

RAPORTARE ȘTIINȚIFICĂ

Proiect complex ReTeRom. Proiect component CoBiLiRo

Activitatea A2.2: Raport asupra soluțiilor de armonizare a reprezentărilor colecțiilor existente text /vorbire (metadate și adnotări)

Faza de predare: iunie 2019

Autori: Ionuț Pistol, Diana Trandabăț, Andrei Scutelnicu, Cristian Pădurariu

Colaborator extern: Șerban Boghiu

1. Rezumatul etapei

A doua etapă (2019) a proiectului CoBiLiRO prevede realizarea infrastructurii pentru gestionarea și stocarea corpusului bimodal. Activitățile complementare prevăzute includ descrierea și implementarea unor soluții de armonizare a reprezentărilor colecțiilor existente text/vorbire (metadate și adnotări). Ulterior, aceste soluții urmează să fie utilizate pentru a armoniza formatele colecțiilor existente și a le încărca pe platforma CoBiLiRO. Similar celorlalte etape, este prevăzută o activitate de diseminare atât la evenimente științifice cât și în mass-media. De interes special pentru platforma dezvoltată este respectarea drepturilor de autor și a anonimizării solicitate pentru contributorii de resurse pe platformă.

2. Rezumatul activității

Activitatea 2.2 are ca obiectiv descrierea unor soluții de conversie automată a celor trei formate identificate ca fiind utilizate de contributorii actuali la formatul CoBiLiRO, descris în raportul activității A1.3. Conversia are ca scop facilitarea mecanismelor de căutare și comparare a resurselor contribuente pe platforma proiectului, precum și facilitarea utilizării formatului propus ca standard pentru resursele aliniate text-voce. Pe viitor, în cazul în care noi formate vor fi identificate, vor putea fi propuse procese de conversie similare.

3. Descrierea științifică și tehnică

3.1 Introducere

În etapa 2018 a proiectului, două dintre activitățile finalizate în cadrul CoBiLiRO au fost A1.2. *Inventarierea colecțiilor de date lingvistice românești disponibile la parteneri sau în terțe coalitii* și A1.3. *Proiectarea funcțională și arhitecturală a infrastructurii care va găzdui resursele și instrumentele de prelucrare și acces ale consorțiului și realizarea unui prototip*, ambele fiind descrise în raportul etapei anterioare. În urma acestor activități au fost identificate trei formate predominant utilizate în reprezentarea resurselor partenerilor. De asemenea, au fost identificate

cerințele platformei implementate în această etapă și a fost descris un format de reprezentare ce va fi folosit pentru punerea la dispoziție către utilizatori a resurselor contribuie. Pentru descrierea detaliată a formatului agreat de parteneri, ce va fi folosit pentru stocarea resurselor pe platforma CoBiLiRO, poate fi consultat raportul corespunzător activității A1.3 [3] aferent etapei 2018. În acest raport, acest format va fi referit ca formatul CoBiLiRo.

Acest raport continuă în Secțiunea 2 cu o descriere pe scurt a celor trei formate menționate. Secțiunea 3 propune soluții de armonizare pentru conversia acestor trei formate de intrare la formatul platformei. Procesele de conversie propuse țin cont și de eventualele incompatibilități sau omisiuni ale formatelor originale. Secțiunea 4 a raportului discută și posibile optimizări necesare pe serverul platformei CoBiLiRO în vederea facilitării conversiei automate a unui volum mare de fișiere.

3.2 Formatele folosite în colecțiile existente

Cele trei formate identificate în resursele partenerilor sunt: formatul PHS/LAB, formatul MULTEXT/TEI și formatul TEXTGRID.

2.1 Formatul 1 (PHS/LAB)

Formatul PHS/LAB este reprezentat de calupuri de câte 4 fișiere cu următoarele caracteristici:

- Un fișier *.wav* ce conține înregistrarea audio a intrării;
- Un fișier *.txt* ce conține transcrierea textului, rostit în cadrul înregistrării audio;
- Un fișier *.lab* ce conține varianta procesată a textului din fișierul *.txt*, în care sunt eliminate semnele de punctuație;
- Un fișier *.phs* care conține o aliniere la nivel de fonem a textului transcris. Pentru fiecare fonem este precizat momentul începerii rostirii lui, respectiv momentul terminării rostirii.

