Mobility scenarios supported by the Erasmus Without Paper Network

Janina Mincer-Daszkiewicz (jmd@mimuw.edu.pl), Faculty of Mathematics, Informatics, and Mechanics, University of Warsaw, Banacha 2, 02-097 Warszawa, Poland

1. ABSTRACT

EWP (*Erasmus Without Paper*) is the European project co-financed in years 2015-2017 by the *Erasmus+ Programme, Key Action 3 (Prospective Initiatives — Forward Looking Cooperation Projects*). Funding has been prolonged by EACEA for years 2018-2019.

At the final conference of EWP 1.0 (September 2017) the state of development of the EWP Network was presented. Now, when EWP 2.0 starts (January 2018), the architecture of the EWP Network is already fully designed (the specification in publicly available in GitHub), including highly secure communication protocols, the EWP Registry holding the binding information and the digital services to be delivered by the participating institutions. The project partners developed connectors between their local Student Information Systems and the Network. The Registry which holds the URLs of the connectors has been implemented and deployed (demo installation). The reference connector has been set up to help testing the communication within the Network and as the reference implementation. Some of the partners also offer stable demo versions of their installations to be used for testing. The EWP Network together with the Registry and the institutional connectors constitutes a stable and mature proof of concept of the idea of mobility without paper.

During the first two years of the project the partners concentrated, on the one hand, on the business aspects of the mobility, which were the basis for specification of the required digital services, and on the other hand, on highly technical aspects of the Network development. The next two years should however be spent to show the academic community that the network is there and that it really supports daily processes of International Relation Offices which are business owners of the mobility procedures. It is time to go forward and deploy the Network in the production environment. This is the main goal of EWP 2.0. A Competence Centre will be established to support this goal and in particular to help other higher education institutions to join the Network.

The aim of this paper is to show the state of development of the EWP Network, to demonstrate how the Network will support the electronic exchange of data between partners in mobility excluding use of paper, and to share the plans for the European-wide roll out of the EWP Network.

The ultimate goal is to help the authorities, students, IRO staff or — generally speaking — end users to understand what does EWP mean to higher education and mobility, in **practical terms**.

2. INTRODUCTION

EWP (*Erasmus Without Paper*) [4], [8] is the European project funded in years 2015-2017 by the *Erasmus+ Programme, Key Action 3 (Prospective Initiatives — Forward Looking Cooperation Projects)*. The application for a follow-up has been approved and EWP will continue as the European project in years 2018-2019.

At the final conference of EWP 1.0, which took place in Brussels in September 2017, the state of development of the EWP Network was presented [10]. The architecture of the EWP Network is fully designed (specification in publicly available in GitHub [7]), including highly secure communication protocols, the EWP Registry holding the binding information and the digital services to be offered by the participating institutions. The project partners developed connectors between their local Student Information Systems and the Network. The Registry which holds the URLs of the connectors

has been implemented and deployed (demo installation [6]). Testing was carried out using the reference connector and a couple of test installations set up by the development teams. Data came from institutional databases but were scrambled to protect privacy. In chapter 3, we give a short overview of the current state of the Network from the technical perspective, and demonstrate what tools and resources are offered to support developers.

However for end users — authorities of higher education institutions, decision makers, IRO staff involved in daily mobility routines and mobile students — the main concern is not a **technical perspective** but what are the **noticeable benefits** of digital, paperless mobility. Does it mean no paper at all or no exchange of paper between the mobility partners? What data are exchanged electronically? Who/what triggers data transfer? Is privacy of data respected? Who is in charge and where does responsibility lie? And finally — does it mean *less burden* on all parties involved, *better quality* of the procedures or *more fun* for ambitious ICT staff? These aspects of the EWP project constitute the main subject of this paper and will be dealt with in chapter 4.

What is the most important is the roll-out of the Network across Europe, dissemination and sustainability. What measures are taken into account to make the Network operational in production with many mobility partners connected? Be it a medium size higher education institution with a homemade Student Information System, or a client of the commercial mobility software providers, like SOP, or QS Unisolution, or a member of a consortium of HEIs using the same SIS — all should know how to start, what steps to take, and where to ask for help and advice. These are the important issues that will eventually decide about success or failure of paperless mobility in EHEA. This topic will be dealt with in chapter 5.

