

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

NÁVRH
Športový klub

2016

Jaroslav Fúška
Tomáš Sláma
Martin Heinz
Michal Puškel

Obsah

1	Analýza technológií	3
1.1	HTML	3
1.2	CSS	3
1.3	Bootstrap	3
1.4	LESS	4
1.5	PHP	4
1.6	JavaScript	4
1.7	Laravel	4
1.8	SQL	5
1.9	SQLite	5
1.10	WAMP / LAMP	5
2	Diagramy	6
2.1	Use-case diagram	6
2.2	Entitno-relačný diagram	7
2.3	Stavové diagramy	8
2.4	Sekvenčný diagram	9
2.5	Triedny diagram	10
2.6	Component diagram	11
3	Dátový model	13
4	Testovacie scenáre	14
4.1	Registrácia	14
4.2	Prihlásenie	15
4.3	Zmena údajov profilu	15
4.4	Možnosti bežca po prihlásení	15
4.5	Možnosti trénera po prihlásení	16
4.6	Možnosti administrátora po prihlásení	17

Kapitola 1

Analýza technológií

1.1 HTML

- HTML (Hypertext Markup Language) bude použité na základnú štruktúru webovej aplikácie a rozmiestnenie objektov.
- využitá bude verzia HTML 5.0
- HTML bude kombinované s viacerými jazykmi, ako napríklad PHP, CSS...

1.2 CSS

- CSS (Cascading Style Sheets) bude použité na vytvorenie, úpravu dizajnu a štylizovanie webovej aplikácie.
- využitá bude verzia CSS 3.0
- CSS bude využité aj na vytvorenie responzivity webovej aplikácie

1.3 Bootstrap

- Bootstrap je jednoduchá voľne stiahnuteľná sada nástrojov na tvorbu webových aplikácií
- obsahuje návrhárske šablóny založené na HTML a CSS
- pre využívanie je potrebná dobrá znalosť HTML a CSS
- využíva LESS deklarácie, čo umožňuje používanie napr. premenných a funkcií priamo v CSS kóde
- dôvod použitia: efektívnejšia práca pri navrhovaní a vytváraní responzívnej aplikácie

1.4 LESS

- LESS je "dynamic style sheet" jazyk, ktorý je možné skompilovať do CSS
- umožňuje definovať a používať premenné a funkcie
- dôvod použitia: efektívnejšia práca pri úpravách CSS, eliminácia duplicity (napr. zmena farieb...)

1.5 PHP

- PHP je skriptovací jazyk, ktorý sa používa najmä na programovanie klient-server aplikácií
- dôvod použitia: PHP dokáže spolupracovať s relačnými databázami, ako napr. MySQL alebo SQLite, ktoré budú súčasťou nášho projektu
- predpokladané použitie verzie PHP 5.5 / 7

1.6 JavaScript

- JS je multiplatformový, objektovo orientovaný skriptovací jazyk na strane klienta
- dôvod použitia: využitie na interakciu s používateľom, prácu s dátami z GoogleMaps API

1.7 Laravel

- open source framework pre PHP
- Laravel zahŕňa ako základné funkcie:
 - autentifikáciu - kontrolu prístupu
 - routovanie - spravovanie, smerovanie a spracovanie dotazov na jednom mieste
 - databázu - všetky nástroje potrebné na komunikáciu s databázou
 - mail - posielanie emailov s prílohami a vloženými súbormi
 - sessios
 - caching
- dôvod použitia: zjednodušenie práce pri programovaní funkcií, bezpečnejšie a efektívnejšie algoritmy na autentifikáciu a komunikáciu s databázou

1.8 SQL

- štruktúrovaný dopytovací jazyk určený na manipuláciu (výber, vkladanie, úpravu a mazanie) a definíciu údajov v relačných databázových systémoch
- dôvod použitia: jedna z hlavných súčastí projektu budú relačné databázy a práca s nimi

1.9 SQLite

- je relačná databáza pracujúca bez servera, čiže používa iba klientsku časť
- nie je potrebná jej inštalácia ani žiadne konfiguračné súbory
- jednoduchý import a export tabuliek
- dôvod použitia: využitie na uloženie dát potrebných pre fungovanie webovej aplikácie

