X Modal
X Cultural
X Lingual
X Domain
X Site
Global OER Network

Grant Agreement Number: 761758
Project Acronym: X5GON
Project title: X5gon: Cross Modal, Cross Cultural, Cross Lingual, Cross Domain, and Cross Site Global OER Network
Project Date: 2017-09-01 to 2020-08-31
Project Duration: 36 months
Document Title: D2.4 – Final Evaluation Report
Author(s): Erik Novak
Contributing partners: JSI
Date:
Approved by:
Type: P
Status: Draft
Contact: Erik Novak (erik.novak@ijs.si)

Dissemination Level		
PU	Public	x
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Revision

Date	Lead Author(s)	Comments
26/11/2020	Erik Novak	First draft
15/12/2020	Erik Novak	Added Content to Different Sections
16/12/2020	Ayse Saliha Sunar	Internal Review and Comments
22/12/2020	Kristijan Perčič	Review and Comments

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

Acronyms	Definitions
OER	Open Educational Resource
API	Application Programming Interface
JSON	JavaScript Object Notation
URL	Uniform Resource Location

ABSTRACT

This document contains the final evaluation report of the X5GON products and services. We perform a technical evaluation with the objective to identify the current status of the services viewed from different technical lenses – going from the hardware requirements and security, maintainability and integration, to monitoring and alerting the admins. We found that the X5GON system has a firm technical foundation but does require improvements to ensure it will last longer.

This document also includes the basic statistics of the X5GON database and highlights regarding two open problems in open education: i) automatic OER identification and indexing, and ii) OER availability decision. Solving these problems will improve the visibility of the scattered OERs.

1. INTRODUCTION

The purpose of this document is to report on the final evaluation of the X5GON services and products. The evaluation covers several technical aspects that provide insight in how technically sound the solutions developed in the project are. We justify the final evaluation with the final database statistics. Finally, we present two of the open problems in open education that we think it is worth considering for further improvements.

The document is structured as follows. Section 2 contains the evaluation report. The services and products have been evaluated based on the following technicalities: hardware architecture, security, integration, maintainability, monitoring, alerting, reusability, and documentation. These technical points are identified as important to determine the soundness of the technical solutions. Section 3 presents the database statistics. It contains the final number of OER, its providers, the number of transcriptions and translations per languages, as well as the number of user visits and interactions with the X5GON recommender engine. Section 4 describes two of the open problems in open education. The first is how to automatically find and index OER resources around the globe. The second is how to decide when an OER is not available anymore. We find that solving these problems is important for the OER community. Finally, we conclude the document in Section 5.

2. EVALUATION REPORT

In this section we present the evaluation report. The evaluation has been performed from the more technical perspective with the objective to identify the current overall status of the services and products – this include the security of services, the difficulty of maintaining existing services and integrating new ones, how we monitor the service processes, and if they can be reused in a domain outside of open education.

2.1 SERVICES AND PRODUCTS

There are a number of services and products that have been developed during the project's duration. Some of the services were developed to be used in multiple products, while others are tailored for the specific tasks that a product requires to function.

Since all of the services and products are described in detail in deliverable D2.5 – Final Set of X5GON Products and Services, we present a list of services and products to make this report complete.

Services and Products	Description
X5GON Platform	The product in charge of indexing, processing, and storing the OER metadata, which can be accessed through a public API.
X5GON Search Engine	The search engine for OERs.
X5GON Discovery	The product that allows the user to search through and filter OERs.
X5GON Connect Service	The library used to connect OER repositories to the X5GON Platform.
X5GON Recommender System	The recommender system providing OER recommendations to the user.
Learning Analytics Machine	The product providing different metrics for measuring the learning difficulty, computing coherent order of the OERs, and other learning analytics tasks.
X5-Moodle	The Moodle Plugin that integrates X5GON recommendations and playlists into the Moodle learning environment.
X5GON Blind	The learning environment tailored for the visually impaired students.
X5Learn	The learning environment that exploits the full capacity of the X5GON services.

Table 1. The list of services or products.