Summary chapter concludes the paper.

3. STATE OF DEVELOPMENT OF THE EWP NETWORK

The reference document for the developers is the **Developers Guide** [5]. It gives the overview of resources and tools supporting development:

1. Documents and specifications.
2. Libraries and tools.
3. Echo API Validator.
4. XML Schema Validator.

The most important resource offering the full description of the technical aspects of the EWP Network is GitHub repository [7]. In fact, GitHub contains not only description of the Network, its architecture, common data types, communication protocols, Application Programming Interface (API), but also XSD schemas and examples of exchanged data in XML. Last, but not least, it contains the section **EWP Mobility process explained** (https://github.com/erasmus-without-paper/ewp-specs-mobility-flowcharts/) explaining how the Student Mobility Business Process is modeled within the EWP Network, using flowcharts like the one in **Figure 1.** It helps to get a quick grasp on which APIs are used by whom and when. The attached flowchart demonstrates in particular that data can be exchanged in different ways: one institution may start data transfer by **pulling** the data from another institution (using first **index** to get the whole list and then **get** to obtain one particular piece of data, e.g. one particular bilateral agreement) or one institution may **notify** the other that the new data is ready and then the partner **get**s the data using the data identifier from the notification.

The specification covers APIs for the most important procedures of the mobility process, like signing bilateral agreement, sending list of the nominated students or exchange of transcript of records when the mobility ends (see chapter 4). Global identifiers have to be stored in both databases to enable data synchronization.

The only central element of the Network is the EWP Registry which stores URLs of the institutions connected to the Network (**Figure 2**). Under this URL the institution keeps the Manifest file (service) with the information of supported APIs (services). The Registry is an active part of the Network invoking **Discovery API** to get update on the information stored in the Manifest files and post it in the Registry. It also gives access to the page **Manifest Importer Status**, showing the list of currently

defined Manifest sources and their statuses. The other important page is the *HEI/API Coverage Matrix* showing the list of APIs supported by partner institutions, with the version numbers.
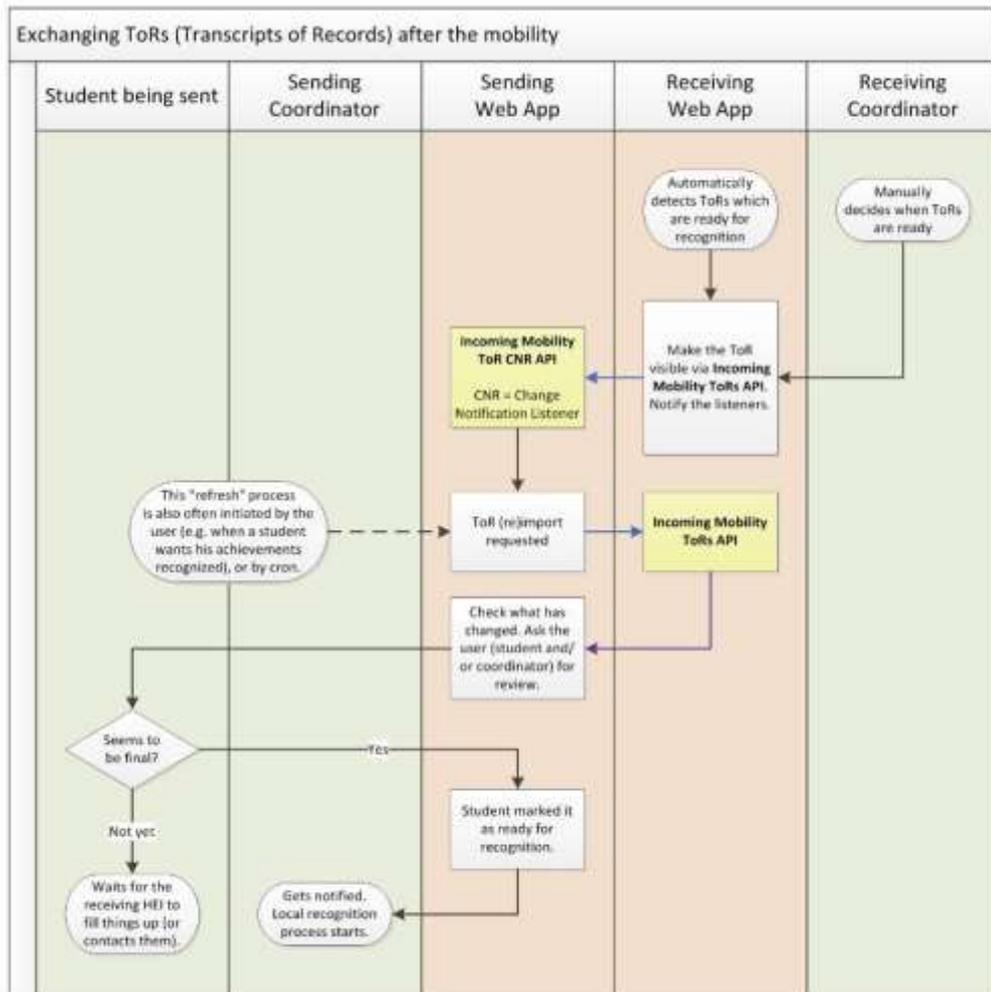

