

TEI and Microsoft: a marriage made in...: Using proprietary tools for producing standardly encoded editions

Tomaž Erjavec
Dept. of Knowledge Technologies
Jožef Stefan Institute
Ljubljana, Slovenia
tomaz.erjavec@ijs.si

1. XML and editorial interventions

It is generally assumed that scholarly annotated digital texts, be they text-critical editions or linguistically annotated corpora, should be stored in XML to ensure longevity as well as platform and software independence. Furthermore, the TEI Guidelines (Sperberg-McQueen and Burnard, 2002) are the de-facto standard for constructing the XML schemas for most such texts. But while XML is well-suited for machine processing, it is less than ideal for authorial or editorial interventions into the text, esp. when used with the complex TEI-derived schemas. It is of course possible to edit XML documents directly in a plain text editor, or better yet in specialised XML editors that support on-the-fly validation against a schema and schema-dependent drop-down menus. But, depending on the text type and required manual interventions, these generic editors are too clumsy for extensive work, and do not enable complex constraints on the allowed content and changes in the annotations. An additional problem with using XML editors for editorial work is the fact that many humanities scholars or students who are most likely to be doing this work have no knowledge of XML or TEI or any experience in editing it. This problem becomes all the more relevant in collaborative projects in which, say, a large number of students are hired to annotate a certain text or text collection. The effort required to first teach them how to use an XML editor and the underlying concepts might be prohibitive, and saving time necessary to perform each editorial intervention is essential.

Such problems of manual intervention can be resolved as they had been before the advent of XML: by developing specialised editing programs for the task at hand which store the data in the required format, and are optimised to perform validity checking and enable fast and easy editing of particular texts types or annotations. But such development is expensive in programming time, (human) editors need to be taught the specifics of the system and last but

not least, the program might need to be installed on many different computers or on a Web server for which an uninterrupted internet connection is required.

On the other hand, standard desktop office editors are very versatile, can be easily configured for particular tasks, and most computer users are literate in their usage and already have them installed. In the context of scholarly text editing and annotation, two editors are especially relevant:

- Microsoft Word, the most used text editor, allows easy text authoring (e.g. spell-checking) and editing (e.g. hot keys), definition of complex document structures (sections, tables, notes), has good support for Unicode, allows the inclusion of graphics, etc.; and
- Microsoft Excel, a widely used spreadsheet editor, allows sorting, content-dependent formatting, cell protection, drop-down menus, multiple sheets, etc.

Both editors have applications in scholarly digital texts production: Word is appropriate for text authoring and editing and simple alignments (e.g. between the transcription and its facsimile), while Excel can be usefully employed for linguistic annotation, especially lemmatisation and part-of-speech tagging.

The missing link is, of course, the transformation from the file format of the respective editor to the format defined by a specific (TEI) XML schema. The implementation of such a transformation brings about two problems:

- how to parse complex input files the formats of which are under the control of a software company, hence without the necessarily accessible specifications and with no guarantees on modifications from one version of the software to another; and
- how to reconcile the very free (“visual”) structure allowed by the editors to the much stricter (“semantic”) XML schema controlled output.

This paper explains how we overcame these two hurdles in the otherwise appealing scenario: humanities editors are free to use tools they are familiar with (Word, Excel), no time investment into project specific (computer) editors is required, yet the final digital edition is stored in a standardised, well-documented and processable format, the TEI XML.

2. The conversion Web service

The conversion architecture is centred on a Web service which takes as input (possibly a combination of) Word and Excel documents and returns XML and HTML documents. The conversion consists of:

1. parsing the input documents;
2. converting (and merging) them to a TEI document;
3. validating the TEI document;
4. converting the TEI document to HTML;
5. (converting the TEI document to Excel);

Step 1, the first hurdle mentioned above, would have been much more difficult (if not impossible) to overcome some years ago, as we would need to develop or obtain software to parse native editor formats (RTF, Excel) or fall back on the lowest common denominator (plain text, tab separated file) as well as deal with encoding problems. Now, however, many applications offer XML formats for their data. Both Word and Excel (at least in the Office Professional edition) support saving and opening documents as XML. This gives us the necessary window into the source, as XML documents are easy enough to reverse-engineer, even without explicit and documented schemas. Possible changes in the format between versions are hence also relatively easy to accommodate.

XSLT is used to convert between documents of one XML schema to another. Step 2 typically consists of a pipeline of XSLT transformations which convert the source XML into a simple TEI and from there into the project-required TEI encoding.

Step 3 is the syntactic validation of the created TEI by a simple XML validation against the TEI schema. Step 4 produces a “readable” version of the document. On the one hand, this has to be done for end-users (of digital libraries), but it is also of crucial importance in overcoming the second hurdle mentioned above, i.e. how to reconcile the unconstrained and presentation-oriented nature of documents, especially the ones created in Word, with the strict and interpretative TEI schemas. Namely, the HTML format, esp. with the generated table of context, indexes, use of colour etc. gives the humanities editors the feedback they need in order to validate whether their source documents are indeed well-formed; only if the structure and annotations are correct in the produced HTML, are they valid in the source. While, as

will be discussed below, good guidelines are still needed, the proposed approach also enables self-correction and self-teaching of editors who ultimately produce an exact TEI document.

Finally, Step 5 is used in certain scenarios to generate Excel (XML) documents which then serve as input to the editing process, and are uploaded to the server after they had been corrected. Such automatically generated Excel documents can be quite sophisticated using a simple trick: a template Excel document is created by hand, with certain cells containing “hooks”, and stored as XML. The conversion then takes this template and replaces the hooks with actual data from the TEI document, duplicating the rows as necessary.

The web service runs under Linux/Apache, using CGI/Perl. The Perl script:

1. takes the uploaded file, possibly compressed, with the archive containing multiple files;
2. calls various transformations with user-selected parameters;
3. returns the result, either directly via HTTP or as an archive file; and
4. logs each transaction, possibly archiving the input and output files.

This architecture has the following characteristics:

- it enables the editors to work with familiar and powerful tools yet produces TEI conformant output;
- it allows for a gradual learning process and step-wise refinement of the target documents;
- the data is standards-compliant (TEI, XML, (X)HTML, XSLT);
- the software components are Open Source, (Linux, Apache, Perl, libxml); and
- it is not very difficult to modify for new projects.

