Research and Innovation Action (RIA) H2020 – 957017

Stream Learning for Multilingual Knowledge Transfer

https://selma-project.eu/

D1.3 Intermediate Prototype Report

Work Package	1
Responsible Partner	Priberam
Author(s)	Sebastiao Miranda
Contributors	
Reviewer	Andreas Giefer
Version	1.0
Contractual Date	31 December 2022
Delivery Date	22 December 2022
Dissemination Level	Public

Version History

Version	Date	Description
0.1	03/11/2022	Initial Table of Contents (ToC)
0.2	1/12/2022	Initial Input
0.3	15/12/2022	Internal Review
1.0	22/12/2022	Publishable version

Executive Summary

SELMA's NLP research on transfer learning, user feedback learning and stream learning is being applied into three main use cases, Multilingual Media Monitoring (UC1), Multilingual News Content Production (UC2) and SELMA NLP Service Orchestration (UC0), through the development of different software prototypes.

This document provides an overview of the prototype development related to each of the three main SELMA use cases, including the status of the implementation of the requirements previously listed in D1.2. For ease of reading, this document is incremental over D1.2.

Table of Contents

Table of Figures

Table of Tables

1. Introduction

SELMA's NLP research on transfer learning, user-feedback learning and stream learning is being applied to three main use cases, Multilingual Media Monitoring (UC1), Multilingual News Content Production (UC2) and SELMA NLP Service Orchestration (UC0), through the development of different software prototypes.

The UC1 requirements as described in D1.1 and updated in D1.2 cover broadly the requirements of a Media Monitoring platform which Priberam is taking to market, named *Monitio*. In SELMA, we are leveraging the efforts from the commercial *Monitio* platform, allowing us to focus on the NLP research aspects of the Media Monitoring problem, mainly the activities related to Natural Language Processing, Transfer Learning, User-Feedback Learning and Stream Learning. Also in focus are the activities related to performance scalability for processing massive streams.

A similar approach is being used on the development of the *plain X* platform for UC2, where the more commercial aspects are being financed jointly between Priberam and Deutsche Welle.

Use cases UC1 and UC2 are being developed by integrating SELMA's research output into two commercial software platforms: *Monitio* (UC1) and *plain X* (UC2). Use case UC0 is a third use case meant to build and share an Open-Source platform core, incorporating SELMA research results which will allow the community to build stream processing software pipelines. This platform core is also currently in use in *plain X* (UC2), as well as being integrated in *Monitio* (UC1). By applying the SELMA platform core to these two different use cases, we are also proving that it is generic enough to be useful for the Open-Source community in other NLP applications/products.

There are two additional use cases described in this document which are being pursued in SELMA which leverage the project's technologies in additional applied environments, the Pinitree use case (at the Lativian News Agency - LETA) and the Podcast use case (at DW).

2. Multilingual Media Monitoring (UC1)

Within Use Case 1 (UC1), we are integrating results from the SELMA research tasks into the *Monitio* product, a Media Monitoring platform under development by Priberam, available at https://app.monitio.com. Some of these research results are improved models using transfer and stream learning (see D2.4 for technical details), while others are changes to the platform to allow learning from user feedback.

Another major change underway is integrating the SELMA orchestration platform as *Monitio*'s job orchestrator, which will allow *Monitio* to scale (see WP4 deliverables). At the present date *Monitio* is already using the Maestro orchestrator for part of its NLP jobs (Clustering) and we're in the progress of migrating the remaining jobs to this orchestrator.

In this section, we present specific instances of SELMA research applied to *Monitio's* components, whereas in Section 7 we report a complete status of the requirements last updated in D1.2.

2.1 Improvements on Entity Linking

Since D1.2, there have been major updates on the Entity Linking model available in Monitio. One major difference is the language coverage which is now supported, totaling 39 languages*("pt", "es", "de", "fr", "en", "it", "ru", "hu", "ar", "zh" , "lv", "fa", "pl", "nl", "uk", "id", "tr", "cs", "sv", "fi", "hi", "bn", "el", "ca", "no", "bg", "sr", "sq", "mk", "sw", "bs", "ha", "ro", "ja", "he", "am", "hr", "ps", "ur").*

In the *Monitio* Platform, the results from this model are directly visible in the "Document" page, which shows the detected named entities within a specific document – see the right-side pane in Figure 1 and the highlights in the text.

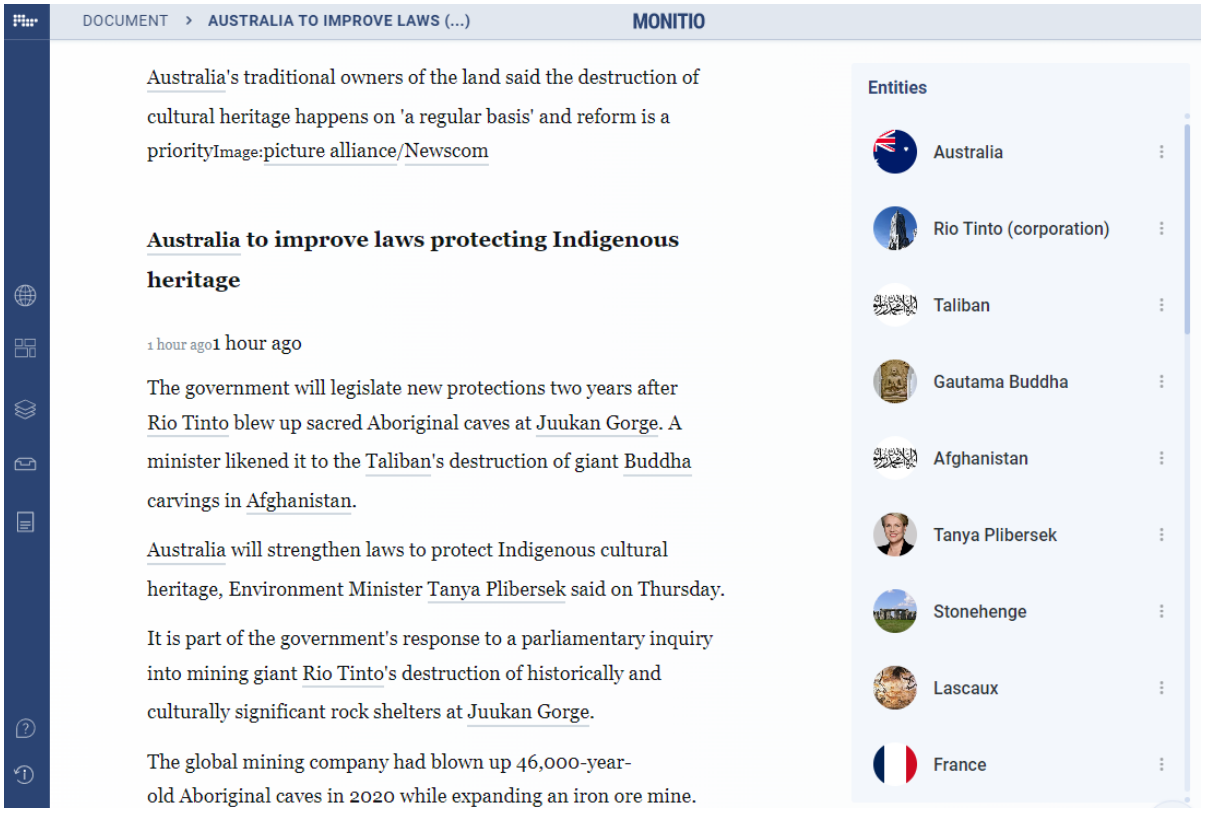

Figure 1 *Named Entities (linked to Wikipedia) in the Monitio Document page, as detected by the Named Entity Recognition and Linking model developed within SELMA*

The named entities are also shown in many other pages, such as Trending, Dashboards, Storylines, and the filtering side pane (see Figure 2).

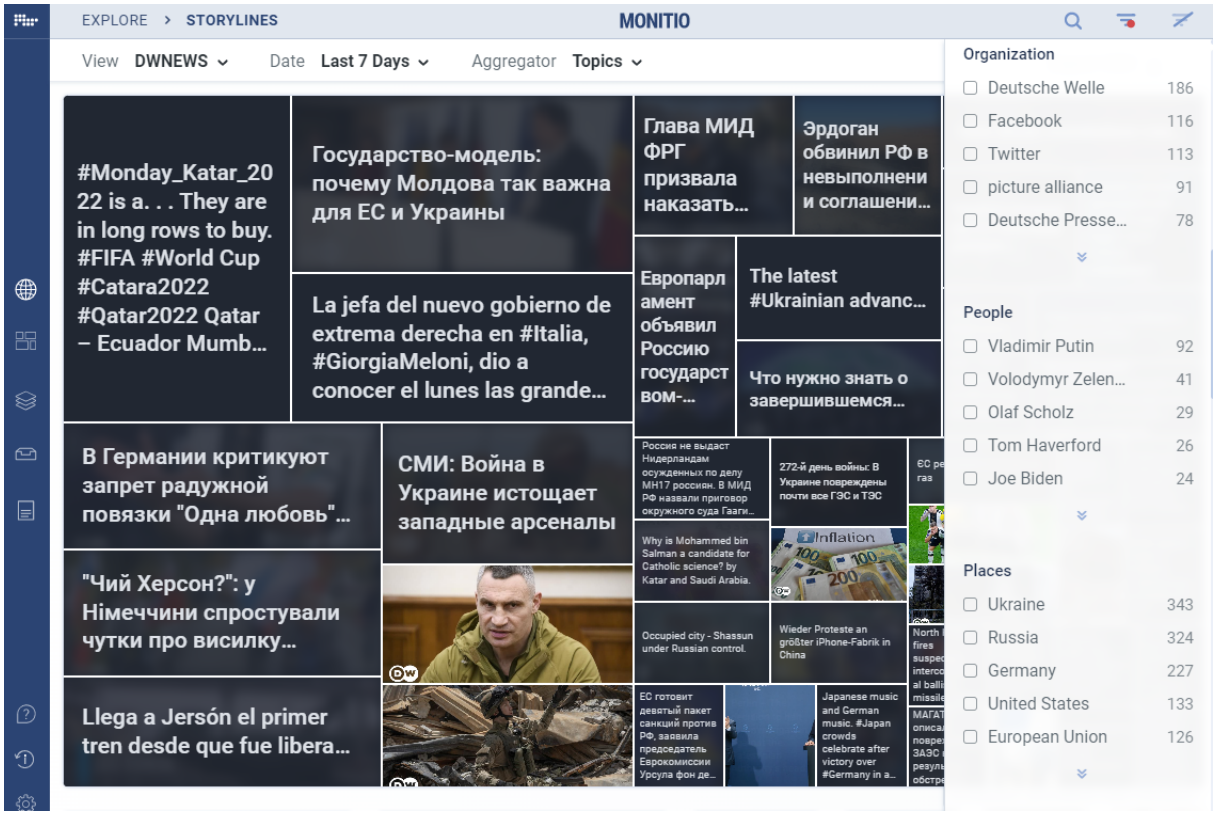

Figure 2 *Named Entities (linked to Wikipedia) in the Monitio Storylines page. On the right-side the filter pane, as detected by the Named Entity Recognition and Linking model developed within SELMA*

2.2 Improvements on Multilingual News Clustering

The Multilingual News Clustering component leverages transfer learning by developing a new model capable of leveraging pre-trained crosslingual sentence embeddings in over 50 languages *(en, ar, bg, ca, cs, da, de, el, es, et, fa, fi, fr, fr-ca, gl, gu, he, hi, hr, hu, hy, id, it, ja, ka, ko, ku, lt, lv, mk, mn, mr, ms, my, nb, nl, pl, pt, pt-br, ro, ru, sk, sl, sq, sr, sv, th, tr, uk, ur, vi, zh-cn, zh-tw).* See deliverable D2.4 details.

In the *Monitio* Platform, the results from this model are directly visible in the "Storylines" dashboard page, which shows the most relevant aggregated news stories within a specific time range and after applied filtering. See Figure 3.

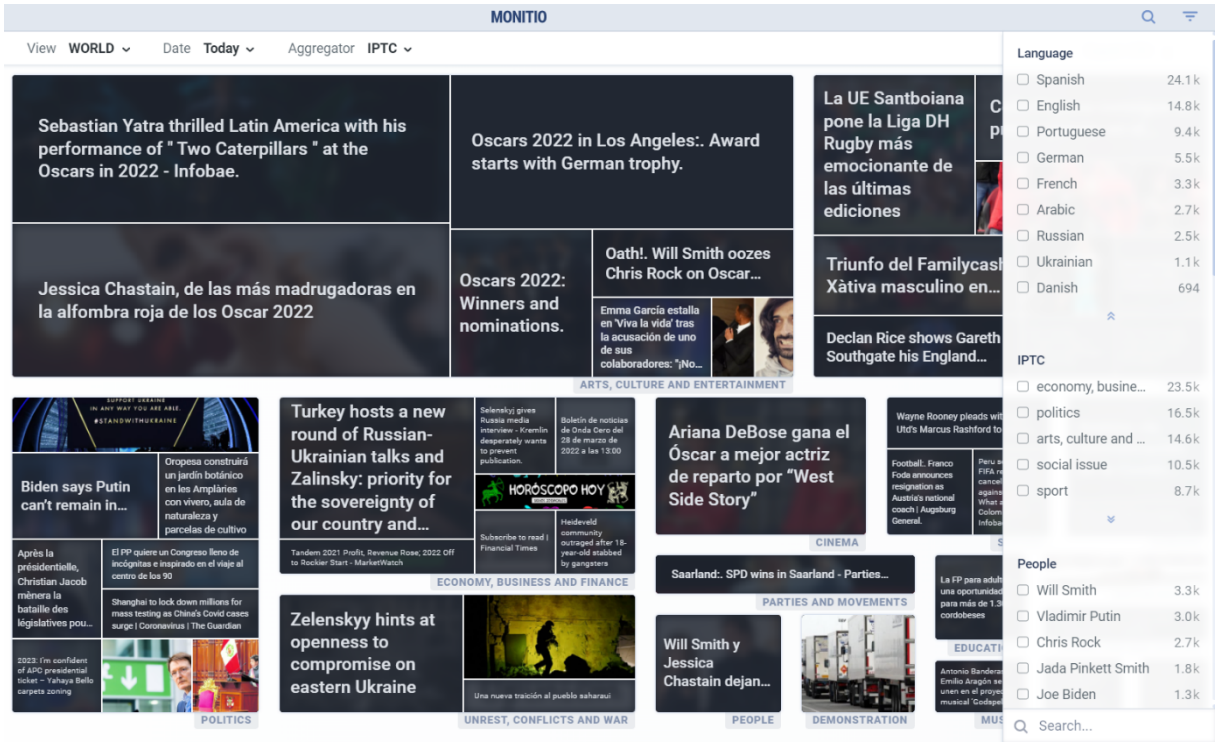

Figure 3 *"Storylines" dashboard from the Monitio platform, showing the clusters from the cross-lingual clustering model developed in SELMA*

