
UNIWeb
Interconnected Research Networks
User Introduction

1 Introduction

UNIWeb is a web-based, research information system that lets you maintain your academic information in a single data file and reuse it every time you apply for funding or submit reports. UNIWeb converts your data automatically to help you upload it to funding websites, and create the documents you need for your institution, research projects and partner organizations.

UNIWeb also includes powerful networking and collaboration tools. UNIWeb connects you to other individuals in your network by identifying common research interests. By forming these relationships, UNIWeb aims to facilitate collaboration within your local network.

UNIWeb provides valuable web visibility for your research and that of your colleagues. Visitors to UNIWeb are able to search for specialized areas of study within your network or use UNIWeb's smart clusters to browse for broader subjects. These users can also employ the map tool to discover studies being done by your researchers and international partners worldwide.

UNIWeb is proudly developed in Ottawa by Proximify. For any questions not covered by this manual please contact support@proximify.ca.

The Problem

- You are constantly asked for updated CV information by your institution, projects, research centres and networks. This requires a great deal of effort and diverts your attention from producing new knowledge;
- External funding agencies require you to provide up-to-date information each year when you apply for funding;
- Keeping research information on your institutional website up to date is time consuming and requires more duplication of your CV information;
- It is difficult to be aware of the research work of your colleagues within your institution;
- When prospective graduate students compare universities, they can easily miss out on key research groups that have no web visibility.

The UNIWeb Solution

- A central CV platform that can be used for both internal and external needs;
- A "YellowPages" directory of faculty members, postdoctoral fellows and graduate students;
- A complement to other social research networks with a focus on your institution;
- Better visibility of the research work done within your local network;
- A way of letting prospective graduate students see what the research ecosystem of your university looks like, and where they would fit in;
- New channels of communication and engagement with graduate students and fellow researchers;

- A complete set of collaboration tools, which allow you to work with your colleagues directly within UNIWeb.

2 Your Curriculum Vitae

Your academic CV is the cornerstone of your UNIWeb account. You can use your UNIWeb CV to apply for funding, complete annual reports and create a public profile. Your institution can use your CV data to maintain institutional websites and to calculate metrics and produce annual reports on your faculty's performance.

The UNIWeb CV offers many benefits. UNIWeb is a web-based application. This means that you can access your CV data at any time, from any web browser. There is nothing to download or install and you will always have an up-to-date version of your CV.

An academic CV can become a very lengthy document over the length of a career. The UNIWeb CV is all contained on a single page. This allows you to use your web browser's Find function to search for specific information. The UNIWeb CV also has various section headers to help you navigate to precise points in the document (see ① in Fig. 1).

You can either begin composing your CV from scratch, using UNIWeb's intuitive interface, or you can import your existing data from a variety of sources as outlined in sections 3.1 and 3.2.

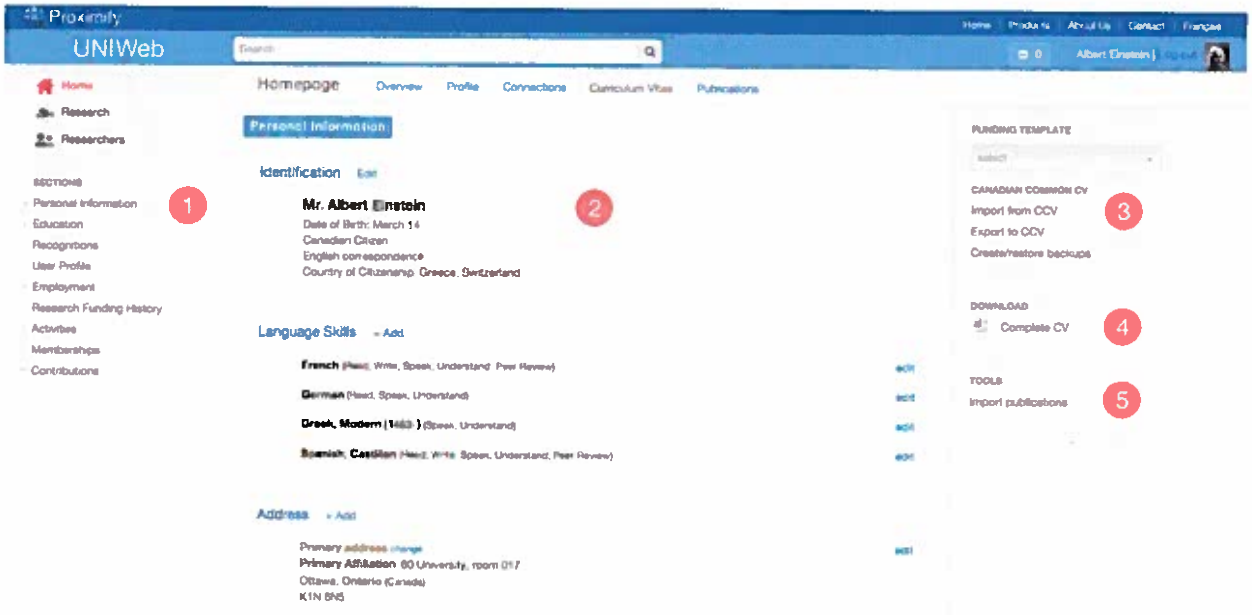


Figure 1: *CV Page Overview ① Quick shortcuts to CV sections ② Current CV template view and editing area ③ Import from or export to the CCV website, restore CCV backups and CCV funding templates ④ Download various CV formats (Generic, OCGS) and file types (.pdf, .doc, .odt) ⑤ Import publications*

2.1 Importing from the Canadian Common CV

If you are a Canadian researcher who already has data on the CCV website, you can import that data into UNIWeb.