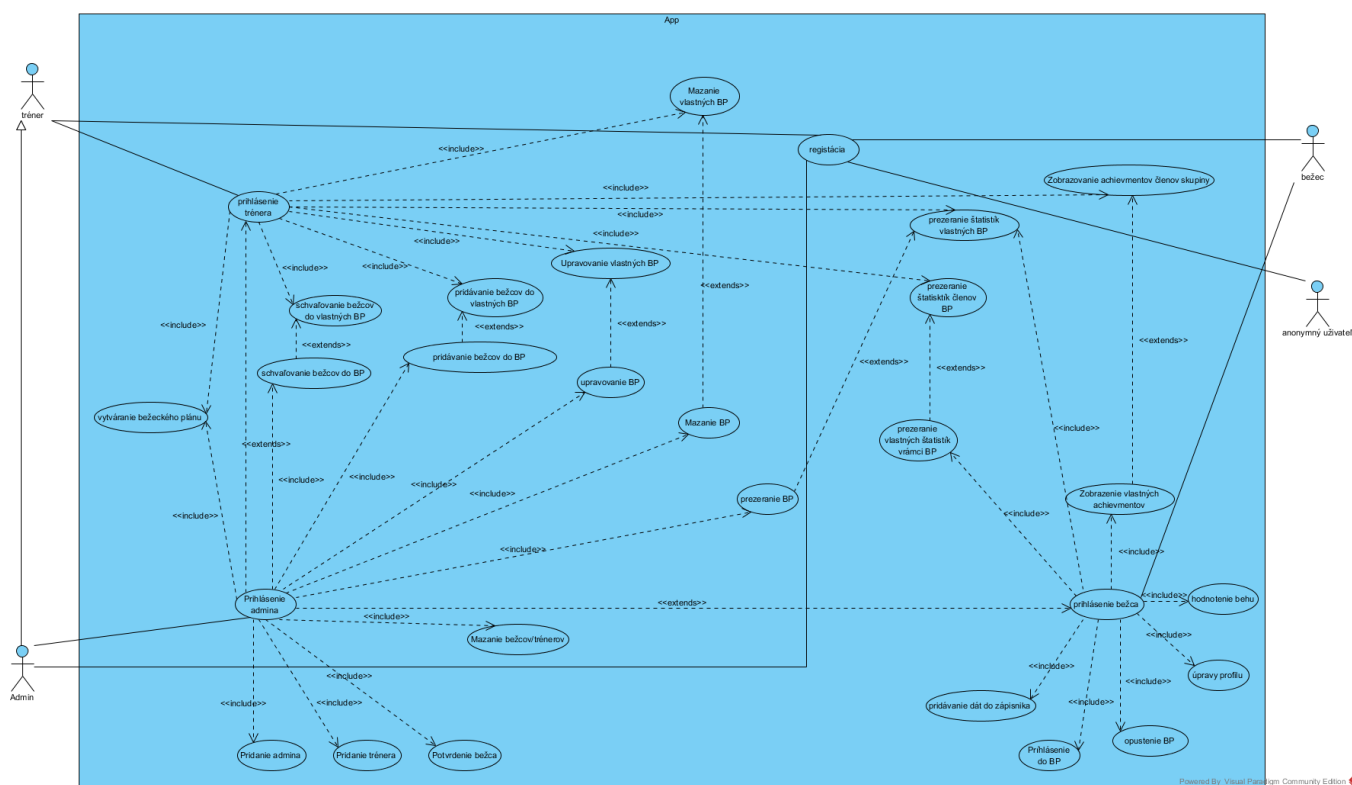
1.10 WAMP / LAMP

- apache, MySQL a PHP aplikačná serverová platforma
- dôvod použitia: využitie počas testovania webovej aplikácie bez prístupu k serveru, na ktorom bude výsledná aplikácia bežať

