Univerzita Hradec Králové
Fakulta informatiky a managementu
Katedra informatiky a kvantitativních metod

Aplikace pro podporu lezeckého sportu

Bakalářská práce

Autor: Lukáš Ondráček
Studijní obor: Aplikovaná informatika

Vedoucí práce: doc. RNDr. Petra Poulová, Ph.D.

Odborný konzultant: Mgr. David Chaloupský, Ph.D.
Katedra rekreologie a CR

Hradec Králové Duben 2019

Prohlášení:

Prohlašuji, že jsem bakalářskou práci zpracoval samostatně a s použitím uvedené literatury.

V Hradci Králové dne 26. 4. 2019 Lukáš Ondráček

Poděkování:

Děkuji vedoucí bakalářské práce doc. RNDr. Petře Poulové, Ph.D. za metodické vedení práce a odbornému konzultantovi Mgr. Davidu Chaloupskému, Ph.D. za poskytnuté rady a pomoc. Dále děkuji rodině a kolegům ze zaměstnání za podporu a vstřícnost.

Anotace

Bakalářská práce se zabývá tvorbou aplikace pro podporu a sdílení informací specifické oblasti cestovního ruchu věnující se skalnímu lezení na základě analýzy stávajících aplikací. Práce zahrnuje analýzu aplikací sloužících pro sdílení lezeckého sportu, analýzu a výběr implementačních nástrojů, návrh databáze a webové aplikace. Cílem aplikace je přehledně zobrazovat počty výjezdů a lezců, kteří vycestovali do konkrétní oblasti v určitém časovém rozmezí a umožnit snadné přenesení těchto čísel do dalších aplikací. Jedná se o webovou aplikaci využívající PHP framework Laravel a MySQL databázi.

Annotation

Title: The Application for Support for Climbing Sport

Bachelor Thesis is focused on creating application for supporting and sharing information of specific part of tourism focused on rock climbing based on the analysis of existing applications. The thesis contains analysis of applications for sharing climbing sport, analysis and selection of implementation tools, suggestion of database and web application. The goal of this application is to clearly display the number of trips and climbers who have traveled to a specific area within a certain time frame and allow easy transfer of these numbers to other applications. It is a web application using PHP framework Laravel and MySQL database.

Obsah

1 Úvod ... 1

2 Cíl práce ... 2

3 Dostupné technologie ... 3

3.1 Klient-server ... 3

3.2 World Wide Web ... 3

3.3 HTTP ... 4

3.4 Kódování ... 4

3.5 Dynamické webové stránky ... 4

3.5.1 HTML ... 5

3.5.2 Kaskádové styly (CSS) ... 6

3.5.3 JavaScript ... 6

3.6 Model relační databáze ... 7

3.6.1 SQL ... 7

3.7 Skriptovací jazyky ... 8

3.7.1 PHP ... 8

3.7.2 Python ... 11

3.8 Frameworky ... 12

3.8.1 Laravel ... 12

3.8.2 Symfony ... 14

3.8.3 Nette ... 14

4 Získání dat ... 15

4.1 Horolezecké portály ... 15

4.1.1 Lezec ... 15

4.1.2 Skalní oblasti ČR ... 16

4.1.3 8a.nu ... 16

4.1.4 The Crag 17
4.1.5 Climb.sk 17
4.2 Stažení dat 18
4.2.1 Získání identifikátorů uživatelů (UID) 18
4.2.2 Získání uživatelů 18
4.2.3 Získání cest 19
4.2.4 Získání výlezů 19
4.2.5 Získání výjezdů 19
4.2.6 Získání států, oblastí a regionů 20
4.2.7 Shrnutí 21
5 Návrh aplikace 22
5.1 Podobné webové aplikace 24
5.1.1 Vyletnik.cz – Tipy na výlety 25
5.1.2 Geologické lokality 25
5.1.3 Booking.com 25
5.1.4 Invia.cz 25
5.1.5 Eurostat 26
6 Implementace 27
6.1 Tvorba databáze 27
6.2 Tvorba aplikace 29
6.2.1 Modely 29
6.2.2 Pohledy 29
6.2.3 Kontroléry 30
6.2.4 JavaScript 30
6.2.5 Zobrazení 31
6.3 Umístění aplikace na Internet 32

6.3.1 Chráněný přístup 33
6.4 Chyby v aplikaci 33
7 Shrnutí výsledků 35
8 Závěry a doporučení 36
9 Seznam použité literatury 37
10 Přílohy 38

Seznam obrázků

Obrázek 1 Ukázka využití dat – Evropa....22
Obrázek 2 Ukázka využití dat – Německo....23
Obrázek 3 Diagram případů užití....24
Obrázek 4 Návrh databáze....28
Obrázek 5 Výsledná aplikace....31

1 Úvod

Jako téma bakalářské práce jsem si zvolil vytvořit aplikaci pro podporu lezeckého sportu, protože jsem chtěl, aby práce byla nějakým způsobem prospěšná. Měla by sloužit především pro přednášející a studenty v oboru cestovního ruchu Fakulty informačních technologií Univerzity Hradec Králové. Především ji bude využívat pan doktor David Chaloupský pro účely odborných prací a konferencí, kterých se pravidelně účastní.

Ve 2. kapitole se snažím vystihnout cíl bakalářské práce. V kapitole 3 jsou popsány dostupné technologie pro tvorbu webových aplikací. Postup, jakým způsobem jsem získal potřebná data je uvedený v kapitole 4. V 5. kapitole se nachází návrh aplikace a dále jsou zde uvedeny podobné webové stránky, které posloužily jako zdroj inspirace. Kapitola 6 popisuje postup při vytváření aplikace. V závěrečných dvou kapitolách se nachází shrnutí a závěr.

2 Cíl práce

Cílem bakalářské práce je vytvořit aplikaci pro podporu lezeckého sportu, která by měla usnadnit práci pracovníkům katedry cestovního ruchu při psaní odborných prací s tématikou aktivního cestování za účelem přírodního skalního lezení. Jedná se o webovou aplikaci s vlastní databází založenou na datech z portálu Lezec, kde horolezci vyplňují záznamy o svých zdolaných skalních cestách. Bohužel na webu Lezec se nezobrazují absolutní čísla, ale každý uživatel má vlastní deníček, kde se zobrazují jeho cesty. Proto je získání potřebných dat pro pracovníky katedry cestovního ruchu časově velmi náročné, jelikož musí postupně projít všechny záznamy z deníčků a data přepisovat do tabulkového editoru. Navíc cílem jejich výzkumu typicky nejsou jednotlivé zdolané skalní cesty, ale výjezdy do horolezeckých oblastí a států. Vytvořená aplikace má zobrazovat právě tato čísla.

3 Dostupné technologie

I když webové stránky existují řadu let, způsob jejich tvorby se stále mění, protože nástroje pro jejich produkci se rychle vyvíjejí a přibývají nové, což umožňuje vytvářet stále propracovanější aplikace.

3.1 Klient-server

Systém klient-server se fyzicky skládá minimálně ze tří částí: klient, který je reprezentován počítačem, poslední dobou stále častěji mobilním zařízením, jako jsou smartphone či tablet. Druhou částí je server, v tomto případě databázový. Jedná se o počítač s dostatečně rychlým procesorem, dostatkem operační paměti a úložištěm pro uložení dat. Nezbytnou součástí systému je i síť, která dvě předchozí části propojuje. Může se jednat o lokální síť (LAN) nebo větší síť, která sdružuje více lokálních sítí dohromady (WAN) nebo Internet, bez kterého by se webové aplikace neobešly. (Murach, 2015)

Klient a server potřebují odpovídající software, v případě klienta to je aplikační software, může se jednat o zakoupený software, např. program pro správu účetnictví, nebo software vytvořený na zakázku pro specifické použití. Server používá systém pro správu databáze (DBMS = Database Management System), např. MySQL a Microsoft SQL Server. Ačkoli aplikační software běží na straně klienta, data, se kterými pracuje, jsou uložená na serveru. K vyřešení komunikace mezi klientem a serverem je potřeba mít rozhraní pro programování aplikace (API = Application Programming Interface). Klient komunikuje se serverem pomocí SQL příkazů (queries), které přes API posílá ke zpracování DBMS. Po vykonání příkazu zašle DBMS odpověď klientovi. (Murach, 2015)

Klient je nainstalovaný lokálně na stroji, ze kterého se přistupuje k serveru. Server se může nacházet kdekoli, může běžet lokálně na stejném stroji jako klient nebo na druhé straně planety, a to díky rozsáhlé síti. Pro připojení k MySQL serveru je potřeba znát uživatelské jméno, název hostitele a heslo. (Dubois, 2014)

3.2 World Wide Web

Na počátku 90. let prováděli vědci v laboratořích CERNu rozsáhlé experimenty, které generovaly obrovské množství dat. Tato data potřebovali sdílet se svými

kolegy po celém světě. Internet již existoval a bylo k němu připojeno několik set tisíc počítačů, proto Tim Berners-Lee navrhl metodu navigace mezi nimi pomocí hyperlinkového rámce, který se stal známým jako HTTP. Vytvořil také značkovací jazyk HTML a napsal první webový prohlížeč a webový server. Dnes tyto nástroje považujeme za samozřejmost, ale tehdy byl jeho koncept revoluční. (Nixon, 2014)

3.3 HTTP

HTTP (Hypertext Transfer Protocol) je komunikační standard, který řídí požadavky a odpovědi mezi webovým prohlížečem počítače koncového uživatele a webovým serverem. Úkolem serveru je přijmout požadavek od klienta a pokusit se mu odpovědět smysluplným způsobem, často je to předání webové stránky. Klient se serverem komunikuje nejčastěji pomocí Internetu a server obvykle řeší více připojení s klienty. Pokud s žádným klientem nekomunikuje, tráví čas nasloucháním pro příchozí spojení, když žádost dorazí, server odešle odpověď a potvrdí příjem. (Nixon, 2014)

3.4 Kódování

Kódování stanovuje, které znaky je možné v souboru použít. Pro jistotu správné interpretace češtiny a dalších jazyků, které používají oproti anglické abecedě další znaky, je vhodné použít kódování UTF-8 (Unicode Transformation 8-bit), které je zpětně kompatibilní s formátem ASCII a aktuálně umožňuje napsat okolo sta tisíc znaků. (Ullmann, 2012)

3.5 Dynamické webové stránky

Dnešní uživatelé webu očekávají stránky, které jsou často upravovány a poskytují jim zážitek. Na takové stránky se poté uživatelé často vrací, proto správci stránek chtějí, aby se daly snadno aktualizovat a udržovat. Z těchto důvodů jsou v současnosti standardem pro dynamické webové stránky PHP a MySQL. Dynamické webové stránky jsou v podstatě aplikace, protože reagují na různé parametry, např. verze webového prohlížeče návštěvníka stránek, mají „paměť“, která umožňuje registraci a přihlášení uživatelů, e-commerce atd. Téměř vždy je pro návštěvníky umožněné vyhledávání. Často správci mají rozhraní, kde mohou

upravovat obsah webu. Snadněji se u těchto stránek provádí údržba a aktualizace než u statických stránek. (Ullmann, 2012)

Pro vytvoření dynamických webových stránek existuje mnoho nástrojů, nejběžnějšími jsou ASP.NET (Active Server Pages) od Microsoftu, JSP (Java Server Pages), ColdFusion, Ruby on Rails a PHP. Dynamické webové stránky nemusí nutně využívat databází, většina se jich ale bez databáze neobejde. Nejčastější aplikací pro správu databáze je MySQL. (Ullmann, 2012)

