

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

**Student:** Martin Král  
**Studijní program:** Kybernetika a robotika (bakalářský)  
**Obor:** Robotika  
**Název tématu:** Využití výsledků projektů realizovaných přes portál Spolupráce s průmyslem

### Pokyny pro vypracování:

1. Seznamte se s portálem Spolupráce s průmyslem budovaným na Fakultě informačních technologií Českého vysokého učení technického v Praze.
2. Navrhněte komponenty, které povedou k lepšímu využití výsledků projektů realizovaných přes portál. Soustředte se zejména na:
  - a) doplnění o volbu režimu ochrany duševního vlastnictví autorů projektu, včetně možnosti pozdější změny zvoleného režimu,
  - b) vytvoření GUI pro publikaci článku shrnujícího výsledky projektu,
  - c) využití hodnocení projektu a jeho výsledků jeho řešitelem (studentem),
  - d) využití hodnocení projektu a jeho výsledků jeho zadavatelem (průmyslovým partnerem),
  - e) integraci portálu se systémem pro správu verzí zdrojového kódu, který je výsledkem projektu.
3. Tyto komponenty realizujte jako portlety v systému Liferay.
4. Držte se zásad pro návrh a realizaci softwarového díla.
5. Specifikujte případy užití, na nichž komponenty otestujete a demonstujete jejich funkčnost.

### Seznam odborné literatury:

- [1] Fišer Aleš - Posílení zpětné vazby z průmyslu na univerzity - Praha 2014
- [2] JSR 168: Portlet Specification - 2003 - Dostupné z: <http://jcp.org/en/jsr/detail?id=168>
- [3] JSR 286: Portlet Specification 2.0 - 2008 - Dostupné z: <http://jcp.org/en/jsr/detail?id=286>
- [4] Liferay User Guide - 2014 - Dostupné z: <http://www.liferay.com>

**Vedoucí bakalářské práce:** Ing. Aleš Fišer

**Platnost zadání:** do konce letního semestru 2015/2016

L.S.

doc. Dr. Ing. Jan Kybic  
**vedoucí katedry**

prof. Ing. Pavel Ripka, CSc.  
**děkan**

V Praze dne 5. 2. 2015



ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA ELEKTROTECHNICKÁ

KATEDRA KYBERNETIKY



Bakalářská práce

## Využití výsledků projektů realizovaných přes portál Spolupráce s průmyslem

*Martin Král*

Vedoucí práce: Ing. Aleš Fišer

22. května 2015



---

## Poděkování

Tímto bych rád poděkoval mému vedoucímu Ing. Aleši Fišerovi za jeho cenné rady během vývoje, dále za jeho odborné vedení a podnětné připomínky při tvorbě této práce.



---

# Prohlášení

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

V Praze dne 22. května 2015

.....

České vysoké učení technické v Praze

Fakulta elektrotechnická

© 2015 Martin Král. Všechna práva vyhrazena.

*Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě elektrotechnické. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí, je nezbytný souhlas autora.*

### **Odkaz na tuto práci**

Král, Martin. *Využití výsledků projektů realizovaných přes portál Spolupráce s průmyslem*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta elektrotechnická, 2015.



---

# Abstrakt

Tato práce pojednává o možnostech rozšíření portálu Spolupráce s průmyslem, které povedou k lepšímu využití výsledků přes něj realizovaných projektů.

Práce uvádí přístupy k řešení některých podobných problémů na jiných portálech. Dále je v práci obsaženo seznámení s portálem Spolupráce s průmyslem, uživatelskými rolemi a se základními procesy. Popsány jsou zde i nejdůležitější technologie, které jsou při jeho vývoji a provozu využívány.

Za další práce představuje a analyzuje návrhy a popisuje způsob realizace nových komponent a procesů, vedoucích k řešení určitých problémů souvisejících s využitím výsledků projektů. Závěrem popisuje způsob jejich zakomponování do stávajícího systému a jejich přínos.

**Klíčová slova** Spolupráce s průmyslem, GUI, Liferay, portál, portlet, JSF, Activiti, Java EE

---

# Abstract

This thesis describes possibilities for Cooperation with Industry portal extension which leads to better utilization of projects, completed with this portal.

At the beginning there are described several ways of solving similar problems, used on the other portals. The next chapter presents the Cooperation with Industry portal, its user roles and basic processes. Also there are specified important technologies used for application development and running.

Then this thesis brings and analyzes some options of new components and processes and describes the way of their implementation. At the end there are described benefits of added components and their incorporation into current system.

**Keywords** Cooperation with Industry, GUI, Liferay, portal, portlet, JSF, Activiti, Java EE

---

# Obsah

<b>Úvod</b>	<b>1</b>
<b>1 Rešerše</b>	<b>3</b>
1.1 Hodnocení na jiných portálech . . . . .	3
1.2 Prezentace vývojářů . . . . .	5
<b>2 Teoretická část</b>	<b>7</b>
2.1 Seznámení s portálem Spolupráce s průmyslem . . . . .	7
2.2 Využívané technologie . . . . .	9
2.3 Architektura aplikace . . . . .	14
<b>3 Analýza a návrh řešení</b>	<b>17</b>
3.1 Využití hodnocení spolupráce . . . . .	17
3.2 Ošetření ochrany duševního vlastnictví . . . . .	20
3.3 Zveřejnění výsledků spolupráce . . . . .	22
3.4 Integrace se systémem pro správu verzí . . . . .	24
<b>4 Realizace</b>	<b>27</b>
4.1 Ošetření ochrany duševního vlastnictví . . . . .	27
4.2 Hodnocení spolupráce . . . . .	28
4.3 Zveřejňování dokončených zadání . . . . .	29
4.4 Integrace se systémem pro správu verzí . . . . .	30
<b>5 Testování</b>	<b>35</b>
5.1 Uživatelské testování . . . . .	35
5.2 Testování procesů . . . . .	35
5.3 Jednotkové a integrační testy . . . . .	36
<b>Závěr</b>	<b>37</b>

<b>Literatura</b>	<b>39</b>
<b>A Seznam použitých zkratk</b>	<b>41</b>
<b>B Obsah přiloženého CD</b>	<b>43</b>

---

## Seznam obrázků

1.1	Hodnocení obchodníka na serveru Heureka.cz [2]	3
1.2	Hodnocení obchodníků na serveru Aukro.cz [3]	4
1.3	Hodnocení videa na serveru YouTube [4]	5
1.4	Profil uživatele na GitHub [6]	5
1.5	Otázka a nejvhodnější odpověď na Stack Overflow [8]	6
2.1	Ukázka jednoho z procesních diagramů v Activiti	15
2.2	Schéma architektury systému Spolupráce s průmyslem [16]	16
3.1	Návrh zobrazení hodnocení na profilu studenta	18
3.2	Návrh zobrazení hodnocení na profilu zadavatele	19
3.3	Návrh volby režimu IP - výběr z rozevíracího menu	21
3.4	Návrh volby režimu IP - tlačítka s popisem v dialogu	21
3.5	Návrhy volby režimu IP - tlačítka s popisem pod nimi	21
3.6	Návrh portletu pro změnu režimu ochrany duševního vlastnictví	22
3.7	Návrh portletu s detailem článku o dokončeném zadání	23
3.8	Návrh přehledu dokončených zadání	25
3.9	Návrh zobrazení informací o repozitáři	26
4.1	Portlet pro návrh změny režimu ochrany duševního vlastnictví	27
4.2	Zjednodušený proces změny režimu ochrany duševního vlastnictví	28
4.3	Realizované hodnocení na profilu studenta	29
4.4	Realizované hodnocení na profilu zadavatele	30
4.5	Zjednodušený proces publikace dokončeného zadání	31
4.6	Portlet pro schvalování publikace dokončeného zadání	31
4.7	Portlet pro vytvoření návrhu článku zadání	32
4.8	Portlet pro schvalování návrhu článku zadání	33
4.9	Detail dokončeného zadání	34
4.10	Seznam dokončených zadání	34



---

# Úvod

Portál Spolupráce s průmyslem je rozsáhlá webová aplikace, vyvíjená na Fakultě informačních technologií ČVUT v Praze, sloužící primárně ke zprostředkování spolupráce mezi studenty a průmyslem. Tento portál se snaží co nejvíce zpřístupnit a přiblížit studentům praxi, zajistit pro ni administrativní podporu a zabezpečit její právní rámec.

Důležitou součástí portálu je, že umožňuje spolupráci provázat se souvisejícími studovanými předměty, což přináší další motivaci ke spolupráci pro studenty vedle finanční odměny za dokončené zadání.

Portál Spolupráce s průmyslem je v provozu od roku 2013. Od této doby prošel mnoha modifikacemi a i nadále je rozvíjen a vylepšován. V současné době je do portálu registrováno více než 800 studentů a téměř 100 společností.

Jedním z problémů portálu však zůstává nedokonalé zužitkování výsledků dokončených zadání na straně univerzity. Výsledky zadání jsou jednak výstupy řešitelů zadání, neboli řešení zadaného problému, a dále hodnocení samotné spolupráce mezi řešiteli a průmyslovým partnerem.

Jelikož se jedná o univerzitní systém, je žádoucí, aby zadání byla přínosná nejen jejich řešitelům a zadávajícímu průmyslovému partnerovi, ale i ostatním studentům pro možnost jejich dalšího rozvoje. Proto je vhodné promyslet způsob jejich uveřejnění pro ostatní uživatele systému a to takovým způsobem, aby nedošlo k narušení soukromí účastníků zadání. Se zužitkováním výstupů zadání se však váží další problémy.

Prvním problémem je nepřiliš jasná definice duševního vlastnictví odevzdané práce. Výstupy projektů mohou podléhat pouze jednomu režimu ochrany duševního vlastnictví autorů, který nemusí vyhovovat všem průmyslovým partnerům, či studentům. Dalším problémem je, že práce vůbec nemusí být odevzdána přes portál Spolupráce s průmyslem a tím pádem nemusí být možná její publikace a případný rozvoj.

## Cíle práce

Cílem této práce je navrhnout způsoby využití výsledků spolupráce mezi studenty a průmyslovými partnery tak, aby byl možný další rozvoj odevzdané práce a bylo dosaženo větší popularity a tím i využívanosti portálu.

