

PŘÍRODOVĚDECKÁ FAKULTA UNIVERZITY PALACKÉHO
KATEDRA INFORMATIKY

BAKALÁŘSKÁ PRÁCE

Repositář dat pro FCA



2011

Jiří Tempír

Anotace

Cílem práce je vytvořit repositář vstupních datových souborů pro formální konceptuální analýzu (FCA) s online webovým rozhraním. Uživateli bude nabízet data, správci repositáře potom rozhraní pro snadné nahrávání nových datových souborů a editaci metadat.

Na tomto místě bych chtěl poděkovat rodině, která mi byla oporou nejen při tvorbě této práce, ale i během celého studia. Také bych chtěl poděkovat svému vedoucímu bakalářské práce Mgr. Janovi Outratovi, Ph.D., za odborné vedení a cenné rady.

Obsah

| | |
|--|-----------|
| 1. Úvod | 7 |
| 1.1. Definice pojmů | 7 |
| 1.1.1. Dataset | 7 |
| 1.1.2. Škálování (Scaling) | 7 |
| 1.1.3. Klíčové slovo (Keyword) | 8 |
| 1.1.4. Typ klíčového slova | 8 |
| 1.1.5. Charakteristiky a Metadata | 8 |
| 2. Specifikace aplikace | 9 |
| 3. Použité technologie | 10 |
| 3.1. CSS | 10 |
| 3.2. JavaScript | 10 |
| 3.3. PHP | 11 |
| 3.4. PostgreSQL | 11 |
| 3.5. Apache | 11 |
| 4. Implementace | 12 |
| 4.1. Datový model | 12 |
| 4.2. Struktura aplikace | 12 |
| 4.2.1. Model | 14 |
| 4.2.2. View | 14 |
| 4.2.3. Controller | 15 |
| 4.3. Data v souborovém systému | 15 |
| 4.3.1. Uložení datasetů | 15 |
| 4.3.2. Stahování datasetů | 17 |
| 4.3.3. Adresářová struktura aplikace | 17 |
| 4.4. Web | 18 |
| 4.4.1. Pohled na datasety | 18 |
| 4.4.2. Bezpečnost aplikace | 18 |
| 4.5. Vzhled stránek | 18 |
| 4.5.1. Záhloví stránky | 20 |
| 4.5.2. Těla stránek | 20 |
| 4.5.3. Zápatí stránky | 21 |
| Závěr | 22 |
| Conclusions | 23 |
| Reference | 24 |

| | |
|---|-----------|
| A. Manuál | 25 |
| A.1. Úvod | 25 |
| A.2. Popis | 25 |
| A.3. Technické požadavky | 25 |
| A.4. Instalace | 25 |
| A.4.1. Příprava databáze PostgreSQL | 25 |
| A.4.2. Nastavení aplikace | 26 |
| A.4.3. Apache | 27 |
| A.4.4. Syslog | 27 |
| A.5. Odinstalace | 27 |
| A.6. Veřejná část | 28 |
| A.6.1. Datasets | 28 |
| A.6.2. Donate a dataset | 29 |
| A.6.3. Contact | 31 |
| A.7. Administrace | 31 |
| A.7.1. Přihlášení | 32 |
| A.7.2. Ovládání | 32 |
| A.7.3. Úprava datasetů | 34 |
| A.7.4. Omezení vkládaných hodnot | 36 |
| A.7.5. Úprava klíčových slov (Keywords) | 37 |
| A.7.6. Úprava uživatelů (Users) | 37 |
| A.8. Závěr | 37 |
| B. Obsah přiloženého CD | 39 |

Seznam obrázků

| | | |
|-----|---|----|
| 1. | ER-diagram | 13 |
| 2. | UML diagram hlavních tříd | 16 |
| 3. | Adresářový strom datasetu. | 17 |
| 4. | Hlavní menu | 28 |
| 5. | Seznam datasetů | 28 |
| 6. | Darovací formulář | 30 |
| 7. | Přihlašovací dialog | 32 |
| 8. | Odhlášení | 32 |
| 9. | Přehled novinek | 33 |
| 10. | Úprava novinky | 33 |
| 11. | Detail novinky | 33 |
| 12. | Seznam datasetů - Růžové datasety nejsou publikované. | 34 |
| 13. | Nepublikovaná škálování jsou zvýrazněna červeným rámováním. | 35 |
| 14. | Editace klíčového slova typu Enumeration | 38 |
| 15. | Editace uživatele - změna hesla | 38 |

Seznam tabulek

| | | |
|----|---|----|
| 1. | Třídy ve vrstvách datového modelu | 14 |
| 2. | Práva rolí, R = zobrazení, W = úprava (editace) | 31 |
| 3. | Omezení hodnot klíčových slov | 36 |
| 4. | Ostatní omezení | 37 |

1. Úvod

Hlavním důvodem pro vytvoření repositáře dat pro Formal Concept Analysis (FCA) bylo to, že žádný takový repositář dosud neexistuje. Velice známý je repositář [Machine Learning Repository](#) (MLR) na Kalifornské univerzitě v Irvine. Ten však není vhodný pro vstupní data pro FCA (tzv. formální kontext), protože vícehodnotové datasety, které jsou v MLR uloženy, musí být pro potřeby FCA převedeny prostřednictvím vhodného konceptuálního škálování. Tato škálování není možné v MLR přiložit k originálnímu datasetu. Nicméně jsem tento repositář vzal jako jakýsi standard, od kterého jsem se nechtěl moc odlišit, abych usnadnil používání tohoto repositáře uživatelům, kteří již Machine Learning Repository důvěrně znají.

Jelikož se tato práce nezabývá samotnou formální konceptuální analýzou, odkáži zájemce o tuto problematiku na skripta [1] nebo [2] Prof. RNDr. Radima Bělohávkovi, DSc.

1.1. Definice pojmů

V předchozím odstavci jsem použil několik pojmů, které je potřeba vysvětlit.

1.1.1. Dataset

Pojem dataset může být chápán ve dvou významech.

1. Dataset jako vlastní datový soubor.
2. Dataset jako celek obsahující datový soubor, charakteristiky a metadata tohoto souboru, spolu se všemi škálováními.

Aby se tyto dva pojmy nepletly, budu používat pojem dataset ve druhém významu a pro první význam budu používat fráze „datový soubor“ nebo „soubor datasetu“.

1.1.2. Škálování (Scaling)

I pojem škálování má dva významy. Jsou však víceméně stejné jako u pojmu dataset. Škálování může opět označovat datový soubor datasetu, tentokrát ale převedený, nebo chcete-li „naškálovaný“. Těchto škálování může mít dataset více. Mohou se lišit konceptuálním škálováním nebo formátem datového souboru.

Druhý význam pojmu škálování je opět celek sestávající se z datového souboru škálování a jeho metadat.

U pojmu škálování budu postupovat stejně jako u pojmu dataset.

1.1.3. Klíčové slovo (Keyword)

Klíčové slovo označuje v aplikaci jednu charakteristiku nebo metadata date-setu.

1.1.4. Typ klíčového slova

Typ klíčového slova je datový typ hodnoty, které klíčové slovo obsahuje. Jsou odvozeny od datových typů, které zná PostgreSQL.

1.1.5. Charakteristiky a Metadata

V této aplikaci se termíny Charakteristiky a metadata liší tím, že metadata se ukládají jako texty a charakteristiky jako ostatní datové typy (integer, boolean, real, atd.). Charakteristiky se v aplikaci vypisují do tabulky, kdežto metadata jako odstavce textu.

2. Specifikace aplikace

Požadavky na repositář jsou následující:

- Repositář bude webová aplikace přístupná komukoliv, ale určená vědcům zabývajících se formální konceptuální analýzou.
- Zobrazení katalogu dat s řazením podle charakteristik (počet objektů, počet atributů, oblast, atd.)
- Hromadné stažení dat datasetů vybraných v katalogu
- Formulář pro přidání datasetu a editaci jeho charakteristik a metadat správcem repositáře
- Formulář pro zaslání datasetu dárce
- Uložení původních dat, ze kterých data pro FCA vznikla
- Možnost přidání klíčových slov
- Možnost uložení více škálováních
- Darované datasety ukládat mimo publikované, před zveřejněním je musí zkontrolovat správce repositáře.
- Novinky v datasetech nebo repositáři
- Zpracování dle webových standardů
- Běh pod webovým serverem Apache 2.2, doporučení implementace serverové části na platformě PHP 5.3
- Uživatelský manuál

3. Použité technologie

V této kapitole popíší použité technologie, což jsou takové základní kameny, na kterých je repositář postaven.