Fișierul *.phs* este structurat pe 4 coloane, astfel: (1) prima coloană reprezintă momentul începerii pronunțării fonemului; (2) a doua coloană prezintă momentul terminării rostirii fonemului; (3) pe a treia coloană se regăsește fonemul; (4) a patra coloană este prezentă doar în cazul fonemelor ce marchează începutul cuvintelor și cuprinde cuvântul întreg, ce urmează a fi rostit.

```
0 3700000 pau pau
3700000 4400000 p purt\304\203torul
4400000 4900000 u
4900000 5200000 r
5200000 5600000 t
5600000 6200000 @
```

6200000 6700000 t
6700000 7200000 o
7200000 7500000 r
7500000 8100000 u
8100000 8400000 l
8400000 8800000 d de
8800000 9200000 e

Textul de mai sus reprezintă un extras dintr-un fișier *.phs*. De remarcat este modul în care se salvează diacriticele. De exemplu, litera “ă” este reprezentată printr-o serie de caractere “\304\203”. Acest lucru poate genera o eroare la parsarea acestor fișiere, dacă nu sunt tratate corespunzător diacriticele.

Un dezavantaj al acestui format este faptul că nu prezintă metadate legate de sursa înregistrării sau detalii tehnice despre fișierul audio. Astfel, pentru a converti fișierele din acest format în codificarea agreată în raportul A1.3 [3], este necesară adăugarea acestor informații.

Fișierele *.wav* prezintă în general înregistrării la nivel de propoziție. Acest lucru este util, deoarece permite stocarea unei alinieri atât la nivel de propoziție, cât și la nivel de fonem, folosindu-se informațiile din fișierul *.phs*.

3.2.1 Formatul 2 (MULTEXT/TEI)

MULTEXT (*Multilingual Text Tools and Corpora*) este un proiect de cercetare finalizat ce și-a propus să sprijine dezvoltarea și utilizarea unor resurse multilingve prin proiectarea de standarde de adnotare, dezvoltarea de tehnologii de prelucrare și crearea unor corpusuri. Acest proiect a fost continuat prin MULTEXT-EAST [1] ce a avut aceleași obiective, de data asta însă pentru limbile vorbite în estul Europei. Una din resursele create în acest proiect este și un corpus aliniat voce-text ce va fi încărcat pe platforma CoBiLiRO. În acest context, acest format a fost analizat și a fost investigată o modalitate de armonizare a sa cu formatul agreat în raportul A1.3 [3].

În vederea adnotării corpusului aliniat, MULTEXT-EAST a plecat de la standardul propus de TEI 4.0 [2] pentru textul propriu-zis, la care a adăugat un marcaj specific pentru înregistrarea audio corespunzătoare.

```
<div id="sro.2" n="01" type="block">
<head>*BLOCK: 01</head>
<p id="sro.2.2"><s id="sro.2.2.1">Am o problem&#x103; cu aparatul de
dedurizare a apei.</s>
...</p>
<ab>[<xref url=" ../spc/spch01-ro.wav">speech file</xref>]</ab></div>
```

Spre deosebire de formatul CoBiLiRO, unde se preferă o aliniere unu-la-unu între fișierul text și cel audio, în acest format textul este asociat cu mai multe fișiere *.wav*, corespunzătoare