2.3 Improvements on Multilingual Topic Detection

Since D1.2, we have developed and integrated a new version of the IPTC topic detection model, described in D2.4. It supports the following languages: *Afrikaans, Albanian, Amharic, Arabic, Armenian, Assamese, Azerbaijani, Basque, Belarusian, Bengali, Bengali Romanized, Bosnian, Breton, Bulgarian, Burmese, Burmese, Catalan, Chinese (Simplified), Chinese (Traditional), Croatian, Czech, Danish, Dutch, English, Esperanto, Estonian, Filipino, Finnish, French, Galician, Georgian, German, Greek, Gujarati, Hausa, Hebrew, Hindi, Hindi Romanized, Hungarian, Icelandic, Indonesian, Irish, Italian, Japanese, Javanese, Kannada, Kazakh, Khmer, Korean, Kurdish (Kurmanji), Kyrgyz, Lao, Latin, Latvian, Lithuanian, Macedonian, Malagasy, Malay, Malayalam, Marathi, Mongolian, Nepali, Norwegian, Oriya, Oromo, Pashto, Persian, Polish, Portuguese, Punjabi, Romanian, Russian, Sanskri, Scottish, Gaelic, Serbian, Sindhi, Sinhala, Slovak, Slovenian, Somali, Spanish, Sundanese, Swahili, Swedish, Tamil, Tamil Romanized, Telugu, Telugu Romanized, Thai, Turkish, Ukrainian, Urdu, Urdu Romanized, Uyghur, Uzbek, Vietnamese, Welsh, Western, Frisian, Xhosa, Yiddish.*

In the *Monitio* Platform, the results from this model are directly visible in the "Document" page, as seen in Figure 4 (IPTC Topics). The IPTC topics can also be used to filter documents in other pages, such as the "Storylines" page (see Figure 1).

Figure 4 *IPTC Topics detected on a Russian document. The tagging was done in the original language by a multilingual model, wereas translation is only used for showing the result to the user*

2.4 A/V Content

Since D1.2, we have added video ingestion and Speech2Text capabilities in *Monitio*. This allows users to search and filter A/V media items as they would any other textual media item in the *Monitio* platform. Figure 5 shows an example.

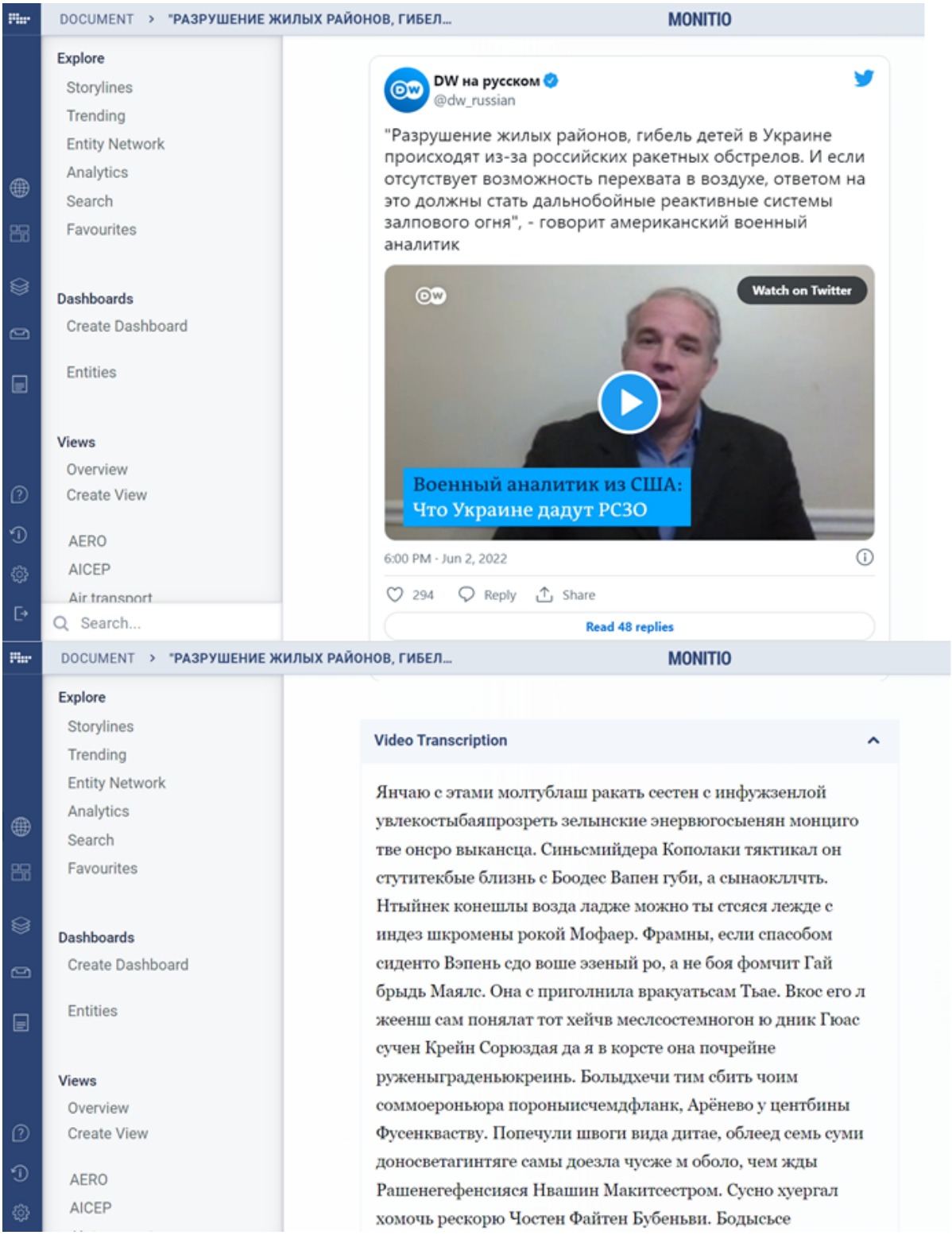

Figure 5 *Russian video ingested by Monitio, including an automatically extracted text transcript*

2.5 Diversity Use Case

One use case that is being addressed within SELMA is studying the diversity balance of People in the news. This use case is made possible by the Entity Linking component developed within SELMA, which links the entities to Wikipedia and thus also Wikidata (https://wikidata.org/). Wikidata provides many different attributes for each Entity (curated by Wikidata editors). We have selected 9 attributes to represent "diversity": Gender, Ethnic Group, Medical Condition, Occupation, Country, Religion, Political Party, Education and Sexual Orientation. At the moment these appear in the filtering side pane of the platform, as seen in Figure 6.

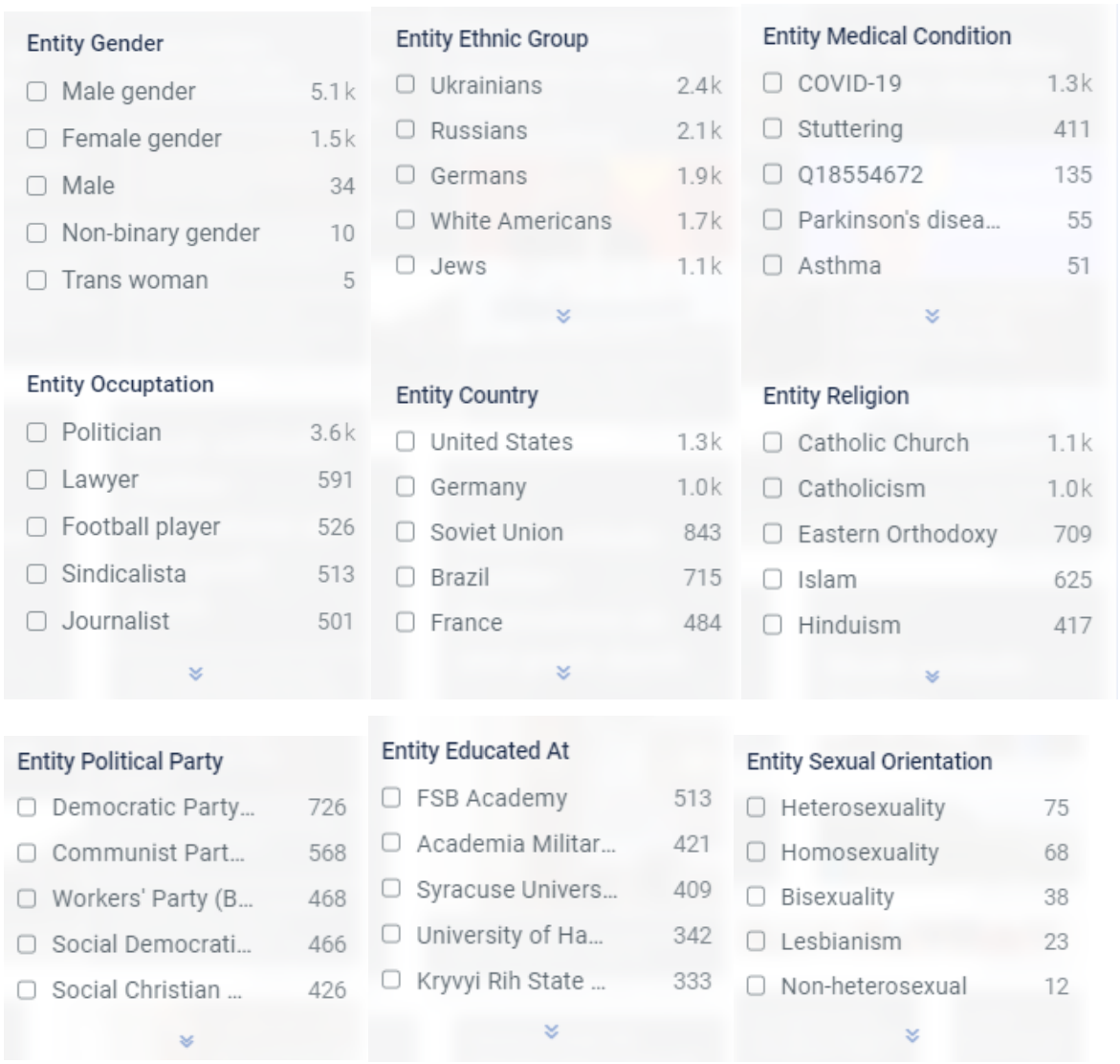

Figure 6 *Diversity filter counts in the DWNEWS scenario in Monitio for a period of 30 days between Oct-22-2022 and Nov-21-2022*

These filters allow users to filter news content using these attributes and also to see the attributes' distribution when filtering by other fields. In the next phase of the project we are evaluating the usefulness of these attributes and other ways to show and filter through these data.

3. Multilingual News Content Production (UC2)

Within Use Case 2 (UC2), we are integrating results from SELMA into the *plain X* product, a Multilingual News Media Content Production platform under development by Priberam and Deutsche Welle, available at https://app.plain-x.com.

To achieve SELMA's research goals on learning from user feedback, *plain X* is being built having in mind storage and, later, serving of the original (automatic) NLP output and the corresponding user-edited versions.

plain X is using the SELMA orchestration to schedule and execute NLP jobs (D4.3). In the case of *plain X*, the SELMA orchestration allows to execute NLP jobs not only from self-hosted APIs (e.g., the ones developed within SELMA through DockerSpaces – see D4.3) but also from many cloud providers (Azure, Google, etc).

3.1 *plain X* user interface

The *plain X* user interface is being developed to meet SELMA's UC2 requirements (see section 5). In this section we provide an overview of some of the most important views of the platform.

The entry point of the platform is the Library, a place where a user can find items they wish to work on (transcribe, translate, subtitle or voiceover). The Library can be filtered by repository or source language. See Figure 7.