3.1. CSS

Cascading Style Sheets (CSS), česky kaskádové styly, je jazyk pro popis vzhledu dokumentů napsaných značkovacími jazyky HTML, XHTML a XML. Jazyk byl navržen a standardizován organizací [W3C](#), jako reakce na počínání výrobců webových prohlížečů, kteří začaly svévolně rozšiřovat jazyk HTML o formátovací značky. V současné době je nejpoužívanější verze CSS 2.1 a pracuje se na verzi CSS3.

Cílem jazyka CSS je oddělení vzhledu dokumentu od jeho obsahu. Kaskádové styly popisují vzhled elementů (tagů) HTML dokumentu. Jak z názvu vyplývá, jednotlivé definice stylů se na sebe mohou kaskádovitě vrstvit, avšak vždy platí poslední definice. Tímto přístupem je zajištěna snadná modifikace vlastností pro velké množství elementů.

Hlavní výhody použití CSS:

- Rozsáhlé možnosti formátování
- Oddělení vzhledu od obsahu
- Snadná modifikace vzhledu
- Různé styly pro různá výstupní média (screen, print, handheld, braille, atd.)

Snad jedinou nevýhodou kaskádových stylů je špatná implementace nebo nedodržení standardů ve webových prohlížečích. To způsobuje rozdílný vzhled v různých prohlížečích a v horších případech úplný rozpad formátování.

Kompletní specifikaci CSS 2.1 najdete zde [\[6\]](#).

3.2. JavaScript

JavaScript je objektově orientovaný skriptovací jazyk primárně určený k rozšíření funkčnosti webových stránek. Je interpretován webovým prohlížečem, čili na straně klienta. Používá se ke zvýšení dynamičnosti a interaktivity webových stránek.

JavaScript byl původně vyvinut společností Netscape. Nyní ve vývoji pokračuje [Mozilla Foundation](#) a standardizaci pod názvem [ECMAScript](#) provádí [ECMA](#)(European Computer Manufacturers Association) [\[3\]](#).

3.3. PHP

Název PHP vznikl jako zkratka Personal Home Page. Nyní se uvádí jako rekurzivní zkratka PHP: Hypertext Preprocessor. Vývoj PHP odstartoval Rasmus Lerdorf v roce 1994. V současnosti řídí vývoj PHP Group a vydává jej pod PHP licenci, která je kompatibilní s GNU General Public License. Aktuální verze je 5.3.6.

PHP je skriptovací jazyk navržený pro generování dynamických webových stránek. PHP kód je součástí HTML kódu a je interpretován webovým serverem s PHP modulem, který generuje webové stránky. PHP je platformně nezávislý, i když se najde několik rozdílů v systémově závislých funkcích.

Hlavní výhody PHP:

- Specializace na dynamické webové stránky
- Velké množství knihoven a podpora mnoha protokolů
- Nativní podpora mnoha databázových systémů
- Množství hostingů nabízejících PHP
- Ohromné množství svobodných projektů a knihoven

Podrobnější popis najdete na [Wikipedii](#) nebo přímo na webu PHP [4].

3.4. PostgreSQL

PostgreSQL je objektově-relační databázový systém, který je šířen pod licenci typu MIT. Vývoj není řízen jedinou firmou, ale PostgreSQL Global Development Group, což je komunita vývojářů a firem. PostgreSQL má bohatou historii, která sahá až do roku 1977, kdy se na Kalifornské univerzitě v Berkeley začal vyvíjet předchůdce, systém Ingres.

[PostgreSQL](#) je multiplatformní, spolehlivý a bezpečný databázový systém. Plně podporuje cizí klíče, operace JOIN, pohledy, spouště a uložené procedury. Obsahuje většinu standardních datových typů. Výkonnostně je srovnatelný s komerčními systémy a častokrát je i poráží.

3.5. Apache

Apache HTTP Server je multiplatformní webový server s otevřeným kódem (licence Apache). V současné době je nejpoužívanějším webovým serverem na celém světě. Vývoj předchůdce, server NCSA HTTPd, začal v NCSA na Illinoiské univerzitě v roce 1993. Aktuální verze je 2.2.17.

[Apache](#) podporuje velké množství různých vlastností. Mnoho z nich jsou implementovány jako moduly rozšiřující základní funkcionalitu. Jedna instalace Apache může obsloužit několik virtuálních webů.

4. Implementace

4.1. Datový model

Hlavním požadavkem na repositář bylo, aby mohly být přidávána a ubírána klíčová slova. Toto značně zkomplikovalo návrh databáze. Protože jsem okamžitě zavrhl přidávání a mazání sloupce tabulky, musel jsem místo jedné tabulky pro celý dataset použít tabulek několik.

První možností, nad kterou jsem přemýšlel, je mít tabulku datasetů se základními atributy, bez kterých dataset nemůže být. Pak tabulku klíčových slov s atributem pro typ klíčového slova a vazební tabulku mezi datasety a klíčovými slovy s atributy pro každý typ klíčového slova. Jelikož v typech klíčových slov převládají typy integer a text, uvažoval jsem o zjednodušení vazební tabulky na dva atributy, s tím, že by se jiné typy než integer ukládaly jako text. Co se mi v tomto návrhu nepodařilo uspokojivě vyřešit je ukládání výčtového typu. Ten musí mít vlastní vazební tabulku, což rozbilo mou snahu o co největší jednoduchost a malý počet tabulek. To už může mít každé klíčové slovo svou vlastní tabulku. Na této myšlence je založen můj výsledný návrh.

Jak můžete vidět na obrázku číslo 1., krom výše popsanych tabulek `dataset` a `keyword` (klíčová slova), má každé klíčové slovo svou vlastní tabulku, ve které se uchovávají hodnoty konkrétních datasetů. Její název tvoří zkratka `kw`, podtržítka a identifikační číslo z tabulky klíčových slov. Například `kw_2`. Typ atributu `value` je dán typem klíčového slova, který je uložen v samostatné tabulce nazvané `kw_type`.

