

Sem vložte zadání Vaší práce.

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
KATEDRA SOFTWAREVÉHO INŽENÝRSTVÍ



Diplomová práce

Webová databáze referencí

Bc. Jan Kubálek

Vedoucí práce: doc. Ing. Petr Fišer, Ph.D.

29. dubna 2015

Poděkování

Rád bych poděkoval vedoucímu své diplomové práce panu doc. Ing. Petru Fišerovi, Ph.D. za cenné rady a připomínky, které mi poskytl při zpracování mé diplomové práce.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou, a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu), licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 29. dubna 2015

.....

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2015 Jan Kubálek. Všechna práva vyhrazena.

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

Odkaz na tuto práci

Kubálek, Jan. *Webová databáze referencí*. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2015.

Abstrakt

Tato diplomová práce popisuje a analyzuje současnou aplikaci pro správu publikací, kterou používá skupina pedagogů a studentů z ČVUT. Dále je provedeno srovnání obdobných existujících aplikací a to jak „open-source“, tak komerčních. Na základě analýzy je poté navrhována, naimplementována a na samotný závěr otestována aplikace nová, která překlenuje nedostatky aplikace původní. Nová aplikace je webová, postavena na moderních technologiích, responzivní a plně interaktivní s možností přidávat publikace včetně dokumentů, třídit je, fulltextově vyhledávat v přiložených souborech a podobně. Další funkcionalitou je import a export z/do známých referenčních formátů jako BibTeX, RefWorks, EndNote. Neméně důležitou funkcionalitou je podpora kategorizace, správa uživatelů, uživatelské skupiny a propojení s aplikací třetích stran Springer.

Klíčová slova Aplikace pro správu publikací, BibTeX, RefWorks, EndNote, Webová aplikace, Responzivní design, Nette, jQuery, Bootstrap, Springer API.

Abstract

This diploma thesis describes and analyzes the contemporary reference management software, which is currently utilized by a group of educators and

students at CTU. Furthermore, comparison of the similar applications is made in this thesis – both the open-source ones and the commercial ones. A new application, which surpasses the flaws of the former application, is then designed, implemented and tested on the grounds of the before mentioned analysis. The new application is web based, built on modern technologies, fully responsive and interactive and is provided with many features which allow for example to add publications (including documents), to categorize them and finally to full-textually search in the added files. Yet another feature is the possibility to import to/export from the well-known reference formats such as BibTeX, RefWorks and EndNote. Equally essential traits are also the categorization support, user and user-group administration and the interconnection with the third-party application Springer.

Keywords Reference management software, BibTeX, RefWorks, EndNote, Web application, Responsive design, Nette, jQuery, Bootstrap, Springer API.

Obsah

Úvod	1
1 Cíl práce	3
2 Zadání práce	5
2.1 Obecný popis	5
2.2 Rozbor vztahů a zdefinování pojmů	6
3 Analýza původní aplikace	9
3.1 Obecný popis	9
3.2 Uživatelské role	9
3.3 Cílová skupina	10
3.4 Popis technických parametrů	10
3.5 Analýza struktury a kódu	10
3.6 Analýza bezpečnosti	10
3.7 Analýza databáze	11
3.8 Analýza UI a nefunkční funkcionality	11
3.9 Shrnutí analýzy	19
4 Přehled stávajících řešení	23
4.1 „Open-source“ systémy	23
4.2 Komerční systémy	25
4.3 Shrnutí	31
5 Výběr technologií	35
5.1 Webová aplikace	35
5.2 Server side jazyky/technologie	35
5.3 Framework	38
5.4 Client side jazyk	42
5.5 Mobilní verze stránek	43

6	Use cases	45
6.1	Uživatelské role	45
6.2	Případy užití	46
7	Databáze	61
7.1	Popis změn v tabulkách	61
8	Uživatelské rozhraní a vzhled	67
8.1	Návrh UI (uživatelského rozhraní)	67
9	Realizace	73
9.1	Struktura aplikace	73
9.2	Implementace	76
9.3	Zabezpečení	99
10	Testování	103
10.1	Úvod	103
10.2	White box	103
10.3	Black box	104
	Závěr	107
	Shrnutí všech částí	107
	Přínos práce	109
	Srovnání s konkurencí	109
	Možná rozšíření	109
	Literatura	111
A	Seznam použitých zkratk	117
B	Databázové tabulky	121
C	Instalační příručka	129
C.1	Vlastnosti HTTP Serveru	129
C.2	Aplikace	129
C.3	Databáze	130
D	Uživatelská příručka	131
D.1	Přihlašování	131
D.2	Uživatelský účet	132
D.3	Domovská stránka – vyhledávání	135
D.4	Seznam výsledků vyhledávání	135
D.5	Detail publikace	136
D.6	Přidání nové publikace	138
D.7	Editace publikace	145

D.8 Seznamy	146
D.9 General settings	150
D.10 Waiting for check	150
E Obsah přiloženého CD	153

Seznam obrázků

3.1	Původní aplikace – homepage	12
3.2	Původní aplikace – seznam výsledků vyhledávání Authors/Publication search	13
3.3	Původní aplikace – přidávání nové publikace – přiřazení časopisů	14
3.4	Původní aplikace – přidávání nové publikace – přiřazení autorů	15
3.5	Původní aplikace – přidávání nové publikace – přiřazení konferencí a ročníku konferencí	16
3.6	Původní aplikace – přidávání nové publikace – přidávání souborů	17
3.7	Původní aplikace – přidávání nové publikace – přidávání dodatečných atributů	18
3.8	Původní aplikace – seznam autorů	19
3.9	Původní aplikace – schvalování publikací	20
3.10	Původní aplikace – ER model používající Crow’s foot notaci	21
5.1	Diagram znázorňující komunikaci komponent MVP	41
7.1	Nová aplikace – ER model používající Crow’s foot notaci	62
8.1	Drátěný model – login	70
8.2	Drátěný model – seznamy – schvalování publikací	71
8.3	Drátěný model – detail publikace.	72
9.1	Diagram znázorňující životní cyklus presenteru	75
9.2	Nová aplikace – validátor duplicity autorů	79
D.1	Nová aplikace – login	132
D.2	Nová aplikace – ztráta hesla	133
D.3	Nová aplikace – uživatelský účet – detail	133
D.4	Nová aplikace – uživatelský účet – poznámky	134
D.5	Nová aplikace – uživatelský účet – uživatelská nastavení	134
D.6	Nová aplikace – homepage	135

D.7	Nová aplikace – seznam výsledků vyhledávání	136
D.8	Nová aplikace – detail publikace	137
D.9	Nová aplikace – detail publikace – poznámky	138
D.10	Nová aplikace – detail publikace – definice (export)	139
D.11	Nová aplikace – detail publikace – citace	140
D.12	Nová aplikace – přidání nové publikace	140
D.13	Nová aplikace – přidání nové publikace – import	141
D.14	Nová aplikace – přidání nové publikace – Springer	142
D.15	Nová aplikace – přidání nové publikace – autoři	142
D.16	Nová aplikace – přidání nové publikace – kategorie	143
D.17	Nová aplikace – přidání nové publikace – uživatelské skupiny	144
D.18	Nová aplikace – přidání nové publikace – vydavatelé	144
D.19	Nová aplikace – přidání nové publikace – časopisy	144
D.20	Nová aplikace – přidání nové publikace – konference	145
D.21	Nová aplikace – přidání nové publikace – atributy	145
D.22	Nová aplikace – přidání nové publikace – soubory	146
D.23	Nová aplikace – seznamy – uživatelé	146
D.24	Nová aplikace – seznamy – konference	147
D.25	Nová aplikace – seznamy – formáty citací	147
D.26	Nová aplikace – seznamy – publikace	148
D.27	Nová aplikace – seznamy – moje publikace	148
D.28	Nová aplikace – seznamy – uživatelské skupiny	149
D.29	Nová aplikace – seznamy – moje uživatelské skupiny	149
D.30	Nová aplikace – seznamy – obecné nastavení aplikace	150
D.31	Nová aplikace – seznamy – schvalování publikací	151

Seznam tabulek

4.1	Srovnání systémů pro správu publikací na základě pokročilých funkcí	31
4.2	Srovnání systémů pro správu publikací na základě požadavků na funkcionalitu	33
8.1	Bootstrap: podpora prohlížečů	67
9.1	EndNote tagy	88
9.2	RefWorks tagy	91
9.3	Springer API: Přehled kolekcí a návratových formátů	96
9.4	Springer API: Přehled metod	97
B.1	Tabulka annotation	121
B.2	Tabulka attributes	121
B.3	Tabulka attrib_storage	122
B.4	Tabulka author	122
B.5	Tabulka author_has_publication	122
B.6	Tabulka categories	122
B.7	Tabulka categories_has_publication	123
B.8	Tabulka conference2	123
B.9	Tabulka conference_year	123
B.10	Tabulka documents	124
B.11	Tabulka format	124
B.12	Tabulka general_settings	124
B.13	Tabulka journal	124
B.14	Tabulka publication	125
B.15	Tabulka publisher	126
B.16	Tabulka retrieve	126
B.17	Tabulka submitter	126
B.18	Tabulka submitter_has_publication	126
B.19	Tabulka user_settings	127
B.20	Tabulka group	127

SEZNAM TABULEK

B.21 Tabulka group_has_publication	127
B.22 Tabulka submitter_has_group	127

Úvod

Internet je největším informačním nástrojem dnešní doby. Většina lidí zvládne bez větších problémů informace na internetu vyhledávat a také zveřejňovat. Díky jednoduchosti zveřejňování informací a dokumentů je nárůst informací na internetu enormní. Toto všechno s sebou ovšem přináší několik problémů. Jedním z nich je nestandardní, či špatná struktura informací. Informace zveřejněné na internetu mnohokrát nedodržují standardy pro elektronické publikování a ani standardy jazyka HTML. Bohužel, to znesnadňuje jejich zpracování, uspořádání a také jejich vyhledávání.

Dalším problémem je pravost informací, jelikož informace nebývají většinou nikým kontrolovány a následně ani opravovány. U dokumentů také může chybět jejich autor. Co se týče datování dokumentů, nastává zde další problém – někdy datum chybí úplně, někdy uvedeno, je ale zase není zřejmé, zda se jedná o datum zveřejnění, nebo datum poslední editace.

Na trhu existuje několik řešení pro uchování a třídění informací. Jedním z nich jsou i systémy pro správu publikací. Těchto systémů je nepřeberné množství, přičemž každý má samozřejmě své výhody a nevýhody. Moje práce zahrne jejich analýzu, srovnání a podrobně se podívá na stávající aplikaci pro správu publikací, kterou používá skupina pedagogů a studentů z ČVUT. Tato aplikace byla vyvíjena v rámci dvou bakalářských [1][2] a jedné diplomové práce[3] ale obsahuje řadu nedostatků, které se objevily během jejího dlouhaletého používání.

Pokud analýza prokáže, že aplikace trpí závažnými nedostatky (nevhodná architektura, špatný návrh, chyby ve funkcionalitě, trhliny v zabezpečení, nepřívětivé uživatelské rozhraní), bude aplikace navržena a naprogramována od začátku a to tak, aby tyto nedostatky překlenula.

Cíl práce

Cílem této práce je tedy zanalyzovat existující systémy a zvážit, zda má smysl rozšiřovat stávající aplikaci pro správu referencí, jak již bylo zmíněno v úvodu. Pokud se v analýze prokáží značné nedostatky této aplikace, tak proběhne návrh a implementace nové. Na samotný závěr pak proběhne testování aplikace.

Základní požadavky na funkcionalitu jsou:

- Databáze s webovým přístupem.
- Možnost uchovávání bibliografických informací o publikacích a vyhledávání podle nich.
- Možnost uchovávání souborů (textů publikací).
- Import a export z/do známých referenčních formátů (BibTex, RefWorks, EndNote).
- Vyhledávání a agregace podle jednotlivých položek (autor, zdroj publikace, ...).
- Podpora kategorizace – možnost zařazení publikací do uživatelských skupin.
- Správa uživatelů, možnost sdílení, uživatelské skupiny.

Zadání práce

V této kapitole bych chtěl detailněji rozepsat zadání diplomové práce.

2.1 Obecný popis

Aplikace by měla obsáhnout veškerou funkcionalitu, kterou disponuje stávající aplikace 3.1 a zároveň přidat funkcionalitu novou. V následující části tedy shrnu vše, co by měla ve finále umět.

Aplikace by měla být webová, aby mohla být přístupná z jakéhokoli zařízení s webovým prohlížečem a připojením k internetu. Měla by zde být možnost přidávat bibliografické informace o publikacích a přikládat jejich texty ve formátu PDF. Stávající záznamy o publikacích editovat, mazat a také v nich vyhledávat – to jak fulltextově v dokumentech, tak klasicky v názvech a attributech. Dále by zde měly být seznamy publikací, vydavatelů, kategorií, časopisů, autorů, konferencí, ročníků konferencí, uživatelů, rozšířených atributů a formátů citací, ve kterých by měl mít uživatel možnost vyhledávat, řadit je, jednotlivé položky mazat, editovat a přidávat nové.

Dále by měl být realizovaný import/export publikací do/z formátů BibTeX, RefWorks a EndNote. Uživatel by měl mít možnost zobrazit si detail publikace, kde budou vypsány všechny její atributy (autoři, kategorie, . . .), uživatelské poznámky a také přiložené soubory, které si bude moct každý uživatel stáhnout. Užitečnou funkcionalitou budou styly publikací, které by měly být zobrazeny u každé publikace. Uživatel by si také měl moct přidat publikaci do „svých publikací“ nebo k ní přidat vlastní poznámku – buď soukromou, anebo veřejnou. Dále by zde měla být možnost vytvářet uživatelské skupiny, do kterých se posléze mohou přiřadit publikace. Tyto uživatelské skupiny by si uživatel měl mít možnost přidat do „svých uživatelských skupin“. Uživatelé by také měli mít možnost editovat svůj profil a měnit svá uživatelská nastavení (počet položek u stránkování, . . .).

V aplikaci by měly být čtyři typy rolí – nepřihlášený uživatel, čtenář, zadavatel a administrátor. Podrobnější popis rolí je v části 6.1. Administrátor

by měl mít možnost spravovat uživatele – přidávat je, editovat, mazat, dále měnit globální nastavení aplikace a schvalovat publikace.

Nebylo by také od věci propojit aplikaci skrz API s nějakým katalogem publikací třetích stran – například Springer. Díky tomu by se mohly importovat publikace do databáze například na základě DOI, ISSN nebo ISBN. Uživatel by tedy nemusel vše zdlouhavě opisovat.

Aplikace by také měla obsahovat moderní UI prvky – drag&drop přidávání souborů, drag&drop řazení autorů, zAJAXované formuláře, našeptávače a přinést trochu inovace (notifikace, ...). Určitě by také chtělo zvážit, zda aplikaci neoptimalizovat pro mobilní zařízení (tablety, smartphony) za použití responzivního designu.

2.2 Rozbor vztahů a zadefinování pojmů

V této sekci bych chtěl provést rozbor zadání z hlediska vztahů mezi jednotlivými částmi a také zadefinovat pojmy. Pro tento účel nazývejme tyto části aplikace „objekty“. Objektem rozumějme nějaký předmět, který má specifické vlastnosti (atributy) – publikace, uživatel, kategorie apod. Necht jsou také mezi jednotlivými objekty povinné a nepovinné relace. V DP bude tedy potřeba rozlišovat mnoho níže rozepsaných objektů.

Hlavním objektem bude publikace. Publikace se bude dělit na 13 typů, které budou vycházet z BibTeX formátu 9.2.7.1. Jedná se o následující typy: article, book, booklet, inbook, incollection, inproceedings, manual, misc, mastersthesis, phdthesis, proceedings, techreport, unpublished. Každý typ publikace bude mít určité povinné (název, typ, ...) i nepovinné atributy (abstrakt, poznámka, ...) a zároveň povinné i nepovinné relace na další objekty, které zmíním níže.

Každá publikace může spadat do několika kategorií, a proto bude vytvořen objekt kategorie. Vztah mezi publikacemi a kategoriemi bude M:N, protože publikace spadá do více kategorií a kategorie může být přiřazená k více publikacím.

Pokud je publikace typu „book“, „incollection“ nebo „inbook“, jedná se o knihu, část knihy mající svůj název nebo sekci/kapitolu v knize. U těchto typů je potřeba, kromě názvu a ostatních atributů, evidovat informaci o vydavateli. Tato informace představuje objekt – vydavatel. Každá kniha má právě jednoho vydavatele, ale vydavatel může vydávat více knížek. Proto je mezi objekty vydavatel a publikace vztah 1:N.

Pokud je publikace typu „article“, jedná se o novinový článek nebo článek z nějakého magazínu. Z toho důvodu je také výhodné udržovat informace o časopisech odděleně od samotného objektu publikace. Informace o časopisech budou tedy představovat samostatný objekt a bude mít vztah s publikací 1:N.

Publikace může být také typu „inproceeding“ nebo „proceeding“, čímž rozumíme článek nebo sborník článků z konference. U těchto typů bude také

výhodné udržovat informaci o konferenci zvlášť a umístit ji do separátního objektu, stejně jako v případě časopisu. Každá konference může mít několik ročníků, a tak je třeba ještě oddělit tuto informaci. Vztah mezi konferencí a ročníkem konference bude 1:N a vztah mezi ročníkem konference a publikací bude také 1:N. Na konferenci vznikne sborník prací, který bývá vydán nějakým vydavatelem. Proto ještě přibude vztah mezi vydavatelem a ročníkem konference 1:N.

Publikace také mohou mít autory, kteří se zároveň mohli podílet na tvorbě mnoha dalších publikací. Proto budou autoři odděleni od publikace a bude pro ně vytvořen vlastní objekt. Vztah mezi objektem autorů a publikací bude M:N.

Ke každé publikaci by také mělo být umožněno přidávat rozšířené atributy a to v případě, že by nedostačovaly základní atributy publikace. Pro tento účel bude tedy vytvořen objekt reprezentující tyto rozšířené atributy a bude mít vztah s publikací M:N.

Pro styly publikací bude také potřeba vytvořit vlastní objekt, protože každá publikace by měla mít možnost být naformátována pomocí všech těchto stylů. Relace potřeba nebude.

Za účelem fulltextového vyhledávání se bude muset také vytvořit objekt, který bude reprezentovat informace získané z příložených dokumentů. Mezi tímto objektem a publikací bude vztah 1:1.

Aplikaci budou používat uživatelé, proto i uživatelé budou představovat samostatný objekt.

Každý uživatel by měl mít možnost ukládat své preference aplikace, takže tyto preference budou udržovány v dalším objektu. Vztah mezi uživatelem a uživatelskými preferencemi bude 1:1.

Uživatelé budou moct přidávat poznámky k publikacím, takže poznámky budou dalším objektem se vztahem N:1 k publikaci a N:1 k uživateli.

Uživatelské skupiny jsou jedním z požadavků na funkcionality. Tyto skupiny budou samostatným objektem se vztahem N:1 k publikaci a N:1 k uživateli.

Uživatel by si také měl mít možnost přidat publikaci do „my publications“, a proto zde bude muset být další objekt reprezentující tuto informaci. Vztah mezi tímto objektem a publikací bude N:1 a pak ještě N:1 s uživatelem.

Globální nastavení aplikace bude další objekt, avšak bez jakékoliv relace, protože ta není potřeba.

Analýza původní aplikace

V této kapitole bych chtěl zanalyzovat původní aplikaci – popsat funkcionalitu, zmínit problémy, kterými tato aplikace trpí a navrhnout jejich řešení. Nakonec zhodnotit, zda má smysl v aplikaci pokračovat, anebo bude výhodnější aplikaci naprogramovat a navrhnout od začátku.

3.1 Obecný popis

Aplikace slouží pro uchovávání vědeckých publikací. Její uživatelé mohou přidávat nové publikace, stávající vyhledávat, editovat a mazat. Vyhledávat lze podle názvů publikací, kategorií, autorů, klíčových slov a pak také fulltextově v nahraných dokumentech.

U každé publikace lze zobrazit její detail, kde jsou vypsány veškeré informace jako její typ, autoři, kategorie, rok vydání, počet stran atd. Uživatel si taktéž může stáhnout nahrané dokumenty, zobrazit BibTeX definici, přidat publikaci do svých „publikací“, anebo přidat vlastní poznámku. Poznámka může být buď public (viditelná pro všechny), nebo private (viditelná pouze pro konkrétního uživatele). Dále jsou zde automaticky generované citace pole norem ČSN, ISO 690, IEEE a možnost importu publikace pomocí BibTeX definice.

3.2 Uživatelské role

Aplikace rozlišuje čtyři uživatelské role – nepřihlášený uživatel, čtenář, zadavatel a administrátor. Role s vyšším oprávněním dědí všechna oprávnění od role nižší. Nejnižší oprávnění má uživatel s rolí nepřihlášený uživatel.

Nepřihlášený uživatel se může přihlásit do systému nebo si nechat zaslat odkaz pro resetování hesla.

Čtenář může vyhledávat a zobrazovat publikace, dále přidávat/editovat/mazat vlastní poznámky.

Zadavatel má stejná práva, jako čtenář. Navíc však může přidávat/editovat/mazat publikace, kategorie, autory, vydavatele, časopisy, konference, ročníky konference a dodatečné atributy. Mezi další pravomoc patří možnost přidat/odebrat publikaci do/ze svých publikací, BibTeX import publikace.

Administrátor rozšiřuje schopnosti zadavatele o přidávání/editování/mazání uživatele a schvalování nově vytvořených publikací.

Podrobnější popis je pak v bakalářské práci slečny Kateřiny Pospíšilové [2].

3.3 Cílová skupina

Původní aplikaci používá skupina pedagogů a studentů z ČVUT.

3.4 Popis technických parametrů

Aplikace je psaná ve staré verzi PHP 5.2 v kombinaci se šablonovacím systémem Smarty. Data jsou uložena v MySQL databázi ve verzi 5.

3.5 Analýza struktury a kódu

Při analýze struktury aplikace a kódu, jsem narazil na následující nedostatky. Aplikace je postavena na třívrstvé architektuře (nebo se alespoň tak prezentuje), avšak její filosofii nedodrží – není zde zřejmé, které třídy jsou modelové a které patří do controlleru, protože všechny dělají všechno.

Dále jsou zde kusy kódu, které nejsou vůbec používány. Tyto mrtvé části kódu by bylo vhodné odstranit.

Hned v úvodu tak vyvstává otázka, jestli má smysl aplikaci „opravovat“, anebo naprogramovat od začátku za použití moderních technologií. Uvidíme tedy, co ukáže následující část analýzy.

3.6 Analýza bezpečnosti

V této části popíšu bezpečnostní problémy původní aplikace.

V prvé řadě je zde zásadní bezpečnostní problém a to nefunkční ACL (Access control list) 9.3.2, které je řešeno jen pomocí skrývání položek v menu a to na základě role aktuálně přihlášeného uživatele. Pokud tedy uživatel s nižší rolí zná URL stránek, ke kterým nemá přístup, může si je bez problému zobrazit a vykonávat na nich veškeré akce, které mají povolené pouze role s vyšším oprávněním. ACL je tedy potřeba naimplementovat od začátku.

Další bezpečnostní problém se týká hesel v databázi, která jsou šifrována zastaralou hashovací funkcí MD5 a neobsahují příměs (salt), která by útočníkovi zkomplikovala (v případě získání databáze) dešifrování hesla. Hesla tedy bude potřeba lépe zabezpečit – například SHA funkcí se zmíněným saltem.

3.7 Analýza databáze

V této sekci popíšu chyby v databázi.

V databázi jsem narazil na problémy týkající se sloupců, v nichž jsou uložena data (množné číslo od datum). Pokud ve sloupcích nebyl vyplněn záznam o datu, byla namísto něj uložena nesmyslná hodnota -0001-11-30 00:00:00. Tento problém se týká sloupců `w_year`, `w_to`, `w_from` tabulky `conference_year`, sloupce `first_year` tabulky `conference2` a také sloupce `issue_date` tabulky `publication`.

Tyto nesmyslné záznamy bude potřeba opravit a to nahrazením nulovou hodnotou `NULL`.

3.8 Analýza UI a nefunkční funkcionality

V této části bych chtěl zanalyzovat UI a funkcionality původní aplikace, upozornit na problémy a poté navrhnout jakým způsobem by šly opravit. Na obrázku 3.10 je vidět ER model.

3.8.1 UI Obecně

Vzhled aplikace působí dosti archaickým dojmem, nicméně v tomto případě nehraje až tak důležitou roli, protože se jedná se o „backendovou“ aplikaci u které je především podstatné intuitivní UI.

První věc, která není příliš šťastně vyřešena, je menu, protože je příliš roztáhlé a zabírá hodně místa. Zde by bylo vhodnější použít horizontální vysouvací menu.

Dále chybí jakákoliv informace o aktuálním umístění uživatele, takže těžko zjistíme, kde se uživatel nachází. Tento problém by šel vyřešit zvýrazněním aktuální položky v menu a pak drobečkovou navigací.

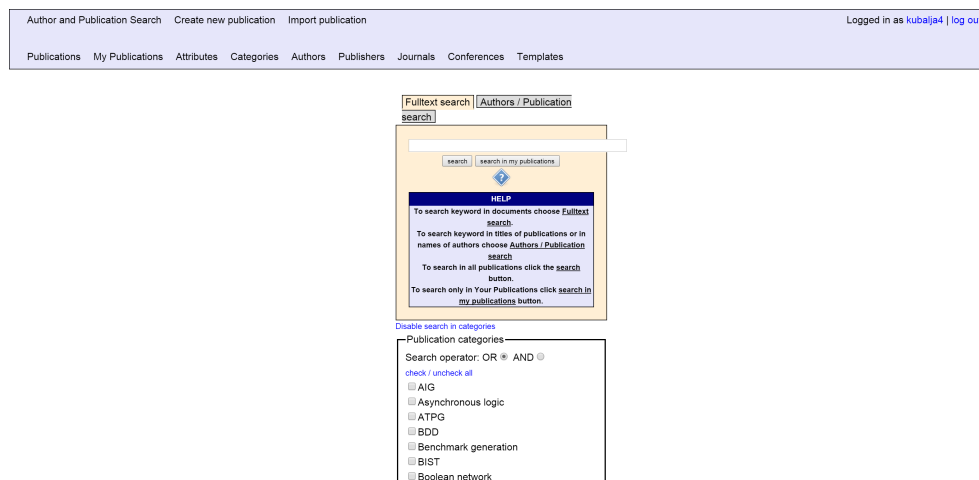
Co se týče formulářů, zde je také několik nedostatků. Povinná formulářová pole jsou sice označena hvězdičkou u popisku, avšak tato hvězdička není vidět. Nejideálnějším řešením bude hvězdičku zvýraznit jinou barvou, případně zvýraznit celé políčko.

Chybové zprávy z validátorů formulářů se nezobrazují u konkrétního chybně vyplněného políčka, ale pouze na začátku formuláře, takže uživatel musí vyrolovat stránku na začátek formuláře, přečíst si chybovou hlášku a poté opět srolovat dolu k danému políčku.

3.8.2 Domovská stránka

Na domovské stránce se nachází vyhledávač v publikacích, který ale zabírá hodně místa, respektive jeho defaultně rozbalené pokročilé nastavení. Vše je vidět na náhledu z aplikace 3.1. Určitě bych toto pokročilé nastavení skryl a nechal na uživateli, zda si jej aktivuje, nebo ne.

3. ANALÝZA PŮVODNÍ APLIKACE



Obrázek 3.1: Původní aplikace – homepage.

Ve vyhledávání také zabírá mnoho prostoru nápověda (pokud ji rozklikneme). Chytřejší řešení by bylo nápovědu umístit do nějakého modálního okna, které bychom poté mohli zavřít.

Poslední výtka patří k dvěma malým vyhledávacím tlačítkům „search“ a „search in my publications“. Nahradil bych je jedním velkým tlačítkem „search“ a poté checkboxem, kde by si uživatel zvolil, zda bude vyhledávat pouze v „my publications“, nebo ne.

3.8.3 Výsledky vyhledávání

3.8.3.1 Fulltext search

Pokud vyhledáme publikace fulltextovým vyhledáváním, tak se nám zobrazí výpis výsledků. V tomto výpisu už chybí samotné vyhledávání – takže pokud chce uživatel změnit parametry vyhledávání, musí se znovu navrátit na domovskou stránku, parametry zadat a pak opětovně vyhledat.

Toto by šlo vyřešit velice jednoduše a to tak, že by se nacházelo na stránce s výpisem i samotné vyhledávání se zadanými parametry.

3.8.3.2 Authors/Publication search

Pokud vyhledáme publikace klasickým vyhledáváním v názvech publikací/jménech autorů, tak se nám zobrazí výpis výsledků. V tomto výpisu je i samotné vyhledávání, avšak na zcela nestandardní pozici – na pravé straně obrazovky. Dále jsou zde nesmyslně vloženy dvě smarty šablony za sebou, takže stránka obsahuje dvakrát html tagy `<head></head>` a `<body></body>` včetně opětovně vložených `.css` a `.js` souborů. Vše je vidět na náhledu z aplikace 3.2.

3.8. Analýza UI a nefunkční funkcionality

The screenshot shows a web application interface for searching authors and publications. At the top, there is a navigation bar with links like 'Author and Publication Search', 'Create new publication', and 'Import publication'. Below this, there are two main sections: 'AUTHOR results' and 'PUBLICATION results'. Each section includes a search bar, a list of authors, and a list of categories. A sidebar on the right contains a 'Publication categories' list with checkboxes for various terms. The page is titled 'Page 1'.

AUTHOR results
Order by: title | year
[New multivalued functional decomposition algorithms based on MDDs \(article\)](#)
Authors: Perkowski A, Marek, Files M, C aig
Categories: Hardware, Logic design, Decomposition, BDD

Page 1

PUBLICATION results
Order by: title | year
[New multivalued functional decomposition algorithms based on MDDs \(article\)](#)
Authors: Perkowski A, Marek, Files M, C aig
Categories: Hardware, Logic design, Decomposition, BDD

Page 1

Publication categories

- Search operator: OR AND
- check / uncheck all
- AIG
- Asynchronous logic
- ATPG
- BDD
- Benchmark generation
- BIST
- Boolean network
- Boolean network flexibility
- Boolean network optimization
- Cellular automata
- CNF
- Covering problem
- Craig Interpolants
- Cut technique
- Decomposition
- Delay estimation
- Delay Test
- Dependability
- Don't cares
- Dynamic Learning
- EXOR Synthesis
- FPGA
- Full state encoding
- Genetics
- Hardware
- Logic design
- Low power
- Model checking
- Multi-valued logic
- On-line testing
- Orthonormalization
- Parallelism
- Radiation
- Randomness
- rewriting
- Robustness
- SAT
- Self-checking
- SEU
- Simulated annealing
- Simulation
- Spectral methods
- SPFD
- Symmetric functions
- Technology mapping
- Test compression
- Tries
- Two-level minimization
- Verification
- XcHr
- ZDDs

Obrázek 3.2: Původní aplikace – seznam výsledků vyhledávání Authors/Publication search.

3.8.4 Přidávání nové publikace

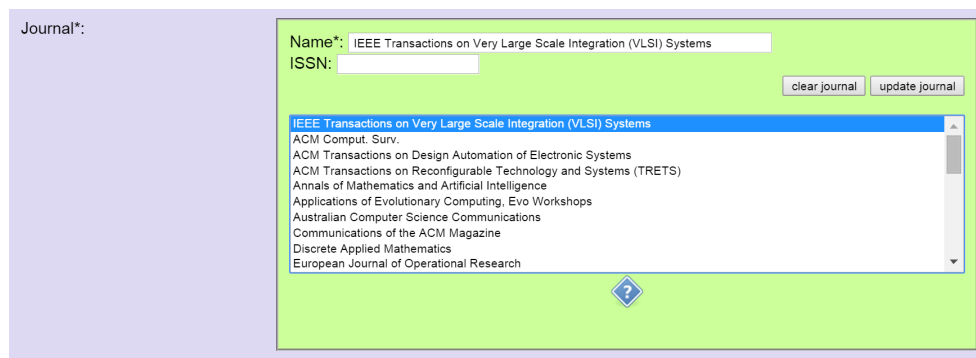
Přidávání publikace je rozděleno do tří kroků mezi kterými se uživatel přepíná tlačítka „previous“ a „next“. Bohužel, uživatel se nedozví ve kterém kroku zrovna je a kolik kroků ho ještě čeká. Tato informace by tedy zde měla být určitě uvedena. Také by byla možnost udělat přidávání pouze jednokrokové.

3.8.4.1 Časopisy

Přiřazení časopisů k publikaci je řešeno pomocí boxu, ve kterém si uživatel vybere časopis, a tak ho tedy přiřadí k dané publikaci. Tímto výběrem se také přednastaví data časopisu do sdíleného formuláře pro editaci a přidávání, který je zabudovaný přímo ve stránce.

Pokud chce uživatel vybraný záznam editovat, tak jej zedituje a potvrdí

3. ANALÝZA PŮVODNÍ APLIKACE



Obrázek 3.3: Původní aplikace – přidávání nové publikace – přiřazení časopisů.

změny kliknutím na tlačítko „update“. Pokud však chce v tomto momentě přidat nový záznam, tak musí nejprve kliknout na tlačítko „clear“, které smaže výběr a přednastavené hodnoty ve formuláři, poté může vyplnit nové hodnoty a kliknutím na tlačítko „add“ hodnoty uloží. Vše je vidět na náhledu z aplikace 3.3. Mazání záznamů pak zde není implementováno vůbec.

Tento styl řešení se mi zdá poněkud nešťastný. Elegantnější by bylo nahradit celý tento sdílený formulář selectboxem a pod ním by se uživateli po výběru časopisu zobrazily jen „read only“ informace.

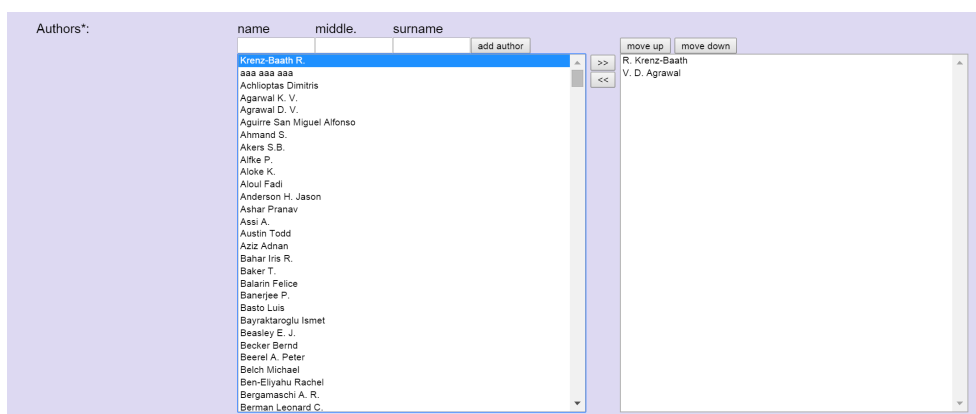
Editace vybraného časopisu by probíhala kliknutím na tlačítko „edit“, kdy by se uživateli zobrazil předvyplněný formulář v modálním okně. Uživatel by záznam zeditoval a změny uložil kliknutím na tlačítko „done“. Podobným stylem by probíhalo přidávání nového časopisu – kliknutím na tlačítko „add new“ by se vyvolal formulář, uživatel by formulář vyplnil a potvrdil tlačítkem „done“.

