

**Univerzita Pardubice**

**Fakulta ekonomicko-správní**

**Identifikace a zpracování požadavků**

**Lenka Horáková**

**Diplomová práce**

**2019**

## ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Lenka Horáková**  
Osobní číslo: **E17538**  
Studijní program: **N6209 Systémové inženýrství a informatika**  
Studijní obor: **Informatika ve veřejné správě**  
Název tématu: **Identifikace a zpracování požadavků**  
Zadávací katedra: **Ústav systémového inženýrství a informatiky**

### Z á s a d y p r o v y p r a c o v á n í :

Cílem práce je zachycení změnových požadavků a jejich zpracování, s důrazem na sdílení nápadů a zkušeností v rámci organizace. Výstupem bude návrh aplikace pro sběr požadavků. Práce bude ověřována ve vybraném prostředí veřejné správy. V práci budou využity nástroje datového modelování.

Osnova:

- Identifikace požadavků.
- Analýza a modelování.
- Návrh aplikace.

Rozsah grafických prací:

Rozsah pracovní zprávy: **cca 50 stran**

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

**ČEGAN, Lukáš.** *Vývoj webových aplikací v PHP a NetBeans.* Pardubice: Univerzita Pardubice, 2015. ISBN 978-80-7395-858-9.

**KALUŽA, Jindřich a Ludmila KALUŽOVÁ.** *Modelování dat v informačních systémech.* Praha: Ekopress, 2012. ISBN 978-80-86929-81-1.

**ROUDENSKÝ, Petr.** *Kvalita softwaru: teorie a praxe.* Prostějov: Computer Media, 2016. ISBN 978-80-7402-294-4.

**ŠIMONOVÁ, Stanislava.** *Databázové systémy I.* Pardubice: Univerzita Pardubice, 2013. ISBN 978-80-7395-702-5.

**ŠIMONOVÁ, Stanislava.** *Procesní řízení.* Pardubice: Univerzita Pardubice, 2014. ISBN 978-80-7395-766-7.

*Šimonová*

Vedoucí diplomové práce:

**doc. Ing. Stanislava Šimonová, Ph.D.**

Ústav systémového inženýrství a informatiky

Datum zadání diplomové práce: **3. září 2018**

Termín odevzdání diplomové práce: **30. dubna 2019**



doc. Ing. Romana Provozničková, Ph.D.  
děkanka

L.S.



doc. Ing. Pavel Petr, Ph.D.  
vedoucí ústavu

V Pardubicích dne 3. září 2018

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 9/2012, bude práce zveřejněna v Univerzitní knihovně a prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 27. 3. 2019

Lenka Horáková

## **PODĚKOVÁNÍ**

Na tomto místě bych ráda poděkovala doc. Ing. Stanislavě Šimonové, Ph.D. za odborné vedení mé diplomové práce, cenné rady, konzultace a čas, který mi a mé práci věnovala. Dále bych ráda poděkovala zaměstnancům Úřadu práce ČR, Kontaktního pracoviště Jablonec nad Nisou Krajské pobočky v Liberci za spolupráci při testování navržené aplikace. V neposlední řadě bych chtěla poděkovat rodině a přátelům za poskytnutou podporu během celého studia.

## **ANOTACE**

Tato práce se zabývá sběrem a zpracováním požadavků, které vznikají v organizacích veřejné správy. K tomuto účelu je navržena aplikace, která jednotlivé požadavky eviduje a pomáhá při nalezení jejich řešení. Návrh aplikace je především zaměřen na možnost vzájemného sdílení nápadů, jak vzniklé požadavky řešit, zkušeností a námětů mezi zaměstnanci dané organizace. Součástí aplikace je i hodnotící systém navržených řešení, který pomáhá vedení organizace v rozhodování o nejvhodnějším řešení vloženého požadavku. Aplikace je v rámci této práce otestována na konkrétním požadavku od jeho založení až po vyhodnocení nejvhodnějšího řešení.

## **KLÍČOVÁ SLOVA**

identifikace požadavků, návrh aplikace, datové modelování, aplikační programové rozhraní, grafické uživatelské rozhraní, validace

## **TITLE**

Identification and processing of requirements

## **ANNOTATION**

This thesis deals with the collection and processing of requirements that arise in public administration organizations. For this purpose, an application is designed to record requests and help to find their solutions. The design of the application is primarily focused on the possibility of sharing ideas how to solve these requirements, experience and ideas among the employees of the organization. The application includes an evaluation system of suggested solutions that helps the organization's leaders in deciding what the most appropriate solution for the requirements is. In this thesis, the application is tested on a specific request from its establishment to the evaluation of the most suitable solution.

## **KEYWORDS**

identification of requirements, application design, data modeling, application programming interface, graphical user interface, validation

# Obsah

|   |           |
|---|-----------|
| ÚVOD .....  | 13        |
| <b>1 POŽADAVKY V ORGANIZACI.....</b>                                      | <b>14</b> |
| 1.1 VÝZNAM POŽADAVKŮ .....  | 14        |
| 1.2 NÁSTROJE PRO IDENTIFIKACI POŽADAVKŮ .....                             | 15        |
| 1.2.1 <i>Sběr informací</i> .....   | 15        |
| 1.2.2 <i>Generování námětů</i> .....                                      | 16        |
| 1.2.3 <i>Třídění a sdružování námětů</i> .....                            | 16        |
| 1.2.4 <i>Benchmarking</i> .....   | 16        |
| 1.2.5 <i>Identifikace příčin problémů</i> .....                           | 17        |
| 1.3 DÍLČÍ SOUHRN .....  | 18        |
| <b>2 NAKLÁDÁNÍ S POŽADAVKY VE VYBRANÉ ORGANIZACI VEŘEJNÉ SPRÁVY .....</b> | <b>19</b> |
| 2.1 CHARAKTERISTIKY ORGANIZACE .....                                      | 19        |
| 2.2 CHARAKTER POŽADAVKŮ.....  | 20        |
| 2.3 DÍLČÍ SOUHRN .....  | 21        |
| <b>3 POTŘEBA KOMUNIKAČNÍHO NÁSTROJE PRO IDENTIFIKACI POŽADAVKŮ .....</b>  | <b>22</b> |
| <b>4 KOMUNIKAČNÍ NÁSTROJ – NÁVRH APLIKACE .....</b>                       | <b>26</b> |
| 4.1 STANOVENÍ FUNKČNÍCH A NEFUNKČNÍCH POŽADAVKŮ .....                     | 26        |
| 4.2 NÁSTIN FUNGOVÁNÍ APLIKACE A PRÁV UŽIVATELŮ.....                       | 27        |
| 4.3 VÝCHOZÍ GRAFICKÝ NÁVRH .....  | 29        |
| 4.4 HLEDÁNÍ PŘÍPADNÝCH NEDOSTATKŮ V NÁVRHU .....                          | 29        |
| 4.5 NÁVRH MOŽNÝCH ROZŠÍŘENÍ APLIKACE.....                                 | 33        |
| 4.6 MODELOVÁNÍ FUNKČNOSTI APLIKACE – DIAGRAMY PŘÍPADŮ UŽITÍ.....          | 34        |
| 4.7 VÝSLEDNÝ GRAFICKÝ NÁVRH APLIKACE.....                                 | 36        |
| 4.7.1 <i>Požadavky</i> .....  | 36        |
| 4.7.2 <i>Řešení</i> .....   | 42        |
| 4.7.3 <i>Správa</i> .....   | 42        |
| 4.7.4 <i>Přihlášení</i> .....   | 44        |
| 4.8 SCÉNÁŘE PŘÍPADŮ UŽITÍ.....  | 44        |
| 4.9 DATOVÉ MODELOVÁNÍ .....   | 46        |
| 4.9.1 <i>Konceptuální model</i> .....                                     | 46        |
| 4.9.2 <i>Technologický model</i> .....                                    | 49        |
| 4.9.3 <i>Implementační model</i> .....                                    | 51        |
| <b>5 IMPLEMENTACE.....</b>  | <b>53</b> |
| 5.1 DATABÁZE .....  | 53        |
| 5.2 APLIKAČNÍ PROGRAMOVÉ ROZHRANÍ .....                                   | 54        |
| 5.3 GRAFICKÉ UŽIVATELSKÉ ROZHRANÍ.....                                    | 58        |

|          |   |            |
|----------|---|------------|
| <b>6</b> | <b>VÝSLEDNÉ APLIKACE .....</b>                                | <b>59</b>  |
| 6.1      | APLIKAČNÍ PROGRAMOVÉ ROZHRANÍ .....                           | 59         |
| 6.2      | GRAFICKÉ UŽIVATELSKÉ ROZHRANÍ .....                           | 62         |
| 6.2.1    | <i>Nastavení aplikace.....</i>                                | 62         |
| 6.2.2    | <i>Číselník oddělení .....</i>                                | 62         |
| 6.2.3    | <i>Správa uživatelů .....</i>                                 | 65         |
| 6.2.4    | <i>Registrace uživatele.....</i>                              | 69         |
| 6.2.5    | <i>Přihlášení .....</i>                                       | 69         |
| 6.2.6    | <i>Nastavení uživatele.....</i>                               | 70         |
| 6.2.7    | <i>Založení požadavku .....</i>                               | 72         |
| 6.2.8    | <i>Prohlížení požadavků .....</i>                             | 73         |
| 6.2.9    | <i>Navržení řešení požadavků a hodnocení řešení .....</i>     | 78         |
| 6.2.10   | <i>Vyhodnocení navržených řešení .....</i>                    | 80         |
| 6.2.11   | <i>Statistika nejlepších řešitelů.....</i>                    | 83         |
| 6.2.12   | <i>Zobrazení vybraných řešení .....</i>                       | 84         |
| 6.2.13   | <i>Další funkce.....</i>                                      | 85         |
| <b>7</b> | <b>VALIDACE APLIKACE.....</b>                                 | <b>87</b>  |
| 7.1      | VERIFIKACE A VALIDACE .....                                   | 87         |
| 7.2      | VÝBĚR POŽADAVKU .....   | 87         |
| 7.3      | OVĚŘENÍ PROCESU IDENTIFIKACE POŽADAVKŮ .....                  | 88         |
| 7.4      | PRŮBĚH PROCESU V PRAXI .....                                  | 90         |
| 7.4.1    | <i>Nastavení aplikace.....</i>                                | 90         |
| 7.4.2    | <i>Zadání požadavku.....</i>                                  | 90         |
| 7.4.3    | <i>Analýza příčin a důsledků .....</i>                        | 91         |
| 7.4.4    | <i>Navržená řešení.....</i>                                   | 91         |
| 7.4.5    | <i>Hodnocení řešení.....</i>                                  | 93         |
| 7.4.6    | <i>Analýza problémových vlivů a jejich důsledků.....</i>      | 93         |
| 7.4.7    | <i>Vyhodnocení nejvhodnějšího řešení .....</i>                | 97         |
| 7.4.8    | <i>Zhodnocení otestování aplikace na daném požadavku.....</i> | 97         |
| 7.5      | DALŠÍ VYUŽITÍ APLIKACE .....                                  | 97         |
| 7.6      | NÁVRHY A PŘIPOMÍNKY ZAMĚSTNANCŮ ZVOLENÉ ORGANIZACE.....       | 98         |
| <b>8</b> | <b>ZÁVĚR .....</b>  | <b>99</b>  |
|          | <b>POUŽITÉ ZDROJE.....</b>                                    | <b>101</b> |
|          | <b>PŘÍLOHA A – SCÉNÁŘE PŘÍPADŮ UŽITÍ .....</b>                | <b>105</b> |
|          | <b>PŘÍLOHA B – REST API.....</b>                              | <b>113</b> |



## Seznam tabulek

|   |     |
|---|-----|
| Tabulka 1: Zobrazení jednotlivých prvků aplikace .....                            | 41  |
| Tabulka 2: Registrace uživatele.....  | 45  |
| Tabulka 3: Symboly Crow's Foot notace.....  | 47  |
| Tabulka 4: Ukázka relace oddeleni .....   | 51  |
| Tabulka 5: Použité datové typy .....  | 52  |
| Tabulka 6: Stavové kódy architektury REST .....                                   | 55  |
| Tabulka 7: FMEA před zavedením změn.....  | 94  |
| Tabulka 8: FMEA s navrženými řešeními.....  | 96  |
| Tabulka 9: Příklad užití: Registrace uživatele .....                              | 105 |
| Tabulka 10: Příklad užití: Změna hesla uživatelem .....                           | 105 |
| Tabulka 11: Příklad užití: Reset hesla uživatelem.....                            | 106 |
| Tabulka 12: Příklad užití: Založení požadavku .....                               | 106 |
| Tabulka 13: Příklad užití: Vložení řešení požadavku / reakce na dané řešení ..... | 106 |
| Tabulka 14: Příklad užití: Vložení hodnocení řešení .....                         | 107 |
| Tabulka 15: Příklad užití: Uzavření požadavku .....                               | 108 |
| Tabulka 16: Příklad užití: Vložení / oprava poznámky k řešení požadavku .....     | 108 |
| Tabulka 17: Příklad užití: Označení použitého řešení.....                         | 109 |
| Tabulka 18: Příklad užití: Reset hesla správcem systému.....                      | 109 |
| Tabulka 19: Příklad užití: Oprava uživatele .....                                 | 109 |
| Tabulka 20: Příklad užití: Smazání uživatele .....                                | 110 |
| Tabulka 21: Příklad užití: Založení nového uživatele .....                        | 111 |
| Tabulka 22: Příklad užití: Založení nového oddělení.....                          | 111 |
| Tabulka 23: Příklad užití: Oprava oddělení nebo ukončení jeho platnosti.....      | 112 |
| Tabulka 24: Příklad užití: Smazání oddělení.....                                  | 112 |
| Tabulka 25: REST API.....   | 113 |

## Seznam zdrojových kódů

|  |    |
|--|----|
| Zdrojový kód 1: Ukázka souboru config.php..... | 59 |
| Zdrojový kód 2: Ukázka souboru config.js ..... | 62 |

## Seznam obrázků

|   |    |
|---|----|
| Obrázek 1: Diagram případů užití .....                                | 35 |
| Obrázek 2: Grafický návrh – otevřený požadavek .....                  | 38 |
| Obrázek 3: Grafický návrh – uzavřený požadavek .....                  | 39 |
| Obrázek 4: Grafický návrh - oddělení .....                            | 43 |
| Obrázek 5: Konceptuální model .....                                   | 48 |
| Obrázek 6: Upravený konceptuální model .....                          | 49 |
| Obrázek 7: Technologický model.....                                   | 50 |
| Obrázek 8: API – ukázka zobrazení dat v internetovém prohlížeči ..... | 57 |
| Obrázek 9: API – autorizace v internetovém prohlížeči.....            | 57 |
| Obrázek 10: API – autorizace.....                                     | 60 |
| Obrázek 11: API – formát dat.....                                     | 60 |
| Obrázek 12: API – nevyplněné parametry .....                          | 61 |
| Obrázek 13: API – vložení parametrů .....                             | 61 |
| Obrázek 14: GUI – číselník oddělení .....                             | 63 |
| Obrázek 15: GUI – nové oddělení .....                                 | 63 |
| Obrázek 16: GUI – oprava oddělení.....                                | 64 |
| Obrázek 17: GUI – potvrzení smazání oddělení .....                    | 64 |
| Obrázek 18: GUI – varování – záznam nelze smazat.....                 | 65 |
| Obrázek 19: GUI – seznam uživatelů.....                               | 65 |
| Obrázek 20: GUI – založení nového uživatele .....                     | 66 |
| Obrázek 21: GUI – zpráva o zaslání hesla .....                        | 67 |
| Obrázek 22: GUI – e-mailová zpráva o registraci uživatele .....       | 67 |
| Obrázek 23: GUI – oprava uživatele .....                              | 68 |
| Obrázek 24: GUI – registrace uživatele .....                          | 69 |
| Obrázek 25: GUI – přihlášení uživatele .....                          | 70 |
| Obrázek 26: GUI – menu uživatele .....                                | 70 |
| Obrázek 27: GUI – změna hesla.....                                    | 71 |
| Obrázek 28: GUI – nastavení uživatele .....                           | 71 |
| Obrázek 29: GUI – menu Požadavky .....                                | 72 |
| Obrázek 30: GUI – nový požadavek .....                                | 72 |
| Obrázek 31: GUI – výběr požadavků .....                               | 74 |
| Obrázek 32: GUI – stromová struktura příspěvků.....                   | 75 |
| Obrázek 33: GUI – označení nově zadaných příspěvků.....               | 76 |

|  |    |
|--|----|
| Obrázek 34: GUI – uzavřený a vyhodnocený požadavek .....                       | 76 |
| Obrázek 35: GUI – návrh řešení je uveden v požadavku .....                     | 77 |
| Obrázek 36: GUI – opětovně otevřený požadavek.....                             | 78 |
| Obrázek 37: GUI – navržení a hodnocení řešení.....                             | 79 |
| Obrázek 38: GUI – zobrazené hodnocení řešení .....                             | 79 |
| Obrázek 39: GUI – zobrazené podrobnosti hodnocení .....                        | 80 |
| Obrázek 40: GUI – uzavření požadavku .....                                     | 80 |
| Obrázek 41: GUI – uzavřený požadavek.....                                      | 81 |
| Obrázek 42: GUI – uzavřený požadavek – označení řešení a vložení poznámky..... | 81 |
| Obrázek 43: GUI – vybrané řešení .....   | 82 |
| Obrázek 44: GUI – oprava poznámky a zobrazené odkazy na vybraná řešení .....   | 82 |
| Obrázek 45: GUI – seznam nejlepších řešitelů .....                             | 83 |
| Obrázek 46: GUI – úprava sestavy pro tisk.....                                 | 84 |
| Obrázek 47: GUI – náhled tisku sestavy .....                                   | 84 |
| Obrázek 48: GUI – zobrazení vybraných řešení .....                             | 85 |
| Obrázek 49: GUI – řazení sloupců tabulky .....                                 | 86 |
| Obrázek 50: Analýza příčin a důsledků.....                                     | 92 |

## Seznam zkratek

|       |   |
|-------|---|
| API   | Application Programming Interface - rozhraní pro programování aplikací                                |
| AJAX  | Asynchronous JavaScript and XML - technologie využívaná ve webových aplikacích                        |
| CSS   | Cascading Style Sheets – jazyk pro popis zobrazení jednotlivých prvků                                 |
| FMEA  | Failure Mode and Effects Analysis – Analýza možných způsobů a důsledků poruch                         |
| FMECA | Failure mode, effects and criticality analysis – Analýza způsobů, důsledků a kritičnosti poruch       |
| GUI   | Grafické uživatelské rozhraní   |
| HTML  | Hypertext Markup Language – značkovací jazyk používaný pro tvorbu webových stránek                    |
| HTTP  | Hypertext Transfer Protocol – protokol pro přenos různých souborů mezi webovým serverem a prohlížečem |
| JSON  | JavaScript Object Notation – jednotný formát pro výměnu dat   |
| KoP   | Kontaktní pracoviště  |
| KrP   | Krajská pobočka   |
| MPSV  | Ministerstvo práce a sociálních věcí  |
| OZ    | Oddělení zprostředkování  |
| PFMEA | Process Failure Mode Effects Analysis – Analýza možných způsobů a důsledků poruch v procesu           |
| PHP   | Hypertext Preprocessor – skriptovací programovací jazyk   |
| REST  | Representational State Transfer - rozhraní pro jednotný a snadný přístup ke zdrojům                   |
| SQL   | Structured Query Language – standardizovaný strukturovaný dotazovací jazyk                            |
| SSP   | Státní sociální podpora   |
| UI    | User interface – uživatelské rozhraní   |
| ÚP ČR | Úřad práce České republiky  |
| URI   | Uniform Resource Identifier – jednotný identifikátor zdroje   |
| URL   | Uniform Resource Locator – jednotná adresa zdroje   |
| XML   | eXtensible Markup Language – rozšiřitelný značkovací jazyk  |

# Úvod

S rozmachem internetu a sociálních sítí je čím dál tím více běžné, že lidé mezi sebou sdílejí různé informace, zážitky a zkušenosti, navazují nová přátelství, učí se novým dovednostem apod. Informace se stávají dostupnější pro kohokoliv, kdo je požaduje a lidé také čím dál tím více využívají internet k jejich získání, a to nejen prostřednictvím různých článků s návody a postupy, ale i různých internetových fór, kde se sdružují lidé se stejnými zájmy a kde si vzájemně radí, vyměňují názory a předávají zkušenosti. Lidé se tak snaží prostřednictvím internetu dozvědět od druhých jejich názor na danou věc, problém či situaci a do snahy jim pomoci a poradit se zapojují různí lidé, kteří s nimi své zkušenosti a rady sdílejí.

Tato práce se bude zabývat možností využití vzájemného sdílení informací, rad a zkušeností, ne v rámci internetu, ale v rámci organizace veřejné správy. Ve veřejné správě je zatím příležitostí, jak sdílet své zkušenosti, požádat o radu či o vyřešení nějakého problému či požadavku tímto způsobem, daleko méně. Přitom se může jednat o situace, které již někdo řešil, má s nimi nějakou zkušenost, řešil podobný problém nebo má nápad, jak situaci vyřešit. Většinu těchto situací však ve veřejné správě řeší pouze vedení organizace, které se snaží najít co nejlepší řešení, ovšem jen na základě svých poznatků a zkušeností. Sdílení zkušeností pak většinou probíhá jen v rámci pořádaných porad.

Cílem této práce tedy bude navržení systému, který by byl inspirován principem internetových fór a přinesl tak možnosti vzájemného sdílení zkušeností do prostředí veřejné správy a tím zapojení do procesu hledání řešení vzniklých situací či požadavků všech zaměstnanců, resp. těch, kteří by se na nich chtěli podílet. Na hledání řešení by se tak nepodíleli pouze vedoucí pracovníci, jak je to obvyklé.

Aby bylo takovéto sdílení požadavků, rad a zkušeností mezi všemi zaměstnanci možné, bude navržena aplikace, která bude umožňovat všem zaměstnancům zadávat své požadavky či náměty, sdílet je s ostatními a podílet se na jejich řešení. Všichni zaměstnanci tak budou mít příležitost vyslovit problém nebo námět a každý se bude moci zapojit do jeho řešení.

Navržená aplikace bude vyzkoušena na konkrétním požadavku od jeho založení až po vyhodnocení navržených řešení. Aplikace by tak mohla přispět k větší příležitosti zapojení zaměstnanců do chodu organizace, větší informovanosti zaměstnanců, ke zlepšení jednotlivých procesů, které se v organizacích veřejné správy vykovávají, pracovních podmínkách zaměstnanců, jejich celkové spokojenosti apod.

# 1 Požadavky v organizaci

## 1.1 Význam požadavků

Co je požadavek? Požadavkem může být nějaká potřeba, žádost či přání, jinými slovy něco, co požadujeme, aby bylo splněno. Tyto požadavky mohou vznikat za různým účelem a v různém prostředí. Asi nejčastěji se vyskytují požadavky na konkrétní výrobek či službu, resp. na jejich vlastnosti. Většina lidí má nějakou představu o tom, co by daný výrobek či služba, kterou chtějí, měla splňovat, tedy, co od něj požadují. Pokud jsou tyto požadavky v určité míře splněny, jsou ochotni za vybraný výrobek či službu vynaložit finanční prostředky.

Na plnění požadavků je zaměřena většina firem. Firmy, které obchodují se zbožím či nabízejí nějaké služby, se snaží zjistit, jak jsou zákazníci spokojeni s nabízeným zbožím či službami a jaké požadavky na tyto výrobky či služby mají. Podobné zaměření mají i firmy, které zboží vyrábějí. Čím je zákazník spokojenější, tím je ochotnější za službu či výrobek vynaložit více peněžních prostředků, což pro firmu představuje vyšší zisk.

To, kolik jsou lidé ochotni za daný výrobek či službu vynaložit finančních prostředků, se samozřejmě může lišit v závislosti na prostředí, majetkových poměrech lidí, jejich zvyklostech, klimatických podmínkách apod. Pokud se firmám podaří s akceptovatelnými náklady splnit požadavky zákazníků, mají šanci dosáhnout vyšších příjmů a obstát v mnohdy velké konkurenci. V těchto oblastech je tedy zjišťování požadavků velmi důležité a má své nezastupitelné místo.

Na plnění požadavků však nejsou zaměřeny pouze soukromé firmy, ale požadavky se vyskytují i ve veřejné správě, která by měla vykonávat svou správu v zájmu veřejnosti, což by ve většině případů mělo znamenat v souladu s požadavky veřejnosti, resp. občanů. Občané své požadavky mohou vyjádřit jednak prostřednictvím voleb či referend, nebo i prostřednictvím podané konkrétní žádosti o vyřízení určité záležitosti.

Různé požadavky tedy mohou mít jak zákazníci, tak i občané. Zapomínat by se ovšem nemělo ani na ty, kteří lidem pomáhají tyto požadavky prostřednictvím firem či jiných organizací splnit – tedy na zaměstnance. Ti mohou mít požadavky vzhledem ke své vykonávané práci, pracovnímu prostředí, týkajících se jich samotných apod. O tyto požadavky by se měl zajímat zaměstnavatel, který by se měl snažit o spokojenost svých zaměstnanců, ať už se jedná o soukromou nebo veřejnou sféru. Tato snaha by měla být ze

strany zaměstnanců oplácena vysokým pracovním nasazením a snahou o dobré jméno úřadu, společnosti či firmy, tj. o kvalitní výrobky či služby, což v případě firem znamená opět vyšší zisky, u veřejné správy spokojenější klienty, resp. občany.

Jak bylo uvedeno výše, požadavky zaměstnanců se mohou týkat jich samotných, ale i procesů, které vykonávají a které by podle jejich úvahy potřebovaly nějakou změnu. Ta by měla vést k zlepšení procesu, tj. jeho zjednodušení, zrychlení, zpřehlednění apod., což by opět mělo vést k vyšší efektivitě, resp. zisku firmy.

Ve většině případů jsou tyto procesy nastaveny vedením společnosti či jejím managementem. Ten ovšem někdy nemusí mít k dispozici všechny informace, které mají zaměstnanci, kteří v daném procesu přímo pracují. Proto by vedení společnosti, popř. management, neměli opomíjet požadavky a postřehy svých zaměstnanců, neboť mnohdy mohou být velmi cenné a inspirující.

Dobry zaměstnavatel by měl tedy nejen umět využít potenciálu každého zaměstnance, jeho zkušeností a nápadů, ale naslouchat i různým požadavkům, které by vedly k větší spokojenosti zaměstnanců. K tomu je samozřejmě potřeba, aby měl každý zaměstnanec možnost své náměty a požadavky vyjádřit a byl v tom vedením organizace či firmy podporován.

## **1.2 Nástroje pro identifikaci požadavků**

K účelu zjištění možností, jakými lze jednotlivé požadavky od zaměstnanců získávat a zpracovávat, budou prozkoumány jednotlivé nástroje pro identifikaci požadavků. Následující výčet nástrojů pro identifikaci požadavků je uveden na základě knihy Zlepšování podnikových procesů. [51]

### **1.2.1 Sběr informací**

*Pozorování* – slouží především k pochopení toho, jak proces probíhá, neboť bez toho, aby byl probíhající proces dobře pochopen, nelze požadavky či náměty podávat. Každý námět a požadavek musí vycházet z dobrého porozumění realitě. [51]

*Analýza písemné dokumentace* – patří mezi nezastupitelné nástroje, neboť z písemné dokumentace je možné zjistit podrobnosti o chování procesu. [51]

*Interview s účastníky procesu* – používá se v kombinaci s výše uvedenými metodami. [51]

### **1.2.2 Generování námětů**

K získávání požadavků a námětů se používají diskuze. Jsou založeny na jednotlivcích, kteří mohou být zdrojem různých nápadů, námětů či postřehů, ať už jsou v procesu zainteresování nebo ne. Aby bylo možné získat neocenitelné nápady či postřehy, musí se pracovní skupina správně motivovat a řídit. Skupinové diskuze mají také důležitou úlohu při hodnocení, kdy se k navrženým námětům mohou účastníci skupiny vyjádřit a pomoci sestavit žebříček priorit. [51]

Ke generování námětů lze použít tyto nástroje [51]:

*Řízená diskuze* – je významný nástroj k dosažení nějakého cíle a získání podmětných námětů.

Diskuzi musí vést odborník, který ji směřuje k požadovanému cíli a podněcuje iniciativu zúčastněných.

*Brainstorming* – jedná se o skupinovou práci, při které se shromažďují náměty a porovnávají se a hodnotí různé varianty. Vítáno je co nejvíce nápadů a neobvyklá řešení, žádný z nápadů není předem zavrhnut, ale všechny jsou podrobeny přezkoumání. [53]

*Individuální diskuze* – může být variantou brainstormingu. Je však zkreslena tím, že je na daný proces nahlíženo individuálním pohledem.

### **1.2.3 Třídění a sdružování námětů**

Pokud je k dispozici velké množství námětů, které jsou si podobné, mohou se použít diagramy pro třídění a sdružování námětů. Náměty jsou sdružovány podle společné vlastnosti. Zpravidla se jedná o komplikované a rozsáhlé problémy, které vykazují stejné nebo příbuzné náměty. [51]

### **1.2.4 Benchmarking**

Benchmarking je založen na porovnání výrobků, služeb a procesů s těmi organizacemi, které jsou uznané v oboru jako nejlepší [53] a pomáhá v aplikaci těch postupů, které jsou ověřeny praxí podobných podniků nebo i zcela odlišných organizací [51]. Mimo jiné rozlišujeme:

*Procesní benchmarking*, který je zaměřen na zavedení dobré praxe nebo dílčích postupů, které používají jiné společnosti a které se osvědčily. [51]

*Produktový benchmarking*, který je zaměřen na porovnání vlastností výrobků nebo poskytovaných služeb. [51]



### 1.2.5 Identifikace příčin problémů

Příčiny problémů lze identifikovat buď subjektivním hodnocením např. v rámci skupinové diskuze, nebo více objektivním hodnocením pomocí metod, které jsou založeny na strukturální analýze a logice. K identifikaci příčin problémů slouží nástroje jednak pro odhalení potenciálních příčin problémů, tak i pro ověření, zda se potenciální příčiny na vzniklém problému skutečně podílejí. [51]

Nejčastějšími metodami jsou [51]:

*Kontrolní seznamy* – mohou být ve formě různých tabulek obsahující sledy kroků, položek určených pro závěrečné ověřování nebo tabulek pro následné ověření a porovnání vlastností dvou podobných produktů. Tento nástroj je vhodný pro různé zkoušky a testy, a to i opakované, nebo pro stabilizaci implementovaných změn.

*Paretovy diagramy* – umožňují identifikovat prioritní problémy [40]. Jsou postaveny na teorii, že osmdesát procent následků je způsobeno dvaceti procenty příčin. [50] Paretův diagram je speciálním typem sloupkového grafu [40], který zobrazuje jednotlivé příčiny problémů a pomáhá se zaměřit na 20 % nejdůležitějších příčin, které způsobují 80 % problémů [41].

*„Pětkrát proč?“* – je metoda pro zjištění skutečné základní (kořenové) příčiny [53] a vede účastníky k tomu, aby se hlouběji zamysleli nad případnými příčinami problémů, a to zpravidla v rámci diskusní skupiny. Metoda je založena na kladení otázek, které vedou k jádru problému, přitom každá další otázka jde do hlubšího detailu problému než předchozí.

*Analýza příčin a důsledků* – je také známá jako diagram „Rybí kost“. Jedná se o efektivní nástroj pro popis problému a jeho příčin. Diagram „Rybí kost“ zobrazuje souvislosti mezi následkem a jeho možnými příčinami. Problém (následek) na diagramu tvoří „hlavu ryby“. Příčiny jsou znázorněny jako „rybí kosti“. Nejprve jsou identifikovány v rámci jednotlivých kategorií hlavní (primární, prvotní) příčiny, následně pak vedlejší (sekundární, druhotné) příčiny. [53]

*Analýza problémových vlivů a jejich důsledků* nebo také analýza možných způsobů a důsledků závad [12] (FMEA), která se v oblasti procesů označuje jako PFMEA. FMEA je zaměřená na identifikaci způsobů, jakými může proces, výrobek či služba selhat. V této metodě jsou evidovány možnosti vzniku vad, jejich příčiny a následky. Rozšířenou metodou je metoda FMECA (Analýza způsobů, důsledků a kritičnosti

poruch) [7], která navíc eviduje hodnocení závažnosti selhání, možného výskytu poruchy a šance na odhalení poruchy kontrolním mechanismem. Z jednotlivých hodnocení se vypočítá míra rizika. Míra rizika ukazuje na prioritu zavedení nápravných opatření, která by měla eliminovat příčiny selhání procesu. [13] FMEA může být použita i v nevýrobních oblastech, např. pro analyzování rizik v procesu státní správy nebo při hodnocení bezpečnostního systému. FMEA by se měla použít při vzniku nového procesu, modifikaci stávajícího procesu nebo použití stávajícího procesu v novém prostředí. [1]

### **1.3 Dílčí souhrn**

Z výše uvedeného vyplývá, že získávání požadavků je velmi důležité, ať již se jedná o jakékoliv prostředí, neboť může mimo jiné přinést firmám větší zisky, zaměstnancům větší spokojenost a celkově může dojít k zlepšení jednotlivých procesů v organizaci. K účelu zjištění možností, jakými lze jednotlivé požadavky získávat a zpracovávat, byly vyhledány nástroje k tomu určené. Z nich je patrné, že pro identifikaci požadavků je nutné znát procesy, kterých se identifikace požadavků týká a shromáždit co nejvíce námětů a nápadů pomocí skupinových nebo individuálních diskuzí. Jednotlivé náměty, které si jsou navzájem podobné, by se měly sdružovat do skupin. Využít se mohou i nápady, které jsou již ověřené jinou organizací či jiným oddělením. Pro kompletní identifikaci je dobré zjistit příčiny problémů, popř. jejich důsledky.

## **2 Nakládání s požadavky ve vybrané organizaci veřejné správy**

### **2.1 Charakteristiky organizace**

Charakter požadavků, které mohou v prostředí veřejné správy vznikat, bude zjišťován na základě informací získaných z vybrané organizace veřejné správy. Zvolena bude střední nebo větší organizace, pro kterou je vzhledem k její velikosti již nemožné, aby požadavky a náměty zaměstnanců byly osobně sdíleny mezi všemi zaměstnanci. Zvolená organizace musí také být pro náměty a názory svých zaměstnanců otevřená. Jednou z takových organizací je Úřad práce České republiky (ÚP ČR), Kontaktní pracoviště (KoP) Jablonec nad Nisou Krajské pobočky (KrP) v Liberci, která byla také k výše uvedeným účelům vybrána.