2.2 HARDWARE ARCHITECTURE

All of the services developed within the project are running on Posita [1], the Posta Slovenije Cloud Infrastructure. There are two servers on which the services are running:

- **The production server.** All of the services and products developed within the X5GON project are running on this server. The server has the following specifications: **CPU 12 Cores, 64 GB of RAM, 2 TB of disk space.**

- **The database server.** This server is dedicated for storing and redirecting the data from one service to another. It contains the PostgreSQL database [2] (for storing OER metadata, user activities, machine learning models, and other service specific data), Elasticsearch [3] (the OER search engine index), the OER processing pipeline developed within the project, and Apache Kafka [4] (used for streaming messages between the services, the processing pipeline, the database, and Elasticsearch). The server has the following specifications: **CPU 8 Cores, 32 GB of RAM, 500 GB of disk space.**

The production server has more RAM and disk space because of the technical requirements of the services – these employ machine learning models that require space both on the RAM (during runtime) and on the disk.

Both servers are on the same internal network allowing the services on the production server to communicate with the services on the database server.

Improvements. The servers are sufficient enough for supporting the current services and products as well as for storing the data. In the future, we would need to monitor the hardware resource consumption to ensure the services are running as they should.

2.3 SECURITY

The servers are behind a firewall and one can connect to it only via the SSH cryptographic network protocol. Currently, only a few developers within the X5GON consortium have access to the servers – those are also the people that are developing and maintaining the services.

In line with the General Data Protection Regulation (GPDR) we store anonymized user activity data – each user is presented as a random string which is generated without any of the user's personal data. The data is secured on the database server, and is not exposed to the public.

The OER data cannot be accessed directly but rather through public API endpoints that first validate the request before executing it and returning the data. The API endpoints for retrieving the data are available on the X5GON Platform website[1].

Improvements. We do not perform data backups which would be required in the future. In addition, data versioning would improve both the development process as well as data security – with it one could update the database and revert to its previous version if security vulnerabilities or bugs are introduced.

2.4 INTEGRATION

The services and products are integrated within a semi-microservice architecture: Although each service is developed independently and has its own architecture design, they are connected with each other due to the interdependency of the service functionalities, e.g. The Moodle Plugin requires the playlist functionality of X5Learn, while X5Learn requires the access to the search and recommender engine.

Because the services provide their own API endpoints to access their functionalities, integrating a new product or service into the X5GON system requires minimal effort.

[1] https://platform.x5gon.org/products/feed/

Improvements. The X5GON system is small enough to have an overview how the services and products are integrated. In the future, as the number of services increase, a documentation of how the services are integrated and what functionalities the services provide would be required for both maintenance of the services and having an overview of the functionalities that is supported by the X5GON services.

2.5 MAINTAINABILITY

The service and product code bases are available on the X5GON Github organization [5]. To ensure maintainability of the services we use the following mechanisms:

API Versioning. Versioning is a crucial part of the service API design. It enables us to improve our API without breaking the applications that use the services' functionalities.

Upgrading Libraries and Applications. When developing we upgrade all existing libraries used by the service to their newest version. In addition, applications installed on the operating system are periodically updated.

Continuous Integration. For some services we enabled continuous integration, checking if the service is correctly installed and built.

Continuous Deployment. For the X5GON Platform and Discovery user interfaces, we configured to be automatically deployed on the production server when a new version of the interface has been published on Github.

Improvements. While continuous integration and deployment are setup just for a subset of services, it should be enabled for all services in the future. In addition, unit testing should be added to ensure that new changes do not break the current service functionalities.

2.6 MONITORING

The X5GON service processes are under basic monitoring showing: 1) the status of the process, 2) the time of the processes creation, 3) its uptime, 4) date of creation, 5) number of times the process restarted, and 6) the number of unstable restarts. The number of OERs and number of user interactions per OER provider is also monitored. These statistics can be seen on the X5GON Admin pages which is, as a security measure, restricted only to a few people within the project consortium.

All of the services also perform logging of requests which are stored either in log files (logs are stored based on the service configuration) or in the database (user activities on OER sites, recommender system interactions, etc.).

