

Univerzita Karlova v Praze

Matematicko-fyzikální fakulta

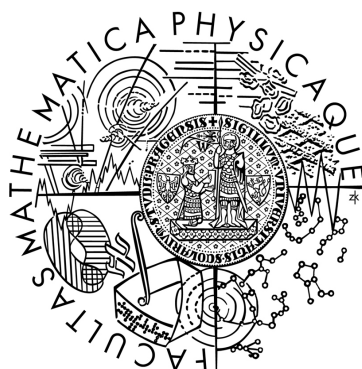
BAKALÁŘSKÁ PRÁCE

2013

Martin Pospíšil

Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

BAKALÁŘSKÁ PRÁCE



Martin Pospíšil

Workflow management systém pro řízení redakce

Katedra softwarového inženýrství

Vedoucí bakalářské práce: RNDr. Michal Kopecký, Ph.D.

Studijní program: Informatika

Studijní obor: Programování

Praha 2013

Děkuji panu RNDr. Michal Kopeckému, Ph.D. za odborné vedení práce, za cenné připomínky k textu a za čas, který mi věnoval.

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V Praze dne 2.8.2013

Martin Pospíšil

Název práce: Workflow management systém pro řízení redakce

Autor: Martin Pospíšil

Katedra / Ústav: Katedra softwarového inženýrství

Vedoucí bakalářské práce: RNDr. Michal Kopecký, Ph.D., Katedra softwarového inženýrství

Abstrakt: Obsahem práce je návrh a implementace redakčního systému podporujícího řízení procesů ve společnosti zabývající se provozováním webových portálů zaměřených na tematiku lékařské vědy a medicíny. Cílem je zefektivnit práci redakce, zpřehlednit její řízení tak, aby ubylo manuálních činností, které pracovníci redakce provádějí, a tím snížit náklady na provoz redakce. Redakční systém zajišťuje kompletní evidenci a řízení procesu vzniku a vydávání článků. Následně umožňuje automatické vytváření výkazů práce, které slouží jako podklady pro vyplácení odměn pracovníkům redakce.

Klíčová slova: redakční systém, workflow management, zdravotnictví

Title: Editor's office workflow management system

Author: Martin Pospíšil

Department: The Department of Software Engineering

Supervisor: RNDr. Michal Kopecký, Ph.D., The Department of Software Engineering
Abstract: The goal of this bachelor thesis is to analyze and implement an editorial system that supports the process management in a company which operates and administrates several web portals aimed to health care and medical science. The goal is to make the work within the company more effective and make the process management more transparent to decrease the amount of work that must be performed manually. It should lead to reducing the costs. The system ensures a complete evidence and management of the process of creating and publishing articles. Consequently it allows to automatically generate work reports which serve as the records for paying the editorial's office staff.

Keywords: editorial and content management systém, workflow management, health care

Obsah

Úvod.....	9
1.Základní koncepce a pojmy.....	11
1.1.Popis stavu před vznikem redakčního systému.....	11
1.2.Cíle nového systému.....	12
1.3.Existující alternativní řešení.....	13
2.Specifikace požadavků.....	15
2.1.Slovník pojmů.....	15
2.2.Uživatelské role.....	16
2.3.Popis životního cyklu článku.....	16
2.4.Popis životního cyklu výkazu práce.....	18
2.5.Základní případy užití.....	19
2.6.Ostatní požadavky na systém.....	25
2.7.Nefunkční požadavky.....	26
3.Analýza a návrh řešení.....	27
3.1.Použité pojmy z teorie workflow.....	27
3.2.Architektura systému.....	27
3.3.Návrh workflow modulu.....	28
3.4.Logický datový model.....	29
4.Programátorská dokumentace.....	33
5.Uživatelská dokumentace a přístup do aplikace.....	35

Úvod

Cílem práce je navrhnout, implementovat a uvést do provozu informační systém pro řízení činnosti redakce (dále jen „Redakční systém“) ve společnosti zabývající se tvorbou webových projektů zaměřených na problematiku medicíny a zdravotnictví. Společnost zaměstnává tým lékařů, kteří jsou autory obsahu webových projektů, a řídicích a administrativních pracovníků, jejichž úkolem je řídit chod redakce. Mezi činnostmi, které provádí lékaři, patří psaní odborných článků z různých oborů lékařské vědy, psaní textů se zdravotnickou tematikou pro širokou veřejnost a interakce s uživateli webových portálů (odpovědi v poradnách, řízení diskusních fór, ...). Administrativní a řídicí činnosti zahrnují řízení týmu autorů článků, správu a schvalování obsahu textů, korektury, určování, vyhodnocování a schvalování odměn za práci v rámci redakce, atd. Všechny tyto činnosti byly dříve prováděny s pomocí několika běžných kancelářských softwarových systémů, které nebyly vzájemně propojeny. Komunikace mezi pracovníky a předávání dat probíhalo pouze s využitím emailů, a tak neexistovala možnost centralizovaně řídit a kontrolovat chod redakce. Z této situace vyšel požadavek na vznik systému, který by tyto nedostatky odstraňoval a umožňoval jednotným způsobem řídit procesy týkající se tvorby a zprávy obsahu webových portálů.

Základními požadavky na systém jsou implementace řízení a automatizace procesů probíhajících v redakci a vytvoření centrálního úložiště dat potřebných k provádění těchto procesů. Příklady takových procesů jsou zpracování článků a zpracování podkladů pro odměňování a vykazování práce členů redakčního týmu. Jak bude dále popsáno, tyto dva procesy zahrnují všechny nejpodstatnější činnosti redakce, a proto jejich implementace tvoří i podstatnou část aplikace a v jejím uživatelském rozhraní je na ně kladen největší důraz. Na základě manažerských rozhodnutí mohou v systému vznikat nové procesy a stávající procesy se v čase mohou měnit tak, aby probíhaly co nejefektivněji. Proto je součástí aplikace obecný workflow management systém, který umožňuje bez změny kódu upravovat a přidávat definice procesů. Tyto definice mají jednotnou strukturu pro všechny druhy datových entit, na které jsou aplikovány (např. článek nebo výkaz práce), a proto je lze jednoduše rozšiřovat a aplikovat i na jiné druhy entit. Jak bude podrobněji popsáno v dalších kapitolách, workflow management systém je dokumentově orientovaný, což – oproti workflow

systemům orientovaným na aktivity – přináší výhodu přímého propojení obsahu datových entit („dokumentů“) s procesem. Doplňujícími funkčními požadavky na systém jsou možnost správy uživatelských nastavení a práv na základě přidělování uživatelských rolí, možnost tvorby měsíčních edičních plánů, exporty článků do běžných formátů, exporty souhrnných reportů a propojení systému s hodnocením článků od čtenářů.

