

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

Webová aplikace pro přípravu na závody
Bakalářská práce

Autor: Alex Zamastil
Studijní obor: Aplikovaná informatika

Vedoucí práce: Ing. Jan Krunčík

Hradec Králové

duben 2024

Prohlášení:

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

V Hradci Králové

Alex Zamastil

Poděkování:

Děkuji vedoucímu této bakalářské práce Ing. Janu Krunčíkovi za metodické vedení práce, předání cenných rad z praxe a pomoc se zpracováním této práce.

Abstrakt

Tato bakalářská práce se věnuje tvorbě webové aplikace, určené pro přípravu na závod v běhu, nebo v běhu s překážkami. Cílem je využít moderní technologie v prospěch zlepšení kvality života uživatelů aplikace. Z technického pohledu je cílem vytvořit komplexní projekt, který bude tento teoretický cíl umožňovat. V práci jsou nejprve představeny konkurenční aplikace a teoretické principy, které by měly být použity pro tvorbu tréninku a jídelníčku. Pak jsou popsány požadavky na aplikaci a samotná implementace práce, spolu s představením použitých technologií. U implementace je kladena pozornost zejména na algoritmus generování tréninkových dat, což je jádrem celé aplikace. Na závěr jsou shrnuty možnosti budoucího rozvoje aplikace, kde se nabízí mnoho možností.

Abstract

Title: Web application for run preparation

This bachelor's thesis focuses on the creation of a web application designed for race preparation, either in running or obstacle course running. The aim was to utilize modern technologies to improve the quality of life for application users. From a technical perspective, the goal was to develop a comprehensive project capable of achieving the theoretical objective. The thesis first introduces competitive applications and theoretical principles that should be used for training and meal planning. Then, it describes the requirements for the application and the implementation itself, along with an introduction to the technologies used. Special attention is paid to the implementation, particularly the algorithm for generating training data, which forms the core of the entire application. Finally, the possibilities for future development of the application are summarized, offering numerous opportunities for further enhancement.

Klíčová slova: Webová aplikace, Trénink, Sportovní příprava, Strojové učení, Spring Boot, React

Keywords: Web application, Training, Sports preparation, Machine learning, Spring Boot, React

Obsah

1	Úvod.....	1
2	Cíl a metodika práce.....	3
2.1	Požadavky	3
2.1.1	Funkční požadavky	3
2.1.2	Nefunkční požadavky	3
2.2	Cíl.....	4
2.3	Přiznání o využití AI.....	4
3	Teoretický základ	5
3.1	Cílová skupina	5
3.2	Existující služby.....	5
3.2.1	Nike Run Club	5
3.2.2	RunReps.....	6
3.2.3	Fitify.....	6
3.2.4	RunningCoach	7
3.3	Shrnutí průzkumu trhu	7
3.4	Princip tvorby tréninku	8
3.4.1	Stanovení cíle	8
3.4.2	Progresivita tréninku a dosažení cíle.....	11
3.4.3	Regenerace a bezpečnost	15
3.5	Princip tvorby jídelníčku	16
3.5.1	Energetický příjem	16
3.5.2	Sacharidy	17
3.5.3	Bílkoviny	17
3.5.4	Tuky	18
3.5.5	Odměňování.....	18
3.5.6	Shrnutí jídelníčku.....	19
4	Návrh a vývoj	20
4.1	Funkcionalita	20
4.2	Uživatelské rozhraní	22
4.3	Použité technologie.....	22
4.3.1	Backend	22

4.3.2	Frontend.....	23
4.3.3	Databáze.....	25
4.3.4	Server	26
4.4	Architektura projektu	27
4.4.1	Komunikace vrstev	27
4.5	Zabezpečení	29
4.5.1	JSON Web Token	29
4.5.2	Autorizační filtr.....	30
4.5.3	Další zabezpečení	31
5	Proces generování	33
5.1	Rozhodovací strom J48.....	33
5.2	Tvorba algoritmu	34
5.2.1	Změna přístupu	36
5.2.2	Finální verze algoritmu.....	38
5.3	Porovnání tréninku.....	39
6	Testování a vyhodnocení.....	42
6.1	Jednotkové testování.....	42
7	Shrnutí výsledků.....	44
8	Závěr	45
9	Seznam obrázků	1
10	Seznam zdrojových kódů	2
11	Seznam tabulek	3
12	Literatura	4
13	Přílohy	1
14	Zadání práce z IS.....	1

1 Úvod

Žijeme v době, kdy technologický pokrok ovlivňuje skoro každý aspekt lidského života a sport není výjimkou. Na jednu stranu nám technologie dávají velmi široké možnosti zkvalitnění a zpříjemnění sportovních aktivit, např. chytré sportovní hodinky, měřící v průběhu aktivit různé atributy našeho těla, plný internet informací a rad od profesionálních sportovců a trenérů, nebo třeba široká dostupnost výživových doplňků pro sportovce. Velmi pravděpodobně žijeme v době s nejlepšími předpoklady pro úspěch ve sportu.

Na druhou stranu ale technologie mohou mít na člověka velmi negativní vliv, a to z psychické i fyzické stránky. Hrozí závislosti na sociálních sítích, videohrách, a další problémy spojené s technologickým rozvojem. Nadměrná produkce nezdravého jídla a další faktory vedou ke globálnímu problému se špatnou životosprávou a nadměrnou obezitou populace. [1] Lze to vidět např. v USA, jedné z nejvyspělejších zemí světa, kde je v mnoha státech obezních přes 35 % dospělých jedinců. Tato statistika má za posledních 10 let pouze rostoucí tendenci. [2] Konzumace přidaných tuků v potravinách se zvedla od roku 1970 o 45 % v průměru a rychlé občerstvení tvoří 11 % stravy průměrného Američana.

Aplikace pro přípravu na závod má za cíl bojovat s tímto problémem tím, že nabízí vstup do světa sportu začátečníkům, kteří se sportem nemají zkušenosti. Smyslem aplikace je příprava na konkrétní, uživatelem specifikovaný závod, lze chápat i jako cíl (např. uživatel chce za půl roku být schopný uběhnout 10 km). Aplikace není určená pro profesionální sportovce, kteří si vhodný trénink volí dle vlastních preferencí a za pomoci trenérů a odborníků. Trénink je založený na konkrétních proporcích a cílech uživatele, ale jde o obecné principy, které pro profesionály pravděpodobně nebudou stačit.

Cílem práce je vytvoření funkčního webu, splňujícího všechny funkční i nefunkční požadavky. Web má být intuitivní, přehledný, s moderním designem a odpovídající běžným, obecným praktikám, používaných při tvorbě webu.

Existuje mnoho webů, blogů a aplikací zaměřených na pomoc sportovcům, práce je relevantní v tom, že spojuje tvorbu personalizovaného tréninku a jídelníčku dohromady a zároveň obsahuje rady, bezpečnostní upozornění, návody a cenné informace, které začátečník ocení. Zvolenou disciplínou je běh, protože je to jeden z nejdostupnějších sportů, je vhodný pro lidi, co nemají se sportem zkušenosti a je to vhodná aerobní aktivita pro spalování tuků, což bývá častým podmětem k započetí cesty sportovce. Následující obrázek ukazuje porovnání spalování tuků při běhu v porovnání s dalšími aerobními aktivitami. Kromě běhu je i možnost přípravy na takzvaný Obstacle course racing, česky překážkový

závod, či Spartan race, nebo Gladiator race. Tento druh závodu vyžaduje všestranné fyzické dovednosti a vyžaduje komplexnější přípravu než pouhý běh. Byl zvolen jako výzva pro už zkušenější sportovce.

Obecně se bude jednat o užitečný nástroj, který se snaží poskytnout všechny relevantní informace na jednom místě.

Aktivita	Váha	59	62	65	68	71	74	77	80
Volejbal		3.0	3.1	3.3	3.4	3.6	3.7	3.9	4.0
Cyklistika		5.9	6.2	6.5	6.8	7.1	7.4	7.7	8.0
Tenis		6.4	6.8	7.1	7.4	7.7	8.1	8.4	8.7
Plavání		7.6	7.9	8.3	8.7	9.1	9.5	9.9	10.2
Běh při tempu 5 min/km		12.5	13.1	13.6	14.2	14.8	15.4	16.0	16.5

Tabulka 1: Energetický výdej aerobních aktivit (zdroj: [10])

2 Cíl a metodika práce

Tato kapitola obsahuje stručný popis požadavků na vyvíjenou aplikaci, cíle projektu a přiznání o využití AI v průběhu tvorby této práce.

2.1 Požadavky

Požadavky na aplikaci lze rozdělit na funkční a nefunkční dle jejich charakteristiky. Funkční požadavky aplikace jsou konkrétní funkce, které aplikace nabízí a konkrétní chování systému. Jinými slovy určují, co aplikace umí. Nefunkční požadavky neurčují, co aplikace umí, ale jak pracuje. Stanovují, jak by aplikace měla fungovat, ať už z hlediska výkonnosti, spolehlivosti, nebo úrovně zabezpečení.

2.1.1 Funkční požadavky

- Základní uživatelské operace – registrace, přihlášení, odhlášení, změna údajů
- Vypočítání relevantních údajů na základě hodnot uživatele – BMI, doporučený příjem vody, hodnota bazálního metabolismu
- Možná existence administrátorských účtů, které umožňují nadstandartní funkce, ke kterým běžný uživatel nemá přístup.
- Funkce tvorby tréninku, který bude personalizován vstupním datům uživatele.
- Možnost prohlížení uživatelských dat v sekci profil a tréninkových dat v sekci trénink
- Poskytnutí relevantních bezpečnostních upozornění

2.1.2 Nefunkční požadavky

- Moderní a působivý vzhled aplikace
- RESTful komunikace mezi webovou a backendovou částí
- Rychlá odezva aplikace
- Kvalitní zabezpečení komunikace
- Responzivní design a lokalizace pro český a anglický jazyk
- Konzistentní vzhled aplikace na většině internetových prohlížečů
- Uložení citlivých informací do databáze v zašifrované podobě

- Intuitivní navigace

2.2 Cíl

Cílem projektu je vytvořit webovou aplikaci, která bude umožňovat uživateli vytvořit si tréninkový plán na nadcházející závod v překážkovém, nebo normálním běhu. Z technického hlediska by měla být vytvořena webová aplikace, která dokáže tento nelehký úkol splnit. Pro tvorbu tréninkového plánu by měl být využit některý z algoritmů strojového učení.

2.3 Přiznání o využití AI

Během této práce byly využity nástroje umělé inteligence (AI). Byly použity 2 druhy chatbota: ChatGPT 3.5 od společnosti OpenAI a Gemini od společnosti Google. Tyto nástroje byly použity k brainstormingu a ke generování doporučení, týkajících se projektu z hlediska struktury a jazyka. Nástroje umělé inteligence nebyly použity pro získávání faktických informací. Za pravdivost informace v projektu ručí autor. Převzaté faktické informace mají citovaný původ. Informace bez citací jsou autorovy domněnky, zkušenosti, nebo běžné praktiky (např. vzorec pro doporučený příjem vody nemá zdroj, jde o běžnou praxi). Nutno také podotknout, že mnoho informací v oboru zdraví a sportu mohou být subjektivní.

3 Teoretický základ

Tato kapitola obsahuje průzkum obdobných služeb na trhu, poznatky získané zkoumáním, teoretické principy pro stanovení cíle a principy použité pro tvorbu tréninku a jídelníčku.

3.1 Cílová skupina

Cílovou skupinou projektu budou zdraví, dospělí jedinci, muži i ženy, sportovní začátečníci. Aplikace nebude vhodná pro děti, protože se vychází ze studií a experimentů, zkoumaných převážně na dospělých, a protože vývoj lidského těla je individuální a těžko se tvoří personální trénink a bezpečnostní upozornění s velkým množstvím proměnných. Dále aplikace není vhodná pro těhotné a kojící ženy, hlavně kvůli důležitosti specifické stravy, potřebné pro zdravý vývoj dítěte a také kvůli fyziologickým změnám těla během těhotenství. Většina tréninkových aktivit, které budou uživatelům doporučovány, by mohly být během těhotenství nevhodné nebo i nebezpečné.

3.2 Existující služby

Na internetu existuje mnoho různých aplikací s podobným zaměřením, většinou ale mají určitá omezení. Mezi mladými běžci je populární např. mobilní aplikace Nike Run Club, na obchodu Play exceluje česká aplikace Fitify s více než 10 miliony stažení a co se týče desktopových aplikací, úspěšné jsou RunningCoach, nebo například RunReps.

3.2.1 Nike Run Club

[27] Nike Run Club funguje pouze jako mobilní aplikace, umožňující sledování výkonu během běhu. Aplikace obsahuje i inovace ve stylu hlasových návodů a povzbuzování, které uživatel slyší ze sluchátek během samotného tréninku. Mimo to aplikace přehrává hudbu, která by se měla k aktuální intenzitě běhu hodit.

Co se týče plánování tréninku, obsahuje aplikace 4 tréninkové plány (maraton, půlmaraton, 5 km a začátečník) a 267 předpřipravených běhů různých intenzit s odpovídající hudbou a pokyny (267 běhů ke dni 14.8.2023).

Otázku stravy, bezpečnosti a regenerace aplikace neřeší. Jako nástroj pro zaznamenání tréninku a sdílení záznamů z běhu je aplikace dostačující a má skvělý design, na druhou stranu ale není příliš vhodná pro začátečníka – ze 267 druhů tréninku je náročné si vybrat

nějaká vhodný, obzvlášť, když uživatel např. ještě nikdy žádný trénink nezkusil. Další drobná nevýhoda je fakt, že je aplikace dostupná pouze v angličtině.

3.2.2 RunReps

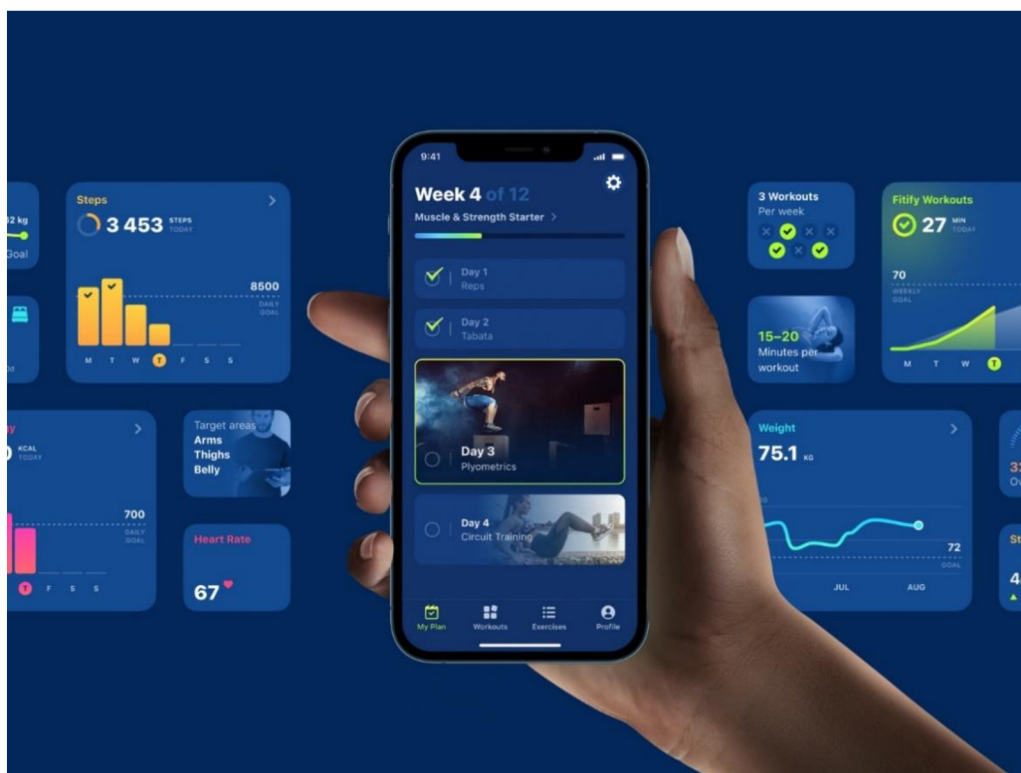
[29] RunReps je Webová stránka, která slouží pro generování běžeckého tréninkového plánu. Web nemá možnost přihlášení, plán je velmi prostý – je generován na základě 4 atributů. Vygenerovaný trénink je tabulkový formát, která lze exportovat do formátu PDF, či CSV. Ukázka tréninku je pro lepší čitelnost uvedena v příloze č. 1.

3.2.3 Fitify

[28] Fitify je velmi dobrá mobilní aplikace vytvořená českou firmou Fitify Workouts s.r.o. Nejedná se o přímou konkurenci pro webovou aplikaci pro přípravu na závody, protože se nezaměřuje na běh, ale na cvičení obecně. Aplikace má velmi hezky zpracované uživatelské prostředí, viz následující obrázek. Tréninky jsou personalizované na základě vstupního dotazníku, který zabere zhruba 5 minut. Uživatel si může vybrat jednoho z několika virtuálních trenérů, se kterým může komunikovat, trenéři fungují na bázi umělé inteligence. Také samotný tréninkový plán je údajně generován pomocí umělé inteligence.

Nevýhoda vytvořeného tréninku je, že jej nelze dále upravovat. Takže pokud si začátečník, který nemá představu o tom, co zvládne, nastaví příliš náročný trénink, je nucen začít znovu bez předchozího pokroku. Hlavní nevýhoda aplikace je, že personalizovaný trénink a většina předem sestavených tréninků jsou pouze pro uživatele, kteří platí měsíční poplatek 249kč (nebo 1799kč ročně).

Otázku stravy řeší Fitify pomocí záložky jídla, kde najdeme desítky receptů, které lze filtrovat podle atributů, např. počet kalorií nebo druh jídla. Je to validní řešení, nicméně to začátečníkovi nepomůže se stravovacími návyky. Je předhozen před desítky receptů, kde netuší, co jeho tělo zrovna potřebuje.



Obrázek 1: Ilustrační obrázek Fitify UI (zdroj: [25])

3.2.4 RunningCoach

Tento web nabízí registraci, personalizovaný plán různých druhů běhu (vytrvalostní, intervalový, ...). Otázku jídelníčku řeší pouze pomocí blogu, kde jsou zveřejňovány příspěvky mimo jiné o stravě. [30] RunningCoach poskytuje statistiky zobrazované v grafech a kalendář, ve kterém jsou plánované běhy. Bohužel jde opět o placenou službu, zdarma jsou první 2 týdny tréninku, pro další je nutné platit měsíční členství začínající na 10 eurech. Za 88 euro je k dispozici prémiová verze umožňující komunikaci se skutečným trenérem.

3.3 Shrnutí průzkumu trhu

Na trhu existují různé podobné aplikace, ale žádná z těch nejrelevantnějších přímo neřeší problematiku personalizovaného generování tréninku a zároveň jídelníčku pro začátečníky.

K překvapení na zmíněných stránkách/aplikacích nejsou na první pohled vidět bezpečnostní rady, nebo návod na protažení svalů po běhu. Velmi pravděpodobně tam jsou, ale ve formě příspěvku, nebo blogu, na což běžný uživatel nemusí narazit.

Zjistilo se, že si uživatelé stěžují na chybějící možnost upravit plán v průběhu (někdo si např. nastavil plán na 3 měsíce a po týdnu zjistil, že je plán příliš náročný). Proto je důležité tuto funkci v implementaci tréninku nevynechat – je nutné, aby uživatel měl možnost změnit průběh tréninku buď pomocí zpětné vazby, nebo aby i cíleně mohl trénink ovlivnit k obrazu svému.

