# 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:** D4.1 – Initial Prototype of User Modelling Architecture
**Author(s):** Erik Novak
**Contributing partners:** JSI
**Date:**
**Approved by:**
**Type:** P
**Status:** Draft/Final
**Contact:**

| **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 |
|---|---|---|
| April 3rd | Stefan Kreitmayer | Add more information about the user modelling architecture scalability. |
| | | |
| | | |
| | | |

## TABLE OF CONTENTS

## LIST OF FIGURES



| Acronyms | Definitions |
|---|---|
| OER | Open Educational Resources |
| API | Application Programming Interface |
| JSON | JavaScript Object Notation |
| URL | Uniform Resource Locator |

## ABSTRACT

Within the X5GON project we will develop a recommender system to recommend material from multiple OER repositories allowing the users to gain relevant knowledge from multiple resources. The system will incorporate cross-site and cross-linguality methods and services which will allow recommending materials provided in different languages. In this report we present the initial steps to achieving this goal – the development of user modelling architecture.

This architecture is designed to allow users manually input their interests as well as dynamically extract their interests based on the OER materials he or she viewed. User models are designed to be used in interest analysis and in the next steps of recommendation engine development which is development of cross-site and cross-lingual material recommendation methods. We also present the initial recommendation methods we developed to achieve the goals presented within the project.

# 1. INTRODUCTION

User modelling is a subdivision of human-computer interaction which describes the process of building up and modifying a conceptual understanding of the user [1]. This can be done in multiple ways: by asking the user to give information about themselves and their interests, using machine learning methods to extract the user's interests from the content he or she is looking at or a combination of both approaches. This data can be used for recommending content the user is interested in. Additionally, it can be used for learning analytics – giving insight in what are the user's interests and how they evolve through time.

In this document we present the initial user modelling architecture prototype which includes the modelling architecture, the data that is fed to it and how is it going to be used in the final cross-lingual recommendation engine. The architecture considers two approaches:

- Users login to the X5GON dashboard developed within WP2 and give information about their interests and learning patterns which is used for creating the user's model
- Create user model by extracting concepts from material the user has viewed on OER repositories and use them for determining user interests

The architecture also has a mechanism for merging both approaches which enables real-time updating of user models. This mechanism is based on the X5GON user activity tracker library described in D2.1 – Requirements & Architecture report. The library creates a cookie which is persistent across different domains enabling user identification both on OER repositories as well as the X5GON dashboard.

The document is structured as follows. Section 2 describes the data used in creating user's model. The data consists of both OER material information and user activity data. Next, in section 3 we present the user modelling architecture and describe each step in the process. How is the architecture used within the recommendation methods is shown in section 4. Finally, we present future steps of recommendation engine development in section 5 and conclude the report in section 6.

# 2. OER MATERIAL AND USER ACTIVITY DATA

Before we start developing the recommendation engine we need to understand the data that is and will be available to us. We identified there are two types of data that will be useful in the development process: OER material data and user activity data. In this section we give a brief description both types of data and how it will be used in user modelling. The data acquisition process of both types is described in D2.1 – Requirements & Architecture Report.

## 2.1 OER MATERIAL DATA

The OER material data contain information of the material available on OER repositories. It is presented in a JSON format containing the following attributes:

- **title:** the title of the material
- **provider:** the OER provider of the material
- **materialURL:** the URL to the material
- **author:** who created the material *(optional)*
- **created:** when was the material created or published *(optional)*
- **type:** what is material type, i.e. video, audio, presentation, image, text, etc.
- **language:** the language in which it is written, i.e. en, sl, es, etc.
- **metadata:** additional metadata acquired from the material, i.e. Wikipedia concepts, extracted features, etc.