Z nefunkčních požadavků je podstatný požadavek na využití infrastruktury jednotného uživatelského rozhraní pro vnitro-firemní aplikace, kterou zadavatel využívá i u jiných svých aplikací. Tato infrastruktura využívá technologie PHP a *My-SQL*. Proto je musí využívat i aplikace, která je předmětem této práce. Součástí této infrastruktury jsou obecné komponenty pro zobrazování, filtrování a úpravy dat z databáze, jednotný vzhled základní struktury uživatelského rozhraní (moduly, menu, záložky, ...), a proto jejich implementace není součástí práce. Tam kde to nebude zřejmé, bude explicitně uvedeno, co je komponenta nebo část existující infrastruktury a co je nově implementovaná funkcionality.

Práce je členěna do následujících kapitol:

Kapitola 1 definuje základní pojmy z byznys domény zadavatele a popisuje základní koncepci systému.

Kapitola 2 podrobněji popisuje uživatelské požadavky na systém.

Kapitola 3 analyzuje požadavky, popisuje návrh řešení a porovnává jej s alternativními řešeními, které nebyly použity.

Kapitola 4 obsahuje programátorskou dokumentaci a popis datového modelu.

1. Základní koncepce a pojmy

Redakce je tým lidí, který má na starost obsah webových portálů, které patří do portfolia zadavatele. Tyto portály se zabývají výhradně tematikou medicínské vědy a zdravotnictví. Základním prvkem tvořícím obsah webových portálů je článek. Proto většina důležitých procesů prováděných v rámci redakce pracuje s článkem, což znamená, že aktéři těchto procesů buď tvoří nebo upravují článek, případně mění jeho další atributy a stavy s cílem vydat tento článek na příslušném webovém portále. Tento proces je nazýván životním cyklem článku a dále zahrnuje mimo jiné i zajištění korektnosti článku po věcné a gramatické stránce, včasné dodání článku a správné finanční ohodnocení autora článku. Podrobně je životní cyklus článku popsán v Kapitole 3.

Další obsahový prvek portálu, jehož zpracování je součástí požadavků na systém, je odpověď odborníka v lékařské poradně. Podobně jako u článků je zde požadavek na finanční ohodnocování autorů těchto odpovědí.

1.1. Popis stavu před vznikem redakčního systému

Prvotním impulsem pro rozhodnutí o tvorbě nového systému pro řízení redakce byla snaha o zefektivnění a zpřehlednění její činnosti. Během sběru požadavků na nový systém byly zjištěny tyto nedostatky ve stávajícím procesu řízení redakce:

- Neexistuje jednotné úložiště článků. Tyto jsou předávány mezi aktéry procesu pomocí e-mailů a upravovány na lokálních pracovních stanicích v textových editorech. Neexistuje on-line přístupný přehled o stavu článků.
- Celý proces zpracování článku od jeho vytvoření, přes korekturu, schvalování, odborné posuzování obsahu až po vydání je řízeno manuálně šéfredaktorem pomocí e-mailů a evidence v tabulkovém procesoru.
- Nelze automatizovaně hlídat termíny provádění jednotlivých úkonů nad články. Opět je vše kontrolováno manuálně šéfredaktorem.
- Autoři článků a další pracovníci musí manuálně evidovat provedenou práci a tuto odesílat e-mailem ke schválení manažerovi redakce, tak aby mohla být vyplacena odměna za jejich práci. Kontrola správnosti údajů v evidenci práce

je velmi náročná kvůli nutnosti manuálního ověřování všech údajů (např. počet slov v článku atd.)

1.2. Cíle nového systému

Na základě analýzy nedostatků v řízení redakce byly stanoveny následující cíle pro nový systém:

- Mělo by existovat jednotné úložiště článků, které bude přístupné on-line všem členům týmu redakce a bude poskytovat informace o aktuálním stavu článků a nad nimi běžících procesů.
- Řízení procesu životního cyklu článku by mělo být automatizované bez nutnosti manuálních zásahů šéfredaktora, pokud proces probíhá standardně. Součástí tohoto řízení bude směřování jednotlivých úkolů na správné aktéry procesů a také podpora provádění těchto úkolů na úrovni uživatelského rozhraní aplikace.
- Měla by existovat automatizovaná kontrola a sledování termínů provedení jednotlivých úkolů, které jsou součástí příslušného procesu (Např. termín dodání a vydání článku).
- Měly by být odesílány automatické notifikace o důležitých událostech v systému a o požadavcích na provedení úkolů příslušnými uživateli.
- Výkazy práce sloužící jako podklady pro finanční ohodnocení pracovníků by se měly automaticky generovat na základě dat evidovaných v systému.
- Veškeré informace a podklady potřebné pro řešení úkolů souvisejících s příslušnými procesy by měly být evidovány systémem, takže nebude nutné jejich předávání jiným způsobem, např. e-mailem.
- Systém by měl umožňovat definovat uživatelská práva založená na rolích. Tato práva dovolují omezit přístup jak k celým modulům, tak k datům, a to v závislosti na stavu workflow, které v kontextu příslušných dat probíhá (Např. text článku, který je již po korektuře, nemůže být autorem upravován, ale může být zobrazen, atd.).

1.3. Existující alternativní řešení

Před začátkem implementace vlastního řešení byla zvažována i varianta použití dostupného existujícího CMS systému, kterých je na trhu poměrně velké množství. V následujícím přehledu jsou popsány základní výhody a nedostatky třech systémů, které připadaly v úvahu:

. *WordPress*

Jde o velmi rozšířený systém, který vyniká jednoduchostí a snadnou obsluhou. Je velmi oblíbený například pro vytváření blogů, což bylo i původní zaměření tohoto systému. Nevýhodou systému slabá podpora workflow a také jeho původní zaměření na drobné projekty a blogy.

. *Plone*

Variabilní CMS systém, který disponuje možností dodatečného přidávání rozšiřujících modulů. Díky větší komplexnosti nemá tak intuitivní uživatelské rozhraní jako *WordPress*. Obsahuje podporu uživatelských rolí a workflow.

. *Joomla!*

CMS systém s podobnou úrovní rozšiřitelnosti jako *Plone*. Nabízí také podporu workflow managementu a správy uživatelských rolí a práv. Výhodou proti *Plone* je existence většího množství doplňků.

