FP7-ICT-2011.4.2
Contract no.: 288342
www.xlike.org

XLike

Deliverable D1.2.2

Requirements for demonstrator

Editor:	Achim Rettinger, KIT
Authors:	Achim Rettinger, KIT; Lei Zhang, KIT; Blaž Fortuna, JSI; Gregor Leban, JSI; Aljoša Rehar, STA; Pat Moore, Bloomberg; Esteban García Cuesta, ISOCO
Deliverable Nature:	R
Dissemination Level: (Confidentiality)	PU
Contractual Delivery Date:	M15
Actual Delivery Date:	2.4.2013
Suggested Readers:	All project partners
Version:	1.0
Keywords:	use case requirements; demonstrator

Disclaimer

Full Project Title:	Cross-lingual Knowledge Extraction
Short Project Title:	XLike
Number and Title of Work package:	WP1 – Definition and Data Provision
Document Title:	D1.2.2 – Requirements for demonstrator
Editor (Name, Affiliation)	Achim Rettinger, KIT
Work package Leader (Name, affiliation)	Blaž Fortuna, JSI
Estimation of PM spent on the deliverable:	7

Executive Summary

This document provides the functional specification for the demonstrator, based on the previous deliverable D1.2.1, which presents description of the functional requirements for the early prototype. In order to keep pace with the technology developments within the project, the requirements will be revised at the beginning of each year based on the feedback from the validation process and this document will serve as a basis for the deliverable D1.2.3, which will provide the functional specification for the final prototype.

The goal of this task T1.2 is to gather and survey the requirements for the XLike system based on the case studies. The main outcome of the task will be three requirements documents on (a) how to integrate solution for cross-lingual information linking and knowledge extraction with the business processes within the companies, (b) to define a set of possible services built on the top of the XLike technology which will be used within case studies and (c) to position supported languages in the context of case studies.

This deliverable is the second outcome of the task T1.2, which provides functional specifications for the demonstrator, based on the feedback from the early prototype, and the technology developed in the second year of the project.

Table of Contents

List of Figures

List of Tables

Abbreviations

STA	Slovenian Press Agency
BLP	Bloomberg

1 Introduction

The purpose of this document is to gather and survey the functional requirements for the demonstrator of the XLike project. We will review the use cases based on the feedback from the early prototype and identify the new requirements needed by the use cases.

1.1 Use Cases

The following two use cases will be evaluated for major languages (English, Spanish, German, and Chinese), as well as minority languages (Slovene).

- ***Bloomberg (BLP)*** – **financial news source** of information for businesses and professionals. Bloomberg combines analytic, data, news, display and distribution capabilities to deliver critical information via their service and multimedia platforms. Bloomberg's media services cover the world with more than 2,200 news and multimedia professionals at 146 bureaus in 72 countries.
- ***Slovenian Press Agency (STA)*** – national press agency covering domestic and international events. STA offers **general news service** in Slovenian and daily English service and is the only provider of daily news in English for the expatriate community in Slovenia and for English-speaking readers abroad.

The use cases have been carefully selected in order to demonstrate the advantages of the project results, in respect to the three issues from the vision statement. Both case studies are news agencies. The first, Bloomberg, is mainly focused on fast and accurate delivery of financial and business news in English. The second, Slovenian Press Agency, is focused on delivering general news, in English and Slovene with focus on events happening in Slovenia or related to Slovenia.

As introduced in the deliverable D1.2.1 the project will cover four general applications:

- **Cross-lingual Summarization**
- **Cross-lingual Contextualization**
- **Cross-lingual Personalization**
- **Cross-lingual Plagiarism Detection**

Regarding the Bloomberg use case, identifying the related and relevant articles and displaying it as a ranked list (cross-lingual recommendation) are the main tasks in Y1. The goal for Y2 is to extend this by taking users' history on Bloomberg.com into account (cross-lingual personalization). Articles are selected in a way as to optimize click rate on recommendations (number of clicks / number of page views). For cross-lingual contextualization, the task is new with respect to Y1. The goal is to help social media people at Bloomberg advertising their content in specific regions.