3.5.1 HTML

Jazyk HTML (Hypertext Markup Language) je základem všech webů. Kód HTML je zapsán v textových dokumentech a skládá se z prvků (elementů). Prvky jsou obvykle ohraničeny dvěma značkami (tagy), které jsou zapsány uvnitř hranatých závorek (znaky větší než a menší než), zavírající značka obsahuje navíc oproti otevírající značce lomítko, např. <h1>Nadpis</h1>. Každý prvek HTML informuje prohlížeč o informaci, která se nachází mezi jeho otevírajícími a zavírajícími značkami. Více informací o obsahu prvku poskytují atributy, např. Odstavec v českém jazyce. . (Duckett, 2011)

Veškerý HTML kód se nachází uvnitř tagu html, dále je soubor rozdělen do tagů head a body. Část head obsahuje informace o stránce, např. metainformace (použité kódování, jazyk…), odkazy na potřebné dokumenty (CSS, favicon…), titulek (zobrazuje se v záhlaví prohlížeče) atd. Vše zapsané uvnitř části body se zobrazí uvnitř hlavního okna prohlížeče. Kromě HTML velké množství webů používá i další technologie jako jsou PHP, MySQL, JavaScript a CSS. (Duckett, 2011)

3.5.1.1 HTML5

HTML5 je nový standard HTML, který je vyvíjen od roku 2004, kdy první návrh vypracovali vývojáři dvou oblíbených prohlížečů Mozilla Foundation a Opera Software. Konečný návrh byl předložen konsorciu W3C (World Wide Web Consortium) na začátku roku 2013. Není to ale konečné řešení a pracuje se na verzi 5.1, která by měla přinést další vylepšení. HTML5 oproti HTML přináší mnoho nových věcí, a téměř žádné nebyly odstraněny nebo změněny. Nová je např. možnost práce s vektorovou grafikou ve webovém prohlížeči bez použití zásuvných

modulů (Flash) nebo vkládání videí a zvuků do webových stránek. Dále přináší nové prvky (nav, footer) a dvě nové technologie vykreslování: MathML pro zobrazování matematických vzorců a SVG pro vytváření grafických prvků mimo nový element canvas. (Nixon, 2014)

3.5.2 Kaskádové styly (CSS)

Kaskádové styly slouží k formátování webových stránek, pomocí nich lze rychle a snadno přeformátovat všechny prvky HTML. Řešením, jak nastavit styl konkrétního jednoho prvku je použití ID. Pro specifikaci ID v CSS souboru slouží znak „#“. Pokud je potřeba aplikovat stejný styl na více prvků, tak toho lze docílit použitím třídy. Definování v CSS souboru je stejné jako u ID, ale místo znaku „#“ se používá tečka. Vlastnosti prvku jsou oddělené středníkem. Pokud dojde k zadání stejné vlastnosti jednomu prvku vícekrát, použije se poslední zadaná hodnota. Pro vložení komentáře slouží párová značka „/*“ pro začátek komentáře a „*/“ pro jeho konec. Je vhodné kaskádové styly uložit do samostatných souborů a do HTML souborů je pouze importovat. (Nixon, 2014)

3.5.3 JavaScript

JavaScript je jazyk, který slouží k vytváření dynamických webových stránek, běží uvnitř prohlížeče a má přístup ke všem prvkům dokumentu. Poprvé se objevil v roce 1995 v prohlížeči Netscape Navigator. S programovacím jazykem Java nemá mnoho společného. Jeho pojmenování je pouze marketingová snaha těžit z popularity Javy u tohoto skriptovacího jazyka. JavaScript, stejně jako PHP, vychází ze syntaxe programovacího jazyka C. (Nixon, 2014)

JavaScript poskytuje prostředek pro dynamickou interakci uživatele, např. kontrolu platnosti e-mailové adresy ve vstupních formulářích a zobrazování výzev. Mimo to je stále častěji používán pro Ajax (Asynchronous JavaScript and XML), což je termín pro přístup k webovému serveru na pozadí. Ajax je hlavním pilířem Webu 2.0, kde se webové stránky začínají podobat samostatným programům, protože nemusí být zcela načteny a místo toho může rychlé volání v Ajaxu aktualizovat jeden prvek na stránce, např. změna fotografie na sociální síti.

JavaScript lze používat i ve formě frameworku, který ušetří čas vývojáři při práci s webovými stránkami, nejznámějším je jQuery. (Nixon, 2014)

3.6 Model relační databáze

V roce 1970 vyvinul Edgar Frank Codd model pro nový typ databáze, nazvaný relační databáze. Relační model snižuje redundanci dat, což snižuje potřebné místo k uložení dat a zvyšuje efektivitu získávání dat. Data jsou uložená v jedné nebo více tabulkách, každá tabulka je dvourozměrná matice složená z řádků (záznamů) a sloupců (polí). Relační databáze jsou dnes standardem databázových aplikací. (Murach, 2015)

Sloupce reprezentují entity objektů reálného světa, např. jméno, adresa, telefon, datum narození atd. a každý řádek reprezentuje instanci tohoto objektu. Průsečík sloupce a řádku se nazývá hodnota (buňka). Jestliže tabulka obsahuje sloupec nebo sloupce, které jedinečně identifikují každý řádek, potom tento sloupec nebo tyto sloupce se označují jako primární klíč tabulky. Kromě primárního klíče lze definovat i neprimární (unikátní) klíč, který rovněž nabývá pouze unikátních hodnot. (Murach, 2015)

Efektivní přístup k hodnotám v řádcích tabulky, založený na hodnotách v jednom nebo více sloupcích, umožňuje index. Protože aplikace typicky přistupují k řádkům v tabulce odkazováním na jejich klíče, jsou automaticky indexované všechny klíče v tabulce, navíc je možné vytvořit index pro kterýkoliv sloupec. Stejně jako klíč, i index může být složený z více sloupců. (Murach, 2015)

3.6.1 SQL

SQL (Structured Query Language) je nejběžnější jazyk pro přístup k systémům správy relačních databází, používá se pro uložení i získávání dat z databází MySQL, Oracle, PostgreSQL, Sybase, Microsoft SQL Server atd. Pro definování databází se používá jazyk DDL (Data Definition Language) a pro dotazování databází slouží jazyk DML (Data Manipulation Language), SQL pokrývá oba dva jazyky. Standard pro SQL je stanoven organizací ANSI (American National Standards Institute). (Welling a Thomson, 2009)

3.6.1.1 MySQL

MySQL vzniklo v roce 2000 a dnes je nejoblíbenějším open-source systémem řízení databází. Jeho škála použití je veliká, od nadšených amatérů až po velké společnosti jako jsou Wikipedie, Facebook nebo Twitter. Komunikace s MySQL databází probíhá pomocí jazyka SQL, ve většině případů se jedná o komunikaci klient-server. (Murach, 2015)

MySQL je rychlé, databáze se snadně vytvářejí, používají a spravují. Může běžet na mnoha verzích Unixu a Windowsu. Programy využívající MySQL mohou být napsány v mnoha jazycích, včetně C, C++, Eiffel, Go, Java, Perl, PHP, Python, Ruby atd. (Dubois, 2014)

MySQL využívá vlastní SQL jazyk, který se od standardu SQL definovaným ANSI v některých detailech liší, některé z rozdílů budou smazány v následujících verzích MySQL SQL a některé odlišnosti se zde nachází úmyslně. (Welling a Thomson, 2009)

3.7 Skriptovací jazyky

Skriptovací jazyky jsou ve většině případů interpretovanými jazyky, které jsou opakem ke kompilovaným jazykům, např. Java nebo C++. U kompilovaných jazyků programátor píše kód do textového souboru, ale potom spustí příkaz. Tyto textové soubory se změní na binární soubory, tj. pro člověka nečitelné kódy, které používá počítač. Na druhou stranu interpretované jazyky, např. PHP a JavaScript, nepotřebují spoustu nástrojů ani kroků, stačí jen napsat kód a hned vyzkoušet ve webovém prohlížeči. Kód skriptovacího souboru je zaznamenán ve formě skriptů, což jsou textové soubory. Tyto soubory jsou interpretovány řádek po řádku při každém přístupu k souboru. (McLaughlin, 2012)

3.7.1 PHP

Zkratka PHP původně znamenala Personal Home Page, jazyk PHP vymyslel Rasmus Lerdorf v roce 1994 a použil ho pro vytvoření svého on-line životopisu. Jazyk se postupně vyvíjel a stával se více schopným a užitečným. Začal se používat v komerčních projektech a došlo ke změně významu na rekurzivní PHP: Hypertext Preprocessor (McLaughlin, 2012).

PHP je široce používaný obecný skriptovací jazyk, který je zvláště vhodný pro vývoj webových stránek a může být vložen do HTML. PHP skript běží pouze po nějaké události, např. když uživatel odešle formulář nebo přejde na adresu URL (Uniform Resource Locator). Kód PHP je vykonávaný na straně serveru, kde se nachází aplikace jako Apache nebo IIS (Internet Information Services) od Microsoftu. Server odesílá webové stránky na žádost klienta (webový prohlížeč návštěvníka webu). Když se návštěvník dostane na web napsaný v PHP, server čte kód PHP a zpracovává ho. (Ullmann, 2012)

Výsledkem je kód HTML, který je předán webovému prohlížeči a ten ho následně považuje za standardní stránku HTML. To je rozdíl od statické stránky HTML, kde se při požadavku na server pouze odešle kód HTML do webového prohlížeče a neexistuje zde žádná interpretace na straně serveru. Jelikož není vyžadována žádná akce na straně serveru, ve webovém prohlížeči se spouští stránky HTML zcela bez použití serveru. Pro koncového uživatele a webový prohlížeč neexistuje žádný zřetelný rozdíl mezi tím, jak bude vypadat stránka home.html a home.php, ale způsob, jakým byl obsah této stránky vytvořen, bude výrazně odlišný. (Ullmann, 2012)

Jazyk PHP je multiplatformní, tudíž může běžet na operačních systémech Windows, Unix a Macintosh. PHP skripty napsané na jednom serveru budou fungovat na jiném s drobnými nebo žádnými změnami. PHP je oproti dalším nástrojům ke tvorbě dynamického webu vhodnější, rychlejší a snadnější na učení. Další velkou výhodou je fakt, že se jedná o open-source jazyk, tj. software s otevřeným zdrojovým kódem. Z toho vyplývá, že je k dispozici zdarma a není potřeba platit za licenci. Důkazem, že jeho aplikace je velmi vhodná pro webové stránky, je jeho uplatnění ve více než 76 % všech webů a ze všech programovacích jazyků je pátým nejoblíbenějším. PHP využívají i giganti Internetu jako jsou Yahoo!, Wikipedie a Facebook. (Ullmann, 2012)

3.7.1.1 Syntaxe PHP

Kód PHP se vkládá přímo do souboru s příponou .php obsahující kód HTML, začátek bloku kódu PHP se značí „<?php“ a ukončovacími znaky jsou „?>“. Všechno napsané mezi tyto tagy je považováno za PHP a tento kód bude zpracován PHP interpretem.

