

MASARYKOVA UNIVERZITA
FAKULTA INFORMATIKY



Integrace DMS a workflow v Liferay Portal

DIPLOMOVÁ PRÁCE

Marek Tlačbaba

Brno, 2014

Prohlášení

Prohlášení

Vedoucí práce: RNDr. Jaroslav Ráček, Ph.D.

Poděkování

Poděkování

Shrnutí

Abstrak

Klíčová slova

Klíčová slova

Obsah

1	Úvod	2
2	Podnikové portály	3
2.1	Konkrétní portálová řešení	4
2.2	Liferay Portal	4
3	Vývoj portletů	6
3.1	Portlet	6
3.2	Specifikace	7
3.2.1	Životní cyklus portletu	7
3.2.2	Režimy portletu	8
3.2.3	Stavy okna	8
3.2.4	Struktura portálové aplikace	9
4	Workflow	10
4.1	Typy workflow systému	10
4.1.1	Hledisko charakteru procesů	10
4.1.2	Hledisko orientace procesů	11
4.1.3	Hledisko technologické infrastruktury	11
4.2	Obecný model workflow systému	11
4.2.1	Fáze workflow	13
5	Liferay a workflow	15
5.1	Activiti	15
5.2	JBoss's jBPM	15
5.3	Kaleo	15
5.4	Závěrečná volba workflow systému	15
6	Analýza a návrh grafického designeru	16
7	Závěr	17
A	Položky seznamu	19

Kapitola 1

Úvod

Oficiální zadání

Popsat principy tvorby podnikových portálů. Zaměřit se zejména na otázku implementace workflow pro procesy pracující s vnitropodnikovými dokumenty. Pro Liferay Portal verze CE analyzovat, navrhnout, implementovat, otestovat a zdokumentovat vlastní grafický workflow editor pracující v základní notaci BPMN, který bude plně kompatibilní se standardním workflow systémem, který je součástí Liferay Portal CE. Výstup bude mít podobu funkčního prototypu.

Kapitola 2

Podnikové portály

Pojem portál bývá definován různými způsoby v závislosti na situaci, ve které je používán. Dříve se portály zaměřovaly hlavně a jenom kolem jednotného přístupu k informacím nezávisle na jejich původu a umístění. S dalším vývojem informačních technologií se postupně tento přístup dále rozšiřoval a do definice portálu přibýly pojmy jako integrace a agregace služeb a aplikací.

V následujícím textu je uvedeno několik definic portálu a podnikového portálu.

- Například Gála definoval portál jako jednotné rohraní, ve kterém lze pracovat s běžnými službami a nástroji jako jsou, například zpravodajství a komunikace, mimo to je v něm zaručen přístup k všeobecným aplikacím jako jsou vlastní stránky a blogy, a k aplikacím specializovaným, například slovníkům. [1]
- Tato definice portálu je převzata ze standardu JSR-286 a zní následovně: Portál je webová aplikace, která (obvykle) poskytuje možnost personalizace, autentifikace, agregace obsahu z různých zdrojů a slouží jako prezentační vrstva informačních systémů. Agregací se rozumí integrace obsahu z různých zdrojů v rámci jedné webové stránky. Portál může mít vysoce propracované nástroje pro personalizaci, které poskytují obsah upravitelný podle přání uživatele. Stránky portálu mohou mít různé skupiny portletů¹, které vytvářejí obsah pro různé uživatele. [2]
- Předchozí definice se týkají portálu obecně, proto zde uvedu také definici podnikového portálu jako takového. Například podle Čecha je podnikový portál definován jako internetové nebo intranetové stránky, které slouží jako vstupní bod, respektive brána k různým datovým, informačním a znalostním zdrojům v organizaci. Jejich cílem je zpřístupnit tyto zdroje specifické skupině lidí. Může to být jak zákazníkům, tak také vlastním zaměstnancům nebo partnerům. [3]