Figure 1 Flowchart showing exchange of *Transcript of Records* after the mobility and used APIs

Figure 2 The DEV Registry of the EWP Network

The functioning of the Registry and the structure of the Manifest file is also explained in GitHub. Developers should start with implementing *Discovery API*. Discovery Manifest files announce which HEIs the local system covers, which features (APIs) have been implemented, and which credentials the local clients are going to use when fetching the data from the EWP Network. The next one to implement is *Echo API* which allows beginner EWP developers to test the security of their EWP Network connections. It doesn't "do" anything, but it requires the developer to implement the core security framework (which will be needed by *all the other* APIs later on). The Registry implements the *Registry API* which is used by the clients to get the binding information from the Registry.

Having implemented these basic APIs the institution may become part of the Network. Implementation of other APIs follows according to the needs and priorities of the institution.

The *Echo API Validator* helps to determine if the implementation meets the basic EWP standards (in particular, its security requirements). *Echo API* has been designed to serve two purposes:

    a. to make developers aware of the specific security features required by the EWP Network,
    b. to allow running automated tests on all existing implementations (thus reducing the risk of security misconfiguration).

The example report produced by the Validator is shown in **Figure 3**. The Validator performs thousands of tests, only a small part of the report is presented.

Another available tool is the *XML Schema Validator*. This tool helps to write EWP XML documents. For example a developer can check the contents of the local Manifest file before uploading it onto the production site. The Validator allows validation of any XML document described in all released specifications. This tool will validate against the schema only! Even if such validation succeeds, the file may still be invalid (if, for example, the developer didn't adhere to the guidelines described in documentation elements included in the XSD files).

Figure 3 Echo API Validator

Last but not least there is a reference connector up and running at https://ewp.its.umu.se/ewp-reference-connector/. The reference connector has been set up to help in testing the communication within the Network and as the reference implementation (the source code is available in GitHub, https://github.com/erasmus-without-paper/ewp-reference-connector). Some of the partners also offer stable demo versions of their installations to be used for testing, e.g. there is a demo installation of the University of Warsaw (see chapter 4). Credentials needed to use this installation are available upon request.

4. MOBILITY SCENARIOS SUPPORTED BY THE EWP NETWORK (EXAMPLE)

The aim of this chapter is to show a few mobility scenarios as they are supported by the EWP Network in the University of Warsaw. The example scenarios are very simple, just to give a feel of what paperless mobility means in practical terms. Note however, that it is up to the local implementation what scenarios will be supported and how sophisticated they will be. What we present here is the perspective of the University of Warsaw.

Let us assume that the institution called *UW* (*University of Warsaw*) wants to start cooperation with a new partner — institution called *HEI*. We use two separate demo installations of the system *USOSadm* (SIS used in many HEIs in Poland, in particular in the University of Warsaw) to show behavior of each of the partners.

4.1. New mobility partner

The first scenario starts when *UW* decides to sign an agreement with the new partner — *HEI*. This will be the first agreement between these two institutions. UW first checks in the local system if *HEI* is connected to the EWP Network and finds out that *HEI* runs a fully functional EWP connector with many supported APIs. This will allow *UW* to exchange data with *HEI* using the EWP Network.