Úřad práce ČR byl zřízen 1. 4. 2011 zákonem č. 73/2011 Sb., o Úřadu práce České republiky a o změně souvisejících zákonů. Je organizační složkou státu a správním úřadem s celostátní působností, který vykonává veřejnou správu. Jeho nadřízeným orgánem je Ministerstvo práce a sociálních věcí (MPSV), které ho také řídí. Úřad práce ČR je tvořen generálním ředitelstvím, podřízenými krajskými pobočkami a pobočkou pro hlavní město Prahu, dále kontaktními pracovišti, která jsou součástí krajských poboček. [38]

Na Úřadu práce ČR, Krajské pobočce v Liberci, Kontaktním pracovišti Jablonec nad Nisou pracuje zhruba 100 zaměstnanců. V čele úřadu je ředitelka KoP Jablonec n. N. ÚP ČR. Zvláštností úřadu může být, že na ÚP ČR, KrP v Liberci, KoP Jablonec nad Nisou pracují nejen zaměstnanci Kontaktního pracoviště Jablonec nad Nisou, ale i zaměstnanci Krajské pobočky v Liberci. Vzhledem k tomu, že zaměstnanci Krajské pobočky v Liberci mohou pracovat na čtyřech různých kontaktních pracovištích, není výjimkou, že na jednom oddělení na daném kontaktním místě pracuje jen jeden nebo dva zaměstnanci.

Zaměstnanci KoP Jablonec nad Nisou jsou rozděleni do následujících šesti oddělení [32]:

- Oddělení sekretariátu
- Oddělení trhu práce
- Oddělení zprostředkování a podpory v nezaměstnanosti
- Oddělení poradenství a dalšího vzdělávání
- Oddělení státní sociální podpory, dávek pěstounské péče, příspěvků na péči a dávek pro osoby se zdravotním postižením
- Oddělení hmotné nouze

Zaměstnanci KrP v Liberci pracující na KoP Jablonec nad Nisou jsou rozděleni do následujících sedmi oddělení [32]:

- Oddělení ekonomické
- Oddělení majetku a investic
- Oddělení informatiky
- Oddělení trhu práce
- Oddělení projektů EU
- Oddělení právní
- Oddělení specializovaných kontrol

Další odlišností od ostatních úřadů veřejné správy může být to, že jednotlivá kontaktní pracoviště mohou být mezi sebou různě provázaná. V tomto případě Kontaktní pracoviště Jablonec nad Nisou zajišťuje pro Kontaktní pracoviště Tanvald organizační, mzdové a personální záležitosti. Jednotlivá kontaktní pracoviště také řídí zřízená detašovaná pracoviště. Kontaktní pracoviště Jablonec nad Nisou tak řídí Detašované pracoviště Železný Brod. Žádná z těchto odlišností však nemá zásadní vliv na charakter požadavků organizace.

## **2.2 Charakter požadavků**

V této organizaci mohou od zaměstnanců vznikat různé požadavky, které se mohou týkat jich samotných, organizačních záležitostí, jejich pracovního prostředí, ale i procesů, které v rámci vyřizování žádostí či jiných úkonů vykonávají. Tyto požadavky tak mohou např. přispívat ke zrychlení procesů, snížení rizika či zvýšení kvality procesů.

Tyto požadavky či návrhy mohou vzniknout buď změnou zákona, na jejímž základě je nutné změnit dotčené procesy či nastavit jiné kontrolní mechanismy, nebo příchodem nového pracovníka, který vidí nastavené procesy a ostatní oblasti nezaujatým pohledem, návštěvou jiného pracoviště nebo pouhým zamyšlením, zda by nešly věci dělat jinak. Možností, proč navrhnout nějakou změnu může být celá řada.

Být v navrhování změn aktivní by mělo být v zájmu každého zaměstnance, neboť se jedná především o zlepšení jeho pracovních záležitostí. Zájem by mělo mít i vedení organizace, neboť jednotlivé návrhy mohou přispět jak k větší spokojenosti zaměstnanců, k lépe a rychleji odvedené práci, tak i k větší spokojenosti klientů.

Zaměstnanci samozřejmě mohou podávat své podněty nebo postřehy u svých nadřízených. Ti mohou požadavek řešit buď sami, nebo v rámci porady, a to jak porady vedení, tak i porady jednotlivého oddělení – to záleží na povaze požadavku. Tímto způsobem se však málokdy o daném požadavku dozvědí všichni zaměstnanci, neboť se daný požadavek může řešit jen v rámci užší skupiny zaměstnanců, nebo může být vedoucím oddělení ihned zamítnut jako nerealizovatelný, či z jiného důvodu.

Přínosné by však bylo, kdyby se o navržených požadavcích dozvěděli všichni zaměstnanci, neboť i když je požadavek odmítnutý, tak může některého zaměstnance inspirovat k jinému realizovatelnému požadavku. Na řešení požadavku by se tak mohli podílet všichni zaměstnanci. Tím by se zvýšila šance, že se najde nejvhodnější řešení. Ideálně by se všichni zaměstnanci měli dovědět i o všech navržených řešeních. Tak by na ně mohli reagovat a upozornit na některá úskalí, která nejsou všem známa, nebo rozvinout či upravit již navržené řešení někoho jiného. Zaměstnanci by si tak mohli navzájem více pomáhat, radit si a sdílet a prezentovat své názory.

## **2.3 Dílčí souhrn**

V této kapitole byla charakterizována vybraná organizace veřejné správy a nastíněn vznik a řešení požadavků v této organizaci. Uvedeny byly i nevýhody zavedeného způsobu podávání požadavků. Z uvedeného vyplynulo, že přínosem by bylo umožnění vzájemného sdílení vzniklých požadavků mezi všemi zaměstnanci. Sdílet by se měla i navržená řešení daných požadavků tak, aby zaměstnanci mohli na jakékoliv požadavky a návrhy řešení otevřeně reagovat.

### **3 Potřeba komunikačního nástroje pro identifikaci požadavků**

Vzájemně sdílet požadavky a návrhy řešení by bylo možné například:

- ústně
- e-mailem
- prostřednictvím „knihy přání a stížností“
- použitím anonymní krabice
- pomocí aplikace

Každá z těchto uvedených možností má však svá úskalí. Ústní forma předávání požadavků by byla vhodná pouze pro organizaci s velmi malým počtem zaměstnanců. Uskutečňovat by se mohla prostřednictvím porad, kterých by se účastnili všichni zaměstnanci. Navíc by zaměstnanci museli s novým řešením čekat vždy na další poradu nebo řešení vymyslet během omezené doby konání porady.

Podávání požadavků a řešení prostřednictvím e-mailu by sice poskytlo zaměstnancům možnost se kdykoliv k požadavku či navrženému řešení vyjádřit, ale tento způsob by byl velmi nepřehledný. Pro zaměstnance by bylo také časově náročné dohledat všechna navržená řešení k danému požadavku, zvláště v případě, kdyby se řešilo několik požadavků najednou.

Ani „Kniha přání a stížností“ nenabízí přehlednou formu, jak k daným požadavkům psát navržená řešení a další reakce. Navíc by do knihy nemohlo v jednu chvíli psát více zaměstnanců najednou – to se týká i čtení zapsaných požadavků a reakcí ostatních. Kniha by také byla umístěna na jednom místě, ke kterému by měli někteří zaměstnanci horší přístup než jiní.

Podobné problémy by vyvstaly v případě použití anonymní krabice, do které by zaměstnanci mohli vhazovat své požadavky. Ty by se museli nějakou vhodnou formou zveřejňovat, aby na ně mohli ostatní zaměstnanci reagovat. Vyřešit by se muselo i zveřejňování navržených řešení. Problém by mohl nastat i v případě, že by navržené řešení bylo špatně formulováno a nebylo zcela jasné, k jakému požadavku patří.

Nejvhodnějším řešením by byl návrh aplikace, která by umožňovala evidenci požadavků a jejich řešení a do které by měli přístup všichni zaměstnanci a mohli tak do ní přispívat svými postřehy. Dále by bylo vhodné do této evidence zaznamenávat vybraná řešení z navržených řešení, která budou následně realizována. Použitím aplikace pro uvedené účely

vznikne i větší časový prostor pro vymýšlení optimálního řešení (zaměstnanci budou moci navrhnout svá řešení několik dní), který by např. na poradě nevznikl. Nevýhodou této formy sdílení požadavků je fakt, že se jedná o nejnáročnější řešení (pokud aplikace neexistuje) z výše uvedených. To ovšem nemění nic na faktu, že se jedná o nejvhodnější řešení a proto bude dále zpracován návrh na zmíněnou aplikaci.

Návrh aplikace bude zpracován na základě informací získaných z vybrané organizace veřejné správy. Jelikož se předpokládá, že mezi jednotlivými organizacemi veřejné správy by neměl být žádný zásadní rozdíl, který by měl vliv na návrh aplikace, měla by navržená aplikace být použitelná pro všechny organizace veřejné správy.

Menší rozdíly mohou být pouze v organizační struktuře organizace, a to jak v její struktuře, tak i v názvech jednotlivých složek – některé organizace mohou mít odbory, jiné oddělení či referáty. Rozdíly samozřejmě budou i ve složitosti organizační struktury. Společnými vlastnostmi však je, že za každou organizační složku je odpovědný vedoucí, kterému jsou podřízeni určití zaměstnanci a že každá organizace veřejné správy v principu vykonává stejnou činnost – vyřizuje různé žádosti či záležitosti a/nebo vykonává nějaký proces podle zákona a stanovených postupů.

Vzhledem k tomu, že ve vybrané organizaci a nejspíše ve většině organizací veřejné správy existuje aplikace typu „HelpDesk“ určená na zadávání požadavků na opravu různých zařízení, výměnu tonerů či hlášení nefunkčnosti softwaru apod., nebude navržená aplikace určena k zadávání těchto požadavků, které vyžadují úkon jednoho pracovníka nebo mají daný postup řešení. Aplikace by měla spíše sloužit pro zadávání a sdílení situací a problémů, ke kterým potřebují zaměstnanci organizace najít vhodné řešení. Vybraná organizace nemá aplikaci, pomocí níž by se mohly řešit požadavky, které se netýkají hardwaru nebo softwaru, ale např. organizačních záležitostí nebo která by umožňovala vzájemně sdílet zkušenosti při řešení různých problémů k dispozici a ani nebylo zjištěno, že by podobnou aplikaci měla některá organizace veřejné správy.

Návrh aplikace na sběr a řešení požadavků zaměstnanců veřejné správy bude inspirován výše uvedeným nástroji pro identifikaci požadavků. Jednotlivé nástroje budou do návrhu aplikace implementovány v tomto rozsahu:

### **Sběr informací**

Nástroj pro sběr informací, resp. získávání znalostí jednotlivých procesů, nebude do návrhu aplikace implementován, neboť jednotlivé požadavky budou navrhnout zaměstnanci,

kteří dotčené procesy znají, popř. může být dán v aplikaci prostor na vysvětlení konkrétního procesu nebo jeho části, pokud to bude někdo požadovat.

### **Generování námětů**

Ke generování námětů, resp. požadavků, bude využita skupinová diskuze. Návrh aplikace může být inspirován brainstormingem tak, aby bylo uživateli navrhováno co nejvíce nápadů a námětů. Přitom musí platit, že žádný námět nebo nápad nesmí být z aplikace smazán, ale musí se jím někdo zabývat. Různé nápady a náměty by se měly dát mezi sebou navzájem porovnávat, resp. nějakým způsobem je hodnotit s jasným výsledkem.

Za každou skupinu námětů, resp. za zadaný požadavek, by měl být stanoven alespoň jeden odpovědný pracovník, který by měl inspirovat a podněcovat ostatní uživatele k navrhování různých nápadů a zaujmout k navrženým námětům a nápadům konečné stanovisko. To může být projednáno v rámci určené pracovní skupiny.

### **Třídění a sdružování námětů**

Navržená řešení by se měla přiřazovat k odpovídajícímu požadavku, aby tak vznikaly jednotlivé skupiny se společnými vlastnostmi, resp. se společným požadavkem a daný požadavek se řešil „na jednom místě“.

### **Benchmarking**

Navržené požadavky a řešení mohou být samozřejmě inspirovány již vyzkoušenými řešeními z jiných organizací. Radit si a dávat podněty na vyřešení situace či problému by měli mít možnost i zaměstnanci napříč odděleními.

### **Identifikace příčin problémů**

Pro možnost získání většího množství námětů lze využít nástroje pro identifikaci příčin problémů. Zjištěné příčiny mohou být inspirací k navržení dalších námětů, resp. návrhů na řešení těchto příčin. K tomuto účelu může být využita především analýza příčin a důsledků. Vzhledem k charakteru tohoto nástroje se ale nepředpokládá, že bude součástí navržené aplikace. Tato analýza bude nejspíše prováděna jen u složitějších nebo rozsáhlejších požadavků, a to vedoucími pracovníky.

Dále je možné použít analýzu problémových vlivů a jejich důsledků pro zjištění, zda navrhované náměty nebudou mít nějaké negativní důsledky. Tento nástroj však také nebude součástí navrhované aplikace a bude použit pouze u složitějších požadavků. I když samotné



poslední dvě zmiňované analýzy nebudou součástí navrhované aplikace, měla by navrhovaná aplikace umožňovat výsledky těchto analýz zaznamenat.

## 4 Komunikační nástroj – návrh aplikace

### 4.1 Stanovení funkčních a nefunkčních požadavků

Při vzniku softwaru je nejprve nutné zjistit, co má požadovaný software dělat. Nejprve se tedy musí nashromáždit všechny požadavky, které mají uživatelé na požadovaný software, neboť každá dodatečná změna softwaru je samozřejmě nákladnější než zapracování změny ve fázi sběru požadavků. Proto je nutné se pečlivě věnovat definování požadavků a tuto fázi nepodceňovat, neboť může ušetřit nemalé finanční prostředky a investovaný čas.

Pod pojmem požadavek na software je myšleno popis určité funkce nebo vlastnosti, kterou uživatel systému požaduje, aby software obsahoval. V podstatě je to soupis přání uživatele požadovaného softwaru, který vyjadřuje, co by měl systém dělat. Přitom neřeší, jak by to měl systém dělat, resp. jak má být daného požadavku dosaženo. [2], [29]

Uživatel může poskytnout buď podrobný popis požadavků, nebo má jen hrubou představu o fungování softwaru a detaily nechá na navrhovateli systému. Ten musí požadavky rozpracovat do detailního návrhu, který následně konzultuje s uživateli systému. V některých případech se uživatel systému spolehne na zkušenosti navrhovatele a i detaily systému nechá na něm. [29]

Požadavky lze rozdělit na [29]:

- *Funkční* – specifikují funkce systému
- *Nefunkční* – specifikují vlastnosti, příp. omezení systému

Pro aplikaci na sběr a řešení požadavků zaměstnanců veřejné správy byly stanoveny požadavky na software vycházející mimo jiné z výše uvedených vybraných nástrojů na identifikaci požadavků.

*Funkční požadavky:*

- přístup do aplikace všem zaměstnancům
- zadávání požadavků či námětů
- zadávání návrhů řešení
- zadaný požadavek s navrženými řešeními bude tvořit jednu ucelenou skupinu
- evidovat, kdo požadavek, námět či řešení zadal
- nemožnost smazat jakýkoliv požadavek, námět či řešení
- umožnit hodnocení návrhů řešení

- zvolení odpovědného pracovníka za daný požadavek
- vyhodnocení nejvhodnějšího řešení odpovědným pracovníkem

Nefunkční požadavky budou především vycházet z prostředí, pro které je aplikace navrhovaná. Snaha bude o co nejjednodušší údržbu aplikace, proto bude zvolena webová aplikace, jejíž hlavní výhodou je fakt, že aplikace je umístěna na serveru a uživatelům stačí mít pouze internetový prohlížeč, který je již na každém počítači nainstalován. Ani v případě nasazení nové verze aplikace není potřeba u uživatelů nic měnit nebo zajišťovat kompatibilitu jednotlivých verzí.

*Nefunkční požadavky:*

- webová aplikace
- přívětivé prostředí
- intuitivní a snadné ovládání

Tyto požadavky musí být samozřejmě rozpracovány do většího detailu tak, aby uživatelé systému měli lepší představu o jednotlivých funkcích navrhované aplikace.

## **4.2 Nástin fungování aplikace a práv uživatelů**

Aplikace by měla zaměstnancům umožňovat jednoduše zadávat své požadavky na vyřešení nějakého problému či situace nebo podávat své náměty, a to pomocí názvu a stručného popisu požadavku (námětu), a navrhopvat na ně řešení, resp. psát názory. Řešení bude moci navrhnout zaměstnanec jak na své požadavky, tak samozřejmě i na požadavky kteréhokoliv z kolegů. Následně budou moci zaměstnanci jednotlivá řešení ohodnotit, aby se zjistilo, které řešení zaměstnancům vyhovuje nejvíce. Hodnotit navržené řešení budou moci všichni zaměstnanci, a to pomocí hodnot od 1 do 5. Čím vyšší hodnota, tím lepší hodnocení.

Ke každému požadavku se bude kromě názvu a popisu zadávat oddělení organizace, kterého se zadaný požadavek týká, aby bylo možné zjistit, kteří zaměstnanci budou požadavkem přímo dotčeni (budou ze stejného oddělení). Pokud se požadavek bude týkat všech oddělení, tzn. celé organizace, zaznamená se i tato skutečnost. Dále se při založení nového požadavku nepovinně uvede termín, do kterého je potřeba požadavek vyřešit.

Pro každé oddělení bude určen alespoň jeden odpovědný pracovník, který bude mít oprávnění vyhodnocovat jednotlivé náměty a navržená řešení požadavku – zpravidla to bude

vedoucí oddělení. Požadavek bude řešen tak, že odpovědný zaměstnanec – tzn. ten, který bude mít oprávnění k řešení požadavků týkajících se příslušného oddělení, nejprve požadavek uzavře. Uzavřením požadavku se znemožní uživatelům vkládat další řešení a hodnocení. Následně se vyhodnotí všechna navržená řešení s přihlédnutím k jejich hodnocením. Odpovědný zaměstnanec v aplikaci označí navržené řešení, resp. všechna řešení, která byla vybrána pro uvedení do praxe, a popř. napíše k zadanému požadavku svou poznámku o způsobu nasazení vybraného, resp. vybraných řešení. Tím bude vyhodnocení požadavku v aplikaci ukončeno.

*Proces evidence a zpracování požadavků tedy prochází těmito kroky:*

1. Zaevidování požadavku
2. Vkládání návrhů řešení a hodnocení navržených řešení
3. Uzavření požadavku
4. Výběr a označení nejvhodnějších řešení pro realizaci
5. Vložení poznámky o způsobu realizace vybraných řešení

Aby nedocházelo ke zbytečnému zneužívání aplikace, budou s ní moci pracovat jen zaregistrovaní uživatelé. Registrace bude umožněna a zpřístupněna všem zaměstnancům. Povinnými údaji budou uživatelské jméno, heslo, e-mailová adresa a oddělení, ve kterém zaměstnanec pracuje.

*Aplikace bude mít tři různá oprávnění:*

**Zaměstnanec** bude mít oprávnění se zaregistrovat, změnit si heslo, zadávat požadavky, zadávat řešení a hodnotit jednotlivá řešení.

**Vedoucí oddělení nebo zaměstnanec, který má oprávnění řešit požadavky příslušného oddělení** bude moci dělat stejné úkony jako zaměstnanec. Navíc bude moci uzavřít požadavek, napsat poznámku k požadavku a označit všechna řešení, která budou použita.

**Správce systému** bude mít stejné oprávnění jako zaměstnanec, navíc bude mít oprávnění opravovat a mazat zaměstnance či ukončovat aktuálnost zaměstnance (pokud zaměstnanec ukončí pracovní či služební poměr). Smazat zaměstnance bude moci pouze v případě, že zaměstnanec ještě nenavrhl žádný požadavek, řešení či žádné řešení nehodnotil. Pokud zaměstnanec zapomene heslo do aplikace, bude mu moci správce nechat vygenerovat heslo nové, které mu aplikace zašle na jeho e-mailovou adresu.

Správce systému bude také moci k jednotlivým zaměstnancům zadávat oprávnění, a to jak oprávnění správce systému, tak i oprávnění pro vyhodnocování navržených řešení požadavků týkajících se jednotlivých oddělení. Dále bude spravovat číselník oddělení, tzn. zadávat nová, opravovat stávající, ukončovat již neplatná oddělení či oddělení mazat (to v případě, že bylo zadáno omylem a toto oddělení není použito u žádného zaměstnance nebo požadavku).

### 4.3 Výchozí grafický návrh

Aplikace musí být navržena nejen z funkčního pohledu, ale aby byla uživatelsky ovladatelná, tak musí být vypracován i grafický návrh aplikace, resp. musí být navrženo, jak bude aplikace vypadat. Grafické uživatelské rozhraní aplikace bude vycházet z výše uvedeného návrhu.

Grafický návrh aplikace může také pomoci větší představě o fungování aplikace a může pomoci si uvědomit případné nedostatky návrhu aplikace (viz následující kapitola). Jelikož po zpracování grafického návrhu bude dále návrh aplikace podroben dalšímu ověřování, je předpoklad, že navržená grafická podoba nebude v této fázi konečná. Proto její konečný popis po případné úpravě bude uveden dále.

### 4.4 Hledání případných nedostatků v návrhu

Při navrhování informačního systému je potřeba si neustále klást otázky, např.: „Co se stane, když...?“ apod. Aplikace by tak měla umět řešit všechny možné situace, které nastanou, ne jen ty standardní, neboť uživatelé se mnohdy chovají nestandardně. Níže jsou uvedeny případy, které mohou nastat a k nim navržené chování systému:

- *Zaměstnanec by se mohl duplicitně zaregistrovat.*

Do aplikace bude při registraci doplněna kontrola duplicity uživatelského jména a e-mailové adresy. Tato kontrola bude aplikována i při opravě e-mailové adresy uživatele správcem systému.

- *Zaměstnanci se mohou zaregistrovat vícekrát a tak ovlivňovat hodnocení řešení.*

Aplikace může buď po registraci zaslat informační e-mail správci systému s uvedeným zaregistrovaným uživatelským jménem a e-mailovou adresou (správce systému pak může popř. uživatele smazat či ho nastavit jako neaktivní), nebo nebude dovoleno zadat jinou

než pracovní e-mailovou adresu (zaměstnanec nemůže mít dvě). Další možností je zakázat uživatelům se registrovat a uživatele by zakládal pouze správce systému.

Do aplikace budou zapracovány všechny varianty s tím, že možnost kontroly zadané e-mailové adresy, resp. domény se bude moci nastavit v konfiguračním souboru. Nastavit se bude moci i to, zda se uživatelé budou moci sami registrovat či nikoliv.

- *Zaměstnanec při registraci uvede jméno, popř. e-mailovou adresu jiného zaměstnance a bude tak v aplikaci vystupovat jako někdo jiný.*

Jedním řešením je, jak bylo uvedeno výše, zakázat zaměstnancům, aby se do aplikace sami registrovali. Pokud bude zaměstnancům registrace povolena, musí být zajištěno, aby zaměstnanec uvedl minimálně svou e-mailovou adresu. To lze zajistit tak, že uživatel nebude své heslo při registraci zadávat do formuláře, ale bude mu zasláno vygenerované heslo na uvedenou e-mailovou adresu. Bez tohoto hesla se zaměstnanec do aplikace nepřihlásí. Kontrolu, zda uvedená e-mailová adresa odpovídá zvolenému uživatelskému jménu, může provést správce systému na základě aplikací zaslání e-mailu informujícího o registraci nového uživatele.

Další možností by bylo automatické vygenerování uživatelského jména na základě uvedené e-mailové adresy. Vzhledem k možnosti zachování alespoň částečné anonymity a k možnosti vlastní volby uživatelského jména, pokud nebude zavádějící a pokud tomu bude dotyčná organizace nakloněna, nebude automatické generování uživatelského jména implementováno. Uživatelé budou uživatelské jméno zadávat buď podle určitých pravidel daných organizací, nebo si budou moci zvolit jakékoliv – to však neznamená, že budou vystupovat zcela anonymně, neboť v případě porušení pravidel či nevhodného chování je uživatel lehce identifikován na základě jeho e-mailové adresy.

- *Dojde k překlepu při zadání uživatelského jména*

Oprava uživatelského jména bude umožněna pouze správci systému, a to pouze pro případ opravy překlepu, popř. změny jména u zaměstnankyně z důvodu svatby.

- *Zaměstnanci chtějí psát požadavky a hodnotit navržená řešení anonymně, resp. bez registrace.*

Jak bylo uvedeno výše, zcela anonymní zadávání požadavků (tzn. bez registrace) nebude umožněno. Při anonymním hodnocení navržených řešení by mohlo docházet k vícenásobnému hodnocení jednoho řešení jedním uživatelem a tím k zásadnímu

ovlivnění hodnocení daného řešení, proto i hodnocení jednotlivých řešení bude umožněno pouze registrovaným uživatelům.

- *Jeden zaměstnanec bude chtít ovlivnit vybrané řešení tím, že bude stejné řešení hodnotit vícekrát.*

Při zadání hodnocení bude doplněna kontrola, zda zaměstnanec dané řešení již hodnotil. Pokud ho ještě nehodnotil, hodnocení se přidá jako nové. Pokud ho již hodnotil, hodnocení zaměstnance se k danému řešení pouze opraví, nepřibude tak nové, které by mohlo nekorektně ovlivnit přání zaměstnanců.

- *Hodnotit řešení mohou i zaměstnanci, kterých se požadavek bezprostředně netýká (nepracují na příslušném oddělení) a tak mohou zapříčinit špatné vyhodnocení řešení, neboť z hodnocení nebude možné určit, jak řešení požadavku hodnotili dotčení zaměstnanci. Tím může být vybráno řešení, které dotčeným zaměstnancům nebude nakonec vyhovovat.*

Jednou variantou je umožnit hodnotit navržené řešení pouze zaměstnancům z dotčeného oddělení. Toto řešení však neumožňuje zjistit, zda navržené hodnocení nějakým způsobem nevadí zaměstnancům z ostatních oddělení.

Druhou variantou je umožnit hodnotit řešení všem zaměstnancům s tím, že hodnocení jednotlivých řešení bude rozděleno na hodnocení pracovníků z oddělení, kterého se požadavek týká, a na hodnocení pracovníků z ostatních oddělení. Požadavky, které se budou týkat celé organizace, se rozlišovat nebudou. Může se tak rozlišit váha hodnocení navrženého řešení přímo dotčenými zaměstnanci a ostatních zaměstnanců. Zároveň bude zjištěno, zda navržené řešení nějakým způsobem nevadí zaměstnancům z ostatních oddělení. Implementováno bude toto řešení, neboť nabízí více možností.

- *Zaměstnanec může hodnotit své vlastní řešení?*

Hodnocení vlastníka řešení by mělo být započítáno do celkového hodnocení, proto to bude umožněno.

- *Řešení může být navrženo již v zadaném požadavku.*

Funkce pro hodnocení řešení a popř. pro výběr řešení pro následnou realizaci budou k dispozici i přímo u požadavku.

- *Jak je možné zjistit, zda ostatní zaměstnanci mají zájem o řešení vloženého požadavku, resp. zda nechtějí zůstat u stávajícího řešení a nechtějí nic měnit?*

Pokud zaměstnanci nechtějí nic měnit, mohou tuto skutečnost navrhnout jako řešení a hodnocením tohoto řešení vyjádřit svůj názor (dát mu kladné hodnocení). Další možností by bylo dát negativní hodnocení k ostatním navrženým řešením – z toho však nelze určit, zda zaměstnanec pouze odmítá navržená řešení či nechce vůbec nic na stávající situaci měnit. Proto je lepší návrh na zachování stávajícího řešení zapsat do aplikace jako návrh na řešení požadavku.

- *Došlo k nechtěnému uzavření požadavku nebo je potřeba po uzavření požadavku znovu umožnit uživatelům do něj vkládat řešení a hodnocení řešení.*

Uzavřený požadavek bude možné znovu otevřít a pokračovat v zadávání a hodnocení řešení. Je však nutné rozhodnout, zda pokud byl požadavek již uzavřen, byla vybrána řešení k realizaci a/nebo napsána poznámka k vybraným řešením a poté byl požadavek znovu otevřen, zda se mají označení vybraných řešení k realizaci a uvedená poznámka uživatelům zobrazit či nikoliv.

V případě, že došlo k nechtěnému uzavření požadavku, tak před jeho opětovným otevřením nedošlo k označení vybraných řešení ani k vložení poznámky. Tato situace se tedy nemusí řešit. Pokud by byl požadavek uzavřen záměrně a označeny jednotlivá řešení k realizaci, popř. vložena poznámka k vybraným řešením a pokud by bylo potřeba z nějakého důvodu pokračovat v hledání řešení a hodnocení, bylo by nejspíše vhodné, aby vybraná řešení nebyla zobrazena, aby uživatelé nebyli tímto výběrem ovlivněni. Na druhou stranu pokud by vybraná řešení byla vidět, mohla by se tato skutečnost využít k „dvoukolovému výběru“. Uživatelé by pak připomínkovali a hodnotili jen vybraná řešení, resp. rozhodovali by se mezi danými variantami.

Zachování označených řešení a vložené poznámky tak může nabídnout více možností, proto bude tato varianta vybrána pro realizaci. Pokud by uživatelé neměli vidět již jednou vybraná řešení a vloženou poznámku, musel by si odpovědný vedoucí řešení a poznámku poznamenat zvlášť mimo aplikaci a zrušit označení vybraného řešení a popř. smazat vloženou poznámku. Tento případ by se však měl stát ojediněle, neboť požadavek se uzavírá ve chvíli, kdy už uživatelé zadali své návrhy na řešení a ohodnotili zadané řešení a neměl by být důvod k opětovnému otvírání požadavků (kromě výše zmíněného).

- *Je potřeba napsat reakci, rozšířit řešení či se odkázat na řešení, které bylo zadané již dříve a může být „ztracené v historii“.*



Aby mohli uživatelé reagovat na jakýkoliv návrh řešení či k němu dopsat např. jeho rozšíření, opravu či komentář, bude umožněno vkládat návrhy řešení do tzv. stromu. Jednotlivá řešení se nebudou zapisovat jen pod sebe, ale jednotlivé „druhy“ řešení budou moci tvořit větve. Celé zobrazení a zadávání řešení tak dostane více formu přehledné diskuze.

- *Zaměstnanec se nedoví, že byl vložen nový požadavek.*

Zaměstnanec bude mít možnost si nechat zasílat informační e-mail, že byl zadán nový požadavek. Tuto volbu může uživatel kdykoliv změnit, aby popř. nedostával nechtěné velké množství e-mailů.

- *Zaměstnanec se nedoví, že bylo vloženo nové řešení požadavku nebo reakce na příspěvek*

Jelikož je předpokládáno, že vložených řešení, popř. reakcí na řešení bude větší množství a zasilání informačního e-mailu by bylo pro uživatele obtěžující, bude do seznamu požadavků vložen sloupec s datem a časem posledního vloženého příspěvku k danému požadavku. Uživatel si tak může lehce zkontrolovat, zda přibyly nějaké nové příspěvky. Navíc budou pro větší přehled vložených požadavků vytvořeny výběrové podmínky, pomocí níz si může uživatel omezit zobrazení seznamu požadavků. Jednotlivé sloupce pak bude možné řadit, lze si tak seznam srovnat dle data nejnovějšího příspěvku.

- *Jeden zaměstnanec bude mít oprávnění vyhodnocovat řešení požadavků, které se týkají různých oddělení, popř. bude vyhodnocovat řešení požadavků týkajících se kteréhokoliv z oddělení.*

U zaměstnance bude moci správce systému zadat oprávnění k jakémukoliv počtu oddělení. Jeden vedoucí tak může být zodpovědný za více oddělení, popř. za všechna oddělení.

## **4.5 Návrh možných rozšíření aplikace**

Aplikaci je možné samozřejmě rozšířit o další funkce jako např.:

- Umožnit uživateli nechat si u zadaného požadavku zvýraznit všechna vložená řešení zadaná za posledních X dní a Y hodin. Počet dní a hodin by si uživatel zvolil.
- Pro větší přehlednost umožnit vygenerovat všechna použitá řešení požadavků, které byly uzavřeny v rozmezí dní, které si uživatel zadá. Uživatel tak nebude muset procházet jednotlivé požadavky a hledat jejich vybraná řešení k realizaci.

- Vzhledem k tomu, že u každého požadavku jsou označena použitá řešení a kdo tato řešení navrhl, je možné vygenerovat statistiku „nejlepších řešitelů“, která může být vedoucím oddělení podkladem pro různá hodnocení zaměstnanců.
- Zajistit, aby se při zobrazení detailu požadavku zobrazilo v adresním řádku internetového prohlížeče i příslušné URL s id požadavku. To by v případě potřeby umožňovalo zaslat e-mailem odkaz na konkrétní požadavek.
- K zobrazenému uzavřenému požadavku vložit odkazy na vybraná řešení pro realizaci. Jednotlivá vybraná řešení pro realizaci by tak uživatel nemusel hledat ve stromové struktuře mezi ostatními řešeními, ale dalo by se na ně přesunout kliknutím na zmíněné odkazy uvedené v detailu požadavku.
- Aplikaci je možné naprogramovat s aplikačním programovým rozhraním a tím umožnit jednoduše naprogramovat další aplikace pro různá prostředí, např. zjednodušenou mobilní aplikaci či exporty do Microsoft Excelu.

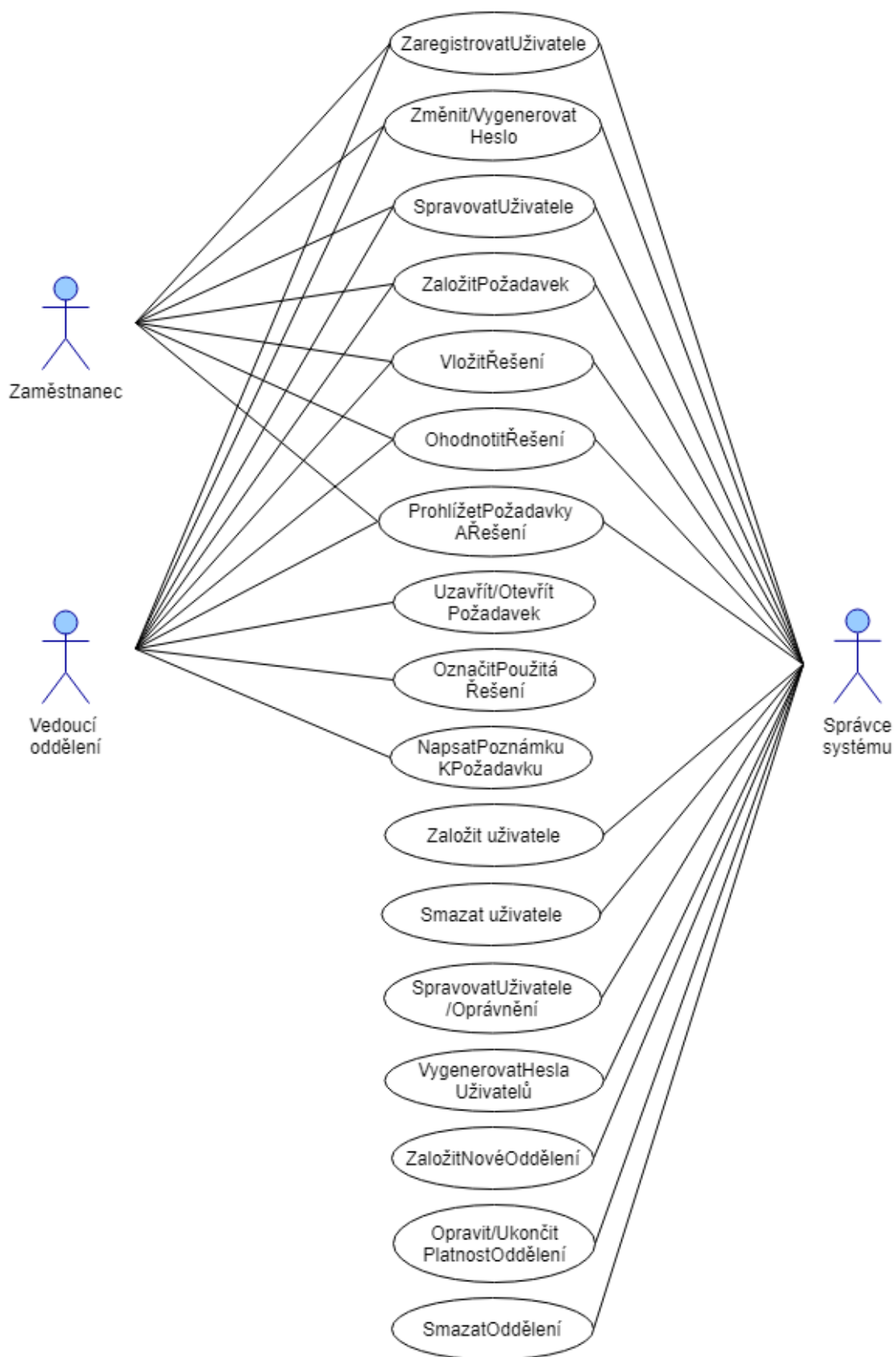
Pokud by byla aplikace naprogramována včetně navržených rozšíření, poskytovala by uživateli větší komfort při jejím používání, proto budou do aplikace implementována všechna uvedená rozšíření včetně aplikačního programového rozhraní, které umožní větší možnosti v případě potřebného rozšíření aplikace do různých prostředí.