fiecărui paragraf. O resursă încărcată pe platformă în acest format va fi compusă dintr-un fișier *.xml* și un set de fișiere *.wav* cu scurte înregistrări audio. Fișiere audio nu au o calitate deosebită (doar 16kbps), dar conțin o singură voce, fără alte zgomote și au dimensiuni relativ mici (aproximativ 1MB pentru un paragraf). Header-ul TEI inclus în fișierul XML conține majoritatea informațiilor cerute în formatul CoBiLiRo, iar codificarea este similară. Conținutul de text și alinierea sunt cuprinse în elementul `<text>` ce conține și atributele `id` (un identificator unic pentru document) și `lang` (limba textului din document). Textul este segmentat în paragrafe, iar pentru a marca legătura dintre un paragraf și fișierul/fișierele audio este folosit elementul `xref`, definit în standardul TEI. Descrierea elementelor și atributelor este bine documentată, tagset-ul folosit fiind TEI 4.0. Formatul fișierului XML, exclusiv header-ul TEI, este prezentat mai jos.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:element name="TEI.2">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="text"/>
      </xs:sequence>
      <xs:attribute name="id" use="required" type="xs:NCName"/>
      <xs:attribute name="lang" use="required" type="xs:NCName"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="text">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="body"/>
      </xs:sequence>
      <xs:attribute name="id" use="required" type="xs:NCName"/>
      <xs:attribute name="lang" use="required" type="xs:NCName"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="body">
    <xs:complexType>
      <xs:sequence>
        <xs:element maxOccurs="unbounded" ref="div"/>
      </xs:sequence>
      <xs:attribute name="lang" use="required" type="xs:NCName"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="div">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="head"/>
        <xs:element ref="p"/>
        <xs:element ref="ab"/>
      </xs:sequence>
      <xs:attribute name="id" use="required" type="xs:NCName"/>
      <xs:attribute name="n" use="required" type="xs:NCName"/>
      <xs:attribute name="type" use="required" type="xs:NCName"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="head" type="xs:string"/>
  <xs:element name="p">
    <xs:complexType>
      <xs:sequence>
        <xs:element maxOccurs="unbounded" ref="s"/>
      </xs:sequence>
      <xs:attribute name="id" use="required" type="xs:NCName"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="s">
    <xs:complexType mixed="true">
      <xs:attribute name="id" use="required" type="xs:NCName"/>
    </xs:complexType>
  </xs:element>

```

```

</xs:complexType>
</xs:element>
<xs:element name="ab">
  <xs:complexType mixed="true">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" ref="xref"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="xref">
  <xs:complexType mixed="true">
    <xs:attribute name="url" use="required"/>
  </xs:complexType>
</xs:element>
</xs:schema>

```

3.2.2 Formatul 3 (TEXTGRID)

Cel de-al treilea format identificat, formatul TEXTGRID, este alcătuit din 3 fișiere:

- Un fișier *.wav* ce conține înregistrarea audio a intrării;
- Un fișier de tip *.TextGrid* care conține o aliniere a tuturor vocalelor din text. Rândurile trei și patru din fișier precizează momentul începerii rostirii și momentul terminării rostirii (exprimate prin două variabile globale - *xmin* și *xmax*, respectiv timpul de început și sfârșit). Ulterior, este marcat un vector de elemente (*items*), care sunt vocalele din text, fiecare având un interval - cu *xmin*, *xmax* și vocala transcrisă fonetic;
- Al treilea fișier este de tip *.txt* și conține locația fișierului, dimensiunea sa în kilobiți, data înregistrării, precum și un vector de valori care prezintă, pentru fiecare vocală identificată:
 - a. un identificator
 - b. durata în milisecunde
 - c. energia măsurată în decibeli
 - d. frecvența pentru formatul *F1*, *F2* și *F3* măsurată în Hz

Exemplu format TextGrid

```

File type = "ooTextFile" - tipul fișierului
Object class = "TextGrid" - formatul fișierului

```

```

xmin = 0
xmax = 1.8284375
tiers? <exists>
size = 1
item []:
  item [1]: - indicele colecției
    class = "IntervalTier"
    name = "voyelles"
    xmin = 0
    xmax = 1.8284375
    intervals: size = 17
    intervals [1]:
      xmin = 0
      xmax = 0.21698313449079956
      text = ""
    intervals [2]:

```

```
xmin = 0.21698313449079956
xmax = 0.3118133965778926
text = "e"
intervals [3]:
  xmin = 0.3118133965778926
  xmax = 0.3553420414703287
  text = ""
intervals [4]:
  xmin = 0.3553420414703287
  xmax = 0.5139106764356318
  text = "'la:"
intervals [5]:
  xmin = 0.5139106764356318
  xmax = 0.590085804997395
  text = ""
intervals [6]:
  xmin = 0.590085804997395
  xmax = 0.6926890393867089
  text = "a:"
intervals [7]:
  xmin = 0.6926890393867089
  xmax = 0.7517636288835865
  text = ""
intervals [8]:
  xmin = 0.7517636288835865
  xmax = 0.8792403746400065
  text = "'1\ef:"
intervals [9]:
  xmin = 0.8792403746400065
  xmax = 0.916550641690666
  text = ""
intervals [10]:
  xmin = 0.916550641690666
  xmax = 1.0160446871590916
  text = "\ic"
intervals [11]:
  xmin = 1.0160446871590916
  xmax = 1.095329004641743
  text = ""
intervals [12]:
  xmin = 1.095329004641743
  xmax = 1.2787711509741526
  text = "'la:."
intervals [13]:
  xmin = 1.2787711509741526
  xmax = 1.3689376296799132
  text = ""
intervals [14]:
  xmin = 1.3689376296799132
  xmax = 1.473095458529671
  text = "\sw:"
intervals [15]:
  xmin = 1.473095458529671
  xmax = 1.5197332923429954
  text = ""
intervals [16]:
  xmin = 1.5197332923429954
  xmax = 1.6254457156531976
  text = "i\nva:"
intervals [17]:
  xmin = 1.6254457156531976
  xmax = 1.8284375
```