UW enters minimal information about *HEI* to the local system — name of the institution and **SCHAC** which is a global unique identifier of the institution in the EWP Network.


Figure 4 USOSadm in *UW* — entering name and SCHAC of *HEI* to the system

The other data can be obtained electronically straight from *HEI* using the EWP Network — data on the institution level and units of HEI. The operator pushes ***Get data from EWP*** button.

Figure 5 USOSadm in *UW* — getting institutional data of *HEI* to the system

Details about the partner institution can be copied one by one to the local system. One of the data is URL of the *HEI*'s factsheet. All the partner institutions of *UW* are listed in a student portal called *USOSweb*. On the *Partner institutions* page there is a list of partner institutions with their factsheets — uploaded locally (link *show from USOS*) or obtained from the partner by the EWP Network (link *show from partner*). The filter in the upper part of the page allows limiting the list to the institutions with institution factsheets and/or unit factsheets.

Figure 6 USOSweb in *UW* — factsheets of the partner institutions

There is also **Get data from EWP** button for getting all units of the partner. It is up to the operator which one will be copied to the local system — probably only those involved in the mobility.
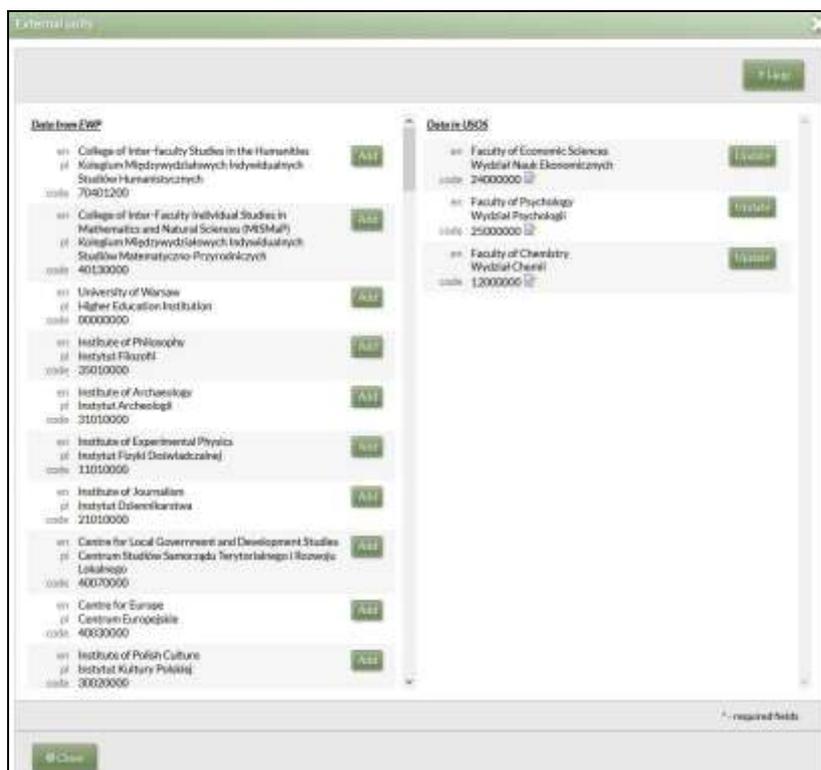
Figure 7 USOSadm in **UW** — getting organizational units of **HEI** to the system

