Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Bakalářská práce

Generátor a parser formulářů recenzí příspěvků na konferenci TSD

Plzeň 2019 Vojtěch Danišík

Místo této strany bude zadání práce.

Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.
V Bakalářské práci jsou použity názvy programových produktů, firem apod., které mohou být ochrannými známkami nebo registrovanými ochrannými známkami příslušných vlastníků.

V Plzni dne 17. března 2019

Vojtěch Danišík

Poděkování

Děkuji panu Ing. Kamilu Ekšteinovi Ph.D. za ochotu při vedení bakalářské práce a rady s jejím vypracováním.

Abstract

Generator and Parser of Submission Review Forms for the TSD Conference. The goal of this thesis is to create PHP module, which will be easily integrate into existing information system for managing TSD conference. First part of the thesis explains standard PDF format and forms created in PDF. Subsequently, there are described existing PHP libraries for generating off-line PDF forms of scientific contribution, their advantages and disadvantages. Second part of the thesis describing existing PHP libraries for parsing PDF file. The module was tested by conference system users and multiple PDF browsers were used. Test results are part of this thesis.

Abstrakt

Cílem bakalářské práce je vytvořit jednoduše integrovatelný PHP modul do již existujícího informačního systému pro správu konference TSD. První část práce důkladně vysvětluje standardní formát PDF a formuláře vytvořené v PDF. Následně jsou popsány existující PHP knihovny pro generování off-line PDF formuláře daného vědeckého příspěvku, jejich výhody a nevýhody. Druhá část práce popisuje existující PHP knihovny pro parsování souborů ve formátu PDF. Modul byl otestován uživateli konferenčního systému a bylo použito více PDF prohlížečů. Výsledky testování jsou součástí této práce.

Obsah

1 Úvod

TSD (**T**ext, **S**peech and **D**ialogue) je konference zabývající se problémy zpracování přirozeného jazyka. Mezi nejčastěji probíraná témata se řadí: rozpoznávání řeči, modelování řeči,textové korpusy, značkování textu a mnoho dalších. Konference se koná každý rok v září a místo konání se střídá mezi Brnem (pořadatelem je Fakulta informatiky Masarykovy Univerzity) a Plzní (pořadatelem je Fakulta aplikovaných věd Západočeské univerzity v Plzni). Tento rok bude konference organizována právě Západočeskou univerzitou, a poprvé se bude konat za hranicemi České Republiky, přesněji na Slovinsku ve městě Ljubljana.

Ke každé konferenci existuje webový portál vytvořený daným pořadatelem, na nějž jsou od uživatelů nahrávány vědecké příspěvky. Tyto příspěvky jsou poté hodnoceny recenzenty (převážně členy programového výboru) formou online formuláře a na základě konečného hodnocení jednotlivých parametrů a na doporučení recenzentů jsou tyto příspěvky schváleny organizátorem a mohou být prezentovány na konferenci, nebo jsou zamítnuty z důvodu nedostatečného hodnocení. Modul vytvářený autorem bude implementován do webového portálu organizovaný Fakultou aplikovaných věd.

Cílem této práce je prostudovat strukturu PDF formátu, který je pro vytváření editovacích formulářů nejvhodnější a byl vybrán zadávajícím jako standard, tak i funkcionalitu volně dostupných PHP knihoven pro generování a parsování PDF souborů obsahujících editovatelný formulář, aby existovala možnost ohodnocení daného vědeckého příspěvku i v místech, kde není dostupné internetové připojení, neboli off-line. Tento PDF soubor musí obsahovat hodnotící formulář se všemi hodnotícími parametry, text vědeckého příspěvku doplněný o vodoznak. Pro generování a parsování musí být použity výhradně knihovny v jazyce PHP, jelikož není vhodné využívat aplikace třetích stran spustitelné z terminálu. Modul musí být nezávislý na platformě a lze ho upravovat v jakémkoliv PDF prohlížeči nezávisle na verzi PDF. Před vytvořením modulu na testovací verzi webového portálu bude potřeba projít zdrojové soubory webového portálu pro seznámení s již existujícími funkcionalitami a zařadit do portálu i náš modul. Z dřívějších let je zde naimplementován totožný modul pro generování a parsování PDF souborů, bohužel tento modul nesplňuje veškeré body zadání právě z důvodu použití nevhodného parseru.

2 Formát PDF

Formát **PDF** (**P**ortable **D**ocument **F**ormat) je souborový formát vyvinutý společností Adobe v roce 1992. PDF formát byl vyvinut za účelem konzistentní prezentace dokumentů (spustitelné na více zařízeních a různých platformách). Díky konzistenci lze dosáhnout toho, že PDF soubor vytvořený a uložený v systému Windows bude zobrazen totožně na systémech Mac, na všech distribucích Linuxu nezávisle na použitém PDF prohlížeči (Adobe Reader, Foxit a další).