Co se týče otázky placené části obsahu, filozofie projektu je vyřešení problému, a ne vydělání peněz. Z psychologického pohledu také není placená část vhodná, mohla by mnoho začátečníků odradit. Na druhou stranu lze říct, že když už za danou službu zákazník zaplatí, přikládá jí určitou důležitost. Placené členství by pravděpodobně vedlo k mnohem méně uživatelům, kteří by ale v průměru dosahovali lepších výsledků.

Funkce exportu tréninku do souboru s tabulkou dává smysl v projektu RunReps, kde není k dispozici přihlášení a tréninkový den má pouze 2 atributy: délku a typ běhu. U komplexnějších tréninků by taková tabulka byla velmi nepřehledná.

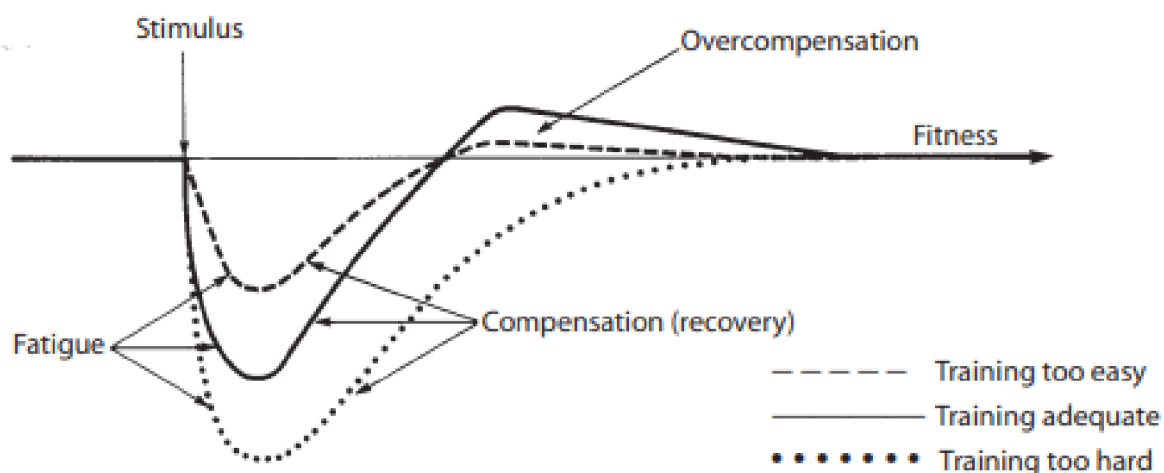
Za zmínku stojí velká konkurence v oblasti vzhledu uživatelského rozhraní, aplikace Fitify a Nike Run Club nabízí velmi propracované uživatelské rozhraní s intuitivní, jednoduchou navigací, při tvorbě aplikace bude třeba klást na tento aspekt důraz.

3.4 Princip tvorby tréninku

Tato část se věnuje teoretickým základům, ze kterých lze při implementaci generačního algoritmu vycházet. Konkrétní implementace generačního algoritmu bude popsána v kapitole 4 – Implementace a vývoj.

3.4.1 Stanovení cíle

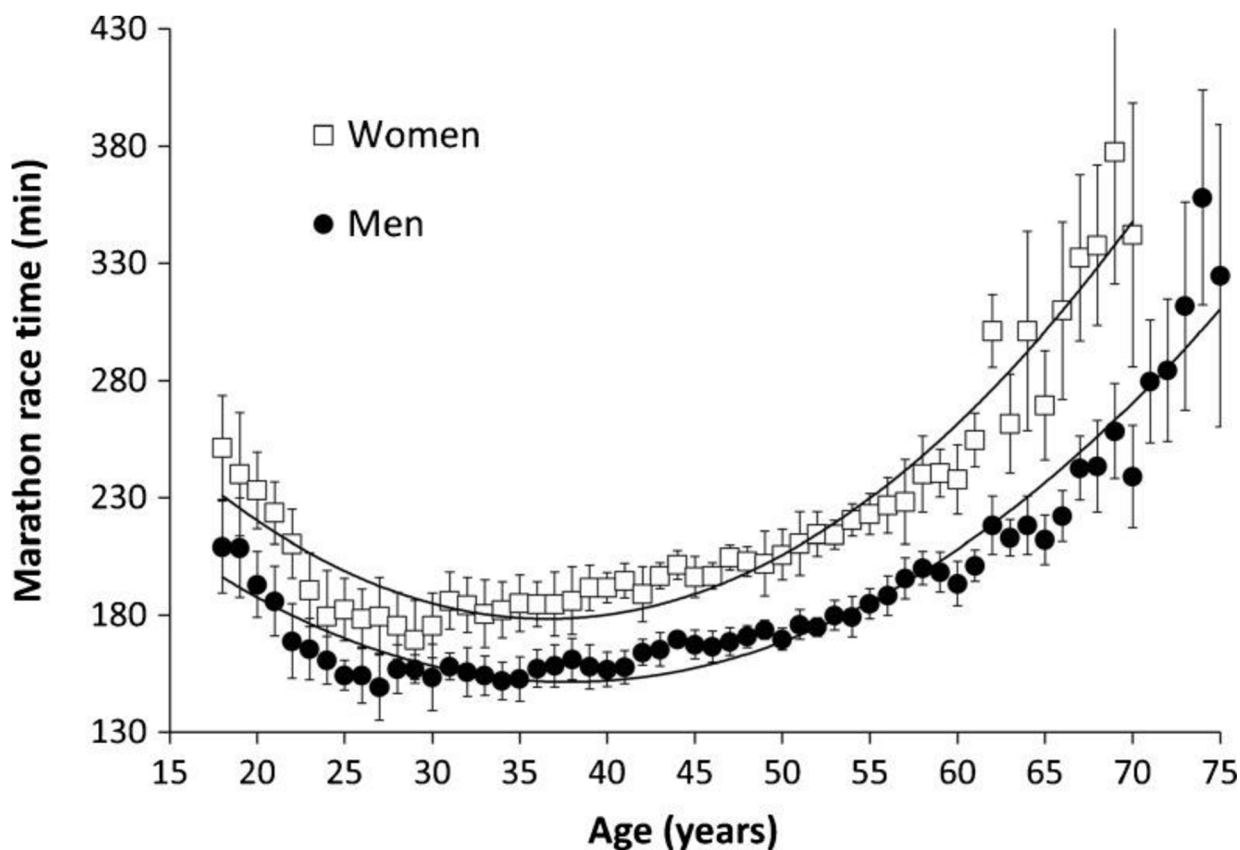
Stanovení cíle je psychologický konstrukt silně ovlivňující průběh a úspěšnost tréninku. Uznávaný kineziolog R. S. Weinberg uvádí, že klíčová vlastnost cíle je jeho měřitelnost, schopnost vyjádřit pokrok čísly. [5] Běžci používající měřitelné cíle mají průměrně mnohem lepší výsledky. Pokud si běžec řekne „poběžím, jak nejlíp mi to půjde“, pak neexistuje neúspěch. Každý výkon stačí a běžec nemá zpětnou vazbu. Studie prokázaly, že pravidlo „čím ambicióznější cíle, tím lepší výsledky“ neplatí mezi amatérskými, ani profesionálními atlety. [3, 4] Naopak stanovení velmi náročného cíle přineslo pro amatérské atlety menší zlepšení než snadno zvládnutelný cíl a více olympijských sportovců preferuje snadno dosažitelné cíle nad velmi náročnými. Zásadní důležitost volby ideální zátěže je ilustrována na následujícím obrázku.



Obrázek 2: Princip adekvátního tréninku (zdroj: [9])

Pro adekvátní personalizaci cíle je nutné brát v potaz klíčové, fyziologické faktory, a to: věk, pohlaví, fyzický stav a časové dispozice. Dále konkrétní náročnost běhu – členitost terénu a zdolané převýšení.

[6, Obrázek 3] Relevantní studie *The relationship between age and running time in elite marathoners is U-shaped* z roku 2014 zkoumá výsledky více než 40 000 běžců New Yorkského maratonu v letech 2010 a 2011. Studie ukazuje, že nejvýkonnější je člověk mezi 25. a 30. rokem života. Nejúspěšnější muž byl ve věku 27 let a nejúspěšnější žena ve věku 29 let. Před výkonnostním maximem se výkonnost zlepšovala průměrně o 4.4 % ročně u mužů i žen a po dosažení maxima klesala o 2.4 % u mužů a 2.5 % u žen ročně. Výkonnostní rozdíl mezi muži a ženami činí v průměru 18.7 % ve věkovém rozmezí a nejmenší rozdíl nastává ve věku 29 let, kde činí 10.2 %. Nutno podotknout, že průměrný účastník této studie není začátečník. Důvodem relevance je určení obecného trendu z velkého množství běžců, platícího pro populaci obecně.



Obrázek 3: Vliv věku a pohlaví na běh (zdroj: [6])

Další měření ukazuje vliv pohlaví na více běžeckých disciplín. [7] Měření na 8 odlišných běžeckých disciplínách potvrzuje, že výkonnostní rozdíl v pohlaví není specifický trend pro maraton, ale musíme s ním počítat pro celé spektrum.

Co se týče konkrétního fyzického stavu uživatele, trénink bude ovlivněn jeho BMI neboli Body Mass Indexem. BMI je index vyjadřující vztah mezi tělesnou hmotností a tělesnou výškou člověka. Sám o sobě však není plně vypovídající, jelikož pro velmi svalnatého kulturistu, který má mnoho kilogramů svalové hmoty by určil jeho fyzický stav např. na nadváhu, či obezitu. Pro relevanci tohoto údaje je nutné získat od uživatele údaj o jeho aktuální formě.

Další klíčový faktor, který musí být zohledněn je časová dispozice uživatele. Čím více dní v týdnu má uživatel možnost trénovat, tím lepší má potenciál pro zvolení a úspěšné dosažení ambicióznějšího cíle. Neznamená to ale, že pokud má čas na trénink 7 dní v týdnu tak bude trénovat 7 dní v týdnu. Je nutné do tréninku začlenit adekvátní regeneraci. [41] Dle WHO je pro dospělé jedince vhodné alespoň 150 až 300 minut středně náročné fyzické aktivity týdně, spolu se dvěma aktivitami posilující svaly. Na základě toho bude minimální počet tréninků týdně nastaven na tři. Tři tréninky týdně lze považovat za dostatečné úvodní

množství i pro začátečníky a také to může mít pozitivní vliv tím, že je to naučí disciplíně, která je pro dlouhodobé zlepšení naprosto klíčová.

Délka samotného tréninkového plánu je také velmi důležitý parametr. Těžko se určuje hranice, jak dlouhý trénink má pro uživatele smysl. Pro krátké běhy to může být pár týdnů, na maraton se někteří sportovci připravují i řadu měsíců, či roky. V rámci tohoto projektu bude tato hranice stanovena na 1 měsíc minimum a 1 rok maximum. Kratší trénink, než 1 měsíc člověku s fyzickým stavem znatelně nepomůže, ale nutit nováčka do dlouhého tréninku by naopak mohlo uživatele demotivovat. Tato hranice je ale subjektivní, jelikož každý člověk má specifické potřeby.

Na základě aktuálních schopností uživatele a požadovaného cíle lze určit potřebné procentuální týdenní zlepšení pro dosažení cíle. Pokud je uživatel začátečník a nezná svoji výkonnost, bude před začátkem tréninku požádán o testovací běh pro získání vstupních údajů.

Důležitý aspekt je délka tratě závodu. Jelikož je aplikace cílená převážně na začátečníky, nebude nabízet přípravu na delší vzdálenost, než půlmaraton (21 km). Delší, náročné závody, jako např. maraton už vyžadují komplexnější přípravu, než realisticky dokáže tento projekt nabídnout a pro začátečníka s během také není vhodné si dát jako první cíl hned maraton. Existují i sportovci běžající přes 100 km najednou, ale ti jsou na tak vysoké fyzické a znalostní úrovni, že by pro ně bylo používání aplikace bezpředmětné. Spodní hranice kondičního běhu bude nastavena na 2 km. Překážkové závody bývají kratší a běžecké závody pro veřejnost mají v dospělých kategoriích ve velké většině případů vzdálenost vyšší, než je tato hranice. Pro přípravu na sprint se hodí jiná, specifická příprava, které se tento projekt nevěnuje. V rámci zlepšení fyzické kondice začínajícího běžce je vhodné věnovat se aerobním aktivitám spíše delším než kratším. [41] Na základě doporučení WHO budou generované aktivity odpovídat minimálně 30 minutám fyzické aktivity v závislosti na charakteristice cviku. Pro dlouhé, vytrvalostní běhy není problém strávit na aktivitě 90 minut, zatímco čas intenzivního cvičení v posilovně bude kratší.

Trénink by měl také zohlednit terén tratě, pokud se jedná o závod s vysokým převýšením, je potřeba dbát vyšší důraz na vytrvalost.

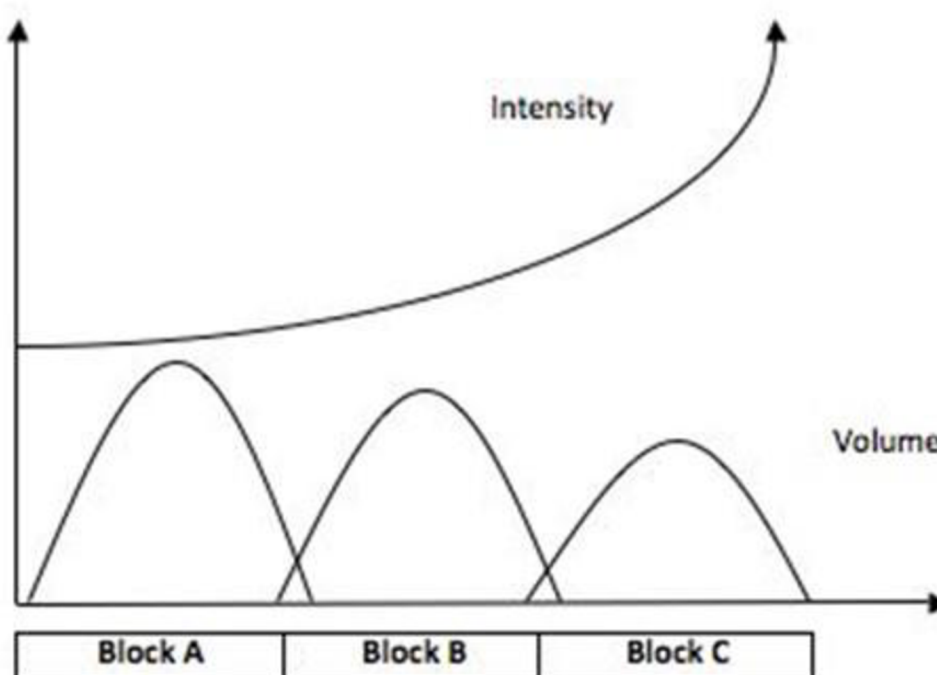
3.4.2 Progresivita tréninku a dosažení cíle

Zde je nutné určit a číselně vyjádřit očekávanou, minimální a maximální míru zlepšování. Tato hodnota se bude pro každého uživatele velmi lišit na základě jeho predispozicí a odhodlanosti. Cíl je hodnota, konkrétní snadno vyjádřitelný parametr.

Buď může mít uživatel vytyčený vlastní cíl, např. uběhnout závod na 5 km pod 25 minut, nebo může stanovit za cíl pouze danou vzdálenost uběhnout. V takovém případě vycházíme z definice pro běh a jogging. Jogging je mírný, nezávodní běh s kondičními a zdravotními účely. Tempo joggingu je stanoveno na rozmezí od 6 do 10 km/h. [8] Pro srovnání, průměrná chůze nabývá rychlosti 4 až 5 km/h. Stanoví-li tedy uživatel za cíl pouze vzdálenost a nikoli cílený čas, aplikace bude pro začátek počítat se spodní hranicí joggingu, tedy 6 km/h. Pravděpodobně bude jednat o případ, kdy je uživatel ve stavu, ve kterém tento úsek ještě není schopný uběhnout.

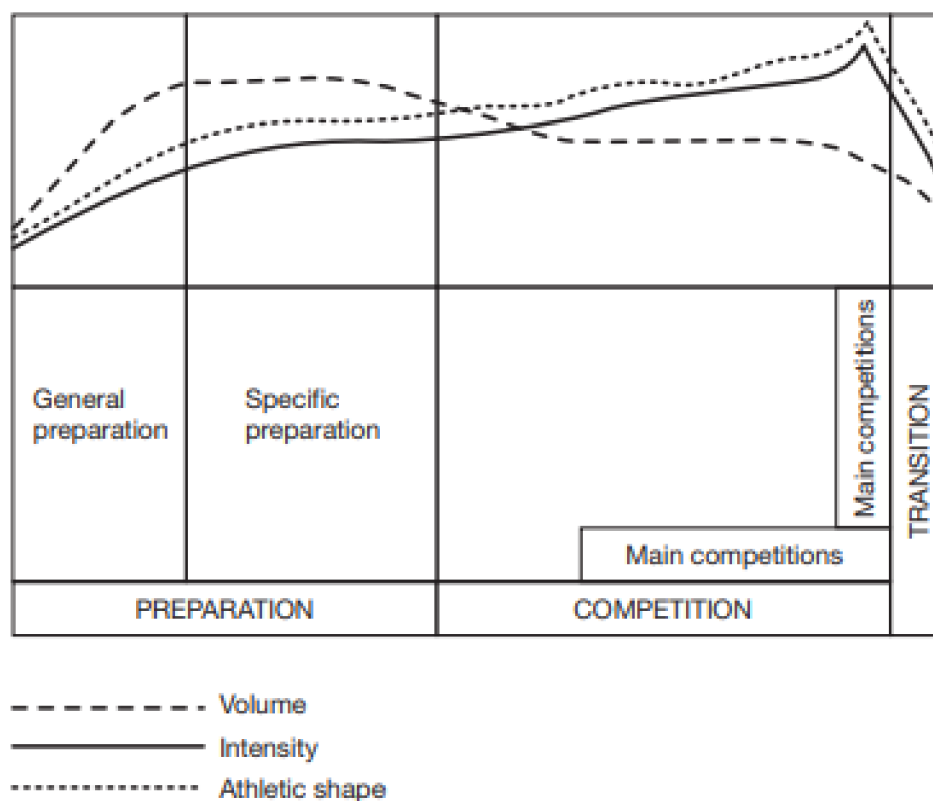
Problematika začátečníků v běhu je ale ještě o něco složitější. Pokud člověk, co nikdy neběhal, najednou začne s tvrdým tréninkem, nebude se zlepšovat, protože jeho svaly a zejména srdce nejsou na aktivitu zvyklé. Běh s velmi vysokou tepovou frekvencí tělu více škodí, než pomáhá a je častým problémem nejen začátečníků s nadváhou. Pro takového člověka je pro začátek vhodná pouze rychlá chůze na vzduchu, nebo zmíněný jogging kombinovaný s chůzí. [21] V ideálním případě se uživatel řídí měřenou tepovou frekvencí, pro kterou jde přesně určit rozmezí. Dle zdroje Journal of the American College of Cardiology se maximální tep u netréňovaného člověka dá určit vzorcem $(211 - 0.8 * \text{věk})$. [22] Ideální rozmezí pro běh je mezi 60 a 85 % této hodnoty pro začátečníka a o něco více pro vytrvalostně trénovaného běžce. Ideální tepová frekvence pro rozvoj vytrvalostních dovedností je 75 až 85 % maximální tepové frekvence, s touto hodnotou bude pracovat uživatel, který už má alespoň základní zkušenosti s během. Uživatelům, kteří si nemají, jak změřit tepovou frekvenci při výkonu nezbyvá, než běžet podle pocitu a snažit se spíše o klidnější tempo.