Kapitola 2

Diagramy

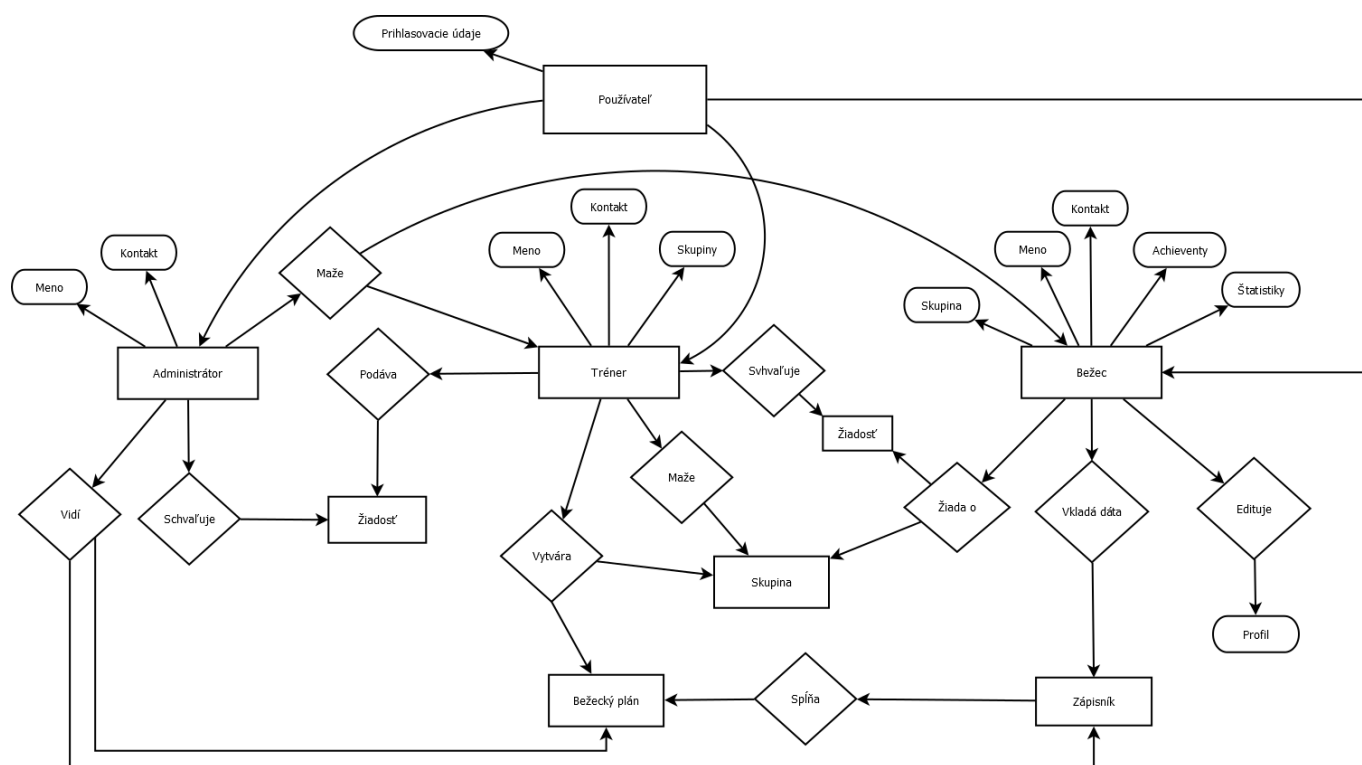
2.1 Use-case diagram



Obr. 2.1: Use-case diagram

Používateľ aplikácie je buď tréner, admin, bežec alebo anonymný používateľ. Anonymný používateľ má možnosť len registrovať sa keďže aplikácia má funkcie prístupné len po prihlásení. Bežec si po prihlásení môže prezerať svoje štatistiky a aj štatistiky svojej skupiny, takisto si môže prezerať achievementy seba aj skupiny, môže hodnotiť beh, upravovať profil a prihlasovať a opúšťať skupiny. Tréner aj admin môžu vytvárať, upravovať a mazať skupiny, schvalovať a pridávať bežcov do skupín a prezerať si štatistiky skupín s tým, že tréner má tieto funkcie povolené len pre vlastné skupiny a bežecké plány. Admin môže ešte navyše aj pridávať potvrdzovať a mazať užívateľov.