Pro mazání záznamu by zde bylo tlačítko „delete“, které by stiskem vyvolalo varovnou hlášku, zda si uživatel přeje smazání doopravdy provést. Opětvorným potvrzením by se záznam smazal. Ještě by bylo užitečné zobrazit přidružené publikace vztahující se k aktuálně vybranému časopisu (associated publications). Uživatel by díky tomu měl přehled, kde všude se změny projeví.

3.8.4.2 Autoři

Přiřazení autorů k publikaci je řešeno pomocí dvou boxů, v jednom je seznam všech autorů a ve druhém seznam vybraných autorů. Chybí zde však popisky nad jednotlivými boxy, díky kterým by se uživatel dozvěděl, k čemu vlastně který box slouží. U tohoto formulářového pole by byla také vhodná nápověda, která nyní chybí. Vše je vidět na náhledu z aplikace 3.4.

Dalším problémem je fakt, že pokud uživatel přiřadí autory k publikaci (pomocí tlačítek), tak i nadále zůstávají v seznamu (boxu) všech autorů a uživatel tím pádem má možnost přiřadit více stejných autorů k jedné publikaci.



Obrázek 3.4: Původní aplikace – přidávání nové publikace – přiřazení autorů.

Je zde sice validátor, který přiřazení zabrání, ale to až v případě, že si uživatel označí autora a klikne na tlačítko výběru. Uživatel pak není informován o neprovedení akce.

Určitě by také bylo užitečné implementovat podporu drag&drop přiřazení autorů a to v kombinaci s klasickým přiřazením přes tlačítka. Přiřazení autorů není řešeno JavaScriptem/AJAXem, a tak se stránka pokaždé celá načítá znovu. Toto řešení je velmi nešťastné, protože je to pomalé a uživatel se pokaždé zobrazí začátek stránky, takže si autory musí opětovně vyhledat.

Uživatel může pomocí tlačítek „move up“ a „move down“ měnit pořadí autorů, což je ale opětovně implementováno bez použití JavaScriptu/AJAXu, takže uživateli „skáče“ stránka.

Výše zmíněné přidávání a přerazování pořadí bych určitě vyřešil za pomoci JavaScriptu/AJAXu. Úplně nejelegantnější řešení by bylo najít nějaký hotový doplněk, který by požadovanou funkcionalitu uměl.

Co zde ale chybí je filtrování/vyhledávání autorů. Tato funkcionalita se ukázala jako nezbytná, protože autorů je veliké množství a těžko se v nich uživatel orientuje. Toto by šlo řešit jednoduše přes vyhledávací pole, do kterého by uživatel psal jméno autora, a dle tohoto řetězce by se autoři filtrovali.

Validátor, který by zamezil přidávání stejných autorů jejichž jména mají pouze jiný formát (J. Kubálek vs. Jan Kubálek) zde není implementován vůbec.

3.8.4.3 Vydavatelé

Výběr vydavatelů je řešen podobně jako výběr časopisů a trpí naprosto stejnými problémy.

3. ANALÝZA PŮVODNÍ APLIKACE

The screenshot shows a web form titled "Conference:" with two main sections. The top section is for adding a new conference. It includes fields for "Name*" (International ACM Symposium on Field-Programmable Gate Arrays), "Abbreviation" (FPGA), and "First year" (1993). Below these is a "Description" field and two buttons: "clear conference" and "update conference". A dropdown menu lists various conferences, with "FPGA (International ACM Symposium on Field-Programmable Gate Arrays)" selected. The bottom section is for adding a new conference year. It includes fields for "Name*" (7th ACM/SIGDA international symposium on Field programmable gate arrays), "Year" (1999), "From" (0000-00-00), and "To" (0000-00-00). It also has fields for "Location", "ISBN" (1-58113-088-0), and "Description". A "Publisher" dropdown menu is set to "Select publisher", with an "unset publisher" button. Below are "clear conference year" and "update conference year" buttons. A dropdown menu lists years from 1996 to 2012, with "1999 (7th ACM/SIGDA international symposium on Field programmable gate arrays)" selected.

Obrázek 3.5: Původní aplikace – přidávání nové publikace – přiřazení konferencí a ročníku konferencí.

3.8.4.4 Konference, ročník konference

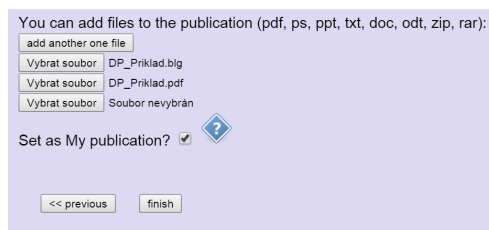
Uživatel může k publikaci přiřadit ročník konference, který zároveň spadá pod konkrétní konferenci. K ročníku konference je také možnost přiřadit vydavatele. Vše je vidět na ER diagramu 3.10 a náhledu z aplikace 3.5.

Výběr konferencí a ročníků konferencí je také řešen poněkud nešťastně – formulář pro přidávání a editaci je zobrazen přímo ve stránce (stejně jako u výběru časopisů) a je pro obě akce sdílený. Uživatel si výběrem v boxu může přiřadit ročník konference k publikaci (tím se současně přednastaví hodnoty do sdíleného formuláře). V okamžiku, kdy si uživatel vybere konferenci nebo ročník konference, ale pak si své plány rozmyslí a chce přidat nový záznam, tak musí nejprve vymazat hodnoty z formuláře kliknutím na tlačítko „clear“.

Pokud chce uživatel vybraný záznam editovat, tak jej může pozměnit a změny potvrdit kliknutím na tlačítko „update“. Mazání záznamů pak zde není implementováno vůbec.

3.8.4.5 Kategorie

Uživatel také může přiřadit k publikaci kategorii, respektive vybrat kategorii, do kterých publikace patří. Nové kategorie lze přidávat, stávající ale nelze



Obrázek 3.6: Původní aplikace – přidávání nové publikace – přidávání souborů.

editovat a mazat. Přidávání je řešeno stejně jako v případě časopisů natvrdo zobrazeným formulářem ve stránce.

Tento problém bych vyřešil stejně jako u časopisů – formulář by se vyvolal stiskem tlačítka a zobrazil v modálním okně.

Jedním z požadavků bylo, aby šly vytvářet stromy kategorií. Toto by šlo určitě vyřešit nějakým hotovým doplňkem, který podporuje hierarchickou strukturu.

3.8.4.6 Přidávání souborů

Přidávání souborů je řešeno také uživatelsky nepřívětivě. Uživatel může přiložit více souborů, musí však každý soubor přiložit zvlášť a to tak, že klikne na tlačítko „add another file“ a tím se mu zobrazí tlačítko pro vybrání souboru. Poté může uživatel konečně soubor vybrat. Vše je vidět na náhledu z aplikace 3.6.

Podpora drag&drop souborů zde také chybí. Tento problém bych vyřešil tak, že by uživatel měl možnost přidat najednou více souborů, které by vybral přes tlačítko, nebo drag&drop přetažením.

3.8.4.7 Dodatečné atributy

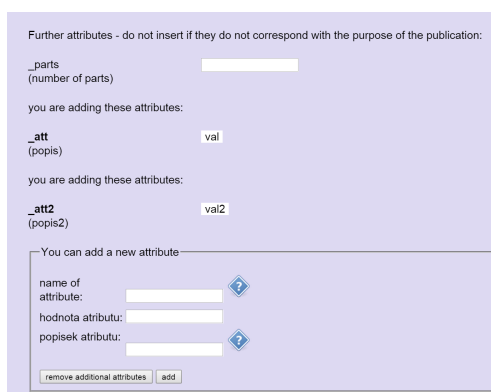
Uživatel může přidat dodatečné atributy. Formulář pro přidávání je však natvrdo umístěný ve stránce, takže stejně jako v případě časopisů, autorů apod. zbytečně zabírá místo a znepřehledňuje celou stránku. Bohužel chybí editace atributu a mazání je řešeno nešikovně a to tak, že se kliknutím na tlačítko „remove additional attributes“ smažou všechny nově přidané atributy. Není možnost smazat pouze jeden konkrétní. Vše je vidět na náhledu z aplikace 3.7.

Toto by šlo také vyřešit daleko lépe – u každého atributu mít tlačítko pro editaci a smazání, které by vyvolalo příslušnou akci. Zvlášť mít pak tlačítko pro přidávání vyvolávající formulář pro přidávání.

3.8.5 Import publikace

Import publikace je tvořen samostatnou stránkou, což je trochu zbytečné. Přidal bych ho jako součást přidávání nové publikace, kde by se po kliknutí

3. ANALÝZA PŮVODNÍ APLIKACE



Obrázek 3.7: Původní aplikace – přidávání nové publikace – přidávání dodatečných atributů.

na tlačítko „import publication“ zobrazilo modální okno s formulářem pro vložení importu.

Samotná funkcionalita také obsahuje řadu chyb a nedostatků. Například pokud není BibTeX ve 100% správném tvaru, tak ho to vůbec nenaimportuje.

Dále chybně zpracovává autory – špatně rozparsuje nebo nenajde v databázi.

3.8.6 Detail publikace

Detaily publikace jsou v aplikaci ze záhadných důvodů dva. Jeden si může zobrazit pouze administrátor nebo zadavatel a to skrz seznam všech publikací. Do druhého se dostane každý uživatel a to skrz vyhledané výsledky z vyhledávání.

Tato duplicita detailů je zbytečná, a proto bych nechal jen jeden. Každý z těchto dvou detailů má odlišně naformátované výpisy a vedou z něj také odkazy na odlišené akce dělající ale to samé – například na editaci publikace.

3.8.7 Editace publikace

Editace publikace, jak jsem již zmínil výše, je duplicitně řešena, takže by ji chtělo sloučit do jedné. Proces editace publikace pak trpí stejnými problémy jako přidávání nové publikace.

3.8.8 Seznamy

Veškeré seznamy publikací, mých publikací, kategorií, autorů, atributů, vydavatelů, časopisů, konferencí a šablon nejsou stránkované, takže jsou uživatelům vylistovány všechny položky najednou. V případě seznamu publikací to je cca 350 položek, takže je zde při načítání jemná prodleva, než se vše načte z databáze a vykreslí. Příklad seznamu autorů je vidět na náhledu z aplikace 3.8.

surname	middlename	name	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="add author"/>
Krenz-Baath		R.	<input type="button" value="change"/> <input type="button" value="delete"/> <input type="button" value="publications"/>
aaa	aaa	aaa	<input type="button" value="change"/> <input type="button" value="delete"/> <input type="button" value="publications"/>
Achlioptas		Dimitris	<input type="button" value="change"/> <input type="button" value="delete"/> <input type="button" value="publications"/>
Agarwal	K.	V.	<input type="button" value="change"/> <input type="button" value="delete"/> <input type="button" value="publications"/>
Agrawal	D.	V.	<input type="button" value="change"/> <input type="button" value="delete"/> <input type="button" value="publications"/>
Aguirre	San Miguel	Alfonso	<input type="button" value="change"/> <input type="button" value="delete"/> <input type="button" value="publications"/>
Ahmand		S.	<input type="button" value="change"/> <input type="button" value="delete"/> <input type="button" value="publications"/>
Akers		S.B.	<input type="button" value="change"/> <input type="button" value="delete"/> <input type="button" value="publications"/>
Alfke		P.	<input type="button" value="change"/> <input type="button" value="delete"/> <input type="button" value="publications"/>
Aloke		K.	<input type="button" value="change"/> <input type="button" value="delete"/> <input type="button" value="publications"/>
Aloul		Fadi	<input type="button" value="change"/> <input type="button" value="delete"/> <input type="button" value="publications"/>
Anderson	H.	Jason	<input type="button" value="change"/> <input type="button" value="delete"/> <input type="button" value="publications"/>
Ashar		Pranav	<input type="button" value="change"/> <input type="button" value="delete"/> <input type="button" value="publications"/>
Assi		A.	<input type="button" value="change"/> <input type="button" value="delete"/> <input type="button" value="publications"/>

Obrázek 3.8: Původní aplikace – seznam autorů.

V seznamech se také špatně pohybuje, protože jsou příliš vertikálně roztažené, nefunguje řazení podle názvu sloupců a nelze v nich vyhledávat. Dále je zde natvrdo zobrazený formulář pro přidávání, který bych určitě nechal schovaný a zobrazoval po zavolání v modálním okně.

Seznamy bych udělal stránkované, dále bych do horní části stránky umístil vyhledávací pole a přidal filtrování podle začátečních písmen.

Také bych sloučil seznam všech publikací a mých publikací do jednoho seznamu a do horní části stránky bych umístil tlačítka, kterými by se mezi seznamy přepínalo.

3.8.9 Schvalování publikací

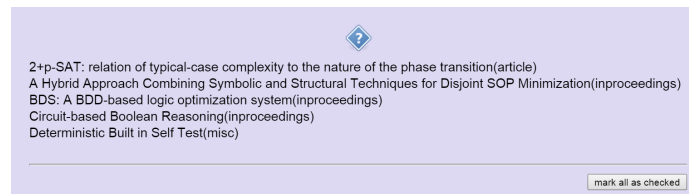
Publikace mohou schvalovat pouze administrátoři a to pouze hromadně skrz seznam publikací ke schválení. Chybí zde ale možnost výběru konkrétních publikací ke schválení. Vše je vidět na náhledu z aplikace 3.9.

Také by nebylo špatné notifikovat administrátora v případě, že se objeví nová publikace ke schválení. Notifikaci bych umístil nahoru do menu lišty, kde by byl vždy vidět počet publikací ke schválení.

3.9 Shrnutí analýzy

Analýza ukázala, že aplikace používá neaktuální webové technologie a je ve velmi špatném stavu – to jak po stránce struktury a kvality kódu, tak po

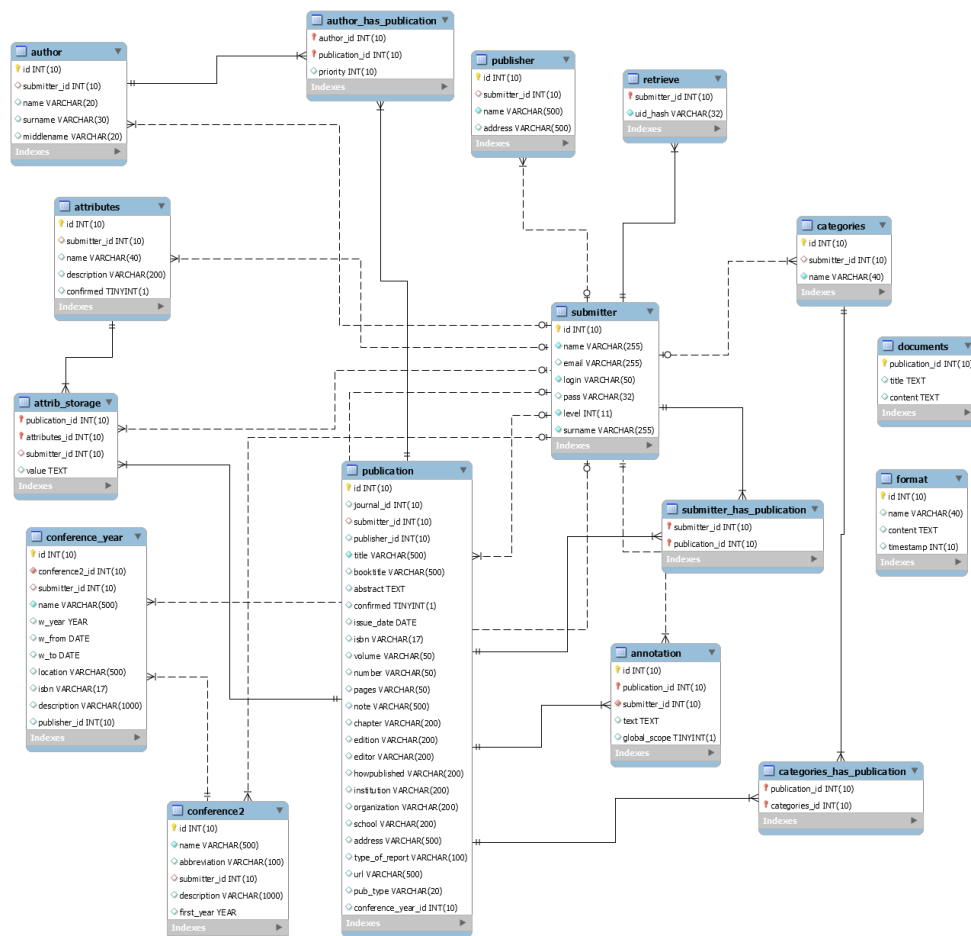
3. ANALÝZA PŮVODNÍ APLIKACE



Obrázek 3.9: Původní aplikace – schvalování publikací.

stránce UI, bezpečnosti a funkčnosti. Je zde několik zásadních chyb, které by bylo velmi pracné odstraňovat.

Na jejím vývoji se postupně vystřídal více studentů, což se právě odrazilo na struktuře a odlišné kvalitě kódu. Jedná se tedy o takový „slepenec“. Proto jsem došel k závěru, že tuto práci nemá smysl opravovat, ale naprogramovat znovu a to za použití aktuálních technologií a trendů moderních webových aplikací.



Obrázek 3.10: Původní aplikace – ER model používající Crow's foot notaci.

Přehled stávajících řešení

V této kapitole bych chtěl uvést přehled neznámějších a nejpoužívanějších programů pro správu publikací. Programy rozřadím do dvou skupin – „open-source“ programy a komerční programy. U každého programu zmíním jeho funkcionalitu a přidám své dojmy a zkušenosti.

4.1 „Open-source“ systémy

4.1.1 BibTeX

BibTeX je nástroj pro správu referencí, který se typicky používá v kombinaci se sázečím systémem LaTeX. BibTeX byl vytvořen v roce 1985 dvěma pány – Oren Pataschnik a Leslie Lamport [4]. Jedna z hlavních myšlenek celého systému je oddělit strukturu a obsah bibliografického dokumentu od jeho vzhledu. Stejnou myšlenku sdílí webové stránky, kde HTML dokument udržuje strukturu stránek a o vzhled se stará CSS.

Výměnou BibTeXových stylů můžeme tedy lehce měnit vzhled seznamu i množství informací, které v něm budou obsaženy. Nemusíme se tedy starat o to, jak jednotlivé reference vysázet. Styly jsou uloženy v textovém souboru s příponou `.bst`, takže je můžeme jednoduše editovat. Stačí je otevřít v nějakém textovém editoru. K dispozici je veliké množství hotových stylů jako AGSM, APA, CHICAGO, ACM, ABBRV atd.

Seznam referencí pro BibTeX má tvar textového souboru s příponou `.bib`. Tento seznam tvoří jakousi strukturovanou databázi položek. Položka představuje bibliografické dílo jako je kniha, časopis, diplomová práce a podobně. Na tyto položky se pak můžeme odkazovat z LaTeX dokumentu, přičemž tu samou databázi může sdílet více dokumentů. BibTeX také disponuje funkcí, kdy do výsledného seznamu referencí zahrne jen ty reference, na které je v textu odkazováno.

Formát BibTeX definice je rozebrán podrobněji v sekci 9.2.7.1.

4.1.2 Zotero

Zotero je „open-source“ software pro správu bibliografických informací a dokumentů. První verze spatřila světlo světa v roce 2006 [5] a byla dostupná pouze jako doplněk do webového prohlížeče Firefox. V roce 2010 vyšla verze číslo 2 stále ještě formou doplňku do Firefoxu, ale změnila licenční podmínky z Educational Community License na GPLv3. Jako samostatná aplikace pro desktopové operační systémy MS Windows, GNU/Linux a Mac OS X vyšel Zotero v roce 2012. Spolu s ním vyšly i doplňky pro webové prohlížeče Safari a Chrome. Nyní je ve verzi 4. Níže bych chtěl v několika bodech zmínit jeho vlastnosti.

- Přidávání, editování, mazání položek [6].
 - Položky si uživatel může řadit do kolekcí, přidávat k nim poznámky, tagy.
 - Položky lze přidat pomocí ISBN, DOI nebo PubMed ID. Zotero poté načte informace o položce z internetu a předvyplní formulář pro přidávání.
- Pokročilé vyhledávání, jehož kritéria si může uživatel uložit jako novou kolekci. Položky v této kolekci jsou průběžně updatovány.
- Díky doplňku do webového prohlížeče a funkci „Web translators“ umožňuje Zotero import záznamů ze zdrojů jako je wikipedia apod. Uživatel pouze při prohlížení stránek klikne na „save to Zotero“ a doplněk vyextrahuje potřebná data ze stránky (včetně přiložených dokumentů) a následně je uloží do databáze.
- Přidávání dokumentů a operace s nimi.
 - Podpora drag&drop přidávání a přidávání skrz webový prohlížeč.
 - Přidávání webových stránek, které jsou uloženy jako snapshoty a jsou přístupné i v režimu offline.
 - Organizování dokumentů – přidávání do složek, oblíbených apod.
 - Přidávání poznámek.
 - Full-textové vyhledávání napříč nahranými PDF dokumenty.
- Mnoho stylů citací [7], které si uživatel může stáhnout z oficiálních stránek. Stávající citace lze editovat či vytvořit zcela nové za pomoci CSL (Citation Style Language).
- Synchronizace knihovny (databáze) mezi počítači, která je prováděna skrz Zotero servery. Jednotlivé položky takto synchronizované databáze jsou přístupné online.

- Práce se skupinami.
 - Uživatelské profily.
 - Vytváření soukromých nebo veřejných skupin, do kterých může uživatel zvát například své kolegy.
 - Interakce s uživateli v rámci skupiny – zprávy, sdílení apod.
 - Vlastní webová stránka pro každou skupinu.
- Doplněk do textových procesorů MS Word a OpenOffice Writer, který umožňuje vkládat citace do dokumentu a snadno je formátovat do různých stylů.

Na Zoteru především oceňuji integraci s webovými prohlížeči. Uživatel tráví prohlížením stránek typu wikipedia.org mnoho hodin denně a díky této úzké integraci nemusí žádné informace manuálně kopírovat. Stačí zavolat příslušnou akci a Zotero se o zbytek postará sám.

4.2 Komerční systémy

4.2.1 EndNote

EndNote je komerční bibliografický software [8] pro správu publikací a referencí, za kterým stojí firma Thomson Reuters. Software je dostupný v několika verzích:

- EndNote X7 – tato verze je určena pro desktopové operační systémy MS Windows a Mac OS X.
- EndNote for iPad – tato verze je určena pro tablety Apple iPad s operačním systémem iOS.
- EndNote Basic – online verze ořezaná o všechny pokročilé funkce.

Já měl tu možnost vyzkoušet si EndNote X7 pro MS Windows v třicetidenní trial verzi, a proto se jí budu v následující části věnovat.

EndNote X7 ocení studenti při psaní rozsáhlejších dokumentů, jako jsou bakalářské, diplomové a disertační práce a to hlavně z důvodu skvělého doplnění do Microsoft Wordu „Cite While You Write“. Své uplatnění si taktéž najde v akademické sféře, takže ho využijí i akademičtí pracovníci při vytváření článků a odborných publikací. Nyní bych chtěl v několika bodech uvést hlavní vlastnosti a vyžití softwaru EndNote X7.

- Disponuje funkcí online vyhledávání, které je zabudováno přímo v programu [9]. Uživatel může vyhledávat v několika online katalozích jako British Library, Library of Congress, Lista (EBSCO) a PubMed.

4. PŘEHLED STÁVAJÍCÍCH ŘEŠENÍ

- Vyhledané informace v online katalozích si lze ukládat do tzv. knihoven. Knihovna je jakási databáze bibliografických položek.
- Položky v knihovně si uživatel může třídit do skupin.
- Přidávat položky lze několika způsoby.
 - Manuálně – uživatel si vybere typ publikace, vyplní příslušná pole, přiloží PDF soubory.
 - Importem – uživatel naimportuje soubor obsahující publikace. Podporovány jsou formáty bibtex, endnote apod.
 - Vyhledáním v online katalozích a následným přidáním.
- Rozlišuje 51 typů publikací 9.2.7.2.
- Nabízí kontrolu pravopisu, tzv. spell checker.
- Obsahuje slovníky slov, které uživatel může dále rozšiřovat a upravovat.
- Přiložené PDF soubory lze zobrazit přímo v programu, takže uživatel nepotřebuje pro jejich prohlížení žádný dodatečný software. K PDF souborům lze psát poznámky, zvýrazňovat důležité části, vyhledávat v nich atd.
- Export citací je možný do HTML, XML, RTF, anebo do čistého textového souboru.
- Doplněk „Cite While You Write“ do Microsoft Wordu, který je součástí instalace EndNote X7, propojí oba programy tak, že se autor textu nemusí starat o rozdílné citační styly. Celý dokument se po stisku jediného tlačítka převede do jiné citační normy.
- Doplněk do Microsoft PowerPoint umožňuje uživateli pohodlně vkládat citace do snímků.
- Umí pokročile vyhledávat v knihovně publikací za pomoci booleovských operátorů.
- Nabízí přes 6000 bibliografických stylů formátování.
- Disponuje funkcí sdílení knihovny až s 14 kolegy kdekoli na světě.

V sekci 9.2.7.2 je pak uveden přehled EndNote tagů, typů referencí a příklad položky.

4.2.2 RefWorks

RefWorks je komerční, webově založený software pro správu publikací. Nyní je vyvíjen firmou RefWorks-COS, založen byl roku 2001. Umožňuje uživatelům založit si vlastní databázi publikací v cloudu [10]. Na oficiálních stránkách se uživateli po registraci zpřístupní 30 denní trial verze. Hlavní vlastnosti toho systému jsou zmíněny níže.

- Uživatel může RefWorks využít [11] jako přímý vyhledávací nástroj pro řadu online zdrojů.
- RefWorks umožňuje uživatelům přístup do nespočet veřejně dostupných online databází, jako např. PubMed, dále také do několika univerzitních online katalogů a předplacených online služeb (např. Ovid, ProQuest).
- RefWorks je kompatibilní s podobnými nástroji umožňující správu bibliografických informací. Uživatel tedy může do RefWorks převádět existující bibliografické informace např. z programu EndNote.
- Uživatel má možnost přiložit k bibliografickému záznamu soubory různých formátů (PDF, TIF, JPG, GIF atd.).
- Write-N-Cite je doplněk do Microsoft Wordu, který umožňuje pohodlně formátovat a vkládat citace do Microsoft Wordu, je potřeba si ho dodatečně stáhnout a nainstalovat.
- Možnost importu a exportu citací do Firefox doplňku Zotero, který umožňuje spravovat citace.
- RefGrab-It je doplněk do webového prohlížeče, který umožňuje získat bibliografické informace z webu a nainportovat je do RefWorks účtu. Podporovány jsou prohlížeče Internet Explorer a Firefox.
- RefMobile je mobilní verze RefWorks [12], která je ořezaná o všechny pokročilé funkce a disponuje pouze základními, jako jsou vyhledávání publikací, vytváření/mazání/editace složek, přidávání komentářů apod.
- V RefWorks je integrovaný RSS Feed Reader, takže uživatel může jednoduše přidávat jeho oblíbené RSS kanály, zobrazovat si informace a importovat data do jeho soukromé RefWorks databáze.
- RefWorks nabízí tisíce výstupních stylů, ze kterých si má uživatel možnost vybírat. Pokud není s žádným stylem spokojen, může použít Output Style Editor, ve kterém si může vytvořit úplně nový styl, nebo upravit stávající do jakékoliv podoby.
- RefWorks nabízí funkci pokročilé vyhledávání pro vyhledávání informací v konkrétních polích (např. autor (Author), titul (Title) apod.). Pomocí

booleovských operátorů (AND, OR, NOT) si uživatel sestaví dotaz, na-
konec může vybrat konkrétní adresáře, nad kterými bude dotaz vykonán.

Výsledky pokročilého vyhledávání jsou standardně zobrazeny abecedně
podle autora se zvýrazněnými vyhledávací termíny, pokud nejsou uživa-
telem parametry speciálně přednastaveny jinak.

- RefShare je nástroj pro sdílení publikací a výzkumných prací s dalšími
uživateli tohoto softwaru. Uživatel může sdílet adresář se záznamy nebo
celou osobní databází.
- Dodatečný obsah a formuláře jsou zobrazovány v modálních oknech

V sekci 9.2.7.3 je pak uveden přehled RefWorks tagů, typů referencí a
příklad položky.

4.2.3 Mendeley

Mendeley je v bezplatný (v základní verzi) desktopový a webový program pro
správu a sdílení publikací. Kombinuje programy Mendeley desktop (správce
PDF souborů a správce referencí) a Mendeley web (online sociální síť pro
vědecké pracovníky). Je dostupný pro desktopové operační systémy Microsoft
Windows [13], Max OS X, Linux a mobilní operační systém iOS.

Mendeley byl založen roku 2007 německými PhD studenty [14] – Victor
Hennin a Jan Reichelt. První veřejná beta byla vypuštěna v roce 2008.

Mendeley taktéž vyhrál řadu ocenění, přičemž za zmínku stojí následující
tři.

- Plugg.eu – Evropský start-up roku 2009 [15].
- TechCrunch Europeas – Nejlepší sociální inovace, která má přínos pro
společnost (2009).
- The Guardian – 6. příčka v top 100 technických mediálních společností.

Mezi investory patří bývalý předseda firmy služby Last.fm, bývalý zakla-
datel programu Skype, bývalý šéf oddělení zabývající se digitálními strategi-
emi ve Warner Muis Group a několik akademických pracovníků z university
Cambridge a z Johns Hopkins University. Přehled základních vlastností tohoto
programu uvedu níže.

- Přidávání PDF dokumentů a operace s nimi.
 - Podpora drag&drop přidávání.
 - Automatická extrakce informací z přiložených PDF dokumentů.
Extrahovány jsou základní atributy jako autoři, název, rok apod.
 - Organizování dokumentů: přidávání do složek, oblíbených apod.

-
- Full-screen čtečka PDF dokumentů, která podporuje přidávání vlastních poznámek a zvýrazňování textu přímo v dokumentu. Tato funkcionality je velice zajímavá a přínosná například pro studenty vysokých škol, kteří píšou nějakou práci, čerpají z elektronických dokumentů a potřebují si k částem textu psát vlastní poznámky. Sám jsem tuto funkcionality nespočetněkrát využil.
 - Sdílení PDF dokumentu včetně přidaných poznámek s ostatními uživateli. Jakmile uživatel poznámku edituje, nebo přidá novou, okamžitě se to promítne všem uživatelům, se kterými daný dokument sdílí.
 - Otvírání více PDF dokumentů najednou, které se zobrazí v záložkách.
 - Ukládání a tisknutí PDF dokumentů včetně přidaných poznámek.
 - Full-textové vyhledávání napříč nahranými PDF dokumenty.
- Import a export.
 - Importování bibliografických položek z webových stránek za pomoci bookmark nástroje „web importer“. Uživatel si jednoduše klikne na přidanou záložku „Import to Mendeley“ a poté se automaticky vyextrahují a zobrazí informace z aktuálně načtené stránky. Uživatel si tyto informace může uložit do své databáze.
 - Import může být taktéž prováděn z BibTeX, RIS a EndNote formátů, takže uživatel může snadno přenést celou svou databázi z programů jako EndNote, Papers a Zotero.
 - Export do BibTeX, RIS a EndNote XML formátů.
 - Doplněk do textových procesorů MS Word, Mac Word a LibreOffice. Doplněk umožňuje stejně jako v případě EndNote doplňku „Cite While You Write“ vkládat citace do dokumentu a snadno je formátovat do různých stylů.
 - Práce v týmu (zpoplatněná funkce).
 - Vytváření soukromých nebo veřejných skupin, do kterých může uživatel pozvat další uživatele.
 - Neomezený prostor pro nahrávání a sdílení dokumentů.
 - Sdílení informací s lidmi v týmu – veškeré úpravy, které uživatel provede, se ostatním členům týmu ihned promítnou.
 - Automatická synchronizace knihovny napříč zařízeními.
 - Zabudované vyhledávání v online katalozích.

- Sociální funkce.
 - Osobní prezentace – online profil přístupný všem uživatelům.
 - Navazování kontaktů.
 - Komentování příspěvků.

4.2.4 Papers

Papers je software pro správu publikací a citací. Podporuje desktopové operační systémy Microsoft Windows a Mac OS X a mobilní operační systém iOS. Papers vznikl pod rukama dvou studentů – Alexander Griekspoor a Tom Groothuis během jejich PhD studia na Netherlans Cancer Institute [16].

Ze zmíněných bibliografických správců je nejmladší, první veřejná verze vyšla roku 2007 a to pouze na Mac Os X. Při vývoji byl kladen důraz na podobnost aplikace s hudební aplikací iTunes. Verze na Mac Os X byla přijata s velikým nadšením a proto v roce 2012 přišla i verze pro MS Windows následována pak verzí pro iOS a webovou službou Papers Online.

Papers je nyní ve verzi 3 pro všechny platformy a přináší kompletně přepracované UI a další mnohá vylepšení. Měl jsem možnost si vyzkoušet verzi pro MS Windows a níže přináším vlastnosti toho programu:

- Zabudované online vyhledávání napříč 25 katalogy (PubMed, Scopus, arXiv, apod.) [17]. Vyhledané publikace lze pak snadno importovat do lokální databáze.
- Navrhování článku na základě toho, co právě čteme.
- Přidávání PDF dokumentů a operace s nimi.
 - Full-screen čtečka PDF dokumentů, která podporuje přidávání vlastních poznámek a zvýrazňování textu přímo v dokumentu.
 - Otevírání více PDF dokumentů najednou, které se zobrazí v záložkách.
 - Vyhledávání v dokumentech.
- Organizace publikací a dokumentů do složek.
- Přes 7000 bibliografických stylů formátování, možnost si vytvořit nové nebo modifikovat stávající.
- Sdílení či synchronizace databáze přes webové uložení Dropbox.
- Webová služba Papers Online:
 - Sdílení položek, kolekcí článků.
 - Spolupráce v týmu.

- Doplněk do textových procesorů MS Word a Mac Word, který umožňuje vkládat citace do dokumentu a snadno je formátovat do různých stylů.

4.3 Shrnutí

Z hlediska UI, se mi nejvíce líbil program Papers. Jeho UI na mě působí svěžím dojmem, líbí se mi vzhled, který se veze na aktuálním trendu „flat designu“ s důrazem na typografii. Co se týče funkčnosti a běhu, tak zde ale Papers ztrácí. Například oproti Mendeley nenabízí tolik funkcí a několikrát se mi stalo, že zamrzl a pomohlo až ukončení procesu.

Z hlediska množství funkcí mě nejvíce oslovil Mendeley, protože i v rámci bezplatné verze toho nabízí v porovnání s ostatními správci suverénně nejvíce. Srovnání pokročilých funkcí jednotlivých programů jsem pak zanesl do tabulky 4.1.