1. Bližší informace v dalších částech této práce.

V předchozích definicích portálu jsou naznačeny některé základní rysy podnikových portálů. Mezi základní vlastnosti můžeme považovat jediný přístupový bod, integrace, federace, přizpůsobivost, personalizace, kontrola přístupu a vyhledávání v podnikovém obsahu, tak jak jsou uvedeny v [4]

2.1 Konkrétní portálová řešení

Existuje a používá se celá řada konkrétních portálových řešení. Nejčastěji bývají implementovány v jazyce Java a podporují standardy JSR-168 a JSR-286² pro tvorbu portletů v jazyce Java. Existují také portály i pro jiné technologie, například pro platformu .NET je možné využít Microsoft SharePoint, ale nejvíce řešení existuje pro jazyk Java, kde mezi nejznámější patří IBM Websphere Portal, JBoss GateIn, Oracle WebCenter či Liferay Portal. Tyto portály již většinou obsahují velké množství již připravovaných portletů, např. wiki, správa souborů, komunikace a další podobné.

V praktické části této práce bude vytvořen portlet pro tvorbu workflow pomocí grafického designeru v portálu Liferay.

2.2 Liferay Portal

Liferay portal je open-source podnikový portál založený na jazyce Java vyvíjený společností Liferay, Inc.. Podporuje specifikace JSR-168 a JSR-286 pro tvorbu portletů v jazyce Java. Liferay Portal je k dispozici ve dvou edicích, Community Edition a Enterprise Edition.

Community Edition je bezplatná verze portálu, která je volně dostupná ke stažení. Tato verze je ale bez oficiální podpory, ta je dostupná pouze prostřednictvím Liferay komunity. Druhá verze portálu, Liferay Portal Enterprise Edition je placená edice. U této verze je kladen velký důraz na stabilitu, bezpečnost a výkon. K této edici patří dlouhodobá podpora od společnosti Liferay, Inc. nebo jejích partnerů.

Ihned po instalaci portálu je k dispozici množství již předinstalovaných portletů. Je možné například ihned využívat wiki, fórum, kalendář, galerii obrázků. Liferay portál podporuje různé způsoby, jak jej lze upravovat či rozšiřovat. Pro změnu vzhledu můžeme využít témata či šablony, dále funkcionalitu portálu je možno rozšířit pomocí portletů, měnit vlastnosti portálu či portletů je umožněno díky tzv. hooks a větší a zásadnější změny přímo v jádru portálu se provádí pomocí pluginu Ext. [5]

2. Bližší informace v dalších částech této práce

Mezi hlavní a důležité vlastnosti portálu Liferay bývají nejčastěji řazeny tyto: [6]

- zabezpečené jednotné přihlášení,
- nástroje pro správu workflow,
- snadná instalce samotného portálu i nových aplikací a rozšíření,
- od verze 6.1 CE GA2 umožňuje instalaci rozšíření a aplikací z Liferay Marketplace,
- obsahuje systém pro správu obsahu (Liferay CMS), který umožňuje oprávněným uživatelům vytvářet a spravovat obsah webu přímo z prohlížeče i bez znalosti programování,
- vícejazyčné uživatelské rozhraní,
- personalizace portálových stránek,
- jednoduchá úprava stránek způsobem táhni a pusť,
- umožňuje dělit uživatele do organizací, či do skupin uživatelů a podle těchto rolí zobrazovat různý obsah či aplikace,
- automatické nahrávání souborů,
- podpora pro různé platformy - různé aplikační servery, operační systémy a databáze,
- podporuje různé metody integrace včetně SOAP, REST, RSS a také různá proprietární rozhraní.

To jsou některé z důvodů, proč se podnikový portál Liferay stal rozšířenou a oblíbenou platformou pro vývoj firemních webů a systému v jazyce Java.