The attributes labelled as optional are not necessarily present. Within the metadata attribute each material contain Wikipedia concepts that are associated with the material's content. These concepts represent a semantic space, i.e. vocabulary, in which we can compare materials. Additionally, Wikipedia concepts can be viewed as topics or interests the material covers. This information will be used to describe user's interests and recommend content in the future. Furthermore, we will use all of the material data to do material analysis which will give knowledge about the acquired materials such as the language and topic coverage, provider traffic and learning pathway creation.

## 2.2 USER ACTIVITY DATA

User activity data tell us which user viewed which material and when. It also includes the technology, i.e. operating system and browser, the user used to access the material. The data is presented in a JSON format and contains the following attributes:

- **userId:** the identification of the user accessing the OER repository
- **materialURL:** the link to the material the user visited
- **referrerURL:** from which website did the user accessed the material
- **accessDate:** when was the material accessed
- **userAgents:** what technology did the user used to get to the material

The materialURL attribute will be used to connect the viewed OER material with the user model where we'll map the Wikipedia concepts found in the material to user interests. It will also be used to track how the user moved from one material to another – showing the learning pathway he or she took to acquire the desired knowledge.

# 3. USER MODELLING ARCHITECTURE

For the purposes of the X5GON project user modelling is viewed as modelling user's interests regarding to the OER material. We describe the user interests with a set of Wikipedia concepts that were extracted from the OER materials viewed by the user. The interests will be also manually given by the user on the X5GON dashboard, a website developed within the project as part of WP2.

In this section we present the user modelling architecture and its workflow. A visual description of the architecture is shown in Figure 1.

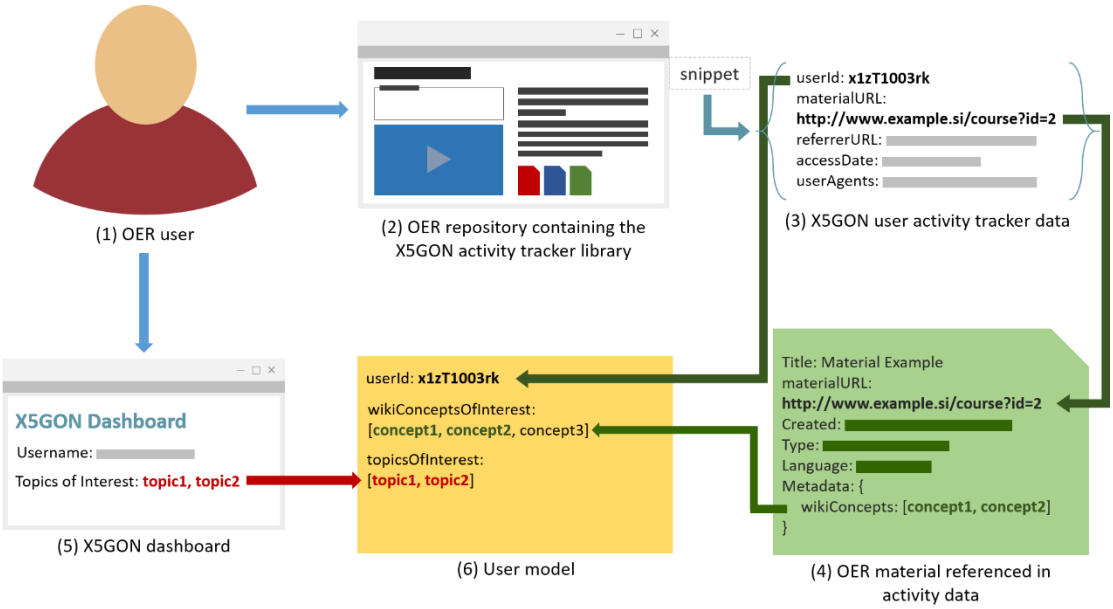


***Figure 1:*** *User Modelling Architecture*

The architecture is designed to handle three approaches:

- **OER Provider Approach:** User models are dynamically updated using the information of OER material viewed by the user
- **X5GON Dashboard Approach:** User manually gives his or her topics of interests in the X5GON dashboard which is used to modify the user model
- **Combined Approach:** A combination of previous approaches

In the following sections we describe how the architecture handles each of the mentioned approaches. The description is accompanied by a figure highlighting the elements used in the described approach. Elements not used in the approach are grey-scaled. Also, elements are presented with labels, i.e. numbers, which are used in the description for better understanding. The source code of the user modelling architecture is found on Github (https://github.com/JozefStefanInstitute/x5gon).