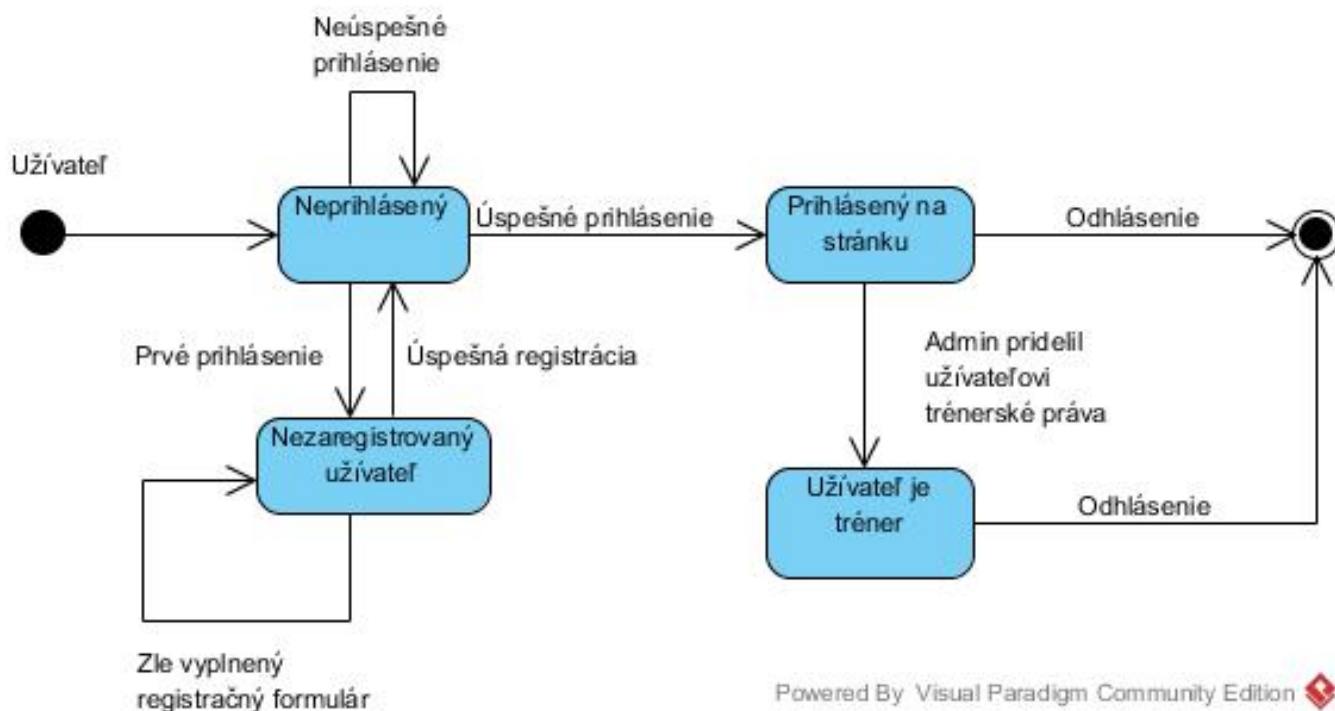
2.2 Entitno-relačný diagram



Obr. 2.2: ERD

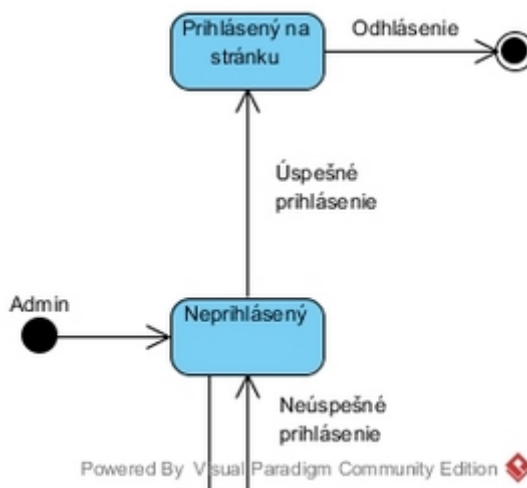
Entitno-relačný diagram (obr 2.2) zobrazuje entity vystupujúce v systéme a relácie (vzťahy) medzi nimi. Diagram slúži pre lepšiu orientáciu počas návrhu dátového diagramu (databázy) a aj počas navrhovania diagramov popisujúcich samotnú funkcionality systému. ER diagram znázorňuje delenie používateľov do troch skupín na Administrátora, Trénera a Bežca a popisuje jednotlivé kľúčové funkcie a atribúty ktoré bude aplikácia obsahovať.

2.3 Stavové diagramy



Obr. 2.3: User state diagram

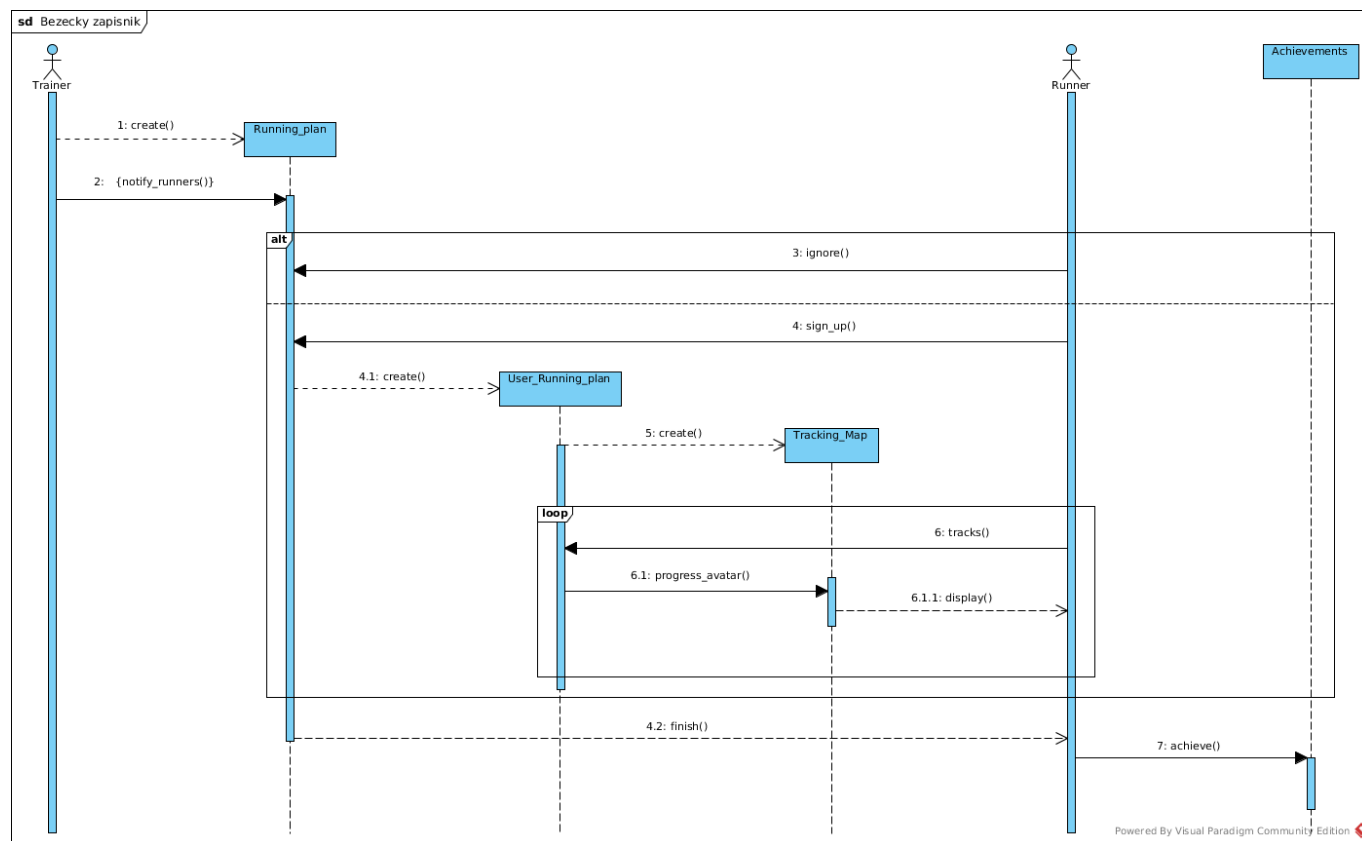
Stavový diagram užívateľa (obr. 2.3) zobrazuje pasívne stavy užívateľa v systéme a udalosti/akcie, na základe ktorých sa dostáva do iného stavu.



Obr. 2.4: Admin state diagram

Stavový diagram admina (obr. 2.4) zobrazuje pasívne stavy admina v systéme a udalosti/akcie, na základe ktorých sa dostáva do iného stavu.