V PDF souboru lze uchovávat velice širokou škálu dat, včetně formátovaného textu, vektorové grafiky a rastrových obrazů, nebo například informace o rozložení, velikosti a tvaru stránky. Informace definující umístění jednotlivých položek (jsou zde zahrnuty i editovací objekty pro formuláře) na stránce jsou zde uloženy též. Do dokumentu lze ukládat i metadata. Metadata jsou informace uložené v hlavičce souboru a lze do nich uložit název dokumentu, autora dokumentu, předmět a klíčová slova. Je zde možnost uložit heslo, aby byl dokument přístupný pouze autorizovaným uživatelům. Všechny tyto informace jsou uloženy ve standardním formátu [2, 5].

2.1 Objekty

PDF Objekty jsou základním stavebním kamenem pro uchovávání dat v dokumentu. Množinou PDF objektů lze reprezentovat bitmapové a vektorové objekty, barevné prostory, text, fonty aj. [6].

2.1.1 Základní objekty

V PDF můžeme najít celkem 5 základních objektů:

- **Celá a reálná čísla** - Celá čísla jsou reprezentována jako jedno nebo více desetinných čísel z rozsahu 0..9 se znaménkem + nebo - před číslem. Reálné číslo je celé číslo rozšířené o desetinnou část s ideálně jedním desetinným číslem (reálná čísla nelze popsat exponenciálním způsobem). Přesnost a rozsah celých a reálných čísel je definován jednotlivými implementacemi PDF. V některých implementacích platí pravidlo které přetypuje celé číslo na reálné po přesáhnutí předem daného rozsahu.

- **Řetězce** - Řetězec je reprezentován jako množina po sobě jdoucích bytů vepsaných mezi jednoduché závorky. Jako příklad lze uvést: *(Hello, World!)*. Pro zobrazení zpětného lomítka a jednoduchých závorek je potřeba před tyto znaky přidat zpětné lomítko pro jejich správné zobrazení v dokumentu. V tabulce 2.1 lze vidět využití zpětného lomítka pro zobrazení odřádkovacích znaků:

Sekvence znaků	**Význam**
$\backslash n$	*Line feed (LF)*
$\backslash r$	*Carriage return (CR)*
$\backslash t$	*Tab*
$\backslash b$	Backspace

Tabulka 2.1: Odřádkovací sekvence znaků

  Řetězce můžou být reprezentovány i jako sekvence hexadecimálních čísel vložených mezi znaky $<$ a $>$.
  Jako příklad lze uvést: *<4F6EFF00> →0x4F, 0x6E, 0xFF, 0x00.*

- **Jména** - Jméno je reprezentováno jako sloučení zpětného lomítka a řetězce (př. */Jmeno*). Za jméno se pokládá i zpětné lomítko bez řetězce. Pokud bychom potřebovali nadefinovat v dokumentu jméno, jenž bude obsahovat mezery, musíme do řetězce přidat i sekvenci znaků **#20**, jelikož v ASCII tabulce je hexadecimální hodnota 20 vyjádřena jako prázdný znak. Jména jsou case-sensitive, proto */Jmeno* a */jmeno* jsou rozdílná jména. Jeho využití v PDF je prosté, slouží jako klíče ve slovnících a pro definice složitějších (vícehodnotových) objektů.

- **Boolean (pravdivostní) hodnoty** - Logické hodnoty **true**/**false** a vyskytuje se v jednotlivých záznamech ve slovníku jako příznak.

- **Hodnota null** - Nabývá hodnot *f* (free) nebo *n* (use) a vyjadřuje, zda je objekt vyobrazen v dokumentu.

2.1.2 Složené objekty

Složený objekt je takový objekt, který obsahuje seřazenou/neseřazenou množinu základních objektů i množinu složených objektů.

- **Pole** - Pole je v PDF reprezentováno jako seřazená množina základních i složených PDF objektů (v poli může být uložen například i slovník nebo pole) nezávisle na typech (v poli lze uchovávat například řetězec a číslo zároveň). Hodnoty pole jsou vloženy mezi znaky [a].

- **Slovníky** - Slovník se skládá z množiny dvou hodnot: klíče a hodnoty, pomocí kterých se slovník namapuje. Klíč je reprezentován pomocí **jména**, zatímco hodnota může být kterýkoliv PDF objekt, povoleny jsou i slovníky nebo pole. Slovníky jsou uloženy mezi znaky « a ».
- **Datové proudy** - Datové proudy slouží především pro uložení binárních dat a skoro ve všech případech jsou zkomprimovány různými kombinacemi algoritmů, které jsou popsány v kapitole 2.2, proto datové proudy musí být zároveň i nepřímým odkazem. Skládají se ze slovníků a části binárních dat. Slovník je využit pro ukládání parametrů binárních dat, jako například délka binárních dat aj.