Kapitola 3

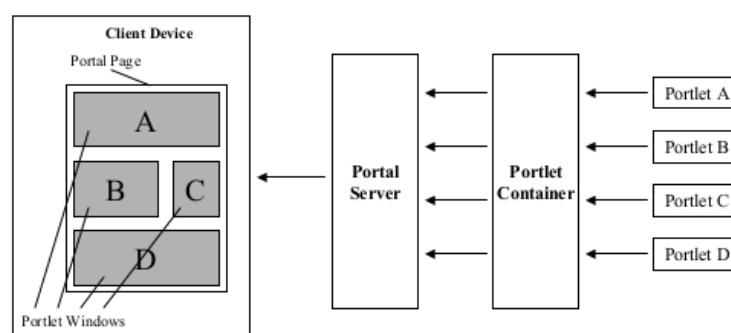
Vývoj portletů

V této kapitole bude popsán vývoj portletů, které tvoří obsah portálů.

3.1 Portlet

Portlet je webová aplikace, která poskytuje uživatelům specifický obsah, typicky informaci nebo službu. Portlet je spravován portletovým kontejnerem, který zpracovává požadavky portletu a následně generuje dynamický obsah v podobě fragmentů v některém ze značkovacích jazyků – HTML, XHTML, WML. Tyto fragmenty jsou pak spojovány a dohromady vytváří portálovou stránku.

Portletový kontejner má tedy opovědnost za řízení životního cyklu jednotlivých portletů a je v něm také uloženo nastavení uživatele pro daný portlet. Ale jednotlivé části generované jednotlivými portlety slučuje dohromady na stránku portál. Portál a portletový kontejner mohou postaveny jako jedna komponenta ale mohou být také jako komponenty dvě. Na následujícím obrázku 3.1 je zobrazen průběh vytváření stránky v portálu tak, jak bylo již výše naznačeno.



Obrázek 3.1: Vytváření stránky v portálu (převzato z [2])

3.2 Specifikace

Důvodem pro vytvoření portletové specifikace byla situace, kdy různí dodavatelé používali svá vlastní API, která neumožňovala přenositelnost mezi portálovými servery.

První verze Java Portlet Specification 1.0 (JSR-168) byla vydána v roce 2003. V této specifikaci ještě bylo stále různé nedostatky a chyby, které pak tvůrci portletů stejně řešili každý po svém. To způsobovalo i nadále, že portlety byly mezi portálovými servery stále problematicky přenositelné či nepřenositelné úplně. Proto vznikla druhá verze Java Portlet Specification 2.0 (JSR-286) a ta byla vydána v červnu 2008. Mezi nejdůležitější přínosy patří následující:

- sdílení parametrů mezi portlety skrz veřejné render parametry,
- meziportletová komunikace pomocí zasílání událostí,
- transformace informací nově definovanými portletovými filtry,
- možnost portletů poskytovat zdroje jako jsou například PDF soubory.

Většina portletových kontejnerů sice poskytuje k základním požadavkům i svá rozšíření. Tyto rozšíření však nemusíme využívat a tak můžeme zachovat myšlenku snadné přenositelnosti.

3.2.1 Životní cyklus portletu

Každý portlet musí implementovat rozhraní **Portlet**. Toto rozhraní obsahuje čtyři základní metody, které musí být implementovány a které řídí životní cyklus portletu - **init()**, **processAction()**, **render()**, **destroy()**.

Metoda **init()** je volána ihned při vytváření portletu a je používána například pro přípravu zdrojů potřebných portletem.

Metoda **processAction()** je volána vždy po provedení nějaké akce uživatelem, když se touto akcí nějak mění stav portletu. Může to být například akce potvrzení formuláře a následné zpracování a uložení dat v databázi.

Metoda **render()** má za úkol generovat fragment stránky, který je pak dostupný uživateli v portletu. V této metodě by se neměl měnit stav portletu.

Poslední metodou je **destroy()**. Ta je volána před odstraněním portletu z paměti. Poskytuje poslední možnost pro uvolnění zdrojů, které byly portletem používány. [2]

Zpracování požadavků v portletech může procházet několika fázemi zpracování.