4.2. Interinstitutional agreement

UW enters the draft version of the agreement with **HEI** to the local system typing local information only. The rest will be added by the partner and then synchronized between both systems. The agreement is signed under the ERASMUS+ programme.
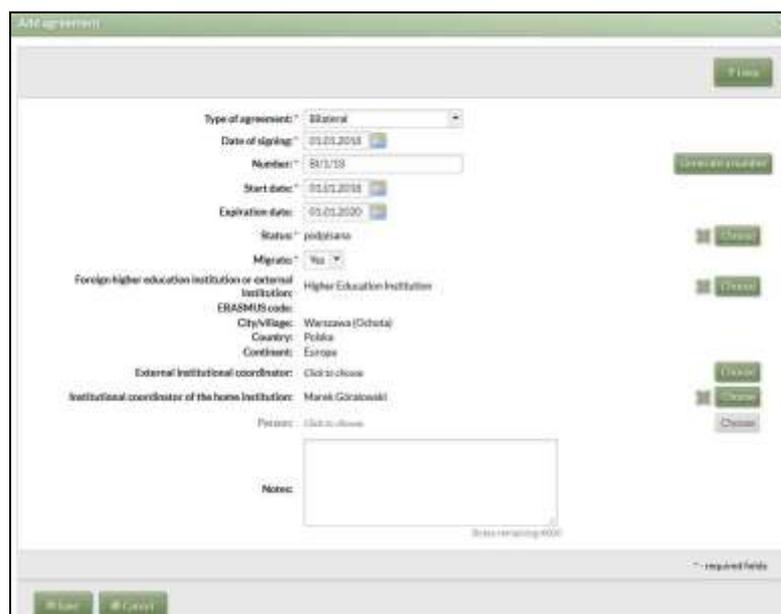
Figure 8 USOSadm in **UW** — entering draft version of the bilateral agreement with **HEI** to the system

UW informs the partner by sending EWP notification that the draft of the agreement is ready.
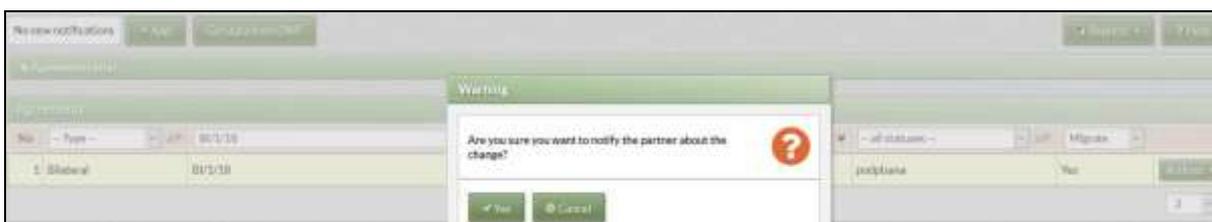
Figure 9 USOSadm in *UW* — sending notification about the agreement to *HEI*

HEI obtains notification about new agreement (or changes in the existing agreement).
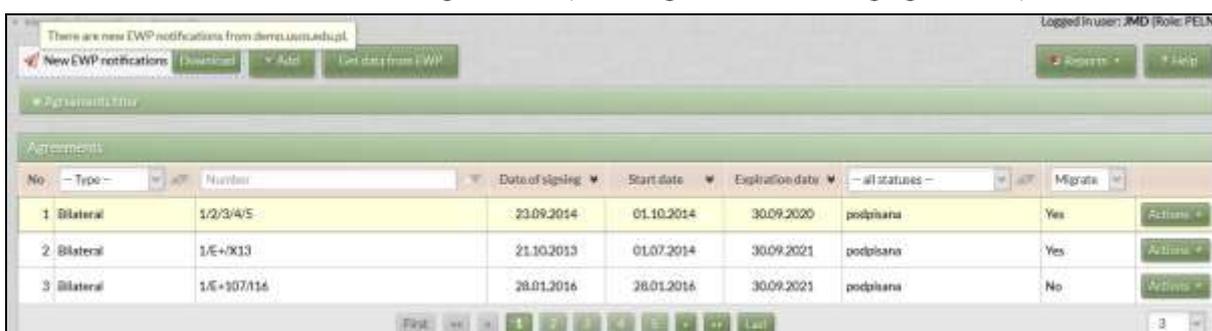
Figure 10 USOSadm in *HEI* — the operator is informed about the notification from *UW*

*HEI* gets the agreement electronically from *UW* by the EWP Network and saves the copy in the local system — can either use *Download* or *Get data from EWP*.

In a similar way, details of the cooperation conditions of the agreement can be entered by one of the partners and copied by the other. Notifications help to synchronize local versions of the bilateral agreement before it is finally approved.

4.3. Nominations

UW nominates a student to spend winter semester 2017 in *HEI* under the newly signed bilateral agreement.

Figure 11 USOSadm in *UW* — nominating student for the outgoing mobility in *HEI*

UW notifies the partner about the nomination. *HEI* gets the notification.