## 3.1 OER PROVIDER APPROACH

The first approach handling we describe is when the user views material located on an OER repository that included the X5GON activity tracker library. How the architecture handles this scenario is presented in Figure 2.

**Pipeline description.** When the user (1) visits a material located on an OER repository (2) the included activity tracker library sends user activity data (3) to the X5GON platform where it is stored in the database. The sent data contains attributes described

in section 2.2. Next, *materialURL* and *userId* attributes in the user activity data are used to find the corresponding OER material (4) and user model (6), respectively. Finally, we map the Wikipedia concepts located under the OER material's *metadata* attribute to the user model. As mentioned in section 2.1 these Wikipedia concepts are viewed as topics and interests the material covers. By viewing it we determine the user is interested in topics the material covers. Additionally, the number of times a Wikipedia concept was mapped to the user model indicates the rate of interest described with these concepts.

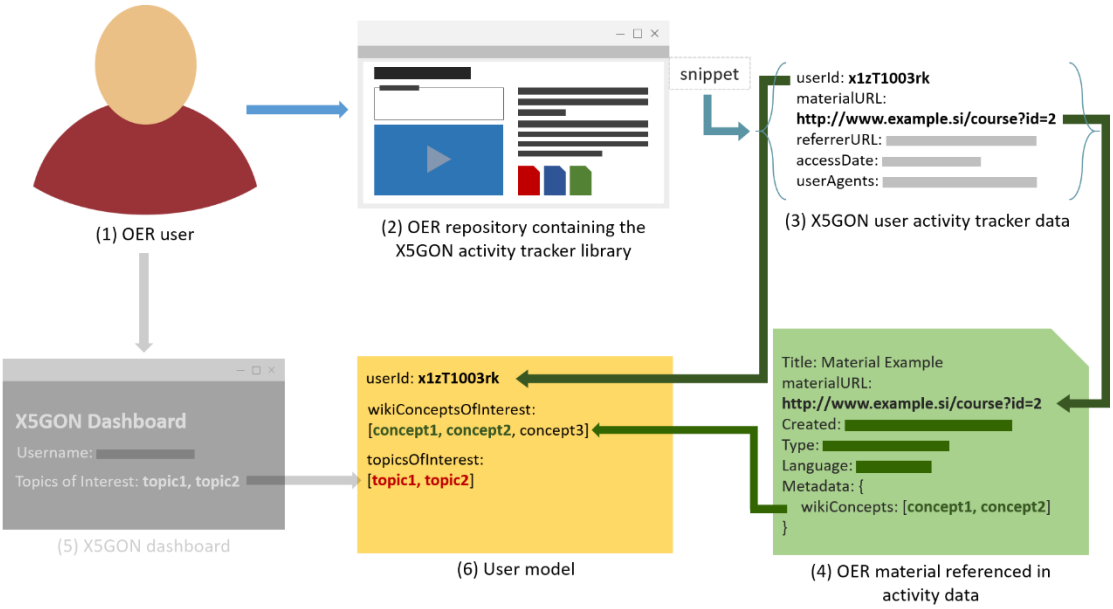


***Figure 2:*** *User Modelling Architecture – OER Provider Approach*

If the OER material associated with the *materialURL* attribute is not found the material is not yet acquired and stored in the database. In this case, we first crawl and enrich the OER material as described in D2.1 – Requirements & Architecture Report. Once we acquired the material we use the approach described above.