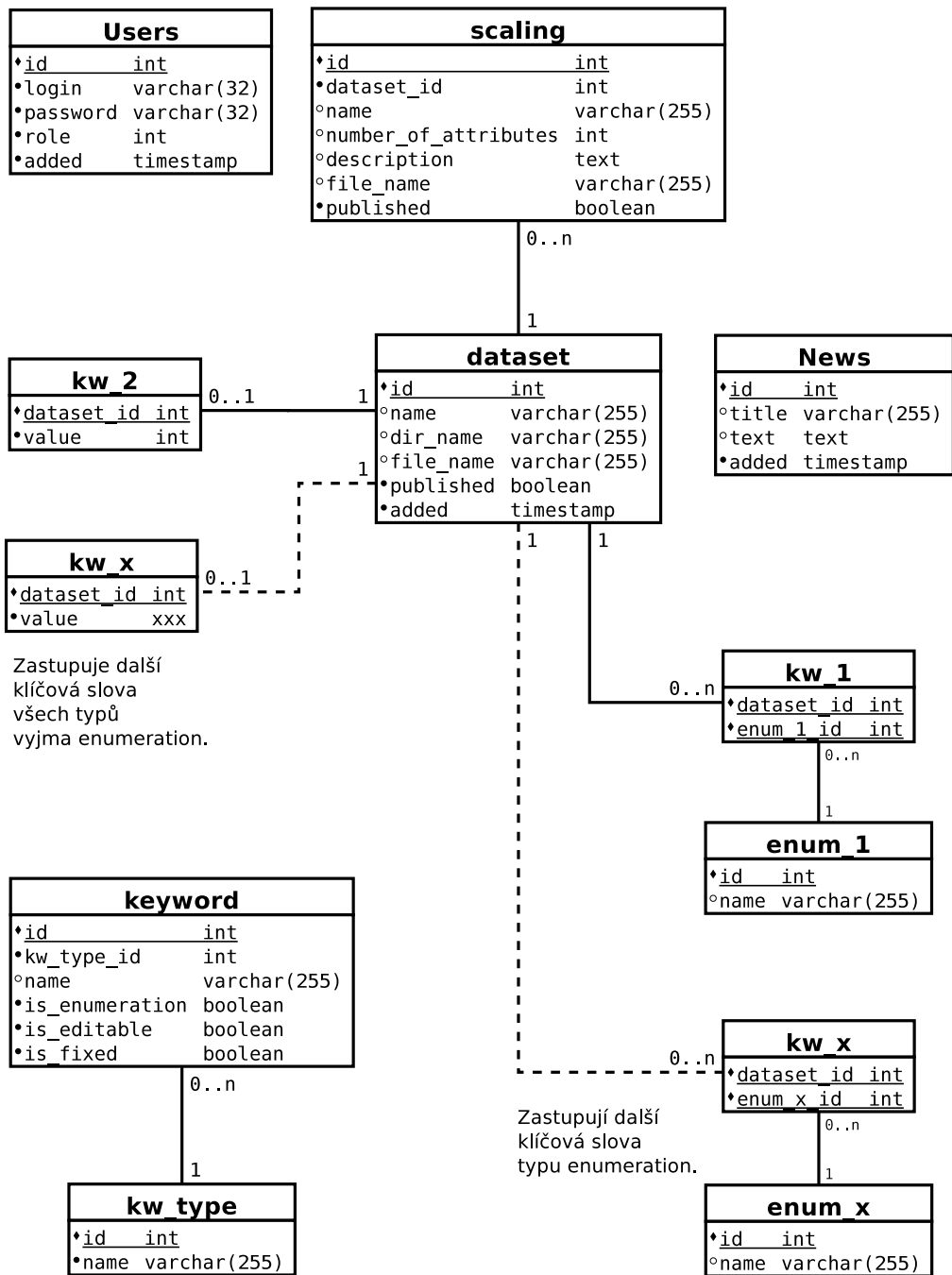
Samostatnou tabulku má škálování, které má na rozdíl od datasetu pevně danou a neměnnou strukturu.

Možnost přidávat a odebírat klíčová slova je v diagramu znázorněna entitami, které mají v názvu znak `x` místo identifikačního čísla a vazební čára je kreslena čárkovaně.

4.2. Struktura aplikace

Při návrhu a následném programování jsem se snažil o dodržení návrhového vzoru `MVC` Model-View-Controller. Návrhový vzor nebo, chcete-li softwarová architektura, Model-View-Controller dělí aplikaci na datový model, uživatelské rozhraní a řídicí logiku. Tyto tři části jsou na sobě nezávislé, čímž je umožněna změna jednotlivých částí s minimálním dopadem na části ostatní.

Další návrhový vzor, který jsem se pokusil implementovat je rozdělit datový model do pěti vrstev, jak to popsal Jan Tichý [5]. Jedná se o rozšíření datového modelu `Active record` o další vrstvy, které odstraňují jeho nevýhody. Mé rozdělení tříd do navrhovaných vrstev ukazuje tabulka číslo 1..



Obrázek 1. ER-diagram

| Vrstva | Třídy |
|------------|------------------------------|
| Service | - |
| Entita | potomci DomainObjectAbstract |
| Repository | potomci RepositoryAbstract |
| Mapper | potomci DbMapperAbstract |
| Úložiště | Db + PDO |

Tabulka 1. Třídy ve vrstvách datového modelu

4.2.1. Model

Model tedy tvoří třídy potomci tříd uvedených v tabulce 1.. Všechny jsou uloženy v adresáři `models`. Třídy mají na starosti ukládání a načítání dat z databáze a mapují je na objekty. V aplikaci jsou tyto třídy objektů ukládající své vlastnosti do databáze:

- Dataset
- Enumeration
- Keyword
- KwType
- News
- Scaling
- User

Jak jste si jistě všimli, názvy tříd se shodují s entitami datového modelu. UML diagram tříd Vám osvětlí vztahy mezi třídami.

Ke komunikaci s databází PostgreSQL jsem využil rozšiřující knihovnu [PDO](#) (PHP Data Objects), která poskytuje jednotné API pro různé databázové systémy. Tím je usnadněn přechod k jinému databázovému serveru, jako je například MySQL.

Pro navázání spojení s databází a nejčastější úlohy jsem napsal statickou třídu `Db`. Třída `SqlQuery` slouží pro konstrukci SQL dotazů.

4.2.2. View

Pohled tvoří PHP soubory v adresáři `views`. Jsou to vlastně šablony obsahující HTML kód a PHP kód, kterým se kompletuje celá HTML stránka. Tuto funkcionalitu zajišťuje třída `Template`.

Do části pohledu musím také zahrnout statické prvky jako jsou obrázky, JavaScript a CSS. Ty jsou uloženy v podadresářích `css`, `js`, `img` adresáře `public`.

4.2.3. Controller

Řízení celé aplikace je rozděleno do dvou úrovní. První úroveň tvoří třída `WebController`, která zjistí, co návštěvník požaduje. Když je požadavek relevantní, předá řízení kontroléru, který zajistí splnění požadavku. Druhou úroveň tedy tvoří kontroléry řídící provedení požadované akce a zobrazení odpovědi ve formě HTML stránky.

V adresáři `controllers` najdete potomky abstraktní třídy `ControllerAbstract`, kteří řídí zobrazení menu, stahování škálování, přihlášení, darování datasetu a další.

V adresáři `library` jsou společné, pomocné knihovnické třídy. Mezi ně patří třídy pro komunikaci s databází, manipulaci se soubory, řízení přístupu, abstraktní třídy, třídy výjimek a tak dále.

Hlavní třídy aplikace jsem nakreslil do UML diagramu tříd na obrázku číslo 2.

4.3. Data v souborovém systému

4.3.1. Uložení datasetů

Datové soubory datasetů se ukládají do souborového systému serveru, na kterém běží webový server Apache. Rozhodl jsem se ukládat zvlášť soubory publikovaných a nepublikovaných datasetů. Je to z toho důvodu, aby mohly být publikované datasety přístupné přes FTP server. Adresáře publikovaných a darovaných datasetů se nastavují v konfiguračním souboru `config.php`.

