

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

SYSTÉM PRO ZÁZNAM A VYHODNOCOVÁNÍ DAT Z GPS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

ŠIMON ŽÁČIK

BRNO 2015



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

SYSTÉM PRO ZÁZNAM A VYHODNOCOVÁNÍ DAT Z GPS

SYSTEM FOR RECORDING AND EVALUATION OF GPS DATA

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

ŠIMON ŽÁČEK

VEDOUcí PRÁCE

SUPERVISOR

Ing. VLADIMÍR BARTÍK, Ph.D.

BRNO 2015

Abstrakt

Bakalářská práce se zabývá návrhem a implementací webového systému pro vyhodnocování výsledků a sledování průběhu športových soutěží na základe údajů GPS. Získání GPS dat je možné použitím mobilné aplikace pro platformu Android nebo importem souborů typu TCX. Uživatelské rozhraní umožňuje sledovat pohyb závodníků v reálnem čase, případně přehrávání průběhu podujetí zpětně. Součástí je automatické generování průběžného a celkového pořadí účastníků. Systém je implementován v jazyku PHP, klientské rozhraní bylo vytvořené za pomoci HTML, CSS, javascriptových knižnic jQuery, Bootstrap a OpenLayers pro práci s mapami.

Abstract

The aim of the thesis is design and implementation of web system for results evaluation and tracking of sports events based on GPS data. GPS data acquisition is possible by mobile application for Android operating system or by import TCX files. User interface allows realtime competitors tracking or replay of competition progress. The part of application is automatic generating of intermediate and overall standings of athletes. System is implemented in PHP scripting language, user interface was developed in HTML, CSS, javascript libraries jQuery, Bootstrap and OpenLayers for work with maps.

Klíčová slova

GPS, webová aplikace, mapy, zaznamenávání polohy, automatické generování výsledků, sledování průběhu závodu, TCX, OpenLayers, Android

Keywords

GPS, web application, maps, position recording, automatic results creation, tracking during race, TCX, OpenLayers, Android

Citace

Šimon Žáčik: Systém pro záznam a vyhodnocování dat z GPS, bakalářská práce, Brno, FIT VUT v Brně, 2015

System pro záznam a vyhodnocování dat z GPS

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Vladimíra Bartíka Ph.D.

.....
Šimon Žáčik
17. května 2015

Poděkování

Rád bych poděkoval panovi Ing. Vladimírovi Bartíkovi Ph.D za vedení práce a odbornou pomoc při jej řešení.

© Šimon Žáčik, 2015.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1 Úvod	3
2 Princípy tvorby webových aplikácií	4
2.1 Technológie na strane servera	4
2.1.1 PHP	5
2.1.2 MySQL	5
2.2 Technológie na strane klienta	5
2.2.1 HTML	5
2.2.2 CSS	5
2.2.3 Javacript	6
2.3 Vizualizácia GPS dat vo webovom prostredí	6
2.3.1 Google Maps	6
2.3.2 Openlayers	7
3 Systém GPS	8
3.1 Segmenty systému GPS	8
3.1.1 Kozmický segment	8
3.1.2 Riadiaci segment	8
3.1.3 Užívateľský segment	9
3.2 Družicové signály	9
3.3 Súradnicové systémy	9
3.4 Nadmorská výška	9
3.5 Spôsoby merania	9
3.6 Čas GPS	10
3.7 Presnosť systému GPS	10
4 Prijímače GPS a prenos dát na server	11
4.1 Špecifické prijímače s GSM	11
4.2 Univerzálne športové prijímače	12
4.3 Prijímače zabudované v mobilných zariadeniach	12
5 Uchovávanie GPS a geografických dát v textových formátoch a ich zobrazovanie	13
5.1 Formát GeoJson	13
5.2 Formát KML	13
5.3 Formát TCX	15

6	Návrh riešenia	17
6.1	Špecifikácia požiadavkov	17
6.2	Voľba implementačných prostriedkov	17
6.2.1	PHP a MySQL	17
6.2.2	HTML, CSS, Javascript	17
6.2.3	Android	19
6.3	Dátový model	19
6.4	Zhrnutie návrhu	19
7	Popis funkčnosti aplikácie a implementácia	21
7.1	Štruktúra webovej aplikácie	21
7.2	Registrácia a prihlasovanie užívateľov	21
7.3	Správa profilu	22
7.4	Založenie a správa závodu	22
7.5	Vytvorenie a zrušenie účasti na závode	23
7.6	Odoslanie údajov o polohe	23
7.7	Prijímanie a spracovávanie údajov o polohe	24
7.8	Import súboru TCX	25
7.9	Sledovanie priebehu závodu	26
7.10	Spätné prehrávanie závodu	27
7.11	Vyhodnocovanie výsledkov	28
7.12	Inštalácia systému	29
8	Testovanie	30
8.1	Testovanie prenosu údajov na server	30
8.2	Testovanie zobrazovania priebehu závodu	30
8.3	Testovanie importu údajov z GPS zariadení	31
8.4	Testovanie užívateľského rozhrania	31
9	Možnosti pokračovania projektu	33
10	Závěr	34
A	Obsah CD	37

Kapitola 1

Úvod

Systém globálneho určovania polohy (GPS) zaznamenáva v posledných rokoch výrazné rozšírenie možností svojho využitia. Služba pôvodne vytvorená pre armádne účely je dnes neodmysliteľnou súčasťou civilných navigácií a stáva sa stále obľúbenejšou v mnohých ďalších odvetviach.

Zaujímavou skupinou užívateľov sú účastníci, respektíve organizátori športových súťaží, ktoré sú založené na vyhodnocovaní dosiahnutého času na vopred stanovenej trati. Sú to napríklad bežecké alebo cyklistické podujatia. Dostatočne presné údaje zo systému GPS v kombinácii s ich správnym spracovaním dokážu v porovnaní s inými spôsobmi poskytnúť omnoho detailnejšie informácie o priebehu takýchto súťaží ako iné metódy tvorby výsledkov. Táto práca sa zaoberá návrhom a implementáciou webového portálu pre jednoduchú správu závodu. Od registrácie užívateľa na závod, cez sledovanie priebehu závodu, s možnosťou spätného prehrávania až po vyhodnotenie celkových výsledkov.

Kapitola 2 je venovaná popisu spôsobov tvorby moderných webových aplikácií, vrátane možností využitia voľne dostupných služieb a knižníc pre prácu s mapami. Čitateľa do problematiky systému GPS uvádza kapitola 3, ktorá stručne vysvetľuje jeho základné princípy. V kapitole 4 sú vypísané typy GPS prijímačov využiteľných v tejto práci spolu s ich výhodami a nevýhodami. Kapitola 5 sa zaoberá možnosťami ukladania GPS dát v počítači v textových formátoch, ktoré boli využité pri tvorbe práce. Návrh aplikácie je popísaný v kapitole 6. Vysvetlenie riešenia a implementácia jej jednotlivých častí v kapitole 7. V nasledujúcej kapitole 8 sa nachádza popis spôsobov a výsledkov uskutočneného testovania. Ďalšie možnosti pokračovania práce a rozširovania aplikácie sú uvedené v kapitole 9.

Kapitola 2

Princípy tvorby webových aplikácií

Webové stránky a aplikácie sú v súčasnosti jednou z najpoužívanejších služieb internetu. Nie vždy však vyzerali tak ako v dnešnej dobe. Od roku 1990¹, kedy sir Tim Berners-Lee v organizácii CERN² spustil prvú webovú stránku³ prešli obrovským vývojom. Z pôvodných statických dokumentov určených na prezeranie sa vyvinuli dynamické a interaktívne aplikácie, v ktorých môžu užívatelia ovplyvňovať ich obsah. V tejto kapitole sú popísané základné technológie, ktoré sú v súčasnej dobe najviac využívané pri ich tvorbe. Technológie rozdelím na tie, ktoré pracujú na strane servera a tie, ktoré sa využívajú na strane klienta, zvyčajne ako súčasť internetového prehliadača.

2.1 Technológie na strane servera

Technológie na strane servera slúžia pre dynamické generovanie obsahu webových stránok. Narozdiel od pôvodných webových stránok sa teda na jednej adrese môže v rôznom čase zobrazovať rozličný obsah alebo môže byť obsah variabilný dokonca pre každého užívateľa. Zjednodušene to funguje tak, že užívateľ pomocou svojho internetového prehliadača odošle HTTP požiadavku na webový server. Ten ho spracuje a zavolá odpovedajúci skript. Tento skript vygeneruje obsah ktorý sa ako HTML stránka zašle naspäť klientovi, ktorý o ňu žiadal a zobrazí sa mu v jeho klientskej aplikácii.

Podrobnejšie sa táto kapitola venuje jazyku PHP v kombinácii s databázou MySQL, nakoľko práve tieto boli použité v tomto projekte. K ďalším podobným technológiám patria napríklad [4]:

- CGI (Common Gateway Interface) - spúšťanie spustiteľných súborov a následné generovanie HTML stránok
- SSI (Server Side Includes) - príkazy vložené do HTML súborov, ktoré server pre odoslaním spúšťa
- ASP (Active Server Pages) - technológia Microsoftu použiteľná na platforme Windows
- SSJS (Server Side JavaScript) - skripty vykonávané na serveri a výsledok odosielaný ako súčasť stránok
- MSSQL - databázový server od spoločnosti Microsoft

¹<http://home.web.cern.ch/topics/birth-web>

²European Organization for Nuclear Research - Európska organizácia pre jadrový výskum

³<http://info.cern.ch/hypertext/WWW/TheProject.html>

2.1.1 PHP

PHP je voľne šíriteľný open-source skriptovací jazyk, umožňujúci procedurálne aj objektovo orientované programovanie. Jeho súčasné využitie je hlavne pre programovanie interaktívnych webových stránok a aplikácií. História začína v roku 1995, keď Rasmus Lerdorf vytvoril sadu skriptov pre počítanie prístupu ku svojim stránkam, ktoré boli neskôr uverejnené pod názvom Personal Home Page Tools [8]. Od tohto roku prešiel jazyk dlhým vývojom a súčasná najnovšia verzia je PHP 5.6 ⁴.

Syntax jazyka PHP je veľmi podobná jazykom C alebo Java. Bloky kódu sú vkladané medzi značky `<? a ?>`. Všetok text mimo týchto značiek je posúdený ako kód značkovacieho jazyka HTML. Tieto segmenty sa pred odoslaním spoja a užívateľ tak dostáva vygenerovaný HTML dokument bez toho aby mal predstavu o tom, čo sa deje na strane servera.

2.1.2 MySQL

Len veľmi malé percento webových aplikácií postavených na PHP pracuje bez nejakej databázy. Databáza je kolekcia dát v štruktúrovanej forme uložená v počítačovom systéme [2]. Najčastejšie využívanou databázou v jazyku PHP je relačná⁵ databáza MySQL, ktorú možno v spojení s PHP používať bezplatne[11]. Pre chod databázy je potrebný databázový server a nastavenie pripojenia cez PHP skripty. Ako napovedá názov s databázou sa pracuje pomocou dotazovacieho jazyka SQL.

2.2 Technológie na strane klienta

Správne zobrazenie webovej stránky získanej zo servera je úlohou internetového prehliadača. Využívajú sa pri tom tri základné technológie, ktoré stručne popíšem v nasledujúcich podkapitolách.