## **4.6 Modelování funkčnosti aplikace – diagramy případů užití**

Výše uvedený návrh aplikace lze pro větší přehlednost vyjádřit pomocí diagramu případů užití (viz obrázek 1). Jedná se o způsob modelování, pomocí něhož se získávají a dokumentují funkční požadavky na informační systém a pomáhá těmto funkčním požadavkům lépe porozumět. [14]

Model případů užití obsahuje čtyři komponenty [2]:

- Hranice systému
- Aktéři
- Případy užití
- Relace



**Obrázek 1:** Diagram případů užití

Hranice systému vyznačuje hranici modelovaného systému, tj. co ještě je a co už není součástí systému. Aktéři jsou uživatelé systému a představují určitou roli, kterou aktér v systému vykonává (např. uživatel, správce systému, účetní apod.) Aktérem nemusí být pouze člověk, ale také např. systém. Jeden aktér (role) může představovat několik např. pracovníků, na druhou stranu jeden pracovník může vystupovat v několika rolích. [29]

Aktéři provádějí případy užití. Jeden případ užití může být vykonáván jedním nebo více aktéry a jeden aktér může vykonávat jeden nebo více případů užití. [14] Případ užití pak představuje jednu požadovanou funkčnost systému, resp. činnost, kterou bude aktér vykonávat. [29] Relace pak vyjadřují vztahy mezi aktéry a případy užití. [2]

## 4.7 Výsledný grafický návrh aplikace

Prvotní grafický návrh aplikace byl postupně přepracován vzhledem k výše navržených změnám a rozšířením. Jeho výsledná podoba je popsána níže.

### 4.7.1 Požadavky

Menu aplikace bude rozděleno do čtyř kategorií. První kategorií budou *Požadavky*, ve které bude položka *Nový* pro zadávání nového požadavku a položka *Seznam požadavků* pro zobrazení seznamu zadaných požadavků. Kliknutím na položku *Nový* se zobrazí formulář pro vyplnění názvu požadavku, jeho popisu, oddělení, kterého se požadavek týká, a termínu, do kterého by měl být požadavek vyřešen.

Na obrazovce *Seznam požadavků* budou vytvořeny formulářové pole pro výběrové podmínky – zobrazit se budou moci požadavky zadané přihlášeným uživatelem, všechny otevřené požadavky, uzavřené požadavky a všechny požadavky bez omezení, dále požadavky vybraného oddělení, požadavky zadané ve zvoleném rozmezí dní, uzavřené ve zvoleném rozmezí dní a požadavky, u kterých byl zadán příspěvek za posledních X zvolených dní, resp. Y hodin. Seznam požadavků bude obsahovat sloupce s datem vytvoření požadavku, názvem požadavku, názvem oddělení, kterého se požadavek týká, a datem a časem vložení nejnovějšího návrhu řešení k danému požadavku.

Detail požadavku se zobrazí výběrem ze seznamu požadavků a jeho vzhled bude záležet na stavu požadavku a na oprávnění přihlášeného uživatele. Navržená řešení požadavku budou tvořit stromovou strukturu. Nad detailem požadavku bude formulářové pole pro zadání počtu dní a hodin, které určují, které vložené návrhy řešení se mají zvýraznit – zvýrazní se ty, které

budou zadané za posledních zvolených X dní a Y hodin. Samotný zobrazený detail požadavku bude obsahovat tyto prvky:

*Text „Název“* – název požadavku.

*Text „Popis“* – stručný popis požadavku, resp. toho, co uživatel požaduje.

*Text „Hodnocení řešení:“* – zobrazení hodnocení daného řešení jednotlivými uživateli.

*Tlačítko „Zobrazit podrobnosti“* – zobrazí podrobnosti hodnocení, resp. rozložení jednotlivých hlasů. Pokud jsou podrobnosti zobrazeny, tlačítko se změní na „Skrýt podrobnosti“.

*Texty „V oddělení hodnotilo:“, „Mimo oddělení hodnotilo:“ a „Celkem hodnotilo:“* – jednotlivá hodnocení řešení budou rozdělena na tyto skupiny, za textem budou počty uživatelů, kteří dané řešení hodnotili.

*5 hvězdiček u každé skupiny* – zbarvením jednotlivých hvězdiček bude graficky znázorněno průměrné hodnocení daného řešení za každou skupinu a za hvězdičkami bude uveden číselný údaj průměrného hodnocení zaokrouhlený na dvě desetinná čísla.

*Indikátory průběhu* – budou zobrazeny po stisknutí tlačítka „Zobrazit podrobnosti“, v každé skupině bude 5 indikátorů pro jednotlivá hodnocení (1 – 5), resp. pro daný počet hvězdiček, indikátory budou zobrazovat procentuální zastoupení jednotlivých hlasů a za indikátorem průběhu bude uveden počet uživatelů, kteří řešení hodnotili daným počtem hvězdiček.

*Tlačítko „Navrhnout řešení“, „Reagovat“* – po jeho stisknutí se objeví pole pro zadání navrženého řešení na daný požadavek nebo reakce na vložené řešení.

*Text „Ohodnotit“ a 5 hvězdiček* – pomocí hvězdiček se budou hodnotit jednotlivá řešení, při najetí myši se zbarví jejich příslušný počet a při kliknutí na danou hvězdičku se provede hodnocení řešení.

*Tlačítko „Uzavřít požadavek“* – uzavře požadavek – již se nebudou se moci zadávat další návrhy řešení a navržená řešení hodnotit.

*Zaškrtnutí tlačítko s textem „Použít řešení“* – jeho zaškrtnutí slouží k označení daného řešení jako vybrané k následnému použití v praxi.

*Text „Poznámka k vybraným řešením:“* – nadpis pro níže uvedené textové pole nebo text.

*Textové pole pro vložení poznámky* – slouží pro vložení poznámky o způsobu realizace vybraných řešení, termínu realizace apod.

*Tlačítko „Uložit poznámku“* – slouží k uložení poznámky zadané v textovém poli.

*Text poznámky* – uložený text vložené poznámky – viz výše.

*Tlačítko „Upravit poznámku“* – slouží ke změně textu poznámky na textové pole pro vložení poznámky.

*Text „Uzavřeno“* – informace o uzavření požadavku – bude uvedeno i datum a čas uzavření požadavku.

*Text „Řešení“* – informace o označení daného navrženého řešení pro použití v praxi.

*Tlačítko „Otevřít požadavek“* – zruší uzavření požadavku – budou se moci opět zadávat další návrhy řešení a navržená řešení hodnotit.

*Text „Odkazy na vybraná řešení:“ a vygenerované odkazy* – vygenerované odkazy na řešení, která byla označena k použití v praxi – bude sloužit k rychlému přesunu na detail vybraného řešení.

|                                    |                                    |                                    |        |
|------------------------------------|------------------------------------|------------------------------------|--------|
| <b>Název</b>                       |                                    |                                    |        |
| Popis                              |                                    |                                    |        |
| Založeno dne                       | Uživatel                           | Týká se oddělení                   | Termín |
| <b>Hodnocení řešení:</b>           | Zobrazit/skrýt podrobnosti         |                                    |        |
| V oddělení hodnotilo:              | Mimo oddělení hodnotilo:           | Celkem hodnotilo:                  |        |
| „5x indikátor průběhu“             | „5x indikátor průběhu“             | „5x indikátor průběhu“             |        |
| „5 hvězdiček - průměrné hodnocení“ | „5 hvězdiček - průměrné hodnocení“ | „5 hvězdiček - průměrné hodnocení“ |        |
| <b>Ohodnotit</b>                   | „5 hvězdiček“                      | Navrhnout řešení                   |        |
| Uzavřít požadavek                  |                                    |                                    |        |

**Obrázek 2:** Grafický návrh – otevřený požadavek

Textové pole pro zadávání návrhu řešení nebo reakce na řešení a tlačítko „Uložit“ – zobrazí se pod požadavkem/řešením a bude sloužit k zadání navrženého řešení požadavku nebo reakce na vložený návrh řešení.

|  |                                    |                                    |        |
|--|------------------------------------|------------------------------------|--------|
| <b>Název</b>                           |                                    |                                    |        |
| Popis                                  |                                    |                                    |        |
| Založeno dne                           | Uživatel                           | Týká se oddělení                   | Termín |
| <b>Hodnocení řešení:</b>               |                                    |                                    |        |
| Zobrazit/skrýt podrobnosti             |                                    |                                    |        |
| V oddělení hodnotilo:                  | Mimo oddělení hodnotilo:           | Celkem hodnotilo:                  |        |
| „5x indikátor průběhu“                 | „5x indikátor průběhu“             | „5x indikátor průběhu“             |        |
| „5 hvězdiček - průměrné hodnocení“     | „5 hvězdiček - průměrné hodnocení“ | „5 hvězdiček - průměrné hodnocení“ |        |
| <input type="checkbox"/> Použít řešení | Řešení                             |                                    |        |
| Poznámka k vybraným řešením:           |                                    | Upravit poznámku                   |        |
| Poznámka                               |                                    |                                    |        |
| Uložit poznámku                        |                                    |                                    |        |
| Odkazy na vybraná řešení:              |                                    |                                    |        |
| Odkazy                                 |                                    |                                    |        |
| Otevřít požadavek                      |                                    | Uzavřeno dne                       |        |

**Obrázek 3:** Grafický návrh – uzavřený požadavek

U každého požadavku bude zobrazen řádek, ve kterém bude uvedeno *datum a čas založení požadavku*, *uživatelské jméno* uživatele, který požadavek založil, *název/zkratka oddělení*, kterého se požadavek týká, a *termín*, do kterého by měl být požadavek vyřešen. Každý návrh řešení bude mít podobný řádek s uvedeným *datem a časem vložení návrhu řešení* a *uživatelským jménem* uživatele, který dané řešení navrhl.

Jak bylo řečeno výše, jednotlivé prvky se budou zobrazovat dle stavu požadavku a oprávnění přihlášeného uživatele. Na obrázcích 2 a 3 jsou uvedeny návrhy rozložení jednotlivých prvků na stránce. Význam těchto prvků je popsán výše. Na obrázku 2 je grafický návrh pro zobrazení požadavku v otevřeném stavu, na obrázku 3 pak návrh pro zobrazení požadavku v uzavřeném stavu. Návrhy uvedených stránek jsou udělány tak, aby na nich bylo zobrazeno co nejvíce prvků – některé z prvků nemusí být vzhledem k oprávněním přihlášeného uživatele nakonec zobrazeny.

Pro lepší přehlednost, jaké prvky má systém zobrazovat při daném stavu požadavku a oprávnění přihlášeného uživatele a jaké ne, byla vytvořena tabulka s přehledem stavu požadavku (otevřený/uzavřený), rolí uživatele a jednotlivými prvky se zaškrtnutými políčky, které vyjadřují zobrazení příslušného prvku (viz tabulku 1). Pokud je zobrazení daného prvku závislé na splnění určité podmínky, je tato podmínka v tabulce uvedena. Některé prvky se budou vyskytovat i u jednotlivých řešení, která budou zobrazena pod požadavkem ve stromové struktuře. Tyto prvky jsou označeny ve sloupci *Řešení*.

#### **Uvedené podmínky v tabulce 1 znamenají:**

*Nová nebo oprava* – daný prvek se zobrazí, pokud není v poznámce ještě uložen žádný text nebo bylo stisknuto tlačítko pro opravu poznámky.

*Poznámka je zadána* – daný prvek se zobrazí, pokud je text poznámky již zadán.

*Počet vybraných řešení > 0* – pro zobrazení daného prvku je potřeba, aby počet označených řešení k realizaci byl větší než 0, resp., aby bylo označeno alespoň jedno řešení.

*Počet hodnocení > 0* – daný prvek se zobrazí, pokud již někdo příslušné řešení hodnotil.

*U vybraného řešení* – daný prvek se zobrazí jen u řešení, které je označeno k použití v praxi.

*Pro vybrané řešení* – daný prvek se zobrazí, pokud je označeno příslušné řešení k použití v praxi.

*Otevřený požadavek* – daný prvek se zobrazí, pokud je požadavek otevřený.

*Uzavřený požadavek a vedoucí* – daný prvek se zobrazí, pokud je požadavek uzavřený a zároveň je přihlášen uživatel s rolí vedoucího oddělení.



**Tabulka 1:** Zobrazení jednotlivých prvků aplikace

| Stav požadavku  | Otevřený požadavek         |                            | Uzavřený požadavek                                 |                            | Řešení   |
|---|----------------------------|----------------------------|--|----------------------------|--|
|   | Uživatel                   | Vedoucí                    | Uživatel   | Vedoucí                    |  |
| Název   | X                          | X                          | X  | X                          |  |
| Popis   | X                          | X                          | X  | X                          | X  |
| Tlačítko „Navrhnout řešení“, „Reagovat“ na příspěvek              | X                          | X                          |  |                            | otevřený požadavek                                     |
| Text „Ohodnotit“ a zobrazení hvězdiček                            | X                          | X                          |  |                            | otevřený požadavek                                     |
| Zobrazení hodnocení řešení  | počet hodnocení > 0        | počet hodnocení > 0        | počet hodnocení > 0                                | počet hodnocení > 0        | počet hodnocení > 0                                    |
| Informace o uzavření požadavku                                    |                            |                            | X  | X                          |  |
| Tlačítko na uzavření požadavku                                    |                            | X                          |  |                            |  |
| Tlačítko na opětovné otevření požadavku                           |                            |                            |  | X                          |  |
| Zaškrtnuté pole „Použít řešení“                                   |                            |                            |  | X                          | uzavřený požadavek a vedoucí                           |
| Označení jednotlivých řešení, která mají příznak vybraného řešení | u vybraného řešení         | u vybraného řešení         | u vybraného řešení                                 | u vybraného řešení         | u vybraného řešení                                     |
| Text „Odkazy na vybraná řešení“                                   | počet vybraných řešení > 0 | počet vybraných řešení > 0 | počet vybraných řešení > 0                         | počet vybraných řešení > 0 |  |
| Vložené odkazy vybraných řešení                                   | pro vybrané řešení         | pro vybrané řešení         | pro vybrané řešení                                 | pro vybrané řešení         |  |
| Text „Poznámka k vybraným řešením“                                | poznámka je zadána         | poznámka je zadána         | poznámka je zadána                                 | X                          |  |
| Textové pole pro vložení poznámky                                 |                            |                            |  | nová nebo oprava           |  |
| Tlačítko „Uložit poznámku“  |                            |                            |  | nová nebo oprava           |  |
| Text poznámky   | poznámka je zadána         | poznámka je zadána         | poznámka je zadána                                 | poznámka je zadána         |  |
| Tlačítko „Opravit poznámku“                                       |                            |                            |  | poznámka je zadána         |  |
| Oddělovat textu pod hodnocením příspěvku                          | X                          | X                          | poznámka je zadána nebo počet vybraných řešení > 0 | X                          | otevřený požadavek nebo (uzavřený požadavek a vedoucí) |

Výše uvedená tabulka řeší nejen zobrazení/skrytí jednotlivých prvků při prvotním zobrazení detailu požadavku, ale je možné z ní vyčíst i změnu zobrazení při různých aktivitách uživatele. Zobrazení jednotlivých prvků se samozřejmě změní při zvolení uzavření nebo znovu otevření požadavku, ale i vložení hodnocení navrženého řešení nebo vložení poznámky či označení řešení k použití v praxi. V těchto případech je také rozhodující, zda se jedná v případě vložení poznámky nebo hodnocení řešení o prvotní vložení dat nebo v případě označení použitého řešení o označení prvního řešení nebo v případě zrušení označení řešení, zda je ještě označeno nějaké jiné řešení či nikoliv. Aplikace také samozřejmě musí neustále rozlišovat jaká oprávnění má přihlášený uživatel.

### 4.7.2 Řešení

Další kategorií v menu aplikace bude *Řešení*, ve které bude položka *Seznam řešení* a *Nejlepší řešitelé*. V *Seznamu řešení* budou zobrazeny požadavky spolu s vybranými řešeními pro implementaci. Tyto seznamy se tedy budou týkat pouze požadavků, které jsou již uzavřené. Pro lepší přehlednost bude nad zobrazenými požadavky zobrazeno formulářové pole pro zadání rozmezí dní, ve kterých byly požadavky uzavřeny. Nebudou tedy zobrazovány všechny požadavky, ale jen ty z požadovaného období. Zobrazen bude vždy detail požadavku a pod ním pouze vybraná řešení včetně jejich hodnocení.

Po kliknutí na druhou položku *Nejlepší řešitelé* v kategorii *Řešení* bude zobrazena tabulka s počty navržených řešení, které byly označeny k implementaci, a to pro každého uživatele, který toto řešení navrhl, zvlášť. Tyto počty se opět týkají pouze uzavřených požadavků, proto budou nad tabulkou opět zobrazeny formulářové pole pro výběr rozmezí dní, ve kterém byly dané požadavky uzavřeny.

### 4.7.3 Správa

Další kategorie v menu bude *Správa*. Ta bude přístupná jen pro správce systému. Zde bude položka *Oddělení*. Po jejím zvolení se zobrazí tabulka s jednotlivými odděleními. Nad ní bude zatrhávací políčko pro volbu zobrazení i již neplatných oddělení a tlačítko *plus* pro zadání nového oddělení. Kliknutím na toto tlačítko se objeví formulář pro vyplnění názvu oddělení a tlačítka na uložení oddělení a zrušení akce. Viz obrázek 4.

Po najetí myši na řádek s názvem oddělení v tabulce se seznamem oddělení, se na tomto řádku objeví ikonky pro opravu názvu oddělení a pro smazání oddělení. Při kliknutí na ikonku opravy oddělení se opět objeví formulář, na kterém bude kromě názvu také zaškrťovací políčko s informací, zda je název oddělení stále platný či nikoliv. Při kliknutí na ikonku pro

smazání oddělení bude ještě uživateli zobrazen kontrolní dotaz na provedení akce. Viz obrázek 4.

| Nové oddělení / Oprava oddělení            |                                     |                                       |
|--|-------------------------------------|---------------------------------------|
| Název:                                     | <input type="text"/>                |                                       |
| Aktivní:                                   | <input type="checkbox"/>            |                                       |
| <input type="button" value="Uložit"/>      |                                     | <input type="button" value="Storno"/> |
| <b>Oddělení</b>                            |                                     |                                       |
| <input type="checkbox"/> včetně neplatných | <input type="button" value="plus"/> |                                       |
| Oddělení                                   |                                     |                                       |
| Název oddělení                             |                                     |                                       |
| Název oddělení                             | oprav                               | smaž                                  |
| Název oddělení                             |                                     |                                       |
| Název oddělení                             |                                     |                                       |

**Obrázek 4:** Grafický návrh - oddělení

Další položkou v kategorii *Správa* bude *Uživatelé*. Tato volba bude graficky řešena podobně jako položka Oddělení – bude zde zaškrťovací políčko pro zobrazení i již neplatných uživatelů a budou stejným způsobem zobrazeny ikonky pro zadání nového uživatele, jeho opravu a smazání. Lišit se samozřejmě budou jednotlivé formuláře pro zadání nového uživatele a pro jeho opravu, resp. změnu nastavení.

U nového uživatele budou na formuláři zobrazena políčka pro uživatelské jméno, e-mailovou adresu a oddělení (to bude formou rozbalovacího seznamu). Dále zde budou zaškrťovací políčka, kde se bude volit oprávnění uživatele. První zaškrťovací políčko bude označeno jako Správce systému, další budou zobrazena s názvy jednotlivých oddělení pro zadání oprávnění vyhodnocovat řešení požadavků z daného oddělení.

Formulář pro opravu uživatele bude shodný s předchozím, pouze bude zobrazeno navíc zaškrťovací políčko, zda je uživatel stále platný (aktivní) a u tlačítek na uložení nebo zrušení záznamu bude přidáno tlačítko na reset hesla uživatele.

#### 4.7.4 Přihlášení

Poslední kategorií bude *Přihlášení*. Pokud nebude uživatel přihlášen, bude tato kategorie obsahovat položku *Přihlásit se*. Výběrem této položky se zobrazí formulář pro přihlášení uživatele, tzn., že bude obsahovat formulářové pole pro zadání uživatelského jména a hesla a tlačítko *Přihlásit se*. Navíc bude formulář obsahovat informaci pro případ, že uživatel heslo zapomněl spolu s polem pro zadání e-mailové adresy a tlačítkem na zaslání nového hesla.

Další položkou bude *Registrace* – ta bude zobrazena pouze v tom případě, že je uživatelům povolena. Po jejím zvolení se zobrazí formulář s formulářovými políčky pro zadání uživatelského jména, e-mailové adresy, oddělení a zaškrtačím políčkem pro možnost zaslání informačního e-mailu v případě, že byl založen nový požadavek. Formulář samozřejmě bude obsahovat tlačítka na uložení a zrušení záznamu.

Pokud je uživatel již přihlášen, změní se název kategorie *Přihlášení* na uživatelské jméno uživatele a zobrazí se položky *Změna hesla*, *Nastavení* a *Odhlášení*. Formulář pro změnu hesla bude obsahovat formulářové pole pro zadání původního hesla, dvakrát nového hesla a tlačítka na uložení a zrušení záznamu. Formulář pro nastavení uživatele bude obsahovat pouze zaškrtačící políčko pro volbu zaslání informačního e-mailu v případě, že bude založen nový požadavek, a tlačítko na uložení změny. Při zvolení položky *Odhlášení* se zobrazí informace, že byl uživatel odhlášen.

#### 4.8 Scénáře případů užití

Aby bylo možné aplikaci naprogramovat, musí být jasné, v jakém sledu a jaké akce bude vykonávat systém, zaměstnanec, vedoucí oddělení či správce systému (tzn. jednotlivé role) při konkrétním užití systému. K tomu slouží scénáře případů užití.

Scénář je posloupnost kroků popisující interakci mezi aktérem a systémem. Je zapisován formou tabulky. Přitom není definován žádný standard, jak by měly tabulky přesně vypadat. Je možné např. použít tabulku s pořadovými čísly kroků, rolemi, které jednotlivé kroky vykonávají, a popisem vykonávané akce (viz tabulku 2). Tyto scénáře, resp. sady scénářů spojuje dohromady společný cíl. Tento cíl je výše zobrazen jako případ užití, u scénářů je pak uveden v názvu tabulky. [29]

**Tabulka 2:** Registrace uživatele

| <b>Krok</b> | <b>Role</b> | <b>Akce</b>   |
|-------------|-------------|---|
| 1           | Zaměstnanec | Z menu zvolí registraci uživatele.  |
| 2           | Systém      | Zobrazí formulář pro registraci uživatele a naplní rozbalovací seznam aktuálními odděleními.  |
| 3           | Zaměstnanec | Vyplní formulář (uživatelské jméno, e-mailovou adresu, oddělení a příznak zaslání e-mailu při založení nového požadavku).                           |
| 4           | Zaměstnanec | Buď registraci uloží (a), nebo ji zruší (b).  |
| 5a          | Systém      | Zkontroluje povinně vyplněná pole – v případě jejich nevyplnění na tuto skutečnost upozorní uživatele a data neuloží (konec).                       |
| 6a          | Systém      | Zkontroluje formát e-mailové adresy, popř. doménu – pokud je chybný/á, upozorní na to uživatele a data neuloží (konec).                             |
| 7a          | Systém      | Zkontroluje duplicitu uživatelského jména nebo e-mailové adresy – v případě duplicity na tuto skutečnost upozorní uživatele a data neuloží (konec). |
| 8a          | Systém      | Vygeneruje heslo a zašle ho na uvedenou e-mailovou adresu.  |
| 9a          | Systém      | Uloží data uživatele.   |
| 10a         | Systém      | Zjistí e-mailové adresy uživatelů s oprávněním správce systému a zašle jim e-mail s informací o nové registraci.                                    |
| 11a         | Systém      | Informuje uživatele o výsledku registrace a zaslání hesla na e-mailovou adresu.   |
| 12a         | Systém      | Přesměruje stránku na přihlášení uživatele.   |
| 5b          | Systém      | Smaže vyplněná pole formuláře.  |

Některé ze scénářů případů užití pro navrhovanou aplikaci jsou zpracovány v příloze A. Tyto scénáře vycházejí z výše uvedených návrhů aplikace. Grafický návrh aplikace určuje zobrazení jednotlivých formulářů a grafických prvků, stisknutí tlačítek uživatelem apod. Funkční návrh aplikace řeší princip, jakým bude aplikace fungovat.

Z výše uvedených návrhů je již zcela jasné, jak bude výsledná aplikace vypadat a fungovat. Následující kapitola se již bude věnovat datovému modelování, které bude řešit návrh aplikace z hlediska datové struktury.

## 4.9 Datové modelování

Datové modelování je proces, jehož výsledkem je datový model, který zobrazuje návrh datové struktury aplikace a vzájemné vztahy mezi jednotlivými prvky datového modelu.

Datové modelování je možné rozdělit do tří úrovní [52]:

- Konceptuální úroveň
- Technologická úroveň
- Implementační úroveň

V podstatě se jedná o tři fáze, resp. tři úrovně návrhu aplikace, přičemž každá další úroveň je více podrobná a rozvádí úroveň předchozí. Nejprve tedy vzniká obecný pohled na aplikaci, který postupně dostává konkrétnější obrysy.

### 4.9.1 Konceptuální model

Konceptuální model popisuje obsah systému, aniž by se zabýval jeho implementací nebo technologií. Pro vyjádření konceptuálního modelu neexistuje přesný návod nebo přesná forma. Základem tohoto modelu jsou entity, což jsou v podstatě objekty reálného světa. [28] Tyto entity se slučují do entitních množin, které představují např. zaměstnance, požadavek, oddělení úřadu apod. Entita je pak konkrétní výskyt patřící do entitní množiny – např. ekonomické oddělení. Každá entita má své atributy neboli vlastnosti. Vlastností oddělení je např. jeho název. Tato vlastnost může nabývat jen určitých hodnot označovaných jako doména atributu. [52]

Vztahy mezi entitami jsou charakterizovány parcialitou a kardinalitou. Parcialita vztahu určuje, zda entity mají povinnost se vztahu účastnit či nikoliv. Pokud je vztah nepovinný, jedná se o vztah parciální. Pokud je vztah povinný, jedná se o vztah totální. [52] Zaměstnanec musí mít pracovní pozici (totální vztah), ale nemusí mít průkaz sociálního pracovníka (parciální vztah).

Kardinalita vztahu vyjadřuje počet výskytů entit, které se mohou účastnit jednotlivých vztahů mezi entitami. Může být vyjádřena vazbou [30]:

- 1:1 – na obou stranách vazby se vyskytuje pouze jedna entita (např. zaměstnanec může mít pouze jeden zaměstnanecký průkaz)
- 1:N – jedna entita na jedné straně může být ve vztahu s více entitami na straně druhé (např. jedno oddělení může mít více zaměstnanců)

- M:N – na obou stranách se vztahu může účastnit více entit (např. zaměstnanec je zapojen do více projektů a jeden projekt může řešit více zaměstnanců)

Dále rozlišujeme stupeň vztahu, který vyjadřuje počet množin entit, které jsou součástí vztahů [37]. Jestliže se vztahu účastní pouze jedna entitní množina – je tedy ve vztahu sama se sebou, jedná se o unární vztah. Pokud se vztahu účastní dvě množiny entit, jedná se o binární vztah, v případě tří entit o ternární vztah atd.

Jak bylo zmíněno výše, pro zobrazení tohoto modelu neexistuje přesná forma. Nejčastěji se však používá ER diagram neboli diagram entit a vztahů mezi těmito entitami. ER diagramů je celá řada a používají se různé formy zobrazení. V této práci bude využita notace Crow's Foot [46]. V této notaci je kardinalita vyjádřena pomocí svislé čáry (vstup jedné entity do vztahu) nebo tří spojených čar (vstup více entit). Parcialita je znázorněna svislou čarou pro totální vztah a prázdným kolečkem pro parciální. [11] V tabulce 3 je uveden přehled notace Crow's Foot. Entitní množina je vyjádřena obdélníkem s uvedením svého jména v jeho horní části. [28]

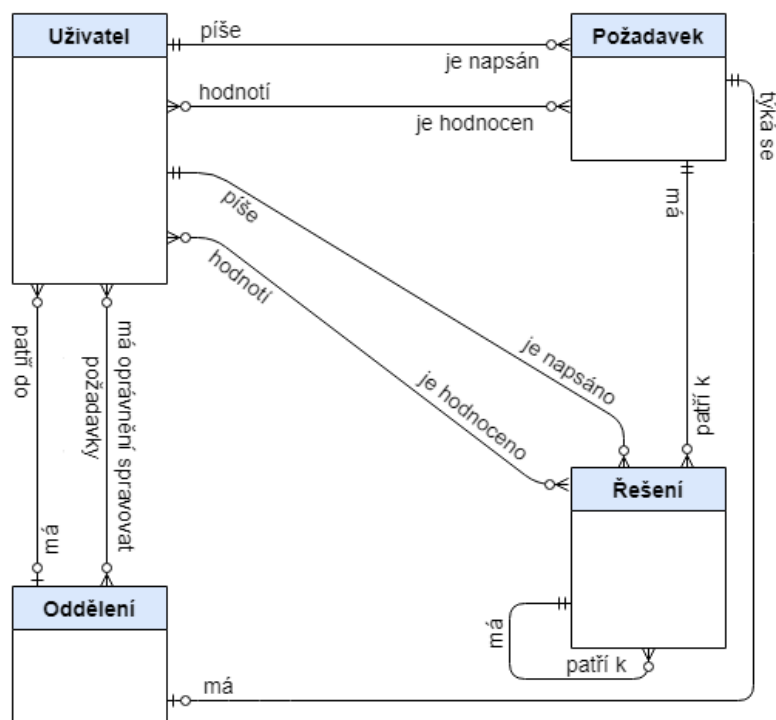
**Tabulka 3:** Symboly Crow's Foot notace

| Symbol | Význam  |
|--------|---|
| ⊕—     | Vztahu se povinně účastní pouze jedna entita      |
| +○—    | Vztahu se nepovinně účastní pouze jedna entita    |
| ⋈—     | Vztahu se povinně účastní jedna nebo více entit   |
| ⋈○—    | Vztahu se nepovinně účastní jedna nebo více entit |

Konceptuální model pro návrh aplikace pro sběr a řešení požadavků bude obsahovat entitní množinu Zaměstnanec, resp. Uživatel, který bude s aplikací pracovat a vkládat do ní své požadavky a řešení. Z toho vyplývá, že dalšími entitními množinami budou Požadavek a Řešení s příslušnými vazbami na entitní množinu Uživatel (Uživatel píše Požadavek/Řešení). Další vazbou, resp. vztahem bude hodnocení Řešení Uživatelem. Řešení může být vloženo buď samostatně, nebo je součástí Požadavku, proto může Uživatel hodnotit jak Řešení, tak i Požadavek.

U hodnocení je potřeba rozlišit, zda uživatel hodnotí řešení požadavku týkající se stejného oddělení, ve kterém pracuje, či nikoliv. K tomu je potřeba založit entitní množinu Oddělení, která bude obsahovat číselník oddělení. K této entitní množině se tedy připojí jak entitní množina Uživatel (Uživatel patří do Oddělení), tak entitní množina Požadavek (Požadavek se týká Oddělení). Číselník oddělení bude spravovat správce systému. Správce systému bude vlastností uživatele.

Dále je nutné zpracovat oprávnění uživatele vyhodnocovat požadavky přiděleného oddělení. Vznikne tedy další vazba mezi entitní množinou Uživatel a entitní množinou Oddělení (Uživatel má oprávnění spravovat požadavky daného/daných Oddělení). Tito oprávnění uživatelé budou uzavírat požadavky, psát k nim poznámky a označovat řešení, která budou vybrána pro realizaci. Jelikož se počítá s tím, že oprávněnými uživateli budou pouze vedoucí oddělení, popř. jejich zástupci, není nutné evidovat, kdo tyto úkony provedl. Uzavření požadavku, poznámka a označení vybraného řešení pro implementaci budou tedy pouze vlastnosti entitní množiny Požadavek, resp. Řešení.

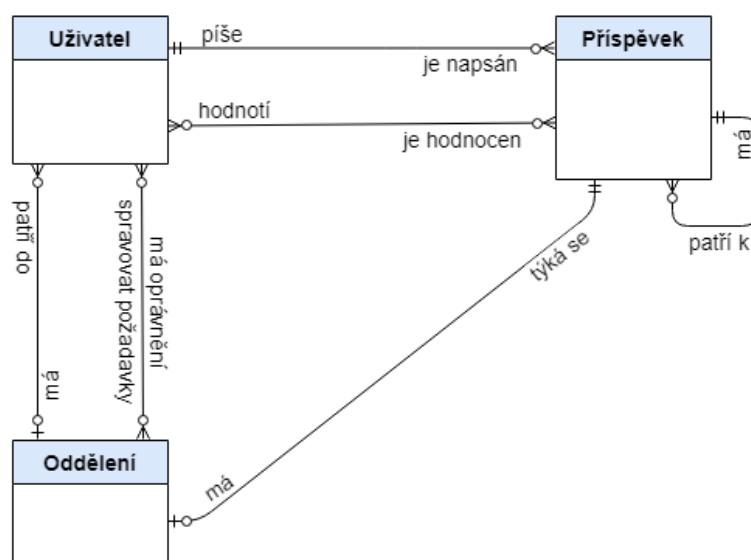


**Obrázek 5:** Konceptuální model



Jelikož celá aplikace bude fungovat na principu diskuzního fóra, bude entitní množina Řešení ve vztahu sama se sebou, aby uživatelé mohli reagovat a tedy se odkazovat na jakékoliv zadané řešení. Celý konceptuální model je vidět na obrázku 5.