Figure 12 USOSadm in *HEI* — getting the data about the nominated student

HEI gets details of the nomination from the EWP Network. Not all data sent by the Network are saved in the local system. Implementing the connector the developers decide how to handle exchanged data and how to map the data items from the API to the local data model.
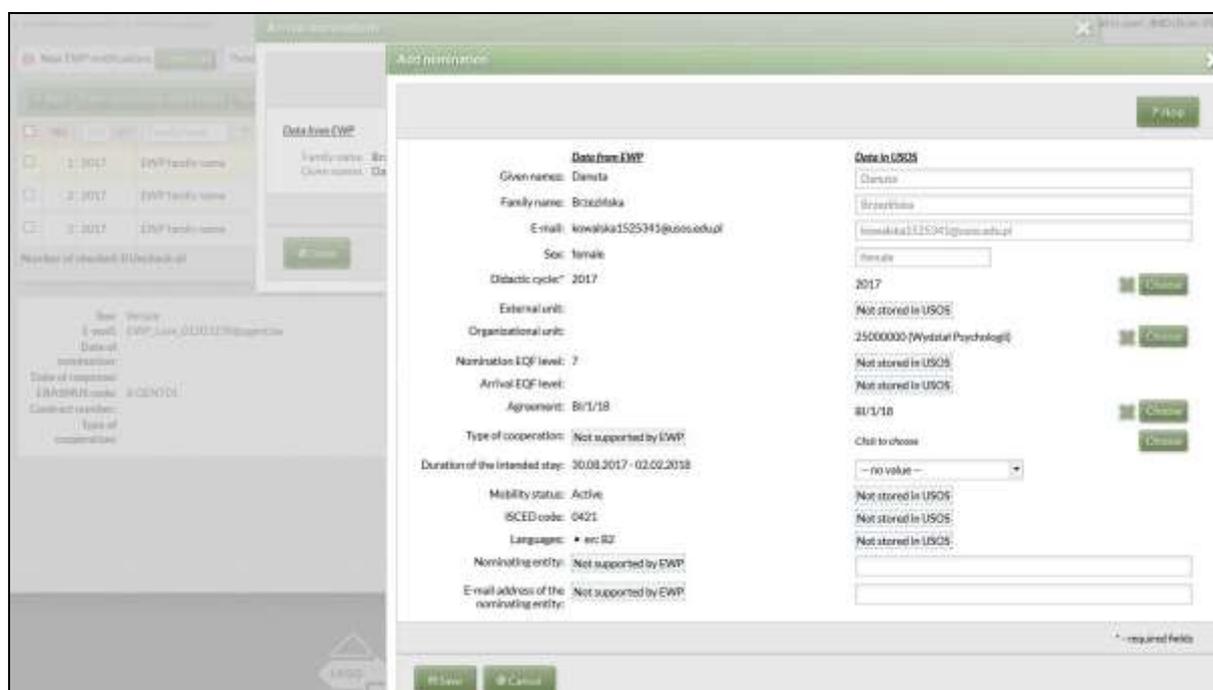
Figure 13 USOSadm in *HEI* — getting details of the incoming mobility

HEI edits the new nomination supplementing the missing data. The operator may *accept* or *reject* the nomination and notify the partner about the change in the nomination. Eventually accepted nominations become arrivals.

HEI may decide to skip the nomination phase and accept all obtained nominations as arrivals.

4.4. Learning Agreement

Institutions will continue handling Learning Agreement using their tools of choice.

4.5. Transcript of Records

When the mobility ends, *HEI* generates *Transcript of Records* for incoming students. This can be done per student or for a group of students (see checkboxes in the column on the left).

Figure 14 USOSadm in *HEI* — generating Transcript of Records for the incoming students

HEI notifies the partner about the available transcripts. *UW* obtains the notification.