Vyvstává otázka, zda má vůbec smysl aplikaci vyvíjet, když je zde několik kvalitních existujících systémů. Odpověď zní – ano, smysl to má, protože žádný ze systémů nepokrývá všechny požadavky na funkcionalitu zmíněné v kapitole 2. V tabulce 4.2 je vidět, které požadavky na funkcionalitu systému splňují a které nikoliv.

Tabulka 4.1: Srovnání systémů pro správu publikací na základě pokročilých funkcí.

Systémy	Mendeley	EndNote	RefWorks	Zotero	Papers
Základní balíček.	Free	250\$	100\$	Free	79\$
Volně dostupné webové uložení.	2GB	1GB	-	300MB	-
Správa dokumentů:					
Organizace PDF a jiných dokumentů.	ANO	ANO	NE	ANO	ANO
Plugin pro vkládání citací do MS Wordu.	ANO	ANO	ANO	ANO	ANO
Plugin pro vkládání citací do LibreOffice.	ANO	ANO	NE	ANO	ANO
Poznámky / zvýrazňování v PDF dokumentech.	ANO	ANO	NE	NE	ANO

Pokračování na další stránce

4. PŘEHLED STÁVAJÍCÍCH ŘEŠENÍ

Tabulka 4.1 – Pokračování z předešlé stránky

Systémy	Mendeley	EndNote	RefWorks	Zotero	Papers
Synchronizace napříč zařízeními.	ANO	NE	NE	ANO	NE
Získávání znalostí:					
Volně dostupná databáze obsahující řádově 100 milionů dokumentů.	ANO	NE	NE	NE	NE
Personalizovaná doporučení.	ANO	NE	NE	NE	NE
Statistiky čtenosti a komunitní tagy.	ANO	NE	NE	NE	NE
Open Web API.	ANO	NE	NE	ANO	NE
Fulltextové vyhledávání napříč dokumenty.	ANO	ANO	NE	ANO	ANO
Vyhledávání napříč externími databázemi.	NE	ANO	NE	ANO	ANO
Spolupráce:					
Soukromé skupiny.	ANO	ANO	ANO	ANO	NE
Veřejné skupiny.	ANO	NE	NE	ANO	NE
Sociální síť.	ANO	NE	NE	ANO	ANO
Kolabkrační newsfeed.	ANO	NE	NE	ANO	NE
Technologie:					
Webová aplikace.	ANO	ANO	ANO	ANO	NE
Desktopová aplikace.	ANO	ANO	NE	ANO	ANO
Kompatibilita se všemi moderními webovými prohlížeči.	ANO	NE	ANO	NE	ANO
Kompatibilita s WIN/MAC/Linux.	ANO	NE	NE	ANO	NE

Pokračování na další stránce

Tabulka 4.1 – Pokračování z předešlé stránky

Systémy	Mendeley	EndNote	RefWorks	Zotero	Papers
Mobilní/iPad aplikace.	ANO	NE	NE	ANO	NE
Aktivní fórum.	ANO	ANO	NE	ANO	ANO
Integrace do knihovných systémů/EZProxy Support.	NE	ANO	ANO	ANO	ANO
Extrakce metadat:					
Extrakce DOI z PDF dokumentů.	ANO	ANO	ANO	ANO	ANO
Extrakce PubmedID a ArxivID z PDF dokumentů.	ANO	NE	NE	ANO	ANO
Extrakce přiložených metadat z dokumentů.	ANO	ANO	NE	ANO	ANO
Extrakce detailů citací z PDF dokumentů bez přiložených metadat.	ANO	NE	NE	ANO	NE

Tabulka 4.2: Srovnání systémů pro správu publikací na základě požadavků na funkcionalitu.

Systémy	Mendeley	EndNote	RefWorks	Zotero	Papers
Databáze s webovým přístupem.	ANO	ANO	ANO	ANO	NE
Responzivní design.	NE	NE	NE	NE	NE
Moderní UI.	ANO	NE	NE	NE	ANO
Správa publikací.	ANO	ANO	ANO	ANO	ANO
Uchovávání bibliografických informací o publikacích.	ANO	ANO	ANO	ANO	ANO

Pokračování na další stránce

4. PŘEHLED STÁVAJÍCÍCH ŘEŠENÍ

Tabulka 4.2 – Pokračování z předešlé stránky

Systémy	Mendeley	EndNote	RefWorks	Zotero	Papers
Uchovávání textů publikací.	ANO	ANO	ANO	ANO	ANO
Přidávání poznámek.	ANO	ANO	ANO	ANO	ANO
„My publications“.	ANO	ANO	ANO	ANO	ANO
Fulltextové vyhledávání.	ANO	ANO	NE	ANO	ANO
Správa autorů.	ANO	NE	NE	NE	NE
Správa konferencí.	NE	NE	NE	NE	NE
Správa ročníků konferencí.	NE	NE	NE	NE	NE
Správa vydavatelů.	NE	NE	NE	NE	NE
Správa časopisů.	NE	NE	NE	NE	NE
Styly publikací.	ANO	ANO	ANO	ANO	ANO
Dodatečné atributy.	NE	NE	NE	NE	NE
Správa uživatelů.	NE	NE	NE	NE	NE
Uživatelské role.	NE	NE	NE	NE	NE
Uživatelské skupiny, jejich sdílení.	ANO	ANO	NE	ANO	NE
Podpora hierarchické kategorizace.	NE	NE	NE	NE	NE
Export do referenčních formátů: BibTex, RefWorks, EndNote.	ANO	NE	NE	ANO	ANO
Import z referenčních formátů: BibTex, RefWorks, EndNote.	ANO	ANO	ANO	ANO	ANO
Propojení s aplikacemi třetích stran.	ANO	ANO	NE	ANO	ANO

Výběr technologií

Jedním z nejdůležitějších kroků je volba správného programovacího jazyka, respektive webové technologie. Na trhu je nespočet webových technologií, a proto bych v následující kapitole chtěl představit ty nejpoužívanější a dále provést jejich srovnání. Nakonec uvedu, jaké webové technologie jsem pro implementaci diplomové práce vybral a proč.

5.1 Webová aplikace

Jedním z požadavků DP je, aby aplikace byla webová. Co to tedy vůbec webová aplikace je?

Webová aplikace je jakýkoliv program [18], který běží na webovém serveru, je přístupný skrze počítačovou síť internet nebo vnitropodnikovou obdobu Intranet a používá webový prohlížeč jako svého klienta. Klient poté komunikuje se serverem za pomoci HTTP protokolu.

Jedním z hlavních důvodů oblíbenosti webových aplikací je fakt, že se nemusí nikam instalovat. Jakékoliv „update“ probíhají na serveru a uživatel má tedy vždy přístup k aktuální zveřejněné verzi aplikace. Uživatel je odstíněn od rizika, že by mu daná aplikace neběžela na jeho počítači, nebo že by si při instalaci aplikace nainstaloval nějaký vir. Vývojář zase vyvíjí aplikaci pouze na jednu platformu. Díky rozšířenosti internetové konektivity a výše zmíněným důvodům se staly webové aplikace velmi populární.

5.2 Server side jazyky/technologie

5.2.1 PHP

Nejrozšířenějším jazykem pro vývoj webových aplikací je bezesporu PHP s podílem 82% [19]. PHP je skriptovací programovací jazyk, který běží na straně serveru, je nezávislý na platformě a lze ho tedy jednoduše přenášet na deskto-

pové operační systémy [20]. Nejčastěji však běží na operačním systému Linux za použití webového serveru Apache.

PHP obsahuje ve svém základu celou řadu „open-source“ knihoven (přes 5500), modulů pro přístup k databázovým serverům (MySQL, Oracle, PostgreSQL) a modulů pro práci s internetovými protokoly (HTTP, FTP, IMAP, POP3).

PHP je vhodné pro menší až středně velké projekty, nicméně ani velké projekty nejsou pro tento jazyk překážkou – ukázkovým příkladem je sociální síť Facebook.com nebo internetová encyklopedie Wikipedia.org. Z českých projektů pak na PHP běží například iPrima.cz, CSFD.cz, CRO.cz apod. Pomocí PHP jsou taktéž napsány webové frameworky jako Zend, Symfony, nebo v Česku velice populární Nette.

Výhody:

- Snadné a rychlé programování.
- Multiplatformost.
- Podpora na hostingových službách [21].
- Veliké množství knihoven a hotového kódu, který je volně dostupný.
- Velmi kvalitně zpracovaná dokumentace.

Nevýhody:

- Neudržuje kontext aplikace.
- Nejednotné pojmenování funkcí.
- U proměnných se nestanovuje datový typ.

5.2.2 ASP.NET

ASP.net je webová technologie, která je součástí .NET frameworku a umožňuje psát aplikace v jakémkoliv jazyce podporující CLR [22], tedy Visual Basic.NET, JScript.NET, C#, Managed C++. Aplikace postavené na ASP.NET by měly být rychlejší než aplikace napsané skriptovacími jazyky, jelikož jsou předem zkompileovány.

Tato technologie spadá pod křídla Microsoftu a s podílem 17,1% [19] se řadí na pomyslnou druhou příčku co se používá pro vývoj server-side aplikací týče. Na ASP.NET jsou postaveny portály jako Ebay.com, Amazon.com a obecně je vhodná převážně pro větší projekty.

Výhody:

- Využití jazyků podporující CLR.
- Velký důraz na bezpečnost.
- Mnoho předpřipravených komponent jako přihlašování atp.
- Rychlost aplikace díky předzkompilovaným souborům.
- Možnost vyvíjet na mocném vývojovém prostředí Visual Studio.

Nevýhody:

- Vysoká cena licencí (Windows server, SQL server, Visual studio, ...).
- Není multiplatformní.

5.2.3 JavaServer Pages

JavaServer Pages je webová technologie, která je založena na jazyce Java a je součástí platformy JavaEE. Od roku 2010 je vyvíjena firmou Oracle [23].

Stejně jako technologie ASP.Net je vhodná pro velké projekty, pro malé je to tak trochu „kanon na vrabce“. Tato technologie je v současné době využita na 2,8% [19] webových projektů.

Výhody:

- Výkonnost a škálovatelnost.
- Zpětná kompatibilita verzí.
- Multiplatformnost.

Nevýhody:

- Není oddělena business logika a prezentační vrstva.
- Složitě debugování díky překladu a následné kompilaci do Java servletů.
- JSP soubory potřebují dvojnásobné místo na disku.

5.2.4 Shrnutí

Jelikož se jedná o středně veliký projekt a s vývojem v PHP mám několikaleté zkušenosti, rozhodl jsem se právě pro tento programovací jazyk. Dalším krokem bude zvážit, zda nevyužít některý z webových frameworků.

5.3 Framework

Framework (aplikační rámec) je softwarová struktura, která slouží jako podpora při programování, vývoji a organizaci softwarových projektů. Může obsahovat podpůrné programy, knihovny API, podporu pro návrhové vzory nebo doporučené postupy při vývoji tzv. „Best Practice“.

Jednou z hlavních předností frameworků je znovupoužitelnost kódu, díky čemuž se šetří čas vývojářů a zároveň peníze zákazníka. Další nespornou výhodou je vyšší bezpečnost – většina frameworků má minimalizována bezpečnostní rizika. Framework taktéž pomáhá udržovat určitou strukturu aplikace [24].

Pokud se vývojář řídí všemi „coding standards“ daného frameworku, může se pak na projektu podílet bez problému více vývojářů bez jakéhokoliv rozsáhlejšího zaučování.

Neméně podstatnou výhodou je oficiální podpora frameworku a také komunita lidí sdružující se na fórech, která ve většině případů ochotně se vším pomůže.

Pokud se zaměřím na zápory frameworku, tak mezi největší zápory patří nemožnost modifikace chování frameworku a delší doba potřebná pro naučení práce s frameworkem.

5.3.1 Shrnutí

Pokud zvážíme všechny výhody a nevýhody frameworků, tak nám jasně vyjde, že využití frameworku dává smysl, a proto ho využijí při vývoji aplikace.

5.3.2 Výběr frameworku

Dalším krokem bude výběr konkrétního webového frameworku. Zaměřím se pouze na PHP frameworky, protože použiji PHP jazyk pro implementaci. Nejpopulárnější PHP frameworky jsou Zend a Symfony, v České republice pak jednoznačně vládne Nette Framework.

5.3.3 Symfony

Symfony je PHP framework založený na MVC architektuře a je poskytován pod MIT licencí [25]. Symfony je silně inspirován webovými frameworky Ruby on Rails, Django a Spring. Taktéž využívá „open-source“ PHP projekty jako: vrstvu pro objektově relační mapování Doctrine, databázovou abstraktní vrstvu PDO, unit testy PHPUnit, šablonovací systém Twig apod.

Symfony také poskytuje komponenty, které si vývojář může použít pro svůj projekt, pokud nechce využít celý framework. Jedná se například o komponenty:

- **Symfony YAML:** parser, který parsuje YAML řetězce a převádí je do PHP polí a naopak [26].
- **Symfony Event Dispatcher:** komponenta, která implementuje návrhový vzor Mediator (Prostředník) a umožňuje pluginům komunikovat navzájem pomocí vysílání a odposlouchávání událostí [27].
- **Symfony Dependency Injector:** komponenta, která umožňuje standardizovat a centralizovat způsob jakým jsou vytvářeny objekty [28].
- **Symfony Templating:** komponenta, která poskytuje všechny nástroje pro tvorbu vlastního šablonovacího systému [29].
- **Symfony Debug:** komponenta, která poskytuje nástroje pro jednoduché debugování chyb [30].

Symfony pohání velké projekty jako CMS Drupal8, Joomla!, nebo systém pro vytváření fóra phpBB.

Současná verze Symfony, kterou jsem měl možnost vyzkoušet je 2.6. Musím říci, že Symfony je dobrý webový framework známý především v zahraničí, který je vhodný pro větší enterprise projekty. Ve vývojovém prostředí je ale velice pomalý, v produkčním pak díky kešování rychlý. Toto masivní kešování je leckdy trochu otravné, protože je potřeba velmi často po úpravě kódu také manuálně pročistit cache (anebo pomocí příkazu), aby se změny v kódu navenek vůbec promítly.

5.3.4 Zend

Zend Framework je „open-source“ objektově orientovaný framework napsaný v PHP [31]. Založený byl pány: Andi Gutmans a Zeev Suraski, kteří se přímo podílejí na vývoji jazyka PHP.

ZF se skládá z komponent, které na sebe nemají téměř žádnou závislost a vývojáři tedy mohou použít pouze ty komponenty, které chtějí. Tento návrh se nazývá „use-at-will“. Mezi základní komponenty [32] můžeme řadit:

- `Zend\Authentication`: komponenta pro autentizaci.
- `Zend\Mvc`: komponenta pro MVC aplikace.
- `Zend\Db`: komponenta pro přístup a řízení databáze.
- `Zend\Debug`: komponenta pro debugování chyb.
- `Zend\Form`: komponenta pro formuláře.
- `Zend\Validator`: komponenta pro validaci vstupů.

ZF je nyní ve verzi 2.4, která se dle předních PHP vývojářů příliš nepovedla [33][34][35]. Jako hlavní nedostatky uvádí: příliš dlouhé názvy tříd (např. ZendDbTableGatewayAbstractTableGateway), zbytečné velké množství souborů potřebné pro založení modulu, složitá adresářová struktura (např. controllery se nachází až v 5. úrovni), nevyužívá Dependency Injection a stále používá návrhový vzor ActiveRecord.

5.3.5 Nette Framework

Nette Framework je moderní, objektově orientovaný PHP framework, šířen jako svobodný software s dvěma typy licencí [36] – New BSD a GNU General Public License. Vytvořen byl v roce 2004 Davidem Grudlem, nyní se o jeho vývoj stará organizace Nette Foundation. Nette podporuje všechny moderní technologie a koncepce jako AJAX/AJAJ, Dependenci Injection, DRY, KISS, MVC a Web 2.0.

Fungují na něm projekty jako Slevomat, ČSFD, GE Money, Socialbakers, Uložto, DámeJidlo, osobní stránky Václava Klause, Lupa.cz, Rádio Impuls, Root.cz, Deník.cz a mnoho dalších [37].

5.3.5.1 MVC architektura

Nette Framework staví na třívrstvé Mode-View-Controller (MVC) architektuře. MVC architektura je softwarová architektura, která rozděluje aplikaci do tří nezávislých komponent [38]:

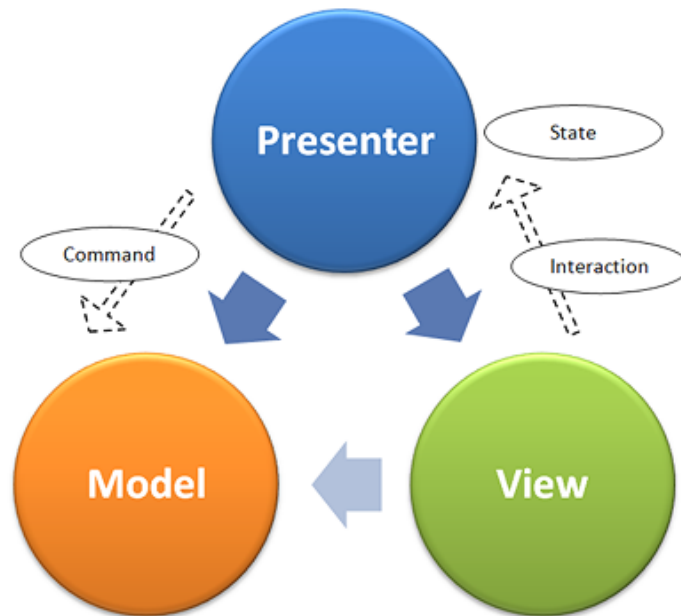
- **Model** je datový a funkční základ aplikace. Obsahuje aplikační logiku (business logiku) a o existenci view nebo controlleru neví.
- **View** neboli pohled zobrazuje uživatelské rozhraní uživateli.
- **Controller** neboli řadič (v případě Nette tedy Presenter) zpracovává požadavky uživatele a podle nich provádí změny v modelu a v pohledu.

Schéma MVP architektury je vidět na obrázku 5.1.

5.3.5.2 Bezpečnost

Nette používá technologie, které eliminují výskyt bezpečnostních děr [37] a jejich zneužití, jako je např. XSS, CSRF, session hijacking, session fixation atd.

Jedna z technologií se nazývá Context-Aware Escaping a zbavuje rizika XSS. Veškeré výstupy automaticky ošetřuje, a tak se nemůže stát, že by útočník do stránky podstrčil svůj vlastní kód a tím stránku pozměnil, anebo získal nějaké citlivé údaje.



Obrázek 5.1: Diagram znázorňující komunikaci komponent MVP (převzato z [39]).

5.3.5.3 Debugování chyb

Pro debugování chyb existuje v Nette nástroj, kterému nikdo neřekne jinak než „Laděnka“. Laděnka pomáhá rychle odhalit a opravit chyby, logovat, vypisovat proměnné a měřit čas.

Součástí Laděnky je taktéž Debugger Bar – plovoucí panel, který se zobrazuje ve spodní části okna prohlížeče. Tento bar programátora informuje o alokované paměti, počtu dotazů apod.

Neméně užitečná věc je zobrazování chyb, které srozumitelně informuje zvýrazněným řádkem, kde k chybě došlo.

V produkčním režimu Laděnka zachycuje všechny chyby do error logu. Je zde taktéž možnost, kterou ocení každý profesionální vývojář a to nechat se informovat e-mailem o každé nově objevené chybě v error logu [40].

5.3.5.4 Doplnky, pluginy, komponenty

V čem Nette ale bezesporu exceluje, je množství hotových doplňků, pluginů a komponent. Vývojáři se činí a podělují se s ostatními o jejich výtvořiny na oficiálních stránkách nette.org. Můžeme zde nalézt doplňky pro SEO, vícenásobné nahrávání souborů, oauth autorizaci, stránkování apod. [41].

5.3.5.5 Latte

Pro psaní view (pohledů) používá Nette vlastní šablonovací jazyk Latte. Latte je rychlý, bezpečný a má intuitivní syntaxi. V mnoha ohledech je podobný šablonovacímu jazyku Smarty [42]. Hojně využívá makra a filtry, dále umožňuje zanořovat bloky do sebe a přidává dědičnost.

5.3.6 Shrnutí

Rozhodnout, který framework pro mou DP použít bylo vcelku jednoduché.

Zend Framework jsem ihned vyloučil, protože je po právu vývojáři kritizován za značné nedostatky. Rozhodoval jsem se tedy pouze mezi Nette a Symfony.

Moje DP je spíše menší projekt a zrovna Symfony je zbytečně robustní framework vhodný pro větší enterprise projekty. Nette oproti tomu neobsahuje tolik pro mě nepotřebných částí a je velice populární u nás v České republice, takže případné řešení komplikací při vývoji je o to snazší.

Díky výše zmíněným důvodům jsem se tedy rozhodl pro použití Nette Frameworku.

5.4 Client side jazyk

Každá moderní aplikace by měla být taktéž dynamická a interaktivní. Pro tento účel se používá skriptovací jazyk JavaScript, který se pro programování těchto „featur“ hodí a který mimo jiné využívá i samotný Nette fw. V dnešní době člověk nenarazí na webovou aplikaci, která by neobsahovala části JavaScriptového kódu.

5.4.1 JavaScript

JavaScript je multiplatformní objektově orientovaný skriptovací jazyk, který nejčastěji běží na straně klienta, ale je ho i možné použít na straně serveru, například za pomoci softwarového systému node.js. Jeho syntaxe patří do rodiny jazyků C/C++, Java [43].

JavaScript vyvinula společnost Netscape v roce 1995 a v roce 1998 byl standardizován organizací ISO. Stal se základem pro další programovací jazyky, jako například ActionScript.

Na straně klienta se nejčastěji používá za účelem rychlé validace formulářů, animací a efektů obrázků, vyskakování oken, řazení položek apod. Nevýhodou tohoto jazyka je, že každý internetový prohlížeč ho interpretuje jinak a právě z toho důvodu je vhodné využít některý JavaScriptový framework, který tento problém řeší.

Pro DP tento jazyk určitě využiji, protože bez něj by bylo nemožné naprogramovat uživatelsky přívětivou aplikaci.

5.4.2 JQuery

JQuery je JavaScriptový framework, který je šířen pod licencí typu MIT. JQuery si klade za cíl zredukovat množství kódu, které by bylo jinak vynaloženo i pro naprosto základní operace, jako je změna barvy písma v odstavci.

Syntaxe jQuery je jednoduchá a intuitivní. Pokud vývojář umí pracovat s CSS selektory, bude mít značnou výhodu, protože se zde hojně využívají při manipulaci s prvky DOMu [44]. Právě jednoduchá práce s elementy DOMu je jednou z hlavních předností toho frameworku. Práce s událostmi je taktéž důležitá a v jQuery to jde opravdu snadno. Velikou doménou toho frameworku je obrovské množství doplňků, které jsou na něm postaveny. Nette Framework ho již v základu používá pro validaci formulářů, AJAX a mnoho dalších věcí, a proto je využití tohoto frameworku celkem logickým a nezbytným krokem.

5.4.3 JQWidgets

JQWidgets je sada více jak 40 doplňků [45] postavených na jQuery a optimalizovaných jak pro desktopové prohlížeče, tak pro mobilní prohlížeče. Můžeme zde nalézt doplňky pro renderování grafů, tvorbu stromových struktur, dále různé progress bary, nav bary, doplňky pro hodnocení a podobně. JQWidgets také obsahují nepřeberné množství témat (skinů) a zároveň i možnost nechat si skin vygenerovat na základě zadaných parametrů. Doplňky tedy po této customizaci působí v aplikaci zcela přirozeně. Co se týče licence, tak pro nekomerční účely je tato sada poskytována jako „open-source“.

Sadu těchto doplňků použiji kvůli velké škále a možností jednotlivých doplňků, a pak hlavně proto, že jako jedna z mála JavaScriptových sad je poskytována pro účely mojí DP zcela zdarma.

Do DP zakomponuji doplněk jqxTree pro strom kategorií a jqxListBox pro výběr, filtrování a řazení autorů.

5.5 Mobilní verze stránek

Chytrá mobilní zařízení (telefony, tablety) se stala nedílnou součástí našeho života. Celosvětové prodeje chytrých telefonů (telefony s operačním systémem) již v roce 2013 překonaly prodeje telefonů hloupých [46]. K tomu si přičtíme fakt, že z mobilních telefonů přistupuje na weby již téměř polovina českých uživatelů [47]. Celkem logicky z toho vyplývá, že je vhodné webové aplikace taktéž optimalizovat pro mobilní zařízení.

5.5.1 Responzivní web design

Responzivní web design [48] je způsob stylování HTML dokumentu, který zaručí, že zobrazení stránky bude optimalizováno pro všechny druhy nejruz-

nějších zařízení. Optimalizací je myšleno, že se samotná stránka a její obsah přizpůsobí zařízení, na kterém je prohlížena.

Hlavní výhodou tohoto řešení je, že není třeba vytvářet nové view (pohledy) pro mobilní verzi stránek. Díky tomu se ušetří čas a s tím související peníze.

Na trhu existují dva hlavní CSS frameworky, které responzivitu stránek řeší za nás. Jedná se o Bootstrap a Foundation. V krátkosti bych je chtěl v následujících odstavcích představit.

5.5.2 Foundation

Foundation je „open-source“, front-end a mobile first framework pro tvorbu responzivních webových stránek [49]. Framework obsahuje několik HTML a CSS návrhů pro formuláře, tlačítka, typografii i samotnou strukturu stránek. Založen byl v roce 2011 firmou ZURB, která se specializuje na interaction design.

5.5.3 Bootstrap

Bootstrap je taktéž „open-source“ [50], front-end a mobile first framework pro tvorbu responzivních webových stránek. Nabízí hotové graficky zpracované elementy, jako jsou tlačítka, boxy atp. Založen byl v roce 2011 dvěma lidmi z Twitteru (Mark Otto a Jacob Thornton).

5.5.4 Srovnání

Měl jsem tu možnost pracovat s oběma frameworky, a proto mohu s klidným srdcem prohlásit, že jsou si frameworky dosti podobné a je tedy celkem jedno, který z nich si vývojář pro svou aplikaci zvolí. Já jsem se rozhodl pro Bootstrap, protože jsou mi sympatičtější jeho grafické prvky, stojí za ním vývojáři z Twitteru, je celkově rozšířenější a má širší podporu internetových prohlížečů.

Use cases

V této kapitole zmíním uživatelské role a k nim příslušné případy užití.

6.1 Uživatelské role

Aplikace bude rozlišovat čtyři typy uživatelských rolí stejně jako aplikace původní 3.2. Každá uživatelská role kromě nepřihlášeného uživatele má svého předka a dědí všechna jeho práva.

6.1.1 Nepřihlášený uživatel

Nepřihlášený uživatel je uživatel aplikace, který není přihlášený do systému. Může si pouze zobrazit přihlašovací obrazovku a zažádat o resetování hesla.

6.1.2 Čtenář

Čtenář je uživatel aplikace, který je přihlášen do systému a má nejnižší možná oprávnění. Tato role budou sloužit, jak už název napovídá, pouze pro čtenáře. Jeho předkem je nepřihlášený uživatel. Oproti původní aplikaci bude moci navíc:

- Změnit si své heslo.
- Zobrazit/editovat/smazat uživatelský profil.

6.1.3 Zadavatel

Zadavatel je uživatel aplikace, který je přihlášen do systému a jeho předkem je čtenář, takže od něj dědí všechna práva. Oproti původní aplikaci bude moci navíc:

- Zakládat/editovat/mazat uživatelské skupiny, sdílet je a přidávat do svých oblíbených.

- Veškeré přidružené záznamy spravovat přímo z formuláře pro přidávání/editování publikace.
- Provádět import publikace z EndNote a RefWorks formátů.
- Větvit kategorie do stromu.
- Filtrovat, stránkovat a řadit seznamy publikací, uživatelských skupin, autorů, časopisů, vydavatelů, konferencí, atributů a formátů.

6.1.4 Administrátor

Administrátor je uživatel aplikace, který je přihlášen do systému a jeho předkem je zadavatel, takže od něj dědí všechna práva. Jedná se o roli s nejvyšším oprávněním. Oproti původní aplikaci bude moci navíc:

- Měnit globální nastavení pro celou aplikaci.

6.2 Případy užití

V této podkapitole bych chtěl uvést případy užití, jehož aktéři jsou uživatel a systém.

6.2.1 UC1

Popis: Zaslání nového hesla.

Aktéři: Nepřihlášený uživatel, systém.

Podmínky pro spuštění: Uživatel se musí nacházet na přihlašovací obrazovce.

Hlavní scénář:

1. Use case začíná, když si nepřihlášený uživatel chce nechat zaslat nového heslo.
2. Nepřihlášený uživatel klikne na požadovanou akci „Forgotten password“.
3. Systém zobrazí nepřihlášenému uživateli modální okno s formulářem.
4. Nepřihlášený uživatel vyplní formulář (zadá e-mail nebo login) a potvrdí akci.
5. Systém provede požadovanou akci, informuje nepřihlášeného uživatele o výsledku akce a odešle mu na e-mail odkaz pro resetování hesla.
6. Nepřihlášený uživatel klikne na odkaz z e-mailu.

7. Systém si ověří, že požadavek na reset hesla byl oprávněný, vygeneruje nové heslo a zašle ho nepřihlášenému uživateli na e-mail.

Alternativní scénář:

4. Pokud nepřihlášený uživatel zadá chybný e-mail nebo login a potvrdí akci.
5. Systém ho informuje, že jím zadané informace jsou neplatné a vyzve ho k opravě zadání.

6.2.2 UC2

Popis: Zobrazení publikace.

Aktéři: Čtenář, systém.

Podmínky pro spuštění: Čtenář se musí nacházet na domovské stránce.

Hlavní scénář:

1. Use case začíná, když si čtenář chce zobrazit publikaci.
2. Čtenář zadá vstupní kritéria pro vyhledávání a zmáčkne tlačítko vyhledat.
3. Systém zobrazí stránkovaný seznam vyhledaných výsledků.
4. Čtenář si vybere publikaci ze seznamu výsledků a pro její zobrazení klikne na název publikace.
5. Systém zobrazí detail publikace.

Alternativní scénář:

3. Pokud zadaná kritéria neodpovídají ani jednomu výsledku, systém tuto skutečnost oznámí a případ užití skončí.

6.2.3 UC3

Popis: Přidání/editace/mazání poznámky.

Aktéři: Čtenář, systém.

Podmínky pro spuštění: Čtenář se musí nacházet na detailu publikace.

Hlavní scénář:

1. Use case začíná, když čtenář chce přidat/editovat/smazat poznámku.

6. USE CASES

2. Čtenář vybere záložku Annotations.
3. Systém zobrazí poznámky.
4. Čtenář klikne na požadovanou akci (Edit/Add new/Delete).
5. Systém zobrazí čtenáři modální okno s:
 - a) Formulářem v případě přidávání a editace.
 - b) Dotazem, zda chce akci doopravdy provést v případě smazání.
6. Čtenář (provede změny a) potvrdí akci.
7. Systém provede požadovanou akci, informuje čtenáře o výsledku akce a zobrazí mu pozměněný obsah.

Alternativní scénář 1:

7. Pokud čtenář zadá chybné informace.
8. Systém ho informuje, že jím zadané informace jsou neplatné a vyzve ho k opravě zadání.

Alternativní scénář 2:

6. Čtenář nepotvrdí změny a akci zruší.

6.2.4 UC4

Popis: Stažení přiloženého dokumentu.

Aktéři: Čtenář, systém.

Podmínky pro spuštění: Čtenář se musí nacházet na detailu publikace.

Hlavní scénář:

1. Use case začíná, když si čtenář chce stáhnout přiložený dokument.
2. Čtenář klikne na požadovanou akci „Download file“.
3. Systém zobrazí okno s dokumentem pro stažení.
4. Čtenář potvrdí stažení dokumentu a souboru.
5. Systém provede požadovanou akci a zobrazí čtenáři pozměněný obsah.

Alternativní scénář 1:

2. Publikace neobsahuje přiložené soubory.

Alternativní scénář 2:

4. Čtenář stahování dokumentu zruší.

6.2.5 UC5

Popis: Zobrazení uživatelského profilu.

Aktéři: Čtenář, systém.

Podmínky pro spuštění: Čtenář se musí nacházet na domovské stránce.

Hlavní scénář:

1. Use case začíná, když si čtenář chce zobrazit svůj uživatelský profil.
2. Čtenář klikne v horním menu na ikonku se svým loginem.
3. Systém zobrazí stránku s jeho uživatelským profilem.

6.2.6 UC6

Popis: Editace/smazání uživatelského profilu.

Aktéři: Čtenář, systém.

Podmínky pro spuštění: Čtenář se musí nacházet na stránce se svým uživatelským profilem.

Hlavní scénář:

1. Use case začíná, když si čtenář chce změnit/smazat svůj uživatelský profil.
2. Čtenář klikne na požadovanou akci (Edit/Delete).
3. Systém zobrazí čtenáři modální okno s:
 - a) Formulářem v případě přidávání a editace.
 - b) Dotazem, zda chce akci doopravdy provést v případě smazání.
4. Čtenář (provede změny a) potvrdí akci.
5. Systém provede požadovanou akci, informuje čtenáře o výsledku akce a zobrazí mu pozměněný obsah.

Alternativní scénář 1:

5. Pokud čtenář zadá chybné informace.
6. Systém ho informuje, že jím zadané informace jsou neplatné a vyzve ho k opravě zadání.

Alternativní scénář 2:

4. Čtenář nepotvrdí změny a akci zruší.

6.2.7 UC7

Popis: Zobrazení seznamu všech autorů, časopisů, vydavatelů, konferencí, kategorií, rozšířených atributů a formátů citací.

Aktéři: Zadavatel, systém.

Podmínky pro spuštění: Zadavatel se musí nacházet na domovské stránce.

Hlavní scénář:

1. Use case začíná, když si zadavatel chce zobrazit příslušný seznam položek.
2. Zadavatel klikne v horním menu na položku „Administration“ a poté vybere příslušný seznam položek.
3. Systém zobrazí stránkovaný seznam položek.

6.2.8 UC8

Popis: Přidání/editace/mazání autorů, časopisů, vydavatelů, konferencí, kategorií, rozšířených atributů nebo formátů citací.

Aktéři: Zadavatel, systém.

Podmínky pro spuštění: Zadavatel se musí nacházet příslušném seznamu položek.

Hlavní scénář:

1. Use case začíná, když zadavatel chce přidat/editovat/smazat autory, časopisy, vydavatele, konference, kategorie, rozšířené atributy nebo formáty citací.
2. Zadavatel klikne na požadovanou akci (Edit/Add new/Delete).
3. Systém zobrazí zadavateli modální okno s:
 - a) Formulářem v případě přidávání a editace.
 - b) Dotazem, zda chce akci doopravdy provést v případě smazání.
4. Zadavatel (provede změny a) potvrdí akci.
5. Systém provede požadovanou akci, informuje zadavatele o výsledku akce a zobrazí mu pozměněný obsah.