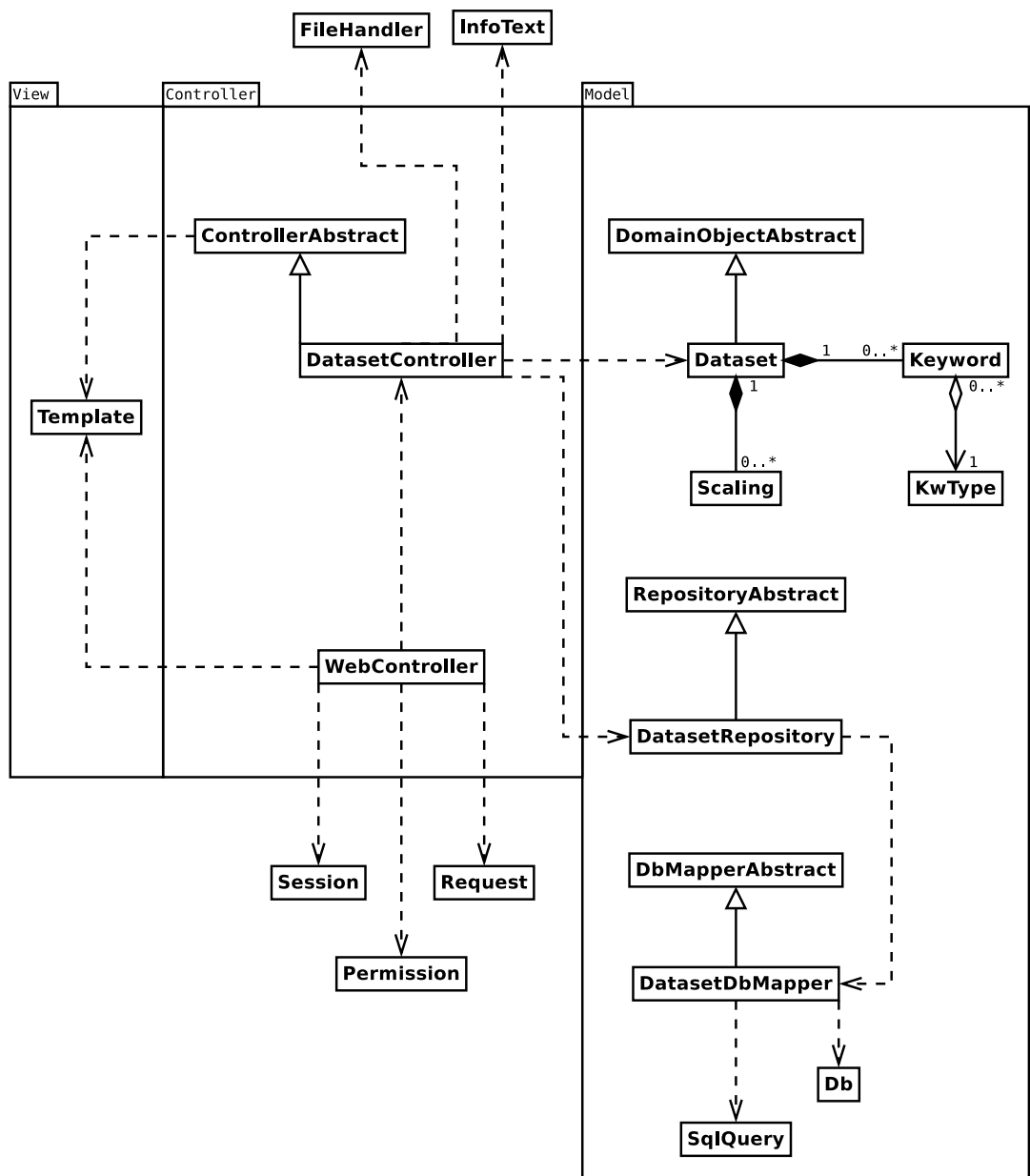
Každý dataset má samostatný adresář. V tomto adresáři je, krom datového souboru datasetu, také informační soubor datasetu a podadresáře škálování. Adresář škálování obsahuje datový soubor škálování a informační soubor tohoto škálování. Jasnější Vám to bude po pohledu na obrázek číslo 3.

Informační soubor obsahuje stejné informace, které vidí návštěvník stránek na stránce datasetu. Není to HTML soubor, ale čistě textový soubor. Informační soubor škálování obsahuje, krom informací o datasetu, informace o konkrétním škálování. Informace o ostatních škálováních v něm nejdou přítomny.

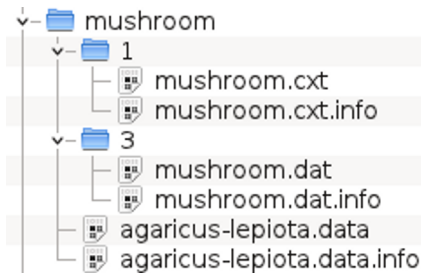
Při publikování se celý adresář datasetu přesune z adresáře pro darované datasety do adresáře pro publikované datasety. Opačný postup se realizuje při stažení („odpublikování“) datasetu.

4.3.2. Stahování datasetů

Jelikož stahovaný dataset tvoří minimálně dva soubory, což jsou datový soubor datasetu a informační soubor datasetu, musí být před odesláním uživateli



Obrázek 2. UML diagram hlavních tříd



Obrázek 3. Adresářový strom datasetu.

zabaleny do jediného souboru. Využívám k tomu souborový formát [zip](#). Tento postup má výhodu v tom, že oba soubory jsou zkomprimovány a tudíž se přenáší menší množství dat. Pro lepší manipulaci s datasetem jsou oba dva soubory zabaleny i s adresářem datasetu.

Stejně tak i škálování je posláno uživateli jako zip archiv adresáře škálování s datovým souborem škálování a informačním souborem škálování. Tentokrát však není adresář škálování umístěn v adresáři datasetu, ale je samostatně. Při stahování všech škálování datasetu jsou tyto umístěny v adresáři datasetu. Když uživatel požaduje stáhnout celý dataset, je uživateli poslána celá adresářová struktura, tak jak jsem ji popsal [zde](#), vyjma nepublikovaných škálování.

4.3.3. Adresářová struktura aplikace

Obsah většiny adresářů jsem již popsal v předešlém textu. Teď nadešel čas představit celou adresářovou strukturu. Názvy adresářů jsou všeříkající.

```

dataset/repository
    /donated

web/app/controllers
    /models
    /views
/config
/library
/public/css
    /img
    /js
  
```

4.4. Web

Vkládání informací do aplikace se provádí prostřednictvím webových formulářů, které tvoří grafické uživatelské rozhraní aplikace. Není doporučen jiný způsob manipulace s daty, než pomocí aplikace.

4.4.1. Pohled na datasety

Pro snížení zátěže databázového serveru PostgreSQL se seznam datasetů načítá z pohledu `whole_dataset`. I všechna řazení, krom jednoho, se provádí nad tímto pohledem. V tomto pohledu se výčtové položky a počty naškálovaných atributů škálování datasetu ukládají do jednoho sloupce jako text oddělený čárkami. Celý SQL příkaz, kterým se pohled vytváří najde [zde](#).

4.4.2. Bezpečnost aplikace

Kontrola zadaných dat je prováděna na dvou místech. Nejprve JavaScriptem ihned po zapsání do políčka formuláře na straně uživatele. Následně je kontrola provedena na straně serveru při prvním použití.

JavaScript kontroluje vložená data pomocí regulárních výrazů. Na serveru jsem využil vestavěné funkce PHP. Například `ctype_alnum`, `is_numeric` nebo `intval`. Pro kontrolu data, URL a názvů souborů jsem použil regulární výrazy i na straně serveru.

Na obou místech také omezují velikost vkládaných dat. U uživatele to je nastavením atributu `maxlength` formulářového prvku `<INPUT>`. Na serveru využívám funkci `mb_substr`. Protože jsou znaky kódovány kódováním UTF-8, musím použít `mutlibyte` variantu.

V případě, že vstupní data projdou kontrolou, jsou uložena do databáze. Speciální znaky jsou ošetřeny technikou vázání proměnných dostupnou v PDO.

Protože uložený text může obsahovat nechtěné HTML značky, nahrazuji nebezpečné znaky funkcí `htmlspecialchars` při zobrazení HTML stránky. Ve stejný čas také nahrazuji DokuWiki [syntaxi externího linku](#) (s dvojitými hranatými závorkami) HTML značkou ` `.

Omezení přístupu do adresářů s knihovnami aplikace je realizována zákazem v konfiguračním souboru Apache `.htaccess`, který platí pro aktuální adresář a jeho podadresáře. Uživatelé se do zakázaných adresářů nedostanou, kdežto webový server ano.

Aplikace, ve výchozím nastavení, používá pro přihlášení protokol HTTPS. Uživatel, který nemá webový server patřičně nastaven, si jej může vypnout v konfiguračním souboru.

4.5. Vzhled stránek

Snažil jsem se navrhnout jednoduchý líbivý vzhled, který nebude zbytečně

```

CREATE OR REPLACE VIEW whole_dataset (
    id,
    name,
    file_name,
    published,
    kw_attribute_types,
    kw_area,
    kw_number_of_instances,
    kw_number_of_attributes,
    kw_number_of_scaled_attributes
)
AS
SELECT
    dataset.id,
    dataset.name,
    dataset.file_name,
    dataset.published,
    (SELECT array_to_string(array(
        SELECT CAST(enum_1.name AS text)
        FROM enum_1, kw_1 as kw
        WHERE kw.enum_1_id = enum_1.id
            AND kw.dataset_id = dataset.id
    ), ', '))
),
    kw_5.value,
    kw_2.value,
    kw_3.value,
    (SELECT array_to_string(array(
        SELECT CAST(scaling.number_of_attributes AS text)
        FROM scaling
        WHERE scaling.dataset_id = dataset.id
            AND scaling.published = true
    ), ', '))
FROM
    (((dataset
    LEFT OUTER JOIN kw_5 ON dataset.id = kw_5.dataset_id )
    LEFT OUTER JOIN kw_2 ON dataset.id = kw_2.dataset_id )
    LEFT OUTER JOIN kw_3 ON dataset.id = kw_3.dataset_id )
;

```

poutat pozornost při práci. Za základní barvu jsem zvolil modrou a použil převážně její světlejší odstíny, které doplňuje šedá barva. Pro zvýraznění chybových hlášení jsem vybral červenou a odkazů žluto-hnědou barvu. Základní rozdělení stránky repositáře je provedeno šablonou `Page.php`. V šabloně je stránka rozdělena na:

Hlavičku obsahující pouze hlavní nadpis.