```
text = ""
```

```
D:\Arad\9F75twp3.txt size: 29255 - locația fișierului  
28-Aug-2013 - data înregistrării
```

	duration [ms]	energy [dB]	fo1	fo2	fo3 [Hz]
1	95	62	203	202	200
2	159	62	199	193	191
3	103	64	191	183	182
4	127	62	180	171	167
5	99	61	173	161	162
6	183	55	168	168	185
7	104	62	295	305	297
8	106	58	262	259	259

values at:

```
3472 4230 4989 5685 6954 8223 9441 10262 11083 12028 13048 14068 14665 15461 16257 17525 18993  
20460 21903 22736 23570 24316 25161 26007
```

Pentru fiecare valoare de tip *interval*, se va genera tag-ul *unit* aferent, completat cu atributele *start* și *stop* luate din câmpurile *xmin* și *xmax*. Atributul *text* va fi mapat folosind câmpul *text* din fișierul *textgrid*.

Fiecare înregistrare de tipul :

```
intervals [1]:  
  xmin = 0  
  xmax = 0.21698313449079956  
  text = ""
```

va fi convertită la o înregistrare specifică *teiHeader*:

```
<teiHeader>  
  <unit>  
    <speech start="0.0000" stop = "0.21698313449079956"/>  
    <text></text>  
  </unit>  
</teiHeader>
```

3.3 Soluții de armonizare propuse

3.3.1 Formatul 1 (PHS/LAB)

Având în vedere că *formatul PHS/LAB* nu conține niciun fel de metadata dintre cele cerute de header-ul, inspirat de standardul TEI, al formatului CoBiLiRo, partenerul care va încărca resursa pe platformă va trebui să completeze în interfața grafică câmpurile aferente acestor metadata. Valorile lor vor fi preluate și stocate în header.

Textul aferent fiecărei înregistrări audio va fi extras din fișierul *.txt* și va fi convertit pentru formatul *xml* CoBiLiRo într-un tag de tip *unit*. Acest tag va conține și numele fișierului audio

aferent textului. De asemenea alinierea se poate face și la nivel de fonem, folosind informația din fișierul *.phs*, unde există două valori, ce indică începutul și sfârșitul unui fonem în milisecunde.

Pe lângă fișierul *xml* ce conține metadatele și informația textuală, platforma va stoca separat și fișierele audio. În tabelul de mai jos prezentăm o posibilă mapare a elementelor formatului PHS/LAB la elementele formatului CoBiLiRo.

Format PHS/LAB	Format CoLiBiRo	
<i>Valoare</i>	<i>tag</i>	<i>atribut</i>
Numele fișierului audio	Unit>speech	speechFile
Conținutul fișierului <i>.txt</i>	Unit>text	
Prima coloana a fișierului <i>.phs</i>	Unit>speech	start
A doua coloana a fișierului <i>.phs</i>	Unit>speech	stop
A treia coloana a fișierului <i>.phs</i>	Unit>text	

3.3.2 Formatul 2 (MULTEXT/TEI)

Spre deosebire de primul format, formatul MULTEXT prezintă o serie de metadate, descrise într-o manieră similară cu cele din formatul agreed în raportul 1.3 [3]. Astfel, pentru armonizarea resurselor disponibile în acest format, primul pas ar fi selectarea din xml-ul aferent formatului 2 a acelor câmpuri de care ne putem folosi pentru a completa header-ul formatului CoBiLiRo. Aceste date vor fi preluate automat, rămânând ca utilizatorul ce încarcă resursa să completeze în interfața platformei și celelalte câmpuri necesare, care nu sunt incluse în formatul MULTEXT.

Conținuturile fișierelor audio vor fi preluate din xml-ul formatului 2 (din tag-ul *text* ce prezintă o serie de tag-uri de tip *div* pentru textul fiecărui fișier audio) și introduse într-un tag de tip *unit* alături de denumirea fișierului audio aferent textului. Fișierele audio vor fi reținute și ele, separat de fișierul *xml* TEI.

Format 2	Format 1.3	
<i>Valoare</i>	<i>tag</i>	<i>atribut</i>

text>div>ab	Unit>speech	speechFile
text>div>p	Unit>text	
teiHeader>fileDesc>titleStmnt>title	teiHeader>title	

3.3.3 Formatul TEXTGRID

Acest format este folosit de aplicația de analiză și adnotare a semnalului audio Praat [4], una din cele mai populare în domeniu, cu o istorie de aproape 20 de ani. La fel ca Formatul 1 (*PHS/LAB*), formatul TextGrid nu conține niciun fel de metadata conform cu cele prezente în header-ul standard.

