

České vysoké učení technické v Praze
Fakulta elektrotechnická

katedra počítačů

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Kryštof Šplíchal**

Studijní program: Otevřená informatika

Obor: Softwarové systémy

Název tématu: **Software pro evidenci a analýzu sportovního tréninku**

Pokyny pro vypracování:

Seznamte se s aktuálním stavem aplikace ScApp, která vznikla v rámci bakalářské práce O. Janaty a s populárními aplikacemi pro řízení sportovního tréninku. Definujte konkrétní požadavky pro dokončení aplikace a jejich prioritu. Zaměřte se především na oblast cviků, jejich realizaci a začlenění do stávajícího systému. Následně vybrané požadavky podrobně analyzujte, implementujte a implementaci ověřte testy.

Seznam odborné literatury:

1. Janata Ondřej: Software pro evidenci a analýzu sportovního tréninku, Bakalářská práce, ČVUT FEL, 2014

Vedoucí: Ing. Ondřej Macek, Ph.D.

Platnost zadání: do konce letního semestru 2015/2016



V Praze dne 26. 3. 2015

České vysoké učení technické v Praze
Fakulta elektrotechnická
Katedra počítačů



Bakalářská práce

Software pro evidenci a analýzu sportovního tréninku

Kryštof Šplíchal

Vedoucí práce: Ing. Ondřej Macek, Ph.D.

Studijní program: Otevřená informatika, Bakalářský

Obor: Softwarové systémy

20. května 2015

Poděkování

Chtěl bych tímto poděkovat vedoucímu práce Ing. Ondřeji Mackovi, Ph.D. za veškeré rady a pomoc poskytnuté v průběhu realizace práce a za to, že se ujal vedení mé práce. Dále náleží velké poděkování Bc. Ondřeji Janatovi za veškerý poskytnutý čas a energii v rámci konzultací.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 20. 5. 2015

.....

Abstract

The main goal of this work is to expand the functionality of an opensource application for sport training management ScApp with detailed specification of exercises and exercise plans. The motivation is to offer a wide spectrum of new possible uses to the freely available application and thus support an alternative of the commercial solutions. I designed and implemented the extension based on analysis of the original application and new requirements, the whole new functionality is tested with BDD tests.

Abstrakt

Cílem práce je rozšířit opensource aplikaci pro správu sportovních tréninků ScApp o část pro detailní specifikaci cviků a cvičebních plánů. Motivací je poskytnout volně dostupné aplikaci další široké spektrum možných využití a podpořit tak alternativu komerčních řešení. Rozšíření jsem navrhl a implementoval na základě analýzy původní aplikace a nově vzniklých požadavků, veškerá nová funkcionality je otestována pomocí BDD testů.

Obsah

| | | |
|----------|---|-----------|
| 1 | Úvod | 1 |
| 1.1 | Cíle práce | 1 |
| 1.2 | Podobné aplikace | 2 |
| 1.2.1 | Virtuagym | 2 |
| 1.2.1.1 | Význam vývoje ScApp | 2 |
| 2 | Analýza | 3 |
| 2.1 | Seznámení se s původní aplikací | 3 |
| 2.2 | Důležité používané pojmy | 3 |
| 2.3 | Popis problému | 4 |
| 2.3.1 | Správa cviků | 4 |
| 2.3.2 | Správa cvičebních plánů | 4 |
| 2.4 | Systémové požadavky | 5 |
| 2.4.1 | Funkční požadavky | 5 |
| 2.4.2 | Nefunkční požadavky | 6 |
| 2.4.3 | Ostatní požadavky na projekt | 6 |
| 2.5 | Případy užití | 7 |
| 2.5.1 | Pokrytí požadavků uživatelskými scénáři | 9 |
| 3 | Návrh aplikace | 11 |
| 3.1 | Doménový model | 11 |
| 3.1.1 | Exercise | 11 |
| 3.1.1.1 | Accessibility | 12 |
| 3.1.1.2 | ExerciseWithSets | 12 |
| 3.1.2 | ExerciseSetup | 12 |
| 3.1.2.1 | ExerciseSetSetup | 12 |
| 3.1.3 | ExerciseMeasurement | 13 |
| 3.1.3.1 | ExerciseSetMeasurement | 13 |
| 3.1.4 | ExerciseStep | 13 |
| 3.1.5 | ExerciseImage | 13 |
| 3.1.6 | ExerciseBundle | 13 |
| 3.1.7 | Unit | 13 |
| 3.1.8 | Plan | 14 |
| 3.1.9 | ExerciseRealization | 14 |
| 3.1.9.1 | ExerciseSetRealization | 14 |

| | | |
|----------|--|-----------|
| 3.1.10 | ExerciseRealizationSetup | 14 |
| 3.1.10.1 | ExerciseSetRealizaitonSetup | 14 |
| 3.1.11 | ExerciseRealizationMeasurement | 14 |
| 3.1.11.1 | ExerciseSetRealizationMeasurement | 15 |
| 3.2 | Diagram doménového modelu | 15 |
| 3.3 | Podrobné diagramy modelu aplikace | 17 |
| 3.4 | Implementace | 19 |
| 3.4.1 | Framework Ruby on Rails | 19 |
| 3.4.1.1 | Model-View-Controller | 19 |
| 3.4.1.2 | Převod na Rails 4.1 | 19 |
| 3.4.2 | Knihovna JQuery | 19 |
| 3.5 | Ukázky výsledné implementace | 20 |
| 4 | Testování | 25 |
| 4.1 | Behavior Driven Development a Ruby on Rails | 25 |
| 4.1.1 | Ukázka <code>step definitions</code> pro <code>Exercise</code> | 25 |
| 4.1.2 | JQuery v testování | 26 |
| 4.2 | Akceptační testy | 27 |
| 4.3 | Testování s uživateli | 27 |
| 4.3.1 | Výběr účastníků testu | 27 |
| 4.3.2 | Průběh testování | 27 |
| 4.3.3 | Metody pro testování | 28 |
| 4.3.3.1 | Metoda testování kognitivním průchodem | 28 |
| 4.3.3.2 | Metoda testování heuristickou evaluací | 28 |
| 4.3.4 | Výsledky testování s uživateli | 29 |
| 4.3.4.1 | Testování metodou kognitivního průchodu | 30 |
| 4.3.4.2 | Testování metodou heuristické evaluace | 31 |
| 4.3.4.3 | Ostatní postřehy uživatelů vzniklé při testování | 32 |
| 4.4 | Zhodnocení testování | 32 |
| 4.4.1 | Zhodnocení BDD | 32 |
| 4.4.2 | Zhodnocení testování s uživateli | 33 |
| 5 | Zhodnocení a budoucnost projektu | 35 |
| 5.1 | Testovací nasazení | 35 |
| 5.2 | Budoucnost | 35 |
| 5.2.1 | Měření realizací | 36 |
| 5.2.2 | Mobilní aplikace | 36 |
| 5.2.3 | Globální repozitář cviků, sad a plánů | 36 |
| 6 | Závěr | 37 |
| | Literatura | 37 |
| A | Obsah přiloženého CD | 41 |

| | | |
|----------|--|-----------|
| B | Výsledky testování s uživateli | 43 |
| B.1 | Metoda testování kognitivním průchodem | 43 |
| B.2 | Metoda testování heuristickou evaluací | 48 |

Seznam obrázků

| | | |
|-----|---|----|
| 3.1 | Doménový model | 16 |
| 3.2 | Diagram tříd - část pro správu cviků | 17 |
| 3.3 | Diagram tříd - část pro správu plánů | 18 |
| 3.4 | Ukázka obrazovky pro editaci cvičebních plánů | 20 |
| 3.5 | Ukázka obrazovky pro detail a editaci cviku | 21 |
| 3.6 | Ukázka oken pro editaci nastavení a sérií realizace cviku | 22 |
| 3.7 | Ukázka obrazovky pro editaci kroků cviku | 23 |

Seznam tabulek

| | | |
|-----|---|----|
| 2.1 | Pokrytí funkčních požadavků uživatelskými scénáři | 10 |
| 4.2 | Seznam nálezů pro metodu kognitivního průchodu | 30 |
| 4.4 | Seznam nálezů pro metodu heuristické evaluace | 31 |

Seznam výpisů kódu

| | | |
|-----|---|----|
| 4.1 | Ukázka <code>step</code> definitions pro Exercise v jazyce Ruby | 25 |
| 4.2 | Ukázka testování jQuery Sortable | 26 |
| 4.3 | Ukázka testování jQuery Draggable | 26 |

Kapitola 1

Úvod

Cílem práce je vytvořit systém pro podporu plánování sportovních tréninků.

Práce rozšiřuje systém vytvořený a popsáný v bakalářské práci Bc. Ondřeje Janaty[7]. V práci autor vytvořil aplikaci pro správu sportovních tréninků ScApp. K práci a aplikaci [7] se budu ve své práci pro přehlednost odkazovat termíny *původní aplikace*, *původní práce*. Rozšíření, které jsem vytvořil, budu nazývat *aplikace*, *práce*.

Původní aplikace dobře pokrývá oblast správy samotných tréninků, chyběla však možnost *specifikovat konkrétní cviky a vytvořit detailní cvičební plán* - tedy využít vytvořené cviky v rámci tréninkové hodiny. To může být velmi žádaná vlastnost např. pro posilovny, kde trenéři chtějí detailně specifikovat obsah svých tréninků. Rozšíření, které jsem v rámci práce vytvořil, se pokusí poskytnout původní aplikaci právě tuto funkcionalitu.

1.1 Cíle práce

Hlavním cílem práce je vytvořit aplikaci pro obecnou specifikaci *cviků a tréninkových plánů*, do kterých jsou tyto cviky zařazeny jako tzv. *realizace cviků*. Aplikace bude posléze začleněna do původní aplikace, kterou tak rozšíří. Nově vytvořená aplikace tedy musí následovat použité technologie a konvence tak, aby celý systém mohl být dále vyvíjen a rozšiřován v duchu *opensource* vývoje¹. Toto následuje ideu původní práce, kde její autor systém začal navrhovat od počátku s tímto záměrem.

Funkcionalita specifikace jednotlivých cviků a jejich realizací původní aplikaci přidává další široké spektrum možných využití (zejména v posilovnách a podobných typech sportu, kde je žádoucí specifikovat přesně i obsah tréninků spolu se samotnými hodinami a souvislými organizačními záležitostmi), proto si práce klade za cíl celou aplikaci posunout dále v univerzálnosti tak, aby s případnými drobnými úpravami bylo možné ji použít na mnoha dalších místech. Původní aplikace byla vyvíjena primárně *pro potřeby tenistů*, se snahou o to, aby bylo možné ji dále bezproblémově rozšířit i pro další druhy sportů. Této myšlenky se snaží tato práce držet a aplikovat ji i na zde vzniklé nové součásti, aby systém mohl být dále dobře rozvíjen.

¹Autor původní aplikace se ji rozhodl šířit pod *opensource* licencí GPLv3[17], aby byl umožněn vývoj komunitou.

1.2 Podobné aplikace

Autor původní práce podal přehled aktuálních aplikací s podobným zaměřením.[7] Výsledkem průzkumu byl fakt, že na trhu je *malé množství aplikací podobného typu* a rozsahu a případné *alternativy jsou komerční*. Jako nejbližší alternativu autor původní práce označil aplikaci **TrainingPeaks**[24], která ovšem mimo absence *opensource* filosofie postrádala i možnost specifikovat detailně cviky a cvičební plány.

Situace na trhu je v době psaní této práce podobná, s výjimkou aplikace **Virtuagym**.

1.2.1 Virtuagym

Virtuagym [25] je velmi rozsáhlá platforma, jejíž oblasti funkcionality se ve velké míře překrývají s projektem **ScApp**. Z oblasti již zpracované funkcionality **ScApp** umožňuje **Virtuagym** kompletní správu tréninků s obdobným rozsahem, správu finančních záležitostí, uživatelských vztahů a obecných měření. Z nově vznikající funkcionality je to správa cviků a cvičebních plánů a měření vázaných na realizace cviků. Dále stojí za zmínku správa nutričních plánů, možnost specifikovat cíle pro hráče (slouží pro motivaci ke cvičení), možnost zavést v rámci klubu virtuální měnu a spravovat akce v systému pomocí této měny, je zde i dobře zpracován sociální aspekt. Veškerá tato funkcionalita je zpracována v moderním rozhraní, existuje i populární mobilní aplikace, která zjednodušuje práci v dynamickém prostředí.

1.2.1.1 Význam vývoje ScApp

I při existenci velmi podobné a dobře zpracované aplikace, jakou je **Virtuagym**, má další vývoj **ScApp** velký význam - vzniká *silná alternativa* s možností *využití zdarma* a *vývoje přispěvateli z celého světa*. Projekt má tedy *větší volnost* v rámci možných směrů vývoje. Projekty podobné **Virtuagym** mohou v budoucnu posloužit jako inspirace.

Kromě těchto obecných výhod poskytuje **ScApp** v této fázi i některé konkrétní výhody plynoucí z návrhu aplikace (kapitola 3) - *detailnější nastavení* cviků a jejich sérií, možnost zaznamenávat *měření pro jakoukoli obecnou nebo konkrétní vlastnost* a některé další.

V jiných oblastech pochopitelně **ScApp** může mít oproti komerčním alternativám nedostatky, avšak je zde potenciál pro jejich odstranění v případě dalšího aktivního vývoje, pokud se podaří vybudovat stabilní komunitu - tomuto procesu by měla napomoci snaha autorů projektu tvořit systém tak, aby bylo možné jej v budoucnu dobře rozšiřovat.

Kapitola 2

Analýza

V průběhu analýzy požadavků a návrhu aplikace byl kladen důraz na vytyčené cíle a pokračování ve vzorech, které byly zvoleny pro původní aplikaci.

Funkční požadavky poskytly základ pro tvorbu uživatelských scénářů. V kombinaci s *Behavior Driven Development* (popsáno v sekci 4.1) tedy byl dobře a přesně formulován obsah aplikace ještě před samotnou implementací.

2.1 Seznámení se s původní aplikací

Před zahájením návrhu a implementace jsem se důkladně seznámil s původní prací. Autor v ní popsal a navrhnul kostru rozšíření, kde byly specifikovány hlavní entity, které jsou v této práci využity. [7] Tento návrh jsem v rámci konzultací s autorem původní práce upravil na stávající model (sekce 3.1).

Před návrhem aplikace jsem se s autorem původní práce domluvil na podobě rozšíření - pro budoucnost celého projektu bude v této práci nejvíce důležitý co nejúplnější popis hlavních entit a důraz by měl být kladen na budoucí možnost rozšíření. Původní model tedy prošel změnami, o kterých podrobně pojednává sekce 3.1.