Menu sloužící k navigaci v repositáři.

Oblast uživatele kde se zobrazuje přihlášený uživatel.

Oblast zpráv kde se zobrazují varování a chybové zprávy.

Obsah kam se vkládá obsah ostatních stránek.

Patičku obsahující copyright.

Vzhled stránek se řídí kaskádovými styly CSS verze 2.1 uloženými v souboru `style.css`.

Pevná šířka stránky se mi pro aplikaci nezdála být vhodná. Nastavil jsem pouze minimální šířku a to takovou, která dovoluje zobrazit celé hlavní menu. Maximální šířku stránky si tedy může každý řídit šířkou okna prohlížeče, jak mu to vyhovuje.

Obrázky některých stránek nebo jejich částí najdete v příloze [A](#). Z nich si uděláte lepší představu o vzhledu aplikace.

4.5.1. Záhloví stránky

V hlavičce, blíže k levé straně je umístěn velký název aplikace. Je to také odkaz na úvodní stránku aplikace. Viditelnou, ale ne skutečnou, součástí hlavičky je hlavní menu. Aktivní položka menu má bílé pozadí, čímž uživateli oznamuje, v které části aplikace se nachází. Pod hlavním menu, u pravého okraje, je odkaz pro přihlášení do aplikace. Po přihlášení se na tomto místě zobrazí jméno přihlášeného uživatele a odkaz pro odhlášení.

Dále následuje část stránky, kde se zobrazují chybová hlášení. Text hlášení je zobrazen centrovane, tučným písmem a jak je uvedeno výše, tmavě červenou barvou. V případě, že není třeba nic oznamovat, není tato sekce ve stránce přítomna.

4.5.2. Těla stránek

Obsah stránky tvoří většinou tabulka se seznamem všech datasetů, klíčových slov, uživatelů a ostatních entit aplikace. Přes odkaz tvořený názvem položky se uživatel dostane na její detail. Uživateli s rolí Editor nebo Administrátor přibudou v tabulce dva sloupce s odkazy pro editaci a smazání položky.

Při zobrazení detailu se v levé dolní části nachází odkaz pro návrat na seznam, ze kterého si uživatel detail zobrazil. Součástí detailu je také „editační menu“. Jsou to textové odkazy TOP, EDIT a DELETE oddělené znakem |. Je-li přihlášen uživatel s rolí editora. V opačném případě se uživateli zobrazí pouze odkaz TOP.

4.5.3. Zápatí stránky

Zápatí je od těla odděleno tenkou šedou čarou. V současné době obsahuje pouze copyright. Nenapadlo mne, co jiného do zápatí dát. Při implementaci se obsah jistě změní.

Závěr

Cílem práce je vytvořit repositář vstupních datových souborů pro formální konceptuální analýzu (FCA) s online webovým rozhraním. To se podle mého názoru podařilo. Aplikace splňuje všechny funkční a nefunkční požadavky.

Při vývoji této aplikace jsem získal spoustu zkušeností a rozšířil své vědomosti o všech použitých technologiích, architektuře webových aplikací a modelech ukládání objektů v relačních databázích. Tyto vědomosti zajisté v budoucnu využiji.

Jako pokračuje vývoj většiny aplikací, mohl by a může pokračovat vývoj i této. V případě větší návštěvnosti by stálo za to realizovat anonymní diskuzi k datasetům. Velký počet datasetů může vyvolat potřebu implementovat vyhledávání. Teprve praxe ukáže, zda je aplikace kvalitní a jaká rozšíření nebo úpravy budou třeba.

Conclusions

The aim of this bachelor's diploma thesis is create data repository for Formal Concept Analysis (FCA) with on-line web interface. In my humble opinion, it succeeded. The application implement all functional and non-functional requirements.

During development I gain a lot of experience and extend my knowledge about all used technologies, web application architecture and data source architectural patterns. I will certainly use this knowledge in the future.

As is continued development of most of application It could and can continue development of this one. In the case of bigger attendance users would appreciate the implementation of anonymous discussion. A large number of data sets may necessitate implementation of the searching. Only practice will show whether the application is good and what extensions or modifications are needed.

Reference

- [1] Bělohávek, Radim. *Konceptuální svazy a formální konceptuální analýza*. Univerzita Palackého, Olomouc, 2004.
- [2] Bělohávek, Radim. *Introduction to Formal Concept Analysis*. Univerzita Palackého, Olomouc, 2008.
- [3] Ecma International Standard ECMA-262. *ECMAScript Language Specification Edition 5.1*. Elektronická publikace, 2011.
- [4] PHP: Hypertext Preprocessor. <http://www.php.net/>.
- [5] Tichý, Jan. *Pět vrstev modelu*. Elektronická publikace, 2010.
- [6] W3C. *Cascading Style Sheets Level 2 Revision 1 (CSS 2.1) Specification*. Elektronická publikace, 2011.

A. Manuál

A.1. Úvod

Jsem rád, že hodláte používat mnou napsanou aplikaci Repodat. Aplikace vznikla jako bakalářská práce při studiu Aplikované informatiky na katedře informatiky, Přírodovědecké fakulty Univerzity Palackého v Olomouci.

A.2. Popis

Aplikace slouží jako veřejný sklad datasetů a jejich škálování, s možností jejich stažení a formulářem pro darování datasetu. Administrační rozhraní umožňuje úpravu metadat o datasetu a jeho škálováních nebo nahrání datasetů a škálování do aplikace. Aplikace rovněž obsahuje stránku s novinkami, což jsou krátké informační zprávičky o novinkách. Dataset je vstupní datový soubor pro FCA (Formal Concept Analysis) a škálování (binarizování) je jeho převedení do binární podoby (objekt buď atribut má nebo nemá). Těchto škálování může mít dataset více. Záleží na výběru způsobu škálování a datovém formátu. V této aplikaci pojem dataset také označuje souhrn všech informací (charakteristik a metadat) o datovém souboru spolu s ním a jeho škálováními.

A.3. Technické požadavky

Aplikace pro svůj běh vyžaduje minimálně databázový server PostgreSQL 8.2, webový server Apache 2.2 s PHP 5.2 a s modulem pgsq. Operační systém není určen. Je předpokládán GNU Linux. Doporučuji použít aktuální verze výše uvedených serverů i s opravami. Samotná aplikace nezabere téměř žádné místo. Avšak na disku si musíte vyhradit místo pro ukládané datasety a jejich škálování. Metadata a Charakteristiky se ukládají do databáze.

A.4. Instalace

Instalace se skládá z konfigurace serverů, které aplikace používá, vytvoření tabulek v databázi a nakopírování php skriptů do adresáře webového serveru. Nebudu zde popisovat instalaci a nastavení jednotlivých serverů, ale pouze věci týkající se této aplikace. Popisuji instalaci na stroji s Linuxem (Debian Squeeze).

A.4.1. Příprava databáze PostgreSQL

V adresáři `scripts` jsou SQL soubory pro instalaci databázové části aplikace. V souboru `createDB.sql` jsou příkazy, pomocí kterých lze vytvořit novou databázi a uživatele. Tyto příkazy musí provádět uživatel s patřičnými právy.

Jestliže už máte databázi připravenou, není potřeba tyto příkazy použít.