1. From the Curriculum Vitae page, choose the Import from CCV option (see ③ in Fig. 1).
2. Enter your account username (see ① in Fig. 2).
3. Enter your account password.
4. Select Import from CCV.

This may take several minutes depending on traffic to the CCV website as well as how much of your CCV you have completed.

Once your data has been imported, you can begin working on your CCV from directly within UNIWeb. To learn how to export your data back to the CCV see section 5.

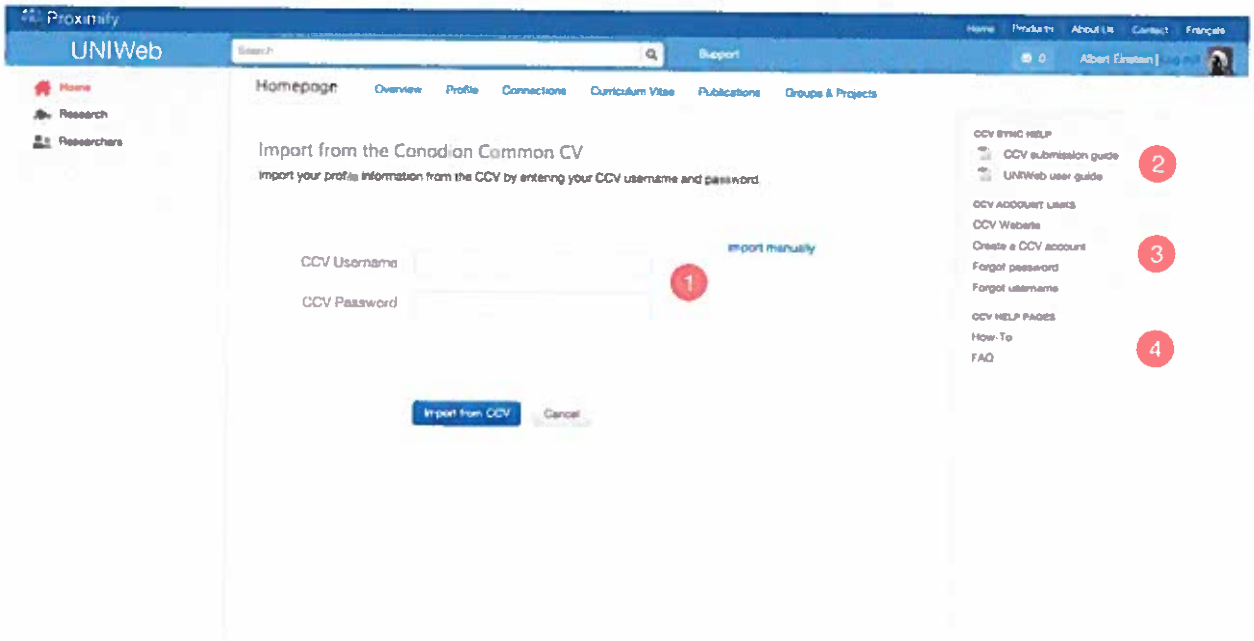


Figure 2: *Import from the Canadian Common CV ① CCV login ② Proximify help documents ③ CCV account links ④ CCV help documents*

2.2 Importing your Publications

UNIWeb can import publications from a variety of sources including EndNote, BibTex, Google Scholar and PubMed (see ⑤ in Fig. 1).

You can follow a simple, three-step process to import your publications from any of these sources. UNIWeb also ensures that it does not duplicate any of your existing publications. For

BibTex and EndNote, you also have the additional benefit of only importing publications for which you are the author or editor.

2.3 Downloading your CV

In addition to giving you an interface to edit your CV online, UNIWeb also offers the ability to download your CV.

1. From the Curriculum Vitae page, choose the Download Complete CV option (see ④ in Fig. 1).
2. Select your desired CV layout.
3. Choose how many years of data you would like included in your CV.
4. Choose your desired file format (.pdf, .doc, .odt).
5. Select Download.

If you select the Generic CV layout, your CV will retain the basic UNIWeb CV formatting. However, your institution may choose to include their own layout option here.

If you select your institution's layout, all of your CV data will be automatically entered into the template and formatted to fit the specific requirements of your institution. These layouts can format your CV data into lists and tables and can also perform calculations to add up your funding, providing the important data your institutions need.

3 Identifying Potential Collaborators and Advisors

More and more, research grants are awarded to multidisciplinary groups with the right combination of expertise, knowledge and experience. UNIWeb allows researchers to find potential collaborators and graduate students within their own university. The software's objective is to achieve a measurable increase in the number of grants awarded to faculty members.