## 3.2 X5GON Dashboard Approach

In this section we present the second approach the architecture can handle – when the user manually inputs the topics that he or she is interested. For the user to give this information we require a website which allows the user to express their interests. To this end, we will develop the X5GON dashboard – a website where the user has the option of creating his or her own learning pathways, save OER materials and set their own goals he or she wishes to achieve. The dashboard will be developed within WP2.

Once the dashboard will be developed the user will have the option to input their interests. Upon saving, he or she sends this data to the X5GON platform where the data is stored in his or her user model. This transition of data is shown in *Figure 3*. What follows is a detail description of how the architecture handles this approach.

**Pipeline description.** When the user (1) will access the X5GON dashboard he or she will have the option to input his or her interests. This will be done by inserting a free-form text into a specific field or by selecting multiple Wikipedia concepts that have been extracted from the acquired OER material. Upon submitting the data is sent to the X5GON platform where it is stored within the user model. The free-form text will then

be sent through the Wikification process described in D2.1 – Architecture & Requirement Report. The extracted concepts are then added to the Wikipedia concepts of interest within the user model. Additionally, any user input is stored in the user model to preserve the state in which the user submitted his interests. By doing this we will be able to analyse the change in the user's interest which will be used in the next steps of Recommendation Engine development.

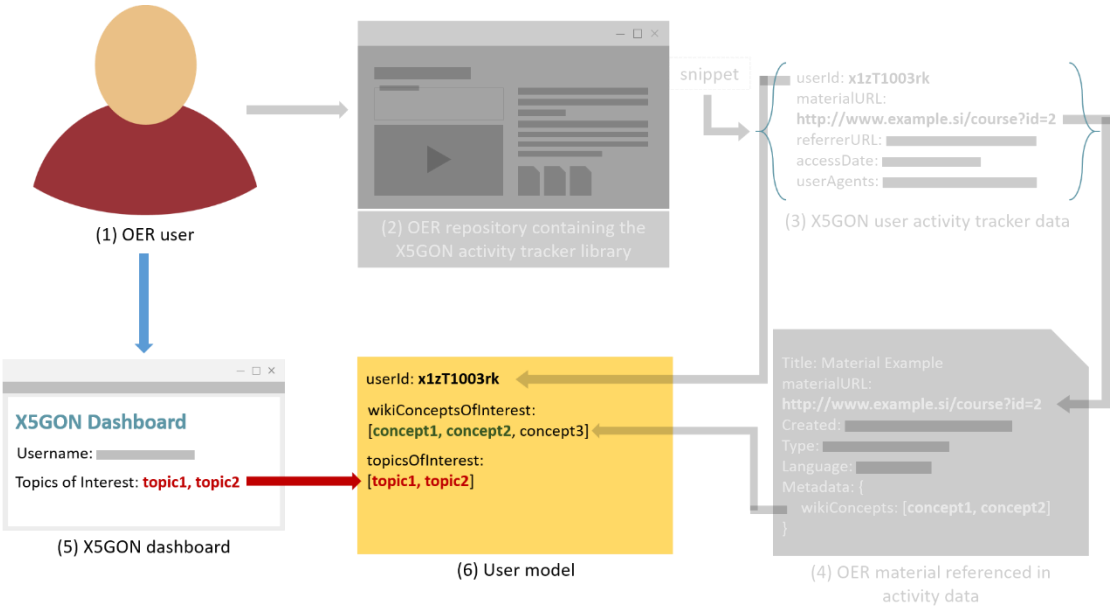


***Figure 3:*** *User Modelling Architecture – X5GON Dashboard Approach*

## 3.3 Combined Approach

When the user both inputs his or her interests into the X5GON dashboard and visits materials on OER repositories that included the X5GON user activity tracker library the architecture needs to consider how to handle both data sources.

We have decided to prioritize user input data given through the X5GON dashboard since this data is given directly by the user. The Wikipedia concepts extracted from his or her input will receive a much larger weight in the user model which will influence the recommendations for that user.