S tím souvisí následující úkoly:

- Určení způsobu stanovení režimu ochrany duševního vlastnictví autorů odevzdaného řešení, včetně definice procesu pro jeho dodatečnou změnu
- Analýza možností integrace systému pro správu verzí do portálu pro lepší dostupnost odevzdané práce a její další rozvoj
- Vytvoření procesu pro publikaci dokončeného zadání a jeho výsledků
- Návrh prezentace dokončených zadání a jejich výsledků
- Návrh využití vzájemného hodnocení spolupráce mezi průmyslovým partnerem a studenty
- Realizace výše popsaných návrhů
- Otestování všech realizovaných součástí a jejich zakomponování do portálu

## Členění obsahu práce

Práce je rozdělena do kapitol:

**Rešerše** - zkoumá možné přístupy k řešení daných problémů, zejména pak způsoby hodnocení na jiných portálech a možnosti prezentace vývojářů

**Teoretická část** - obsahem je seznámení se základními funkcemi portálu, uživatelskými rolemi, základními procesy a s technologiemi, využívanými při vývoji a provozu portálu

**Návrh řešení** - představuje možné přístupy k řešení výše popsaných problémů, jejich rozbor a možné přínosy.

**Realizace** - zde je popsán vývoj nových částí aplikace a představeno jejich výsledné řešení

**Testování** - popisuje testy s uživateli a další způsoby testování vytvářené části aplikace



# Rešerše

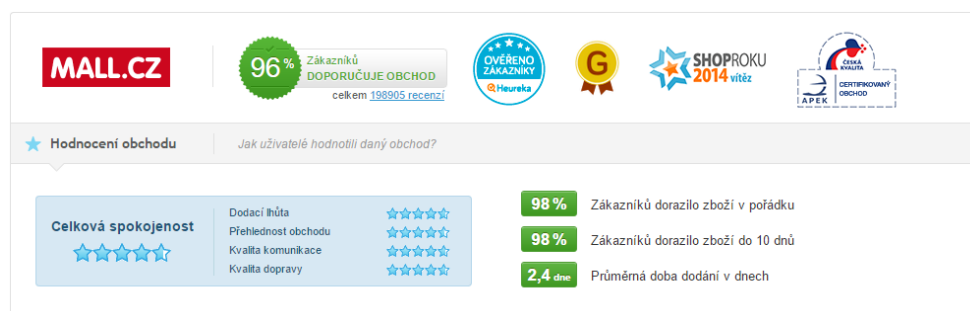
Tato část zkoumá různé přístupy k řešení uživatelského hodnocení a možnosti veřejné prezentace vývojářů a jimi odvedené práce.

## 1.1 Hodnocení na jiných portálech

V současné době existuje nepřehledné množství webových aplikací, obsahujících nějakou formu hodnocení uživatelů, či produktů a služeb. Zde je několik příkladů.

### 1.1.1 Heureka

Zajímavým příkladem je hodnocení obchodníků na serveru srovnávače cen Heureka [1]. Jak je patrné z obrázku 1.1, je zde zachován čistý design a tím i přehlednost této komponenty. Pod celkovým hodnocením pak následují jednotlivé uživatelské recenze.



Obrázek 1.1: Hodnocení obchodníka na serveru Heureka.cz [2]

### 1.1.2 Aukro

Podobně jako na serveru Heureka.cz vypadá i hodnocení na inzertním portálu Aukro [3]. Zajímavým způsobem zde však probíhá samotný proces hodnocení obchodníka.

Jak je patrné z obrázku 1.2, při jiném, než plném hodnocení je nutné ještě vybrat důvod tohoto negativního hodnocení. V případě negativního komentáře je nutné potvrdit, že uživatel vyčerpal všechny možnosti a skutečně nedošlo pouze k nedorozumění. To může působit dojmem, že uživatel provádí něco nestandardního a navíc tak získává čas si svůj negativní komentář rozmyslet, což může předejít špatnému hodnocení "v afektu".

V důsledku toho je u naprosté většiny obchodníků více než 95% komentářů kladných a v průměru hodnocení lépe než 4,5/5. Dobré hodnocení pak samozřejmě podporuje obchod a tím i využívanost portálu. Těžko však posoudit relevanci takovýchto hodnocení, nicméně uživatelé nemají důvod ke stížnostem, neboť je stále možnost přidání negativního komentáře.

**Ohodnotit jednotlivé oblasti prodeje**

Ohodnoťte jednotlivé oblasti prodeje pomocí stupnice od 1 do 5. 1 hvězdička znamená nejnižší známku, 5 hvězdiček znamená nejvyšší hodnocení. Pokud zvolíte 4 a méně hvězdiček, je třeba ze seznamu vybrat důvod takového hodnocení.

Shoda předmětu s popisem	★★★★★
Kontakt s Prodejcem	★★★★☆
	<input type="radio"/> Na odpověď jsem čekal/a příliš dlouho <input type="radio"/> Nezdvořilý přístup ze strany Prodejce <input type="radio"/> Neobdržel/a jsem odpověď na dodatečné otázky <input type="radio"/> Při pokusu o kontakt Prodejce nereagoval <input type="radio"/> Jiný důvod
Rychlost odeslání zboží	★★★★★
Náklady na dopravu	★★★★★

**Udělení komentářů**

Typ komentáře  Pozitivní  Neutrální  Negativní

- Opravdu jste vyčerpal všechny možnosti, jak kontaktovat Vašeho obchodního partnera? Berte prosím na vědomí, že komentáře neslouží pro komunikaci s obchodním partnerem. Komentář, který udělíte, bude viditelný na kartě Uživatele a nebude již možné jej odstranit.
- Pokud si přejete s obchodním partnerem cokoliv vyjasnit, napište mu.

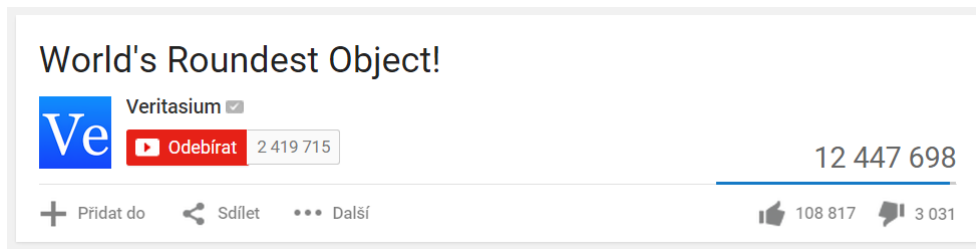
[Chci udělit komentář](#)

Obrázek 1.2: Hodnocení obchodníků na serveru Aukro.cz [3]

### 1.1.3 YouTube

Jiný způsob hodnocení je využíván například na serveru YouTube [4]. Jedná se o binární hodnocení - tedy se dvěma možnostmi: líbí nebo nelíbí. Tento způsob je velmi přehledný, uživatelsky přístupný a rychlý, jelikož nad hodno-

cením není třeba dlouho přemýšlet. Využívá se především tam, kde probíhá hodnocení mnoha uživateli. V současné době je tento, či unární způsob hodnocení (pouze s možností líbí) velice rozšířen na různých populárních portálech a uživatelských sítích (například na uživatelské síti Facebook [5]).



Obrázek 1.3: Hodnocení videa na serveru YouTube [4]

## 1.2 Prezentace vývojářů

Některé webové služby umožňují prezentaci vývojáře a jeho vlastní tvorby, která by mohla být přínosem pro jiné, často samotnému vývojáři neznámé uživatele. Většinou jsou pak tyto služby bez přímého nároku na odměnu.

### 1.2.1 GitHub

Obrázek 1.4: Profil uživatele na GitHub [6]

Jednou z takových je GitHub [7], který slouží především jako kvalitní hosting Git repositářů pro open-source projekty. Mimo to ale slouží uživatelům pro prezentaci jejich práce. Každý uživatel zde má vlastní profil, ze kterého lze přistoupit na jeho veřejné projekty<sup>1</sup>, prohlédnout si je, stáhnout repositář, či případně do projektu zkusit přispět.

Po vzoru sociálních sítí je možné vývojáře odebírat, či sdílet jeho práci. Na profilu vývojáře lze nalézt zejména základní informace, jeho kontaktní údaje, počet odběratelů a základní statistiky vývojáře. Jednotlivé projekty je pak možné označit hvězdičkou, pokud se uživateli líbí, či odebírat informace o změnách.

### 1.2.2 Stack Overflow

Jinou službou je Stack Overflow [8]. Ta slouží pro rychlou podporu od zkušenějších vývojářů. Uživatel zde položí dotaz, na který mohou ostatní uživatelé odpovídat, či hodnotit odpovědi ostatních uživatelů. Za každou akci roste uživateli jeho reputace. Ta slouží ke zpřístupnění dalších možností v rámci této služby. Kromě toho je však zobrazena i na uživatelském profilu, kde lze nalézt například i uživatelovy příspěvky, základní statistiky a podobně. Vývojářův profil tak může sloužit jako jeho vizitka.

The screenshot shows a Stack Overflow question titled "Parse JSON in JavaScript?". The question text is "I want to parse a JSON string in JavaScript. The response is something like" followed by a code block: `var response = '{"result":true,"count":1}';`. The question asks "How can I get the values result and count from this?". It has 795 votes and 213 answers. The question was asked on Feb 8 '11 at 16:34 by user605334 and edited on Oct 31 '14 at 13:35 by ProgramFOX. The top answer, by Andy E, has 1191 votes and is marked as the best answer. The answer text is "Most browsers support `JSON.parse()`, which is defined in ECMA-262 5th Edition (the specification that JS is based on). Its usage is simple:" followed by a code block: `var json = '{"result":true,"count":1}';  
obj = JSON.parse(json);  
alert(obj.count);`. The answer was answered on Feb 8 '11 at 16:38 and edited on Apr 2 '13 at 9:40.

Obrázek 1.5: Otázka a nejhodnější odpověď na Stack Overflow [8]

---

<sup>1</sup>GitHub poskytuje především veřejné repositáře, ale za poplatek hostuje i neveřejné

---

## Teoretická část

### 2.1 Seznámení s portálem Spolupráce s průmyslem

Jak již bylo zmíněno v úvodu, portál Spolupráce s průmyslem je rozsáhlá webová aplikace rozvíjená na Fakultě informačních technologií Českého vysokého učení technického v Praze. Tato aplikace obstarává především zprostředkování spolupráce mezi studenty a průmyslovými partnery a zajišťuje její právní formu. Dále zajišťuje propojení spolupráce se studovanými předměty, které souvisejí s jednotlivými zadáními.

V portálu jsou nadále zpracovávány studijní výsledky studentů, na jejichž základě jsou určovány dovednosti studentů. Ty následně slouží mimo jiné průmyslovým partnerům pro lepší představu o kvalitách studentů. Studijní výsledky jsou také využívány při informování vhodných<sup>2</sup> studentů o novém zadání.

#### 2.1.1 Základní uživatelské role

V portálu Spolupráce s průmyslem má každý uživatel přidělenou svoji roli, případně kombinaci více rolí, podle nichž probíhá autorizace na určité akce.

##### **Zadavatel**

Zástupce společnosti, který může vytvářet zadání.

##### **Student**

Uživatel, který se může ucházet o jednotlivá zadání a stát se jejich řešitelem.

---

<sup>2</sup>Jsou informováni ti studenti, kteří kvalitami, vypočítanými na základě studijních výsledků, nejvíce odpovídají kvalitám, které požaduje zadavatel

### **Učitel**

Uživatel, spravující vyučované předměty. Může schvalovat předměty k jednotlivým zadáním a rozhodovat o přijetí řešení do daného předmětu.

### **Univerzitní expert**

Univerzitní expert má za úkol pomáhat zadavatelům, kteří si o to požádají, s vytvářením zadání, přijímáním řešitelů do týmu a přijímáním odevzdaných řešení.