3. Practical implementations of the Web service

So far, the presented web service has been used in three projects / editions:

1. The eZISS (Erjavec and Ogrin, 2005) digital edition of the Collected Poems of Anton Podbevšek (1898–1981), who was a central figure of the Slovenian literary avant-garde. The edition contains facsimiles of the manuscripts and the author's emended copy of *The Man with Bombs*, transcriptions of the poems, editorial comments, notes and indexes. The variant poems and facsimiles are inter-linked. This edition has been

completed and is available in HTML and TEI under the Creative Commons licence at <http://nl.ijs.si/e-zrc/podbevsek/>.

2. The AHLib digital library / corpus of XIXth century Slovene books. Each book in AHLib is represented by the facsimile and a structured diplomatic transcription, hand-corrected from OCR. The text is automatically lemmatised, using the methods described in Erjavec et al. (2005), and then corrected manually. This is still work in progress, although the Web interface has already been implemented, as well as a “debug” version of the TEI to HTML conversion.

3. The JOS morphosyntactic specification and hand-annotated corpus. JOS is meant as a widely-available resource for language technology research into Slovene, and we aim to hand annotate a 1M word corpus with morphosyntactic information according to the JOS specification, itself a XML/TEI document. The JOS corpus is based on the FIDA+ automatically annotated 600M reference corpus (<http://www.fidaplus.net>) of present-day Slovene according to the MULTEXT-East multilingual morphosyntactic specification (<http://nl.ijs.si/ME/>). The work on this project is just beginning, and a preliminary Web interface for manual tagging has so far been implemented, as well as the conversion of the RTF specifications into TEI and HTML.

For each project, more details about the configuration and the use of the web service are given next.

3.1 eZISS

The Web interface in this case supports uploading of the RTF file (the complete Collected Poems), and display or download of the derived TEI and HTML files; a screenshot from the HTML file is given in Figure 1. In this setting, Word is used primarily as an authoring environment. This was the first application of the interface, and it served as valuable experience for further development. In particular, it soon became apparent that quite exact guidelines are needed to enable the production of sufficiently constrained Word documents to enable further processing. This is why editors in all the subsequent projects have been given short course, as well as written, quite specific guidelines about what and how to annotate the source document, together with a Word dot file containing the styles used in the project.

3.2 *AHLib*

The AHLib Web interface, as illustrated in Figure 2, is used for correcting two types of errors. The first are errors in the text itself for which each book must be proof-read from the OCR original. At this stage text structure is annotated as well, e.g. headings (divisions), footnotes, figures, page breaks (for alignment with the facsimile), foreign language passages, critical corrections (in case of typos in the original), etc. The second type of errors concerns linguistic annotation. Each word token in the text is first automatically lemmatised and then this lemmatisation is corrected by hand.

The set-up and the intended workflow in this application are rather complicated, mainly due to the fact that there is no simple way of splitting these two annotation types into two separate stages. In particular, checking the lemmatisation often reveals overlooked OCR errors in the text which can only be corrected by going back to the RTF. A further problem is that the automatic lemmatiser (a machine learning program, coupled with a trainable tagger) has been trained on contemporary Slovene which differs considerably from the (non-standardised) language of a century ago. Therefore the lemmatiser consistently makes errors with certain archaic words.

We solved these problems by splitting the process into three stages, allowing for multiple file input and output, and up- or downloading partially corrected files:

1. The user uploads the RTF file and receives either the TEI or HTML; an example is given in Figure 3. This stage is appropriate for structuring the document and initial proof-reading.
2. The user uploads the RTF or TEI file which is automatically lemmatised and the (structured and) lemmatised version returned as TEI or HTML; an example lemmatised file is given in Figure 4. More importantly, the document lemmas are checked against a large dictionary of Slovene. The unknown lemmas are returned with word-forms and context from the corpus in an Excel table, as illustrated in Figure 5. The table is then manually checked: in case a word is a typo, it is corrected in the RTF file and deleted from the Excel table; if a word is lemmatised incorrectly, its lemmatisation is corrected, and the corrected Excel table is uploaded to the service where it serves as a gold-standard lexicon for the lemmatisation of further texts. It is also possible to perform this process cyclically by submitting the RTF/TEI files and the (partial) Excel table of unknown words together.

3. The user uploads the RTF or TEI documents, and the complete lemmatised text is returned as an Excel table, as illustrated in Figure 6. In it, the user has to check / correct the lemmatisation of each token, and finally submit the RTF/TEI together with the corrected Excel in order to arrive at the final structured and linguistically annotated TEI. Again, cyclic improvement is possible by submitting the RTF/TEI together with partial Excel. This step is slightly more complicated as Excel imposes an upper limit of 64,000 rows per table, while a book can have more than that number of words. We therefore support the download of multiple Excel files, each containing a portion of the book. Furthermore, the user has the option of retrieving the Excel in the text order or sorted alphabetically.

3.3. JOS

The JOS Web service is currently being developed for manual morphosyntactic tagging correction and tagset definition. The corpus already contains automatically assigned morphosyntactic descriptions (MSDs) and lemmas, and the annotators should be able to correct these as efficiently as possible. As Slovene is a highly inflected language, the MSD tagset is quite large, around 2,000 tags. Apart from that, the work is to be preformed by students, which is why it is imperative to ensure easy-to-reach documentation, try to prevent mistagging, and indicate who corrected what, and when.

As the first step, a small corpus from FIDA+ was extracted by random sampling (»protoJOS«, 10,000 words). Next, its lean XML schema, giving annotations as attributes of word elements, was replaced by the one that can store possibly several, possibly ambiguous annotations with each token, and also gives their origin. This XML/TEI format is illustrated in Figure 7. This corpus is stored as a database, and the user of the Web service can choose which parts she wants to annotate (currently, selection by MSD prefix is supported) and then downloads the Excel files containing the automatically annotated text. Each Excel contains several sheets, one with the annotation guidelines; one with a complete list of all MSDs with their expanded names and frequencies from the FIDA+ corpus; a front page (illustrated in Figure 8), where the annotator's name is stored together with some summary statistics (all tokens, corrected tokens, etc.); and, of course, the sheet with the MSD and lemma tagged tokens, as shown in Figure 9. For known words, the selection of the correct tag is performed via a drop down menu, where all the lexical MSDs of the token are listed. The spreadsheet has other bells and whistles, such as highlighting the corrected annotation and »focus« words, giving the expansion of automatically assigned MSDs, the KWIC for the word, etc.