2.2.1 HTML

Jazyk HTML bol vytvorený k tomu, aby dal webovým stránkam zmysel [3]. Slúži pre vytváranie logickej štruktúry dokumentov. Syntax jazyka je zložená s html značiek (tagov), ktoré sa delia na párové (`<h1>Nadpis prvej úrovne</h1>`, `<p>Odstavec</p>`) a nepárové (`
`). Zmyslom týchto značiek je prideliť zobrazovanému textu nejaký význam, pričom neurčujú vzhľad dokumentu. Pomocou jazyka HTML je text možné členiť do blokov, vkladať obrázky a iné typy súborov alebo vytvárať webové formuláre. V súčasnosti je najnovšou verziou jazyka HTML5⁶.

2.2.2 CSS

CSS alebo kaskádové štýly⁷ sú nástrojom pre dotváranie vzhľadu webových stránok, popísaných jazykom HTML. Slúžia pre nastavenie, farby, veľkosti alebo pozície jednotlivých elementov. Možno ich do dokumentu vkladať tromi spôsobmi:

⁴<http://php.net/>

⁵V relačnej databáze sú údaje uložené v tabuľkách (reláciach) zložených s riadkov (jednotlivé záznamy) a stĺpcov (atribúty).

⁶http://www.w3schools.com/html/html5_intro.asp

⁷<http://www.w3schools.com/css/>

- priame štýly - atribútom style v HTML elemente, najmenej prehľadné ale občas užitočné riešenie
- štýly v hlavičke - vloženie na začiatok html dokumentu
- externý súbor - vložením súboru .css do html dokumentu

2.2.3 Javacript

Javascript je programovací jazyk interpretovaný vo webovom prehliadači a spúšťa sa až po načítaní stránky. V súčasnosti sa využíva najmä k dynamickým zmenám obsahu na stránke, tvorbe navigačných prvkov alebo validáciu formulárov. Pre zaistenie bezpečnosti do neho vývojári nezabudovali možnosť pracovať so súborami na počítači, na ktorom je spúšaný [19]. Syntax jazyka je znova veľmi podobná známym programovacím jazykom ako napríklad C. Na základe javascriptu vzniklo mnoho knižníc, ktoré rozširujú jeho funkcionality, prípadne zjednodušujú komplikované funkcie do krátkych príkazov. Najznámejšou je knižnica jQuery⁸ s rozšírením jQuery UI⁹ pre prácu s užívateľským rozhraním.

2.3 Vizualizácia GPS dat vo webovom prostredí

Nakoľko sa táto práca zaoberá vyhodnocovaním údajov na základe GPS, bolo potrebné do systému pridať podporu pre zobrazovanie interaktívnej mapy. Pre webové aplikácie existuje mnoho rôznych služieb ktoré to dokážu zabezpečiť. Pri návrhu práce som zvolil voľne dostupné Google Maps API pre vytvorenie mapového podkladu. Javascriptovú knižnicu Openlayers 3 som použil pre zabudovanie mapy do webového rozhrania, prácu s mapou a pre pridávanie ďalších objektov do mapy.

2.3.1 Google Maps

Aplikácia Google Maps bola spustená v roku 2005 spoločnosťou Google, Inc. [20]. Ide o voľne dostupnú službu s celosvetovým pokrytím a zobrazením v režime so zobrazením ciest a ulíc (viď obr. 2.1), terénu alebo mapy zloženej zo satelitných snímok. Umožňuje vyhľadávanie na základe adresy, tvorbu trás medzi dvoma zadanými miestami, pridávanie obrázkov k zaujímavým miestam alebo 3D zobrazenie ulíc zloženého z panoramatických fotografií. Neustále však pribúda nová funkcionality. Okrem samotnej webovej aplikácie Google poskytuje aj aplikačné rozhranie (API) pre vloženie mapy do vlastnej stránky, kde mapa zostáva aj naďalej plne interaktívna. Túto službu momentálne využíva viac ako 800 000 webov [5], je teda veľký predpoklad, že bude aj naďalej plne podporovaná [14] [17].

Využívanie Google API je pre tvorcov webových aplikácií bezplatné za určitých podmienok. Pre užívateľov musí byť web dostupný bezplatne a verejne. Je obmedzené rozlíšenie zabudovanej mapy na 640x640px, obmedzený počet žiadostí na zistenie namorskej výšky, adresy a niekoľko ďalších pomienok [16]. Pomocou API sa dajú do mapy dynamicky vkladať geometrické objekty, popisy miest alebo tvoriť trasy, je teda vhodné pre rôzne typy webových aplikácií.

⁸<https://jquery.com/>

⁹<https://jqueryui.com/>



Obrázek 2.1: Zobrazenie základnej mapy od spoločnosti Google

2.3.2 Openlayers

Openlayers je javascriptová knižnica poskytujúca funkcie pre zobrazenie mapových dát vo webových prehliadačoch. Funguje podobne ako napríklad Google Maps ale je vyvíjaná ako opensource software a je vydávaná pod licenciou BSD [10]. Openlayers pracuje s mapou pomocou vrstiev. Základnou vrstvou je mapový podklad, pre ktorý je možné využiť rôzne druhy máp, napríklad Google Maps alebo OpenStreetMaps. Do ostatných vrstiev je možné pridávať objekty a zobrazovať ich podľa potreby užívateľov. Podobne ako Google Maps aj Openlayers poskytuje javascriptové API pre zabudovanie do webovej aplikácie.

Kapitola 3

System GPS

Global positioning system (GPS) vznikol ako dôsledok potreby ľudstva čo najoptimálnejšieho určovania polohy a jednoznačnej trasy pohybu (navigácie) k iným bodom. V počiatkoch navigácie prevažovala orientácia podľa známych bodov na Zemi, prípadne pomocou hviezd a planét. S rozvojom techniky sa tieto metódy rozšírili o uhlové merania k týmto bodom, čím sa získala veľmi vysoká presnosť určenia aktuálnej polohy. Avšak nevýhodou je nutnosť viditeľnosti daného objektu. Po vynájdení rádiových vysielačov sa vyvíja systém určovania polohy založený na diaľkomernom princípe, čo je meranie vzdialenosti k niekoľkým vysielačom so známou polohou a následným výpočtom aktuálnej polohy na základe týchto údajov [7]. Vysielače boli najskôr umiestnené na Zemi, s rozvojom kozmického programu sa presúvali do vesmíru, čím vznikla súčasná podoba GPS - určovanie polohy pomocou družíc obiehajúcich okolo Zeme. Primárne vznikol ako vojenský projekt americkej armády NAVSTAR GPS, avšak rýchlo sa stal dostupným aj pre civilné účely. V súčasnosti má po celom svete desiatky miliónov užívateľov [12]. Existuje mnoho ďalších alternatív k GPS ako napríklad GLONASS, GALILEO alebo Transit avšak ich rozšírenie je omnoho menšie, preto sa nasledujúce podkapitoly venujú výhradne systému GPS.

3.1 Segmenty systému GPS

System GPS tvoria 3 základné segmenty, ktoré sú vzájomne nezávislé. Previazané sú iba presným časom, základným prvkom celého systému.

3.1.1 Kozmický segment

Kozmický segment je tvorený družicami na obežnej dráhe Zeme. Každá družica má nemennú dráhu obehu, pričom čas obehu je približne 12 hodín. Sú systematicky rozmiestnené tak, aby v každom okamihu v každom mieste na Zemi boli viditeľné minimálne 4 z 28 obiehajúcich družíc [12]. Každá družica nepretržite (mimo času potrebného k údržbe) vysiela navigačné signály. Popisom signálov sa zaoberá kapitola 3.2.

3.1.2 Riadiaci segment

Riadiaci segment je tvorený sústavou pozemných monitorovacích staníc, hlavnou riadiacou stanicou a stanicami pre komunikáciu s družicami. Všetky sú umiestnené na amerických základniach rozmiestnených po celom svete. Hlavnou úlohou tejto, pre bežného užívateľa

neviditeľnej súčasti systému, je vysielanie údajov o obežnej dráhe družiciam, nastavovanie hodín a aktualizovanie navigačných správ.

3.1.3 Užívateľský segment

Užívateľský segment sa skladá z GPS prijímačov, užívateľov a vyhodnocovacích postupov. Prijímače slúžia na spracovanie prijatých signálov z družíc. Pre určenie polohy sú potrebné súčasné údaje z minimálne štyroch družíc aby bolo možné vypočítať všetky štyri súradnice. Tri priestorové a presný čas. Existujú rôzne typy GPS prijímačov podľa využitia, napríklad pre navigácie, geodetické práce alebo poskytovanie maximálne presného času. Výber prijímačov pre potreby tejto práce sa nachádza v kapitole 4.

3.2 Družicové signály

Všetky signály prichádzajúce z družíc sa skladajú z nosnej vlny, diaľkomerného kódu a navigačnej správy. Každý signál je vysielaný na dvoch nosných frekvenciách. Frekvencia L1 (1575.42MHz) je zložená z dvoch diaľkomerných kódov. Presný P-kód, ktorý sa používa najmä pre vojenské účely a môže byť šifrovaný a hrubý/dostupný C/A kód. Tieto signály sa označujú ako signály štandardnej polohovej služby. Každá družica má pridelený svoj vlastný C/A kód. Frekvencia L2 (1227.60 MHz) obsahuje iba P-kód a je používaná pre presnú polohovú službu, ale môžu ju spracovávať iba špeciálne vybavené prijímače [12]. Nosnou vlnou je ešte navyše prenášaný binárny kód, obsahujúci navigačnú správu. Tá obsahuje parametre obežných dráh, na základe ktorých je prijímač schopný vypočítať polohu družice. Ďalšími údajmi v navigačnej správe sú napríklad informácie o stave družice alebo almanach, ktorý obsahuje menej presné údaje o družiciach a umožňuje rýchlejšie vyhľadanie družíc aktuálne viditeľných v danej oblasti.

3.3 Súradnicové systémy

Systém GPS pracuje s dvoma rôznymi súradnicovými systémami. Jeden pre pozemné segmenty (geocentrický systém), druhý pre kozmický segment (heliocentrický systém). Prevody medzi týmito dvoma systémami sú riešené prevádzkovateľom GPS. Prijímač GPS poskytuje polohu v súradniciach WGS-84. Do u nás bežne používaných systémov S-JTSK a S-42 sa musia transformovať dodatočne, pretože prijímače s týmito systémami nevedia pracovať.

3.4 Nadmorská výška

Nadmorská výška je počítaná z pravouhlých súradníc WGS-84, ktorá sa ale označuje ako výška nad elipsoidom. Vzhľadom na to, že Zem nemá pravidelný tvar ale tvar geoidu, musí sa brať do úvahy odchýlka pre dané územie. Odchýlka je buď pre územie známa¹ alebo sa musí dopočítavať pomocou interpolačných vzťahov [12].

3.5 Spôsoby merania

Systém GPS používa k presnému zisteniu polohy družice tri základné princípy. Pomocou kódového merania je možné z prijatého C/A alebo P - kódu zistiť zdanlivú vzdialenosť,

¹Napríklad pre Českú republiku sa táto odchýlka pohybuje od 42 do 47 metrov

ktorá je zaťažená chybou. U civilne dostupného signálu C/A je maximálna dosiahnuteľná presnosť 3-6m. Fázovým meraním je možno dosiahnuť presnosť až na milimetre avšak problémom je nejednoznačnosť merania, počet možných vzdialeností sa rovná počtu celých vlnových dĺžok medzi družicou a prijímačom. Dopplerovské meranie využíva relativistický Dopplerov princíp, ktorý umožňuje na základe meniacej sa vlnovej dĺžky určovať rýchlosť a smer pohybu telesa [1]. Kratšia vlnová dĺžka znamená približovanie, dlhšia vzdiaľovanie. Na základe tohto merania je výpočtami možné zistiť polohu družice, avšak prijímačmi je využívaný prevažne k určovaniu rýchlosti akou sa prijímač pohybuje.