Z obrázku 5 je patrné, že z entitní množiny Uživatel do entitních množin Požadavek a Řešení vedou stejné vazby. Entitní množiny Požadavek a Řešení budou mít i některé shodné atributy – minimálně se bude registrovat text požadavku/řešení, datum a čas založení požadavku/řešení, uživatel, který požadavek/řešení zadal a příznak použitého řešení. Dalo by se tedy uvažovat o zjednodušení celého modelu, a to sloučením entitní množiny Požadavek a Řešení do jedné, např. Příspěvek (viz obrázek 6).



**Obrázek 6:** Upravený konceptuální model

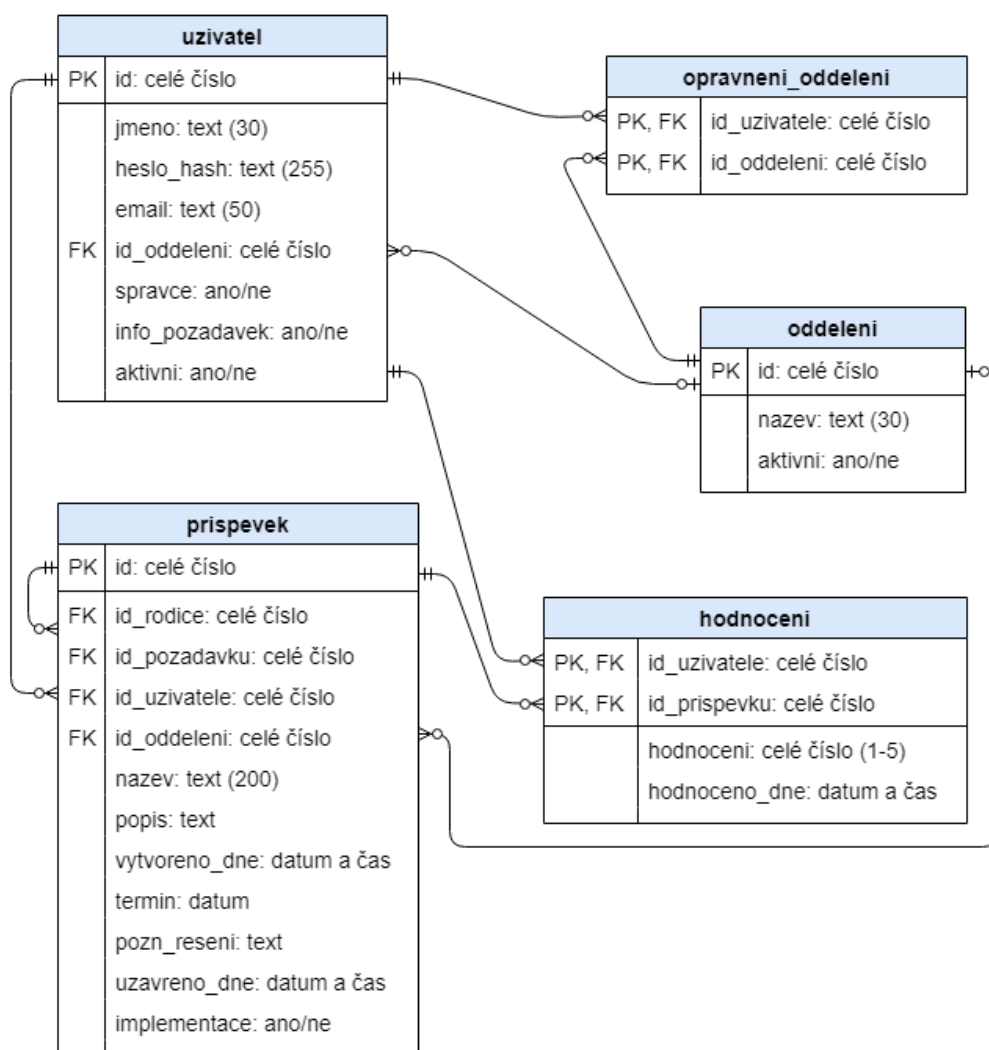
Sloučením entitní množiny Požadavek a Řešení do jedné množiny může navíc dojít ke zjednodušení jednotlivých dotazů do databáze. Na druhou stranu, pokud bude potřeba se dotázat na seznam požadavků, budou se prohledávat všechny entity Příspěvků místo pouze entit Požadavků. To by však nemělo nějak zásadně časově ovlivnit příslušný dotaz do databáze. Výhodnější tedy bude sloučení uvedených entitních množin.

#### 4.9.2 Technologický model

Další úroveň datového modelování je technologický model. Technologický model je též nazýván logickým nebo databázovým modelem. [52] Tato úroveň modelu zahrnuje technologické řešení aplikace. Model vychází z konceptuálního modelu, ale již zohledňuje

způsob uložení dat, tzn., zda budou data uložena např. v objektové nebo relační databázi. Relační databáze bude použita i pro aplikaci na sběr požadavků. Tento model je však ještě nezávislý na implementaci.

Na této úrovni se řeší vzniklé vztahy s kardinalitou M:N, které se musí převést na vztahy s kardinalitou 1:N. Tím vzniknou nové entitní množiny. První novou entitní množinou je Hodnocení, která řeší vazbu M:N mezi entitními množinami Uživatel a Příspěvek. Druhou nově vzniklou entitní skupinou je Oprávnění\_oddělení, která řeší vazbu M:N mezi entitními množinami Uživatel a Oddělení (viz obrázky 6 a 7).



**Obrázek 7:** Technologický model

Každá entita musí být jednoznačně identifikovatelná. Entity se identifikují pomocí primárního klíče, který je v rámci entity jednoznačný. Může být použit buď jednoduchý klíč,

nebo klíč složený z více atributů. [28] Nejčastěji se k tomuto účelu používá jednoduchý klíč ve formě identifikačního čísla (id), jeho hodnotu systém postupně přiděluje tak, že každá nově vzniklá entita má o jednotku vyšší identifikační číslo než ta předchozí. Složený klíč ze dvou atributů je použit u entitních množin `Oprávnění_oddělení` a `Hodnocení`.

Pomocí primárního klíče jsou realizovány jednoznačné vztahy mezi entitami, a to tak, že primární klíč jedné entity se zároveň vyskytuje v entitě druhé – zde vystupuje jako cizí klíč. Tím je jedna entita jednoznačně přiřazena k druhé. [55] V modelu (viz obrázek 7) jsou viditelné jak jednoduché primární (PK), tak cizí (FK) klíče, pomocí níž jsou vytvářeny jednotlivé vazby mezi entitami.

Entitní množiny jsou zobrazeny se všemi jejími atributy. Jak bylo uvedeno výše, atributy jsou vlastnosti entitní množiny. Pokud jsou uváděny datové typy, jsou zcela nezávislé na implementaci, tzn. na tom, zda bude použita databáze Oracle, MySQL či jiná. Technologický model navržené aplikace je uveden na obrázku 7.

### 4.9.3 Implementační model

Implementační model je také nazýván fyzickým modelem. Tento model je zcela závislý na implementaci, tj. na konkrétní použité databázi a programovacím jazyku. V této fázi již probíhá realizace datové struktury. Jednotlivé entitní množiny se změny na tabulky, které se nazývají relace. [52] Atributy tvoří sloupce tabulky a řádky tabulky pak obsahují jednotlivé výskyty entit. Viz tabulku 4 s ukázkou relace `oddeleni`.

**Tabulka 4:** Ukázka relace `oddeleni`

| id | nazev       | aktivni |
|----|-------------|---------|
| 1  | Sekretariát | 1       |
| 2  | SSP         | 1       |
| 3  | OZ          | 1       |

Pro návrh aplikace bude použit databázový server MariaDB verze 10.1.10 [39] a jazyk SQL [49]. Použité datové typy [8] jsou uvedeny v tabulce 5. Pro hodnotu ano/ne nepoužívá databázový server MariaDB obvyklý datový typ `boolean`, ale datový typ `tinyint` s délkou 1 byte. Hodnota 0 je považována za hodnotu `false` (ne), ostatní nenulové hodnoty za hodnotu `true` (ano). [3]

**Tabulka 5:** Použité datové typy

| Datový typ   | Význam      | Rozsah / formát   |
|--------------|-------------|---|
| INT          | celé číslo  | -2 147 483 648 až 2 147 483 647 ( $-2^{31}$ až $2^{31}-1$ ) |
| TINYINT      | celé číslo  | -128 až 127   |
| TINYINT      | ano/ne      | -128 až 127   |
| VARCHAR(max) | text(max)   | maximální počet znaků uveden v závorce                      |
| TEXT         | text        | maximální počet znaků je 65 535                             |
| DATE         | datum       | ve formátu YYYY-MM-DD                                       |
| DATETIME     | datum a čas | ve formátu YYYY-MM-DD HH:MM:SS                              |

Implementací se zabývá i následující kapitola, ve které jsou uvedeny podrobnější informace k databázi a dále se již věnuje samotným naprogramování aplikace, resp. aplikací – aplikačního programového rozhraní a grafického uživatelského rozhraní.

## 5 Implementace

### 5.1 Databáze

Databáze je propracovaný systém pro ukládání dat a jejich následné zpracování. [9]. Také lze říci, že databáze je organizovanou sbírkou dat. [5] Lze si jí představit např. jako kartotéku, ve které jsou data uložena podle určitého systému. Tento systém umožňuje pracovat s daty velmi efektivně. [55] Jak bylo uvedeno výše, jako databázový server bude použit MariaDB [39], který zajišťuje bezpečnou práci s daty, přístup k datům více uživatelům najednou, efektivní vyhledávání a řazení dat atd.

Struktura dat bude vycházet z uvedeného implementačního modelu. Jednotlivé entitní množiny budou tedy ve formě tabulek [44], které budou mimo dalších sloupců obsahovat primární a cizí klíče. Sloupce s jednoduchými primárními klíči jsou pojmenovány `id` a jsou datového typu `INT` s definicí atributu `AUTO_INCREMENT`, který zajišťuje automatické přidělování celého čísla s hodnotou vždy o jednotku větší, než bylo naposledy přidělené číslo. [4] Cizí klíče pak v názvu většinou obsahují kromě „`id`“ i název dotčené tabulky.

Manipulace s daty bude umožněna pomocí jazyka SQL (Structured Query Language) [49], což je standardizovaný strukturovaný dotazovací jazyk určený pro relační databáze [44]. Pomocí něj lze mimo jiné zakládat nové databáze, vytvářet databázové tabulky, získávat data z databáze, ukládat do databáze nová data, upravovat je či mazat. Pro manipulaci s daty se používají tyto základní příkazy [55], [4]:

`SELECT` – slouží k vybírání záznamů, resp. řádků a sloupců z databáze

`WHERE` – specifikuje podmínku, které konkrétní záznamy budou vybírány

`ORDER BY` – umožňuje řazení záznamů (řádku) podle daného sloupce

`JOIN` – slouží ke spojování tabulek v jednu množinu záznamů

`INSERT` – slouží k vložení nového záznamu, resp. řádku do tabulky

`UPDATE` – umožňuje změnu záznamu

`DELETE` – slouží k odstranění záznamů, resp. řádků z tabulky

## 5.2 Aplikační programové rozhraní

Požadovaná webová aplikace bude naprogramována včetně aplikačního programového rozhraní (API). Toto rozhraní umožňuje vytvářet aplikace pro různá prostředí, aniž by se znovu musely programovat základní funkce programu – ty obstarává API. Těmito funkcemi jsou správa databáze, tzn. získávání, uložení, oprava a smazání záznamu z databáze včetně kontroly vstupních dat.

API bude dále zajišťovat bezpečný přístup k databázi, a to tím, že bude vyžadovat přihlášení uživatele a kontrolovat jeho přístupová práva pro požadovanou operaci s daty. Pokud uživatel nebude přihlášen nebo nebude mít požadovaná oprávnění, nebude požadovaná operace s daty provedena. API bude také definovat, které operace mohou být vykonány bez přihlášení uživatele.

K naprogramování API bude použit programovací jazyk PHP verze 5.6 [43], což je skriptovací jazyk na straně serveru navržený speciálně pro web. Jedná se o velmi rychlý jazyk, který umí obsluhovat od malého kontaktního formuláře až po webovou stránku typu Facebook. [54] PHP pracuje na straně serveru, to znamená, že se na serveru, kde jsou uloženy zdrojové soubory webových stránek, spustí PHP script a prohlížeči, který je na straně klienta, pošle pouze výsledek. [42]

API bude využívat architekturu REST (Representational State Transfer) [47], která zajišťuje jednotný a snadný přístup ke zdrojům, kterými mohou být data nebo stavy aplikace. Zdroje jsou definovány URI a pro přístup k nim, resp. k požadovaným operacím s nimi se používají dané metody. REST je založeno na CRUD operacích, tzn. pořízení dat (Create), získání dat (Retrieve), změně dat (Update) a smazání dat (Delete). Těmito operacím odpovídají čtyři metody, které jsou zasílány spolu s URI v HTTP požadavku. [36]

*Těmito metodami jsou:*

GET – slouží pro získávání dat

– např. pro získání seznamu všech zadaných požadavků se zašle HTTP s URL „/požadavky“

– pro získání dat konkrétního požadavku je nutno použít identifikátor požadavku, tzn. v případě, že je např. požadováno vypsání dat požadavku s `id 3`, zadá se URL „/požadavek/3“

POST – slouží pro vytvoření dat  
– spolu s URL (např. pro založení nového požadavku bude URL „/požadavek“) je potřeba zaslat i data, která mají být vytvořena

PUT – metoda pro změnu dat  
– v URL musí být uveden parametr (např. id požadavku), který definuje, která data mají být změněna (např. „/požadavek/3“) a zaslána nová data pro opravu stávajících

DELETE – slouží ke smazání dat  
– v URL musí být opět uveden parametr, který definuje, která data mají být smazána, např. pro smazání oddělení s id 4, je zasláno URL „/oddeleni/4“

Spolu s výstupními daty zasílá API i stavový kód, který informuje o stavu odpovědi, resp. jeho výsledku. REST poskytuje mnoho různých stavových kódů [19]. V tabulce 6 jsou vypsány nejběžnější, které se budou vyskytovat i v API navržené aplikace.

**Tabulka 6:** Stavové kódy architektury REST

| Číselný kód | Význam  |
|-------------|---|
| 200         | OK – HTTP požadavek byl proveden                      |
| 201         | Vytvořeno – nejčastěji odpověď na metodu POST         |
| 400         | Zaslán chybný požadavek                               |
| 401         | Neautorizováno – nezasláno přihlašovací jméno a heslo |
| 409         | Konflikt  |
| 500         | Interní chyba serveru                                 |

Získaná výstupní data mohou být v různém formátu, např. XML nebo JSON. Pro účely aplikace byl vybrán formát JSON (JavaScript Object Notation), neboť to je rozšířený, dobře čitelný textový formát, který slouží pro výměnu dat. [27] Tento formát je zcela nezávislý na programovacím jazyku, i když je založen na JavaScriptu. [5] Implementaci JSONu najdeme i v PHP. [17] Hlavní výhodou JSONu je v jeho snadném zpracování. [23] Přehlednost výstupního formátu je možné vidět v případě jedné položky na obrázku 8, v případě více položek na obrázku 11.

API je naprogramováno pomocí frameworku Slim 3 [35], což je mikro framework, který je používán PHP vývojáři pro vývoj API a služeb vyhovujících pravidlům REST. [56] Slim je využíván jako rychlý směrovač URL, který reaguje na HTTP požadavek, tzn. na URI a mimo jiné na výše uvedené metody GET, POST, PUT a DELETE. Framework Slim také podporuje zasílání potřebných parametrů. Na základě HTTP požadavku vykoná příslušný kód a vrátí HTTP odpověď. [35]

HTTP požadavky mohou být provedeny bez autorizace nebo s autorizací. Pokud je autorizace vyžadována, musí být spolu s požadavkem zasláno uživatelské jméno a heslo, jinak nebude požadavek proveden. Tímto je aplikační programové rozhraní chráněno před neoprávněnými zásahy, resp. před zásahy do databáze neoprávněnými uživateli. Tento bezpečný přístup k databázi bude zajištěn pomocí knihovny slim-basic-auth [16].

Databáze bude dále chráněna před SQL Injection, což je útok, při kterém útočník využije předání dat ke změně SQL dotazu a tím může např. smazat databázovou tabulku nebo celou databázi. Obrana proti tomuto útoku bude zajištěna pomocí parametrizovaných dotazů, a to tak, že v SQL dotazech se místo hodnot zadají zástupné znaky. Hodnoty a dotaz do databáze jsou pak předávány úplně odděleně a databáze si požadované hodnoty do dotazu vloží sama tak, aby to bylo bezpečné. [31]

Dále bude API zajišťovat znemožnění spuštění javascriptového kódu, který může být uložen do databáze. Spuštěním tohoto kódu by mohl být útočníkem pozměněn HTML dokument a tím i vzhled stránek nebo by stránky přestaly úplně fungovat. Toto bezpečnostní riziko bude řešeno úpravou dat pomocí funkce `htmlspecialchars`, a to před jejich výpisem. Tato funkce nahrazuje pomocí escapování všechny speciální znaky a znemožňuje tak útočníkovi spustit javascriptový kód.

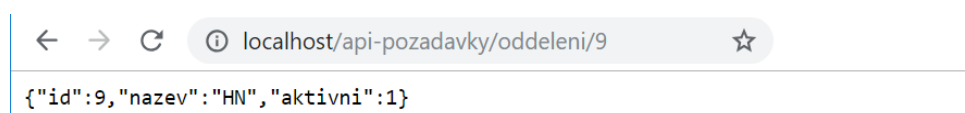
Seznam všech URL a jednotlivých metod, které poskytuje API v rámci aplikace na zadávání a řešení požadavků je uveden v příloze B, v tabulce 25. Pokud URL obsahuje datum, je ve formátu YYYY-MM-DD. V tabulce jsou uvedeny i povinné parametry, které je třeba zadat – v případě metod POST pro vložení záznamu a PUT pro opravu, resp. změnu záznamu. V závorce jsou pak uvedeny parametry nepovinné.

V dalším sloupci je uvedeno oprávnění, které musí uživatel mít, resp. musí být přihlášen uživatel s uvedeným oprávněním, aby požadovaný HTTP požadavek byl vykonán. Pokud je v tabulce uvedeno „správce“, musí mít uživatel oprávnění správce systému. Pokud je uvedeno „oddělení“, musí mít uživatel oprávnění spravovat požadavky oddělení, kterého se zadaný



požadavek týká. Pokud URL obsahuje „/verejne/“, k vykonání HTTP požadavku není potřeba, aby byl uživatel přihlášen. Pro vykonání ostatních HTTP požadavků je přihlášení uživatele vyžadováno.

HTTP požadavky je možné zadávat buď pomocí internetového prohlížeče – to v případě použití metody GET (viz obrázek 8), nebo pomocí REST klienta, kterým může být např. v Google Chrome rozšíření Postman [45]. Rest klient umožňuje zaslání jakékoliv metody HTTP požadavku spolu s požadovanými parametry. Vracená data jsou pak přehledně vypsána ve formátu JSON (či jiném zvoleném formátu) spolu s uvedeným stavovým kódem (viz obrázek 11).



**Obrázek 8:** API – ukázka zobrazení dat v internetovém prohlížeči

V případě použití internetového prohlížeče pro získání dat, která jsou přístupná jen pro přihlášené uživatele nebo uživatele s požadovaným oprávněním, bude vyžadováno přihlášení uživatele pomocí přihlašovacího okna (viz obrázek 9). Pokud bude využit REST klient, je způsob přihlášení uživatele uveden níže (viz obrázek 10).

A screenshot of a login form titled 'Přihlaste se'. The form is located on the page 'http://localhost'. It contains two input fields: 'Uživatelské jméno' (Username) and 'Heslo' (Password). Below the input fields are two buttons: a blue 'Přihlaste se' button and a white 'Zrušit' button.

**Obrázek 9:** API – autorizace v internetovém prohlížeči

API bude jednak komunikovat s databází a zajišťovat všechny výše uvedené funkce, tak také i s grafickým uživatelským prostředím, které uživatelům zajistí přívětivé prostředí pro práci s daty. Díky tomu, že grafické uživatelské prostředí bude komunikovat pouze s API, tak bude zcela oddělené od samotné databáze a nebude se muset zabývat funkcemi správy dat v databázi, jak bude patrné z následující kapitoly.

## 5.3 Grafické uživatelské rozhraní

Grafické uživatelské rozhraní aplikace bude uděláno formou webových stránek. Jejich vzhled bude vytvořen pomocí HTML5 [18]. HTML (HyperText Markup Language) je značkovací jazyk sloužící k vytváření struktury dokumentů, resp. webových stránek [6], a to pomocí příznaků zadaných v úhlových závorkách [5]. HTML verze 5 obsahuje i CSS3, které je v aplikaci použito. CSS (Cascading Style Sheets) neboli kaskádové styly slouží k úpravě vizuálního vzhledu webových stránek [6] tím, že je pomocí něj definován vzhled jednotlivých elementů na webových stránkách [24], [5].

K naprogramování funkčnosti webových stránek bude použit programovací jazyk PHP [43], knihovny jQuery, jQuery UI, Stupid-Table-Plugin a technologie AJAX. JQuery [25] je javascriptová knihovna, která usnadňuje práci s javascriptem. Její předností je jednoduchost, rychlost a dobrá čitelnost, a to vše s použitím minimálního množství kódu. [21] V aplikaci bude využita ke snadné manipulaci s obsahem webu, resp. s HTML prvky. Dále bude použit jQuery UI, který mimo jiné umožňuje implementaci různých efektů pro webové stránky [26], [20]. Pomocí jQuery UI budou v aplikaci vytvořena vyskakovací dialogová okna. Knihovna Stupid-Table-Plugin [15] bude použita k řazení dat v tabulkách. Tato knihovna bude v aplikaci umožňovat seřadit data v tabulce podle uživatelem zvoleného sloupce.

Použitá technologie AJAX (Asynchronous Javascript And XML) umožňuje asynchronní načítání částí stránek a odesílání dat formulářů bez obnovení celé webové stránky. [22] Umožňuje tedy měnit obsah stránek bez nutnosti jejich znovunačtení. [33], [54] Asynchronnost v AJAXu pak spočívá ve schopnosti javascriptu zavolat na serveru nějaký skript a nečekat na odpověď, ale místo toho pokračovat dál v provádění kódu. V momentě, kdy přijde odpověď na požadavek, se provádění hlavního kódu zastaví a provede se funkce, která se má provést v případě, že ze serveru přijde odpověď. AJAX je tedy způsob, kterým se vyměňují data mezi aplikací a např. databází. [34]

V aplikaci bude technologie AJAX používána k výměně dat mezi GUI a API. Cesta k API bude nastavena v souboru *config.js* v kořenovém adresáři. „Hezké URL adresy“ budou v aplikaci vytvořeny pomocí frameworku Slim 3. Aplikace bude optimalizovaná pro prohlížeč Google Chrome verze 69.0.

## 6 Výsledné aplikace

Tato kapitola může sloužit jako návody na použití aplikací (API a GUI). Takto bude využita pro vybrané zaměstnance vybrané organizace veřejné správy, kteří budou mimo jiné tímto způsobem seznámeni s grafickým uživatelským prostředím. Následně budou mít možnost si aplikaci vyzkoušet.

### 6.1 Aplikační programové rozhraní

Aby bylo aplikační programové rozhraní funkční, je nejprve nutné nastavit konfigurační údaje pro připojení k databázi. Ty se nastavují v souboru *config.php*. V tomto souboru si lze také zvolit, zda se bude kontrolovat doména e-mailových adres jednotlivých uživatelů při jejich registraci a při zadání změny e-mailové adresy uživatelů. V případě požadované kontroly e-mailové adresy se do zmíněného souboru do proměnné *DOMENA* zadá název domény e-mailových adres (např. „uradprace.cz“). Pokud kontrola není požadována, nebude proměnná obsahovat žádné znaky (viz zdrojový kód 1).

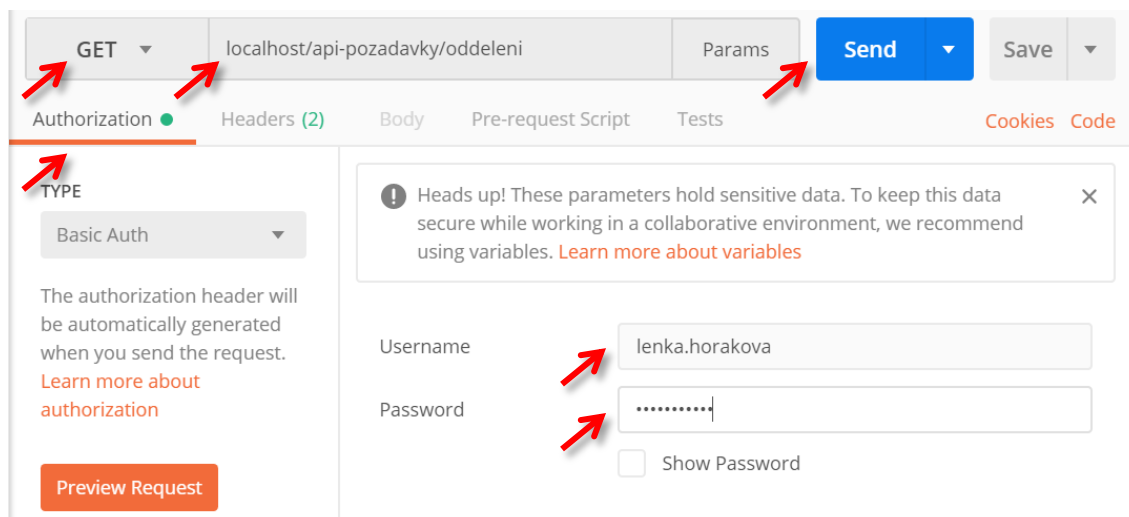
```
1  <?php
2
3  // nastavení databáze
4  define('DB_HOST', 'localhost');
5  define('DB_USERNAME', 'požadavky');
6  define('DB_PASSWORD', 'zdeBudeHeslo');
7  define('DBNAME', 'db_pozadavky');
8
9  // doména e-mailových adres uživatelů
10 define('DOMENA', '');
```

**Zdrojový kód 1:** Ukázka souboru config.php

Jak bylo uvedeno výše, jednotlivé HTTP požadavky lze zadávat pomocí REST klienta, kterým může být např. v Google Chrome rozšíření Postman. Jelikož testování API bude probíhat pomocí této aplikace, bude i používání API vysvětleno na aplikaci Postman. Tuto aplikaci lze nainstalovat z internetové adresy <https://www.getpostman.com/downloads/> [10].

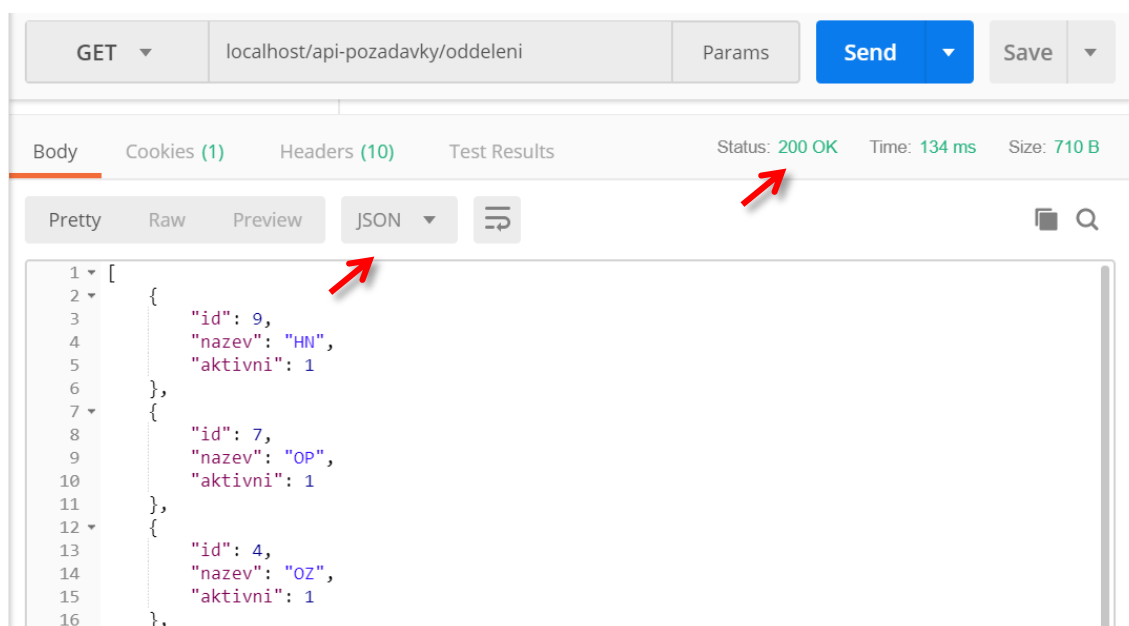
Pro zpracování HTTP požadavků je nutné zadat URL a typ metody dle přílohy B, tabulky 25 (viz obrázek 10). V případě zadání HTTP požadavků, které nemají v URL uvedenou cestu „.../verejne/...“, je nutné zaslat spolu s HTTP požadavkem také uživatelské jméno a heslo. Tyto údaje lze zadat na záložce *Authorization* vyplněním polí

*Username a Password* (viz obrázek 10). Tím se vyplněná data (uživatelské jméno a heslo) budou zasílat v hlavičce každého zaslání HTTP požadavku. HTTP požadavek se odesílá tlačítkem *Send*.



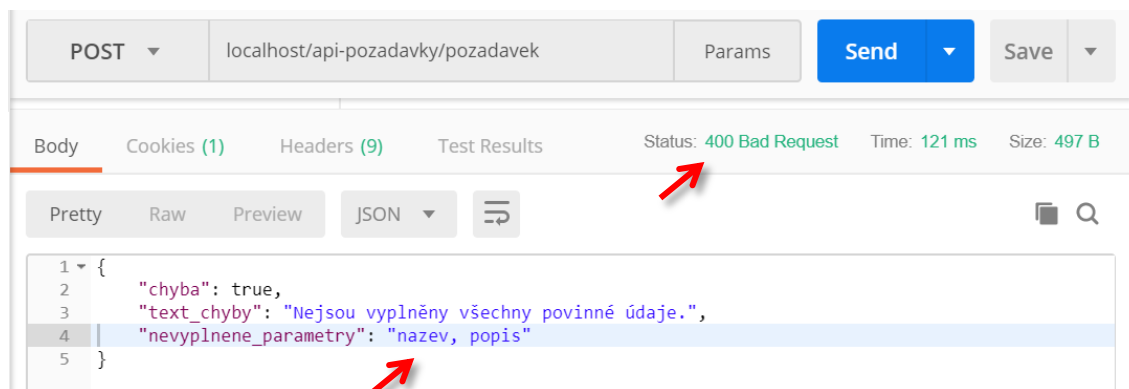
**Obrázek 10:** API – autorizace

Při použití metody GET jsou data přehledně vypsána ve formátu JSON (viz obrázek 11), lze však vybrat i jiný formát (např. XML, HTML, TEXT). Spolu s výsledkem operace (v tomto případě s požadovanými daty) je zaslán i stavový kód s informací o výsledku operace.



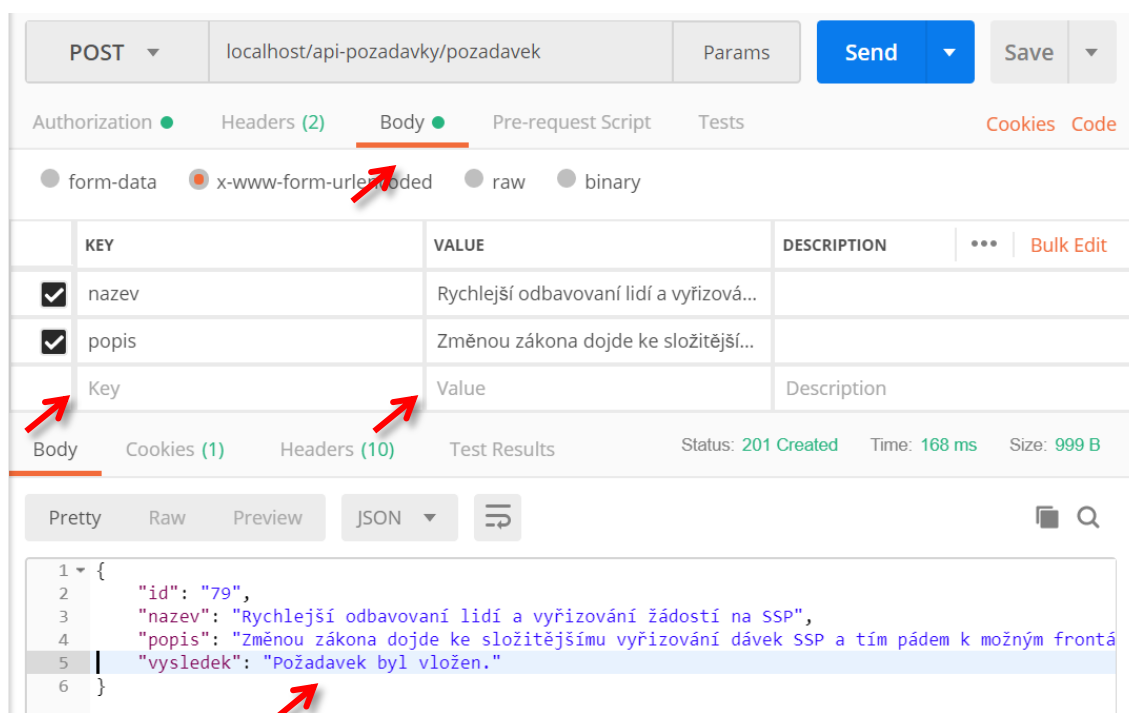
**Obrázek 11:** API – formát dat

Při použití metod POST pro založení nového záznamu a PUT pro opravu stávajícího záznamu musí být vyplněny příslušné parametry. Ty mohou být jak povinné, bez kterých nejde záznam založit, resp. opravit, tak i nepovinné. Pokud povinné parametry nejsou vyplněny, API na to uživatele upozorní (viz obrázek 12) a požadovaná operace není vykována.



Obrázek 12: API – nevyplněné parametry

Požadované parametry se vyplňují na záložce *Body*. Do sloupce *KEY* se uvede název parametru – ten je vypsán v odpovědi jako „nevyplněné parametry“ (viz obrázek 12) a do sloupce *VALUE* se uvede jeho hodnota. Z obrázku 13 je patrné, že po zaslání parametrů byl požadavek vykonán a záznam byl založen.



Obrázek 13: API – vložení parametrů

## 6.2 Grafické uživatelské rozhraní

Grafické uživatelské rozhraní bylo uděláno na základě grafického návrhu popsaného výše. Vzhledem k tomu, že se u detailů požadavků zobrazuje větší množství prvků, bylo zjištěno, že původní návrh nebyl zcela přehledný, proto byl upraven a to především tím, že celý řádek s informacemi o založení požadavku/řešení (kdy byl požadavek založen, kdo ho založil, jakého se týká oddělení a termín, do kterého by se měl vyřešit) byl přesunut výše a byl dále využit pro informaci o vybraném řešení. Využito bylo i barevné zvýraznění některých prvků aplikace. Celá aplikace je optimalizovaná pro internetový prohlížeč Google Chrome verze 69.0 a z tohoto prohlížeče jsou i následující ukázky aplikace.