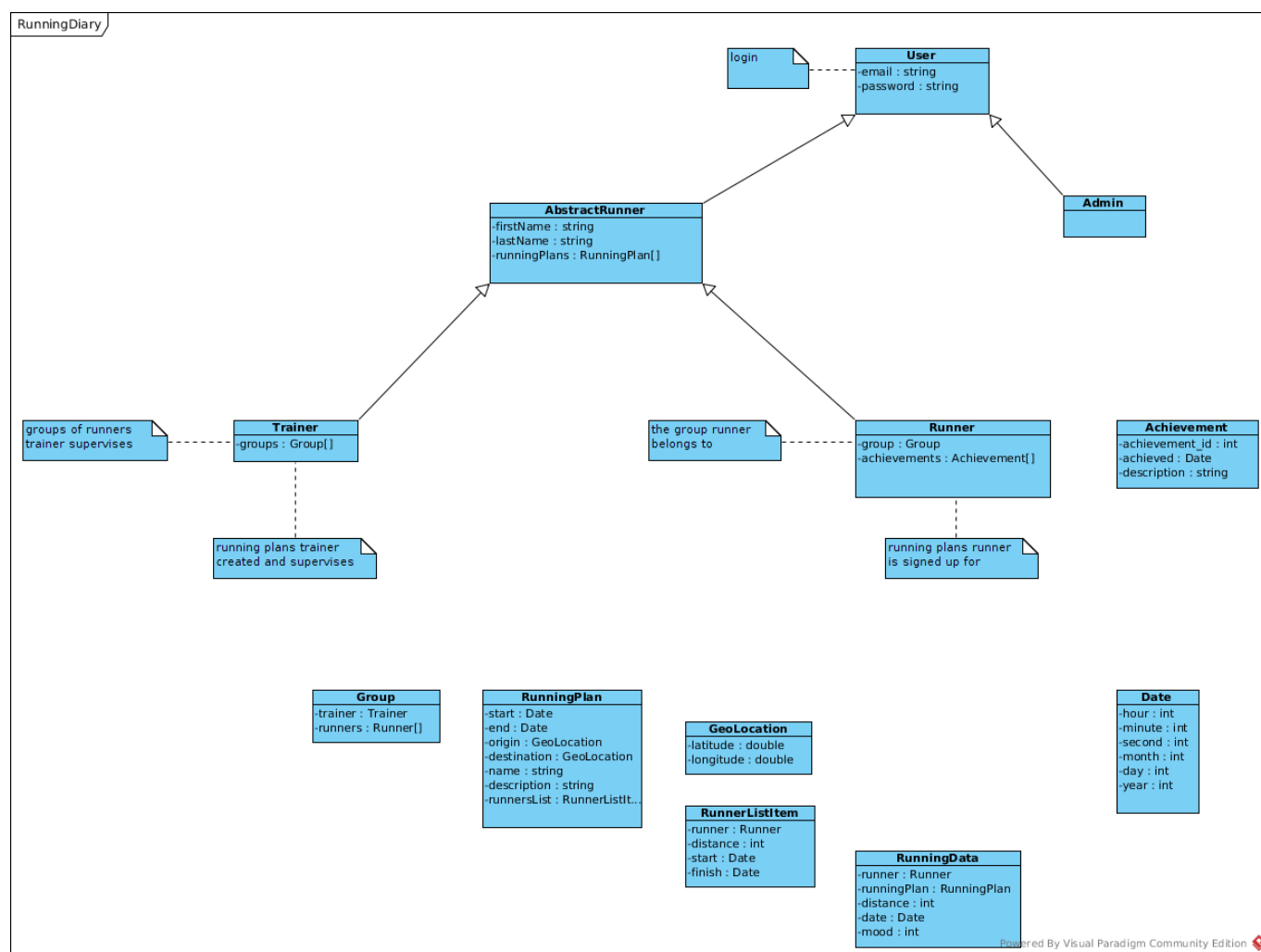
2.4 Sekvenčný diagram



Obr. 2.5: Sekvenčný diagram

Sekvenčný diagram (obr. 2.5) graficky zachytáva sled nadväzných udalostí zaznačovania pokroku v plnení bežeckého plánu, vytvoreného trénerom a postupnými iteráciami naplňajúceho sa bežcom.