After they have been corrected, the Excel files are uploaded and joined into the XML database; each token contained in the spreadsheet simply receives a new annotation in the database. Currently two HTML views of the corpus are provided – the first one highlights different PoS with different colours, while the second displays all the annotations on each token (in a pop-up window, including the annotator’s name and date of the annotation).

The idea behind this somewhat complex set-up is that, ultimately, the annotators will not receive complete stretches of text to be annotated but rather only the tokens with a high probability of having a wrong MSD or lemma, thus maximising the effectiveness of the manual annotation.

The other usage of the architecture in JOS is the conversion of morphosyntactic specifications from Word to TEI. As mentioned, the Slovene MSD tagset is large, and MSD tags are structured, so that they represent feature structures. The definitions of the part-of-speech dependent attributes and their values, their mapping to MSD tags, constraints on their well-formedness, and possibly localisations and mappings to other tagsets, as well as commentary text and notes, are enshrined in the morphosyntactic specifications. In FIDA+ and other corpora we have used the MULTTEXT-East specifications which already provided a mapping to TEI feature structures, but the source was stored and edited as a LaTeX file. The specifications for JOS, currently under development, have been significantly changed from the multilingual MULTTEXT-East specification with a view of arriving at an optimal morphosyntactic tagset for Slovene. The specifications have also been recast in Word with the formal parts written as tables. An XSLT script transforms the Word/XML specifications to a canonical TEI encoding as shown in Figure 10, and checks it for consistency. From this formalised (but still tabular) and non-redundant canonical TEI, two further TEI documents can be generated, each one with a dedicated XSLT script. The first contains the TEI features structure library that should be a part of any corpus using the MSDs, while the second is a reader-friendly TEI version, with generated indexes and cross-references which can be turned into HTML with generic XSLT stylesheets for TEI, illustrated in Figure 11.

4. Conclusions

The paper has presented an environment for manual interventions into XML-based scholarly editions. The basic assumption is that it is easier for authors/editors/annotators to use generic and readily available editors than to edit the XML as well as faster for computer linguists to write or modify XSLT scripts from the XML produced by editors to TEI than to

develop specialised editors for particular projects. The architecture relies on a Web service that transforms input documents into a standardised format, validates them syntactically, and returns them for semantic validation. The approach was illustrated in a setting in which Word and Excel are used for authoring or correcting the base text and word-level linguistic annotation respectively.

Our experience with the presented approach shows that it is important to give annotators a tutorial and detailed guidelines, and that the approach is mostly appropriate for shallow encodings. For example, trying to unambiguously represent nested annotations in Word (e.g. a correction consisting of a deletion and addition) or cross-references is very difficult, and complex XSLT transformations are needed to capture this information in TEI. We therefore see the usefulness of this approach esp. in collaborative projects with each annotator investing minimal time in training and annotation. Such an annotation scenario is becoming increasingly popular in the HLT community, and wider. So, for example, Mihalcea and Chklovski (2004a, 2004b) describe their “Open Mind Word Expert” site where e.g. student contributors are presented with a set of natural language (e.g., English) sentences that include an instance of the ambiguous word and are asked to indicate the most appropriate meaning with respect to the definitions provided, thus building a word-sense disambiguated corpus. Similarly, Good et al. (2006) report on an experiment where volunteers untrained in knowledge engineering developed a partial ontology via a Web interface. In a non-HLT context, a distributed approach to annotating is used by the Mechanical Turk (<http://www.mturk.com/>) by Amazon (also dubbed “Artificial Artificial Intelligence) where humans are paid to classify instances of e.g. pictures or texts into predetermined categories. Such tasks are posted to the Turk by companies that need large annotated datasets, typically for training machine learning systems. Both examples above are much more sophisticated than ours in terms of the number of users they allow, but much simpler in the kinds of annotations they envisage – the main difference is that they directly employ a web interface, while our architecture assumes off-line editing in Word or Excel.

As could be noticed, our approach is built around the notion of open standards and software, so a common question is why we have opted to support Microsoft Word and Excel, rather than their open source Open Office (<http://www.openoffice.org/>) equivalents, OO Writer and OO Calc. The reason is simple: most of the editors and annotators that we worked with, have Microsoft Office already installed on their computers, and are reluctant to install the OO suite and learn to use it. Also, in our experience, OO tools still lag behind Microsoft in

terms of usability. However, the use of the XML-based OpenDocument standard as the native format for OO applications is a significant advantage, so we might reconsider our decision in future version of the Web service.

In our further work we also plan to address the question of version control, which is currently lacking in our system, to enable multiple editors to correct a set of documents without the danger of conflicts.

References

Erjavec, T., Ogrin, M. (2005) Digitalisation of literary heritage using open standards. In Paul Cunningham, Miriam Cunningham (eds.). *Innovation and knowledge economy: issues, applications, case studies, (Information and communication technologies and the knowledge economy)*. Amsterdam [etc.]: IOS Press, 999-1006.

Erjavec, T., Ignat, C., Pouliquen, B., Steinberger, R. (2005) Massive multilingual corpus compilation: ACQUIS Communautaire and totale. In Proc. of the Second Language Technology Conference. April 2004, Poznan.

Good, B. M., E. M. Tranfield, P. C. Tan, M. Shehata, G. K. Singhera, J. Gosselink, E. B. Okon, and M. D. Wilkinson. (2006) Fast, Cheap and Out of Control: A Zero Curation Model for Ontology Development *Pacific Symposium on Biocomputing II*.
<http://psb.stanford.edu/psb-online/proceedings/psb06/good.pdf>

Mihalcea, R. and T. Chklovski, Teaching Computers: Building Multilingual Linguistic Resources with Volunteer Contributions over the Web, in *The LISA Newsletter - Globalization Insider*, September 2004.
http://www.lisa.org/archive_domain/newsletters/2004/3.3/mihalceaChklovski.html

Mihalcea, R. and T. Chklovski, Building Sense Tagged Corpora with Volunteer Contributions over the Web, book chapter in *Current Issues in Linguistic Theory: Recent Advances in Natural Language Processing*, Editors Nicolas Nicolov and Ruslan Mitkov, John Benjamins Publishers, 2004.