Zbývající kód je odeslán prohlížeči jako standardní HTML. HTML a PHP jsou tedy vzájemně provázány ve stejném dokumentu nebo mohou být rozděleny do samostatných souborů s příponami .html a .php. (Ullmann, 2012)

Stejně jako většina jazyků založených na jazyce C, i PHP umožňuje dva typy komentářů. Jedním z nich je jednořádkový komentář, to znamená, že všechny znaky následující za klíčovým znakem (nebo kombinací znaků) jsou ignorovány interpretem. Toho lze využít v situaci, kdy je potřeba napsat krátký komentář přímo za PHP kód do jednoho řádku. Pro začátek řádkového komentáře slouží znak „#“, stejnou službu provede i dvojice lomítek „//“. (Ullmann, 2012)

Druhým typem komentáře je víceřádkový. Začíná kombinací znaků „/*“ a končí obrácenou kombinací stejných znaků „*/“. Mimo psaní dlouhých komentářů se tento typ používá i k zakomentování kódu. Pokud je potřeba, aby některá část kódu nebyla dočasně vykonávána, není nutné ji smazat ze souboru, ale stačí umístit výše uvedené znaky před a za požadovanou část kódu. (Ullmann, 2012)

PHP má mnoho vestavěných funkcí. K nejčastěji používaným patří echo a print, obě dvě slouží k výpisu (text se zobrazí jako by byl napsaný v HTML), např. echo 'Hello, world!'; nebo print 'Hello, world!';. Místo apostrofů lze použít stejným způsobem uvozovky. Pokud je potřeba použít apostrof uvnitř apostrofů (nebo uvozovky uvnitř uvozovek), je nutné tento znak escapovat zpětným lomítkem, např. echo 'I\'m happy.';. Každý příkaz musí být ukončen středníkem. Bílé znaky (mezery, tabulátory a nové řádky) jsou PHP interpretem ignorovány a slouží pouze pro přehlednost kódu. Stejně tak i pokud bude napsán např. obsah apostrofů u funkce echo na více řádků, nebude toto rozdělení na výsledné stránce viditelné. K odřádkování slouží HTML značka <br>. (Ullmann, 2012)

3.7.1.2 Proměnné

K dočasnému uložení hodnoty slouží proměnná. V PHP její název začíná značkou dolaru „$“ a dále obsahuje kombinaci písmen, číslovek a podtržítko. Po znaku dolaru nemůže přímo následovat číslovka. PHP podporuje osm typů proměnných: 4 skalární (jednohodnotové) – Boolean (pravda nebo nepravda), integer (celé číslo), floating point (reálné číslo s plovoucí řádovou čárkou) a string (řetězec znaků); 2 neskalární (vícehodnotové) – pole a objekt; 2 speciální – resource (pro práci s ex-

terními zdroji jako je třeba databáze) a NULL (nemá žádnou hodnotu). (Ullmann, 2012)

PHP je částečně case-sensitivní. Proměnné, konstanty a klíče polí jsou case-sensitivní, z toho vyplývá, že $var, $Var a $VAR jsou tři různé proměnné, zatímco funkce, klíčová slova a konstruktory jsou case-insensitivní, takže echo, Echo a ECHO je stále ta samá funkce. PHP je neformální jazyk, takže proměnné se nemusí deklarovat (vybrat typ) ani inicializovat (přiřadit hodnotu ihned po vytvoření) a je možné bez problému převádět proměnnou mezi typy. Přiřazení hodnoty do proměnné se nastavuje pomocí znaku „=“. Spojování řetězců znaků se provádí pomocí operátoru tečka, např. $user = $first_name . ' ' . $last_name;. (Ullmann, 2012)

Číselné proměnné se musí zadávat bez uvozovek a apostrofů, např. $pi = 3.14;, v opačném případě by se jednalo o řetězec znaků. Pro operace s nimi se používají aritmetické operátory „+“ (sčítání), „-“ (odčítání), „*“ (násobení), „/“ (dělení), „%“ (zbytek po celočíselné dělení), „++“ (inkrementace = zvýšení čísla o jedničku) a „--“ (dekrementace = snížení čísla o jedničku). (Ullmann, 2012)

3.7.1.3 Konstanty

Konstanty jsou stejně jako proměnné používány k dočasnému uložení hodnoty, ale liší se v mnoha směrech. Pro vytvoření konstanty se používá funkce define() na rozdíl od operátoru přiřazení (=) u proměnných. U konstant se nepoužívá jako počáteční znak symbol dolaru. Hodnotu konstanty nelze změnit. Nepsaným pravidlem je použití velkých písmen pro jméno konstanty, např. define ('PI', 3.14);. PHP běží s několika předdefinovanými konstantami, jako jsou PHP_VERSION (verze běžícího PHP), PHP_OS (operační systém serveru) atd. (Ullmann, 2012)

3.7.2 Python

Python je univerzální programovací jazyk, který se často používá při vytváření skriptů. Jedná se o objektově orientovaný jazyk. Není plně kompilovaný, což znamená, že zdrojový kód není přeložený až do binárního kódu. Je ale přeložen do bajtového kódu, který se nachází mezi zdrojovým kódem a binárním kódem u plně kompilovaných jazyků, jako je např. jazyk C. Tím je zajištěna přenositelnost,

protože bajt kód není závislý na platformě, ale některé programy mohou běžet pomaleji, než by běžely v plně kompilovaném jazyce. (Lutz, 2009)

Python vznikl v roce 1991 a je široce používán, proto je stabilní a robustní. Používá ho více než milion programátorů a je uplatňován i ve významných projektech firem Google, Intel, Cisco, Hewlett-Packard, NASA a další. Jeho obecná povaha umožňuje použití pro téměř všechny oblasti. Skoro každá podstatná organizace, která vytváří software, používá Python, ať pro krátkodobé úkoly, jako je testování, nebo pro dlouhodobý vývoj produktů. Python je vhodný pro obě dvě domény. (Lutz, 2009)

3.8 Frameworky

Framework slouží k vyřešení typických problémů dané oblasti, a tím umožní vývojářům soustředit se na řešení vlastního projektu. Framework za vývojáře vyřeší např. směrování HTTP požadavků, využije se kód od člověka, který je v tomto oboru specialistou a vývojář nemusí tuto oblast studovat a ve výsledku vyřešit hůře (např. z hlediska bezpečnosti) než specialista. Framework obsahuje sbírku komponent třetích stran spolu s propojením těchto komponent, jako jsou konfigurační soubory, poskytovatelé služeb, předepsané adresářové struktury a bootstrapy. Výsledkem je konzistence a flexibilita. Frameworky dále poskytují konvence, které snižují množství kódu, který musí nový vývojář pochopit u již existujícího projektu. Příkladem PHP frameworků jsou Ruby on Rails, CakePHP, Symfony, Laravel, Silex, Lumen, Slim, Nette a mnoho dalších. (Stauffer, 2016)

3.8.1 Laravel

První beta verze Laravelu byla vydána v roce 2011 Taylorem Otwellem a byla složena ze zcela vlastního kódu. Pár měsíců poté byly vydány verze Laravel 2 a 3. V roce 2013 byl celý framework přepsán od základů a vydán Laravel 4, který je založený na balících Symfony a Illuminate získávané přes Composer. Laravel 4 dále nově obsahoval fronty, fasády, komponentu pro elektronickou poštu a nástroj pro automatické vyplňování databáze vzorovými daty. (Stauffer, 2016)

Laravel 5 byl představen v roce 2015 a přinesl upravenou strukturu adresářů, nástroj pro autentizaci sociálních sítí, plánovač úloh a zcela nový příkazový

řádek. Hlavní filozofií Laravelu je přinést vývojáři zrychlení a usnadnění práce při vytváření webu. Zbavit vývojáře zbytečných překážek by mělo vést k vytváření lepších produktů s jasným, jednoduchým a dobře čitelným zdrojovým kódem. Laravel je úspěšný open-source, který staví základy na dobré dokumentaci a na bohaté a živé komunitě plné lidí, kteří se zajímají o Laravel od prvních dní jeho existence. (Stauffer, 2016)

Pro instalaci Laravelu je vhodné nejprve nainstalovat balík s vývojovým prostředím Laravel Homestead, což je možné provést dvěma obvyklými způsoby. Starší řešení využívá Composer a novější klonování z repozitáře Git. Samotný Laravel je možné nainstalovat s pomocí instalátoru nebo s pomocí Composeru příkazem create-project, výsledek je stejný. (Blankenship, 2016)

Laravel využívá návrhový vzor MVC (Model View Controller) a v kořenovém adresáři se nachází devět složek: app, bootstrap, config, database, public, resources, storage, tests a vendor. V adresáři app se nachází logika celé aplikace, jsou zde kontroléry, služby, filtry, příkazy a mnoho dalších vlastních tříd. V adresáři bootstrap se kromě samotné knihovny nachází složka cache s uloženou mezipamětí. Nastavení databáze, e-mailu atd. se nachází v adresáři config. Složka database obsahuje databázové migrace a nástroje pro generování dat do databáze „seeder“. Obrázky, CSS a JavaScript a jiné veřejné soubory jsou ve složce public. Pohledy se vkládají do resources. Složku storage Laravel používá pro uložení relací, mezipamětí, šablon, logů atd. Jak název napovídá, v adresáři test se nachází testovací soubory, např. PHPUnit. Závislosti composeru (třídy Symfony, PHPUnit atd.) jsou umístěny ve složce vendor. (Wu, 2016)

Jedním z nejdůležitějších souborů Laravelu je routes.php nacházející se v adresáři /app/Http/, který slouží pro směrování. Jinými slovy je zde definováno, které akce se mají vykonat na konkrétní URL, např. pro URL „/“ se vrátí pohled home, což je typicky soubor home.blade.php uložený v /resources/views a obsahuje kód HTML. Pro řízení slouží kontroléry, které lze generovat automaticky pomocí Artisanu, což je nástroj Laravelu v příkazovém řádku. Stejným způsobem lze vytvářet modely, pohledy, migrace atd. (Wu, 2016)

3.8.2 Symfony

Framework Symfony byl zveřejněn už v roce 2005. Jedná se o spolehlivou a vyspělou platformu. Je jediným známým frameworkem, který se řídí PHP a webovými standardy. Hodí se pro rozsáhlé projekty a pro jeho použití jsou vyžadováni spíše zkušenější programátoři. Používá ho např. redakční systém Drupal. (Njenga, 2018)

3.8.3 Nette

Hlavním autorem Nette Frameworku je David Grudl, který je považován za jednoho z největších českých odborníků na webové aplikace. Nette slouží pro vytváření malých i velkých projektů, podporuje účelný návrh a klade důraz na budoucí rozšiřitelnost. Řadu let je nejoblíbenějším PHP frameworkem v České republice. (Grudl, 2019)

4 Získání dat

Před začátkem vývoje samotné aplikace bylo nutné nejprve zajistit potřebná data, bez kterých by aplikace nemohla fungovat. Získaná data poté usnadnila návrh aplikace a zamezila kritickým chybám v plánování aplikace.

4.1 Horolezecké portály

Po konzultacích bylo rozhodnuto, že budou použita data z portálu Lezec, protože prioritou aplikace je sledování českých horolezců a na tomto portálu se jich nachází nejvíce. Existují ale i jiné, především zahraniční, portály. Jejich rozbor se nachází v následujících podkapitolách. Při návrhu aplikace je nutné myslet na rozšiřitelnost databáze o údaje z těchto dalších portálů.