### **Administrátor**

Spravuje systém jako celek. Má za úkol především přidávat do systému nové studenty, průmyslové partnery a s nimi související zadavatele a další osoby.

## **2.1.2 Standardní uživatelské akce**

### **Vytvoření zadání**

Zadání vytváří zadavatel. Ten při vytváření vyplní především titulek, krátký popis, cíle, které si přeje dosáhnout, vymezí složení řešitelského týmu, včetně určení požadovaných dovedností a stanoví odměnu za dokončení zadání, odhad časové náročnosti a termíny zahájení a dokončení.

### **Nominace předmětu k zadání**

Nominovat předmět k zadání může téměř jakýkoliv uživatel a tuto akci lze provést kdykoliv. Nominaci je však nutné potvrdit, což může provést pouze učitel daného předmětu, který k tomu má právo. Při potvrzování obvykle učitel určí podmínky uznání práce do předmětu formou poznámky.

### **Sestavování řešitelského týmu**

Po vytvoření zadání se zahájí proces sestavování řešitelského týmu. V první řadě jsou e-mailem osloveni vhodní studenti – ti jsou stanoveni na základě hodnocení v dovednostech, vztahujícím se k danému zadání. Dále se zadání objeví v seznamu zadání daného uživatele.

Na zadání může být student nominován zadavatelem nebo se o zadání může sám ucházet, načež se zadavatel rozhodne, zda o studenta má zájem. V obou případech poté ještě následuje nutnost potvrzení zájmu o danou roli studentem.

V tomto stádiu je možné (například při velkém, či naopak malém zájmu o zadání) změnit zadané role.

Po obsazení a vzájemném potvrzení všech pozic má zadavatel možnost zahájit práci na zadání. Poté již složení řešitelského týmu nelze upravovat.

### **Odevzdání řešení**

Po vyřešení zadání jeden z členů řešitelského týmu odevzdá řešení. Toto řešení může zadavatel přijmout nebo s poznámkou vrátit k přepracování. Pokud k zadání byl přiřazen univerzitní expert, může počkat na jeho vyjádření.

Nezávisle na odevzdání řešení zadavateli lze odevzdat řešení do souvisejícího předmětu. O přijetí, či odmítnutí rozhoduje učitel daného předmětu.

### **Vyplácení odměny**

Po přijetí řešení zadavatel určí, jak má být finanční odměna rozdělena v rámci řešitelského týmu, přičemž má možnost ještě navýšit celkovou výši rozpočtu.

### **Hodnocení spolupráce**

Po přijetí řešení zadavatelem následuje vzájemné hodnocení spolupráce mezi řešitelským týmem a zadavatelem.

### **Úprava profilu**

Studenti, univerzitní experti, průmysloví partneři a jejich zadavatelé mají v portálu Spolupráce s průmyslem vlastní profily. Tyto profily je možné doplnit o fotografii, respektive logo společnosti, kontaktní údaje, krátký popis a další informace.

Na profilu studenta je možné nalézt přehled jeho dovedností s objektivním a subjektivním hodnocením. Profily studentů mohou být skryté nebo veřejné, které jsou pak k dispozici v seznamu veřejných profilů studentů.

## **2.2 Využívané technologie**

Celý systém Spolupráce s průmyslem pracuje na portálu Liferay, který uchovává informace o uživateli, získané přes Shibboleth. Pro řízení procesů v Systému je využíván procesní stroj Activiti. Pro ukládání dat slouží relační databáze PostgreSQL a dále je využíván datový sklad fakulty. Soubory a dokumenty jsou ukládány pomocí systému Alfresco DMS. Schéma závislosti jednotlivých komponent systému je patrné ze schématu 2.2.

### **2.2.1 Liferay portal**

Portál je obecně webová aplikace, tvořená souborem stránek, která má na starost mimo jiné správu uživatelských účtů, řízení uživatelského přístupu k jednotlivým stránkám pomocí SSO (Single Sign On) autentizace, vzhled stránek a další.

Liferay [9] je open-source webový portál, založený na Javě, který umožňuje správu jednotlivých stránek portálu, včetně nastavení jejich vzhledu pomocí různých témat. Samozřejmě také obstarává správu uživatelských účtů a k nim přidružených uživatelských profilů včetně zabezpečení procesu autentizace SSO.

Mezi jeho základní funkce také patří možnost vícejazyčného uživatelského rozhraní, což ulehčilo lokalizaci portálu Spolupráce s průmyslem do českého a anglického jazyka.

Nejjednodušší cesta pro správu portálu a vkládání nového obsahu (nových stránek), kterou Liferay poskytuje, je Liferay CMS – grafické uživatelské rozhraní, přístupné přímo z prohlížeče, které umožňuje provádět veškeré potřebné akce bez nutnosti zásahu do zdrojových kódů.

### 2.2.2 Portlet

Portlety jsou komponenty webové aplikace – moduly, které se instalují do portálu a následně vkládají do připravené šablony stránky. Generují XHTML obsah a poskytují tedy prezentační vrstvu systému. Jednotlivé stránky s vloženými portlety pak tvoří uživatelské rozhraní portálu.

Portlety se instalují na jednotlivé stránky portálu Liferay například pomocí uživatelského rozhraní Liferay CMS. Obecně je však portlety možné použít i na jiných portálech založených na Java EE, díky jednotnému API, definovanému v Portlet Specification JSR 168 [10] a JSR 286 [11].

Portlety jsou definovány v XML souborech a jsou řízeny kontejnerem – obalem, který obstarává jejich životní cyklus a poskytuje jim perzistentní úložiště.

### 2.2.3 Java Server Faces

Java Server Faces (zkráceně JSF) je framework, pro vývoj profesionálních webových aplikací. Umožňuje oddělit vývoj uživatelského rozhraní a aplikační logiky.

**Uživatelské rozhraní** je vytvářeno formou XML souborů, za použití speciálních XML značek. Způsob vytváření stránek je patrný v ukázce kódu 2.1



```

<pcc:secureContent
  portletPageBean="#{simpleLookForAssignmentBean}"
  <h:form class="form-horizontal">
    <h2>
      #{simpleLookForAssignmentBean.assignments.size()}
      #{languages['assignmentsFound']}
    </h2>
    <ui:repeat
      value="#{simpleLookForAssignmentBean.assignments}"
      var="assignment">
      <h3 class="title">
        #{assignment.assignmentTitle}
      </h3>
    </ui:repeat>
  </h:form>
</pcc:secureContent>

```

Algoritmus 2.1: Ukázka JSF kódu

**Aplikační logika** (ManagedBean) zajišťuje předávání informací z backendu do uživatelského rozhraní. Tato vrstva je již vytvářena v Javě a jako její příklad můžeme použít ukázkou 2.2.

```

@ManagedBean(name = "simpleLookForAssignmentBean")
@ViewScoped
public class SimpleLookForAssignmentBean implements
  PortletPageBean, Serializable {

  private List<AssignmentPreviewDtoNew> assignments;

  @Override
  public Status isUserAuthorized() {
    if (PortletUtil.isPortletInitialized()) {
      return Status.OK;
    }
    final SimpleAssignmentBrowserService service =
      SimpleAssignmentBrowserServiceUtil.getService();
    final Long assigneeId = securityHelper.getAssigneeId();
    assignments = service.findAllAssignments(assigneeId);
    return Status.OK;
  }

  public List<AssignmentPreviewDtoNew> getAssignments() {
    return assignments;
  }
}

```

Algoritmus 2.2: Ukázka Managed bean

### 2.2.4 Java Persistence API

Aplikace Spolupráce s průmyslem využívá k ukládání dat databázi PostgreSQL. Tento open-source databázový systém umožňuje práci s daty běžnými SQL příkazy, ale pro elegantní propojení aplikace s databází je v portálu Spolupráce s průmyslem využíváno objektově-relační mapování pomocí standardu JPA (Java Persistence API) a jeho implementace - frameworku Hibernate.

Tento přístup umožňuje pracovat se záznamy v databázi jako s objekty. Třídy, používané pro objektově-relační mapování, nazýváme entitními. Operace s těmito třídami obstarává třída EntityManager – ta umožňuje především vytvářet, mazat a editovat záznamy v databázi na základě příslušných objektů.

Entitní třídy obsahují privátní proměnné, pojmenované buď jako jednotlivé sloupce příslušné tabulky nebo v případě cizích klíčů jako objekt, na který daný sloupec odkazuje. Jednotlivé entity mezi sebou mohou mít vztahy one-to-one, one-to-many, many-to-one nebo many-to-many. Dále entitní třídy musejí obsahovat gettery a settery.

```
@Entity
public class Sponsor {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long sponsorId;

    private Boolean deleted;

    @ManyToOne
    @JoinColumn(name = "institutionId")
    private Institution institution;

    @OneToMany
    @JoinColumn(name = "sponsorId")
    private List<SponsorCooperationEvaluation>
        sponsorCooperationEvaluations;

    ...

    // getters and setters
}
```

Algoritmus 2.3: Ukázka entitní třídy

V ukázce 2.3 vidíme entitní třídu Sponsor, která odpovídá stejnojmenné tabulce, která obsahuje mimo jiné sloupce sponsorId (BIGINT), deleted (BOOLEAN), institutionId (BIGINT). Tyto sloupce jsou namapovány na příslušné proměnné, přičemž objekty SponsorCooperationEvaluations jsou mapovány z jiné tabulky na základě sponsorId.

### 2.2.5 Spring a Inversion of Control

Spring je open-source framework využívající návrhový vzor Inversion of Control. Ten umožňuje přesunutí odpovědnosti za vytvoření a provázání objektů z aplikace na framework. Vytvořené objekty jsou nazývány Beans. Tímto způsobem je možné odstranit těsné vazby mezi objekty, které by měly negativní vliv na přehlednost projektu a znesnadňovaly by samotný vývoj i testování aplikace.

Tento princip funguje tak, že ve třídách není třeba vytvářet instance objektů, ale Spring framework sám do těchto tříd dosadí potřebné instance. Třída se přitom již nemusí starat o to, jaká implementace jí byla předložena, jelikož pracuje pouze s rozhraním, které je společné pro všechny implementace daného objektu.

Pro každý objekt je možné definovat jeho životnost - například singleton (společná instance pro celý projekt), či prototype (pro každé použití nová instance). K definování závislostí a určení životností jednotlivých objektů je použit samostatný XML soubor.