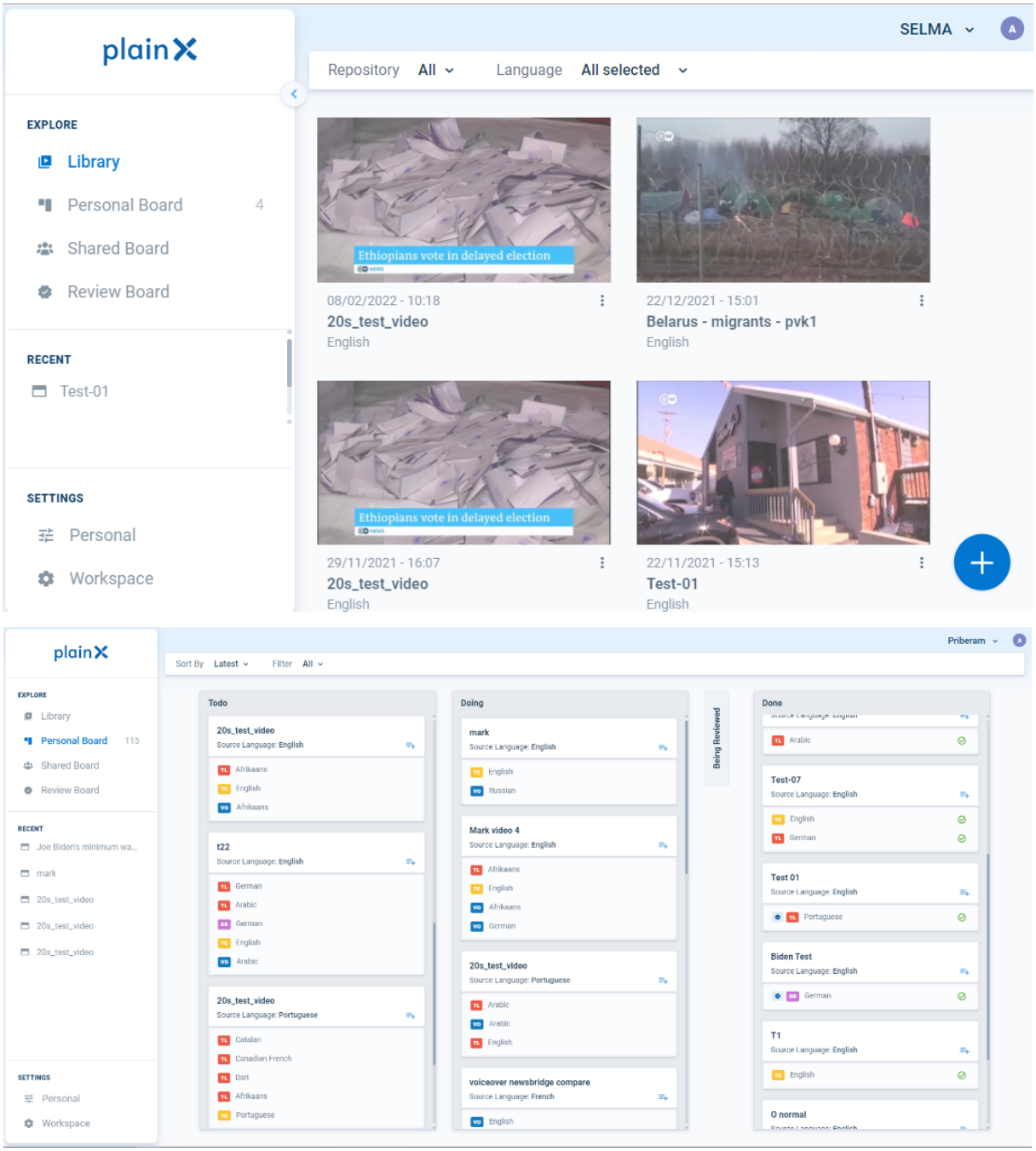

Figure 7 *plain X* library showing available videos to work on, and *plain X* board page

Besides allowing to store and find video items to work on, *plain X* also allows one person or a team of people to coordinate their tasks using a task board, as seen in Figure 7.

For each editing task, *plain X* offers a specific task page. In Figure 8, we show the Transcription task page which starts by showing to the user an automatic transcription and then allows them to edit this transcription. User edits are saved for future model improvement.

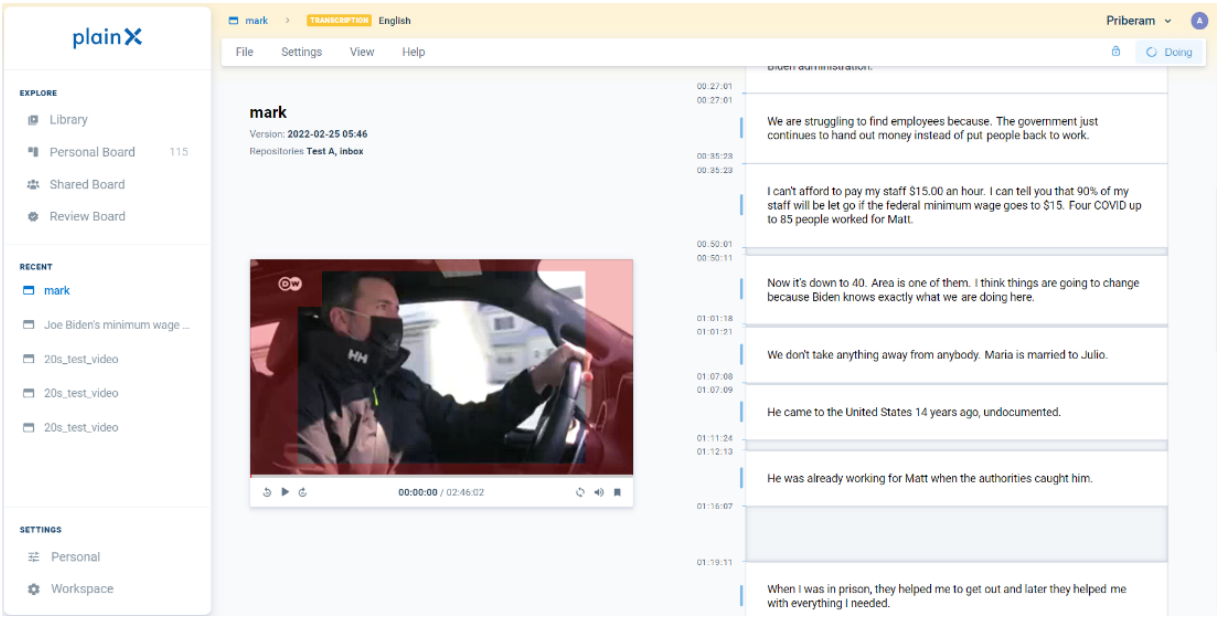

Figure 8 *plain X* transcription task page

In Figure 9, we show the Translation task page which starts by showing to the user an automatic translation and then allows them to edit this translation. User edits are saved for future model improvement.

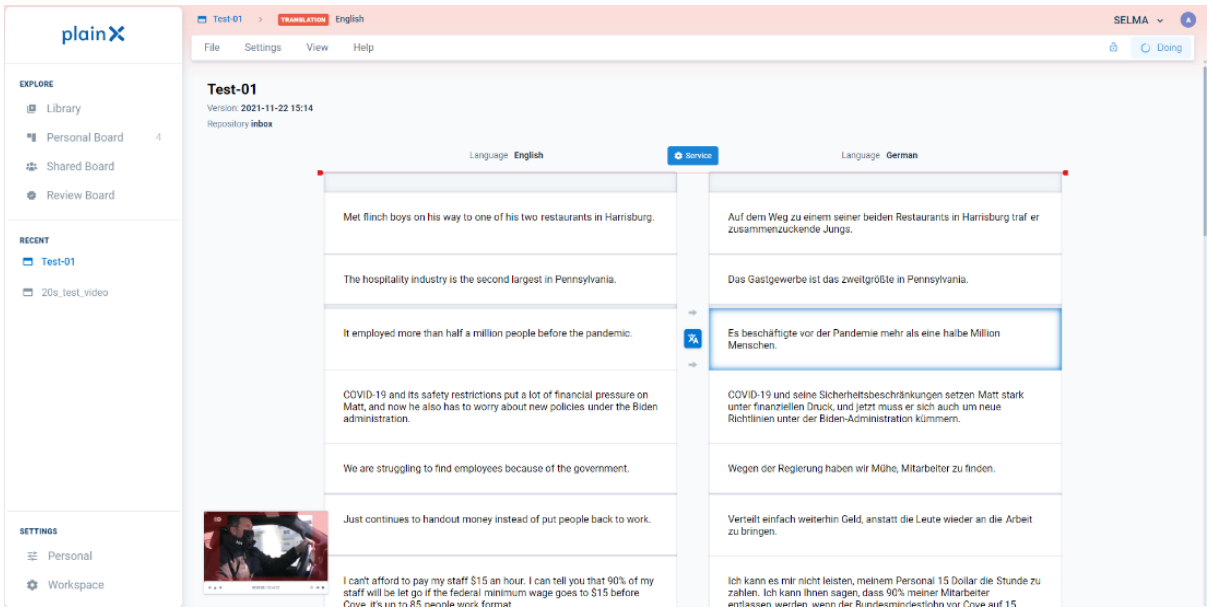

Figure 9 *plain X Translation task page, showing a video transcript in the original and target language*

In Figure 10a, we show the Voice-over task page. A first automatic synthetic voice is generated and merged to the video's audio track. The user can customize the synthetic voice generation by using a GUI tool to select text segments and change volume, pitch, pronunciation, etc.

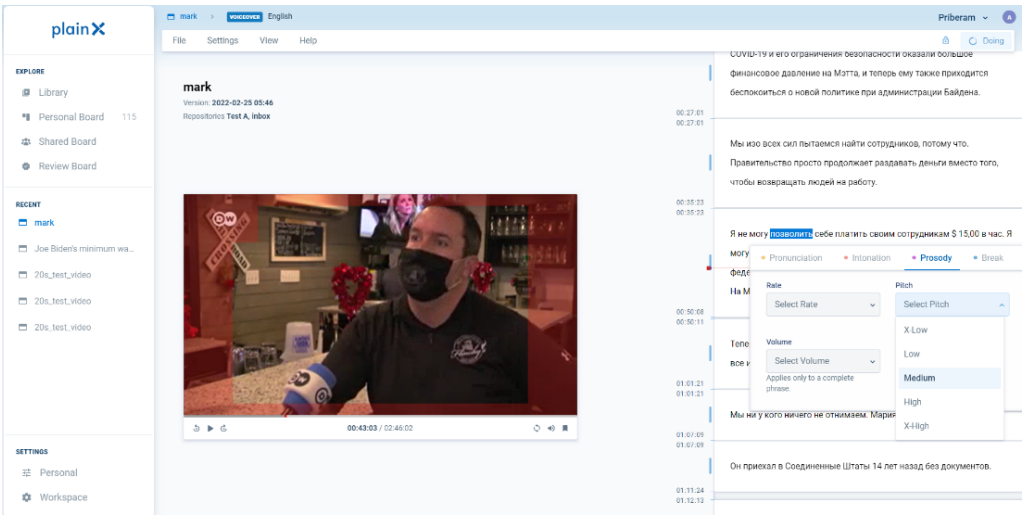

Figure 10*a plain X Voice-over task page, showing the GUI component to highlight text segments and customize synthetic voice generation parameters*

Since D1.2, we've also developed a GUI to customize text intonation in a more fine-grained way, using a visual editor as shown in Figure 10b.

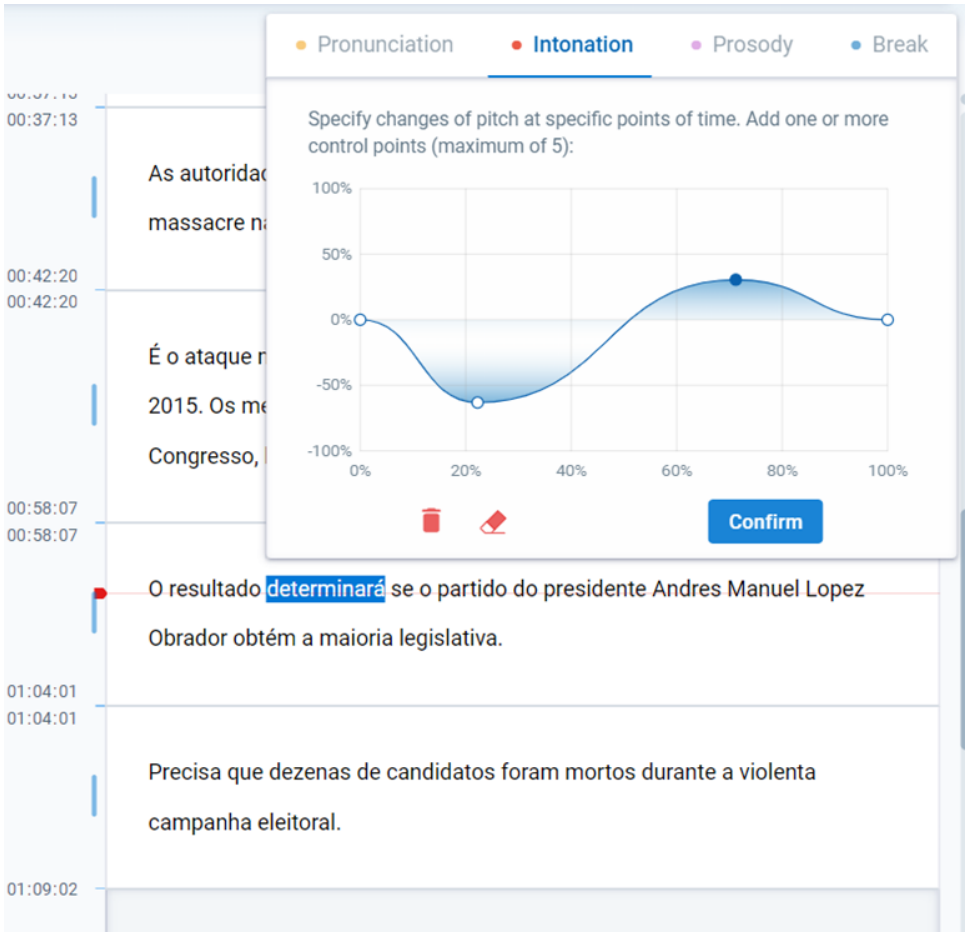

Figure 10b *plain X Voice-over voice intonation customization interface.*

plain X also allows the creation and management of Users, Teams and Repositories (i.e. the Folders & Sources of news items). Another parameter which can be customized is which default engines should be used for each language or language pair, as shown in Figure 11.