Všechny existující CMS systémy jsou ze své podstaty orientovány především na správu obsahu dokumentů a jejich publikování. Na řízení procesů v redakci není kladen takový důraz, protože jde o funkcionalitu, kterou nelze vždy dostatečně obecně navrhnout tak, aby vyhovovala všem zákazníkům. Pokud navíc požadavky zadavatele obsahují i integraci návazných procesů, jako je vykazování práce pro vyplácení odměn, je varianta použití hotového CMS systému nerealizovatelná. Druhou velkou nevýhodou jsou jejich omezené možnosti v oblasti workflow managementu.

Další alternativou by mohl být nákup komplexnějšího podnikového informačního systému, do kterého by byl CMS systém zaintegrovan. Taková varianta je ovšem pro zadavatele nepřijatelná z ekonomických důvodů.

Poslední zvažovaný způsob řešení bylo použití existujícího obecného workflow management systému (např. *Joget*), který by byl rozšířen o prvky specifické pro řízení redakce a o prvky CMS systému. Hlavním úskalím takového přístupu je právě obecnost takových systémů, kde náklady na jejich rozšíření a přizpůsobení by byly pravděpodobně vyšší než vývoj vlastního řešení šitého na míru a zároveň by obecnost takového systému nebyla plně využita. Nevýhodou by bylo i oddělení uživatelského rozhraní pro provádění procesů od CMS části systému.

Následuje výčet hlavních důvodů pro volbu varianty vlastního vývoje:

1. Existence PHP knihoven a infrastruktury pro tvorbu jednotného uživatelského rozhraní

Zadavatel používá tuto infrastrukturu i pro své ostatní vnitro-firmní aplikace a jeho využití se předpokládá i u Redakčního systému. Díky tomu nebude nutné vyvíjet obecně použitelné moduly a třídy, které slouží např. pro zobrazování, filtrování a editaci dat uložených v relační databázi, správu uživatelských účtů, definování jednotného vzhledu menu a dalších ovládacích prvků, které jsou součástí každého uživatelského rozhraní. Další výhodou je snadnější integrace jiných systémů vyvinutých nad touto platformou a také to, že uživatelé si nemusí zvykat na jiný styl uživatelského rozhraní.

2. Existence požadavku na integraci modulu pro vykazování práce

Dostupné CMS systémy nemohou pokrývat tak specifické požadavky jako je např. kalkulace odvedené práce podle počtu normo-stran, znaků nebo řádků v textu, definice sazebníků pro různé druhy článků, generování textu dohody o provedení práce ve formátu PDF apod.

3. Existence požadavku na implementaci uživatelských práv závislých na stavu článku v průběhu workflow procesu

Jedním z důležitých funkčních požadavků je možnost povolit nebo zakázat přístup k jednotlivým atributům článku nejen v závislosti na uživatelských rolích, ale také na stavu článku v rámci procesu jeho zpracování. Tuto funkcionalitu nemá žádný z uvažovaných CMS systémů.

2. Specifikace požadavků

Klíčovým funkčním požadavkem na nový systém byla automatizace procesů pro zpracování článků a zpracování výkazů práce. Proto je na úvod uveden popis životního cyklu článku a výkazu práce ve formě stavových diagramu. Dále jsou popsány základní případy užití a nakonec ostatní požadavky na podpůrné funkce systému (Uživatelská práva, definice sazebníků, atd.) a nefunkční požadavky.

2.1. Slovník pojmů

Článek – Text s lékařskou tematikou, který bude uveřejněn na některém z webových portálů zadavatele. Článek si nese také další informace jako je nadpis, datum vzniku, autor, použité zdroje, atd.

Projekt – Konkrétní webový portál, pro který jsou připravovány články. Každý článek spadá do jednoho projektu, což znamená, že od začátku psán s cílem vydat jej na odpovídajícím webovém portále.

Ediční plán – Předpis toho, jaké články by měly být vydány pro konkrétní projekt v daný měsíc. Za vytváření edičních plánů jsou odpovědní projekt manažeři (Viz kapitola Uživatelské role). Každý ediční plán má alespoň jednu položku.

Položka edičního plánu – Předpis pro vytvoření jednoho konkrétního článku pro daný ediční plán. Položka obsahuje nadpis nebo téma požadovaného článku. Z položek edičního plánu vytváří šéfredaktor nové články, které jsou následně zadávány ke zpracování autorům.

Výkaz práce - Soupis odvedené práce konkrétního pracovníka za daný měsíc. Slouží jako předpis pro vygenerování textové podoby faktury nebo dohody o provedení práce. Například pro roli Autor obsahuje výkaz seznam napsaných článků za příslušný měsíc, částku, která má být vyplacena, atd.

Akce – Konkrétní činnost nebo úloha, která má být provedena v rámci konkrétního workflow procesu nad článkem, výkazem práce nebo jinou entitou v systému. V uživatelském rozhraní je implementována jako ovládací prvek, který je přístupný vždy v kontextu procesu podle role uživatele a stavu subjektu nad kterým běží workflow proces.

Modul – Část aplikace, která implementuje ucelenou část požadavků. Na uživatelském rozhraní je reprezentován položkou menu na nejvyšší úrovni a zobrazuje se jako samostatná stránka v prohlížeči. Příklady modulů jsou „Články“, „Výkazy práce“, „Sazebník cen“, „Ediční plány“.

2.2. Uživatelské role

Kapitola popisuje uživatelské role a jejich základní činnosti, které uživatelé s těmito rolemi provádějí v Redakčním systému.

Šéfredaktor – Uživatel s touto rolí dohlíží nad správným prováděním procesů nad články. Mezi důležité akce, které může v systému provádět patří např. schvalování článků, zadávání článků na autory, atd.

Projekt manažer – Role určená pro uživatele, který je odpovědný za konkrétní projekt. Projekt manažer vytváří ediční plány a účastní se také procesu zpracování článku. Provádí schvalování zdrojů článků a koordinuje komunikaci s klientem, pro kterého je projekt vytvořen.

Autor – Uživatel, který tvoří texty článků.

Medical advisor – Uživatel, který provádí kontrolu obsahu textů po odborné stránce.

Korektor – Uživatel, který provádí korekturu textů článků.

Manažer – Uživatel, který dohlíží nad fungováním redakce jako celku a provádí schvalování výkazů práce.