Tréninkový plán by se měl postarat o to, aby uživatel pracoval s postupně navyšující se zátěží a aby tato zátěž byla pro tělo adekvátní. Pro tvorbu dlouhodobého tréninkového plánu je možné použít jeden ze dvou základních druhů periodizace tréninku, a to buď tradiční, nebo blokový. [11] Blokovaná periodizace je přístup rozdělení tréninku do bloků, kde každý blok obsahuje cyklus velmi intenzivních tréninků s úzkou specializací na 2 až 3 konkrétní vlastnosti. [13] U blokované periodizace je běžná několikadenní regenerační pauza mezi bloky tréninku. Tento princip tréninku dle mnoha experimentů přináší lepší výsledky než běžná tradiční periodizace. [13] Na druhou stranu ale bývá náročnější a vyžadující mnohem větší disciplínu díky velmi vysoké intenzitě tréninků. Dlouhé pauzy v trénování nejsou pro začátečníka vhodné, protože ho nevedou k pravidelnosti a riskuje se, že po dlouhé pauze už se začátečník nebude chtít k intenzivnímu tréninku vrátit. Tento princip je vhodný pro profesionální a zkušené atlety, kteří jsou zvyklí intenzivně trénovat, aktivně regenerovat a jsou disciplinovaní.



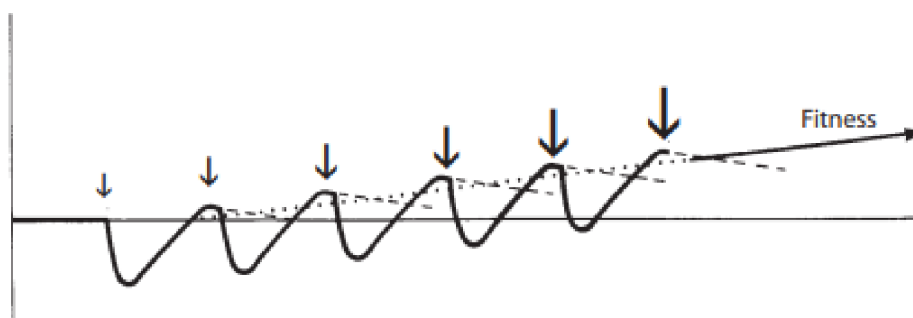
Obrázek 4: Princip blokové periodizace (zdroj: [9])

[12, Obrázek 5] Pro začátečníka bude lepší využít tradiční neboli lineární periodizaci tréninku. Tradiční periodizace je založena na principu postupného snižování objemu tréninku a zvyšování intenzity. Aplikace v tréninku tedy vychází z tradiční periodizace, v některých případech obohacenou o „taper phase“, neboli fázi snižování zátěže. Tato fáze je vhodná u těch nejdelších tratí, většinou začíná okolo 2 týdnů před závodem a jde o snížení objemu i intenzity tréninku. Obecný princip tohoto přístupu k tréninku je velmi jednoduchý: začíná s budováním vytrvalosti jedince, která je pro závod klíčová, bez ní by to nešlo. Postupem tréninku dochází k aktivnímu trénování potřebného závodního tempa a postupnému snižování délky tréninku. Tato fáze je však možná až v momentu, kdy jsme si jisti, že má závodník dostatečnou výdrž. Těsně před koncem přípravy u dlouhých tratí dojde k odpočinkové fázi. Tato fáze je pro závodníka přínosnější než intenzivní trénink, tělo do závodu nastupuje řádně zregenerované a je silně zredukováno riziko zranění těsně před závodem, které by závod znemožnilo.



Obrázek 5: Křivka množství a intenzity tréninku – lineární trénink (zdroj: [9])

[Obrázek 6] Důležité je, aby křivka navyšování objemu práce během fáze budování vytrvalosti nebyla příliš strmá, což je těžké stanovit. [4, 6] Zatímco atleti běžající maraton se zlepšují průměrně do **4,4 % ročně**, u mladých atletek ze studie *The Effect of Goal Difficulty and Goal Orientation on Running Performance in Young Female Athletes* se počítalo s odpovídajícím dosažitelným zlepšením **2,5 % týdně**. [Obrázek 2, 4] Vyžadování příliš velkého zlepšení by vedlo k demotivaci a zhoršení míry zlepšování. Pokud si uživatel zvolí příliš ambiciózní cíl, aplikace mu nedovolí takový trénink započít, s tím, že mu uvede důvod, např. „Toto tempo neodpovídá aktuálním výsledkům, zvolte pomalejší tempo“. Maximální míra zlepšení se bude odvíjet od počtů dnů tréninku. Čím častěji bude uživatel trénovat, tím vyšší bude možná míra zlepšení. To zachová možnost uživateli, aby si sám zvolil vlastní výzvu, ale zároveň mu nepovolí trénink, který by byl určitě nedosažitelný. Tento údaj je subjektivně stanoven, nelze určit přesný maximální koeficient zlepšení kvůli individualitě jedinců.



Obrázek 6: Vliv soustavného tréninku (zdroj: [9])

3.4.3 Regenerace a bezpečnost

Zejména u začátečníků musí být kladen důraz na osvojení správných bezpečnostních a regeneračních návyků. [4] Studie *Prospective study of beginner running groups: psychological predictors and outcomes of participation* uvádí, že 33 % ze všech účastníků studie utrpělo zranění během devítitýdenního tréninkového běžeckého programu pro začátečníky. Příčin zranění může být mnoho: špatná technika, nedostatečné protažení, nevhodná obuv, příliš náročné tempo, a další. U každé aktivity, ať půjde o běh, posilování, či plavání budou k dispozici relevantní bezpečnostní upozornění. Může jít například o upozornění týkající se zahřátí před výkonem, správné techniky, nebo strečinku po výkonu.

Velmi důležitý aspekt přípravy na závod je kvalitní hydratace organismu a přísun potřebných živin. Aplikace bude vypočítávat vhodný přísun vody pro daný den. Tato hodnota bude personalizována, a to na základě váhy uživatele. Obecně se doporučuje 30 až 45 ml vody na 1 kg tělesné váhy denně. Pro zdravého, dospělého člověka bude stanovený přísun na: **tělesná váha v kilogramech * 0,033 + 0.2** až 1 litr vody v závislosti na náročnosti tréninku pro daný den.

Regeneraci dělíme na aktivní a pasivní. Pro náročný trénink je vhodné využít oba druhy. Pasivní regenerace je aktivita, kdy člověk nevyvíjí žádnou fyzickou aktivitu, tedy např. spánek (ideálně 7 až 9 hodin pro dospělého), masáž, nebo pobyt v sauně či ve vířivce. Formu pasivní regenerace nebude aplikace uživateli určovat, aby si každý zvolil, co mu vyhovuje. [14] Aktivní regenerace je většinou lehká aerobní aktivita prováděná při aktivitě okolo 60 až 70 % maximální tepové frekvence. Důležitost aktivní regenerace spočívá ve zvýšení průtoku krve ve svalech. Během náročných tréninků se ve svalech hromadí laktát. Laktát je konjugovaná báze kyseliny mléčné a v těle způsobuje pocit svalové únavy, někdy přecházející až ve svalovou bolest. Zvýšením krevního průtoku ve svalech dochází k urychlení odplavení laktátu. Vyplavování laktátu ze svalů je proces, který většinou trvá

několik hodin po náročném tréninku, proto je aktivní regeneraci vhodné provádět během této doby. Další výhodou aktivní regenerace je možné zlepšení mobility a zlepšení kondice. Vhodná aktivní regenerace je např. jogging, kompenzační cvičení (pomalé, kontrolované pohyby), nebo pomalé plavání (vhodné pro klouby, zároveň dochází k podpoře cirkulace krve a protažení svalů). Obecně je regenerace velmi důležitá část tréninku, která by měla být aplikací podporována.

3.5 Princip tvorby jídelníčku

Před tvořením jídelníčku bude uživatel požádán o určení jeho stravovacích preferencí. Může se jednat o alergie na určité ingredience, nebo osobní preference, např. vegetariánství. Preference se stanoví vytvořením seznamu zakázaných ingrediencí. Jídla, která tyto ingredience obsahují, nebudou uživateli doporučována.

3.5.1 Energetický příjem

Jedna ze základních měřitelných vlastností jídla je jeho energetická hodnota. Měří se v kilojoulech (kJ) a kilokaloriích (kcal). Podle přísunu kalorií určujeme kalorický deficit a přebytek, určený za základě příjmu a výdaje energie.

Většina energie se během dne spaluje pomocí BMR. Basal Metabolic Rate, česky bazální rychlost organismu, je celkové množství energie vydané v klidovém stavu v teplotně neutrálním prostředí na lačno. Tato energie je vydána pouze na provoz orgánů a svalů a je potřebná pro zachování životních funkcí. [15] Nelze přesně určit, ale dá se dobře odhadnout a tvoří 60 až 75 % denního kalorického výdaje. Pro získání optimálního příjmu kalorií se hodnota BMR vynásobí koeficientem fyzické aktivity (1,22 až 1,9) v závislosti na fyzické aktivitě jedince.

Pro hubnutí je nutné přijímat méně kalorií, než je optimální příjem. Čím větší je kalorický deficit, tím více tělo spotřebuje z tukových zásob, ale příliš velký kalorický deficit vede k fyzickým i psychickým poruchám a neracionálnímu chování organismu. Pro efektivní hubnutí se doporučuje přijímat zhruba o 500 méně kalorií, než je optimální kalorický příjem. Nabírání hmoty je snazší než ztrácení, lze nabírat i s větším kalorickým přebytkem. [15, 16]

Aplikace vychází z **Mifflin-St Jeorovy rovnice**. Jedná se o nejpoužívanější a pravděpodobně nejpresnější rovnici pro výpočet BMR.

Pro muže platí: **$BMR = 10 \times W + 6,25 \times H - 5 \times A + 5$**

Pro ženy platí: $BMR = 10 \times W + 6,25 \times H - 5 \times A - 161$

W = váha v kg, H = výška v cm, A = věk v letech

Trénink se bude zaměřovat na dostatek energie pro dané fyzické aktivity a pokud bude uživatel nad hranicí doporučeného BMI, bude pracovat s lehkým kalorickým deficitem. Pokud se bude uživatel připravovat na překážkový běh, na který je potřeba velká fyzická síla, bude trénink pracovat s lehkým kalorickým přebytkem a jídelníčkem bohatým na bílkoviny, pro zajištění dostatečných zdrojů pro růst svalů.

3.5.2 Sacharidy

Sacharidy (po staru karbohydráty) jsou velmi obsáhlou a důležitou složkou vytrvalostních běžců. Jedná se o primární zdroj glukózy, která je svaly používána jako palivo. Je vhodné zvolit kvalitní sacharidy (ovesné vločky, brambory, batáty, čočka, ...) s nízkým glykemickým indexem, aby se zabránilo příliš rychlému přísunu glukózy do krevního řečiště. Pro vytrvalostní běh je lepší pomalé, pozvolné uvolňování. Vyšší obsah vlákniny v jídle může také zpomalit uvolňování glukózy. [18] Dle WHO jsou nejvhodnější zdroje sacharidů celá zrna, zelenina, ovoce a luštěnina. [17] Příjem sacharidů má pozitivní vliv na dlouhé tratě, od délky běhu 90 minut. [19] Obecně mají sacharidy tvořit mezi 50 a 70 % stravy. Jsou vhodné ke konzumaci před výkonem, během výkonu i po výkonu. Po výkonu je doporučena konzumace sacharidů společně s proteiny. [23] Pro představu, etiopští vytrvalostní běžci převážně dodržují dietu bohatou na sacharidy (cca 64 %). Tato dieta obsahuje také 23 % tuků a 12 % bílkovin.

3.5.3 Bílkoviny

Bílkoviny neboli proteiny jsou další podstatnou složkou stravy. Přispívají k regeneraci a jsou používány pro opravu svalových vláken. Během náročné fyzické aktivity se ve svalech tvoří mikrotrhliny a proteiny jsou zodpovědné za jejich opravu. Dlouhodobý vytrvalostní trénink může vést ke ztrátě svalové hmoty, dostatek bílkovin pomáhá svalovou hmotu udržovat. Před výkonem je vhodný pouze lehký zdroj bílkovin, nejvíce by se mělo konzumovat ihned po výkonu.

[20] Více studií se shodlo na výsledku, že běžci potřebují více bílkovin, než je doporučovaný denní příjem dle instituce United States Food and Drug Administration (0.8 g na 1 kg hmoty – tato hodnota stačí pro člověka bez fyzické aktivity). Na konkrétním čísle

se neshodly, hodnoty se pohybují od 0.94 do 1.37g/kg pro běžce. Sportovci zvedající těžké váhy (s těmi aplikace taky musí počítat kvůli překážkovému běhu) mají ještě vyšší potřebu proteinu a to 1.4 až 1.8 g/kg/den. Aplikace bude dbát důraz na příjem bílkovin hlavně po náročných trénincích pro regeneraci a růst svalů.

3.5.4Tuky

Tuky se dělí na více druhů, do jídelníčku rozhodně nepatří trans-tuky. Tyto tuky se vyskytují převážně v průmyslově vyrobeném jídle a jsou zdraví škodlivé. Podporují rozvoj kardiovaskulárních chorob a zvyšují hladinu LDL cholesterolu („špatný“ cholesterol, způsobující usazeniny na cévních stěnách) v krvi.

Nasyčené mastné kyseliny jsou pro tělo potřebné, ale pouze v malém množství. Měly by tvořit pouze zlomek z celkového příjmu tuků. Ve větším množství zvyšují hladinu LDL cholesterolu a přispívají k hormonální nestabilitě a diabetes. Druh tuků, na který se zaměříme jsou nenasycené mastné kyseliny. Sportovec by měl přijímat 25 až 35 % energie z tuků, a to převážně z nenasycených mastných kyselin. Tyto tuky snižují hladinu LDL cholesterolu a zvyšují hladinu HDL cholesterolu, který je pro tělo, zejména srdce přínosný. Dále pomáhají zlepšit citlivost na inzulin, což vede k regulaci cukru v krvi. Některé nenasycené mastné kyseliny, např. Omega3 (ryby, kapusta, ořechy, chia semínka) hrají podpůrnou roli v regeneraci. Obecným zdrojem nenasycených mastných kyselin jsou ořechy, olivový olej, avokádo, semínka a ryby.

3.5.5Odměňování

Mnoho začínajících sportovců skokově přechází z nezdravého životního stylu a dodržování kvalitní stravy může být velmi náročné a vyžadující silnou disciplínu. [24] Existuje populární dietní pravidlo 80/20 stanovující, že pro úspěšnou dietu stačí, aby 80 % přijímaného jídla splňovalo dietní požadavky a zbylých 20 % si uživatel zvolí sám, čistě na základě jeho chutí. Je to prevence před selháním diety a před deprivací. Malé množství potencionálně nezdravého jídla je mnohem menší zlo než kompletní upuštění od diety. Aplikace bude s tímto principem pracovat a také bude odměňovat uživatele v podobě občasného, jednorázového upuštění od jídelníčku.

3.5.6 Shrnutí jídelníčku

Základní požadavek na jídelníček je umožnění stravovacích preferencí z důvodů alergií a osobních preferencí. Dále je nejpodstatnější údaj denní příjem a výdej kalorií. Podle tohoto údaje dochází k hubnutí, nebo nabírání tělesné hmoty. Aplikace by měla přizpůsobovat doporučení kalorií BMI uživatele. Jídelníček by obecně měl obsahovat kvalitní potraviny. Měl by se zaměřit na velký přísun sacharidů, vhodných jako palivo pro dlouhé, vytrvalostní aktivity a dále na dostatečné množství bílkovin pro umožnění růstu svalů. Na základě tělesné hmotnosti a fyzické aktivity lze stanovit orientační doporučený příjem vody. Odměna uživatele ve formě vlastní volby jídla v malém rozsahu určitě nebude škodit, či brzdit postup, naopak může uživatele motivovat a odměnit za konzistentní výkony.

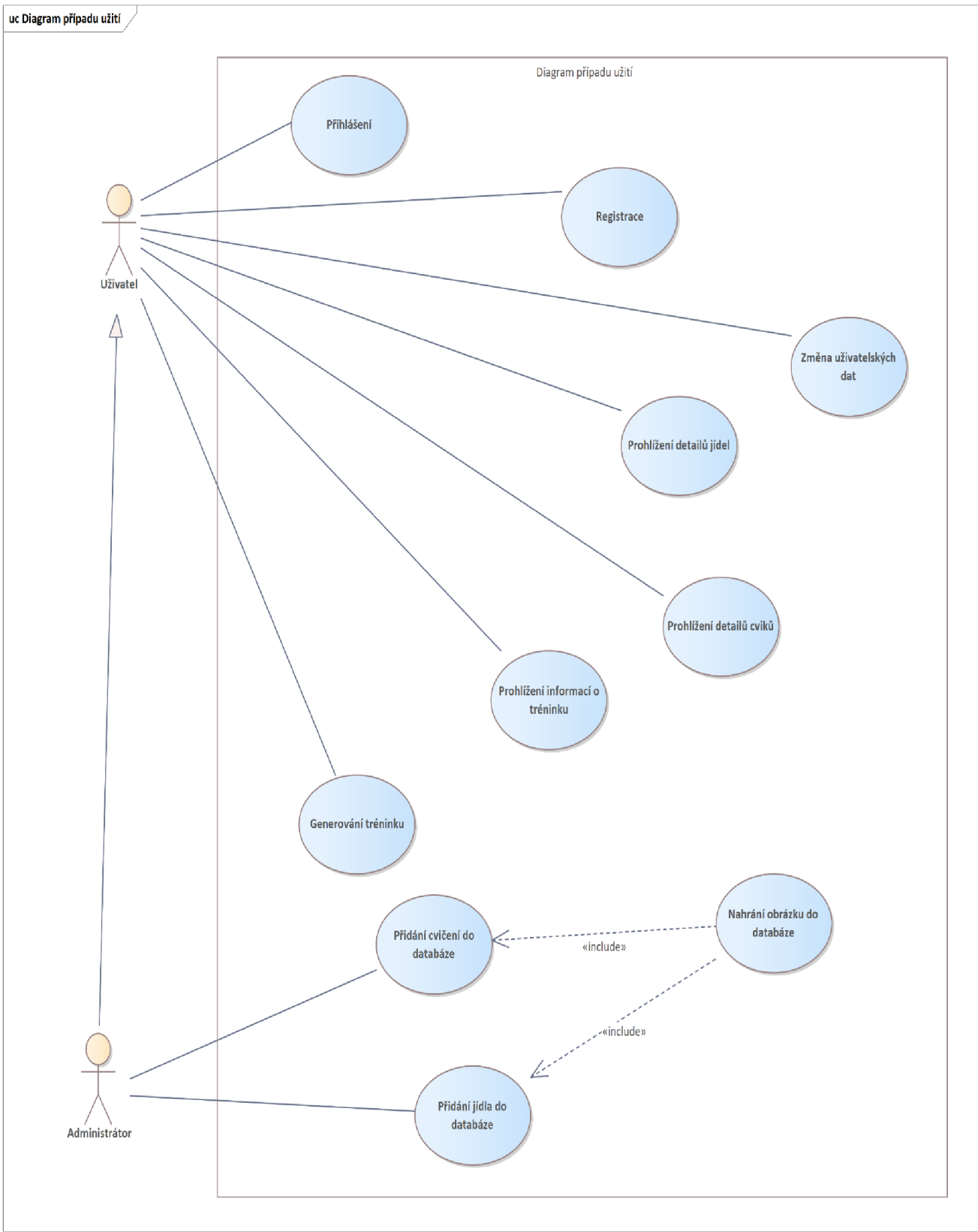
4 Návrh a vývoj

Tato kapitola definuje a popisuje použité technologie a vysvětluje principy zabezpečení a komunikace v rámci aplikace.