2.1.3 Linkovací objekty

PDF objekty můžou být různě velké. Pokud je objekt až příliš veliký, pak jsou v kódu dokumentu využity nepřímé odkazy. Na obrázku 2.1 si lze všimnout využití nepřímých odkazů ve slovníku.

```
<<
/Resources 10 0 R      <--- znak R reprezentuje nepřímý odkaz na objekt s ID 10 a gen. číslem 0
/Contents [4 0 R]
>>
```

Obrázek 2.1: Ukázka nepřímého odkazu

2.2 Komprese dat v PDF

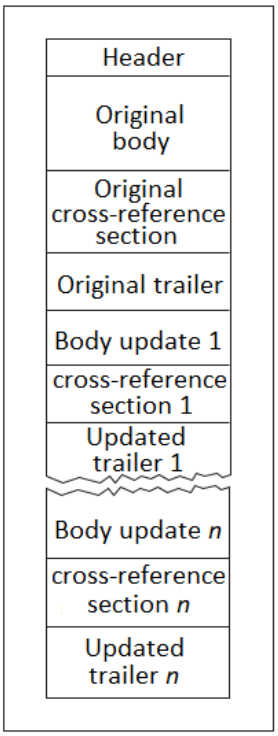
PDF soubory moho být poměrně kompaktní, o mnoho menší než ekvivalentní postscriptové soubory. Tato vlastnost je dosažena nejen lepší strukturou dat, ale i díky kompresním algoritmům, které jsou velice efektivní. Typ komprese dat PDF souboru lze zjistit pomocí textového editoru, který dokáže zpracovat binární data, vyhledáním klíčového slova **/Filter**. Níže jsou popsány kompresní algoritmy využívané v PDF [1].

- **CCITT G3/G4** - Algoritmus je bezeztrátový a využívá se pro vykreslení černobílých obrázků.
- **JPEG** - JPEG algoritmus může být jak ztrátový, tak i bezeztrátový. V Acrobatu se využívá pouze ztrátový s 5 stupni komprese. Využívá se pro barevné a šedotónové obrázky.
- **JPEG2000** - Rychlejší algoritmus na bázi JPEGu. Víceméně se nepoužívá, jelikož není kompatibilní se staršími systémy a vysokými nároky na procesor.

- **Flate** - Bezeztrátový algoritmus, vychází z kompresních algoritmů LZ77 a Huffmanova kódování.
- **JBIG2** - Alternativní k CCITT. V Dnešní době se nevyužívá z důvodu pomalejší komprese než je u jeho protějšku.
- **LZW** - Komprimací LZW algoritmem lze dosáhnout až o polovinu menší velikosti díky komprimaci veškerého textu a operátorů v souboru.
- **RLE** - Bezeztrátový algoritmus pro vykreslování černobílých obrázků. Nahrazen efektivnějším algoritmem CCITT.
- **ZIP** - Bezeztrátový algoritmus, učinější než jeho protějšek LZW.

2.3 Vnitřní struktura PDF

Vnitřní reprezentace PDF souboru je rozdělena na sekce, které jsou znázorněny na obrázku 2.2.



Obrázek 2.2: Interní struktura PDF souboru

Z obrázku lze vyčíst, že se zde vyskytují 4 hlavní sekce: *Header, Body, Cross-reference a trailer*. Díky jedné z vlastností PDF formátu se při úpravě

souboru staré sekce neodstraní, místo toho se pouze na jeho konci vytvoří nové sekce [4].

- **Header** - Hlavička souboru je uložena na první řádce, obsahující primárně použitou verzi PDF.

```
%PDF-1.4  <---- Hlavička souboru
%âãÐÓ
```

Obrázek 2.3: Ukázka hlavičky

- **Body** - V těle dokumentu jsou uložena veškerá data objektů reprezentující celý dokument. Objekty jsou referencovány v tabulce Cross-reference z důvodu rozprostření částí dat patřících k danému objektu po celé sekci.Pokud se v dokumentu vyskytuje jeden obrázek/zvukový záznam vícekrát než jednou, tak se poté všechny objekty reprezentující obrázky odkazují na jedny data [3].

```
4 0 obj    <--- start objektu
<</Filter /FlateDecode /Length 1882>>
stream     <--- data objektu
xśíśßoÚ6□ÇąWżî□ °ŧ´ŔĂŠ□ő□□öĐiĐ□CÓĆŘ□Ö=ŘŽăx‰“ÍVšv ýŽ§□-Ďrě,>#0□#’,QüńîH~(S’üµ-_ĽŃRk□□□lźĚŁv+ß□
ĐÁŘöq9Â\,†&¯−□ĚD□□+%Íj□*]Xfč-ń□Íźkeş€ń□pŮ§□Aú±,Ä□C``Ž, ţ ®tó/ŘÍ-óąfÓŃ?ŕŘ□ú®¤ë+ę"z.ónžm0R|~V□ö
endstream  <--- konec dat objektu
endobj     <--- konec objektu
```