Sperberg-McQueen, C. M. and L. Burnard, (eds.) (2002). Guidelines for Electronic Text Encoding and Interchange, the XML Version of the TEI Guidelines. The TEI Consortium,
<http://www.tei-c.org/>

Figures

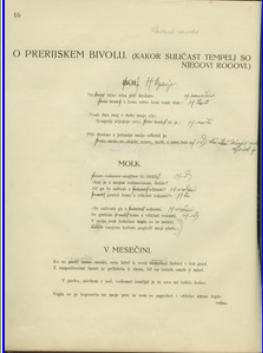
**Figure 1. eZISS Project; Collected Poems of Anton Podbevšek.
HTML view showing facsimile, with two transcriptions**

http://nl.ijs.si/e-zrc/podbevsek/html/pdb.2.2.html#pdb.2.2.4.8

Korekture JOS e-Podbevšek: 2. Pesniška zbirka ...

O prerijskem bivolu.

Bol, Molk, V mesečini

2.3. Faksimile ročno popravljene izvoda	2.2. Prepis	2.4. Kritični prepis ročno popravljene izvoda
<p>Bol, Molk, V mesečini ↑</p> 	<p>BOL. ↑</p> <p>Na moje okno trka ptič droban: "Bela ženka s koso ostro kosi vsak dan." "Vsak dan maj v dušo mojo sije. Kragulji kljujejo srce, bele ženke ni je." Ptič droban z jutranjo zarjo odletel je. Dušo mojo so objele sence, molk v njej bel je.</p> <p>MOLK. ↑</p> <p>Breze volnene srajčice so slekle. "Kaj je z mojim rožmarinom, dekle? Ali ga še zalivaš z bolnimi solzami? V noči grabiš temo z vitkimi rokami?" "Ne zalivam ga z bolnimi solzami. Ne grabim v noči teme z vitkimi rokami. V moje srce bolečine legle so in mraki. Kakor ranjene košute pogledi moji plahi."</p> <p>V MESEČINI. ↑</p> <p>Ko so padli name mraki, sem hitel k svoji tenkolasi ljubici v črn grad. Z razpuščenimi lasmi je prihitela k oknu, bil na ustnih smeh ji mlad. V parku, zavitem v noč, vodomet šumljal je in srce mi tolklo bolno. Vrgla se je koprneča na moje prsi in sem se zagrebel v obleke njene toplo volno.</p>	<p>UPANJE. ¹³ ↑</p> <p>Na zamreženo okno trka ptič droban: "Smrt s koso ostro kosi vsak dan." "Vsak dan maj v dušo mojo sije. Kragulji kljujejo srce, smrti ni je." Ptič droban z jutranjo zarjo odletel je. Na steni češnjev cvet obvisel je.</p> <p>MOLK. ¹⁴ ↑</p> <p>"Kaj je z mojim rožmarinom, dekle? Ali ga še zalivaš z vročimi solzami in grabiš temo z vitkimi rokami?" "Ne zalivam ga z vročimi solzami. Ne grabim teme z vitkimi rokami. Ko ranjene košute pogledi moji plahi, v mojem srcu bolečine so in mraki."</p> <p>V MESEČINI. ¹⁵ ↑</p> <p>Ko se je zmračilo, sem hitel k svoji dolgolasi ljubici v črn grad. Z razpuščenimi lasmi je prihitela k oknu, bil na ustnih smeh ji mlad. V parku, zavitem v noč, vodomet šumljal je in srce mi tolklo bolno. Vrgla se je koprneča na moje prsi in sem se zagrebel v obleke njene toplo volno.</p>

Find: Match case

Figure 2. AHlib Project; Web interface.

