatelském rozhraní, které mají pomocí tříd bootstrapu nastaveno zobrazení pouze při

tisku. Zároveň jsou všechny ostatní elementy na stránce při tisku skryty díky vlastnímu CSS stylu s využitím pravidla `@media print`. Při tisku je proto zobrazeno pouze logo CATV umístěno v tisknutelném elementu a výsledky funkce, u které bylo tisk spuštěn, viz obrázek 8.13.



Czech last names for VANECHCK:

VANIČEK (57.67%)
VANEČEK (41.94%)
VANIČEK (0.39%)

Obrázek 8.13: Příklad tisku výsledků

Zobrazení nápovědy

Druhým společným funkčním požadavkem uživatelské strany pluginu je zobrazení nápovědy k jednotlivým funkcím. Texty nápovědy jsou uloženy přímo v kódu třídy `Gt_Html_Public` a nelze je tedy upravovat bez úpravy kódu pluginu.

Zobrazení nápovědy na stránce lze provést tlačítky se symbolem `?` u nadpisů jednotlivých funkcí. Text nápovědy se po kliknutí zobrazí pod nadpisem v modrém rámečku, kterým je odlišen od ostatních textů s informacemi. Opětným stiskem tlačítka lze nápovědu skrýt.

8.5.5 Video návod

Kromě textové nápovědy byl také vytvořen video návod³ vytvořený nahráváním kroků potřebných k obsluze pluginu a přidáním doplňovacích textů využitých v nápovědách byl umístěn do čtvrté záložky pod menu Genealogy, kde se předtím nacházely informace k původnímu pluginu. Tyto informace byly ze stránky odstraněny a zůstal zde pouze video návod a odkazy na zdroje mimo CATV.

Pro manuál byl vytvořen vlastní shortcode ve třídě **Gt_Public_Html**, takže video je přímo zabudováno do pluginu a nikoliv staticky umístěné na stránce. Díky tomu je video ovlivněno responzivním designem frameworku Bootstrap a zároveň pod ním je zobrazen seznam jednotlivých témat ve videu a tlačítek, které po kliknutí přetočí video na odpovídající téma, podle funkčního požadavku číslo 9 (viz obrázek 8.14).



In this tutorial you can find help for the following topics. Click on the timestamp to rewind the tutorial to that section.

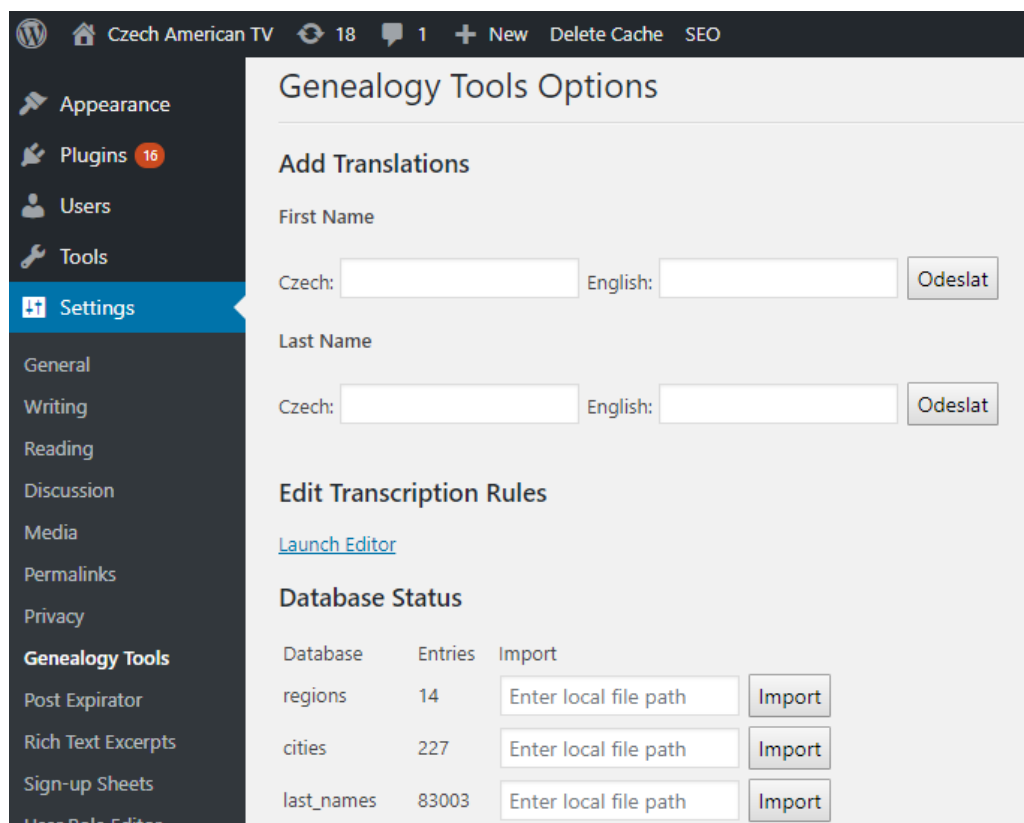
Changing Names	0:52
German "CZECH" Terminology	1:32
German Handwriting	2:23
Genealogy Map	4:04
Displaying Help	5:32
Copying & Printing	5:48