UNIWeb lets you recognize and find potential collaborators through a multitude of channels, allowing you to build stronger teams of experts. The subsections below discuss the different tools and strategies that can help you to identify people of interest within your university.

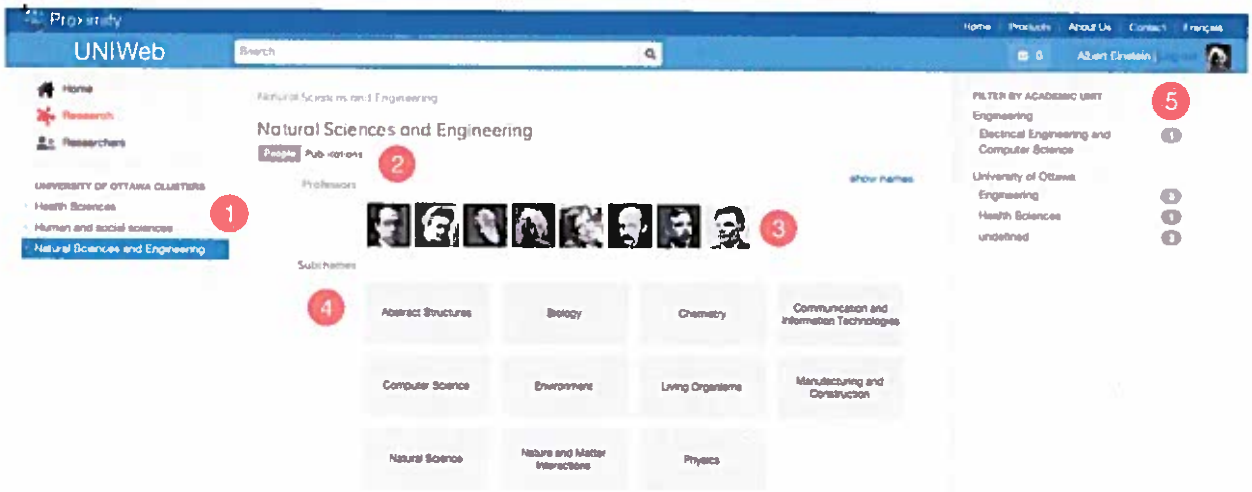


Figure 3: *Research Cluster Overview ① Hierarchal list of research clusters ② Toggle between displaying People and Publications ③ List of people grouped by type ④ List of sub-themes within which those people are interested ⑤ Filter by academic unit*

3.1 Using Filters

All UNIWeb network pages include powerful filters to help you easily browse or search your colleagues (see ⑤ in Fig. 3). Page content can be filtered by selecting options under different filter categories. Clicking one filtering option will narrow the results, while selecting more options under the same category will make the filter wider.

Selecting filtering options under different categories creates a narrower filter that combines the options across categories. You can turn off a filtering option by clicking it again.

3.2 Finding People within Specific Research Areas

Research clusters span multiple departments and faculties (Fig. 3). When viewing a research area, you can see the multidisciplinary nature of its members without being restricted to individual faculties or departments.

You can use UNIWeb's search bar feature to find individuals by name or you can find researchers that share particular interests by exploring the research clusters. Choosing a specific research area allows you to view people who are interested in that research area as well as the sub-themes of that area.

By default, all people belonging to the research cluster being viewed share a common high-level interest or connection. The filters on the right allow you to display individuals and sub-themes that are found in specific academic units.

3.3 Finding People with Common Research Interests

Selecting a person under the Researchers sections allows you to view their Profile and Connections. By selecting their Connections you can view the people with whom they share a common research interest (Fig. 4). You can then use the filters on the right to narrow down their connections by category, academic unit and research interest.