Provedl jsem tzv. `fork`¹ původního repozitáře na `GitHub`², aby bylo možné posléze jednoduše integrovat novou funkcionalitu do původní aplikace pomocí tzv. `pull requestu`³ nebo pokračovat s vývojem v nové verzi, podle společného rozhodnutí o budoucnosti projektu.

2.2 Důležité používané pojmy

Z důvodu snadnější orientace v textu uvádím některé pojmy, které budou hojně používány v následujících částech. Část pojmů vychází z původní aplikace[7], ostatní pojmy se vyskytují nově v rámci analýzy této aplikace.

¹Standardní způsob zkopírování repozitáře `GitHub` s jednoduchou možností zpětné integrace

²Původní repozitář[8]: <https://github.com/hack006/scapp>, nový repozitář[26]: <https://github.com/krystofspl/scapp>

³Standardní způsob integrace rozšíření v rámci `GitHub` repozitáře

Trénink Aktivita v rámci sportovního zařízení, jde o nejobecnější specifikaci této aktivity, jsou zde definováni především pravidelní hráči, trenéři a *tréninkové hodiny*

Tréninková hodina Obecná instance *tréninku*, je zde specifikován zejména (pravidelný) čas konání, délka a finance

Realizace tréninkové hodiny Konkrétní instance *tréninkové hodiny*, tvoří se pro ni *cvičební plány* pro jednotlivé hráče, specifikují se další přítomní trenéři a hráči a metriky

Cvik Obecná specifikace jednotky cvičební aktivity. Cviky lze podrobně specifikovat a řadit je do *sad* (soukromých i veřejných skupin s nějakou společnou vlastností). Podrobněji popsáno v 2.3.1.

Realizace cviku Využití *cviku* v rámci konkrétního *cvičebního plánu* pro nějakého hráče v rámci *realizace tréninkové hodiny*

Cvičební plán Sestavení řazených *realizací cviků* pro hráče zapsané na konkrétní *realizaci tréninkové hodiny*. Podrobněji popsáno v 2.3.2.

2.3 Popis problému

Aplikace poskytuje nově funkcionalitu, kterou lze rozdělit na dvě hlavní skupiny - *správa cviků* a *správa cvičebních plánů* v rámci realizací tréninkových hodin.

Tato ohniska jsou doplněna o různé propojovací a rozšiřující prvky.

Entity jsou popsány podrobně včetně atributů v sekci 3.1.

2.3.1 Správa cviků

Cvik je základní jednotkou aplikace. Specifikuje jej trenér buď pro využití v rámci některého ze svých tréninků, nebo pro využití celého klubu, což systém striktně rozlišuje. Může se jednat o jednotku tréninku např. v posilovně nebo cvik může popisovat i obecnější aktivitu - jedná se tedy o nedělitelnou⁴ část libovolného tréninku (zde myšleno obecně, bez vazby k entitám aplikace).

Správa cviků umožňuje tyto cviky vytvářet, detailně specifikovat, umisťovat do *sad* a prohlížet s různými kritérii.

Použití nějakého cviku ve cvičebním plánu budu nazývat *realizace cviku*.

2.3.2 Správa cvičebních plánů

Cvičební plán spojuje realizace cviků v rámci realizací tréninkových hodin (entita **Training-LessonRealization**). Tedy realizace tréninkové hodiny obsahuje pro každého zúčastněného hráče právě jeden plán, který obsahuje žádnou až n realizací cviků. Tento počet je omezen celkovou délkou realizace tréninkové hodiny.

Správa cvičebních plánů tedy umožňuje trenérům specifikovat *průběh a obsah* svých tréninků *pro každého zúčastněného hráče*. Tyto plány mohou následně být zobrazeny na přehledné obrazovce jak trenérem, tak zúčastněným hráčem.

⁴Kromě sérií cviku, popsáno v sekci 3.1.1.2.

2.4 Systémové požadavky

Pro úspěšnou analýzu problému a návrh řešení bylo nutné nejprve formulovat požadavky na systém. Ty jsem podle konvencí [1] rozdělil na 3 hlavní kategorie - *funkční*, *nefunkční* a *ostatní* požadavky.

2.4.1 Funkční požadavky

Funkční požadavky popisují, co má výsledná aplikace splňovat v rámci funkcionality. Na základě těchto požadavků se specifikují uživatelské scénáře a formuje celkový návrh aplikace. Každý požadavek by měl být nedělitelný a specifikovat jednu konkrétní měřitelnou funkci systému, pokud jsou však některé požadavky velmi podobné a úzce souvislé, uvádím je pohromadě. [1]

- F1: **Správa cviků**
Systém umožní trenérům přidávat, prohlížet, upravovat a mazat vlastní cviky.
- F2: **Specifikace nastavení a měření cviků**
Systém umožní trenérům specifikovat nastavení a měření, která se týkají cviku.
- F3: **Klonování cviků**
Systém umožní trenérům zkopírovat libovolný použitý, jim dostupný cvik včetně nastavení, měření a jiných vlastností v rámci jedné akce.
- F4: **Filtrování cviků**
Systém bude poskytovat možnost filtrovat cviky podle názvu a data vytvoření.
- F5: **Správa sad cviků**
Systém umožní trenérům přidávat, prohlížet, upravovat a mazat vlastní sady cviků.
- F6: **Propojení sad cviků se cviky**
Systém umožní trenérům přidávat jejich i veřejné cviky do sad.
- F7: **Správa realizací cviků v rámci cvičebního plánu**
Systém umožní trenérům přidávat, prohlížet, upravovat a mazat realizace cviků pro hráče v jejich tréninkové hodině.
- F8: **Specifikace nastavení pro realizaci cviku**
Systém poskytne možnost určit nastavení pro každou realizaci cviku, spolu s příslušnou hodnotou a poznámkou.
- F9: **Zobrazení cvičebních plánů**
Systém umožní zobrazit cvičební plán trenérům (pro všechny hráče v jejich hodině) a hráčům (pouze jejich plán) na přehledné obrazovce.

2.4.2 Nefunkční požadavky

Nefukční požadavky popisují další podmínky, které má systém splňovat, netýkají se však funkcionality systému. Spíše jsou zde definovány kvalitativní požadavky, které jsou také nedílnou součástí specifikace - např. použité technologie, bezpečnost, výkon. [1]

- N1: Aplikace bude implementována ve frameworku **Ruby on Rails** [20] verze 4, dále bude využívat technologie **HAML** ⁵, **SCSS** ⁶, **jQuery** [22], **AJAX** [15].
- N2: Aplikace bude plně funkční v aktuálních verzích prohlížečů Mozilla Firefox a Google Chrome (v době psaní práce po řadě verze 37 a 42).
- N3: Aplikace bude podporovat na hlavních obrazovkách mobilní zařízení, bude tedy použit tzv. *responsivní design*.
- N4: Veškeré akce v systému budou omezeny v rámci role uživatele.

2.4.3 Ostatní požadavky na projekt

- D1: Aplikace bude otestována pomocí akceptačních testů.
- D2: Zdrojový kód bude obsahovat komentáře.
- D3: Aplikace bude připravena na začlenění do původní aplikace.

⁵<http://haml.info/>

⁶<http://sass-lang.com/>

2.5 Případy užití

Případy užití jsou definovány v rámci akceptačních testů (sekce 4.1). Ty jsou tvořeny ve formě, která je ekvivalentní standardnímu zápisu scénářů [2]. Zde uvedu některé z hlavních případů užití pro ilustraci použití systému.

U uvedených případů užití uvádím odpovídající `feature`⁷, pokrytí funkčních požadavků `features` je přehledně uvedeno v tabulce v sekci 2.5.1.

UC1: Přidat nový soukromý cvik

Odpovídající `feature`: `exercises/new_exercise.feature` - Scenario: As coach I can add a new private exercise

Předpoklady: Uživatel má roli trenéra. Uživatel je přihlášen.

1. Případ užití začíná, když uživatel navštíví stránku s výčtem jemu dostupných cviků.
2. Systém zobrazí tlačítko s nápisem „Přidat nový cvik“.
3. Uživatel klikne na zobrazené tlačítko.
4. Systém zobrazí formulář pro vyplnění údajů o cviku.
5. Uživatel vyplní formulář daty, povinně s následujícími položkami, volitelně další nepovinná pole.
 - **Název:** Název cviku.
 - **Popis:** Popis cviku.
 - **Typ cviku:** Výběr typu cviku z možností *Jednoduchý cvik* a *Cvik se sériemi*.
6. Uživatel potvrdí vložení tlačítkem „Přidat cvik“.
7. Systém provede validaci korektnosti dat (formát, unikátnost názvu).
8. Systém zobrazí obrazovku s detailem vloženého cviku.

Alternativní scénář: Zadaná data nebyla validní.

1. Systém zobrazí varování nad formulářem a označí špatně vyplněná pole.
2. Uživatel znovu vyplní formulář s opravenými daty.
3. Uživatel potvrdí vložení tlačítkem „Přidat cvik“.
4. Systém provede validaci korektnosti dat (formát, unikátnost názvu).
5. Systém zobrazí obrazovku s detailem vloženého cviku.

⁷Jednotka akceptačních testů odpovídající scénáři, viz sekce 4.1.

UC2: Zařadit cvik do sady cviků

Odovídající feature: `exercises/exercise_bundles/connect_exercise.feature` - Scenario: As a coach I can connect my private/global exercise to (my) private exercise bundle

Předpoklady: Uživatel má roli trenéra. Uživatel je přihlášen.

1. Příklad užití začíná, když uživatel navštíví stránku s výčtem jeho sad cviků.
2. Systém zobrazí tabulku se všemi sadami cviků, které jsou dostupné uživateli.
3. Uživatel klikne v tabulce na odkaz s názvem libovolné sady cviků.
4. Systém zobrazí detail sady cviků.
5. Uživatel klikne na tlačítko „Přidat / Odebrat cviky“.
6. Systém zobrazí tabulku s dostupnými cviky. Cviky již obsažené v sadě budou zvýrazněny zelenou barvou, ostatní cviky červenou barvou.
7. Uživatel klikne v řádku pro daný cvik na tlačítko „Přidat“.
8. Systém zobrazí potvrzovací hlášku.
9. Systém zobrazí v tabulce právě přidaný cvik zvýrazněný zelenou barvou.

UC3: Přidat realizaci cviku v rámci cvičebního plánu

Odovídající feature: `exercises/exercise_realizations/new_exercise_realization.feature` - Scenario: As coach I can add a new realization

Předpoklady: Uživatel má roli trenéra. Uživatel je přihlášen. Uživatel je vlastníkem tréninkové hodiny.

1. Příklad užití začíná, když uživatel navštíví stránku pro editaci plánů v rámci své tréninkové hodiny.
2. Uživatel najde v tabulce cviků cvik, jehož realizaci chce vytvořit.
3. Uživatel „uchopí“ myší v tabulce s cviky vybraný cvik.
4. Uživatel přetáhne myší cvik do plánu na pozici mezi ostatní realizace, kam chce cvik zařadit.
5. Systém vizuálně potvrdí přidání realizace a aktualizuje posunuté časové údaje všech ostatních realizací v plánu.

UC4: Řadit série v realizaci cviku

Odovídající feature: `exercises/exercise_set_realizations/edit_exercise_set_realizations.feature` - Scenario: As a coach, who created the realization, I want to change order of sets in it

Předpoklady: Uživatel má roli trenéra. Uživatel je přihlášen. Uživatel je vlastníkem tréninkové hodiny. Vybraný cvik je typu „Cvik se sériemi“. Vybraný cvik obsahuje alespoň dvě série.

1. Případ užití začíná, když uživatel navštíví stránku pro editaci plánů v rámci své tréninkové hodiny.
2. Uživatel najede myší na některou realizaci cviku v rámci některého z plánů.
3. Systém zobrazí v oblasti realizace tlačítko „Upravit“.
4. Uživatel klikne na tlačítko „Upravit“.
5. Systém zobrazí editační obrazovku.
6. Uživatel klikne na tlačítko „Upravit série“.
7. Systém zobrazí editační obrazovku pro editaci sérií.
8. Uživatel „uchopí“ myší v tabulce se sériemi některou ze sérií a potažením myši změni pořadí sérií.
9. Systém aktualizuje a zobrazí tabulkový výpis sérií cviku a jejich časů.

2.5.1 Pokrytí požadavků uživatelskými scénáři

Veškeré uživatelské scénáře jsou umístěny ve složce **features/**, umístění konkrétního scénáře je vždy v příslušném řádku v následující tabulce. Všechny soubory mají příponu **.feature**.

| Název scénáře | Umístění scénáře | Funkční požadavky | | | | | | | | |
|------------------------------|-------------------------------------|-------------------|----|----|----|----|----|----|----|----|
| | | F1 | F2 | F3 | F4 | F5 | F6 | F7 | F8 | F9 |
| Cviky | exercises/ | | | | | | | | | |
| Přidat cvik | new_exercise | X | | | | | | | | |
| Zobrazit cviky | show_exercises | X | | | X | | | | | |
| Upravit cvik | edit_exercise | X | | X | | | | | | |
| Smazat cvik | delete_exercise | X | | | | | | | | |
| Sady cviků | exercise_bundles/ | | | | | | | | | |
| Spravovat cviky | connect_exercise | | | | | | X | | | |
| Odebrat sadu | delete_exercise_bundle | | | | | X | | | | |
| Upravit sadu | edit_exercise_bundle | | | | | X | | | | |
| Přidat sadu | new_exercise_bundle | | | | | X | | | | |
| Zobrazit sady | show_exercise_bundle | | | | | X | | | | |
| Měření cviku | exercise_measurements/ | | | | | | | | | |
| Odebrat měření cviku | delete_exercise_measurement | | X | | | | | | | |
| Upravit měření cviku | edit_exercise_measurement | | X | | | | | | | |
| Přidat měření cviku | new_exercise_measurement | | X | | | | | | | |
| Zobrazit měření cviku | show_exercise_measurements | | X | | | | | | | |
| Realizace cviku | exercise_realizations/ | | | | | | | | | |
| Odebrat realizaci cviku | delete_exercise_realization | | | | | | | X | | |
| Upravit realizaci cviku | edit_exercise_realization | | | | | | | X | | |
| Zobrazit realizace cviku | list_exercise_realizations | | | | | | | | | X |
| Přidat realizaci cviku | new_exercise_realization | | | | | | | X | | |
| Nastavení realizace | exercise_realization_setups/ | | | | | | | | | |
| Odebrat nastavení realizace | delete_exercise_realization_setups | | | | | | | | X | |
| Upravit nastavení realizace | edit_exercise_realization_setup | | | | | | | | X | |
| Přidat nastavení realizace | new_exercise_serealization_setup | | | | | | | | X | |
| Nastavení cviku | exercise_setups/ | | | | | | | | | |
| Odebrat nastavení cviku | delete_exercise_setup | | X | | | | | | | |
| Upravit nastavení cviku | edit_exercise_setup | | X | | | | | | | |
| Přidat nastavení cviku | new_exercise_setup | | X | | | | | | | |
| Zobrazit nastavení cviku | show_exercise_setups | | X | | | | | | | |
| Série realizace cviku | exercise_set_realizations/ | | | | | | | | | |
| Odebrat sérii realizace | delete_exercise_set_realization | | | | | | | X | | |
| Upravit sérii realizace | edit_exercise_set_realizations | | | | | | | X | | |
| Spravovat nastavení r. | manage_setups | | | | | | | | X | |
| Přidat sérii realizace | new_exercise_set_realizations | | | | | | | X | | |
| Kroky cviku | exercise_steps/ | | | | | | | | | |
| Přidat kroky cviku | add_exercise_step | X | | | | | | | | |
| Odebrat kroky cviku | delete_exercise_step | X | | | | | | | | |

Tabulka 2.1: Pokrytí funkčních požadavků uživatelskými scénáři

Kapitola 3

Návrh aplikace

Tato kapitola popisuje návrh aplikace. Pro implementaci bylo zvoleno pokračovat v používání frameworku *Ruby on Rails* (podsekce 3.4.1). Důraz byl kladen na navázání na vzory, které zvolil autor původní práce.