3.6 Čas GPS

Veľmi dôležitou súčasťou systému GPS je neustále udržiavanie presného času. Čas GPS je riadený hlavnými kontrolnými hodinami, na základe ktorých sú synchronizované hodiny jednotlivých družíc. Čas je udávaný v týždňoch a sekundách, ktoré uplynuli od 5.1.1980 24:00:00. Narozdiel od času UTC neobsahuje mechanizmus prestupných sekúnd, ktoré vznikajú dôsledkom nepravidelnej rotácie Zeme a je preto nutné prepočítavať čas GPS na UTC. Údaje na to potrebné sú uvádzané v navigačných správach družíc. Každá družica si udržiava vlastný čas pomocou niekoľkých atómových hodín s minimálnou odchýlkou, ktoré môžu byť v prípade potreby nastavované aby sa udržala odchýlka pod jednu milisekundu.

3.7 Presnosť systému GPS

V závislosti od kvality zariadenia sa presnosť výpočtu polohy môže pohybovať od 100 metrov až po niekoľko centimetrov. Pri civilných prijímačoch je prevádzkovateľom zaručená horizontálna presnosť 100 metrov pri 95 percent prípadoch. U horizontálnej presnosti je to 156 metrov. Bežne dosahovaná presnosť pri turistických a navigačných prijímačoch je v rozmedzí 5 až 15 metrov. Presná polohová služba (PPS) má túto presnosť omnoho väčšiu, ale je využívaná výhradne pre armádne účely. Presnosť systému ovplyvňuje niekoľko faktorov:

- počet viditeľných družíc
- stav družíc
- chyby hodín
- spôsoby merania
- typ prijímača a iné

Prevádzkovateľ systému GPS - ministerstvo obrany USA má možnosť kedykoľvek znížiť presnosť verejne dostupnej služby.

Kapitola 4

Prijímače GPS a prenos dát na server

Užívateľským segmentom systému GPS sa chápu zariadenia na prijímanie a spracovanie signálov vysielaných družicami. Podľa zamerania sa vyskytujú rôzne druhy prijímačov. V tejto kapitole sú rozobraté typy GPS prijímačov vhodných pre prácu s navrhovaným systémom. Hlavnou požiadavkou je pri tom presnosť daného zariadenia a možnosť prenosu získaných dát na webový server, kde sú ďalej spracovávané.

4.1 Špecifické prijímače s GSM

Zariadením spĺňajúcim potreby tejto práce by bol malý programovateľný GPS prijímač s možnosťou internetového pripojenia. Existuje niekoľko zariadení, ktoré sa takémuto ideálnemu zariadeniu približujú. Ide prevažne o GPS lokátory so slotom pre SIM kartu mobilného operátora (obr. 4.1). Určené sú napríklad pre sledovanie vozidla. S takto vybaveným zariadením je možné komunikovať pomocou textových správ a získavať tak jeho aktuálnu polohu. Výhodou je relatívne nízka cena, avšak čas potrebný pre posielanie a spracovávanie SMS správ by veľmi ovplyvnil odozvu systému, ktorý je navrhnutý tak aby pracoval s dátami reálnom čase. Ideálnym riešením by tak bol prijímač, ktorý by cez mobilné pripojenie získal prístup na internet a bolo by možné naprogramovať ho tak, aby v zadaných intervaloch posielal získané GPS dáta na webový server. Na takéto zariadenie som však v čase písania práce nenarazil, bolo teda potrebné hľadať inú alternatívu.



Obrázek 4.1: GPS prijímač UniSafe GPS-GSM Tracker G200



Obrázek 4.2: GPS prijímač Garmin Forerunner 405

4.2 Univerzálne športové prijímače

Druhým použiteľným typom zariadení sú športové GPS prijímače (obr. 4.2). Využívané sú hlavne pri turistike, behu alebo iných športových aktivitách. Umožňujú zaznamenávanie športovej aktivity, uloženie dát do internej pamäte zariadenia a ich neskorší export do špecializovaných softvérov pre vyhodnocovanie. Niektoré navyše poskytujú aj navigačné funkcie. Aj pri tomto type zariadení však chýba možnosť priameho prenosu dát cez internet okamžite po ich získaní. Do vytvoreného systému bola aj napriek tomu pridaná podpora takýchto zariadení hlavne pre ich veľkú obľubu medzi užívateľmi.

4.3 Prijímače zabudované v mobilných zariadeniach

V dnešnej dobe je už takmer u každého mobilného zariadenia samozrejmosťou prítomnosť zabudovaného GPS prijímača. Prístup k nemu je možné získať pomocou aplikácií. Pre účely testovania systému bola využitá práve táto alternatíva. Bola teda vyvinutá jednoduchá mobilná aplikácia pre platformu Android, ktorá zabezpečuje priame prepojenie zariadenia s webovým serverom v reálnom čase. Jej detailnejší popis sa nachádza v kapitolách 6 a 7. Výhodou tohto prístupu je cena a jednoduchosť riešenia. Problémom je nekompatibilita s niektorými typmi mobilných zariadení, čo má za následok znemožnenie práce so systémom pre časť potencionálnych užívateľov. Táto nevýhoda však bola pri tvorbe a testovaní systému zanedbateľná.

Kapitola 5

Uchovávanie GPS a geografických dát v textových formátoch a ich zobrazovanie

Existuje mnoho rôznych textových reprezentácií geografických a GPS dát. Pri vytváraní aplikácie pre túto prácu bolo vybraných niekoľko ktoré spĺňali všetky požiadavky určené pri návrhu a zároveň neobsahovali príliš mnoho zbytočných dát. V nasledujúcich podkapitolách sú rozobraté všetky použité formáty a ku každému z nich je pridané stručné zhrnutie účelu na ktorý bol v tejto práci využitý.

5.1 Formát GeoJson

Formát GeoJson je založený na Json (JavaScript Object Notation), ktorý je v súčasnosti jedným z najrozšírenejších formátov pre výmenu dát na webe. Hlavným prínosom je stručnosť oproti iným druhom zápisov, ako napríklad XML [6]. Zápis v týchto formátoch je platným zápisom v jazyku javascript (viď príklad zápisu kód 5.1), čím sa stávajú veľmi použiteľnými pri tvorbe webových aplikácií. Geografická nadstavba GeoJson umožňuje popis geometrických štruktúr určených súradnicami v súradnicovom systéme.

Tento formát sa v práci uplatnil pre prenos a zobrazenie údajov, ktoré sa často menia a je nutné ich v krátkych intervaloch znovu sťahovať zo serveru. Jeho využitím sa skracuje doba prenosu a vďaka kompatibilite s webovým prostredím aj doba spracovania. Pomocou GeoJson je zobrazovaná aktuálna poloha závodníkov v reálnom čase (viď obr. 5.1). Ku každej štruktúre (bod určený súradnicami) je možné pridať atribúty, ktorými je aplikácii pridaná ďalšia funkcionálna.

5.2 Formát KML

Formát KML (Keyhole Markup Language) je zapísaný v XML notácii (viď kód 5.2). KML bol vyvinutý firmou Keyhole, Inc. pre jej virtuálny glóbus, ktorý je v súčasnosti známy ako Google Earth od firmy Google Inc. V roku 2008 sa svojou verziou 2.2 stal štandardom Open Geospatial Consortium, čo je organizácia zaoberajúca sa vývojom geopriestorových dát a služieb [13].

V tejto práci bol formát KML využitý pre zobrazenie priamok určených geografickými súradnicami. Tieto údaje predstavujú trate jednotlivých závodov. Každý bod sa skladá z

Listing 5.1: Príklad zápisu vo formáte GeoJson

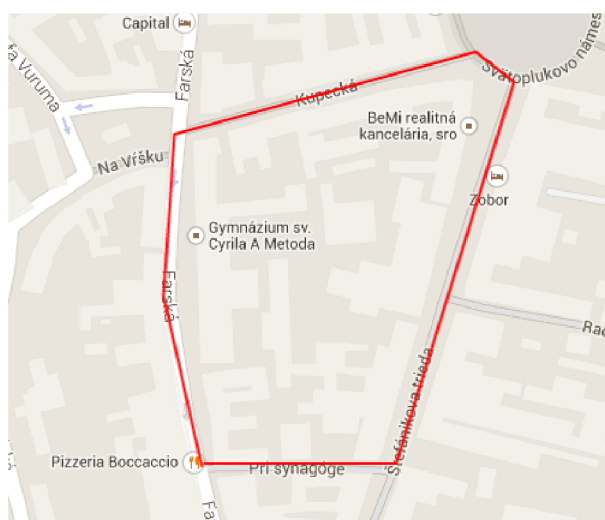
```
{ "type": "FeatureCollection",  
  "features": [  
    { "type": "Feature", "geometry":  
      { "type": "Point", "coordinates": [18.091021701693535, 48.31605033379283] },  
      "properties": { "name": "SZ", "color": "#5e81ff" } },  
    { "type": "Feature", "geometry":  
      { "type": "Point", "coordinates": [18.090517781674862, 48.31554509096332] },  
      "properties": { "name": "PZ", "color": "#f254c5" } }  
  ] }
```



Obrázek 5.1: Príklad zobrazenia údajov získaných z formátu GeoJson

Listing 5.2: Príklad zápisu vo formáte KML

```
<kml xmlns="http://www.opengis.net/kml/2.2"
...
<Placemark>
  <LineString>
    <coordinates>
      18.08559894561797,48.3114359857187 18.085298538208303,48.312320786049526
      18.08538436889678,48.31316275921452 18.08774471282988,48.313590875834564
      18.088045120239553,48.313433900157634
      18.087122440338426,48.311450256813515
      18.085620403290086,48.311450256813515
    </coordinates>
  </LineString>
</Placemark>
</kml>
```



Obrázek 5.2: Príklad zobrazenia údajov získaných z formátu KML

dvojice súradníc. Ich následným pospájaním vzniká lomená čiara, popisujúca trať závodu (viď obr. 5.2).

5.3 Formát TCX

Formát TCX (Training Center XML) bol predstavený v roku 2007 firmou Garmin ako súčasť ich produktu Garmin Training Center [15]. Využíva XML notáciu veľmi podobnú formátu GPX (viď kód 5.3), ktorý je jedným z najrozšírenejších spôsobov pre uchovávanie trás a dát pre potreby navigácie. Ten však umožňuje pracovať iba s jednotlivými GPS bodmi. Do TCX súboru je navyše možné ukladať aj iné údaje získané počas športovej aktivity a prepojiť ich s konkrétnym bodom trasy. Dá sa tak získať napríklad kompletný prehľad o srdcovom tepe, kadencii šľapania alebo spálených kalóriách počas celého trvania aktivity.