### 6.2.1 Nastavení aplikace

Nejprve je nutné v souboru *config.js* nastavit cestu k API, které bude aplikace využívat. Dále lze nastavit volbu, zda se budou uživatelé moci do aplikace sami registrovat či tuto registraci bude dělat správce systému. Možnost registrace uživateli je doporučována, pokud nejsou stanovena žádná pravidla pro tvorbu uživatelských jmen a uživatelé si mohou zvolit jakékoliv jméno. Při této volbě je dobré nastavit v API kontrolu zadané domény e-mailových adres, a to z důvodu zabránění duplicitních registrací. Pokud je registrace uživatelem povolena, nastaví se konfiguračním souboru hodnota proměnné `registrace` na 1. Pokud bude uživatel registrovat správce systému, bude proměnná `registrace` nastavena na hodnotu 0. Viz zdrojový kód 2.

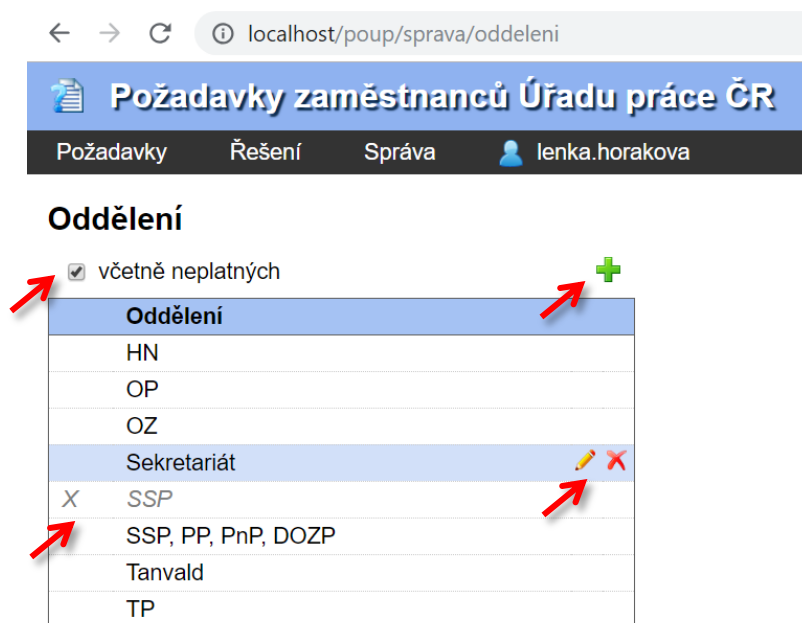
```
1 // nastavení url
2 var api_path = "/api-pozadavky";
3
4 // registrace = 1 -> v menu je pro uživatele přístupná registrace
5 // registrace = 0 -> registrace není uživatelům povolena
6 var registrace = 1;
```

**Zdrojový kód 2:** Ukázka souboru config.js

### 6.2.2 Číselník oddělení

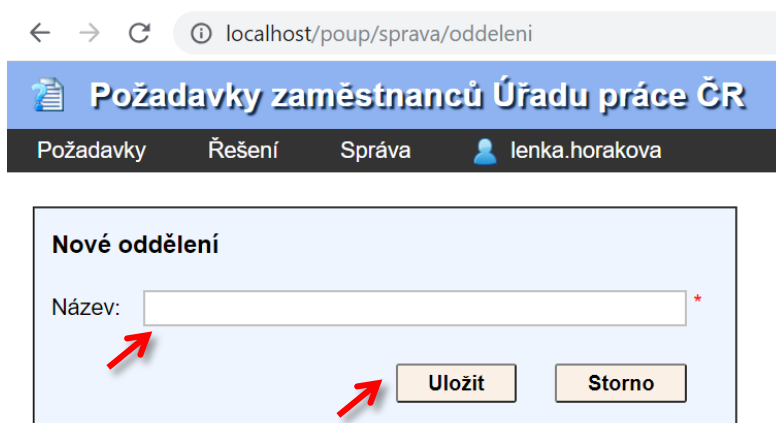
Aby byla aplikace plně funkční, je potřeba nejprve vytvořit číselník oddělení, který slouží pro naplnění rozbalovacích seznamů v různých formulářích. Pro naplnění rozbalovacích seznamů jsou použita pouze oddělení, která jsou označena jako platná, tj. aktivní. **Číselník oddělení** je po přihlášení uživatele (viz dále) přístupný z menu *Správa*,

a to pro uživatele s oprávněním správce systému. V číselníku oddělení si lze nechat zobrazit pouze aktuální (platné) oddělení nebo všechna oddělení, a to zaškrtnutím volby *včetně neplatných* (viz obrázek 14). Již neplatná oddělení jsou označena vlevo křížkem a zobrazena šedivým písmem.



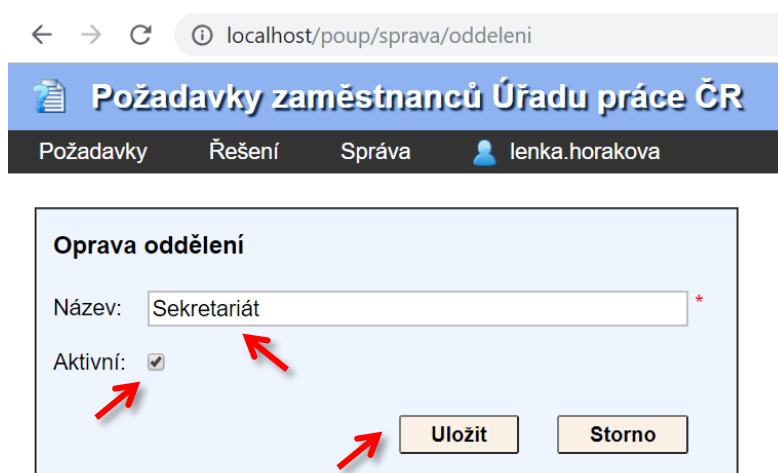
Obrázek 14: GUI – číselník oddělení

**Nové oddělení** se založí kliknutím na *plus* (viz obrázek 14) a vyplněním názvu, resp. zkratky oddělení (viz obrázek 15). Je vhodné volit vždy co nejkratší název, ale tak, aby byl všem uživatelům srozumitelný. Pokud je používána pro oddělení nějaká zkratka, je dobré ji pro tento účel použít. Při ukládání názvu oddělení aplikace kontroluje jeho duplicitu, tzn., že zvolený název nesmí být v aplikaci již použit. Uložené oddělení má automaticky příznak platného oddělení.



Obrázek 15: GUI – nové oddělení

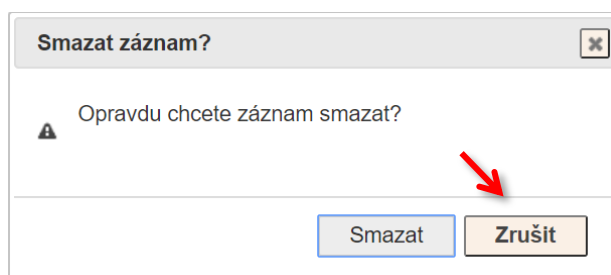
**Oprava oddělení** se provede najetím myši na příslušné oddělení a kliknutím na *tužku* nebo dvojklikem na zvoleném řádku s oddělením (viz obrázek 14). Tím se zobrazí formulář pro opravu oddělení, v kterém je oproti formuláři na založení nového oddělení navíc volba, zda je oddělení stále platné, tzn. aktivní (viz obrázek 16). Touto volbou tak lze rušit staré a již neplatné názvy oddělení.



**Obrázek 16:** GUI – oprava oddělení

Oprava názvu oddělení opět podléhá kontrole, zda zvolený název nebyl již v aplikaci použit. Tato oprava by se měla používat výjimečně – pokud např. dojde k překlepu nebo se plný název mění ve zkratku, ale jedná se o totéž oddělení. V žádném případě se nesmí opravou názvu přepisovat jedno oddělení na jiné. Tím by došlo k přepsání daného oddělení u všech uživatelů a u založených požadavků na jiné a záznamy by již neodpovídaly skutečnosti.

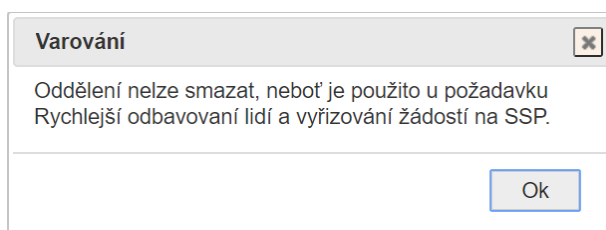
**Smazání názvu oddělení** lze provést najetím myši na příslušný řádek a kliknutím na *červený křížek* (viz obrázek 14). Akci lze ještě zrušit v dialogovém okně potvrzující smazání záznamu (viz obrázek 17).



**Obrázek 17:** GUI – potvrzení smazání oddělení



Pokud je dialog potvrzen, záznam je smazán. Dané oddělení však nesmí být použito u žádného uživatele, jeho oprávnění, ani u založeného požadavku. V takovém případě je na tuto skutečnost uživatel upozorněn a záznam smazán není (viz obrázek 18). Pokud dojde ke zrušení oddělení, lze ho vyřadit z rozbalovacích seznamů na formulářích zrušením příznaku *Aktivní* (viz výše).



**Obrázek 18:** GUI – varování – záznam nelze smazat

### 6.2.3 Správa uživatelů

Jednotlivé uživatele může spravovat správce systému, a to zvolením položky *Uživatelé* z menu *Správa* (ta je přístupná pouze pro správce systému). Zvolením této položky se zobrazí seznam uživatelů (viz obrázek 19). Zaškrtnutím volby *včetně neaktivních* se zobrazí všichni uživatelé, kteří byli do systému registrováni. Neaktivní uživatelé jsou zobrazeni šedivým písmem s křížkem na levé straně řádku.

|   | Uživatelské jméno | Oddělení           |
|---|-------------------|--------------------|
|   | adela.novotna     | OZ                 |
|   | lenka.horakova    | Sekretariát        |
|   | pavel.novak       | OZ                 |
| X | petra.moudra      | OP                 |
|   | sarka.pilna       | Sekretariát        |
|   | tomas.novy        | OZ                 |
| X | veronika.nova     | SSP, PP, PnP, DOZP |

**Obrázek 19:** GUI – seznam uživatelů

Pro **založení nového uživatele** je potřeba kliknout na *plus*. Zobrazí se formulář (viz obrázek 20), do kterého se povinně vyplní uživatelské jméno, e-mailová adresa a oddělení. Je vhodné si zvolit nějaký systém pro volbu uživatelských jmen, např. použít e-mailové adresy, a to jejich část před zavináčem. Je ale samozřejmě možné i volit libovolná uživatelská jména. Stejně uživatelské jméno nesmí být použito dvakrát. Kontrola na duplicitu je i u e-mailové adresy, u které se navíc kontroluje její formát a pokud je zapnuta kontrola domény, tak i ta.

Dále se zvolí v rozbalovacím seznamu oddělení, ve kterém dotyčný uživatel pracuje. Rozbalovací seznam je naplněn pouze názvy oddělení, které mají příznak, že jsou aktivní. Další volby slouží pro zadání různých oprávnění. Zatřetím volby *Správce* se danému uživateli přiřadí oprávnění správce systému, tzn., že bude moci spravovat číselník oddělení a uživatelů.

← → ↻ localhost/poup/sprava/uzivatele

Požadavky zaměstnanců Úřadu práce ČR

Požadavky Řešení Správa lenka.horakova

**Nový uživatel**

Uživatelské jméno:  \*

E-mail:  \*

Oddělení:  \*

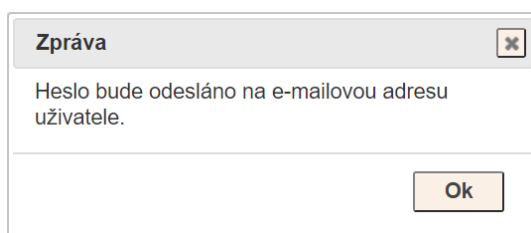
Správce:

**Oprávnění k oddělením:**

- Nezadané oddělení
- HN
- OP
- OZ
- Sekretariát
- SSP, PP, PnP, DOZP
- Tanvald
- TP

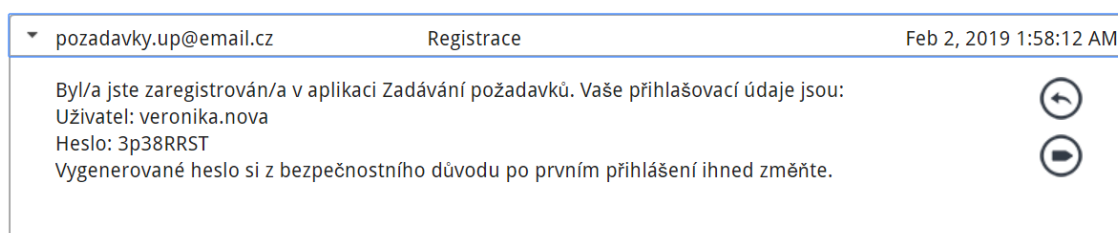
**Obrázek 20:** GUI – založení nového uživatele

Zaškrtnutím volby u jednotlivých oddělení, která jsou vypsána pod textem *Oprávnění k oddělením*, se uživateli přiřazuje oprávnění vyhodnocovat řešení požadavků týkajících se zaškrtnutého oddělení (jednotlivá oddělení jsou opět načtena z číselníku oddělení s příznakem aktivní). Uživatel bude moci tyto požadavky uzavírat, vybírat navržená řešení, která se uplatní v praxi, a psát k požadavkům poznámku. Při zatržení volby *Nezadané oddělení* bude moci uživatel vyhodnocovat řešení požadavku, u kterého není zadané žádné oddělení, tzn., že se týká celé organizace. Data nového uživatele se uloží tlačítkem *Uložit*.



**Obrázek 21:** GUI – zpráva o zaslání hesla

Uložením uživatele dojde k vygenerování hesla, které se skládá z 8 znaků. Toto heslo je uživateli odesláno na e-mailovou adresu uvedenou na formuláři (viz obrázek 21). Z bezpečnostních důvodů je doporučeno si toto heslo ihned po prvním přihlášení změnit (viz dále), na což je uživatel v e-mailu upozorněn (viz obrázek 22).



**Obrázek 22:** GUI – e-mailová zpráva o registraci uživatele

**Opravit uživatele, změnit mu oprávnění či vygenerovat nové heslo** je možné najetím myši na řádek s uživatelem a dvojklikem ho zvolit nebo na daném řádku kliknout na *tužku* (viz obrázek 19). Zobrazí se formulář s daty uživatele (viz obrázek 23). Zde je možné mimo jiné i opravit uživatelské jméno. To by se však mělo opravovat jen ve výjimečných případech a zásadně se nesmí přepisovat na jiného uživatele. Tato změna se uživateli nezasílá na danou e-mailovou adresu – změnu uživatelského jména musí správce systému uživateli sdělit jiným způsobem. Stejná pravidla platí pro opravu e-mailové adresy. Při opravě uživatelského jména

a e-mailové adresy se také kontroluje, zda již tyto údaje nejsou v aplikaci použity, pokud ano, musí se zvolit jiné.

**Oprava uživatele**

Uživatelské jméno:  \*

E-mail:  \*

Oddělení:  \*

Aktivní:

Správce:

**Oprávnění k oddělením:**

- Nezadané oddělení
- HN
- OP
- OZ
- Sekretariát
- SSP, PP, PnP, DOZP
- Tanvald
- TP

**Obrázek 23:** GUI – oprava uživatele

Oprava uživatele slouží i pro **zneplatnění jeho účtu**, a to zrušením zatržení volby *Aktivní* (viz obrázek 23). Dále lze volbou *Správce* přiřadit nebo zrušit oprávnění správce systému a volbami u jednotlivých oddělení přiřadit či zrušit oprávnění vyhodnocovat řešení požadavků týkajících se daného oddělení. Oprava uživatele také umožňuje **vygenerovat uživateli nové heslo** kliknutím na tlačítko *Reset hesla*, pokud ho uživatel zapomněl. Toto heslo je následně zasláno na e-mailovou adresu uživatele. Uživatel je opět upozorněn, aby si vygenerované heslo ihned změnil.

Zadaného uživatele lze i **smazat**, a to najetím na příslušný řádek a kliknutím na *červený křížek* (viz obrázek 19). Smazání uživatele se musí ještě potvrdit na dialogovém okně. Aplikace však dovolí smazat pouze uživatele, který ještě nenapsal žádný požadavek, nenavrhl žádné řešení či nenapsal jakýkoliv příspěvek a nehodnotil žádné řešení.

## 6.2.4 Registrace uživatele

Pokud se uživatelé mohou registrovat sami (resp. je tato možnost povolena), je možné registraci provést zvolením *Registrace* z menu *Přihlášení*. Zde (viz obrázek 24) si uživatel vyplní své uživatelské jméno dle pokynů správce systému, tak aby byl zachován dohodnutý systém, pokud je stanoven, dále vyplní svou pracovní e-mailovou adresu (pokud není se správcem systému dohodnuto jinak) a oddělení, ve kterém pracuje. Dále si zvolí, zda si chce nechat zasílat informační e-mail o založení nového požadavku – tento e-mail bude zaslán s každým nově zadaným požadavkem.

Vyplněná data se uloží kliknutím na tlačítko *Uložit*. Pokud je zvoleno již použité uživatelské jméno, aplikace na to upozorní a musí být zvoleno jiné. Po uložení dat je na uvedenou e-mailovou adresu zaslán e-mail s vygenerovaným heslem, které je potřeba si po prvním přihlášení změnit (viz dále).

The screenshot shows a web browser window with the address bar displaying 'localhost/poup/uzivatel/registrace'. The page title is 'Požadavky zaměstnanců Úřadu práce ČR'. Below the title is a navigation menu with items: 'Požadavky', 'Řešení', 'Správa', and 'Přihlášení'. The main content area is titled 'Registrace' and contains a registration form. The form has three input fields: 'Uživatelské jméno:', 'E-mail:', and 'Oddělení:'. Each field has a red asterisk to its right. Below these fields is a checkbox labeled 'Zasílat e-mailem informaci o založení nového požadavku'. At the bottom right of the form are two buttons: 'Uložit' and 'Storno'. Red arrows point to the input fields and the checkbox.

Obrázek 24: GUI – registrace uživatele

## 6.2.5 Přihlášení

Přihlášení je přístupné z menu *Přihlášení*, položka *Přihlásit se*. Do zobrazeného formuláře se vyplní přihlašovací jméno a heslo a klikne se na *Přihlásit se* (viz obrázek 25). Pokud uživatel zapomněl své přihlašovací údaje, může si je nechat zaslat na svoji e-mailovou adresu a to vyplněním e-mailové adresy do příslušné kolonky a kliknutím na tlačítko *Zaslat nové heslo* (viz obrázek 25). Aplikace vygeneruje nové heslo a zašle ho na uvedenou

e-mailovou adresu. Z bezpečnostních důvodů je nutné si toto vygenerované heslo ihned po přihlášení do aplikace změnit (viz dále).

← → ↻ localhost/poup/uzivatel/prihlaseni

**Požadavky zaměstnanců Úřadu práce ČR**

Požadavky   Řešení   Správa   Přihlášení

### Přihlášení

Uživatelské jméno:  \*

Heslo:  \*

**Zapomněli jste přihlašovací údaje?**  
Nevadí, zadejte svou e-mailovou adresu a bude Vám zasláno nové heslo.

E-mailová adresa:

Obrázek 25: GUI – přihlášení uživatele

## 6.2.6 Nastavení uživatele

Po přihlášení do aplikace se menu *Přihlášení* změní na přihlašovací jméno uživatele a zpřístupní se volby pro změnu hesla uživatele, jeho nastavení a samozřejmě odhlášení uživatele z aplikace (viz obrázek 26).

← → ↻ localhost/poup/uzivatel/prihlaseni

**Požadavky zaměstnanců Úřadu práce ČR**

Požadavky   Řešení   Správa   lenka.horakova

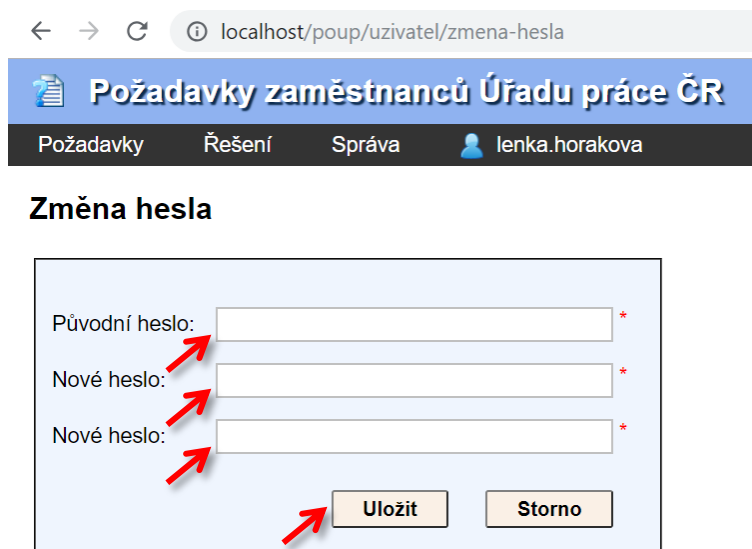
### Přihlášení

Uživatel je přihlášen.

- Změna hesla
- Nastavení
- Odhlášení

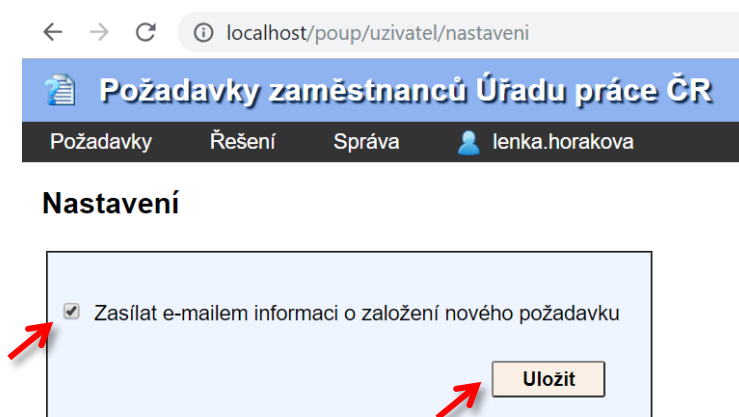
Obrázek 26: GUI – menu uživatele

**Změna hesla** se provede volbou *Změna hesla* z menu uživatele a vyplněním původního hesla (kterým se uživatel přihlásil do aplikace) a dvakrát nového hesla (viz obrázek 27). Tato změna se uloží tlačítkem *Uložit*. Pro uložení hesla je nutné, aby obě zadaná nová hesla byla shodná. Zvolené heslo může obsahovat jakékoliv znaky.



**Obrázek 27:** GUI – změna hesla

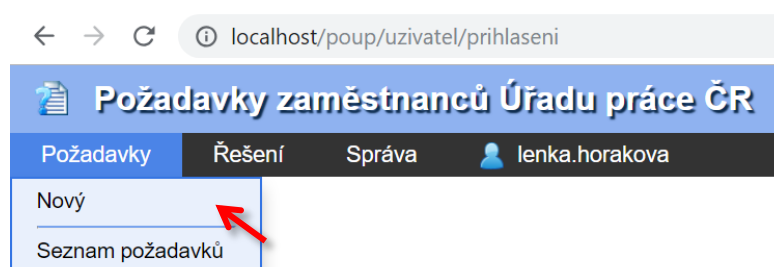
V menu *Nastavení* je možné si změnit volbu pro zasílání upozornění na založení nového požadavku, a to zaškrtnutím této volby a uložením změny (viz obrázek 28). E-mail je zasílán, pokud je založen jakýkoliv požadavek od jakéhokoliv uživatele bez rozlišení.



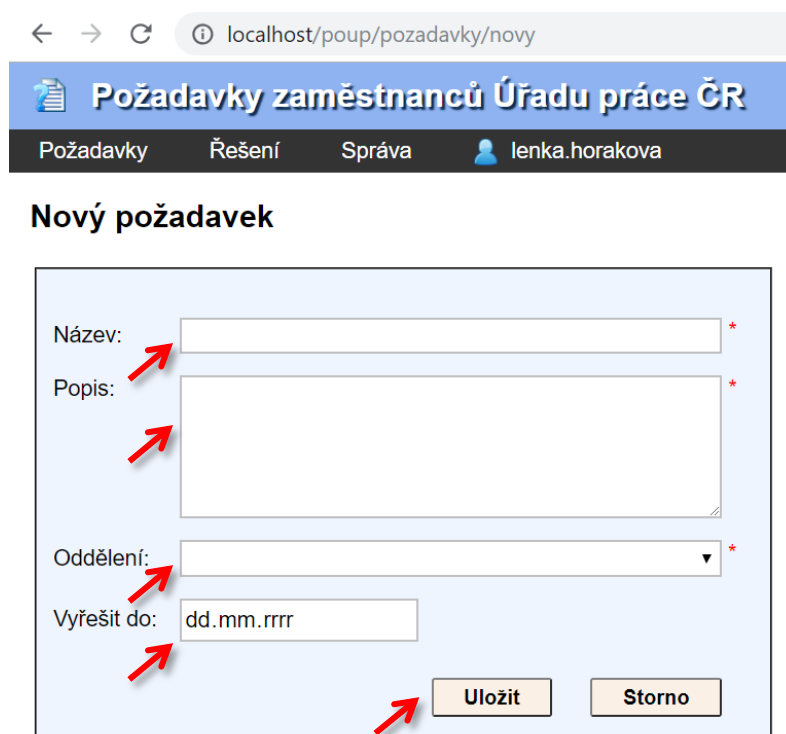
**Obrázek 28:** GUI – nastavení uživatele

## 6.2.7 Založení požadavku

Nový požadavek se vytvoří zvolením položky *Nový* z menu *Požadavky* (viz obrázek 29). Po tomto výběru se zobrazí příslušný formulář (viz obrázek 30). Povinnými položkami k vyplnění jsou název, popis požadavku a oddělení, kterého se požadavek týká. Název a popis požadavku by měl být krátký a výstižný. Pokud je vyplněno konkrétní oddělení, bude se zvlášť evidovat hodnocení řešení uživatelů patřících do oddělení, kterého se daný požadavek týká, a zvlášť hodnocení ostatních uživatelů. Pokud se požadavek týká celé organizace, do oddělení se vyplní „ – “ a hodnocení navržených řešení se nebude nijak rozlišovat.



Obrázek 29: GUI – menu Požadavky



Obrázek 30: GUI – nový požadavek



Na formuláři lze vyplnit i termín, do kterého by se měl požadavek vyřešit. Tento údaj má pouze informační charakter a aplikace v tomto termínu s požadavkem neprovádí žádnou akci. Zadaný termín slouží hlavně vedoucím (oprávněným uživatelům) pro informaci, do kdy by měli jednotlivá řešení vyhodnotit a napsat k požadavku nějaké stanovisko (viz dále). Požadavek se uloží kliknutím na tlačítko *Uložit*, tlačítko *Storno* slouží ke smazání vyplněných polí formuláře.

### 6.2.8 Prohlížení požadavků

Všechny zadané požadavky si lze prohlédnout v *Seznamu požadavků*, který je přístupný z menu *Požadavky* (viz obrázek 29). Zde se zadají výběrové podmínky pro zobrazení požadovaného seznamu požadavků. Možnosti jsou následující (viz obrázek 31):

- *Moje* – zobrazí jen požadavky zadané přihlášeným uživatelem
- *Otevřené* – na tyto požadavky je ještě možné navrhopat řešení a navržená řešení hodnotit
- *Uzavřené* – na tyto požadavky již nelze navrhopat řešení ani navrhovaná řešení hodnotit, jsou zpřístupněny volby a funkce pro oprávněné uživatele (vedoucí oddělení) sloužící k vyhodnocení navržených řešení
- *Všechny* – jsou zobrazeny všechny zadané požadavky
- *Oddělení* – zobrazí se jen požadavky, které se týkají zadaného oddělení
- *Vytvoření požadavku od – do* – zobrazí se požadavky vytvořené v zadaném rozmezí dní
- *Uzavření požadavku od – do* – zobrazí se požadavky uzavřené v zadaném rozmezí dní
- *S příspěvky zadanými za posledních X dní, Y hodin* – zobrazí se požadavky, u kterých byl zadán příspěvek za posledních zadaných X dní a Y hodin

V seznamu požadavků jsou rozlišeny otevřené požadavky, které jsou zobrazeny tučným písmem, a uzavřené požadavky zobrazeny normálním písmem. Detail požadavku lze zobrazit buď kliknutím na ikonku *tužky* na vybraném řádku, nebo dvojklikem na daný řádek (viz obrázek 31). Po zobrazení detailu požadavku zůstávají výběrové podmínky spolu s načteným seznamem požadavků zobrazeny pod detailem požadavku. Pro zobrazení jiného požadavku tak stačí pouze kliknout na příslušný požadavek v již načteném seznamu, popř. načíst jiný seznam požadavků změnou výběrových podmínek.

← → ↻ localhost/poup/poradavky/vyber

**Požadavky zaměstnanců Úřadu práce ČR**

Požadavky    Řešení    Správa    lenka.horakova

### Seznam požadavků

Moje   
 Otevřené   
 Uzavřené   
 Všechny

Oddělení:

Vytvoření požadavku od:  do:

Uzavření požadavku od:  do:

S příspěvky zadanými za posledních  dní  hodin

| Vytvořeno         | Název   | Oddělení           | Nejnovější příspěvek    |
|-------------------|---|--------------------|-------------------------|
| 03.02.2019        | Zpožděné fasování kancelářských potřeb                |                    | 12.02.2019 10:45        |
| <b>03.02.2019</b> | <b>Sledování zakázek</b>                              | <b>Sekretariát</b> | <b>03.02.2019 21:16</b> |
| 04.02.2019        | Rychlejší odbavování lidí a vyřizování žádostí na SSP | SSP, PP, PnP, DOZP | 02.03.2019 14:08        |
| 19.02.2019        | Zrušení abecedy na přepážkách SSP                     | SSP, PP, PnP, DOZP | 23.02.2019 15:06        |
| 19.02.2019        | Kdo bude registrovat smlouvy                          |                    | 19.02.2019 22:38        |

**Obrázek 31:** GUI – výběr požadavků

Vzhled detailu požadavku je dán oprávněním přihlášeného uživatele a stavem požadavku. Následující obrazovky jsou uváděny pro uživatele, kteří nemají oprávnění vyhodnocovat navržená řešení požadavků. Na těchto obrazovkách jsou postupně vidět různé varianty zobrazení detailů požadavků, s kterými se mohou uživatelé setkat. Každý požadavek obsahuje název požadavku, pod kterým je popis požadavku. Nad názvem je zobrazen řádek s informacemi, kdy byl požadavek založen, kdo ho založil, jakého oddělení se týká a termínem, do kterého má být požadavek vyřešen (pokud jsou údaje zadány). U navrženého řešení je to podobné, s tím rozdílem, že je zobrazen pouze popis navrženého řešení a v řádku nad ním je pouze informace o tom, kdy a kým bylo řešení navrženo.

Na obrázku 32 je vidět detail otevřeného požadavku, tzn., že je možné k němu navrhnout řešení a tato řešení hodnotit (v seznamu požadavků je zobrazen tučným písmem). Detaily jednotlivých požadavků jsou zobrazeny se všemi zadanými řešeními, resp. reakcemi uživatelů. Tato navržená řešení jsou umístěna pod požadavkem a tvoří stromovou strukturu podle toho, na jaké příspěvky uživatelé psali své reakce či další řešení (viz obrázek 32).

← → ↻ localhost/poup/pozadavky/79

**Požadavky zaměstnanců Úřadu práce ČR**

Požadavky   Rešení   Správa   lenka.horakova

Červeně zvýraznit řešení zadané za posledních  dní  hodin

Založeno: 04.02.2019 18:58   Uživatel: pavel.novak   Týká se oddělení: SSP, PP, PnP, DOZP   Vyřešit do: 28.02.2019

**Rychlejší odbavování lidí a vyřizování žádostí na SSP**  
 Změnou zákona dojde ke složitějšímu vyřizování dávek SSP a tím pádem k možným frontám u přepážek a k prodloužení lhůt vyřizování žádostí. Potřebovali bychom nápady na rychlejší odbavování lidí a vyřizování žádostí.

Ohodnotit ☆☆☆☆☆   **Navrhnout řešení**

22.02.2019 10:37   sarka.piina  
 Mohl by nám někdo pomoci při vybírání žádostí od klientů kopírovat přílohy k žádostem, razítkovat je a dávat na ně štítky.

Ohodnotit ☆☆☆☆☆   **Reagovat**

25.02.2019 14:08   lenka.horakova  
 Co udělat veřejně přístupnou tabulku, kde by se na daný termín psali se svolením vedoucích kolegové z jiných oddělení, kteří mohou a chtějí pomoci?

Ohodnotit ☆☆☆☆☆   **Reagovat**

22.02.2019 10:06   lenka.horakova  
 Pan Lounský by mohl pomáhat lidem se zorientovat a navést je, kam mají jít.

Ohodnotit ☆☆☆☆☆   **Reagovat**

20.02.2019 22:56   lenka.horakova  
 Mohla by paní Kusá pipat dokumenty do krabice na digitalizaci?

Ohodnotit ☆☆☆☆☆   **Reagovat**

**Obrázek 32:** GUI – stromová struktura příspěvků

Pro lepší orientaci v nově přidaných řešeních či jiných reakcích je možné si nechat červeně zvýraznit příspěvky zadané za posledních X dní a Y hodin, a to zadáním těchto údajů do příslušných polí a kliknutím na ikonku *znovunačtení dat*. Na obrázku 33 je zvýrazněno navržené řešení, které bylo zadáno za poslední 1 den, resp. 24 hodin. Tato funkce je přístupná pro požadavky v jakémkoliv stavu a pro jakéhokoliv uživatele bez rozlišení jeho oprávnění.

Pokud je již požadavek uzavřen (v seznamu požadavků je zobrazen normálním písmem), nelze již k němu navrhnout žádná řešení či navržená řešení hodnotit. Informace o uzavření požadavku spolu s datem a časem uzavření je zobrazena pod názvem požadavku (viz obrázek 34). Pokud byla navržená řešení uživateli hodnocena, je toto hodnocení zobrazeno u každého takového řešení (viz dále).

← → ↻ ⓘ localhost/poup/pozadavky/79

**Požadavky zaměstnanců Úřadu práce ČR**

Požadavky   Rešení   Správa   lenka.horakova

Červeně zvýraznit řešení zadané za posledních  dní  hodin

Založeno: 04.02.2019 18:58   Uživatel: pavel.novak   Týká se oddělení: SSP, PP, PnP, DOZP   Vyřešit do: 28.02.2019

**Rychlejší odbavování lidí a vyřizování žádostí na SSP**

Změnou zákona dojde ke složitějšímu vyřizování dávek SSP a tím pádem k možným frontám u přepážek a k prodloužení lhůt vyřizování žádostí. Potřebovali bychom nápady na rychlejší odbavování lidí a vyřizování žádostí.

Ohodnotit ☆☆☆☆☆  

---

22.02.2019 10:37   sarka.pilna

Mohl by nám někdo pomoci při vybírání žádostí od klientů kopírovat přílohy k žádostem, razítkovat je a dávat na ně štítky.