4.1 Funkcionalita

Webová aplikace by měla poskytovat možnost tvorby uživatelského účtu a přihlášení do účtu. Uživatel by měl mít možnost změnit své osobní údaje (přezdívka, heslo, hmotnost, ...). Dále má aplikace poskytovat možnost volby tréninku a jídelníčku na míru uživateli. Web by měl být stručný, přehledný, dostupný a s náležitým bezpečnostním upozorněním – aby případný uživatel počítal s tím, že se jedná o studentskou práci, a ne odborná lékařská doporučení. S tímto upozorněním bude muset souhlasit každý uživatel při tvorbě účtu. Dále musí být web bezpečný, viz kapitola 5.5 - Zabezpečení. Jelikož aplikace uchovává osobní data uživatelů, musí také splňovat obecné požadavky na ochranu osobních údajů neboli GDPR (general data protection regulation). Web by se měl řídit i obecnými pravidly pro tvorbu webu: implementace responzivního designu, intuitivní navigace atd.

Na následujícím obrázku je zobrazen diagram případu užití. Tento diagram definuje běžné akce, které projekt nabízí. Pracuje se dvěma základními typy uživatelů a to „User“ a „Administrator“, v kódu zkráceně „Admin“. Dostupná funkcionalita není nijak rozsáhlá, ale pro účel projektu to stačí. Kromě uživatelských operací má administrátor pravomoc přidávat do databáze nové cviky a jídla, ze kterých jsou samotné tréninky tvořeny.



Obrázek 7: Diagram případu užítí (zdroj: vlastní)

4.2 Uživatelské rozhraní

Vzhled a prezentace webu odpovídá jeho cílové skupině, a to začínajícím sportovcům ve věku od 18 let. Webové stránky používají barvy a prvky, které kombinují sport a přírodu (většina běhů i gladiator race probíhají někde v přírodě). Toto téma je částečně subjektivní a každý si může asociovat jiné barvy pro jiné podměty, ale pro přírodu je klasicky asociována zelená (louky, lesy). Pro sport může být asociována např. žlutá (energická, pozitivní, rychlá). Proto bude web stylizován do těchto 2 barev.

Je vhodné, kdyby měl uživatel všechny své údaje k dispozici na jednom místě, v sekci profilu. Dále je na jedné stránce zobrazen dopodrobna celý aktuální trénink s datem začátku, datem konce a s cílem (závodem). Také je k dispozici samostatná stránka pro zobrazení konkrétního dne s cvičeními, jídly a údaji o spálených a přijatých kaloriích. Pro konkrétní cvičení, nebo jídlo by měl být zobrazen jeho detailní popis, obrázek, případně odkaz na video. U cvičení jsou evidovány cílené svalové partie a u jídel určité tagy příslušnosti (např. bez cukru, veganské, snídaně, oběd, nízkokalorické, ...).

4.3 Použité technologie

4.3.1 Backend

Backend je část aplikace, která je zodpovědná za zpracování dat, vykonávání aplikační logiky a interakcí s databází apod. Zjednodušeně se jedná tu část aplikace, kterou uživatel nevidí.

[32] Jako backendová technologie byl zvolen Spring Boot – nadstavba nad open-source frameworkem Spring. Spring je framework, sloužící k usnadnění vývoje enterprise aplikací a je vyvíjen společností VMware. [34] Nabízí např. „*Inversion of Control*“, který uvolňuje pevné vazby mezi objekty v Javě (přesouvá odpovědnost za vznik z programátora na framework). Dále nabízí moduly Spring Data (abstrakce nad daty, podpora JDBC, JPA), Spring Security (zabezpečení aplikace autentizací, autorizací a ochrana před útoky) Spring Cloud (vývoj cloud-native aplikací) a další. Spring Boot je rozšíření tohoto frameworku, umožňující snadný vývoj aplikací.

[35] Jeho hlavní výhodou je, že je založen podle paradigmatu „*convention over configuration*“. Vývojář díky tomu nemusí sám explicitně definovat všechny konfigurace, vše je nakonfigurováno předem, podle zavedených konvencí. [31] Mezi další výhodou a důvody zvolení právě tohoto frameworku jsou jeho vestavěné servletové kontejnery (např,

Apache Tomcat) a také velká komunita uživatelů Spring Bootu, která pomáhá při řešení různých problémů.

Mezi nevýhody Spring Bootu patří určité množství nevyužitých závislostí, což má za následek zvýšení velikosti a zpomalení startu běhu, dále omezená flexibilita a velká závislost na Spring ekosystému. [33] K dalšímu zpomalení může vést používání Java Virtual Machine, která s sebou přináší další výhody i nevýhody. Další úroveň abstrakce např. umožňuje spuštění kódu na různých platformách bez potřeby změn v kódu.

Jako alternativu Spring Bootu by šlo využít např. NodeJS – multiplatformní, JavaScriptové runtime prostředí, také s velkou komunitou uživatelů. Toto prostředí nabízí velmi vysokou výkonnost, díky využívání tzv. událostmi řízeného, asynchronního modelu a je v porovnání se Spring Bootem jednodušší. Jde o prostředí vhodné pro tvorbu malých, až středních aplikací.

Spring Boot byl zvolen z velké části kvůli osobní preferenci a znalostech Javy. Tato volba byla také motivována tím, že Spring je robustní, v praxi hojně využívaný ekosystém a pro vývoj backendu pro webovou aplikaci tohoto rozsahu se perfektně hodí. Nevyužité závislosti a pomalejší start se u projektu se spíš menším rozsahem příliš neprojeví. Při vývoji nebyl výkon Springu rozhodně překážkou. Jediná část aplikace, která měla obtíže s pomalou odezvou byl endpoint „training/getActiveTraining“, kde docházelo k přenosu většího objemu dat. To bylo vyřešeno vrácením pouze části tréninku, která je potřebná pro zobrazení aktuálních informací o tréninku (požadavek na tento endpoint nevrací údaje o všech den tréninku, ale pouze nadcházející týden).

4.3.2 Frontend

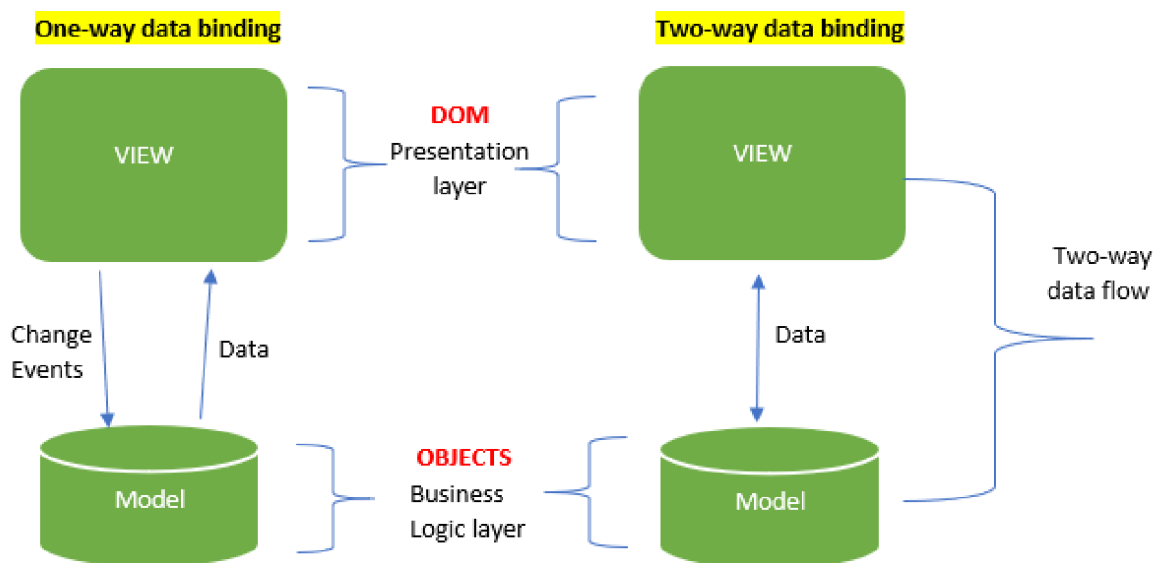
Frontend je prezentační vrstva aplikace neboli část aplikace, sloužící k interakci s uživatelem. V případě tohoto projektu jde o webové stránky.

Jako frontendová technologie byl zvolen React. Jde o JavaScriptový framework, určený pro tvorbu uživatelského rozhraní, vyvíjený společností Meta. Jedná se také o open-source projekt.

[37] Mezi hlavní rysy a výhody Reactu patří skládání uživatelského rozhraní ze znovupoužitelných komponent, které mohou nabývat stavů („states“). Další výhodou Reactu je podpora JSX. [36] JavaScript XML je rozšíření syntaxe JavaScriptu, které umožňuje kombinovat JavaScript kód s HTML, pomocí konverze HTML tagů na React elementy. Používání tohoto rozšíření může mít pozitivní vliv na SEO.

[37] React obecně nemá žádné podstatné nevýhody nebo omezení, za zmínku stojí pouze rychlé tempo vývoje. React jako ekosystém se rychle vyvíjí, dochází k častým změnám a aktualizacím. To způsobuje i problém s dokumentací, která není dostatečně dobře a aktuálně zpracována.

Jako alternativa pro React by šel využít např. Angular. Angular 2+ je TypeScriptový framework vytvořený Googlem. [38] Na rozdíl od Reactu aplikuje princip „*two-way data binding*“, kde data proudí obousměrně mezi uživatelským rozhraním a modelem dat. V Reactu v základu funguje proud dat z rodičovské komponenty na potomka, ale *two-way data bindingu* lze také dosáhnout např. použitím externích knihoven. Rozdíl je znázorněn na následujícím obrázku. Angular přichází i s odlišným přístupem k DOM (Document Object Model). Zatímco React používá koncept virtuálního DOM neboli abstrakci nad reálným DOM, v Angularu se pracuje s reálným DOM. To vyplývá s z použití *two-way data binding*, kde při každé změně dat dochází k manipulaci s DOM, za účelem aktualizace uživatelského rozhraní.



Obrázek 8: Porovnání one-way a two-way data binding (zdroj: [39])

Vývoj webu v Reactu byl zpočátku náročný z důvodu neovládání základních principů tohoto frameworku, ale postupně se vše vyřešilo a React se osvědčil. Hooky `useState` a `useEffect` a `useParams`, které jsou používány napříč celým webe zjednodušují práci a Material UI komponenty a React knihovny, jako `18n`, `React-router` a další dělají soustu věcí za vývojáře. U Material UI je výhoda definování jednotného stylu, který je pak konzistentní

napříč všemi komponentami webu. Hlavním důvodem zvolení Reactu byla snaha o lepší poznání a naučení se použití této technologie. Jedná se o komplexní framework, vyvíjený obří společností, hojně využívaný pro vývoj webových stránek, což z něj dělá skvělou volbu.

4.3.3 Databáze

Definice databáze je „organizovaná kolekce strukturovaných informací, nebo dat“. Každá aplikace, která jakkoli zpracovává data je musí ukládat do nějaké databáze. Lze pro to využít různé databázové systémy.

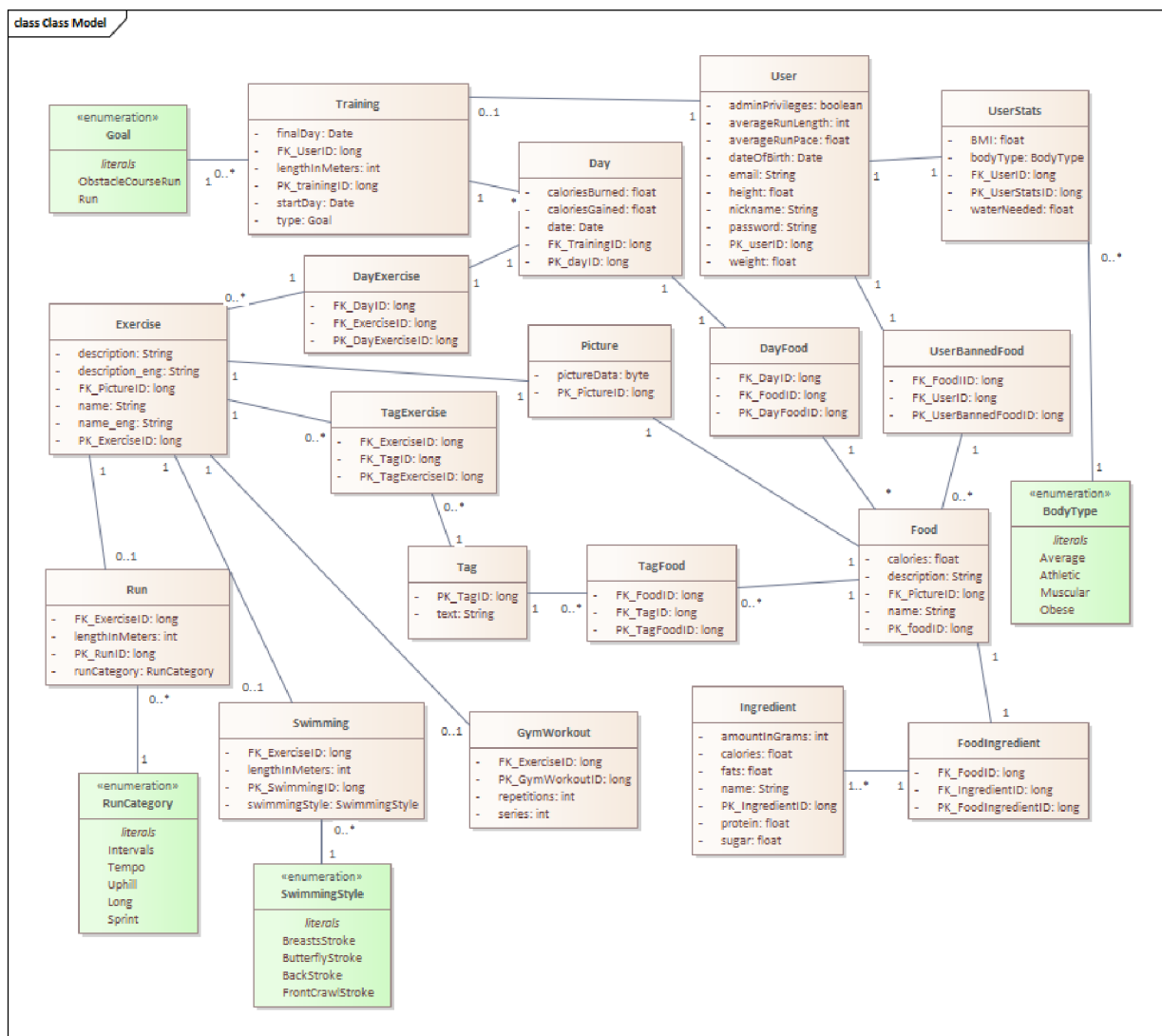
Pro tento projekt byl zvolen PostgreSQL, zejména kvůli pozitivním zkušenostem z minulého projektu. PostgreSQL, zkráceně Postgres, je open-source objektově-relační databázový systém, na jehož vývoji se podílí globální komunita firem a jednotlivců. Jde o vhodný systém pro malé i velké projekty.

[40] Jedná se o velmi spolehlivý a odolný systém, mezi jeho výhody patří např. podpora ACID transakcí, ukládání geografických objektů, nebo podpora mnoha rozšíření. Pro správu PostgreSQL databází existuje software s názvem PgAdmin, který je velmi přehledný a usnadňuje práci s databázemi. V tomto softwaru byly během vývoje zobrazovány aktuální data databáze a také sloužil k změnám v databázi (přidávání/úprava tabulek během vývoje). Přimo v databázi nebyly definovány databázové procedury, transakce, či funkce. O tuto funkcionalitu se ze 100 % postaraly rozhraní JpaRepository, viz kapitola 4.4.1 - Komunikace vrstev.

[40] Hlavní nevýhodou PostgreSQL je fakt, že se nejedná o jeden z nejpoužívanějších systémů. Díky tomu není možné PostgreSQL integrovat do mnoha aplikací, nebo technologií obecně. Další nevýhodou tohoto systému je menší výkon než některé konkurenční databázové systémy. Např. systém MySQL je rychlejší/výkonnější a také je mnohem častěji podporován externími aplikacemi. Pro tento projekt by byl systém MySQL také o vhodné řešení, ale např. otázka rychlosti není pro tak malý projekt klíčová.

Následující obrázek představuje UML diagram tříd, zobrazující schéma databáze použité v projektu. Jde o abstraktní návrh, neobsahující PostgreSQL datové typy, ale pouze základní reprezentaci v datových typech Javy.

Tvorba databáze byla jednoduchá, vychází z SQL scriptu, který byl vygenerován z DDL diagramu v aplikaci Enterprise Architect. Volba PostgreSQL byla dobrým řešením, ale využití jakéhokoli jiného databázového systému z těch nejpoužívanějších by pravděpodobně přineslo stejný výsledek.



Obrázek 9: Ukázka schéma databáze (zdroj: vlastní)

4.3.4 Server

Server v rámci tohoto projektu slouží k hostování a provozu webové aplikace. Server je poskytnut univerzitou Hradec Králové a jedná se o Ubuntu Server 22.04.4 LTS s přidělenou operační pamětí 6 GB a úložištěm pevného disku 60 GB. Verze Ubuntu Server 22.04 „Jammy Jellyfish“ je distribuce operačního systému Ubuntu s tzv. Long Term Support. Verze s LTS dostávají bezpečnostní aktualizace a opravy chyb ještě dlouhou dobu po vydání. Server používá Linux Kernel 5.15. Jde o virtualizovaný server pomocí softwaru VMware. Tento software umožňuje efektivní využití hardwarových prostředků serveru. Na jednom fyzickém serveru totiž může běžet více virtuálních serverů, kde každý má své vlastní určené parametry. Nasazení aplikace na tento server slouží pro testovací a vývojové účely. Nejde o produkční prostředí.

Alternativa tohoto řešení by mohla být např. použití jiné distribuce Linuxu (např. Debian, nebo Fedora), použití jiného operačního systému (např. FreeBSD, nebo Windows Server), nebo použití některé z cloudových platform (např. AWS, nebo Microsoft Azure).

4.4 Architektura projektu

Celý projekt by šel považovat za implementaci třívrstvé architektury. [47] Tato architektura rozděluje aplikaci do tří vzájemně oddělených logických vrstev a to prezentační, aplikační a datová. Každá tato vrstva plní v aplikaci specifickou úlohu. Datová vrstva je reprezentace informací, se kterými aplikace pracuje. V případě tohoto projektu jde o PostgreSQL databázi. Prezentační vrstva slouží k interakci s koncovým uživatelem. Tato vrstva zobrazuje uživateli informace a zároveň je od něj i čerpá pomocí interaktivních prvků, jako jsou tlačítka, formuláře, textová pole apod. V rámci tohoto projektu jde o webové stránky. Aplikační vrstva je prostředník mezi těmito dvěma vrstvami, v tomto případě jde o Spring Boot backend. Reaguje na události, přicházející od uživatele z view, zpracovává data a předává je modelu. Tato vrstva obsahuje business logiku aplikace. Klíčová charakteristika této architektury je, že prezentační a datová vrstva spolu nikdy nekomunikují, veškerá komunikace jde vždy přes aplikační vrstvu.

4.4.1 Komunikace vrstev

Samotná komunikace mezi vrstvami probíhá pomocí dvou standardů. Spring Boot komunikuje s databází pomocí Hibernate a s webem pomocí RESTful API.