Figure 15 USOSadm in *UW* — getting notification about the available transcripts

UW uses *Get data from EWP* to get Transcript of Record for the outgoing student straight to the local system in the context of the outgoing mobility. The transcript is transferred in the ELMO format (designed for EMREX, see [1], [9]) which means that it contains courses, grades, ECTS points in a structed XML from which data can be copied to the local tables and used for further processing. PDF version for pretty-printing in embedded in XML. Both XML and PDF can be digitally signed.
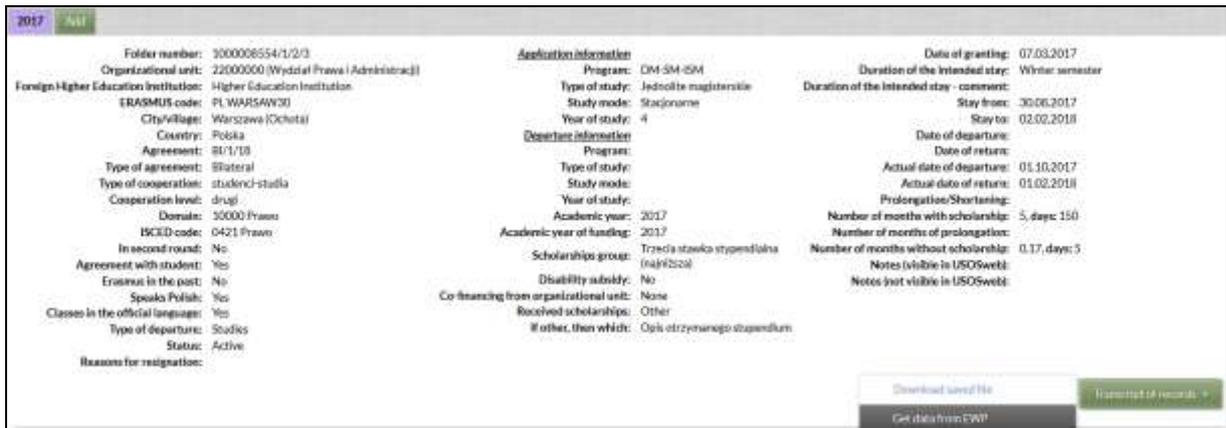
Figure 16 USOSadm in *UW* — getting Transcript of Record for the outgoing student

4.6. Final remarks

There are more APIs and more scenarios supported by them, but we had to choose a sample to give a flavor of the system. Notifications play an important role in data synchronization between partner institutions. Data objects are identified by GUIDs (global unique identifiers) which can be stored in a local system to identify objects coming from the Network and recognize them in the local system.

It is up to the local stakeholders how to automate data exchange and in particular whether it should be triggered by the operator or a system daemon. Eventually, when the local users will start trusting the Network, more and more data can be synchronized between the institutions fully automatically.

5. STEPS TO JOIN THE EWP COMMUNITY

One of the dissemination tasks undertaken by the partners in the project will be to define the entry procedure for those who want to join the EWP Network. Following is a tentative list of the steps to be taken by a Higher Education Institution to become part of the EWP Community.

1. The local representative gets in touch with experts from the EWP Competence Centre who can help in getting a full picture of the required activities. Help will be offered to both technical and administrative staff of Higher Education Institutions.

2. The local authorities (including IRO staff as the business owner of the EWP processes) make a decision about joining the EWP community.

3. Depending on the situation, the institution either develops its own connector integrated with the homemade Student Information System, or deploys the one provided by the consortia running the same SIS (the most cost-effective scenario), or the one offered by commercial providers. Smaller HEIs, which do not use any sort of IT system for managing student data, will be catered for through the creation of the EWP Hub, an online platform allowing for the use of the EWP Network.

   This is the task not only for ICT specialists but also domain experts from the IRO who should cooperatively work out new procedures in the paperless mobility process. These new procedures may enforce changes in the institutional regulations concerning the mobility.

4. When the local connector is ready or at least the basic network APIs are available, the URL of the local Manifest file is entered to the DEV Registry. The EWP Technical Support Team should be contacted to make it happen. Then testing starts in the development environment. The testing phase which should be concluded with the acceptance testing of the new installation carried out by the EWP Technical Support Team. Tools and resources mentioned in chapter 3 can be used.

5. After getting approval from the EWP Technical Support Team, the local installation is accepted in the production EWP Registry.

6. The IRO staff of the institution enjoys easy data exchange with the mobility partners and the institution gets recognized as the reliable and digitally matured partner in the mobility.

Running helpdesk is a crucial task of the EWP Competence Centre, providing general support to the members of the EWP Network, in particular new comers.