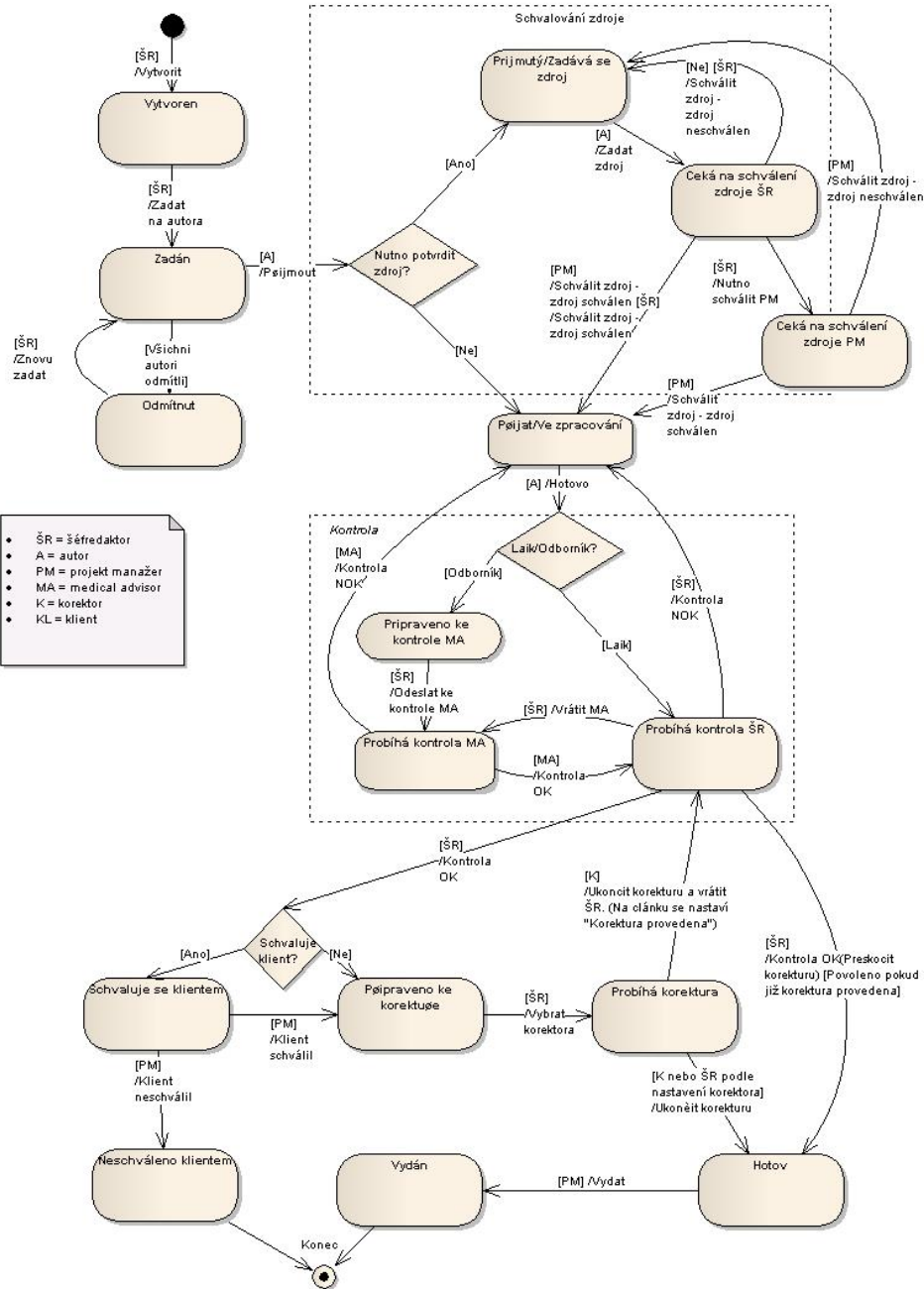
Účetní – Uživatel odpovědný za evidenci výkazů práce a jejich vyplácení pracovníkům.

2.3. Popis životního cyklu článku

Životní cyklus článku je popsán stavovým diagramem, viz Obrázek 1.

Přechody mezi stavy představují uživatelské akce, které uživatel může provést nad článkem v daném stavu a určují přípustné scénáře pro zpracovávání článku. V hranatých závorkách je uvedena role, která může danou akci vykonávat.

Stavy článku



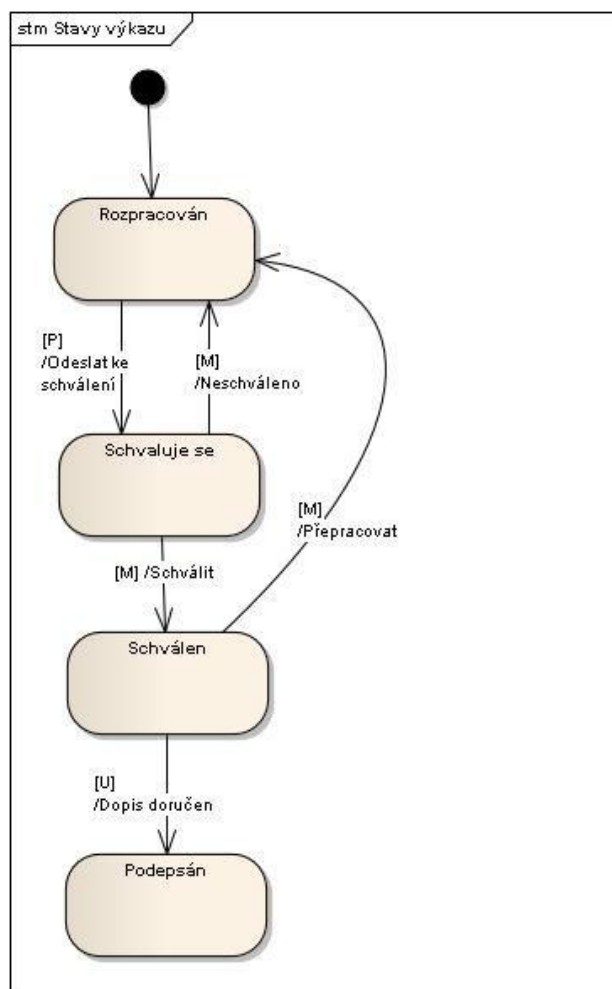
- ŠR = šéfredaktor
- A = autor
- PM = projekt manažer
- MA = medical advisor
- K = korektor
- KL = klient

[Obrázek 1]

2.4. Popis životního cyklu výkazu práce

Životní cyklus výkazu práce je popsán stavovým diagramem, viz Obrázek 2.

Přechody mezi stavy představují uživatelské akce, které uživatel může provést nad výkazem práce v daném stavu a určují přípustné scénáře pro zpracovávání výkazu. V hranatých závorkách je uvedena role, která může danou akci vykonávat.