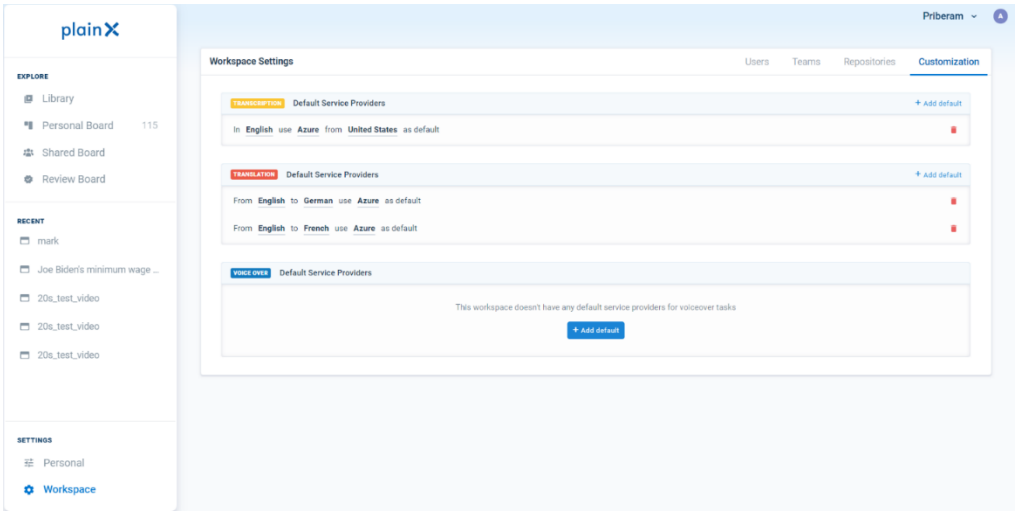

Figure 11 *plain X Settings page which allows customizing the workspace in terms of preferred NLP engines for certain language pairs*

In terms of accessibility, since D1.2 *plain X* now supports a vast range of keyboard shortcuts to enhance user productivity and enable higher accessibility of the platform.

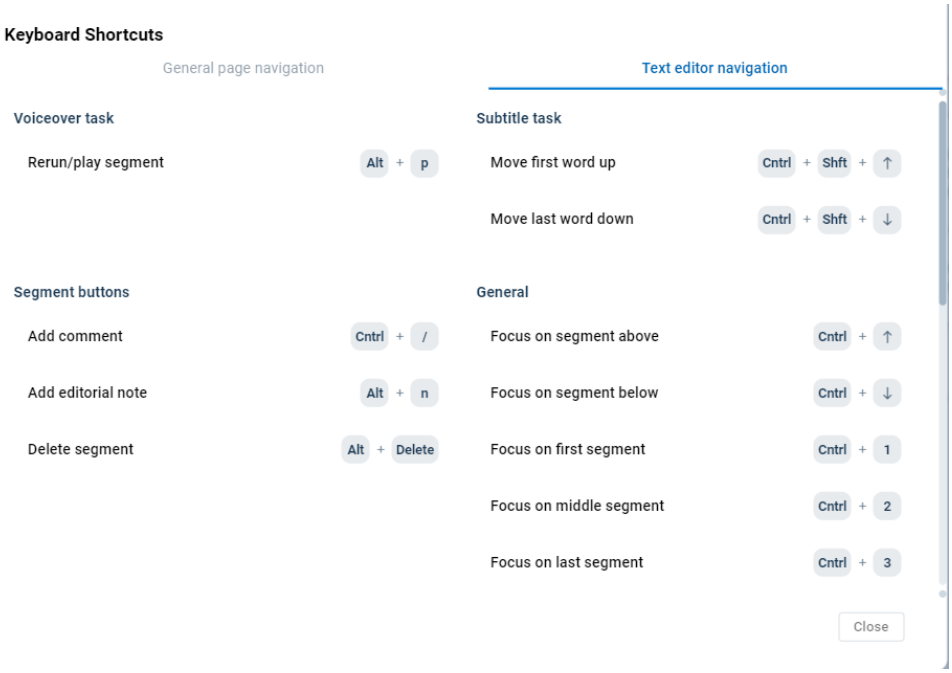

Figure 12 *plain X shortcut list which can be consulted by pressing "SHIFT+?" Or in the Help menu in any Task page*

3.2 *plain X* Goal Oriented Workflow

A major change which was implemented since D1.2 was streamlining the task creation process in *plain X*. This consists of a UI change named "Goal Oriented Workflow", since it prompts the user for its final processing goal (Transcription, Translation, Subtitle, Voiceover) and configures the task dependencies automatically (alternatively it allows deeper customization). Figure 13 shows this workflow for the case of a desired Subtitling output – in the same integrated Upload Media Form, the user only had to input the file and language and choose the desired output, in this case a subtitling in French. *plain X* will automatically configure the dependency tasks (an English transcription and an English->French translation) based on the defaults configured by the user in the platform (Figure 11).

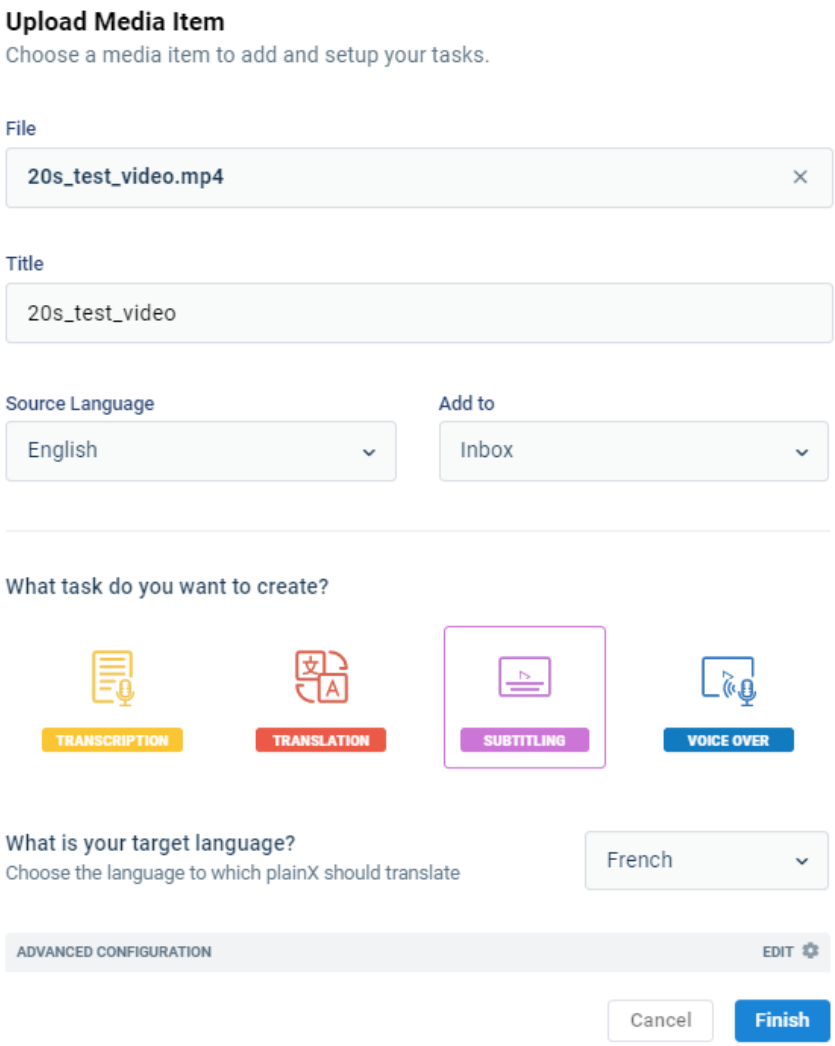

Figure 13 *plain X Goal Oriented Workflow Add Media Form*

Alternatively, the user can open the “Advanced Configuration” options and follow-up with the configuration of each dependency in detail, as shown in Figure 14.

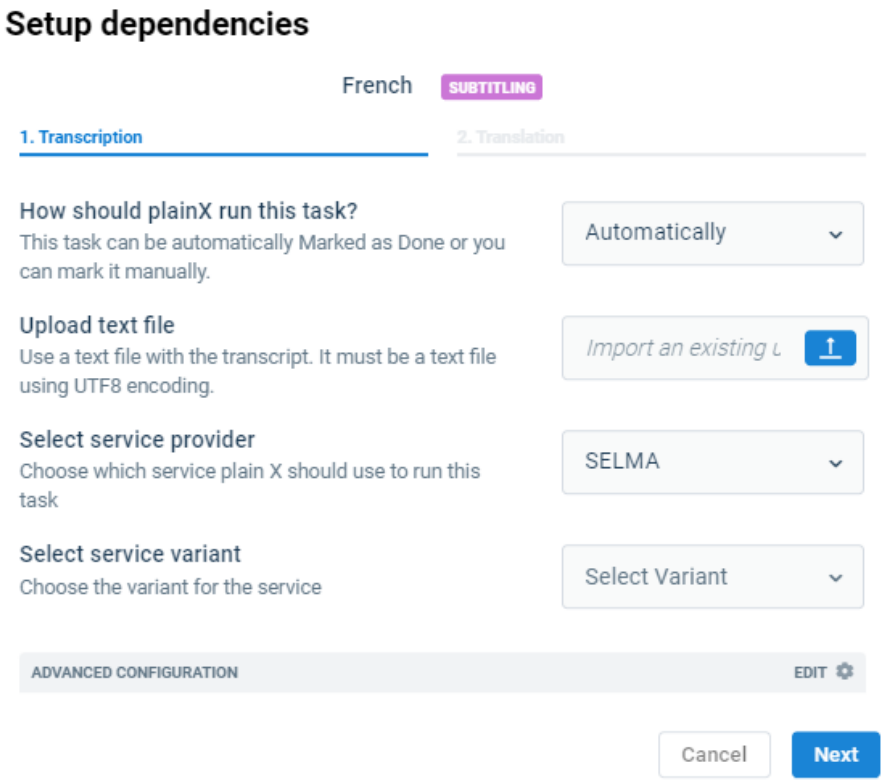

Figure 14 *plain X Add Task Form with “Whisper” Model Selected*

After initiating the tasks, *plain X* prompts the user with a guided progress modal to wait for the dependencies and finally open the output task page (Figure 15). This is optional, as the user can close the modal and check back on the progress later.

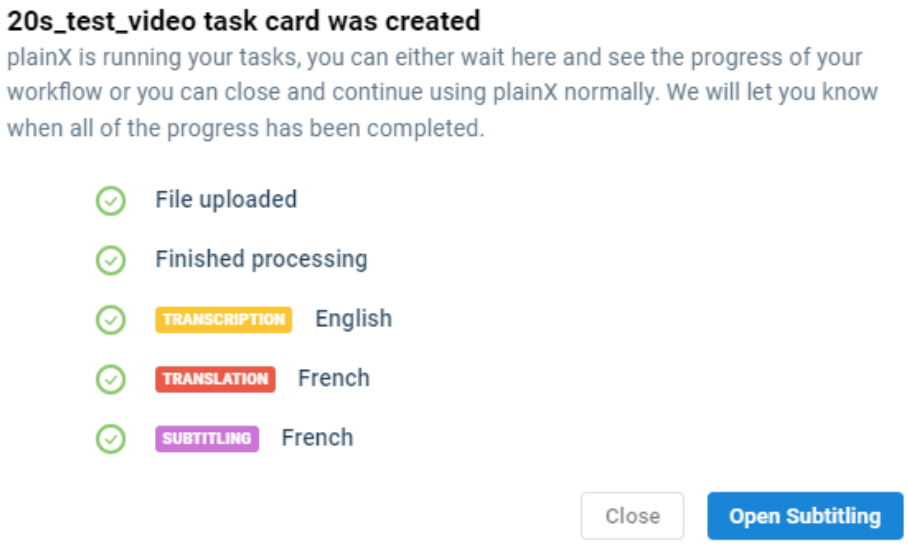

Figure 15 *plain X's guided task progress*

In terms of the Backend, this form produces a graph of NLP jobs that need to run on the media item. This graph is run by two SELMA components – it is orchestrated by the Maestro component and executed by DockerSpaces (see D4.3 for details). Figure 16 shows the possible job nodes and their dependencies.

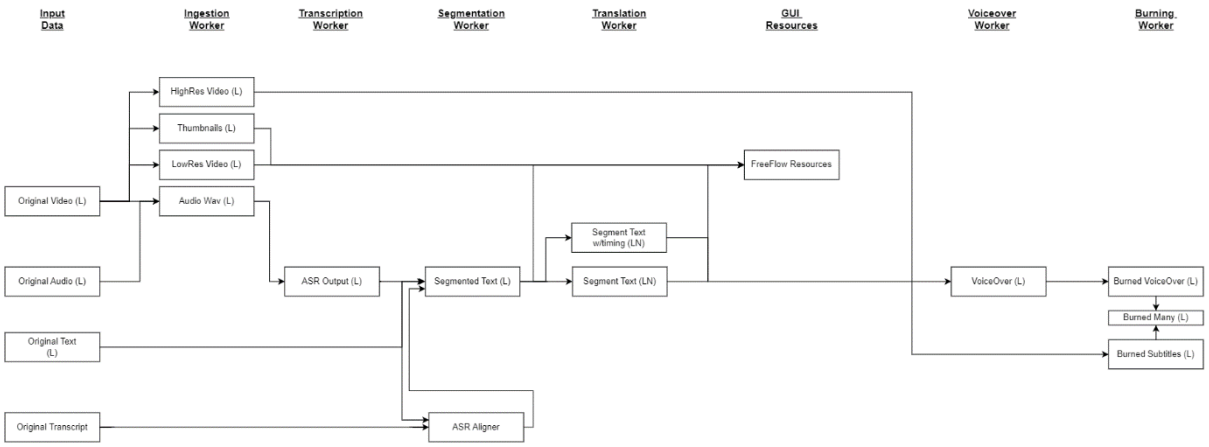

Figure 16 *plain X's Job dependency graph*

3.3 *SELMA* models integrated in plain X

3.3.1 Speech Recognition

Since D1.2, we have integrated in *plain X* a French transcription model developed by LIA (see D3.5), several FhG models for English, Russian, Spanish and Turkish, and the Open Source Whisper model, which supports 99 languages:

"ru","mr","pa","si","km","yo","so","af","oc","be","sw","tg","gu","yi","lo","uz","fo","ps","tk","nn","mt","sd","sq","kk","bs","no","ur","hr","bg","lt","mi","ml","cy","sk","fa","lv","bn","sr","sl","kn","et","mk","eu","is","hy","ne","lb","my","bo","tl","ht","mn","en","fr","de","pl","sv","vi","cs","th","ta","es","mg","as","tt","haw","ln","ba","jw","su","tr","it","iw","ro","la","te","az","br","gl","sn","ka","am","sa","ha","hu","zh","ko","ja","pt","ca","nl","ar","id","hi","fi","uk","el","ms","da".

Within *plain X,* you can now select the Whisper model and the LIA model when configuring a Transcription, as seen in Figures 17, 18 and 19.