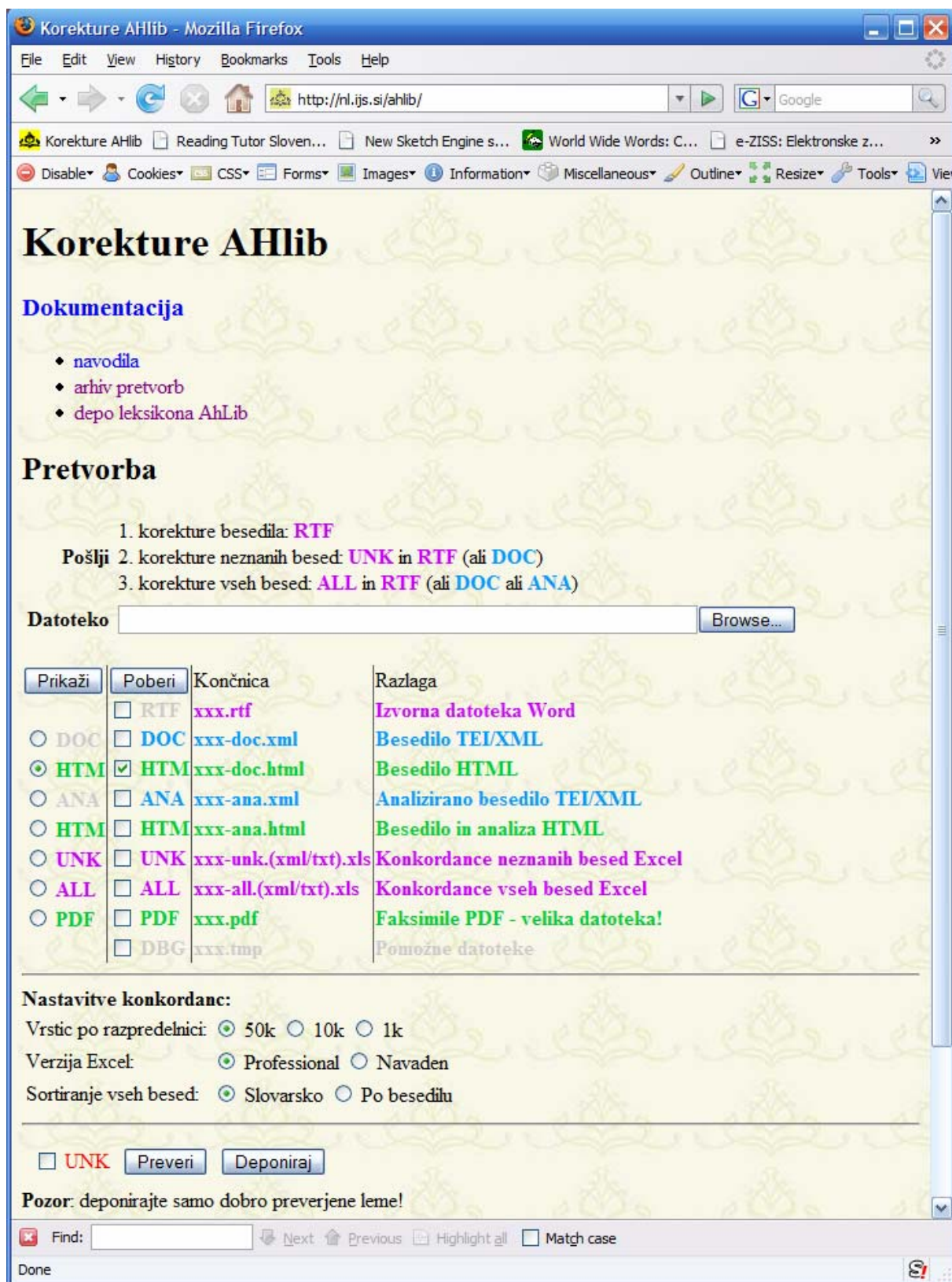


Figure 3. Ailib Project; HTML, start of text view.

http://nl2.ijs.si/ahlib/cgi/ahlib-cnv-v2

Index of /ahlib/doc/fa...
 (5.1) I.
 (5.2) II.
 (5.3) III.
 (5.4) IV.
 (5.5) V.
 (5.6) VI.
 • (6) B. Kanarčik.
 (6.1) I.
 (6.2) II.
 (6.3) III.
 (6.4) IV.
 (6.5) V.
 (6.6) VI.

Kazalo po straneh

01[000], 02[5], 03[5], 04[6], 05[7], 06[8], 07[9], 08[10], 09[11], 10[12], 11[13], 12[14], 13[15], 14[16], 15[17], 16[18], 17[19], 18[20], 19[21], 20[22], 21[23], 22[24], 23[25], 24[26], 25[27], 26[28], 27[29], 28[30], 29[31], 30[32], 31[33], 32[34], 33[35], 34[36], 35[37], 36[38], 37[39], 38[40], 39[41], 40[42], 41[43], 42[44], 43[45], 44[46], 45[47], 46[48], 47[49], 48[50], 49[51], 50[52], 51[53], 52[54], 53[55], 54[56], 55[57], 56[58], 57[59], 58[60], 59[61], 60[62], 61[63], 62[64], 63[??]

Besedilo (63 strani)

[000]



DVE POVESTI
iz pisem
Kristofa Šmida.
A. Golobčik.
B. Kanarčik.

p.1: Poslovenil A. P., bogoslovec
 p.2: v Ljubljanski duhovščini.
 p.3:
 Drugi natis.
 p.4: V LJUBLJANI.
 p.5: Natisnil, založil in na prodaj jih ima Jožef Blaznik, na brégi, Nr. 190.
 p.6: 1853.

[5][5]

I.

p.7: Na Sokólovski grajšini je živel pred več sto let serčni vitez Teobald s svojo pobožno ženó Otilijo. Vitez je bil ravno takó blaziga kot junaškiga serca. Za vse stiskance po celi deželi se je zló poganjal, pa zató še hvala ni hotel. Rádost, ljudi osrečiti, mu je bila že dovolj plačila. Otilija je réveže obilno obdarovala, je obiskovala bolnike po kóčah bližnjih dolin, in njen grad je bil zavetje vsim ubozim, ki so bili pomoči vredni. Tudi Neža, edina hči teh zalih staršev, gospodična per osmih létih, je bila zgolj dobrotna in perjaznost do ljudi. Zanjó ni bila večiga veselja, kot druge oveselováti. Staríse in hčer je vse spoštovalo, in kdor koli je visóki Sokólovski stolp ali turn od délec ugledal, je blagovával dobrotne, tam stanujúče ljudi. In gotovo je bil božji blagoslòv (ali žegen) práv vidama nad Teobaldam, Otilijo in Nežo. Desiravno so veliko razdelili, so vendar vsiga dovolj imeli, ker so bili smed nar_ bogatíših blagoródnih družin tiste dežele.

p.8: Bil je lep, jasen poléten dan, ko greste Otilija in Neža po obédí skosi vratíca in dvornim ozidjím po kamnitih stopnicah na vert, ki je bil navdól hriba. Z veseljem zagledate, kako lepó simkljati (ali plavkasti) vóhrovrt raste, in se vertnični popki prézajo; kako se bób ovija, in óšnjeje med začernélim perjem že rudéčijo. Nekaj

[6] časa postojíte per vodokóki v sredi verta, in se razveselujete nad igranjem vodé, ki je v solnčni svitlobi bistra ko golot (ali kristál) kviško vrela, in v kapljicah, pisanih ko mavrica, zopet spadávala. Po tem se usédete v lepó omréženo lopo, ob ktéri se vinske terté ovijajo, in pridno dodelujete oblačilo za néko siróto. Vse po verti je tiho in pokojno; le pénica je še verh bližnjiga drevesa prelepó popévala, in od vodokóka se je neprenéhama slíšalo perjetno vodno šumenje.

p.9: Nanáglima, de še viditi neste bile v stani, kaj de si je, nekaj v lopo perleti. Obé se stráhama pogledujete. Ko bi trenil, se velik ptíč z razprostertimi perutami pred lopo spusti, pa ljudi v njej zagledávši práč spet odleti. Neža je vsa plašna, de se ne gane; še ozréti se, si ne upa, kaj de bi bilo takó urno v lopo perletélo. Tode mati ji réče posmehljaje se: „Nikar se ne boj! Kaj če néki biti, kot kaki ptíček, ki je jástrobu ubéžal“. Pogléda in zavpije: „Glej, glej golobčika, beliga kakor snég! Ves prestrašen se ti je za herbét skril“. Ga prime, ter se v Nežo ozre in ji réče: „Vés kaj, za večerjo ti bom golobčika spekla“.