[Obrázek 2]

2.5. Základní případy užití

Kapitola popisuje základní případy užití v Redakčním systému. Každý scénář obsahuje krátký textový popis shrnující význam případu, určení hlavního účastníka, případně vedlejší účastníky případu, hlavní scénář případu a případně alternativní scénáře.

UC1 - Vytvoření nového článku

Stručný popis: Případ popisuje vznik nového článku bez předchozí existence edičního plánu.

Primární účastníci: Šéfredaktor

Hlavní scénář:

1. Případ užití začíná, když šéfredaktor otevře formulář pro zadání nového záznam o článku
2. Šéfredaktor vyplní údaje o článku. Povinnými údaji jsou „Název“ a „Projekt“.
3. Systém uloží článek a nastaví jeho stav na hodnotu „Vytvořen“ (Viz Obrázek 1.)

UC2 - Vytvoření edičního plánu

Stručný popis: Případ popisuje vznik nového edičního plánu.

Primární účastníci: Projekt manažer

Hlavní scénář:

1. Případ užití začíná ve chvíli, kdy uživatel v modulu „Ediční plány“ otevře formulář pro vytvoření nového záznamu.
2. Uživatel vyplní údaje o edičním plánu. Povinná pole jsou „Projekt“, a „Měsíc/Rok“.
3. Systém uloží záznam o edičním plánu. Zatím pro ediční plán neexistují žádné

položky

4. Uživatel otevře přehled položek edičního plánu
5. Uživatel otevře formulář pro přidání položky k edičnímu plánu
6. Uživatel vyplní údaje o položce. Povinná pole jsou „Název“ a „Datum dokončení“.
7. Systém uloží položku k edičnímu plánu a aktualizuje výpis položek edičního plánu
8. Kroky 5-7 se opakují dokud není ediční plán hotov.

UC3 - Vytvoření nového článku z položky edičního plánu

Stručný popis: Případ popisuje vznik nového článku z existující položky edičního plánu.

Primární účastníci: Šéfredaktor

Hlavní scénář:

1. Případ užití začíná, když šéfredaktor otevře detail položky edičního plánu a provede akci „Vytvořit článek“.
2. Systém zobrazí formulář pro vznik článku a předvyplní jeho název, projekt, zdroje a požadované datum dodání podle údajů na položce edičního plánu.
3. Šéfredaktor zkontroluje, upraví a doplní údaje na článku a provede uložení.
4. Systém uloží článek a nastaví jeho stav na hodnotu „Vytvořen“ (Viz Obrázek 1.)

UC4 - Provedení workflow akce nad článkem nebo výkazem práce

Stručný popis: Případ popisuje provedení jedné akce nad článkem jako součást workflow procesu zpracování článku. Scénář probíhá identicky i pro výkaz práce.

Primární účastníci: Jakýkoli přihlášený uživatel

Hlavní scénář:

1. Případ užití začíná zobrazením přehledu všech článků, pro které má aktuálně přihlášený uživatel možnost provést akci.
2. Systém ke každému článku v přehledu vyhodnotí pomocí definice workflow procesu a rolí přiřazených uživateli, jaké akce může uživatel nad daným článkem provádět.
3. Systém zobrazí seznam akcí u každého článku
4. Uživatel zvolí požadovanou akci a případně vyplní její parametry
5. Uživatel provede akci
6. Systém dle definice workflow procesu vyhodnotí do jakého stavu přejde článek a nastaví mu tento stav.
7. Systém zobrazí informaci o provedení akce
8. Systém aktualizuje přehled zobrazených článků

Alternativní scénář: Před provedením akce může uživatel zvolit možnost zrušení provádění akce a vrátit se tak na výpis článků bez změny stavu příslušného článku.

UC5 - Zadání výkazu práce

Stručný popis:

Scénář popisuje jakým způsobem uživatel zadá nový výkaz práce do systému. Scénář začíná odesláním upozorňujícího emailu na pracovníka.

Primární účastníci: Jakýkoli přihlášený uživatel

Hlavní scénář:

1. Systém odešle N dní před nastaveným termínem pro zadání výkazů mail na pracovníka s požadavkem na zadání a odeslání ke schválení výkazu pro příslušný měsíc. Hodnotu N (počet dní) a termín pro zadání výkazu je možné v systému nastavit.
2. Pokud již výkaz existuje (systém ho pro daný měsíc mohl vytvořit automaticky např. při ukončení korektury článku, nebo ukončení kontroly,

kdy uživatel zadává do akce na nad článkem konkrétní cenu procedury, takže je nutné rovnou částku uložit do nové položky výkazu pro daný měsíc a uživatele), uživatel jej vyhledá a otevře pro editaci. Pokud výkaz neexistuje, uživatel zadá nový. Uživatel zadá potřebné informace o výkazu. (seznam polí viz kapitola Logický datový model).

3. Systém nastaví typ výkazu dle typu vyplácení pracovníka (Faktura/Dohoda, viz kapitola Logický datový model)
4. Systém dle měsíce, pro který je výkaz vytvořena automaticky dohledá počet článků, které uživatel v daném měsíci zpracoval a vygeneruje pro ně položku výkazu obsahující počet článků a druh činnosti „Vypracování článku“. Pouze články, které již prošly kontrolou šéfredaktora a klientem (jde o články ve stavu „Připraveno ke korektuře“), jsou ve výkaze zohledněny. Systém umožňuje dodatečně aktualizovat položky výkazu, pokud články splňující podmínky vzniknou později než výkaz (viz. scénář Aktualizace rozpracovaného výkazu)
5. Systém dohledá odpovědi z projektu „U lékaře“, které uživatel v daném měsíci zodpověděl a vygeneruje pro ně položku výkazu obsahující počet odpovědí a druh činnosti „Odpověď na dotaz v projektu U lékaře“
6. Volitelné: Uživatel vyplní položky Výkazy pro druhy činností, které systém nedotáhne automaticky. Seznam polí viz kapitola Logický datový model
7. Systém dopočítá částky k proplacení za články, odpovědi a ostatní druhy prací, které jsou zadané na položkách výkazu, dle hodnot v sazebníku.
8. Systém nastaví stav na „Rozpracován“ a uloží výkaz.

UC6 - Schvalování výkazu práce

Stručný popis: Scénář popisuje jak probíhá schvalování výkazu Manažerem. Scénář začíná ve chvíli kdy pracovník nad uloženým výkazem provede akci „Odeslat ke schválení“.