- **Render** – vykreslovací fáze, ve které se typicky nemění stav portletu, odpovídá jí metoda `render()`.
- **Action** – portlet zpracovává nějakou akci a typicky mění svůj stav, odpovídá jí metoda `processAction()`.
- **Resource** – portletem je poskytován nějaký zdroj, např. PDF soubory. Aby portlet mohl podporovat tuto funkcionalitu, musí implementovat rozhraní `ResourceServingPortlet`.
- **Event** – portlet přijímá nějakou událost a nějak na ni reaguje. Portlet v tomto případě musí implementovat rozhraní `EventPortlet`.

Všechny tyto rozhraní implementuje abstraktní třída `GenericPortlet`. Tato třída obsahuje implementaci všech povinných metod a několika dalších, které je možné využít.

Po každé action fázi musí následovat fáze render. Pokud se na portálové stránce vyskytuje více portletů, je metoda render zavolána u všech zobrazených portletů.

3.2.2 Režimy portletu

Portletová specifikace obsahuje tři režimy portletu, které umožňují zobrazit jiný obsah v závislosti na požadovaném úkolu. Tyto režimy využívá metoda `render()`, která podle nich vygeneruje odpovídající obsah.

- **View** – základní režim používaný při klasickém zobrazení portletu na portálové stránce.
- **Edit** – nepovinný, uživateli umožňuje nastavovat portlet.
- **Help** – nepovinný, zobrazuje se nápověda k danému portletu.

Ve třídě `GenericPortlet` jsou k těmto účelům definovány metody `doView()`, `doEdit()` a `doHelp()`. Jednotlivé portletové režimy by měli být implementovány přetížením těchto metod. Každý portál si také může definovat vlastní portletové režimy. Každý portlet musí mít zdefinováno, které režimy podporuje.

3.2.3 Stavy okna

Portletová specifikace definuje tři základní stavy.

- **Minimized** – okno portletu je minimalizované.

- Normal – původní velikost portletu, které umožňuje zobrazení všech portletů na portálové stránce.
- Maximized – okno portletu skryje ostatní portlety na stránce a zobrazuje se pouze tento maximalizovaný portlet, nebo zakrývá většinu dostupné plochy.

3.2.4 Struktura portálové aplikace

Portálové aplikace jsou uloženy ve standardní archivním souboru jazyka Java, Web ARchive (WAR). Mohou obsahovat více portletů, servlety, JSP soubory a další. Musí obsahovat webový deskriptor a také navíc portletový deskriptor implementace `portlet.xml`.

Kapitola 4

Workflow

Jednoznačná definice workflow neexistuje, jelikož každá oblast, ve které je workflow používáno, si ho definuje po svém. Obecně je ale možné definovat workflow jako proces automatizace podnikových procesů. O sjednocení této terminologie se již v roce 1996 pokusila institut Workflow Management Coalition (WfMC), kdy vydal terminologický slovník, ve kterém je workflow definováno takto:

Workflow znamená automatizaci celého nebo části podnikového procesu, během kterého jsou dokumenty, informace nebo úkoly předávány od jednoho účastníka procesu k druhému podle sady procedurálních pravidel tak, aby se dosáhlo nebo přispělo k plnění celkových/globálních podnikových cílů. [8]

V počítačových systémech je workflow definováno jako systém řízení workflow. V tomto systému je možné workflow definovat, vytvořit a celý průběh procesu řídit. Mezi jeho úkoly patří provádět definici procesu, komunikovat s účastníky workflow, případně spouštět další aplikace nutné k vykonání workflow.