Vzhľadom na veľké rozšírenie formátu TCX a možnosti jeho exportu z väčšiny tréningových softvérov bola do aplikácie pridaná jeho podpora. Nahratím tohto typu súboru

Listing 5.3: Príklad zápisu vo formáte TCX

```
<?xml version='1.0' encoding='UTF-8'?>
...
<Activities>
  <Activity Sport="Running">
    <Id>2015-04-18T09:09:27Z</Id>
    ...
    <Track>
      <Trackpoint>
        <Time>2015-04-18T09:09:28Z</Time>
        <Position>
          <LatitudeDegrees>49.1950663459</LatitudeDegrees>
          <LongitudeDegrees>16.6083838977</LongitudeDegrees>
        </Position>
        <AltitudeMeters>214.030639648</AltitudeMeters>
      </Trackpoint>
    </Track>
  </Activity>
  ...
</Activities>
```

cez webové rozhranie je užívateľ automaticky zaradený do výsledkovej listiny bez nutnosti odosielania dát o svojej polohe v reálnom čase.

Kapitola 6

Návrh riešenia

Táto kapitola sa zaoberá návrhom samotnej aplikácie. V kapitole 6.1 sú podrobnejšie rozobrané požiadavky na funkčnosť. Ďalej nasleduje popis prostriedkov zvolených pre implementáciu v kapitole 6.2 a v kapitole 6.3 sa nachádza návrh dátového modelu aplikácie.

6.1 Špecifikácia požiadavkov

Cieľom práce bolo vytvoriť systém, ktorý dokáže zaznamenávať GPS dáta, spracovávať ich a vyhodnocovať. Jeho primárne uplatnenie by malo byť pri organizovaní športových podujatí ako prostriedok pre automatické vyhodnocovanie výsledkov. Potencionálnych užívateľov systému som rozdelil do troch skupín. Prvou skupinou sú organizátori podujatí, ktorý musia byť schopní zakladať závody, prípadne s nimi vykonávať ďalšie úkony. Druhou skupinou sú závodníci prihlasujúci sa k týmto podujatiam a posledným typom užívateľov sú neregistrovaní návštevníci, ktorí majú možnosť prehliadať priebeh závodu počas jeho priebehu, prípadne spätne po jeho ukončení. Všetky možnosti použitia aplikácie sú špecifikované Use case diagramom zobrazeným na obrázku 6.1.

6.2 Voľba implementačných prostriedkov

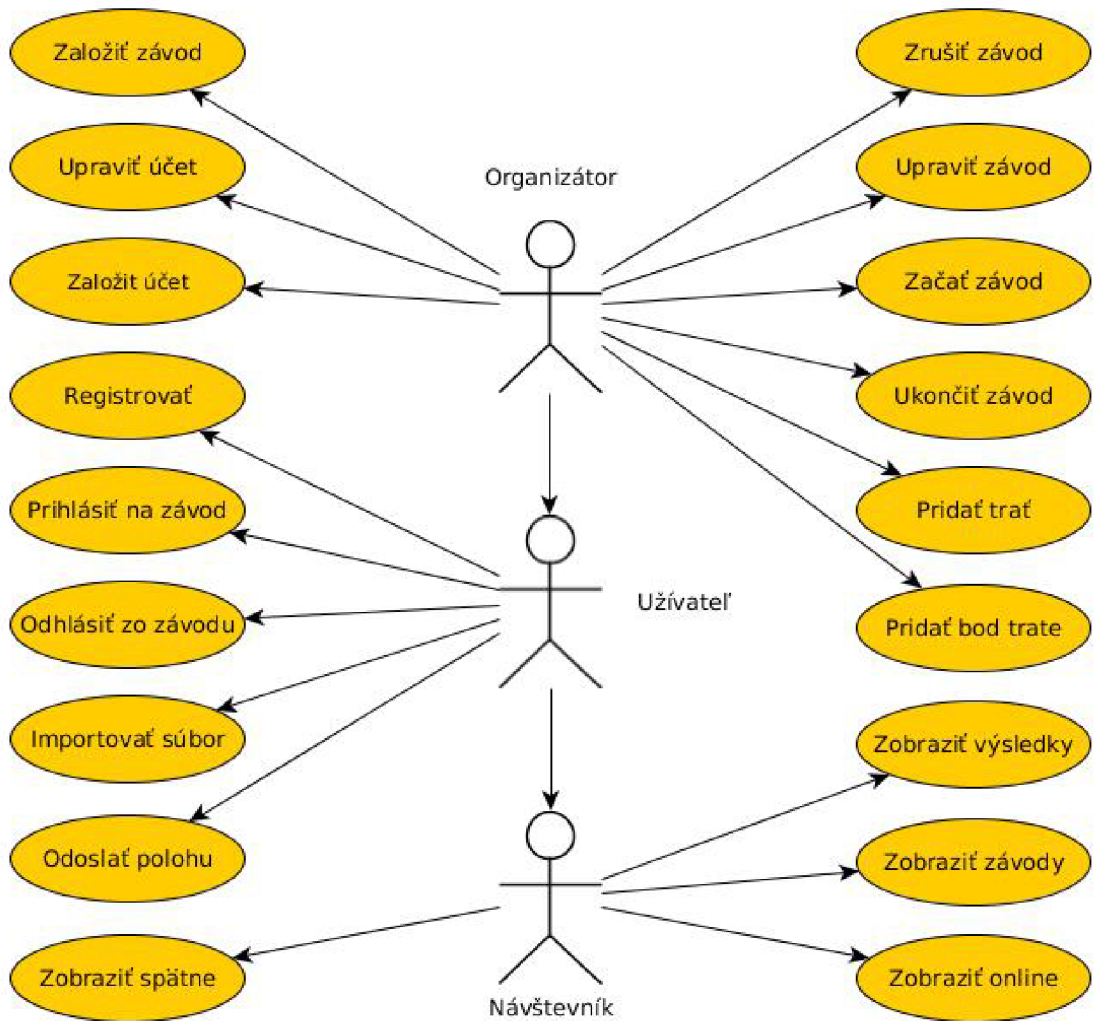
Pre tvorbu aplikácie bolo zvolené webové prostredie. Pre zabezpečenie plnej funkčnosti som použil viacero dostupných technológií, ktorých spojením vznikol kompletný systém zložený zo serveru a klientskej webovej aplikácie.

6.2.1 PHP a MySQL

Časť systému bežiacu na strane serveru som implementoval v jazyku PHP verzie 5.5.12 pod open-source licenciou BSD. Dáta sú ukladané takisto vo voľne dostupnej databáze MySQL verzie 5.5.43. Pre zjednodušenie práce s databázou som využil triedu Tmysql jazyka PHP s licenciou GLPL.

6.2.2 HTML, CSS, Javascript

Zobrazenie klientskej časti aplikácie vo webovom prostredí je zabezpečené pomocou značkovacieho jazyka HTML5, s formátovaním zobrazovaných dát kaskádovými štýlmi jazyka CSS3. Pre vizuálne efekty som použil javascriptovú knižnicu jQuery a jej rozšírenie jQuery



Obrázek 6.1: Use case diagram

UI pre interaktívne užívateľské rozhranie. V jazyku Javascript je tiež implementovaná knižnica OpenLayers3, ktorá sa používa pre zobrazenie geografických dát, pričom ako podklad slúžia Google Maps od spoločnosti Google, Inc. Pre užívateľské rozhranie som zvolil responzívny dizajn, aby sa aplikácia správne zobrazovala na rôzne veľkých obrazovkách. Túto funkciu plní knižnica Bootstrap 3.

Požiadavkou pri tvorbe bola možnosť práce s údajmi v reálnom čase. Preto sa pre asynchrónnu zmenu obsahu webových stránok bez potreby znovunačítavania celej stránky využíva AJAX. Jeho podpora v klientskej časti je zabudovaná v knižnici jQuery, obsluhu na strane servera zabezpečujú skripty v jazyku PHP.

6.2.3 Android

Druhá časť klientskej strany systému je určená k prenosu dát z GPS zariadenia na webový server. Zvolenou platformou je operačný systém Android. Programovací jazyk pre implementáciu aplikácie je Java 7 s využitím popisu užívateľského rozhrania pomocou značkovačieho jazyka XML.

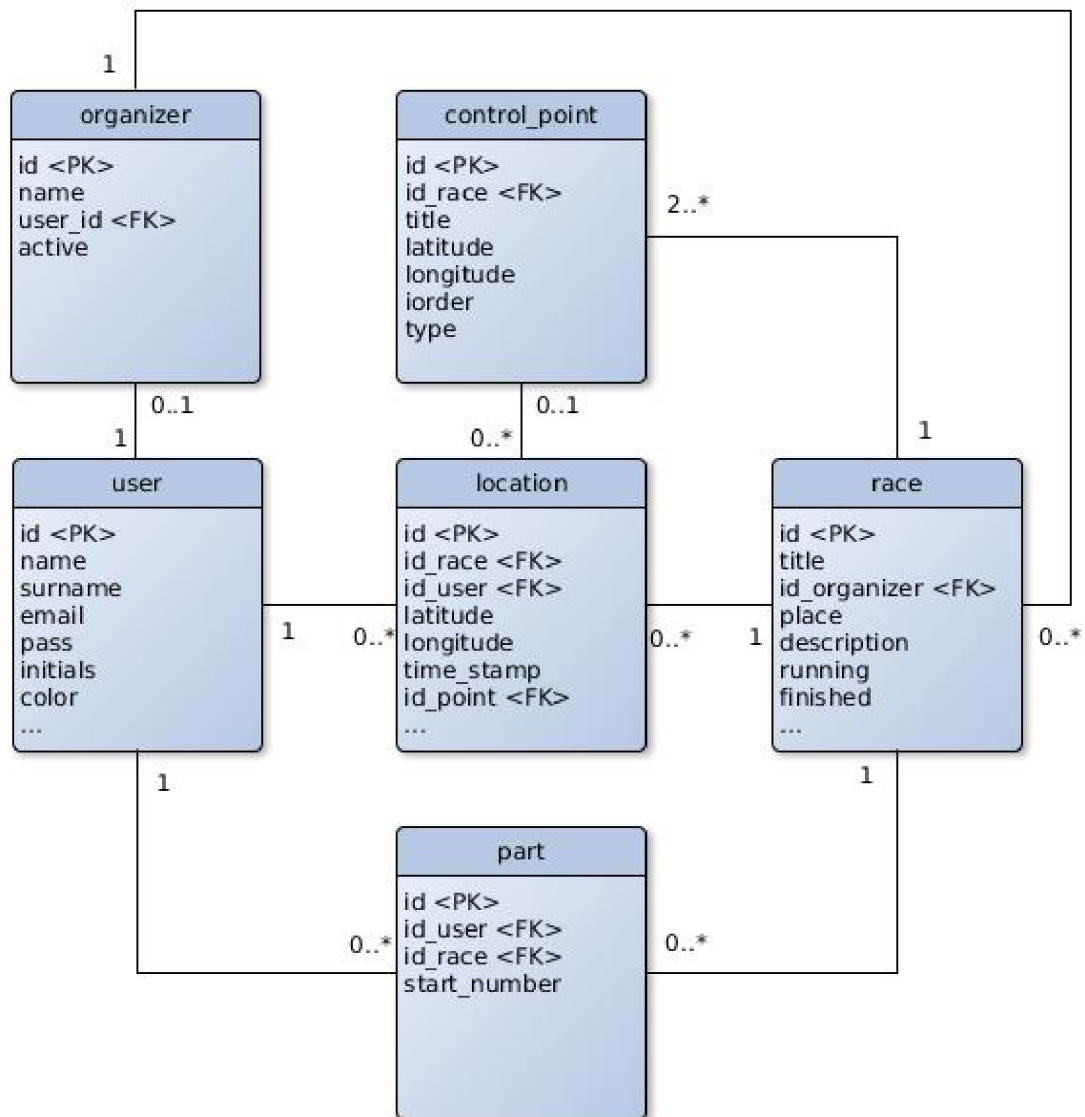
6.3 Dátový model

Dáta systému sú ukladané v relačnej databáze na server a časť geografických dát sa ukladá do textových súborov typu GeoJson a KML v súborovom systéme serveru. Dátový model som navrhol pomocou 6 samostatných tabuliek v databáze. Informácie o užívateľovi sa nachádzajú v tabuľke užívateľ **user**. Každý užívateľ môže byť zároveň organizátorom, detaily ktorého sú v tabuľke organizátor **organizer**. Tabuľka závod **race** popisuje jednotlivé založené závody, pričom každý závod ma jednoznačne určeného organizátora. K závodu sa môže prihlásiť neobmedzený počet užívateľov a preto, aby sa predišlo vzťahu M:N bola pridaná pomocná tabuľka účasť **part**. V tabuľke poloha **location** sa ukladajú informácie o polohe a čase užívateľa v danom závode. Poloha je určená v geografických súradniciach, čas pomocou časového razítka, aby bolo možné v databáze vyhľadávať na základe tohto stĺpca. Poslednou tabuľkou je kontrolný bod **control_point** ktorá znázorňuje vytvorené miesta v závode, na ktorých sa vyhodnocuje priebežné poradie účastníkov. V správne prebehnutom závode by u každého závodníka by mal byť kontrolný bod priradený k práve jednému bodu polohy. Dátový model je detailnejšie popísaný E-R diagramom na obrázku 6.2