Hibernate je součástí modulu Spring Data JPA. Java Persistence API neboli Jakarta Persistence je standard umožňující objektově relační mapování v Javě. Kromě Hibernate existují ještě další implementace tohoto standardu, např. EclipseLink nebo OpenJPA. V praxi Hibernate mapuje Java objekty a jejich vztahy na entity v relační databázi pomocí mapovacích souborů, nebo anotací. V tomto projektu jsou využity anotace. V podadresáři model jsou ve třídách definovány databázové objekty pomocí anotací „@Entity“, „@Table“, „@Column“, „@Id“ a dalších. S PostgreSQL je Hibernate kompatibilní. Nastavení připojené databáze je v projektu v konfiguračním souboru application.properties v podadresáři resources. Zde jsou klíčové konfigurace spring.datasource a spring.jpa.

[48] Spring také poskytuje rozhraní JpaRepository, což je nástroj, který usnadňuje dotazování na databázi. JpaRepository poskytuje CRUD operace nad databázovými entitami a také automaticky implementuje pojmenované metody pro příslušné databázové entity.

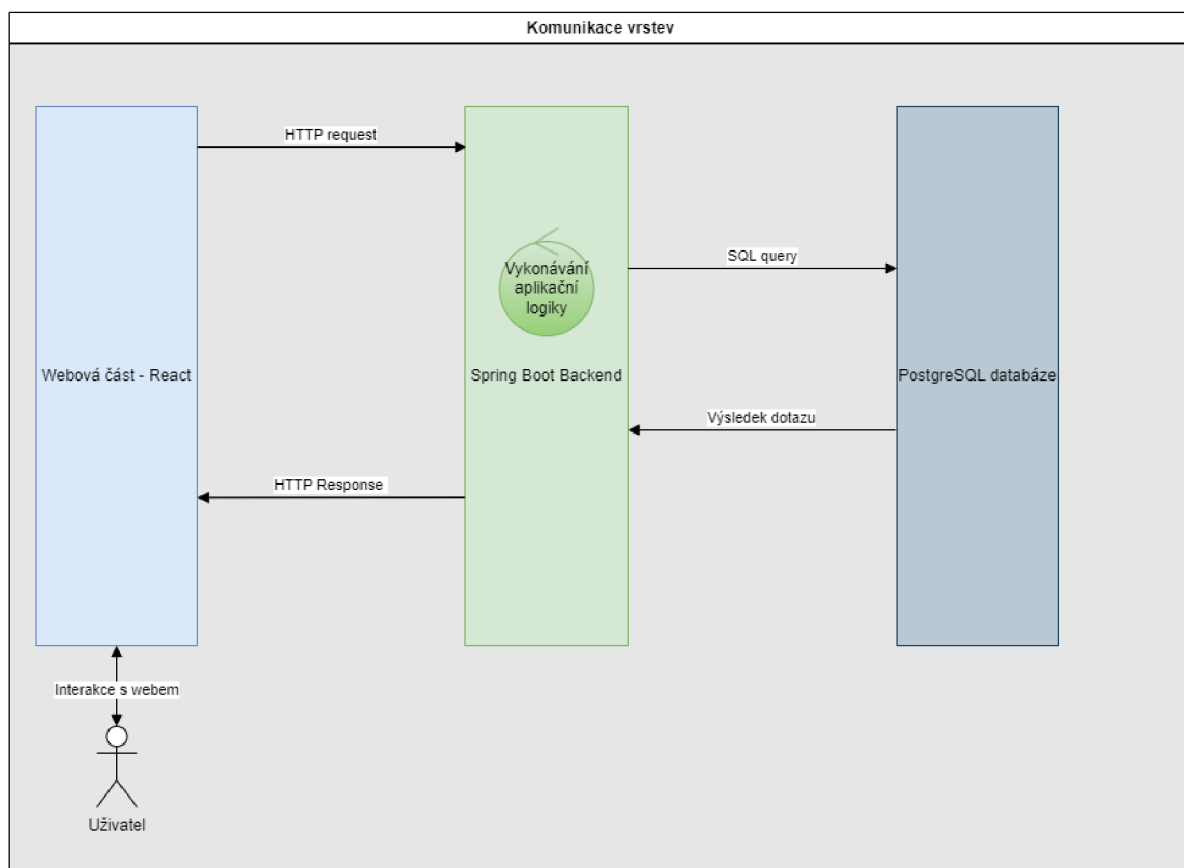
Následující ukázka kódu znázorňuje databázový dotaz (query), který je volán s parametrem dotazu. Další metoda znázorňuje právě automatickou implementaci pojmenovaných metod – tato metoda sice byla v kódu definována, ale není jí explicitně přiřazen kód. Zjednodušeně řečeno ji stačí definovat a ona na vrátí trénink na základě požadavku, který je definován názvem této metody. Některé metody tohoto repositáře lze volat v kódu bez nutnosti inicializace v repositáři. Např., find(), findAll(), findById(), deleteAll(), delete(), save(), atd. lze využít automaticky.

```
public interface TrainingRepository extends JpaRepository<Training, Long> {  
  
    @Query("SELECT t FROM Training t JOIN t.user u WHERE u.id = :userId")  
    List<Training> findTrainingsContainingUser(@Param("userId") Long userId);  
  
    Training findTrainingByUserAndRacedayIsAfter(User user, LocalDate date);  
  
    @Transactional  
    @Modifying  
    @Query("DELETE FROM Day d WHERE d.training = :training")  
    void deleteDaysByTraining(Training training);  
  
    @Transactional  
    @Modifying  
    @Query("DELETE FROM Training t WHERE t = :training")  
    void deleteTraining(Training training);  
  
    default void deleteTrainingWithData(Training training) {  
        deleteDaysByTraining(training);  
        deleteTraining(training);  
    }  
}
```

Zdrojový kód 1: Ukázka JpaRepository (zdroj: vlastní)

Komunikace s webovými stránkami funguje na základě RESTful API. [49] Representational state transfer, zkráceně REST je paradigma softwarové architektury distribuovaných systémů, používané hlavně v oblasti webu. Nabízí standardizovaná pravidla a metody pro komunikaci v relaci klient-server pomocí HTTP volání. Mezi hlavní principy této softwarové architektury patří „stateless“ komunikace (server nezaznamenává stav klienta, nezáleží na předchozích komunikacích) a použití jedinečných identifikátorů prostředků – URI.

RESTful API implementuje REST principy tvorbou konkrétního rozhraní. Na základě REST principů definuje URL adresy, reprezentaci přenášených dat a HTTP metody pro manipulaci s daty (POST, GET, DELETE, ...). V rámci tohoto projektu jsou data přenášena ve formátu JSON, ale není to podmínka. Data lze posílat v HTTP requestech i pomocí XML, či jiných datových struktur.



Obrázek 10: komunikace vrstev projektu (zdroj: vlastní)

4.5 Zabezpečení

Velmi důležitá část projektu je jeho zabezpečení a ochrana před možnými hrozbami a útoky. O zajištění bezpečnosti aplikace se na backendu stará modul Spring Security.

4.5.1 JSON Web Token

JSON Web Token je základním prvkem zabezpečení aplikace. [50] Jde o vygenerovaný řetězec, který v sobě nese zakódované informace. V případě tohoto projektu jde o identifikační číslo uživatele, název organizace, zakódovaný řetězec pomocí hashovací

funkce, čas expirace tokenu a použitý kódovací algoritmus. Token musí obsahovat nějakou formu utajeného prvku, protože algoritmus pro dekódování je veřejně známý a token stačí zadat do volně dostupného debuggeru a kdokoli má přístup k informacím, které jsou v něm zakódovány a může zkoušet token replikovat. Tento token je posílán v rámci hlavičky zabezpečených HTTP requestů. Logika tokenu je definována v souboru util/JWTUtils. Pro hashování řetězce je použita stejná funkce jako pro hashování uživatelských hesel a to BCryptPasswordEncoder, dostupný ze Spring Security modulu. Následující ukázka vyobrazuje proces generování JWT.

```
public String generateJWTToken(Long userID) {
    String var = hashHandler.hashString(secretVar);
    long day = 7;
    Date expirationDate = Date.from(new Date().toInstant().plus(day,
ChronoUnit.DAYS));
    JWTCreator.Builder jwtBuilder = JWT
        .create()
        .withClaim("PK", userID)
        .withClaim("org", "UHK-FIM")
        .withClaim("var", var)
        .withExpiresAt(expirationDate);

    Algorithm algorithm = Algorithm.HMAC256(secret);
    return jwtBuilder.sign(algorithm);
}
```

Zdrojový kód 2: Ukázka generování JWT (zdroj: vlastní)

Nutno zmínit, že token se na webové části ukládá do úložiště localStorage, což by mohlo být zneužito potenciálním útočníkem. Při každém odeslání požadavku se ale porovnávají údaje z hlavičky http požadavku s údajem v databázi. Aktuální JWT je pro každého uživatele uchovávan v databázi. Je generován při přihlášení uživatele a mazán při odhlášení, takže potenciální doba, kdy lze token zneužít pro nechtěné/neoprávněné požadavky je omezena na čas, kdy je uživatel přihlášen. V rámci projektů s vyššími bezpečnostními nároky by musel být ukládán jinde.

4.5.2 Autorizační filtr

Tento filtr je definován ve třídě security/AuthFilter a rozšiřuje „OncePerRequestFilter“. Tato třída je volána jednou pro každý příchozí HTTP request a její metoda „doFilterInternal“ je zodpovědná za zpracování requestu. Logika je taková, že je příchozí request kontrolován různými testy a pokud všemi projde, pak je propuštěn dále. V opačném případě se nastaví chybový status a chybová hláška a request je vrácen. V případě tohoto projektu se testy

v tomto filtru týkají kontroly tokenu, který je posílán společně s requestem. Kontroluje se jeho přítomnost, expirace a validace.

4.5.3 Další zabezpečení

Projekt v třídě SecurityConfig definuje další bezpečnostní prvky – CORS, CSRF protekce, XSS protekce a funkce pro bezpečná ukládání hesel.

[51] Cross Origin Resource Sharing je politika aplikace, určující povolené zdroje (origins), metody a hlavičky HTTP requestů. Slouží k tomu, aby se předešlo cross-origin útokům. Ty probíhají tak, že útočník vytvoří webovou stránku, která obsahuje škodlivý kód a odkazuje na zranitelný server. Tato stránka může sloužit např. ke krádeži dat, nebo provádění akcí jménem uživatele bez jeho souhlasu. Pokud má server správně nastavenou CORS politiku, tak mají ke komunikaci s ním přístup pouze definované domény, mezi které doména útočníka zaručeně nepatří.

[52] Cross-Site Request Forgery je druh útoku na internetové aplikace, který pracuje na bázi provádění nezamýšlených požadavků v aplikaci, pocházejících z nelegitimního zdroje. Většinou zneužívá akce uživatelů, kteří jsou v okamžik útoku do aplikace přihlášení. Jako ochrana proti těmto útokům se používá implementace autorizačního tokenu (XSRF-TOKEN), jehož úkolem je ověřit, že požadavek na server byl odeslán z legitimního zdroje, ne ze strany útočníka. V případě tohoto projektu je token zaslán serverem jako cookie a následně je načten webovou částí a posílán v hlavičce požadavků, které toto zabezpečení vyžadují. V následující ukázce z třídy SecurityConfig je znázorněna inicializace CookieCsrfTokenRepository, která se stará o vytvoření cookie a nastavení vlastností cookie. Za zmínku stojí nastavení „withHttpOnlyFalse“. Toto nastavení umožňuje přístup JavaScriptovým scriptům k obsahu cookie. V tomto projektu je na webové části token získáván právě pomocí JavaScriptu, což u projektu malého rozsahu, bez velkých bezpečnostních hrozeb není problém, ale při využití tohoto principu u větších projektů by toto nebylo vhodné řešení. V takových případech by se hodilo využít např. ukládání jako session data na straně serveru. Komunikace webu se serverem by měla probíhat pomocí zabezpečeného HTTPS protokolu atd.

```

@Bean
public CookieCsrfTokenRepository csrfTokenRepository() {
    CookieCsrfTokenRepository repository =
    CookieCsrfTokenRepository.withHttpOnlyFalse();
    repository.setCookieName("XSRF-TOKEN");
    repository.setSecure(true);
    repository.setCookiePath("/");

    return repository;
}

```

Zdrojový kód 3: Ukázka nastavení CSRF tokenu (zdroj: vlastní)

[53] Cross-Site Scripting je dalším typem útoku, proti kterým se webové aplikace musí bránit. Jde o vkládání škodlivého kódu do webové stránky, která je následně spuštěna v prohlížeči uživatele. Důsledky tohoto útoku mohou být různé, včetně krádeže dat, přesměrování uživatele na cizí internetovou stránku, nebo v krajních případech i SQL injection (kombinace dvou útoků). Škodlivý kód bývá vkládán jako vstupní hodnota do formuláře, nebo např. jako URL parametr. Některé prohlížeče mají implementovanou ochranu proti XSS útokům, ale na to se tvůrce aplikací nemůže spoléhat. Zamezení případných XSS útoků je povinnost kódu na straně serveru. V rámci tohoto projektu je nastaveno blokování vykonávání nebezpečného skriptu. Dále je definováno povolení spuštění skriptu pouze ze stejného zdroje, jako je webová stránka ("script-src 'self' "). Tím se zabraňuje spuštění skriptů třetích stran, což dále snižuje riziko XSS útoku.

Velmi důležitá součást bezpečnosti je šifrování hesel uživatelů. Pro hashování je použit již zmíněný BCryptPasswordEncoder. Šifrování pomocí této funkce je založeno na algoritmu BCrypt. Jde o velmi bezpečný, kryptografický algoritmus, který je velmi náročný na prolomení hrubou silou. V rámci projektu jsou všechna hesla hashována po přijetí na backendu a do databáze jsou ukládány v zahashované podobě. [54] V rámci testování a porovnávání během vývoje byl použit také online Bcrypt validátor, který porovná zahashovanou hodnotu s řetězcem a vyhodnotí shodu.

5 Proces generování

Tréninkový plán je základem pro systematickou sportovní přípravu. Sportovec těžko dosáhne nadprůměrných výsledků bez opakovaného přemýšlení, co zlepšit, co dělat jinak, čemu se vyhnout atd. Ti nejlepší sportovci k tomu mají profesionální trenéry, kteří několik hodin denně věnují optimalizaci přípravy svých svěřenců. Tito trenéři vycházejí z vlastních zkušeností, ze zaběhlých pravidel, či z doporučení sportovních lékařů.

Sportovci, co ale nejsou na tak vysoké úrovni a trenéra nemají musejí tréninkový plán získávat jinak. Mohou např. navštěvovat skupinové lekce a řídit se pokyny instruktora. Nebo si můžou zkusit trénink sestavit sami. To může fungovat, ale s riziky, jako jsou špatné návyky/technika u daného sportu, přetěžování, nebo naopak nevyčerpání určitých svalových partií atd.

Existuje ale ještě další možnost, a to spojení sportovní přípravy s využitím moderních technologií. V dnešní době lze použít komplexní algoritmy, které zohlední možná úskalí tréninku, zpracují vstupní data uživatele a na základě vzorových dat desítek ostatních tréninků stanoví zájemci tréninkový plán na míru. Tato vize je i v omezeném rozsahu cílem tohoto projektu.

Obecně ke generování nových dat existuje více přístupů, mohou se využít např. genetické/evoluční algoritmy, Bayesovské sítě, neuronové sítě, nebo rozhodovací stromy. V rámci tohoto projektu je cílem vygenerovat odpovídající tréninkový plán a jídelníček za pomoci algoritmu s odpovídající složitostí.

5.1 Rozhodovací strom J48

Pro generování v tomto projektu byl zvolen rozhodovací strom J48. [43] Tento algoritmus je open source Java implementace algoritmu C4.5, který slouží ke klasifikaci dat a je obsažen v knihovně Weka. Weka je rozsáhlá knihovna, obsahující mnoho nástrojů, určeným k využití strojového učení. Kromě rozhodovacího stromu J48 obsahuje ještě další druhy algoritmů strojového učení, jako např. tzv. naivní Bayesův klasifikátor, Bayesovské sítě, Logistická regrese, nebo algoritmus k-nejbližších sousedů. [42] Problematika klasifikace dat se zabývá procesem přiřazování kategorií/tříd k jednotlivým datovým záznamům na základě jejich charakteristik. Cílem klasifikace je schopnost vytvořit model, jenž je schopný automaticky přiřazovat nové záznamy do určených kategorií na základě tréninkových dat modelu. Jde o klíčový proces v oblasti strojového učení a využívá se v mnoha funkcích, jako je třeba rozpoznávání obrazu, nebo i predikce nemocí, či predikce úrodnosti půdy. Tento

algoritmus nejlépe funguje, když operuje nad velkou testovací sadou dat, která zde není k dispozici. S malou sadou dat přichází obtíže typu omezení statistických vlastností, nebo nižší spolehlivost výsledků. Proto musí být generační algoritmus vhodně ošetřen, aby nedocházelo k nesmyslným doporučením.

5.2 Tvorba algoritmu

Vygenerování tréninku je nejdůležitější část celého projektu, protože z ní vychází i generování jídelníčku. Vstupní údaje jsou třída User, která obsahuje potřebné údaje, jako BMI, druh postavy atp. a údaje o plánovaném tréninku – údaje o závodu (datum, délka, druh, dostupné dny tréninku případně požadované tempo). Tyto údaje jsou nejprve testovány, zda odpovídají realitě (např. aby nedošlo k tomu, že začátečník spustí přípravu na maraton, která potrvá 2 dny). Pokud vstupní údaje projdou úspěšně všemi testy, přistoupí se k samotnému generování.

To probíhá tvorbou instance Day pro každý den tréninku. Třída Day obsahuje kromě datumu ještě seznam cvičení a jídel pro konkrétní den a informace o kalorickém příjmu a výdeji. Využití klasifikace dat pro generování dat vůbec není lehký úkol, obzvlášť pokud není k dispozici dostatečně obsáhlá sada tréninkových dat. Výsledek prvních experimentů s generováním nových dat na základě sady tréninkových dat je vidět v následující ukázce.

Tato sada dat je vytvořena ve formátu ARFF (Attribute Relationship File Format), který je výchozím typem souboru pro knihovnu Weka a obsahuje zhruba 40 záznamů, které vycházejí z teoretických principů pro tvorbu tréninku. Jsou uměle tvořeny tak, aby odpovídaly předpokladům.

@relation PersonalizedTrainingPlan

@attribute weight numeric

@attribute height numeric

@attribute bmi numeric

@attribute sex {MALE, FEMALE}

@attribute bodyType {Average, Athletic, Obese}

@attribute availableTrainingDays numeric

@attribute trainingLengthInDays numeric

@attribute usersRunLengthInMeters numeric

@attribute usersRunPace numeric

@attribute raceLengthInMeters numeric

@attribute racePaceToAchieve numeric

@attribute raceType {Run, OCR}

@attribute raceElevation {FLAT_ROAD, CROSS, UPHILL}

@attribute trainingsPerWeek numeric

@attribute trainingsDemandingness {Easy, EasyMedium, Medium, MediumHard, Hard, HardWithJogging}

%principles

% OCR requires specific training and is a goal for experienced sportsmen

% high track elevation require harder training

% minimal pace for run = 10min/km, slower pace for OCR

% easier training for obese

@data

70, 175, 22.9, MALE, Average, 5, 90, 5000, 6.5, 10000, 6.5, Run, FLAT_ROAD, 4, MediumHard

85, 180, 26.2, FEMALE, Obese, 4, 120, 2000, 7.2, 7000, 10, Run, CROSS, 3, EasyMedium

63, 168, 24.1, MALE, Athletic, 6, 60, 5000, 6.0, 6000, 6.0, OCR, CROSS, 5, Hard

72, 172, 24.3, FEMALE, Athletic, 5, 90, 8000, 6.8, 6000, 6.8, OCR, CROSS, 4, MediumHard

58, 160, 22.7, MALE, Average, 6, 75, 6000, 6.0, 6000, 5.5, OCR, CROSS, 5, MediumHard

78, 185, 22.8, FEMALE, Athletic, 4, 100, 8000, 7.5, 10000, 7.0, Run, CROSS, 4, MediumHard

95, 190, 26.3, MALE, Obese, 3, 150, 3000, 8.0, 6000, 8.0, Run, FLAT_ROAD, 3, Easy

67, 170, 23.2, FEMALE, Average, 5, 120, 6000, 6.2, 6000, 6.0, Run, CROSS, 5, MediumHard

Zdrojový kód 4: Ukázka experimentální sady tréninkových dat (zdroj: vlastní)

I přes to, že se data uměle tvořila tak, aby vše odpovídalo požadavkům, evaluace této sady dat vyhodnotila nepoužitelnost těchto dat, viz následující ukázka.