3.1 Doménový model

Následuje popis všech entit v systému. Tento popis koresponduje s diagramem tříd (sekce 3.3). Všechny tyto entity byly v systému implementovány, s výjimkou *Exercise(Set)-RealizationMeasurement*. Důvody a stav jsou popsány v popisu entity (podsekce 3.1.11).

Oproti původnímu modelu [7] zde není entita *tagů*, kterou jsme se z nedostatku času a pro malou důležitost rozhodli odložit do další verze systému. Naopak entity *ExerciseStep*, *Plan*, *FavoritePlan* přibyly a ostatní použité entity byly ve většině případů zpřesněny, včetně vztahů a pomocných entit.

3.1.1 Exercise

Exercise je centrální entitou v systému - reprezentuje obecný cvik. Cviky mají povinně i atribut *version*, který společně s *code*, který udává unikátní zkratku použitelnou pro URL, jednoznačně identifikuje každý cvik. Verzování bylo zvoleno jako volitelná akce, pokud si chce *trenér* vytvořit personalizovanou verzi nějakého cviku nebo pokud je potřeba provést větší úpravy u již použitého cviku. Je tedy možné, aby v systému existovaly 2 cviky se stejným *code*, musí je však rozlišovat *version*.

Kromě navázaných entit popsanych níže lze specifikovat v attributech vedle názvu a popisu některé další vlastnosti, jejichž význam pro úplnost uvádím:

- **AuthorName** - jméno autora cviku (původní autor cviku se může lišit od uživatele, který jej přidal do systému, dále může v budoucnu sloužit jako specifikace pro účely umístění do globálního repozitáře)
- **Sources** - výčet zdrojů na literaturu či web, odkud je cvik čerpán

- `YoutubeUrl` - odkaz na video se cvikem, zde jsme zvolili službu YouTube, protože je zde dostupné velké množství videí tohoto typu a služba umožňuje videa i jednoduše nahrávat.

3.1.1.1 Accessibility

Atribut `Accessibility` určuje přístupnost cviku jednotlivým *trenérům*. Všechny varianty cviků jsou viditelné a spravovatelné *administrátorem* bez rozdílu.

Pokud je cvik `private`, je spojen s konkrétním *trenérem* a není přístupný jiným *trenérům*. To znamená, že *trenér*, který tento cvik nevytvořil, jej nemůže zobrazit ani nikde použít.

Cvik, který je `global` může vytvořit pouze *administrátor* a je viditelný všem *trenérům*. Takový cvik může být zobrazen i použit kterýmkoliv z *trenérů*.

V budoucnu mohou přibýt další varianty `Accessibility`, např. pokud by k systému přibyl internetový repozitář cviků. (podsekce 5.2.3) Systém je na takové rozšíření připraven, pro účely tohoto rozšíření však uvažují používání pouze uvedených typů.

3.1.1.2 ExerciseWithSets

`ExerciseWithSets` je potomek entity `Exercise`, reprezentuje *cvik se sériemi*, tedy cvik, který se dá dělit na určitý počet oddělených částí. Entita nemá žádné další atributy, je však klíčová pro identifikaci cviku tohoto typu. Jsou na ni napojeny podděně verze `ExerciseSetup` (podsekce 3.1.2.1) a `ExerciseMeasurement` (podsekce 3.1.3.1) a je tak poskytnuta větší flexibilita systému, neboť lze blíže specifikovat jednotlivé série, stejně jako celý cvik.

3.1.2 ExerciseSetup

`ExerciseSetup` specifikuje konfiguraci cviku. Základní cvik takto může být rozšířen o různá nastavení - příkladem může být typ použitého vybavení, povrch, teplotní podmínky, apod.

Entita obsahuje doplňkovou entitu `ExerciseSetupType`, která volitelně určuje některé základní (a později rozšířitelné) typy těchto konfigurací. Zvolení nějakého ze základních typů bude v budoucnu umožňovat např. připojení zásuvného modulu, který bude z nasbíraných dat vytvářet statistiku.

3.1.2.1 ExerciseSetSetup

`ExerciseSetSetup` je potomkem `ExerciseSetup`, váže se k verzi cviku `ExerciseWithSets`, tedy se jedná o konfiguraci popisující sérii cviku. Tato entita může popisovat např. různou váhu závaží pro konkrétní sérii cviku.

3.1.3 ExerciseMeasurement

`ExerciseMeasurement` zajišťuje měření nějaké veličiny týkající se cviku. Může to být např. tepová frekvence po ukončení cviku, úspěšnost, čas k provedení, apod.

Atribut `OptimalValue` umožňuje specifikovat, zda je „lepší“ vyšší nebo nižší naměřená hodnota.

3.1.3.1 ExerciseSetMeasurement

Obdobně jako u `ExerciseSetSetup` (podsekce 3.1.2.1) je tato entita rozšířením `ExerciseMeasurement` a váže se ke cviku se sériemi.

3.1.4 ExerciseStep

`ExerciseStep` specifikuje *kroky* (oddělené části) cviku. Cvik může mít libovolný počet kroků, jejich pořadí lze měnit. V případě např. cviku v posilovně to mohou být jednotlivé fáze provedení cviku.

3.1.5 ExerciseImage

`ExerciseImage` zastřešuje primárně obrázky pro ilustraci `ExerciseStep` *kroků*, je možné zadat obrázky pro správné a špatné provedení (atribut `correctness`). Dalším využitím je úvodní obrázek ke cviku.

3.1.6 ExerciseBundle

Původní entitu `ExerciseSet` jsem přejmenoval na `ExerciseBundle`, z důvodu anglické terminologie - *set* zde znamená *série cviku*, *bundle* lépe vyjadřuje podstatu entity (sekce 3.1.6).

`ExerciseBundle` reprezentuje sadu cviků. Pod „sada cviků“ rozumíme seskupení cviků s nějakou společnou charakteristikou. Může to být personalizovaná sada z pohledu trenéra nebo nějaký rys spojující cviky. Tato entita byla tvořena s myšlenkou možného budoucího rozšíření aplikace o *rezpozitář cviků*, kde by byla možnost importovat do lokální instance `ScApp` celé sady cviků naráz. (podsekce 5.2.3)

Podobně jako u `Exercise` je zde atribut `accessibility`, který udává přístupnost. V `ExerciseBundle`, který má `accessibility` nastaveno na `global`, mohou být zařazeny `Exercise`, které jsou `global`. Takovou sadu může vytvořit pouze administrátor a slouží zejména pro často používané skupiny cviků. V `ExerciseBundle`, který má `accessibility` nastaveno na `private`, mohou být zařazeny `Exercise`, které jsou `global` nebo `private`. Taková sada je určena primárně pro vytvoření *trenérem*.

3.1.7 Unit

`Unit` představuje jednotku pro `ExerciseSetup` nebo `ExerciseMeasurement`. Jako jeden z atributů je zde `UnitType`, tedy typ jednotky, který je realizován jako *výčet*. Tento atribut určuje, jakého typu hodnot může veličina nabývat. V systému jsou implicitně hodnoty `integer`, `decimal`, `string` a `time`.

3.1.8 Plan

Plan slouží pro identifikaci skupiny *realizací cviků* v rámci jedné *treninkové hodiny* (entita `TrainingLessonRealization`, která je specifikována v původní práci) pro jednoho *hráče*. Zároveň specifikuje *uživatele*, který je *autorem plánu*.

Entita `FavoritePlan` umožňuje *trenérovi* uložit si s komentářem některý z *plánů*, které vytvořil, pro znovupoužití na jiné *hodině*. Tato funkcionalita je v současné době základní a *systém* je připraven pro případné budoucí rozšíření, které poskytne další výrazné zjednodušení práce s *plány* - obzvláště při častém vyplňování *treninků* s podobným obsahem by byla tato rozšířená funkcionalita výhodou.

3.1.9 ExerciseRealization

`ExerciseRealization` je vedle `Exercise` další hlavní entitou a reprezentuje realizaci *cviku* v rámci *plánu*. *Realizace* jsou v plánu umístěny v daném pořadí, které lze měnit při editaci plánu, dokud hodina nebyla *uzavřena*. Atribut `TimeDuration` udává *délku trvání realizace*, `RestAfterExercise` *odpočinek* po dokončení cvičení. Je možné specifikovat *poznámku*, jakým způsobem má cvik být vykonán. Atribut `Completed` aktuálně není využit, je zde pro indikaci, zda se *hráči* podařilo *cvik* vykonat.

3.1.9.1 ExerciseSetRealization

`ExerciseSetRealization` je ekvivalent `ExerciseRealization` pro specifikaci jedné série realizace `Exercise`. Entita má stejné atributy, vše se však váže k „*sériovým*“ verzím entit, tj. entit, které mají v názvu `set` a specifikují *sérii cviku*, *ne cvik samotný*.

3.1.10 ExerciseRealizationSetup

Tato entita zajišťuje připojení nastavení `ExerciseSetup` k realizaci `ExerciseRealization`. V závislosti na *jednotce Unit* je možno specifikovat *numerickou* či *textovou hodnotu* a *poznámku*.

3.1.10.1 ExerciseSetRealizationSetup

`ExerciseSetRealizationSetup` je ekvivalentní entitou jako `ExerciseRealizationSetup` pro sérii cviku `ExerciseSetRealization`.

3.1.11 ExerciseRealizationMeasurement

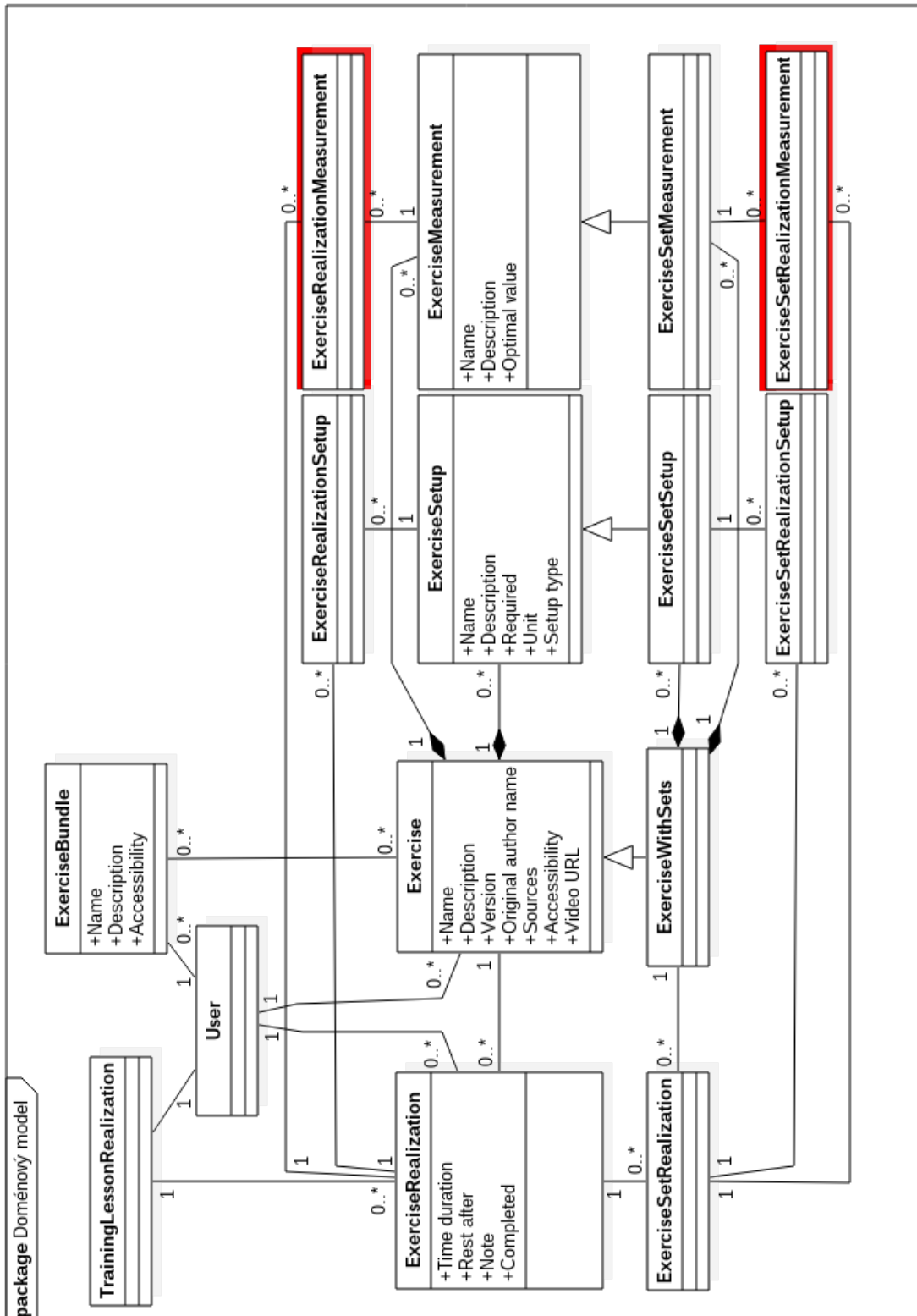
Tato entita zajišťuje připojení měření `ExerciseMeasurement` k realizaci `ExerciseRealization`. Z důvodu nedostatku času jsem nestihl dokončit vývoj rozhraní spojených s touto entitou, v systému je však připraven základ pro toto rozšíření, které bude realizováno v budoucnu. Návrh systému s touto entitou počítá, bude se jednat o další užitečný celek aplikace.

3.1.11.1 ExerciseSetRealizationMeasurement

ExerciseSetRealizationMeasurement bude ekvivalentní entitou jako Exercise-RealizationMeasurement pro sérii cviku ExerciseSetRealization.