Zvyšná časť dát je uložená je na server v textových súboroch. Jedná sa o dráhy tratí závodov a súradníc kontrolných bodov. Formáty týchto súborov boli detailne popísané v kapitole 5.

6.4 Zhrnutie návrhu

Pri návrhu som kládol dôraz na čo najväčšiu jednoduchosť dátového modelu, pri súčasnom zachovaní všetkých zadaných podmienok. Taktiež som sa snažil vytvoriť čo najprehľadnejšie grafické rozhranie pre užívateľov, ktorí s ním budú pracovať. Hlavnou časťou rozhrania je interaktívny mapový podklad, kde som pre zabezpečenie funkcionality využil rôzne voľne dostupné knižnice.



Obrázek 6.2: Entity relationship diagram

Kapitola 7

Popis funkčnosti aplikácie a implementácia

Nasledujúca kapitola sa zaoberá štruktúrou vytvorenej aplikácie. Proces implementácie som rozdelil do logických celkov. Každému celku je venovaná samostatná kapitola, kde je bližšie popísaná jeho funkčnosť. Na miestach kde je to potrebné je podrobnejšie vysvetlený popis riešenia danej funkcionality.

7.1 Štruktúra webovej aplikácie

Pri implementácii aplikácie som sa snažil minimalizovať miesta s duplicitným zdrojovým kódom. Každé načítanie stránky, pokiaľ sa jedná o zobrazenie vo webovom prehliadači začína v súbore `index.php` v koreňovom adresári projektu. Tu sa na základe nastavení v konfiguračnom súbore `config.php` vytvorí prepojenie na databázu a načítajú sa všetky pomocné funkcie z `functions.php`. `index.php` obsahuje HTML hlavičku, navigáciu a päť, ktoré sú naprieč celou aplikáciou rovnaké. Súčasťou tohto súboru je vloženie kaskádových štýlov z adresáru `/css` a načítanie javascriptových súborov z adresáru `/js`. Na základe GET parametrov získaných zo zadanej url sa do súboru vkladajú ďalšie zdrojové kódy. Tieto php skripty sú umiestnené v adresári `/codes`, pričom každý z nich sa stará o zobrazenie istej logickej časti systému. Všetky akcie vykonávané pri interakcii so systémom sú umiestnené v súbore `actions.php` a volané pomocou GET alebo POST požiadavkov. Zjednodušene je tento skript jedným príkazom `switch`, kde každá jeho vetva predstavuje jednu konkrétnu akciu. Súčasťou štruktúry serveru sú adresáre `/images` pre obrázky a `/data` pre ukladanie súborov vytvorených alebo nahratých užívateľmi.

7.2 Registrácia a prihlasovanie užívateľov

Prístup k aplikácii je možný bez nutnosti prihlásenia. Avšak rozsah akcií neprihláseného užívateľa je obmedzený na prezeranie dát bez možnosti pridávania a ovplyvňovania obsahu na stránke. K sprístupneniu všetkých funkcií je potrebná registrácia a následné prihlásenie. Registračný formulár (viď obr. 7.1) sa nachádza v súbore `register.php`. Užívateľ vyplní údaje, po odoslaní formuláru metódou POST prebehne kontrola na duplicitu vloženého emailu a vytvorí sa nový záznam v tabuľke `user`. Vložené heslo je funkciou `md5()` zakódované, z dôvodu ochrany osobných údajov. Po úspešnej registrácii prebehne automatické prihlásenie

Register

The image shows a registration form titled "Register". It consists of four input fields stacked vertically: "E-mail", "Name", "Surname", and "Password". Below these fields is a prominent red button labeled "Register". Underneath the button is a text link labeled "Login".

Obrázek 7.1: Formulár pre registráciu užívateľa

a presmerovanie na stránku s užívateľským profilom. Pri prihlasovaní užívateľa je vyžadovaný registračný email a heslo. Ak vložené údaje odpovedajú dátam v databáze prebehne prihlásenie. Informácie o prihlásenom užívateľovi sa uchovávajú v premennej `$_SESSION` a vymažú sa až po vypršaní času určeného prehliadačom alebo manuálnym odhlásením. Manuálne odhlásenie sa vykonáva cez hlavnú navigačnú lištu aplikácie (záložka Profile - Logout). V prípade zabudnutého hesla aplikácia umožňuje automaticky vygenerovať nové heslo a odoslať ho užívateľovi na emailovú adresu. Predpokladom je správne nastavený emailový server.

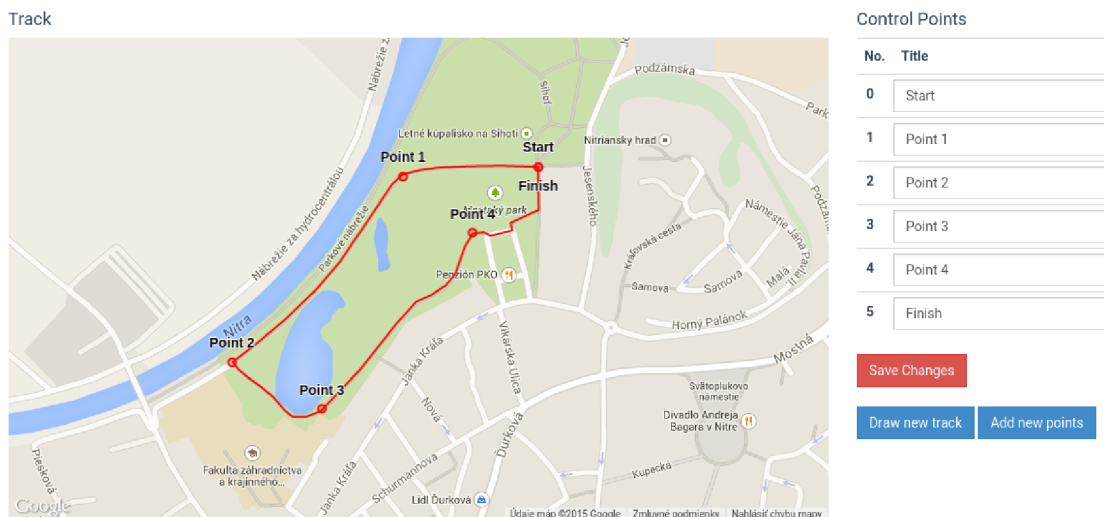
7.3 Správa profilu

Každý prihlásený užívateľ má v rámci aplikácie možnosť úpravy svojich osobných údajov. Zobrazenie profilu je realizované súborom `profile.php`. Zmenou položiek v textových poliach a odoslaním formuláru sa vykoná aktualizácia údajov v databáze SQL príkazom `UPDATE`. Stránka tiež obsahuje všetky závody prihláseného užívateľa spolu so zobrazením ich základných parametrov. Súčasťou správy profilu je vytvorenie organizátorského účtu pre možnosti zakladania nových závodov. Vytvorenie sa vykoná cez hlavnú navigáciu (Profile - Organizer). Jedinou podmienkou je vloženie názvu organizátora ktorý je neskôr editovateľný. Informácie o organizátorovi sa zobrazujú skriptom `organizer.php`. Okrem základných údajov obsahuje stránka aj výpis jeho spravovaných závodov.

7.4 Založenie a správa závodu

Každý užívateľ s organizátorským účtom je oprávnený zakladať nové závody. Táto akcia prebieha na stránke s údajmi organizátora tlačidlom Create Race a vyplnením názvu, typu a dátumu. Ďalším krokom je uloženie ostatných informácií o závode (miesto, dĺžka, popis). Vzhľadom na to, že sa jedná primárne o systém pre bežecké závody, alebo závody podobného typu je potrebné definovať trasu (viď obr. 7.2).

Pre definovanie trasy závodu sa využívajú funkcie knižnice Openlayers3 (OL). Kliknutím na tlačidlo Draw New Track sa zavolá funkcia `map.addInteraction()` nad mapou, čím sa aktivuje kreslenie do mapy. Každým kliknutím do mapy sa pridá bod trasy a medzi dvoma poslednými bodmi sa vykreslí čiara. Pre ukončenie kreslenia postačuje opätovné kliknutie na posledný vykreslený bod, čím sa interakcia s mapou ukončí. Tlačidlom Save



Obrázek 7.2: Rozhranie pre kreslenie trasy závodu do mapy.

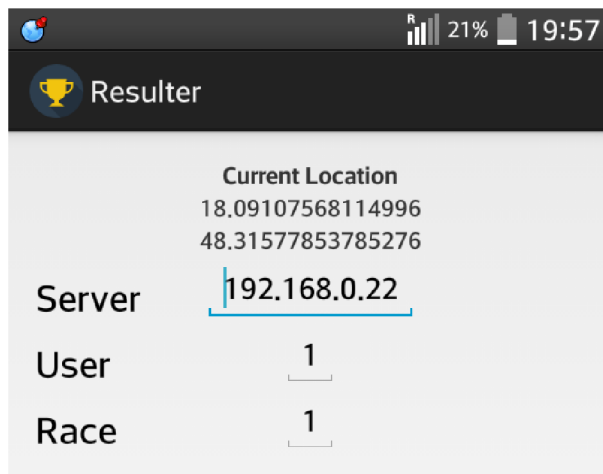
sa nová trať uloží. Pomocou OL sa nakreslená trasa prevedie do textového formátu KML a ako položka formuláru sa odošle na server. PHP skript na strane servera prijatý reťazec uloží do súboru s názvom `trackid.kml`, kde `id` je jednoznačný identifikátor závodu. Trasu závodu je povolené zmeniť, prepíše sa tak pôvodný súbor. Pre správny priebeh závodu musí mať každá trať určené kontrolné body, slúžiace pre automatické generovanie výsledkov na medzičasoch a v cieľi. Minimálny počet bodov je 2 (štart a cieľ). Princíp pridávania bodov je podobný kresleniu trate. Kliknutím na tlačidlo `Add new points` sa aktivuje kreslenie pomocou OL. Body sa vložia do mapy a formulár sa odošle. Kontrolné body sú uložené vo textovom formáte GeoJson a zároveň ako položka v tabuľke `control_points` v databáze pre urýchlenie vyhodnocovania výsledkov. K bodom môže organizátor pridať popis, ktorý sa zobrazí v mape pri danom bode. V detaile závodu sa nachádza tlačidlo `Start`. V okamihu keď je závod pripravený, organizátor ho použije pre zahájenie závodu, do databázy sa uloží čas štartu a začne sa zaznamenávať jeho prebeh až pokiaľ organizátor závod rovnakým spôsobom neukončí.