p.10: „Kaj spekli?“ réče Neža z zavzétjem, ter z obéma rokama golobčika popade, ravno kakor bi ga hotla zažugani smerti otéti: „Ne, ne, ljuba mati,“ ji réče, „to ni bila vsa resnica! Uboga živalica je k meni perbežala, kakó bi jo koli zamogla umoriti? Le poglejte, kakó je lép! Gotovo je bel ko sneg, in njegove nožice, le poglejte, kakó so lepó rúdeče kot koralde. Vite, kakó mu še sercé utripa! Z svojima nedolžnima očéscama me takó mlo pogleduje, kakor bi mi

[7] hotel réči: Nikar mi nič žaliga ne stóri! — Práv nič ti ne bom storila, ljuba živalica, nési se zastonj k meni zatekla. Per meni ti ima dobro biti“.

p.11: „Práv imaš, ljubi otrok“, ji mati perjazno réče. „mojo misel si zadela. Le skušiti sim te hotla. Nési golobčika v svojo stanico, in mu dai piče.

Figure 4. AHlib Project; HTML end of text and start of annotation view.

na toliko, da od obeda ostavšo meso u večer smejo pojesti, tudi tisti, kateri težko delajo, ino kateri po presodi vračitelja mesene jedi potrebujejo. Nikakor se nedopuša, da bi u takih dnevih, po kterih je mesena jed iz perzanesljivosti dovoljena, kak tudi ne po nedeljah svetega posta, k mesenim jedlom ribe se pristavljale.

p.18: Zapoved od mesojeje se zderžavati ne preteže se na bolene, tudi ne na siromake, kateri od darov dobrih ljudih živijo, ali pa drugač u velikem siromaštvu se znajdejo.

p.19: Kteri pa bi zbog posebnih zrokov obilnejšega dovoljenja potrebuval, si ga ma od škofivstva sprostiti.

p.20: Kteri se stem podeljenega perpuščenja poslužjo, imajo vsaki den, kada od mesenoga jejo, pet očanašov ino pet češena si Maria na čast britkemu terpljenju ino Jezusove smerti moliti, ali pa svojemu premoženju primerjeno vboğaime dati.

[8]

p.21: Kloštterski se imajo postiti, kuk njim njihove postave zapovedajo. Častitim duhovnikom se stem nalaže da to pastirsko pismo svojim farmanom na masno nedelju oznanijo, ino opomembe, ktere bi morebiti po različnih krajih potrebne bile, pristavijo.

p.22: Dano u Gradcu na nedelju septuagesime, 8. svečnja 1852.

p.23: Jožef Otmar, Knezškof.

p.24: Natisjen pri Kienreich.

Analiza

p.1

- ♦ Br.330/2br.330/2

p.2

- ♦ Leto leto 1852 .

head.1

- ♦ Jožef Jožef Otmar otmar , po milosti milost božji knezoškof Sekovskisekovski . oskerbnik oskrbnik Lujbnske lujbnski škofijješkofija , itd. itd. itd. vsemves vernimveren Sekovske sekovski ino in Lujbnske lujbnski škofijješkofija vseves dobrodober od Bogabog želimzeleti

p.3

- ♦ Kerščansko kerščanski prepričanje (vera) jebiti življenja življenje duh , kateri kateri vsega ves človeka človek prevzeme prevzem , ino in kateri kateri se očituje očitovati u vsakom vsak njegovom njegovnjeg djanju dejanje ino in tersenju tersenje .
- ♦ Vseves postave postava in naredbe naredba cerkvene cerkven imajo meti za nalogo nalog , da dati zbudijo zbuditi ino ina jačijo jačiti toto tot kerščansko kerščanski prepričanje , da dati ga on čistijo čistiti ino in branijo braniti ; ino in ravno zato se preteže pretežiti njihov uplív na vseves okolnosti okolnost posameznih posamezen ljudi človek , ino in skupčine skupčina .
- ♦ Od od tega ta se jasno jasen ino in popolnoma prepričamo prepričati , ako poglednemo pogledniti na svedkovanje nedelje nedelja , kak jebiti po prepisu prepis katoliček katoliček cerkve cerkev svetiti zapovedana zapovedati .

p.4

- ♦ Človek človek se mora morati skerbeti skrbeti skoz upotrebovanje svojih svoj telesnih telesen moči moč za potrebe potreba časnoga življenja življenje .
- ♦ Skoz skoz nevtrudljivo nevtrudljivo delavnost si se mora morati iz zemlje zemlja sad pridelati , iz sirovoga plodaplod njenoga kruh načiniti , kateri kateri ga on hrani hrani , oblačilo oblačiti stori , s katerim kateri se oblači oblačiti , ino in hram si se podignuti podignuti , pod katerim kateri najdenajti zavetje ino in strehu .
- ♦ Težavna težaven ino in dolgoterpa dolgoterpen reč jebiti , ktera katera velik velik trudatrud ino in napinjanja napinjanje stojštati , dokler se človeško človeški družtvo družtvo u dobrovredjeno dobrovredjen edinost spravi spravi tak , da dati eno en na drugem drug naslonjeno nasloniti skoz pravičnost vseves se ravna ravnati ino in brani braniti .
- ♦ Papa pri vsemves tem ta se terja terjati , da dati jebiti namen , ino in serce srce človeka človek od posvetnih posveten reči reč prosto prost , ino in obernjeno obmiti u višave višava nadzemeljske nadzemeljski , da dati vseves , kaj god človek poprime poprjeti ino in deladelo , tak deladelo , kak Bog bog hoče hoteti , ino in kajti jebiti njemu on tak dopadljivo dopadljiv ; drugač drugače se znebi znebiti zasluzbe , ino in jebiti sirota , ako ravno misli misiti , da dati jebiti bogat .
- ♦ Milostiva milostiv previdnost vsemogočega vsemogoč Bogabog jebiti podala podati človeku človek eden den za počitek , u katerim kateri se naj spomni spomniti z svojega svoj cilja cilj , u katerim kateri , oslobodjen od posvetnih posveten skerbih skerbi ino in poslov posel , se naj skerbih skerbi za večnost .

Figure 7. JOS Project; XM/TEI, Annotation.