- vytvoří nového uživatele:
CREATE USER jouza WITH PASSWORD 'peslw-';
- vytvoří novou databázi:
CREATE DATABASE repodat OWNER jouza;
- přidělí práva na databázi uživateli:
GRANT ALL PRIVILEGES ON DATABASE repodat TO jouza;

Také je potřeba pamatovat na to, aby se mohl vytvořený uživatel připojit k databázovému serveru. Toto se nastavuje v `/etc/postgresql/8.4/main/pg_hba.conf`. V souboru `createTables.sql` jsou příkazy vytvářející jednotlivé tabulky. Provedením SQL příkazů ze souboru `sampleData.sql` vytvoříte dva ukázkové datasety. Toto už může udělat dříve vytvořený uživatel. Na prvním řádku souboru je uveden tento příklad použití SQL souborů.

```
psql --file createTables.sql --log-file createTables.log \  
--output createTable.out -U jouza repodat > createTables.err 2>&1
```

A.4.2. Nastavení aplikace

Soubor `config.php` v adresáři `config` obsahuje konfiguraci aplikace. V tomto souboru musíte nastavit údaje pro připojení k databázi a cesty k adresářům, kam se budou ukládat datasety s jejich škálováními. Jedná se o proměnné:

`dbHost` - jméno stroje s databází

`dbPort` - port, na kterém databáze poslouchá

`dbName` - jméno databáze

`dbUser` - uživatel databáze, který má právo vytvářet tabulky

`dbPassword` - uživatelské heslo

`DIR_TMP` - adresář pro dočasné soubory (Při PHP "save_mode:On" není /tmp přístupný.)

`DIR_REPOSITORY` - adresář pro datasety

`DIR_DONATED` - adresář pro darované (nezveřejněné) datasety

Do všech těchto adresářů musí mít právo zápisu uživatel, pod kterým běží Apache. Také by bylo dobré nastavit těmto adresářům taková práva, aby ostatní uživatelé serveru nemohly s uloženými daty a škálováními manipulovat.

Dále se v konfiguračním souboru nastavuje, zda bude přihlášení do aplikace probíhat zabezpečenou komunikací, protokolem HTTPS nebo nezabezpečeným HTTP. Při výchozím nastavení bude použit zabezpečený protokol ('HTTPS', 'YES').

A.4.3. Apache

Adresář `repodat` umístít do adresáře dostupného z webovému serveru Apache. Můžete speciálně pro tuto aplikaci nastavit virtuálního hosta v konfiguraci Apache. Pokud chcete využívat protokol HTTPS, musíte odpovídajícím způsobem nastavit Apache. Jinak změňte nastavení v konfiguračním souboru.

A.4.4. Syslog

V adresáři `scripts/rsyslog.d` je soubor `repodat.conf`, který je třeba umístít do `/etc/rsyslog.d`. Slouží pro filtrování logování aplikace do souboru `/var/log/repodat.log`. Jestliže to neuděláte, budou se chybová hlášení aplikace ukládat do souboru `/var/log/messeges`.

A.5. Odinstalace

POZOR !!! Odinstalováním smažete všechny daty a jejich škálování. Odinstalace je vlastně zrušení toho, co jste vytvořili při instalaci. Čili:

1. smazání databáze
`DROP DATABASE IF EXISTS repodat ;`
2. smazání databázového uživatele
`DROP USER IF EXISTS jouza ;`
3. Je-li třeba, zrušení nastavení pro aplikaci v konfiguraci Apache
4. smazání aplikace z adresáře Apache
5. smazání adresářů `DIR_TMP`, `DIR_REPOSITORY` a `DIR_DONATED` z disku
6. smazání souboru `/etc/rsyslog.d/repodat.conf`

A.6. Veřejná část

Návštěvník webu se většinou dostane na úvodní stránku, která je tvořena přivítáním a posledními novinkami. Pro následnou navigaci po aplikaci je určeno hlavní menu, které je v horní části okna prohlížeče (Obrázek číslo 4.).



Obrázek 4. Hlavní menu

Odkazy mají v aplikaci světle modrou barvu, která se po najetí myši změní na světle hnědou. Chybová hlášení se zobrazují pod hlavním menu.

A.6.1. Datasets

Stránka Datasets obsahuje přehled publikovaných datasetů ve formě tabulky. Můžete ji vidět na obrázku číslo 5..

Dataset list

| | Name ↕ | Attribute types ↕ | # Instances ↕ | # Original attributes ↕ | Area ↕ | # Scaled attributes ↕ | | |
|-------------------------------------|---|----------------------------|---------------|-------------------------|----------|-----------------------|-----|----|
| <input type="checkbox"/> | Arrhythmia | Categorical, Integer, Real | 452 | 279 | Life | 542 | 873 | |
| <input type="checkbox"/> | Contraceptive Method Choice | Categorical, Integer | 1473 | 9 | Life | 51 | | |
| <input type="checkbox"/> | Image Segmentation | Real | 2310 | 19 | Image | 95 | 77 | |
| <input checked="" type="checkbox"/> | Mushroom | Categorical | 8124 | 22 | Nature | 120 | 120 | |
| <input type="checkbox"/> | Spambase | Integer, Real | 4601 | 57 | Computer | 228 | 351 | |
| <input checked="" type="checkbox"/> | Sponge | Categorical, Integer | 76 | 45 | Nature | 212 | | |
| <input type="checkbox"/> | Tic-Tac-Toe Endgame | Categorical | 958 | 9 | Game | 36 | 47 | 39 |
| <input type="checkbox"/> | Water Treatment Plant | Integer, Real | 527 | 38 | Physical | 190 | 167 | |

Download Origin

Download Scaling

TOP

Obrázek 5. Seznam datasetů

Datasety v tabulce můžete řadit kliknutím na název sloupce nebo na šipku. Prvním kliknutím se příslušný sloupec seřadí vzestupně, druhým sestupně a třetím se řazení zruší. U šipek prvním kliknutím třídění zapnete a druhým jej vypnete. Šipka směřující nahoru značí vzestupné řazení a šipka dolu sestupné. Aktivní řazení je značeno černou barvou šipky. Výchozí řazení je dle času přidání datasetu. Čili novinky najdete nahoře.

U každého datasetu najdete v prvním sloupci zaškrtačací políčko (checkbox). Řádky se zatrženými datasety se zvýrazní šedým pozadím. Takto vybrané datasety můžete hromadně stáhnout. K tomu slouží tlačítka v pravém dolním rohu. Tlačítkem **Download Origin** stáhnete původní datové soubory všech vybraných datasetů. Tlačítkem **Download Scaling** stáhnete všechna škálování všech vybraných datasetů. Jak napovídá název, tlačítkem **Download All** stáhnete všechny datové soubory všech vybraných datasetů (i všech jejich škálování).

Kliknutím na název datasetu přejdete na stránku s podrobnými informacemi (metadaty) o tomto datasetu. Najdete zde také tlačítka pro stažení původního datasetu, jednotlivých škálování, všech škálování i všech souborů datasetu. Ať vyberete kteroukoliv z výše uvedených možností, vždy si stáhnete zkomprimovaný (zip) adresář , který krom datového souboru, obsahuje i textový soubor s informacemi o datasetu. I každé škálování má svůj informační soubor, jehož součástí jsou informace o původním datasetu.

Do počtu stažení se počítá kliknutí na jakékoliv tlačítko stáhnout na stránce datasetu.

A.6.2. Donate a dataset

Stránka **Donate a dataset** je určená pro darování datasetů. Kdokoliv může formulář vyplnit a darovat dataset.

Hvězdičkou označené položky jsou povinné a je nutné je vyplnit.

Do políček není dovoleno vkládat HTML značky. Do polí pro popis (textarea) je možné vložit odkaz DokuWiki syntaxí. Např. `[[http://www.inf.upol.cz|katedra informatiky]]` . Do dvou hranatých závorek napište URI a za svislítko (rouru) text, který bude zobrazen jako odkaz. Když svislítko a text vynecháte, bude zobrazen přímo URI.

Popis darovacího formuláře

Darovací formulář je zobrazen na obrázku číslo 6. Uvádím popis jednotlivých položek formuláře.

Name - jméno datasetu

Attribute type - datový typ atributů

Number of instances - počet záznamů v datasetu

Number of original attributes - počet atributů (položek) jednoho záznamu

Missing values - chybí nějaké položky?

Dataset description - popis datasetu

Attribute description - popis atributů (položek) datasetu

Donate a Dataset

Name: *

Attribute type: *

Categorical Integer Real Boolean

Number of instances: *

Number of original attributes: *

Missing values: *

Yes

No

Dataset description:

Attribute description:

Source:

Relevant papers:

Data file: *

 Procházet...