Obrázek 8.14: Stránka s video návodem

³<https://www.youtube.com/watch?v=GLYass12em8>

8.5.6 Administrace pluginu

Posledním funkčním požadavkem je možnost přidávání překladů do databáze administrátory webu. Rozhraní samotné je zařazeno pod jedno z hlavních menu WordPressu, **Settings** (nastavení) a je implementováno třídou `Gt_Html_Admin`.



Obrázek 8.15: Rozhraní pro administraci pluginu

Aby bylo zajištěno správné zadání českého jména nebo příjmení bez nutnosti použití české klávesnice, je opět použito automatické doplňování. V administrátorském rozhraní se dále nachází stav databáze (počet záznamů v tabulkách) s možností importu dat z lokálního disku. Dále byl přidán odkaz na editaci prepisovacích pravidel pomocí vestavěného editoru WordPressu, viz obrázek 8.15.

8.6 Mimofunkční požadavky

Další částí vývoje byla kontrola a případná úprava implementace pro splnění mimofunkčních požadavků na plugin. Většina mimofunkčních požadavků

byla zohledněna již při implementaci jednotlivých funkcí a proto nebylo nutné provádět velké úpravy již implementovaných částí.

8.6.1 Snadná instalace

Prvním mimofunkčním požadavkem je snadná instalace i nastavení pluginu. Díky snížení velikosti databáze lze plugin nainstalovat včetně všech datových zdrojů ze zip archivu, stejně jako jakýkoliv jiný plugin. Další obsluha z administrační části se pak týká pouze přidávání překladů a upravování, k čemuž je určeno rozhraní, jehož implementace byla popsána v kapitole 8.5.6.

8.6.2 Responzivní design

Dalším požadavkem je responzivní design použitelný pro mobilní zařízení i stolní počítače. Všechny funkce pluginu jsou plně responzivní díky frameworku Bootstrap, ale jakýkoliv jiný text vložený na stejnou stránku (například seznam odkazů pod videem z kapitoly 8.5.5) musí být vložen jako obsah do shortcode `[gt_text]`, čímž se zaručuje responzivita odpovídající pluginu a zároveň snadná editace textu přímo z prostředí WordPressu.

8.6.3 Rychlé provádění dotazů

Nový design databáze zdatelně urychlil dobu vyhledávání výsledků a všechny funkce byly implementovány se snahou o co nejlepší optimalizaci. Přesto se u některých dotazů při větší serverové zátěži může dotaz vykonávat několik sekund, obzvláště při přepisu delších jmen nebo vyhledávání velmi častého příjmení na mapě. Uživatelé jsou však vždy upozorněni, že funkce provádí výpočet a neměli by proto mít pocit že aplikace neodpovídá.

8.6.4 Uživatelské rozhraní v anglickém jazyce

Uživatelské rozhraní na straně pro návštěvníky i administrátory je kompletně v anglickém jazyce. Veškeré texty nápovědy, nadpisů i další informační texty použité v pluginu byly kontrolovány rodilým mluvčím.

8.6.5 Zadávání pouze znaků anglické abecedy

Požadavek na vyřešení situace, kdy uživatel nemá k dispozici znaky české nebo německé abecedy na klávesnici byl z většinové části vyřešen implementací funkce automatického doplňování, viz kapitola 8.5.1. U funkce zobra-

zování textu v německém písmu (kde nemůže být automatické doplňování použito, protože vstupy nejsou známé) je požadavek řešen přidáním tlačítek pro zápis německých znaků.