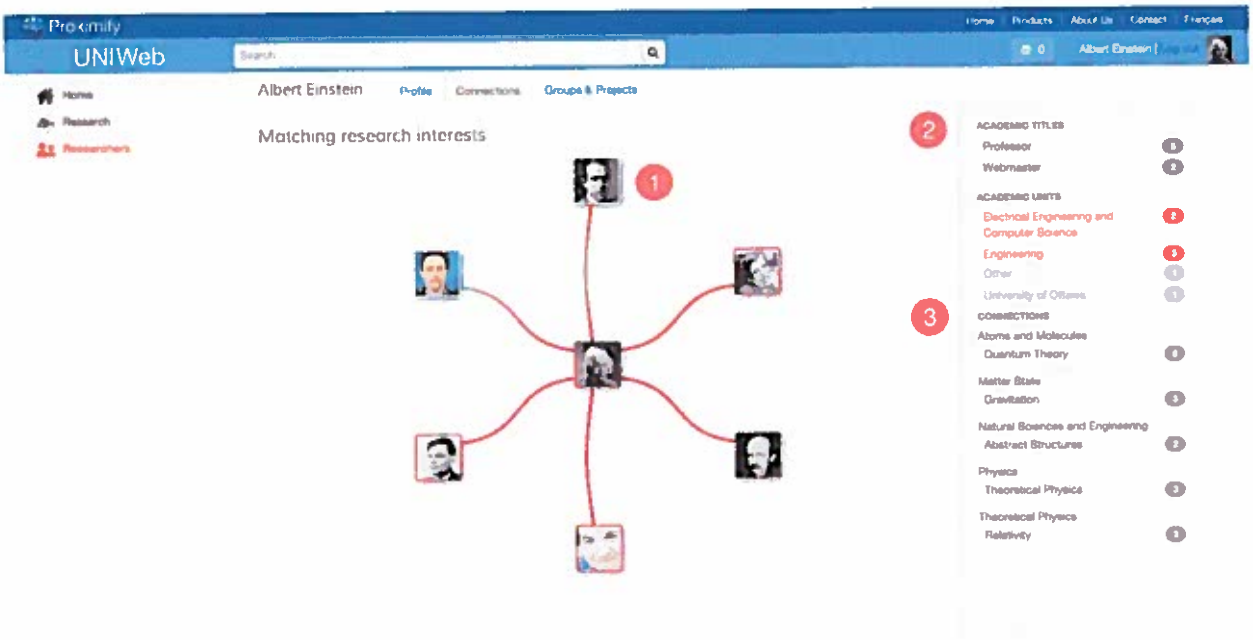


Figure 4: *Connections Page Overview ① Mouseover thumbnails or connections for detailed information, click on a thumbnail to jump to that person's network ② Filter connections by categories and academic units ③ Filter connections by research interests*

The easiest way to find colleagues with common research interests to yourself is to view your own public connections. These can be found by either selecting the Connections tab at the top of the Home page or by finding your name in the list of researchers and viewing your connections.

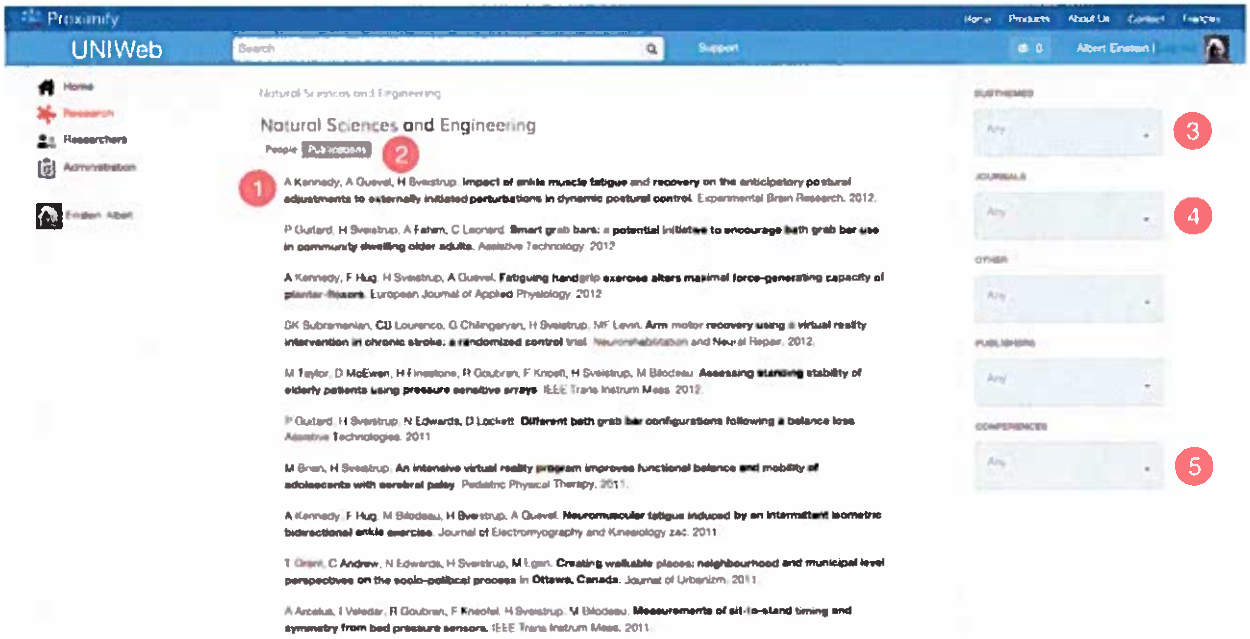


Figure 5: *Research Cluster Publications Overview ① List of publications ② Toggle between displaying People and Publications ③ List of sub-themes of the publications ④ Filter by journal name ⑤ Filter by conference name*

3.4 Finding People who Publish in Specific Journals or Conferences

You can find people who publish within a particular research area (and its sub-themes) by searching for it in the UNIWeb's search bar and then clicking the Publications tab (see ② in Fig. 5). In this view you will have access to various filters that allow you to identify the researchers who publish their work in specific journals and/or conferences (see ④ and ⑤ in Fig. 5).

Selecting a journal filter will display results that span various themes. Conversely, selecting a journal with a specific research area and sub-theme will refine the results to only those that meet both criteria.

3.5 Finding People Based on Geographical Location

UNIWeb boasts a powerful GIS (geographic information system) feature that allows you to search for members of your network based on location. This can be extremely helpful when searching for collaborators who have expertise in a specific region.