Alternativní scénář 1:

5. Pokud zadavatel zadá chybné informace.

6. Systém ho informuje, že jím zadané informace jsou neplatné a vyzve ho k opravě zadání.

Alternativní scénář 2:

4. Zadavatel nepotvrdí změny a akci zruší.

6.2.9 UC9

Popis: Zobrazení přidružených publikací u autorů, časopisů, vydavatelů, konferencí, kategorií nebo rozšířených atributů.

Aktéři: Zadavatel, systém.

Podmínky pro spuštění: Zadavatel se musí nacházet v příslušném seznamu položek.

Hlavní scénář:

1. Use case začíná, když si zadavatel chce zobrazit přidružené publikace u autorů, časopisů, vydavatelů, konferencí, kategorií nebo rozšířených atributů.
2. Zadavatel klikne na požadovanou akci „Associated Publications“.
3. Systém zobrazí zadavateli modální okno se seznamem přidružených publikací.

6.2.10 UC10

Popis: Zobrazení seznamu všech publikací.

Aktéři: Zadavatel, systém.

Podmínky pro spuštění: Zadavatel se musí nacházet na domovské stránce.

Hlavní scénář:

1. Use case začíná, když si zadavatel chce zobrazit seznam všech publikací.
2. Zadavatel klikne v horním menu na položku „Publications“ a poté „All Publications“.
3. Systém zobrazí stránkovaný seznam všech publikací.

6.2.11 UC11

Popis: Přidání/editace/mazání publikací.

Akteři: Zadavatel, systém.

Podmínky pro spuštění: Zadavatel se musí nacházet v seznamu všech publikací.

Hlavní scénář:

1. Use case začíná, když zadavatel chce přidat/editovat/smazat publikace.
2. Zadavatel klikne na požadovanou akci (Edit/Add new/Delete)
3. Systém zobrazí zadavateli:
 - a) Stránku s formulářem v případě přidávání a editace.
 - b) Modální okno s dotazem, zda chce akci doopravdy provést v případě smazání.
4. Zadavatel (provede změny a) potvrdí akci.
5. Systém provede požadovanou akci, informuje zadavatele o výsledku akce a zobrazí mu pozměněný obsah.

Alternativní scénář 1:

3. Zadavatel může přidat/editovat/smazat autory, časopisy, vydavatele, konference, kategorie, rozšířené atributy nebo uživatelské skupiny.

Alternativní scénář 2:

5. Pokud zadavatel zadá chybné informace.
6. Systém ho informuje, že jím zadané informace jsou neplatné a vyzve ho k opravě zadání.

Alternativní scénář 3:

4. Zadavatel nepotvrdí změny a akci zruší.

6.2.12 UC12

Popis: Zobrazení seznamu mých publikací.

Aktéři: Zadavatel, systém.

Podmínky pro spuštění: Zadavatel se musí nacházet na domovské stránce.

Hlavní scénář:

1. Use case začíná, když si zadavatel chce zobrazit seznam svých publikací „My Publications“.
2. Zadavatel klikne v horním menu na položku „Publications“ a poté „My Publications“.
3. Systém zobrazí stránkovaný seznam jeho publikací „My Publications“.

6.2.13 UC13

Popis: Přidání/odebrání publikace do/z mých publikací.

Aktéři: Zadavatel, systém.

Podmínky pro spuštění: Zadavatel se musí nacházet na seznamu všech publikací.

Hlavní scénář:

1. Use case začíná, když si zadavatel chce přidat/odebrat publikaci do/ze svých publikací „My Publications“.
2. Zadavatel klikne u konkrétní publikace na ikonku „add to my publications“, anebo „remove from my publications“ podle toho, zda chce publikaci přidat, nebo odebrat ze svých publikací „My Publications“.
3. Systém provede požadovanou akci, informuje zadavatele o výsledku akce a zobrazí mu pozměněný obsah.

6.2.14 UC14

Popis: Přidání nové publikace pomocí BibTex/EndNote/RefWorks importu.

Aktéři: Zadavatel, systém.

Podmínky pro spuštění: Zadavatel se musí nacházet na stránce pro přidávání nové publikace.

Hlavní scénář:

6. USE CASES

1. Use case začíná, když zadavatel chce přidat publikaci pomocí BibTex/End-Note/RefWorks importu.
2. Zadavatel klikne na ikonu „Import Definition“.
3. Systém zobrazí zadavateli modální okno formulářem pro import.
4. Zadavatel vyplní formulář a potvrdí akci.
5. Systém provede požadovanou akci, informuje zadavatele o výsledku akce a zobrazí mu pozměněný obsah – předvyplněný formulář pro přidání nové publikace hodnotami z provedeného importu.
6. Zadavatel (provede změny a) potvrdí akci.
7. Systém provede požadovanou akci, informuje zadavatele o výsledku akce a zobrazí mu pozměněný obsah – detail nově přidané publikace.

Alternativní scénář 1:

6. Zadavatel může přidat/editovat/smazat autory, časopisy, vydavatele, konference, kategorie, rozšířené atributy nebo uživatelské skupiny.

Alternativní scénář 2:

7. Pokud zadavatel zadá chybné informace.
8. Systém ho informuje, že jím zadané informace jsou neplatné a vyzve ho k opravě zadání.

Alternativní scénář 3:

6. Zadavatel nepotvrdí změny a akci zruší.

6.2.15 UC15

Popis: Přidání nové publikace pomocí stažení dat ze Springer katalogu.

Aktéři: Zadavatel, systém.

Podmínky pro spuštění: Zadavatel se musí nacházet na stránce pro přidávání nové publikace.

Hlavní scénář:

1. Use case začíná, když zadavatel chce přidat publikaci pomocí stažení dat ze Springer katalogu.
2. Zadavatel klikne na ikonu „Fetch data from Springer“.

3. Systém zobrazí zadavateli modální okno formulářem pro stažení dat.
4. Zadavatel vyplní formulář a klikne na „Fetch data“.
5. Systém zobrazí zadavateli stažená data.
6. Zadavatel vybere konkrétní data a klikne na „Import data“.
7. Systém provede požadovanou akci, informuje zadavatele o výsledku akce a zobrazí mu pozměněný obsah – předvyplněný formulář pro přidání nové publikace hodnotami ze stažených dat ze Springer katalogu.
8. Zadavatel (provede změny a) potvrdí akci.
9. Systém provede požadovanou akci, informuje zadavatele o výsledku akce a zobrazí mu pozměněný obsah – detail nově přidané publikace.

Alternativní scénář 1:

8. Zadavatel může přidat/editovat/smazat autory, časopisy, vydavatele, konference, kategorie, rozšířené atributy nebo uživatelské skupiny.

Alternativní scénář 2:

9. Pokud zadavatel zadá chybné informace.
10. Systém ho informuje, že jím zadané informace jsou neplatné a vyzve ho k opravě zadání.

Alternativní scénář 3:

8. Zadavatel nepotvrdí změny a akci zruší.

6.2.16 UC16

Popis: Zobrazení seznamu všech uživatelských skupin.

Aktéři: Zadavatel, systém.

Podmínky pro spuštění: Zadavatel se musí nacházet na domovské stránce.

Hlavní scénář:

1. Use case začíná, když si zadavatel chce zobrazit seznam všech uživatelských skupin.
2. Zadavatel klikne v horním menu na položku „Groups“ a poté „All Groups“.
3. Systém zobrazí stránkovaný seznam všech uživatelských skupin.

6.2.17 UC17

Popis: Zobrazení seznamu mých uživatelských skupin.

Aktéři: Zadavatel, systém.

Podmínky pro spuštění: Zadavatel se musí nacházet na domovské stránce.

Hlavní scénář:

1. Use case začíná, když si zadavatel chce zobrazit seznam svých uživatelských skupin „My Groups“.
2. Zadavatel klikne v horním menu na položku „Groups“ a poté „My Groups“.
3. Systém zobrazí stránkovaný seznam jeho uživatelských skupin „My Groups“.

6.2.18 UC18

Popis: Přidání/editace/mazání uživatelských skupin.

Aktéři: Zadavatel, systém.

Podmínky pro spuštění: Zadavatel se musí nacházet na seznamu všech skupin.

Hlavní scénář:

1. Use case začíná, když zadavatel chce přidat/editovat/smazat uživatelské skupiny.
2. Zadavatel klikne na požadovanou akci (Edit/Add new/Delete).
3. Systém zobrazí zadavateli modální okno s:
 - a) Formulářem v případě přidávání a editace.
 - b) Dotazem, zda chce akci doopravdy provést v případě smazání.
4. Zadavatel (provede změny a) potvrdí akci.
5. Systém provede požadovanou akci, informuje zadavatele o výsledku akce a zobrazí mu pozměněný obsah.

Alternativní scénář 1:

5. Pokud zadavatel zadá chybné informace.
6. Systém ho informuje, že jím zadané informace jsou neplatné a vyzve ho k opravě zadání.

Alternativní scénář 2:

4. Zadavatel nepotvrdí změny a akci zruší.

6.2.19 UC19

Popis: Přidání/odebrání skupiny do/z mých skupin.

Aktéři: Zadavatel, systém.

Podmínky pro spuštění: Zadavatel se musí nacházet na seznamu všech skupin.

Hlavní scénář:

1. Use case začíná, když si zadavatel chce přidat/odebrat uživatelskou skupinu do/ze svých uživatelských skupin „My Groups“.
2. Zadavatel klikne u konkrétní publikace na ikonku „add to my groups“ nebo „remove from my groups“ podle toho, zda chce publikaci přidat, nebo odebrat ze svých uživatelských skupin „My Groups“.
3. Systém provede požadovanou akci, informuje zadavatele o výsledku akce a zobrazí mu pozměněný obsah.

6.2.20 UC20

Popis: Zobrazení seznamu všech uživatelů.

Aktéři: Administrátor, systém.

Podmínky pro spuštění: Administrátor se musí nacházet na domovské stránce.

Hlavní scénář:

1. Use case začíná, když si administrátor chce zobrazit seznam všech uživatelů.
2. Administrátor klikne v horním menu na položku „Administration“ a poté „All Users“.
3. Systém zobrazí stránkovaný seznam uživatelů.

6.2.21 UC21

Popis: Přidání/editace/mazání uživatelů.

Aktéři: Administrátor, systém.

Podmínky pro spuštění: Administrátor se musí nacházet příslušném seznamu uživatelů.

Hlavní scénář:

6. USE CASES

1. Use case začíná, když zadavatel chce přidat/editovat/smazat uživatele.
2. Administrátor klikne na požadovanou akci (Edit/Add new/Delete).
3. Systém zobrazí administrátorovi modální okno s:
 - a) Formulářem v případě přidávání a editace.
 - b) Dotazem, zda chce akci doopravdy provést v případě smazání.
4. Administrátor (provede změny a) potvrdí akci.
5. Systém provede požadovanou akci, informuje administrátora o výsledku akce a zobrazí mu pozměněný obsah.

Alternativní scénář: 1

5. Pokud administrátor zadá chybné informace.
6. Systém ho informuje, že jím zadané informace jsou neplatné a vyzve ho k opravě zadání.

Alternativní scénář: 2

4. Administrátor nepotvrdí změny a akci zruší.

6.2.22 UC22

Popis: Zobrazení seznamu všech publikací čekající na schválení.

Aktéři: Administrátor, systém.

Podmínky pro spuštění: Administrátor se musí nacházet na domovské stránce.

Hlavní scénář:

1. Use case začíná, když si administrátor chce zobrazit seznam všech publikací čekající na schválení.
2. Administrátor klikne v horním menu na položku „Administration“ a poté „Waiting for check“.
3. Systém zobrazí seznam publikací čekajících na schválení.

6.2.23 UC23

Popis: Hromadné schvalování publikací.

Aktéři: Administrátor, systém.

Podmínky pro spuštění: Administrátor se musí nacházet na seznamu všech publikací.

Hlavní scénář:

1. Use case začíná, když si administrátor chce hromadně schválit publikace.
2. Administrátor klikne na tlačítko „Select/Deselect all“.
3. Systém zaškrtně všechny publikace na schválení.
4. Administrátor potvrdí hromadné schválení všech publikací kliknutím na tlačítko „Confirm“.
5. Systém provede požadovanou akci, informuje administrátora o výsledku akce a zobrazí mu výsledek akce.

Alternativní scénář:

4. Administrátor nepotvrdí změny a akci zruší.

6.2.24 UC24

Popis: Zobrazení globálního nastavení aplikace „General Settings“.

Aktéři: Administrátor, systém.

Podmínky pro spuštění: Administrátor se musí nacházet na domovské stránce.

Hlavní scénář:

1. Use case začíná, když si administrátor chce zobrazit globální nastavení aplikace.
2. Administrátor klikne v horním menu na položku „General Settings“.
3. Systém zobrazí stránku s globálním nastavením aplikace.

6.2.25 UC25

Popis: Editace globálního nastavení aplikace „General Settings“.

Akteři: Administrátor, systém.

Podmínky pro spuštění: Administrátor se musí nacházet na stránce s globálním nastavením.

Hlavní scénář:

1. Use case začíná, když administrátor chce změnit globální nastavení aplikace.
2. Administrátor klikne na požadovanou akci Edit.
3. Systém zobrazí administrátorovi modální okno s formulářem.
4. Administrátor (provede změny a) potvrdí akci.
5. Systém provede požadovanou akci, informuje administrátora o výsledku akce a zobrazí mu pozměněný obsah.

Alternativní scénář: 1

5. Pokud administrátor zadá chybné informace.
6. Systém ho informuje, že jím zadané informace jsou neplatné a vyzve ho k opravě zadání.

Alternativní scénář: 2

4. Administrátor nepotvrdí změny a akci zruší.

Databáze

V této kapitole se zaměřím na rozbor databáze.

Aplikace bude stavět na databázi, kterou používala původní aplikace. Jedná se o MySQL databázi, která jako storage engine (formát uložení dat) používá pro své tabulky až na výjimku 7.1.10 InnoDB.

InnoDB [51] je moderní storage engine, který podporuje transakce, cizí klíče a je takovým standardem dnešních MySQL databází.

MyISAM je oproti InnoDB zastaralý formát, nicméně podporuje fulltextové indexy, které jsou právě využity pro funkci fulltextového vyhledávání.

V databázi se nachází 22 tabulek – některé jsem nově přidal a některé upravil. Níže bych chtěl rozepsat seznam všech tabulek včetně změn, které jsem na nich provedl. Na obrázku 7.1 je vidět ER model. Podrobnější popis všech tabulek včetně schématu se pak nachází v příloze B.

7.1 Popis změn v tabulkách

7.1.1 annotation

Tabulka `annotation` slouží pro uchovávání komentářů k publikacím. Poznámka může být veřejná nebo soukromá.

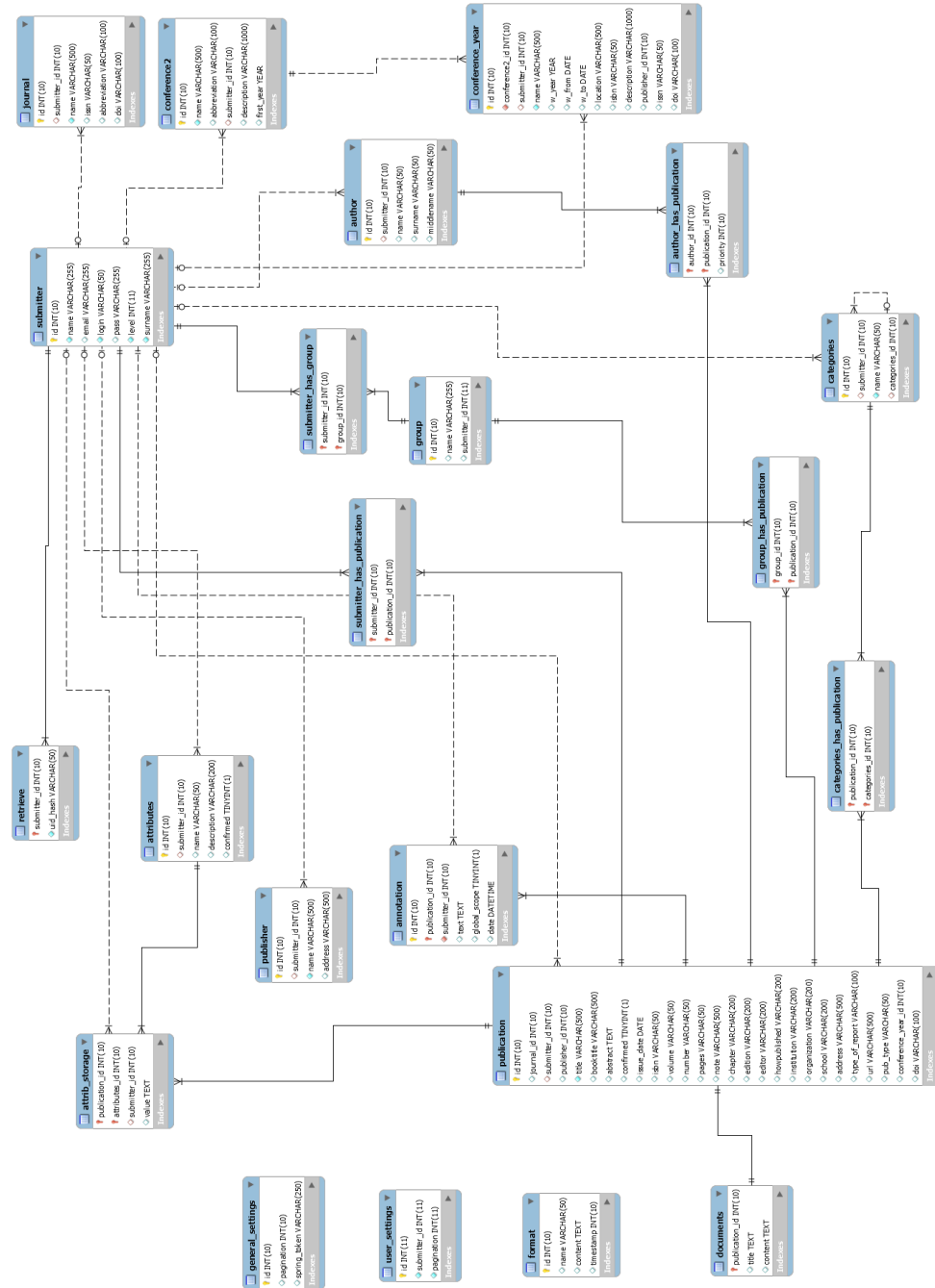
Změny:

- Přidán atribut `date`, do kterého se ukládá datum přidání/editace poznámky.

7.1.2 attributes

Tabulka `attributes` slouží pro uchovávání rozšířených atributů publikací.

7. DATABÁZE



Obrázek 7.1: Nová aplikace – ER model používající Crow's foot notaci.

7.1.3 `attrib_storage`

Tabulka `attrib_storage` slouží jako pomocná tabulka pro vztah M:N mezi `attributes` a `publication`.

7.1.4 `author`

Tabulka `author` slouží pro uchovávání autorů publikací.

7.1.5 `author_has_publication`

Tabulka `author_has_publication` slouží jako pomocná tabulka pro vztah M:N mezi `author` a `publication`.

7.1.6 `categories`

Tabulka `categories` slouží pro uchovávání kategorií.

Změny:

- Přidán atribut `categories_id`, který je referencí na tu samou tabulku `categories`. Tímto je docíleno toho, že se můžou tvořit stromy kategorií.

7.1.7 `categories_has_publication`

Tabulka `categories_has_publication` slouží jako pomocná tabulka pro vztah M:N mezi `categories` a `publication`.

7.1.8 `conference2`

Tabulka `conference2` slouží pro ukládání konferencí.

Změny:

- Opraven problém s formátem dat ve sloupci `first_year`.

7.1.9 `conference_year`

Tabulka `conference_year` slouží pro uchovávání ročníků konferencí. Pokud je publikace typu `inproceedings` nebo `proceedings`, pak je v tabulce `publication` ve sloupci `conference_year_id` uložena reference právě na tuto tabulku ročníků konferencí `conference_year`.

Změny:

- Opraven problém s formátem dat ve sloupcích `w_year`, `w_from`, `w_to`.
- Přidány atributy:

7. DATABÁZE

- `issn` – international standard serial number
- `doi` – digital object identifier

7.1.10 `documents`

Tabulka `documents` slouží pro uchovávání textů z přiložených dokumentů a to za účelem fulltextového vyhledávání. Jako jediná tabulka není typu InnoDB ale MyISAM, kvůli fulltextovým indexům.

7.1.11 `format`

Tabulka `format` slouží pro uchovávání Latte šablon k formátování citací.

Změny:

- Původní Smarty šablony byly nahrazeny za Latte šablony.

7.1.12 `general_settings`

Tabulka `general_settings` slouží pro uchovávání defaultního nastavení aplikace.

Změny:

- Tabulka je přidána zcela nově.

7.1.13 `group`

Tabulka `group` slouží pro uchovávání uživatelských skupin.

Změny:

- Tabulka je přidána zcela nově.

7.1.14 `group_has_publication`

Tabulka `group_has_publication` slouží jako pomocná tabulka pro vztah M:N mezi `group` a `publication`.

7.1.15 `journal`

Tabulka `journal` slouží pro uchovávání časopisů. Pokud je publikace typu `article`, odkazuje atribut `journal_id` u `publication` právě na tuto tabulku.

Změny:

- Opraven problém s formátem dat ve sloupci `issue_date`.

- Přidány atributy:
 - `abbreviation` – zkratka časopisu.
 - `doi` – digital object identifier.

7.1.16 `publication`

Tabulka `publication` slouží, jak již název napovídá, pro uchování publikací. Struktura vychází z BibTeXu, takže jsou zde atributy pro všech 13 typů publikací 9.2.7.1.

Změny:

- Opraven problém s formátem dat ve sloupci `issue_date`.
- Přidán atribut `doi` – digital object identifier.

7.1.17 `publisher`

Tabulka `publisher` slouží pro uchovávání vydavatelů.

7.1.18 `retrieve`

Tabulka `retrieve` slouží pro uchovávání tokenu, který se používá při resetování hesla.

7.1.19 `submitter`

Tabulka `submitter` slouží pro uchovávání uživatelů.

Změny:

- Navýšena velikost sloupce `pass`. Data (hashe hesel) z této tabulky byly lépe zabezpečeny (funkce SHA v kombinaci se saltem).

7.1.20 `submitter_has_group`

Tabulka `submitter_has_group` slouží jako pomocná tabulka pro vztah M:N mezi `group` a `submitter`.

Změny:

- Tabulka je přidána zcela nově.

7.1.21 `submitter_has_publication`

Tabulka `submitter_has_publication` slouží jako pomocná tabulka pro vztah M:N mezi `publication` a `submitter`. Pokud si uživatel přidá publikaci do svých oblíbených, vytvoří se záznam právě v této tabulce.

7.1.22 `user_settings`

Tabulka `user_settings` slouží pro uchovávání uživatelských nastavení.

Změny:

- Tabulka je přidána zcela nově.

Uživatelské rozhraní a vzhled

V této kapitole rozeberu návrh uživatelského rozhraní aplikace. Zmíním kroky, kterými jsem docílil výsledné podoby aplikace a pak v přehledné tabulce srovnám prohlížeče, které aplikace podporuje 8.1.

8.1 Návrh UI (uživatelského rozhraní)

UI aplikace bylo vytvářeno úplně od začátku. Odrazový můstkem pro mě byla původní aplikace, kterou jsem zanalyzoval, vyhledal hlavní problémy a při návrhu nové aplikace se jim snažil vyvarovat.

Při návrhu jsem se snažil využívat standardizované grafické prvky frameworku Bootstrap a UI tedy působí čistě, vzdušně a sjednoceně.

Aplikace je díky použití Bootstrapu optimalizovaná pro všechny běžné desktopové a mobilní prohlížeče včetně jejich starších verzí – například Internet Explorer je podporován ve verzích 8–11.

Layout je postaven na grid systému (systém mřížek) [52], který rozděljuje layout aplikace do několika řádků (row) a jednotlivé řádky pak do 12 vertikální oddílů. Každý oddíl může mít odlišnou velikost – takže v jednom řádku může být například 6 oddílů o velikosti 2 oddílů. Tímto způsobem vzniká nespočet kombinací různých velikostí, součet musí být však vždy 12.

V případě změny velikosti okna prohlížeče (mobilní telefony, tablety apod.) se oddíly různě transformují – mění svou velikost a řadí se pod sebe. Uživatelé je tedy pokaždé zobrazen obsah stránky v nejlepší možné podobě [53].

Tabulka 8.1: Bootstrap: Podpora prohlížečů (převzato z [53]).

	Chrome	Firefox	Internet Explorer	Opera
Android	Podporuje.	Podporuje.	Neexistuje.	Nepodporuje.

Pokračování na další stránce

Tabulka 8.1 – Pokračování z předešlé stránky

	Chrome	Firefox	Internet Explorer	Opera
iOS	Podporuje.	Neexistuje.	Neexistuje.	Nepodporuje.
Mac OS X	Podporuje.	Podporuje.	Neexistuje.	Podporuje.
Windows	Podporuje.	Podporuje.	Podporuje.	Podporuje.
Linux	Oficiálně nepodporuje, nicméně vše funguje.	Oficiálně nepodporuje, nicméně vše funguje.	Oficiálně nepodporuje, nicméně vše funguje.	Oficiálně nepodporuje, nicméně vše funguje.

Samotný návrh probíhal v následujících krocích:

1. Analýza cílů a požadavků: požadavky jsou sepsány v sekci 1.
2. Analýza UI a funkcionality původní aplikace: určení problémů a návrhu, jak se jím v nové aplikaci vyhnout 3.8.
3. Zjištění a stanovení cílové skupiny, její způsob uvažování a zvyky: cílová skupina je zmíněna v sekci 3.3.
4. Návrh scénářů: pomocí Use Case diagramů, které jsou v sekci 6.2.
5. Návrh drátěných modelů (wireframe).
 - Drátěné modely jsem vytvářel v komerčním programu WireframeSketcher, jelikož je mimo jiné nabízen formou doplňku do IDE Eclipse, se kterým mám zkušenosti. UI aplikace je intuitivní a aplikace má v základu hotovy všechny důležité části webového UI, takže není potřeba nic nového modelovat. Příklady modelů jsou vidět na obrázcích 8.18.28.3.
6. Vytvoření funkčního prototypu.
 - Prototyp byl vytvořen pro část aplikace. Při návrhu jsem se snažil využívat standardní grafické prvky frameworku Bootstrap, kterých nabízí opravdu hodně – tlačítka, progress bary, ikony, formulářová pole apod. Díky tomu UI působí čistě, vzdušně a sjednoceně.
 - Při analýze se ukázalo, že většina UI prvků v původní aplikaci je navrhnutá naprosto špatně. Nebudu zde znovu rozepisovat všechny změny, kterými UI prošlo, protože vše je uvedeno v části 3.8. Rozepíšu tedy jen hlavní změny.

Analýza ukázala, že by bylo výhodné zobrazovat veškeré formuláře v modálních oknech, protože nemusíte přesměrovat uživatele na novou stránku s formulářem a poté zase zpět na stránku původní. Vše je sice implementačně složitější – odesílání dat z formuláře a poté překreslení pozmeněné části stránky musí být řešeno AJAXem na pozadí, nicméně pro uživatele je vše daleko plynulejší a příjemnější. Při analýze stávajících systémů, se moje myšlenka potvrdila, protože většina aplikací pro správu publikací má formuláře řešeny právě také pomocí modálních oken.

- Dále se také ukázalo, že by bylo vhodné použít tematické ikony pro tlačítka vyvolávající následující akce:
 - Přidej novou položku.
 - Edituj položku (zobraz editační formulář).
 - Smaž položku.
 - Zobraz asociované záznamy.
 - Přidej do mých publikací.
 - Přidej do mých skupin.
 - Odeber z mých publikací.
 - Odeber z mých skupin.
 - Stáhni příložený soubor.
- Tato tlačítka byla taktéž barevně odlišena, takže uživatel na první pohled pozná, co dané tlačítko dělá:
 - Zelená barva – přidej novou položku.
 - Modrá barva – edituj položku.
 - Červená barva – smaž položku.
 - Žlutá barva – přidej do mých publikací/skupin.
 - Defaultní bílá barva – zruš změny, zobraz asociované záznamy, stáhni soubor, zobraz nápovědu.
 - Veškeré položky v horním menu byly v původní aplikaci vysázeny vedle sebe, což zabíralo mnoho místa. V nové aplikaci jsem je přesunul do rozbalovacího menu, které nejen šetří místo, ale hlavně logicky položky seskupuje.

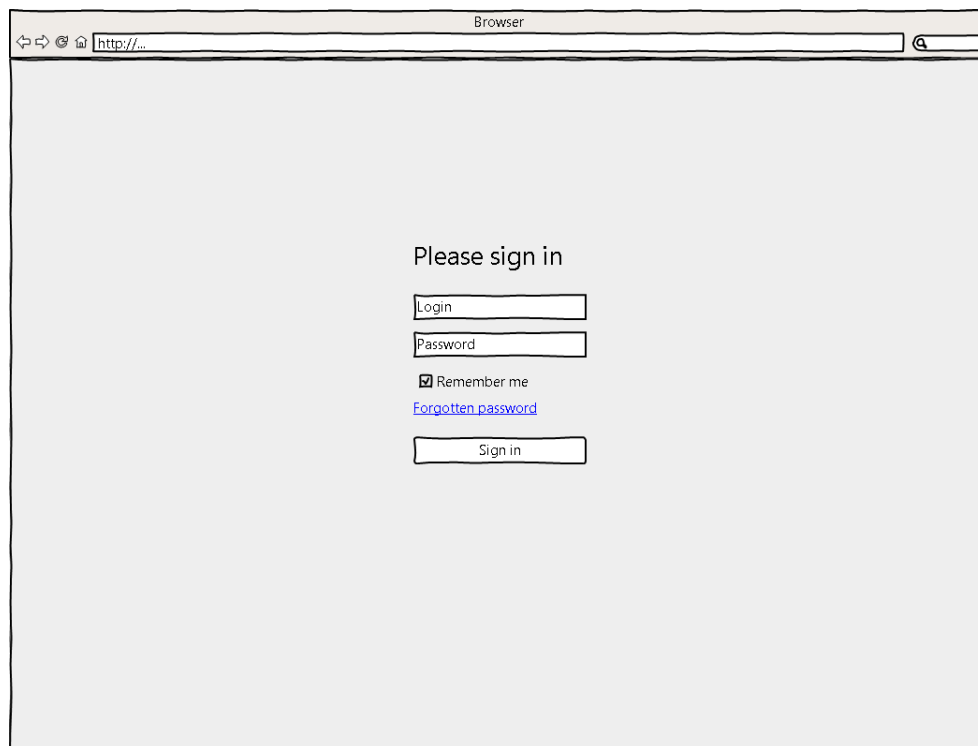
7. Vytvoření funkční aplikace.

- Pro finální aplikaci jsem využil hotové šablony z prototypu, které jsem však upravil, aby vyhovovaly finální aplikaci.

Při návrhu UI jsem prováděl Nielsonovu heuristickou analýzu, která se opírá o 10 pravidel použitelnosti [54]:

1. Viditelnost stavu systému.

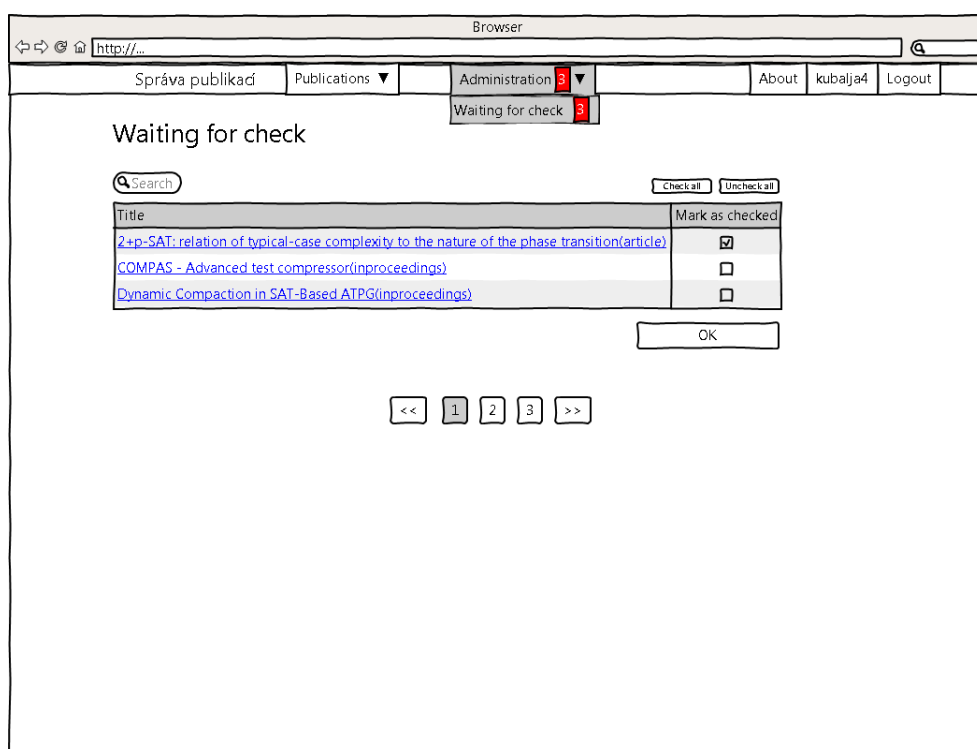
8. UŽIVATELSKÉ ROZHŘANÍ A VZHLED



Obrázek 8.1: Drátěný model – login.

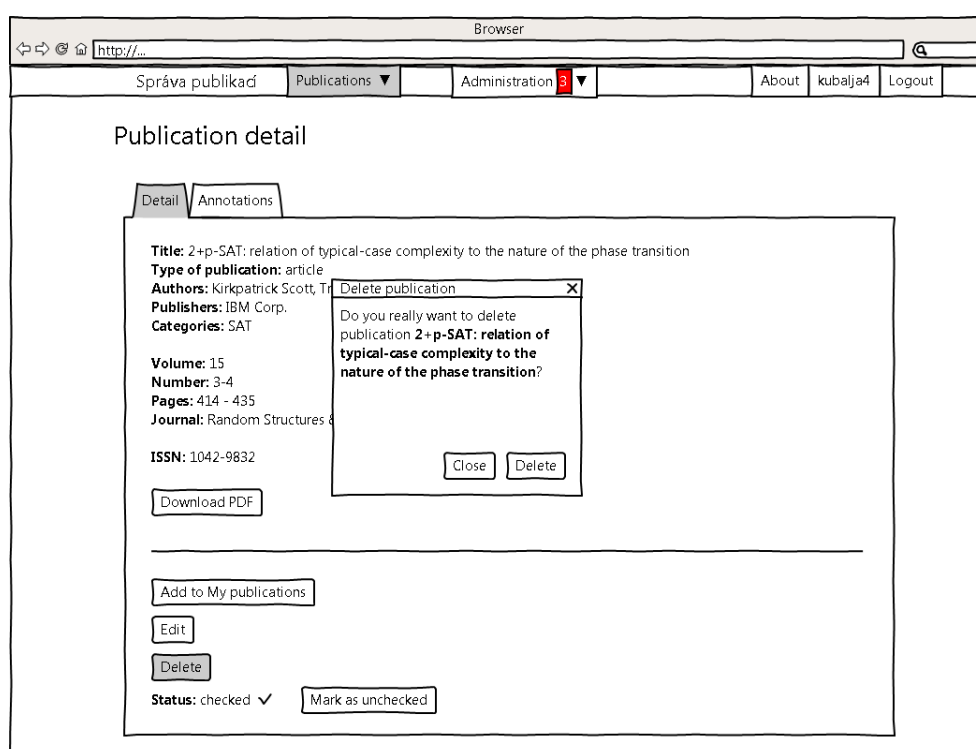
2. Propojení systému a reálného světa.
3. Uživatelská kontrola a svoboda.
4. Standardizace a konzistence.
5. Prevence chyb.
6. Rozpoznání namísto vzpomínání.
7. Flexibilní a efektivní použití.
8. Estetický a minimalistický.
9. Pomoc uživatelů pochopit, poznat a vzpamatovat se z chyb.
10. Náповěda a návod.