Ohodnotit ☆☆☆☆☆  

Obrázek 33: GUI – označení nově zadaných příspěvků

Založeno: 04.02.2019 18:58   Uživatel: pavel.novak   Týká se oddělení: SSP, PP, PnP, DOZP   Vyřešit do: 28.02.2019

**Rychlejší odbavování lidí a vyřizování žádostí na SSP**

**Uzavřeno 26.02.2019 18:29** ←

Změnou zákona dojde ke složitějšímu vyřizování dávek SSP a tím pádem k možným frontám u přepážek a k prodloužení lhůt vyřizování žádostí. Potřebovali bychom nápady na rychlejší odbavování lidí a vyřizování žádostí.

**Poznámka k vybraným řešením:** ←  
Pomoc z ostatních oddělení bude probrána na nejbližší poradě.

**Odkazy na vybraná řešení:** ←  
[Mohla by paní Kusá pítat dokumenty do krabice na...](#)  
[Pan Lounský by mohl pomáhat lidem se zorientovat...](#)  
[Mohl by nám někdo pomoci při vybírání žádostí od...](#)

---

22.02.2019 10:37   sarka.pilna   **Řešení**

Mohl by nám někdo pomoci při vybírání žádostí od klientů kopírovat přílohy k žádostem, razítkovat je a dávat na ně štítky.

**Hodnocení řešení:**  ←

V oddělení hodnotilo: 5   Mimo oddělení hodnotilo: 2   Celkem hodnotilo: 7

☆☆☆☆☆ 4   ☆☆☆☆☆ 2   ☆☆☆☆☆ 3,43

---

25.02.2019 14:08   lenka.horakova

Co udělat veřejně přístupnou tabulku, kde by se na daný termín psali se svolením vedoucích kolegové z jiných oddělení, kteří mohou a chtějí pomoci?

Obrázek 34: GUI – uzavřený a vyhodnocený požadavek

Jednotlivá řešení, která byla vybrána pro použití v praxi, jsou označena zeleně zbarveným řádkem zobrazeným nad názvem požadavku a poznámkou *Řešení*. Pro jejich vyhledání není nutné prohledávat celou historii požadavku, resp. celý strom řešení a reakcí na ně, ale v detailu požadavku jsou vygenerovány jednotlivé odkazy na tato řešení. Po kliknutí na tento odkaz je stránka posunuta na zvolené řešení. Pokud byla k vyhodnocení požadavku, resp. k vybraným řešením, napsána poznámka, je uvedena pod popisem požadavku.

Pokud není u daného požadavku poznámka zobrazena a nejsou označena vybraná řešení, ale požadavek je již uzavřen, znamená to, že je ve stavu vyhodnocování navržených řešení, tzn., že již k němu nelze přidávat žádná řešení nebo řešení hodnotit, ale odpovědný pracovník (vedoucí oddělení) ještě navržená řešení nevyhodnotil a nedal k požadavku žádné stanovisko.

V případě, že samotný požadavek ve svém popisu obsahuje zároveň i navržené řešení, je možné toto řešení hodnotit a toto hodnocení je zobrazeno přímo u požadavku. Hodnocení se tedy netýká navrženého požadavku, ale navrženého řešení, které požadavek obsahuje. Toto řešení tak může být i následně vybráno pro realizaci v praxi. Tato situace je vidět na obrázku 35. Pokud tedy samotný požadavek nenabízí zároveň i řešení, neměl by hodnocení vůbec obsahovat.

Založeno: 19.02.2019 22:22    Uživatel: lenka.horakova    Týká se oddělení: SSP, PP, PnP, DOZP    Řešení

### Zrušení abecedy na přepážkách SSP

Uzavřeno 28.02.2019 22:17

Navrhuj z důvodu spravedlivějšího rozdělení klientů zrušit abecedu na přepážkách SSP.

**Hodnocení řešení:** [Zobrazit podrobnosti](#)

V oddělení hodnotilo: 1    Mimo oddělení hodnotilo: 1    Celkem hodnotilo: 2

★★★★★ 5    ★★☆☆☆☆ 2    ★★★★★☆ 3,5

---

**Poznámka k vybraným řešením:**  
Abeceda bude zrušena k datu 1.3.2019.

**Odkazy na vybraná řešení:**  
[Navrhuj z důvodu spravedlivějšího rozdělení...](#)  
[V tom případě by se měly žádosti po jejich...](#)  
[Mělo by se zajistit, aby klienti nechodili jen k...](#)

**Obrázek 35:** GUI – návrh řešení je uveden v požadavku

Pokud je požadavek z nějakého důvodu po svém uzavření a vyhodnocení opět otevřen, zpřístupní se uživatelům znovu možnost navrhnout řešení požadavku a navržená řešení hodnotit. Přitom zůstanou zeleně označena vybraná řešení pro následnou realizaci, dále jsou zobrazeny vygenerované odkazy na tato řešení a zobrazena poznámka k vybraným řešením (viz obrázek 36).

Založeno: 19.02.2019 22:22    Uživatel: lenka.horakova    Týká se oddělení: SSP, PP, PnP, DOZP    Řešení

### Zrušení abecedy na přepážkách SSP

Navrhuj z důvodu spravedlivějšího rozdělení klientů zrušit abecedu na přepážkách SSP.

**Hodnocení řešení:** [Zobrazit podrobnosti](#)

V oddělení hodnotilo: 1    Mimo oddělení hodnotilo: 1    Celkem hodnotilo: 2

★★★★★ 5    ★★☆☆☆ 2    ★★★★★☆ 3,5

**Ohodnotit** [Navrhnout řešení](#)

**Poznámka k vybraným řešením:**  
Abeceda bude zrušena k datu 1.3.2019. Prosím o vyjádření se, zda a jakým způsobem přerozdělit vybrané žádosti.

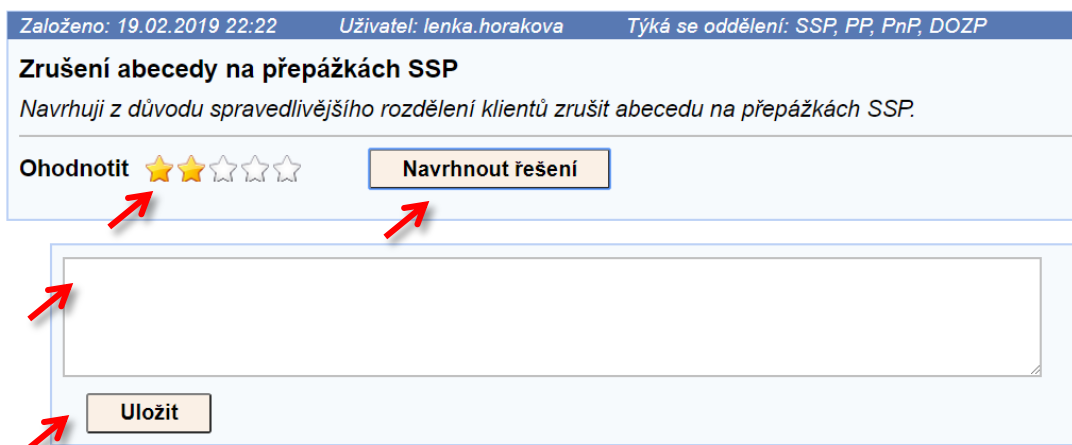
**Odkazy na vybraná řešení:**  
[Navrhuj z důvodu spravedlivějšího rozdělení...](#)  
[V tom případě by se měly žádosti po jejich...](#)  
[Mělo by se zajistit, aby klienti nechodili jen k...](#)

**Obrázek 36:** GUI – opětovně otevřený požadavek

Pokud dojde k této situaci, je dobré si vždy přečíst poznámku k vybraným řešením, neboť může být tímto způsobem od uživatelů požadován konkrétní postup, např. hodnocení pouze vybraných řešení, resp. tímto způsobem může být od uživatelů požadováno, aby se rozhodli mezi dvěma či více vybranými řešeními, která z nich se by se měla realizovat.

### 6.2.9 Navržení řešení požadavků a hodnocení řešení

Navrhnout řešení požadavku či reagovat na navržené řešení lze kliknutím na tlačítko *Navrhnout řešení*, resp. u jednotlivých řešení na tlačítko *Reagovat* (viz obrázek 37). Tím se zobrazí textové pole, do kterého je možné napsat navržené řešení či svou reakci na daný požadavek, resp. na navržené řešení. Danou reakci je nutné uložit kliknutím na tlačítko *Uložit*. Skrýt textové pole lze opětovným kliknutím na tlačítko *Navrhnout řešení*.



**Obrázek 37:** GUI – návržení a hodnocení řešení

Ohodnotit zadané řešení lze zvolením počtu hvězdiček zobrazených u jednotlivých řešení (viz obrázek 37). Jedna hvězdička znamená nejnižší ohodnocení, resp. že navržené řešení není vyhovující, pět hvězdiček znamená nejlepší hodnocení, tzn., že s daným řešením uživatel plně souhlasí a podporuje ho. Pokud bylo dané řešení již uživatelem hodnoceno, aplikace k řešení nepřidá nové hodnocení, ale opraví původní zadané. Uživatelé tak nemohou hodnotit dané řešení vícekrát. Hodnotit lze i požadavek, pokud obsahuje řešení, tzn., že se hodnotí navržené řešení požadavku, ne samotný požadavek.



**Obrázek 38:** GUI – zobrazené hodnocení řešení

Pokud je u požadavku zadáno oddělení, kterého se požadavek týká (je vidět v řádku zobrazeném nad názvem požadavku), rozlišuje se hodnocení na hodnocení uživatelů, kteří pracují v dotčeném oddělení, a na hodnocení ostatních uživatelů. Hodnocení je zobrazeno u příslušného navrženého řešení (viz obrázek 38). U jednotlivých skupin hodnocení (hodnocení uživatelů v/mimo oddělení, celkové hodnocení) jsou vidět počty uživatelů, kteří



dané řešení hodnotili. Pod tímto údajem jsou vypočítané průměrné hodnocení v jednotlivých skupinách, které jsou i graficky znázorněny pomocí zbarvených hvězdiček.

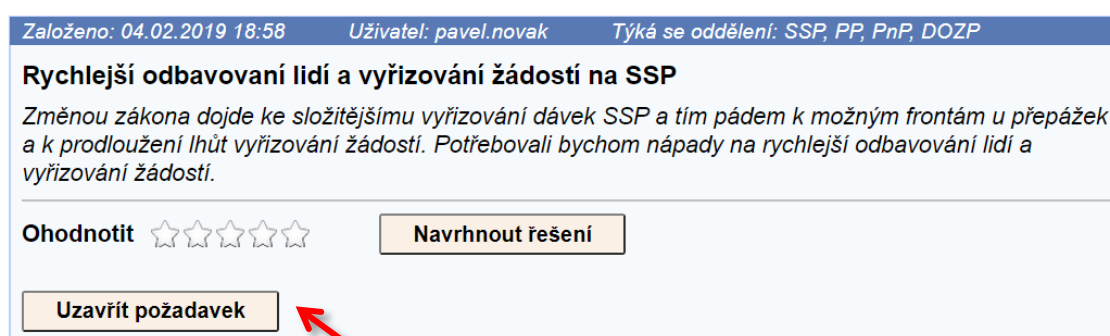
Hodnocení lze zobrazit s většími podrobnostmi, a to kliknutím na tlačítko *Zobrazit podrobnosti* (viz obrázek 38). Tím se u jednotlivých skupin hodnocení zobrazí rozložení jednotlivých hlasů, resp. hodnocení uživatelů dle počtu hvězdiček (viz obrázek 39). V závorce jsou pak uvedeny počty uživatelů, kteří navržené řešení hodnotili příslušným počtem hvězdiček. Skrýt tyto podrobnosti lze tlačítkem *Skrýt podrobnosti*.



Obrázek 39: GUI – zobrazené podrobnosti hodnocení

## 6.2.10 Vyhodnocení navržených řešení

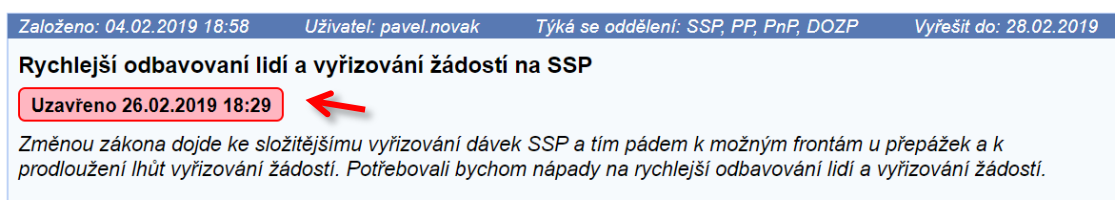
Po tom, co uživatelé navrhli svá řešení daného požadavku a ohodnotili ostatní navržená řešení, je možné tato řešení začít vyhodnocovat. Funkce pro vyhodnocení navržených řešení jsou přístupné pouze pro uživatele s patřičným oprávněním. Zpravidla jsou těmito uživateli vedoucí příslušného oddělení, kterého se daný požadavek týká.



Obrázek 40: GUI – uzavření požadavku

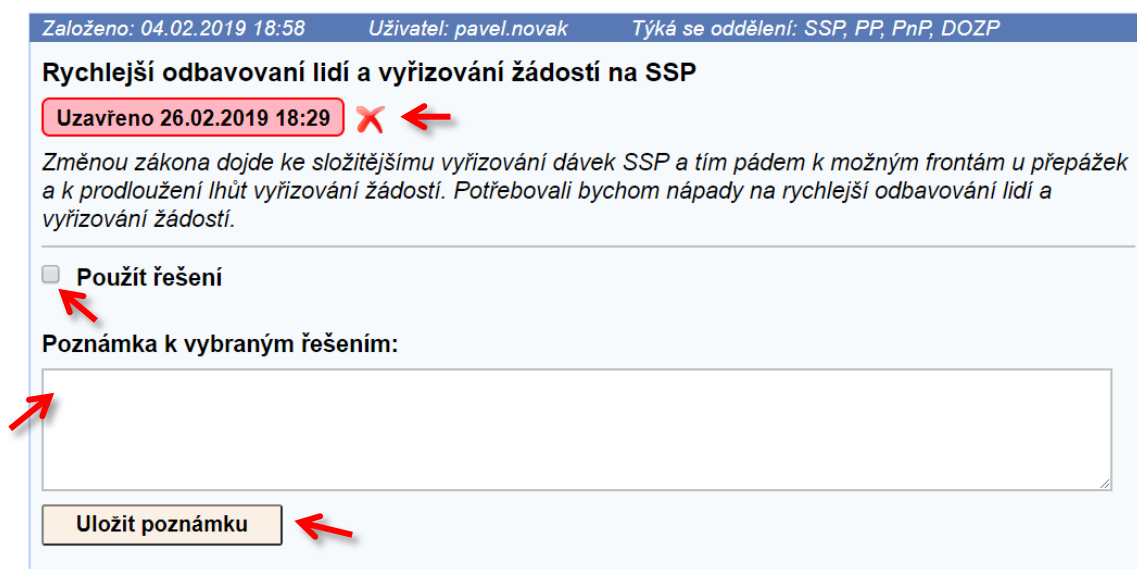


Před vlastním vyhodnocením navržených řešení je nejprve nutné požadavek uzavřít, a to na detailu požadavku tlačítkem *Uzavřít požadavek* (viz obrázek 40). Uzavřením požadavku se uživatelům znemožní přidávat další navrhované řešení a zadávat či měnit hodnocení řešení (viz obrázek 41). Dále se u požadavku objeví informace o uzavření požadavku s datem a časem uzavření.

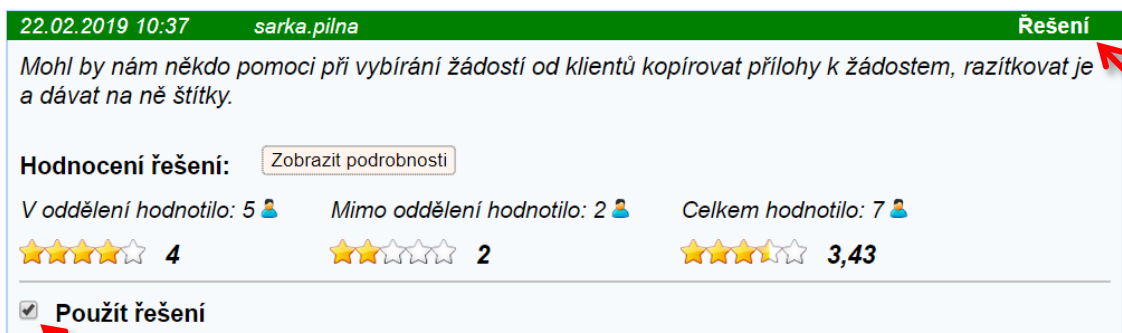


**Obrázek 41:** GUI – uzavřený požadavek

Pro oprávněné uživatele (vedoucí oddělení) se vedle informace o uzavření požadavku zpřístupní ikonka *křížku* (viz obrázek 42), pomocí níž je v případě potřeby možné uzavření požadavku zrušit – tím by mohli uživatelé pokračovat v navrhování dalších řešení a navržená řešení hodnotit. Pokud by již bylo před zrušením uzavření požadavku nějaké řešení označeno jako vybrané pro realizaci, toto označení řešení zůstane zachováno i po znovuootevření požadavku. Zachován zůstane i text poznámky k vybraným řešením. Tyto údaje tedy uvidí všichni přihlášení uživatelé (viz obrázek 36).

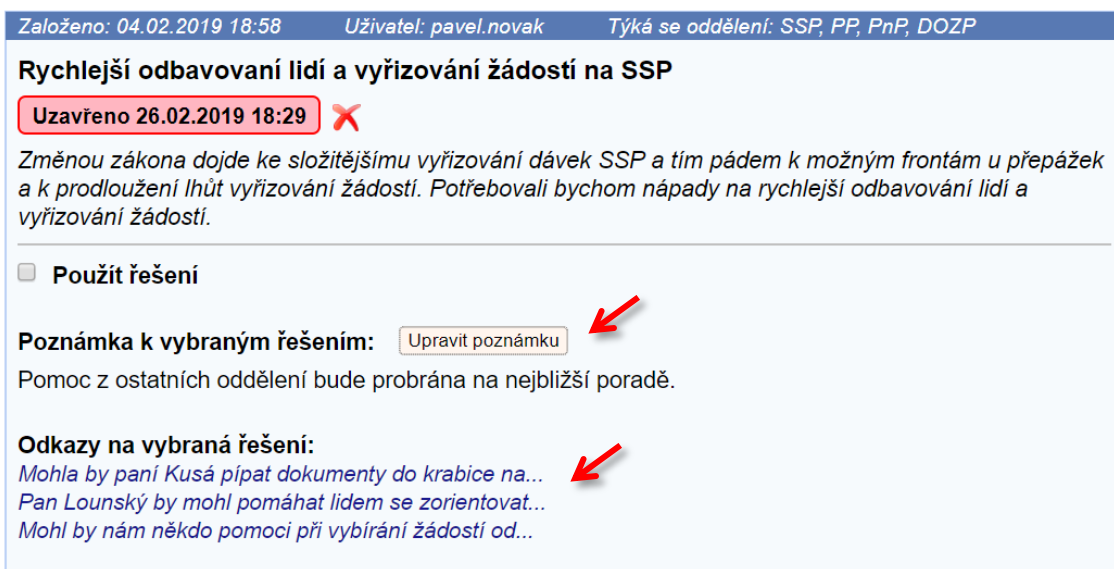


**Obrázek 42:** GUI – uzavřený požadavek – označení řešení a vložení poznámky



Obrázek 43: GUI – vybrané řešení

Po uzavření požadavku a vyhodnocení jednotlivých navržených řešení, se označí všechna navržená řešení, která byla vybrána pro realizaci v praxi, a to zaškrtnutím volby *Použít řešení* (viz obrázek 42). Tato volba je u každého navrženého řešení, resp. příspěvku, a je přístupná pro oprávněného uživatele (vedoucího oddělení). Zaškrtnutím této volby se zbarví řádek zobrazený nad názvem požadavku, resp. nad popisem řešení, na zelenou a objeví se v něm poznámka, že se jedná o vybrané řešení (viz obrázek 43). Dále se u požadavku vytvoří odkaz na dané řešení (viz obrázek 44), na který je možné kliknout pro zobrazení tohoto řešení.



Obrázek 44: GUI – oprava poznámky a zobrazené odkazy na vybraná řešení

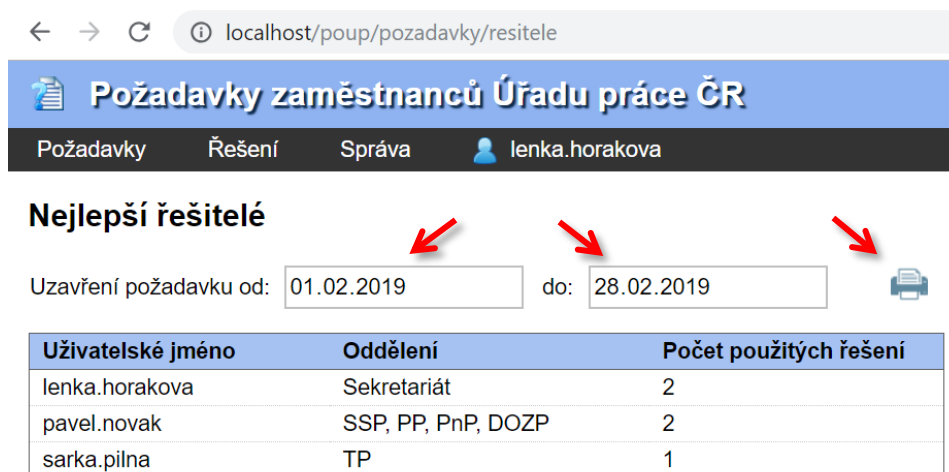
Pro oprávněné uživatele se u uzavřeného požadavku také zpřístupní textové pole pro zadání poznámky (viz obrázek 42). Tato poznámka by měla sloužit jako komentář k vybraným řešením – uživatelé by se z ní měli dovědět podrobnější informace k nasazení

vybraných řešení do praxe, např. jakým způsobem se bude postupovat v realizaci vybraných řešení, kdy budou tato řešení realizována apod. Poznámku je nutné uložit kliknutím na tlačítko *Uložit poznámku*.

Uložením poznámky se textové pole změní na text (viz obrázek 44). Pokud je potřeba poznámku opravit, je to umožněno kliknutím na tlačítko *Upravit poznámku*, tím se opět zpřístupní textové pole pro opravu a tlačítko *Uložit poznámku*. Označením vybraných řešení pro realizaci a uložením poznámky proces vyhodnocování požadavku končí.

### 6.2.11 Statistika nejlepších řešitelů

Aplikace umožňuje u každého uživatele zobrazit celkový počet řešení, které navrhl a které byly zároveň označeny pro použití v praxi. Tuto statistiku je možné zobrazit zvolením položky *Nejlepší řešitelé* v menu *Řešení*. Na této obrazovce se zvolí rozmezí dní, ve kterém byly uzavřeny požadavky, z kterých mají být jednotlivé počty označených řešení sečteny (viz obrázek 45). Po vyplnění tohoto období se automaticky zobrazí tabulka s požadovanými údaji. Zobrazení nejsou všichni uživatelé, ale pouze ti, kteří navrhli alespoň jedno řešení, které bylo označeno pro realizaci.



| Uživatelské jméno | Oddělení           | Počet použitých řešení |
|-------------------|--------------------|------------------------|
| lenka.horakova    | Sekretariát        | 2                      |
| pavel.novak       | SSP, PP, PnP, DOZP | 2                      |
| sarka.pilna       | TP                 | 1                      |

Obrázek 45: GUI – seznam nejlepších řešitelů

Pokud je potřeba statistiku vytisknout, je možné kliknout na ikonku s *tiskárnou* a stránka se upraví pro tisk (viz obrázky 45 a 46). Pro návrat na původní stránku se použije dvojklik myši na kterémkoliv místě na stránce. Náhled tisku je pak viditelný na obrázku 47.

## Nejlepší řešitelé

Uzavření požadavku od: 01.02.2019

do: 28.02.2019

| Uživatelské jméno | Oddělení           | Počet použitých řešení |
|-------------------|--------------------|------------------------|
| lenka.horakova    | Sekretariát        | 2                      |
| pavel.novak       | SSP, PP, PnP, DOZP | 2                      |
| sarka.pilna       | TP                 | 1                      |

Obrázek 46: GUI – úprava sestavy pro tisk

| Nejlepší řešitelé                 |                    |                        |
|-----------------------------------|--------------------|------------------------|
| Uzavření požadavku od: 01.02.2019 |                    | do: 28.02.2019         |
| Uživatelské jméno                 | Oddělení           | Počet použitých řešení |
| lenka.horakova                    | Sekretariát        | 2                      |
| pavel.novak                       | SSP, PP, PnP, DOZP | 2                      |
| sarka.pilna                       | TP                 | 1                      |

Obrázek 47: GUI – náhled tisku sestavy

### 6.2.12 Zobrazení vybraných řešení

Pokud je potřeba zobrazit přehled všech řešení vybraných k realizaci, je to možné výběrem položky *Seznam řešení* z menu *Řešení*. Na této stránce (viz obrázek 48) se zvolí rozmezí dní, ve kterém byly uzavřeny požadavky, pro které se mají vybraná řešení zobrazit. Po zadání těchto dat jsou automaticky příslušné požadavky zobrazeny spolu s řešeními označenými pro nasazení v praxi včetně jejich hodnocení (pokud nějaké obsahují). I zde si uživatelé mohou zobrazit podrobnosti tohoto hodnocení. Zobrazeny jsou pouze ty požadavky, u kterých je označeno alespoň jedno navržené řešení k realizaci. Pokud žádné řešení k realizaci použito nebylo, aplikace o tom uživatele informuje. V případě, že je potřeba stránku vytisknout, lze její vzhled upravit kliknutím na ikonku *tiskárny*.

← → ↻ ⓘ localhost/poup/pozadavky/reseni

**Požadavky zaměstnanců Úřadu práce ČR**

Požadavky Rešení Správa lenka.horakova

**Použitá řešení**

Uzavření požadavku do:  do:

Založeno: 04.02.2019 18:58 Uživatel: pavel.novak Týká se oddělení: SSP, PP, PnP, DOZP

**Rychlejší odbavování lidí a vyřizování žádostí na SSP**

**Uzavřeno 26.02.2019 18:29**

Změnou zákona dojde ke složitějšímu vyřizování dávek SSP a tím pádem k možným frontám u přepážek a k prodloužení lhůt vyřizování žádostí. Potřebovali bychom nápady na rychlejší odbavování lidí a vyřizování žádostí.

**Poznámka k vybraným řešením:**  
Pomoc z ostatních oddělení bude probrána na nejbližší poradě.

---

Založeno: 20.02.2019 22:56 Uživatel: lenka.horakova **Řešení**

Mohla by paní Kusá pípat dokumenty do krabice na digitalizaci?

---

Založeno: 22.02.2019 10:06 Uživatel: lenka.horakova **Řešení**

Pan Lounský by mohl pomáhat lidem se zorientovat a navést je, kam mají jít.

---

Založeno: 22.02.2019 10:37 Uživatel: sarka.pilna **Řešení**

Mohl by nám někdo pomoci při vybírání žádostí od klientů kopírovat přílohy k žádostem, razítkovat je a dávat na ně štítky.

**Hodnocení řešení:**

V oddělení hodnotilo: 5 Mimo oddělení hodnotilo: 2 Celkem hodnotilo: 7

★★★★☆ 4      ★★☆☆☆ 2      ★★★★★ 3,43

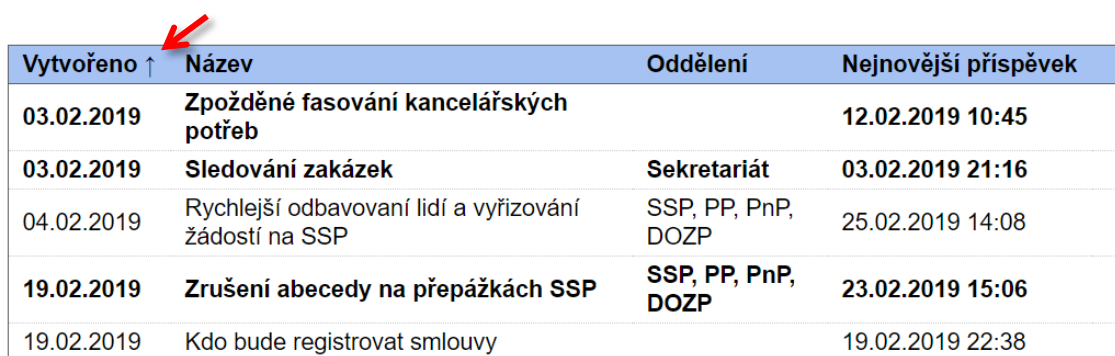
Obrázek 48: GUI – zobrazení vybraných řešení

### 6.2.13 Další funkce

Pokud je někde v aplikaci zobrazena tabulka se záznamy, je možné tyto **záznamy řadit**, a to kliknutím na název příslušného sloupce tabulky. Směr šipky (viz obrázek 49) ukazuje směr řazení dat. Tímto způsobem si lze např. seřadit jednotlivé požadavky podle naposledy vložených příspěvků či seřadit uživatele podle oddělení, ve kterých pracují apod.

Pro lepší komunikaci mezi uživateli se při zobrazení detailu požadavku zobrazí v adresním řádku internetového prohlížeče i příslušné **URL s id požadavku**. Tím je umožněno kopírovat URL z adresního řádku a např. ho zaslat e-mailem jinému uživateli,

který si daný požadavek zobrazí kliknutím na zasláný odkaz. Pokud nebude uživatel aktuálně přihlášen do aplikace, aplikace ho před zobrazením požadavku k tomu vyzve.



| Vytvořeno ↑ | Název   | Oddělení           | Nejnovější příspěvek |
|-------------|---|--------------------|----------------------|
| 03.02.2019  | Zpožděné fasování kancelářských potřeb                |                    | 12.02.2019 10:45     |
| 03.02.2019  | Sledování zakázek                                     | Sekretariát        | 03.02.2019 21:16     |
| 04.02.2019  | Rychlejší odbavování lidí a vyřizování žádostí na SSP | SSP, PP, PnP, DOZP | 25.02.2019 14:08     |
| 19.02.2019  | Zrušení abecedy na přepážkách SSP                     | SSP, PP, PnP, DOZP | 23.02.2019 15:06     |
| 19.02.2019  | Kdo bude registrovat smlouvy                          |                    | 19.02.2019 22:38     |

**Obrázek 49:** GUI – řazení sloupců tabulky

## 7 Validace aplikace

V předchozí kapitole byla představena výsledná aplikace. Ta musí odpovídat daným požadavkům na aplikaci, návrhu a účelu aplikace. V této kapitole tedy bude řešena verifikace a validace navržené aplikace. Účelnost a funkčnost aplikace bude ověřována nejen zpracovatelem práce, ale kvůli objektivnímu pohledu i vybranými zaměstnanci z vybrané organizace.

### 7.1 Verifikace a validace

Verifikace je ověření shody softwarového produktu s jeho specifikací. Tato shoda je ověřována v jednotlivých dílčích produktech. Začíná schválením specifikace požadavků, s kterými jsou následně porovnávány dílčí produkty – od návrhu softwaru až po finální produkt. V jednotlivých dílčích produktech se zjišťují odchylky od dané specifikace, resp. požadavků na software. Podstatou verifikace tedy je, zda je produkt správně vytvářen. [48]

Naopak validace zjišťuje, zda produkt odpovídá požadavkům a očekáváním zákazníka týkající se jeho zamýšleného použití. Začíná již fází plánování, sběru a vývoje specifikace požadavků a pokračuje přes všechny dílčí produkty. Tato kontrola je velmi důležitá, neboť specifikace požadavků se nakonec může lišit od skutečných požadavků a očekávání. Problémy během validace tak mohou vést ke změně specifikace požadavků. Podstatou validace tedy je, zda je vytvářen správný produkt. Zjišťuje tak použitelnost produktu v kontextu jeho zamýšleného použití. [48]

Validace i verifikace byly prováděny v každé fázi vývoje softwaru a nebyly zjištěny žádné odchylky od specifikace, resp. od zamýšleného účelu softwaru. V následující kapitole bude vybrán požadavek, na kterém bude otestován nejen celý proces identifikace požadavků, ale i samotná aplikace.

### 7.2 Výběr požadavku

K otestování aplikace bude vybrán jednoduchý požadavek, který je srozumitelný bez větší znalosti zákonů nebo prostředí ÚP ČR. K tomuto účelu byl vybrán problém s fasováním kancelářských potřeb, a sice požadavek na jejich rychlejší dodávku. Na něm bude ověřen celý proces identifikace požadavků dle první a třetí kapitoly a výběr nejvhodnějšího, resp. nejvhodnějších řešení.

Kancelářské potřeby se vydávají jednou za tři měsíce, a to na základě objednávky jednotlivých oddělení. Objednávka je tvořena zápisem požadovaného počtu kusů nebo balení k jednotlivým položkám zboží do listu vytvořeného v aplikaci Microsoft Excel. Každé oddělení má k tomuto účelu přidělenou svou záložku pojmenovanou podle svého oddělení, kam své požadavky zapisuje.

Po vytvoření objednávky jednotlivými odděleními udělá pracovník ekonomického oddělení její sumář. Následně zkontroluje zásoby na skladu a podle nich upraví objednávku tak, aby po vydání objednaného zboží oddělením vznikla na skladě přiměřená rezerva. Objednávku postoupí krajské pobočce, která jí zašle na MPSV. MPSV shromáždí požadavky z celé ČR a vyhlásí výběrové řízení na dodávku požadovaných kancelářských potřeb. Po té jsou vítězem výběrového řízení postupně objednané kancelářské potřeby dodávány na jednotlivá kontaktní pracoviště ÚP ČR. Po dodávce je zboží rozděleno na jednotlivá oddělení.

### **7.3 Ověření procesu identifikace požadavků**

Pro podání a následné řešení požadavku je nejprve nutné získat o procesu co nejvíce informací (*sběr informací*). Základní informace by měli dotčení pracovníci mít, neboť s daným procesem pracují. Podrobnější informace je možné získat buď osobně formou interview s účastníky procesu, nebo lépe pomocí aplikace, a to formou diskuze, kterou aplikace dovoluje – k jednotlivým návrhům řešení nebo reakcí je možné psát vysvětlení probíhajícího procesu. Výhodou proti osobnímu interview je, že vložené informace jsou viditelné pro všechny uživatele. Tím není dotčena možnost použít jako doplňující metody analýzu písemné dokumentace nebo formu pozorování.

*Generování námětů* je v aplikaci umožněno formou diskusního fóra. Každý z účastníků může podat jakýkoliv návrh, žádný z nich není smazán nebo okamžitě zavržen. Na jednotlivé návrhy lze reagovat a navrhovat jejich další vylepšení či rozšíření. Jednotlivé varianty řešení jsou porovnávány pomocí hodnocení, které může každý účastník využít k vyjádření svého názoru na navržené řešení požadavku. Tato forma generování námětů je blízká *brainstormingu*.

*Třídění a sdružování námětů* je v aplikaci řešeno předpokladem, že uživatelé budou psát jednotlivé návrhy řešení k požadavku, kterého se navržená řešení týkají. Tím vznikne jakási skupina návrhů řešení se společnými vlastnosti, resp. se společným požadavkem na vyřešení konkrétní situace.