8.6.6 Bezpečnost

Posledním mimofunkčním požadavkem je zabezpečení pluginu pro návštěvníky i provozovatele webu. Během vývoje pluginu byly dodržovány bezpečnostní doporučení stanovená v Plugin Handbook [13] a plugin by tedy měl být zabezpečen proti útokům z externích zdrojů i případných nebezpečných pluginů na stejné instalaci WordPressu. Testování bezpečnosti je dále popsáno v kapitole 9.3.

9 Testování

Předposlední kapitola práce se věnuje testování implementovaných funkcí pomocí frameworku Selenium, testování pluginu v na kombinaci různých prohlížečů a zařízení a testování zabezpečení pluginu. těchto testů byl plugin také testován v počátečních fázích vývoje koncovými uživateli a podle zpětné vazby bylo upraveno uživatelské rozhraní, viz kapitola 8.

9.1 Systémové testy

Testy správnosti funkčních požadavků byly provedeny pomocí Selenium IDE, rozšíření pro webové prohlížeče Chrome a Firefox určené k automatizovanému testování webových stránek. Příklad testu pro funkční požadavek číslo 1 (přepis příjmení) je k vidění v tabulce 9.1.

Příkaz	Cíl	Hodnota	Popis
open	www.catvusa.com		Otevřít web
mouse over	linkText=Genealogy		najet na záložku Genealogy
click	linkText=Changing Names		Kliknout na odkaz Changing Names
click	id=gt-transcription-lname-input		kliknout do pole pro přepis příjmení
type	id=gt-transcription-lname-input	Vashicheck	napsat příjmení
send keys	id=gt-transcription-lname-input	ENTER	stisknout Enter
wait for element not editable	id=gt-transcription-lname	1000	zkontrolovat, že se výsledky načítají
wait for element editable	id=gt-transcription-lname	5000	počkat na výsledky
verify value	id=gt-transcription-lname	V A Š Í Č E K	zkontrolovat správnost

Tabulka 9.1: Příklad Selenium testu na funkci přepisu příjmení

Zároveň se v testech kontrolují i dva mimofunkční požadavky - požadavek na vykonání výpočtu v rádech několika sekund a požadavek na možnost zadávání českých či německých znaků bez nutnosti mít tyto znaky na klávesnici. Testy také kontrolují správnou integraci do stránek CATV - všechny testy začínají na hlavní stránce webu a k jednotlivým funkcím přistupují pomocí odkazů v menu stejným postupem jako uživatelé.

Testů je celkem 9, každý z nich označen číslem funkčního požadavku, jehož implementaci testuje a názvem. Testy mimofunkčních požadavků jsou součástí všech testů - při čekání na výsledky je nastavený timeout 5s, čímž je kontrolována rychlá odezva programu a při zadávání vstupů jsou použity pouze znaky anglické abecedy v kombinaci s funkcí automatického doplňování. U funkce zobrazování textu v německém písmu je také kontrolována funkcionality tlačítek pro zápis německých znaků.

Všechny testy byly provedeny na lokální kopii webových stránek i přímo na online verzi webu CATV. V případě testování online webu se vyjíměčně (v 1 případě z 10 spuštění všech testů) dosáhlo 5 sekundového timeoutu u testu výsledků interaktivní mapy, způsobeného zátěží na serveru. V ostatních případech proběhly testy na lokální i online kopii bez chyb. Soubor testů **genealogy_tools.side** je k dispozici na přiloženém DVD.