3.2 Diagram doménového modelu

Pro zvýšenou přehlednost uvádím diagram doménového modelu, ve kterém figurují pouze hlavní entity aplikace. Entity zvýrazněné červenou barvou nebyly implementovány z důvodu nedostatku času. Podrobnější diagramy se všemi entitami se nacházejí v sekci 3.3.

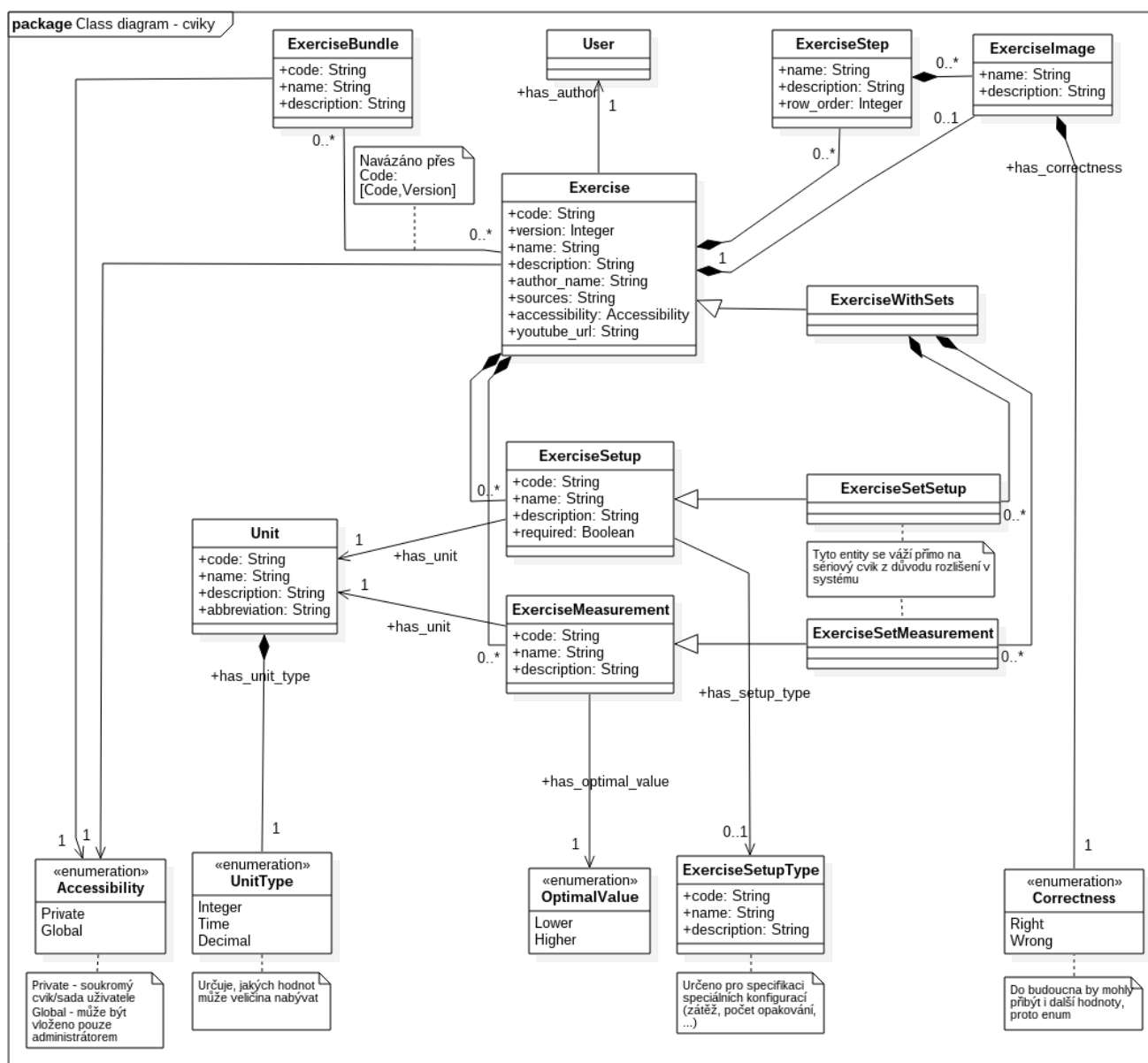


Obrázek 3.1: Doménový model

3.3 Podrobné diagramy modelu aplikace

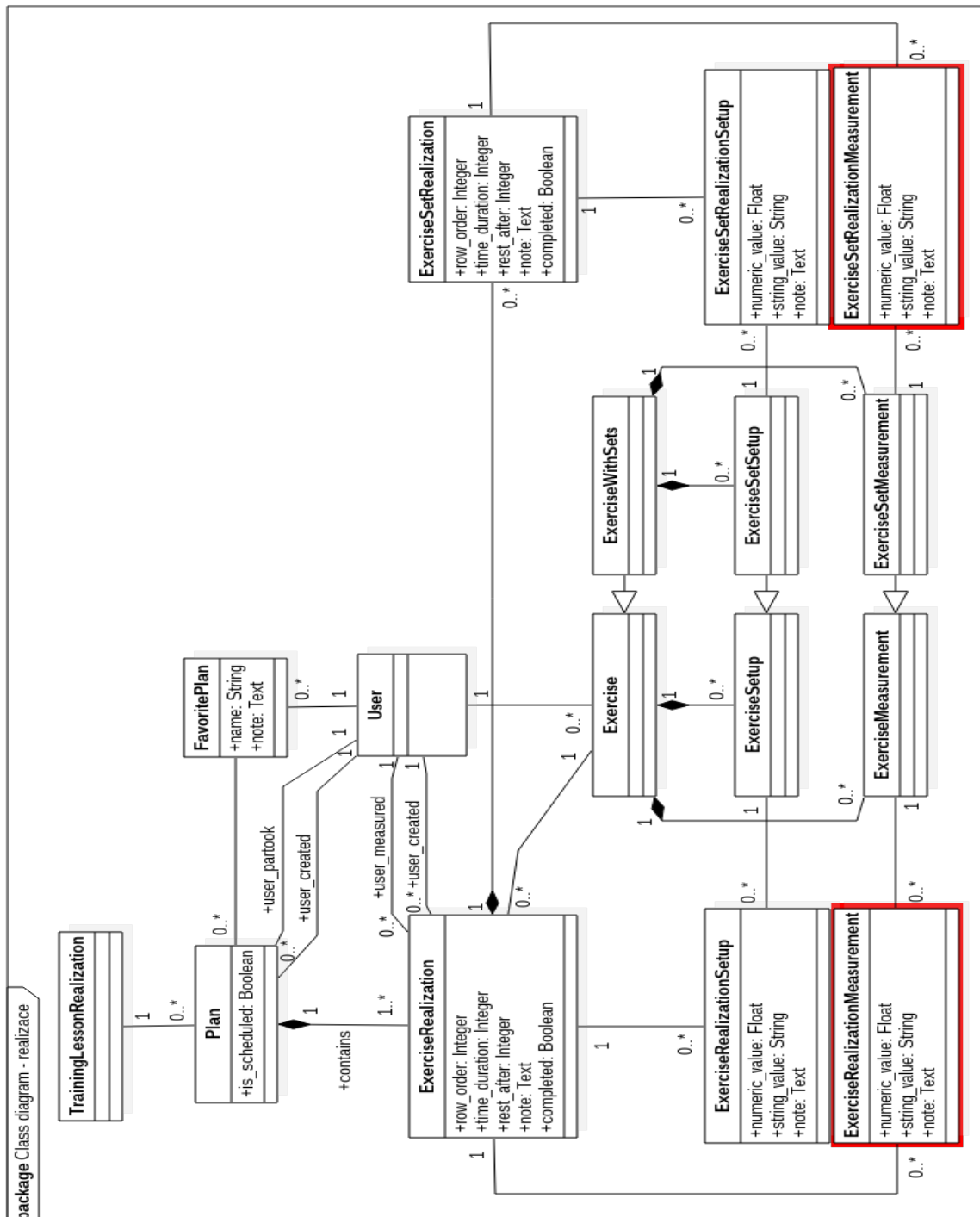
Následují diagramy tříd - oba diagramy popisují dohromady aplikaci, z důvodu přehlednosti jsem diagramy rozdělil - první diagram znázorňuje část aplikace pro správu cviků. Druhý diagram znázorňuje část aplikace pro správu plánů.

Obrázek 3.2 znázorňuje třídy aplikace spojené se specifikací cviků.



Obrázek 3.2: Diagram tříd - část pro správu cviků

Obrázek 3.3 znázorňuje třídy aplikace spojené se specifikací tréninkových plánů. Entity zvýrazněné červenou barvou nebyly implementovány z důvodu nedostatku času (podrobněji popsáno v popisu příslušné entity).



Obrázek 3.3: Diagram tříd - část pro správu plánů

3.4 Implementace

3.4.1 Framework Ruby on Rails

Původní aplikace je napsána ve webovém *frameworku* Ruby on Rails [4]. Použití tohoto nástroje bylo dalším podnětem, který napomohl mému zvolení tohoto tématu, chtěl jsem se s *frameworkem* důkladně seznámit. Jedná se o v poslední době obecně velmi oblíbený a propracovaný nástroj s aktivní komunitou a kvalitní dokumentací.

Použití Ruby on Rails poskytuje řadu výhod. Jako hlavní důsledek koncepce, kterou autoři *frameworku* zvolili, vnímám na mnoha úrovních velmi dobře zvládnutou redukci „zbytečné“ práce, což vývojáři poskytuje možnost soustředit se hlavně na to důležité a zároveň tak *framework* poskytuje jednotnou formu realizace často řešených problémů.

Často využívanou vlastností Ruby on Rails je možnost rozšíření pomocí tzv. *gemů* [9], což je ekvivalent knihoven v ostatních jazycích. Jsou to balíčky dostupné z repozitáře, často aktualizované a udržované komunitou. *Gemy* jsou jednoduše připojitelné k aplikaci specifikací v souboru *Gemfile* a následným spuštěním nástroje, který se stará o jejich instalaci a aktualizaci. Velmi kvalitních a spolehlivě použitelných *gemů* je již k dispozici opravdu velké množství a tedy je dobře pokryta většina často se objevujících podobných problémů. Pokud tedy vývojář chce nějaký problém řešit za pomoci hotové knihovny, je zde opravdu velký výběr.

3.4.1.1 Model-View-Controller

Model-View-Controller (dále MVC) je návrhový vzor široce využívaný v rozmanitých aplikacích [19]. Odděluje v aplikaci *datový model*, *uživatelské rozhraní* a *řídící logiku*. Vznikají tak separátní komponenty, což výrazně zvyšuje přehlednost kódu a míru nezávislosti jednotlivých komponent navzájem. U Ruby on Rails jsou tyto vlastnosti dobře dosažitelné, pokud se vývojář drží konvencí, které se *framework* snaží prosazovat.

3.4.1.2 Převod na Rails 4.1

Z původní verze *frameworku* Rails 4.0 jsem během implementace přešel na verzi 4.1.7, což vyžadovalo aktualizovat i některé další závislé *gemy* a opravit části aplikace, které přestaly fungovat. Toto dobře odhalily kvalitní *akceptační testy*.

Hlavní důvod přechodu na novou verzi bylo rozhodnutí o použití *složeného primárního klíče*, který je použit u entity *Exercise*, kde určuje instanci pomocí atributů *Code* a *Version*. Dobře použitelná verze *gemu* obstarávajícího tuto funkcionalitu vyžadovala novější verzi a upgrade poskytoval i některé další drobné benefity.

3.4.2 Knihovna JQuery

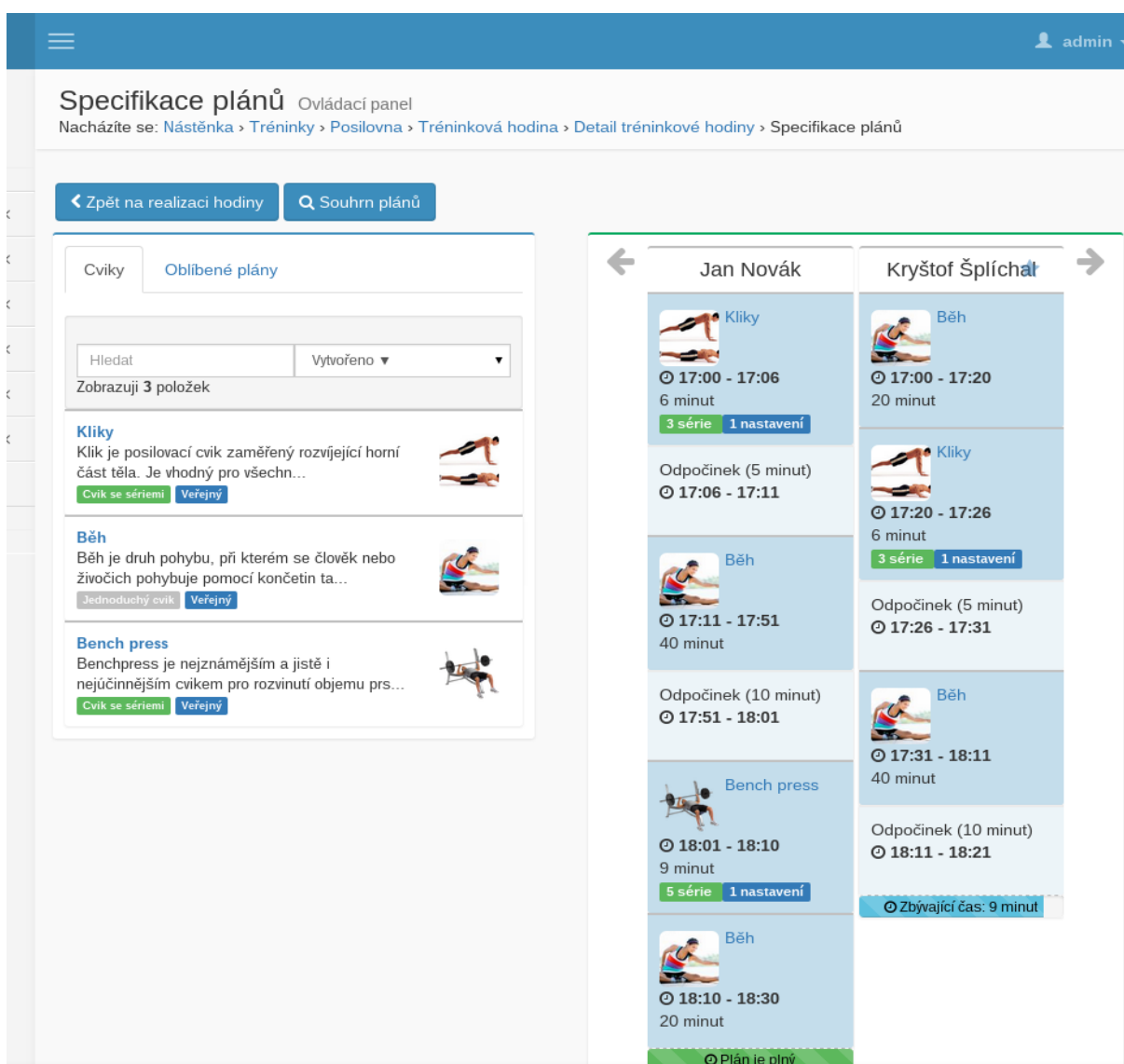
Zejména v části, kde se specifikují realizace cviků, je hojně využito knihoven *jQuery* a *jQuery-UI* [22, 23], které jsou šířeny pod licencí MIT [18]. Tyto knihovny zprostředkovávají zjednodušené a standardizované rozhraní pro *skriptování* na klientské straně.