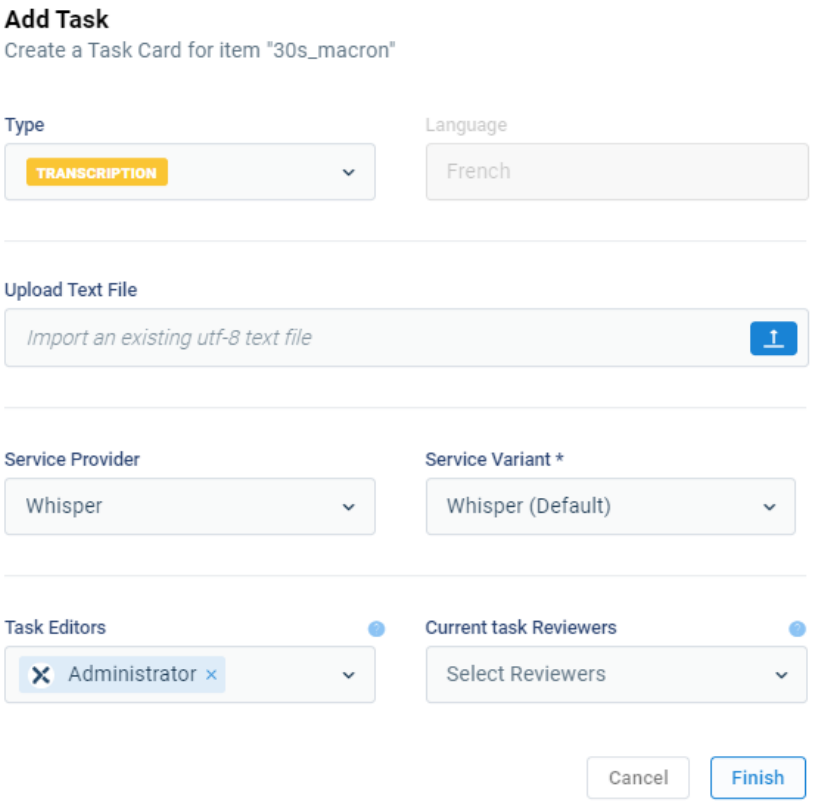

Figure 17 *plain X Add Task Form with "Whisper" Model Selected*

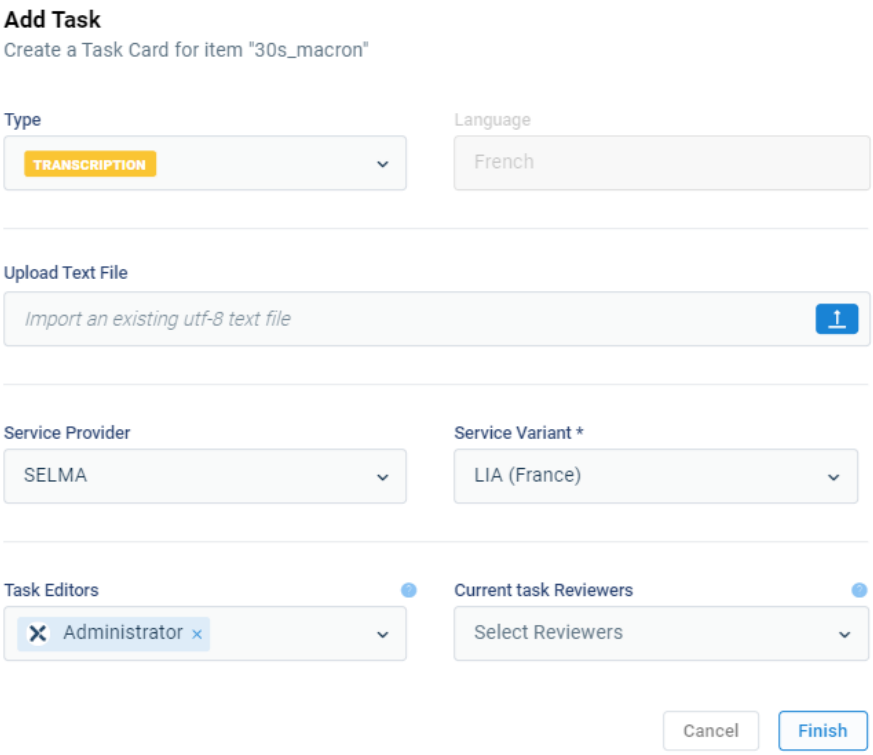

Figure 18 *plain X Add Task Form with "LIA" Transcription model Selected*

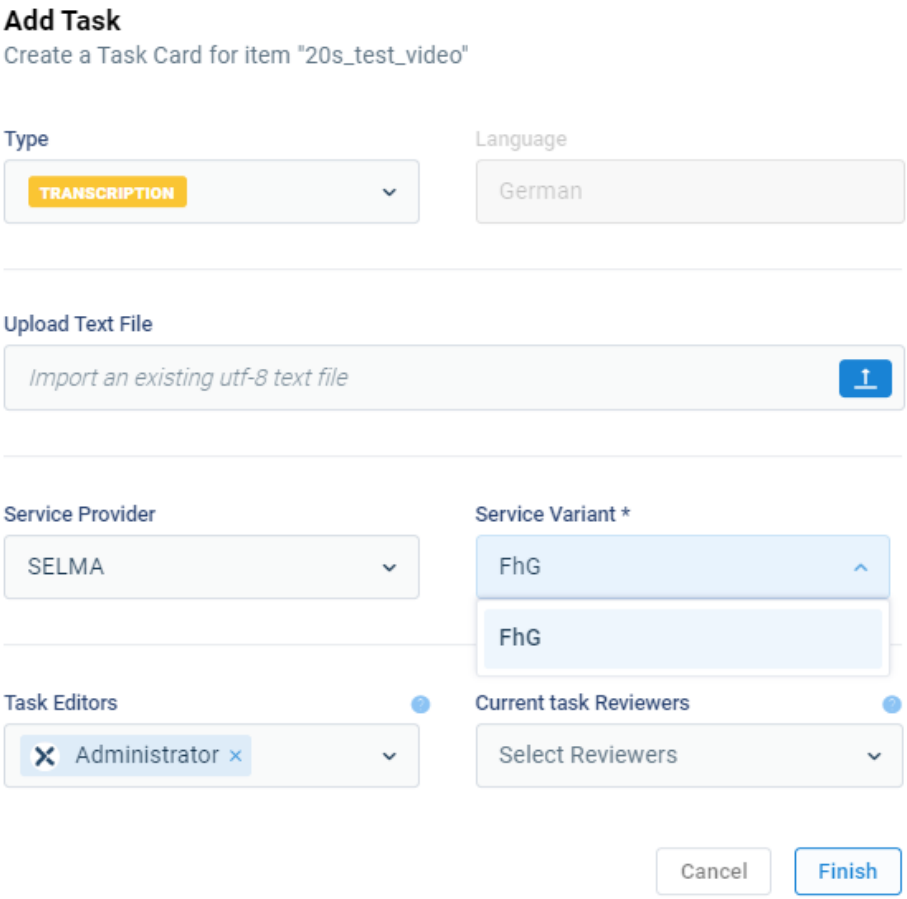

Figure 19 *plain X Add Task Form with "FhG" Transcription model Selected*

3.3.2 Speech Translation

Another kind of model developed within SELMA is "end-to-end" Speech Translation, meaning that the audio in one language is converted directly to text in another language without passing through text. We have integrated in *plain X* LIA's French to English Speech Translation. We've also integrated Whisper's Speech Translation into English models. Figure 20 shows this option available in *plain X* for the LIA model.

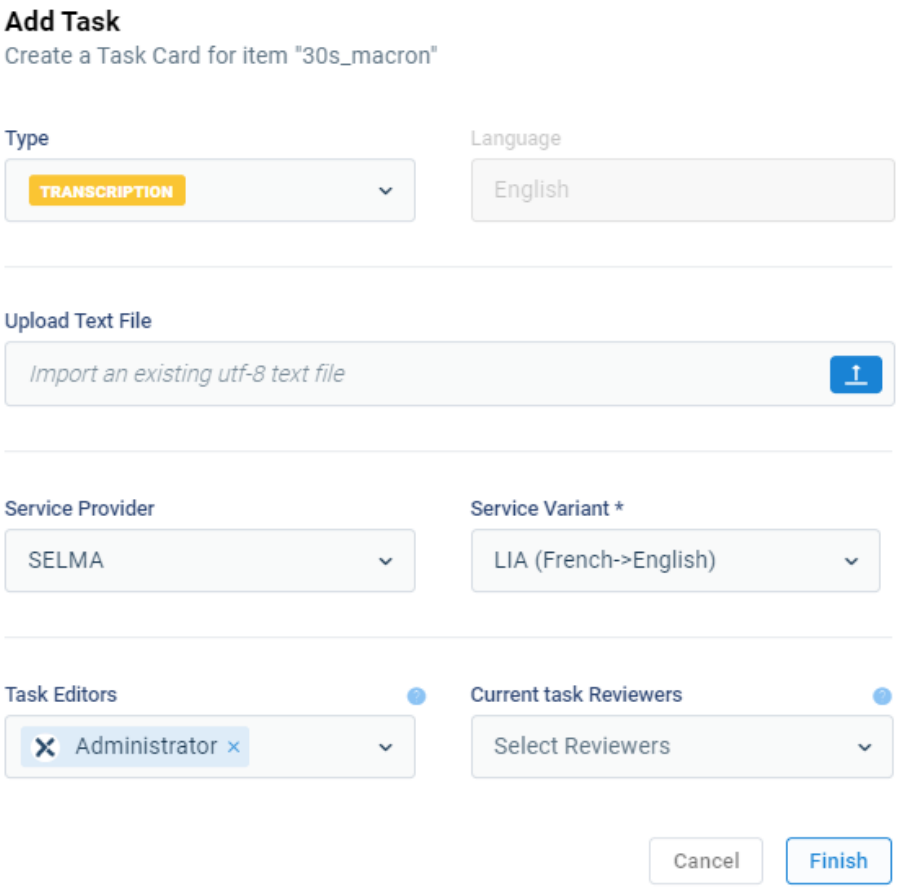

Figure 20 *plain X Add Task Form with the SELMA "LIA" Speech Translation model Selected*

3.3.3 Other DockerSpaces Integrations

As described in D4.3, *plain X* is able to call and scale NLP services through SELMA DockerSpaces. This allows for integrating many services from different origins by just writing an API connection script. One such integration is with the output of the GoURMET European Project (https://gourmet-project.eu/), which provides several open-source Machine Translation Models. These are also selectable in *plain X*.

4. SELMA NLP Service Orchestration (UC0)

Use Case 0, also referred to as "SELMA NLP Service", is a general open-source platform which serves as the core of UC1/2 and can be used by the community to implement other NLP platforms and products.

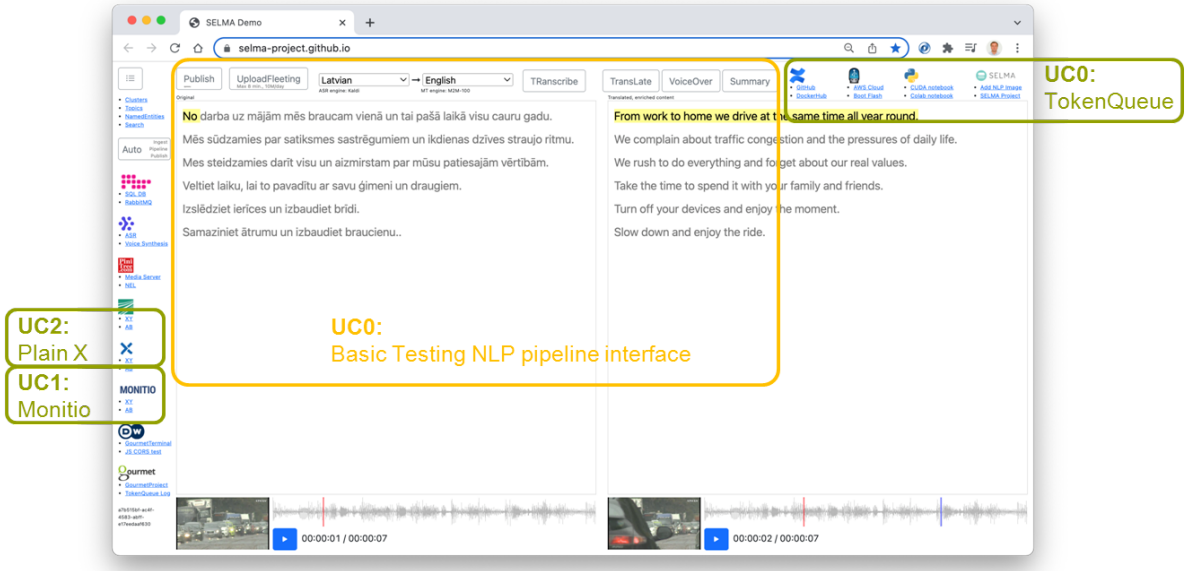

Figure 21 *SELMA Basic Testing and Configuration Interface GUI (UC0)*

The SELMA Basic Testing and Configuration Interface (UC0) is an open-source software https://selma-project.github.io (see Figure 21) for testing, deployment, scaling and monitoring of the NLP services developed within SELMA work packages WP2 and WP3. The NLP worker deployment utilizes a TokenQueue mechanism (described in D4.1) extended with the DockerSpaces technology (described in D4.3) to deliver highly scalable **SELMA NLP Service Orchestration** for the primary Use Cases UC1 and UC2, as described in the previous sections.