4.1 Typy workflow systému

Workflow systémy se mohou dělit do několika skupin podle různých hledisek či typů. [9]

4.1.1 Hledisko charakteru procesů

- **Administrativní** workflow systémy jsou určeny k vyřizování běžných administrativních úkonů, které jsou většinou jednoduché, opakující se a dobře strukturované.
- **Ad-hoc** workflow systémy jsou založeny na náhodnosti vzniku. Procesy bývají jedinečné a je nutné je definovat až při jejich vzniku.
- **Kolaborativní** workflow systémy podporují především týmovou spolupráci. Zde je typickým znakem existence nějakého dokumentu, skrze

který si uživatelé vyměňují poznatky a většinou se tento dokument stává i výsledkem společné práce. Často obsahuje opakování několika iterací stejného kroku dokud nedojde ke shodě.

- **Produkční** workflow systémy podporují hlavní podnikové procesy, které vytvářejí přidanou hodnotu k finálnímu produktu.

4.1.2 Hledisko orientace procesů

- **Procesy orientované na lidi.** U těchto systémů spolehají účastníci sami na sebe. Předávané informace jsou proměnlivé a procesy nejednotné a špatně předpověditelné. Z těchto důvodů jsou průběhy závislé na jednotlivcích.
- **Procesy orientované na sebe.** Tyto systémy jsou zaměřené na klíčové procesy, které obvykle bývají hlavními aktivitami podniku. Jejich pravidla řešení a jejich zpracování jsou pevně daná.

4.1.3 Hledisko technologické infrastruktury

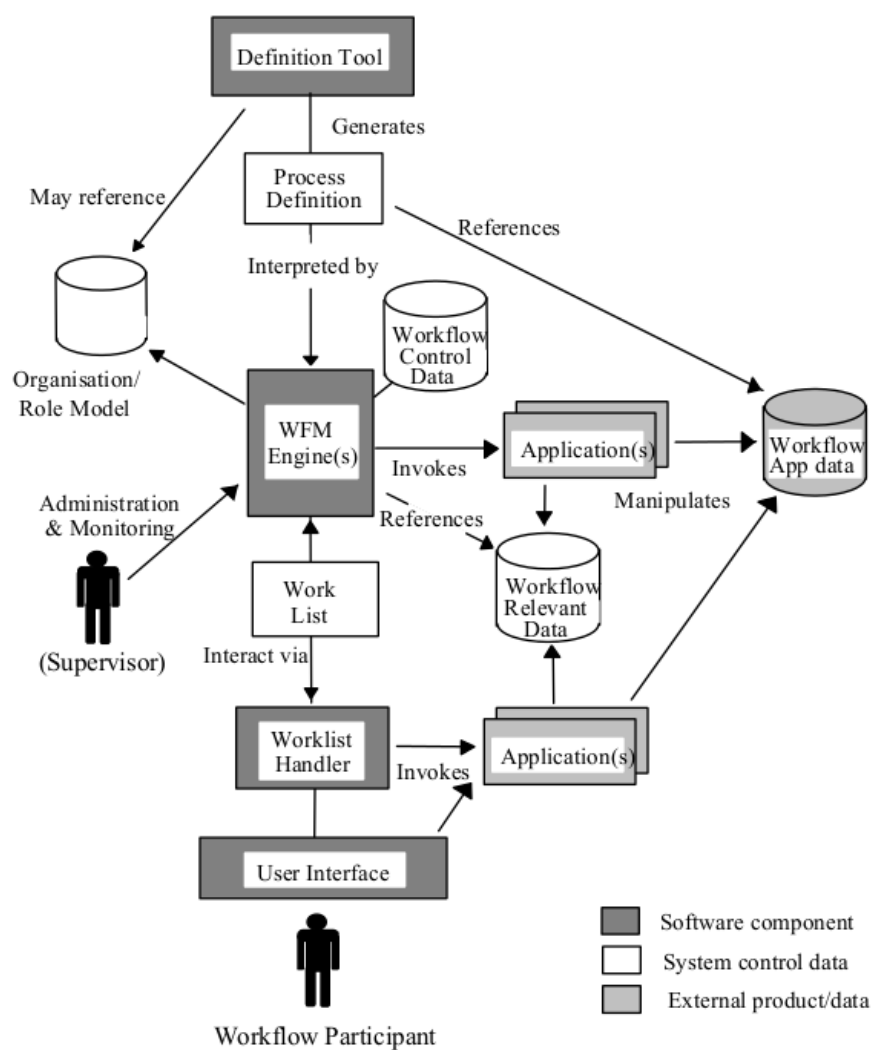
Podle technologické infrastruktury, nad kterou je systém workflow postaven, lze produkty rozdělit do několika skupin.