9.2 Podpora prohlížečů a OS

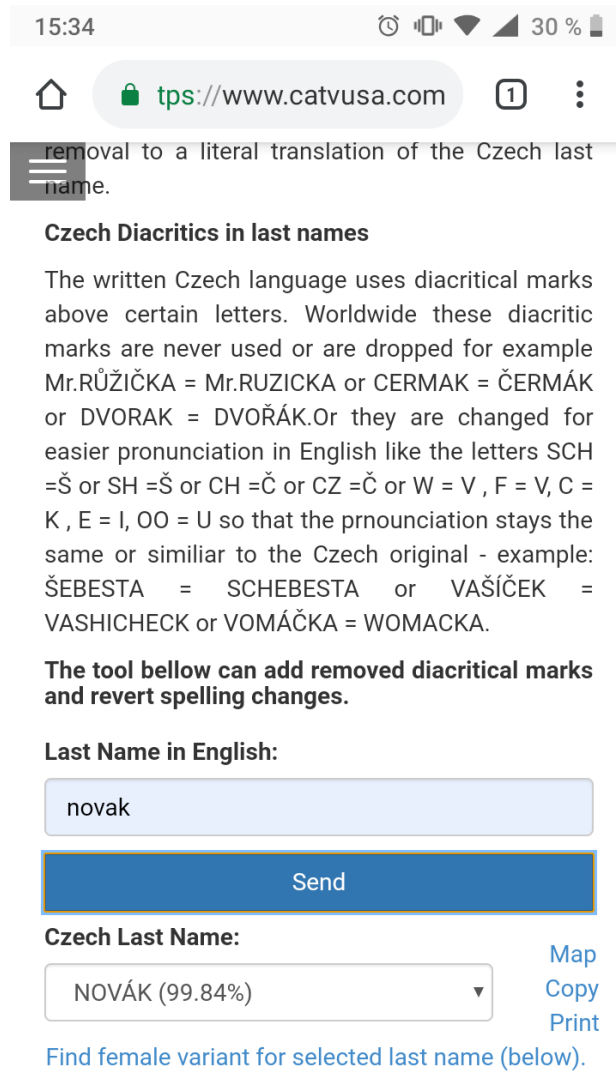
Druhou částí testování byla kontrola funkčnosti pluginu na různých prohlížečích a operačních systémech. Testování proběhlo na pěti nejběžnějších operačních systémech a webových prohlížečích. V tabulce 9.2 jsou písmenem X vyznačeny úspěšné testy, písmeny N/A nedostupné kombinace a pomlčkou kombinace, které nebyly testovány z důvodu nedostupnosti zařízení s vybraným operačním systémem.

OS	Chrome	Firefox	Opera	Safari	Edge
Windows 10	X	X	X	N/A	X
Android 9	X	X	X	N/A	X
Ubuntu 18.04	X	X	X	N/A	N/A
iOS12	X	-	-	X	-
macOS10.14	X	-	-	X	N/A

Tabulka 9.2: Příklad Selenium testu na funkci přepisu příjmení

Díky použití několika frameworků které slibují konzistenci napříč prohlížeči bylo dosaženo kvalitní podpory ve všech větších prohlížečích, s drob-

nými vizuálními rozdíly, nepodstatnými pro funkci pluginu. Díky použití frameworku Bootstrap a implementace rozhraní pomocí jeho responzivních tříd bylo navíc dosaženo i responzivního designu s dynamickým zobrazením na mobilních zařízeních, viz obrázek 9.1.



Obrázek 9.1: Ukázka UI na mobilním zařízení

9.3 Testování zabezpečení

Lokální kopie pluginu byla manuálně testována proti následujícím útokům:

- **SQL Injection** - všechna vstupní pole byla otestována proti SQL Injection pomocí dotazu `’;DROP TABLE last_names;’`, který by při

nedostatečném zabezpečení dotazů smazal tabulku příjmení. Všechna vstupní pole se prokázala jako ošetřená jak při přímém vepsání dotazu, tak při odeslání AJAX požadavku s ručně upravenými údaji pomocí konzole prohlížeče.

- **XSS** - stránka s mapou, která jako jediná obsahuje vypisování parametru URL do kódu stránky, a je proto náchylná na Cross-Site Scripting, byla testována změněním parametru **gt-name**, využitého při odkazování z jiných stránek, na `<script> alert("test"); </script>`. Skriptový tag byl z URL úspěšně odstraněn a vnitřní funkce **alert()** nebyla provedena.

10 Závěr

V teoretické části práce byl popsán systém pro správu obsahu WordPress se zaměřením na tvorbu a využití pluginů a další technologie použité k tvorbě výsledného pluginu. Dále byl též popsán problém amerikanizace českých jmen a příjmení a zapisování historických českých dokumentů v němčině. Dále byl v práci analyzován původní pluginu pro genealogii, dostupné datové zdroje a požadavky CATV na nový plugin. Podle analýzy bylo navrženo nové optimalizované schéma databáze pluginu a uživatelské rozhraní pro požadované funkce.

Pro nově vytvořený plugin byly použity dostupné zdroje dat v optimalizované struktuře vhodné k rychlému vyhledávání. Během vývoje byl kladen důraz na přehledné rozhraní a plugin byl podle zpětné vazby upravován tak, aby vyhovoval koncovým uživatelům. Plugin je již od ledna 2019 nainstalovaný do webové aplikace Czech-American TV a je používán návštěvníky webu. Pro plugin je dále plánováno rozšíření o další funkce a snaha o rozšíření nebo aktualizaci používané databáze.