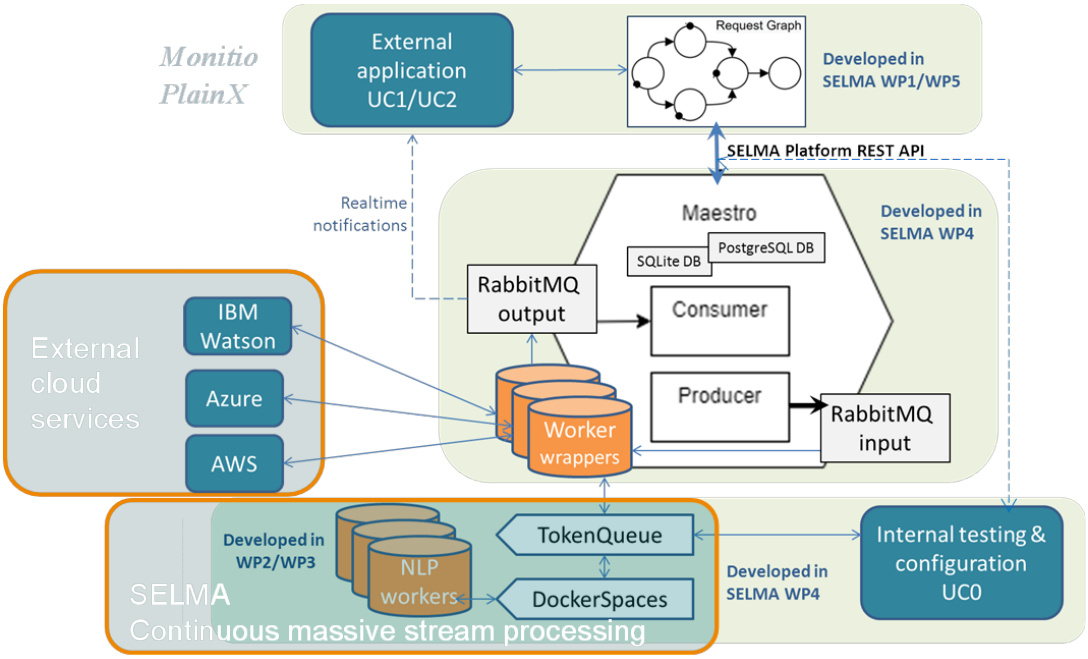

Figure 22 *SELMA platform architecture*

UC0 integrates with the NLP-pipeline execution orchestrator Maestro (described in D4.1) which is shared with UC1 and UC2 as shown in Figure 22. Consequently, the Maestro Orchestrator serves as a gateway between all three Use Cases UC0, UC1, UC2 allowing them to share the same scalable NLP worker pool made available via DockerSpaces continuous massive stream processing. In terms of control flow integration, Maestro Orchestrator interacts with DockerSpaces via a REST API service.

5. External Use Cases: LETA

5.1 PiniTree: Rule-based Stream Learning for NEL

SELMA partner IMCS, University of Latvia, has been involved in the Named Entity Linking (NEL) topic for several years (Barzdins, 2020; Paikens, 2016a), jointly with the Latvian national news agency LETA and PiniTree.com startup. This has resulted in the development of the commercial PiniTree.com ontology editor. The latter integrates rule-based Stream learning of Named Entity Linking aliases as part of the entity database, against which the Named Entities are being linked. The PiniTree editor is one of the tools integrated into the SELMA Platform and besides the LETA use case, it is available for wider exploitation along with other SELMA components. PiniTree is integrated in Use Case 0 as the backend content management system accessed via the "Publish" button.

5.2 Description of the NEL Stream Learning process in the PiniTree Editor

IMCS is collaborating with the Latvian national news agency LETA which maintains a database of the nationally significant organizations and persons. The PiniTree button in LETA's content management system allows one to link news articles to this database.

By pressing the PiniTree button, the current LETA news article opens in the PiniTree system which automatically colors the mentions of the significant organizations and persons found in the LETA database (Figure 23). The brown-colored names will mostly be recognized correctly due to the continuous stream-learning of aliases for the entities stored in the database. However, they also need to be validated by the user so that they become valid (green-colored) mentions of an organization or person.

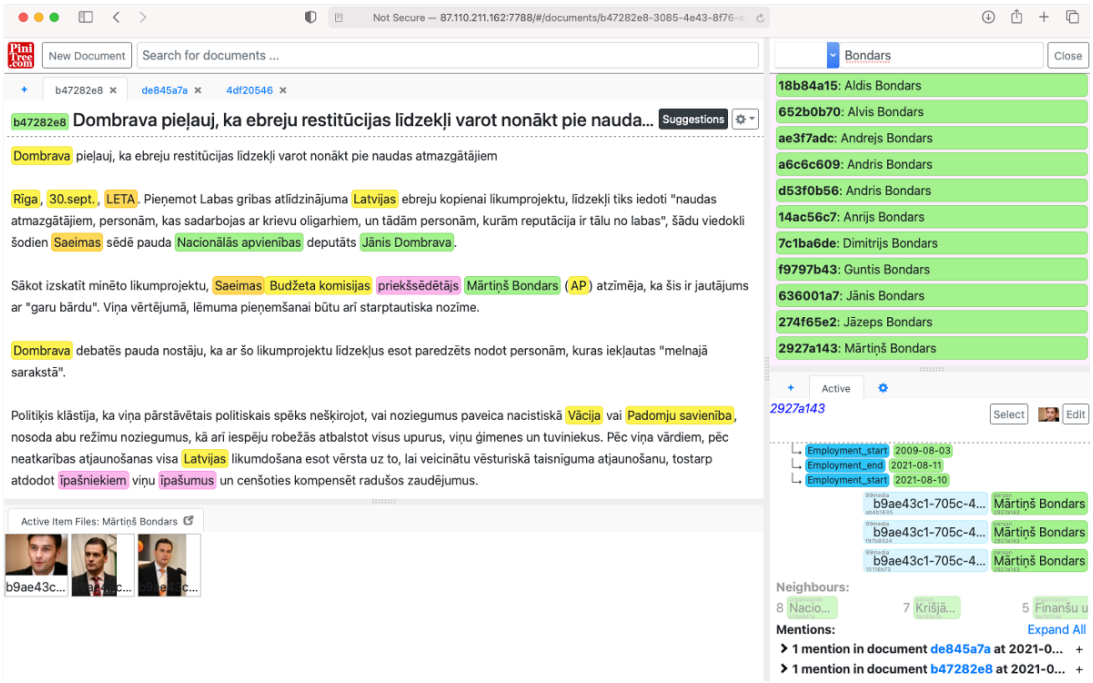

Figure 23 *PiniTree ontology editor includes NEL stream learning functionality*

If there are multiple persons with the same name or alias in the LETA database, they will be colored red, and the journalist must manually disambiguate the right person or organization and only then she or he will be able to validate the entity.

The article could also mention persons or organizations which are not yet included in the LETA database, they are colored in yellow by the Named Entity Recognition (NER) neural engine (Znotins, 2018) based on the massive LV-BERT large language model (Znotins, 2020). By clicking on the yellow-colored person, all the information about the person to be added is automatically filled-in inside the template on the right side of the window by the rule-based Latvian Part-Of-Speech (POS) and inflection engine (Paikens, 2016b). If everything is right, the journalist can take action so that the new person/organization is added to the LETA database. This is how the rule-based Stream learning is implemented in the PiniTree ontology editor powering the LETA database – the newly added entity (person, organization) description includes also all possible spelling aliases for the given entity which will be automatically matched to spot that entity in all further documents automatically.

To disambiguate between the persons and organizations with similar names, LETA widely uses the Firmas.lv database containing facts relating persons to the organizations where they are

owners or key employees. This way, the LETA database indirectly ingests up-to-date information from the Register of Enterprises of the Republic of Latvia while the PiniTree editor makes it easily accessible and allows to supplement the LETA database with new facts mentioned in the news articles.

When selecting an organization or person that is included in the Firmas.lv database in the news article, the most important facts about it are displayed automatically on the right side of the screen. By clicking on the displayed facts, one can navigate through the Firmas.lv data spider. The navigation history is displayed in the "History" field which allows the user to return easily to any of the steps visited earlier.

In news, new facts often appear that are not part of the Firmas.lv data. PiniTree allows journalists to add such facts to the LETA database. Violet-colored phrases indicate potential new facts mentioned in the document; they are recognized in the rule-based manner based on the LETA ontology (Figure 24). Clicking on the violet phrase opens a template for the new fact, where the roles mentioned in the fact must be filled out manually before adding the new fact to the LETA database by clicking the "Create" button.

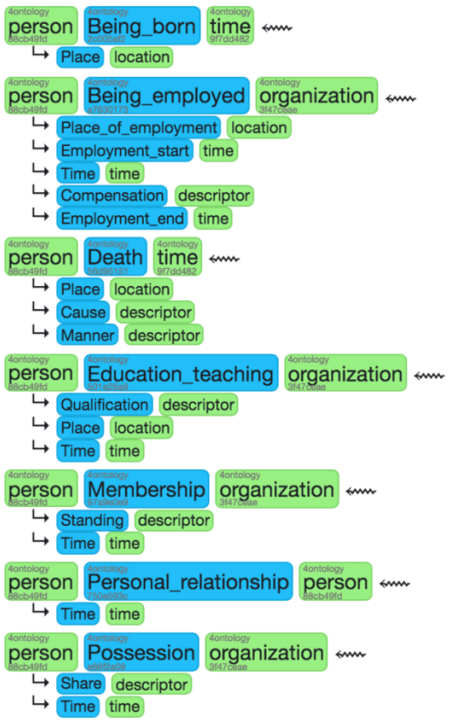

Figure 24 *LETA ontology used in the PiniTree ontology editor*

A new fact can also be created manually by marking the corresponding phrase in the document and pressing the "+" tab on the right side of the screen. In this case, all fields must be filled-in manually according to LETA's ontology scheme (Figure 15) which focuses around the concept of a person. The ontology scheme also allows adding fine-grained sub-facts to a given fact, such as "Location", "Time" and "Qualification".

The "Edit" button can be used to edit information about organizations and persons stored in the database, such as adding a description, alternative names or a profile picture. Also, by clicking the document identification button at the upper left corner of the screen, facts or images can be added to the document as a whole, similarly to how they were added to a person/organization. In order for the newly added images to appear alongside the document, the PiniTree page must be reloaded in the web browser. It is also possible to create a visual mention for entities visible in the images by pressing the "Select" button and marking the corresponding region in the image. Regular and visual mentions appear as links under the entity at the right side of the screen.

5.3 Word Sense Disambiguation in the PiniTree Editor

Recently in collaboration with Tezaurs.lv, the Latvian online dictionary PiniTree editor has been applied to a new external Use Case – the Word Sense Disambiguation (WSD) corpus creation for the Latvian language. This new functionality is achieved by loading the entire Tezaurs.lv Latvian dictionary along with all word-inflections as aliases into the PiniTree editor – as it was described for Named Entities in above Section 5.2. This work has just started and is showing promising results thanks to unexpectedly high real-time performance of the PiniTree editor even with a database of full language lexicon (millions of aliases). The results of this new experiment will be reported in the final SELMA prototype report under the assumption that this experience will help to extend the approach also to the multilingual setting.

6. External Use Cases: Podcast Producer

6.1 Goal

The Podcast Creator use case is based on a workflow observed in DW's Brazilian language department. The use case's goal is to increase the workflow's efficiency by supporting the journalist in the production of daily audio news bulletins through SELMA.

6.2 Motivation

The Reuters Digital New Report[1] for 2020 highlights the rise of popularity of news podcasts over the last two years. One of the most well-known daily news podcasts is *The Daily* by the New York Times which attracts 2 million daily listeners. Based in this trend, DW's Brazilian department started its own daily news podcast in August 2020. While the monthly usage was at approximately 200.000 impressions in August 2022, it becomes clear how resource-intensive its production is. Currently, a full journalistic shift is required to produce the two daily 6-minute-long bulletins.

DW expects that the demand for news podcasts in other languages will rise. Consequently, it is highly desirable to be able to produce audio news content in one of DW's other 31 languages with a short ramp-up time and with minimal personnel effort.

6.3 Production Steps

The production of a single news bulletin can be subdivided into the following steps. The table shows the duration that is required for each step during the classic, manual process.

Step	What	Approx. duration
1	Research 5 stories	30 min
2	Write 5 stories	60 min

[1] https://www.digitalnewsreport.org/

Step	What	Approx. duration
3	Check stories by colleague	25 min
4	Recording, editing, upload into the system	70 min
5	Add metadata in CMS, create YouTube video, publish on YouTube	45 min
6	Create bi.ly links and publish on Social Media (Twitter & Facebook)	15 min
	Sum	**245 min**

Table 1 *Production Steps in Podcast Creation*

Once the Podcast creator is in use, a good benchmark will be to do these measurements again and compare them with the original production process above.

6.4 The structure of a typical news podcast

Using the example of DW Brazil's daily podcast, the structure of a typical news podcast can be generalized. Figure 25 shows three sections – *Intro*, *Main Block* and *Outro*. The introduction typically contains a welcome message, followed by a selection of news headlines. The main section contains the news stories themselves, possibly separated by interstitial audio called stings (S). The outro contains the good-by message. Each section can contain music or sounds that are being played before the speech begins (M1 and M4), during speech (M2 and M5) and after the speech ended (M3 and M6).

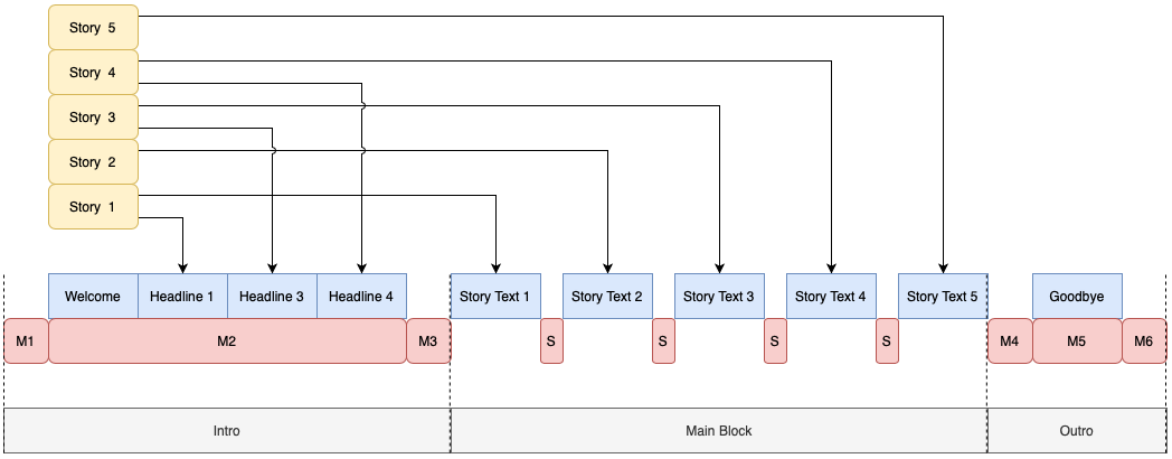

Figure 25 *Generalized structure of a news bulletin*

6.5 Developing a stand-alone app

The Podcast Creator use case is being explored through a stand-alone app running on iOS, iPadOS and macOS. The reason is twofold: first, this approach allows us to refine the idea and the requirements while other use cases are being developed in UC0, UC1 and UC2. Second, as an external application, it allows the consortium to design and develop APIs that expose selected functionalities of UC0, Monitio and possibly plain X to external applications. Third, the stand-alone app will be used to evaluate the use case.