7.5 Vytvorenie a zrušenie účasti na závode

Aby užívateľ mohol byť zaradený do závodu musí sa k nemu prihlásiť. V súbore `race.php` (záložka `Events`) je tabuľkové zobrazenie závodov ktoré ešte neprebehli. Ak je užívateľ prihlásený v systéme, zobrazí sa mu možnosť registrovať sa na závod. Potvrdením účasti sa vytvára záznam v tabuľke `part`. Účasť je jednoznačne určená identifikátorom závodu a užívateľa. Ku každej účasti je pridelené štartové číslo. Štartové číslo sa prideluje automaticky, počínajúc od 1. Účastníkovi sa závod zobrazí v stránke s profilom. Pokiaľ závod ešte neprebehol závodník má právo svoju účasť zrušiť tlačidlom `Unregister`.

7.6 Odoslanie údajov o polohe

V práci som využil platformu Android pre odosielanie GPS dát na webový server v reálnom čase. K testovacím účelom som vytvoril jednoduchú aplikáciu s jednou obrazovkou (viď obr.



Obrázek 7.3: Rozhranie mobilnej aplikácie pre operačný systém Android

7.3). Po jej spustení sa užívateľovi zobrazia predvyplnené polia. Do pola Server sa zadáva IP adresa alebo doména serveru na ktorom systém beží. User je identifikátor užívateľa a Race identifikátor závodu. Závodník musí údaje správne vyplniť, pomôcka sa nachádza vo webovom rozhraní v detaile závodu.

Po zapnutí lokalizačnej služby aplikácia začína odosielať aktuálnu polohu zariadenia ako požiadavok POST skriptu `actions_android.php`. Údaje o polohe sa odosielajú pri každej zmene polohy. Funkciou `requestLocationUpdates(LocationManager.GPS_PROVIDER, 1000, 2, this)` je nastavený minimálny interval medzi dvoma zaznamenanými zmenami polohy na 1000ms, pričom vzdialenosť medzi geografickými bodmi musí byť minimálne 2 metre. Predchádza sa tým zbytočnému odosielaniu duplicitných údajov na server. Vďaka mechanizmu `partial wakelock`, ktorý zabraňuje zariadeniu prejsť do režimu spánku dokáže aplikácia zbierať a odosielať údaje aj v prípade vypnutej obrazovky alebo minimalizovanej aplikácie [18].

7.7 Prijímanie a spracovávanie údajov o polohe

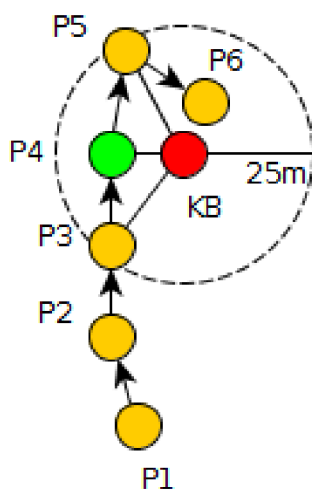
Dáta prijaté z mobilnej aplikácie sú na serveri spracovávané skriptom `actions_android.php`. Každému bodu je priradený niektorý z kontrolných bodov závodu. Body sa priradujú podľa poradia určeného organizátorom. Ak teda závodník vynechá niektorý z kontrolných bodov kontrola na ostatných bodoch zlyhá a jeho poradie nebude správne určené. Pre výpočet vzdialenosti medzi bodom polohy a kontrolou som zvolil Haversine Metódu. Táto metóda aproximuje zemský povrch povrchom gule a pre výpočet používa vzťah medzi vzdialenosťami troch bodov na povrchu gule a uhlom, ktorý zvierajú (viď vzorec 7.1) [9].

U niektorých bodov polohy prebieha k potvrdeniu priblíženia ku kontrolnému bodu za predpokladu splnenia podmienok:

1. Vzdialenosť bodu určujúceho polohu a kontrolného bodu je menšia ako hraničná vzdialenosť. Východzia hraničná hodnota je 25 metrov ale v súbore `config.php` je možné ju zmeniť. Na obrázku 7.4 splňujú body P3-P6.
2. Aktuálna vzdialenosť od kontrolného bodu je väčšia ako vzdialenosť predošlej polohy a kontrolného bodu. Na obrázku 7.4 nastáva v bode P5, t.j. bod P4 sa vyhodnotí ako

$$\begin{aligned}
 a &= \sin^2(\Delta\varphi/2) + \cos\varphi_1 * \cos\varphi_2 * \sin^2(\Delta\Lambda/2) & (7.1) \\
 c &= 2 * \operatorname{atan2}(\sqrt{a}, \sqrt{1-a}) \\
 d &= R * c
 \end{aligned}$$

kde φ je geografická šírka, Λ je geografická dĺžka, R je polomer Zeme (6,371km) a d vzdialenosť medzi dvoma bodmi



Obrázek 7.4: Príklad zachytenia užívateľa na kontrolnom bode

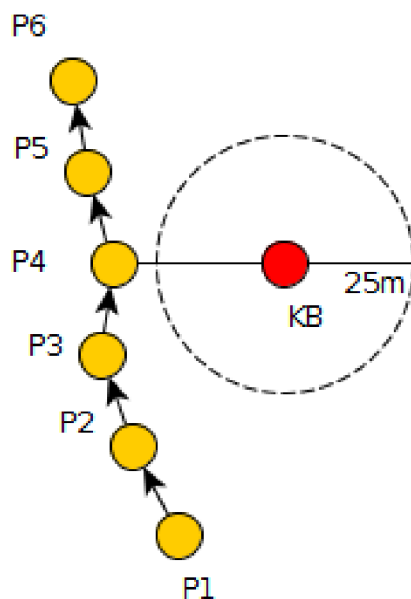
najbližší bod a potvrdí sa jeho priblíženie. Bod P6 aj napriek tomu, že jeho vzdialenosť od kontrolného bodu je menšia ako pri P4 sa ignoruje, pretože obidve podmienky už boli splnené.

Na obrázku 7.5 je načrtnutý príklad prechodu okolo kontrolného bodu bez potvrdenia polohy z dôvodu nesplnenia podmienky 1. V tomto prípade je znehodnotený záznam závodníka, preto je dôležité správne zvoliť hraničný polomer s ohľadom na parametre trate.

V poslednom kroku sa záznam o polohe uloží do tabuľky `location` v databáze. Súčasťou záznamu sú okrem identifikátorov užívateľa a závodu geografické súradnice určené zemepisnou šírkou a výškou, čas prijatia údajov s presnosťou na sekundy a vzdialenosť od nasledujúceho kontrolného bodu.

7.8 Import súboru TCX

Ako alternatívu priameho odosielania GPS údajov som do aplikácie zabudoval podporu súborov TCX, ktoré sú jedným z možných výstupov športových GPS prijímačov. Import prebieha po ukončení závodu. V súbore `race.php` sa nachádza formulár so vstupom typu súbor. Prihlásený užívateľ vyberie s lokálneho súborového systému zodpovedajúci súbor. Odošle sa



Obrázek 7.5: Príklad neúspešného prechodu okolo kontrolného bodu

požiadavok na skript `actions.php`, kde sa spracováva funkciou `simplexmlloadfile()`. V získanom objekte sa nájdu všetky body určené polohou zariadenia a obdobným postupom aký bol vysvetlený v predchádzajúcej kapitole sa vytvoria záznamy v databáze. Okrem toho sa založí účasť na závode, čím sa zaručí zobrazenie vo výsledkovej listine.

7.9 Sledovanie priebehu závodu

Jednou z hlavných funkcií aplikácie je zobrazovanie priebehu závodu v reálnom čase. Predpokladá sa, že účastníci závodu majú vo svojom mobilnom zariadení správne nastavenú aplikáciu popísanú v kapitole 7.6 a zariadenie má povolené určovanie polohy na základe GPS. Zobrazovanie polohy online zabezpečuje skript `live.php`. Vo výbere sa zobrazia závody ktoré majú v databáze nastavenú hodnotu `running` na 1 a zároveň `finished` na 0. Podstatnú časť užívateľského rozhrania tvorí mapa (viď obr. 7.6). Počiatočné nastavenie mapy je pomocou knižnice `openlayers` nastavené na miesto, ktoré je stredom trate závodu (počíta sa ako priemer súradníc všetkých bodov trasy). Celý proces zobrazovania v užívateľskom rozhraní rieši skript `maps-live.js`. Funkciou `setInterval(function, 1000)` sa nastaví opakujúce sa volanie funkcie `showMarkers()` a vykonanie kódu zodpovedného za zobrazovanie aktuálnych výsledkov každých 1000ms.

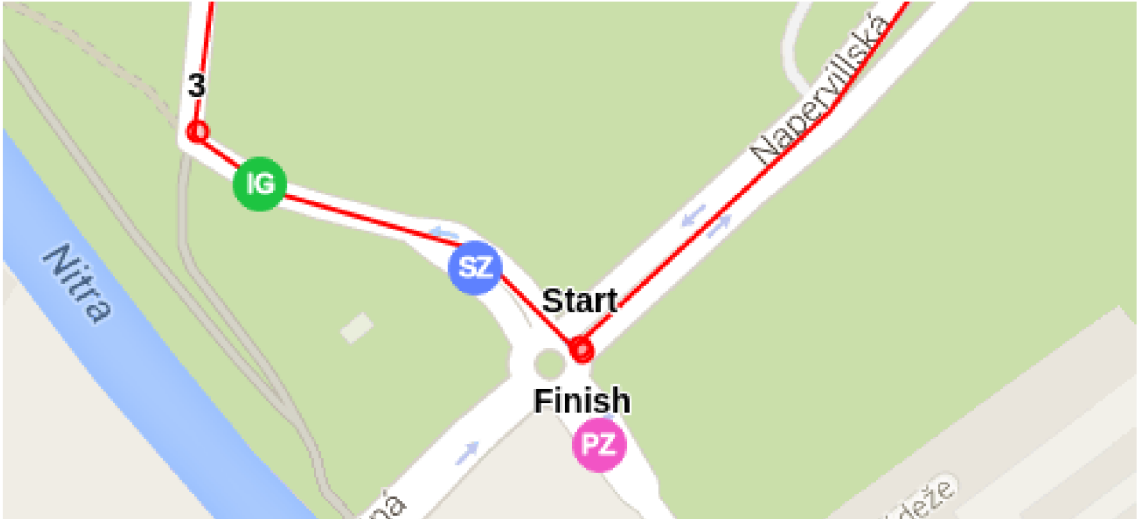
Pri každom volaní funkcie `showMarkers()` sa mapa čiastočne prekresluje. Vytvorí sa nová vektorová vrstva ako objekt triedy `ol.layer.Vector` (viď 7.1).