On your profile page, there is a section titled Research Places. This allows you to define a location where you have conducted a research activity. You begin by entering an address. Then, you can add a description, choose a type of research activity and even tag your location with research interests. Finally, you have the option to specify a start and end date.

Once you have added locations to your profile, you will now appear in the 'Research Places' section of UNIWeb. To find other researchers, you can either search by name or location, or you can filter by title or academic unit.

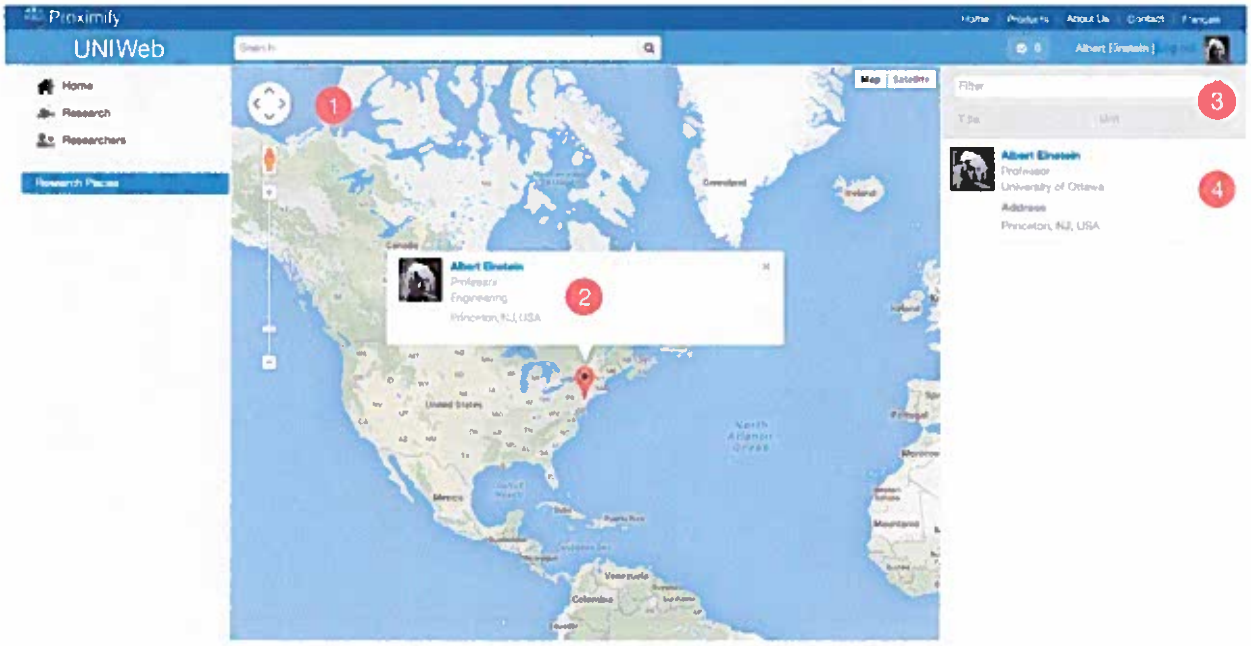


Figure 6: *Research Places Overview ① Researchers displayed on map ② Profile summary ③ Filter by name, address, title and academic unit ④ Search results*

4 Research Interests

Research interests are one of the most important components of a UNIWeb network. Research interests are what creates the connections between members of your network.

Every UNIWeb network comes with an extensive, pre-defined list of research interests. This list comes directly from the CCV. However, as a user, you also have the option to add additional research interests.

Research interests are nested. Meaning that parent interests are subsequently broken down as they get more specific. When adding a new research interest, it is important to be as specific as possible. For more information on the hierarchy of research interests, please see the example in 4.1.

4.1 Adding Research Interests to your Profile

To add a research interest to your profile, begin by navigating to the Research Interests section. There are two ways to add a research interest from your network's existing list. First, click within the text box and scroll through the list of available options. Second, begin typing your desired interest in the text box and UNIWeb will refine your results.

You also have the option of creating a new research interest. To do this, you must select a parent interest and then create a new interest name (Fig. 7). In some cases, you may need to add several nested interests at once. To do this, simply separate their names with ">".

Example

Nest new theme under: **Health Sciences**

New theme name: **Aging > Chronic Diseases in the Elderly > Alzheimer's Disease**

This will create Aging as a new theme and nest it under the existing theme of Health Sciences. In turn, Chronic Diseases will be created and nested under Aging and Alzheimer's Disease will be created and nested under Aging. Finally, Alzheimer's Disease will be added to your Research Interests.