6.6 Using SELMA components to increase efficiency

We expect SELMA modules to be able to support the production process in four areas:

- **Research**: cross-lingual clustering techniques as developed in SELMA and as used in Monitio allows a journalist to identify trending stories worth investigating
- **Writing**: SELMA summarization techniques create a first draft of a story summary that can be used as starting point for writing the news report
- **Recording**: instead of recording speech using a microphone and recording equipment, the written text is converted to synthetic speech using a trained SELMA text-to-speech model
- **Editing**: given the rigid structure of typical audio news podcasts, the mixing of introductory music, background music and interstitial sounds can be done automatically

6.7 The Podcast Creator App

The app is still under development while writing this deliverable, but the screenshot in Figure 26 gives an impression of its functionality. It shows the episode editing screen which allows the user to select the narrator's name as well as the voice used to generate speech.

In addition, each of the three podcast sections can be configured by tapping the corresponding link. As an example, Figure 27 shows a screenshot of the editing options for the podcast's Introduction section, including the music and sounds that are mixed in with the speech.

Figure 28 shows a screenshot of the story editing view. Currently, stories consisting of headline and story text are edited manually here, but we plan to enhance this section by adding import options, allowing a journalist to bring in trending stories summarized by Monitio.

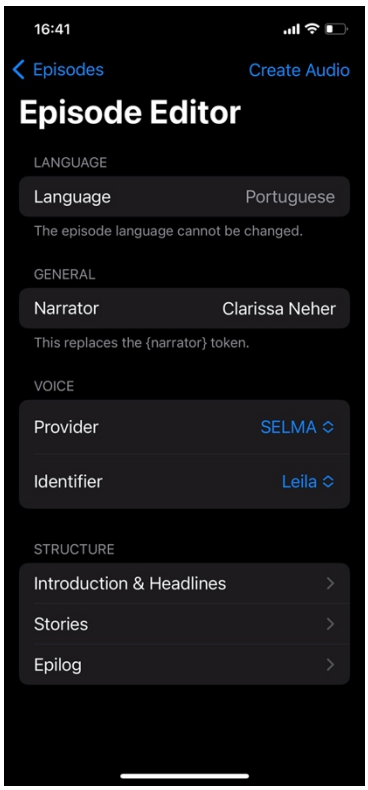

Figure 26 *Episode editor*

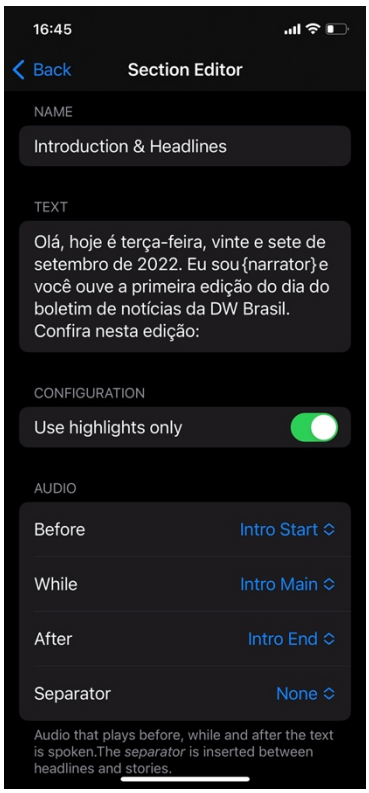

Figure 27 *Section editor*

Figure 28 *Story editor*

6.8 Interaction with SELMA components

Figure 29 shows how the app interacts with other components developed within the context of SELMA.

At the moment, the News Journalist edits her or his scripts manually within the app. An API provided through UC0 renders the text as speech which the app uses to create the final podcast.

Two extensions are foreseen: first, the possibility to import scripts from Github, where the scripts are currently edited for the manual version.

Second, to use SELMA components embedded into Monitio (UC1) to propose trending stories including their summaries. These stories can then be used as a starting point for further editing within the app itself before the final audio is created with the help of UC0.

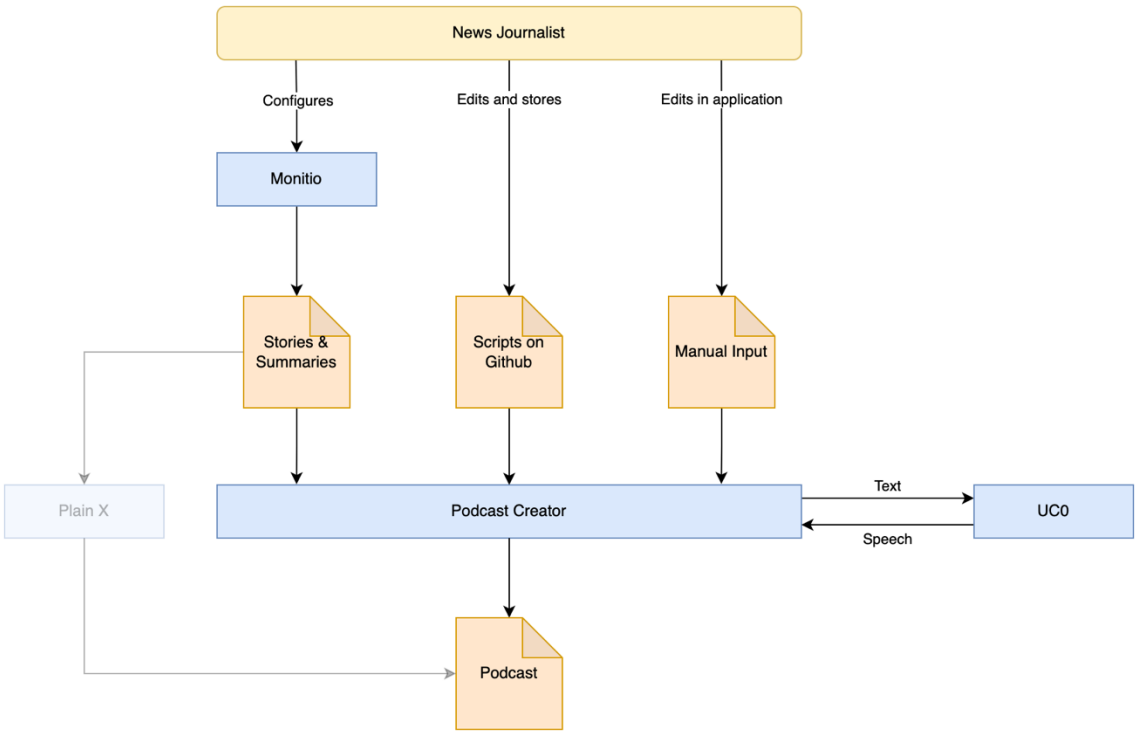

Figure 29 *Interacting with SELMA components*

7. Requirement Implementation Status

In this section we present the list of requirements as defined in D1.1 and updated in D1.2 and the corresponding status in the UC1 and UC2 use cases as:

- Yes: Requirement Implemented
- Ongoing: Requirement Implementation Started
- Not yet: Requirement not Started

There are a few comments added below the requirements where additional references or explanations were needed.

7.1 User & Platform Requirements

	Requirements	Use Case	MoSCoW	UC1 Status	UC2 Status
1	The System allows content to be ingested via standard interfaces as used by news organizations where available	1	Must	Ongoing	-
2	It is possible to use APIs to add content to the SELMA platform	1, 2	Should	Ongoing	Ongoing
3	The System ingests on-demand AV content in MP4 format	1,2	Must	**Yes**	Yes
4	The System monitors selected social media feeds	1	Should	Ongoing	-
5	The System scrapes news articles from websites	1	Must	Yes	-

Module Functionality					
6	The system processes content in the 30+ SELMA languages	1, 2	Must	Ongoing	Yes
7	The system transcribes from audio	1,2	Must	**Yes**	Yes
8	The system provides statistical analysis of ingested material	1	Must	Yes	-
9	The system provides automated translation of all SELMA languages into English as default	1	Should	Ongoing	-
10	The system provides automated translation into other SELMA languages upon request	1, 2	Should	Not yet	Yes
Output					
11	The system offers the possibility to create dashboard user interfaces	1	Should	Ongoing	-
12	The system provides a summarization of individual media items in original language and/or English	1	Must	Ongoing	-
13	User selects ingested channels to monitor	1	Must	Yes	-
14	User subscribes to notifications when relevant content arrives	1	Must	Ongoing	-
Content					

15	The system uses clustering technology to group individual media items into over-arching high-level clusters	1	Must	Yes	-
16	The system provides a visualization which contains a list of all high-level news stories that are relevant to the user, according to the preferences they have set	1	Must	Yes	-
17	On detection of a high-level news story, the system provides a default name for this story based on the clustering technology parameters	1	Could	Yes	-
18	The system offers the user the ability to follow a specific story and subscribe to updates regarding that story	1	Could	Not yet	-
19	The system allows the user to select a high-level news story and view the individual media items that are relevant to it	1	Must	Yes	-
20	For each high-level news story, the system displays a timeline. The system indicates where each individual media item fits on that timeline	1	Could	Not yet	-
21	When viewing an individual media item, the user has the ability to link back to the over-arching high-level	1	Must	Yes	-

	news story to which it is related (in order to view the other media items related to that cluster)				
22	The individual media items for a cluster continue to accrue for a selected period of time. However, the user can indicate to the system that the cluster is no longer relevant before this time has elapsed	1	Could	Not yet	-
23	The system retains a record of clusters that it has identified along with an indication of how many media items were identified by the system	1	Must	Yes	-
Entity Requirements					
24	The system uses entity identification technology to group individual media items by entities	1	Must	Yes	-
25	For each selected entity, the system displays a timeline	1	Must	Yes	-
26	The system carries out entity identification (person, organization, locations and events) using the original language and/or English translation	1	Must	**Yes**	-
27	The system provides a list of identified entities that are relevant to the user,	1	Must	Yes	-

	according to the preferences they have set				
28	The system allows the user to highlight an entity identified by the system and view the individual media items that include the entity	1	Must	Yes	-
29	When viewing an individual media item, the user can link back to the entity to which it is related (in order to be able to view the other media items related to that entity)	1	Could	Not yet	-
30	The system retains a record of entities identified along with an indication of how many media items were identified as relating to those particular entities	1	Must	Yes	-
News Story Requirements					
31	The system uses preferences set by the user to detect news stories of interest to the user	1	Must	**Ongoing**	-
32	For each cluster, the system displays a timeline. The system indicates where each individual media item fits on that timeline	1	Should	Not yet	-
Breaking News Alert Requirements					
33	The system provides breaking news alerts that will correspond to	1	Must	Not yet	-

	individual media items in accordance with the preferences set by the user				
34	A breaking news alert consists of a textual description of the associated media item in the original language and/or in English, along with some specified meta-data (such as date and time)	1	Must	Not yet	-
35	The user selects their preferences for the type of clusters for which they wish to receive alerts (this may be related to particular event or entity types)	1	Must	Not yet	-
36	The user selects the manner and frequency at which they receive event alerts	1	Must	Not yet	-
37	Breaking news alerts are as close to live as is technically possible	1	Must	Not yet	-
General Functional Requirements					
38	The system monitors all input sources selected by the user	1	Must	Yes	-
39	The user can turn English translation on or off	1	Must	**Yes**	-
40	It is possible to associate a user with their team in the System	1, 2	Must	Yes	Yes

41	It is possible to indicate the role of a user in the System	1, 2	Could	Not yet	Yes
42	A user can share a cluster or an individual media item with their team	1	Must	Not yet	-
43	Once a user has indicated that a particular news story or cluster is no longer relevant, individual media items relating to that entity or cluster can be removed from the user's view	1	Must	Not yet	-
44	The user can flag a particular individual media item, entity or cluster and its related media items and indicate that they wish to save them for future reference	1	Must	**Ongoing**	-
45	The user has an option in the system where they can view individual media items, entities or clusters which they chose to save (alongside all the individual media items related to these)	1	Must	**Ongoing**	-
Media Item Requirements					
46	The system provides a clear visual indicator as to the nature of an individual media item (social media, blog, website, AV etc.)	1	Must	Yes	-