Pred jej vytvorením sa pôvodná vrstva priradí do pomocnej premennej aby bolo možné jej zmazanie. Ako formát zdroja vrstvy je zvolený `GeoJson`. Zdrojovým súborom vkladaným do vrstvy je `load_location.php`. Funkciou `createPointStyleFunction()` sa vrstve nastaví štýl zobrazenia.

Každý závodník je reprezentovaný kružnicou vyplnenou unikátnou farbou, pridelenou pri registrácii a inicálami, vytvorenými z mena a priezviska. Novovytvorená vrstva sa vkladá

Listing 7.1: Vytvorenie vektorovej vrstvy

```
vectorPoints = new ol.layer.Vector({
  source: new ol.source.GeoJSON({
    projection: 'EPSG:3857',
    url: 'codes/load_location.php'
  }),
  style: createPointStyleFunction()
});
```



Obrázek 7.6: Časť grafického užívateľského rozhranie pre online zobrazenie

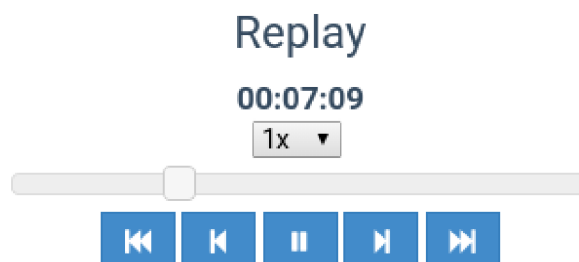
do mapy a následne sa pôvodná vrstva zmaže. Výstupom PHP skriptu `load_location.php` je zápis vo formáte GeoJson. Je to teda kolekcia štruktúr, kde každá štruktúra predstavuje informácie o jednom závodníkovi. Z tabuľky `location` v databáze sa príkazom `SELECT` vyberie posledný záznam o jeho polohe a vloží sa do štruktúry, okrem súradníc sa posielajú ďalšie informácie o závodníkovi (farba, iniciály).

V jednom webovom prehliadači je počet súčasných sledovaní závodu obmedzený na 1. ¹

7.10 Spätné prehrávanie závodu

Po ukončení závodu sa vo výpise výsledkov objaví odkaz na spätné prehrávanie priebehu závodu (tlačidlo `Replay`). Zobrazovanie mapy je implementačne identické s priamym prehrávaním závodu. Jedinou zmenou v užívateľskom rozhraní je pribudnutie ovládacích prvkov prehrávania do bočného panelu (viď obr. 7.7). Tie umožňujú návštevníkovi stránky zobraziť stav závodu v každom okamihu od jeho štartu až po ukončenie. Ovládacie prvky sa skladajú z hodín zobrazujúcich čas od štartu, posuvníku pre rýchlu navigáciu po časovej osi, výberového menu pre zmenu rýchlosti a tlačidlá pre spustenie, zastavenie a krokovanie prehrávania.

¹Tento kompromis bol zavedený z dôvodu potreby ukladania niektorých údajov o sledovanom závode do premennej `$_SESSION`



Obrázek 7.7: Ovládacie prvky prehrávania záznamu

Aktualizácia mapy sa vykoná vždy pri zmene času od štartu princípom aký bol popísaný v kapitole 7.9. Zmenu času je možné dosiahnuť rôznymi spôsobmi, pri všetkých boli použité funkcie knižnice JQuery. Stopky sa spúšťajú stlačením tlačidla Play, pričom každú sekundu sa čas zmení o hodnotu zvolenú z menu rýchlosti prehrávania. Súčasne dochádza k zmene polohy posuvníku. Pri manuálnom pohybe s posuvníkom sa okamžite posiela požiadavok na server a poloha sa aktualizuje iba s minimálnym oneskorením. Posledným spôsobom zmeny času je krokovanie pomocou tlačidiel, čím dochádza k jeho posunutiu o 1 alebo 5 sekúnd. Narozdiel od živého zobrazenia môžu byť do spätného prehrávania vložené aj účastníci, ktorí pri závode nepoužívali mobilnú aplikáciu. V prípade, že závod absolvovali z GPS zariadením umožňujúcim export TCX súboru, po jeho importe do systému budú automaticky zobrazovaní na mape.

7.11 Vyhodnocovanie výsledkov

Posledným celkom systému, v ktorom sa pracuje s GPS údajmi je vyhodnocovanie celkových a čiastkových výsledkov. Základ vyhodnocovania je položený už vkladáním záznamov o polohe do databázy, kedy sa niektoré body párujú s kontrolnými bodmi závodu. Správne vytvoreným príkazom `SELECT` jazyka SQL potom možno získať poradie na ktoromkoľvek kontrolnom bode. Celkové výsledky závodu sú zobrazené ako tabuľka v detaile závodu (prístup cez tlačidlo Results). Priebežné výsledky sú súčasťou rozhrania pre online sledovanie závodu. V pravom paneli vedľa mapy sa nachádza zoznam kontrolných bodov (viď obr. 7.8). Po otvorení záložky s konkrétnym bodom sa zobrazí výpis účastníkov závodu, ktorí boli na danom bode zaznamenaní spolu s ich priebežným časom. Špeciálnym typom kontrolného bodu je cieľ podujatia. Po otvorení výpisu kontrolného bodu sú dostupné profily účastníkov. V každom profile sa znova zobrazujú informácie o absolvovaných medzičasoch (čas, poradie).

Panel s priebežnými výsledkami sa aktualizuje pravidelne každú sekundu. Pre obnovenie obsahu som využil funkciu knižnice JQuery `load()`, ktorá do zvoleného elementu vloží obsah vygenerovaný php skriptom. Pre generovanie výsledkov v aktuálnom čase je to skript `load_results.php` a pre profil užívateľa skript `load_athlete.php`. Na stránke s prehrávaním závodu sa výsledky zobrazujú podobným spôsobom ale narozdiel od zobrazenia v reálnom čase nieje nutné aktualizovať obsah panelu, pretože sa po otvorení stránky načítajú výsledky v čase ukončenia podujatia. Práca s panelom a jeho aktualizácia je implementovaná v súbore `maps-live.js`.

Start List		2
Simon Zacik		Times:
		Point 1: 00:05:53 (2)
		Point 2: 00:10:36 (1)
		Point 3: 00:18:26 (2)
		Point 4: 00:22:39 (2)
		Finish time: 00:24:09 (1)
Results		
Point 1		00:05:51
Point 2		00:10:36
1.	<input type="checkbox"/> 2 Simon Zacik	00:10:36
2.	<input type="checkbox"/> 1 Michal Hagara	+ 00:00:06
Point 3		00:18:20
Point 4		00:22:38
Finish		00:24:09

Obrázek 7.8: Panel rozhranie pre sledovanie priebežných časov závodníkov

7.12 Inštalácia systému

Celý systém je možné spustiť na vlastnom webovom serveri. Zdrojové súbory sa nachádzajú v adresári `/src` na priloženom CD. Po ich skopírovaní je potrebné v súbore `config.php` nastaviť pripojenie k MySQL databáze a zvoliť polomer zachytávania polohy okolo kontrolných bodov. Pre vytvorenie databázových tabuliek treba spustiť skript `install.php` a nastaviť adresáru `/data` práva na zápis.

Kapitola 8

Testovanie

Aplikáciu som začal testovať takmer okamžite po začiatku implementácie. Pomohlo mi to k odladeniu niektorých chýb, ktoré som si počas návrhu neuvedomil a prejavili sa až po nejakom čase. Testovanie teda prebiehalo na etapy, pričom vždy som sa snažil testovať niektorý funkčný celok aplikácie tak aby som plne overil jeho spoľahlivosť. Po dokončení práce som ešte testoval aplikáciu s dobrovoľníkmi, ktorí mali s podobnými aplikáciami skúsenosti na rôznej úrovni.

8.1 Testovanie prenosu údajov na server

Aby som mohol overiť funkčnosť kompletného systému potreboval som pracovať s reálnymi, prípadne pseudoreálnymi datami. Preto som zvolil ako prvé testovanie mobilnej aplikácie, pomocou ktorej sú údaje získané zo systému GPS prenášané na server. Aplikáciu som testoval na štyroch rôznych zariadeniach s operačným systémom Android:

- LG L90 (Android 4.4.2)
- Sony Xperia Miro (Android 4.0.4)
- Sony Ericsson WT19i (Android 4.0.4)
- Lenovo A319 (Android 4.4)

S každým o zariadením som absolvoval test aplikácie v reálnych podmienkach, keď som cez dátové pripojenie odosielal údaje o aktuálnej polohe na testovací server. Následne som prijaté dáta analyzoval vo webovom rozhraní s mapou. V prevažnej väčšine prípadov sa údaje zhodovali s miestami, v ktorých som sa pohyboval. Odchýlky boli zaznamenané hlavne po štarte aplikácie, čo mohlo byť spôsobené oneskoreným fixovaním polohy, ktoré je u GPS príímačov pomerne bežné. Aplikáciu som testoval aktívnu, minimalizovanú, ale aj s vypnutou obrazovkou zariadenia, kedy som overoval funkčnosť mechanizmu wakelock.

8.2 Testovanie zobrazovania priebehu závodu

Hlavnou funkciou vytvoreného systému je zobrazenie aktuálneho stavu závodu a vyhodnocovanie priebežných výsledkov. Pre zefektívnenie procesu implementácie a testovania použil som voľne dostupnú android aplikáciu Fake GPS¹ stiahnuteľnú z obchodu Google Play. Táto

¹<https://play.google.com/store/apps/details?id=com.fakegps.mock>

umožňuje simulovať polohu zariadenia pohybom po mape. Pre túto etapu testovania som si vytvoril niekoľko virtuálnych závodov, do ktorých som prihlásil dvoch až troch súťažiacich. Pre každého z nich som pomocou samostatného mobilného zariadenia simuloval jeho polohu tak aby to približne odpovedalo priebehu závodu². Sledoval som pritom zobrazovanie na mape v implementovanom užívateľskom rozhraní. Taktiež som sa sústredil na overenie mechanizmu vyhodnocovania výsledkov. Pri tomto bode som riešil otázku určenia polomeru oblasti okolo kontrolného bodu v ktorej je možné potvrdenie polohy. Po viacerých testoch (niektoré v teréne) som zvolil hodnotu na 25 metrov, ktorá postačuje pri slabšej presnosti GPS signálu a zároveň výrazne neobmedzí presnosť vyhodnotenia výsledkov. Navyše som ponechal možnosť jej jednoduchšej zmeny v konfiguračnom súbore.

8.3 Testovanie importu údajov z GPS zariadení

Pre dodatočné zaradenie do závodu systém podporuje možnosti importu údajov vo formáte TCX z GPS zariadení. Túto funkčnosť som testoval pomocou športových GPS zariadení a s mobilnými zariadeniami s dvoma rôznymi aplikáciami.

- Garmin Forerunner 405
- Garmin Forerunner 610
- LG L90 s aplikáciou Strava
- LG L90 s aplikáciou Endomondo

Táto časť testovania opäť prebiehala v teréne. S GPS zariadením som absolvoval Brnenský polmaratón³. Záznam som importoval do systému a pomocou funkcie spätného sledovania som analyzoval funkčnosť aplikácie⁴. Priebeh závodu presne odpovedal reálnemu behu a vyhodnotený čas v systéme sa odlišoval o cca 30 sekúnd oproti času nameranému organizátormi, čo ale bolo spôsobené nepresným uložením kontrolných bodov. Po ich presnom zameraní som dosiahol presnosť v jednotkách sekúnd, preto som test zhodnotil ako úspešný.