Vytvořený plugin umožňuje americkým uživatelům zjistit nejpravděpodobnější původ jejich příjmení a nalezení města, ve kterém se nejčastěji příjmení vyskytuje na interaktivní mapě a to i pokud již o svých předcích mají minimum informací. Pro dokumenty o emigraci psané v německém jazyce ručním písmem umožňuje plugin překlad německých názvů měst a osad a zobrazení textu v několika německých písmech, který je pak možné porovnat s vlastními dokumenty. Dále plugin umožňuje snadné a přesné vyhledání přechýlených příjmení, což výrazně usnadňuje vyhledání předků v případě, kdy první předek přestěhovaný do USA byla žena. Kromě příjmení zvládá plugin i běžné překlady českých jmen do angličtiny, které bohužel nejsou tak spolehlivé, jelikož křestní jména si lidé typicky při emigraci do USA volili sami.

Plugin bohužel není schopen vyhledávat žijící vzdálené příbuzné, což je kvůli ochraně osobních údajů a nedostatku zdrojů a propojitelných API velice komplikovaná úloha. Plugin však poskytuje možnost si zjištěná data uložit nebo vytisknout a dále pokračovat ve vlastním hledání, ať již na online matrikách odkazovaných CATV, rodokmenových aplikacích, sociálních sítích nebo třeba osobní návštěvou matrik s vytištěnými výsledky vyhledávání.

Nad rámec funkčních požadavků byla do pluginu též přidána například funkce automatického doplňování, značně usnadňující zadávání údajů uživatelům bez české klávesnice. Dále byla kompletně předělána původní in-

teraktivní mapa - rozšířením o zobrazení bez nutnosti načítat zpoplatněné Google Maps, vyhledáváním informací o krajích České Republiky a propojením s ostatními funkcemi pluginu.

Správnost implementovaných funkcí pluginu byla testována funkčními testy pomocí frameworku Selenium IDE. Dále byla provedena kontrola kompatibility s populárními webovými prohlížeči a operačními systémy a kontrola funkčnosti na mobilních zařízeních. Závěrem testování byla manuální kontrola odolnosti na nejčastější útoky na webové stránky, SQL Injection a XSS.

Plugin je psán se snahou o modulární design a je proto velmi snadné ho rozšiřovat o další funkce - například o další německá ruční písma, nebo o vyhledávání četnosti křestních jmen, které se na stránkách CATV ukázalo jako nevhodné, ale mohlo by být použito jako součást pluginu v rámci jiné webové aplikace na bázi WordPressu. Díky použití frameworku Bootstrap používá plugin minimum vlastních stylů a je ho proto možné velice snadno přizpůsobit při změně grafického designu stránky kde je použit.

Přehled Zkratek

AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
CATV	Czech-American TV
CMS	Content Management System
CSS	Cascading Style Sheet
CSV	Comma Separated Values
GT	Genealogy Tools
HTML	HyperText Markup Language
IDE	Integrated Development Enviroment
JS	JavaScript
JSON	JavaScript Object Notation
OS	Operační Systém
PHP	PHP: Hypertext Preprocessor
SQL	Structured Query Language
UI	User Interface
URL	Uniform Resource Locator
WP	WordPress
wpdb	WordPress Database
XSS	Cross-Site Scripting