Obrázek 2.4: Ukázka dat objektu

- **Cross-reference table** - Jinak nazývána **xref** je tabulka obsahující reference na veškeré objekty uložené v těle a v kódu začíná řetězcem *xref*. Reference uložená v tabulce je reprezentována na 2 řádcích pomocí řetězce a skládá se z 5 částí o celkové velikosti 20 bytů včetně oddělovačů *CRLF*:
  - *Číslo objektu* - Jednoznačný číselný identifikátor objektu.
  - *Počet subobjektů* - Počet částí daného objektu vyskytujícího se v dokumentu.
  - *Začátek objektu* - Tvoří většinu řetězce (prvních 10 bytů) a určuje offset od začátku PDF dokumentu až po začátek daného objektu.
  - *Generační číslo objektu* - Vyjadřuje jak často byl objekt vymazán při úpravě dokumentu.
  - *Identifikátor využití* - Nabývá hodnot *f* (free) nebo *n* (use) a vyjadřuje, zda je objekt vyobrazen v dokumentu.

```
xref       <--- start tabulky
0 1        <--- ID objektu a počet subobjektů
0000000001 65535 f
31 1
0000423765 00000 n
```

Obrázek 2.5: Ukázka jednoduché xref tabulky

- **Trailer** - Trailer je seznam informací, ze kterých lze snadno zjistit například velikost nebo umístění xref tabulky. Trailer může obsahovat tyto elementy:
  - *Size* - Udává počet objektů referencovaných v xref tabulce.
  - *Prev* - Offset od začátku dokumentu k předchozí xref tabulce.
  - *Root* - Odkazuje na objekt obsahující informace ohledně katalogu xref tabulek.
  - *Encrypt* - Specifikuje komprimující algoritmus použití pro daný dokument.
  - *Info* - Obsahuje dodatečné informace ohledně katalogu xref tabulek.
  - *ID* - 2-bytový identifikátor PDF dokumentu.
  - *XrefStm* - Offset od začátku dokumentu až k dekódovanému xref streamu. Využívá se pouze u hybridně-referencovaných souborů pouze tehdy, kdy hledaný objekt není nalezen v xref tabulce (před tím, než se volá element *Prev*).

```
trailer        <--- start traileru
<<
/Size 742      <--- velikost xref tabulky
/Root 741 0 R  <--- odkaz na objekt odkazující na objekt katalogu dokumentů
/Info 740 0 R  <--- odkaz na informační slovník dokumentů
/ID [<009feb05c3e899ac1d26612f86bb56aa> <009feb05c3e899ac1d26612f86bb56aa>]  --->
<--- identifikátor souboru
>>
startxref    <--- offset tabulky xref
408764
%%EOF
```

Obrázek 2.6: Ukázka traileru

2.4 Formuláře v PDF

3 Výsledky testování modulu

4 Závěr

Literatura

[1] *Compression in PDF files* [online]. Prepressure, 2017. [cit. 2017/01/05]. Dostupné z: `https://www.prepressure.com/pdf/basics/compression`.

[2] Christenssion, P. *PDF Definition* [online]. TechTerms. Sharpened Productions, 2018. [cit. 2018/04/05]. The Tech Terms Dictionary. Dostupné z: `https://techterms.com/definition/pdf`.

[3] King, J. *Introduction to the Insides of PDF* [online]. Adobe, 2005. [cit. 2005/04/29]. Dostupné z: `https://www.adobe.com/technology/pdfs/presentations/KingPDFTutorial.pdf`.

[4] Lukan, D. *PDF File Format: Basic Structure* [online]. InfoSec, 2018. Dostupné z: `https://resources.infosecinstitute.com/pdf-file-format-basic-structure`.

[5] Rouse, M. *Portable Document Format (PDF)* [online]. TechTarget, 2010. [cit. 2010/05/20]. Dostupné z: `https://whatis.techtarget.com/definition/Portable-Document-Format-PDF`.

[6] Whitington, J. *PDF Explained, Chapter 3. File Structure.* O'Reilly Media, Inc., 2011. ISBN 9781449310028.