In the case of STA, the Y1 task includes a combination of linking their articles with other similar articles in other languages (cross-lingual contextualization) and from this set, isolating ones that look to much as an exact translation (cross-lingual plagiarism detection). For Y2, we are more interested in the identification of events from news articles (cross-lingual summarization) and their storage and representation in an event registry.

2 Requirements for Demonstrator

In this section, we will review the use cases. Based on the detailed analysis of case studies needs and the feedback from the early prototype, we derive the functional requirements for the demonstrator that are essential to the use cases. To make this document self-contained, the use cases and requirements covered by D1.2.1 will also be included and updated.

2.1 Bloomberg Use Cases

Bloomberg's business is the delivery of financial information. The core of their business is based on Bloomberg Terminals, a specialized platform for financial professionals. Besides this, they also maintain a more mainstream oriented news portal at Bloomberg.com. The Bloomberg use case in XLike will focus on the website, by evaluating techniques for cross-lingual integration of news articles.

2.1.1 Related or Relevant Articles (Y1 + Y2)

Bloomberg.com provides personalized list of suggested articles along each article. The list is assembled from the recent Bloomberg articles and custom fitted for the specific user, based on his/her history. This task in Bloomberg use case is to extend the suggested articles by including external mainstream sources across more languages, which is already included in Y1. The goal for Y2 is to extend this by taking users' history on Bloomberg.com into account.

Formally, the task is defined as follows. Given a user u, with visit history $H(u) = \{a_1, ..., a_n\}$, identify relevant recent articles from multi-lingual news stream. All articles from $H(u)$ are in English and published by Bloomberg. Assembling a relevant recent articles list requires cross-lingual integration with Bloomberg.com articles.

Table 1. Related or Relevant Articles in Bloomberg Use Case.

Identifier	**UC1**
Name	**Related or Relevant Articles**
Application	Cross-lingual contextualization and Cross-lingual recommendation
Input	a) Bloomberg news steam b) Mainstream news stream c) Social media stream
Output	a) A set of matching articles b) Summarization and Visualization of the matching articles
Languages	XLike languages
Related tasks in Y1	**T1.3 – Data infrastructure** must provide sufficient corpora of existing articles for experimentation and sufficient coverage of relevant mainstream news services for article tracking **T4.1 – Statistical cross-lingual document linking** used for article linking and user recommendation **T5.2 – Information visualization** used for visualization of matching articles
Related tasks in Y2	**T2.3 – Analysis of informal languages** used for construction of parallel collections of informal textual expressions (words, phrases, sentences) paired with their formal counterpart **T2.4 – Extracting structure from informal language corpora** used for extraction of bag-of-words vector from social media feeds (blogs, Twitter, Facebook)
Evaluation	a) Accuracy of identified articles matching Bloomberg.com articles b) Relevancy of recommended articles (focused user study)

2.1.2 Content Advertising (Y2)

This task is new with respect to Y1. The goal is to help social media people at Bloomberg advertising their content in specific regions. For example, there is a special Bloomberg.com fan page on Facebook for Germany featuring articles that are mostly relevant or of interest to Germans.

The task will be implemented in a form of a supporting tool, which for a particular region:

- identifies relevant topics at the moment by monitoring local mainstream news and social media,
- at the beginning, suggests recent articles from Bloomberg.com that would be of relevance in near-real time (e.g. each few minutes),
- in the later stage, automatically publishes relevant articles from Bloomberg.com to various feeds in some larger predefined intervals (e.g. hour), depending on the region, feed, and availability of relevant Bloomberg.com articles.

Relevance of article is a combination of several scores:

- top topics,
- age of the article in combination with how evergreen the content is (e.g. earnings report have shorter lifecycle compared to opinion articles),
- last time similar topic was published to the feed.

The task will need to support the following feeds:

- Facebook,
- Google+,
- Twitter,
- Orkut.

Table 2. Content Advertising in Bloomberg Use Case.