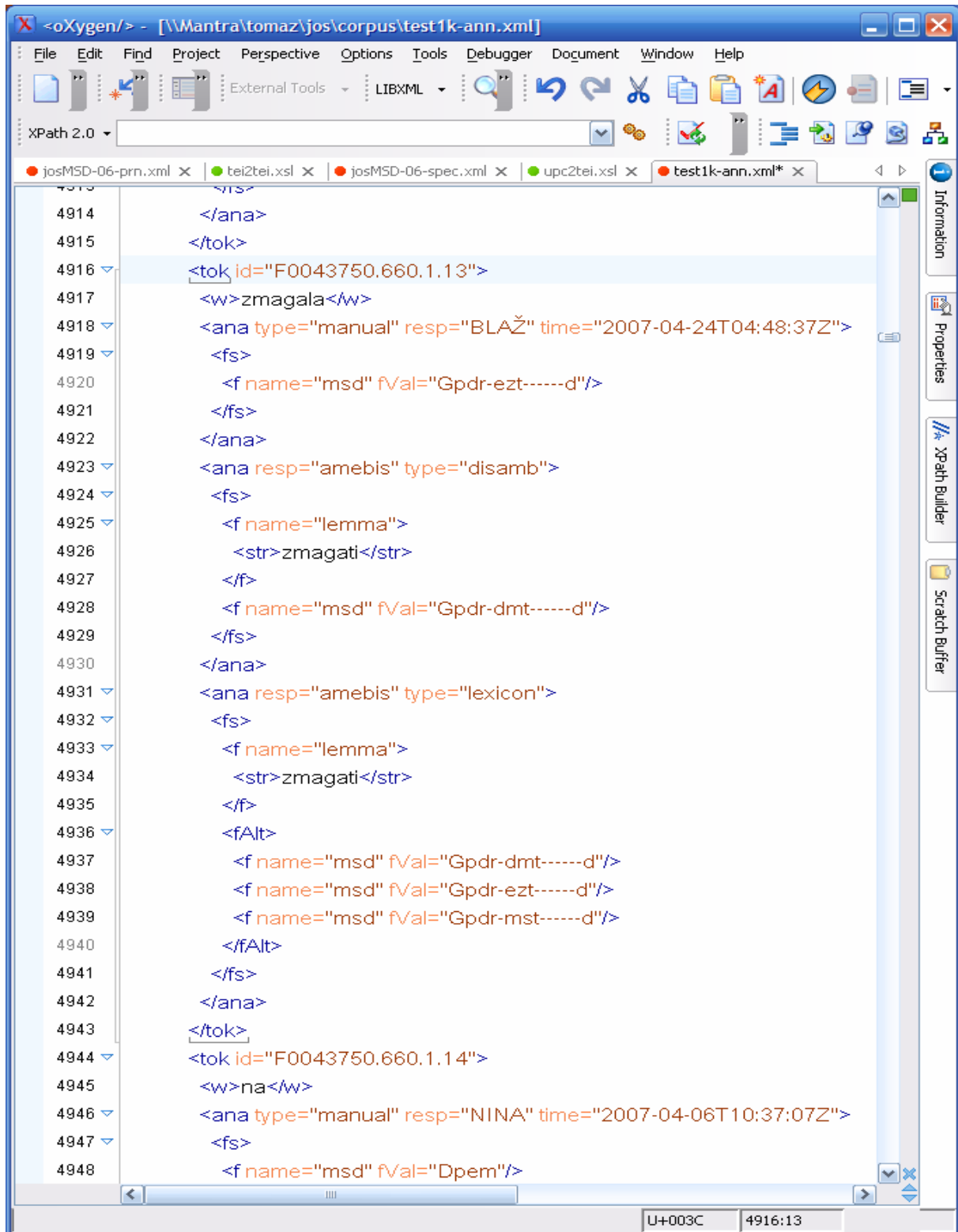


Figure 8. JOS Project; Excel, Front sheet.

ID	N/A Filter	G.*	Vrstic	
4				JOS ana2xls 0.1
5	2007-03-23	xml	Pojavnice	1.214
6	2007-03-12	Sortiranje	Vseh besed	1.007
7	2007-03-28	Rezina	Izbranih besed	185

Rezultati označevanja	
Popravljenih MSD	18
Popravljenih lem	2
Vnešenih opomb	-
Popravkov tokenizacije	N/A

Figure 9. JOS Project; Excel, Morphosyntactic Annotation Sheet.

A	C	D	E	H	I	J	K	L	
1	n beseda	lema	MSD	amb	Razvezano	Og	levi kontekst	beseda	desni kontekst
2	1 <div/>								
3	2 <p/>								
4	3 <s/>								
5	4 Slavko	Slavko	Slmei	2/3	moški ednina imenovalnik				
6	5 Dragovan	Dragovan	Pkomein	2	a imenovalnik -določnost				
7	6,								
8	7 župan	župan	Somei	1/2	moški ednina imenovalnik				
9	8 občine	občina	Sozer	3/5	ne ženski ednina rodilnik				
10	9 Metlika	Metlika	Slzei	1/2	nski ednina imenovalnik				
11	10 :								
12	11 Se	še	L	1	Členek				
13	12 nikoli	nikoli	Rso	1/7	Prislov splošni osnovnik				
14	13 me	jaz	Zop-et--d	3	naslonka samostalniški				
15	14 ni	biti	!Gvpste--d	3	ijk tretja ednina zanikani	očine Metlika : Še nikoli me	ni	bilo tako strah pod	
16	15 bilo	biti	!Gvdr--est	2	ijk ednina srednji tvorik	ne Metlika : Še nikoli me ni	bilo	tako strah podpisat	
17	16 tako	tako	Rso	3/7	Prislov splošni osnovnik				
18	17 strah	strah	Somei	2/3	moški ednina imenovalnik				
19	18 podpisati	podpisati	!Gpn-----	1	enski nedoločnik dovršni	nikoli me ni bilo tako strah	podpisati	kakšne pogodbe kc	
20	19 kakšne	kakšen	Zv-zer----	6/8	dnina rodilnik pridevniški				
21	20 pogodbe	pogodba	Sozer	3	ne ženski ednina rodilnik				
22	21 kot	kot	Dpei	6/8	g enostaven imenovalnik				
23	22 prav	prav	Rso	4	Prislov splošni osnovnik				
24	23 pogodbo	pogodba	Sozet	2	ne ženski ednina tožilnik				
25	24 za	za	Dpet	2/3	redlog enostaven tožilnik				
26	25 vrtino	virtina	Sozet	2	ne ženski ednina tožilnik				
27	26 .								
28	27 <s/>								
29	28 Podpisuješ	podpisovat	!Gppsde--n	1	ina nezanikani nedovršni			Podpisuješ nekaj , za kar de	
30	29 nekaj	nekaj	Zntset----	2/3	na tožilnik samostalniški				
31	30								

Figure 10. JOS Project; XML/TEI, Morphosyntactic Specifications.