8.1. Návrh UI (uživatelského rozhraní)



Obrázek 8.2: Drátěný model – seznamy – schvalování publikací.

8. UŽIVATELSKÉ ROZHRANÍ A VZHLED



Obrázek 8.3: Drátěný model – detail publikace.

Realizace

V této kapitole je popsána implementace aplikace. Nejprve zmíním strukturu aplikace, poté rozeberu jednotlivé komponenty, popíšu funkci jednotlivých presenterů, akcí, signálů a nakonec ukážu nestandardní implementační řešení.

9.1 Struktura aplikace

Aplikace dodržuje strukturu Nette Frameworku, který je, jak již bylo zmíněno v části 5.3.5.1, postaven na MVC architektuře. Struktura aplikace je vidět níže na schématu 9.1.

```

www/.....kořenový adresář webu
├── DP/ ..... adresář se spustitelnou formou implementace
│   ├── app/ ..... adresář webové aplikace
│   │   ├── components/ ..... komponenty
│   │   ├── config/ ..... konfigurační soubory
│   │   ├── forms/ ..... formuláře
│   │   ├── helpers/ ..... pomocné třídy
│   │   ├── model/ ..... třídy modelové vrstvy
│   │   ├── presenters/ ..... třídy presenterů
│   │   ├── templates/ ..... šablony
│   │   ├── router/ ..... konfigurace URL adres
│   │   └── bootstrap.php/ ..... spouštěcí soubor
│   └── vendor/ ..... knihovny pro aplikaci
│       ├── nette/ ..... Nette fw
│       ├── log/ ..... chybové logy
│       ├── temp/ ..... místo pro dočasné soubory (cache, sessiony, atd.)
│       ├── test/ ..... adresář pro unit testy
│       └── www/ ..... místní kořenový adresář webu – pouze tento adresář je
│           přístupný z internetu

```

9.1.1 Modely

Modely se skládají z několika tříd, kde každá obstarává část logiky aplikace. Většina jich názvem odpovídá názvům tabulek v databázi: `Annotation`, `AttribStorage`, `Attributes`, `Author`, `AuthorHasPublication`, `Categories`, `CategoriesHasPublication`, `Conference2`, `ConferenceYear`, `Documents`, `Format`, `GeneralSettings`, `GroupHasPublication`, `Journal`, `Publication`, `Publisher`, `Retrieve`, `Submitter`, `SubmitterHasGroup`, `UserSettings`, `SubmitterHasPublication`, `Group`.

V Každé modelové třídě jsou metody, které nad danou tabulkou operují: získávají a formátují data. Nachází se ve složce `www/DP/app/model` jak je vidět na schématu 9.1.

9.1.2 Controllery

Controllery neboli presentery, jak již bylo zmíněno 5.3.5.1, zpracovávají požadavky uživatele a podle nich provádí změny v modelu a v pohledu. V aplikaci se nachází 17 presenterů: `AdminPresenter`, `GroupPresenter`, `AuthorPresenter`, `BasePresenter`, `ConferencePresenter`, `UserPresenter`, `ErrorPresenter`, `JournalPresenter`, `SecuredPresenter`, `RepairPresenter`, `SignPresenter`, `CategoriesPresenter`, `PublicationPresenter`, `PublisherPresenter`, `FormatPresenter`, `AttributesPresenter` a `HomepagePresenter`.

Diagram znázorňující životní cyklus presenteru je vidět na obrázku 9.1. V každém presenteru je několik druhů metod [55], které bych chtěl níže rozebrat.

9.1.2.1 `startup()`

Metoda, která inicializuje proměnné nebo ověří uživatelská oprávnění. Volá se po vytvoření presenteru.

9.1.2.2 `action<Action>()`

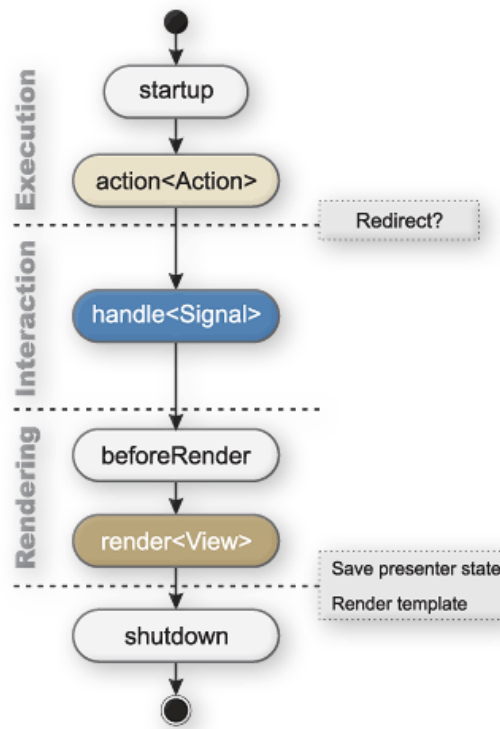
Metoda, která zpracovává požadavky od uživatele, dává příkazy modelu (CRUD) a poté předává data pro vykreslení metodě `render<View>()` (není však podmínkou).

9.1.2.3 `handle<Signal>()`

Metoda, která má za úkol zpracovávat tzv. subrequesty. Signály rozumíme akce, které neovlivňují view.

9.1.2.4 `beforeRender()`

Metoda `beforeRender()`, je volána před metodou `render<View>()` a nejčastěji obsahuje nastavení šablony.



Obrázek 9.1: Diagram znázorňující životní cyklus presenteru (převzato z [55]).

9.1.2.5 `render<View>()`

Metoda, která předává data do šablony.

9.1.2.6 `shutdown()`

Metoda, která je volána při ukončení životního cyklu presenteru.

9.1.3 Pohledy

Pohledy mají za úkol vykreslit a naformátovat data získaná od presenteru. Nachází se ve složce `twww/DP/app/templates/<nazevPresenteru>` jak je vidět na schématu 9.1. Jejich název, respektive název souboru odpovídá názvu akce. Veškerý kód je psán pomocí šablonovacího jazyka Latte.

9.1.4 Formuláře

Formuláře jsou od aplikace odděleny a nachází ve složce `www/DP/app/forms`. Pro přidání i editaci prvku je vždy použit stejný formulář.

9.2 Implementace

V této podkapitole bych chtěl popsat nestandardní implementační řešení. Nebudu se tedy zabývat úplně všemi částmi aplikace.

9.2.1 Seznamy

Veškerou editaci a mazání jednotlivých položek v seznamech jsem se rozhodl implementovat pomocí AJAXu na pozadí, protože tato změna není natolik důležitá, aby muselo dojít k přesměrování. Existují dva způsoby, jak implementaci provést.

Nejjednodušší způsob je AJAXově znovu překreslit všechny položky, protože člověk se nemusí starat o to, který řádek konkrétně překreslit. Toto řešení je však neefektivní, protože se zbytečně načítají znovu všechna data z databáze.

Nejelegantnější řešení tedy je překreslit pouze editovaný/smazaný záznam (řádek). Pro tento účel jsou v Nette dynamické snippety.

Dynamickými snippety je myšleno použití makra `snippet` v šabloně, kdy je v názvu snippetu použita proměnná. Tyto snippety musí být obaleny statickým (obyčejným snippetem bez použití proměnné).

9.2.2 Demonstrace dynamických snippetů na seznamu časopisů

Při editaci časopisu se do šablony nepředá kolekce se všemi časopisy, ale pouze s jediným – tím, který jsme pozměnili a chceme vykreslit a odeslat v payloadu do prohlížeče. Invalidujeme však celou statickou obálku – snippet `journalShowAllRecords`. `foreach` v šabloně proběhne jen jednou a žádné snippety navíc se nevykreslí 9.1.

Metoda `publicationEditJournalFormSucceeded`, která pak zpracuje data z formuláře, uloží data do databáze a rozhodne, který řádek překreslit. Vše je vidět na příkladu 9.2.

9.2.3 Autoři

V této sekci bych chtěl rozebrat funkcionalitu týkající se autorů.

9.2.3.1 Přiřazení autorů k publikaci

Během přidávání nové publikace má mít uživatel možnost přiřadit k dané publikaci autory a to v pořadí podle významnosti. Nejvýznamnější autor (nejvyšší priorita) se má vždy nacházet na prvním místě seznamu.

Bylo mi jasné, že toto nepůjde vyřešit pouze za použití jazyka PHP a budu muset sáhnout po nějakém JavaScriptovém doplňku. Nabízela se taktéž možnost, že bych si tento doplněk napsal sám. Usoudil jsem však, že by mi

Listing 9.1: Seznam časopisů – šablona (jedná se o zkrácený kus zdrojového kódu vyňatého z aplikace).

```
<table class="table table-striped">
<tbody n:snippet="journalShowAllRecords">
  {foreach $records as $record}
<tr n:snippet="record-{$record->id}">
  <td>{$record->name}</td>
  <td>{$record->abbreviation}</td>
  <td>{$record->issn}</td>
  <td>{$record->doi}</td>
  <td>
  <button type="button" id="deleteJournalButton-{$record->id}"
  </button>
  <button type="button" id="editJournalButton-{$record->id}"
  </button>
  <button type="button" id="relatedJournalButton-{$record->id}"
  </button>
  </td>
</tr>
  {/foreach}
</tbody>
</table>
```

to zabralo spoustu času a stejně by se to funkcemi nemohlo rovnat doplňkům hotovým, které vyvíjejí velké týmy programátorů.

Nalezl jsem sadu 40 doplňků JQWidgets, které zmiňuji v části 5.4.3. Přesně pro tento účel se mi hodil doplněk jqxListBox. Doplněk podporuje drag&drop výběr prvků, umí pracovat s JSON, XML, filtrovat, bindovat akce na klávesy apod. JqxListBox jsem tedy nasadil a otestoval. Během testování jsem však odhalil několik chyb:

- Filtrování sice fungovalo, pokud jsem však chtěl přesunout vyfiltrované autory přes tlačítko (na které byla nabindovaná akce přesuň) a nikoliv pomocí drag&drop, tak se nepřesunovali vyfiltrování autoři, ale autoři naprosto náhodní.
- Nefungovaly metody:

```
– jqxListBox('getSelectedItem');
– jqxListBox('clearSelection');
```

Výše zmíněné problémy jsem nahlásil a programátoři z JQWidgets je během několika dnů opravili. Nyní tedy vše funguje jak má – autory lze filtrovat, přesouvat a následně řadit jak myši (drag&drop), tak za pomoci tlačítek.

Listing 9.2: Seznam časopisů – metoda, která zpracovává data z editačního formuláře (jedná se o zkrácený kus zdrojového kódu vyňatého z aplikace).

```
public function publicationEditJournalFormSucceeded($form) {
// ziskej data z~formulare
$formValues = $form->getValues();
$formValues['submitter_id'] = $this->user->id;
$this->drawAllowed = false;
// updatni zaznam v~databazi
$this->context->Journal->update($formValues);
$this->template->journalEdited = true;
if ($this->name == "Publication") {
// pokud je presenter Publication, nahraj pozmenena data do selectboxu
$this->journals = $this->context->Journal->getJournalsNames();
$this["publicationAddNewForm"]["journal_id"]->setItems($this->journals);
$this["publicationAddNewForm"]["journal_id"]->setDefaultValue($form->values->id);
$this->template->journalInfo = $this->context->Journal->find($form->values->id);
} elseif ($this->name == "Journal") {
// pokud je presenter Journal, obnov konkretni radek v~tabulce
$this->template->records = array();
$this->template->records[$form->values->id] =
$this->context->Journal->find($form->values->id);
}
// vypis zpravu o~vysledku akce
$this->flashMessage('Operation has been completed successfully.', 'alert-success');
if (!$this->presenter->isAjax()) {
// pokud není ajax, redirectni na tu samou stranku
$this->presenter->redirect('this');
} else {
// pokud je ajax, prekresli snippety
$form->setValues(array(), TRUE);
$this->redrawControl('publicationEditJournalForm');
$this->redrawControl('flashMessages');
if ($this->name == "Publication") {
$this->redrawControl('publicationAddNewForm-journal_id');
$this->redrawControl('publicationJournalInfo');
} elseif ($this->name == "Journal") {
$this->redrawControl('journalShowAllRecords');
}
}}
```

9.2.3.2 Duplicita autorů

Jedním z požadavků bylo vyřešit duplicitu autorů. Původní aplikace trpěla nedostatkem, že při přidávání nového autora do databáze, nedokázala rozlišit, zda se přidávaný autor v databázi již nenachází. Autor přidávaný do databáze může mít totiž více tvarů:

- Martin Luther King.
- M. L. King.
- Martin L. King.

Tuto funkcionalitu jsem tedy do aplikace implementoval, takže při přidávání nového autora se aplikace podívá do databáze a pokusí se vyhledat všechny možné tvary zadaného autora. Pokud některý z tvarů odpovídá záznamu v databázi, aplikace tento tvar vypíše uživateli na obrazovku včetně varovné hlášky, zda se nejedná o stejného autora.

Publications Groups Administration

Add new Author

Fill form correctly!

- Alfke P.
- Similiar author has been found, check his name above. If it is not him, press DONE button again.

Name*

Middlename

Surname*

Similiar author has been found, check his name above. If it is not him, press DONE button again.

Close Done

Obrázek 9.2: Nová aplikace – validátor duplicity autorů.

V případě že se jedná o stejného autora, uživatel může přidávání zrušit kliknutím na tlačítko „close“, v opačném případě potvrdí přidání toho autora tlačítkem „done“. Vše je vidět na náhledu 9.2.

9.2.4 Kategorie

Další funkcionalita, kterou bylo potřeba naimplementovat, byl strom kategorií. V původní aplikaci byly všechny kategorie na stejné hierarchické úrovni a nešel tvořit strom.

Potřeboval jsem tedy takový doplněk, který podporuje stromovou strukturu a zároveň funguje jako checkbox. V sadě JQWidgets mým potřebám naprosto vyhovoval doplněk jqxTree, takže jsem ho použil. jqxTree umožňuje tvořit stromovou strukturu, umí načítat data z JSON, XML, bindovat akce na klávesy apod.

Dále byly potřeba udělat změny v databázi. Upravil jsem tedy tabulku `categories` a to tak, aby mohla odkazovat sama na sebe a tím šly tvořit stromy kategorií (potomci, rodiče). Přidal jsem sloupec `categories_id` 7.1.6, který odkazuje sám na sebe (jedná se o referenci).

V případě, že záznam je typu podkategorie, má v tomto sloupci uložené ID rodiče. Pokud se jedná o kategorii (nemá rodiče) je hodnota sloupce nastavena na NULL.

9.2.5 Přidávání souborů

Každá publikace by měla mít možnost přiložení několika dokumentů současně, proto bylo potřeba implementovat vícenásobné uploadování souborů. Pro tento účel jsem našel perfektní doplněk přímo pro Nette Framework s názvem MultiFileUpload.

Doplněk podporuje vícenásobné drag&drop přidávání souborů a je postaven na AJAXu s využitím HTML5.

9.2.6 Formát citací

Zadavatel si může také přidat vlastní formát citací, který naformátuje metadata dané publikace. Vše se dělá jednoduše přes administrační rozhraní.

Jediným předpokladem pro vytváření vlastních formátů je, že zadavatel zná šablonovací jazyk Latte. V původní aplikaci byl využit šablonovací jazyk Smarty, ze kterého ale Latte vychází. Vše funguje tak, že zadavatel vytváří Latte šablonu, která se ukládá namísto do souboru, do databáze. Celá logika převodu šablony uložené v databázi do formátu vhodného pro vykreslení je vidět na příkladu 9.3. V aplikaci se nyní nachází tři formáty normovaných citací:

- ČSN ISO 690.
- IEEE.
- APA.

Nebudu tyto formáty podrobně rozebírat, jelikož byly rozebrány v předchozích pracích – od pana Martina Vodičky [1][3] a slečny Kateřiny Pospíšilové [2].

9.2.7 Import

Další funkcionalitu, kterou bylo potřeba implementovat, jsou importy citací z formátů BibTeX, EndNote a RefWorks. Jedná se o velmi zajímavou funkci, jelikož dokáže uživateli značně usnadnit přidávání publikace do systému. Uživatel je odstíněn od kroku, kdy by musel sám kopírovat veškeré atributy do konkrétních formulářových políček.

Při přidávání publikace 6.2.11 uživatel klikne na tlačítko „Import publication“, vyplní příslušná políčka a naimportuje publikaci. Aplikace předvyplní políčka a uživatel pouze dovyplní části (pokud je třeba), které import neobsahoval nebo nezvládnul. Celý proces je popsán v UC14 6.2.14

V následujících částech bych chtěl formáty rozebrat a uvést jejich příklady.

Listing 9.3: Formát citací – logika převodu šablony (jedná se o zkrácený kus zdrojového kódu vyňatého z aplikace).

```

public function renderShowPub() {
    $_this = $this;
    // zaregistruj helper
    $this->template->registerHelper('template', function($text) use ($_this) {
        $template = new Nette\Templating\Template();
        $template->control = $template->_control = $_this;
        $template->presenter = $template->_presenter = $this->getPresenter(FALSE);
        $template->registerFilter(new Nette\Latte\Engine);
        // dodane promenne
        $template->user = $this->user;
        $template->baseUri = $this->template->baseUri;
        $template->basePath = $this->template->basePath;
        $template->pubCit = $this->template->pubCit;
        $template->pubCit['author_array'] = $this->template->pubCit['author_array'];
        $template->pubCit['author'] = $this->template->pubCit['author'];
        // flash message
        $presenter = $this->getPresenter(FALSE);
        if ($presenter->hasFlashSession()) {
            $id = $this->getParameterId('flash');
            $template->flashes = $presenter->getFlashSession()->$id;
        } else {
            $template->flashes = array();
            $template->setSource($text);
            $txt = $template->__toString();
            return $txt;
        }
    });
    // predej data do template
    $this->template->formats = $this->context->Format->findAll();
}

```

9.2.7.1 BibTeX

BibTeX formát 4.1.1 rozlišuje 13 typů publikací (položek), přičemž každá je tvořena poli [4]. V závislosti na typu publikace jsou některá pole povinná a některá nepovinná. Zde bych chtěl uvést seznam všech typu publikací včetně jejich povinných a nepovinných polí. Po něm bude následovat seznam standardních polí a jejich popis, nakonec pak příklad BibTeX položky.

Seznam položek

@article

- Popis: článek z novin nebo časopisu/magazínu.
- Povinná pole: author, title, journal, year, volume.
- Nepovinná pole: number, pages, month, note, key.

@book

- Popis: kniha s jednoznačně identifikovatelným vydavatelem.
- Povinná pole: author/editor, title, publisher, year.

- Nepovinná pole: volume/number, series, address, edition, month, note, key.

@booklet

- Popis: publikace, která je vytisknutá a svázaná, ale bez uvedeného vydavatele nebo sponzora.
- Povinná pole: title.
- Nepovinná pole: author, howpublished, address, month, year, note, key.

@conference

- Popis: zápis z konference či z jednání.

@inbook

- Popis: část knihy bez vlastního názvu – například kapitola, sekce nebo text určený rozsahem stran.
- Povinná pole: author/editor, title, chapter/pages, publisher, year.
- Nepovinná pole: volume/number, series, type, address, edition, month, note, key.

@incollection

- Popis: část knihy s vlastním názvem.
- Povinná pole: author, title, booktitle, publisher, year.
- Nepovinná pole: editor, volume/number, series, type, chapter, pages, address, edition, month, note, key.

@inproceedings

- Popis: článek z konference.
- Povinná pole: author, title, booktitle, year.
- Nepovinná pole: editor, volume/number, series, pages, address, month, organization, publisher, note, key.

@manual

- Popis: manuál/technická dokumentace.
- Povinná pole: title.

- Nepovinná pole: author, organization, address, edition, month, year, note, key.

@mastersthesis

- Popis: diplomová práce.
- Povinná pole: author, title, school, year.
- Nepovinná pole: type, address, month, note, key.

@misc

- Popis: nezařaditelná položka.
- Povinná pole: nejsou.
- Nepovinná pole: author, title, howpublished, month, year, note, key.

@phdthesis

- Popis: disertační práce.
- Povinná pole: author, title, school, year.
- Nepovinná pole: type, address, month, note, key.

@proceedings

- Popis: sborník článků z konference.
- Povinná pole: title, year.
- Nepovinná pole: editor, volume/number, series, address, month, publisher, organization, note, key.

@techreport

- Popis: report vydaný školou nebo jinou institucí.
- Povinná pole: author, title, institution, year.
- Nepovinná pole: type, number, address, month, note, key.

@unpublished

- Popis: dokument, u kterého je uveden autor a název, formálně však není vydaný.
- Povinná pole: author, title, note.
- Nepovinná pole: month, year, key.

Standardní pole

address: Adresa vydavatele nebo jiného typu instituce. U větších vydavatelů se vynechává.

annotate: Anotace, pro anotované bibliografické styly.

author: Jména autorů.

booktitle: Název knihy – část, ze které se cituje.

chapter: Kapitola či sekce.

crossref: Klíč položky v databázi, na kterou se křížově odkazuje.

edition: Vydání knihy. Tato položka by měla být tvořena řadovou číslovkou nebo slovně psanou řadovou číslovkou s prvním velkým písmenem, například „Druhé“.

editor: Jméno editora.

howpublished: Popis, jak bylo nějaké dílo zvláště publikováno. První slovo by mělo být psáno velkými písmeny.

institution: Název sponzorské instituce technického reportu.

journal: Název časopisu/magazínu.

key: Klíč, který se používá pro řazení a křížové odkazování.

month: Měsíc publikace – pokud není publikace vydána, tak se jedná o měsíc vzniku.

note: Poznámka, která může pomoci čtenáři. První slovo by mělo být psáno velkými písmeny.

number: Číslo časopisu/magazínu nebo technické reportu v sérii. Časopis je často identifikován svazkem a číslem. Většina publikací nemá pole „number“, nýbrž „volume“.

organization: Organizace, která sponzoruje konferenci nebo vydává manuál.

pages: Číslo stránek nebo počet stránek. Číslo mohou být psána výčtem (1,2,3) nebo rozsahem (45–99).

publisher: Jméno vydavatele.

school: Název školy, ve které byla napsána práce (diplomová, disertační, ...).

series: Jména sérií nebo souborů knih. Pokud citujeme celou knihu, pole title udává název knihy a volitelné pole series udává jména sérií, ve kterých je kniha publikována.

title: Název práce.

type: Typ technické reportu – například výzkumný report.

volume: Svazek časopisů nebo knih.

year: Rok vydání publikace – pro nevydanou práci, rok kdy byla napsána. Měl by se skládat ze čtyř číslic.

Ostatní pole

URL: URL adresa publikace.

ISBN: Mezinárodní identifikační číslo knihy (The International Standard Book Number).

ISSN: Mezinárodní identifikační číslo časopisů (The International Standard Serial Number).

LCCN: The Library of Congress Call Number.

abstract: Abstrakt práce.

keywords: Klíčová slova používána pro vyhledávání.

price: Cena dokumentu.

copyright: Informace o copyrightu.

language: Jazyk dokumentu.

contents: Tabulka obsahu.

Položka v BibTeX formátu Položka v BibTeX formátu je vidět na příkladu 9.4. Struktura položky byla rozebrána podrobně v předchozích pracích [1][3][2], proto se v této práci na BibTeX formát nebudu soustředit.

9.2.7.2 EndNote

EndNote formát 4.2.1 rozlišuje 51 typů publikací (položek) [56]. Aplikace však pracuje s 13 typy (vychází z BibTeXu), proto jsou tyto odlišné EndNote typy sloučeny s těmi z BibTeXu.

Nejprve bych chtěl uvést přehled všech typů, které EndNote rozlišuje, poté seznam polí (tagů) a nakonec příklad položky.

Listing 9.4: Příklad položky v BibTeX formátu

```
@Book{hicks2001 ,
  author   = "Selman, Bart and Monasson, Remi",
  title    = "Design of a Carbon Fiber Composite Grid Structure
             for the GLAST Spacecraft Using a
             Novel Manufacturing Technique",
  publisher = "Stanford Press",
  year     = 2001,
  address  = "Palo Alto",
  edition  = "1st",
  isbn     = "0-69-697269-4"
}
```

Typy publikací

%0 Aggregated Database: sloučeno s *@misc*

%0 Ancient Text: sloučeno s *@misc*

%0 Artwork: sloučeno s *@misc*

%0 Audiovisual Material: sloučeno s *@misc*

%0 Bill: sloučeno s *@misc*

%0 Blog: sloučeno s *@misc*

%0 Book: sloučeno s *@book*

%0 Book Section: sloučeno s *@inbook*

%0 Case: sloučeno s *@misc*

%0 Catalog: sloučeno s *@misc*

%0 Classical Work: sloučeno s *@misc*

%0 Computer program: sloučeno s *@misc*

%0 Conference Paper: sloučeno s *@inproceedings*

%0 Conference Proceedings: sloučeno s *@proceedings*

%0 Dataset: sloučeno s *@misc*

%0 Dictionary: sloučeno s *@book*

%0 Edited Book: sloučeno s *@book*

%0 Electronic Article: sloučeno s *@misc*

%0 Electronic Book: sloučeno s *@book*

%0 Electronic Book Section: sloučeno s *@inbook*

%0 Encyklopedie: sloučeno s *@misc*

%0 Equation: sloučeno s *@misc*

%0 Figure: sloučeno s *@misc*

%0 Film od Broadcast

%0 Generic: sloučeno s *@misc*

%0 Government Document: sloučeno s *@misc*

%0 Grant: sloučeno s *@misc*

%0 Hearing: sloučeno s *@misc*

%0 Chart or Table: sloučeno s *@misc*

%0 Interview: sloučeno s *@misc*

%0 Journal Article: sloučeno s *@article*

%0 Legal Rule or Regulation: sloučeno s *@misc*

%0 Magazine Article: sloučeno s *@article*

%0 Manuscript: sloučeno s *@manual*

%0 Map: sloučeno s *@misc*

%0 Music: sloučeno s *@misc*

%0 Newspaper Article: sloučeno s *@article*

%0 Online Database: sloučeno s *@misc*

%0 Online Multimedia: sloučeno s *@misc*

%0 Pamhlet: sloučeno s *@booklet*

%0 Patent: sloučeno s *@misc*

%0 Personal Communication: sloučeno s *@misc*

%0 Podcast: sloučeno s *@misc*

%0 Press Release: sloučeno s *@misc*

%0 Report: sloučeno s *@techreport*

%0 Serial: sloučeno s *@misc*

%0 Standard: sloučeno s *@misc*

%0 Statute: sloučeno s *@misc*

%0 Thesis: sloučeno s *@mastersthesis* nebo s *@phdthesis*

%0 Unpublished Work: sloučeno s *@unpublished*

%0 Web Page: sloučeno s *@misc*

Seznam tagů V tabulce 9.1 je vidět přehled všech EndNote tagů, které se běžně používají.

Tabulka 9.1: EndNote tagy.

Tag	Název pole v EndNote
%A	Author
%B	Secondary Title (of a Book or Conference Name)
%C	Place Published
%D	Year
%E	Editor /Secondary Author
%F	Label
%G	Language
%H	Translated Author
%I	Publisher
%J	Secondary Title (Journal Name)
%K	Keywords
%L	Call Number
%M	Accession Number
%N	Number (Issue)
%P	Pages
%Q	Translated Title
%R	Electronic Resource Number
%S	Tertiary Title
%T	Title
%U	URL
%V	Volume
%W	Database Provider
%X	Abstract
%Y	Tertiary Author
%Z	Notes
%0	Reference Type

Pokračování na další stránce

Tabulka 9.1 – Pokračování z předešlé stránky

Tag	Název pole v EndNote
%1	Custom 1
%2	Custom 2
%3	Custom 3
%4	Custom 4
%6	Number of Volumes
%7	Edition
%8	Date
%9	Type of Work
%?	Subsidiary Author
%@	ISBN/ISSN
%!	Short Title
%#	Custom 5
%%\$	Custom 6
%]	Custom 7
%&	Section
%(Original Publication
%)	Reprint Edition
%*	Reviewed Item
%+	Author Address
%^	Caption
%>	Link to PDF
%<	Research Notes
%[Access Date
%=	Last Modified Date
%	Name of Database

Položka v EndNote formátu Položka v EndNote formátu je vidět na příkladu 9.5. Každý tag, který symbolizuje nějaký typ pole se skládá ze dvou znaků. Prvním znakem je pokaždé % a poté následuje znak znázorňující druh pole. Za tagem je poté mezera a následuje samotná hodnota.

Z příkladu je také zřejmé, že se každé pole odděluje novým řádkem. Někdy se oddělují novým řádkem i samotné hodnoty (pokud jich je více) toho samého pole a to tak, že se vždy na nový řádek zopakuje tag následovaný mezerou a danou hodnotou (autoři).

Významnější autor je uveden dříve.

Listing 9.5: Příklad položky v EndNote formátu

```
%0 Conference Paper
%1 843625
%A Sybille Hellebrand
%A Hua-Guo Liang
%A Hans-Joachim Wunderlich
%T A MIXED MODE BIST SCHEME BASED ON RESEEDING..
%B Proceedings of the 2000 IEEE International Test Conference
%@ 0-7803-6546-1
%P 778
%D 2000
%I IEEE Computer Society
```

9.2.7.3 RefWorks

RefWorks formát 4.2.2 rozlišuje 31 typů publikací (položek) [57]. Aplikace však pracuje s 13 typy (vychází z BibTeXu), proto jsou stejně jako v případě EndNote formátu, tyto odlišné typy sloučeny s těmi z BibTeX.

Nejprve bych chtěl uvést přehled všech typů, které RefWorks rozlišuje, poté seznam polí (tagů) a nakonec příklad položky.

Typy publikací RT tag udávající typ publikace se musí nacházet na prvním místě položky.

RT Abstract: sloučeno s *@misc*

RT Artwork: sloučeno s *@misc*

RT Bills/Resolutions: sloučeno s *@misc*

RT Book, Section: sloučeno s *inbook*

RT Book, Edited: sloučeno s *@book*

RT Book, Whole: sloučeno s *@book*

RT Case/Court Decisions: sloučeno s *@misc*

RT Computer Program: sloučeno s *@misc*

RT Conference Proceedings: sloučeno s *@misc*

RT Dissertation/Thesis: sloučeno s *@mastersthesis* nebo s *@phdthesis*

RT Dissertation/Thesis, Unpublished: sloučeno s *@unpublished*

RT Generic: sloučeno s *@misc*

- RT Grant:** sloučeno s @misc
- RT Hearing:** sloučeno s @misc
- RT Journal Article:** sloučeno s @article
- RT Journal, Electronic:** sloučeno s @misc
- RT Laws/Statutes:** sloučeno s @misc
- RT Magazine Article:** sloučeno s @article
- RT Map:** sloučeno s @misc
- RT Monograph:** sloučeno s @misc
- RT Motion Picture:** sloučeno s @misc
- RT Music Score:** sloučeno s @misc
- RT Newspaper Article:** sloučeno s @article
- RT Online Discussion Forum:** sloučeno s @misc
- RT Patent:** sloučeno s @misc
- RT Personal Communication:** sloučeno s @misc
- RT Report:** sloučeno s @techreport
- RT Sound Recording:** sloučeno s @misc
- RT Unpublished Material:** sloučeno s @unpublished
- RT Video/ DVD:** sloučeno s @misc
- RT Web Page:** sloučeno s @misc

Seznam tagů V tabulce 9.2 je vidět přehled všech RefWorks tagů, které se běžně používají.

Tabulka 9.2: RefWorks tagy.

Tag	Název pole v RefWorks
Tag	Název pole v RefWorks programu
RT	Reference Type
SR	Source Type (field is either Print(0) or Electronic(1))
ID	Reference Identifier
A1	Primary Authors

Pokračování na další stránce

Tabulka 9.2 – Pokračování z předešlé stránky

Tag	Název pole v RefWorks
T1	Primary Title
JF	Periodical Full
JO	Periodical Abbrev
YR	Publication Year
FD	Publication Data, Free Form
VO	Volume
IS	Issue
SP	Start Page
OP	Other Pages
K1	Keyword
AB	Abstract
NO	Notes
A2	Secondary Authors
T2	Secondary Title
ED	Edition
PB	Publisher
PP	Place of Publication
A3	Tertiary Authors
A4	Quaternary Authors
A5	Quinary Authors
T3	Tertiary Title
SN	ISSN/ISBN
AV	Availability
AD	Author Address
AN	Accession Number
LA	Language
CL	Classification
SF	Subfile/Database
OT	Original Foreign Title
LK	Links
DO	Digital Object Identifier
CN	Call Number
DB	Database
DS	Data Source
IP	Identifying Phrase
RD	Retrieved Date
ST	Shortened Title
U1	User 1
U2	User 2

Pokračování na další stránce

Tabulka 9.2 – Pokračování z předešlé stránky

Tag	Název pole v RefWorks
U3	User 3
U4	User 4
U5	User 5
U6	User 6
U7	User 7
U8	User 8
U9	User 9
U10	User 10
U11	User 11
U12	User 12
U13	User 13
U14	User 14
U15	User 15
UL	URL
SL	Sponsoring Library
LL	Sponsoring Library Location
CR	Cited References
WT	Website Title
A6	Website editors
WV	Website version
WP	Date of Electronic Publication
OL	Output Language (see codes for specific languages below)
PMID	PMID
PMCID	PMCID

Položka v RefWorks formátu Položka v RefWorks formátu je vidět na příkladu 9.6. Každý tag, který symbolizuje nějaký typ pole se skládá ze dvou znaků. Prvním znakem je pokaždé velké písmeno a poté následuje buď druhé velké písmeno, nebo číslo. Za tagem je poté mezera a následuje samotná hodnota.

Z příkladu je také zřejmé, že se každé pole odděluje novým řádkem stejně jako u EndNote formátu. Někdy se oddělují novým řádkem i samotné hodnoty toho samého pole (pokud jich je více) a to tak, že se vždy na nový řádek zopakuje tag následovaný mezerou a danou hodnotou (autoři, klíčová slova).