2.5 Triedny diagram

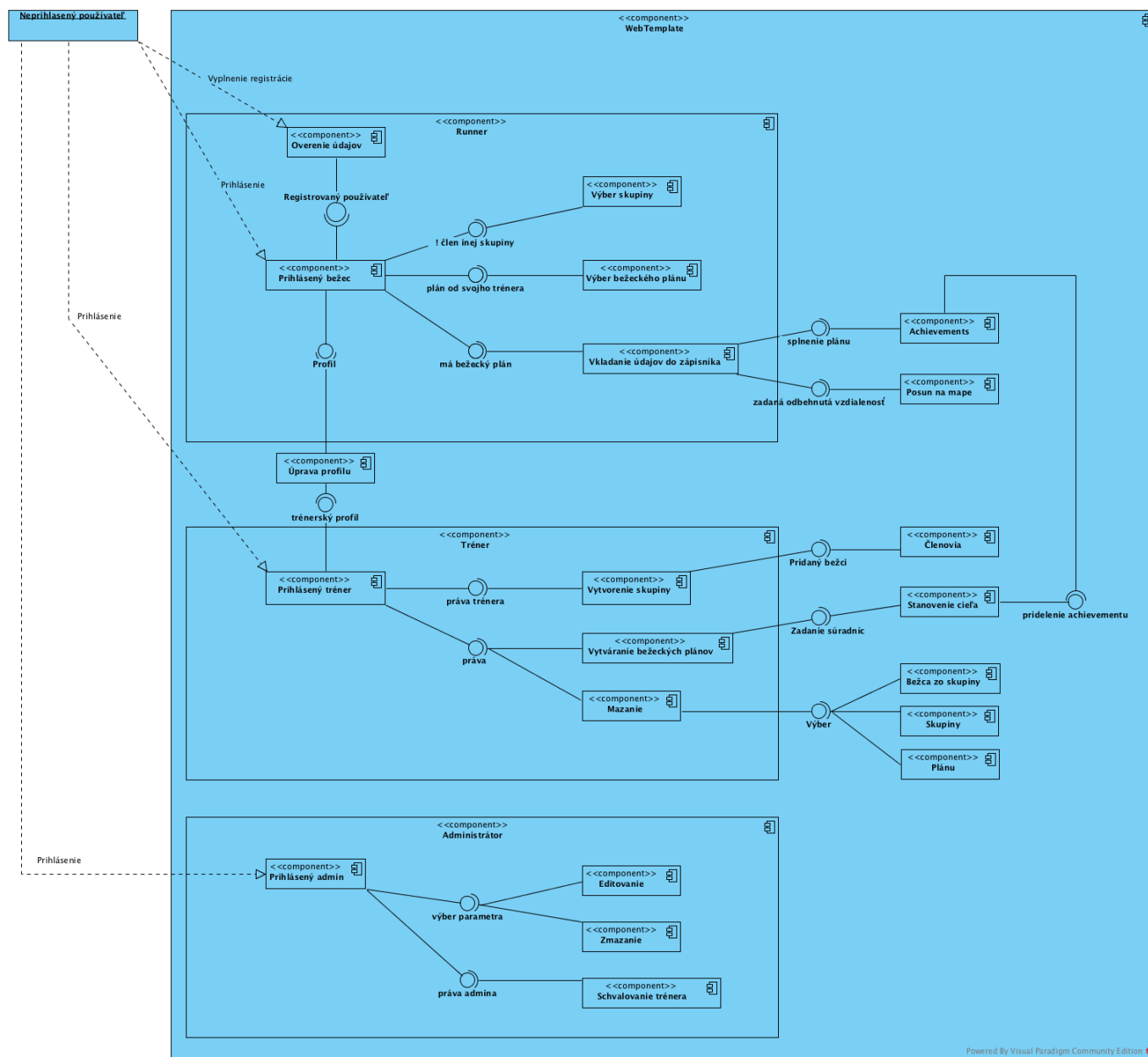


Obr. 2.6: Triedny diagram

Triedny diagram (obr. 2.6) je zhmotnením abstrakcie, keď vidíme ako jednotlivé PHP-čkovské triedy dedia spoločné atribúty či metódy od svojich predkov, ako je to aj v prípade **AbstractRunnera** (resp. **Trainera** či **Runnera**) a **Admina** nakoľko oni všetci podliehajú nutnej podmienke prihlásenia sa do systému v zmysle dodržania dohody o ochrane osobných údajov so súladom s požiadavkami zadávateľa a disponujú triednymi vlastnosťami email a password.

Perličkou je zdedenie `runningPlans` v triedach **Trainer** a **Runner** kde tento atribút predstavuje úplne iný význam v oboch triedach, čo však vďaka dobre zdokumentovanému návrhu nikdy nebude nesprávne pochopené ani implementačným tímom tretích strán. Konkrétne `runningPlans` v triede **Trainera** predstavujú plány, ktoré **Trainer** vytvoril a spravuje, kdežto v triede **Runnera** sú to plány, v ktorých je konkrétny bežec prihlásený.