Scaling

Name of scaling:

Number of scaled attributes:

Scaling description:

Scaling data file:

 Procházet...

Donate

[TOP](#)

Obrázek 6. Darovací formulář

Source - kontakt a další informace o dárci

Relevant papers - odkazy na dokumenty, které už dříve darovaný dataset citovaly

Data file - soubor s daty

Name of scaling jméno škálování

Number of scaled attributes - počet naškálovaných atributů

Scaling description - popis škálování a atributů škálování

Scaling data file - soubor s naškálovanými daty

A.6.3. Contact

Jak název napovídá, na stránce **Contact** je kontakt na správce aplikace. Na něj je možné zaslát své připomínky, náměty nebo dotazy.

A.7. Administrace

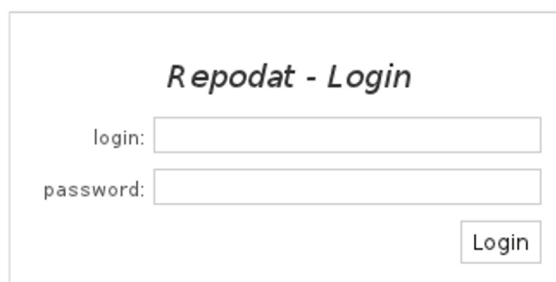
Administrace slouží hlavně k úpravě darovaných datasetů a jejich následné publikaci. Můžete také vkládat a upravovat novinky. Administrátor má navíc práva k editaci uživatelů a klíčových slov. Aby jste mohli upravovat datasety a novinky musíte mít v aplikaci uživatelský účet a roli editora. Uživatele zakládá a roli jim přiděluje administrátor. Po nainstalování aplikace jsou v ní nachystány dva uživatelské účty s rolemi: admin = administrátor a edie = editor. Tito uživatelé mají hesla stejná jako jejich jména. **Z bezpečnostních důvodů jim musíte změnit hesla nebo uživatele vymazat.** V tabulce číslo 2. najdete práva rolí na jednotlivé oblasti.

| Oblast | Editor | Administrátor |
|----------|--------|---------------|
| News | W | W |
| Dataset | W | W |
| Keywords | R | W |
| Users | - | W |

Tabulka 2. Práva rolí, R = zobrazení, W = úprava (editace)

A.7.1. Přihlášení

V případě, že máte svůj účet, můžete se do aplikace přihlásit. Kliknutím na odkaz **Administration** se Vám zobrazí přihlašovací obrazovka, kam vyplníte své přihlašovací údaje a potvrdíte enterem nebo kliknutím na tlačítko **Login**. Přihlašovací dialog je na obrázku číslo 7..



The image shows a login dialog box with the title "Repodat - Login". It contains two input fields: "login:" and "password:". Below the "password:" field is a "Login" button.

Obrázek 7. Přihlašovací dialog



Obrázek 8. Odhlášení

Po přihlášení Vás aplikace vrátí na stránku, na které jste byli před kliknutím na odkaz **Administration**. Že jste přihlášení a na jaký účet, je signalizováno změnou odkazu **Administration** za Vaše uživatelské jméno a odkaz **logout** (viz. obrázek číslo 8.). Tento odkaz použijete pro odhlášení z administrace.

A.7.2. Ovládání

Máte-li roli editora nebo administrátora, změní se stránka s novinkami na tabulku se sloupci **EDIT** a **DELETE** (viz. 9. obrázek). Jejich název plně odpovídá tomu, co se stane, když na odkaz kliknete.

Odkaz pro přidání nové zprávičky je vpravo pod tabulkou.

Kliknutím na název novinky si zobrazíte její detail.

V editaci provedené změny uložíte kliknutím na tlačítko **Submit**. Jestliže nechcete změny uložit, můžete se vrátit zpět na seznam novinek kliknutím na odkaz **Back to list**. Editační formulář vidíte na obrázku 10.

Na stránce s detailem najdete odkazy pro úpravu a smazání novinky. Odkazy jsou umístěné uprostřed nad patičkou stránky. Odkaz **View list** pro návrat na seznam novinek najdete vlevo pod textem novinky. Opět příkládám obrázek. Tentokráté má číslo 11..

News list

| Added | Title | | |
|----------|--------------------------------------|------|--------|
| 07/17/11 | <i>Aplikace připravena na vydání</i> | EDIT | DELETE |
| 06/05/11 | <i>Přidán první dataset</i> | EDIT | DELETE |
| 11/21/10 | <i>Aplikace poprvé spuštěna</i> | EDIT | DELETE |

[Add news](#)

[TOP](#)

Obrázek 9. Přehled novinek

Edit news: "Aplikace připravena na vydání"

Title:

Text:

Myslím, že je aplikace připravena. Samozřejmě se najdou věci, které by šly ještě vylepšit, ale to bych stále vylepšoval a nikdy nebyl hotov.

[Back to list](#)

Obrázek 10. Úprava novinky

Aplikace připravena na vydání

07/17/11

Myslím, že je aplikace připravena. Samozřejmě se najdou věci, které by šly ještě vylepšit, ale to bych stále vylepšoval a nikdy nebyl hotov.

[View list](#)

[TOP](#) | [EDIT](#) | [DELETE](#)

Obrázek 11. Detail novinky

Odkazy, popsané u detailu, editace a seznamu novinek, najdete také u datasetů, klíčových slov a uživatelů.

A.7.3. Úprava datasetů

Jste-li přihlášený a máte roli editor nebo administrátor, uvidíte v seznamu datasetů nezveřejněné (nepublikované) datasety zvýrazněné růžovým podbarvením. Mezi nezveřejněné datasety patří darované a přidané datasety. Pokud se editor podívá na detail kteréhokoliv datasetu, uvidí jej i se všemi jeho škálováními, tedy i s těmi nepublikovanými.

Editační formulář je velice podobný tomu darovacímu, proto jej zde nebudu popisovat. Můžete ho vidět na 12. obrázku. Darovací formulář je pevně daný. Do editačního budou přibývat políčka pro nově přidaná klíčová slova.

Dataset list

| <input type="checkbox"/> | Name | Attribute types | # Instances | # Original attributes | Area | # Scaled attributes | | | | |
|-------------------------------------|-----------------------------|----------------------------|-------------|-----------------------|----------|---------------------|-----|------|--------|--------|
| <input type="checkbox"/> | Wine Quality | Real | 4898 | 12 | | | | EDIT | DELETE | |
| <input type="checkbox"/> | Arrhythmia | Categorical, Integer, Real | 452 | 279 | Life | 542 | 873 | EDIT | DELETE | |
| <input checked="" type="checkbox"/> | Tic-Tac-Toe Endgame | Categorical | 958 | 9 | Game | 36 | 47 | 39 | EDIT | DELETE |
| <input checked="" type="checkbox"/> | Spambase | Integer, Real | 4601 | 57 | Computer | 228 | 351 | EDIT | DELETE | |
| <input type="checkbox"/> | Image Segmentation | Real | 2310 | 19 | Image | 95 | 77 | EDIT | DELETE | |
| <input type="checkbox"/> | Artificial Characters | Categorical, Integer, Real | 6000 | 7 | Computer | | | 37 | EDIT | DELETE |
| <input type="checkbox"/> | Contraceptive Method Choice | Categorical, Integer | 1473 | 9 | Life | | | 51 | EDIT | DELETE |
| <input type="checkbox"/> | Water Treatment Plant | Integer, Real | 527 | 38 | Physical | 190 | 167 | EDIT | DELETE | |
| <input type="checkbox"/> | Sponge | Categorical, Integer | 76 | 45 | Nature | | | 212 | EDIT | DELETE |
| <input type="checkbox"/> | YearPredictionMSD | Real | 515345 | 90 | Music | | | | EDIT | DELETE |
| <input type="checkbox"/> | Mushroom | Categorical | 8124 | 22 | Nature | 120 | 120 | EDIT | DELETE | |

Download Origin

Download All Scalings

Download All

[Add dataset](#)

[TOP](#)

Obrázek 12. Seznam datasetů - Růžové datasety nejsou publikované.