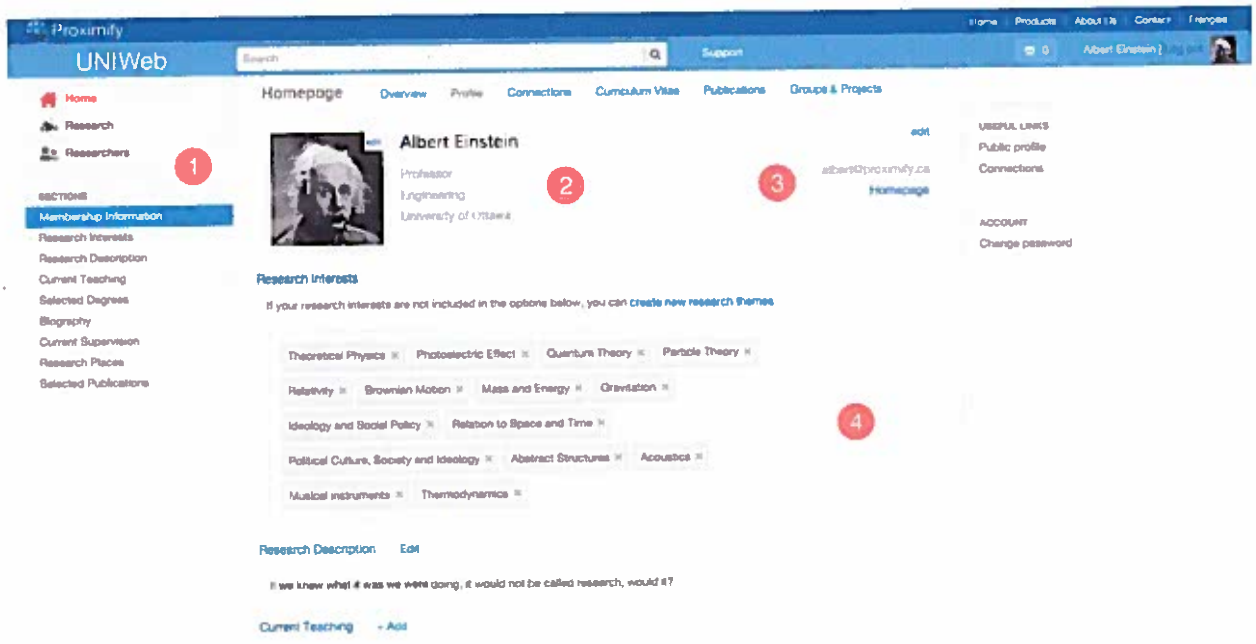


Figure 7: *Profile Overview ① Profile sections ② Name, title, department, institution ③ Contact information ④ Research interests*

4.2 Tagging your Publications

UNIWeb gives you the option to tag one, or many, of your publications with a research interest. Other users will then be able to locate this publication by research interest and it will appear on your UNIWeb profile.

To begin, navigate to the Publications page or the Publications section of your CV. Select your desired publication or publications and choose Add Tags from the right-hand menu. You will be presented with a list of research interests that you have already applied elsewhere in UNIWeb. However, you also have the option to add new research interests from your network's list.

To remove a tag, simply select your desired publication or publications and then choose Remove Tags from the right-hand menu and follow the same process.

4.3 Tagging your Research Places

When you add or edit a research place, UNIWeb gives you the option to tag it with a research interest. To find your desired interest from your network's list, you can either scroll through the list or you can begin entering the name in the search field and UNIWeb will refine your results.

By tagging your research places with interests, you will increase your visibility on the Research Places page and will allow your colleagues to find you based on your research interests.

5 Applying for Funding with the Canadian Common CV

The process of applying to a funding agency that requires a CCV submission begins by exporting from UNIWeb to the CCV website. The necessary steps are described below.

On UNIWeb

1. Enter your information into the UNIWeb Curriculum Vitae interface.
    a. If you have already begun a CCV on the government site, you can import your work directly into UNIWeb (see section 3.1).
    b. If you know the agency to which you are applying, selecting your desired Funding Template (see ③ in Fig. 8). This will condense the CV to only the required sections and fields of that agency, and will verify that all required fields have been correctly filled.[1]
2. Select the Export to CCV option from your right-hand menu (see ③ in Fig. 8). You will be prompted to enter your username and password.

1 The generic CCV is almost void of restrictions for required data but provides the most rich data options. The individual agencies have stricter field requirements and fewer fields.

Note: You must **not** be logged into the CCV website while you are exporting to the CCV. If you remain signed in, your new UNIWeb CV will not be able to replace your existing CV.