Primární účastníci: Manažer

Sekundární účastníci: Jakýkoli přihlášený uživatel

Hlavní scénář:

1. Uživatel provede akci „Odeslat ke kontrole“ nad vytvořeným výkazem
2. Pokud výkaz neobsahuje všechny články a odpovědi, které pracovník daný měsíc vypracoval, systém upozorní uživatele, že je třeba výkaz aktualizovat a nabídne zrušení provedení akce.
3. Systém přiřadí výkaz do příslušného měsíce pro výplatu dle aktuálního data a dle nastavení hraničního dne pro vyplácení.
4. Systém po provedení akce informuje uživatele, do kterého měsíce bude výkaz zařazen-proplacen. Pokud má být výkaz zařazen až do následujícího měsíce, nabídne systém uživateli možnost akci neprovádět a ponechat výkaz ve stavu „Rozpracován“, aby uživatel případně mohl ještě doplňovat další činnosti před vypršením hraničního dne pro následující měsíc.
5. Pokud je výkaz zařazen do aktuálního měsíce, uživatel (role Manažer) obdrží od systému email s informací o provedení akce „Odeslat ke kontrole“. Email slouží k upozornění, že je potřeba výkaz co nejdříve schválit, aby mohl být v aktuálním měsíci proplacen.
6. Systém nastaví stav na „Schvaluje se“
7. Systém nastaví pole Od-Do
8. Systém nastaví pole Počet hodin
9. Manažer rozhodne o schválení/neschválení výkaz pomocí akce „Schválit výkaz“, resp. „Neschválit výkaz“. Při spuštění akce „Neschválit výkaz“ systém vyžaduje zadání důvodu neschválení.
10. Pokud je výkaz schválen, odešle systém informační email pracovníkovi (v příloze je i vygenerovaná dohoda/faktura) a stav výkazu je nastaven na „Schváleno“. Od této chvíle není možné výkaz měnit (Výkaz lze dodatečně změnit, pokud manažer spustí akci Přepracovat).
11. Pokud výkaz není schválen, je odeslán email s upozorněním na pracovníka a stav je nastaven na „Rozpracováno“.

UC7 - Aktualizace rozpracovaného výkazu práce

Scénář popisuje jak probíhá automatická aktualizace již vytvořeného výkazu. Scénář je možné provést pouze nad výkazem ve stavu rozpracováno.

1. Uživatel zvolí akci „aktualizovat výkaz“.
2. Systém zjistí, které články a odpovědi nejsou ve výkazu zahrnuty a vygeneruje pro ně nové položky.
3. Uživatel doplní údaje na položkách výkazu, pokud je to potřeba.
4. Systém uloží výkaz

UC8 - Automatické vytvoření výkazu z pro korektora a uživatele s rolí medical advisor

Scénář popisuje automatický vznik nového výkazu při provedení akcí “Ukončit korekturu” nebo “Kontrola MA je-není OK” nad článkem.

1. Uživatel je vyzván k zadání typu sazby pro spočítání ceny za korekturu/kontrolu článku. Uživatel zadá typ sazby a potvrdí provedení akce
2. Systém zkontroluje zda pro daný měsíc a uživatele již existuje výkaz práce. Pokud neexistuje, vytvoří se nový výkaz. Systém automaticky vyplní:
 - Měsíc a rok
 - Uživatele
 - Typ výkazu (faktura/dohoda) dle uživatel
 - Vazbu na článek.
3. Systém vytvoří novou položku výkazu s typem práce dle prováděné akce (korektura, kontrola uživatelem s rolí „Medical Advisor“) a s částkou vypočítanou dle sazebníku.

2.6. Ostatní požadavky na systém

UC9 - Zobrazení a tisk výkazu

Systém musí umožňovat vygenerovat uložené výkazy jako PDF dokument, který lze použít pro tisk.

Systém musí umožňovat hromadné generování balíku všech výkazů na daný měsíc (případně vyfiltrovaných výkazů), pokud je potřeba aby je účetní vytiskla a odeslala k podepsání. Za normálních okolností toto nebude nutné, protože pracovníci by si měli tisknout dohodu sami ihned po schválení.

UC10 - Měsíční kontrola a upozornění na nedoručené dohody

Systém pravidelně jednou za měsíc zkontroluje, které výkazy nebyly doručeny mzdové účetní a odešle email s urgencí na příslušné pracovníky.

UC11 - Práva na data výkazů práce

Systém omezuje přístup k datům dle uživatelských rolí:

- Manažer – vidí všechny výkazy
- Pracovník – vidí jen vlastní výkazy . Může editovat jen své výkazy ve stavu Rozpracováno.
- Účetní – vidí všechny výkazy ve stavu Schváleno a Doručeno.

UC12 - Notifikace při akcích

Některým akcím nebo vypršeným termínům jsou přiřazeny notifikace ve formě odeslání emailu na určenou osobu.

UC13 - Notifikace o termínech a jiných událostech v systému

- 3 dny před defaultním hraničním dnem pro odeslání výkazu upozornit pracovníky, kteří pro daný měsíc ještě neodeslali výkaz ke schválení
- Měsíc po schválení odeslat upozornění pracovníkům, jejichž výkazy ještě nejsou označené v systému jako „Doručené“, tzn. Účetní neobdržela podepsaný výkaz.

- V den uzavření měsíce odeslat roli Manažer email s připomenutím, pokud ještě měsíc není uzavřen.
- Po skutečném uzavření měsíce manažerem odeslat všem pracovníkům email s informací, že od dané chvíle budou odeslané výkazy vyplaceny v následujícím měsíci.

2.7. Nefunkční požadavky

- Systém musí fungovat ve všech běžných webových prohlížečích
- Systém musí využívat stávající infrastrukturu a PHP knihovny pro tvorbu uživatelských rozhraní pro zpracovávání dat z *My-SQL* databáze.

3. Analýza a návrh řešení

3.1. Použité pojmy z teorie workflow

V této kapitole jsou vyloženy obecné pojmy a principy z oblasti workflow management systémů, tak jak jsou popsány např. v [1] a [2], a vysvětluje využití těchto pojmů a principů v Redakčním systému.

Samotný pojem **workflow** lze definovat různými způsoby. Organizace Workflow Management Coalition (www.wmfc.org), která se zabývá tvorbou standardů a vzděláváním v oblasti definování a automatizace procesů informačními technologiemi, popisuje pojem workflow jako “automatickou proceduru reprezentující byznys proces jako celek nebo jeho část, během které jsou informace a úlohy předávány ke zpracování z jednoho účastníka na druhého podle určené množiny pravidel.“

Jednoduchá, avšak výstižná definice je uvedena v [3]: „Workflow je reprezentace byznys procesu ve strojově čitelném formátu“.