2.6 Component diagram



Obr. 2.7: Component diagram

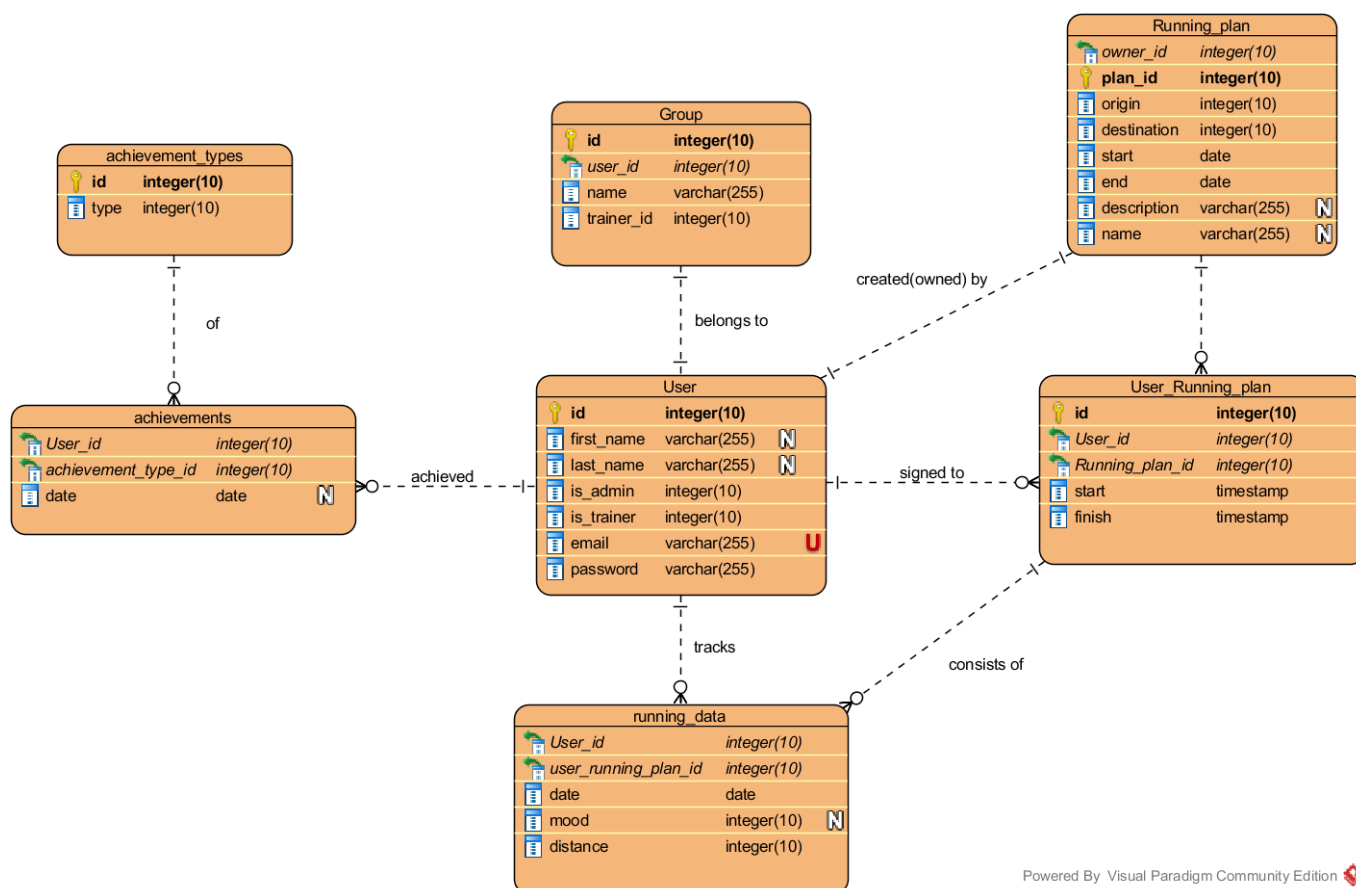
Component diagram (obr. 2.7) zobrazuje komponenty, ktoré tvoria systém a závislosti medzi nimi. Stručný popis základnej funkcionality kľúčových komponentov:

- Component WebTemplate obsahuje celé používateľské rozhranie a dizajn webovej aplikácie. Componet tvoria ďalšie componenty, ktoré môžeme rozdeliť do troch podkategórií do Runner, Tréner, Administrátor a ktoré budú spúšťané podľa výsledku autentifikácie.
- Component Úprava profilu umožní Bežcovi alebo Trénorovi upraviť svoj profil, a to napríklad výmenou profilovej fotografie

- Component Schvaľovanie trénera umožňuje administrátorovi udeliť obyčajnému bežcovi vyššie právomoci a teda dať mu status Trénera. Tento status mu môže samozrejme aj odobrať.
- Component Zmazanie umožňuje trénerovi mazať vybrané údaje.
- Component Vkladanie údajov do zápisníka umožňuje Bežcovi vložiť odbehnutú vzdialenosť ktorá sa následne zapíše do databázy a pomocou Componentu Posun na mape Bežec uvidí svoj progres pomocou posunu značky na mape po vyznačenej trase.
- Component Mazanie umožňuje Trénerovi mazať dáta podľa zvolených vstupných podmienok. Funkčnosť je podobná (skoro totožná) s Componentom Zmazanie pre Administrátora.
- Component Vytvorenie skupiny umožní Trénorovi vytvoriť novú skupinu a pridať do nej bežcov.
- Component overenie údajov slúži na overenie pravosti údajov zadaných počas registrácie, ak prebehne overenie bez problémov používateľ bude informovaný e-mailom o úspešnej registrácii.

Kapitola 3

Dátový model



Powered By Visual Paradigm Community Edition

Obr. 3.1: Dátový model

Vysvetlivky k diagramu:

- biele N znamená, že field je nullable
- červené U znamená, že field je unique
- zelené šípku označujú cudzie kľúče a tučným sú označené primárne kľúče

Kapitola 4

Testovacie scenáre

4.1 Registrácia

- – Vstup: Správne vyplnenie registračného formulára.
– Výstup: Vytvorenie neaktívneho konta a profilu v databáze, zobrazenie informácie o obdržaní emailu v prípade, že požiadavku schváli admin, zaslanie požiadavky adminovi.
– Stav testovania: Treba otestovať
- – Vstup: Potvrdenie požiadavky o registráciu adminom.
– Výstup: Odoslanie informácie o potvrdení registrácie užívateľovi, aktivácia užívateľovho konta.
– Stav testovania: Treba otestovať
- – Vstup: Odmietnutie požiadavky o registráciu adminom.
– Výstup: Odoslanie informácie o odmietnutí registrácie užívateľovi, vymazanie užívateľovho konta z databázy.
– Stav testovania: Treba otestovať
- – Vstup: Nesprávne vyplnenie registračného formulára
– Výstup: Zobrazenie informácie užívateľovi o zle vyplnených údajoch.
– Stav testovania: Treba otestovať
- – Vstup: Snaha o „CODE Injection“ pri vyplňovaní registračného formulára.
– Výstup: Zobrazenie informácie užívateľovi o zle vyplnených údajoch.
– Stav testovania: Treba otestovať