In addition, the X5GON Platform is monitored using Pingdom [6]. This service periodically performs requests to the service to check if it is responsive or not, measuring its uptime. In the scenario when the platform is not responsive it notifies the registered admin about its inactivity.

The data processing is monitored internally on the database server but can easily get its status through the following API endpoint:

https://platform.x5gon.org/api/v2/upload/status?material_url={material-url}

Where the **{material-url}** is the URL of the material. Through this endpoint any OER provider can check in which processing and enriching stage their OER is.

Improvements. In the future, log analysis would provide insight into how to improve the system. With log analysis we would get insight into which services are more used than others, which service functionalities are working as expected and which don't, as well as how to improve the overall architecture of the services.

2.7 ALERTING

Apart from using Pingdom to monitor the X5GON Platform and alert the registered admin about its inactivity, there have been no alerting mechanisms implemented into the services and products.

Improvements. In the future, triggering alerts when specific processes fail or take longer to execute would be beneficial for the development of the services. Setting up monitoring for all services would be required. In addition, weekly or monthly reports of the database statistics (number of new OERs, number of processed OERs, number of updated OERs, number of unsuccessfully processed OERs, number of user interactions, number of new OER providers, number of recommender interactions, etc.) would provide immediate insight into how the database is growing.

2.8 REUSABILITY

While the products and services were developed to perform a specific task, some of the services are designed to be used in other scenarios and on different types of data data. The service codes are available on the X5GON Github organization and are open source – allowing anyone to copy the code and modify them for their specific task.

As an example of such reusability, let's say we are an organization that contain private documents containing rules and regulations for different scenarios. Since there are a lot of scenarios and documents we would like to find the most relevant ones for the current task at hand as fast as possible.

In this example, we would need to 1) specify a database schema for storing the documents, 2) setup the search engine for our documents, and 3) configure the processing pipeline to retrieve the title, content and possible translations of our documents. With the developed X5GON search engine and processing pipeline we are able to setup such a system quite easily.

Improvements. In the future, we would need to maintain the level of reusability of the services, possibly opening other opportunities for developing systems within the open education domain and beyond.

2.9 DOCUMENTATION

Every service is created using well documented code. The code documentation allows multiple developers work on the same services. In addition, most of the API endpoints have a publicly available documentation.

Improvements. While some of the services have their documentations automatically generated, others require manual implementation. Automatic documentation generation would be required for high maintainability of the services, easy integration, and smooth development process.

3. DATA STATISTICS

The X5GON platform currently contains **118.307 OERs** from **14** different repositories found in Slovenia, Germany, France, USA, Canada, and several countries in Africa. Five of the repositories also have the X5GON Connect Service integrated providing the X5GON Platform user activity information. Table 2 shows the distribution of the materials and user visits per OER provider. The **34.027.487 user visits** have been created by **6.918.806 unique users.**

OER Providers	Material Count	User Visits
Videolectures.NET	27.424	921.790
Universtat Politécnica de Valéncia	4.177	990.777
Nantes University	34	2.512
University Osnabrück	507	282
MIT OpenCourseWare	46.568	0
University of Bologna Digital Library	10.439	0
eUčbeniki	12.539	32.106.695
OpenStax CNX	7.925	0
OpenLearnWare	698	0
TIB AV-Portal	53	0
Engage NY	4.619	0
The Siemens Stiftung Media Portal	2.130	0
OER Africa	317	0
eCampus Ontario Open Library	276	0

Table 2. Material and user visits count per OER provider.

The OERs are found in different formats, such as pdf, mp4, mp3, etc. We cluster the resources into three groups: text, video, and audio. The X5GON database does not contain any images, but can be found via the X5GON search engine through the use of the Creative Commons Search [7]. Figure 1 shows the distribution of OERs per group type. The majority of resources are found in text format while others are either videos or audio files.