Významnější autor je uveden dříve a jeho jméno by mělo být v následujícím tvaru: příjmení, jméno (nebo iniciála jména), iniciála druhého jména.

Listing 9.6: Příklad položky v RefWorks formátu

```
RT Journal Article
SR Electronic(1)
ID 271
A1 Allan ,Steven
A1 Gilbert ,Paul
T1 Anger and anger expression in relation to perceptions of social rank
JF Personality & Individual Differences
YR 2002
FD Feb
VO 32
IS 3
SP 551
OP 565
K1 Anger
K1 Self Report
K1 Status
K1 Depression (Emotion)
K1 Symptoms
K1 self-report measures
K1 anger expression
K1 social rank
K1 entrapment
K1 depressive symptoms
AB Explored the relationship between self-report \dots
PB Elsevier Science , England , [URL:http:// www.elsevier.nl]
SN 0191-8869
AD Kingsway Hosp, Dept of Clinical Psychology , Derby , UK;
AN 2002-00282-017
LA English
CL 3120 Personality Traits & Processes
SF Print (Paper); Journal Article; Empirical Study
LK http:// bmj.com/content/vol325/issue7371/twib.shtml#325/7371/0
OL English (30)
```


9.2.8 Uživatelské skupiny

Do aplikace jsem také implementoval uživatelské skupiny, které může uživatel (zadavatel, administrátor) přidávat, editovat a mazat 6.2.17.

Při přidávání nebo editování publikace uživatel zvolí, do kterých skupin publikace patří 6.2.11. Poté si přidá samotné skupiny do svých skupin „My Groups“ 6.2.18 a díky tomu se mu publikace v „My Groups“ budou zobrazovat.

9.2.9 Springer API

Springer je jeden z největších vydavatelů vědeckých knih a časopisů na světě. Vydává téměř 500 vědeckých a odborných časopisů a je součástí skupiny Springer Science+Business Media a působí ve 20 zemích v Evropě, USA a Asii [58].

Springer nabízí čtyři druhy API, které podporují řadu operací a rozdílných návratových formátů 9.3. Jedná se o následující API:

- Springer Metadata API – poskytuje metadata online dokumentů (články z časopisů, kapitoly knih, protokoly), kterých je více jak 7 milionů.
- Springer Meta API – poskytuje novou verzi metadat více než 7 milionů online dokumentů (články z časopisů, knih, kapitoly knih, protokoly).
- Springer Open Access API – poskytuje metadata, obsah, obrázky pro více než 80,000 volně dostupných článků z časopisů BioMed Central a SpringerOpen.
- Integro API – poskytuje prostředky pro přístup k informacím (název, vydavatel, DOI, datum) o časopisech, které patří Springer.

Springer API podporuje RESTful dotazy, jejichž URI musí pokaždé obsahovat následující části [59]:

- Kolekci, nad kterou je dotaz proveden.
- Návratový formát.

Další parametr URI je volitelný a jedná se o metodu 9.4, která je nad kolekcí provedena. Následuje pak volitelná hodnota metody.

Poslední částí URI je API klíč, který uživatel obdrží po zaregistrování aplikace na stránkách <https://dev.springer.com>. Obdržený API klíč je pak potřeba uvést při každém dotazu.

Obecný RESTful dotaz:

- `http://api.springer.com/{kolekce}/{návratový_formát}/{metoda}/{parametr_metody}?api_key=api_klíč_aplikace`

Uživatel může tvořit složitější dotazy kombinací různých parametrů. Omezil jsem se pouze na základní – vyhledávání pomocí třech metod `doi`, `issn`, `isbn`, které jsou vidět v tabulce 9.4.

Dotaz v aplikaci provádí PHP knihovna `cURL`, která je pro tento účel navrhnutá. Celá aplikační logika se nachází v modelu `Springer` ve funkci s názvem `fetchData`. Tato metoda tvoří samotný dotaz podle vstupních parametrů a poté volá funkci `curlDownload`, která pracuje s knihovnou `cURL` a vykonává stažení obsahu.

Návratový formát jsem zvolil `JSON`, protože je v `PHP` snadno zpracovatelný. Funkce `json_decode` převede získaná `JSON` data do objektu, takže se s nimi dobře pracuje.

Získané položky (`data`) jsou poté zobrazeny uživateli a ten si vybere, kterou položku chce naimportovat. Pokud vše proběhne v pořádku, je uživatelem vybraná položka rozparsována a přednastavena do políček formuláře pro přidávání nové publikace. Celý proces je popsán v UC15 6.2.15.

Konkrétní RESTful dotaz:

- `http://api.springer.com/metadata/json/doi/10.1007/11527695_15?api_key=318191e1dc228af0c36214a7b7aee84e`

Výsledek dotazu ve formátu `JSON` je vidět na příkladu 9.7.

Tabulka 9.3: Springer API: Přehled kolekcí a návratových formátů (převzato z [59]).

Kolekce	Popis kolekce	Návratový formát	Popis návratového formátu
meta versioned	Ukládá články z časopisů a kapitoly knih v nové verzi.	pam	Navrací XML, kde je každý záznam ve formátu PRISM Aggregator Message.
		json	Navrací JSON.
		jsonp	Navrací JSON v „závorce“.
metadata	Ukládá články z časopisů a kapitoly knih.	pam	Navrací XML, kde je každý záznam ve formátu PRISM Aggregator Message.
		json	Navrací JSON.
		jsonp	Navrací JSON v „závorce“.

Pokračování na další stránce

Tabulka 9.3 – Pokračování z předešlé stránky

Kolekce	Popis kolekce	Návratový formát	Popis návratového formátu
openaccess	Poskytuje volně dostupný obsah.	app	Navrací XML ve formátu Springer A++.
		json	Navrací JSON.
		jsonp	Navrací JSON v „závorce“.
integro	Ukládá Springer časopisy.	xml	Navrací XML.

Tabulka 9.4: Springer API: Přehled metod (převzato z [59]).

Metoda	Popis metody
doi	Metoda, která nalezne článek s daným DOI.
issn	Metoda, která vyhledá množinu článků v časopise s daným ISSN.
isbn	Metoda, která vyhledá množinu kapitol knihy s daným ISBN.

9.2.10 Globalní nastavení aplikace

V průběhu implementace DP, jsem narazil na několik případů, kdy by se hodilo, aby uživatel měl možnost měnit globální nastavení aplikace. V původní aplikaci bylo vše uloženo v konfiguračním souboru, který mohl měnit pouze uživatel s přístupem k souborům aplikace. Napadlo mě, že by vše šlo řešit daleko elegantněji – a to ukládat nastavení, která jsou potřeba čas od času změnit, do databáze. K tomuto nastavení by měl přístup jen administrátor a to skrze webové rozhraní.

Přidal jsem tedy novou tabulku `general_settings` do databáze a vytvořil editační formulář, který je přístupný pouze pro administrátory a nachází se v menu pod názvem „General Settings“. Administrátor může měnit:

- Defaultní hodnotu počtu položek na stránku, která se přednastaví každému nově přidanému uživateli. Ten si tuto hodnotu může poté změnit v uživatelském nastavení.
- Springer API klíč aplikace.

Listing 9.7: Springer API – výsledek dotazu ve formátu JSON.

```
{
  "query": "doi:10.1007/11527695_15",
  "apiKey": "318191e1dc228af0c36214a7b7aee84e",
  "result": [
    {
      "total": "1",
      "start": "1",
      "pageLength": "10"
    }
  ],
  "records": [
    {
      "identifier": "doi:10.1007/11527695_15",
      "url": [
        {
          "format": "",
          "platform": "",
          "value": "http://dx.doi.org/10.1007/11527695_15"
        }
      ],
      "title": "Clause Form Conversions for Boolean Circuits",
      "creators": [
        {
          "creator": "Jackson, Paul"
        },
        {
          "creator": "Sheridan, Daniel"
        }
      ],
      "publicationName": "Theory and Applications of Satisfiability Testing",
      "openaccess": "false",
      "doi": "10.1007/11527695_15",
      "printIsbn": "978-3-540-27829-0",
      "electronicIsbn": "978-3-540-31580-3",
      "isbn": "978-3-540-27829-0",
      "publisher": "Springer",
      "publicationDate": "2005-01-01",
      "volume": "",
      "number": "",
      "startingPage": "",
      "copyright": "2005 Springer-Verlag Berlin Heidelberg",
      "genre": "OriginalPaper",
      "abstract": "The Boolean circuits is well established"
    }
  ],
  "facets": [ \dots ]
}
```

9.2.11 Uživatelské nastavení

Možnost každého uživatele přizpůsobit si chování nebyla původně v plánu, avšak ukázala se v průběhu vývoje jako zcela nezbytná, protože bylo potřeba, aby si každý uživatel mohl měnit v seznamech počet položek na stránku.

V databázi se tedy nachází nová tabulka `user_settings` a uživatel si může počet položek měnit v nastavení svého účtu v záložce „User Settings“.

9.2.12 Fulltextové vyhledávání

V aplikaci je taktéž realizováno fulltextové vyhledávání, které principiálně funguje stejně jako v původní aplikaci. Na hlavní stránce se nachází dvě záložky – „Fulltext search“ a „Authors/Publication search“, vyhledávací pole a pokročilé nastavení vyhledávání.

Pokročilé nastavení vyhledávání je defaultně skryté a hlavní stránka tedy působí velmi čistým dojmem. Pokud uživatel chce toto nastavení aktivovat, klikne na tlačítko Enable/Disable advanced search. Poté vybere buď operátor OR, nebo AND a nakonec kategorie publikací. Operátory uživatel specifikuje, zda vyhledané publikace mají patřit do všech kategorií zároveň, nebo alespoň do jedné.

Pokud má uživatel vybranou záložku „Authors/Publication search“, vyhledává aplikace klíčová slova v názvech publikací a jménech autorů a poté zobrazí seznam výsledků.

Pokud má však uživatel vybranou záložku „Fulltext search“, vyhledává aplikace klíčová slova v přiložených dokumentech, přesněji řečeno ve vyextrahovaných textech uložených v databázi a poté zobrazí seznam výsledků.

Celý proces je popsán v UC2 6.2.2.

9.2.12.1 Extrakce textu

Při ukládání nové publikace se provádí extrakce textu z přiložených dokumentů. Vyextrahovaný text se ukládá do databáze do tabulky `documents`.

Extrakci z PDF dokumentů obstarává knihovna Smalot PDF Parser, která staví na TCPDF parseru. Knihovna je poskytována zdarma pod GNU licencí.

9.3 Zabezpečení

V následujících několika částech bych chtěl rozebrat ACL, poté uvést přehled nejčastějších metod útoků a nakonec zmínit, jak je proti nim aplikace zabezpečena.

9.3.1 Úvod

Zabezpečení webových aplikací je v dnešní době jednou z nejdůležitějších částí vývojového cyklu. Stále totiž existují jednoduché metody, které vedou k naru-

Listing 9.8: Nette – příklad šablony.

```
<p onclick="alert({ $journal})">{ $journal}</p>
<script>var journal = { $journal};</script>
```

šení jejich aplikační bezpečnosti. Ve většině případů k tomu stačí přístup na úrovni běžného uživatele webu.

9.3.2 Access control list (ACL)

ACL je v překladu „seznam pro řízení přístupu“ a jedná se o seznam oprávnění, kde je uvedeno kdo, nebo co, má povolení přistupovat k objektu (akci, stránce) [60].

Původní aplikace nic podobného neměla 3.6 a tím pádem mohl uživatel s jakoukoliv rolí provádět veškeré akce, které jsou povolené pouze pro role s vyšším oprávněním.

Nette disponuje třídou `Nette\Security\Permission`, která slouží jako autorizátor a zároveň poskytuje programátorovi ACL vrstvu pro řízení práv a přístupů. Programátor si jednoduše definuje role, zdroje a jednotlivá oprávnění.

V diplomové práci se ACL logika nachází v modelové třídě `Ac1`, která dědí od `Nette\Security\Permission` [61]. Na začátku třídy jsou definovány role příkazem `$this->addRole('role');` a poté následuje seznam všech zdrojů `$this->addResource('zdroj');` (presenterů a akcí). Nakonec jsou vždy ke každé roli přiřazena určitá oprávnění (přístup ke zdroji) a to pomocí funkce `$this->allow('role', 'zdroj');`.

9.3.3 Nejčastější typy útoků

9.3.3.1 Cross-Site Scripting (XSS)

Cross-Site Scripting je metoda narušení webových stránek, která využívá bezpečnostní chyby ve skriptech a to převážně neošetřené vstupy. Útočníkovi pak nečiní problém podstrčit do stránek svůj vlastní kód a tím stránku pozměnit, či získat citlivé údaje [62].

Jediná funkční ochrana proti tomuto útoku je ošetřením všech řetězců. Je tedy potřeba „escapovat“ vypisovaná data, tj. převod znaků majících v daném kontextu speciální význam na jiné odpovídající sekvence [55].

V případě, že kodér na „escapování“ zapomene, vznikne v aplikaci bezpečnostní díra. Latte disponuje technologií Context-Aware Escaping, která kodérovi šetří práci, protože vše automaticky „escapuje“.

Příklad šablony 9.8 a poté výstupu šablony 9.9. Předpokládáme, že proměnná `$journal` obsahuje řetězec `'Amarcord & 8 1/2'`.

Listing 9.9: Nette – výstup šablony.

```
<p onclick="alert('Amarcord & 8 1\2&quot;');">
Amarcord & 8 1/2
</p>
<script>var journal = "Amarcord & 8 1\2";</script>
```

Listing 9.10: Nette – příklad parametrizovaného dotazu.

```
$table->where("field > ?", $val);
```

9.3.3.2 Cross-Site Request Forgery (CSRF)

Cross-Site Request Forgery je metoda útoku do internetových stránek, která spočívá v tom, že přiměje uživatele navštívit stránku napadané aplikace, která provádí nějakou akci, o které uživatel nemá tušení. Nejčastěji zneužívá toho, že je uživatel zrovna přihlášen do aplikace typu wiki, blog, fórum [63], kde útočník předem zná, nebo dokáže odhadnout URL adresy vedoucí například k editaci článku.

Proti útoku se lze bránit generováním a ověřováním autorizačního tokenu [64]. V Nette lze ochránit formuláře příkazem `$form->addProtection()`; Veškeré formuláře v mé aplikaci jsou tedy chráněny tímto příkazem.

9.3.3.3 SQL Injection

SQL Injection je metoda útoku na databázovou vrstvu programu a to vsunutím kódu přes neošetřený vstup. Tento škodlivý kód je vlastně pozměněný SQL dotaz, kterým útočník může získat důležitá data z databáze, jako jsou hesla apod. [65].

Zastaralý způsob ochrany je „escapování“ potenciálně nebezpečných znaků, který však není 100%. Nejlepší ochrana proti tomuto útoku je vyhnout se vkládání proměnných do dotazů a namísto toho používat parametrizované dotazy tzv. „Prepared Statements“.

Parametrizovaným dotazem rozumíme takový dotaz, kde namísto proměnné dáme zástupný znak. Databáze si pak sama dosadí do těchto znaků hodnoty proměnných.

V Nette se jako zástupné znaky používají otazníky, do kterých pak databázová vrstva Nette database hodnoty proměnných dosadí. Vše je vidět na příkladu 9.10.

Testování

V této kapitole bych chtěl rozepsat testy, kterými byla moje aplikace podrobena.

10.1 Úvod

Testování softwaru je velice důležitou součástí jeho vývoje, jelikož vede ke zjištění chyb, které by jinak negativně ovlivňovaly jeho chod. Bezesporu je potřeba software testovat již v průběhu vývoje, protože čím dříve se na chyby přijde, tím snáze a levněji jdou odstranit. Testování můžeme rozdělit na dvě základní kategorie – black box a white box.

10.2 White box

Při použití testů white box jsou zdrojové soubory aplikace k dispozici a je tedy známá její struktura [66]. Tester testuje aplikaci právě na základě zdrojových souborů.

10.2.1 Developer testing

Developer testing, neboli testy programátorem se řadí mezi základní testy, které by měly být provedeny po dokončení určité části kódu. Tyto testy spočívají v tom, že programátor ihned otestuje kód, který napsal a program je tedy otestován na nejnižší možné úrovni.

Běžně jsou tyto testy opomíjeny a přitom oprava chyb v této části testování je nejméně nákladná. Ze své zkušenosti pak mohu říci, že jsem se setkal s případem, kdy tato metodika testování byla vynechána a programátor odevzdal svůj neotestovaný zkompileovaný kód testerům k další fázi testování. Testerům se aplikaci nepodařilo ani spustit, takže ji museli ihned vracet zpět programátorovi, což všechny naprosto zbytečně časově zdrželo.

Tato metodika testování byla aplikována během celého vývoje aplikace a díky ní se podařilo odhalit valnou většinu chyb.

10.3 Black box

Při použití black box testů se na software nahlíží jako na černou skříňku [66], do které se nelze podívat (zdrojový kód není viditelný) a je pouze vidět, jak vypadá a jak se chová navenek. Není tedy zřejmé, jak pracuje a můžeme pouze sledovat, jaký výsledek navrácí pro vložení vstupních dat.

10.3.1 Compatibility testing

Aplikace byla otestována z hlediska kompatibility na posledních verzích nej-používanějších webových prohlížečů a to jak desktopových, tak mobilních.

10.3.1.1 Desktopové prohlížeče

Pro testování na desktopu byly vybrány následující prohlížeče:

- Google Chrome.
- Mozilla Firefox.
- Internet Explorer.

Byly provedeny všechny UC scénáře sepsané v kapitole 6. Aplikace se ve všech případech zobrazovala a fungovala zcela korektně.

10.3.1.2 Mobilní prohlížeče

Pro testování na chytrém mobilním telefonu byly vybrány následující prohlížeče:

- Google Chrome.
- Dolphin browser.

Stejně jako v případě testu kompatibility na desktopových webových prohlížečích, byly provedeny ty samé testy i na mobilních webových prohlížečích. Během testu byly objeveny drobné chyby týkající se špatně napozicovaných popisků u formulářových políček.

10.3.1.3 Vyhodnocení

Testy ukázaly drobné chyby týkající se zobrazování na mobilních prohlížečích. Veškeré chyby byly opraveny a aplikace je tedy kompatibilní se všemi výše zmíněnými webovými prohlížeči.

10.3.2 User acceptance testing

User acceptance testing neboli akceptační testy jsou testy prováděny na straně zákazníka [67]. Vykonnávají se zpravidla na samotný závěr testování. Zákazníky jsou v tomto případě studenti a učitelé z ČVUT.

Pokud chceme mít výsledky testů relevantní, je potřeba aplikaci otestovat více testery. Konkrétní počet testerů není nikterak standardizován, ale podle experta na testování Jakoba Nielsena je ideální počet testerů 5 [68]. Oslovil jsem dohromady tedy pět testerů – tři studenty vyšších ročníků a dva pedagogy, kteří odpovídali cílovým zákazníkům. Cílový zákazník je akademický pracovník, který používá původní nebo obdobnou aplikaci pro správu publikací a tedy rozumí dané problematice.

10.3.2.1 Příprava

Aplikace byla naplněna reálnými daty a byla nasazena na webový server. Dostupná byla na adrese <http://dp.jankubalek.cz/>. Všem zúčastněným jsem zaslal níže rozepsané instrukce.

- Odkaz na aplikaci.
- Informace, na kterém prohlížeči aplikaci testovat.
- Přihlašovací údaje do všech třech rolí (čtenář, zadavatel, administrátor).
- Jednotlivé scénáře včetně sady závěrečných otázek.

10.3.2.2 Průběh

Uživatelé si otevřeli webový prohlížeč a zadali URL adresu, kterou obdrželi v instrukcích. Poté se přihlásili do aplikace a postupovali podle jednotlivých scénářů.

10.3.2.3 Scénáře a závěrečné otázky

Scénáře a sada závěrečných otázek vycházely z případů užití. Před každým scénářem bylo uživateli sděleno, do jaké role se má přihlásit. Na závěr testování pak následovala sada následujících otázek:

- Rychlost provádění jednotlivých scénářů – které scénáře zabraly nejvíce času a proč?
- Působí rozmístění prvků aplikace (formulářová pole, tlačítka, menu, ...) přehledně?
- Dělal všechny akce to, co měly?
- Věděl uživatel pokaždé vše, co se po něm chce?

- Jsou chybové hlášky srozumitelné?

10.3.2.4 Vyhodnocení

Výsledky akceptačních testů ukázaly, že některé části aplikace byly pro uživatele lehce matoucí – konkrétně proces přidávání nové publikace, který zabral uživatelům, zcela logicky, nejvíce času.

Uživatel ne vždy věděl, v jakém formátu má zadávat hodnoty do formulářových políček. Z tohoto důvodu byla podrobněji rozepsána nápověda včetně příkladu správných formátů.

Dále byly vylepšeny validátory, které upozorňují na konkrétní problém týkající se nesprávně vyplněného políčka.

Na žádost uživatelů byl přidán doplněk pro výběr data, který proces vyplňování také usnadňuje. Uživatel nyní nemusí myslet na dodržování správného formátu data, pouze myší vybere konkrétní datum z nabídky a doplněk se o naformátování postará sám.

Závěr

Cílem diplomové práce bylo zanalyzovat původní aplikaci pro správu publikací, navrhnout změny a rozhodnout, zda má v původní aplikaci smysl pokračovat, nebo nikoliv. Poté změny zapracovat a aplikaci na závěr otestovat. Tímto byly veškeré požadavky naplněny.

V následující sekci bych chtěl shrnout všechny důležité části vývojového cyklu mé diplomové práce.

Shrnutí všech částí

Cíl práce

V první kapitole *Cíl práce 1* byly stanoveny cíle a požadavky na funkcionalitu aplikace.

Zadání práce

V druhé kapitole *Zadání práce 2* byly podrobně rozebrány požadavky na funkcionalitu, bylo stanoveno zadání aplikace a zdefinovány pojmy.

Analýza původní aplikace

V třetí kapitole *Analýza původní aplikace 3* byl proveden rozbor původní aplikace, který ukázal, že ve stávající aplikaci nemá smysl pokračovat, protože je špatně navržena, obsahuje řadu chyb (ve funkcionalitě i bezpečnosti) a je naprosto nevyhovující dnešním požadavkům na moderní webovou aplikaci.

Přehled stávajících řešení

Ve čtvrté kapitole *Přehled stávajících řešení 4* byl proveden rozbor stávajících řešení a to jak „open-source“, tak komerčních. V tomto srovnání exceloval systém Mendeley z hlediska počtu pokročilých funkcí a systém Papers z hlediska

přehledného, intuitivního UI. Dále bylo provedeno srovnání systémů na základě pokročilých funkcí a požadavků na funkcionalitu. Tato srovnání ukázala, že stávající systémy nesplňují mnohé základní požadavky na funkcionalitu.

Výběr implementačních technologií

V páté kapitole *Výběr implementačních technologií* 5 byla provedena analýza všech možných technologií používaných pro implementaci webových aplikací. Analýza ukázala, že pro implementaci server side části aplikace bude vhodné použít skriptovací jazyk PHP v kombinaci s frameworkem Nette.

Jakožto server-side programovací jazyk byl pak zvolen JavaScript. Dále byl proveden výběr JavaScriptového frameworku (jQuery) a dalších doplňků, které převážně pocházely z rodiny doplňků JQWidgets.

Na konci této kapitoly bylo provedeno srovnání CSS mobile first frameworků, ze kterého vzešel jako vítěz Bootstrap, který byl poté využit i v mojí aplikaci.

Use cases

V šesté kapitole *Use cases* 6 byl proveden rozbor uživatelských rolí včetně případů užití, které byly stanoveny na základě požadavků na funkcionalitu.

Databáze

V sedmé kapitole *Databáze* 7 byla popsána databáze a zmíněn seznam provedených změn.

Uživatelské rozhraní a vzhled

V osmé kapitole *Uživatelské rozhraní a vzhled* 8 jsem se zaměřil na popis layoutu stránek aplikace, následovaný pak popisem uživatelského rozhraní. Na samotný závěr pak byly sepsány kroky, které vedly k finální vizuální podobě aplikace.

Realizace

V deváté kapitole *Realizace* 9 byla popsána struktura aplikace, stěžejní pilíře Nette Frameworku a nakonec byly rozebrány nejzajímavější části aplikace – Springer API, problematika týkající se autorů, kategorií, přidávání souborů a formátů citací.

Dále pak byl popsán import ze známých referenčních formátů, uživatelské skupiny, seznamy, globální nastavení aplikace, uživatelské nastavení a fulltextové vyhledávání. V závěru této kapitoly jsem rozepsal zabezpečení aplikace – konkrétně ACL a ochranu proti metodám útoků XSS, CSRF a SQL Injection.

Testování

V poslední desáté kapitole *Testování* 10 jsem se zaměřil na testování aplikace. Aplikace byla testována již v průběhu vývoje a pak po samotném dokončení. Pro testování byly použity metodiky Developer testing, Compatibility testing a User acceptance testing. Při testování byly objeveny některé nedostatky, které byly samozřejmě odstraněny.

Přínos práce

Tato aplikace je určitě přínosem pro všechny uživatele původní aplikace, kteří dlouhá léta museli pracovat s tímto neintuitivním programem plným chyb. Moje aplikace tedy vyřešila veškeré problémy původní aplikace a reakce uživatelů jsou na ni velice kladné. Všichni oceňují především kompletně předělané UI, které nyní působí svěžím i přehledným dojmem a bezproblémový chod aplikace včetně všech inovací, které jsem do aplikace zanesl.

Aplikace měla samozřejmě i přínos pro mě, protože jsem si vyzkoušel celý vývojový cyklus aplikace a díky tomu jsem si prošel všemi rolmi – od návrháře UI, přes programátora až po testera. Díky implementaci jsem se pak setkal se zajímavými technologiemi, které určitě využiji i na svých budoucích projektech.

Srovnání s konkurencí

Pokud srovnám aplikaci s konkurenčními produkty, tak mohu hrdě prohlásit, že konkurenci v několika směrech převyšuje – především co se jednoduchosti aplikace a intuitivnosti UI týče. Hlavním problémem těchto konkurenčních produktů je fakt, že nesplňují požadavky na funkcionalitu zmíněné v kapitole 2. Některé konkurenční produkty (Mendeley) sice disponují větším množstvím pokročilé funkcionality, nicméně tato „extra“ funkcionalita je pro cílovou skupinu uživatelů zbytečná.

Možná rozšíření

Aplikace by šla určitě rozšířit o sociální funkce jako například: tvorba týmů, zasílání zpráv, veřejné akademické profily apod.

V současné době na rozšíření mé aplikace pracuje již pan Petr Svoboda, který na ní staví svou bakalářskou práci. Pan Svoboda provedl analýzu mé práce, při které zjistil, že aplikace obsahuje část funkcionality, kterou by využil ve své práci, a proto nedávalo smysl, aby tuto společnou funkcionalitu znovu programoval. Moje diplomová práce tedy bude rozšířena o funkcionalitu týkající se pokročilé správy konferencí a jejich ročníků – rozšířené filtrování, správa přidružených workshopů, kategorizace konferencí, možnost archivace

ZÁVĚR

konferencí, indikace indexace sborníku ročníků konference ve veřejných databázích publikací apod.

Rozšířena bude i databáze aplikace. Do ostrého provozu bude poté nasazena tato rozšířená práce pana Svobody, která je vlastně nadmnožinou práce mojí.

Literatura

- [1] Vodička, M.: Databáze publikací. 2006, bakalářská práce.
- [2] Pospíšilová, K.: Rozšířená databáze publikací II. 2011, bakalářská práce.
- [3] Vodička, M.: Rozšířená databáze publikací. 2009, diplomová práce.
- [4] Universität Bremen: *BibTeX*. [cit. 2015-03-09]. Dostupné z: <http://www.fb10.uni-bremen.de/anglistik/langpro/bibliographies/jacobsen-bibtex.html>
- [5] Zotero: Changelog. [cit. 2015-01-30]. Dostupné z: https://www.zotero.org/support/1.0_changelog
- [6] Zotero: *Zotero: Documentation*. [cit. 2015-01-30]. Dostupné z: https://www.zotero.org/support/quick_start_guide
- [7] Zotero: *Zotero: Style Repository*. [cit. 2015-01-30]. Dostupné z: <https://www.zotero.org/styles>
- [8] Thomson Reuters: *EndNote: Product Details*. [cit. 2015-02-06]. Dostupné z: <http://endnote.com/product-details>
- [9] Thomson Reuters: *EndNote Getting Started Guide*. [cit. 2015-02-06]. Dostupné z: <http://endnote.com/training/mats/enuserguide/eng/endnote7/enguide-full.pdf>
- [10] RefWorks-COS: *RefWorks*. [cit. 2015-02-20]. Dostupné z: <https://www.refworks.com/content/products/content.asp>
- [11] RefWorks-COS: *Příručka uživatele RefWorks 2.0*. [cit. 2015-02-20]. Dostupné z: http://www.refworks.com/refworks2/help/RefWorks_QSG_CZ_Dec11.pdf

- [12] RefWorks-COS: *RefMobile puts RefWorks wherever you are!* [cit. 2015-02-20]. Dostupné z: <https://www.refworks.com/content/products/refmobile/content.asp>
- [13] Mendeley: *Mendeley: Versions*. [cit. 2015-02-25]. Dostupné z: <http://www.mendeley.com/download-mendeley-desktop/>
- [14] Shema, H.: Interview with Dr. Victor Henning, Mendeley. Prosinec 2012, [cit. 2015-02-25]. Dostupné z: <http://blogs.scientificamerican.com/information-culture/2012/11/23/interview-with-dr-victor-henning-mendeley/>
- [15] Mendeley: Awards and Reviews. [cit. 2015-02-25]. Dostupné z: <https://www.mendeley.com/review/>
- [16] Fenner, M.: Interview with Alexander Griekspoor. Říjen 2008, [cit. 2015-02-26]. Dostupné z: http://blogs.plos.org/mfenner/2008/10/03/interview_with_alexander_griekspoor/
- [17] Papers: *Papers*. [cit. 2015-02-26]. Dostupné z: <http://www.papersapp.com/>
- [18] Nations, D.: Web Applications. Leden 2011, [cit. 2015-01-20]. Dostupné z: http://webtrends.about.com/od/webapplications/a/web_application.htm
- [19] W3Techs: Usage of server-side programming languages for websites. Březen 2015, [cit. 2015-02-01]. Dostupné z: http://w3techs.com/technologies/overview/programming_language/all
- [20] Tatroe, K.; MacIntyre, P.; Lerdorf, R.: *Programming PHP*. O'Reilly Media, třetí vydání, 2013.
- [21] Janovský, D.: Hosting s PHP. Leden 2014, [cit. 2015-03-11]. Dostupné z: <http://www.jakpsatweb.cz/katalog/hosting-php.html>
- [22] Microsoft: *ASP.NET 5: Overview*. [cit. 2015-04-07]. Dostupné z: <http://www.asp.net/vnext/overview/aspnet-vnext/aspnet-5-overview>
- [23] Bergsten, H.: *JavaServer Pages*. O'Reilly Media, třetí vydání, 2003.
- [24] SensioLabs: *Proč bych měl použít framework?* [cit. 2015-03-26]. Dostupné z: <http://symfony.com/cs/why-use-a-framework>
- [25] Zaninotto, F.: Two years of symfony. Říjen 2007, [cit. 2015-01-20]. Dostupné z: <http://www.skoleniphp.cz/skoleni-zend>

-
- [26] SensioLabs: *Symfony: The Yaml Component*. [cit. 2015-04-07]. Dostupné z: <http://symfony.com/doc/current/components/yaml/introduction.html>
- [27] SensioLabs: *Symfony: The EventDispatcher Component*. [cit. 2015-04-07]. Dostupné z: http://symfony.com/doc/current/components/event_dispatcher/introduction.html
- [28] SensioLabs: *Symfony: The DependencyInjection Component*. [cit. 2015-04-07]. Dostupné z: <http://symfony.com/doc/current/components/templating/introduction.html>
- [29] SensioLabs: *Symfony: The Templating Component*. [cit. 2015-04-07]. Dostupné z: <http://symfony.com/doc/current/components/templating/introduction.html>
- [30] SensioLabs: *Symfony: The Debug Component*. [cit. 2015-04-07]. Dostupné z: <http://symfony.com/doc/current/components/debug/introduction.html>
- [31] Zend Technologies: *Zend: About*. [cit. 2015-04-07]. Dostupné z: <http://framework.zend.com/about/>
- [32] Zend Technologies: *Zend: Programmer 's Reference Guide of Zend Framework 2*. [cit. 2015-03-15]. Dostupné z: <http://framework.zend.com/manual/current/en/index.html>
- [33] Zend Technologies: *Symfony: The Debug Component*. [cit. 2015-04-07]. Dostupné z: <http://www.knesl.com/articles/view/nette-symfony-zend-2>
- [34] Grudl, D.: *Nette 2, Symfony 2, Zend Framework 2*. 2013, [cit. 2015-01-20]. Dostupné z: <http://www.skoleniphp.cz/skoleni-zend>
- [35] Koutný, J.: *Symfony2, těší mě!* Červen 2013, [cit. 2015-01-20]. Dostupné z: <http://www.zdrojak.cz/clanky/symfony2-tesi-me/>
- [36] Nette Foundation: *Nette: Licenční politika*. [cit. 2015-02-05]. Dostupné z: <http://nette.org/cs/license>
- [37] Nette Foundation: *Nette: Zabezpečení před zranitelnostmi*. [cit. 2015-02-05]. Dostupné z: <http://doc.nette.org/cs/2.1/vulnerability-protection>
- [38] Nette Foundation: *Nette: Built with Nette*. [cit. 2015-02-05]. Dostupné z: <http://builtwith.nette.org/>