Aceste metadata vor fi completate de către utilizator, folosind interfața grafică. Datele din fișierele *txt* și *textgrid* vor fi salvate în tabele specifice.

3.4 Soluții tehnice pentru conversia automată

După identificarea formatelor inițiale și a celui final, precum și a soluțiilor de armonizarea avute în vedere, propunem câteva soluții tehnice pentru realizarea unei conversii în care intervenția utilizatorului să fie minimă.

Anterior etapei de conversie are loc o etapă de validare în care o serie de verificări sunt făcute automat peste toate câmpurile resursei, întâi pe *client* și apoi pe *server*.

Ulterior sunt validate și fișierelor asociate unei resurse. Altfel, platforma trebuie să se asigure în primă fază că fișierele urcate respectă măcar unul dintre formatele cunoscute. Dacă se constată prezența unui format cunoscut se trece mai departe la validarea corectitudinii formatului. Asta înseamnă parcurgerea fișierelor de intrare, verificarea conținutului lor și atenționarea contributorului asupra problemelor apărute. Validările se preocupă de aspecte precum: tipul fișierelor audio (acestea ar putea avea un format neagreat), numele și extensiile fișierelor, lipsa unor informații din fișierele urcate (lipsa unor tag-uri sau attribute XML etc.) etc. Aceste validări s-ar putea face atât la nivelul clientului (în browser), ceea ce ar însemna un consum în plus de memorie a mașinii ce găzduiește fișierele, sau la nivelul serverului, ceea ce implică trimiterea fișierelor către server, procesarea lor și trimiterea înapoi la client a unei eventuale liste de erori.

O altă validare foarte importantă ce ar trebui făcută la nivel de client, înainte de trimiterea fișierelor către server, este aceea a verificării extensiilor fișierelor urcate. Un utilizator ar putea încărca un fișier malițios, care odată ajuns pe server, să îi afecteze integritatea sau chiar să-l compromită. De aceea ar trebui să avem o listă predefinită de extensii acceptate pe server.

Un utilizator poate alege să încarce un director de pe disc cu fișierele sale sau o arhivă care să le conțină pe toate. Pentru acest ultim caz, platforma va fi capabilă să dezarhiveze fișierele și să le treacă prin procesul de validare și mai apoi upload. Această validare ar trebui făcută la nivel de

client deoarece urcarea lor pe server și dezarhivarea lor acolo ar însemna încărcarea unor fișiere malițioase ce s-ar putea găsi în arhivă.

În cazul încărcării unor resurse de mari dimensiuni, tot acest proces ar putea dura foarte mult. Astfel, un contributor nu trebuie neapărat să urmărească evoluția procesului din fața calculatorului. Poate să pornească procesul și să revină oricând pe pagina platformei pentru a verifica starea procesărilor. De asemenea, la sfârșit, va primi o notificare ce îl va informa dacă procesul s-a realizat cu succes sau dacă au apărut erori la validare.

Deoarece vorbim de resurse audio, care pot fi semnificative ca dimensiune, ne punem întrebarea ce probleme ar putea apărea dacă dimensiunea fișierelor depășește un 1 GB. Într-un asemenea caz majoritatea serverelor s-ar opri datorită unui time-out mult prea mare sau memoria serverului s-ar putea încărca peste o anumită limită admisă. Posibile soluții ar fi modificarea acestor limite sau spargerea input-ului în arhive mici, urcarea lor pe rând pe server și reasamblarea lor în backend.

Serverul pe care va fi instalată aplicația platformei este modelul HP ProLiant ML350 Gen9, cu un procesor E5-2650 v3 de 2.30GHz, care conține 40 de core-uri.