4.1.1 Lezec

Lezec je komplexní portál s obsáhlým seznamem cest, skal, boulderů, hor a závodů z celého světa existující od roku 2000. Mimo to se zde nachází i články o aktualitách a závodech ve skalním lezení, fotografie, diskuze a deníčky zejména českých lezců. Webová prezentace je umístěná na adrese http://www.lezec.cz (v případě anglické mutace to je http://www.czechclimbing.com) a je zde registrováno více než 4,5 tisíce, hlavně českých, uživatelů, včetně jednoho z nejlepších světových lezců Adama Ondry. Dále se zde nachází přes 735 tisíc cest.

Po stránce designu vypadají tyto webové stránky zastarale, ale jsou stále aktualizovány ze strany správců i registrovaných uživatelů a svůj účel plní vcelku dobře. Stránky se jeví nepřehledné, protože se zde nachází celkem čtyři navigační nabídky, jedna slouží jako filtr pro výběr kategorie – skály, bouldery, hory a závody. Pod ní najdeme hlavní nabídku s odkazy na úvodní stránku, e-shop, diskuzi, inzerci, ankety, zajímavé odkazy, průvodce skalními oblastmi, mapy, fotogalerii a videa. V levé části je umístěno další menu, kde mimo jiné je odkaz na deníčky lezců. Na druhé straně se vyskytuje poslední nabídka s databází cest a novinkami v různých sekcích.

Vyhledání starších příspěvků v diskuzi, fotografií atd. není snadné, protože nejsou žádným způsobem tříděné, ale pouze stránkované. V diskuzi je více než

dva tisíce stran příspěvků, seřadit je nelze, pouze jdou vyfiltrovat podle kategorie, poté je ale jejich počet nadále poměrně velký.

Podobně je to i s deníčky lezců, zobrazí se zde stránkovaný seznam registrovaných uživatelů, který je možné seřadit pouze abecedně podle příjmení. Není možné řadit např. podle počtu výlezů. Po zvolení jednoho z lezců se otevře stránka s jeho základními osobními údaji a statistikami na Lezci (počet zadaných cest, datum prvního a posledního záznamu atd.). Po přepnutí na kartu cesty se zobrazí lezcovy výlezy, které je možné filtrovat podle data, typu a stylu. Po kliknutí na cestu se zobrazí popis cesty a všechny výlezy, které byly na dané cestě registrovanými uživateli provedeny.

4.1.2 Skalní oblasti ČR

Dalším českým portálem o lezení s deníčky horolezců jsou Skalní oblasti ČR, které spravuje Dr. Zdeněk Strnad. Tento web existuje od roku 2004, a oproti Lezci po grafické stránce vypadá moderněji. Má i svou alternativu pro zobrazení na mobilním telefonu, a dokonce aplikaci na zařízení s operačními systémy Android a iOS. Jeho URL adresa je http://www.skalnioblasti.cz. Hned na úvodní stránce se nachází dnešní a zítřejší předpověď počasí, včetně zobrazení vlhkosti jednotlivých skal. Na další mapě jsou vyznačené skály, na kterých je aktuálně zakázané lezení, např. z důvodu hnízdění ptactva.

Tento server obsahuje mnoho zajímavých informací, z pohledu této práce jsou zásadní horolezecké deníky. Je jich přes 500, dále je zde přes 94 tisíc evidovaných cest a 15 tisíc fotografií. Celkem je v denících přes 18 tisíc záznamů, z toho ale je pouze přibližně třetina veřejná. Zbytek jsou privátní záznamy od lezců, kteří se nechtějí svými výkony chlubit a píší si je pouze pro vlastní účely. Z tohoto důvodu není použití dat z tohoto portálu do aplikace moc vhodné, navíc je docela pravděpodobné, že horolezci s veřejným deníkem budou mít svůj deníček i na Lezci.

4.1.3 8a.nu

Obdobou Lezce je mezinárodní webový portál 8a.nu, založený v roce 1999 Jensem Larssenem jako lokální stránka pro lezce z okolí švédského Göteborgu. Postupem času se rozšiřoval o další oblasti a přibývali lezci z celého světa. Aktuálně sídlí

v italském Brixenu, je celý v angličtině a některé části je možné zobrazovat v dalších světových jazycích.

Sekce na tomto webu jsou stejné jako na Lezci a obě dvě prezentace jsou si i vizuálně podobné. V úvahu přichází rozšíření aplikace o data ze serveru 8a.nu, pouze je nutné vyřadit české lezce, kteří jsou registrováni rovněž na portálu Lezec, aby nedošlo k duplikaci dat. Je zde registrovaných více než 71 tisíc uživatelů a zadaných více než 5 milionů cest.

Web se nachází na URL adrese https://www.8a.nu. Zajímavostí je, že doména .nu je národní doménou nejvyššího řádu ostrovu Niue, který leží poblíž Nového Zélandu. Často se ale používá pro komerční projekty z celého světa díky podobné výslovnosti jako je anglický výraz pro nový – new. (OVH Registrar, 2015)

4.1.4 The Crag

Dalším zajímavým portálem o lezení je australský The Crag s URL adresou https://www.thecrag.com. Zde se nachází skalní cesty a horolezci se svými výlezy ze všech kontinentů. Kromě toho jsou zde i fotogalerie, fórum a mapa s umístěním všech cest a oblastí. Databáze je oproti 8a.nu méně rozsáhlá, obsahuje 1,2 milionů cest, avšak okolo 100 tisíc registrovaných lezců z celého světa, největší část jich je ale z Austrálie. Uživatelé z jiných států než Austrálie, jsou ve velké míře registrovaní i na serveru 8a.nu, což by se muselo zohlednit v případě rozšiřování databáze aplikace z tohoto zdroje.

4.1.5 Climb.sk

Největším slovenským lezeckým portálem je Climb.sk, kde stejně jako na Lezci se nachází deníčky lezců s jejich výstupy. Je zde zaregistrovaných přibližně 250 uživatelů ze Slovenska a 42 tisíc cest z celého světa. Rozšířit aplikaci o data z tohoto webu by bylo možné, pouze by opět bylo nutné vyřadit uživatele registrované i na Lezci.

Jedná se o přehledný web s moderním responzivním vzhledem. Kromě deníčků jsou zde i články o lezení, oblasti, sektory, cesty, diskuze a závody. Výlezy lze filtrovat podle kategorie a roku absolvování a seřadit podle data vložení, data

přelezu nebo obtížnosti. Je možné zobrazit výlezy konkrétního uživatele nebo všech najednou. URL adresa tohoto portálu je https://www.climb.sk.

4.2 Stažení dat

Jak již bylo nastíněno v úvodu, pro účely vytvořené aplikace je stěžejní databáze deníčků (absolvovaných cest) lezců. Prvním pokusem bylo získat data od administrátorů Lezce ve formě exportu databáze nebo propojení jejich databáze a databáze zmiňované aplikace přes API rozhraní, které by zajistilo aktuálnost dat ve výsledné aplikaci. Bohužel tato snaha byla zamítnuta, ale alespoň se podařilo získat povolení ke stažení veřejně dostupných dat v souladu s obecným nařízením o ochraně osobních údajů (GDPR). To znamenalo ukládat uživatele bez jména, příjmení a dalších soukromých údajů, přičemž tyto údaje by stejně neměly žádné opodstatnění vzhledem k tématu bakalářské práce.

Z důvodu vysokého počtu záznamů na portálu Lezec nebylo možné data získat manuálně, ale bylo nutné práci zautomatizovat pomocí skriptů. K jejich vytvoření byl použit oblíbený skriptovací jazyk Python. Pro stažení dat byly napsány celkem čtyři skripty a jeden pro jejich úpravu.

4.2.1 Získání identifikátorů uživatelů (UID)

První z nich získává identifikátory uživatelů (UID), v případě Lezce se jedná o hexadecimální číslo (znaky 0–9 a a–f) zakončené znakem „h“, např. 204c614461h. Skript postupně projde všechny stránky s odkazy na deníčky jednotlivých lezců s URL adresou ve tvaru http://www.lezec.cz/deni-ky.php?limn=X, kde X je číslo od 1 inkrementováno o 100 postupně do aktuálně 15001. Na každé této stránce se pomocí regulárního výrazu získají UID z hypertextových odkazů jednotlivých lezců a uloží se do výstupního souboru.

4.2.2 Získání uživatelů

Další skript slouží pro získání uživatelů, konkrétně tedy pouze přiřadí datum založení deníčku k UID uživatele. Mohl by s drobnými úpravami stahovat i jméno, příjmení a login uživatele, tyto údaje ale nejsou v aplikaci podstatné, a navíc by mohlo dojít k právnímu střetu kvůli porušování zákonů GDPR. Skript postupně

projde stránky s URL adresou http://www.lezec.cz/denik.php?uid=X, kde X představuje UID ze seznamu získaného prvním skriptem, a opět pomocí regulárních výrazů získá datum, následně ho spojí s UID a předá do výstupního souboru.

4.2.3 Získání cest

Třetí skript stáhne z Lezce všechny zadané cesty z URL adresy http://www.lezec.cz/cesty.php?lim=X&cchr=Y&cobl=h&csek=h&cpol=0, kde X je číslo od 1 zvyšované postupně o 100 a Y je ID kategorie cesty nabývající hodnotu od 1 do 4 – skály, bouldry, hory a industrial. Skript s pomocí regulárních výrazů načte jméno cesty, klasifikaci, sektor, oblast, region, počet výstupů, počet fotografií, počet komentářů a hodnocení, doplní ID kategorie a uloží je do výstupního souboru.

4.2.4 Získání výlezů

Pro získání výlezů slouží čtvrtý skript, který funguje podobným způsobem jako předchozí. Opět využívá regulárních výrazů k získání datumu výlezu, jména cesty, oblasti, klasifikace, bodů, stylu a komentáři z URL adresy http://www.lezec.cz/denik.php?uid=X&ckat=Y&par=1&crok=9997, kde X je UID ze seznamu získané prvním skriptem a Y je ID kategorie cesty již použité ve třetím skriptu. K získaným údajům se ještě doplní právě UID lezce, ID kategorie a celý záznam se zapíše do výstupního souboru.

4.2.5 Získání výjezdů

Poslední skript již nestahuje data z portálu Lezec, ale pouze pracuje s dříve získanými. Konkrétně se jedná o výlezy získané čtvrtým skriptem. Jde o to, že úkolem aplikace není zobrazovat výlezy, ale výjezdy. To znamená, že pokud lezec absolvuje např. pět výlezů během víkendu v jedné oblasti, jedná se pouze o jeden výjezd. Stanovit počet dnů mezi dvěma výlezy, kdy se ještě jedná o jeden výjezd a kdy už o dva je velmi komplikované.

Pokud např. jeden lezec bude mít každý den během jednoho týdne jeden výlez v německé oblasti poblíž českých hranic, je těžké posoudit, zda bydlí poblíž a na noc se vrací domů, tedy se jedná o sedm výjezdů, nebo zde tráví svoji dovolenou a jedná se tak pouze o jeden výjezd. Naopak pokud budou výlezy např. v australské

oblasti následovat po sobě po deseti dnech, bude se jednat stále o jeden výjezd vzhledem ke vzdálenosti od České republiky.

