**Projekat 4 – Automatsko kreiranje metapodataka o nastavnom materijalu u skladu sa IEEE LOM specifikacijom**

Studenti:

* Nebojša Vuga
* Bogdan Janošević

Tehnologije:

* Frontend – Angular
* Backend – ASP.NET Core
* Baza podataka – Sql Server
* Ekstrakcija metapodataka – Python3

Meta podaci:

* General
  1. Identifier
     + Catalog (Na osnovu ekstenzije dam Catalog)
     + Entry (GUID .NET)
  2. Title (Koriscenje naziva uploadovanog file-a, ako nije opcija onda PyPDF2, from mutagen.mp3 import MP3 from mutagen.id3 import ID3, TIT2 za mp3)
  3. Language ( Dobavim text iz pdf-a i onda i onda langdetect Python, za video ne znam kako bih, mozda ovo from moviepy.editor import VideoFileClip za ekstrakciju zvuka I onda from google.cloud import speech\_v1p1beta1 as speech za dobijanje jezika)
  4. Description (transformers ili sumy u pythonu)
  5. Keyword (Rake za prvih n dokumenata. Onda radim TF-IDF gde gledam koliko se puta pojavljuje rec u uploadovanom dokumentu u odnosu na dokumente iz baze.)
  6. Coverage (Spacy identifikuje geografske entitete u tekstu, ali ne znam da li je to dovoljno)
  7. Structure ( Da li se ovde misli na tip datoteke)
  8. Aggregation Level ( Mozda u zavisnosti ekstenzije da znam npr .zip je collection)
* Life cycle
  1. Version (Mozda da pruzim korisniku mogucnost da izmeni vec postojeci sadrzaj i svaki put kad ga izmeni povecam version za 1 ili tako nesto)
  2. Status ( Ne vidim poentu status polja u kontekstu zavrseno ili nije zavrseno)
  3. Contribute ( Mozda moze nekako da se ekstraktuje iz vec postojecih meta podataka uploadovanog fajla)
     + Role
     + Entity
     + Date
* Metadata-Metadata (Nema mi smisla da se izvlace podaci o podacima, svuda ce biti isti kreator, jezik...)
  1. Identifier (Ovo ce imati sigurno posto cuvamo u bazi sve metapodatke)
     + Catalog
     + Entry
  2. Contribute
     + Role
     + Entity
     + Date
  3. Metadata schema
  4. Language
* Technical
  1. Format (Ekstenzija datoteke)
  2. Size (Size datoteke u biteovima iz uploada Za audio i video datoteke možeš dobiti trajanje koristeći biblioteke poput moviepy ili mutagen.)
  3. Location (Picture, Video mozda neke geografske koordinate mogu da se izvuku)
  4. Requirement ( Ne znam kako bih automatski dobio zahteve za softver i hardver potreban za materijale. Ako su obicni nastavni materijali verovatno ne zahtevaju nista specijalno)
     + OrComposite
       - Type
       - Name
       - MinimumVersion
       - MaximumVersion
  5. Installation remarks(Mozda da se pretrazuju reci kao sto su „Installation“ i da se ta recenica izvadi iz teksta i sacuva ali to mi nema previse smisla)
  6. Other platform requirements(Isto kao proslo)
  7. Duration (Video moviepy)
* Educational

1. Interactivity type (Mozda ako ima „Quiz“ u sebi da bude active, nisam siguran)
2. Learning resource type (Mozda na osnovu extenzije da se zna koji je tip)
3. Interactivity level (Mozda po recima koje se nalaze u tekstu „quiz“, „game“)
4. Semantic density( Preko NLP da se odredi koliko cesto se pojavljuju kljucne reci u odnosu na ostale)
5. Intended end user role (Kao i do sad mozda po recima koje se najcesce pojavljuju „Teacher“ „Student“)
6. Context (Kao i do sad reci koje se nalaze u tekstu)
7. Typical Age Range (Preko reci „Kindergared“...)
8. Difficulty (Kljucne reci „Begginer“ „Advanced“)
9. Typical Learning Time
10. Description (Kao i ranije, ako se na to misli)
11. Language( Ne znam kako da znam jezik korisnika kojima je namenjeno)

* Rights
  1. Cost(Spacy – da prepozna da li ima u tekstu $ ili tako nesto)
  2. Copyright and other restrictions(Spacy – da prepozna ORG ili da se nadje rec CoypRight)
  3. Description(Ne znam kako bih ovo uradio)
* Relation
  1. Kind (Jedino da korisnik ima dodatni unos gde oznaci neki drugi od materijal od kojeg je ovaj materijal deo, ali to onda nije automatski)
  2. Resource(Ako ovo gornje dopustimo onda je zeleno)
     + Identifier
       - Catalog
       - Entry
     + Description
* Annotation( Ako dopustimo korisniku da daje dodatne komentare, ali to nije automatski onda)
  1. Entity
  2. Date
  3. Description
* Classification
  1. Purpose
  2. Taxon path (Mozda da imam neki predefinisan skup klasa i onda na osnovu reci da za svaku klasu gledam koje se reci pojavljuju)
     + Source ( Ovo gore)
     + Taxon (Mozda neke kljucne reci da nadjem)
       - Id
       - Entry
  3. Description (Kao i do sad)
  4. KeyWord (Kao i do sad)

Da vidimo sa kojim llm mozemo da se povezemo i da onda polako krenemo jedan po jedan da vidimo da li llm moze da ih resi.