6. SUMMARY

EWP constitutes a significant innovation in current practices for organizing student mobility and has a strong potential to be mainstreamed with a long-term impact. One aim pursued by the EWP project is the outreach to European and National policy makers to create a shift in administrative culture and the use of ICT tools, by proposing a publicly available network for the exchange of student data and engaging in policy dialogue in preparation of the follow-up programme of Erasmus+. This amounts to a significant contribution for the modernization of higher education, which is one of the tenets of the *Modernizing education in the EU* Communication put forward by the European Commission [1][2]. EWP is mentioned in both of the referenced documents.

The institutions of higher education engaged in student mobility are getting aware that to lower the administrative burden of the mobility they must enhance the digitalization maturity. In that respect institutions depend on one another. The institution which is ready for electronic data exchange will encourage its mobility partners to catch up, as it already happens.

Digital services of the educational institution should be exposed in a unified way, by one central access point — the EWP Registry. Once it offers EWP services, next step may be to integrated EMREX scenario, on-the-fly grade conversion envisioned by the Egracons project (http://egracons.eu/), access to the Mobility Tool+ services. We are aiming for the EWP Registry and Discovery APIs to become common underlying parts of other similar projects focused on higher education in Europe. They are designed to be extendible and can be used for all kinds of APIs (even if some of these APIs clash by serving similar purpose in a different manner, depending on the cultural context).

Data exchange means common work on data standards. EWP reuses ELMO implemented for the EMREX platform for exchange of student achievements records. Followers of both projects will continue cooperation on ELMO.

Common data standards, one registry supporting digital services, common security measures implemented by approved security protocols, common policies like the rules of accepting new

partners, GitHub as repository for code, specifications, documentation and to carry on exchange of ideas — all this will lead to a common digital EHEA for European Higher Education Institutions.

Become part of it.

7. ACKNOWLEDGEMENTS

8. REFERENCES

[1] Communication from the commission to the European parliament, the council, the European economic and social committee and the committee of the regions on a renewed EU agenda for higher education (2017). Retrieved in February 2018 from: https://ec.europa.eu/education/sites/education/files/he-com-2017-247_en.pdf

[2] Communication from the commission to the european parliament, the council, the European economic and social committee and the committee of the regions on the Digital Education Action Plan (2018). Retrieved in February 2018 from: https://ec.europa.eu/education/sites/education/files/digital-education-action-plan.pdf

[3] EMREX project. Retrieved in February 2018 from: http://www.emrex.eu.

[4] Erasmus Without Paper project. Retrieved in February 2018 from: http://www.erasmuswithoutpaper.eu.

[5] Erasmus Without Paper Developers Guide. Retrieved in February 2018 from: https://developers.erasmuswithoutpaper.eu/.

[6] Erasmus Without Paper DEV Registry Service. Retrieved in February 2018 from: https://dev-registry.erasmuswithoutpaper.eu/.

[7] Erasmus Without Paper in GitHub. Retrieved in February 2018 from: https://github.com/erasmus-without-paper/.

[8] Mincer-Daszkiewicz J., Rygielski W. (2016). *Erasmus Without Paper — from the technical perspective*", EUNIS 2016, Thessaloniki, Greece. Published in EUNIS 2016: Crossroads where the past meets the future, Book of Proceedings, p. 70-73.

[9] Mincer-Daszkiewicz J. (2017). *EMREX and EWP offering complementary digital services in the higher education area*, EUNIS 2017, Münster, Germany. Published in EUNIS 2017: Shaping the Digital Future of Universities, Book of Proceedings, p. 354-357.

[10] Mincer-Daszkiewicz J. (2018). Presentation of the running EWP Network at the final conference of the EWP 1.0 project "*Student data portability for mobility students: a look into the future*", Live stream, 26.09.2017, Brussels, Belgium.

9. AUTHOR BIOGRAPHY

Janina Mincer-Daszkiewicz graduated in computer science in the University of Warsaw, Poland, and obtained a Ph.D. degree in math from the same university. She is an associate professor in Computer Science at the Faculty of Mathematics, Informatics and Mechanics at the University of Warsaw specializing in operating systems, distributed systems, performance evaluation and software engineering. Since 1999, she leads a project for the development of a student management information system USOS, which is used in 54 Polish Higher Education Institutions, gathered in the MUCI consortium. Janina takes an active part in many nation-wide projects in Poland. She has been involved in Egracons, EMREX and Erasmus Without Paper European projects.