Vyřešení této problematiky převyšuje hranice bakalářské práce studenta aplikované informatiky a je spíše otázkou pro experta na cestovní ruch. Proto bylo dohodnuto zjednodušení problému. Při vytváření výjezdu z výlezů se nepočítá se vzdáleností oblasti od České republiky a pro všechny oblasti se řeší problém stejně. Pokud se mezi dvěma výlezy nachází méně než šest dnů bez výlezu u jednoho uživatele v jedné oblasti, jedná se o jeden výjezd, jinak se jedná o dva výjezdy. Např. jeden lezec bude mít první, osmý a patnáctý den v měsíci výlez, pak se jedná celkem o tři výjezdy. Když lezec bude mít výlez první, sedmý, jedenáctý a patnáctý den v měsíci do stejné oblasti, bude se jednat o jeden výjezd.

Tento skript má tedy na vstupu soubor se seřazenými výlezy podle UID uživatele a datumu výlezu, výstupem je UID uživatele, datum začátku výjezdu a oblast. Pokud bychom chtěli sledovat i dobu výjezdu, bylo by možné zde přidat datum ukončení výjezdu.

4.2.6 Získání států, oblastí a regionů

Aby se později mohly třídit výjezdy podle států, byl využit seznam všech států z internetové stránky https://www.zemepis24.cz/staty a každému státu bylo přiděleno ID. Dále byl připraven seznam všech oblastí, které byly získány ze seznamu cest pomocí tabulkového editoru Excel a textového editoru Notepad++. Výsledkem je čtveřice ID oblasti, jméno oblasti, region a ID státu.

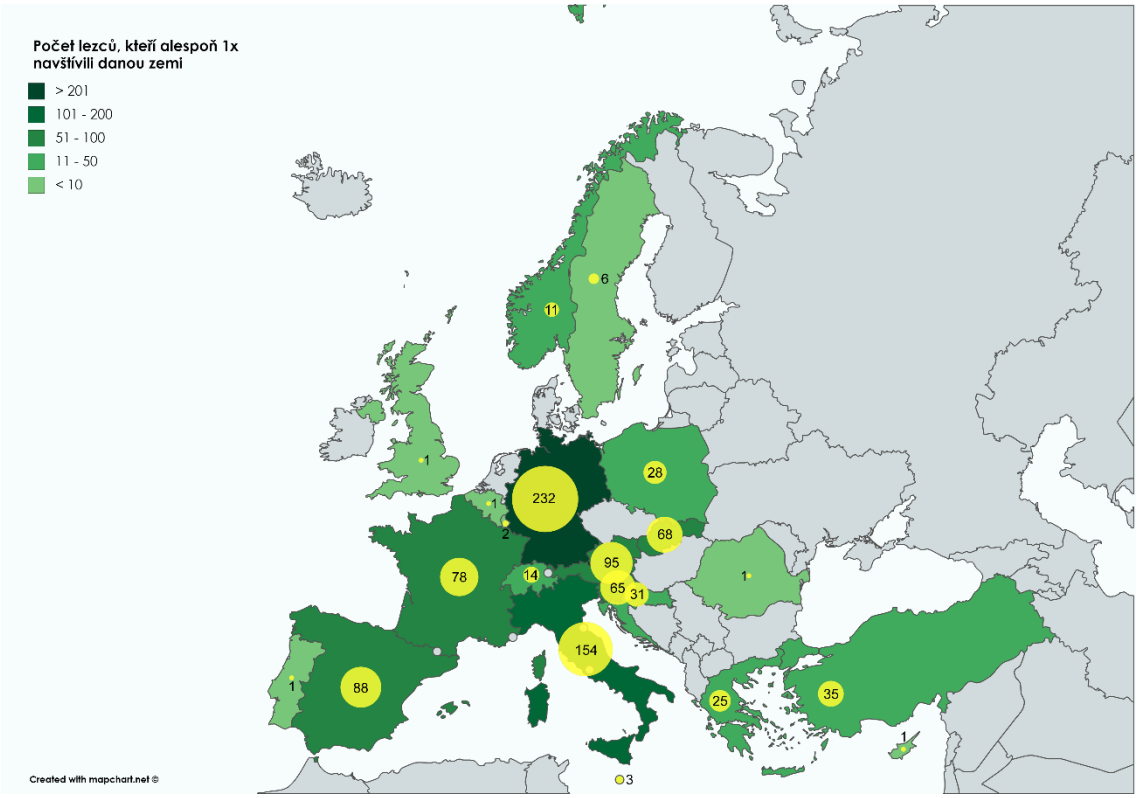
Obdobným postupem byl sestaven seznam regionů skládajících se ze čtyř sloupců. Prvním sloupcem je přidělené ID regionu. Ze seznamu cest byl použit název regionu, čímž vznikl druhý sloupec. Ten je pro zahraniční regiony ve formátu stát – region, např. Německo – Bavorsko, v případě českého regionu je uvedeno pouze jméno regionu bez státu. Následuje sloupec se zkráceným názvem regionu, např. Bavorsko. Ke každému regionu bylo přidáno ID státu, které tvoří poslední sloupec tabulky.

4.2.7 Shrnutí

Uvedeným postupem k 31. 12. 2018 bylo získáno 4 391 horolezců, 127 449 cest, 512 728 výlezů, 107 923 výjezdů, 4 755 oblastí, 218 regionů a 198 států. Pro pozdější bezproblémový import do databáze bylo nutné escapovat uvozovky a apostrofy, smazat speciální znaky (např. smajlíci) a odstranit případné duplicity.

5 Návrh aplikace

Předmětem bakalářské práce bylo vytvořit webovou aplikaci, čímž bude zajištěný přístup k datům z kteréhokoliv zařízení bez instalace dalších programů (kromě webového prohlížeče). Pro programování takovéto aplikace se nejlépe hodí PHP, které za tímto účelem bylo vytvořené. Pro usnadnění práce byl použit framework, konkrétně se jedná o Laravel, protože práce s ním je velmi efektivní a výstupem je dobře čitelný kód, který umožňuje pozdější rozšíření. Pro databázi bylo zvoleno MySQL, protože je nejrozšířenějším systémem řízení databáze a je zdarma.

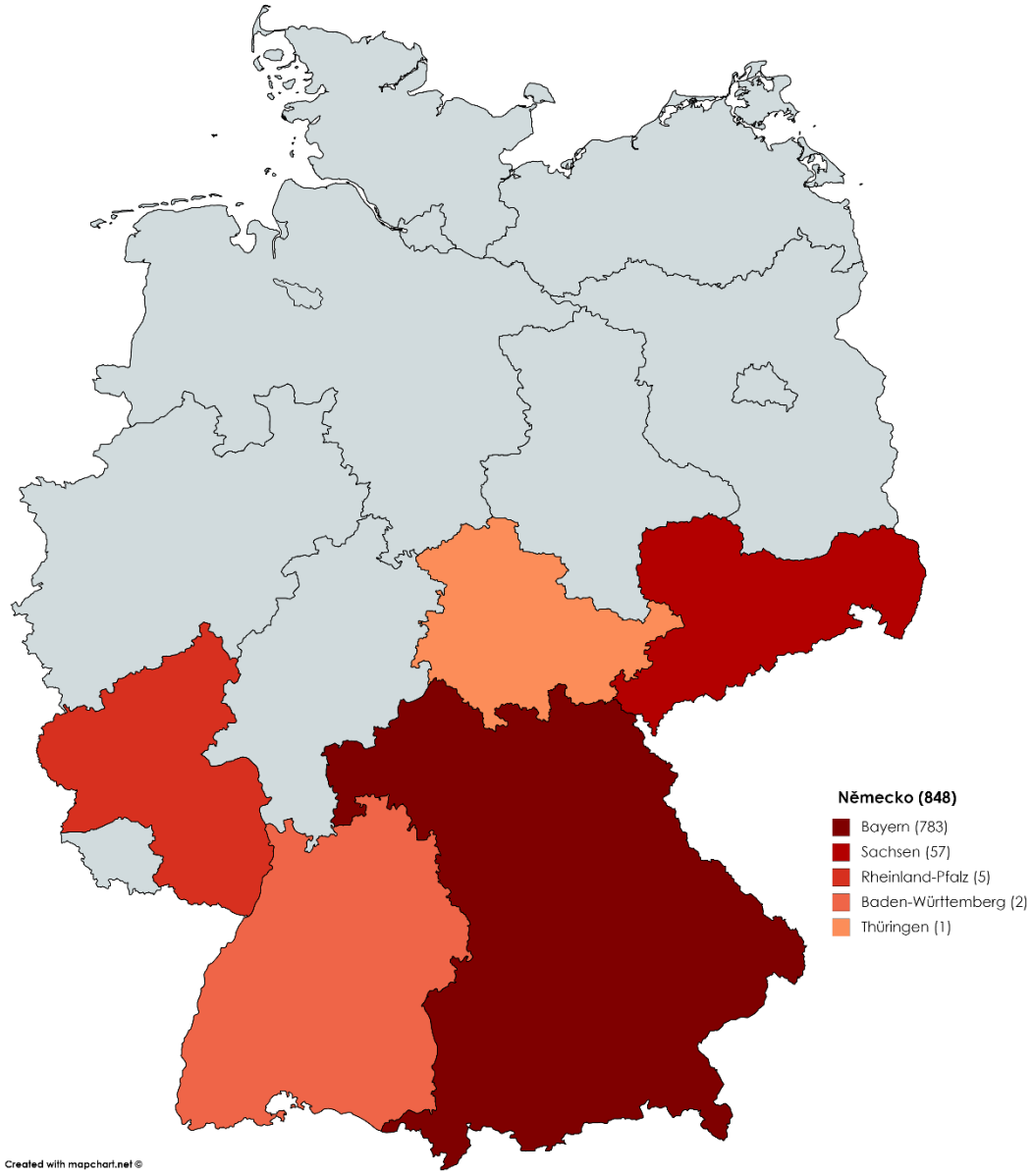


Obrázek 1 Ukázka využití dat – Evropa

Zdroj: vlastní zpracování

Aplikace tedy využívá data získaná z portálu Lezec. Základním požadavkem aplikace bylo přehledné a jednoduché uživatelské prostředí. Hlavním cílem bylo zobrazovat počty výjezdů a horolezců, kteří vycestovali v zadaném období celkem celosvětově, do konkrétního státu, regionu nebo oblasti. Dále byl požadavek na řazení států nebo oblastí podle abecedy, počtu výjezdů nebo horolezců vzestupně nebo sestupně. Aplikace rovněž musí umožňovat snadný export dat pro pozdější