Identifier	**UC2**
Name	**Content Advertising**
Application	Cross-lingual contextualization and Cross-lingual recommendation
Input	a) Bloomberg news stream b) Social media stream
Output	a) Recent Bloomberg articles that would be of relevance in near-real time b) Relevant Bloomberg articles in some predefined intervals (e.g. hour) c) Summarization and Visualization of the relevant articles
Languages	XLike languages
Related tasks in Y1	**T1.3 – Data infrastructure** must provide sufficient corpora of existing articles for experimentation and sufficient coverage of relevant mainstream news services for article tracking **T4.1 – Statistical cross-lingual document linking** used for suggestion of the relevant articles **T5.2 – Information visualization** used for visualization of suggested articles
Related tasks in Y2	**T2.3 – Analysis of informal languages** used for construction of parallel collections of informal textual expressions (words, phrases, sentences) paired with their formal counterpart **T2.4 – Extracting structure from informal language corpora** used for extraction of (a) bag-of-words vector, (b) syntactic and semantic triples from social media feeds (blogs, Twitter, Facebook) **T5.2 – Information visualization** used for visualization of suggested articles
Evaluation	a) Number of suggested Bloomberg articles in different intervals b) Relevancy of suggested articles (focused user study)

2.2 STA Use Cases

STA publishes article in Slovene and English. Primary market for Slovene articles is Slovenia, which either republish the articles or use their content as input to their own articles. The primary target for English articles is foreign news agencies (e.g. Xinhua) or news outlets (e.g. Bloomberg). Some of foreign agencies also use Slovene articles.

2.2.1 Article Tracking (Y1+Y2)

At the moment, STA does not have means to track the republishing of its articles. There are two business cases for why such a tracking is important. First, the main income of the agency is licensing its content, and publish unlicensed material requires their attention. Second, knowing which articles are republished by their subscribers helps the agency to better understand their market, and to provide better coverage for the events relevant for them.

More formally, article tracking should detect the following modification operations to article a_1:

Article a_2 is a near-copy of the other: $a_1 \approx a_2$

Article a_2 is a translation of the original article: $T(a_1) \approx a_2$

Article a_2 includes article a_1: $a_1 \subset a_2$

Article a_2 includes part of article a_1: $P(a_1) \subset a_2$

Ideally, any above operations, or combination of them, should be detectible. Given the technology available, work on the first two operations will be emphasized.

Table 3. Article Tracking in STA Use Case.

Identifier	**UC3**
Name	**Article Tracking**
Application	Cross-lingual contextualization and Cross-lingual plagiarism detection
Input	a) STA article stream b) Mainstream news stream c) Social media stream
Output	a) For each STA article, a set of matching articles from the mainstream news stream b) Summarization and Visualization of the matching articles
Languages	a) STA article are in Slovene and English b) Focus on XLike languages
Related tasks in Y1	**T1.3 – Data infrastructure** must provide sufficient corpora of existing articles for experimentation and sufficient coverage of relevant mainstream news services for article tracking **T4.1 – Statistical cross-lingual document linking** core technique used for the development of this component **T5.2 – Information visualization** used for visualization of matching articles
Related tasks in Y2	**T3.1– Approximate text annotation** with cross-lingual semantic repositories used for entity disambiguation **T5.2 – Information visualization** used for visualization of matching articles
Evaluation	a) Number of tracked articles within the source language b) Number of tracked articles across languages

2.2.2 Topic and Entity Tracking (Y1)

STA covers topics related to Slovenia or Slovenian entities (E.g. companies, athletes). As such, tracking relevant news is an important part of editors' daily routine. Technologies developed within XLike project can improve this process by providing tools for detecting relevant articles across languages and media (mainstream, social media). This task is not new for Y2.

Formally, topic or entity tracking can be seen as a filter applied to a stream of articles. An article is retained by the filter if it matches the topic, or is related to the entity. Topics can be defined as a standard classification task, with articles on the input and set of matching topics on the output. Entities can be detected using named-entity extractors.

For popular topics or entities, the filter can retain a large amount of articles. The information contained within these articles can be visualized or summarized to help the editors in skimming through the content, to identify relevant events.

Similar to Bloomberg use case, we will extend the early prototype in Y1 by employing the word-sense disambiguation and machine translation techniques to improve the quality of topic and entity tracking in STA use case.