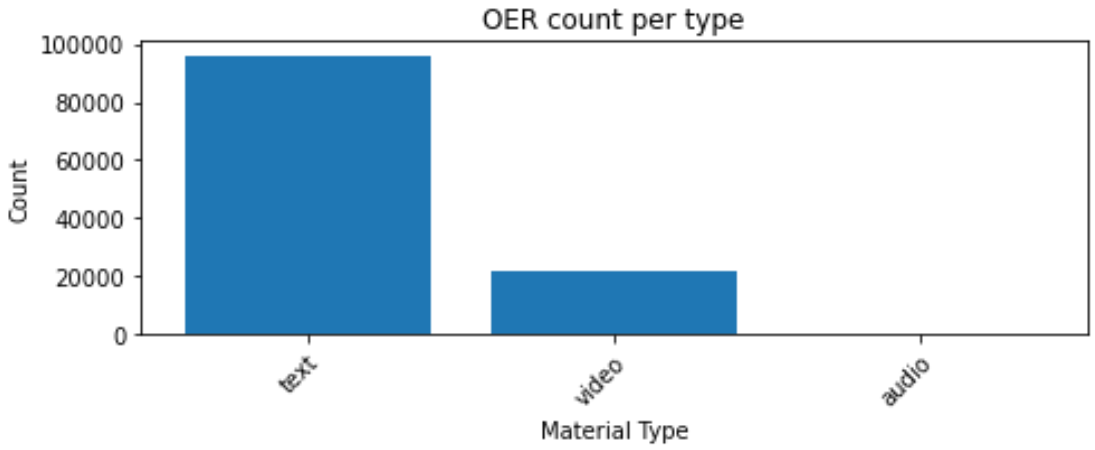

Figure 1. The number of OER per its type (text, audio, and video). The majority of OERs is in the text format.

The OERs have been processed, enriched and translated – resulting in **479.396 OER transcriptions and translations** in Slovene, English, Spanish, German, French, Catalan and Portuguese, as well as a few other languages. Figure 2 shows the distribution of the transcriptions and translations per language.

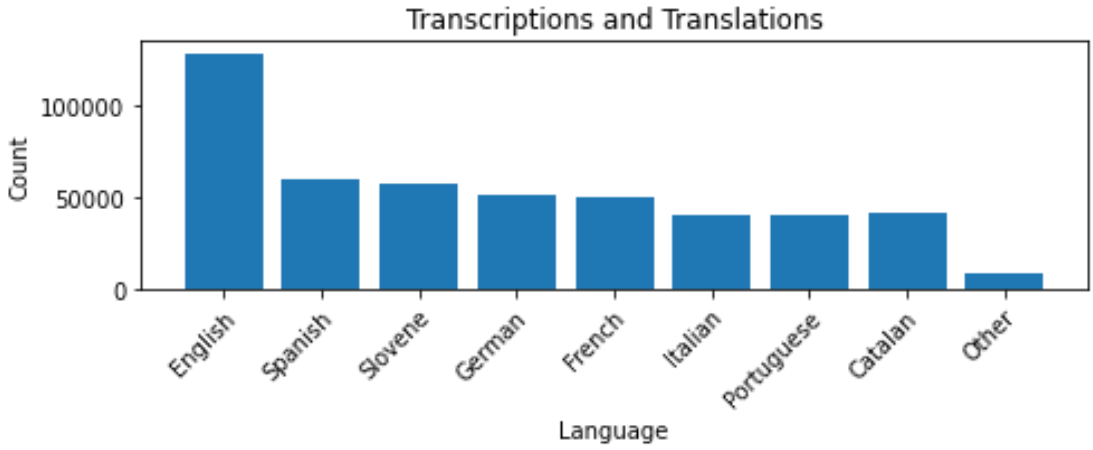

Figure 2. The number of transcriptions and translations per language within the X5GON database.

The database also contains the user selections performed on the X5GON Recommendation Plugin. The users have performed a total of **2.202.105 recommendation selections**. These selections contain the following information:

- The ID of the user that performed the selection.
- The URL of the website on which the user performed the selection.
- The URL of the selected OER or website that contains the OER.
- The whole list of the recommended OERs.
- The position of the selected OER.
- The date when the selection was performed.

While this data was used to evaluate the recommendations and were also reported in the work package 4 and 5 deliverables, we add the statistics here to have a complete report on the database statistics.