*Benchmarking* může být využit k navržení řešení, které úspěšně funguje na určitém oddělení. Toto řešení může sloužit jako vzor pro jiné oddělení. Uživatelé tak mohou v aplikaci sdílet již ověřené postupy, které úspěšně fungují a navrhovat je pro další oddělení či je modifikovat pro jejich potřeby.

*Identifikaci příčin problémů* aplikace neřeší. Ve složitějších případech je vhodné se jí zabývat a i když jí aplikace neřeší, je možné její výsledky do aplikace zaznamenat. V této práci bude použita *Analýza příčin a důsledků*, neboť může odhalit další skryté související problémy, které by se měly řešit a pokusit se na ně najít vhodné řešení. Nalezené problémy tak lze přidat jako další příspěvek, resp. návrh k řešení do aplikace. Tím může být problém řešen v širších souvislostech, než bylo původně zamýšleno, a vybrané řešení může mít větší pozitivní efekt.

Dále se bude práce zabývat *Analýzou problémových vlivů a jejich důsledků*. Tato analýza bude zpracována mimo aplikaci a bude sloužit k ověření navrženého řešení, zda nepřináší větší rizika a tedy negativnější důsledky než předchozí řešení. Zpracovány budou dvě analýzy, jedna pro proces, který probíhal před navržení nového řešení, a druhá pro proces, který by se uskutečnil po výběru a implementaci nejvhodnějšího nového řešení. Tyto analýzy budou následně porovnány. Při porovnání těchto analýz se však musí brát v úvahu další faktory jako např. zmenšení časové náročnosti procesu apod., které analýza neřeší, ale jsou významné pro následné zhodnocení vybraného řešení.

Tato analýza nemusí být zpracována pro všechny zadané požadavky. Velký smysl má především pro ty, kde nasazením změny v procesu může dojít k negativnímu dopadu na jakoukoliv oblast. Pokud je proces komplikovaný s vyšším počtem rizikových úkonů, je namístě zvážit, zda by nebylo lepší použít analýzu FMECA (viz výše), která dovoluje rizika jednotlivých procesů porovnat podle hodnot míry rizika.

Z výše uvedeného je patrné, že navržená aplikace obsáhla celý proces identifikace požadavků kromě identifikace příčin problémů, což bylo i zamýšleno. Identifikace příčin problémů je většinou prováděna jen pro složitější problémy. Její výsledky však lze do aplikace vložit, a to pomocí příspěvků či poznámky. Aplikace je tedy z tohoto pohledu vyhovující.

## 7.4 Průběh procesu v praxi

Nyní bude aplikace ověřena na praktickém příkladu, a to na vybraném požadavku. V aplikaci budou provedeny všechny úkony potřebné k zadání a vyhodnocení řešení požadavku.

### 7.4.1 Nastavení aplikace

V konfiguračním souboru API byl nastaven zákaz kontroly domény e-mailových adres u uživatelů, a to z důvodu snadnějšího testování aplikace. Tato funkce však byla otestována během programování. Dále byla v nastavení povolena registrace uživatelů.

V samotné aplikaci byla nejprve zadána jednotlivá oddělení ÚP ČR, KoP Jablonec nad Nisou, a to pomocí zkratk oddělení. Při zadávání jednotlivých oddělení bylo zjištěno, že není vždy vhodné zachovat organizační strukturu dané organizace. Pokud má oddělení např. pouze jednoho nebo dva zaměstnance (což nastalo na testovaném pracovišti v případě zaměstnanců patřících pod krajskou pobočku), je vhodné spojit více těchto oddělení, které spolu úzce spolupracují, do jednoho. V tomto případě bylo tímto způsobem rozšířeno oddělení sekretariátu. Někdy může být naopak přínosné rozdělit větší oddělení, které zahrnuje více agend, do několika menších oddělení. Jednotlivá oddělení by tedy měla být spíše zakládána podle toho, jak spolu jednotliví zaměstnanci spolupracují, než podle skutečné organizační struktury.

Dále byli v aplikaci založeni uživatelé, resp. se zaregistrovali uživatelé, kteří budou zadávat požadavky, návrhy na jejich řešení a hodnocení řešení. Vybraným uživatelům bylo přidáno oprávnění k vyhodnocení požadavků týkajících se zvolených oddělení. Tím byla aplikace připravena pro vložení požadavku.

### 7.4.2 Zadání požadavku

Pro otestování procesu identifikace požadavků byl do aplikace vložen požadavek *„Kancelářské potřeby jsou dodávány po několika měsících od objednání. Nelze nějakým způsobem celý proces urychlit? Je totiž od nás požadována nová objednávka kanc. potřeb, aniž by byla vyřízena předchozí.“* Jako název byl zadán *„Zpožděné fasování kancelářských potřeb“*.

Pro tento požadavek nebyl dán žádný termín, do kterého je potřeba zadaný požadavek vyřešit. Bylo zadáno, že se bude řešit v rámci celého úřadu, tedy hodnocení všech uživatelů budou rovnocenná. K navrženým řešením se budou také vyjadřovat pracovníci, kteří řeší

dodávání kancelářských potřeb, tj. vyjadřovat se, která řešení jsou reálně možná a která ne. Jelikož budou součástí týmu, který bude vybírat nejvhodnější řešení, nebude potřeba, aby bylo jejich hodnocení jednotlivých řešení zvlášť evidováno.

### **7.4.3 Analýza příčin a důsledků**

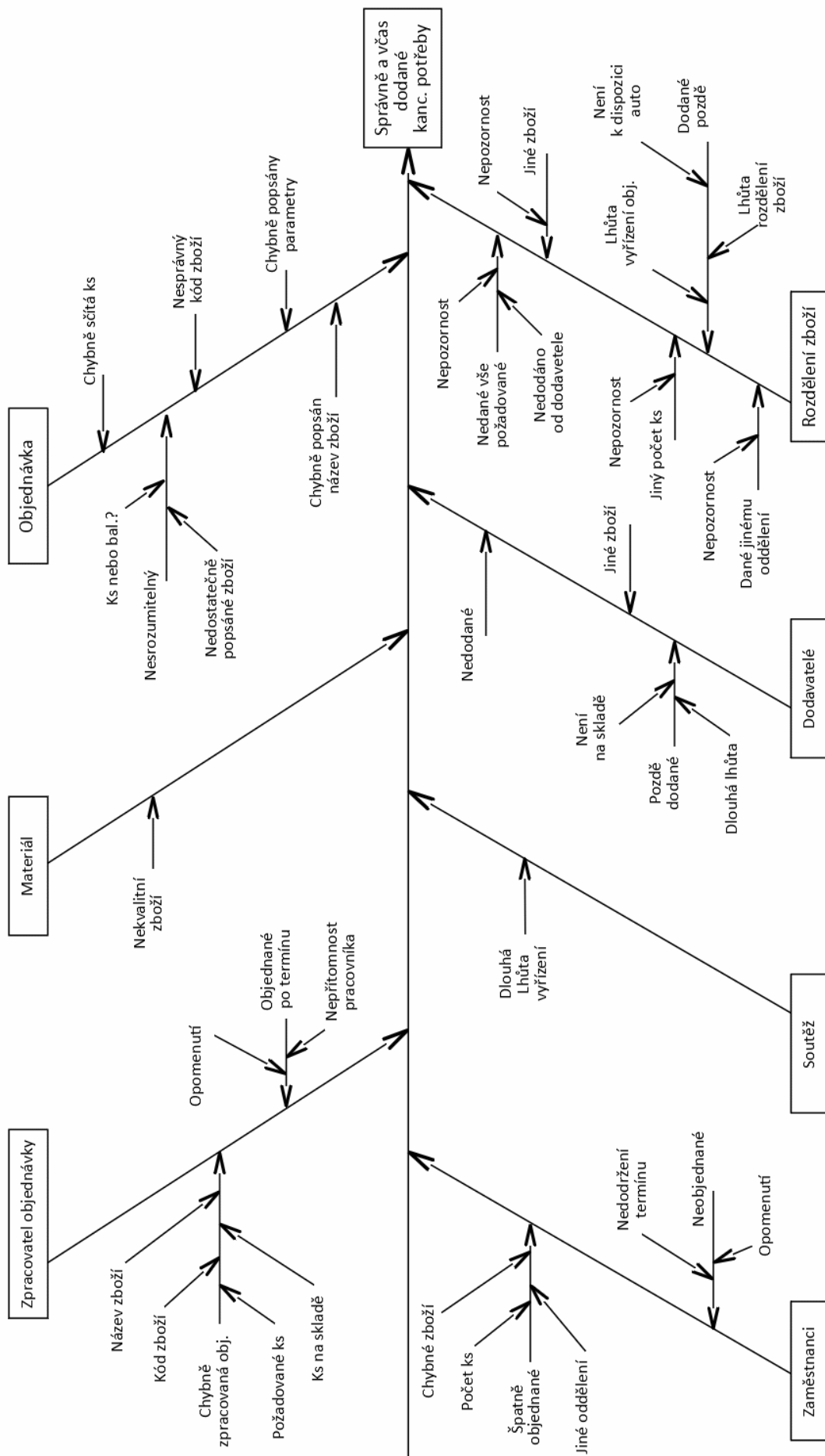
Po zadání požadavku do aplikace je, jak bylo uvedeno výše, v některých případech přínosné zpracovat analýzu příčin a důsledků. Tato analýza byla vytvořena i pro tento požadavek, i když není tak komplikovaný, ale je vhodné jí uvést jako návod pro složitější problémy. Diagram „rybí kosti“ je uveden na obrázku 50. Pro tento požadavek byl diagram rozšířen i na správné dodání kancelářských potřeb, aby byla daná problematika řešena v širších souvislostech.

Jelikož části procesu probíhají na MPSV, není v těchto procesech možné navrhovat jakékoliv změny. Jedná se o kategorie *Soutěž*, *Dodavatelé* a částečně *Materiál*. Na druhou stranu je možné se zabývat kategoriemi *Zaměstnanci*, *Zpracovatel objednávky*, *Objednávka* a *Rozdělení zboží*. Z diagramu je patrné, že v kategorii *Zaměstnanci* je důležité včas a správně vyplnit objednávku. S tím souvisí kategorie *Objednávka*, která musí být, aby mohla být správně vyplněná, dobře srozumitelná, mít jednoznačně a srozumitelně popsané zboží s jasnou informací, zda se objednávají počty kusů či balení. Dále musí být funkční, tzn. správně sčítat počty kusů/balení.

Jelikož není objednávací list vždy všemi zaměstnanci správně pochopen (hlavně rozeznávání, zda se objednávají počty kusů či balení), byl k požadavku o dřívější dodávání kancelářských potřeb přiřazen podnět na vylepšení objednávacího listu, resp. na jeho větší srozumitelnost. Dále je možné řešit oblast pozornosti, s jakou se objednávka vyplňuje, nebo termín pro zpracovatele objednávky pro vytvoření sumáře a pro rozdělení zboží. Přidávat do aplikace náměty na řešení je samozřejmě vhodné jen z problémových oblastí. K navrženému požadavku tedy přibude jen oblast srozumitelnosti objednávacího listu.

### **7.4.4 Navržená řešení**

Průběžně s vytvořením analýzy příčin a důsledků a doplněním do aplikace další problémové oblasti objednávacího listu, byly do aplikace zadávány návrhy řešení uživatelů, byly připisovány různé poznámky nebo byla aplikace využita k dotazu. V aplikaci je také možné vysvětlit stávající situaci nebo důvod, proč je některé řešení nemožné uskutečnit. Při řešení požadavků tak vzniklo diskuzní fórum se všemi jeho aspekty.



Obrázek 50: Analýza příčin a důsledků

Při zadávání návrhů řešení bylo zjištěno, že je uživatele potřeba upozornit na skutečnost, že do navrhovaného řešení nelze psát dvě varianty řešení (tedy např. „mohlo by se toto nebo toto“). Tato formulace znemožňuje jednotlivá řešení ohodnotit, neboť uživatel neví, zda hodnotí variantu 1 nebo variantu 2.

Jak bylo naznačeno výše, navrhovaná řešení se mohou rozšířit do oblasti, která původně v požadavku nebyla zmíněna, ale s požadavkem úzce souvisí či jsou součástí jeho procesu. Taková řešení jsou vkládána k dotčenému požadavku, aby byla všechna související navržená řešení a náměty zadané pod jedním požadavkem a řešily se jako celek. Pokud by byl uživatel požadavkem inspirován k námětu na řešení, které s daným požadavkem nesouvisí, musí být založen požadavek nový. Před jeho založením je však důležité zkontrolovat, zda takový požadavek již neexistuje nebo se již neřešil a tím zabránit zakládání duplicitních požadavků.

#### **7.4.5 Hodnocení řešení**

K jednotlivým návrhům řešení bylo od uživatelů přiděleno hodnocení od 1 do 5. Velkou výhodou je, že pokud uživatel ohodnotil zadané řešení a následně bylo vloženo řešení lepší, mohl uživatel znovu ohodnotit (resp. přehodnotit) původní řešení a snížit tak u něj hodnocení ve prospěch lepšího nově navrženého řešení. Je tak zcela jasné, kterému řešení dává přednost, aniž by musel cokoli vysvětlovat či psát, které hodnocení platí a které již ne, nebo jak změnil svůj názor.

Po uplynutí dostatečného času, ve kterém mohli všichni uživatelé vyjádřit svůj názor hodnocením navržených řešení nebo sami navrhnout nějaká řešení, byl požadavek uzavřen. Tím bylo uživatelům znemožněno vkládat další příspěvky či hodnotit navržená řešení, aby tak nemohla v průběhu vyhodnocování požadavku či provádění analýzy vznikat nová řešení, která by do ní nebyla zahrnuta, nebo měnit podmínky změnou hodnocení jednotlivých řešení.

#### **7.4.6 Analýza problémových vlivů a jejich důsledků**

Jak bylo uvedeno výše, před konečným rozhodnutím, která řešení budou pro daný požadavek nejvhodnější, je v některých případech vhodné zpracovat analýzu problémových vlivů a jejich důsledků. U tohoto požadavku bude analýza zpracována víceméně spíše jako ukázková, neboť požadavek není tak komplikovaný nebo rizikový, aby u něj musela být tato analýza provedena.

V první fázi bude zpracována pro řešený proces, který probíhal před navrhovanými změnami. Jelikož se v rámci aplikace řešily jen procesy objednání zboží zaměstnancem

a rozdání zboží na jednotlivá oddělení, jsou pouze tyto dva procesy řešeny i v rámci zmíněné analýzy. Ve druhé fázi bude analýza zpracována pro nově upravený proces. Porovnáním těchto analýz bude zjištěno, zda po úpravě procesu nedojde v nějaké oblasti k většímu riziku špatného nebo pozdějšího dodání zboží.

**Tabulka 7:** FMEA před zavedením změn

| Č.  | Proces                       | Vznik možné chyby                         | Následky chyby                        | Příčiny chyby                           | Kontroly procesu                          |
|---|------------------------------|---|---------------------------------------|---|---|
| 1   | Objednání zboží zaměstnancem | Objednání jiného zboží                    | Nedodání správného zboží              | Nepozornost                             | V případě nejasnosti volat pověřené osobě |
|   |                              |   |                                       | Nesprávně identifikované zboží          |   |
|   |                              | Objednání chybného počtu ks               | Nedodání správného počtu ks           | Nepozornost                             |   |
|   |                              |   |                                       | Nesprávně spočítané potřebné ks         |   |
|   |                              | Objednán počet kusů místo balení a naopak | Nedodání správného počtu ks           | Nepozornost                             | V případě nejasnosti volat pověřené osobě |
|   |                              |   |                                       | <b>Nepochopení objednávacího listu</b>  |   |
| Zapsání objednávky pod nesprávné oddělení | Nedodání zboží               | Nepozornost                               |                                       |   |   |
| Neobjednání zboží                         | Nedodání zboží               | Nezjištěná potřeba                        |                                       |   |   |
| 2   | Rozdání objednaného zboží    | Dáno jiné zboží                           | Nedodání správného zboží              | Nepozornost                             |   |
|   |                              | Zboží nedodáno                            | Nedodání zboží                        | Nepozornost                             |   |
|   |                              | Dán jiný počet kusů                       | Nedodání správného počtu ks           | Nepozornost                             |   |
|   |                              | Dáno jinému oddělení                      | Nedodání správného zboží              | Nepozornost                             |   |
|   |                              | Dodáno pozdě                              | Objednané zboží není včas k dispozici | Zboží dodáno pozdě od dodavatele        |   |
|   |                              |   |                                       | Rozdělení zboží trvá nepřiměřeně dlouho |   |
| Není k dispozici auto pro rozvoz zboží    |                              |   |                                       |   |   |

Jelikož forma analýzy FMEA může být upravena vzhledem k požadavkům projektu a ve způsobu a prezentaci FMEA existují značné rozdíly [7], byla vytvořena tabulka, která je dostačující k požadovanému účelu a neviduje informace, které by zbytečně časově zatěžovaly zpracovatele analýzy. Kontrola procesu bude vyplněna jen v případě, že se příčina chyby bude vyskytovat i v reálné praxi a bude potřeba ověřit kontrolní mechanismy a popř. navrhnout nějaká opatření. Zmíněná tabulka by měla obsahovat i sloupec *Opatření*. Jelikož však žádné opatření nebylo navrženo, není v tabulce tento sloupec uveden. Analýza procesu před zavedením změn je uvedena v tabulce 7.

Pro rychlejší dodávku kancelářského zboží na jednotlivá oddělení a zmenšení rizika chybně objednaného zboží byly z navržených řešení zadaných do aplikace vybrány k implementaci následující změny v procesu:

- Zboží bude vydáváno ve dvou fázích. V první fázi bude vydáno zboží, které je na skladě, a to ihned po sepsání objednávky. V druhé fázi bude zbytek zboží vydán po jejím dodání z MPSV na kontaktní pracoviště. Tím se zrychlí dodání alespoň některého zboží na jednotlivá oddělení.
- Při výdeji zboží v první fázi bude ke zboží přiložen lístek se seznamem zboží, které bude dodáno dodatečně (tj. ve druhé fázi).
- Do tabulky s objednávacím listem bude přidán sloupec s informací, zda se u objednávaného zboží jedná o počty kusů nebo balení.

Analýza pro navržená řešení je zpracována v tabulce 8. Je patrné, že v tabulce 8 již není příčina chyby „Nepochopení objednávacího listu“, která vedla k chybnému objednání počtu kusů, resp. k záměně počtu kusů s počtem balení. Tím by měly být objednávky méně chybové. Na druhé straně přibyla možnost dodání nesprávného počtu kusů, resp. dodání dvojnásobného počtu kusů z důvodu nově navrženého způsobu dodání zboží, a to ve dvou fázích. Zde nově hrozí riziko, že si pracovník, který kancelářské potřeby rozděljuje, špatně poznamená nebo nepoznamená, jaké zboží již bylo dodáno a jaké má ještě dodat. Výhoda včasného dodání alespoň části zboží na jednotlivá oddělení bude tedy vykoupena potřebou větší pozornosti pracovníka, který zboží vydává, což by ovšem neměl být problém. Jelikož navržený proces v sobě neskrývá větší riziko, budou navržená řešení přijata.

**Tabulka 8:** FMEA s navrženými řešeními

| Č.                                     | Proces                       | Vznik možné chyby                         | Následky chyby                        | Příčiny chyby                                      | Kontroly procesu                          |
|--|------------------------------|---|---------------------------------------|--|---|
| 1                                      | Objednání zboží zaměstnancem | Objednání jiného zboží                    | Nedodání správného zboží              | Nepozornost  | V případě nejasnosti volat pověřené osobě |
|  |                              |   |                                       | Nesprávně identifikované zboží                     |   |
|  |                              | Objednání chybného počtu ks               | Nedodání správného počtu ks           | Nepozornost  | Nesprávně spočítané potřebné ks           |
|  |                              |   |                                       | Nesprávně spočítané potřebné ks                    |   |
|  |                              | Objednán počet kusů místo balení a naopak | Nedodání správného počtu ks           | Nepozornost  |   |
|  |                              | Zapsání objednávky pod nesprávné oddělení | Nedodání zboží                        | Nepozornost  |   |
| Neobjednání zboží                      | Nedodání zboží               | Nezjištěná potřeba                        |                                       |  |   |
| 2                                      | Rozdání objednaného zboží    | Dáno jiné zboží                           | Nedodání správného zboží              | Nepozornost  |   |
|  |                              | Zboží nedodáno                            | Nedodání zboží                        | Nepozornost  |   |
|  |                              | Dán jiný počet kusů                       | Nedodání správného počtu ks           | Nepozornost  |   |
|  |                              | Dáno jinému oddělení                      | Nedodání správného zboží              | Nepozornost  |   |
|  |                              | <b>Zboží dáno dvakrát</b>                 | <b>Dodání nesprávného počtu</b>       | <b>Nepoznamenání si, které zboží bylo již dáno</b> |   |
|  |                              | Dodáno pozdě                              | Objednané zboží není včas k dispozici | Zboží dodáno pozdě od dodavatele                   |   |
|  |                              |   |                                       | Rozdělení zboží trvá nepřiměřeně dlouho            |   |
| Není k dispozici auto pro rozvoz zboží |                              |   |                                       |  |   |



### **7.4.7 Vyhodnocení nejvhodnějšího řešení**

Konečné rozhodnutí tak může být zaznamenáno do aplikace, a to označením vybraných řešení, tj. dodání zboží ve dvou fázích, vložením lístečku se seznamem nedodaného zboží a úpravou objednávacího listu. Do poznámky byla zaznamenána skutečnost o výběru řešení a termín, do kterého má být objednávací list upraven, resp. vložen požadovaný sloupec s poznámkou, zda se při objednávání zboží jedná o kusy či o balení.

Při označování vybraných řešení vyšlo najevo, že pokud by byla statistika úspěšných řešitelů použita vedoucími oddělení jako jeden z podkladů při hodnocení zaměstnanců, tak nemusí být plně vypovídající. Větší šanci navrhnout řešení, které bude následně použito v praxi, mají totiž zaměstnanci, kteří jsou do dotčeného procesu přímo zapojeni a znají všechny souvislosti a možnosti, než ti, kteří do procesu zapojeni nejsou. Podmínky tak nejsou pro všechny zaměstnance rovnocenné. V tomto procesu má tak daleko větší šanci získat kladné body pracovník, který objednávky zpracovává a zboží rozdává. O něco menší šanci má pracovník, který vytváří objednávku za oddělení. Ostatní pracovníci jsou pak v tomto smyslu znevýhodněni nejvíce. Proto by se měla statistika úspěšných řešitelů brát s nadhledem nebo se vždy podívat za jakých okolností jednotlivá řešení vznikla a k tomuto přihlídnout.

### **7.4.8 Zhodnocení otestování aplikace na daném požadavku**

Navržená aplikace byla otestována v celém rozsahu – od nastavení aplikace, přes zadání vybraného požadavku a navržených řešení, včetně jejich hodnocení, až po výběr nejvhodnějších řešení. Po jejím otestování bylo zjištěno, že v aplikaci lze provést všechny potřebné kroky. Zjištěny nebyly žádné situace, které by nešly v aplikaci zadat či vyřešit. Uživatelé si musí pouze dávat pozor, aby do jednoho řešení požadavku nepsali dvě varianty – v tom případě pak nelze řešení správně ohodnotit. Opatrní musí být i vedoucí oddělení při interpretaci statistiky nejlepších řešitelů.

## **7.5 Další využití aplikace**

Při dalším testování aplikace bylo zjištěno, že se dá využít i v jiných oblastech, nejen pro navrhování řešení a jejich hodnocení. Jedním z testovaných požadavků bylo zjišťování, zda je pro zaměstnance lepší dělat na pracovišti revizi elektrických spotřebičů (což znamená vypnutí počítačů) ve čtvrtek nebo v pátek. Aplikace tak byla využita pro jednoduché hlasování, které se uskutečnilo prostřednictvím hodnocení názoru „čtvrtek“ versus „pátek“.

Aplikace je vhodná i pro ujasnění určitých postupů v rámci daného procesu. Její použitelnost je dána víceméně na domluvě a stanovení pravidel, jak se bude a k jakým účelům využívat. V případě potřeby může sloužit i k veřejným dotazům či jiným účelům.

## **7.6 Návrhy a připomínky zaměstnanců zvolené organizace**

Navržená aplikace byla otestována i vybranými zaměstnanci zvolené organizace. K otestování byli vybráni čtyři zaměstnanci včetně jednoho vedoucího oddělení. Nejprve byli seznámeni s navrženou aplikací a byl jim poskytnut manuál k aplikaci. Testování bylo prováděno za přítomnosti zpracovatele práce a bylo sledováno, jak se vybraní zaměstnanci rychle naučili s aplikací pracovat a jak je pro ně srozumitelná a použitelná.

Zaměstnanci neměli s ovládním aplikace žádné problémy a byli schopni provádět všechny akce, které chtěli. K testované aplikaci neměli žádné připomínky ani náměty na její zlepšení. Aplikaci vyhodnotili jako přínosnou. Ocenili možnost okamžitého sdělení problému či dotazu a vyjádření se k navrhovaným řešením. Jako výhodu uvedli evidenci odpovědí a vybraných řešení na dotaz či námět a tím i možnost si tyto informace kdykoliv vyhledat. Jako přínosné se jim zdálo i zjištění názoru ostatních zaměstnanců na daný požadavek.

Diskuze vznikla k anonymitě uživatelů – zda se přihlašovat pod jiným jménem či nikoliv. To se ukázalo jako závislé na vztazích mezi zaměstnanci, otevřeností vedoucích pracovníků a smělostí jednotlivých zaměstnanců. Otázky nastaly i k rozsahu používání aplikace – ten bude nejspíše postupně upravován se získávanou zkušeností práce s aplikací.

## 8 Závěr

Cílem této práce bylo zachycení změnových požadavků a jejich zpracování v rámci organizace. Nejprve tedy byly prozkoumány nástroje, které pomáhají požadavky identifikovat. Dále byla pro ověřování diplomové práce určena taková organizace veřejné správy, kde z důvodu její velikosti není možné osobní sdílení informací. Touto organizací byl zvolen Úřad práce ČR, Kontaktní pracoviště Jablonec nad Nisou Krajské pobočky v Liberci. Pomocí této organizace byla zjištěna povaha vzniklých požadavků a způsob jejich řešení.

Jelikož mělo být při sběru požadavků a jejich zpracování využito sdílení nápadů a zkušeností, byly vzniklé požadavky v organizaci prozkoumány i z tohoto pohledu. Uvedeny byly i nevýhody současného způsobu podávání a řešení požadavků. Ty spočívaly hlavně v tom, že do podávání požadavků a sdílení nápadů a zkušeností nebyli zapojeni všichni zaměstnanci, proto byly nastíněny i jiné možnosti způsobu sběru a řešení požadavků. Jako nejvhodnější pro sběr požadavků a nápadů se ukázalo navržení aplikace, kterým se práce dále zabývala.

Snahou bylo, aby navržená aplikace využila co nejvíce možností, jak získat co největší množství požadavků a návrhů na jejich vyřešení. K tomuto účelu byly využity nastudované nástroje, které slouží k identifikaci požadavků. Z nich pak byly vybrány ty, které byly vhodné pro návrh aplikace. Dále byly vybrány nástroje, které se sice k návrhu zamýšlené aplikace nehodily, ale byly vhodné pro proces získání dalších možných požadavků nebo pro následné prověření vybraných řešení požadavků.

Ze získaných informací ze zvolené organizace a vybraných nástrojů pro identifikaci požadavků byly sestaveny požadavky na navrženou aplikaci. Tyto požadavky byly dále rozvinuty do návrhu aplikace. Tento návrh byl následně podroben úvaze o vzniku možných situací, které by mohly v aplikaci nastat a na tyto situace byly nalezeny řešení. Tím dostal návrh konkrétnější podoby. K tomuto návrhu byl zpracován grafický návrh aplikace a sepsány posloupnosti kroků, které musí zajišťovat uživatel a systém při konkrétním užití aplikace.

Práce dále pokračovala datovým modelováním. Nejprve byl navržen konceptuální model databáze, který byl následně rozšířen do technologického modelu. Po výběru databázového serveru mohl být zpracován implementační model a celý návrh aplikace přešel do fáze implementace, která zahrnovala jak založení databáze, tak i samotné naprogramování aplikací, které bylo uděláno nad rámec práce. Pro větší využití aplikace bylo vytvořeno jak aplikační programové rozhraní, tak i grafické uživatelské rozhraní.

Výsledné aplikace byly v práci dostatečně popsány a tento popis byl využit i jako návod na použití aplikací. Důležitou částí práce byla verifikace a validace navržené aplikace, které byly prováděny v průběhu celé práce. Konečná validace aplikace byla provedena na vybraném požadavku, který byl do aplikace založen a na kterém byl ověřen i celý proces identifikace požadavků. Vyzkoušet a ověřit účelnost aplikace měli možnost po seznámení se s aplikací i zaměstnanci vybrané organizace. Velká pozornost byla věnována jejich případným připomínkám k aplikaci nebo sebemenším postřehům.

Navržená aplikace se ukázala pro sběr a zpracování požadavků jako zcela vyhovující. Testováním bylo ověřeno, že k tomuto účelu umožňuje provést všechny potřebné kroky a využít se dá i k jiným účelům. Jelikož v prostředí veřejné správy aplikace tohoto typu chybí, po jejím nasazení do tohoto prostředí by se rozšířily možnosti, jak by zaměstnanci mohli jednoduše sdělit své požadavky nebo navrhopat různé náměty na zlepšení jak procesů, které vykonávají, tak i jejich pracovního prostředí či jiných věcí.

Největší předností aplikace a zároveň přínosem předložené práce je tedy vzájemné sdílení nejen požadavků, ale i nápadů, námětů, zkušeností a návrhů řešení těchto požadavků či nastalých problémů mezi všemi zaměstnanci. Tím se zvětší šance, že bude vybráno nejvhodnější řešení požadavků, neboť navržená řešení by nepocházela pouze od vedoucích pracovníků, ale od všech zaměstnanců, kteří mohou mít neocenitelné nápady. Aplikace tak daleko více využívá potenciálu každého zaměstnance a při jejím používání by měli i zaměstnanci pocít větší sounáležitosti.