Având acest număr de core-uri, aplicațiile se pot împărți pe fire de execuție (thread-uri). O aplicație nu consumă mai mult de 5 fire de execuție la o rulare. Având la dispoziție 40 de core-uri, nu vor fi ocupate simultan toate, întrucât 15 core-uri vor fi necesare pentru rularea sistemului de operare și a altor programe necesare acestuia și se recomandă să existe un minim de 5 core-uri care să fie lăsate libere pentru orice altă procesare care poate fi realizată de către server - update, generare de log-uri etc. Astfel, avem la dispoziție un număr de 20 de core-uri pentru aplicație, adică 4 procese a câte 5 fire de execuție.

.NET Core, jQuery și MariaDB

Principalul motiv pentru care s-a ales framework-ul ASP.NET Core a fost faptul că tehnologia este *open-source*, *necesită resurse de calcul rezonabile*, *este rapidă și modulară*. Acest framework este în plină dezvoltare și ne pune la dispoziție o serie de librării și pachete *NuGet* ce ne ajută la dezvoltarea rapidă a platformei.

Unul din beneficiile majore aduse de .NET Core este portabilitatea. Acest lucru ne permite să hostăm aplicația noastră pe orice sistem de operare.

Un alt motiv pentru care am folosit această tehnologie a fost faptul că ne oferă o securitate a datelor (protecție pentru atacuri de tip *SQL Injection* și *Cross-site request forgery*) și un mod de a securiza API-urile REST folosind *JSON Web Token*.

Performanța ne este garantată de faptul că aplicația poate fi scalată pe anumite servicii (de exemplu, putem aloca un număr mai mare de instanțe pentru serviciile de conversie - care vor fi mai costisitoare din punct de vedere al procesării).

Pentru programarea la nivel de client - *client-side* s-a folosit framework-ul jQuery. Acesta ne permite manipularea DOM-ului și accesul la anumite animații și validări specifice.

3.5 Concluzii

În acest raport au fost descrise trei formate identificate ca fiind disponibile pentru a fi încărcate de parteneri pe platforma CoBiLiRo și au fost propuse soluții de armonizare în scopul oferirii de către platformă a unui format standardizat, ce facilitează valorificarea întregii colecții de resurse. Formatele originale nu au satisfăcut necesitățile platformei, așa cum au fost descrise în raportul corespunzător activității A1.3 [3]. Formatul propus în acel raport servește ca standard de adnotare la care vor fi convertite celelalte formate, începând cu cele trei descrise în acest raport. În cazul acestor conversii este necesară și completarea informațiilor disponibile în fișierele originale, în special în zona de header, unde informații precum autorul, sursa textului și a înregistrării audio sunt esențiale pentru o platformă de distribuire precum cea dezvoltată în cadrul proiectului. În cazul identificării ulterioare și a altor formate relevante, soluții similare de armonizare pot fi propuse. De asemenea, în cazul în care vor fi necesare ajustări ale standardului propus pentru platforma CoBiLiRo, soluțiile automate de conversie vor fi adaptate.

Toate obiectivele incluse în plan la această activitate au fost realizate.

Bibliografie

- [1] Dimitrova, L., Ide, N., Petkevic, V., Erjavec, T., Kaalep, H. J., & Tufis, D. (1998, August). Multext-east: Parallel and comparable corpora and lexicons for six central and eastern european languages. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1* (pp. 315-319). Association for Computational Linguistics.
- [2] Ide, N., & Véronis, J. (Eds.). (1995). *Text encoding initiative: Background and contexts* (Vol. 29). Springer Science & Business Media.
- [3] Cristea, D., Scutelnicu A. (2018, noiembrie), Raport “*Activitatea A1.3: Proiectarea funcțională și arhitecturală a infrastructurii care va găzdui resursele și instrumentele de prelucrare și acces ale consorțiului și realizarea unui prototip*”, proiect RETEROM
- [4] Boersma, P. (2002). Praat, a system for doing phonetics by computer. *Glott international*, 5.