- Založené na **elektronické poště** – využívají dostupné emailové servery. Uživatelé nepotřebují speciální software.
- Založené na **procesech** – mohou implementovat vlastní komunikační mechanismus a jsou postaveny na určitém databázovém systému. Bývají tvořeny jako komplexní řešení uceleného konceptu workflow.
- Založené na **webu** – využívají jednotného rozhraní intranetové aplikace. Je to univerzální platforma pro sdílení informací.
- Založené na **dokumentech** – tyto systémy byly motivovány představou o směřování dokumentů. Využívají systémy pro správu dokumentů.

4.2 Obecný model workflow systému

Institut WfMC vytvořil obecný model, kde mají produkty podobné komponenty viz obr. 4.1. Jednotlivé komponenty systému se dělí na programové a datové.

Programové komponenty.



Obrázek 4.1: Obecný model workflow (převzato z [8])

- Nástroj pro definici procesů (definition tool) umožňuje definovat procesy, přiřazovat role a stanovat pravidla.
- Výkonné jádro workflow (workflow engine) řídí průběh workflow procesů, spouští nutné externí aplikace a udržuje statistiky o průběhu workflow.
- Správce úkolů (worklist handler) má jako hlavní úkol zprostředkovávat komunikaci mezi jádrem workflow a jednotlivými uživateli.
- Uživatelské rozhraní (user interface) zajišťuje komunikaci mezi správcem úkolů a uživatelem. Často tvoří jeden celek spolu se správcem úkolů.

Datové komponenty.

- Definice procesu (process definition) popisuje strukturu procesu.
- Řídící data workflow (workflow control data) jsou interní data, které zpracovává jádro systému.
- Aplikační data workflow (workflow application data) jsou specifická data aplikací.
- Věcná data workflow (workflow relevant data) jsou data používaná jádrem k vyhodnocování dalších kroků.
- Seznam úkolů (work list) představuje datovou strukturu, ve které jsou uloženy úkoly pro uživatele. Mohou být viditelné všechny najednou, nebo mu úkoly mohou být poskytovány postupně.
- Model organizační struktury (organization/role model) popisuje organizační strukturu podniku. Pokud není definován, musí být úkoly přiřazovány pouze konkrétním uživatelům.

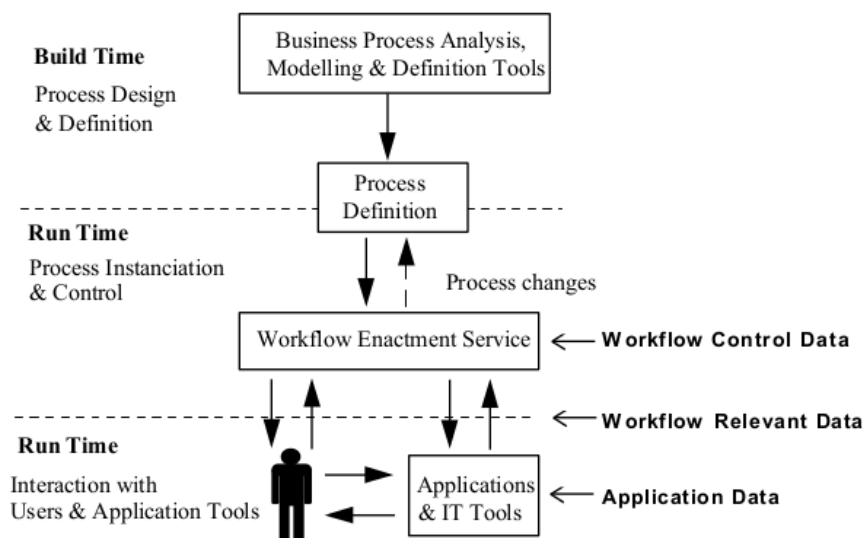
4.2.1 Fáze workflow

Obecně jsou rozlišovány dvě fáze workflow: fáze návrhu workflow a fáze průběhu workflow. [9]