Correctly Classified Instances	3	20	%
Incorrectly Classified Instances	12	80	%
Kappa statistic	-0.0651		
Mean absolute error	0.243		
Root mean squared error	0.4229		
Relative absolute error	90.639	%	
Root relative squared error	115.749	%	
Total Number of Instances	15		

Zdrojový kód 5: Evaluace původní sady dat (zdroj: vlastní)

Správně bylo klasifikováno jen 52.5 % instancí, což je v daném modelu (někde výběr až z 6 kategorií) větší úspěch než využít čistě náhodné generování, ale rozhodně to není uspokojivý výsledek pro generování dat, kterými se budou uživatelé řídit.

5.2.1 Změna přístupu

Strategie generování tedy musela být změněna. Je třeba vzít v potaz charakteristiku J48. Jde o přiřazování dat do kategorií, což může být problém u numerických hodnot. [44] Generovat data na základě numerických hodnot lze. Je to ale prvek, který dokáže snížit výsledek evaluace dat, protože správně klasifikovat celé číslo, kde je teoreticky nekonečně mnoho možných odpovědí je mnohem náročnější, než např. zvolit jednu z pěti kategorií. Může nastat otázka, zda by byl problém, kdyby algoritmus navrhl 7.5 km běhu místo 7 km. Pro evaluaci dat a co nejlepší a nejjednodušší fungování algoritmu je lepší se numerickým hodnotám vyhnout a používat nominální. V nové strategii byl zvolen jiný přístup, a to využití generačního algoritmu lokálně. Pro generování konkrétního cvičení jsou vzaty v potaz minulé tréninky a na základě toho je zvolen následující. Datová sada pro takový úkon obsahuje méně atributů a je možné pracovat bez použití numerických hodnot.

Ve fázi, kdy je přistoupeno k vygenerování nového cvičení a uložení jeho hodnoty do databáze už je přípravnou částí algoritmu vyřešena většina náležitostí – je určeno, jestli je trénink zvládnutelný a jsou alokovány tréninkové dny. Tím pádem se o to nemusí starat J48

a může všechny atributy vymezit jedinému cíli, a to stanovení následujícího cviku pro daný den.

Předchozí postup byl spíše globálního charakteru, kde algoritmus řešil délku závodu, tempo, počet tréninků za týden atd. Kdyby algoritmus uměl detailně zohlednit každý z těchto údajů a upravit na základě toho výsledek pro co nejoptimálnější výstup, bylo by to skvělé, ale pro takovou práci by bylo potřeba mít velice rozsáhlou sadu, velmi dobře zpracovaných historických dat, což rozhodně není v silách tohoto projektu. Jak již bylo dříve zmíněno, podobné algoritmy dokážou s rozsáhlým učícím procesem úspěšně predikovat i např. různé nemoci u lidí, nebo úrodnost zemědělské půdy.

Následně použitá verze pracuje pouze v lokálním měřítku, je volána pro každý den. Během testování lokálního použití algoritmu se zjistilo, že chybovost u různých prototypů malé sady dat je často ještě vyšší než u té původní, dokonce se u relativně přímočaře skládané sady dat, obsahující 3 atributy a 15 záznamů úspěšnost klasifikace pohybovala v rozmezí 0 až 30 %, což je velmi znepokojivý výsledek.

Při hlubším zkoumání problematiky bylo experimentováno s dalšími aspekty J48. Byla měněna evaluační strategie, její parametr „number of folds“ a parametry samotného J48 algoritmu. Při evaluaci dat je použita metoda tzv. křížové validace. Sada dat je rozdělena na několik stejně velkých částí. Model je pak natrénován na určité části dat a testován na zbytku dat. Tento proces se opakuje alespoň jednou pro každou část. Parametr number of folds určuje, na kolik částí se sada dat rozdělí. Většinou je tato hodnota nastavena na 10, ale může se měnit. Pro rozsáhlé sady dat, kde jsou např. desetitisíce záznamů se dají data rozdělit na více částí (za cenu vyšší výpočetní náročnosti) a může tím být dosaženo lépe vypovídající evaluace. Naopak v situaci tohoto projektu může být tento parametr snížen (Při evaluaci 15 řádků nemá smysl sadu dat rozdělovat na 10 částí, protože z dat zmizí aspekt vzájemné provázanosti a tvořit rozhodovací strom z např. jednoho záznamu nedává smysl). Důležité je ale zmínit, že celý proces evaluace nemá žádný vliv na samotný algoritmus generování dat. Jde čistě jen o formu vyhodnocení a optimalizace tohoto aspektu může tvůrci algoritmu poskytnout zpětnou vazbu ohledně kvality algoritmu.

Na druhou stranu parametry algoritmu přímo upravují chování a výstup rozhodovacího stromu. Mezi klíčové parametry J48 patří: confidenceFactor, minNumObj, unpruned a binarySplits. Zlepšení výkonu rozhodovacího stromu přineslo zapnutí parametru „unpruned“. Unpruned stromy jsou zjednodušeně řečeno neprořezávané. Při pruningu dochází k omezení přílišné komplexity stromu tím, že se odstraňují části stromu, které jsou vyhodnoceny jako nadbytečné, nebo příliš specifické. Taková data nepřinášejí velký přínos při klasifikaci nových dat a tím pádem se nevyplatí je ve stromu uchovávat. [45] Pro

představu, jeden z kurzů univerzity Waikato poukazuje na vzorovou situaci, kde byl použit J48 rozhodovací strom na sadě dat o rakovině prsu, jednou s parametrem pruning zapnutým a podruhé s tímto parametrem vypnutým. V případě zapnutého pruningu strom obsahoval celkem 4 listy a 6 uzlů s přesností klasifikace 75.5 %. V druhém testu, kde se tento parametr vypnul výsledný strom obsahoval 152 listů a 179 uzlů. Při tomto obřím nárůstu komplexity stromu se jeho míra úspěšné klasifikace snížila o 6 %. Z toho vyplývá, že komplexnější strom (který má navíc mnohem vyšší výpočetní náročnost) nemusí poskytnout lepší výsledek, než „zjednodušený“ strom. Nevýhodou unpruned stromu jsou tendence snadno se přeučit na popud specifických charakteristik, nebo šumu. Dále již zmíněná vysoká výpočetní náročnost a nadbytečná složitost. V případě tohoto konkrétního projektu parametr unpruned zvýšil míru úspěšnosti klasifikace dat díky tomu, že pracuje s malou sadou dat. U takového případu se není třeba obávat stovek uzlů, ani přeučení. Pruned stromy by byl pro takovou situaci příliš jednoduchý – neměl by zkrátka dostatečnou komplexnost na zachycení vztahů mezi daty. Tato situace je ale v oblasti strojového učení spíše výjimkou, protože v praxi strojové učení zpravidla vychází z velkých sad tréninkových dat a úspěšnost.

5.2.2 Finální verze algoritmu

Po testování různých struktur učicích datových sad bylo přistoupeno k poměrně jednoduché verzi, která ale dobře spolupracuje s logikou aplikace. V lokálním užití algoritmu byla velká nevýhoda, že pokud by administrátor přidal jakékoli cvičení nebo jídlo do databáze, musel by změně přizpůsobit rozhodovací algoritmus. Výsledná verze pouze testuje zásadní parametry uživatele, pro které určí (vygeneruje) jednu ze čtyř možností tréninku. Na základě tohoto výstupu jsou jednotlivé dny tréninku naplněny cviky a jídly z databáze, které odpovídají požadavkům.

Při testování se projevil jasný trend snižování úspěšnosti evaluace dat s rostoucí hodnotou parametru number of folds. Nejlepší výsledek výsledného algoritmu byl 78 % správně klasifikovaných instancí při hodnotě number of folds 2. Pruning u takto jednoduché struktury nevytvořil žádný rozdíl, takže je ve finální verzi vypnutý. Úryvek výsledné tréninkové sady dat je znázorněn na následující ukázce. Zbytek informací o finálním stavu projektu je k dispozici v kapitole 7 – Shrnutí výsledků.

```

@relation FinalDataset

@attribute bodyType { AVERAGE,ATHLETIC,OBESE,MUSCULAR }
@attribute BMI { UNDERWEIGHT, NORMAL, OVERWEIGHT, OBESE }
@attribute percentageIncrease {0, 0.5, 1, 1.5, 2,2.5,3, 3.5,4,4.5,5,5.5 }
@attribute TrainingAlgorithm {HIGH_GAIN_HIGH_BURN,
HIGH_GAIN_LOW_BURN,LOW_GAIN_HIGH_BURN,LOW_GAIN_LOW_BURN}

%high burn correlates with hard training
%underweight people should gain to achieve a healthy weight
%overweight people should burn a lot
%muscular people usually gain lot of calories for maintaining muscle mass

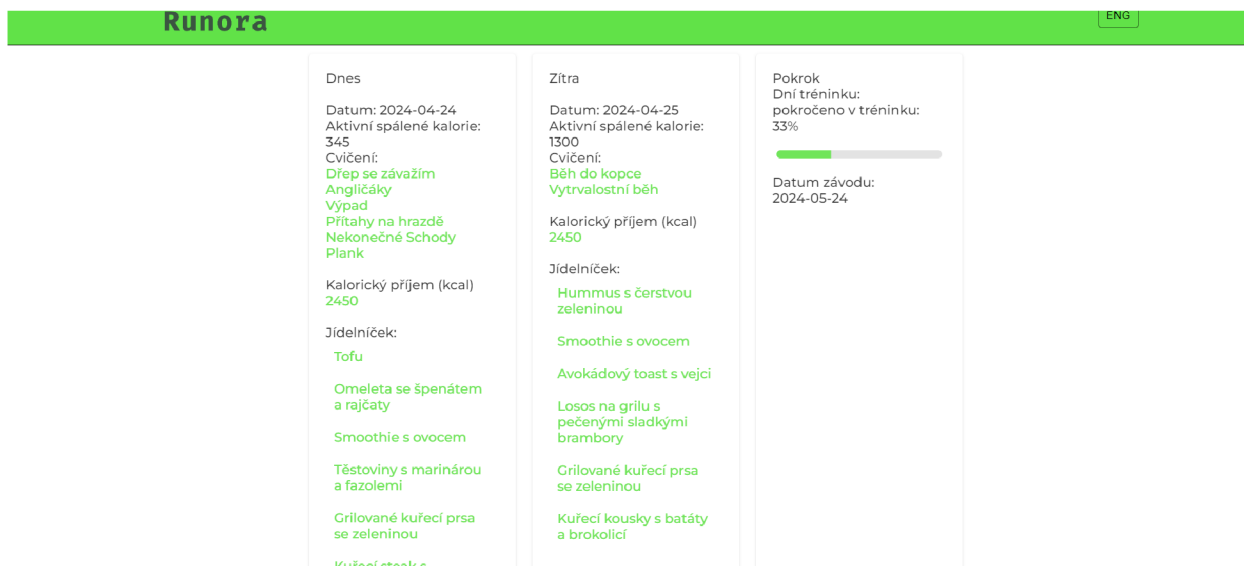
@data
ATHLETIC, NORMAL, 1, LOW_GAIN_LOW_BURN
ATHLETIC, NORMAL, 5.5, HIGH_GAIN_HIGH_BURN
ATHLETIC, UNDERWEIGHT, 5.5, HIGH_GAIN_HIGH_BURN
ATHLETIC, UNDERWEIGHT, 2, HIGH_GAIN_LOW_BURN
ATHLETIC, OVERWEIGHT, 3, LOW_GAIN_HIGH_BURN
ATHLETIC, NORMAL, 5.5, HIGH_GAIN_HIGH_BURN
ATHLETIC, UNDERWEIGHT, 5.5, HIGH_GAIN_HIGH_BURN
AVERAGE, UNDERWEIGHT, 2, HIGH_GAIN_LOW_BURN
AVERAGE, NORMAL, 3, HIGH_GAIN_LOW_BURN
AVERAGE, OVERWEIGHT, 2, LOW_GAIN_HIGH_BURN
AVERAGE, UNDERWEIGHT, 2.5, HIGH_GAIN_LOW_BURN

```

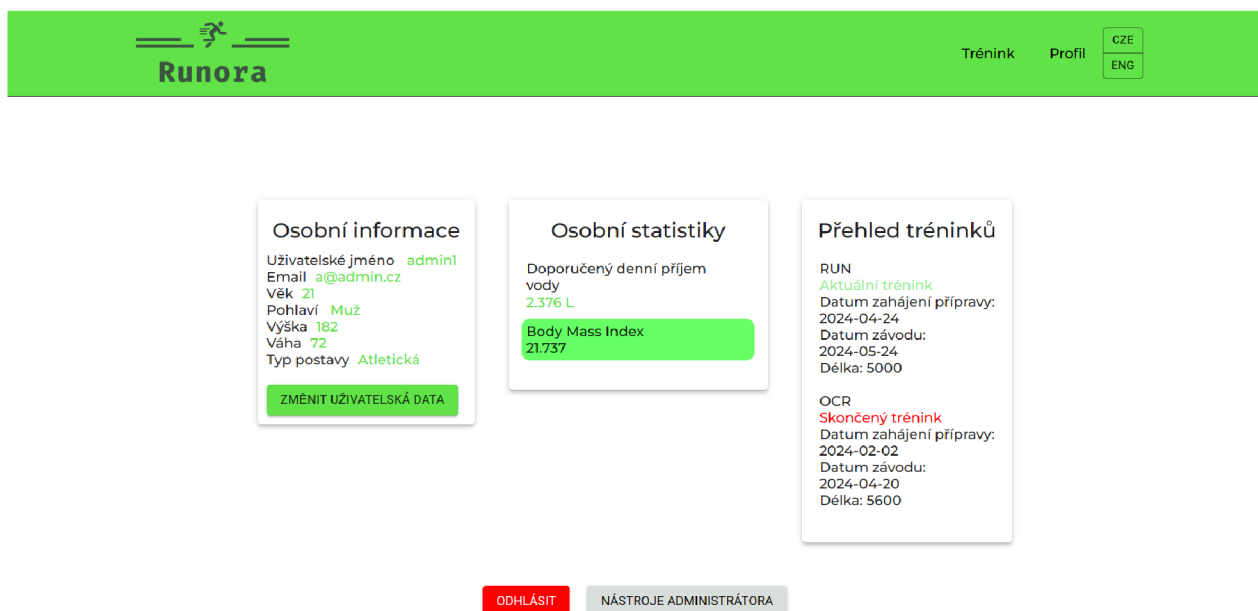
Zdrojový kód 6: Ukázka finální verze zdrojového kódu (zdroj: vlastní)

5.3 Porovnání tréninku

Následující obrázky ukazují výstup tréninku. Doporučená jídla a cviky jsou volena z aktuálních dat v databázi, takže kvalita výstupu se může v čase měnit, podle toho, jak administrátoři přidávají nová jídla a cviky do databáze. Výsledné tréninkové dny odpovídají výsledku algoritmu. V tréninku na obrázku byl zvolen vysoký příjem i výdej kalorií pro člověka vážícího 72 kg s výškou 182 cm. Příprava trvá měsíc a nejde o standardní běh, ale gladiator race. Algoritmus zvolil vysoký kalorický příjem i výdej, což by pro takový cíl bylo vhodné. Doporučené cviky dávají smysl a jejich počet a rozsah by mohly tvořit adekvátní přípravu na podobný závod.



Obrázek 11: Výstup tréninku (zdroj: vlastní)



Obrázek 12: Stránka profilu (zdroj: vlastní)

Při snaze o porovnání a aktivování prémiové verze Fitify bylo touto službou nabídnuto několik obecných, předpřipravených tréninkových plánů s názvy „kardio“, „síla“ a podobně. Konkrétní generování tréninku tato služba nenabízí. RunReps nabídne tabulku s běhy, obsahující nějakou progresivní zátěž s volnějším posledním týdnem před závodem. Fitify nabízí jeden program adekvátní tomuto tréninku, který se zaměřuje na balanc mezi silou a vytrvalostí. Trénink probíhá tak, že je aplikace spuštěna během samotného tréninku a stopuje čas pro jednotlivé cviky. Jde např. o HIIT neboli intervalový trénink vysoké intenzity, nebo klasické cviky, jako např. kliky apod. Rozsah cviků je podobný (u bakalářské práce se v čase

může měnit) a aplikace nabízí i recepty na jídla. Aplikace si jsou podobné, tento projekt nabízí větší personalizaci (volba konkrétního cíle, kalkulace dosažitelnosti cíle), ale menší promyšlenost výsledných tréninků. Velký rozdíl je také v designu aplikace, Fitify má výborně zpracovaný design. Designem se tento projekt rozhodně chlubit nemůže.

6 Testování a vyhodnocení

Testování je proces analýzy kódu softwaru, při které se ověřuje, zda se kód chová podle očekávání a nejsou v něm chyby. Hledání a odstraňování chyb je velkou a nedílnou součástí celého procesu vývoje všech aplikací. Aplikace jsou testovány různými principy i různými lidmi. Hlavní roli v testování mají vývojáři projektu, kteří tvoří kód a logiku fungování aplikace, ale u větších, nebo komerčních projektů je aplikace testována i ze strany uživatele/zákazníka z hlediska ovladatelnosti, přehlednosti, složitosti atd. Nejdůležitějším druhem testování je tzv. jednotkové testování.

6.1 Jednotkové testování

[46] Jednotkové testování je proces testování těch nejmenších funkčních prvků kódu. Těmito prvky jsou v jazyce Java metody tříd. Při jednotkovém testování se mohou ostatní části aplikace nahrazovat falešnými objekty, aby se předešlo ovlivnění správnosti metody nestandardním chováním jiné třídy. Tento proces se nazývá „mocking“.

Mock je simulovaný objekt, který může nahrazovat třídu, nebo objekt, u kterého můžeme předurčit jeho chování. To pomáhá při tvorbě testů, které jsou ovlivněny dalšími okolnostmi, které s konkrétní metodou nesouvisí. Pomocí mock objektů lze simulovat i různé podmínky, kterých by bylo obtížné, nebo nemožné dosáhnout se skutečnými daty. Mezi takové využití mock objektů slouží např. simulování odpovědi ze serveru, databázová operace, uživatelský vstup, nebo simulace výjimky.

V tomto projektu je na backendové části aplikace testována funkčnost vybraných metod, zejména týkajících se entity User. Tento proces probíhá odděleně od zbytku kódu, a to v podadresáři „test“. Jsou zde jednotkové testy, týkající se těch nejdůležitějších metod v kódu, kterými jsou zejména metody, týkající se nakládání s uživatelskými daty. V kódu lze najít anotace „@BeforeEach“ a „@AfterEach“. Tyto anotace definují metody, které jsou volány před a po provedení testu. Slouží k vymazání všech dat v testovací databázi, aby se při provádění testu v databázi nenacházela data z předešlých testů. Některé metody jsou testovány vícekrát s různými průběhy. Zkratka „Hds“ v názvu testu značí koncept „happy day scenario“, též nazývaný „happy path“. Jde o situaci, kdy je metoda použita přesně tak, jak byla vývojářem použita a nedochází k neočekávaným událostem. To ale většinou nebývá jediný možný průchod metodou, např. při přihlášení může uživatel zadat buď správné heslo (HDS), nebo špatné. Na základě uživatelského vstupu je chování a výstup metody změněn. V tomto případě, pokud je heslo špatné, místo přihlášení je uživateli vrácena chybová hláška.