The initial version of the user model will focus on the user's interest in specific topics. This metric was chosen as a bare minimum to get a working prototype out of the ground. By implementing mechanisms for dealing with this metric (via inference and users' self-report), we create a baseline for the system regarding how to deal with similar metrics that are likely to become relevant, such as interest in content type, prior knowledge etc. The decision regarding which metrics to include in the final version depends on evidence about learners' needs in the wild. This evidence will be provided by empirical studies that are part of WP6.

# 4. USER MODELLING AND RECOMMENDATION ENGINE

User modelling will give us insight in what users are interested in and what they would like to learn. The data will also be used to cluster users based on their interests, finding what they are viewing and how their interests change and develop through time. This analysis will help us in the recommendation engine development process.

This section describes how the user modelling architecture fits in the recommendation engine and what types of recommendations will be supported.

## 4.1 USER MODELLING ANALYSIS

Before we start to develop the recommendation engine we will analyse user models. By doing so we will get insight in which users have similar interests, what are the most popular interests and topics and how interests change as time passes. We plan to do the following analysis:

**User clustering.** We will group users based on their interests by using the kmeans clustering algorithm [2]. This algorithm is used to cluster object using a similarity measure, in our case the cosine similarity measure [3]. Within the groups we will be able to extract what are the interests that join the users together and what makes them different from users in other clusters. Centroids of the clusters can be seen as cluster representatives and the average user in the cluster.

**Interest timeline.** One aspect which is interesting is how user's interests change over time. This can be influenced by emergence of a new approach for solving a family of problems, time of the year, i.e. start of exam period in schools, or even lack of interest for a previous passion. User's interest may also shift within the same topic from beginner level to advanced level materials as the user is learning. Additionally, analysing how clusters of users change over time can give us information about which topics or interests are becoming stronger and which are becoming weaker. We can also see how users move from one cluster to another due to change of interest giving us a graph of user transitions between clusters.

The bulk of the analysis will be done using the QMiner platform developed by JSI. The platform includes the kmeans clustering, k-nearest neighbours, active learning and stream analysis algorithms. Additionally, the platform can be used in JavaScript allowing us to easily send the results of the analysis to the X5GON dashboard and to visualize them in various ways – increasing the understanding of the results.

## 4.2 RECOMMENDATION ENGINE

Recommendation engine is a subclass of information filtering system that seeks to predict the preference a user would give to an item [4]. In X5GON project the item is considered to be OER material. There are different ways of creating recommendations. We are considering to use the following:

**Content based recommendation.** When the user gives an OER material and expects to get other OER material that are similar in content, we use content based recommendation. For this type we do not necessarily need user information since the recommendation will be using OER material attributes, more precisely Wikipedia concepts found under the *metadata* attribute. Using content based recommendation is also beneficial in the beginning as it does not suffer from the so called cold start problem, where user data is not yet available.

The recommendations will be done by using the k-nearest neighbour algorithm [5] which for a given OER material searches the k most similar materials based on the

Wikipedia concepts, where k is any natural number. A prototype of this recommendation method has been developed - the source code can be found at [6]. The prototype considers k to be equal 100, i.e. we recommend 100 most similar materials based on the input.

**Interest based recommendation.** Another approach of recommending materials is to use user's interests. As described in section 3 user's interests are presented as Wikipedia concepts which were extracted from the OER material data. We will use these Wikipedia concepts to present the user as a material and use the content based recommendation approach. This will return a list of material most similar to the user's current interests.

**Recommendation based on collaborative filtering.** Collaborative filtering is a method of making automatic predictions about the interests of a user by collecting preferences or taste information from many users [7]. In the context of X5GON project this would be done in the following steps:

1. User *A* views an OER material
2. We search for other users that have also viewed that material – the set of found users is called *viewers*
3. We filter other OER material that have been viewed by at least one user from the *viewers* set and count how many views each material has – the set of (*material, count*) pairs is called *viewedMaterials*
4. From *viewedMaterials* we remove materials that were already viewed by user *A*
5. We sort the v*iewedMaterials* set based on the *count* value and return the list to user *A*

Here we will use user modelling to extract the materials that were viewed prior the start of the recommendation process. The approach will be implemented using the QMiner platform which already has one version of the collaborative filtering method included. We will modify the method to better fit our problem.