Zároveň knihovny poskytly možnost vytvořit uživatelské rozhraní tak, aby byla odezva uživateli pokudmožno okamžitá (technologie AJAX – Asynchronous JavaScript and XML [15]) a rozhraní působilo moderním dojmem.

3.5 Ukázky výsledné implementace

Implemetoval jsem všechny uživatelské scénáře definované v sekci 2.5, zde pro názornost uvedu některé z nejčastěji používaných obrazovek aplikace, které ilustrují hlavní uživatelské scénáře.

Obrazovka na obrázku 3.4 zprostředkovává možnost *editace cvičebních plánů* (odpovídá UC3 v sekci 2.5) pro hráče, cvik se přidá do plánu *přetažením* z nabídky vlevo.



Obrázek 3.4: Ukázka obrazovky pro editaci cvičebních plánů

Obrazovka na obrázku 3.5 zprostředkovává rozhraní pro *zobrazení detailu cviku*, případně jeho *editaci* a *editaci nastavení a měření*.

Rozložení sekcí se mění v závislosti na šířce zobrazujícího zařízení, je tak dosažena vizuální kompatibilita s většinou běžných mobilních zařízení.

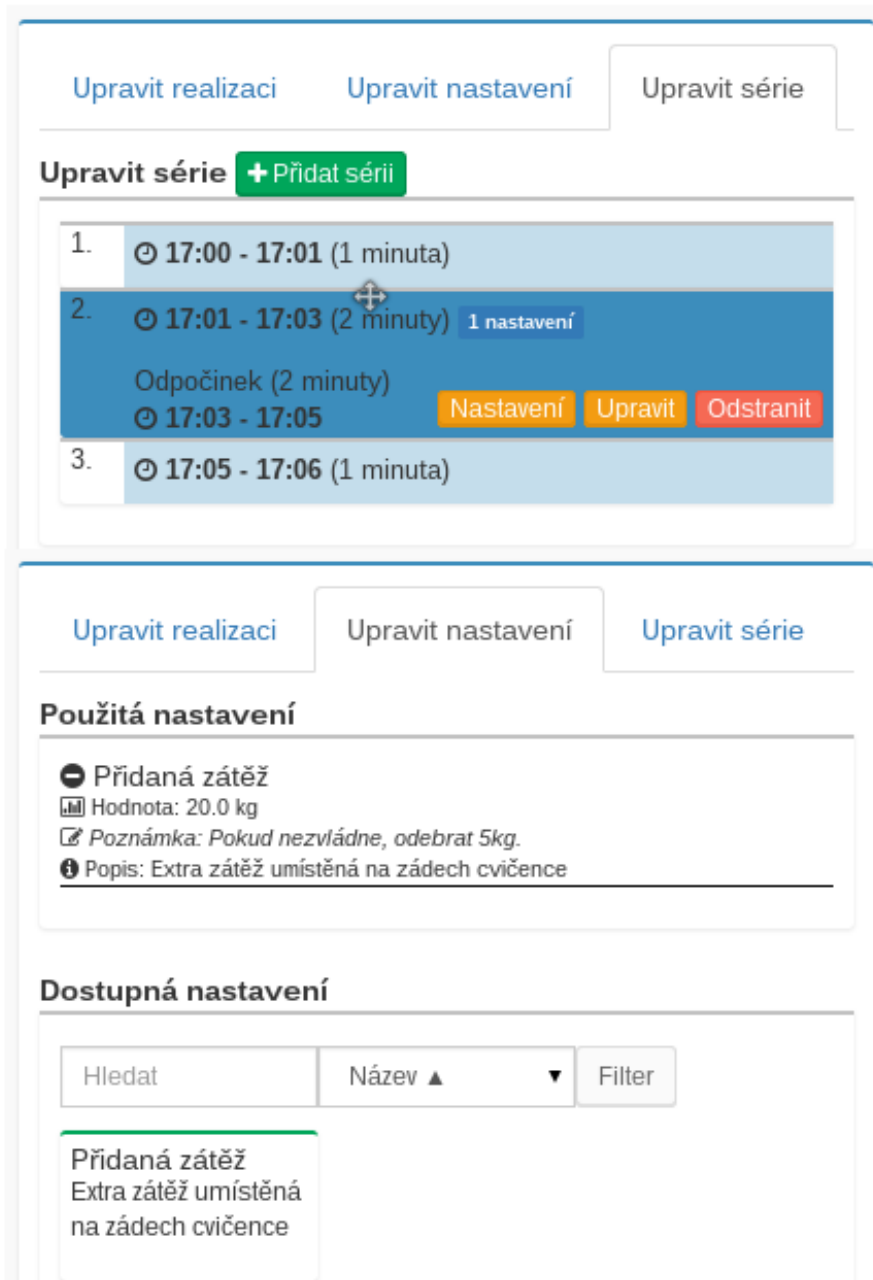
V nastaveních a měřeních lze procházet dynamicky hledáním nebo stránkováním.

The screenshot displays the 'Bench press - detail cviku' page in the Scapp v2 application. The interface is organized into several key sections:

- Header:** Shows the user 'admin' and the page title 'Bench press - detail cviku' with a breadcrumb trail: 'Nacházíte se: Nástěnka > Cviky > Bench press - Detail cviku'.
- Navigation:** A sidebar on the left lists menu items like 'Uživatel', 'Metriky', 'Tréninky', 'Cviky', 'Sady cviků', 'Finance', and 'Nápověda'.
- Detail Section:** Contains metadata for the exercise:
 - Typ:** Cvik se sériemi
 - Vlastník:** admin
 - Přístupnost:** Veřejný
 - V sadách:** Cviky pro sestavy, Nastavení detail
 An image of a person performing a bench press is shown. A description states: 'Benchpress je nejznámějším a jistě i neúčinnějším cvikem pro rozvinutí objemu prsních svalů.'
- Nastavení a Měření Section:** Features two tabs: 'Nastavení cviku' and 'Měření cviku'. A '+ Přidat nastavení cviku' button is present. Below is a search filter with a 'Hledat' input, a 'Název' dropdown, and a 'Vše' dropdown. It indicates 'Zobrazují 3 položek'.
 - Minimální frekvence dechu:** Použito (green), Vyžadováno (red), Jednotka: Sekunda [s], Typ: Nastavení pro celý cvik.
 - Váha:** Použito (grey), Vyžadováno (red), Jednotka: Kilogram [kg], Typ: Nastavení pro sérii cviku.
 - Výška lavice:** Použito (grey), Vyžadováno (red), Jednotka: Jiné (číslo) [], Typ: Nastavení pro sérii cviku.
- Kroky cviku Section:** Titled '(Zobrazit detailní pohled)', it lists five steps:
 - 1 Odejmутí činky ze stojanu:** Po odebrání činky ze stojanu začíná cvik s činkou v napnutých pažích, následuje kontrolovaný pokles, dotek osy na hrudníku a plynulé vytažení zpět do napnutých paží.
 - 2 Stabilní pozice těla:** Zaujměte stabilní polohu těla na lavičce: sedací svaly, vrchní část zad a hlava jsou v průběhu vykonávání cviku v kontaktu s lavičkou.
 - 3 Úchop činky:** Z důvodu bezpečnosti použijte **vždy** palcový úchop, tzn. poloha palce je proti ostatním prstům, nikoliv zároveň s nimi.
 - 4 Provedení zdvihu krát počet opakování**
 - 5 Návrat činky do stojanu**

Obrázek 3.5: Ukázka obrazovky pro detail a editaci cviku

Obrazovky na obrázku 3.6, které se zobrazují v závislosti na přepnutí tabů (ty lze nalézt nad obrazovkami), zprostředkovávají *nastavení sérií cviku* a *editaci nastavení* pro celý cvik. Série lze přesouvat a měnit tak jejich pořadí, aktuálně aktivní série je zvýrazněna tmavší barvou a zobrazí se příslušný kurzor. Scénář řazení sérií v realizaci je popsán v UC4 v sekci 2.5.




Obrázek 3.6: Ukázka oken pro editaci nastavení a sérií realizace cviku

Obrazovka na obrázku 3.7 slouží pro *editaci kroků* cviku. Kroky lze přesouvat obdobně jako série (předchozí obrazovka). Dolní část obrazovky se podle potřeby mění na formulář pro přidání nebo úpravu kroku.

Je možné přidat obrázky „správně“, „špatně“, pro lepší názornost provedení daného kroku.

[← Zpět na cvik](#)

Seznam kroků

| Název | Popis | Obrázky - správně | Obrázky - špatně |
|---|--|--|--|
| ✚ Odejmutí činky ze stojanu | Po odebrání činky ze stojanu začíná cvik s činkou v napnutých pažích, následuje kontrolovaný pokles, dotek osy na hrudníku a plynulé vytlačení zpět do napnutých paží. |  | Upravit Odstranit |
| ✚ Stabilní pozice těla | Zaujměte stabilní polohu těla na lavičce: sedací svaly, vrchní část zad a hlava jsou v průběhu vykonávání cviku v kontaktu s lavičkou. | | Upravit Odstranit |
| ✚ Úchop činky | Z důvodu bezpečnosti používejte vždy palcový úchop, tzn. poloha palce je proti ostatním prstům, nikoliv zároveň s nimi. | | Upravit Odstranit |
| ✚ Provedení zdvihu krát počet opakování | | | Upravit Odstranit |
| ✚ Návrat činky do stojanu | | | Upravit Odstranit |

Upravit krok

*** Jméno**

Popis

A Normal text ▾
Bold
Italic
Underline

↶
↷
↻
🖼️


Po odebrání činky ze stojanu začíná cvik s činkou v napnutých pažích, následuje kontrolovaný pokles, dotek osy na hrudníku a plynulé vytlačení zpět do napnutých paží.

Obrázky - správně

Nahrát obrázky (výběr více)

Soubor nevybrán

Současné obrázky



✖ Odstranit obrázek

Obrázek 3.7: Ukázka obrazovky pro editaci kroků cviku

Kapitola 4

Testování

Aplikace je otestována ze dvou pohledů - *akceptační testy* kontrolují, že aplikace splňuje funkcionalitu požadovanou ve funkčních požadavcích (podsektce 2.4.1) a *testování s uživatelem* poskytuje cenný pohled uživatele systému, tedy zpětnou vazbu, kterou lze dobře využít pro optimalizaci uživatelského rozhraní a některých procesů. [13]

4.1 Behavior Driven Development a Ruby on Rails

Pro vývoj aplikace byl zvolen (v návaznosti na původní aplikaci a dobře osvědčený způsob vývoje) Behavior Driven Development (zkráceně BDD) [16], který lze v Rails dobře realizovat gemem Cucumber [5].

Princip této metody při realizaci v Rails s Cucumber je následující. Nejprve se stanoví požadavky uživatelů (zde tzv. *user stories*) ve formě popisu, jak se systém má chovat. Tyto *user stories* jsou napsány v jazyce Gherkin, který předem danými výrazy spojuje anglické věty. To zaručuje srozumitelnost i pro zákazníka a celkovou dobrou přehlednost. Gem Cucumber umožňuje aplikaci testovat následujícím způsobem. Pro *user stories* se napíše tzv. *step definitions* - jednotlivé kroky, které vykonávají to, co uživatel popsal. Při implementaci všech *step definitions* pro daný test tedy vznikne možnost tento test spustit. Nejprve se test pro úplnost nechá selhat a postupně se implementuje funkcionalita tak, aby test prošel. Jakmile tedy test prochází, víme, že jsme implementovali tu funkcionalitu, kterou zákazník chtěl přednostně, a zároveň je tato funkcionalita dobře pokryta akceptačními testy, které jsou přímo napojeny na požadavky. [5]

Vzhledem k tomu, že náplní práce je rozšířit systém, bylo potřeba mimo tvoření nových analyzovat a zužítkovat některé již použité *step definitions*, které je žádoucí z programátorské praxe znovupoužívat.

4.1.1 Ukázka step definitions pro Exercise

```
1 And(/^Following exercise steps exist in the system$/) do |table|
2   # table is a table.hashes.keys # => [:name, :description, :
      step_number, :exercise]
3   table.hashes.each do |r|
```

```

4     exercise = Exercise.friendly.find([r[:exercise],1])
5     ExerciseStep.create({name: r[:name], description: r[:description]
6       ], row_order: r[:step_number], exercise: exercise})
7   end
8 end
9 And(/^Radio button "([\^"]*)" should be selected$/) do |arg|
10   page.should have_checked_field(arg)
11 end

```

Výpis kódu 4.1: Ukázka step definitions pro Exercise v jazyce Ruby

Tato ukázka ilustruje, jak vypadají step definitions pro jednotlivé kroky scénáře. První scénář obstarává vytvoření ExerciseSteps zadaných v tabulce. Druhý scénář kontroluje, zda je ve stránce přítomen označený Radio Button se zadným popiskem.

4.1.2 JQuery v testování

Z důvodu volby interaktivního rozhraní realizovaného pomocí knihovny jQuery (především v kombinaci s jQuery-UI (User Interface) [23]) bylo potřeba přizpůsobit i testování - podpora v použitých testovacích nástrojích je podle zkušenosti často neúplná, proto je vhodné využít speciální doplňkové knihovny nebo alternativní postupy vedoucí k příslušnému efektu.