Table 4. Topic and Entity Tracking in STA Use Case.

Identifier	**UC4**
Name	**Topic and Entity Tracking**
Application	Cross-lingual contextualization and Cross-lingual summarization
Input	a) A list of topics or entities b) Mainstream news stream
Output	a) A set of matching articles b) Summarization and Visualization of the matching articles
Languages	XLike languages
Related tasks in Y1	**T1.3 – Data infrastructure** must provide sufficient corpora of existing articles for experimentation and sufficient coverage of relevant mainstream news services for article tracking **T2.1 – Shallow linguistic processing** of formal language used for entity extraction from mainstream articles **T3.1 – Approximate text annotation** with cross-lingual semantic repositories used for entity disambiguation **T4.1 – Statistical cross-lingual document linking** used for topic tracking **T5.2 – Information visualization** used for visualization of matching articles
Evaluation	a) Precision of detected articles b) Recall of detected articles on controlled dataset c) Effectiveness of summarization and visualization (user questionnaire)

2.2.3 Event Identification (Y2)

This task is new with respect to Y1. We start by assuming that each news article is describing a particular event. Our goal is to develop methodology to identify the event mentioned in the article and describe it with a set of properties (such as time of the event, involved entities, keywords, etc.). The developed algorithms will be able to assign each article to an event. The identified event will be either new (when this will be the first article describing it) or existing (when we have already seen other articles describing it). Events will be stored in an event registry that will provide querying and editing functionality.

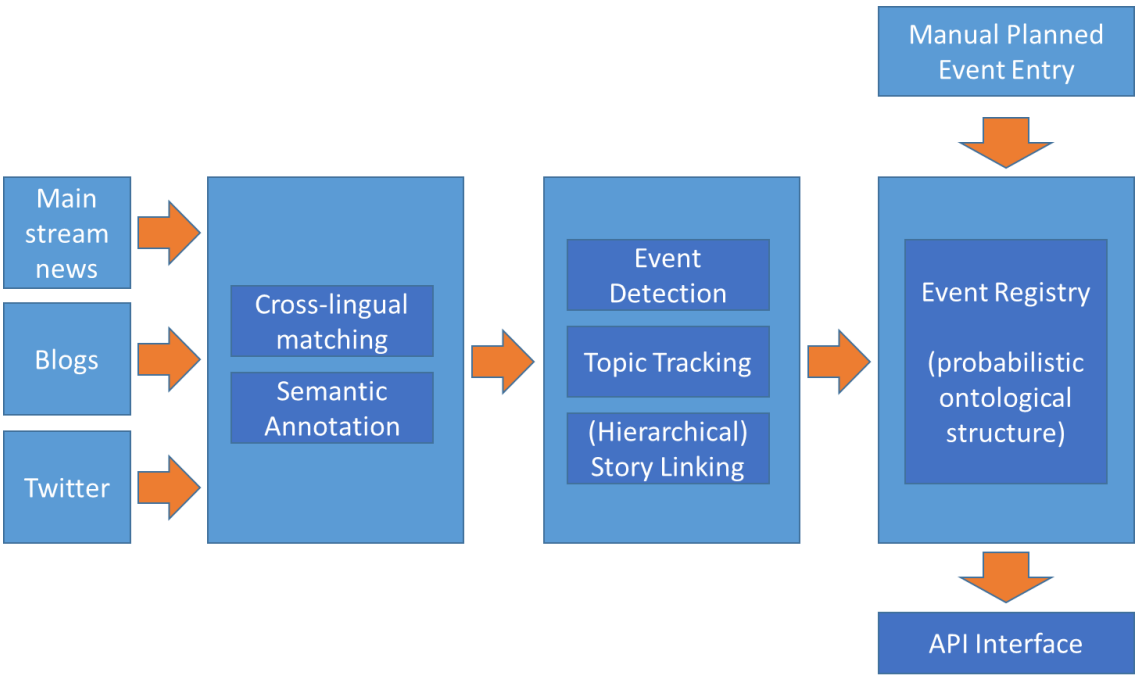

Figure 1. Architecture overview.