- [39] Grudl, D.: Nette Framework: MVC & MVP. Březen 2009, [cit. 2015-02-01]. Dostupné z: <http://zdrojak.root.cz/clanky/doctrine-2-uvod-do-systemu/>
- [40] Nette Foundation: *Nette: Debugování a zpracování chyb*. [cit. 2015-03-04]. Dostupné z: <http://doc.nette.org/cs/2.1/debugging>
- [41] Nette Foundation: *Nette: Addons, plugins and components*. [cit. 2015-03-04]. Dostupné z: <http://addons.nette.org/>
- [42] Nette Foundation: *Latte: amazing template engine for PHP*. [cit. 2015-03-04]. Dostupné z: <http://latte.nette.org/en/>
- [43] Flanagan, D.; Ferguson, P.: *JavaScript: The Definitive Guide*. O'Reilly & Associates, páté vydání, 2006.
- [44] Šerý, R.: JavaScript s jQuery. Květen 2007, [cit. 2015-0-26]. Dostupné z: <https://www.interval.cz/clanky/javascript-s-jquery-lehky-uvod/>
- [45] jQWidgets LTD: *jQWidgets: Products*. [cit. 2015-02-19]. Dostupné z: <http://www.jqwidgets.com/jquery-widgets-demo/>
- [46] Kůžel, F.: Chytré telefony překonaly hloupé, aplikace zvalcovaly SMS. Duben 2013, [cit. 2015-03-26]. Dostupné z: <http://www.mobilmania.cz/bleskovky/chytre-telefony-prekonaly-hloupe-aplikace-zvalcovaly-sms/sc-4-a-1323581/default.aspx>
- [47] ČTK: Z mobilů přistupuje na weby už téměř polovina uživatelů. Leden 2015, [cit. 2015-03-26]. Dostupné z: http://byznys.lidovky.cz/z-mobilu-pristupuje-na-weby-uz-temer-polovina-uzivatelu-pe2-/media.aspx?c=A150106_143037_ln-media_sk
- [48] Marcotte, E.: Responsive web design. *A List Apart*, ročník 306, 2011, [Online][cit. 2015-03-15]. Dostupné z: <http://alistapart.com/article/responsive-web-design>
- [49] ZURB: *About Foundation*. [cit. 2015-03-15]. Dostupné z: <http://foundation.zurb.com/learn/about.html>
- [50] Twitter: *About Bootstrap*. [cit. 2015-03-15]. Dostupné z: <http://getbootstrap.com/about/>
- [51] DuBois, P.: *MySQL*. Addison-Wesley Professional, páté vydání, 2013.
- [52] Twitter: *Bootstrap: CSS*. [cit. 2015-03-15]. Dostupné z: <http://getbootstrap.com/css/>

-
- [53] Twitter: *Bootstrap: Getting started*. [cit. 2015-03-15]. Dostupné z: <http://getbootstrap.com/getting-started/>
- [54] Žikovský, P.: 4. přednáška: Usability Testing with Users. Přednášející předmětu MI-NUR, 2012, [cit. 2015-01-27].
- [55] Nette Foundation: *Nette: MVC aplikace & presentery*. [cit. 2015-03-03]. Dostupné z: <http://doc.nette.org/cs/2.1/presenters>
- [56] Cardiff University: *EndNote: Tags and reference types*. [cit. 2015-02-06]. Dostupné z: http://www.cardiff.ac.uk/insrv/educationandtraining/guides/endnote/endnote_codes.html
- [57] RefWorks-COS: *RefWorks Tagged Format*. [cit. 2015-03-15]. Dostupné z: https://www.refworks.com/refworks/help/refworks_tagged_format.htm
- [58] Springer: *Springer API*. [cit. 2015-03-11]. Dostupné z: <https://dev.springer.com/>
- [59] Springer: *Springer API: RESTful Operations*. [cit. 2015-03-11]. Dostupné z: <https://dev.springer.com/restfuloperations>
- [60] Microsoft: *Managing Authorization and Access Control*. [cit. 2015-03-15]. Dostupné z: <https://technet.microsoft.com/en-us/library/bb457115.aspx>
- [61] Nette Foundation: *Nette: Přihlašování & oprávnění uživatelů*. [cit. 2015-03-30]. Dostupné z: <http://doc.nette.org/cs/2.1/access-control>
- [62] Vrána, J.: Cross Site Scripting. Listopad 2005, [cit. 2015-03-30]. Dostupné z: <http://php.vrana.cz/cross-site-scripting.php>
- [63] Auger, R.: The Cross-Site Request Forgery (CSRF/XSRF) FAQ. Duben 2010, [cit. 2015-01-20]. Dostupné z: <http://www.cgisecurity.com/csrf-faq.html>
- [64] Vrána, J.: Cross-Site Request Forgery. Duben 2006, [cit. 2015-03-30]. Dostupné z: <http://php.vrana.cz/cross-site-request-forgery.php>
- [65] SQL Injection. [cit. 2015-03-30]. Dostupné z: [https://technet.microsoft.com/en-us/library/ms161953\(v=SQL.105\).aspx](https://technet.microsoft.com/en-us/library/ms161953(v=SQL.105).aspx)
- [66] Hlava, T.: Testování bílé a černé skříňky. Srpen 2011, [cit. 2015-04-20]. Dostupné z: <http://testovanisoftwaru.cz/tag/white-box/>
- [67] Hlava, T.: Fáze a úrovně provádění testů. Srpen 2011, [cit. 2015-04-20]. Dostupné z: <http://testovanisoftwaru.cz/druhy-typy-a-kategorie-testu/faze-testu/>

LITERATURA

- [68] Nielsen, J.: Why You Only Need to Test with 5 Users. Březen 2000, [cit. 2015-01-28]. Dostupné z: <http://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/>

Seznam použitých zkratek

- ACL** Access Control List
- AJAX** Asynchronous JavaScript and XML
- AJAJ** Asynchronous JavaScript and JSON
- API** Application Programming Interface
- ASP** Active Server Pages
- BSD** Berkeley Software Distribution
- CLR** Common Language Runtime
- CSL** Citation Style Language
- CRUD** Create, Read, Update, Delete
- CSS** Cascading Style Sheets
- CSRF** Cross-site Request Forgery
- DOI** Digital Object Identifier
- DP** Diplomová práce
- DQL** Doctrine Query Language
- DRY** Don't repeat yourself
- ER** Entity-relationship (model)
- FW** Framework
- GUI** Graphical user interface
- GIF** Graphics Interchange Format

A. SEZNAM POUŽITÝCH ZKRATEK

- GPL** General Public License
- HTML** HyperText Markup Language
- HTTP** Hypertext Transfer Protocol
- IMAP** Internet Message Access Protocol
- ISBN** International Standard Book Number
- ISSN** International Standard Serial Number
- JPEG** Joint Photographic Experts Group
- JS** JavaScript
- JSP** JavaServer Pages
- JSON** JavaScript Object Notation
- KISS** Keep it short and simple
- MVC** Model-View-Controller
- MVP** Model-View-Presenter
- MySQL** My Structured Query Language
- ORM** Objektově-relační mapování
- PDF** Portable Document Format
- PHP** PHP: Hypertext preprocessor (rekurzivní zkratka)
- PNG** Portable Network Graphics
- POP3** Post Office Protocol version 3
- REST** Representational state transfer
- RSS** Rich Site Summary
- SMTP** Simple Mail Transfer Protocol
- SVG** Scalable Vector Graphics
- URL** Uniform Resource Locator
- URI** Uniform Resource Identifier
- UI** User Interface
- WYSIWYG** What You See Is What You Get editor

XHTML Extensible HyperText Markup Language

XML Extensible Markup Language

XSS Cross-site scripting

ZF Zend Framework

Databázové tabulky

Tabulka B.1: annotation

Název	Typ	Popis
id	int (10)	primární klíč
publication_id	int (10)	cizí klíč reference na tabulku publication
submitter_id	int (10)	cizí klíč reference na tabulku submitter

Tabulka B.2: attributes

Název	Typ	Popis
id	int (10)	primární klíč
submitter_id	int (10)	cizí klíč
name	varchar (50)	název
description	varchar (200)	popis
confirmed	tinyint (1)	příznak schválení (0 čeká na schválení, 1 schváleno)

Tabulka B.3: attrib_storage

Název	Typ	Popis
publication_id	int (10)	cizí klíč (reference na tabulku publication), součást primárního klíč
attributes_id	int (10)	cizí klíč (reference na tabulku attributes), součást primárního klíč
submitter_id	int (10)	cizí klíč (reference na tabulku submitter)
value	text	hodnota atributu

Tabulka B.4: author

Název	Typ	Popis
id	int (10)	primární klíč
submitter_id	int (10)	cizí klíč (reference na tabulku submitter)
name	varchar (50)	jméno autora
surname	varchar (50)	příjmení autora
middlename	varchar (50)	prostřední jméno autora

Tabulka B.5: author_has_publication

Název	Typ	Popis
author_id	int (10)	cizí klíč (reference na tabulku categories)
publication_id	int (10)	cizí klíč (reference na tabulku publication)
priority	int (10)	významnost (priorita) autora (1 nejvyšší priorita, 2 nižší priorita, ...)

Tabulka B.6: categories

Název	Typ	Popis
id	int (10)	primární klíč
submitter_id	int (10)	cizí klíč (reference na tabulku submitter)
name	varchar (50)	název
categories_id	int (10)	cizí klíč (reference na tabulku categories)

Tabulka B.7: categories_has_publication

Název	Typ	Popis
publication_id	int (10)	cizí klíč (reference na tabulku publication)
categories_id	int (10)	cizí klíč (reference na tabulku categories)

Tabulka B.8: conference2

Název	Typ	Popis
id	int (10)	primární klíč
name	varchar (500)	název konference
abbreviation	varchar (100)	zkratka názvu konference
submitter_id	int (10)	cizí klíč (reference na tabulku submitter)
description	varchar (1000)	popis
first_year	year (4)	rok (prvního ročníku konference)

Tabulka B.9: conference_year

Název	Typ	Popis
id	int (10)	primární klíč
conference2_id	int (10)	cizí klíč (reference na tabulku conference2)
submitter_id	int (10)	cizí klíč (reference na tabulku submitter)
name	varchar (500)	název ročníku konference
w_year	year (4)	rok konání
w_from	date	datum konání od
w_to	date	datum konání do
location	varchar (500)	místo konání
isbn	varchar (50)	ISBN
descriptipn	varchar (1000)	popis
isbn	varchar (50)	ISSN
doi	varchar (100)	DOI
publisher_id	int (10)	cizí klíč (reference na tabulku publisher)

B. DATABÁZOVÉ TABULKY

Tabulka B.10: documents

Název	Typ	Popis
publication_id	int (10)	primární a cizí klíč (reference na tabulku publication) zároveň
title	text	název publikace
content	text	text publikace (vyextrahovaný z přiložených dokumentů)

Tabulka B.11: format

Název	Typ	Popis
id	int (10)	primární klíč
name	varchar (50)	název citační normy
content	text	kod Latte šablony
timestamp	int (10)	časové razítko šablony

Tabulka B.12: general_settings

Název	Typ	Popis
id	int (10)	primární klíč
pagination	int (10)	defaultní stránkování (počet položek na stránku)
spring_token	varchar (250)	Springer API Key

Tabulka B.13: journal

Název	Typ	Popis
id	int (10)	primární klíč
submitter_id	int (10)	cizí klíč (reference na tabulku submitter)
name	varchar (500)	název časopisu
issn	varchar (50)	ISSN
abbreviation	varchar (100)	zkratka názvu časopisu
doi	varchar (100)	DOI

Tabulka B.14: publication

Název	Typ	Popis
id	int (10)	primární klíč
journal_id	int (10)	cizí klíč (reference na tabulku journal)
submitter_id	int (10)	cizí klíč (reference na tabulku submitter)
publisher_id	int (10)	cizí klíč (reference na tabulku publisher)
title	varchar (500)	název publikace
booktitle	varchar (500)	název svazku
abstract	text	abstrakt
confirmed	tinyint (1)	příznak schválení (0 neschváleno, 1 schváleno)
issue_date	date	datum vydání publikace
isbn	varchar (50)	ISBN
volume	varchar (50)	volume
number	varchar (50)	číslo
pages	varchar (50)	počet stran nebo rozsah stran
note	varchar (500)	poznámka
chapter	varchar (200)	kapitola
edition	varchar (200)	poznámky k vydání
editor	varchar (200)	editor
howpublished	varchar (200)	způsob publikování
institution	varchar (200)	instituce
organization	varchar (200)	organizace
school	varchar (200)	škola
address	varchar (500)	adresa
type_of_report	varchar (200)	typ reportu
url	varchar (500)	URL online dokumentu
pub_type	varchar (50)	typ publikace
conference_year_id	int (10)	cizí klíč (reference na tabulku conference_year)
doi	varchar (100)	DOI

Tabulka B.15: publisher

Název	Typ	Popis
id	int (10)	primární klíč
submitter_id	int (10)	cizí klíč (reference na tabulku submitter)
name	varchar (500)	název vydavatele
address	varchar (500)	adresa vydavatele

Tabulka B.16: retrieve

Název	Typ	Popis
submitter_id	int (10)	cizí klíč
uid_hash	varchar (50)	token (součástí URL pro reset hesla)

Tabulka B.17: submitter

Název	Typ	Popis
id	int (10)	primární klíč
name	varchar (255)	jméno uživatele
e-mail	varchar (255)	e-mail uživatele
login	varchar (50)	přihlašovací jméno
pass	varchar (255)	přihlašovací heslo (kombinace hashe a saltu)
level	int (11)	role (0 administrátor, 1 zadavatel, 2 čtenář)
surname	varchar (255)	příjmení uživatele

Tabulka B.18: submitter_has_publication

Název	Typ	Popis
submitter_id	int (10)	cizí klíč (reference na tabulku submitter)
publication_id	int (10)	cizí klíč (reference na tabulku publication)

Tabulka B.19: user_settings

Název	Typ	Popis
id	int (10)	primární klíč
submitter_id	int (10)	cizí klíč (reference na tabulku submitter)
pagination	int (10)	stránkování (počet položek na stránku)

Tabulka B.20: group

Název	Typ	Popis
id	int (10)	primární klíč
name	varchar (255)	název skupiny
submitter_id	int (10)	cizí klíč (reference na tabulku submitter)

Tabulka B.21: group_has_publication

Název	Typ	Popis
group_id	int (10)	cizí klíč (reference na tabulku group)
publication_id	int (10)	cizí klíč (reference na tabulku publication)

Tabulka B.22: submitter_has_group

Název	Typ	Popis
submitter_id	int (10)	cizí klíč (reference na tabulku submitter)
group_id	int (10)	cizí klíč (reference na tabulku group)

Instalační příručka

V této části popíšeme postup, jak zprovoznit aplikaci na serveru. Nejprve zmíním vlastnosti HTTP serveru, poté popíšeme vše, co je potřeba změnit v aplikaci pro její zprovoznění a nakonec vysvětlím jak zprovoznit databázi.

C.1 Vlastnosti HTTP Serveru

Server musí splňovat následující požadavky:

- Webový server Apache (aplikace byla testována na Apache 2.4.7).
- PHP (aplikace byla testována na Apache 5.5.8).
- Databázový server MySQL (aplikace byla testována na MySQL 5.6.15).
- Mail server.

C.2 Aplikace

Nejprve je potřeba zkopírovat obsah adresáře `implementation` do složky webového serveru určené pro zdrojové soubory php aplikace (většinou složka `www`). Dále je nutné zeditovat konfigurační soubor nacházející se v adresářové struktuře aplikace `app/config/config.neon`.

Je potřeba najít řádky, které obsahují nastavení potřebné k připojení k databázovému serveru. Řádky jsou vidět na příkladu C.1

Zde nahradte řetězce `DATABAZOVY_SERVER` databázovým serverem, dále pak `NAZEV_DATABAZE` názvem databáze, `UZIVATELSKE_JMENO` uživatelským jménem potřebným pro přístup do databáze a `HESLO` heslem potřebným pro přístup do databáze.

Poslední věc, která je potřeba udělat je změnit cestu ke složce `storage`, která obsahuje nauploadované dokumenty publikací. Je tedy potřeba najít soubor `app/model/Files.php`, ve kterém je třída `Files`. V ní se nachází třídni

Listing C.1: Konfigurační soubor aplikace – údaje potřebné pro přístup k databázi.

```
database :  
  dsn : 'mysql:host=DATABAZOVY_SERVER;dbname=NAZEV_DATABAZE'  
  user : UZIVATELSKE_JMENO  
  password : HESLO
```

proměnná `$dirPath`, do které je pak v konstruktoru přiřazena cesta k souborům (k složce `storage`). Tu podle potřeb změňte.

C.3 Databáze

V adresáři `database` se nachází create script databáze `createscript.sql`, kterým se vytvoří databáze s názvem `pubs`.

Uživatelská příručka

Aplikace běží pro testovací účely na adrese <http://dp.jankubalek.cz/>, kde si ji každý uživatel, který zná přihlašovací údaje, může vyzkoušet. Pokud si uživatel aplikaci zprovozní na `localhostu`, tak mu nebude fungovat odesílání e-mailů. Databáze aplikace je naplněna reálnými daty.

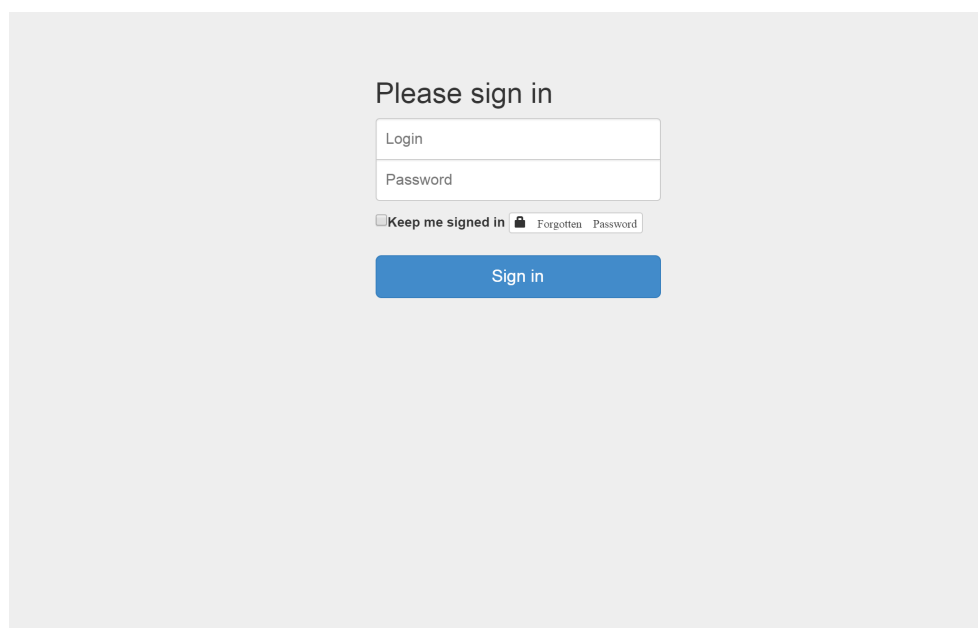
D.1 Přihlašování

Pro testovací účely jsou vytvořeny následující uživatelské účty.

- Role **Čtenář**:
přihlašovací jméno: `reader`
heslo: `heslo`
- Role **Zadavatel**:
přihlašovací jméno: `submitter`
heslo: `heslo`
- Role **Administrátor**:
přihlašovací jméno: `admin`
heslo: `heslo`

Přihlašovací obrazovka je vidět na náhledu D.1. Uživatel se tedy může přihlásit buď pomocí svých přihlašovacích údajů, anebo výše zmíněnými.

Pokud by uživatel zapomněl heslo, může si nechat zaslat odkaz pro jeho resetování a to tak, že na přihlašovací obrazovce klikne na tlačítko „Forgotten Password“. Zobrazí se mu formulář, který je vidět na náhledu D.2, uživatel vyplní login, anebo e-mail a formulář odešle kliknutím na tlačítko „Done“. Poté mu bude zaslán resetovací odkaz na e-mail uvedený v jeho uživatelském účtu. Kliknutím na tento odkaz se mu vygeneruje nové heslo, které mu bude zasláno



Obrázek D.1: Nová aplikace – login.

opět na jeho uživatelský e-mail. Za pomoci tohoto hesla svého uživatelského jména se již může přihlásit do aplikace.

D.2 Uživatelský účet

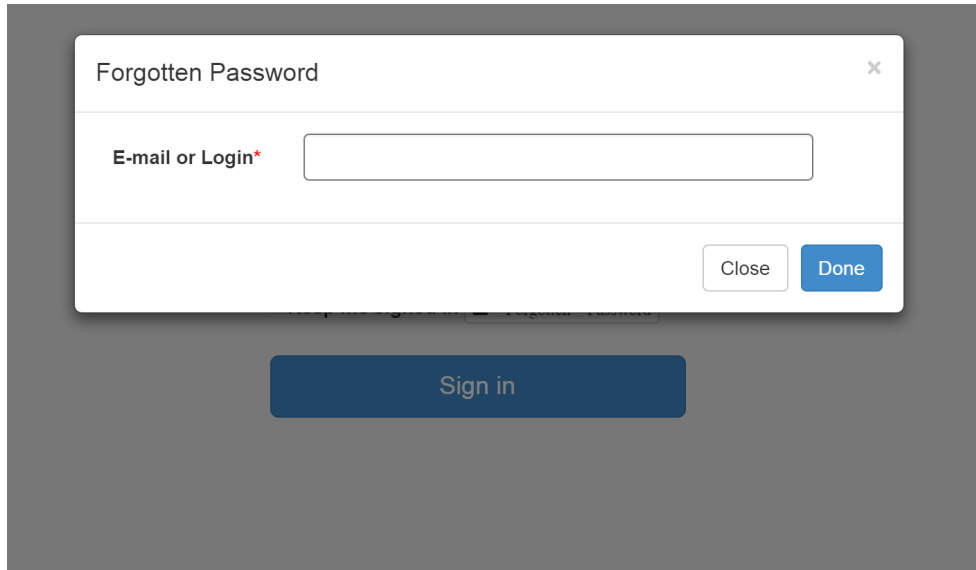
Do detailu svého uživatelského účtu se uživatel dostane kliknutím na položku se svým loginem v menu aplikace. Detail má tři záložky: „User detail“, „My annotations“ a „User settings“.

D.2.1 User detail

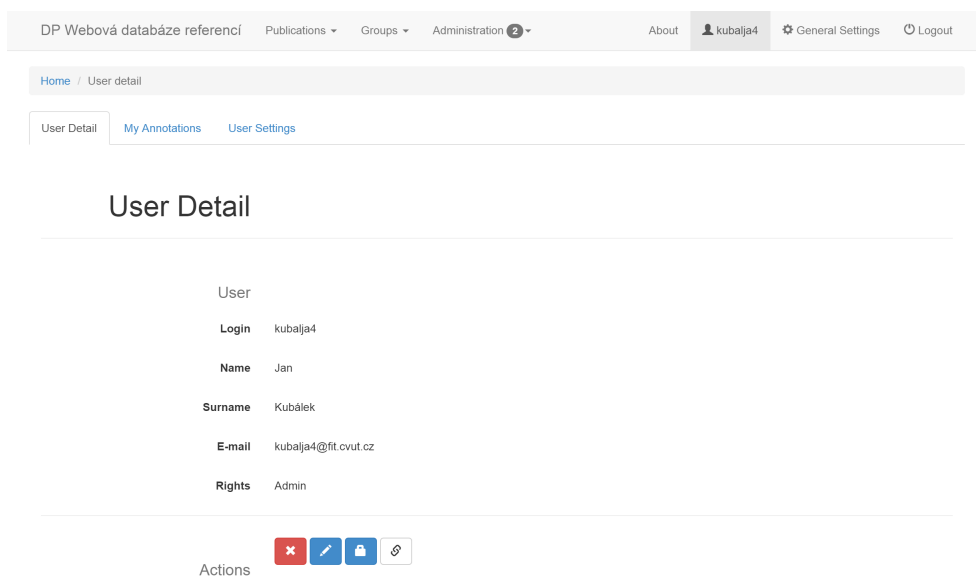
V první (defaultní) záložce „User detail“ vidí uživatel jeho osobní údaje. Vše je vidět na náhledu D.1. Údaje si může změnit kliknutím na tlačítko „Edit user“. Dále si může změnit heslo kliknutím na tlačítko „Change password“, zobrazit publikace, které přidával, anebo editoval kliknutím na tlačítko „Associated publications“, pak smazat svůj uživatelský účet kliknutím na tlačítko „Delete user“.

D.2.2 My annotations

V druhé záložce „My annotations“ se nachází poznámky uživatele. Poznámky může zeditovat a smazat kliknutím na příslušné tlačítko. Dále si může rozkliknout detail publikace, k níž se poznámka vztahuje. Vše je vidět na náhledu D.4.

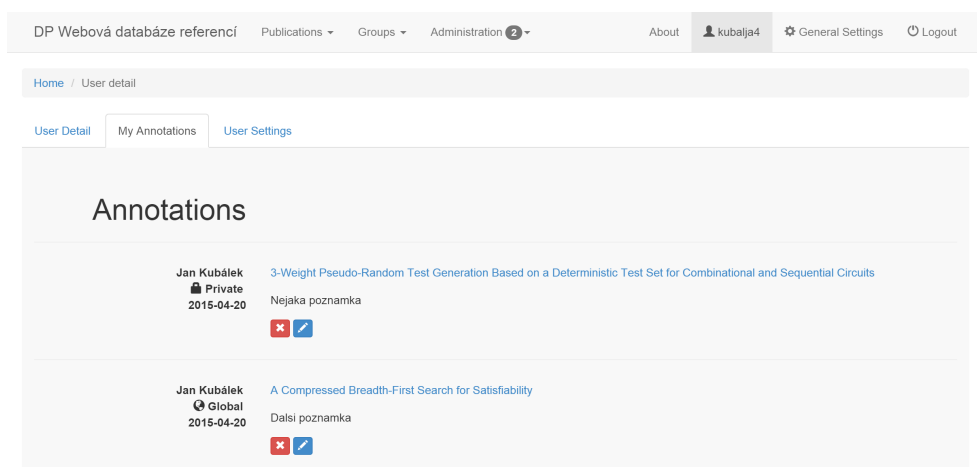


Obrázek D.2: Nová aplikace – ztráta hesla.

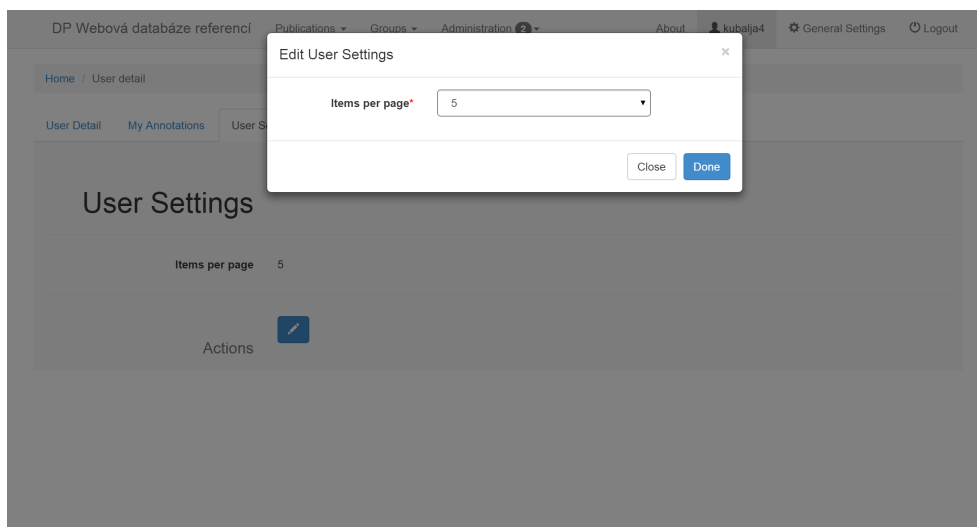


Obrázek D.3: Nová aplikace – uživatelský účet – detail.

D. UŽIVATELSKÁ PŘÍRUČKA



Obrázek D.4: Nová aplikace – uživatelský účet – poznámky.

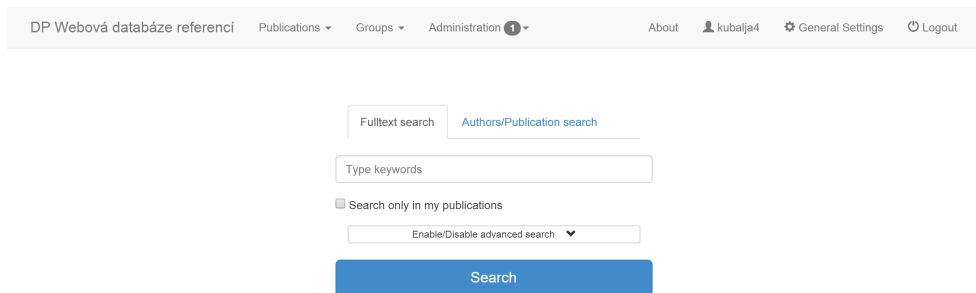


Obrázek D.5: Nová aplikace – uživatelský účet – uživatelská nastavení.

D.2.3 User settings

Ve třetí, poslední záložce „User settings“ se nachází uživatelské nastavení. Zde může uživatel ovlivnit chování aplikace. Kliknutím na „Edit User Settings“ toto nastavení změní. Vše je vidět na náhledu D.5.

D.3. Domovská stránka – vyhledávání



The screenshot shows the search interface of the DP Webová databáze referencí. At the top, there is a navigation bar with links for 'Publications', 'Groups', 'Administration', 'About', 'kubajja4', 'General Settings', and 'Logout'. Below the navigation bar, there are two search options: 'Fulltext search' and 'Authors/Publication search'. A text input field labeled 'Type keywords' is positioned below these options. To the left of the input field is a checkbox labeled 'Search only in my publications'. Below the input field is a dropdown menu labeled 'Enable/Disable advanced search'. At the bottom of the search area is a blue 'Search' button.

Obrázek D.6: Nová aplikace – homepage.

D.3 Domovská stránka – vyhledávání

Na domovské stránce se nachází Fulltextové vyhledávání a vyhledávání v názvech publikací a jménech autorů. Uživatel si mezi těmito dvěma druhy vyhledávání může přepínat, jak je vidět na náhledu D.6. V případě že chce, aby vyhledávání probíhalo pouze v rámci „jeho publikací/my publications“, zaškrtně checkbox „Search only in my publications“. Pokud chce aplikovat pokročilé vyhledávání, klikne na tlačítko „Enable/Disable Advanced search“.

V pokročilém nastavení pak může uživatel vybrat buď operátor OR, nebo AND a nakonec kategorie publikací. Operátory uživatel specifikuje, zda vyhledané publikace mají patřit do všech kategorií zároveň, nebo alespoň do jedné.

Samotné vyhledání se provede kliknutím na tlačítko „Search“. Poté je uživateli zobrazen seznam výsledků.

D.4 Seznam výsledků vyhledávání

Po vyhledání je uživateli zobrazen seznam výsledků. Seznam je možné řadit (Relevancy, Title, Year). Výsledky vyhledávání obsahují úryvky nalezeného textu s tučně vyznačenými nalezenými výrazy, seznam autorů dané publikace a zeleně vyznačený seznam kategorií, ve kterých je publikace zařazena.

Kliknutím na název publikace se uživateli zobrazí detail publikace. Vše je vidět na náhledu D.7.

D. UŽIVATELSKÁ PŘÍRUČKA

DP Webová databáze referencí Publications Groups Administration 1 About kubajka4 General Settings Logout

Home Search Results

Fulltext search Authors/Publication search

boolean

Search only in my publications

Enable/Disable advanced search

Search

Order by: [Relevancy](#) | [Title](#) | [Year](#)

[Circuit-based Boolean Reasoning \(inproceedings\)](#)
Authors: Kuehlmann Andreas, Ganai K. Malay, Paruthi Viresh
15_2.pdf --- Circuit-based **Boolean** Reasoning Andreas Kuehlmann Cadence Berkeley Labs Berkeley, ...
Categories: Hardware, Logic design, BDD, SAT

[Semantical Expansion of Two-Level Boolean Formulas for Minimisation \(misc\)](#)
Authors: Lange Martin
--- Semantical Expansion of Two-Level **Boolean** Formulas for Minimisation Martin...
Categories: Hardware, Logic design, Two-level minimization

[PALMINI---fast Boolean minimizer for personal computers \(inproceedings\)](#)
Authors: Nguyen B. L., Perkowski A. M., Goldstein B. N.
p615-nguyen.pdf --- PALMINI - FAST **BOOLEAN** MINIMIZER COMPUTERS. FOR PERSONAL Lot Bao...
Categories: Hardware, Logic design

[Spectral Transforms for Large Boolean Functions with Applications to Technology Mapping \(inproceedings\)](#)
Authors: Clarke M. E., McMillan L. K., Zhao X, Fujita M., Yang J.
--- Spectral Transforms for Large **Boolean** Functions...
Categories: Hardware, Logic design, Spectral methods

« Back 1 2 3 4 ... 6 7 Next »

Obrázek D.7: Nová aplikace – seznam výsledků vyhledávání.

D.5 Detail publikace

Do detailu publikace se uživatel dostane buď skrz seznam všech publikací „All publications“, nebo skrz seznam vyhledaných výsledků a to vždy kliknutím na název dané publikace. Samotný detail publikace má čtyři záložky: „Publication detail“, „Annotations“, „Definitions“ a „Citations“.

D.5.1 Publication detail

V této první záložce jsou zobrazeny informace o publikaci (název, typ, kategorie, atd.). Dále si zde uživatel může stáhnout přiložené soubory. Pokud má vyšší roli (alespoň roli zadavatel), tak může publikaci editovat, smazat a přidat do „My publications“. Pokud má roli administrátor, tak může publikaci schválit nebo nastavit jako neschválenou. Vše je vidět na náhledu D.8.

DP Webová databáze referencí | Publications | Groups | Administration | About | kubalja4 | General Settings | Logout

Home / Publication detail

Publication Detail | Annotations | Definitions | Citations

Publication Detail

Publication

Title SAT-based logic optimization and resynthesis

Type of publication InProceedings (an article in a conference proceedings)

Abstract The paper develops a technology-independent optimization and post-mapping resynthesis for combinational logic networks, with emphasis on scalability and optimizing power. The proposed resynthesis (a) is capable of substantial logic restructuring, (b) is customizable to solve a variety of optimization tasks, and (c) has reasonable runtime on large industrial designs. The approach is based on several heterogeneous algorithms, which include structural analysis, random and constrained simulation, and manipulation of Boolean functions using a SAT solver. Structural methods include improved windowing, which focuses on reconvergent logic structures rich in functional flexibilities. It is shown how a mainstream SAT solver can be minimally modified by combining it with an interpolation package, which computes Boolean functions of nodes after resynthesis as a by-product of completed feasibility proofs. Experimental results focusing on the minimization of the number of 6-LUTs after high-effort iterative FPGA mapping with structural choices, demonstrate that the proposed resynthesis, applied to 15 benchmarks reduced area by 6.0 % and delay by 2.3 % on average. For 5 benchmarks derived from PLA descriptions, the reduction of 5x in area and 20 % in depth was obtained, which speaks for the powerful nature of Boolean optimization employed in the proposed resynthesis.

Categories Hardware, Logic design, AIG, Cut technique, rewiring, Boolean network optimization, Don't cares

Groups

Pages 358-364

Year and month of publication 2006-11-30

Conference

Name International Workshop on Logic and Synthesis

Abbreviation IWLSs


Year of Conference





Name International Workshop on Logic and Synthesis

Year 2007

Authors Brayton, Robert, K., Mishchenko, Alan, Jang, Stephen, Jiang, Jie-Hong, Roland

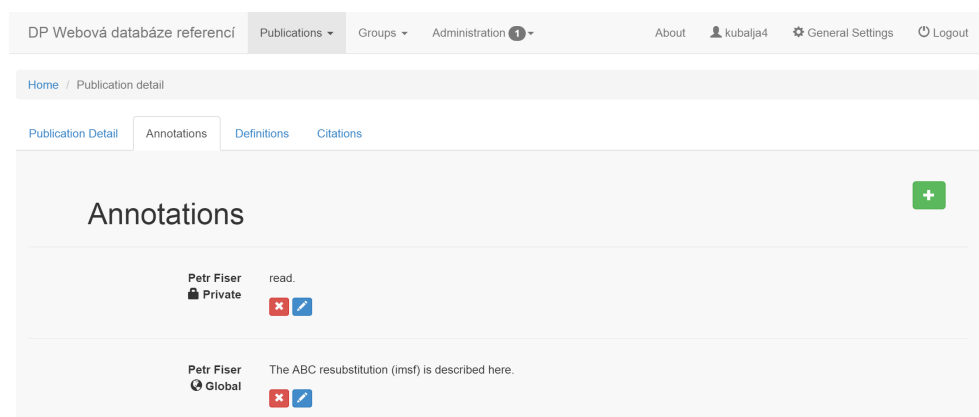
Files to download

iwls07_imfs.pdf 

Actions    

Obrázek D.8: Nová aplikace – detail publikace.