## Použitá literatura

- [1] *Analýza možných způsobů a důsledků poruch (FMEA): referenční příručka*. 4. vyd. Přeložil Ivana PETRAŠOVÁ. Praha: Česká společnost pro jakost, 2008. ISBN 978-80-02-02101-8.
- [2] ARLOW, Jim a Ila NEUSTADT. *UML 2 a unifikovaný proces vývoje aplikací: objektově orientovaná analýza a návrh prakticky*. 2., aktualiz. a dopl. vyd. Brno: Computer Press, 2007. ISBN 978-80-251-1503-9.
- [3] Boolean. *MariaDB*. [online]. © 2018 [vid. 23. 9. 2018]. Dostupné z: <https://mariadb.com/kb/en/library/boolean/>
- [4] BORONCZYK, Tim. *MySQL okamžitě*. Přeložil Milan DANĚK. Brno: Computer Press, 2016. ISBN 978-80-251-4737-5.
- [5] CUESTA, Hector. *Analýza dat v praxi*. Brno: Computer Press, 2015. ISBN 978-80-251-4361-2.
- [6] ČEGAN, Lukáš. *Vývoj webových aplikací v PHP a NetBeans*. Pardubice: Univerzita Pardubice, 2015. ISBN 978-80-7395-858-9.
- [7] ČSN EN 60812 (010675). *Techniky analýzy bezporuchovosti systémů – Postup analýzy způsobů a důsledků poruch (FMEA)*. Praha: Český normalizační institut, 2006.
- [8] Data Types. *MariaDB*. [online]. © 2018 [vid. 23. 9. 2018]. Dostupné z: <https://mariadb.com/kb/en/library/data-types/>
- [9] Databáze. *Adaptic*. [online]. © 2005–2018 [vid. 22. 9. 2018]. Dostupné z: <http://www.adaptic.cz/znalosti/slovnicek/databaze/>
- [10] Download Postman App. *Postman*. [online]. © 2018 [vid. 16. 12. 2018]. Dostupné z: <https://www.getpostman.com/downloads/>
- [11] DYBKA, Patrycja. ERD Notations in Data Modeling. Part 6 – Crow’s Foot Notation. *Vertabelo*. [online]. 31. 3. 2016 [vid. 23. 9. 2018]. Dostupné z: <http://www.vertabelo.com/blog/technical-articles/crow-s-foot-notation>
- [12] EICHLER, Tomáš. Nebojte se FMEA. *Kvalita jednoduše.cz*. [online]. 10. 1. 2016 [vid. 9. 9. 2018]. Dostupné z: <http://kvalita-jednoduse.cz/fmea>
- [13] FMEA. *Process quality management*. [online]. © 2018 [vid. 9. 9. 2018]. Dostupné z: <http://pqm.cz/NVCSS/fmea.html>
- [14] FOWLER, Martin. *Destilované UML*. Praha: Grada Publishing, 2009. Knihovna programátora. ISBN 978-80-247-2062-3.
- [15] GitHub - jquery/Stupid-Table-Plugin: A stupidly small and simple jQuery table sorter plugin. *GitHub*. [online]. 24. 2. 2018 [vid. 8. 12. 2018]. Dostupné z: <https://github.com/jquery/Stupid-Table-Plugin>

- [16] GitHub - tuupola/slim-basic-auth: PSR-7 HTTP Basic Authentication Middleware. *GitHub*. [online]. 31. 8. 2018 [vid. 12. 10. 2018].  
Dostupné z: <https://github.com/tuupola/slim-basic-auth>
- [17] HASSMAN, Martin. JSON : jednotný formát pro výměnu dat. *Zdroják.cz*. [online]. 29. 9. 2008 [vid. 9. 10. 2018].  
Dostupné z: <https://www.zdrojak.cz/clanky/json-jednotny-format-pro-vymenu-dat/>
- [18] HTML5: co přináší a proč se o něj zajímat. *ROOT.cz*. [online]. 29. 8. 2012 [vid. 27. 10. 2018].  
Dostupné z: <https://www.root.cz/clanky/html5-co-prinasi-a-proc-se-o-nej-zajimat>
- [19] HTTP Status Codes. *REST API Tutorial* [online]. [vid. 7. 10. 2018].  
Dostupné z: <http://www.restapitutorial.com/httpstatuscodes.html>
- [20] JADRNÝ, Tomáš. jQuery návod - vše okolo jQuery. *jQuery návod*. [online]. © 2010 [vid. 24. 11. 2018]. Dostupné z: <http://jquery-navod.cz/kategorie-jquery-ui>
- [21] JADRNÝ, Tomáš. jQuery návod - vše okolo jQuery. *jQuery návod*. [online]. © 2010 [vid. 28. 10. 2018].  
Dostupné z: <http://jquery-navod.cz/kategorie-ostatni-clanky/1-uvodni-clanek>
- [22] JAHODA, Bohumil. AJAX. *Je čas*. [online]. 16. 11. 2013 [vid. 10. 11. 2018].  
Dostupné z: <http://jecas.cz/ajax>
- [23] JAHODA, Bohumil. JSON. *Je čas*. [online]. 6. 1. 2016 [vid. 9. 10. 2018].  
Dostupné z: <http://jecas.cz/json>
- [24] JAHODA, Bohumil. Základy CSS. *Je čas*. [online]. 1. 2. 2016 [vid. 27. 10. 2018].  
Dostupné z: <http://jecas.cz/css-zaklady>
- [25] jQuery. *jQuery* [online]. © 2018 [vid. 28. 10. 2018]. Dostupné z: <https://jquery.com>
- [26] jQuery UI. *jQuery user interface* [online]. © 2018 [vid. 24. 11. 2018].  
Dostupné z: <https://jqueryui.com>
- [27] JSON. *Úvod do JSON* [online]. [vid. 9. 10. 2018].  
Dostupné z: <https://www.json.org/json-cz.html>
- [28] KALUŽA, Jindřich a Ludmila KALUŽOVÁ. *Modelování dat v informačních systémech*. Praha: Ekopress, 2012. ISBN 978-80-86929-81-1.
- [29] KANISOVÁ, Hana a Miroslav MÜLLER. *UML srozumitelně*. Brno: Computer Press, 2004. ISBN 80-251-0231-9.
- [30] Kardinalita vztahu. *Informační technologie*. [online]. © 2018 [vid. 23. 9. 2018].  
Dostupné z: <http://informacni-technologie.studentske.cz/2009/02/kardinalita-vztahu.html>
- [31] KONEČNÝ, Martin. Obrana proti útoku SQL injection v PHP. *ITnetwork*. [online]. [2014] [vid. 27. 10. 2018]. Dostupné z: <https://www.itnetwork.cz/php/bezpecnost/tutorial-bezpecnost-v-php-utok-sql-injection-a-obrana>

- [32] Kontakty na zaměstnance. *Integrovaný portál MPSV*. [online]. [vid. 12. 9. 2018].  
Dostupné z: <https://portal.mpsv.cz/web/upcr-lbk/kop/jablonec-nad-nisou/orgstr>
- [33] LÁSLÓ, Petr. Ajax – úvod. *Programujte.com*. [online]. 26. 6. 2008 [vid. 10. 11. 2018].  
Dostupné z: <http://programujte.com/clanek/2008062101-ajax-uvod/>
- [34] Lekce 1 - Úvod do AJAXu. *ITnetwork*. [online]. [2015] [vid. 10. 11. 2018].  
Dostupné z: <https://www.itnetwork.cz/javascript/ajax/uvod-do-ajaxu>
- [35] Lockhart, Josh, Andrew Smith, Rob Allen. Slim Framework. *Slim Framework*. [online].  
16. 9. 2018 [vid. 7. 10. 2018]. Dostupné z: <https://www.slimframework.com>
- [36] MALÝ, Martin. REST: architektura pro webové API. *Zdroják*. [online]. 3. 8. 2009  
[vid. 7. 10. 2018].  
Dostupné z: <https://www.zdrojak.cz/clanky/rest-architektura-pro-webove-api/>
- [37] NOVÁK, David. Kapitola 2: Model vztahu mezi entitami. *Fakulta informatiky Masarykovy univerzity*. [online]. 21. 9. 2005 [vid. 23. 9. 2018].  
Dostupné z: <https://www.fi.muni.cz/~xnovak8/teaching/PB154/pb154-cesky-02>
- [38] O úřadu práce. *Integrovaný portál MPSV*. [online]. [vid. 12. 9. 2018].  
Dostupné z: <https://portal.mpsv.cz/upcr/oup>
- [39] Open Source Database (RDBMS) for the Enterprise. *MariaDB*. [online]. © 2018  
[vid. 23. 9. 2018]. Dostupné z: <https://mariadb.com>
- [40] Pareto diagram. *Svět Produktivity*. [online]. © 2012 [vid. 8. 9. 2018].  
Dostupné z: <http://www.svetproduktivity.cz/slovník/Pareto-diagram.htm>
- [41] Paretův diagram. *Ikvalita.cz*. [online]. [2009] [vid. 8. 9. 2018].  
Dostupné z: <http://www.ikvalita.cz/tools.php?ID=27>
- [42] PHP /základy/. *Tvorba-webu.cz*. [online]. © 2003–2008 [vid. 29. 9. 2018].  
Dostupné z: <https://www.tvorba-webu.cz/php/>
- [43] PHP: Hypertext Preprocessor. *PHP*. [online]. 13. 9. 2018 [vid. 28. 9. 2018].  
Dostupné z: <http://php.net>
- [44] POKORNÝ, Jaroslav a Michal VALENTA. *Databázové systémy: vysokoškolská učebnice*. Praha: České vysoké učení technické v Praze, 2013.  
ISBN 978-80-01-05212-9.
- [45] Postman Docs. *Postman* [online]. © 2018 [vid. 16. 12. 2018].  
Dostupné z: <https://www.getpostman.com/docs>
- [46] RAYAN, C. John. Data Modelling using ERD with Crow Foot Notation. *Code project*.  
[online]. 20. 2. 2015 [vid. 25. 9. 2018]. Dostupné z: <https://www.codeproject.com/Articles/878359/Data-modelling-using-ERD-with-Crow-Foot-Notation>
- [47] REST API Tutorial. *REST API Tutorial* [online]. [vid. 7. 10. 2018].  
Dostupné z: <http://www.restapitutorial.com>

- [48] ROUDENSKÝ, Petr. *Kvalita softwaru: teorie a praxe*. Prostějov: Computer Media, 2016. ISBN 978-80-7402-294-4.
- [49] SQL Tutorial. *w3schools.com*. [online]. © 1999–2018 [vid. 29. 9. 2018].  
Dostupné z: <https://www.w3schools.com/sql>
- [50] STŘELEČ, Jiří. Pareto analýza. *Vlastní cesta*. [online]. 23. 4. 2012 [vid. 8. 9. 2018].  
Dostupné z: <https://www.vlastnicesta.cz/metody/pareto-analyza/>
- [51] SVOZILOVÁ, Alena. *Zlepšování podnikových procesů*. Praha: Grada, 2011. Expert. ISBN 978-80-247-3938-0.
- [52] ŠIMONOVÁ, Stanislava. *Databázové systémy I*. Pardubice: Univerzita Pardubice, 2013. ISBN 978-80-7395-702-5.
- [53] ŠIMONOVÁ, Stanislava. *Procesní řízení*. Pardubice: Univerzita Pardubice, 2014. ISBN 978-80-7395-766-7.
- [54] WELLING, Luke a Laura THOMSON. *Mistrovství PHP a MySQL*. Přeložil Ondřej BAŠE. Brno: Computer Press, 2017. ISBN 978-80-251-4892-1.
- [55] Základy relačních databází, jejich využití v programování webu. *VŠE* [online]. © 2000–2018 [vid. 22. 9. 2018].  
Dostupné z: <https://gml.vse.cz/data/oppa-webdesign/zaklady-db.html>
- [56] 10 nejlepších PHP frameworků pro vývojáře. *Interval.cz*. [online]. 29. 10. 2015 [vid. 6. 10. 2018]. Dostupné z: <https://www.interval.cz/clanky/10-nejlepsich-php-frameworku-pro-vyvojare/>



## Příloha A – Scénáře případů užití

**Tabulka 9:** Případ užití: Registrace uživatele

| Krok | Role        | Akce  |
|------|-------------|---|
| 1    | Zaměstnanec | Z menu zvolí registraci uživatele.  |
| 2    | System      | Zobrazí formulář pro registraci uživatele a naplní rozbalovací seznam aktuálními odděleními.  |
| 3    | Zaměstnanec | Vyplní formulář (uživatelské jméno, e-mailovou adresu, oddělení a příznak zaslání e-mailu při založení nového požadavku).                           |
| 4    | Zaměstnanec | Buď registraci uloží (a), nebo ji zruší (b).  |
| 5a   | System      | Zkontroluje povinně vyplněná pole – v případě jejich nevyplnění na tuto skutečnost upozorní uživatele a data neuloží (konec).                       |
| 6a   | System      | Zkontroluje formát e-mailové adresy, popř. doménu – pokud je chybný/á, upozorní na to uživatele a data neuloží (konec).                             |
| 7a   | System      | Zkontroluje duplicitu uživatelského jména nebo e-mailové adresy – v případě duplicity na tuto skutečnost upozorní uživatele a data neuloží (konec). |
| 8a   | System      | Vygeneruje heslo a zašle ho na uvedenou e-mailovou adresu.  |
| 9a   | System      | Uloží data uživatele.   |
| 10a  | System      | Zjistí e-mailové adresy uživatelů s oprávněním správce systému a zašle jim e-mail s informací o nové registraci.                                    |
| 11a  | System      | Informuje uživatele o výsledku registrace a zaslání hesla na e-mailovou adresu.   |
| 12a  | System      | Přesměruje stránku na přihlášení uživatele.   |
| 5b   | System      | Smaže vyplněná pole formuláře.  |

**Tabulka 10:** Případ užití: Změna hesla uživatelem

| Krok | Role        | Akce  |
|------|-------------|---|
| 1    | Zaměstnanec | Z menu zvolí změnu hesla.   |
| 2    | System      | Zobrazí formulář pro změnu hesla uživatele.   |
| 3    | Zaměstnanec | Vyplní formulář (původní heslo a dvakrát nové heslo).   |
| 4    | Zaměstnanec | Buď změnu uloží (a), nebo jí zruší (b).   |
| 5a   | System      | Zkontroluje povinně vyplněná pole – v případě jejich nevyplnění na tuto skutečnost upozorní uživatele a data neuloží (konec). |
| 6a   | System      | Zkontroluje shodu zadaných hesel – pokud se neshodují, upozorní na to uživatele a data neuloží (konec).                       |
| 7a   | System      | Uloží změnu hesla.  |
| 8a   | System      | Smaže formulář a informuje uživatele o výsledku.  |
| 5b   | System      | Smaže vyplněná pole formuláře.  |

**Tabulka 11:** Příklad užití: Reset hesla uživatelem

| Krok | Role        | Akce   |
|------|-------------|--|
| 1    | Zaměstnanec | Z menu zvolí přihlášení.   |
| 2    | Systém      | Zobrazí formulář pro přihlášení uživatele a zaslání hesla.   |
| 3    | Zaměstnanec | Vyplní e-mailovou adresu a zvolí zaslání nového hesla.   |
| 4    | Systém      | Zkontroluje vyplněné pole – v případě jeho nevyplnění na tuto skutečnost upozorní uživatele (konec). |
| 5    | Systém      | Zkontroluje formát e-mailové adresy – pokud je chybný, upozorní na to uživatele (konec).             |
| 6    | Systém      | Vygeneruje nové heslo a uloží ho.  |
| 7    | Systém      | Zašle uživateli e-mail s vygenerovaným heslem.   |

**Tabulka 12:** Příklad užití: Založení požadavku

| Krok | Role        | Akce  |
|------|-------------|---|
| 1    | Zaměstnanec | Vybere položku Nový z menu Požadavky.   |
| 2    | Systém      | Zobrazí formulář pro založení požadavku a naplní rozbalovací seznam aktuálními odděleními.  |
| 3    | Zaměstnanec | Vyplní formulář (název požadavku, stručný popis požadavku, oddělení, kterého se požadavek týká, popř. termín, do kterého by se měl požadavek vyřešit).            |
| 4    | Zaměstnanec | Zvolí uložení požadavku (a) nebo zrušení (b).   |
| 5a   | Systém      | Zkontroluje povinně vyplněná pole – v případě jejich nevyplnění na tuto skutečnost upozorní uživatele a data neuloží (konec).                                     |
| 6a   | Systém      | Uloží data s aktuálním datem a časem.   |
| 7a   | Systém      | Zjistí, kteří uživatelé chtějí být informováni o založení požadavku a jsou aktivní, zjistí jejich e-mailové adresy a zašle dotyčným uživatelům informační e-mail. |
| 8a   | Systém      | Informuje uživatele o uložení požadavku a smaže vyplněná data z formuláře.  |
| 5b   | Systém      | Smaže vyplněná pole formuláře.  |

**Tabulka 13:** Příklad užití: Vložení řešení požadavku / reakce na dané řešení

| Krok | Role        | Akce   |
|------|-------------|--|
| 1    | Zaměstnanec | Zvolí položku Seznam požadavků z menu Požadavky.                           |
| 2    | Systém      | Zobrazí výběrové podmínky pro zobrazení seznamu požadavků.                 |
| 3    | Zaměstnanec | Vyplní výběrové podmínky.  |
| 4    | Systém      | Zobrazí požadovaný seznam požadavků dle zvolených výběrových podmínek.     |
| 5    | Zaměstnanec | Vybere neuzavřený požadavek.   |
| 6    | Systém      | Zjistí, zda má uživatel oprávnění vyhodnocovat řešení vybraného požadavku. |

| Krok | Role        | Akce  |
|------|-------------|---|
| 7    | System      | Zobrazí detail požadavku s příslušnými řešeními a všemi přístupnými volbami podle zjištěného oprávnění uživatele a podle stavu požadavku (tlačítko pro vložení řešení je viditelné jen u neuzavřeného požadavku). |
| 8    | Zaměstnanec | Vyhledá příspěvek, na který chce reagovat a zvolí „Navrhnout řešení“ nebo „Reagovat“.   |
| 9    | System      | Zobrazí textové pole pro zadání návrhu řešení nebo reakce na řešení spolu s tlačítkem pro uložení dat.  |
| 10   | Zaměstnanec | Vyplní návrh řešení nebo reakci na navržené řešení a uloží ho (a) nebo pro zrušení zvolí opět „Navrhnout řešení“, resp. „Reagovat“ (b).   |
| 11a  | System      | Zkontroluje vyplněné pole – v případě jeho nevyplnění na tuto skutečnost upozorní uživatele a data neuloží (konec).   |
| 12a  | System      | Uloží řešení, resp. reakci na navržené řešení a skryje pole pro vyplnění návrhu řešení, resp. reakce na navržené řešení.  |
| 11b  | System      | Skryje pole pro vyplnění návrhu řešení, resp. reakce na navržené řešení.  |

**Tabulka 14:** Příklad užití: Vložení hodnocení řešení

| Krok | Role        | Akce   |
|------|-------------|--|
| 1    | Zaměstnanec | Zvolí položku Seznam požadavků z menu Požadavky.   |
| 2    | System      | Zobrazí výběrové podmínky pro zobrazení seznamu požadavků.   |
| 3    | Zaměstnanec | Vyplní výběrové podmínky.  |
| 4    | System      | Zobrazí požadovaný seznam požadavků dle zvolených výběrových podmínek.   |
| 5    | Zaměstnanec | Vybere neuzavřený požadavek.   |
| 6    | System      | Zjistí, zda má uživatel oprávnění vyhodnocovat řešení vybraného požadavku.   |
| 7    | System      | Zobrazí detail požadavku s příslušnými řešeními a všemi přístupnými volbami podle zjištěného oprávnění uživatele a podle stavu požadavku (hodnocení je umožněno jen u neuzavřeného požadavku). |
| 8    | Zaměstnanec | Vyhledá řešení, které chce hodnotit.   |
| 9    | Zaměstnanec | Ohodnotí řešení výběrem počtu hvězdiček.   |
| 10   | System      | Zjistí, zda uživatel již toto řešení hodnotil (a) nebo ne (b).   |
| 11a  | System      | Opraví (změní) původní hodnocení uživatele.  |
| 11b  | System      | Uloží nové hodnocení.  |
| 12   | System      | Zjistí, jakého oddělení se požadavek týká a z jakého oddělení jsou hodnotící uživatelé.  |
| 13   | System      | Vypočte počty jednotlivých hodnocení pro každou skupinu (hodnocení v oddělení, mimo oddělení a celkem) a pro daný počet hvězdiček a vypočítá průměrné hodnocení pro každou skupinu.            |
| 14   | System      | Znovu načte grafické a číselné zobrazení hodnocení k řešení.   |

**Tabulka 15:** Příklad užití: Uzavření požadavku

| Krok | Role    | Akce   |
|------|---------|--|
| 1    | Vedoucí | Zvolí položku Seznam požadavků z menu Požadavky.   |
| 2    | System  | Zobrazí výběrové podmínky pro zobrazení seznamu požadavků.   |
| 3    | Vedoucí | Vyplní výběrové podmínky.  |
| 4    | System  | Zobrazí požadovaný seznam požadavků dle zvolených výběrových podmínek.   |
| 5    | Vedoucí | Vybere neuzavřený požadavek.   |
| 6    | System  | Zjistí, zda má uživatel oprávnění vyhodnocovat řešení vybraného požadavku.   |
| 7    | System  | Zobrazí detail požadavku s příslušnými řešeními a všemi přístupnými volbami podle zjištěného oprávnění uživatele a podle stavu požadavku – tlačítko na uzavření požadavku je viditelné jen oprávněným uživatelům (vedoucím oddělení) a u neuzavřených požadavků. |
| 8    | Vedoucí | Uzavře požadavek.  |
| 9    | System  | Uloží aktuální datum a čas uzavření požadavku.   |
| 10   | System  | Znovu načte detail požadavku s příslušnými řešeními a všemi volbami, které jsou přístupné pro uzavřený požadavek spolu s informací o uzavření požadavku.   |

**Tabulka 16:** Příklad užití: Vložení / oprava poznámky k řešení požadavku

| Krok | Role    | Akce  |
|------|---------|---|
| 1    | Vedoucí | Zvolí položku Seznam požadavků z menu Požadavky.  |
| 2    | System  | Zobrazí výběrové podmínky pro zobrazení seznamu požadavků.  |
| 3    | Vedoucí | Vyplní výběrové podmínky.   |
| 4    | System  | Zobrazí požadovaný seznam požadavků dle zvolených výběrových podmínek.  |
| 5    | Vedoucí | Vybere uzavřený požadavek.  |
| 6    | System  | Zjistí, zda má uživatel oprávnění vyhodnocovat řešení vybraného požadavku.  |
| 7    | System  | Zjistí, zda již byla poznámka v minulosti napsána, pokud ano (a) jinak (b).   |
| 8    | System  | Zobrazí detail požadavku s navrženými řešeními a všemi přístupnými volbami podle oprávnění uživatele a stavu požadavku – poznámka k řešení požadavku je přístupná jen oprávněným uživatelům (vedoucím oddělení) a u uzavřených požadavků. |
| 9a   | System  | Zobrazí text poznámky a tlačítko pro její úpravu.   |
| 10a  | Vedoucí | Zvolí úpravu poznámky.  |
| 11a  | System  | Zpřístupní pole na úpravu poznámky a tlačítko na uložení.   |
| 12a  | Vedoucí | Opraví poznámku a uloží jí.   |
| 9b   | System  | Zobrazí pole na vložení poznámky a tlačítko na uložení.   |
| 10b  | Vedoucí | Vloží novou poznámku a uloží jí.  |
| 13   | System  | Zkontroluje povinně vyplněné pole poznámky – v případě jeho nevyplnění na tuto skutečnost upozorní uživatele a poznámku neuloží (konec).  |
| 14   | System  | Uloží poznámku.   |
| 15   | System  | Zobrazí poznámku jako text, zobrazí tlačítko na její úpravu a skryje tlačítko na uložení poznámky.  |

**Tabulka 17:** Příklad užití: Označení použitého řešení

| Krok | Role    | Akce  |
|------|---------|---|
| 1    | Vedoucí | Zvolí položku Seznam požadavků z menu Požadavky.  |
| 2    | System  | Zobrazí výběrové podmínky pro zobrazení seznamu požadavků.  |
| 3    | Vedoucí | Vyplní výběrové podmínky.   |
| 4    | System  | Zobrazí požadovaný seznam požadavků dle zvolených výběrových podmínek.  |
| 5    | Vedoucí | Vybere uzavřený požadavek.  |
| 6    | System  | Zjistí, zda má uživatel oprávnění vyhodnocovat řešení vybraného požadavku.  |
| 7    | System  | Zobrazí detail požadavku s příslušnými řešeními a všemi přístupnými volbami podle zjištěného oprávnění uživatele a podle stavu požadavku – zaškrťovací políčko pro označení řešení jako použité je viditelné jen pro oprávněné uživatele (vedoucí oddělení) a u uzavřených požadavků. |
| 8    | Vedoucí | Vybere požadované řešení a zaškrtně políčko pro použité řešení.   |
| 9    | System  | Uloží příznak použitého řešení.   |
| 10   | System  | Zobrazí informaci o použití označeného řešení.  |

**Tabulka 18:** Příklad užití: Reset hesla správcem systému

| Krok | Role    | Akce   |
|------|---------|--|
| 1    | System  | Zkontroluje, zda má uživatel oprávnění správce. Pokud ano, zpřístupní v menu Správa položku Uživatelé. |
| 2    | Správce | Zvolí z menu Správa položku Uživatelé.   |
| 3    | System  | Zobrazí seznam aktivních uživatelů.  |
| 4    | Správce | Vybere požadovaného uživatele a zvolí jeho opravu.   |
| 5    | System  | Zobrazí formulář s daty vybraného uživatele a jeho oprávněními.  |
| 6    | Správce | Zvolí reset hesla.   |
| 7    | System  | Vygeneruje nové heslo a uloží ho.  |
| 8    | System  | Zašle uživateli e-mail s vygenerovaným heslem.   |

**Tabulka 19:** Příklad užití: Oprava uživatele

| Krok | Role    | Akce  |
|------|---------|---|
| 1    | System  | Zkontroluje, zda má uživatel oprávnění správce. Pokud ano, zpřístupní v menu Správa položku Uživatelé.  |
| 2    | Správce | Zvolí z menu Správa položku Uživatelé.  |
| 3    | System  | Zobrazí seznam aktivních uživatelů.   |
| 4    | Správce | Může si nechat zobrazit i již neplatné uživatele zvolením příslušné volby.  |
| 5    | System  | Reaguje na volbu správce (všechny / jen aktivní uživatele) a podle ní zobrazuje příslušný seznam uživatelů s vyznačením již neplatných záznamů. |
| 6    | Správce | Vybere požadovaného uživatele a zvolí jeho opravu.  |

| Krok | Role    | Akce  |
|------|---------|---|
| 7    | System  | Zobrazí formulář, načte oddělení pro opravu oprávnění a zobrazí data vybraného uživatele a jeho oprávnění.  |
| 8    | Správce | Opraví data uživatele, popř. jeho oprávnění zaškrtnutím příslušných oddělení a zvolí uložení (a) nebo storno (b).                                 |
| 9a   | System  | Zkontroluje povinně vyplněná pole – v případě jejich nevyplnění na tuto skutečnost upozorní správce a data neuloží (konec).                       |
| 10a  | System  | Zkontroluje duplicitu uživatelského jména nebo e-mailové adresy – v případě duplicity na tuto skutečnost upozorní správce a data neuloží (konec). |
| 11a  | System  | Zkontroluje formát e-mailové adresy, popř. doménu – pokud je chybný/á, upozorní na to správce a data neuloží (konec).                             |
| 12a  | System  | Uloží změnu dat.  |
| 13a  | System  | Skryje formulář s nastavením uživatele a znovu načte seznam uživatelů.  |
| 9b   | System  | Skryje formulář s nastavením uživatele.   |

**Tabulka 20:** Příklad užití: Smazání uživatele

| Krok | Role    | Akce   |
|------|---------|--|
| 1    | System  | Zkontroluje, zda má uživatel oprávnění správce. Pokud ano, zpřístupní v menu Správa položku Uživatelé.   |
| 2    | Správce | Zvolí z menu Správa položku Uživatelé.   |
| 3    | System  | Zobrazí seznam aktivních uživatelů.  |
| 4    | Správce | Může si nechat zobrazit i již neplatné uživatele zvolením příslušné volby.   |
| 5    | System  | Reaguje na volbu správce (všechny / jen aktivní uživatele) a podle ní zobrazuje příslušný seznam uživatelů s vyznačením již neplatných záznamů.  |
| 6    | Správce | Vybere požadovaného uživatele a zvolí jeho smazání.  |
| 7    | System  | Zobrazí kontrolní otázku, zda chce uživatel oddělení opravdu smazat.   |
| 8    | Správce | Akci zruší (konec) nebo potvrdí.   |
| 9    | System  | Zkontroluje, zda uživatel není použit u založeného požadavku, vloženého řešení nebo hodnocení řešení. Pokud je uživatel již použit, smazání uživatele neprovede a informuje správce systému (konec). |
| 10   | System  | Smaže uživatele.   |
| 11   | System  | Znovu načte aktuální seznam uživatelů.   |

**Tabulka 21:** Příklad užití: Založení nového uživatele

| Krok | Role    | Akce   |
|------|---------|--|
| 1    | Systém  | Zkontroluje, zda má uživatel oprávnění správce. Pokud ano, zpřístupní v menu Správa položku Uživatelé.   |
| 2    | Správce | Zvolí z menu Správa položku Uživatelé.   |
| 3    | Systém  | Zobrazí seznam aktivních uživatelů.  |
| 4    | Správce | Zvolí založení nového uživatele.   |
| 5    | Systém  | Zobrazí formulář pro založení nového uživatele, načte oddělení pro zvolení oprávnění.  |
| 6    | Správce | Vyplní data uživatele (uživatelské jméno, e-mailovou adresu, oddělení a popř. vybere příslušné oprávnění) a zvolí uložení dat (a) nebo storno (b). |
| 7a   | Systém  | Zkontroluje povinně vyplněná pole – v případě jejich nevyplnění na tuto skutečnost upozorní správce a data neuloží (konec).                        |
| 8a   | Systém  | Zkontroluje duplicitu uživatelského jména nebo e-mailové adresy – v případě duplicity na tuto skutečnost upozorní správce a data neuloží (konec).  |
| 9a   | Systém  | Zkontroluje formát e-mailové adresy, popř. doménu – pokud je chybný/á, upozorní na to správce a data neuloží (konec).                              |
| 10a  | Systém  | Vygeneruje nové heslo a uloží data.  |
| 11a  | Systém  | Zašle uživateli e-mail s vygenerovaným heslem.   |
| 12a  | Systém  | Skryje formulář s nastavením uživatele a znovu načte seznam uživatelů.   |
| 7b   | Systém  | Skryje formulář s nastavením uživatele.  |

**Tabulka 22:** Příklad užití: Založení nového oddělení

| Krok | Role    | Akce  |
|------|---------|---|
| 1    | Systém  | Zkontroluje, zda má uživatel oprávnění správce. Pokud ano, zpřístupní v menu Správa položku Oddělení.                       |
| 2    | Správce | Zvolí z menu Správa položku Oddělení.   |
| 3    | Systém  | Zobrazí seznam aktivních oddělení.  |
| 4    | Správce | Zvolí zadání nového oddělení.   |
| 5    | Systém  | Zobrazí formulář.   |
| 6    | Správce | Vyplní název oddělení a zadání buď potvrdí (a), nebo zruší (b).   |
| 7a   | Systém  | Zkontroluje povinně vyplněné pole – v případě jeho nevyplnění na tuto skutečnost upozorní uživatele a data neuloží (konec). |
| 8a   | Systém  | Zkontroluje duplicitu názvu oddělení – v případě duplicity na tuto skutečnost upozorní správce a data neuloží (konec).      |
| 9a   | Systém  | Uloží nové oddělení.  |
| 10a  | Systém  | Skryje formulář na založení nového oddělení a znovu načte aktuální seznam oddělení.   |
| 7b   | Systém  | Skryje formulář na založení nového oddělení.  |

**Tabulka 23:** Příklad užití: Oprava oddělení nebo ukončení jeho platnosti

| Krok | Role    | Akce  |
|------|---------|---|
| 1    | Systém  | Zkontroluje, zda má uživatel oprávnění správce. Pokud ano, zpřístupní v menu Správa položku Oddělení.   |
| 2    | Správce | Zvolí z menu Správa položku Oddělení.   |
| 3    | Systém  | Zobrazí seznam aktivních oddělení.  |
| 4    | Správce | Může si nechat zobrazit i již neplatná oddělení zvolením příslušné volby.   |
| 5    | Systém  | Reaguje na volbu správce (všechna / jen aktivní oddělení) a podle ní zobrazuje příslušný seznam oddělení s vyznačením již neplatných záznamů. |
| 6    | Správce | Vybere požadované oddělení a zvolí jeho opravu.   |
| 7    | Systém  | Zobrazí formulář s daty vybraného oddělení.   |
| 8    | Správce | Opraví požadovaná data a opravu buď potvrdí (a), nebo zruší (b)   |
| 9a   | Systém  | Zkontroluje povinně vyplněné pole – v případě jeho nevyplnění na tuto skutečnost upozorní správce a data neuloží (konec).                     |
| 10a  | Systém  | Zkontroluje duplicitu názvu oddělení – v případě duplicity na tuto skutečnost upozorní správce a data neuloží (konec).                        |
| 11a  | Systém  | Uloží změnu dat.  |
| 12a  | Systém  | Skryje formulář pro opravu oddělení a znovu načte aktuální seznam oddělení.   |
| 9b   | Systém  | Skryje formulář pro opravu oddělení.  |

**Tabulka 24:** Příklad užití: Smazání oddělení

| Krok | Role    | Akce   |
|------|---------|--|
| 1    | Systém  | Zkontroluje, zda má uživatel oprávnění správce. Pokud ano, zpřístupní v menu Správa položku Oddělení.  |
| 2    | Správce | Zvolí z menu Správa položku Oddělení.  |
| 3    | Systém  | Zobrazí seznam aktivních oddělení.   |
| 4    | Správce | Může si nechat zobrazit i již neplatná oddělení zvolením příslušné volby.  |
| 5    | Systém  | Reaguje na volbu správce (všechna / jen aktivní oddělení) a podle ní zobrazuje příslušný seznam oddělení s vyznačením již neplatných záznamů.                    |
| 6    | Správce | Vybere požadované oddělení a zvolí jeho smazání.   |
| 7    | Systém  | Zobrazí kontrolní otázku, zda chce uživatel oddělení opravdu smazat.   |
| 8    | Správce | Akci zruší (konec) nebo potvrdí.   |
| 9    | Systém  | Zkontroluje, zda oddělení není použito u založeného požadavku nebo uživatele. Pokud je oddělení použito, smazání oddělení neprovede a informuje správce (konec). |
| 10   | Systém  | Smaže příslušné oddělení.  |
| 11   | Systém  | Znovu načte aktuální seznam oddělení.  |



## Příloha B – REST API

Tabulka 25: REST API

| URL {parametr}   | Metoda | Povinné (nepovinné) parametry         | Oprávnění |
|--|--------|---------------------------------------|-----------|
| /api-pozadavky/pozadavek   | POST   | nazev, popis, (id_oddeleni), (termin) |           |
| /api-pozadavky/reseni  | POST   | id_rodice, popis                      |           |
| /api-pozadavky/pozadavek/{id}/poznamka                                 | PUT    | poznamka                              | oddělení  |
| /api-pozadavky/pozadavek/{id}/uzavrit                                  | PUT    |                                       | oddělení  |
| /api-pozadavky/pozadavek/{id}/otevrit                                  | PUT    |                                       | oddělení  |
| /api-pozadavky/reseni/{id}/implementace                                | PUT    | implementace                          | oddělení  |
| /api-pozadavky/reseni/{id}/hodnoceni                                   | POST   | hodnoceni                             |           |
| /api-pozadavky/reseni/{id}/hodnoceni                                   | GET    |                                       |           |
| /api-pozadavky/pozadavek/{id}/reseni/hodnoceni                         | GET    |                                       |           |
| /api-pozadavky/pouzite-pozadavky-a-reseni/uzavrene/{od}/{do}/hodnoceni | GET    |                                       |           |
| /api-pozadavky/pozadavek/{id}  | GET    |                                       |           |
| /api-pozadavky/pozadavek/{id}/reseni                                   | GET    |                                       |           |
| /api-pozadavky/pozadavky   | GET    |                                       |           |
| /api-pozadavky/pozadavky/uzivatel/{id}                                 | GET    |                                       |           |
| /api-pozadavky/pozadavky/oddeleni/{id}                                 | GET    |                                       |           |
| /api-pozadavky/pozadavky/uzavrene                                      | GET    |                                       |           |
| /api-pozadavky/pozadavky/otevrene                                      | GET    |                                       |           |
| /api-pozadavky/pozadavky/{od}/{do}                                     | GET    |                                       |           |
| /api-pozadavky/pozadavky/uzavrene/{od}/{do}                            | GET    |                                       |           |
| /api-pozadavky/prispevky/{dni}/{hodin}                                 | GET    |                                       |           |
| /api-pozadavky/reseni/{id}/reakce                                      | GET    |                                       |           |
| /api-pozadavky/pouzita-reseni/uzavrena/{od}/{do}                       | GET    |                                       |           |
| /api-pozadavky/pouzite-pozadavky-a-reseni/uzavrene/{od}/{do}           | GET    |                                       |           |
| /api-pozadavky/resitele/{od}/{do}                                      | GET    |                                       |           |
| /api-pozadavky/oddeleni  | POST   | nazev                                 | správce   |
| /api-pozadavky/oddeleni/{id}   | PUT    | nazev, aktivni                        | správce   |
| /api-pozadavky/oddeleni/{id}   | DELETE |                                       | správce   |
| /api-pozadavky/oddeleni/{id}   | GET    |                                       |           |
| /api-pozadavky/oddeleni  | GET    |                                       |           |
| /api-pozadavky/verejne/oddeleni/aktivni                                | GET    |                                       |           |

| URL {parametr}                               | Metoda | Povinné (nepovinné) parametry                              | Oprávnění |
|--|--------|--|-----------|
| /api-pozadavky/uzivatele                     | GET    |  | správce   |
| /api-pozadavky/uzivatele/aktivni             | GET    |  | správce   |
| /api-pozadavky/prihlaseni                    | POST   | jmeno, heslo   |           |
| /api-pozadavky/odhlaseni                     | POST   |  |           |
| /api-pozadavky/zmena-hesla                   | PUT    | jmeno, heslo-stare, heslo, heslo2                          |           |
| /api-pozadavky/verejne/vygeneruj-heslo       | GET    |  |           |
| /api-pozadavky/verejne/vygeneruj-hash        | POST   | heslo  |           |
| /api-pozadavky/verejne/posli-heslo           | POST   | adresa   |           |
| /api-pozadavky/verejne/posli-mail            | POST   | adresy, predmet, zprava                                    |           |
| /api-pozadavky/verejne/registrace            | POST   | jmeno, heslo, heslo2, email, (id_oddeleni), info_pozadavek |           |
| /api-pozadavky/uzivatel                      | POST   | jmeno, email, (id_oddeleni), (spravce)                     | správce   |
| /api-pozadavky/uzivatel/{id}                 | GET    |  | správce   |
| /api-pozadavky/uzivatel/{id}                 | PUT    | email, aktivni, spravce, id_oddeleni                       | správce   |
| /api-pozadavky/uzivatel/{id}                 | DELETE |  | správce   |
| /api-pozadavky/uzivatel/{id}/opravneni       | GET    |  |           |
| /api-pozadavky/uzivatel/{id}/opravneni       | POST   | id_oddeleni  | správce   |
| /api-pozadavky/uzivatel/{uz}/opravneni/{odd} | DELETE |  | správce   |
| /api-pozadavky/uzivatel/{id}/opravneni       | PUT    | seznam_id_oddeleni   | správce   |
| /api-pozadavky/uzivatel/{id}/nastaveni       | GET    |  |           |
| /api-pozadavky/uzivatel/{id}/nastaveni       | PUT    | info_pozadavek   |           |
| /api-pozadavky/verejne/spravci/e-maily       | GET    |  |           |
| /api-pozadavky/uzivatele/info-pozadavek      | GET    |  |           |