4. OPEN PROBLEMS IN OPEN EDUCATION

During the project we have identified a number of problems that exist in the open education domain. These problems were also confirmed when talking to experts, policy makers and users in the open educational domain. In this section, we describe two of the major open problems that were identified and should be considered by the stakeholders – not only to increase the usability of the X5GON services but also to improve the visibility of the OERs around the globe.

Automatic OER identification and indexing. The first open problem is "how to automatically identify an OER in the wild and how to retrieve its respective metadata?" While we have tried to tackle this problem directly, we have quickly found that due to the various formats of OERs, their embeddings and positioning in the website, as well as the differences of the provided OER metadata, the task cannot be performed automatically. Because of this, we have developed several web crawlers specific for the OER repositories that are currently indexed in our database. While this is a short-term solution, it will not scale with the number of existing OER repositories. In order to identify the millions of OERs around the globe, one would need to develop a web crawler that would be able to automatically detect the OER metadata (where the license of the resource would have the top priority).

OER unavailability decision. The second problem is "What to do if an OER is not available?" This problem has meaning in the scenario where we identified the OERs location, was processed and stored in the database. What could now happen when our database contains OERs that are not available anymore – but this unavailability can be only temporary due to repository maintenance, power outage, or some other reasons. Because of this one has to be careful in deciding what to do when an OER is unavailable. While we have not tackled this problem, we should have it in mind in the future.

While we do not present solutions to the above problems, we find that these problems are important to think about in order to make OERs even more accessible as it already is.

5. CONCLUSION

In this final evaluation report, we present the current state of the X5GON services and products. We evaluate the services and products from multiple perspectives:

- **Hardware architecture.** The hardware is sufficient for the services to run and have enough resources available for adding additional services and products.
- **Security.** The services can secure – only a few developers have access to them via the SSH protocol. Also, the data is anonymized and following the GDPR regulations. Performing data backups and versioning would increase the data security.
- **Integration.** The services are easy to integrate as they all communicate via the API endpoints. A schema of service dependence would be required as the number of services increase.
- **Maintainability.** To ensure maintainability of the services we employed the following mechanisms: API versioning, periodic library and application upgrading, continuous integration and deployment. Unit testing would improve the maintainability of the services.
- **Monitoring.** Basic monitoring of the service processes and logging is used. Log analysis would provide us insight in both the usability of the services and how to improve the services.
- **Alerting.** No alerting is currently implemented in the system. This would be required for a steady performance of our services.
- **Reusability.** The services as general enough to be reused in a different setting and domain. This level of reusability should be maintained throughout the service development.
- **Documentation.** The services are well documented. While some of the services already have this enabled, automatic documentation generation is desirable for a smooth development, maintenance, and integration process.

We found that the X5GON services and products have a solid technical foundation.

We provide a statistical overview of the data that is available in the X5GON database. Finally, we also present two problems that were identified in open education: 1) automatic OER identification and indexing, and 2) OER unavailability decision. Both problems are not trivial to solve and would require additional effort to solve them.

REFERENCES

[1] “Posita,” Pošta Slovenije d.o.o., [Online]. Available: https://www.posita.si/. [Accessed 14 12 2020].

[2] “PostgreSQL: The world's most advanced open source database,” The PostgreSQL Global Development Group, [Online]. Available: https://www.postgresql.org/. [Accessed 14 12 2020].

[3] “Elasticsearch Service: Managed Elasticsearch on AWS, Google Cloud, and Microsoft Azure,” Elasticsearch B.V., [Online]. Available: https://www.elastic.co/elasticsearch/service. [Accessed 14 12 2020].

[4] “Apache Kafka,” Apache Software Foundation, [Online]. Available: https://kafka.apache.org/. [Accessed 14 12 2020].

[5] “X5GON Github,” Github, Inc., [Online]. Available: https://github.com/X5GON. [Accessed 15 12 2020].

[6] “Website Performance and Availability Monitoring | Pingdom,” SolarWinds Worldwide, LLC., [Online]. Available: https://www.pingdom.com/. [Accessed 15 12 2020].

[7] “CC Search,” Creative Commons, [Online]. Available: https://search.creativecommons.org/. [Accessed 15 12 2020].