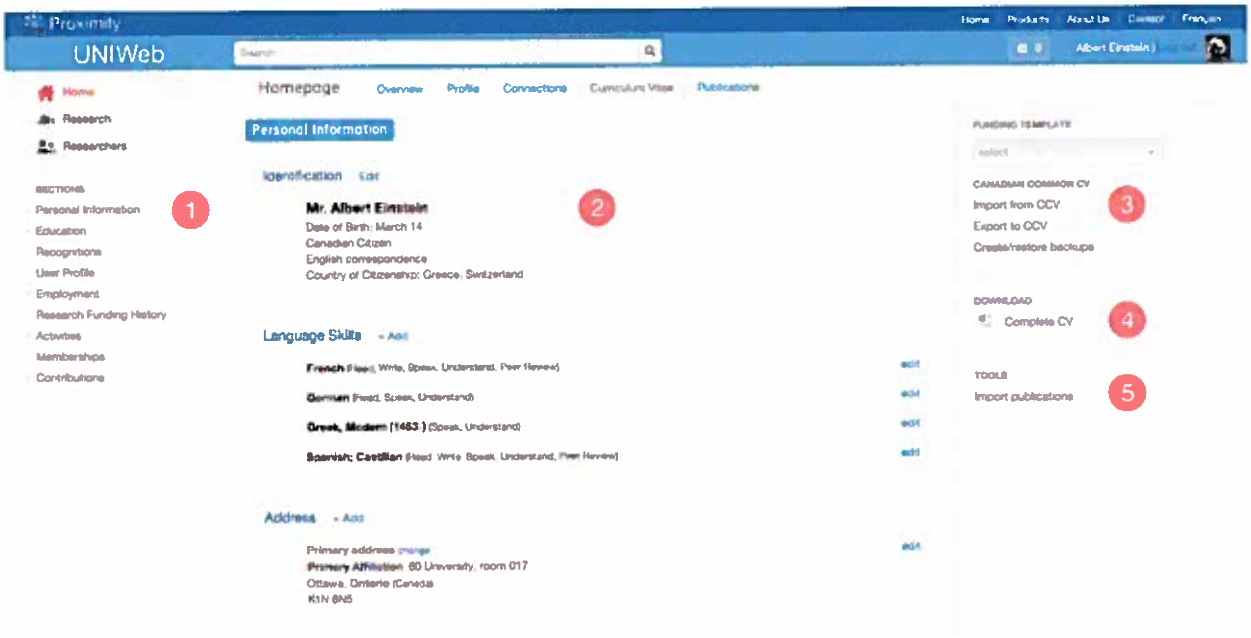


Figure 8: CV Page Overview ① Quick shortcuts to CV sections ② Current CV template view and editing area ③ Import from or export to the CCV website, restore CCV backups and CCV funding templates ④ Download various CV formats (Generic, OCGS) and file types (.pdf, .doc, .odt) ⑤ Import publications

On the CCV Website

Once you have exported your data to the CCV, you must continue the application process on the CCV website.

3. Log on to the CCV website.
4. Navigate to CV / Funding.
5. Select the Funding Source you would like to apply for.
6. Select the CV Type. After which the CCV site will check to see if you have errors.
   a. If you have errors, return to UNIWeb and continue from step 1b.
   b. If you choose to fix your errors directly on the CCV, be sure to return to UNIWeb to re-import your CCV to keep your information synced and up to date.
7. Submit and agree to the Terms of Use and choose whether to be listed in the Researchers Directory (which may require the entry of additional information).

Certain funding agencies expect less information than what is available in the Generic CCV. Part of this "information reduction" is automatically handled by the CCV funding templates, which you can apply in UNIWeb (see ③ in Fig. 8). You can also verify your data prior to exporting by selecting a funding template on the Export to the Canadian Common CV page (see ③ in Fig. 9).

However, there is one aspect that you might need to manually change: some agencies demand specific items removed if they do not meet certain criteria. For example, some SSHRC applications require only the last 6 years of publications. In this case, you must uncheck the 'Submit?' field of each publication that does not meet the condition. This may be done on the CCV website or you can choose the 'Automatically mark publications as 'Submit' based on their date' option from the UNIWeb export page (see ④ in Fig. 9).

Another example is presented in CFI applications, where only one email can be submitted, so you will be required to uncheck any additional emails as shown below.

Note: The 'Submit?' field will remain de-selected when you return to the Generic CV page. If you wish to reset the 'Submit' status of all items, you can do it by simply re-importing your CV from UNIWeb. This will ensure that all items in your CCV are marked as 'submit'.