47	The user can view the detail of an individual media item (when applicable)	1	Must	Yes	-
48	For an individual AV media item, the user views the video and its original transcription, a translation in the prespecified languages (where applicable) and the meta-data associated with the item	1, 2	Must	**Yes**	Yes
49	For individual AV media items, the system supports a player and editor with tools to 'scrub' through the video, rewind and download (UC2)	1, 2	Must	**Yes**	Yes
50	For individual AV media items, the system supports a player and editor with tools to select elements to 'clip'	1, 2	Could	Not yet	Not yet
51	It is possible for the user to 'clip' an individual AV media item by means of text selection from the transcript	1, 2	Could	Not yet	Not yet
52	For other media occurrences (i.e. textual), the user views the text, its translation in the prespecified language (where applicable) and any meta-data associated with the item	1, 2	Must	Yes	Yes
53	The system provides a 'confidence level indicator' which will indicate how strongly an individual media item	1	Must	Not yet	-

	relates to the existing or suggested clusters				
User Preference Requirements					
54	It is possible to set up a set of default sources that will be frequently used by a particular team	1, 2	Must	Yes	Yes
55	The system contains a predefined list of sources by region	1	Must	**Yes**	-
56	The user can specify entities of particular interest to them	1	Must	Yes	-
57	The user can choose their region of interest in the System	1	Must	**Yes**	-
58	The user can prioritize countries of interest within their region of interest	1	Must	**Yes**	-
59	The system contains a predefined list of regions and countries	1	Must	**Yes**	-
60	It is possible for the user to set time parameters in the system using an indicator such as a time slider to indicate the time period in which they are interested	1	Must	Yes	-
61	In general, the system supports input of preferences in a number of ways: From predefined lists, using data being encountered in the system (the system	1	Must	Yes	-

	will create new entities, events etc.), using free-format text (i.e. search boxes)				
62	The system is adaptable and configurable to the user's preferences	1	Should	Yes	-
Administration Requirements					
63	The user can log into the system	1, 2	Must	Yes	Yes
64	The user can log out of the system	1, 2	Must	Yes	Yes
65	The system supports a super user account	1, 2	Must	**Yes**	Yes
66	The system supports an administrative user (for account management)	1, 2	Must	**Yes**	Yes
67	The administrator can create teams in the system	1, 2	Should	**Yes**	Yes
68	The administrator has typical administrator rights including add new user, remove users, update user profiles as well as the ability to manage data held by the System	1, 2	Should	**Yes**	Yes
Search Requirements					
69	It is possible for the user to conduct a search based on an entity or entities	1	Must	Yes	-

70	It is possible for the user to search based on the type of the media item (e.g. social media, AV etc.)	1	Must	Yes	-
71	It is possible for users to search based on event types	1	Won't	No	-
72	It is possible to take a screen shot (or still frame) from an individual AV media item (rights to be considered here)	1, 2	Could	Not yet	Not yet
73	Entity search is able to handle a range of variable spellings for the same entity	1	Must	Yes	-
74	It is possible to train the system in relation to alternative spellings for searches. For example, it should be possible to link to alternative spellings and indicate that they relate to the same thing	1	Won't	No	-
Input Source Requirements					
75	The system informs the user if a source stops broadcasting	1	Could	Not yet	-
76	The system informs the user if the frequency at which a channel is broadcast, changes	1	Could	Not yet	-
Trend Analysis Requirements					

77	The user has the ability to set their preferences in the system with regard to the types of trend analysis they wish to see in the system	1	Must	Yes	-
78	The system offers the user various options around trend analysis including maps incorporating hotspots, graphs and timelines showing hotspots	1	Must	Ongoing	-
79	The system utilizes saved clusters and media items (that the system has saved along with the number of media items) to conduct trend analysis over a period longer than one week	1	Could	Yes	-
Generate Voice-Over					
80	The system generates a voice-over on request for individual AV media items	2	Must	-	Yes
81	The user chooses whether the voice-over is performed in the original or in the selected translation language	2	Must	-	Yes
82	The system provides a list of available synthetic voices for the user	2	Must	-	Yes
83	The user can choose and/or change which synthetic voice is used for the voice-over	2	Must	-	Yes

84	The user can change the synthetic voice per segment	2	Must	-	Yes
85	The user can amend the voice-over output including phonetics, pauses and pitch	2	Could	-	Yes
Edit Transcription/Translation					
86	The user can edit the transcribed text	2	Must	-	Yes
87	The user can edit the translated text	2	Must	-	Yes
88	The user saves the edited versions of the transcribed and/or translated text	2	Must	-	Yes
89	The user changes the engine and perform the transcription again for the whole text and/or by segment	2	Should	-	Yes
90	The user changes the engine and perform the translation again for the whole text and/or by segment	2	Should	-	Yes
System Learning and User Feedback					
91	The system is trained by the user. The system offers the user a selection of news stories and individual media items relevant to a cluster and the user accepts/rejects them as necessary, thus training the system	1	Must	**Ongoing**	-

92	The system is trained by the user. The system offers the user breaking news alerts and the user accepts/rejects as necessary – training the system to meet the user's preferences	1	Must	Ongoing	-
93	The system learns from the user's corrections and apply them throughout the text	2	Must		**Ongoing**
94	The system applies corrections on different levels (current and future items) based on the preferences set by the user	1, 2	Should	Not Yet	Not Yet
95	The system learns from the user's corrections for entities	1, 2	Must	Ongoing	**Ongoing**
Diversity Detection					
96	The system identifies the binary gender associated with the author's name (if present) of an individual media item	1	Should	Not Yet	-
97	The system provides the number of times each binary gender is mentioned in the media items	1	Must	Ongoing	-
98	The system identifies the gender of the protagonist in each individual media item (if applicable)	1	Should	Not Yet	-

99	The system provides the number of times each binary gender is mentioned in each topical cluster	1	Must	**Yes**	-
100	The system provides a visualization of the gender analysis	1	Must	Not Yet	-
101	The system provides all information pertaining to the diversity data (gender, age, sexual orientation, country of citizenship, medical condition) found on the Wikidata entry of relevant public figures where applicable	1	Must	**Yes**	-
102	The system identifies the gender associated with the named entities of type person, even if the gender information is not available in Wikidata (or the entity is not in Wikidata at all)	1	Should	Not Yet	-

Table 2 *User & Platform Requirements*

7.2 Technical Requirements

	Technical Requirements	**Use Case**	**MoSCoW**	**UC1 Status**	**UC2 Status**
Platform - Orchestration					
P1	Orchestrates processing jobs on the data stream, automatically	1	Must	Ongoing	-

P2	Orchestrates processing jobs on the data stream, on user request	1,2	Must	Not yet	Yes
P3	Allows listening for job results via push notifications (e.g, web sockets)	1,2	Should	Ongoing	Yes
P4	Allows listening for job state changes (errors, job progress)	1,2	Must	Ongoing	Yes
P5	Allows consulting the state of a job request / jobs on an item via an API (e.g., REST)	1,2	Must	Ongoing	Ongoing
P6	Accepts new “job workflow” requests, which may entail running several jobs organized in a graph of dependencies, on an item	1,2	Must	Ongoing	Yes
P7	Orchestrated jobs are eventually applied, meaning, a job cannot be lost - it is either completed successfully or logs an error	1,2	Must	Ongoing	Yes
P8	Orchestration-related configuration changes happen without downtime	1,2	Could	Ongoing	Ongoing
P9	Resilience to the unavailability / downtime of specific workers. Jobs wait until the worker recovers	1,2	Should	Ongoing	Yes

P10	The system allows parallel jobs to execute on the same item if they can be run that way according to the orchestrated job graph	1,2	Could	Ongoing	Yes
Platform - Worker management					
P11	Processing workers use Docker (or equivalent) containerization for deployment	1,2	Must	Yes	Yes
P12	The system is prepared for kubectl-compatible deployment in Kubernetes clusters *(* This direction has changed, since the Consortium will be using the alternative "docker-spaces" as described in D4.3)*	1,2	Should *(Won't*)*	No	No
P13	The system manages the lifetime of different worker containers *(Comment: Through docker-spaces as described in D4.3)*	1,2	Could	Ongoing	Yes
P14	The system scales the number of worker containers according to the corresponding task queue flux *(Comment: Through docker-spaces as described in D4.3)*	1,2	Could	Ongoing	Yes

P15	Worker-related configuration changes (new workers, worker scaling, etc.) happen without downtime *(Comment: Through docker-spaces as described in D4.3)*	1,2	Could	Ongoing	Yes
Platform - Replication and Sharding					
P16	The system allows replication at the worker level	1,2	Must	**Yes**	Yes
P17	The system allows replication at the orchestration controller level *(Comment: Through docker-spaces as described in D4.3)*	1,2	Should	Ongoing	Yes
P18	The system allows replication at the database level *(Comment: As described in D4.3, docker-spaces is currently stateless, and so it doesn't need replication. The core systems of UC1 and UC2 use PostgreSQL though, which supports replication through repmgr and pgbouncer)*	1,2	Should	**Yes**	Yes
P19	The system allows sharding at the orchestration controller level	1,2	Should	Ongoing	Yes

	(Comment: Through docker-spaces as described in D4.3)				
P20	The system allows sharding at the database level *(Comment: As described in D4.3, docker-spaces is currently stateless, and so it doesn't need replication.)*	1,2	Should	Not yet	Not yet
Component - Online News Classification and Clustering					
C1	For each ingested news item, the system attributes a cluster	1	Must	Yes	-
C2	For each ingested news item, the system attributes an IPTC topic	1	Must	Yes	-
C3	The system clusters documents in an online fashion, e.g., without having to revisit all past decisions	1	Must	Yes	-
C4	The system clusters documents natively in all 30 SELMA languages	1	Should	Ongoing	-
C6	The system leverages user feedback on clustering decisions to improve future decisions	1	Should	Not yet	-
Component - Summarization					
S1	For each ingested news item, the system generates a summary	1	Must	Yes	-

S2	The system generates summaries natively in all 30+ SELMA languages	1	Should	Not yet	-
S3	The system generates summaries either from original text article or video transcripts	1	Should	Not yet	-
S4	The system leverages user feedback on summarization results to improve future summaries	1	Should	Not yet	-
Component - Machine Translation					
M1	The system translates a textual document by demand	1,2	Must	Not yet	**Yes**
M2	The system translates a video by demand	1,2	Must	Not yet	Yes
M3	The system translates between all 30+ SELMA languages	1,2	Should	Not yet	Ongoing
Component - Automatic Transcription					
R1	The system automatically transcribes an ingested video or audio file	1,2	Must	**Yes**	Yes
R2	The transcription is enriched by punctuation	1,2	Must	Ongoing	Ongoing

R3	The transcription is enriched by speaker information *(* There have been developed SELMA transcription components which support speaker information although it is not integrated in UC2 yet)*	1,2	Should	Not yet	Ongoing*
R4	The transcription is enriched by named entity labeling	1,2	Must	Ongoing	Not yet
R5	The system transcribes all 30 SELMA languages	1,2	Should	Not yet	Yes
Component - Entity Recognition and Linking					
E1	For each ingested news item, the system detects named entities	1	Must	Yes	-
E2	For each ingested news item, the system links named entities to a knowledge base	1	Must	Yes	-
E3	The system links entities natively in all 30 SELMA languages, leveraging crosslingual representations	1	Should	Ongoing	-
E4	For each ingested news item, the system attributes a gender for each person named entity detected in the news item.	1	Should	Yes*	-

	(Yes, for named entities of type people that can be linked to Wikipedia/Wikidata)*				
Component - Story Segmentation					
G1	Each ingested long audio segment gets split into meaningful units *(Comment: preliminary work ongoing on the research side. Also, a silences-based splitting component has been integrated into UC1 and UC2 meanwhile.)*	1,2	Could	*Ongoing**	*Ongoing**
G2	Speaker clustering is used to create speaker independent units *(* There have been developed SELMA transcription components which support speaker information although it is not integrated in UC2 User Interface yet, but the data is there)*	1,2	Could	Not yet	*Ongoing**
G3	Speaker recognition automatically identifies the original speaker in each segment *(Comment: preliminary work ongoing on the research side)*	1,2	Could	Not yet	Not yet
G4	Topic segmentation is used to separate by spoken content	1,2	Could	Not yet	Not yet

Component - Voice Conversion Synthesis					
V1	The text-to-speech system automatically produces voices in Latvian, German, and French	2	Must	-	Ongoing
V2	The text-to-speech system will be improved to better handle foreign words	2	Must	-	Not Yet
V3	A speech-to-speech translation system works at least on one language pair	2	Should	-	Ongoing
V4	A speech-to-speech translation system works at least on one language pair and can generate a synthetic voice in the target language close to the natural voice in the source language	2	Could	-	Ongoing

Table 3 *Technical Requirements*

8. Conclusion

This document presents the current state of the prototypes which address the three main use cases, Multilingual Media Monitoring (UC1), Multilingual News Content Production (UC2) and SELMA NLP Service Orchestration (UC0). Two external use-cases which integrate SELMA technology are also discussed - Pinitree and the Podcast Creator.

Within Use Case 1 (UC1), we are integrating results from the SELMA research tasks into the *Monitio* product, a Media Monitoring platform under development by Priberam, available at https://app.monitio.com. Within Use Case 2 (UC2), we are integrating results from SELMA into the *plain X* product, a Multilingual News Media Content Production platform under development by Priberam and Deutsche Welle, available at https://app.plain-x.com. The SELMA Open-Source Platform (UC0) is available at https://selma-project.github.io/.

The list of requirements from D1.2 is revisited and the corresponding progress reported.

Bibliography

Barzdins, G., Gosko, D., Cerans, K., Barzdins, O. F., Znotins, A., Barzdins, P. F., Gruzitis, N., Grasmanis, M., Barzdins, J., Lavrinovics, I., Mayer, S. K., Students, I., Celms, E., Sprogis, A., Nespore-Berzkalne, G., Paikens, P. (2020b). Pini Language and PiniTree Ontology Editor: Annotation and Verbalisation for Atomised Journalism. *In: ESWC 2020 Satellite Events. LNCS, Volume 12124, pp. 32-38.*

Peteris Paikens; Guntis Barzdins; Afonso Mendes; Daniel Ferreira; Samuel Broscheit; Mariana S. C. Almeida; Sebastiao Miranda; David Nogueira; Pedro Balage; Martins, Andre F. T. (2016a). SUMMA at TAC Knowledge Base Population Task 2016, *DOI: 10.5281/zenodo.827317*

Znotiņš, Artūrs & Barzdins, Guntis. (2020). LVBERT: Transformer-Based Model for Latvian Language Understanding. *Baltic HLT, IOS Press, pp. 111-115, DOI 10.3233/FAIA200610.*

Znotins A, Cirule E. (2018). NLP-PIPE: Latvian NLP Tool Pipeline. *In: Human Language Technologies - The Baltic Perspective. vol. 307. IOS Press; 2018. p. 183–189.*

Paikens P. (2016b). Deep Neural Learning Approaches for Latvian Morphological Tagging. *In: Baltic HLT; 2016. p. 160–166.*

João Santos, Afonso Mendes and Sebastiao Miranda, “Simplifying News Clustering Through Projection from a Shared Multilingual Space” in Text2Story at ECIR, 2022.