Expected input

In order to identify events we will use as input the articles processed and annotated as it is currently done on newsfeed.ijs.si. For each article we expect to get:

- the content of the article with all available article's metadata (time, publisher, language, …)
- a set of disambiguated named entities mentioned in the article and possibly weighted by their importance
- a set of most relevant DMOZ categories for the article
- a set of IDs of most similar articles in different languages (obtained by CCA)

Events

An event can be anything that is happening in the world. Examples of events are Google's I/O conference happening May 15-17 2013, Felix Baumgartner's jump from helium balloon on October 14, 2012 and Vietnam War (1955-1975). Already from these three examples we can see that events can be quite diverse. In order to be able to detect various kinds of possible events we will use several features from the articles. The features we currently plan to use for identifying (and later describing) an event are:

- named entities extracted from the article
- content of the article
- time of the event (extracted from the article)
- publishing time of the article
- event type
- additional article meta data

The two core features that will be most relevant for identifying and distinguishing between different events are named entities and the content of the article. Since news articles can be written in different languages it is important to mention how event detection will work across languages. Use of named entities is not problematic since different named entity recognizers will be used for each language and the obtained entities once disambiguated will be language-independent. Finding events using the article content is however more complicated since text comparison across languages won't yield good results. In order to find descriptions of the same event in other languages we will rely on mapping the article text into language-neutral space using the canonical correlation analysis. Each news article provided by newsfeed already contains information about *n* most similar recent articles in other languages. By inspecting these articles we should be able to identify descriptions of the same event in other languages.

Beside named entities and text, we mentioned also other features relevant for event detection. Time of the event, for example, is a very important feature. Event can occur at a particular time point (e.g. 7:15 on May 12, 2013) or during a time interval (e.g. May 15-17, 2013). The task of information extraction will be to accurately identify and extract time mentions that can occur in various forms. Time reference(s), if identified in article text, should serve as an additional feature in determining the correct event. Similarly important is also the publishing time of the article – articles published only few days apart are more likely about the same topic than articles published months or years apart.

Another feature of the event is also event type. Two types of events are, for example, an earthquake and a football match. Being able to identify the type of the event allows us to identify an additional set of constraints that the article has to match. In case of a football match we should be able to identify in text at least the names of the teams that played, the location, date and potentially what was the result (if the game was in the past). In case of earthquake, we should be able to identify the location of the earthquake and its magnitude. In order to use event types we would first need to define a hierarchy of possible event types and define constraints (requirements) for each type. Once possible event types are defined, the identified event type for an article can be used as learning feature for identifying the event described in the article. The event types will play an even more important role when describing identified events since they provide important semantic frame for each event.

Based on the described features that we will use for identifying events we expect that the structure used for describing the event should contain at least the following properties:

- title of the event
- a weighted list of named entities relevant for the event. This list could be updated as articles are associated or disassociated from the event
- a list of cluster IDs. Each cluster represents a group of articles describing the event, possibly in different languages
- a list of keywords describing the event (in different languages)
- event type
- potentially: Time of event (if valid)
- potentially: A list of sub-events. Vietnam War is a huge and long event that can consist of several smaller events that span across 20 years.
- a list of articles associated with the event

Table 5. Event Identification in STA Use Case.

Identifier	**UC5**
Name	**Event Identification**
Application	Cross-lingual contextualization and Cross-lingual summarization
Input	a) Mainstream news stream b) Social media stream
Output	a) New event templates b) A set of detected events c) Visualization of the detected events
Languages	XLike languages
Related tasks in Y1	**T1.3 – Data infrastructure** must provide sufficient corpora of existing articles for experimentation and sufficient coverage of relevant mainstream news services for article tracking **T2.1 – Shallow linguistic processing** of formal language used for entity extraction from mainstream articles **T3.1 – Approximate text annotation** with cross-lingual semantic repositories used for entity disambiguation **T4.1 – Statistical cross-lingual document linking** used for topic tracking **T5.2 – Information visualization** used for visualization of matching articles
Related tasks in Y2	**T2.2 – Deep linguistic processing** of formal language used for extraction of