Fáze návrhu workflow v sobě zahrnuje funkce pro návrh a definici procesu, které jsou poskytovány analytickými a modelovacími nástroji. Ty umožňují převod procesu z reálného světa do normalizované podoby. Výsledkem je počítačově zpracovatelný popis procesu ve smyslu: kdo, kdy, co, s čím, za jakým účelem a jak má udělat.

Fáze průběhu workflow se dále dělí na funkce pro řízení běhu procesu, které zabezpečují interpretaci procesu, spouštění, provádění a kontrolu průběhu jednotlivých činností. Další část této fáze představuje interakce s uživateli a aplikačními nástroji. To například znamená předávání úkolů ke zpracování, vyžádání manuální činnosti, automatické spouštění jiných aplikací či předávání dat mezi aplikacemi.

Vše znázorněno na následujícím obrázku.



Obrázek 4.2: Fáze workflow (převzato z [8])

Kapitola 5

Liferay a workflow

Portál Liferay implementuje služby pro workflow tak, že může být využito skrze celý portál a může být využito jak pro publikování webového obsahu, schvátování článku blogu, pro práci s dokumenty a v dalších oblastech.

Liferay je v základní edici dodáván s workflow systémem Kaleo, který je jednoduchý a snadnou použitelný a plně dostačuje k základnímu použití pro práci s webovým obsahem nebo pro práci s dokumenty. Pro složitější a komplexnější workflow umožňuje Liferay využít i jiné a rozsáhlejší workflow systému, jako například Activiti nebo JBoss's jBPM. Vždy ale musí být použit na portálu právě jeden workflow systém, proto je nutné před nasazením zvolit ten nejvhodnější.

5.1 Activiti

5.2 JBoss's jBPM

5.3 Kaleo

5.4 Závěrečná volba workflow systému

Kapitola 6

Analýza a návrh grafického designeru

Kapitola 7

Závěr

Literatura

- [1] GÁLA, L. et al. *Podniková informatika*. 1. vyd. Praha: Grada Publishing, 2006.
- [2] *Java™ Portlet Specification*, The Java Community Process, dostupný na <http://www.jcp.org/en/jsr/detail?id=286> [cit. 2014-03-27].
- [3] ČECH, P. *Přínos podnikových portálů pro management znalostí*, in Internet a konkurenceschopnost, Zlín, 2004.
- [4] *Enterprise portal*, Wikipedia, dostupný na http://en.wikipedia.org/wiki/Enterprise_portal [cit. 2014-03-27].
- [5] *Liferay Portal 6.2 Developer's Guide*, Liferay Inc., dostupný na <http://www.liferay.com/documentation/liferay-portal/6.2/development> [cit. 2014-04-02].
- [6] *Portal Features*, Liferay Inc., dostupný na <http://www.liferay.com/products/liferay-portal/features/portal> [cit. 2014-04-02].
- [7] SARIN, Ashish. *Portlets in action*, Shelter Island, NY: Manning, 2011.
- [8] *Terminology & Glossary*, Workflow Management Coalition, dostupný na http://www.wfmc.org/standards/docs/TC-1011_term_glossary_v3.pdf [cit. 2014-04-14].
- [9] CARDA, A., KUNSTOVÁ, R. *Workflow : Nástroj manažera pro řízení podnikových procesů*. 2. vyd. Praha: Grada Publishing, 2003.

Příloha A

Položky seznamu

asdfsadf