Na následující ukázce je průběh testování přiblížen. Zobrazený test je přímočarý, ale princip funguje u všech testů podobně. Nejprve je ověřeno, že do testovací databáze nebyl přidán žádný uživatel. Pak jsou připravena testovací data a následně je spuštěna metoda, nebo skupina metod, které se testují. Na závěr testu jsou porovnávány výsledky. Pokud se jedna, nebo více hodnot neshoduje, test je vyhodnocen chybou.

```
@Test
void login_NonexistentUser() {
    log.info("Logging in via made up credentials. Access should be rejected");
    // Checking for empty database
    assertThat(userRepository.findAll(), empty());
    // Data preparation
    String email= "abc@uhk.cz";
    String password = "Passwd_123!";
    // Execution
    ResponseEntity<?> responseEntity = userService.login(email,password);

    // Verification
    assertThat(responseEntity, notNullValue());
    assertThat(responseEntity.getStatusCode(), equalTo(HttpStatus.BAD_REQUEST));
    assertThat(responseEntity.getBody(), is("Uživatel nenalezen"));
}
```

Zdrojový kód 7: Ukázka jednotkového testu (zdroj: vlastní)

Obecně je testování důležitá součást procesu vývoje a existuje dokonce vývojová metodika, při které se testy píšou dříve než samotné metody (není tím myšlena tato práce). Tzv. Test Driven Development je metodika softwarového inženýrství, kde jsou nejprve detailně popsány funkcionality požadovaného systému. Následně jsou napsány testy, které přesně definují stav projektu po provedení metody. Pak je psána samotná metoda a cílem je napsat co nejnižší a nejprehlednější množství kódu, které vede ke splnění jednotkového testu této metody.

7 Shrnutí výsledků

Cílem práce bylo využít moderní technologie ve prospěch zdraví uživatelů. K tomu výsledný produkt může pomoci. Dalším cílem bylo vytvořit nástroj pro začátečníky v běhu, kde najdou všechny podstatné informace na jednom místě, což nebylo splněno ve velké míře, ale potenciální uživatel se určitě může něco dozvědět (stav BMI, příjem vody, kalorické údaje u cviků a jídel). Některé požadavky, rozebírané v teoretické části nejsou implementovány. Za hlavní nesplnění cíle by šlo považovat chybějící implementace zpětné vazby, která by celý proces tréninku mohla znatelně vylepšit a více ho personalizovat a přiblížit potřebám uživatele. Důvodem nesplnění všech cílů jsou časové dispozice a nedostatečné znalosti všech používaných technologií. Nejnáročnější částí tvorby aplikace byla implementace zabezpečení, seznámení s Dockerem a tvorba rozhodovacího stromu pomocí knihovny Weka.

Co se týče splnění požadavků, nestihlo se úplně vše. Prioritu získaly složitější prvky projektu. Např. bezpečnostní upozornění u jednotlivých cviků by pro uživatele v praxi měly zásadní přínos, ale v rámci prokázání znalostí jde jen o další textové pole na webu. Byl kladen důraz hlavně na implementaci zabezpečení projektu, proces generování tréninku a vytvoření virtualizovaného kontejneru pomocí Dockeru.

Většina požadavků byla splněna a ze základní funkcionality projektu funguje vše. Proces generování tréninku není tak komplexní a přínosný, jak bylo původně zamýšleno, ale pořád jde o komplexní proces, jehož výsledkem jsou data, která ve většině případů odpovídají realitě. Některé implementace strojového učení v praxi pracují s podobnou, ne-li nižší úspěšností algoritmu. Samozřejmě zde záleží na mnoha faktorech, jako je povaha úkolu a tréninková sada dat, která je zde uměle sepsána a testována pro co největší úspěšnost. V rámci tohoto projektu je proces tvorby ještě ztížen faktem, že nabídka cviků a jídel v databázi je dynamická. Administrátor teoreticky může v reálném čase přidat cvik do databáze a od toho momentu může každý vygenerovaný tréninkový plán tento cvik obsahovat.

8 Závěr

Práce je rozdělena na teoretický podklad a praktickou implementaci projektu. V teoretické části jsou zhodnoceny konkurenční služby pro ucelení představy o budoucím výsledku a případné inspiraci. Dále jsou na základě dat, zaběhlých praktik a studií představeny principy, které mohou sloužit jako komplexní vzor pro proces tvorby běžeckého tréninku a jídelníčku.

V praktické části byla popsána implementace projektu, včetně představení použitých technologií, architektury projektu a implementace bezpečnosti. V implementaci je kladen důraz na metody generování tréninku, která využívá metody strojového učení.

Aplikace je přizpůsobena službě Docker, která umožňuje izolaci aplikace do kontejneru. Jde o proces virtualizace, který umožňuje spustit aplikaci v libovolném prostředí, ať už jde o Windows, Linux, či MacOS. Vzhled aplikace je velmi jednoduchý, pro potenciální komerční využití je v tomto stavu nekonkurenceschopný. V porovnání s např. Fitify a podobnými službami by tento projekt neměl šanci.

Tento projekt má velký potenciál budoucího rozvoje, např. rozšiřování dostupných cvičení, vylepšování algoritmu samotného generování tréninku, nebo i možnost rozšíření na mobilní aplikaci, co je schopna pomocí GPS zaznamenávat běžecké tréninky. Konkurence v oboru generování tréninků počítačem existuje, ale není v ní žádný velký monopol a příchod umělé inteligence může také změnit a usnadnit celý proces generování tréninku. Využití strojového učení dodává projektu nový rozměr realizace. Do budoucna by mohly být využity i jiné druhy strojového učení, jako je třeba umělou neuronová síť. Optimální by byla kombinace neuronové sítě se zaznamenáváním a zpracováním tréninku v reálném čase. Bylo by žádoucí i využití nějaké šablony pro konzistentní, robustní web.

9 Seznam obrázků

Obrázek 1: Ilustrační obrázek Fitify UI (zdroj: [25]).....	7
Obrázek 2: Princip adekvátního tréninku (zdroj: [9])	9
Obrázek 3: Vliv věku a pohlaví na běh (zdroj: [6])	10
Obrázek 4: Princip blokové periodizace (zdroj: [9]).....	13
Obrázek 5: Křivka množství a intenzity tréninku – lineární trénink (zdroj: [9]).....	14
Obrázek 6: Vliv soustavného tréninku (zdroj: [9])	15
Obrázek 7: Diagram případu užití (zdroj: vlastní)	21
Obrázek 8: Porovnání ona-way a two-way data binding (zdroj: [39]).....	24
Obrázek 9: Ukázka schéma databáze (zdroj: vlastní)	26
Obrázek 10: komunikace vrstev projektu (zdroj: vlastní).....	29
Obrázek 11: Výstup tréninku (zdroj: vlastní).....	40
Obrázek 12: Stránka profilu (zdroj: vlastní)	40

10 Seznam zdrojových kódů

Zdrojový kód 1: Ukázka JpaRepository (zdroj: vlastní)	28
Zdrojový kód 2: Ukázka generování JWT (zdroj: vlastní)	30
Zdrojový kód 3: Ukázka nastavení CSRF tokenu (zdroj: vlastní)	32
Zdrojový kód 4: Ukázka experimentální sady tréninkových dat (zdroj: vlastní).....	35
Zdrojový kód 5: Evaluace původní sady dat (zdroj: vlastní)	36
Zdrojový kód 6: Ukázka finální verze zdrojového kódu (zdroj: vlastní).....	39
Zdrojový kód 7: Ukázka jednotkového testu (zdroj: vlastní).....	43

11 Seznam tabulek

Tabulka 1: Energetický výdej aerobních aktivit (zdroj: [10]).....	2
---	---

12 Literatura

- 1 THAT. The State of Obesity: Better Policies for a Healthier America. Online. 2022. Dostupné z: https://www.tfah.org/wp-content/uploads/2022/09/2022ObesityReport_FINAL3923.pdf. [cit. 2023-08-14].
- 2 PUBLICHEALTH.ORG. OBESITY IN AMERICA. Online. Dostupné z: <https://www.publichealth.org/public-awareness/obesity>. [cit. 2023-08-21].
- 3 WEINBERG, Robert; YUKELSON, Dave; BURTON, Damon a WEIGAND, Daniel. Perceived Goal Setting Practices of Olympic Athletes: An Exploratory Investigation. Online. 2000. Dostupné z: <https://doi.org/10.1123/tsp.14.3.279>. [cit. 2023-08-15].
- 4 TENENBAUM, Gershon; SPENCE, Ron a CHRISTENSEN, Steven. The Effect of Goal Difficulty and Goal Orientation on Running Performance in Young Female Athletes. Online. 2011. Dostupné z: <https://doi.org/10.1080/00049539908255328>. [cit. 2023-08-15].
- 5 WEINBERG, Robert. Goal setting in sport and exercise: Research and practical applications. Online. 2013. Dostupné z: <https://doi.org/10.4025/reveducfis.v24.2.17524>. [cit. 2023-08-15].
- 6 LARA, Beatriz; SALINERO, Juan José a DEL COSO, Juan. The relationship between age and running time in elite marathoners is U-shaped. Online. 2014. Dostupné z: <https://doi.org/10.1007/s11357-013-9614-z>. [cit. 2023-08-16].
- 7 HALLAM, Lydia a AMORIM, Fabiano. Expanding the Gap: An Updated Look Into Sex Differences in Running Performance. Online. 2021. Dostupné z: <https://doi.org/10.3389/fphys.2021.804149>. [cit. 2023-08-16].
- 8 Jogging. Online. In: Wikipedia: the free encyclopedia. San Francisco (CA): Wikimedia Foundation, 2024. Dostupné z: <https://en.wikipedia.org/wiki/Jogging>. [cit. 2024-08-16].
- 9 WORLDATHLETICS.ORG. Chapter 3 – Principles of training. Online. Dostupné z: <https://worldathletics.org/download/download?filename=f9fa48c2-2a0c-46f3-88b9-149f4d561326.pdf&urlslug=Chapter%203:%20Training>. [cit. 2023-08-16].
- 10 SHEPHERD, John. The complete guide to sports training. Online. A & C Black, 2013. Dostupné z: https://books.google.cz/books?id=r4bCCEXJY4C&pg=PR10&hl=cs&source=gbs_selected_pages&cad=2#v=onepage&q&f=false. [cit. 16.8.2023].
- 11 BOULLOSA, Daniel; ESTEVE-LANAO, Jonathan; CASADO, Arturo; PYÉRE-TARTARUGA, Leonardo A.; GOMES DA ROSA, Rodrigo et al. Factors Affecting Training

- and Physical Performance in Recreational Endurance Runners. Online. 2020. Dostupné z: <https://doi.org/10.3390/sports8030035>. [cit. 2023-08-17].
- 12 BODERSKOV, Max. GUIDE: PERIODIZATION – DEFINITION AND APPLICATION. Online. 2023. Dostupné z: <https://doi.org/10.3390/sports8030035>. [cit. 2023-08-17].
 - 13 CHLÁDEK, Jaroslav. Aktuální trendy v periodizaci a plánování sportovní přípravy. Online, Diplomová práce, vedoucí PhDr. Jan Cacek, Ph.D. Brno: MASARYKOVA UNIVERZITA Fakulta sportovních studií Katedra atletiky, 2016. Dostupné z: https://is.muni.cz/th/r6mh3/DP_aktualni_trendy_v_periodizaci_a_planovani_sportovni_pripravy.pdf. [cit. 2024-08-17].
 - 14 HAVLOVÁ, barbora. Poznej výhody aktivní regenerace a zlepší svůj výkon! Online. 2019. Dostupné z: <https://www.kulturistika.com/magazin/trenink/poznej-vyhody-aktivni-regenerace-a-zlepsi-svuj-vykon>. [cit. 2023-08-20].
 - 15 Basal metabolic rate. Online. In: Wikipedia: the free encyclopedia. San Francisco (CA): Wikimedia Foundation, 2024. Dostupné z: https://en.wikipedia.org/wiki/Basal_metabolic_rate. [cit. 2024-08-20].
 - 16 HVIZDIŠOVÁ, Dominika. CO JE TO BAZÁLNÍ METABOLISMUS A JAK VYPOČÍTAT BMR? Online. 2019. Dostupné z: <https://gymbeam.cz/blog/co-je-to-bazalni-metabolismus-a-jak-vypocitat-bmr/>. [cit. 2023-08-20].
 - 17 WILSON, Patrick B. Does Carbohydrate Intake During Endurance Running Improve Performance? A Critical Review. Online. 2016. Dostupné z: <https://doi.org/10.1519/JSC.0000000000001430>. [cit. 2023-08-20].
 - 18 WORLD HEALTH ORGANIZATION. Carbohydrate intake for adults and children. Online. 2023. Dostupné z: https://www.ncbi.nlm.nih.gov/books/NBK593396/pdf/Bookshelf_NBK593396.pdf. [cit. 2023-08-20].
 - 19 CLEVELAND CLINIC. Runners, Here's How To Fuel Up and Stay Hydrated. Online. Dostupné z: <https://health.clevelandclinic.org/joggers-eat-drink-and-be-healthy/>. [cit. 2023-08-20].
 - 20 CROWTHER, Greg. Protein: how much do runners need? Online. 2000. Dostupné z: <https://faculty.washington.edu/crowther/Misc/RBC/protein.shtml>. [cit. 2023-08-20].

- 21 FINN, Adharanand. Cesta běžce: pohled do bájného světa japonských běžců. Přeložil Anna KUDRNOVÁ. Praha: Mladá fronta, 2017. ISBN 978-80-204-4184-3.
- 22 TANAKA, Hirofumi; MONAHAN, Kevin D. a SEALS, Douglas R. Age-predicted maximal heart rate revisited. Online. 2001. Dostupné z: <https://www.jacc.org/doi/10.1016/S0735-1097%2800%2901054-8>. [cit. 2023-08-21].
- 23 BEIS, Lukas Y.; WILLKOMM, Lena; ROSS, Ramzy; BEKELE, Zeru; WOLDE, Bezabhe et al. Food and macronutrient intake of elite Ethiopian distance runners. Online. 2022. Dostupné z: <https://doi.org/https://doi.org/10.1186/1550-2783-8-7>. [cit. 2023-08-21].
- 24 FREY, Malia. What Is the 80/20 Diet? Online. 2022. Dostupné z: <https://www.verywellfit.com/what-is-the-8020-diet-3495281>. [cit. 2023-08-21].
- 25 FOGLAR, Jakub. Fitify Preview. Online. Dostupné z: <https://dribbble.com/shots/16701069-Fitify-Preview>. [cit. 2023-08-28].
- 26 FITIFY. Generate a Free Running Plan. Online. Dostupné z: <https://runreps.com/tools/running-plan-generator/>. [cit. 2023-08-28].
- 27 NIKE. Nike Run Club. Online. Dostupné z: <https://www.nike.com/cz/nrc-app>. [cit. 2023-08-28].
- 28 FITIFY. Fitify. Online. Dostupné z: <https://gofitify.com/cs/>. [cit. 2023-08-28].
- 29 RUNREPS. RunReps. Online. Dostupné z: <https://runreps.com/>. [cit. 2023-08-28].
- 30 RUNNING.COACH. RunningCoach. Online. Dostupné z: <https://runningcoach.me/en>. [cit. 2023-08-28].
- 31 ORACLE. Servlet Overview. Online. Dostupné z: https://docs.oracle.com/cd/A97688_16/generic.903/a97680/overview.htm. [cit. 2024-03-17].
- 32 IBM. What is Java Spring Boot? Online. Dostupné z: <https://www.ibm.com/topics/java-spring-boot>. [cit. 2024-03-17].
- 33 Java virtual machine. Online. In: Wikipedia: the free encyclopedia. San Francisco (CA): Wikimedia Foundation, 2024. Dostupné z: https://en.wikipedia.org/wiki/Java_virtual_machine. [cit. 2024-03-17].
- 34 KUNČAR, Petr. Spring – IoC Kontejner. Online. Dostupné z: <https://www.itnetwork.cz/java/spring-boot/filmova-databaze/spring-ioc-kontejner>. [cit. 2024-03-17].

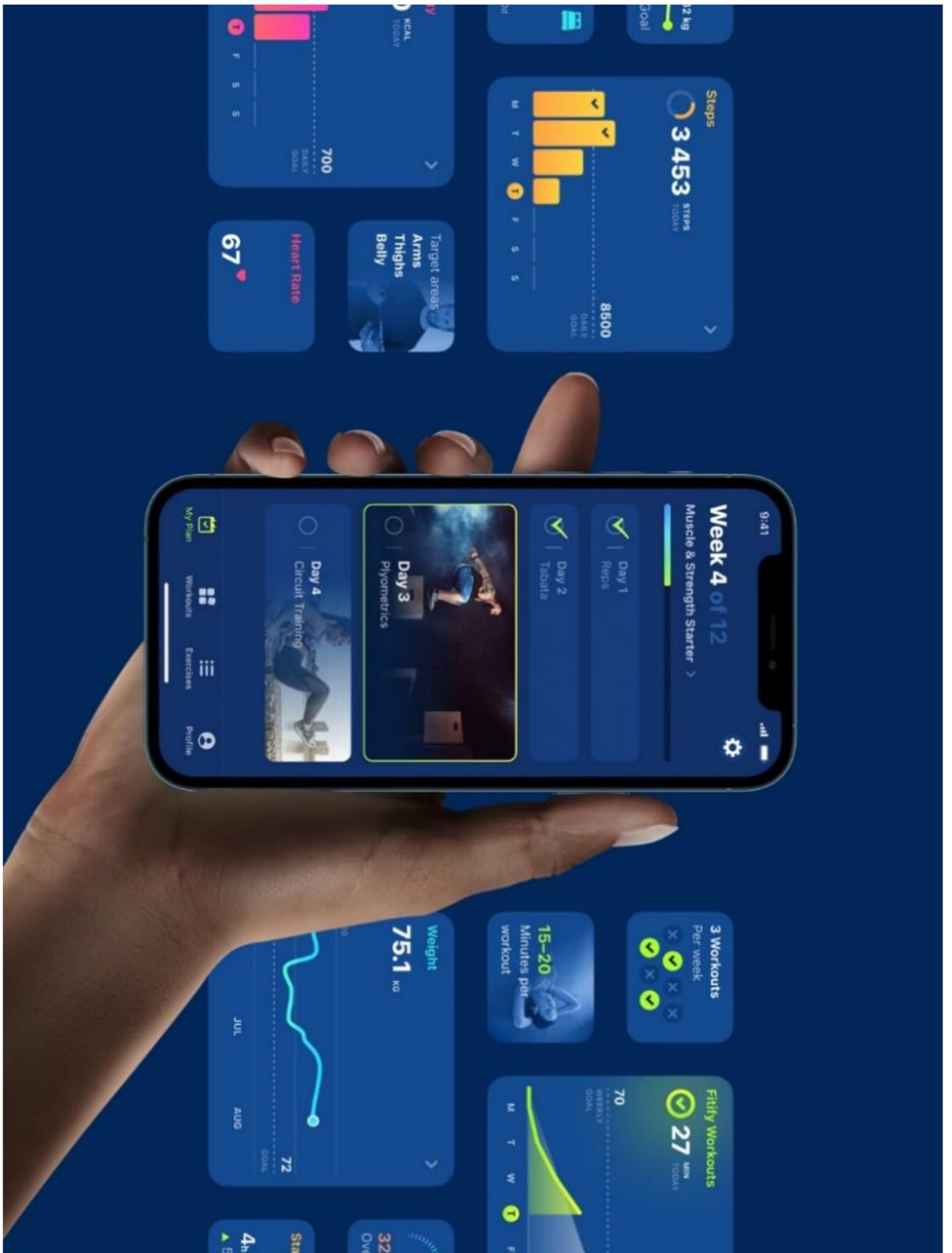
- 35 Convention over configuration. Online. In: Wikipedia: the free encyclopedia. San Francisco (CA): Wikimedia Foundation, 2024. Dostupné z: https://en.wikipedia.org/wiki/Convention_over_configuration. [cit. 2024-03-17].
- 36 W3SCHOOLS. React JSX. Online. Dostupné z: https://www.w3schools.com/react/react_jsx.asp. [cit. 2024-03-20].
- 37 JAVATPOINT. Pros and Cons of ReactJS. Online. Dostupné z: <https://www.javatpoint.com/pros-and-cons-of-react>. [cit. 2024-03-20].
- 38 MITTAL, Akash. Understanding data binding in React. Online. 2023. Dostupné z: <https://handsontable.com/blog/understanding-data-binding-in-react>. [cit. 2024-03-21].
- 39 GEEKSFORGEEKS.ORG. What is the difference between one-way data flow and two-way data binding in vue.js? Online. 2023. Dostupné z: <https://www.geeksforgeeks.org/what-is-the-difference-between-one-way-data-flow-and-two-way-data-binding-in-vue-js/>. [cit. 2024-03-20].
- 40 PETERSON, Richard. What is PostgreSQL? Introduction, Advantages & Disadvantages. Online. 2024. Dostupné z: <https://www.guru99.com/introduction-postgresql.html>. [cit. 2024-03-21].
- 41 WORLD HEALTH ORGANIZATION. Physical activity. Online. 2022. Dostupné z: <https://www.who.int/news-room/fact-sheets/detail/physical-activity>. [cit. 2024-04-10].
- 42 AGGRAWAL, Charu C. Online. Data Classification. In: Data Mining. 2015. ISBN 978-3-319-14141-1. Dostupné z: https://link.springer.com/chapter/10.1007/978-3-319-14142-8_10. [cit. 2024-04-12].
- 43 C4.5 algorithm. Online. In: Wikipedia: the free encyclopedia. San Francisco (CA): Wikimedia Foundation, 2024. Dostupné z: https://en.wikipedia.org/wiki/C4.5_algorithm. [cit. 2024-04-17].
- 44 SOFTWARE TESTING HELP. WEKA Dataset, Classifier And J48 Algorithm For Decision Tree. Online. 2024. Dostupné z: <https://www.softwaretestinghelp.com/weka-datasets/>. [cit. 2024-04-17].
- 45 THE UNIVERSITY OF WAIKATO. Pruning decision trees. Online. Dostupné z: <https://www.futurelearn.com/info/courses/data-mining-with-weka/0/steps/25392>. [cit. 2024-04-17].
- 46 AWS. What is Unit Testing? Online. Dostupné z: <https://aws.amazon.com/what-is/unit-testing/>. [cit. 2024-04-19].