Upravený a uložený dataset je třeba následně zveřejnit (publikovat). Slouží k tomu tlačítko **Publish**. Jistě nemusím připomínat, že by si měl editor dataset před zveřejněním zrevidovat a opravit případné chyby. Při publikaci se dataset neukládá, jen se kontroluje existence datových souborů. Když neexistuje soubor datasetu, nebude možné dataset publikovat. V případě, že neexistuje soubor škálování, nebude publikováno pouze příslušné škálování.

Nepublikovaná škálování jsou v editaci datasetu zvýrazněna červeným rámováním. (Obrázek 13.) Škálování, kterému chybí datový soubor, nemá tlačítko **Download Scaling** pro stažení škálování. Pod posledním škálováním je tlačítko **Add Scaling** pro přidání dalšího škálování.

Name of scaling: 2nd Binarized Spambase

Number of Scaled attributes: 351

Scaling description:
This scaling isn't real. It is only for presentation of application.

Upload Scaling Data File: Procházet...

Download Scaling

Delete Scaling

Name of scaling: 3rd Binarized Spambase

Number of Scaled attributes: 448

Scaling description:
This scaling isn't real. It is only for presentation of application.

Upload Scaling Data File: Procházet...

Delete Scaling

Obrázek 13. Nepublikovaná škálování jsou zvýrazněna červeným rámováním.

Po jakékoli další editaci publikovaného datasetu se tento stáhne („odpublikuje“) a je nutné ho znova publikovat. Stáhnutí můžete provést i sami. Opět příslušným tlačítkem (**Unpublish**). Při uložení datasetu se generují informační soubory datasetu a všech jeho škálování.

Darovaný nebo přidáný soubor datasetu se uloží do adresáře s názvem podle id datasetu. Tento adresář bude umístěn spolu s ostatními nepublikovanými datasety v adresáři `DIR_DONATED`, nastaveném v konfiguračním souboru. Při publikování se celý adresář datasetu přesune do adresáře `DIR_REPOSITORY`. Adresář datasetu je možné přejmenovat v editaci datasetu. Doporučuji tuto možnost využít a pojmenovat adresář tak, aby bylo hned jasné, který dataset obsahuje. Pozor na omezení použitelných znaků v názvu adresáře. Jak je uvedeno v tabulce ostatních omezení, mohou se použít pouze čísla, malá a velká písmena anglické abecedy a znaky „“, „-“ a „_“. Pokud znaky v názvu datasetu vyhovují omezením, bude Vám pro jméno adresáře nabídnut název datasetu převedený na malá písmena s podtržítka místo mezer. Jméno adresáře se použije i při stahování datasetu pro stahovaný soubor.

Každé škálování datasetu je uloženo v samostatném adresáři. Jméno adresáře je podle identifikačního čísla škálování.

Smazáním škálování smažete krom metadat škálování i soubory škálování (datový a informační). Tlačítko na smazání škálování se nachází v pravém dolním rohu rámečku škálování. Smazáním datasetu smažete celý dataset i s jeho soubory a všemi jeho škálováními s jejich soubory. Odkaz na smazání Datasetu najdete v seznamu datasetů nebo v detailu datasetu.

A.7.4. Omezení vkládaných hodnot

Hodnoty vkládané do formulářových políček jsou omezené aplikací nebo vlastnostmi PHP. Po vložení údaje do políčka je obsah zkontrolován, a když neodpovídá typu políčka, je na to uživatel upozorněn. Do políčka nejde vložit větší počet znaků, než je uveden v následujících tabulkách číslo 3. a 4..

| Typ klíčového slova | Omezení |
|---------------------|--|
| integer | Max. 2147483647 (na 32bit. systému) |
| varchar(255) | Max. 100 znaků |
| date | Při použití US formátu 01/01/1970-01/19/2038 (na 32bit. systému) |
| text | Max. 10000 znaků |
| real | Max. 14 znaků |

Tabulka 3. Omezení hodnot klíčových slov

Aplikace používá kódování znaků UTF-8.

| Formulářové pole | Omezení |
|-----------------------------|--|
| Login | Max. 32 znaků |
| Password | Max. 32 znaků |
| News title | Max. 100 znaků |
| News text | Max. 10000 znaků |
| Keyword name | Max. 100 znaků |
| Enumeration item name | Max. 50 znaků (název položky výčtového typu) |
| Dataset name | Max. 100 znaků |
| Dataset directory name | Max. 50 znaků, možné znaky (0-9a-zA-Z.-_) |
| Name of scaling | Max. 100 znaků |
| Number of Scaled attributes | Max. 2147483647 (na 32bit. systému) |
| Scaling description | Max. 10000 znaků |

Tabulka 4. Ostatní omezení

A.7.5. Úprava klíčových slov (Keywords)

V aplikaci jsou klíčová slova, které jsou vestavěná (použitá v darovacím formuláři nebo tabulce datasetů), proto je není možné smazat ani změnit jejich typ. Můžete pouze změnit pouze jejich název.

Klíčová slova můžete přidávat i mazat. Mazat jen v případě, že žádný dataset toto klíčové slovo nepoužívá. To samé platí i pro změnu typu (integer, boolean, real, atd.) klíčového slova. Nově přidaná klíčová slova se přidají na konec základních charakteristik nebo textových popisů datasetu. Jestliže je klíčové slovo typu text, přidá se jako poslední textový popis, v opačném případě se přidá na konec tabulky charakteristik.

Také položky typu enumeration jdou smazat pouze pokud žádný dataset tuto položku nemá nastavenou. Nové položky jdou přidat bez omezení. Ukázka editace klíčového slova je na 14. obrázku.

A.7.6. Úprava uživatelů (Users)

Změna hesla uživatele se provádí v editaci uživatele. Tam také může změnit roli uživatele. Heslo může být i prázdné, to však důrazně nedoporučuji. 15. obrázek představuje úpravu uživatele aplikace.

A.8. Závěr

Pevně věřím, že se Vám s aplikací bude snadno pracovat a přinese Vám řadu zlepšení oproti dosavadní praxi.

Edit Keyword:"Attribute types"

Name:

Type:

Enumeration:

| | No. | Value |
|--------------------------|-----|--|
| <input type="checkbox"/> | 1 | <input type="text" value="Categorical"/> |
| <input type="checkbox"/> | 2 | <input type="text" value="Integer"/> |
| <input type="checkbox"/> | 3 | <input type="text" value="Real"/> |
| <input type="checkbox"/> | 4 | <input type="text" value="Boolean"/> |

[Back to list](#)

Obrázek 14. Editace klíčového slova typu Enumeration

Edit user:"edie"

Login:

Password:

Confirm Password:

Role:

[Back to list](#)

Obrázek 15. Editace uživatele - změna hesla

B. Obsah příloženého CD

V samotném závěru práce je uveden stručný popis obsahu příloženého CD/DVD, tj. závazné adresářové struktury, důležitých souborů apod.

`bin/`

Kompletní adresářová struktura webové aplikace REPODAT (v ZIP archivu) pro zkopírování na webový server. Adresář obsahuje i všechny další potřebné soubory pro bezproblémový provoz na webovém serveru.

`doc/`

Dokumentace práce ve formátu PDF, vytvořená dle závazného stylu KI PřF pro diplomové práce, včetně všech příloh, a všechny soubory nutné pro bezproblémové vygenerování PDF souboru dokumentace (v ZIP archivu), tj. zdrojový text dokumentace, vložené obrázky, apod.

`src/`

Kompletní zdrojové texty programu webové aplikace REPODAT se všemi potřebnými zdrojovými texty a dalšími soubory pro bezproblémové vytvoření adresářové struktury pro zkopírování na webový server (v ZIP archivu).

`readme.txt`

Instrukce pro nasazení webové aplikace REPODAT na webový server, včetně požadavků pro její provoz, a webová adresa, na které je aplikace nasazena pro účel obhajoby práce.

Navíc CD/DVD obsahuje:

`data/`

Ukázková a testovací data použitá v práci a pro potřeby obhajoby práce.

`install/`

Instalátory aplikací, knihoven a jiných souborů nutných pro provoz webové aplikace, které nejsou standardní součástí operačního systému.