Další pojmy, které budou v dalším textu používány jsou:

Proces – Množina úloh, které musí být vykonány, aby byl splněn určitý cíl

Aktivita – Jednotka práce, která představuje jeden konkrétní krok při zpracování procesu.

Workitem – Předpis pro provedení práce nad aktivitou konkrétním uživatelem.

Akce – Činnost nebo událost, kterou může vyvolat účastník procesu, aby řešil konkrétní workitem..

Workflow subjekt – Entita, nad kterou je prováděn workflow proces.

3.2. Architektura systému

Systém je tvořen třemi vrstvami: Databázovou, aplikační a vrstvou uživatelského rozhraní (UI).

Databázová vrstva je implementována jako relační databáze využívající databázový server *My-SQL*. Kompletní struktura databáze je v příloze [1].

Aplikační vrstva definuje datové entity, které slouží k reprezentaci dat z databáze, provádí nad těmito entitami byznys transakce, které naplňují výše popsané případy užití a zpřístupňuje tato data uživateli prostřednictvím vrstvy uživatelského rozhraní. Aplikační vrstva využívá technologii PHP. Tato technologie byla zvolena kvůli snadné návaznosti na vrstvu uživatelského rozhraní (viz následující odstavec).

Vrstva uživatelského rozhraní slouží k zobrazení dat uživateli ve formě webových stránek v internetovém prohlížeči. Je implementována pomocí technologie HTML, Javascript a šablonovacího systému v PHP. Jiná varianta vzhledem k požadavku zadavatele na použití interního frameworku pro tvorbu webových uživatelských rozhraní nepřicházela v úvahu.

3.3. Návrh workflow modulu

Součástí aplikační vrstvy systému je workflow modul, který slouží k provádění procesů popsaných v uživatelských požadavcích. Workflow je dokumentově orientované (document-oriented, content-oriented), což znamená, že probíhá vždy v kontextu konkrétní datové entity (např. článku nebo výkazu práce) a využívá k popisu aktuálního stavu procesu přímo stav entity, nad kterou probíhá. Alternativním přístupem je workflow orientované na aktivity, které zaznamenává provedené aktivity a stav procesu odděleně od datových entit. Výhodou dokumentově orientovaného přístupu je v našem případě jeho jednoduchost a přímočarost, protože nevyžaduje ukládání o provedených krocích mimo datové entity. Pro entity, pro něž je ukládána v redakčním systému historie, jsou informace o provedených akcích v rámci workflow procesu uloženy právě v historických záznamech.

Workflow modul nepracuje díky dokumentově orientovanému přístupu přímo s pojmem aktivita a workitem. Aktivita je v Redakčním systému představována kombinací stavu datové entity, nad kterou proces probíhá, a všech povolených akcí v tomto stavu. Workitem potom odpovídá existenci podmnožiny povolených akcí nad workflow subjektem v konkrétním stavu pro příslušného uživatele.

Hlavní úlohy, které vykonává workflow modulu v Redakčním systému, jsou:

1. Přiřazení práce uživatelům

Workflow modul musí vyhodnotit, jaké uživatelské role má uživatel a jaké akce tyto role mohou vykonávat nad datovou entitou v jejím aktuálním stavu.

2. Definice pracovního postupu (Routing)

Workflow modul musí po provedení dané akce převést entitu do příslušného stavu. V první verzi bude pro změnu stavu určující pouze typ provedené akce. V budoucnu se počítá s rozšířením, které bude vyhodnocovat i další parametry, například data specifická pro konkrétní workflow subjekt.

3.4. Logický datový model

Kapitola popisuje atributy a vztahy nejpodstatnějších datových entit.

Článek

Reprezentuje data článku, na který jsou aplikovány procesy Redakčního systému.

Atributy:

- Stav - Stav ve vztahu k workflow procesu probíhajícím nad článkem
- Název (původní) – Název článku tak jak byl nadefinován v Edičním plánu
- Název (autor) – Název článku tak jak byl zvolen autorem.
- Datum dopsání – Datum, do kterého má být článek dopsán
- Datum vydání – Datum, do kterého má být článek vydán
- Text – Text článku
- Zdroje – Html odkazy na zdroje
- Příznak pro experty-pro laiky

Vztahy (ve formátu: „Název vztahu: Entita1-Entita2 (kardinalita)“)

- Autor: Článek – Uživatel (n:1)
- Korektor: Článek – Uživatel (n:1)
- Projekt: Článek – Projekt (n:1)
- Zdroje(soubory): Článek – Soubor (n:m)

- Položka edičního plánu: Článek – Položka edičního plánu (n:1)

Ediční plán

Reprezentuje ediční plán pro daný měsíc.

Atributy:

- Téma – Textový popis tématu edičního plánu
- Rok
- Měsíc

Vztahy (ve formátu: „Název vztahu: Entita1-Entita2 (kardinalita)“)

- Autor: Článek – Uživatel (n:1)
- Projekt: Článek – Projekt (n:1)

Položka edičního plánu

Reprezentuje položku edičního plánu, tedy předpis pro vznik článku.

Atributy:

- Název článku – název budoucího článku
- Datum dopsání
- Příznak pro experty-pro laiky
- Zdroje – Html odkazy na zdroje

Vztahy (ve formátu: „Název vztahu: Entita1-Entita2 (kardinalita)“)

- Zdroje(soubory): Článek – Soubor (n:m)

Výkaz práce

Reprezentuje výkaz o provedení práce pracovníkem v daném měsíci

Atributy:

- . Číslo výkazu (xxxRRRRyyy)
 - xxx = unikátní číslo pracovníka
 - RRRR = rok dohody
 - yyy = pro každý rok číslování dohod od jedničky
- . Typ výkazu – možné hodnoty: Dohoda, Faktura
- . Měsíc a rok
- . Datum zadání
- . Datum od-do
- . Datum odeslání (ke schválení) – rozhodující pro měsíc výplaty
- . Měsíc výplaty
- . Stav
- . Datum doručení papírové dohody/faktury = Datum podpisu ?
- . Poznámky – např. důvod neschválení
- . Číslo faktury – pokud je typ Faktura

Vztahy (ve formátu: „Název vztahu: Entita1-Entita2 (kardinalita)“)

- Pracovník: Výkaz – Uživatel (n:1)
- Text výkazu(Reference do číselníků textů) (n:1)

Položka výkazu

Reprezentuje informaci o provedení konkrétní činnosti v rámci jednoho výkazu

Atributy:

- Druh činnosti
- Datum
- Počet jednotek

- Cena
- Projekt
- Textový popis práce

Vztahy:

- Výkaz: Položka – Výkaz (n:1)
- Článek: Položka – Článek (n:0-1) – Určuje ke kterému článku se položka výkazu vztahuje, není vyplněn např pro „Vedení projektu“, „Odpověď na dotaz UL“, ...