V ďalšom kroku som testoval import z rôznych zariadení, do nového testovacieho závodu som prihlásil niekoľkých užívateľov. Vytvoril som trať so vzdialenosťou približne 1,25 km a s niekoľkými kontrolnými bodmi. Trať som postupne absolvoval s každým zariadením, rôznou rýchlosťou behu a údaje som nainportoval pod jednotlivými užívateľskými kontami. Aby si záznamy odpovedali musel som ručne pomocou SQL príkazov v databáze upraviť čas u zaznamenananej polohy podľa prvého záznamu. Takto sa mi podarilo vytvoriť fiktívny závod s viacerými súťažiacimi a mohol som tak porovnať importované údaje. Zároveň mi to umožnilo overiť funkčnosť grafického rozhrania s mapou pri zobrazovaní viacerých súťažiacich súčasne⁵. Na základe výsledkov som robil drobné úpravy aby sledovanie priebehu bolo čo najviac plynulé.

8.4 Testovanie užívateľského rozhrania

V poslednej fáze po dokončení systému som otestoval intuitívnosť grafického užívateľského rozhrania. Štyrom dobrovoľníkom som zadal sadu úloh, ktoré mali pri práci so systémom a

²Testovací závod sa nachádza na adrese www.kbnr.sk/resulter/index.php?page=replay&id=7

³<http://www.pulmaratonbrno.cz/>

⁴Testovací závod sa nachádza na adrese www.kbnr.sk/resulter/index.php?page=replay&id=5

⁵Testovací závod sa nachádza na adrese www.kbnr.sk/resulter/index.php?page=replay&id=8

mobilnou aplikáciou vykonať (registrácia a prihlásenie, založenie závodu, nakreslenie trate a pridanie kontrolných bodov, prihlásenie k závodu, prehrávanie priebehu závodu, orientácia v menu, inštalácia a nastavenie mobilnej aplikácie pre odosielanie údajov) a pozoroval som ich prácu.

Podľa predpokladov užívateľ, ktorý mal veľmi malé skúsenosti s prácou s PC mal s úlohami problém. Užívateľia ktorí vedeli s PC pracovať ale nikdy podobný systém nepoužívali mali problém s orientáciou v systéme, ale úlohy po čase zvládli. Dobrovoľník, ktorý sa už s podobným systémom stretol vykonával všetky úlohy takmer bez zaváhania. Následne bol každému dobrovoľníkovi podrobnejšie popísaný spôsob fungovania systému a úlohy mali vykonať znova. Najväčší pokrok som zaznamenal u druhej skupiny, ktorá si po vysvetlení rýchlo osvojila princípy úloh. Užívateľ s minimálnymi skúsenosťami s prácou s PC mal naopak takmer rovnaké problémy ako v prvom kole úloh. Na základe spätnej väzby od dobrovoľníkov som zistil, že intuitívnosť aplikácie je dostatočná na to, aby bola mohla byť reálne využívaná v praxi.

Kapitola 9

Možnosti pokračovania projektu

Pokračovanie tejto práce by sa mohlo uberať viacerými smermi. Na ich realizáciu už v priebehu riešenia práce neostal čas ale určite by bolo zaujímavé sa k nim vrátiť aj mimo tohto projektu, prípadne ako súčasť inej práce.

Aby sa systém mohol uplatniť v praxi ako spôsob reálneho vyhodnocovania výsledkov bolo by potrebné navrhnuť vhodný GPS prijímač. Z vlastnej skúsenosti viem, že takéto zariadenie by muselo byť kompaktné aby závodníkovi nijako neprekážalo v pohybe. Zároveň by muselo byť odolné a vodotesné. Dôležitým parametrom by bola dostatočná presnosť GPS snímača a hlavne prítomnosť GSM modulu, pomocou ktorého by bolo možné odosielať geografické údaje na server okamžite po ich získaní. Vzhľadom na obmedzenú presnosť systému GPS by systém bol využiteľný hlavne pri dlhších závodoch, kde nieje potrebné vyhodnocovanie výsledkov s presnosťou na sekundy alebo desatiny sekundy. Prípadné drobné korekcie by sa museli riešiť dodatočne ručne.

Ďalšou možnosťou rozšírenia projektu by bolo zameranie sa na vylepšenie funkcií samotnej webovej aplikácie. Existuje veľké množstvo mapových pokladov ktoré by sa dali použiť v zobrazení priebehu závodu a užívateľ by mal možnosť výberu, ktorý sa použije. Táto funkcia by mala zmysel v oblastiach, kde by satelitné alebo terénne zobrazenie poskytovalo viac informácií ako súčasné zobrazenie ulíc od Google Maps.

V aplikácii by ďalej bolo možné zobrazovať viacero doplnkových údajov o užívateľoch. Napríklad aktuálna alebo priemerná rýchlosť, presná prejdená vzdialenosť, tepová frekvencia a i.

Možnou alternatívou by bola zmena prístupu ku vytváraniu závodov. Pribudol by nový typ virtuálneho závodu, ktorý by mal presne definovanú trasu aj s kontrolnými bodmi ale nemal by určený čas štartu. Závod by bolo možné absolvovať kedykoľvek, pričom čas štartu by sa určil prechodom okolo štartovacieho bodu. Týmto spôsobom by bolo možné porovnávať svoje výkony s ostatnými prípadne vylepšovať vlastné výsledky bez potreby fyzickej organizácie súťaže.

Kapitola 10

Závěr

Cieľom tejto práce bol návrh a implementácia webovej aplikácie ktorá by v praxi mohla mať praktické využitie ako systém pre automatickú tvorbu výsledkov športových súťaží. Pri tvorbe som sa zameral na to, aby aplikácia bola prehľadná a intuitívna. Pre organizátorov podujatí by mohla poslúžiť ako pomôcka pri vyhodnocovaní poradia alebo ako nástroj pre kontrolu a sledovanie závodníkov. Systém by sa mohol uplatniť hlavne pri dlhších podujatiach, kde organizátori nemajú možnosť pokryť celú trať ľudskými zdrojmi, a umožnil by tak sledovať priebeh závodu v celej jeho dĺžke. Podmienkou správneho fungovania aplikácie je kvalitné pokrytie GPS signálom a v prípade zaznamenávania podujatia v reálnom čase aj mobilným signálom s internetovým pripojením.

Na základe výsledkov testovania a ohlasov od dobrovoľníkov, ktorí sa na ňom podielali hodnotím výsledok práce pozitívne. A to aj napriek tomu, že som nemal možnosť overiť funkčnosť v reálnom závode s viac ako jedným užívateľom. V prípade, že by sa takáto možnosť v budúcnosti naskytla určite by som ju využil.

Počas tvorby projektu som sa zoznámil s rôznymi formátmi súborov vhodných pre ukladanie geografických dát a vyskúšal som si prácu s webovými technológiami pre zobazovanie máp. Za veľmi prínosné považujem skúsenosti s knižnicou Openlayers, ktorá ponúka množstvo užitočných funkcií pre prácu s online mapami a v ďalších podobných projektoch ju budem určite využívať. Taktiež som získal mnoho nových skúseností v oblasti tvorby webových systémov a naučil sa základné postupy programovania mobilných aplikácií pre operačný systém Android.

Literatura

- [1] Bednařík., M.: *Fyzika 1. 1. vyd.* Praha: České vysoké učení technické, první vydání, 2011, ISBN 978-80-01-04834-4.
- [2] Chomo, M.: *Návrh internetových stránek.* bakalářská práce, Vysoké Učení Technické v Brně, Brno, 2009.
- [3] Domes, M.: *Tvorba WWW stránek pro úplné začátečníky.* Brno: Computer Press, první vydání, 2008, ISBN 978-80-251-2160-3.
- [4] Guniš, J.: PHP.
<http://s.ics.upjs.sk/jmurin/files/druhyleto/webStranky/phpIntro.pdf>, 2012-04-15 [cit. 2014-05-06].
- [5] Hamšík, F.: *Webový systém pro publikování GPS dat.* bakalářská práce, Vysoké učení technické v Brně, Brno, 2014.
- [6] Hassman, M.: JSON : jednotný formát pro výměnu dat.
<http://www.zdrojak.cz/clanky/json-jednotny-format-pro-vymenu-dat/>, 2008-09-04 [cit. 2014-05-06].
- [7] a Jan Stankovič, R. H.: *GPS: praktická uživatelská příručka.* Brno: Computer Press, první vydání, 2007, ISBN 978-80-251-1734-7.
- [8] Lacko, M.: Čo je PHP a trošku histórie.
<http://www.phpblog.sk/clanok/70/co-je-php-a-trosku-historie/>, 2008-08-09 [cit. 2014-05-06].
- [9] Majer, T.: Problémy trasovania v rozsiahlych dopravných sieťach.
http://www.fri.uniza.sk/uploads/phd/14s-Majer_autoreferat.pdf, 2013-04-15 [cit. 2014-05-06].
- [10] Polívka, A.: *Webový systém pro správu mapových aplikací digitálních modelů terénu.* diplomová práce, České vysoké učení technické v Praze, Praha, 2010.
- [11] Ponkrác, M.: *PHP a MySQL : bez předchozích znalostí.* Brno: Computer Press, první vydání, 2007, ISBN 978-80-251-1758-3.
- [12] Rapant, P.: *Družicové polohové systémy.* Ostrava: VŠB - Technická univerzita Ostrava., první vydání, 2002, ISBN 80-248-0124-8.
- [13] WWW stránky: KML Reference [online].
<https://developers.google.com/kml/documentation/kmlreference?csw=1>, 2013-12-01 [cit. 2014-05-06].

- [14] WWW stránky: Google Maps API Tutorial.
<http://www.w3schools.com/googleapi/>, 2015-01-01 [cit. 2014-04-08].
- [15] WWW stránky: Training Center XML.
http://en.wikipedia.org/wiki/Training_Center_XML, 2015-02-04 [cit. 2014-05-06].
- [16] WWW stránky: Google Maps API licensing.
<https://developers.google.com/maps/licensing>, 2015-03-01 [cit. 2014-04-08].
- [17] WWW stránky: Google Maps API for Work.
<https://www.google.cz/intx/cs/work/mapsearch/products/mapsapi.html>,
2015-04-06 [cit. 2014-04-08].
- [18] WWW stránky: PoweManager.
<http://developer.android.com/reference/android/os/PowerManager.html>,
2015-05-01 [cit. 2014-05-06].
- [19] Škultéty, R.: *JavaScript : programujeme internetové aplikace*. Brno: Computer Press, druhé vydání, 2004, ISBN 80-251-0144-4.
- [20] Šáfry, M.: Nástroje Google. 4. Google Maps.
<http://ics.muni.cz/bulletin/articles/616.html>, 2009-04-01 [cit. 2014-05-06].

Příloha A

Obsah CD

- `/src` - zdrojové súbory webovej aplikácie
- `/src_android` zdrojové súbory aplikácie pre android
- `/tex` - zdrojové súbory technickej správy
- `/resulter.apk` - instalačný súbor android aplikácie
- `/install.php` - inštalačný súbor pre inicializáciu databázy
- `/technicka_zprava.pdf` - technická správa
- `/README.txt` - súbor so stručnými informáciami o projekte