These recommendation methods will also need to support cross-site and cross-linguality which is partially solved with the X5GON platform architecture design presented in D2.1 – Architecture & Requirements Report as well as the user modelling architecture presented in Section 3. The platform's pre-processing pipeline contains the Wikification step which extracts Wikipedia concepts from textual information about the OER material. This step supports multi- and cross-linguality giving Wikipedia concepts in the same language regardless the language of the input text. Once the Wikipedia concepts are extracted and OER material stored in a database, the X5GON user activity tracker library and snippet give us information what OER material the user is viewing. How this information is used is described in section 3.1 – resulting in cross-site recommendations.

## 5. FUTURE WORK

The user modelling architecture developed currently supports only OER Provider approach. Other approaches will be implemented when the X5GON dashboard will be developed.

We have already developed the content based recommendation method described in section 4.2. We set up a service on the cloud infrastructure provided by Pošta Slovenije and tested the method. This was done using the Videolectures.NET data source. Other methods require user activity data which is provided by the X5GON user activity tracker library. The library needs to be included in OER repositories that are available to the consortium. Once this is done we will start receiving user activity data and test the user modelling architecture and start developing other recommendation methods presented in section 4.2.

The next steps within the work package are:

- Add support for X5GON Dashboard approach to the user modelling architecture.
- User modelling analysis as described in section 4.1.
- Development of recommendation methods presented in section 4.2.

## 6. CONCLUSION

In this report we present the user modelling architecture and how it fits in the final recommendation engine.

First we make a quick overview of what is the data we use in the architecture. In section 2 we describe the OER material data and user activity data that are used in creating user models and recommender engine. The OER material data is the product of the material pre-processing pipeline within the X5GON platform described in D2.1 Architecture and requirements report and the user activity data is provided by the X5GON user activity tracker library and snippet.

In section 3 we present the user modelling architecture. The architecture is design to handle three approaches of user model updating. The first is described in section 3.1 where we dynamically update the user model based on what OER materials the user viewed. The second is presented in section 3.2 where the user send the topics he or she is interested in to the X5GON platform where it is used to update the user model. Third approach is combining both approaches where we prioritize the interests given by the user directly than those extracted using the user activity data.

Next, in section 4 we describe what type of analysis will be done in the future using user models. While user clustering will show us which users have similar interests, analysing user interests through time will give us insight in how user interests change and what material are they viewing.

Finally, we present future work in section 5 which includes live testing, integrating user modelling architecture and developing recommendation methods described in section 4.2.

## REFERENCES


[1] “User modeling - Wikipedia,” Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/User_modeling. [Accessed 19 03 2018].

[2] J. A. Hartigan and M. A. Wong, “Algorithm AS 136: A k-means clustering algorithm,” *Journal of the Royal Statistical Society. Series C (Applied Statistics),* vol. 28, no. 1, pp. 100-108, 1979.

[3] “Cosine similarity - Wikipedia,” Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Cosine_similarity. [Accessed 15 03 2018].

[4] “Recommender system - Wikipedia,” Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Recommender_system. [Accessed 21 03 2018].

[5] “k-nearest neighbor algorithm - Wikipedia,” Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm. [Accessed 15 03 2018].

[6] “x5gon/src/lib/models at master - JozefStefanInstitute/x5gon,” GitHub, Inc., 2018. [Online]. Available: https://github.com/JozefStefanInstitute/x5gon/tree/master/src/lib/models. [Accessed 12 04 2018].

[7] “Collaborative filtering - Wikipedia,” Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Collaborative_filtering. [Accessed 21 03 2018].