	entities, relations and triples from mainstream articles **T3.1 – Approximate text annotation** with cross-lingual semantic repositories used for entity disambiguation **T3.2 – Word-sense-disambiguation** through ontological constraints used for entity disambiguation **T4.2 – Semantic graphs construction** used for merging relations extracted in WP2 into semantic graphs and linking annotations produced in WP3 into semantic graphs **T4.3 – Event extraction from semantic graphs** used for extraction of event templates and their population based on semantic graphs produced in T4.2 **T5.2 – Information visualization** used for visualization of matching articles
Evaluation	a) Accuracy of detected new event templates and the events b) Effectiveness of summarization and visualization (user questionnaire)

Event registry

As articles will be processed an event registry will be kept and updated. For each event we will maintain a structure similar to the one defined at the end of previous section. The registry will provide the users with search capabilities such as finding different articles (potentially in different languages) describing the same event or finding events based on event type or time constraints. Users with sufficient privileges will also be allowed to perform manual updating of events. This updating would include the ability to change all possible properties of the event, as well as merging and splitting capabilities.

2.3 Functional Requirements for Demonstrator

According to the detailed analysis of both use cases, the functional requirements for the demonstrator described below must be implemented to allow the users to perform each use case. Each requirement includes a short description and is referred to the corresponding tasks in the project. To make this document self-contained, the requirements defined in D1.2.1 will also be included and updated.

Table 6. Functional Requirement of Newsfeed.

Identifier	**RQ1**
Name	**Newsfeed**
Description	To provide a clean, continuous, real-time aggregated stream of news articles from RSS-enabled sites across the world.
Input	A list of RSS feeds and a subset of Google News
Output	a) Potential new RSS sources mentioned in the HTML b) Links to news articles b) Clear text version of the article body
Languages	XLike languages
Motivation	Entity tracking (BLP), Related or relevant articles (BLP), Article tracking (STA), Topic and entity tracking (STA), Content Advertising (BLP), Event Identification (STA)
Related task	**T1.3 – Data Infrastructure.**
Evaluation	a) Source diversity b) Data volume c) Latency d) Language distribution

Table 7. Functional Requirement of Shallow Linguistic Processing.

Identifier	RQ2
Name	**Shallow Linguistic Processing**
Description	To prepare and develop tools for shallow linguistic processing of formal language corpora.
Input	A sentence, a document or a set of documents in any XLike language
Output	a) The language given content is in b) Sentences, words, tokens, POS tags, ect. c) Named entities
Languages	XLike languages
Motivation	Entity tracking (BLP), Topic and entity tracking (STA)
Related task	**T2.1 – Shallow linguistic processing of formal language.**
Evaluation	a) Accuracy of the resulting lexical items b) Performance of the shallow linguistic processing

Table 8. Functional Requirement of Text and Semantic Annotation.

Identifier	RQ3
Name	**Text and Semantic Annotation**
Description	To annotate documents with cross-lingual language and knowledge resources
Input	a) One or a set of documents in any XLike language b) Relevant knowledge resources, such as Wikipedia, Linked-Open-Data and Cyc
Output	a) Translation of existing lexical groundings from knowledge resources to all required languages b) Annotations of words in the document with one or more concepts from knowledge resources based on their lexical information
Languages	XLike languages
Motivation	Entity tracking (BLP), Topic and entity tracking (STA)
Related task	**T3.1 – Approximate text annotation with cross-lingual semantic repositories.** **T3.2 – Word-sense-disambiguation through ontological constraints.**
Evaluation	a) Precision of the text annotation b) Recall of the text annotation c) Performance of the translation and the text annotation

Table 9. Functional Requirement of Cross-lingual Document Linking.

Identifier	RQ4
Name	**Cross-lingual Document Linking**
Description	To develop techniques for cross-lingual document linking based on statistical models and cross-lingual knowledge resources
Input	One document in any XLike language
Output	A set of documents based on their similarity to the input document across languages
Languages	XLike languages
Motivation	Entity tracking (BLP), Related or relevant articles (BLP), Article tracking (STA), Topic and entity tracking (STA)
Related task	**T4.1 – Statistical cross-lingual document linking.**
Evaluation	a) Precision of linked documents b) Recall of linked documents c) Performance of the model training and document linking

Table 10. Functional Requirement of Information Visualization.