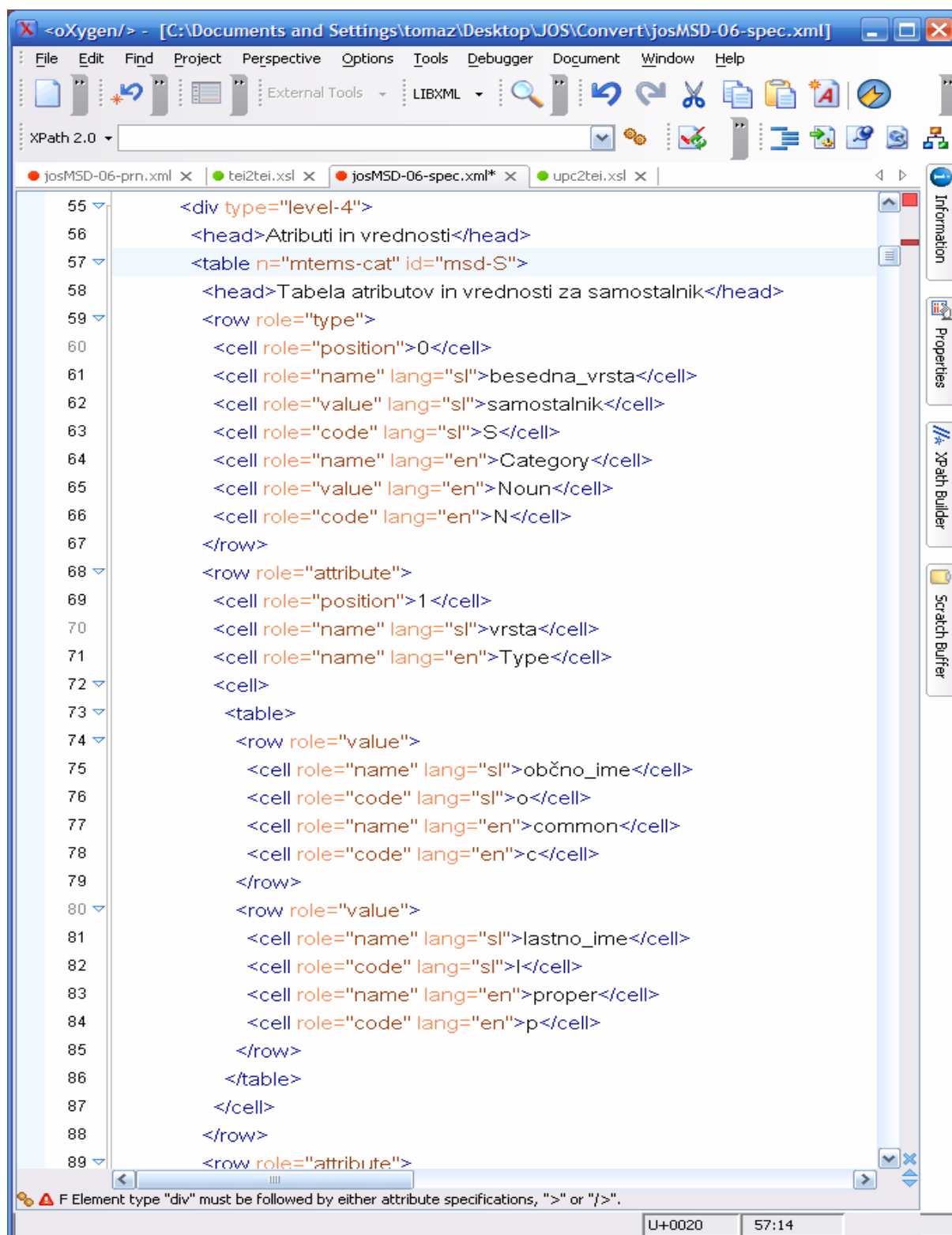


Figure 11. JOS Project; HTML, Morphosyntactic Specifications.

1.2.10. MEDMET
 1.2.10.1. Atributi in vrednosti
 1.2.11. OKRAJŠAVA
 1.2.11.1. Atributi in vrednosti
 1.2.12. NEUVRŠČENO
 1.2.12.1. Atributi in vrednosti
 1.2.12.2. Spremembe glede na FIDA/MULTEXT-East

1. Oblikoslovne specifikacije JOS

Zadnja sprememba 2007-05-27

1.1. Tabela besednih vrst

Se ni!

Splošne opombe:

1. kaj z vidom
2. preverit, da so oznake in atributi konsistentni po vseh besednih vrstah!
3. Tomaž, narediti naslednjo verzijo specifikacij
4. Tomaž et al: kako in kam namestiti avtomatsko preimenovanje MSD glede na lemo-besedo

1.2. Tabele atributov in vrednosti

1.2.1. SAMOSTALNIK

1.2.1.1. Atributi in vrednosti

Table 1. Tabela atributov in vrednosti za samostalnik

P	Atribut	Vrednost	K	Attribute	Value	C
0	besedna_vrsta	samostalnik	S	Category	Noun	N
1	vrsta	občno_ime	o	Type	common	c
		lastno_ime	l		proper	p
2	spol	moški	m	Gender	masculine	m
		ženski	z		feminine	f
		srednji	s		neuter	n
3	število	ednina	e	Number	singular	s
		dvojina	d		dual	d
		množina	m		plural	p
4	sklon	imenovalnik	i	Case	nominative	n
		rodilnik	r		genitive	g
		dajalnik	d		dative	d
		tožilnik	t		accusative	a
		mestnik	m		locative	l
		orodnik	o		instrumental	i
5	živost	da	d	Animate	yes	y