V ukázce 2.4 vidíme praktické využití výše popsaného principu. Třída FullNameLookupService je pouze interface, ale díky definované závislosti v XML souboru a anotaci @Autowired můžeme využívat jejích funkcí.

```
public class AssigneeProfileServiceImpl implements
    AssigneeProfileService {

    @Autowired
    private FullNameLookupService fullNameLookupService;

    ...

    String fullName =
        fullNameLookupService.getFullName(username);

    ...

}
```

Algoritmus 2.4: Ukázka využití Spring frameworku

### 2.2.6 Activiti

Procesy v portálu Spolupráce s průmyslem jsou řízeny procesním strojem Activiti [12]. Tyto procesy jsou modelovány pomocí standardu BPMN 2.0 [13]. Tento způsob především přináší velkou přehlednost a ulehčuje pochopení průběhu procesů v aplikaci. Procesy jsou vytvářeny pomocí následujících komponent:

**User task** Část procesu, která vyžaduje vstup od uživatele. Začíná existovat,

## 2. TEORETICKÁ ČÁST

---

když na tuto úlohu v procesu přijde řada a přestává existovat, jakmile ji uživatel dokončí. Při dokončení je možné navíc předat různé informace pomocí formuláře, přičemž předané hodnoty se uloží jako proměnné daného procesu.

### Service task

Část procesu, která může provést funkci, definovanou v knihovně, se zadanými parametry.

### Script task

Provede příkazy, zadané ve skriptovacím jazyce (Javascript nebo Groovy).

### Exclusive gateway

Rozhodovací bod procesu. Výstup je možný pouze jednou větví, podle podmínek definovaných v dané větvi.

### Parallel gateway

Rozvětvení procesu. Výstup je možný více větvemi.

Jako příklad můžeme uvést diagram 2.1, který je vytvořen na základě standardu BPMN 2.0. Tento diagram popisuje proces hodnocení spolupráce a vyplácení odměny řešitelům po schválení odevzdaného řešení zadavatelem.

### 2.2.7 Git

Důležitou součástí při vývoji aplikace podobného rozsahu je využívání systému pro správu verzí zdrojového kódu a souborů (zkráceně VCS z anglického Version Control System). Takový systém umožňuje elegantně vést historii úprav souborů a v případě problému zjistit, kdo, kdy a jakým způsobem upravoval daný projekt a případně jej vrátit do předchozího stavu.

Při vývoji portálu Spolupráce s průmyslem je využíván verzovací systém Git [14]. Jedná se o distribuovaný<sup>3</sup> systém, který umožňuje správu jakýchkoliv souborů, ale navržen byl zejména na správu verzí zdrojových kódů. Mimo jiné usnadňuje i paralelní spolupráci více vývojářů na stejném projektu.

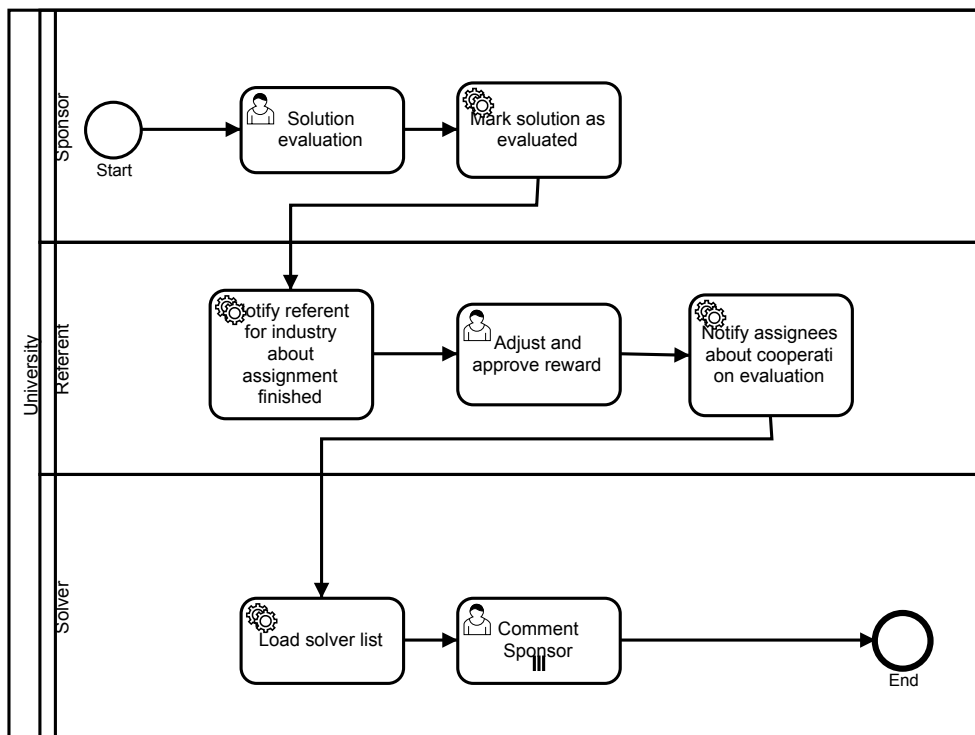
Pro přehlednou správu, grafické zobrazení a spolehlivý hosting repositářů je využívána open-source webová aplikace GitLab [15], fungující na fakultním serveru.

## 2.3 Architektura aplikace

Aplikace Spolupráce s průmyslem je rozdělena do několika vrstev:

---

<sup>3</sup>V případě distribuovaného systému pro správu verzí všichni uživatelé, podílející se na projektu uchovávají a verzují kompletní kopii jeho repositáře. To má pozitivní vliv na bezpečnost, neboť repositář není pouze na jediném (centrálním) stroji, a také na rychlosti práce



Obrázek 2.1: Ukázka jednoho z procesních diagramů v Activiti

**Frontend** - prezentační vrstva, vytvářená přímo pro portál Liferay. Obsahuje především definice portletů, XML soubory a ManagedBeany pro Java Server Faces.

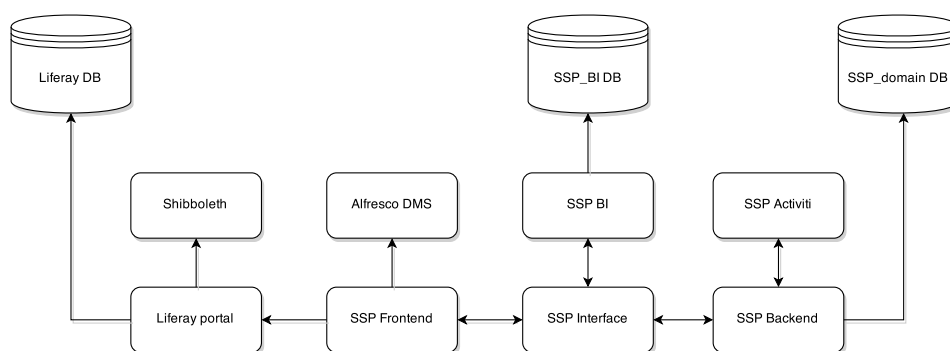
**Backend** - je vrstva pro získávání, zpracování a ukládání dat v databázi za využití výhradně objektově relačního mapování. Dále také zahrnuje implementace servisních tříd. Mimo to také obstarává fungování procesního stroje Activiti.

**Modul BI** - tento modul obstarává především výpočet vhodnosti studentů na jednotlivá zadání. K tomu využívá studijních výsledků jednotlivých studentů, získaných z datového skladu univerzity.

**Interface** - rozhraní mezi frontendem a backendem. Obsahuje objekty pro přenos dat mezi těmito dvěma vrstvami (DTO - Data Transfer Object) a interface servisních tříd.

## 2. TEORETICKÁ ČÁST

---



Obrázek 2.2: Schéma architektury systému Spolupráce s průmyslem [16]

---

## Analýza a návrh řešení

Tato kapitola popisuje současný stav aplikace a některé její současné problémy. Ty jsou zde podrobně analyzovány a následně předloženy možné návrhy uživatelského rozhraní komponent a s nimi souvisejících procesů, vedoucí k jejich řešení. Dále jsou zde popsány případy využití a přínosy jednotlivých částí.

### 3.1 Využití hodnocení spolupráce

Hodnocení spolupráce mezi studentem a zadavatelem nastává po dokončení zadání. Hodnocení již v systému probíhá, nicméně stále ještě není nikde zobrazeno.

Hodnocení studentů je povinné, protože bez jeho dokončení nedojde k vyplacení odměny řešitelům. Po ohodnocení zadavatele řešiteli sice již ne-následují další kroky, ale přesto systém studenta upozorňuje na nevyplněné hodnocení ve formě nesplněného úkolu v postranní liště. Hodnocení řešitelů se od hodnocení zadavatele mírně odlišuje.

V případě hodnocení řešitelů je hodnocení rozděleno na:

**Hodnocení spolupráce a jejích výsledků**, které je pro všechny řešitele společné. V tomto hodnocení zadavatel může několika větami popsat celkovou spokojenost se spoluprací a jejími výsledky a musí vyplnit hodnocení ve formě hvězdiček.

**Hodnocení jednotlivých řešitelů**, kde opět zadavatel může v několika větách popsat spokojenost s řešitelem a jeho přínos k celkovému výsledku. Opět je zde povinné vyplnit hodnocení ve formě hvězdiček.

Hodnocení zadavatele je v současné době rozděleno na hodnocení různých aspektů spolupráce, jako je komunikace se zadavatelem, jeho ochota, profesionalita, kvalita dodaných materiálů a spokojenost s odměnou. Všechny tyto aspekty jsou hodnoceny zvlášť formou hvězdiček.

### 3. ANALÝZA A NÁVRH ŘEŠENÍ

---

Jako vhodné využití tohoto hodnocení se nabízí zobrazovat jej na uživatelských profilech studentů a zadavatelů. Prosté zobrazení hodnocení však má svá úskalí. Především průmysloví partneři si často nepřejí, aby jiní partneři věděli o jejich zadáních. Proto by bylo vhodné při vyplňování hodnocení zadavatelem přidat možnost „zůstat v anonymitě“.

Na začátku části s hodnocením bude vhodné umístit shrnutí všech dosavadních hodnocení. Bude zde informace o celkovém počtu hodnocení a zároveň i průměr všech obdržných hodnocení.

Pod shrnutím budou už k nahlédnutí jednotlivá hodnocení. U každého z nich bude jméno hodnotitele (v případě, že si zadavatel nepřál zůstat v anonymitě), hodnocení ve formě hvězdiček a datum ohodnocení. Pokud byl k hodnocení přidán komentář, bude zobrazen také.

<b>4 dokončená zadání</b>	Průměrné hodnocení: <b>4,0</b> ★★★★☆
<b>Aplikace pro publikaci projektů CWI s.r.o.</b>	★★★★☆ Odevzdáno 11. 5. 2015
<b>Aleš Fišer:</b> Řešitel vstřícný, komunikativní a ochotný.	
<b>Implementace frontendu eshopu Balsa.cz</b>	★★★★☆ Odevzdáno 11. 1. 2015
<b>Radovan Novák:</b> Výborná spolupráce	

Obrázek 3.1: Návrh zobrazení hodnocení na profilu studenta

Jak již bylo popsáno výše, hodnocení zadavatele je rozděleno na více částí. Shrnutí hodnocení na profilu zadavatele tedy bude obsahovat průměrné hodnocení napříč všemi částmi hodnocení, a navíc i průměrné hodnocení v jednotlivých aspektech.