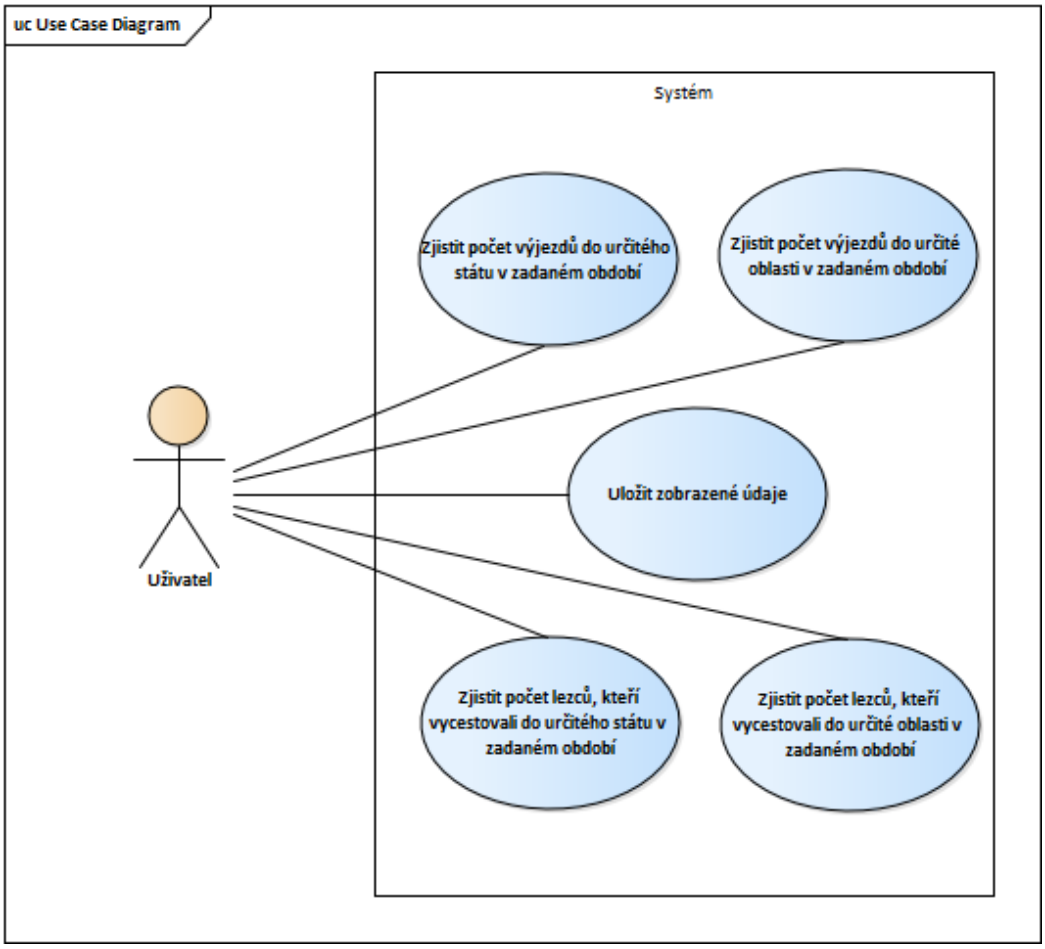
tvorbu grafů v Excelu nebo obarvení map v aplikaci ArcMap z platformy ArcGIS nebo podobných systémech jako je např. MapChart v ukázce na obrázcích 1 a 2.



Obrázek 2 Ukázka využití dat – Německo
Zdroj: vlastní zpracování

Přehled základních požadavků se nachází v diagramu případů užití na obrázku 3. Hlavní část aplikace je jedna webová stránka, která v úvodu má formulář pro výběr kritérií hledaných výjezdů jako jsou časové období, stát nebo region.

Pod formulářem se nachází údaje s celkovými počty výjezdů a horolezců, pod nimi je tabulka s počty v konkrétních oblastech, které splňují podmínky vyhledávání.



Obrázek 3 Diagram případů užití

Zdroj: vlastní zpracování

5.1 Podobné webové aplikace

Najít aplikaci, jenž by stejně jako aplikace, která je předmětem bakalářské práce, zpracovávala počty výjezdů lezců nebo nějakým jiným způsobem souvisela s lezeckým sportem a podobala se této, se bohužel nepodařilo. V následujících podkapitolách jsou popsány webové aplikace, které rovněž využívají PHP a databázi, pracují pravděpodobně podobným nebo stejným způsobem a týkají se cestovního ruchu.

5.1.1 Vyletnik.cz – Tipy na výlety

Na portálu Vyletnik.cz je stránka s tipy na výlety. Ve formuláři si lze zvolit region, oblast a kategorie (hrad, zámek, jeskyně, rozhledna atd.) a poté kliknout na tlačítko Zobrazit. Pod formulářem se zobrazí vizitky všech tipů na výlet, které splňují zadané požadavky. U tipu je uveden název, lokalita, popis, bodové hodnocení a komentáře. Tipy lze seřadit abecedně podle názvu, podle zajímavosti (výchozí hodnocení podle správců Vyletnik.cz) nebo podle hodnocení. V rejstříku se nachází necelých 10 tisíc tipů na výlet.

5.1.2 Geologické lokality

Na URL adrese http://lokality.geology.cz se nachází Geologické lokality, které obsahují databázi se záznamy o místech v České republice. Lokality lze vyhledávat podle názvu, identifikátoru, lokality, charakteristiky, stupně ochrany území nebo údajů o záznamu. Po vyhledání se zobrazí tabulka s ID a názvem lokality, okresem, informací o chráněném území a příznak, zda jsou u lokality fotografie. Po zvolení lokality ze seznamu se zobrazí mnoho informací včetně fotografií a mapy s umístěním lokality.

5.1.3 Booking.com

Booking.com je oblíbený rezervační systém ubytování. V databázi se nachází přes 28 milionů hotelů a dalších ubytovacích jednotek z celého světa. Na úvodní stránce se nachází formulář, kde se vyplní místo pobytu, datum příjezdu a odjezdu, počet dospělých, dětí a pokojů, po kliknutí na tlačítko Hledat se zobrazí stránka s výsledky. Nalezené nabídky lze řadit podle ceny, hodnocení, počtu hvězdiček nebo jejich kombinací.

5.1.4 Invia.cz

Invia je webová aplikace podobná Booking.com, ale nevyhledává pouze ubytovací zařízení, ale kompletní dovolené od cestovních kanceláří. Stránka má také formulář pro zadání kritérií. Prvním polem je výběr země, následuje termín nejzazšího odjezdu a příjezdu, délka pobytu, úroveň hotelu, strava, doprava a typ zájezdu. Po zadání požadavků se zobrazí seznam pobytů, který lze seřadit podle ceny,

četnosti prodejů nebo hodnocení. Vizitky v seznamu nalezených dovolených obsahují fotografii, hodnocení a varianty kombinací datumu příletu a odletu, doby pobytu, stravy a ceny. Po zvolení jedné možnosti se zobrazí stránka s detaily dovolené, jako jsou podrobné informace o hotelu, oblasti, fotografie a recenze zákazníků.

5.1.5 Eurostat

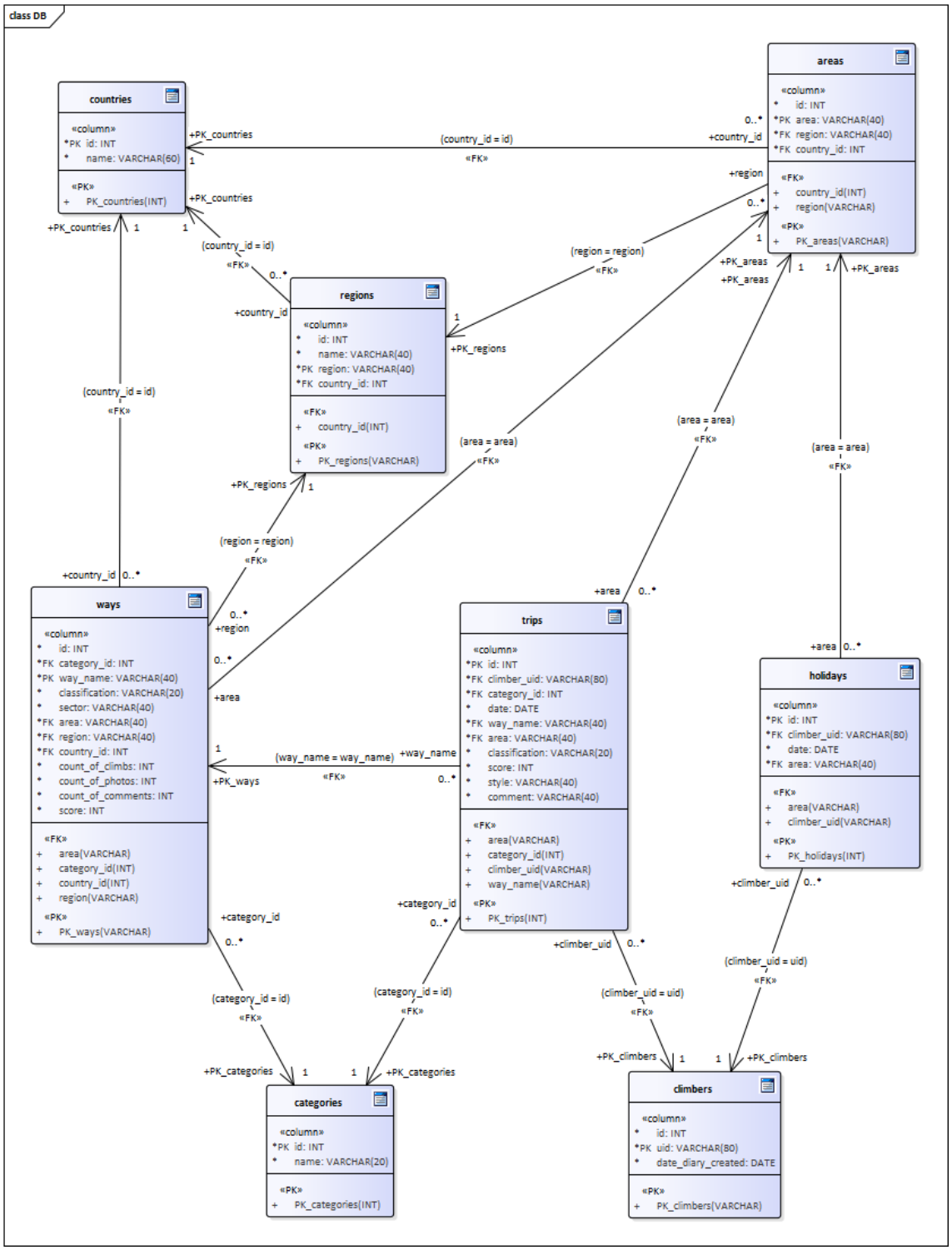
Statistický úřad Evropské unie Eurostat se mimo jiné zabývá i sledováním cestovního ruchu. Statistiky jsou dostupné na adrese https://ec.europa.eu/eurostat/web/tourism/statistics-illustrated ve formě obarvených map. Nad mapou se nachází rozevírací seznam pro výběr sledovaného údaje (počet dní strávených na dovolené, cílové destinace dovolené, náklady spojené s cestovním ruchem atd.). Na mapě je další rozevírací seznam pro výběr roku. Po najetí myší nad stát se objeví absolutní číslo.

6 Implementace

Aplikace byla vyvíjena v PHP frameworku Laravel v operačním systému Linux Mint 18.3. Nejprve bylo nutné vytvořil virtuální webový server, tedy nainstalovat Apache, MySQL a PHP 7.1, toho bylo docíleno prostřednictvím balíku XAMPP. Poté byla provedena samotná instalace Laravelu, čehož bylo docíleno pomocí Composeru. Po vytvoření nového projektu se na adrese 127.0.0.1:8000 zobrazila úvodní stránka Laravelu a mohlo začít programování vlastní aplikace.

6.1 Tvorba databáze

Pro uložení a zpřístupnění dat pro aplikaci bylo nutné vytvořit databázi. Jako systém řízení databáze bylo zvoleno MySQL. Tabulky byly vytvořeny podle diagramu na obrázku 4 a naplněny získanými daty. Vzhledem k vysokému počtu záznamů v tabulkách, bylo nutné vytvořit indexy pro rychlejší získávání výsledků po zadání dotazu a tím pozdější bezproblémový chod aplikace.