Literatura

- [1] *About Bootstrap* [online]. [cit. 15.01.2019]. Dostupné z: <https://getbootstrap.com/docs/4.2/about/overview/>.
- [2] *Usage Statistics and Market Share of Content Management Systems, January 2019*. [online]. [cit. 07.01.2019]. Dostupné z: https://w3techs.com/technologies/overview/content_management/all.
- [3] *Visualization: GeoChart* [online]. [cit. 15.01.2019]. Dostupné z: <https://developers.google.com/chart/interactive/docs/gallery/geochart>.
- [4] *Google Cloud including GCP & G Suite* [online]. [cit. 15.01.2019]. Dostupné z: <https://cloud.google.com/maps-platform/>.
- [5] *PHP: History of PHP - Manual* [online]. [cit. 07.01.2019]. Dostupné z: <http://php.net/manual/en/history.php.php>.
- [6] *Usage Statistics and Market Share of Server-side Programming Languages for Websites, January 2019* [online]. [cit. 07.01.2019]. Dostupné z: https://w3techs.com/technologies/overview/programming_language/all.
- [7] *AJAX in Plugins* [online]. [cit. 12.01.2019]. Dostupné z: https://codex.wordpress.org/AJAX_in_Plugins.
- [8] *Plugin API* [online]. [cit. 12.01.2019]. Dostupné z: https://codex.wordpress.org/Plugin_API.
- [9] *Roles and Capabilities* [online]. [cit. 12.01.2019]. Dostupné z: https://codex.wordpress.org/Roles_and_Capabilities.
- [10] *Writing a Plugin* [online]. [cit. 12.01.2019]. Dostupné z: https://codex.wordpress.org/Writing_a_Plugin.
- [11] *Creating Tables with Plugins* [online]. [cit. 12.01.2019]. Dostupné z: https://codex.wordpress.org/Creating_Tables_with_Plugins.
- [12] *Class Reference/wpdb* [online]. [cit. 12.01.2019]. Dostupné z: https://codex.wordpress.org/Class_Reference/wpdb.
- [13] *Plugin Developer Handbook* [online]. [cit. 12.01.2019]. Dostupné z: <https://developer.wordpress.org/plugins/>.
- [14] *Shortcodes | Plugin Developer Handbook* [online]. [cit. 12.01.2019]. Dostupné z: <https://developer.wordpress.org/plugins/shortcodes/>.

- [15] *File:Anonymer Fotograf PC 4000 Schulkinder aus Hannover und Linden ... Küchenabfälle ... Generalfeldmarschall von Hindenburg ... 68. Geburtstag Adressseite W. Wiedeweg Lüchow.jpg* — *Wikimedia Commons, the free media repository* [online]. [cit. 12.03.2019]. Dostupné z: https://commons.wikimedia.org/wiki/File:Anonymer_Fotograf_PC_4000_Schulkinder_aus_Hannover_und_Linden_..._Küchenabfälle_..._Generalfeldmarschall_von_Hindenburg_..._68._Geburtstag_Adressseite_W._Wiedeweg_Lüchow.jpg.
- [16] CHAFFER, J. – SWEDBERG, K. *Mistrovství v jQuery*. Computer Press, 2013. ISBN 978-80-251-4103-8.
- [17] DUDEK, J. B. The Americanization of Czech Given Names. *American Speech*. 1925, 1, 1, s. 18–22. ISSN 00031283, 15272133. Dostupné z: <http://www.jstor.org/stable/451927>.
- [18] DUDEK, J. B. The Americanization of Czech Surnames. *American Speech*. 1925, 1, 3, s. 161–166. ISSN 00031283, 15272133. Dostupné z: <http://www.jstor.org/stable/452540>.
- [19] KNAPPOVÁ, M. Přechylování příjmení v češtině (Pravidla a systematický přehled). *Naše řeč*. 1979, 62, 5, s. 225–233. ISSN 2571-0893.
- [20] KNAPPOVÁ, M. *Naše a cizí příjmení v současné češtině*. TAX AZ KORT, 2002. ISBN 80-238-8173-6.
- [21] MESSENLEHNER, B. – COLEMAN, J. *Building web apps with WordPress*. O'Reilly Media, 2014. ISBN ISBN978-1-449-36407-6.
- [22] POLÍVKA, M. – EBERHARD, W. – BAHLCKE, J. *Lexikon historických míst Čech, Moravy a Slezska*. Argo, 2001. ISBN 80-720-3402-2.
- [23] RAMEŠ, V. *Po kom se jmenujeme?* Libri, 3. vyd edition, 2003. ISBN 80-727-7197-3.
- [24] SCHWARTZ, B. *High performance MySQL*. O'Reilly Media, 2012. ISBN 9781449332471.
- [25] VRÁNA, J. *1001 tipů a triků pro PHP*. Computer Press, 2010. ISBN ISBN978-80-251-2940-1.

Přílohy

A - Uživatelská příručka

Instalace a nastavení pluginu

Před instalací pluginu je nutné nastavit přihlašovací údaje k databázi jmen zapsáním následujících řádek do konfiguračního souboru **wp-config.php** v kořenovém adresáři instalace WordPressu:

```
define( 'DB_GEN_NAME' , 'jmeno_db' );  
define( 'DB_GEN_USER' , 'uzivatel_db' );  
define( 'DB_GEN_PASSWORD' , 'heslo' );
```

K následné instalaci pluginu je třeba se řídit následujícími kroky, též k vidění na obrázku A.1:

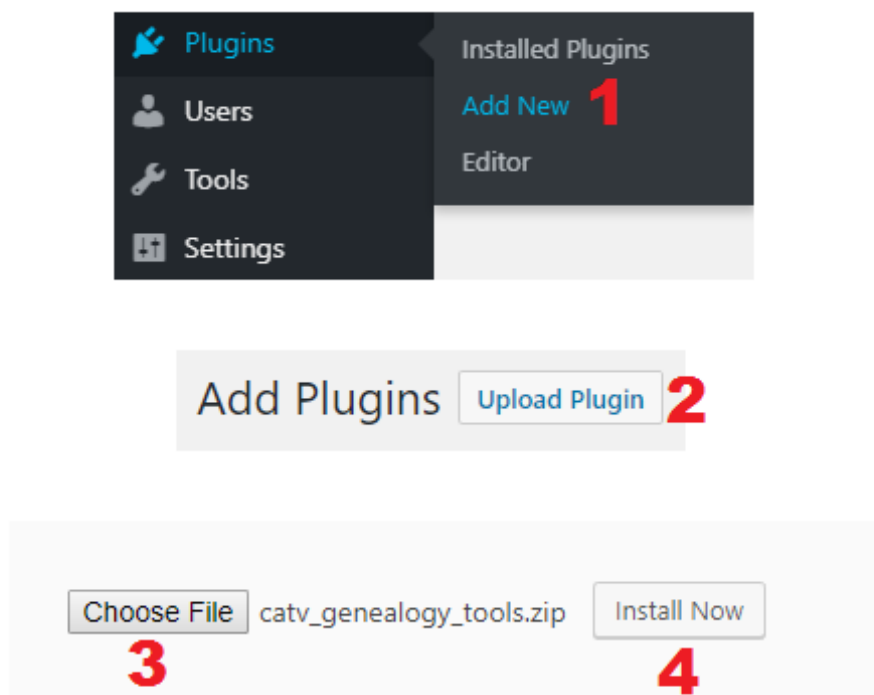
1. Kliknutím na menu **Plugins - Add New** v administračním rozhraní WordPressu.
2. Kliknutím na tlačítko **Upload Plugin** vedle nadpisu v horní části stránky.
3. Kliknutím na nově zobrazené tlačítko **Choose File** a vybráním instalačního zip archivu z disku.
4. Kliknutí na tlačítko **Install now**.

Plugin lze ihned po aktivaci aktivovat potvrzením dotazu na aktivaci, nebo ho aktivovat později přes menu **Plugins**.

Nastavení stránek a zobrazení

Ke správnému zobrazování grafických výstupů pro návštěvníky webu pomocí shortcodes je potřeba nastavit jména stránek, na kterých se shortcodes budou nacházet v hlavičkovém souboru pluginu **catv__genealogy__tools.php** změnou následujících řádek:

```
define( 'GT_PAGE_NAMES' , 'changing-names' );  
define( 'GT_PAGE_GERMAN' , 'german-czech' );
```



Obrázek A.1: Návod k instalaci pluginu

```
define( 'GT_PAGE_MAP', 'map' );
define( 'GT_PAGE_TUTORIAL', 'tutorial' );
```

Druhou možností je ponechat výchozí nastavení a pojmenovat vytvořené stránky tak, aby mu odpovídaly. Po nastavení je možné vkládat funkce do stránek pomocí následujících shortcodes:

[catv_gt_transcription_fname]	-přepis/překlad křestních jmen
[catv_gt_transcription_lname]	-přepis/překlad příjmení
[catv_gt_femvar]	-vyhledání přechýlen. příjmení
[catv_gt_ger_typer]	-psaní v německých písmech
[catv_gt_ger_city]	-překladač německých názvů měst
[catv_gt_map]	-interaktivní mapa
[catv_gt_text]	-libovolný text mezi funkcemi
[catv_gt_tutorial]	-video návod s obsahem

Vechny shortcode mohou být použity jako obyčejné i enclosing, v případě enclosing se pak ve funkci pod jejím nadpisem vypíše text zapsaný v shortcode.