Jednotlivá hodnocení budou při vstupu na stránku obsahovat pouze průměrné hodnocení, vypočítané ze všech částí daného hodnocení. Pokud na něj však uživatel klikne, zobrazí se hodnocení detailní.





Obrázek 3.2: Návrh zobrazení hodnocení na profilu zadavatele

### 3.1.1 Negativní hodnocení

Důležitým cílem je portál Spolupráce s průmyslem udržet populární jak pro studenty, tak i pro průmyslové partnery. Je tedy nutné promyslet způsob nakládání s negativním hodnocením, neboť negativní hodnocení může být vyústěním osobního konfliktu a nemusí tedy být relevantní. Takovéto spory by poté mohly poškodit uživatele, což by mělo negativní dopad na jejich celkovou spokojenost s portálem. Jako řešení se nabízejí tyto možnosti:

**Schválení každého hodnocení administrátorem** Tento přístup by však byl z dlouhodobého hlediska nevhodný, neboť portál se bude rozšiřovat a správce by tak měl příliš mnoho povinností.

**Omezit negativní hodnocení** například nutností jeho explicitního potvrzení autorem, jako například na webovém portálu Aukro [3]. Tento přístup je popsán v sekci 1.1.2.

**Možnost odstranit nevhodné hodnocení administrátorem** V této variantě by bylo vhodné ještě přidat upozornění správce portálu na negativní hodnocení.

**Kombinace výše uvedených možností**

### 3.1.2 Přínosy

Jeden z hlavních problémů portálu Spolupráce s průmyslem je, že se studenti bojí kroku do neznáma, protože se často nechtějí na zadání přihlásit.

Hodnocení na profilu zadavatele částečně přispěje k jeho transparentnosti a ostatní studenti tak budou moci lépe usoudit, co očekávat, pokud se na jeho zadání přihlásí.

Hodnocení na profilu studenta bude informativní pro další zadavatele, kteří by mohli mít na jeho základě o studenta zájem pro své zadání.

## 3.2 Ošetření ochrany duševního vlastnictví

V portálu Spolupráce s průmyslem doposud nebylo možné volit režim ochrany duševního vlastnictví. Pokud nebylo explicitně stanoveno jinak (například formou zvláštní smlouvy) byla na výstupy projektu udělena licence zadavateli a ČVUT. Nyní by se proces stanovení a případné změny režimu ochrany duševního vlastnictví měl zjednodušit.

### 3.2.1 Dostupné režimy ochrany duševního vlastnictví

V portálu Spolupráce s průmyslem budou k dispozici následující režimy ošetření ochrany duševního vlastnictví autorů výstupů zadání:

**Licence zadavateli a ČVUT** - je výchozí režim ochrany duševního vlastnictví, kdy zůstávají výstupy projektu vlastnictvím studenta a zároveň student poskytuje licenci zadavateli a ČVUT.

**Zcela volné užití pro každého** - v tomto režimu jsou výstupy projektu dostupné všem a lze je využít ve vlastním díle bez omezení.

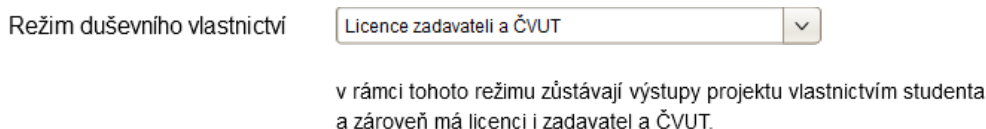
**Volné užití na principu GPL** - zachovává výstupy projektu dostupné všem, stejně jako v případě prvního režimu. Pokud je však uživatel zahrne do svého díla, je nutné toto dílo šířit pod stejnou licenci.

**Výhradní licence pro zadavatele** - výhradní licence pro zadavatele je restriktivní variantou, kdy si zadavatel nepřeje, aby výstupy projektu mohly být dále rozvíjeny. V tomto režimu je nutné uzavřít dodatečnou písemnou smlouvu o výhradní licenci.

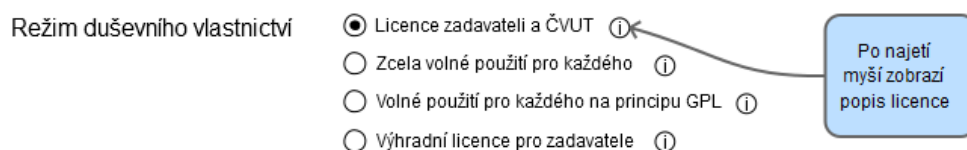
### 3.2.2 Nastavení režimu ochrany duševního vlastnictví

Režim ochrany duševního vlastnictví se bude definovat v závěrečném kroku vytváření zadání.

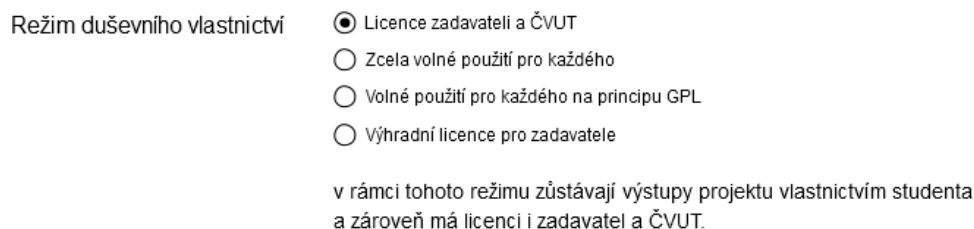
## 3.2. Ošetření ochrany duševního vlastnictví



Obrázek 3.3: Návrh volby režimu IP - výběr z rozevřacího menu



Obrázek 3.4: Návrh volby režimu IP - tlačítka s popisem v dialogu



Obrázek 3.5: Návrhy volby režimu IP - tlačítka s popisem pod nimi

### 3.2.3 Změna režimu ochrany duševního vlastnictví

Zadavatel i řešitel by měli mít možnost navrhnout změnu režimu ochrany duševního vlastnictví. Tuto změnu by však měli odsouhlasit všichni ostatní účastníci zadání. Navíc pokud se bude jednat o výhradní licenci pro zadavatele, bude tuto změnu muset potvrdit administrátor a případně stanovit i administrační poplatek, spojený s nutností podpisu smlouvy a podobně.

### 3.2.4 Přínosy

Přínosy této části jsou zřejmé - tedy jasné stanovení duševního vlastnictví výstupů jednotlivých zadání a možnost jejich dodatečné změny, což doposud nebylo přímo v aplikaci možné.

Díky možnosti výběru režimu ochrany duševního vlastnictví vyjde portál Spolupráce s průmyslem vstříc především těm průmyslovým partnerům, kterým nevyhovovaly výchozí licenční podmínky.

## Aplikace pro využití výsledků projektů

### Změna režimu duševního vlastnictví

<b>Stávající režim</b>	Licence zadavateli a ČVUT ⓘ
<b>Změnit režim</b>	<input type="radio"/> Zcela volné použití pro každého ⓘ <input type="radio"/> Volné použití pro každého na principu GPL ⓘ <input checked="" type="radio"/> Výhradní licence pro zadavatele ⓘ
	Výhradní licence pro zadavatele je restriktivní variantou, kdy si zadavatel nepřeje, aby výstupy projektu mohly být dále rozvíjeny. Uzavírá se samostatná písemná smlouva o výhradní licenci. Zvolený režim musí kromě všech účastníků zadání potvrdit i správce portálu.
<input type="button" value="Zrušit"/>	<input type="button" value="Uložit změny"/>

Obrázek 3.6: Návrh portletu pro změnu režimu ochrany duševního vlastnictví

## 3.3 Zveřejnění výsledků spolupráce

Výstupy projektů nyní zůstávají nezveřejněné, což znemožňuje jejich další rozvoj. I proto by bylo vhodné najít vhodný způsob, jak projekty po dokončení zveřejnit, aniž by mohl být dotčen případný požadavek sponzora na anonymitu v systému.

Vize je taková, že po dokončení zadání bude vytvořen jakýsi krátký článek shrnující průběh a přínosy spolupráce, v němž bude možné nahlédnout do přiložených souborů, případně zdrojových kódů.

### 3.3.1 Proces zveřejňování

Tento článek bude vytvářet správce portálu, případně vznikne zvláštní uživatelská role, specializovaná pro tento účel. Vytvořený článek poté sponzor společně se všemi řešiteli odsouhlasí, případně navrhne úpravu, kterou bude nutné všemi znovu odsouhlasit.

Jelikož je portál Spolupráce s průmyslem lokalizovaný, bude vhodné přidat možnost vytvořit i lokalizované články shrnující zadání.

## Implementace frontendu eshopu

Výsledky spolupráce

Programování GUI Databáze HTML

**Zadavatel:** Ing. Svatopluk Novák - Balsa.cz

**Řešitel:** František Nadaný - programátor

+ 98 uživatelům se líbí tato práce

### Popis spolupráce

Zadáním bylo vytvořit uživatelské rozhraní chystaný eshop. Řešitel se zadání zhostil s nadšením a velice rychle dodal výsledky.

### Využití výsledků

Výstupem bylo uživatelské rozhraní, které bylo téměř bez úprav nasazeno do produkce. Spolupráce mezi zadavatelem a řešitelem nadále pokračuje.

### Přiložené soubory

Report.pdf	238 KB	<a href="#">Stáhnout</a>
Diagram.png	430 KB	<a href="#">Stáhnout</a>

### Zdrojové kódy

Zobrazit repozitář tohoto projektu na GitLabu

[Stáhnout ve formátu .zip](#)

**Režim ochrany duševního vlastnictví**

Licence zadavatelů a ČVUT ⓘ

**Odměna** 5 000 CZK

### Související předměty

Matematika Programování

### Dovednosti

Webové aplikace Programování Datamining Databáze

Obrázek 3.7: Návrh portletu s detailem článku o dokončeném zadání

### 3.3.2 Přínosy

Přestože si někteří zadavatelé přejí zůstat v systému skrytí pro ostatní průmyslové partnery, jiní nemusí klást na anonymitu v systému takový důraz. Naopak by mohli publikaci dokončeného zadání využít jako vlastní propagaci a to jak na úrovni portálu Spolupráce s průmyslem, tak i na úrovni všeobecné - jako reklamu společnosti, kterou zastupují. Touto cestou by průmysloví partneři přispěli k motivaci ostatních studentů a případně i k přihlášení na jejich další zadání.

Touto cestou se zároveň zviditelní i řešitel a bude moci veřejně ukázat své dovednosti, což poslouží jako další kritérium pro ostatní zadavatele, bude-li se ucházet o jiné zadání.

#### 3.3.3 Zachování anonymity

Jak již bylo zmíněno, někteří zadavatelé si pravděpodobně budou přát zůstat v anonymitě, protože bude vhodné přidat do procesu schvalování návrhu možnost anonymizace zadání. Při zvolení této varianty nebude v článku zobrazena společnost ani zadavatel.