Pro testování jQuery-UI widgetu Sortable (použito jako stěžejní část obrazovky specifikace plánů, umožňuje mimo další využití řadit jednotlivé realizace v rámci plánu) jsem využil externí knihovny [14], která umožňuje simulovat toto řazení jednoduchým příkazem. Ukázka začlenění do testu je v následujícím výpisu kódu.

```

1 When(/^I move "([\^"]*)" realization one step down$/) do |arg|
2   page.execute_script %{
3     $('realization:contains("#{arg}")').simulateDragSortable({move:
4       1});
5   }
6 end

```

Výpis kódu 4.2: Ukázka testování jQuery Sortable

Použitá knihovna však neumožňuje testování funkcionality typu drag and drop (přetažení objektu do seznamu) v jQuery-UI widgetu Draggable, kterou jsem využil pro přidání cviku do plánu. Pro tyto účely jsem sestavil jednoduché řešení, kde se jako cíl přetažení vybere první prvek seznamu (seznam nesmí být prázdný). Samotné přetažení již testovací nástroj zvládá dobře.

```

1 When(/^I drag and drop exercise "(.*)" to plan for user "(.*)"$/) do
2   |exercise_name, user_slug|
3   user_id = User.friendly.find(user_slug).id
4   plan = page.find_by_id('player-' + user_id.to_s)
5   target = plan.first('realization')
6   dragged_item = page.first('li[data-exercise-code="' + exercise_name.
7     to_s + '"]')

```

```
6 dragged_item.drag_to(target)
7 end
```

Výpis kódu 4.3: Ukázka testování jQuery Draggable

4.2 Akceptační testy

Díky volbě metody Behavior Driven Development (sekce 4.1) pro vývoj systému lze mít pokrytou funkcionalitu testy, které skutečně ověřují, zda systém splňuje kladené požadavky. [16, 5]

Testy se nachází ve složce `features`, kde jsou tříděné ve složkách podle názvů entit. Pro viditelné oddělení od původního systému a přehlednost jsem jako kořenovou složku všech nových testů zvolil `features/exercises`, zde pak můžeme nalézt další testy entit navázaných na `Exercise`.

Celkem jsem vytvořil v rámci akceptačních testů **82 nových scénářů s 1402 kroky**, všechny tyto testy procházejí.

4.3 Testování s uživateli

Zpětná vazba uživatelů je pro úspěšné používání a vývoj aplikace klíčová, proto jsem do testování zařadil i testy s uživateli.

I přes moji snahu o vytvoření uživatelsky přívětivého a dobře fungujícího rozhraní mohou být některé operace nelogické, zbytečně složité nebo naopak neúplné. Velkou část těchto nedostatků by mělo testování s uživateli odhalit a pomoci odstranit.

4.3.1 Výběr účastníků testu

Pro testování bylo vybráno celkem 6 participantů. Tři z participantů se účastnili testování metodou *kognitivního průchodu* (podsekce 4.3.3.1), druhá polovina participantů se účastnila testování *heuristickou evaluací* (podsekce 4.3.3.2). Výběr metod a rozdělení jsem provedl z důvodu, že obě tyto metody přináší jiný typ výsledků.

Hlavní skupinou uživatelů rozšíření vytvořeného v této práci bude skupina *trenérů*, volil jsem tedy participanty, kteří mají zkušenost se správou tréninků s požadavkem alespoň půl roku praxe - pracovníky fit centra a studenty Fakulty tělesné výchovy a sportu Univerzity Karlovy. Věk ani pohlaví nebyly pro testování relevantní. Nikdo z participantů doposud nepoužíval specializovaný software tohoto typu, pro záznamy používali ruční zápis na papír nebo tabulkový procesor.

4.3.2 Průběh testování

Pro testování jsem použil dvě často využívané metody - *heuristická evaluace* a *kognitivní průchod*. Tyto metody jsou podrobněji popsány v sekci 4.3.3.

Aplikace byla otestována na slovně popsáných scénářích (sekce 2.5). Tyto scénáře byly vybrány, protože ilustrují často používané a nejvíce důležité akce v systému, je tedy prioritní je otestovat s uživateli, kteří mohou podobnou funkcionalitu skutečně využívat.

V případě obou metod testování probíhá obdobně. Participant je seznámen s aplikací a obsahem toho, co se bude testovat. Přítomen u testu je také moderátor, který participanta testem provází - nikoli však navádí či přehnaně zasahuje do průběhu, to by mohlo ovlivnit výsledky testování. Participantovi je důrazně vysvětleno, že jakékoli nejasnosti a vzniklé problémy v aplikaci nejsou jeho chybou, ale naopak chybou aplikace a že zejména z toho důvodu testování probíhá. Dále je participant vyzván, aby veškeré své postupy a myšlenky nahlas komunikoval z důvodu, aby moderátor mohl zaznamenat případné nálezy. [21]

Záznam a klasifikace výsledků je specializovaná činnost, kterou by měl vykonávat tzv. *tester*, pro uživatele bez zkušenosti s testováním software by taková činnost byla příliš složitá, naopak zde participant slouží jako cenný zdroj přímé zkušenosti s aplikací bez ovlivnění znalostí vývoje software. Já jsem v průběhu studia absolvoval kurz Testování uživatelského rozhraní[21], kde jsem se důkladně seznámil s různými metodami testování, včetně použitých metod, proto jsem se zhostil této role.

4.3.3 Metody pro testování

Ve výsledcích testování se odkazují na konkrétní části použitých metod (zejména konkrétní heuristiky), proto je zde pro přehlednost ve stručnosti uvedu.

4.3.3.1 Metoda testování kognitivním průchodem

Při testování kognitivním průchodem se s participantem postupně prochází daný případ užití, který je specifikován přesně danými kroky. Průchod těmito kroky se zaznamená a vyhodnotí se odpovědi na následující otázky. [10]

- Q0** Čeho chce uživatel dosáhnout?
- Q1** Bude uživateli zřejmé, co má nyní udělat?
- Q2** Spojí si uživatel popis se svým cílem?
- Q3** Dostane uživatel dostatečnou a srozumitelnou reakci na svoji akci?

Pokud je odpověď na nějakou z otázek „Ne“, přiloží se k záznamu popis problému sepsaný do stručného komentáře.

4.3.3.2 Metoda testování heuristickou evaluací

Při testování heuristickou evaluací se vyhledávají chyby nebo nedostatky v návrhu vzhledu, jejichž odstranění vede k lepší použitelnosti a průchodnosti systému a předejití zbytečných chyb při používání, které by mohl návrh produkovat. Nálezy se přiřazují některé z následujících heuristik. [12, 11]

1. **Viditelnost stavu systému** – systém by měl vždy dát uživateli vědět co se právě odehrává
2. **Spojení mezi systémem a reálným světem** – komunikace systému s uživatelem by se měla odehrávat uživatelsky příjemným způsobem (srozumitelný jazyk bez odborných termínů).
3. **Uživatelská kontrola a svoboda** – uživatelé při práci se systémem dělají chyby a potřebují proto únikový východ pro návrat do předchozího stavu.
4. **Konzistence a standardizace** – uživatelé by neměli být nuceni přemýšlet jestli různé termíny znamenají to stejné, proto se doporučuje dodržovat obecné zásady.
5. **Prevence chyb** – vyvarovat se chybovým hlášením bezpečným designem, který bude preventivně působit proti problémům.
6. **Rozpoznání místo vzpomínání** – uživatel by neměl být nucen vzpomínat si na provádění operací v systému, instrukce by měly být v systému vždy viditelně umístěny.
7. **Flexibilní a efektivní použití** – umožnění zrychlení práce se systémem pro pokročilé uživatele.
8. **Estetický a minimalistický design** – bez nepotřebných informací a prvků.
9. **Pomoc uživatelů poznat, pochopit a vzpamatovat se z chyb** – chybové hlášky by měly být uváděny v přirozeném jazyce a měly by navrhnout řešení.
10. **Nápověda a návody** – veškeré informace se musí dát lehce vyhledat, nápověda by měla obsahovat postupy v krocích.

4.3.4 Výsledky testování s uživateli

Následuje přehledný výčet nálezů, které vyplynuly z testování s uživateli. Výčet je rozdělen v tabulkách podle použitých metod a seřazen podle priority nálezů. Podrobné průběhy testování jsou v příloze B.

Pro klasifikaci závažnosti nálezů používám tři priority, na jejichž základě lze přistoupit k příslušným opravám v adekvátním pořadí:

1. **Priorita 1** - Závažný nedostatek nebo chyba, která zapříčiňuje obtížné používání systému.
2. **Priorita 2** - Nedostatek nebo chyba, která je pro uživatele nepříjemná, ale ne závažná.
3. **Priorita 3** - Nedostatek nebo chyba, která je spíše kosmetickou chybou, nezávažné.

U nálezů uvádím i navržené řešení. Většinu nalezených problémů nelze kvůli nedostatku času opravit v rámci této práce, tato forma výpisu však (spolu s uvedením hlavních problémů v rámci tzv. `issues` na `GitHub` repozitáři) umožní současným i budoucím vývojářům problémy spolehlivě adresovat.

4.3.4.1 Testování metodou kognitivního průchodu

Testování kognitivním průchodem probíhalo způsobem popsaným v sekci 4.3.3.1. Výsledky jsem zaznamenával do tabulkového editoru (přesný průchod je v příloze B) a po seřazení jsem přidal návrh řešení pro každý nalezený problém.

| Nález | UC | Priorita | Popis | Návrh řešení |
|-------|-----|----------|---|--|
| N2 | UC1 | 2 | Není zřejmý rozdíl mezi poli „Popis“ a „Podrobný popis“ - jak se projeví která varianta. | Přidat k popisku pole formuláře vysvětlení. |
| N3 | UC1 | 2 | Při kliknutí na „Vytvořit cvik bez přidání detailů“ není zobrazena potvrzující hláška, že cvik byl vytvořen. | Přidat potvrzující hlášku. |
| N6 | UC3 | 2 | Bez předchozí detailní instruktaže není zřejmé, jak se plány vyplňují cviky. Po instruktaži nebyly žádné další problémy. | Ideální je předchozí instruktaž uživatele. Možností je přidat do oblasti realizace ikonu kurzoru pro přesun. |
| N7 | UC4 | 2 | Uživatel si nemusí všimnout přeměny levého panelu na panel pro úpravu realizace. Přehlednější by bylo přidat ještě výrazný nadpis. | Přidat k jednotlivým panelům nadpis - aktuálně pouze v tabech, což může být nedostatečně výrazné. |
| N8 | UC4 | 2 | Série vypadají podobně, když nemají specifikovány žádné další vlastnosti - přehlednost systému by zvýšilo přidání potvrzujícího znamení, že série byla přesunuta. | Přidat do systému potvrzovací hlášku, která však nebude rušit uživatele. |
| N1 | UC1 | 3 | Systém zobrazuje popis dvakrát - jednou v levém menu a podruhé nahoře na příslušné obrazovce, to může být matoucí. | Přítomnost obou tlačítek je žádoucí, odlišil jsem však jejich popisky. |
| N4 | UC2 | 3 | Tlačítko pro správu cviků v rámci sady si lze splést s tlačítkem pro upravení sady. | Odlišit barevně tlačítka / změnit ikonu jednoho z nich. |
| N5 | UC2 | 3 | Nadpis „Cviky obsažené v sadě“ nesouhlasí s tím, že v tabulce jsou zobrazeny všechny cviky, byť barevně odlišeny. | Změnit nadpis nebo rozdělit tabulku tak, aby byl jasný rozdíl. |

Tabulka 4.2: Seznam nálezů pro metodu kognitivního průchodu

4.3.4.2 Testování metodou heuristické evaluace

Testování heuristickou evaluací probíhalo způsobem popsaným v sekci 4.3.3.2. Průběh testování jsem nahrával a poté řečené problémy ohodnotil příslušnými heuristikami (konstatováno v příloze B, zde uvádím pouze seznam nálezů). Opět jsem vždy uvedl možné řešení problému.

| Nález | UC | Priorita | Popis problému | Návrh řešení |
|-------|-----|----------|---|---|
| N9 | UC1 | 1 | Filtr funguje na principu aktualizace psaním, je však snadné přehlédnout, že v poli se již něco nachází a poté se zobrazují pouze filtrovaná cvičení. | Zvýraznit pole filtru, když obsahuje nějaký text. |
| N15 | UC4 | 1 | Nelze zavřít editační obrazovku jinde, než v tabu Upravit realizaci. | Přidat tlačítko pro zavření do oblasti editační obrazovky na viditelné místo. |
| N10 | UC1 | 2 | Není jasné, jaký formát mají mít pole Zdroje a Video. | Interaktivně kontrolovat formát nebo uvést vzor pro vyplnění u pole. |
| N12 | UC2 | 2 | V případě velkého množství sad bude obtížné sady procházet, dobrá by byla možnost sady řadit podle různých kritérií. | Přidat funkcionalitu filtrování, obdobně jako je u seznamu cviků. |
| N13 | UC2 | 2 | Stejný problém jako N12, (zobrazení tabulky s dostupnými cviky) | Přidat funkcionalitu filtrování, obdobně jako je u seznamu cviků. |
| N14 | UC4 | 2 | Není nikde viditelné, jaká realizace je aktuálně editována. | Zvýraznit v plánu aktuálně editovanou realizaci. |
| N11 | UC2 | 3 | V případě absence sad není jasně vidět, zda systém žádné sady neobsahuje nebo zda se stala chyba v programu. | Přidat informaci o neexistenci sad. |

Tabulka 4.4: Seznam nálezů pro metodu heuristické evaluace

4.3.4.3 Ostatní postřehy uživatelů vzniklé při testování

Po výstupech z testovacích metod uvádím ještě další postřehy participantů, které z testování vyplynuly. Některé body poskytnou věcné podněty pro další rozvoj aplikace.

- Začátky používání systému mohou odradit uživatele kvůli *nutnosti vyplnit velké množství dat*. Příkladem může být vyplnění cviků v posilovně. Tyto cviky se pro většinu posiloven budou opakovat (budou mít podobný obsah), proto by mělo smysl vytvořit *globální repozitář* (navrženo již v původní práci) nebo *instalační skript*, který by obsahoval často používané cviky jako volitelnou základní sadu, která by měla možnost jednoduché úpravy a tedy *rychlého startu*.
- V systému se *opakují oblasti*, kde je uživatel *nucen opakovaně zadávat podobná data*. Je to např. vyplňování uživatelských e-mailů po jednom, specifikace cviků s podobným obsahem realizace, specifikace celých podobných plánů. U plánů je tento problém částečně řešen možností uložit si plán jako *oblíbený*, ideální by však byla možnost *klonovat realizace i celé plány*. S touto funkcionalitou se počítá v budoucím vývoji aplikace a přidal jsem záznam v podobě *issue* na *GitHub* repozitář.
- Systém neposkytuje možnost *zobrazit si plány pro uživatele v časové souslednosti*. To by byla velmi užitečná funkcionalita pro trenéry i hráče v rámci hodnocení vývoje a dlouhodobějšího plánování. Zároveň by bylo vhodné tuto část propojit s přehledem obecných metrik i měření realizací cviků, které byly na hodině zaznamenány.
- Systém by mohl být rozšířen o možnost hráčů a návštěvníků prohlížet si vypsané tréninky a následně se na ně přihlašovat. Aktuálně systém nabízí pouze správu ze strany trenéra, avšak popsaná funkcionalita je častým případem užití ve sportovních centrech.
- Systém dobře umožňuje specifikaci realizací cviků, pokud známe daný čas trvání cviku nebo série, může však být užitečné délku určit na základě počtu opakování cviku v sérii - taková možnost v systému aktuálně chybí.