Sazebník činností

Atributy:

- Název činnosti
- Jednotka
- Vyžaduje textový popis (ano-ne)

Znamé hodnoty:

- „Psaní článku“ - cena za článek
- „Odpověď na dotaz UL“ - cena za odpověď
- „Odpověď na dotaz – ostatní weby“ - cena za odpověď
- „Korektura“ - cena za NS
- „Programování“ - cena za hodinu
- „Vedení projektu“ - cena za 1 měsíc

4. Programátorská dokumentace

Důležité třídy a metody jsou popsány v příložené generované dokumentaci. Pro snadnější orientaci ve zdrojových kódech popisuje tato kapitola uspořádání adresářů se zdrojovými kódy na příloženém CD. Soubory a adresáře, které jsou součástí obecné infrastruktury, která není součástí této práce, jsou v textu označeny kurzívou.

- */RS/include* – Obecné knihovny, které jsou součástí infrastruktury používané zadavatelem pro všechny jeho aplikace. Jde např. o zobrazování a vyhledávání v tabulkových datech, zobrazení formulářů a obecné funkce pro přístup do databáze. Zdrojové soubory v tomto adresáři nejsou součástí práce.
- */RS/templates* – Soubory HTML šablon pro uživatelské rozhraní. Šablony vyvinuté speciálně pro Redakční systém jsou v podadresáři RS:
 - ***/RS/templates/RS*** - Šablony, které byly vytvořeny speciálně pro RS jsou
- ***/RS/web*** – Zdrojové soubory Redakčního systému
 - ***/RS/web/admin/module*** – Zdrojové soubory aplikační vrstvy. Obsažené soubory s příponou „.inc“ představují jednotlivé moduly aplikace odpovídající položkám menu na nejvyšší úrovni. Např. menu „Články“ odpovídá modulu `articles.inc`, atd. Přesné mapování modulů na položky menu určuje třída `RSMenu` ze souboru `common.inc`.
 - ***/RS/web/admin/module/actions*** – Zdrojové soubory formulářů pro spouštění workflow akcí
 - ***/RS/web/admin/module/articles*** – Zdrojové soubory stránek souvisejících s modulem „Články“.
 - ***/RS/web/admin/module/entities*** – Zdrojové soubory tříd, které představují byznys entity.
 - ***/RS/web/admin/module/work_reports*** - Zdrojové soubory stránek souvisejících s modulem „Výkazy práce“.
 - ***/RS/web/admin/module/workflow*** – Zdrojové soubory workflow

modulu.

- */RS/web/css* – CSS styly.
- ***/RS/web/export*** - Zdrojové kódy hromadného exportu výkazů práce pro účetní.
- */RS/web/img* - Obrázky a jiné binární soubory použité v prezentační vrstvě.
- */RS/web/jscript* – Javascript soubory použité v prezentační vrstvě.

5. Uživatelská dokumentace a přístup do aplikace

Aplikace v testovacím režimu je nasazená na adrese <http://rs2.meditorial.cz/admin>.

V aplikaci je vytvořen testovací uživatel „test_rs“ s heslem „test_rs“

Uživatelskou příručku tvořili sami uživatelé, proto není součástí práce. Má dvě části:

- Náповěda pro Redakční systém obecně:

<http://rs2.meditorial.cz/admin/help/>

- Náповěda pro Výkazy práce:

http://rs2.meditorial.cz/admin/work_reports/help/

Doslov / Závěr

Implementaci Redakčního systému lze považovat za úspěšnou, neboť je již používána pro řízení redakce a odezva od uživatelů je pozitivní. Nasazení systému splnilo všechny na začátku vytyčené cíle a zároveň pomohlo zadavateli šetřit náklady za provoz redakce. Zpětně se ukázala jako dobrá volba dát přednost vlastnímu systému před existujícím CMS systémem, neboť je umožněna integrace systému s dalšími aplikacemi postavenými nad stejnou platformou. Jako příklad lze uvést propojení Redakčního systému se systémem pro administraci lékařské poradny nebo systémem pro evidenci požadavků na změn a hlášení chyb v systému.

Funkcionalita, která ušetřila největší množství manuálních činností pracovníků redakce, je modul pro generování výkazů práce. To je další potvrzení toho, že bylo správnou volbou vyvíjet vlastní systém, který přímo propojuje správu obsahu článků s ohodnocováním jejich autorů.

Systém bude díky úspěšnému nasazení a kladnému hodnocení uživatelů v budoucnu dále rozšiřován. V současnosti jsou známy tyto oblasti pro rozšiřování:

- Propojení modulu pro vykazování práce se systémem pro čtenářské hodnocení vydaných článků a zohlednění tohoto hodnocení ve formě finančních bonusů pro autory.
- Vytvoření uživatelského rozhraní, které umožní uživatelům upravovat pravidla pro workflow procesy.
- Umožnění uživatelsky spravované definice podmínek pro přechod do určitého stavu workflow na základě specifických dat workflow subjektu.

Seznam použité literatury

- [1] Wil van der Aalst, Kees van Hee - Workflow Management: Models, Methods, and Systems, ISBN 0-262-72046-9
- [2] Lanya Fischer - The Workflow Handbook 2004, ISBN 0-9703509-6-1
- [3] Kosek, J.: PHP - tvorba interaktivních internetových aplikací, Grada publishing, ISBN 80-7169-373-1
- [4] WFMC-TC-1011 Ver 3 Terminology and Glossary English -
[<http://www.wfmc.org/Download-document/WFMC-TC-1011-Ver-3-Terminology-and-Glossary-English.html>]
- [5] Wang, J. and A. Kumar. A Framework for Document-Driven Workflow Systems. In: Business Process Management. 2005. p. 285-301.
[<http://php.scripts.psu.edu/faculty/a/x/axk41/BPM05-jerry-reprint.pdf>]

Přílohy

[1] - Databázový model (pouze v elektronické verzi)

[2] – Logický datový model (pouze v elektronické verzi)

[3] – Generovaná programátorská dokumentace (pouze v elektronické verzi)