#### 3.3.4 Ochrana duševního vlastnictví a NDA

Výsledky spolupráce pak budou zveřejňovány pro uživatele podle zvoleného režimu ochrany duševního vlastnictví zadání - tedy pro studenty, zaměstnance fakulty a zadavatele v případě prvního a pro všechny uživatele v případě druhého a třetího režimu. V případě, že byl zvolen režim čtvrtý, restriktivní režim, nebudou výsledky spolupráce publikovány vůbec.

#### 3.3.5 Hodnocení

Otázkou zůstává, zda zveřejňovat vzájemné hodnocení mezi zadavatelem a řešiteli nebo dát možnost ostatním uživatelům ohodnotit výsledky spolupráce.

Pokud by měli čtenáři možnost hodnotit, bylo by vhodné zvolit binární, nebo spíše unární variantu hodnocení, které je využíváno například na sociální síti Facebook [5] k hodnocení příspěvků, jelikož je to uživatelsky přístupnější a modernější způsob, než je hodnocení pomocí hvězdiček. Uživatel by tak získal možnost označit, že se mu dané řešení líbí. Zde však vyvstává problém, jakým způsobem motivovat čtenáře k hodnocení. V současném stavu by totiž pravděpodobně došlo k tomu, že by většina zadání nebyla hodnocena vůbec nebo jen velmi málo.

#### 3.3.6 Seznam dokončených zadání

Pro zobrazení přehledu těchto článků bude existovat portlet s přehledem všech publikovaných zadání.

### 3.4 Integrace se systémem pro správu verzí

Pro lepší koordinaci práce v týmu a zálohování verzí je výhodné využívat systém pro správu verzí. Zadavatelé však doposud nemají možnost ani mimo portál požádat o vytvoření repositáře na fakultním serveru, a proto v případě, že o využívání systému pro správu verzí mají zájem, musejí se uchýlit k využití externí služby. To může vést k situaci, kdy všechny soubory související s příslušným zadáním, zejména pak odevzdané zdrojové kódy, nebudou v portálu k dispozici k nahlédnutí a tím pádem ani pro další rozvoj, přestože by to licenční ujednání umožňovalo.

**Přehled dokončených zadání** Seřadit dle  ▼

**Implementace frontendu eshopu**

**Balsa.cz**

Programování
GUI
Databáze
HTML

Odevzdáno 15. 1. 2015

Zadavatel si přál vytvořit uživatelské rozhraní pro chystaný eshop.

Zcela volně užití pro každého ?

Výsledky spolupráce

LOGO

+ **98** uživatelům se líbí tato práce

Obrázek 3.8: Návrh přehledu dokončených zadání

Aby byla veškerá data bezpečně zálohovaná, a zároveň k nim měli všichni účastníci projektu dobrý přístup, je vhodné využívat společný repositář, umístěný na veřejně dostupném serveru. Z tohoto repositáře všichni uživatelé stahují soubory a zároveň na něj nahrávají vlastní změny. K tomuto účelu existují veřejně dostupné služby, jako například GitHub [7], který je však v bezplatné verzi určen pro open-source projekty.

Alternativou je využití služby GitLab, fungující na fakultním serveru. Přístup k této službě však zadavatelé získat nemohou, protože se nejedná o veřejně dostupnou aplikaci.

Pro tyto účely však existuje GitLab API, které umožní integraci některých akcí přímo do uživatelského rozhraní portálu. Mezi tyto akce bude patřit zejména založení repositáře, přidání vývojáře k projektu, či stažení celé aktuální verze projektu.

Založit repositář pro zadání bude možné až po zahájení spolupráce, aby zbytečně nebyly vytvářeny repositáře, které by zůstaly nevyužité v případě, že by spolupráce vůbec nebyla zahájena. Repositáři budou nastavena oprávnění podle zvoleného režimu ochrany duševního vlastnictví pro dané zadání - tedy v případě nejrestriktivnějšího režimu bude repositář soukromý a budou jej moci zobrazit pouze k němu přiřazení uživatelé. Repositář nemusí být úzce propojený s jediným zadáním, ale mělo by být možné přiřadit k zadání již existující repositář.

Po založení, či přiřazení dříve založeného repositáře se v detailu zadání zobrazí adresa repositáře pro přístup autentizovaný pomocí SSH. Zároveň zde bude možnost zobrazit repositář přímo v aplikaci GitLab.

Po dokončení zadání, v případě jeho zveřejnění, bude u jeho článku kromě souborů odeslaných přes uživatelské rozhraní portálu i možnost stažení celého repositáře.

#### Systém pro správu verzí

Nápověda

Adresa	<input type="text" value="git@gitlab.fit.cvut.cz:portal/moje-zadani.git"/>	<input type="button" value="Zobrazit repositář"/>
--------	--	---

Obrázek 3.9: Návrh zobrazení informací o repositáři

#### 3.4.1 Přínosy

Propojením systému s aplikací GitLab dojde především k jednoduchému zpřístupnění využívání této služby uživatelům a přispěje k lepší kontrole nad odevzaným řešením a jeho případnému rozvoji.



## Realizace

### 4.1 Ošetření ochrany duševního vlastnictví

Režim ochrany duševního vlastnictví je stanoven v závěrečném kroku vytváření zadání. V současné době jsou v systému 4 základní režimy. Ty jsou včetně lokalizovaných názvů a popisů uloženy v databázi pro případnou snadnou úpravu, či definici zcela nového režimu.

#### Implementace frontendu eshopu

Změna režimu ochrany duševního vlastnictví

Současný režim **Zcela volné užití pro každého**

V rámci tohoto režimu jsou výstupy projektu volně k užití pro kohokoliv. Způsob zveřejnění však musí respektovat NDA.

- Nový režim
- Licence zadavateli a univerzitě
  - Zcela volné užití pro každého
  - Volné užití pro každého na principu GPL
  - Výhradní licence pro zadavatele

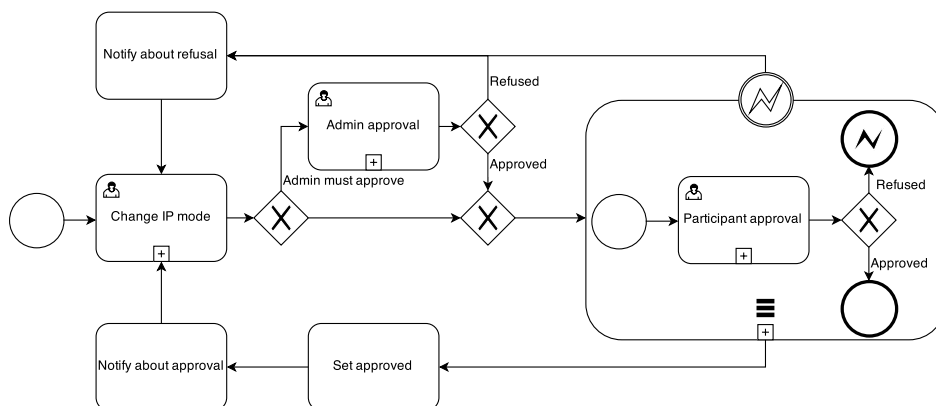
V rámci tohoto režimu jsou výstupy projektu volně k užití pro kohokoliv, ale díla, která je využijí musejí být šířena pod stejnou licenci. Způsob zveřejnění musí respektovat NDA.

Poznámka

← Zpět na zadání

✓ Odeslat žádost

Obrázek 4.1: Portlet pro návrh změny režimu ochrany duševního vlastnictví



Obrázek 4.2: Zjednodušený proces změny režimu ochrany duševního vlastnictví

Režim je vybírán pomocí tlačítek typu radio, pod kterými je zobrazován popis aktuálně zvoleného režimu. Při kliknutí na jiný režim se popis ihned změní.

Zvolený režim je možné kdykoliv před dokončením zadání změnit. Proces změny popisuje zjednodušené<sup>4</sup> schéma procesu na obrázku 4.2. K odeslání požadavku na změnu slouží portlet zobrazený na obrázku 4.1 Tuto změnu může navrhnout jakýkoliv účastník zadání, ale všichni ostatní ji musí přijmout. V případě volby nejvíce restriktivního režimu tuto změnu musí odsouhlasit navíc i administrátor systému, přičemž stanoví i administrační poplatek za tento úkon. Stanovení dodatečného poplatku zatím v systému neprobíhá, protože současný způsob nakládání s financemi na tento krok není připraven.

## 4.2 Hodnocení spolupráce

Jak již bylo zmíněno, hodnocení spolupráce mezi zadavatelem a řešiteli již v systému probíhalo, ale nebylo doposud nikde využíváno. Pro tento účel bylo realizováno zobrazení hodnocení na profilu studenta a na profilu průmyslového partnera.

V procesu hodnocení přibyla pouze možnost pro zadavatele "zůstat v anonymitě", při jejíž volbě nebude u příslušného hodnocení zobrazen uživatel.

Dotčené části portletů mají v obou případech podobný vzhled, jako je zobrazeno v návrhu (viz obr. 3.2 a 3.1). V horní části hodnocení je jakési shrnutí, kolik má student, či zadavatel dokončených zadání a průměrné hodnocení jeho

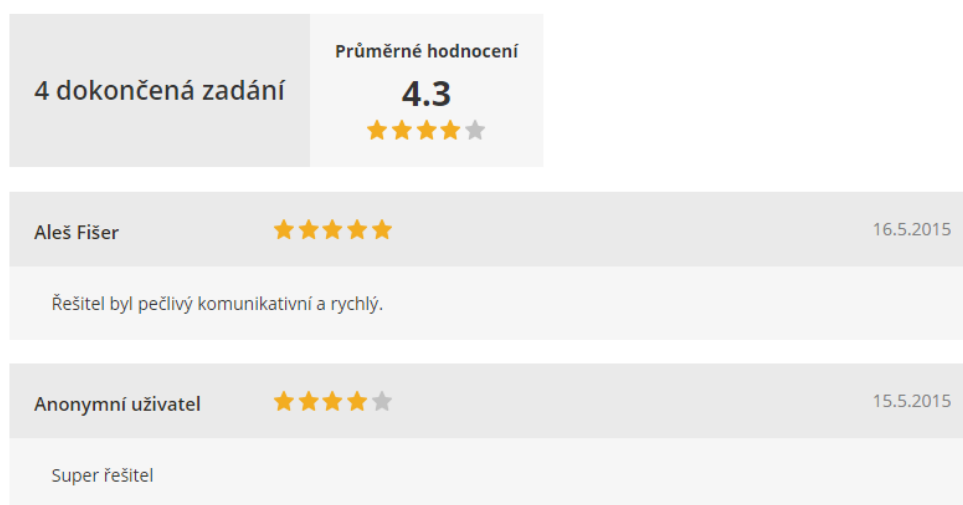
<sup>4</sup>Schéma procesu bylo pro lepší přehlednost zjednodušeno. Nerespektuje tak plně zásady standardu BPMN, přestože využívá jeho komponenty.

spolupráce. V případě profilu zadavatele je průměrné hodnocení počítáno na základě průměrných dílčích hodnocení.

Pod shrnutím je již přehled jednotlivých hodnocení s případným komentářem. V případě hodnocení na profilu zadavatele je možné kliknutím na hodnocení zobrazit jeho details.