D. UŽIVATELSKÁ PŘÍRUČKA



Obrázek D.9: Nová aplikace – detail publikace – poznámky.

D.5.2 Annotations

V záložce „Annotations“ jsou poznámky k publikaci. Všichni uživatelé mohou přidávat poznámky k publikaci a to buď veřejné nebo soukromé.

Uživatelé s rolí administrátor vidí všechny poznámky (veřejné i soukromé), mohou je editovat a mazat.

Uživatelé s rolí čtenář a zadavatel vidí své poznámky a pak veřejné. Editovat a mazat mohou pouze své poznámky.

Vše je vidět na náhledu D.9.

D.5.3 Definitions (export)

V této záložce jsou exporty publikací ve formátech BibTeX, EndNote a RefWorks. Vše je vidět na náhledu D.10.

D.5.4 Citations

V této záložce, jak je vidět na náhledu D.11, jsou styly citací.

D.6 Přidání nové publikace

Do formuláře pro přidávání nové publikace se uživatel dostane skrz menu „Publications/Add new publication“. Uživatel může vyplnit všechna políčka ručně, nebo je nainportovat z nějaké formátu (BibTeX, EndNote, RefWorks), anebo je stáhnout ze Springer katalogu a to na základě ISBN nebo ISSN anebo DOI. Vše je vidět na náhledu D.12.

Pokud uživatel přidá publikaci s rolí čtenář nebo zadavatel, musí být poté schválena administrátorem.

D.6. Přidání nové publikace

DP Webová databáze referencí | Publications | Groups | Administration | About | kuba|ja4 | General Settings | Logout

Home / Publication detail

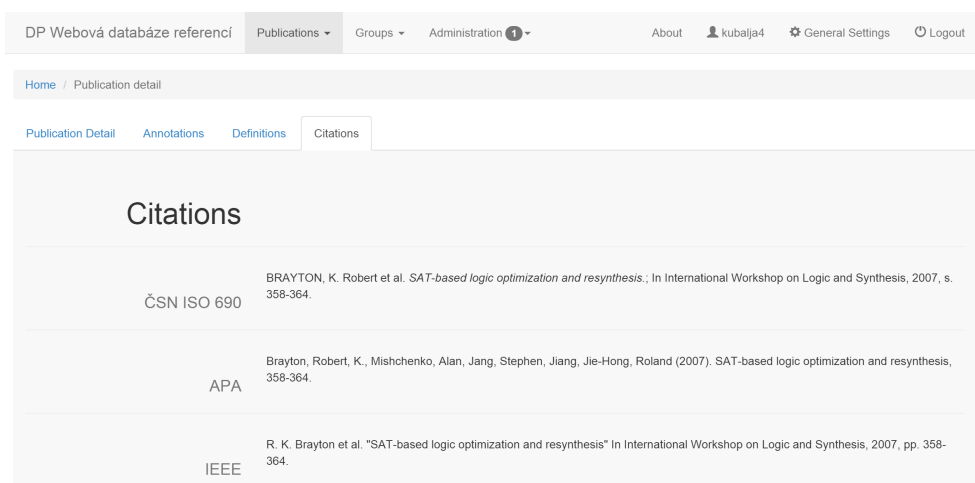
Publication Detail | Annotations | Definitions | Citations

Definitions

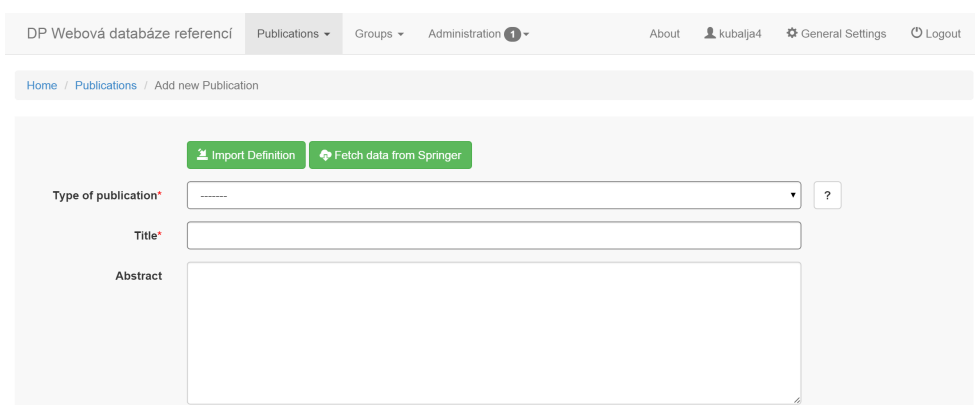
Format	Content
Bibtex	<pre>@inproceedings{2007, author = "Brayton, Robert K. and Mishchenko, Alan and Jang, Stephen and Jang, Jie-Hong, Roland", booktitle = "International Workshop on Logic and Synthesis", title = "SAT-based logic optimization and resynthesis", year = "2007", pages = "358-364" }</pre>
EndNote	<pre>%0 Conference Paper %A Brayton, Robert K %A Mishchenko, Alan %A Jang, Stephen %A Jang, Jie-Hong Roland %T SAT-based logic optimization and resynthesis %B International Workshop on Logic and Synthesis %D 2007 %8 2006-11-30 00:00:00 %P 358-364 %X The paper develops a technology-independent optimization and post-mapping resynthesis for combinational logic networks, with emphasis on scalability and optimizing power. The proposed resynthesis (a) is capable of substantial logic restructuring, (b) is customizable to solve a variety of optimization tasks, and (c) has reasonable runtime on large industrial designs. The approach is based on several heterogeneous algorithms, which include structural analysis, random and constrained simulation, and manipulation of Boolean functions using a SAT solver. Structural methods include improved windowing, which focuses on reconvergent logic structures rich in functional flexibilities. It is shown how a mainstream SAT solver can be minimally modified by combining it with an interpolation package, which computes Boolean functions of nodes after resynthesis as a by-product of completed feasibility proofs. Experimental results focusing on the minimization of the number of 6-LUTs after high-effort iterative FPGA mapping with structural choices, demonstrate that the proposed resynthesis, applied to 15 benchmarks reduced area by 6.0 % and delay by 2.3 % on average. For 5 benchmarks derived from PLA descriptions, the reduction of 5x in area and 20 % in depth was obtained, which speaks for the powerful nature of Boolean optimization employed in the proposed resynthesis.</pre>
RefWorks	<pre>RT Conference Proceedings A1 Brayton, Robert K A1 Mishchenko, Alan A1 Jang, Stephen A1 Jang, Jie-Hong Roland T1 SAT-based logic optimization and resynthesis T2 International Workshop on Logic and Synthesis YR 2007 FD 2006-11-30 00:00:00 SP 358 OP 364 AB The paper develops a technology-independent optimization and post-mapping resynthesis for combinational logic networks, with emphasis on scalability and optimizing power. The proposed resynthesis (a) is capable of substantial logic restructuring, (b) is customizable to solve a variety of optimization tasks, and (c) has reasonable runtime on large industrial designs. The approach is based on several heterogeneous algorithms, which include structural analysis, random and constrained simulation, and manipulation of Boolean functions using a SAT solver. Structural methods include improved windowing, which focuses on reconvergent logic structures rich in functional flexibilities. It is shown how a mainstream SAT solver can be minimally modified by combining it with an interpolation package, which computes Boolean functions of nodes after resynthesis as a by-product of completed feasibility proofs. Experimental results focusing on the minimization of the number of 6-LUTs after high-effort iterative FPGA mapping with structural choices, demonstrate that the proposed resynthesis, applied to 15 benchmarks reduced area by 6.0 % and delay by 2.3 % on average. For 5 benchmarks derived from PLA descriptions, the reduction of 5x in area and 20 % in depth was obtained, which speaks for the powerful nature of Boolean optimization employed in the proposed resynthesis.</pre>

Obrázek D.10: Nová aplikace – detail publikace – definice (export).

D. UŽIVATELSKÁ PŘÍRUČKA



Obrázek D.11: Nová aplikace – detail publikace – citace.



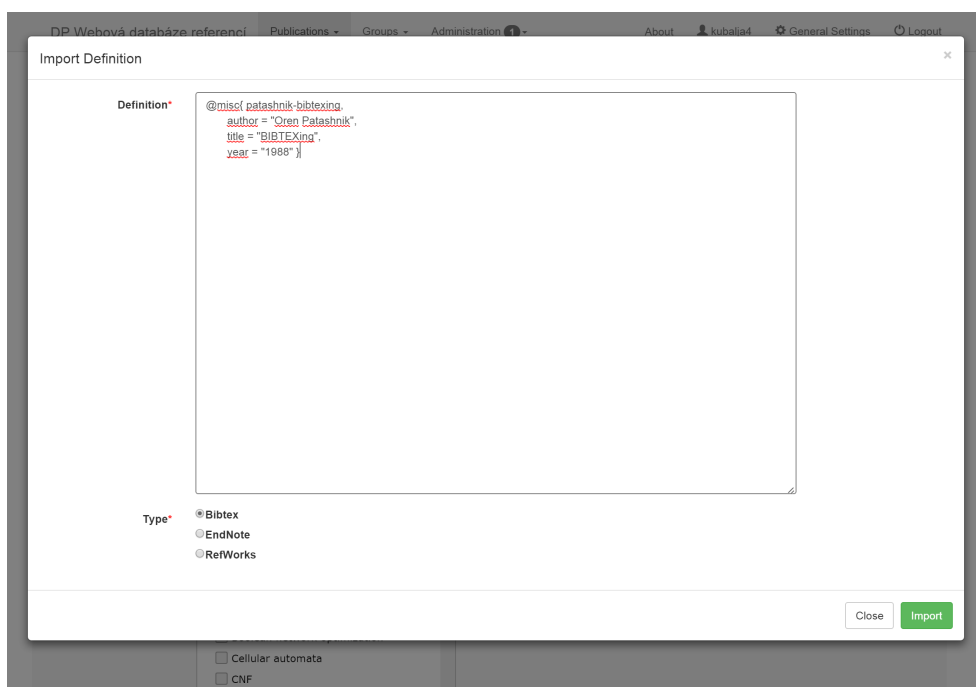
Obrázek D.12: Nová aplikace – přidání nové publikace.

D.6.1 Import publication

Import publikace se vyvolá kliknutím na tlačítko „Import Definition“, poté je uživateli zobrazen formulář, který vyplní a odešle tlačítkem „Import“. Pokud vše proběhne v pořádku, jsou data přednastavena do přidávacího formuláře a uživatel je může ještě poupravit a doplnit. Vše je vidět na náhledu D.13.

D.6.2 Fetch data from Springer

Stažení dat ze Springer katalogu probíhá ve dvou krocích. Uživatel klikne na tlačítko „Fetch data from Springer“, poté zadá buď ISBN nebo ISSN anebo DOI a klikne na tlačítko „Fetch data“. Data jsou poté stažena a uživatel si vybere, která konkrétní chce naimportovat. Import dokončí kliknutím na tla-



Obrázek D.13: Nová aplikace – přidání nové publikace – import.

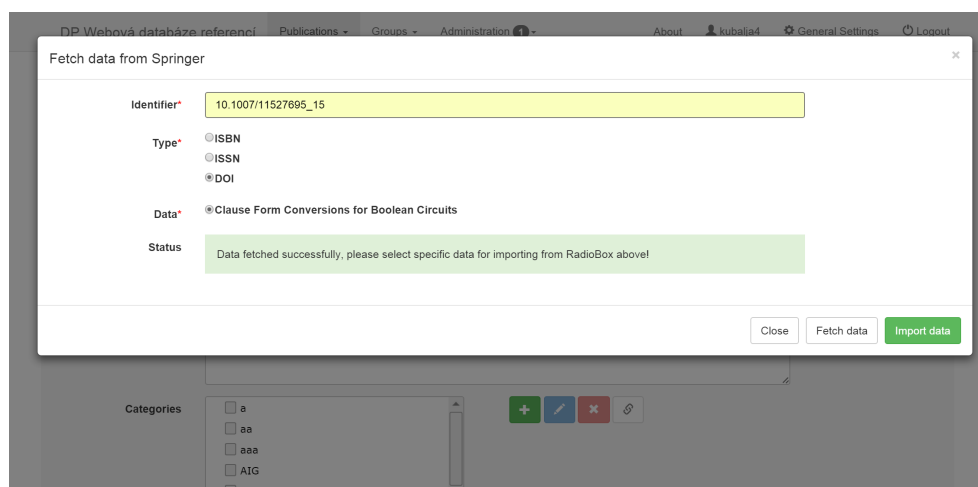
čítko „Import data“. Pokud vše proběhne v pořádku, jsou data přednastavena do přidávacího formuláře a uživatel je může ještě poupravit a doplnit stejně jako v případě importu publikace. Vše je vidět na náhledu D.14.

D.6.3 Manuální vyplnění

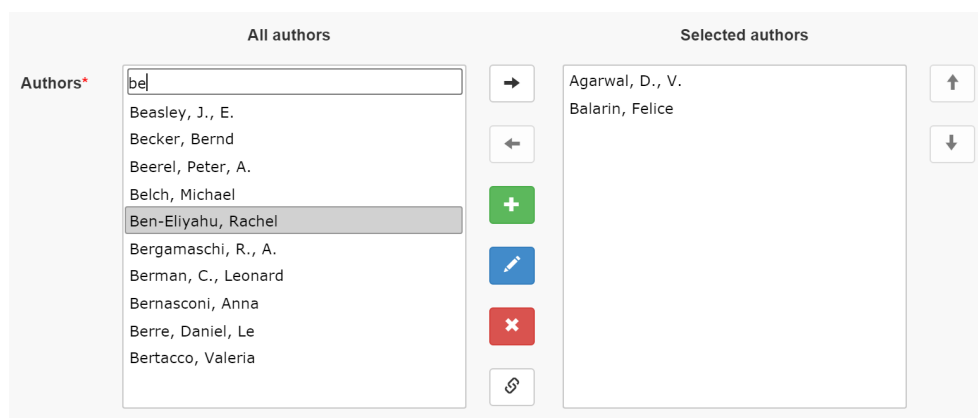
Uživatel může vyplnit řadu informací týkajících se publikací. Nejprve je však potřeba vybrat typ publikace, na základě toho jsou zobrazena příslušná políčka – povinná a nepovinná. Povinná políčka jsou zvýrazněna černou linkou a u popisku je červená hvězdička.

U políček typu autor, kategorie, uživatelská skupina, konference, ročník konference, vydavatel a dodatečný atribut se nachází sada tlačítek – „Edit“, „Delete“, „Add new“, „Associated publications“. Uživatel může tedy v rámci procesu „přidávání publikace“ možnost ovlivňovat téměř veškerý obsah databáze – editovat, mazat, přidávat záznamy do všech tabulek vztahujících se k publikaci a zároveň si zjišťovat, které další publikace touto modifikací ovlivní.

U políček typu datum je pomocný doplněk pro výběr data, takže uživatel nemusí nic manuálně vypisovat. U všech políček, která nemusí být každému ihned jasná, se nachází nápověda.



Obrázek D.14: Nová aplikace – přidání nové publikace – Springer.



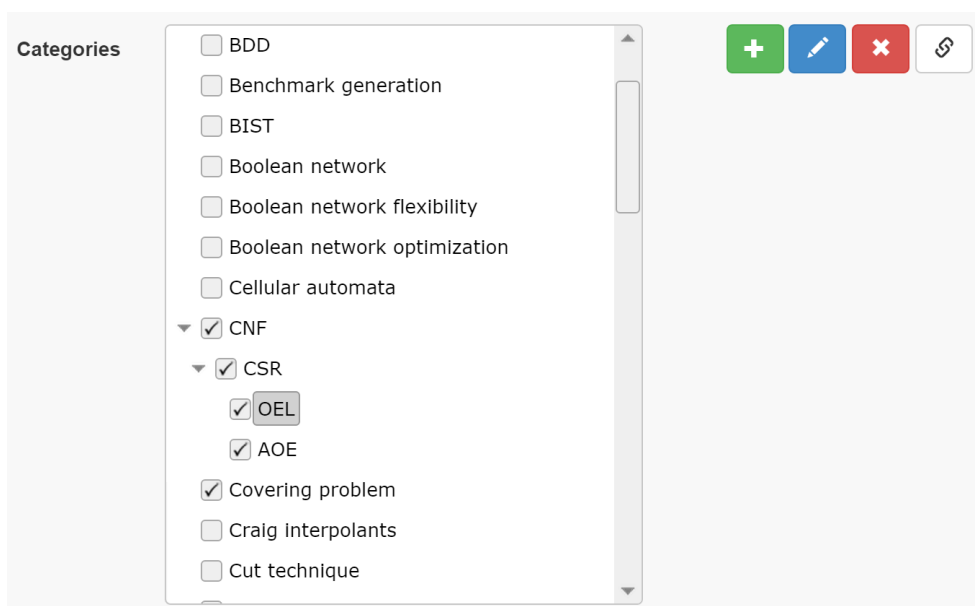
Obrázek D.15: Nová aplikace – přidání nové publikace – autoři.

D.6.3.1 Autoři

Přiřazení autorů k publikaci probíhá následovně. Uživatel si v boxu „All authors“ vybere/vyhledá autora a buď ho drag&drop přetáhne do boxu „Selected authors“, nebo ho pouze označí a klikne na tlačítko „Select“, čímž se provede ta samá operace. Vše je vidět na náhledu D.15.

Přiřazené autory v boxu „Selected authors“ může uživatel také řadit podle významnosti a to buď drag&drop řazením, nebo výběrem a klikáním na tlačítka „Move up“ a „Move down“.

Veškeré autory může uživatel editovat, mazat a přidávat, jak již bylo zmíněno v úvodu.



Obrázek D.16: Nová aplikace – přidání nové publikace – kategorie.

D.6.3.2 Kategorie

Přiřazení kategorií probíhá tak, že uživatel zaškrtně checkboxy konkrétních kategorií či podkategorií. Pokud chce kategorie editovat/mazat/zjistit přiřazené publikace ke kategorii, tak kategorii pouze označí výběrem (nezaškrťává) a provede jednu z akcí. Vše je vidět na náhledu D.16.

Při přidávání nové kategorie, si uživatel může vybrat, zda to bude kategorie na hierarchické úrovni root, anebo zda to bude podkategorie nějaké kategorie. Pokud chce vytvořit podkategorii, je potřeba před samotným přidáváním mít označenou kategorii, která bude jejím rodičem a pod kterou tedy bude spadat.

D.6.3.3 Uživatelské skupiny

Přiřazení uživatelských skupin probíhá naprosto stejně, jako v případě kategorií jen s tím rozdílem, že nejde tvořit hierarchická struktura uživatelských skupin a jsou tedy všechny na stejné root úrovni. Vše je vidět na náhledu D.17.

D.6.3.4 Vydavatelé

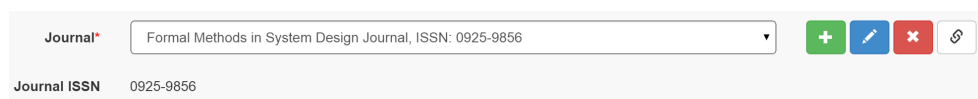
Přiřazení vydavatele probíhá výběrem konkrétního vydavatele v selectboxu vydavatelů. Vybraného vydavatele může uživatel editovat, mazat atd. jak již bylo zmíněno v úvodu. Vše je vidět na náhledu D.18.



Obrázek D.17: Nová aplikace – přidání nové publikace – uživatelské skupiny.



Obrázek D.18: Nová aplikace – přidání nové publikace – vydavatelé.



Obrázek D.19: Nová aplikace – přidání nové publikace – časopisy.

D.6.3.5 Časopisy

Přiřazení časopisu probíhá úplně stejně jako v případě vydavatelů – výběrem konkrétního časopisu v selectboxu časopisů. Vybraný časopis může uživatel editovat, mazat atd. jak již bylo zmíněno v úvodu. Vše je vidět na náhledu D.19.

D.6.3.6 Konference, ročníky konference

Přiřazení konferencí a ročníků konferencí probíhá ve dvou krocích. Nejprve uživatel v selectboxu (konferencí) vybere konferenci, poté se mu zobrazí v se-

Obrázek D.20: Nová aplikace – přidání nové publikace – konference.

Obrázek D.21: Nová aplikace – přidání nové publikace – atributy.

lectboxu (ročníků konferencí) ročníky konferencí této vybrané konference. Uživatel poté vybere konkrétní ročník konference. Vše může editovat, mazat atd. jak již bylo zmíněno v úvodu. Vše je vidět na náhledu D.20.

D.6.3.7 Dodatečné atributy

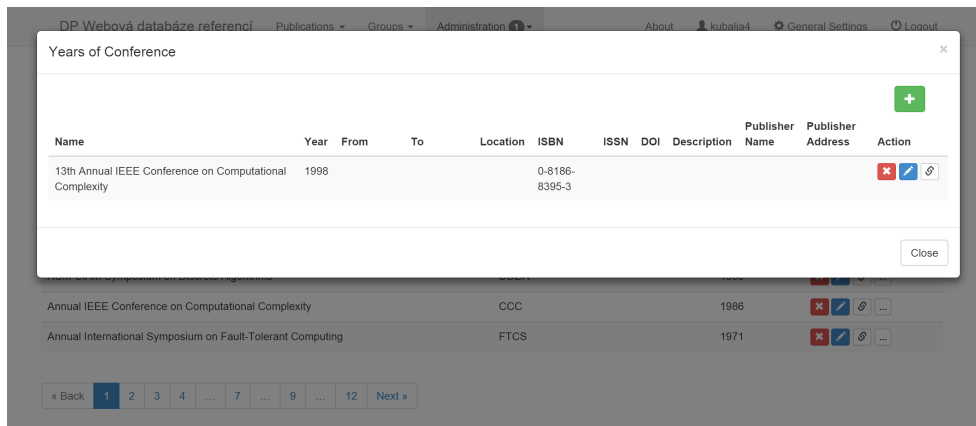
Přiřazení dodatečných atributů probíhá tak, že uživatel vyplní jejich hodnotu. Stávající může editovat mazat atd. Pokud přidává nové, tak je potřeba aby název atributu začínal podtržítkem. Vše je vidět na náhledu D.21.

D.6.3.8 Soubory

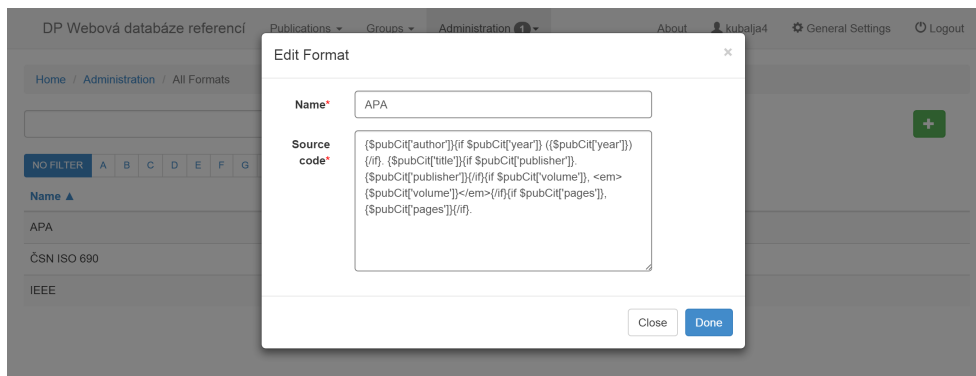
Přidávání souborů probíhá ve dvou krocích. Nejprve uživatel vybere konkrétní soubory a to buď drag&drop přetažením do boxu, anebo manuálně kliknutím na tlačítko „Add Files“. Jakmile uživatel vybere soubory, zahájí samotný upload stisknutím tlačítka „Start upload“. Vše je vidět na náhledu D.21.

D.7 Editace publikace

Editace publikace vypadá podobně jako přidávání nové publikace jen s tím rozdílem, že uživatel nemá možnost importu a stáhnutí dat ze Springer katalogu.



Obrázek D.24: Nová aplikace – seznamy – konference.



Obrázek D.25: Nová aplikace – seznamy – formáty citací.

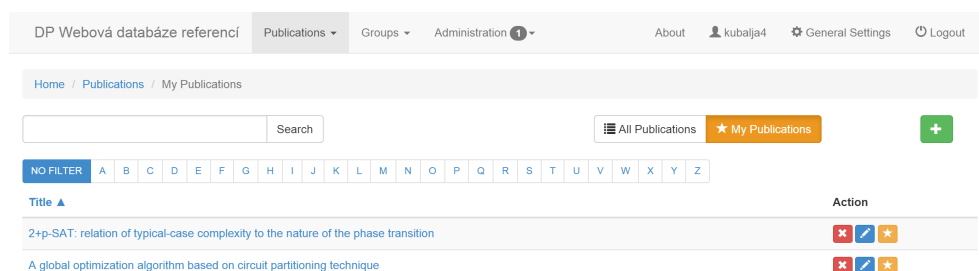
D.8.2 Seznam konferencí

Seznam konferencí je specifický tím, že u každé konference je kromě ma-zání, editace, seznamu přidružených publikací také seznam ročníků konference. Tento seznam se vyvolá stiskem tlačítka „Show Years of Conference“. Seznam se objeví v modálním okně a uživatel může opět veškeré záznamy modifikovat atd. Vše je vidět na náhledu D.24.

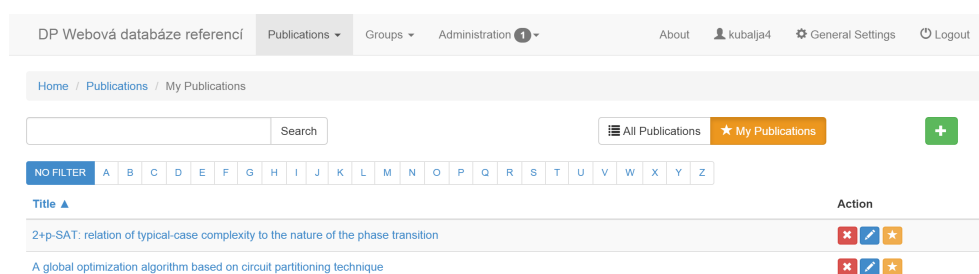
D.8.3 Seznam formátů

Seznam formátů je přístupný jen pro administrátora. Administrátor zde může přidávat formáty citací. Nutnou podmínkou je znalost šablonovacího jazyka Latte. Vše je vidět na náhledu D.25.

D. UŽIVATELSKÁ PŘÍRUČKA



Obrázek D.26: Nová aplikace – seznamy – publikace.



Obrázek D.27: Nová aplikace – seznamy – moje publikace.

D.8.4 Publikace

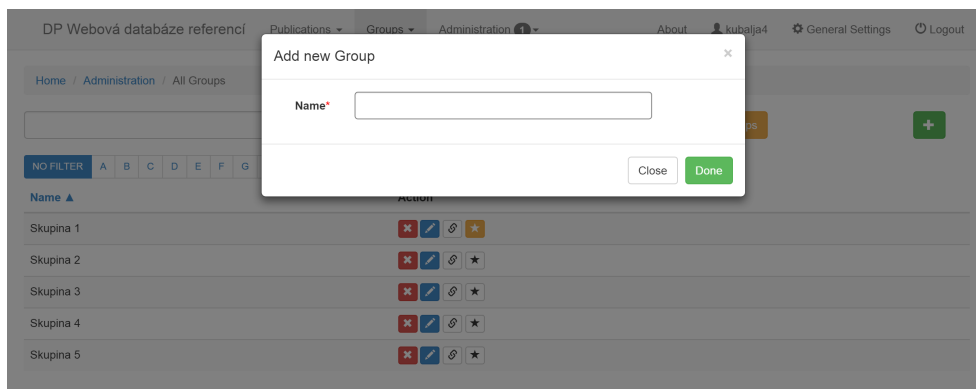
Uživatel si může zobrazit seznamy publikací a to „All Publications“ a „My publications“ skrz menu „Publications“. Všechny položky tohoto seznamu může modifikovat atd.

D.8.4.1 Všechny publikace

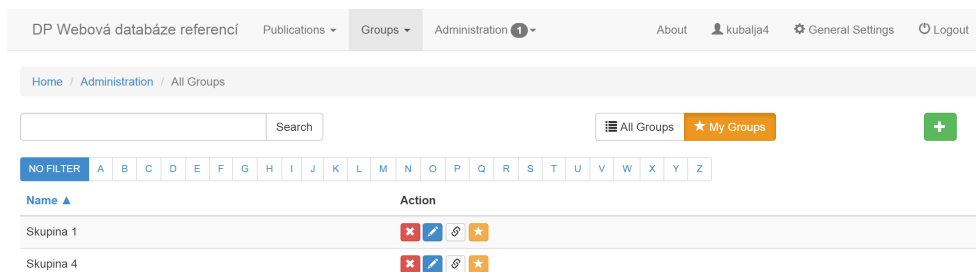
Uživateli je zobrazen seznam všech publikací „All Publications“. Publikace si může kliknutím na tlačítko „Add to my publications“ přidat do svých publikací „My publications“. Publikace se mu poté zobrazuje v „My publications“. Publikaci může samozřejmě uživatel odebrat z „My publications“ kliknutím na tlačítko „Remove from my publications“. Vše je vidět na náhledu D.26.

D.8.4.2 Moje publikace

Uživatel v tomto seznamu vidí své publikace, respektive publikace, které si přidal do seznamu „My publications“. Publikace samozřejmě může ze seznamu odebrat kliknutím na tlačítko „Remove from my publications“. Vše je vidět na náhledu D.27.



Obrázek D.28: Nová aplikace – seznamy – uživatelské skupiny.



Obrázek D.29: Nová aplikace – seznamy – moje uživatelské skupiny.

D.8.5 Uživatelské skupiny

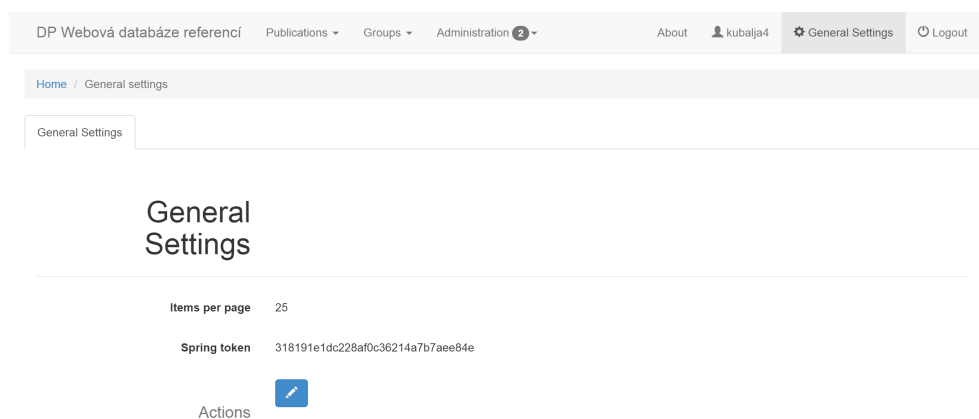
Uživatel si může zobrazit seznamy uživatelských skupin a to „All Groups“ a „My Groups“ skrz menu „Groups“. Všechny položky tohoto seznamu může modifikovat atd. Vše je vidět na náhledu D.28.

D.8.5.1 Všechny skupiny

Principiálně tento seznam funguje jako seznam všech publikací – uživatel si může kliknutím na tlačítko „Add to my groups“ přidat skupinu do svých uživatelských skupin „My Groups“. Skupina se mu poté zobrazuje v „My Groups“. Skupinu může samozřejmě uživatel odebrat z „My Groups“ kliknutím na tlačítko „Remove from my groups“.

D.8.5.2 Moje skupiny

Opět funguje stejně jako „My publications“ – uživatel v tomto seznamu vidí své skupiny, které může ze seznamu odebrat kliknutím na tlačítko „Remove from my groups“. Vše je vidět na náhledu D.29.



Obrázek D.30: Nová aplikace – seznamy – obecné nastavení aplikace.

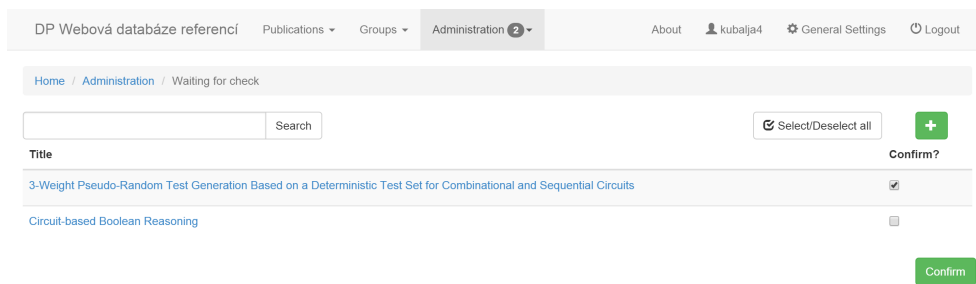
D.9 General settings

Toto nastavení je přístupné skrz položku horního menu „General Settings“. Nastavení je viditelné pouze pro administrátora a ovlivňuje chod celé aplikace. Administrátor zde může nastavit defaultní hodnotu stránkování, která se přednastaví každému nově přidanému uživateli. Ten si pak může hodnotu samozřejmě změnit v uživatelském nastavení „User settings“. Dále zde může měnit API Key ke Spriger API. Vše je vidět na náhledu D.30.

D.10 Waiting for check

Schvalování publikací „Waiting for check“ je přístupné pouze pro administrátora. Nachází se v horním menu Administration/Waiting for check a administrátor zde schvaluje všechny publikace přidané uživateli s rolí čtenář nebo zadavatel. Vše je vidět na náhledu D.31.

Administrátor může všechny publikace označit najednou tlačítkem „Select/Deselect All“, nebo jednotlivě a poté provést samotné schválení kliknutím na tlačítko „Confirm“.



Obrázek D.31: Nová aplikace – seznamy – schvalování publikací.

Obsah přiloženého CD

readme.txt.....	stručný popis obsahu CD
src	
├─ database	script pro vytvoření databáze
├─ implementation.....	zdrojové soubory aplikace
├─ storage	nahrané dokumenty publikací
└─ thesis	zdrojová forma práce ve formátu L ^A T _E X
text	text práce
├─ thesis.pdf	text práce ve formátu PDF
└─ thesis.ps	text práce ve formátu PS