Identifier	**RQ5**
Name	**Information Visualization**
Description	To employ techniques for text and network visualizations for real-time cross-lingual streams to show visual summary of information dynamics across sources, languages and time
Input	Documents or corpus representations from the previous stage
Output	Visualization of the input documents or corpus representations
Languages	XLike languages
Motivation	Entity tracking (BLP), Related or relevant articles (BLP), Article tracking (STA), Topic and entity tracking (STA), Content Advertising (BLP), Event Identification (STA)
Related task	**T5.2 – Information visualization.**
Evaluation	Effectiveness of visualization (user questionnaire)

Table 11. Functional Requirement of Deep Linguistic Processing.

Identifier	**RQ6**
Name	**Deep Linguistic Processing**
Description	To prepare and develop tools for deep linguistic processing of formal language corpora
Input	A sentence, a document or a set of documents in any XLike language
Output	a) Predicative grammatical relations, like b) Agent-predicate-object triples c) Adjunct relations such as temporal, locative, causal, etc
Languages	XLike languages
Motivation	Entity tracking (BLP), Topic and entity tracking (STA), Event Identification (STA)
Related task	**T2.2 – Deep linguistic processing of formal language.**
Evaluation	a) Accuracy of the resulting predicative relations b) Performance of the deep linguistic processing

Table 12. Functional Requirement of Informal Languages Processing.

Identifier	**RQ7**
Name	**Informal Languages Processing**
Description	To prepare and develop tools for informal languages processing
Input	Informal textual units (words, phrases and sentences) from social media feeds (blogs, Twitter, Facebook) in any XLike language
Output	a) Parallel collections of informal textual expressions paired with their formal counterpart b) Bag-of-words vector extracted from informal language sources c) Syntactic and semantic triples extracted from informal language sources
Languages	XLike languages
Motivation	Related or relevant articles (BLP), Article tracking (STA), Content Advertising (BLP)
Related task	**T2.3 – Analysis of informal languages.** **T2.4 – Extracting structure from informal language corpora.**
Evaluation	a) Accuracy of the resulting items b) Performance of the informal languages processing

Table 13. Functional Requirement of Semantic Graphs Construction.

Identifier	**RQ8**
Name	**Semantic Graphs Construction**
Description	To prepare and develop tools for semantic graphs construction
Input	a) A set of triples extracted from documents b) A sequence of annotations extracted from documents
Output	Semantic graphs by merging extracted triples and linking annotations
Languages	XLike languages
Motivation	Event Identification (STA)
Related task	**T4.2 – Semantic graphs construction.**
Evaluation	a) Accuracy of the resulting semantic graphs b) Performance of the semantic graphs construction

Table 14. Functional Requirement of Event Extraction.

Identifier	**RQ9**
Name	**Event Extraction**
Description	To prepare and develop tools for event extraction from semantic graphs
Input	Semantic graphs
Output	Event templates and their population (event)
Languages	XLike languages
Motivation	Event Identification (STA)
Related task	**T4.3 – Event extraction from semantic graphs.**
Evaluation	a) Accuracy of the resulting events b) Performance of the event extraction

3 Conclusions

With respect to the main result of the task T1.2, jointly with this deliverable, there are three requirements documents, one at the beginning of each year, providing detailed analysis of case studies needs and service opportunities using innovations and solutions from the project for the prototype to be developed in the respective year.

This deliverable is the second outcome of the task T1.2, which provides functional specifications for the demonstrator, based on the feedback from the early prototype, and the technology developed in the second year of the project. It based on the M3 deliverable **D1.2.1 – Requirements for early prototype**, which presents description of the functional requirements for the early prototype, and will be further used as source of information for the M26 deliverable **D1.2.3 – Requirements for fully functional prototype**, which will provide the functional specification for the final prototype.