V případě nevhodného hodnocení nebo komentáře má správce systému možnost toto hodnocení odstranit. V současné době jsou v databázi pouze pozitivní hodnocení, a proto se zatím zdá být bezvýznamné tento potenciální problém řešit.

#### Statistika studenta



Obrázek 4.3: Realizované hodnocení na profilu studenta

### 4.3 Zveřejňování dokončených zadání

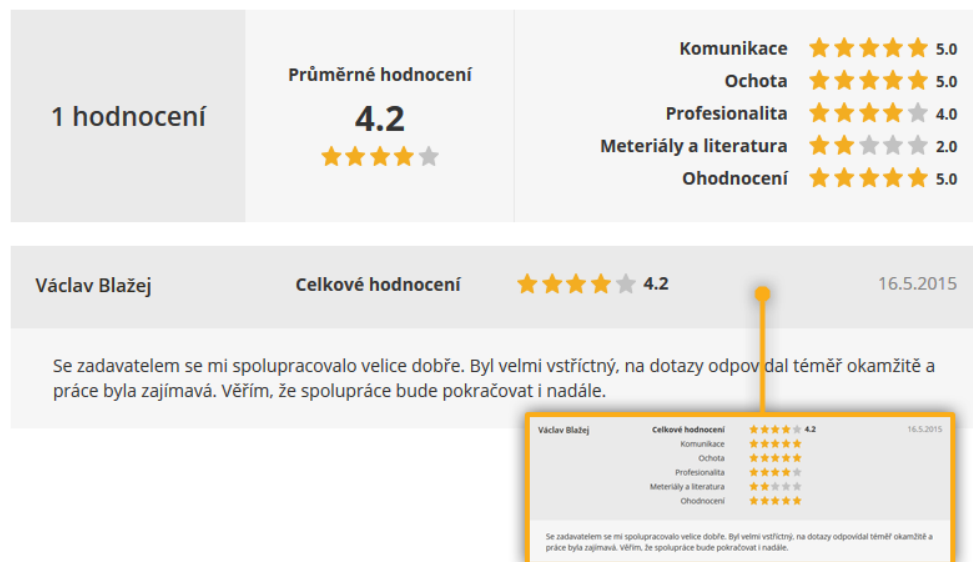
Zjednodušený proces zveřejňování dokončeného zadání je patrný ze schématu 4.5. Zveřejnění dokončeného zadání začíná při hodnocení spolupráce, kdy zadavatel nebo jakýkoliv řešitel může podat návrh.

Po odeslání požadavku na zveřejnění je dotázán administrátor systému, v případě, že nebyl autorem tohoto požadavku. Poté následuje schválení účastníky daného zadání (opět kromě autora požadavku). Portlet sloužící pro potvrzování zveřejnění je na obrázku 4.6 Schéma je navrženo tak, že v případě zamítnutí je možné požadavek opakovat. V uživatelském rozhraní však tato možnost prozatím chybí.

V případě potvrzení jsou všichni účastníci informováni o tom, že požadavek byl schválen a následuje vytváření článku (viz obr. 4.7. Ten opět mohou dle

## 4. REALIZACE

### Statistika zadavatele



Obrázek 4.4: Realizované hodnocení na profilu zadavatele

procesního diagramu vytvářet všichni účastníci, ale uživatelské rozhraní vyzve k vytvoření pouze uživatele, který podal návrh na zveřejnění.

Každý vytvořený článek je pak nutné odsouhlasit všemi ostatními účastníky a administrátorem. K tomu slouží portlet z obrázku 4.8 V případě, že uživatel s článkem nesouhlasí, může navrhnout jeho úpravu. Každou takovou úpravu je pak nutno znovu odsouhlasit. Ke starší verzi článku je možné se kdykoliv vrátit a dodatečně ji odsouhlasit. To uživateli umožňuje například navrhnout úpravu a v případě neodsouhlasení ostatními přijmout původní verzi.

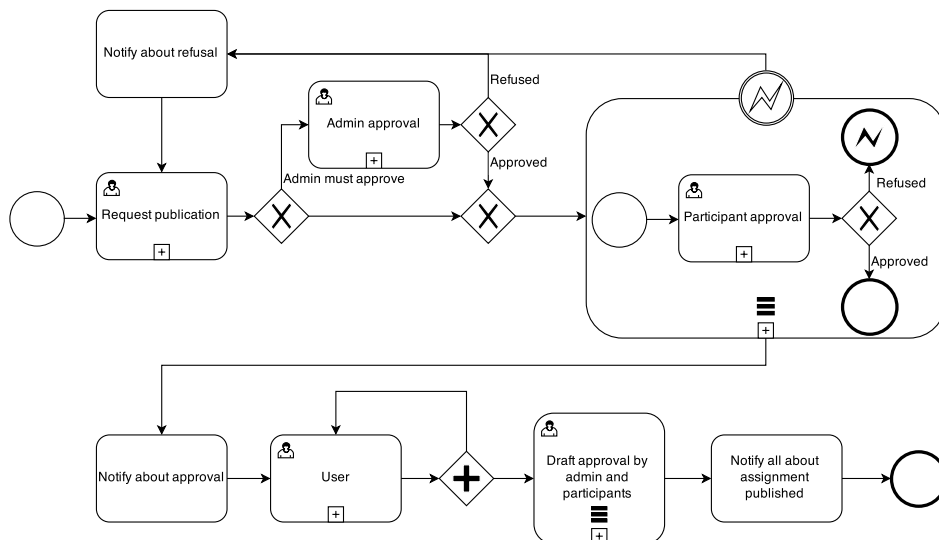
V případě odsouhlasení návrhu je článek označen jako schválený a je odeslán email všem účastníkům, informující o zveřejnění článku. Ten se pak objeví v seznamu dokončených zadání.

Dokončená zadání prozatím nemohou ostatní uživatelé nijak hodnotit. Od této možnosti bylo upuštěno z důvodu, že v současné době je příliš málo uživatelů, kteří by zadání hodnotili. Články by tak pravděpodobně nebyly hodnoceny vůbec nebo jen velmi málo.

### 4.4 Integrace se systémem pro správu verzí

Jak již bylo zmíněno v části 3.4, portál by měl být propojen se službou GitLab. Bohužel se prozatím nepodařilo vyjednat podmínky pro založení vlastní verze

#### 4.4. Integrace se systémem pro správu verzí



Obrázek 4.5: Zjednodušený proces publikace dokončeného zadání

## Implementace frontendu eshopu

Schvalování zveřejnění dokončeného zadání

Václav Blažej	Řešitel	<a href="#">Nevyžádl se</a>
Aleš Fišer	Zadavatel	<a href="#">Přijal</a>

Síň slávy umožňuje zobrazení zadání, včetně jeho výsledků. Po odsouhlasení zařazení do síně slávy je možné vybrat, které z odevzdaných souborů budou zveřejněny a které nikoli. Pro zveřejnění výsledků je vyžadován neprve souhlas referenta a následně všech řešitelů.



Obrázek 4.6: Portlet pro schvalování publikace dokončeného zadání

## 4. REALIZACE

### Implementace frontendu eshopu

Editace detailu dokončeného zadání

Václav Blažej      Řešitel      Čeká na vytvoření článku  
Aleš Fišer      Zadavatel      Čeká na vytvoření článku

Jazyk  CS  EN

Název (CS) Implementace frontendu eshopu

Popis (CS) Zadáním bylo vytvořit uživatelské rozhraní chystaný eshop. Zadání zhostil s nadšením a velice rychle dodal výsledek.

Přínosy (CS) Výstupem bylo uživatelské rozhraní, které bylo téměř bez problémů nasazeno do produkce. Spolupráce mezi zadavatelem a řešitelem nadále pokračuje.

dokumentace.pdf       projekt.zip

Obrázek 4.7: Portlet pro vytvoření návrhu článku zadání

GitLab, fungující pouze pro účely portálu Spolupráce s průmyslem, a proto je zatím integrace ve fázi přípravy.

Aplikace, je připravena pro integraci a to způsobem, který je popsán v části 3.4. Pro komunikaci s aplikací GitLab bude použita existující knihovna GitLab Java API Wrapper [17], která obsahuje všechny potřebné metody pro práci se vzdálenými repositáři, umístěnými na serveru GitLab. Rozhraní GitLab API komunikuje s repositáři jako autentizovaný uživatel. Tento uživatel bude systémový, tedy společný pro celý portál.

Po zahájení spolupráce bude mít zadavatel možnost přiřadit k zadání existující nebo založit zcela nový repositář. Po vytvoření repositáře bude uloženo jeho ID a název do databáze portálu a zároveň dojde k uložení reference na tento záznam u daného zadání.

Podle známého vlastníka (zadavatel) a názvu repositáře lze pak jednoduše sestavit adresu repositáře jak pro náhled, tak pro využívání služeb Git. K repositáři však mají přístup pouze autentizovaní uživatelé. Registraci nového

## Implementace frontendu eshopu

Schvalování návrhu detailu dokončeného zadání

Václav Blažej	Řešitel	Schvaluje článek
Aleš Fišer	Zadavatel	Schválit
Admin	Administrátor	Vytvořit článek

**Název (CS)** Implementace frontendu eshopu

**Popis (CS)** Zadáním bylo vytvořit uživatelské rozhraní chystaný eshop. Řešitel se zadání zhostil s nadšením a velice rychle dodal výsledky.

**Přínosy (CS)** Výstupem bylo uživatelské rozhraní, které bylo téměř bez úprav nasazeno do produkce. Spolupráce mezi zadavatelem a řešitelem nadále pokračuje.

**Příložené soubory**

- [dokumentace.pdf](#)
- [projekt.zip](#)

Navrhnout změny Potvrdit

Obrázek 4.8: Portlet pro schvalování návrhu článku zadání

uživatele již řeší zmíněná knihovna, ale k tomuto kroku je třeba mít administrátorská oprávnění v systému GitLab. Právě získání tohoto oprávnění pro systémového uživatele je jeden z hlavních důvodů, proč je nutností vytvoření verze GitLab pouze pro účely portálu Spolupráce s průmyslem.

Po přiřazení repozitáře k zadání dojde k automatickému přiřazení řešitelů k repozitáři a na detailu zadání se zobrazí jeho adresa. Autentizace všech uživatelů probíhá buď pomocí uživatelského jména a hesla nebo SSH klíčem, který každý uživatel může vyplnit a přistupovat tak k přiřazeným repozitářům přímo z počítače, obsahujícího daný SSH klíč.

Stažení repozitáře může také provést pouze oprávněný uživatel. Proto bude nutné pro účely zveřejnění výsledků stáhnout archiv repozitáře jako systémový uživatel a následně jej poslat uživateli portálu.

## 4. REALIZACE

### Implementace frontendu eshopu

Detail dokončeného zadání

grafický design front-end JavaScript HTML CSS webové služby GUI 16.5.2015


**Řešitel:** Václav Blažej  
**Zadavatel:** Aleš Fišer

**Popis zadání**  
Zadáním bylo vytvořit uživatelské rozhraní chystaný eshop. Řešitel se zadání zhostil s nadšením a velice rychle dodal výsledky. Ty velmi dobře zpracované a podložené kvalitní dokumentací.