4.2 Prihlásenie

- – Vstup: Správne vyplnené prihlasovacie údaje.
– Výstup: Užívateľovi je povolený prístup na stránku.
– Stav testovania: Treba otestovať
- – Vstup: Nesprávne vyplnené prihlasovacie údaje.
– Výstup: Zobrazenie informácie o zle vyplnených údajoch.
– Stav testovania: Treba otestovať
- – Vstup: Snaha o „CODE Injection“ pri vyplňovaní prihlasovacieho formulára.
– Výstup: Zobrazenie informácie užívateľovi o zle vyplnených údajoch.
– Stav testovania: Treba otestovať

4.3 Zmena údajov profilu

- – Vstup: Správne vyplnenie nových údajov.
– Výstup: Zmena údajov v profile užívateľa.
– Stav testovania: Treba otestovať
- – Vstup: Nesprávne vyplnenie nových údajov.
– Výstup: Zobrazenie informácie o zle vyplnených údajoch.
– Stav testovania: Treba otestovať
- – Vstup: Nesprávne zadanie hesla pri potvrdzovaní zmien v profile.
– Výstup: Zobrazenie informácie o zle vyplnených údajoch.
– Stav testovania: Treba otestovať
- – Vstup: Snaha o „CODE Injection“ pri vyplňovaní nových údajov.
– Zobrazenie informácie užívateľovi o zle vyplnených údajoch.
– Stav testovania: Treba otestovať

4.4 Možnosti bežca po prihlásení

- – Vstup: Zobrazenie štatistík.
– Výstup: Zobrazenie štatistík prisluchajúcich danému bežcovi.
– Stav testovania: Treba otestovať

- – Vstup: Požiadavka na pridanie do skupiny.
- Výstup: Trénerovi danej skupiny príde požiadavka na pridanie bežca do skupiny.
- Stav testovania: Treba otestovať
 - * Vstup: Požiadavka na pridanie do skupiny bola akceptovaná.
 - Výstup: Bežec je pridaný do skupiny.
 - Stav testovania: Treba otestovať
 - * Vstup: Požiadavka na pridanie do skupiny bola zamietnutá.
 - Výstup: Bežcovi príde mail s informáciou, že jeho požiadavka na pridanie do skupiny bola zamietnutá.
 - Stav testovania: Treba otestovať
- – Vstup: Opustenie skupiny.
- Výstup: Bežec je vyradený zo skupiny.
- Stav testovania: Treba otestovať
- – Vstup: Pridanie validných dát do zápisníka.
- Výstup: Dáta z behu sa zapíšu do databázy.
- Stav testovania: Treba otestovať
- – Vstup: Pridanie nesprávnych dát do zápisníka.
- Výstup: Zobrazenie informácie o nesprávne vyplnených dátach.
- Stav testovania: Treba otestovať

4.5 Možnosti trénera po prihlásení

- – Vstup: Zobrazenie štatistík skupiny.
- Výstup: Zobrazenie štatistík prisluchajúcich skupine, ktorej je trénerom.
- Stav testovania: Treba otestovať
- – Vstup: Vyhodenie bežca zo skupiny.
- Výstup: Bežec je vyradený zo skupiny, obdrží email o tom že bol vyradený zo
- Stav testovania: Treba otestovať
- – Vstup: Vytvorenie bežeckého plánu.
- Výstup: Bežecký plán sa uloží do databázy.
- Stav testovania: Treba otestovať
- – Vstup: Vytvorenie skupiny.

- Výstup: Skupina je uložená do databázy.
- Stav testovania: Treba otestovať
- – Vstup: Vymazanie skupiny.
 - Výstup: Všetci bežci patriaci do skupiny obdržia mail že bola skupina zrušená, všetky bežecké plány suvisiace so skupinou zaniknú.
 - Stav testovania: Treba otestovať
- – Vstup: Export tabuliek.
 - Výstup: Export tabuliek so štatistikami skupiny vo zvolenom formáte.
 - Stav testovania: Treba otestovať

4.6 Možnosti administrátora po prihlásení

- – Vstup: Vybranie užívateľa za trénera.
 - Výstup: Užívateľ sa stane trénerom a dostane o tom informatívny email.
 - Stav testovania: Treba otestovať
- – Vstup: Úprava údajov užívateľa.
 - Výstup: Užívateľovi sa zmenia údaje.
 - Stav testovania: Treba otestovať
- – Vstup: Vymazanie užívateľa.
 - Výstup: Užívateľské konto je vymazané z databázy, užívateľ dostane email o tom že jeho konto bolo zmazané.
 - Stav testovania: Treba otestovať
- – Vstup: Vymazanie skupiny.
 - Výstup: Všetci bežci patriaci do skupiny obdržia email ,že bola skupina zrušená, tréner tejto skupiny dostane email, že jeho skupina bola vymazaná, všetky bežecké plány suvisiace so skupinou zaniknú.
 - Stav testovania: Treba otestovať