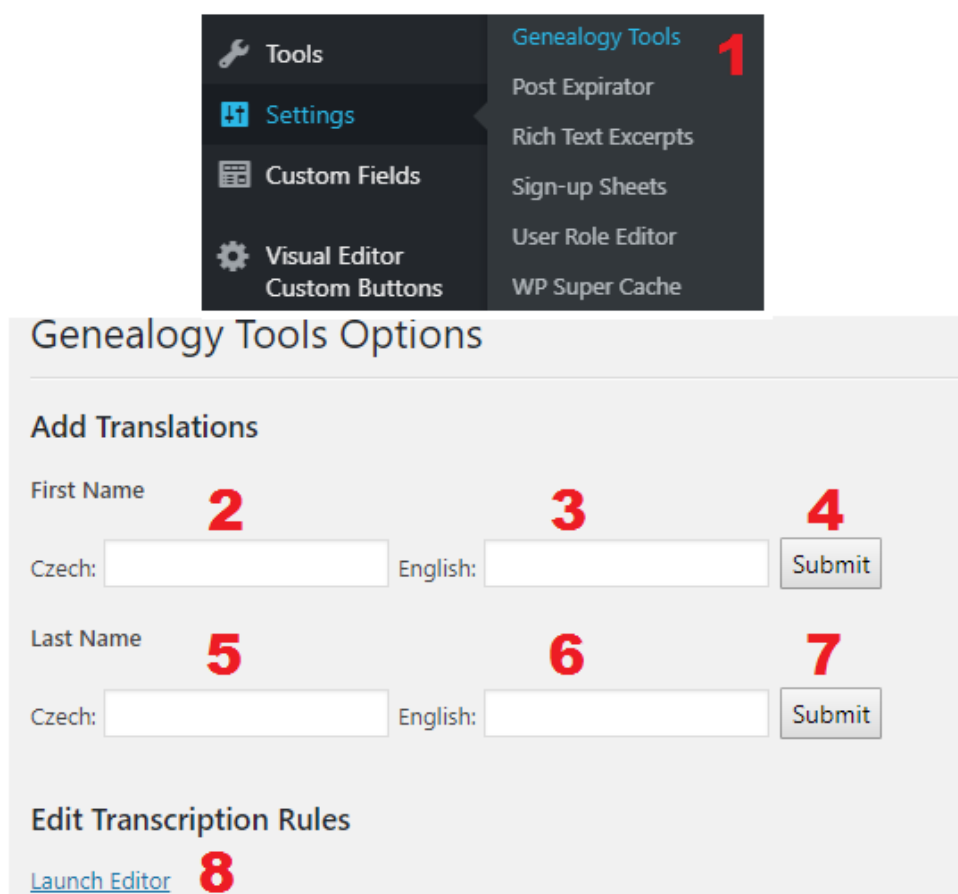
Ovládání pluginu

Administrační strana

Na administrační straně jsou k dispozici funkce pro přidávání překladů jmen a příjmení a funkce editace přepisovacích pravidel. Zobrazení těchto funkcí lze pomocí kliknutí na menu **Settings**, záložka **Genealogy Tools**, viz obrázek A.2, číslo 1.

Překlady křestních jmen a příjmení lze zadáním českého jména/příjmení do pole 2/5 na obrázku A.2, zadáním anglického překladu jména/příjmení do pole 3/6, a odesláním formuláře pomocí tlačítka 4/7.

Editace přepisovacích pravidel pomocí integrovaného editoru WordPressu lze spustit pomocí odkazu **Launch Editor** (číslo 8 na obrázku A.2) a potvrzením zobrazeného dialogového okna. Pravidla lze přidávat na libovolnou řádku v souboru, ale pouze jedno pravidlo na řádek. V souboru se mohou řádky komentovat pomocí znaku #.



Obrázek A.2: Návod k administraci pluginu

Uživatelská strana

Návod k používání všech funkcí pro návštěvníky webu je k dispozici pod tlačítkem nápovědy u každé funkce (modré tlačítko se symbolem ? vedle nadpisu), nebo ve vytvořeném videomanuálu na adrese <https://www.youtube.com/watch?v=GLYass12em8>.

B - Obsah přiloženého DVD

Na přiloženém DVD jsou následující soubory a složky:

- soubor **catv_genealogy_tools.zip**, obsahující zdrojový kód a data pluginu a zároveň sloužící jako instalační archiv.
- soubor **genealogy_tools.side** obsahující testy pro Selenium IDE.
- soubor **thesis.pdf** obsahující text této práce ve formátu pdf
- složka **thesis_source** obsahující zdrojový text této práce pro $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, včetně obrázků, seznamu literatury a použitých balíčků
- složka **pages**, obsahující textové soubory se zdrojovými kódy WordPress stránek zobrazující funkce pluginu
- složka **parser**, obsahující Python skript k přeparsování přiložených dat původního pluginu do formátu využívaného novým pluginem