4.4 Zhodnocení testování

Testování akceptačními testy ověřilo, že funkcionalita definovaná v požadavcích je opravdu splněna a že při průchodu přesně definovanými kroky scénářů v prohlížeči nenastávají žádné problémy.

Testování s uživateli pomohlo odhalit některé nedostatky v návrhu rozhraní a ovládání aplikace a dále přineslo další důležité postřehy uživatelů, které budou v budoucnu zohledněny tak, aby používání aplikace bylo co nejvíce intuitivní.

4.4.1 Zhodnocení BDD

Následování metodiky BDD a tedy důsledné řízení se testy při vývoji pro mě bylo novinkou, ale již ze začátku vývoje jsem ji ocenil, neboť poskytuje některé významné výhody. Uživatelské scénáře jsou přehledně uspořádány a směřují vývojáře k implementaci toho, co

je skutečně potřeba. Zároveň je dobře měřitelný postup v implementaci, pokud máme testy, které dobře pokrývají funkcionalitu.

Bodem hovořícím proti mohou být čas a energie, které je potřeba investovat do vytvoření testů, avšak zastávám názor, že u rozsáhlejšího projektu je čas ve výsledku ušetřen a výsledkem je kvalitnější produkt, neboť testy dobře odhalují drobné problémy, které by jinak zůstaly neobjeveny nebo skryty.

Akceptační testy ve formě BDD mi pomohly odhalit velké množství chyb, které by jinak mohly zůstat neobjeveny a dále mi poskytly větší jistotu při vývoji - mohl jsem se spolehnout na ověřeně funkční základ a soustředit se tedy hlavně na další vývoj. V případě, kdy některá část aplikace přestala být funkční, např. v důsledku změny verze nějakého *gemu*, mne testy až na drobné výjimky včasné upozornily a předešel jsem tak hlubšímu zanášení chyby do systému.

4.4.2 Zhodnocení testování s uživateli

Výsledky testování s uživateli jsou cenným zdrojem informací pro vývojáře projektu, pomohou utvářet další budoucnost projektu. Pro ostré nasazení by bylo vhodné otestovat ještě další části aplikace, aby bylo zamezeno pokudmožno co nejvíce chybám ještě před samotným používáním aplikace.

Kapitola 5

Zhodnocení a budoucnost projektu

Rozšíření aplikace **ScApp**, které jsem v rámci této práce vytvořil, poskytuje projektu významně širší míru použitelnosti v praxi.

Před provozním nasazením bude nutné zohlednit alespoň kritické nálezy vzniklé při testování (podsektce 4.3.4). Dalším důležitým bodem plynoucím z testování bude zajistit množinu základních dat tak, aby motivace klubů používat tuto aplikaci byla podpořena. To bohužel není (alespoň v této fázi) v našich silách, rozhodne především další směr vývoje projektu, vývojáři a uživatelé, kteří se projektu budou dále účastnit.

Podle mého názoru je realizace tohoto rozšíření pro projekt přínosná a pomáhá plnit jeho prvotní účel - poskytnout kvalitní alternativu komerčním aplikacím a umožnit komunitě na této alternativě dále pracovat a zlepšovat ji podle potřeb uživatelů, jsem tedy s výsledkem práce spokojen. Mimo přínosu pro projekt je zde i velký přínos osobní, jelikož jsem se naučil pracovat s řadou technologií a získal jsem velmi cennou zkušenost s realizací většího projektu.

5.1 Testovací nasazení

Aplikace je nasazena na službě **Heroku**¹ [27]. Tato služba umožňuje integraci se službou **GitHub**, je tedy vždy nasazena aktuální funkční verze. Aplikaci plánujeme na této službě udržovat jako aktuální rozpracovanou verzi pro demonstrační a testovací účely do zhotovení první produkční verze.

5.2 Budoucnost

Některé položky, které zmínil autor původní práce v popisu budoucnosti původní aplikace, jsou většími celky a čekají na zpracování [7]. Zde tedy budu popisovat možnou budoucnost úzce spjatou s tímto rozšířením.

¹<http://scapp2.herokuapp.com/>

5.2.1 Měření realizací

Jak již bylo zmíněno v 3.1.11, z důvodu nedostatku času jsem nestihl implementovat *měření realizací cviků*. Tato funkcionality je naplánována i navržena, mělo by tedy jít o první další krok v implementaci projektu. Měření umožní trenérům podrobněji reflektovat výkony svých hráčů na tréninku a v závislosti na vývoji těchto hodnot plánovat tréninky.

5.2.2 Mobilní aplikace

Užitečným i atraktivním rozšířením by byla mobilní aplikace umožňující interakci uživatelů s aplikací jinde, než u počítače. Zejména v prostředích, jako je posilovna, může potřeba fyzické přítomnosti u stolního počítače omezit efektivitu vykonávání práce. Aplikace sice díky *responsivnímu designu* dobře funguje na většině přenosných zařízení, aplikace speciálně určená pro daný mobilní operační systém je z hlediska použitelnosti více vhodná. Rozšířenost chytrých telefonů v této době motivaci pro realizaci mobilní aplikace zvyšuje.

Hlavními oblastmi funkcionality mobilní aplikace pro potřeby tohoto rozšíření by bylo *zadávaní hodnot měření* trenéry přímo na tréninku a *zobrazení tréninkových plánů i detailů cviků* trenéry i hráči, pro zlepšený průběh tréninku. Projekt v této fázi poskytuje řadu možností, které by taková aplikace mohla pokrýt, pro úspěch projektu je však nejvhodnější realizovat nejprve oblasti, které budou nejvíce využívány.

5.2.3 Globální repozitář cviků, sad a plánů

Návrh realizace online repozitáře cviků byl již součástí původní práce [7], zde zdůrazňuji jeho význam a pokusím se jej zpřesnit.

Výsledky testování s uživateli (podsekcí 4.3.4) napovídají, že primárním nedostatkem aplikace je (zejména na začátku používání) nutnost zadávat opakovaně data a nemožnost jednoduše začít provoz se základní sadou. Vhodnou náhradou této sady dodávané se systémem je právě *globální repozitář* dostupný všem instancím aplikace, spravovaný komunitou uživatelů (odborně povolaných tento typ dat vytvářet). Řeší se tak mimojiné otázka kvality a kvantity specifikace cviků (zejména při zavedení kolaborace na specifikaci).

Repozitář může poskytnout i sociální aspekt - diskuse trenérů o plánech, sdílení obsahu, aj. Úspěšná realizace takového rozšíření má potenciál dále podpořit stabilitu používání aplikace i vzniklé komunity.

Kapitola 6

Závěr

V rámci práce jsem se nejprve seznámil s výsledky práce Bc. Ondřeje Janaty [7], navázal jsem na průzkum podobných aplikací a přesvědčil se, že navrhované rozšíření bude pro projekt přínosné, jelikož veškeré alternativy jsou komerční a příslušná funkcionality není jinde v této formě zpracována. Po získání veškerých požadavků na systém jsem provedl jejich analýzu a formuloval uživatelské scénáře, které díky zvolené metodice vývoje BDD zároveň obstaraly kvalitní otestování a podklad pro návrh. Dále jsem implementoval požadovanou funkcionality a začlenil vzniklý systém do původního systému. Aplikace je otestována akceptačními testy a testováním s uživateli.

Práce na projektu pro mne měla velký přínos zejména protože jsem měl možnost realizovat projekt většího rozsahu, dále jsem nabyl zkušeností v oblastech návrhu a implementace - naučil jsem se pracovat s nástroji `Ruby on Rails`, `AJAX`, `Twitter Bootstrap v3`, `HAML`, `SCSS`, `CoffeeScript` a v dalších jsem zlepšil své znalosti (`jQuery`, `GIT` a další).

Literatura

- [1] Arlow J. a Neustadt I. *UML 2 and the unified process: practical object-oriented analysis and design*, kap. 3.6, 3.7. Pearson Education, 2005.
- [2] Arlow J. a Neustadt I. *UML 2 and the unified process: practical object-oriented analysis and design*, kap. 4.5. Pearson Education, 2005.
- [3] Chelimsky D. a kolektiv. *RSpec* [online]. 2014. [cit. 5. 5. 2015]. Dostupné z: <<http://rspec.info/>>.
- [4] Hansson D. H. *Ruby on Rails* [online]. 2014. [cit. 5. 5. 2015]. Dostupné z: <<http://rubyonrails.org/>>.
- [5] Hellesøy A. a kolektiv. *Cucumber* [online]. 2014. [cit. 5. 5. 2015]. Dostupné z: <<https://cucumber.io/docs/reference>>.
- [6] Hellesøy A. a kolektiv. *Gherkin* [online]. 2014. [cit. 5. 5. 2015]. Dostupné z: <<https://github.com/cucumber/cucumber/wiki/Gherkin>>.
- [7] Janata O. *Software pro evidenci a analýzu sportovního tréninku*, Bakalářská práce. ČVUT FEL, 2014.
- [8] Janata O. *ScApp - zdrojový kód na GitHub* [online]. 2014. [cit. 13. 3. 2015]. Dostupné z: <<https://github.com/hack006/scapp>>.
- [9] Kolektiv přispěvatelů. *RubyGems* [online]. 2014. [cit. 5. 5. 2015]. Dostupné z: <<https://rubygems.org/>>.
- [10] Lewis C. a Wharton C. Cognitive walkthroughs. *Handbook of human-computer interaction*. 1997, 2, s. 722.
- [11] Lichnovská P. a Karberová E. *Heuristická evaluace, český překlad heuristik* [online]. 2009. [cit. 12. 5. 2015]. Dostupné z: <<http://human-computer-interaction.webnode.cz/testovani-a-hodnoceni-rozhrani/metody-testovani/heuristicka-analyza/>>.
- [12] Molich R. a Nielsen J. Improving a human-computer dialogue. *Communications of the ACM*. 1990, 33, s. 338-340.
- [13] Myers G., Sandler C. a Badgett T. *The art of software testing*, kap. 6, 7. John Wiley & Sons, 2011.
- [14] Oriordan M. *Simulate Drag on JQuery UI Sortable* [online]. 2012. [cit. 11. 5. 2015]. Dostupné z: <<https://github.com/mattheworiordan/jquery.simulate.drag-sortable.js>>.
- [15] Přispěvatelé Wikipedie. *AJAX* [online]. 2015. [cit. 21. 4. 2015]. Dostupné z: <[http://en.wikipedia.org/wiki/Ajax_\(programming\)](http://en.wikipedia.org/wiki/Ajax_(programming))>.
- [16] Přispěvatelé Wikipedie. *Behavior Driven Development* [online]. 2015. [cit. 5. 5. 2015]. Dostupné z: <http://en.wikipedia.org/wiki/Behavior-driven_development>.
- [17] Přispěvatelé Wikipedie. *Licence GPLv3* [online]. 2015. [cit. 20. 4. 2015]. Dostupné z: <http://cs.wikipedia.org/wiki/GNU_General_Public_License>.

- [18] Příspěvatelé Wikipedie. *Licence MIT* [online]. 2015. [cit. 20. 4. 2015]. Dostupné z: <http://en.wikipedia.org/wiki/MIT_License>.
- [19] Příspěvatelé Wikipedie. *ModelViewController* [online]. 2015. [cit. 10. 5. 2015]. Dostupné z: <<http://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller>>.
- [20] Příspěvatelé Wikipedie. *Ruby On Rails* [online]. 2015. [cit. 5. 5. 2015]. Dostupné z: <http://cs.wikipedia.org/wiki/Ruby_on_Rails>.
- [21] Sporka A. *Průběh testování s uživateli, Přednáška předmětu Y39TUR 2013-S-Lecture-03-3-usability-tests* [online]. 2013. [cit. 12. 5. 2015]. Dostupné z: <<https://cent.felk.cvut.cz/courses/Y39TUR/?page=slides>>.
- [22] The JQuery Foundation. *jQuery* [online]. 2015. [cit. 11. 5. 2015]. Dostupné z: <<https://jquery.com/>>.
- [23] The JQuery Foundation. *jQuery UI* [online]. 2015. [cit. 11. 5. 2015]. Dostupné z: <<https://jqueryui.com/>>.
- [24] Training Peaks. *Training Peaks* [online]. 2013. [cit. 20. 3. 2015]. Dostupné z: <<http://trainingpeaks.com/>>.
- [25] Virtuagym. *Virtuagym* [online]. 2015. [cit. 21. 3. 2015]. Dostupné z: <<https://virtuagym.com/en/software/fitness>>.
- [26] Šplíchal K. *Rozšíření ScApp - zdrojový kód na GitHub* [online]. 2015. [cit. 2. 5. 2015]. Dostupné z: <<https://github.com/krystofspl/scapp>>.
- [27] Šplíchal K. *Rozšíření ScApp - testovací nasazení Heroku* [online]. 2015. [cit. 2. 5. 2015]. Dostupné z: <<http://scapp2.herokuapp.com/>>.

Příloha A

Obsah přiloženého CD

Následuje obsah CD přiloženého v práci.

```
.  
|-- diff  
(Soubor popisující veškeré vytvořené nebo změněné soubory)  
|-- diff_names  
(Soubor obsahující názvy veškerých vytvořených nebo změněných souborů)  
|-- prace.pdf  
(PDF soubor s touto prací)  
|-- prace_zdrojovy_kod.tex  
(Zdrojový TEX kód této práce)  
|-- scapp  
(Složka se zdrojovým kódem celého projektu, včetně nových souborů  
- pro spouštění)  
  '-- scapp_new_only  
(Složka se zdrojovým kódem nově vytvořených nebo změněných souborů)
```


Příloha B

Výsledky testování s uživateli

B.1 Metoda testování kognitivním průchodem

Zápis do tabulky výsledků jsem prováděl při interakci ze strany uživatele (případně hned po reakci systému, jelikož na tu se táže otázka Q3), z toho důvodu jsou některé kroky seskupeny.

Participantů jsou označeni zkratkami P1, P2, P3. Pokud u některého participanta byla odpověď „Ne“, je uvedena v tabulce v rámci příslušné otázky a řádek odkazuje na příslušný nález. Popis vyjadřuje souhrn komentářů participantů - pro úplnost pozitivního i negativního charakteru.

UC1: Přidat nový soukromý cvik

Q0 - Trenér chce přidat nový cvik, který bude přístupný pouze jemu.