Obrázek 4 Návrh databáze

Zdroj: vlastní zpracování

6.2 Tvorba aplikace

Při vytváření aplikace bylo využito softwarové architektury MVC, tedy bylo provedeno rozdělení aplikace do tří oddělených částí na datový model, uživatelské rozhraní a řídící logiku.

6.2.1 Modely

Aplikační logika se nachází v modelech, které jsou uložené v adresáři app ve tvaru Model.php. V případě této aplikace se jedná zejména o dotazování na databázi. V ukázce uvedené níže se do proměnné rows uloží dvourozměrné pole složené z jména oblasti, počtu výjezdů do oblasti a počtu lezců, kteří tuto oblast navštívili, pro všechny oblasti, které vyhovují kritériím vyhledávání a seřadí je podle požadavku. Při práci s databází se v Laravelu používá odlišná syntaxe oproti SQL.

```
$rows = DB::table('areas')

  ->selectRaw('areas.area AS area_name, COUNT(areas.area) AS
    area_count, COUNT(DISTINCT holidays.climber_uid) AS
    climber_count')

  ->join('holidays', 'areas.area', '=', 'holidays.area')

  ->where('areas.country_id', '=', request()->country_id)

  ->where('holidays.date', '>=', request()->date_from)

  ->where('holidays.date', '<=', request()->date_to)

  ->groupBy('areas.area')

  ->orderBy(request()->order_by, request()->order_direction)

  ->get();
```

6.2.2 Pohledy

Pro prezentaci slouží pohledy, které jsou uložené ve složce resources/views a mají formát názvu jako view.blade.php. Nachází se zde obecné pohledy jako hlavička, navigační menu, patička a načtení skriptů, a ty jsou následně vloženy v každém specifickém pohledu. Obsah pohledů je kombinací HTML a PHP. Na uvedeném příkladu je vidět zkrácená a zjednodušená syntaxe oproti čistému PHP,

@foreach($rows as $row) by se v PHP zapsalo jako <?php foreach ($rows as $row) ?> a {{ $row->area_name }} jako <?php echo $ row->area_name ?>.

```
@foreach($rows as $row)
  <tr>
    <td>{{ $row->area_name }}</td>
    <td>{{ $row->area_count }}</td>
    <td>{{ $row->climber_count }}</td>
  </tr>
@endforeach
```

6.2.3 Kontroléry

Kontroléry mají na starost komunikaci mezi uživatelem a aplikací. Vytvářejí modely na základě požadavku a generují pohledy doplněné o data z modelů. Nachází se v adresáři app/Http/Controllers. Ve složce routes se nachází soubor web.php, který definuje zobrazený obsah pro URL adresy. Ukázka směrování je níže, po zadání adresy /oblasti do prohlížeče se zobrazí pohled index ve složce area a tento pohled dostane jako parametr areas všechny instance modelu Area (všechny záznamy v databázové tabulce areas).

```
Route::get('/oblasti', function () {
    return view('area.index', ['areas' => Area::all()]);
});
```

6.2.4 JavaScript

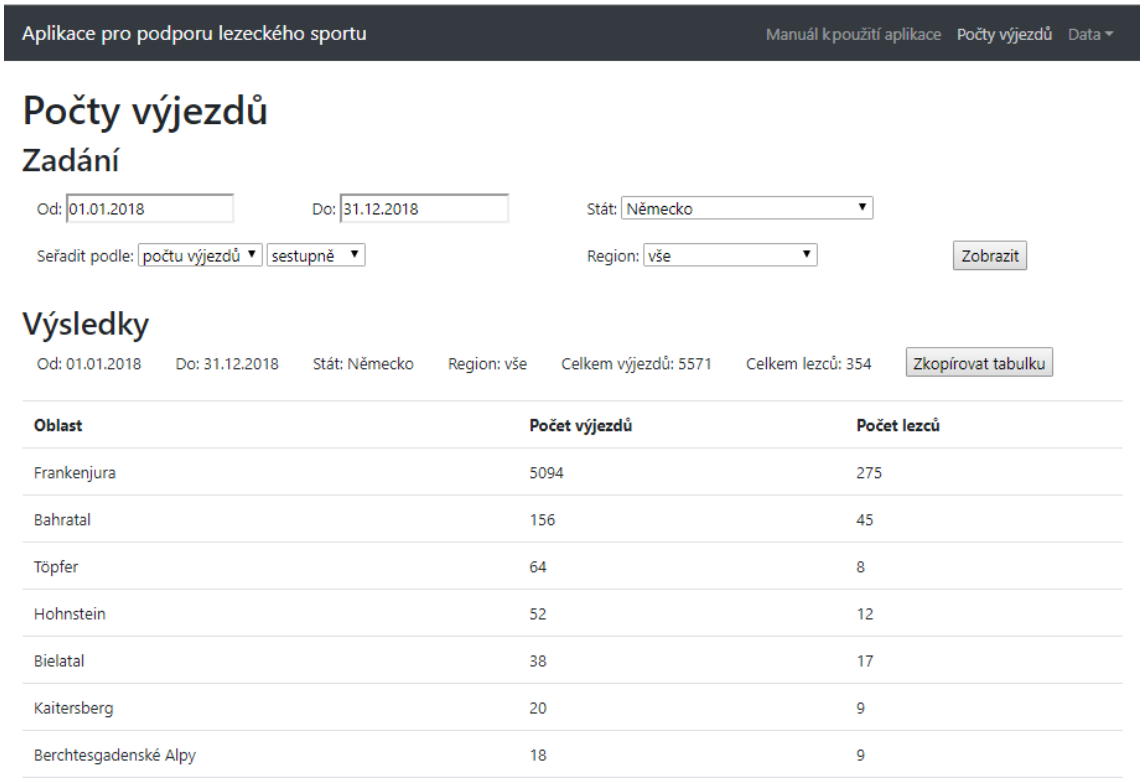
Pro zkopírování výjezdů do schránky bylo využito JavaScriptu, ve kterém byla napsána funkce, která je volána po kliknutí na tlačítko. Její kód se nachází níže, nejprve se vytvoří rozsah, do kterého je vložen obsah elementu, v případě aplikace, která je předmětem této práce to je tabulka s výjezdy lezců. Poté se vybere rozsah a příkazem copy zkopíruje do schránky. Příkaz alert vytvoří vyskakovací okno s informací pro uživatele, že výjezdy byly zkopírovány. Tato funkce nahrazuje ruční

označení celé tabulky myší a následné stisknutí klávesové zkratky Ctrl + C, případně kliknutí pravým tlačítkem myši a klepnutí na kopírovat.

```
function copyToClipboard(container) {
  var range = document.createRange();
  range.selectNode(document.getElementById(container));
  window.getSelection().addRange(range);
  document.execCommand("copy");
  alert("Výjezdy zkopírovány do schránky.")
}
```

6.2.5 Zobrazení

Ukázka výsledné aplikace se nachází na obrázku 5. Důraz byl kladen na přehledné zobrazení a snadné ovládání aplikace. Byly použity barvy, které uživatele nerozptylují. Jedná se pouze o bílou, černou a odstíny šedé.



Obrázek 5 Výsledná aplikace

Zdroj: vlastní zpracování

Pro grafickou úpravu aplikace je použita volně dostupná šablona frameworku Bootstrap. Její aplikace je jednoduchá, stačí nakopírovat soubory s kaskádovými styly a kódem JavaScriptu do příslušných adresářů ve složce public a nastavit cestu k těmto souborům v odpovídajících pohledech. Použitím Boostrapové šablony je zaručený responzivní design, takže aplikaci je možné bez problému používat na různě velkém monitoru, tabletu či mobilním zařízení.

Úplně nahoře se nachází navigační menu, kde se nachází odkaz na úvodní stránku, na které je zadání bakalářské práce a další základní informace, počty výjezdů (tedy nejdůležitější stránka aplikace), návod k použití aplikace. Poslední položkou menu je položka Data, která funguje jako rozevírací seznam, který se skládá z odkazů Státy, Regiony, Oblasti, Kategorie, Cesty, Výlety, Výjezdy a Lezci. Po kliknutí na jeden z těchto odkazů se zobrazí stránka vypisující všechny záznamy z příslušné tabulky. Pokud je šířka obrazovky nižší než 800 px, nezobrazuje se tato klasická nabídka, ale je zastoupeno skrytou navigací „hamburger“ (tři vodorovné čárky).

Šablona využívá dvanácti sloupcovou mřížku, u elementů lze definovat jejich velikost číslem, které udává, kolik sloupců má element k dispozici. Ve formuláři pro zadání hodnot vyhledávání mají pole pro zadání datumů včetně textu od/do každé šířku 3 (3/12). Dohromady tedy obsadí polovinu šířky stránky. Druhá polovina zbyla pro výběr státu, je zde tedy uvedeno číslo 6, stejně jako u nastavení řazení, které se nachází na začátku dalšího řádku. Pro výběr regionu je šířka nastavena na 4, zbytek řádku je určený pro tlačítko „Zobrazit“. V hlavičce výsledků jsou všechny sloupce stejně široké. Samotné výsledky jsou vypsány v tabulce, kde šířka sloupců není nastavena pevným poměrem, ale nastavuje se automaticky podle nejširších záznamů v tabulce.

Na konci každé stránky se nachází patička, která obsahuje rok vytvoření aplikace a jméno autora zarovnané vodorovně na střed stránky.

6.3 Umístění aplikace na Internet

Aby se k vytvořené aplikaci dostali i další uživatelé, bylo nutné ji z lokálního umístění zkopírovat na veřejně dostupný server. Nejprve byly vyexportovány všechny databázové tabulky a naimportovány do databáze webhostingu, poté byl zkopí-

rován celý projekt. V souboru .env umístěném v hlavním adresáři bylo nutné upravit údaje k databázi (hostitele, port, jméno databáze, uživatele a heslo) z localhostu na údaje obdržené od poskytovatele hostingu, změnit prostředí z lokálního na produkční, zakázat režim ladění (debug) a přepsat URL aplikace. Dalším krokem bylo promazat cache paměť, aby se tyto hodnoty změnily i v uložené mězipaměti. Aplikace se nyní nachází na URL adrese http://lezeni.loon.cz.

6.3.1 Chráněný přístup

Pro přístup do aplikace je požadováno přihlášení uživatele. Je použita metoda zabezpečení HTTP Access, která se nastavuje v souboru .htaccess pomocí formule „require valid-user“. Jména uživatelů a jejich hesla jsou uložena v souboru .htpasswd ve formátu jméno:heslo, kde heslo je zašifrováno pomocí fuknce bcrypt, např. ondracek:$2y$10$rtRI0ilHPO1qPv/ToVzC/OvOVT11D4Bu0M6t6WjwOPLKmeZD8u/Ye. Při této metodě zabezpečení je pro neautorizované osoby velmi obtížné se k aplikaci dostat, jedná se o nejspolehlivější ochranu. Pokud uživatel nezadá správné údaje, zobrazí se mu jen bílá stránka s informací, že server nemohl ověřit, zda má oprávnění k přístupu k dokumentu (chyba 401 – Unauthorized / Neoprávněný přístup).

6.4 Chyby v aplikaci

Vzhledem k tomu, že jsou využívána data získaná z portálu Lezec a tam si deníčky vyplňují uživatelé sami, a přitom neprobíhá kontrola správnosti vyplněných dat, tak se může např. stát, že mexická oblast El Potrero Chico podle aplikace spadá do České republiky, protože to nějaký uživatel zadal nesprávně. Tyto chyby nebyly opraveny, vzhledem k vysokému počtu záznamů není možné je všechny projít a zkontrolovat. Strojové opravení by bylo rovněž náročné, pravděpodobně by se dalo vyřešit s využitím umělé inteligence, ale to není předmětem této práce.