- 47 IBM. What is three-tier architecture? Online. Dostupné z: <https://www.ibm.com/topics/three-tier-architecture>. [cit. 2024-04-19].
- 48 PARASCHIV, Eugen. Introduction to Spring Data JPA. Online. 2024. Dostupné z: <https://www.baeldung.com/the-persistence-layer-with-spring-data-jpa>. [cit. 2024-04-19].
- 49 REST. Online. In: Wikipedia: the free encyclopedia. San Francisco (CA): Wikimedia Foundation, 2024. Dostupné z: <https://en.wikipedia.org/wiki/REST>. [cit. 2024-04-19].
- 50 JWT.IO. Introduction to JSON Web Tokens. Online. Dostupné z: <https://jwt.io/introduction>. [cit. 2024-04-19].
- 51 KOSAKA, Mariko. Cross-Origin Resource Sharing (CORS). Online. 2018. Dostupné z: <https://web.dev/articles/cross-origin-resource-sharing>. [cit. 2024-04-19].
- 52 OWASP. Cross Site Request Forgery (CSRF). Online. Dostupné z: <https://owasp.org/www-community/attacks/csrf>. [cit. 2024-04-19].
- 53 OWASP. Cross Site Scripting Prevention Cheat Sheet. Online. Dostupné z: https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html. [cit. 2024-04-19].
- 54 BCRYPT-GENERATOR.COM. Online Bcrypt Hash Generator & Checker. Online. Dostupné z: <https://bcrypt-generator.com/>. [cit. 2024-04-19].

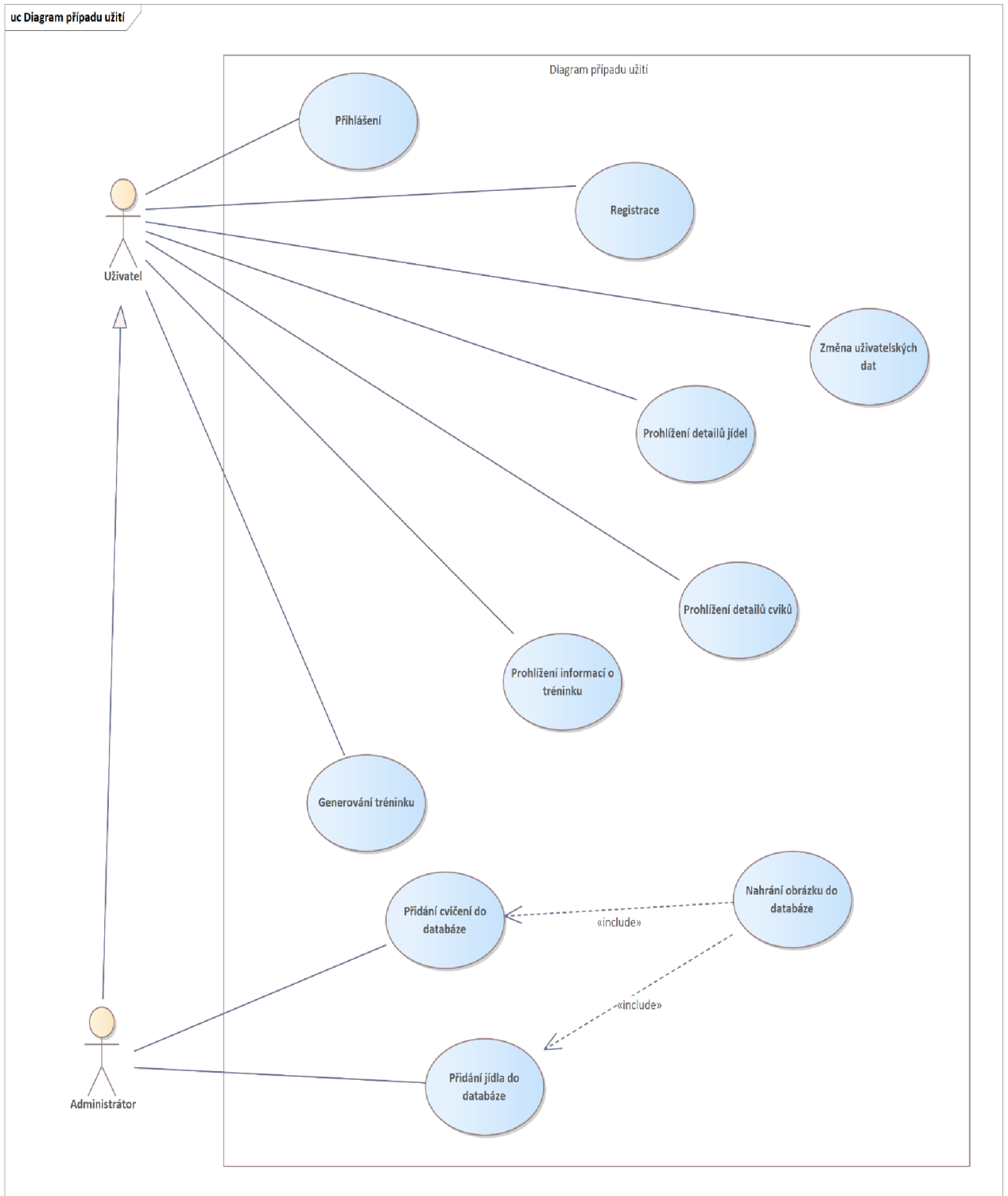
13 Přílohy

Příloha č. 1

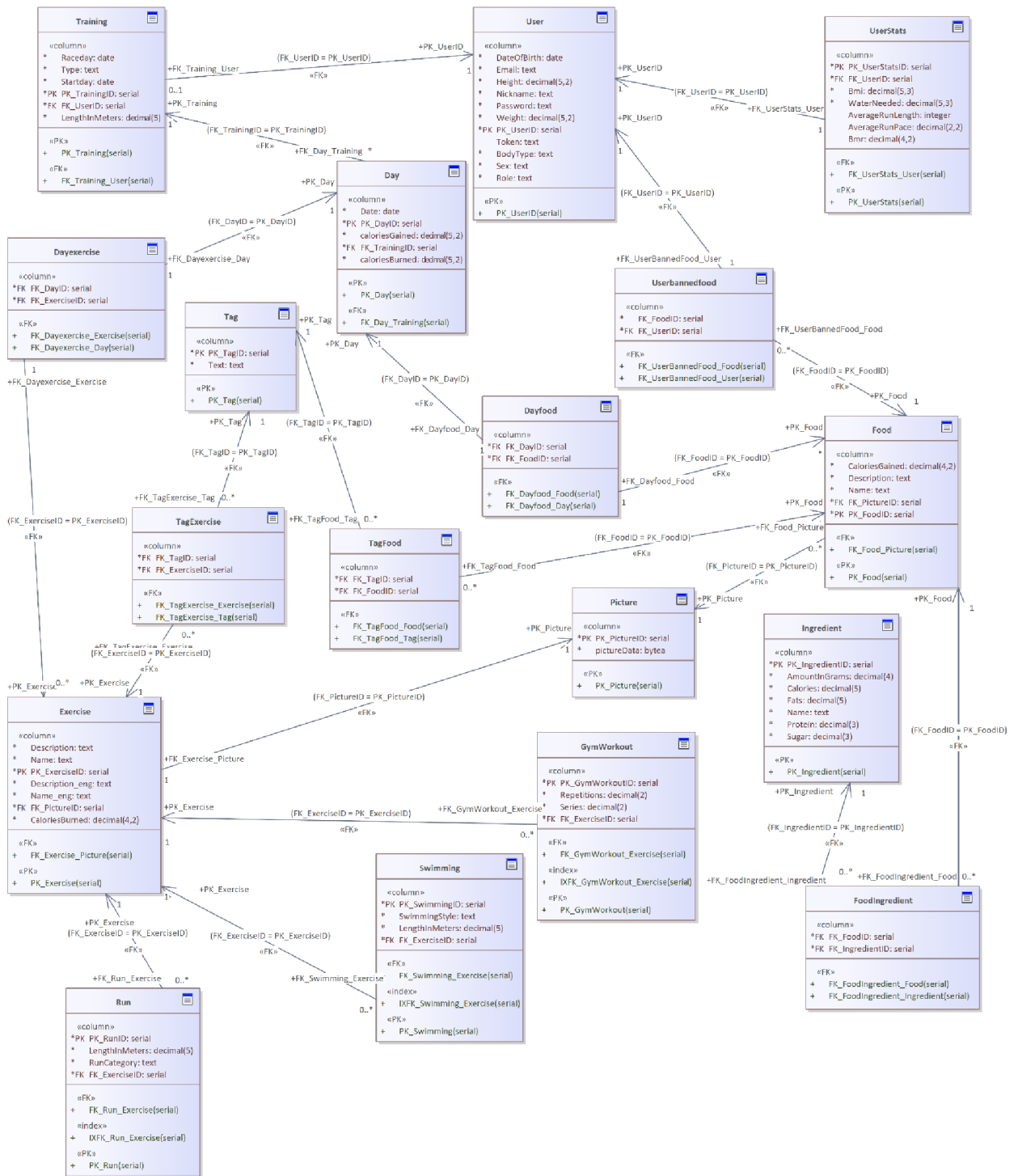
Week	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday	Weekly Aim
1	4000 KMs Steady Run	0 KMs Rest Day	4000 KMs Intervals	0 KMs Rest Day	0 KMs Rest Day	4000 KMs Long Run	0 KMs Rest Day	12000 KMs
2	4204 KMs Steady Run	0 KMs Rest Day	3003 KMs Intervals	0 KMs Rest Day	0 KMs Rest Day	4804 KMs Long Run	0 KMs Rest Day	12011 KMs
3	4207 KMs Steady Run	0 KMs Rest Day	3005 KMs Intervals	0 KMs Rest Day	0 KMs Rest Day	4808 KMs Long Run	0 KMs Rest Day	12020 KMs
4	3 KMs Steady Run	0 KMs Rest Day	3 KMs Intervals	0 KMs Rest Day	0 KMs Rest Day	3 KMs Long Run	0 KMs Rest Day	9 KMs
5	0 KMs Rest Day	0 KMs Rest Day	3 KMs Steady Run	0 KMs Rest Day	0 KMs Rest Day	1 KMs Intervals	10 KMs Race Day	14 KMs

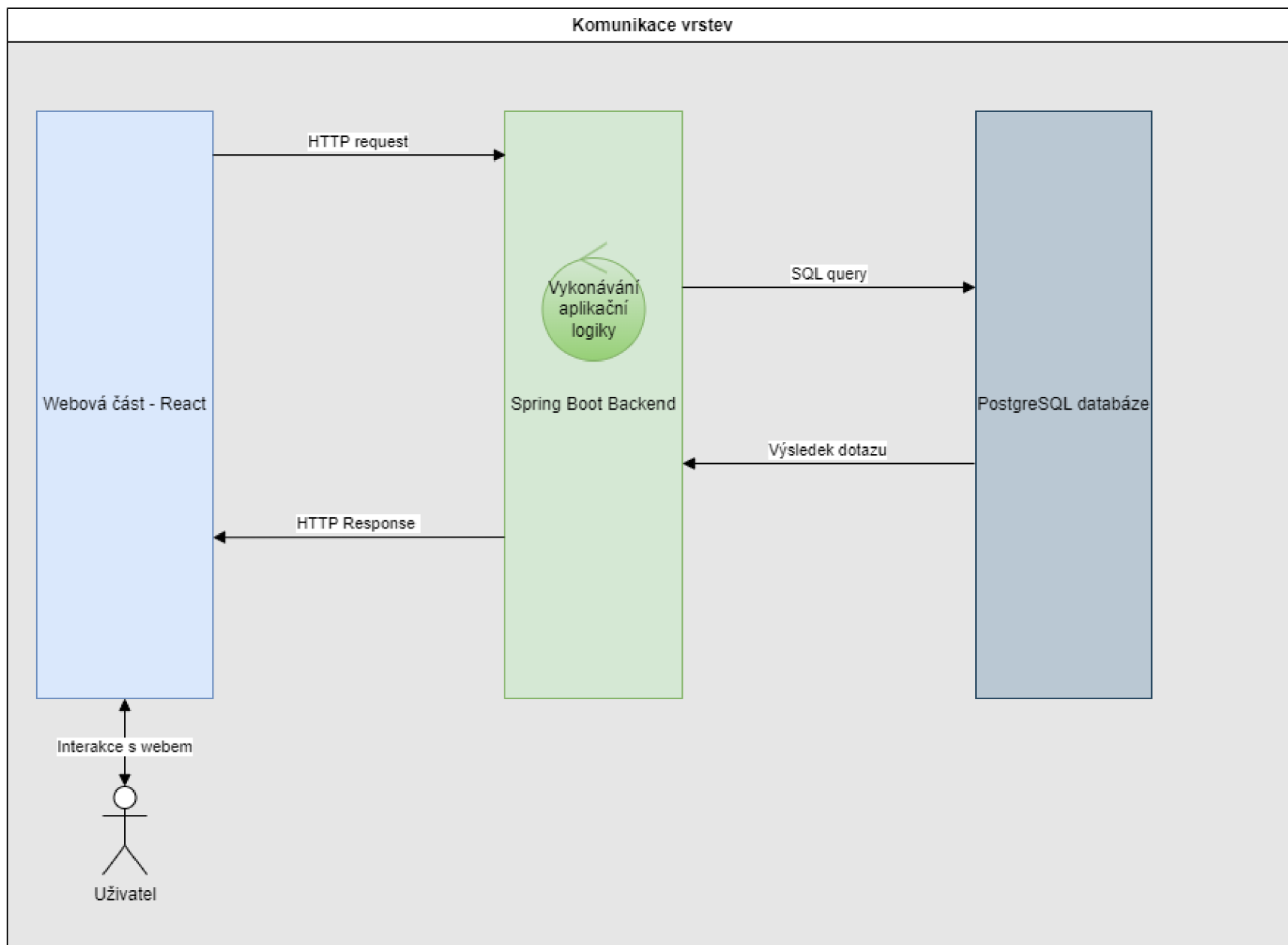


uc Diagram případu užití



class DDL





Runora

ENG

Dnes

Datum: 2024-04-24
Aktivní spálené kalorie:
345
Cvičení:
Dřep se závažím
Angličáky
Výpad
Přítaha na hrazdě
Nekonečné Schody
Plank

Kalorický příjem (kcal)
2450

Jídelníček:

Tofu

Omeleta se špenátem
a rajčaty

Smoothie s ovocem

Těstoviny s marinárou
a fazolemiGrilované kuřecí prsa
se zeleninou

Kuřecí steak s

Zítra

Datum: 2024-04-25
Aktivní spálené kalorie:
1300
Cvičení:
Běh do kopce
Vytrvalostní běh

Kalorický příjem (kcal)
2450

Jídelníček:

Hummus s čerstvou
zeleninou

Smoothie s ovocem

Avokádový toast s vejci

Losos na grilu s
pečenými sladkými
bramboryGrilované kuřecí prsa
se zeleninouKuřecí kousky s batáty
a brokolicí

Pokrok

Dní tréninku:
pokročeno v tréninku:
33%



Datum závodu:
2024-05-24

**Runora**

Trénink

Profil

CZE

ENG

Osobní informace

Uživatelské jméno **admin1**Email **a@admin.cz**Věk **21**Pohlaví **Muž**Výška **182**Váha **72**Typ postavy **Atletická**

ZMĚNIT UŽIVATELSKÁ DATA

Osobní statistiky

Doporučený denní příjem
vody**2.376 L**Body Mass Index
21.737

Přehled tréninků

RUN

Aktuální trénink

Datum zahájení přípravy:

2024-04-24

Datum závodu:

2024-05-24

Délka: 5000

OCR

Skončený trénink

Datum zahájení přípravy:

2024-02-02

Datum závodu:

2024-04-20

Délka: 5600

ODHLÁSIT

NÁSTROJE ADMINISTRÁTORA

14 Zadání práce z IS

Zadání bakalářské práce

Autor:	Alex Zamastil
Studium:	I2100304
Studijní program:	B1802 Aplikovaná informatika
Studijní obor:	Aplikovaná informatika
Název bakalářské práce:	Webová aplikace pro přípravu na závody
Název bakalářské práce AJ:	A web application for race preparation

Cíl, metody, literatura, předpoklady:

Cílem je navrhnout a implementovat webovou aplikaci zaměřenou na přípravu na sportovní závody.

Osnova

1. Úvod
2. Analýza a návrh
3. Implementace
4. Shrnutí výsledků
5. Závěr
6. Literatura

Zadávací pracoviště:	Katedra informatiky a kvantitativních metod, Fakulta informatiky a managementu
Vedoucí práce:	Ing. Jan Krunčik
Datum zadání závěrečné práce:	15.10.2022