1. Příklad užití začíná, když uživatel navštíví stránku s výčtem jemu dostupných cviků.
2. Systém zobrazí tlačítko s nápisem „Přidat nový cvik“.
3. Uživatel klikne na zobrazené tlačítko.
4. Systém zobrazí formulář pro vyplnění údajů o cviku.

| Otázka | Odpověď (participant) | Popis | Nález |
|--------|-----------------------|--|-------|
| Q1 | Ano (P1,P2,P3) | Tlačítko je v očekávané oblasti a dobře viditelné. | - |
| Q2 | Ano (P1,P3) | Popisek je adekvátní akci. | - |
| | Ne (P2) | Systém zobrazuje popisek dvakrát - jednou v levém menu a podruhé nahoře na příslušné obrazovce, to může být matoucí. | N1 |
| Q3 | Ano (P1,P2,P3) | Zobrazený formulář poskytuje dostatečnou reakci systému. | - |

5. Uživatel vyplní formulář daty, povinně s následujícími položkami, volitelně další nepovinná pole.

| Otázka | Odpověď (participant) | Popis | Nález |
|--------|-----------------------|--|-------|
| Q1 | Ano (P1,P2,P3) | Formulář je stručný a rozlišuje mezi povinnými a nepovinnými poli. | - |
| Q2 | Ano (P2,P3) | Popisek je adekvátní akci. | - |
| | Ne (P1) | Není zřejmý rozdíl mezi poli „Popis“ a „Podrobný popis“ - jak se projeví která varianta. | N2 |
| Q3 | Ano (P1,P2,P3) | Zobrazený formuář poskytuje dostatečnou reakci systému. | - |

6. Uživatel potvrdí vložení tlačítkem „Vytvořit cvik“.
7. Systém provede validaci korektnosti dat (formát, unikátnost názvu).
8. Systém zobrazí obrazovku s detailem vloženého cviku.

| Otázka | Odpověď (participant) | Popis | Nález |
|--------|-----------------------|--|-------|
| Q1 | Ano (P1,P2,P3) | Tlačítko je dobře umístěno, případné chybové hlášky jsou jasné a srozumitelné. | - |
| Q2 | Ano (P1,P2,P3) | Popisek je adekvátní akci. | - |
| Q3 | Ano (P1) | Při kliknutí na hlavní tlačítko systém zobrazí detail cviku, jak bylo očekáváno. | - |
| | Ne (P2, P3) | Při kliknutí na „Vytvořit cvik bez přidání detailů“ není zobrazena potvrzující hláška, že cvik byl vytvořen. | N3 |

UC2: Zařadit cvik do sady cviků

Q0 - Trenér chce zařadit existující dostupný cvik do existující sady cviků.

- Případ užití začíná, když uživatel navštíví stránku s výčtem jeho sad cviků.
- Systém zobrazí tabulku se všemi sadami cviků, které jsou dostupné uživateli.
- Uživatel klikne v tabulce na odkaz s názvem libovolné sady cviků.
- Systém zobrazí detail sady cviků.

| Otázka | Odpověď (participant) | Popis | Nález |
|--------|-----------------------|---|-------|
| Q1 | Ano (P1,P2,P3) | Seznam je přehledný. | - |
| Q2 | Ano (P1,P2,P3) | Popisek je adekvátní akci. | - |
| Q3 | Ano (P1,P2,P3) | Zobrazení stránky s detailem sady je adekvátní a očekávanou reakcí. | - |

5. Uživatel klikne na tlačítko „Přidat / Odebrat cviky“.
6. Systém zobrazí tabulku s dostupnými cviky. Cviky již obsažené v sadě budou zvýrazněny zelenou barvou, ostatní cviky červenou barvou.

| Otázka | Odpověď (participant) | Popis | Nález |
|--------|-----------------------|---|-------|
| Q1 | Ano (P1,P2) | Tlačítko je na logickém místě. | - |
| | Ne (P3) | Tlačítko pro správu cviků v rámci sady si lze splést s tlačítkem pro upravení sady. | N4 |
| Q2 | Ano (P1,P2,P3) | Popisek je adekvátní akci. | - |
| Q3 | Ano (P1,P3) | Zobrazení stránky s detailem cviků je adekvátní a očekávanou reakcí. | - |
| | Ne (P2) | Nadpis „Cviky obsažené v sadě“ nesouhlasí s tím, že v tabulce jsou zobrazeny všechny cviky, byť barevně odlišeny. | N5 |

7. Uživatel klikne v řádku pro daný cvik na tlačítko „Zahrnout“.
8. Systém zobrazí potvrzovací hlášku.
9. Systém zobrazí v tabulce právě přidáný cvik zvýrazněný zelenou barvou.

| Otázka | Odpověď (participant) | Popis | Nález |
|--------|-----------------------|--|-------|
| Q1 | Ano (P1,P2,P3) | Tlačítko je na logickém místě. | - |
| Q2 | Ano (P1,P2,P3) | Popisek je adekvátní akci. | - |
| Q3 | Ano (P1,P2,P3) | Zobrazení stránky s detailem cviků je adekvátní a očekávanou reakcí. | - |

UC3: Přidat realizaci cviku v rámci cvičebního plánu

Q0 - Trenér chce přidat do plánu pro hráče realizaci nějakého jemu dostupného cviku.

1. Případ užití začíná, když uživatel navštíví stránku pro editaci plánů v rámci své tréninkové hodiny.
2. Uživatel najde v tabulce cviků cvik, jehož realizaci chce vytvořit.

| Otázka | Odpověď (participant) | Popis | Nález |
|--------|-----------------------|--|-------|
| Q1 | Ano (P1,P2,P3) | Obrazovka je srozumitelně rozdělena a nadepsána. | - |
| Q2 | Ano (P1,P2,P3) | Popisek je adekvátní akci. | - |
| Q3 | Ano (P1,P2,P3) | Systém okamžitě reaguje na uživatelský vstup, vše je koncentrováno v jedné lokalitě. | - |

3. Uživatel „uchopí“ myši v tabulce s cviky vybraný cvik.

| Otázka | Odpověď (participant) | Popis | Nález |
|--------|-----------------------|--|-------|
| Q1 | Ne (P1,P2,P3) | Bez předchozí detailní instruktaže není zřejmé, jak se plány vyplňují cviky. Po instruktaži nebyly žádné další problémy. | N6 |
| Q2 | Ano (P1,P2,P3) | Stejně jako u předchozí otázky. | - |
| Q3 | Ano (P1,P2,P3) | Systém po uchopení zobrazuje pod kurzorem myši kopii cviku, je tedy jasné, že se cvik někam přetahuje - podobně jako v běžných aplikacích. | - |

4. Uživatel přetáhne myši cvik do plánu na pozici mezi ostatní realizace, kam chce cvik zařadit.
5. Systém vizuálně potvrdí přidání realizace a aktualizuje posunuté časové údaje všech ostatních realizací v plánu.

| Otázka | Odpověď (participant) | Popis | Nález |
|--------|-----------------------|---|-------|
| Q1 | Ano (P1,P2,P3) | Vizuální zvýraznění „zkopírovaného cviku“, který je zobrazen při držení tlačítka myši, participantům poskytlo veškeré informace nutné k přidání do plánu. | - |
| Q2 | Ano (P1,P2,P3) | Stejně jako u předchozí otázky. | - |
| Q3 | Ano (P1,P2,P3) | Po přidání cviku systém změnil vzhled sloupce a aktualizuje všechny související položky, zpětná vazba je tedy podle participantů dostatečná. | - |

UC4: Řadit série v realizaci cviku

Q0 - Trenér chce změnit pořadí sérií v realizaci cviku v plánu pro hráče.

- Případ užití začíná, když uživatel navštíví stránku pro editaci plánů v rámci své tréninkové hodiny.
- Uživatel najde myši na některou realizaci cviku v rámci některého z plánů.
- Systém zobrazí v oblasti realizace tlačítko „Upravit“.

| Otázka | Odpověď (participant) | Popis | Nález |
|--------|-----------------------|---|-------|
| Q1 | Ano (P1,P2,P3) | Realizace cviku se jinde na stránce neobjevuje, je logické hledat úpravy zde. | - |
| Q2 | Ano (P1,P2,P3) | Stejně jako u předchozí otázky. | - |
| Q3 | Ano (P1,P2,P3) | Popisek je ihned po najetí myši dobře viditelný. | - |

4. Uživatel klikne na tlačítko „Upravit“.
5. Systém zobrazí editační obrazovku.

| Otázka | Odpověď (participant) | Popis | Nález |
|--------|--------------------------|--|-------|
| Q1 | Ano (P1,P2,P3) | Každý z participantů ihned porozuměl, co má dělat. | - |
| Q2 | Ano (P1,P2,P3) | Popisek je adekvátní. | - |
| Q3 | Ano (P2,P3) | Zde participantům připadala realizace zobrazení obrazovky přehledná. | - |
| | Ne (P1) | Uživatel si nemusí všimnout přeměny levého panelu na panel pro úpravu realizace. Přehlednější by bylo přidat ještě výrazný nadpis. | N7 |

6. Uživatel klikne na tlačítko „Upravit sérii“.
7. Systém zobrazí editační obrazovku pro editaci sérií.

| Otázka | Odpověď (participant) | Popis | Nález |
|--------|--------------------------|--|-------|
| Q1 | Ano (P1,P2,P3) | Každý z participantů ihned porozuměl, co má dělat. | - |
| Q2 | Ano (P1,P2,P3) | Popisek je adekvátní. | - |
| Q3 | Ano (P1,P2,P3) | Tato obrazovka následuje akce již dříve použité jinde (styl výpisu sérií, záměna panelu za jiný), tedy reakce je již intuitivní. | - |

8. Uživatel „uchopí“ myši v tabulce se sériemi některou ze sérií a potažením myši změni pořadí sérií.
9. Systém aktualizuje a zobrazí tabulkový výpis sérií cviku a jejich časů.

| Otázka | Odpověď (participant) | Popis | Nález |
|--------|--------------------------|---|-------|
| Q1 | Ano (P1,P2,P3) | Pro řazení sérií jsou potřeba podobné úkony jako již uživatel dříve viděl, akce tedy byla intuitivní. | - |
| Q2 | Ano (P1,P2,P3) | Stejně jako u předchozí otázky. | - |
| Q3 | Ano (P1,P2) | Odezva je adekvátní. | - |
| | Ne (P3) | Série vypadají podobně, když nemají specifikovány žádné další vlastnosti - přehlednost systému by zvýšilo přidání potvrzujícího znamení, že série byla přesunuta. | N8 |

B.2 Metoda testování heuristickou evaluací

U nálezů se odkazují na konkrétní heuristiky, ty jsou popsány v sekci 4.3.3.2. Participanti jsou označeni P4, P5, P6. Popis vyjadřuje souhrn komentářů participantů - zde pouze negativního charakteru. Některé problémy se opakovaly po metodě kognitivního průchodu, až na výjimky jsem tyto problémy v následujícím záznamu ignoroval, duplicitní hodnoty by nepřinesly nic užitečného.

UC1: Přidat nový soukromý cvik

1. Příklad užití začíná, když uživatel navštíví stránku s výčtem jemu dostupných cviků.

| Popis problému | Porušené heuristiky | Participant | Nález |
|---|---------------------|-------------|-------|
| Filtr funguje na principu aktualizace psaním, je však snadné přehlédnout, že v poli se již něco nachází a poté se zobrazují pouze filtrovaná cvičení. | 1,5 | P4,P5 | N9 |

2. Systém zobrazí tlačítko s nápisem „Přidat nový cvik“.
3. Uživatel klikne na zobrazené tlačítko.
4. Systém zobrazí formulář pro vyplnění údajů o cviku.

| Popis problému | Porušené heuristiky | Participant | Nález |
|---|---------------------|-------------|---------------|
| Není dostatečně odlišen význam polí Popis a Detailní popis. | 10 | P6 | (Shodné s N2) |
| Není jasné, jaký formát mají mít pole Zdroje a Video. | 4,5,6 | P4,P5,P6 | N10 |

5. Uživatel vyplní formulář daty, povinně s následujícími položkami, volitelně další nepovinná pole.
6. Uživatel potvrdí vložení tlačítkem „Vytvořit cvik“.
7. Systém provede validaci korektnosti dat (formát, unikátnost názvu).
8. Systém zobrazí obrazovku s detailem vloženého cviku.

UC2: Zařadit cvik do sady cviků

1. Příklad užití začíná, když uživatel navštíví stránku s výčtem jeho sad cviků.

2. Systém zobrazí tabulku se všemi sadami cviků, které jsou dostupné uživateli.

| Popis problému | Porušené heuristiky | Participant | Nález |
|--|---------------------|-------------|-------|
| V případě absence sad není jasně vidět, zda systém žádné sady neobsahuje nebo zda se stala chyba v programu. | 1 | P5 | N11 |
| V případě velkého množství sad bude obtížné sady procházet, dobrá by byla možnost sady řadit podle různých kritérií. | 7 | P4 | N12 |

3. Uživatel klikne v tabulce na odkaz s názvem libovolné sady cviků.
4. Systém zobrazí detail sady cviků.
5. Uživatel klikne na tlačítko „Přidat / Odebrat cviky“.
6. Systém zobrazí tabulku s dostupnými cviky. Cviky již obsažené v sadě budou zvýrazněny zelenou barvou, ostatní cviky červenou barvou.

| Popis problému | Porušené heuristiky | Participant | Nález |
|--------------------------|---------------------|-------------|-------|
| Stejný problém jako N12. | 7 | P4 | N13 |

7. Uživatel klikne v řádku pro daný cvik na tlačítko „Zahrnout“.
8. Systém zobrazí potvrzovací hlášku.
9. Systém zobrazí v tabulce právě přidaný cvik zvýrazněný zelenou barvou.

UC3: Přidat realizaci cviku v rámci cvičebního plánu

Zde nevznikly žádné nové nálezy, které by již nebyly postihnuty metodou kognitivního průchodu.

UC4: Řadit série v realizaci cviku

1. Případ užití začíná, když uživatel navštíví stránku pro editaci plánů v rámci své tréninkové hodiny.
2. Uživatel najede myší na některou realizaci cviku v rámci některého z plánů.
3. Systém zobrazí v oblasti realizace tlačítko „Upravit“.
4. Uživatel klikne na tlačítko „Upravit“.

5. Systém zobrazí editační obrazovku.

| Popis problému | Porušené heuristiky | Participant | Nález |
|---|----------------------------|--------------------|--------------|
| Není nikde viditelné, jaká realizace je aktuálně editována. | 1,6 | P5,P6 | N14 |

6. Uživatel klikne na tlačítko „Upravit série“.

7. Systém zobrazí editační obrazovku pro editaci sérií.

| Popis problému | Porušené heuristiky | Participant | Nález |
|--|----------------------------|--------------------|--------------|
| Nelze zavřít editační obrazovku jinde, než v tabu Upravit realizaci. | 3 | P4,P5,P6 | N15 |

8. Uživatel „uchopí“ myši v tabulce se sériemi některou ze sérií a potažením myši změni pořadí sérií.

9. Systém aktualizuje a zobrazí tabulkový výpis sérií cviku a jejich časů.