Dalším problémem je nejednotné zadávání cest, oblastí a regionů. To je částečně odstraněno smazáním nadbytečných mezer ve stažených datech a porovnáváním při počítání bez diakritiky a bez ohledu na velikost písmen. Nevyřešil se tím ale problém zcela, např. Malinská Skála a Malínské Skály jsou brány jako dvě oblasti, i když se ve skutečnosti jedná o tu samou. Toto nejde jednoduše

opravit, mohlo by se totiž skutečně jednat o dvě rozdílné oblasti s podobným názvem a k posouzení jsou vyžadovány hlubší znalosti z této oblasti.

Různé překlepy se rovněž nachází v datech, je možné např. zadat 30. 5. 1019 jako datum výlezu. Zde byla provedena pouze korekce dat, kde bylo zřejmé, jaké datum chtěl uživatel zadat. Ve většině případů to ale bylo sporné, a proto údaje zůstaly ponechány v původní podobě.

Všechny výše zmíněné chyby aplikace byly probrány na konzultaci a vzhledem k tomu, že se jich v aplikaci nachází zanedbatelné množství a pro užívání aplikace nejsou významné, bylo rozhodnuto chyby neopravovat. Jejich odstranění by zároveň zabralo velké množství času.

7 Shrnutí výsledků

Vytvořená aplikace funguje podle požadavku, zobrazuje počty výjezdů a lezců v jednotlivých oblastech podle zadaných kritérií vyhledávání jako jsou datum výjezdu, stát nebo region. Aplikace byla vytvořena pomocí PHP frameworku Laravel. Kromě PHP využívá i dalších nástrojů jako JavaScript, HTML, kaskádové styly a MySQL. Data byla získána především z portálu Lezec pomocí skriptů napsaných v jazyce Python. Databáze obsahuje celkem 198 států, 218 regionů, 4 755 oblastí, 4 391 horolezců, 127 449 cest, 512 728 výlezů a 107 923 výjezdů aktualizovaných k datu 31. 12. 2018.

Výjezdy byly získány z výlezů. Pokud se u jednoho uživatele nachází mezi dvěma výlezy v jedné oblasti pět nebo méně dnů bez výlezu, jedná se o jeden výjezd, v opačném případě se jedná o dva výjezdy. Ze získaných dat vyplývá, že průměrný výjezd je složený z přibližně 4,75 výlezů.

Aplikace je umístěná na Internetu na adrese http://lezeni.loon.cz a je zabezpečená pomocí HTTP Access, takže pro přístup k ní je potřeba znát přihlašovací údaje. Vytvořená aplikace bude přínosná hlavně pro přednášející a studenty v oboru cestovního ruchu. I přes chyby uvedené v kapitole 6.4 se jedná o zdařilou aplikaci.

8 Závěry a doporučení

Cílem bakalářské práce bylo vytvořit prototyp webové aplikace pro podporu lezeckého sportu. Vytvořená webová aplikace bez problémů funguje na všech běžným operačních systémech a ve všech obvyklých prohlížečích. Aplikaci je v budoucnu možné rozšířit o data z dalších horolezeckých portálů jako je např. 8a.nu nebo data aktualizovat ze serveru Lezec. Smysl by dávalo do aplikace doplnit dobu výjezdu, to by vyžadovalo k výjezdům doplnit datum posledního výlezu v rámci jedné cesty. Dalším rozšířením by mohlo být automatické generování grafů a barvení map podle zobrazovaných výsledků.

9 Seznam použité literatury

BLANKENSHIP, Gary. (2018). *Laravel 5 Official Documentation* [online]. Dostupné z: https://leanpub.com/laravel-5

DUBOIS, Paul. (2014). *MySQL cookbook.* Third edition. Sebastopol, CA: O'Reilly Media. ISBN 978-1-449-37402-0.

DUCKETT, Jon. (2011). *HTML & CSS: design and build websites.* Indianapolis, IN: Wiley. ISBN 978-1-118-00818-8.

GRUDL, David. (2019). Školení pro. *Vývoj webových aplikací v Nette Framework* [online]. Praha: Grudl. [cit. 2019-04-20]. Dostupné z: https://www.skoleniphp.cz/skoleni-nette-vyvoj-webovych-aplikaci

LUTZ, Mark. (2009). *Learning Python.* Fourth edition. Sebastopol, CA: O'Reilly Media. ISBN 978-0-596-15806-4.

MCLAUGHLIN, Brett. (2012). *PHP & MySQL: the missing manual.* Sebastopol, CA: O'Reilly Media. ISBN 978-0-596-51586-7.

MURACH, Joel. (2015). *Murach's MySQL.* Second edition. Fresno, CA: Mike Murach & Associates. ISBN 978-1-890774-82-0.

NIXON, Robin. (2014). *Learning PHP, MySQL & JavaScript: with jQuery, CSS & HTML5.* Fourth edition. Sebastopol, CA: O'Reilly Media. ISBN 978-1-491-91866-1.

NJENGA, Alice. (2018). Raygun. *10 Popular PHP frameworks in 2019* [online]. Wellington: Raygun. [cit. 2019-04-20]. Dostupné z: https://raygun.com/blog/top-php-frameworks/

OVH Registrar. (2015). *Zaregistrujte si doménu .nu* [online]. Roubaix: OVH. [cit. 2019-04-15]. Dostupné z: https://www.ovh.cz/domeny/dotnu.xml

STAUFFER, Matt. (2016). *Laravel: up and running: a framework for building modern PHP apps.* Sebastopol, CA: O'Reilly Media. ISBN 9781491936085.

ULLMAN, Larry E. (2012). *PHP and MySQL for Dynamic Web Sites.* Fourth edition. Berkeley, CA: Peachpit Press. ISBN 978-0-321-78407-0.

WELLING, Luke a Laura THOMSON. (2009). *PHP and MySQL Web development.* Fourth edition. Upper Saddle River, NJ: Addison-Wesley. ISBN 978-0-672-32916-6.

WU, Nathan. (2016). *Learning Laravel 5: Building Practical Applications* [online]. Dostupné z: https://www.goodreads.com/book/show/28265327-learning-laravel-5

10 Přílohy

1) Manuál k použití aplikace

Manuál k použití aplikace

K vyhledání počtu výlezů je nutné v hlavním menu vybrat položku počty výlezů (URL adresa http://lezeni.loon.cz/pocty). Poté na zobrazené obrazovce zadat počáteční a koncové datum období, ve kterém chceme počty výjezdů sledovat. Datum se zadává v českém formátu DD.MM.RRRR, např. 28.09.1992. To lze provést buď napsáním číslic na klávesnici nebo navolením v kalendáři po kliknutí na velkou šipku směřující směrem dolů. Malé šipky směřující nahoru a dolů slouží ke zvyšování, respektive snižování hodnoty dne, měsíce a roku. Křížek slouží pro smazání zadané hodnoty.

Dalším polem je stát, ten se vybírá ze seznamu. Pro rychlejší vyhledání státu je možné po kliknutí na seznam napsat počáteční písmena a tím se hned dostat na hledaný stát nebo do jeho blízkosti. Následují dva seznamy pro nastavení způsobu řazení, je možné výsledek seřadit abecedně podle názvů oblastí nebo států, počtu výjezdů nebo počtu lezců, vše buď vzestupně nebo sestupně. Poslední pole je region, ten se vybírá stejným způsobem jako stát. Pokud není stát vybraný, nabízejí se všechny regiony, v opačném případě jsou k dispozici pouze regiony ležící ve zvoleném státu.

Tlačítkem „Zobrazit“ se zadaný výběr odešle a zobrazí se tabulka oblastí s počty výjezdů a lezců podle zadaných kritérií. Nad tabulkou se nachází zadané hodnoty ve formuláři. Jsou zde pro případ, že bychom si je ve formuláři změnili a zapomněli, pro které hodnoty se tabulka vygenerovala. Ještě tu leží tlačítko „Zkopírovat tabulku“, které uloží všechny zobrazené údaje do schránky ve formátu vhodném pro import do tabulkových editorů jako je Excel, v něm potom stačí pouze údaje vložit (po kliknutí pravým tlačítkem myši zvolit možnost vložit nebo pomocí klávesové zkratky Ctrl + V) a automaticky se údaje rozdělí do buněk.

V hlavní nabídce se ještě nachází možnost data. Po kliknutí se rozbalí seznam všech tabulek, které jsou v databázi aplikace, po výběru jedné z nich se zobrazí kompletní obsah vybrané tabulky. Další položkou menu je tento manuál k použití aplikace.

Univerzita Hradec Králové
Fakulta informatiky a managementu
Akademický rok: 2018/2019

Studijní program: Aplikovaná informatika
Forma: Kombinovaná
Obor/komb.: Aplikovaná informatika (ai3-k)

Podklad pro zadání BAKALÁŘSKÉ práce studenta

PŘEDKLÁDÁ:	ADRESA	OSOBNÍ ČÍSLO
Ondráček Lukáš	Terézy Novákové 153, Proseč	I1600482

TÉMA ČESKY:

Aplikace pro podporu lezeckého sportu

TÉMA ANGLICKY:

The Application for Support for Climbing Sport

VEDOUCÍ PRÁCE:

doc. RNDr. Petra Poulová, Ph.D. - KIKM

ZÁSADY PRO VYPRACOVÁNÍ:

Cílem bakalářské práce je na základě analýzy stávajících aplikací vytvořit program pro podporu a sdílení informací specifické oblasti cestovního ruchu věnující se skalnímu lezení.

Práce bude zahrnovat:
1) Analýzu aplikací sloužících pro sdílení lezeckého sportu
2) Návrh webové aplikace
   a) Analýza
   b) Výběr implementačních nástrojů
   c) Návrh databáze
   d) Funkční náhled (diagram případů užití)
   e) Prototyp
3) Závěr

Odborný konzultant:
Mgr. David Chaloupský, Ph.D.

SEZNAM DOPORUČENÉ LITERATURY:

DUBOIS, Paul. MySQL cookbook. Third edition. Sebastopol, CA: O'Reilly, 2014. ISBN 978-1-449-37402-0.
MCLAUGHLIN, Brett. PHP & MySQL: the missing manual. Sebastopol , CA: O'Reilly Media, 2012. Missing manual. ISBN 978-0-596-51586-7.
MURACH, Joel. Murach's MySQL. Second edition. Fresno, CA: Mike Murach & Associates, 2015. ISBN 978-1-890774-82-0.
NIXON, Robin. Learning PHP, MySQL & JavaScript: with jQuery, CSS & HTML5. Fourth edition. Sebastopol, CA: O'Reilly Media, 2014. ISBN 978-1-491-91866-1.
ULLMAN, Larry E. PHP and MySQL for dynamic Web sites. Fourth edition. Berkeley, CA: Peachpit Press, 2012. ISBN 978-0-321-78407-0.
WELLING, Luke a Laura THOMSON. PHP and MySQL Web development. Fourth edition. Upper Saddle River, NJ: Addison-Wesley, 2009. Developer's library. ISBN 978-0-672-32916-6.

Podpis studenta: Ondráček **Datum:** 20.9.2018

Podpis vedoucího práce: Paul **Datum:** 20.9.2018