**Popis výsledků**  
Výstupem bylo uživatelské rozhraní, které bylo téměř bez úprav nasazeno do produkce. Spolupráce mezi zadavatelem a řešitelem nadále pokračuje.

**Režim ochrany duševního vlastnictví**  
Licence zadavateli a univerzitě ?  
Způsob užití musí respektovat dohodu NDA ?

**Související předměty**  
Programování v Javě  
Programování v jazyku Javascript  
Magisterský projekt



**Příložené soubory**  
dokumentace.pdf  
projekt.zip

Obrázek 4.9: Detail dokončeného zadání

### Dokončená zadání


**Implementace frontendu eshopu** Ukázat detaily →

grafický design front-end JavaScript HTML CSS webové služby GUI

16.5.2015

Zadáním bylo vytvořit uživatelské rozhraní chystaný eshop. Řešitel se zadání zhostil s nadšením a velice rychle dodal výsledky. Ty velmi dobře zpracované a podložené kvalitní dokumentací.

Licence zadavateli a univerzitě ?



Obrázek 4.10: Seznam dokončených zadání



---

# Testování

Testování probíhalo několika způsoby. Zcela nové i upravené stávající procesy byly otestovány pomocí testů, psaných v jazyce Groovy [18]. Pro rychlejší vývoj a ověření všech funkčností aplikační logiky byly tvořeny jednotkové a integrační testy. Testování uživatelského rozhraní probíhalo manuálně s nejaktuálnější verzí portálu, nasazenou na lokálním serveru.

## 5.1 Uživatelské testování

Uživatelské testování probíhalo již ve fázi návrhu. Jednalo se nejen o testování kompletnosti a intuitivnosti navrhovaného uživatelského rozhraní, ale i o validitu veškerých navrhovaných procesů. Potenciálním uživatelům byly podrobně popsány všechny procesy a k jejich jednotlivým bodům zobrazeny příslušné stránky. Na základě tohoto testování bylo nakonec upuštěno například od možnosti hodnocení dokončeného zadání ostatními uživateli.

## 5.2 Testování procesů

Pro otestování funkčnosti procesů v Activiti [12] slouží psané testy v jazyce Groovy. V těchto testech se vytvoří testovací instance procesů, která existuje pouze po dobu testu. Navíc během testování nejsou využívány skutečné implementace servisních tříd, ale pouze jejich mock - implementace s metodami, které proběhnou bez vykonání jakékoliv akce. Díky tomu nejsou ovlivněna data v žádné databázi.

Při testování se ideálně prověří všechny možné stavy, kterých může proces nabývat. Tyto jednotlivé stavy je však nutné definovat explicitně.

Testování probíhá tak, že se stanoví výchozí podmínky (část given:), poté se provede nějaká akce (část when:) a nakonec se chování procesu v této části porovná s chováním očekávaným, které je definováno v části (then:).

## 5. TESTOVÁNÍ

---

```
def "Waiting for admin approval"() {
  given:
  def instance = runtimeService
  .startProcessInstanceByKey(ProcessConstants
  .CompletedAssignmentPublication.ID, args);

  when:
  def requestTaskId = taskLookupService
  .findCompletedAssignmentPublicationRequestTaskId(aId);
  taskService.claim(requestTaskId, assigneeUsername);
  taskService.complete(requestTaskId, [adminMustApprove: true]);

  then:
  1 * notificationSenderService
  .sendEmailToAdminAboutCompletedAssignmentPublicationRequest(aId);
  taskLookupService
  .findCompletedAssignmentPublicationAdminApprovalTaskId(aId)
  != null;
}
```

Algoritmus 5.1: Ukázka testování procesu

Například v ukázce 5.1 je testována část procesu zveřejňování dokončeného zadání, kdy je očekáváno potvrzení od administrátora. V první fázi se zahájí samotný proces, což již bylo otestováno dříve. V druhé části se nalezne Publication Request User Task, po jehož dokončení (taskService.complete) by se měl vytvořit Admin Approval User Task. Zda se vytvořil a zda se zavolala metoda pro posílání emailu kontroluje třetí část.

### 5.3 Jednotkové a integrační testy

Jednotkové testy mají za úkol ověřit funkčnost samostatné jednotky aplikace. Oproti tomu testy integrační ověřují spolupráci více jednotek dohromady. Při vývoji tedy byly prováděny nejprve jednotkové a následně integrační testy.

Princip integračního testu je patrný například z ukázky 5.2, kde je vytvořen testovací návrh článku pomocí assignmentPublicationService. Následně je ověřováno, zda třída completedAssignmentService záznam nalezla.

```
@Test
public void testCreateDraft() {
  final AssignmentPublicationDto testDraft = createTestDraft();
  final Long assignmentId = testDraft.getAssignmentId();
  assignmentPublicationService.saveDraft(testDraft);
  final CompletedAssignmentDetailDto assignmentDetailDto =
    completedAssignmentService
    .findAssignmentDetailDto(assignmentId);
  assert assignmentDetailDto != null;
}
```

Algoritmus 5.2: Ukázka principu integračního testu

---

## Závěr

Cílem práce bylo dosažení lepšího využití výsledků projektů realizovaných přes portál Spolupráce s průmyslem. Pro tento účel byly prozkoumány největší současné nedostatky portálu v tomto směru. Pro jejich odstranění poté byly navrženy nové komponenty systému.

V první řadě se jedná o komponenty, umožňující volbu režimu ochrany duševního vlastnictví autorů odevzdaných řešení. Stanovení duševního vlastnictví práce je totiž klíčové pro možnost jejího případného zveřejnění a dalšího rozvoje.

Uživatelské rozhraní pro uveřejnění dokončeného zadání a s tím související procesy jsou totiž obsahem dalších přidaných komponent. Zadání je zveřejňováno formou článku, na jehož vytváření se podílí nejen administrátor systému, ale i řešitelé a zadavatel. Zveřejnění zadání přispěje k větší transparentnosti portálu a tedy i k tomu, že studenti budou vědět, co v případě spolupráce očekávat. Zároveň takto publikované zadání přispěje ke zviditelnění jeho řešitelů i samotného zadavatele a poslouží tak jako určitá forma reklamy. Ostatní uživatelé navíc dostanou možnost dále rozvíjet odevzdaná řešení. Toto řešení, zejména pak zdrojové kódy vyvinutých aplikací, bude do budoucna možné již během vývoje ukládat do repositáře, hostovaného službou GitLab, jejíž základní ovládání bude probíhat přímo z portálu. Prozatím se však nepodařilo vyjednat podmínky pro založení této služby.

Poslední přidanou částí je zobrazení vzájemného hodnocení spolupráce mezi zadavatelem projektu a jeho řešiteli, které byly realizovány jako vizuální zobrazení hodnocení na profilech těchto uživatelů. Při návrhu komponent byl kladen maximální důraz na zachování soukromí uživatelů portálu, zvláště pak zachování práva na anonymitu průmyslových partnerů v systému.

Tyto komponenty byly realizovány buď jako zcela nové portlety, nebo nové části portletů stávajících. Zároveň byly realizovány veškeré s nimi související procesy. Díky tomu je možné jejich okamžité využití v produkčním prostředí.



---

## Literatura

- [1] *Heureka.cz - Porovnání cen a srovnání produktů z internetových obchodů.* Dostupné z: <http://heureka.cz/>
- [2] *Heureka.cz - Recenze Mall.cz.* Naposledy navštíveno 15.5.2015. Dostupné z: <http://obchody.heureka.cz/mall-cz/recenze/>
- [3] *Aukro - největší obchodní portál.* Naposledy navštíveno 15.5.2015. Dostupné z: <http://aukro.cz/>
- [4] *YouTube.* Naposledy navštíveno 15.5.2015. Dostupné z: <https://www.youtube.com/>
- [5] *Facebook.* Naposledy navštíveno 15.5.2015. Dostupné z: <https://www.facebook.com/>
- [6] *Glen S West profile - GitHub.* Naposledy navštíveno 15.5.2015. Dostupné z: <https://github.com/glennswest>
- [7] *GitHub - Build software better, together.* Naposledy navštíveno 15.5.2015. Dostupné z: <https://github.com/>
- [8] *Stack Overflow - Homepage.* Naposledy navštíveno 15.5.2015. Dostupné z: <http://stackoverflow.com/>
- [9] *Liferay - Documentation.* Naposledy navštíveno 1.5.2015. Dostupné z: <http://www.liferay.com/>
- [10] *JSR 168: Portlet Specification 1.0.* 2003. Dostupné z: <https://jcp.org/en/jsr/detail?id=168>
- [11] *JSR 286: Portlet Specification 2.0.* 2008. Dostupné z: <https://jcp.org/en/jsr/detail?id=286>

## LITERATURA

---

- [12] *Activiti User Guide*. 2014. Dostupné z: <http://www.activiti.org/userguide/>
- [13] *BPMN 2.0 - Specification*. 2011. Dostupné z: <http://www.omg.org/spec/BPMN/2.0/PDF>
- [14] *Pro Git*. První vydání, 2009. Dostupné z: <http://git-scm.com/book/cs/v1>
- [15] *GitLab - Documentation*. Naposledy navštíveno 16.5.2015. Dostupné z: <http://doc.gitlab.com/ce/>
- [16] Fišer, A.: *Posílení zpětné vazby z průmyslu na univerzitu*. České vysoké učení technické v Praze, Fakulta informačních technologií, 2014.
- [17] Olshansky, T.: *GitLab API Wrapper - GitHub*. Naposledy navštíveno 17.5.2015. Dostupné z: <https://github.com/timols/java-gitlab-api/>
- [18] *Groovy - Documentation*. Naposledy navštíveno 10.5.2015. Dostupné z: <http://www.groovy-lang.org/documentation.html>

## Seznam použitých zkratek

**ČVUT** České vysoké učení technické v Praze

**SSP** Spolupráce s průmyslem

**GUI** Graphical User Interface

**XML** Extensible Markup Language

**XHTML** Extensible Hypertext Markup Language

**SQL** Structured Query Language

**BI** Business Intelligence

**SSO** Single Sign-On

**Java EE** Java Platform, Enterprise Edition

**JSR** Java Specification Request

**JSF** JavaServer Faces

**ORM** Object-Relational Mapping

**API** Application Programming Interface

**BPMN** Business Process Model and Notation

**VCS** Version Control System

**DTO** Data Transfer Object

**SSH** Secure Shell





---

## Obsah přiloženého CD

readme.txt.....	stručný popis obsahu CD
text .....	text práce
└─ BP_kralmar3.pdf.....	text práce ve formátu PDF
src	
└─ impl_src .....	zdrojové kódy projektu
└─ text_src.....	zdrojová forma práce ve formátu $\LaTeX$
└─ dist.....	sestavené soubory typu war a jar