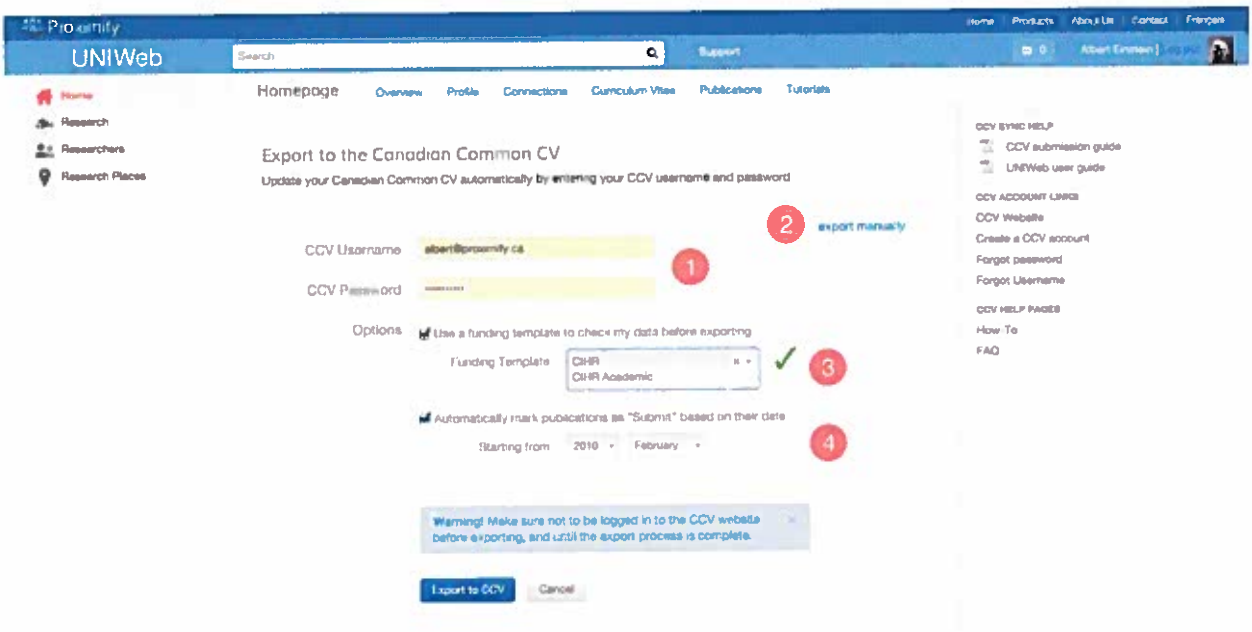


Figure 9: *CVV Export Overview ① CCV username and password ② Export manually to an XML file ③ Funding template to check data before submitting ④ Automatically mark publications as 'Submit' based on their date*

5.1 Backups

The system makes backup copies of UNIWeb's data daily. In addition, whenever you export to the CCV website, a backup of your information on the CCV website is taken. All backups displayed on the backup list page are previous states of your data on the CCV site prior to an export. This can be used in case data in the CCV site was accidentally overwritten during an export operation.

6 Improving Web Visibility

Having a web presence can be important for connecting with colleagues outside of your institution as well as allowing your work to be seen by a larger audience.

When you activate your UNIWeb account, you are given a professional profile. A basic profile will display a name, title, academic unit, contact information and perhaps a profile picture. However, there are benefits to adding additional information to your profile.

First, this profile helps you to connect to, and share information with, members of your network. It is easy to add information by simply clicking on the Add or Edit button next to each section.

Second, this profile serves as a valuable web presence. When you select the Public Profile option, you can see how your profile appears publicly. Potential colleagues can now find up-to-date information on your research interests, publications and more by simply entering your name into their search engine.

Note: It is not mandatory to complete this profile. You are not required to enter any information that you do not feel comfortably sharing publicly.

7 Frequently Asked Questions

Q1. I have already spent 2 weeks creating my Canadian Common CV on their government website, what is the benefit of using UNIWeb's CV conversion function?

We understand the headache involved in using the interface provided by the CCV website. Fortunately, things will be much easier going forward thanks to UNIWeb's modern interface, which is designed to simplify data entry efforts and save you valuable time.

Q2. I don't have to submit an application for funding this year, how is the software useful to me?

In many ways! While professors are often looking to attract talented individuals to help them achieve their goals, researchers (including graduate students and postdoctoral fellows) are similarly trying to find professors working on interesting projects. Getting connected is often the struggle.

The challenge lies in the fact that information pertaining to a given research theme, including work that has already been done within your institution, academic unit or by a group of researchers, is often scattered and hard to find. UNIWeb provides researchers the means to share their achievements in a visually engaging manner, all in one place.

There are two necessary steps to achieve better visibility. First, take a few minutes to input your research interests. This will let anyone in the world who visits your web profile see what you are working on, and how those research interests connect you with other researchers in your network. Second, label your publications with appropriate research themes. This will let visitors and colleagues find you by searching for people that publish in the same journal and conferences. This will also demonstrate your level of expertise in each research topic.

Entering these two critical pieces of information will attract prospective students, and promote interdisciplinary collaboration, all while getting a head start completing your CCV.

Q3. Can't people already find me on the faculty website? Why should I put time into creating a UNIWeb account?

Most university websites are built on the premise that people will know under which faculty and department to look for research done within their areas of interests. However, this structure does not facilitate the interdisciplinary initiative. If the same topics can be studied from a multitude of angles by researchers across a university, why is the web content only accessible through Faculties and departments?

UNIWeb does not follow this model; research themes are not attached to any particular department or faculty. However, viewers can still slice the information by such parameters using the powerful filtering options offered by the software. UNIWeb promotes multidisciplinary research by providing overlapping views of the research done across your university.

By entering your publications and research interests, you will increase your personal web visibility, as well as that of your Faculty. Prospective students, alumni and researchers abroad will be able to understand your work within the context of your colleagues in the university.

Q4. I barely have time to update my CV, how is the software going to help me with writing my CCV?

Submitting your CCV is now an inevitable step when applying for governmental funding. Whether you need to apply for funding this year or not, you will eventually have to go beyond updating your own CV. This is where we come in. You can input your information in UNIWeb's up-to-date CCV templates, without worrying about the complex interface of the CCV website.

In addition to cutting down the time and effort required to create your CCV, the UNIWeb CV lets you reuse that data to generate your academic CV with a choice of layouts, such as that preferred by your institution.

Q5. How is UNIWeb different from other social research networks?

You may be familiar with web solutions like GoogleScholar, Mendeley, ResearchGate or Academia.edu - all focused on providing a platform for document curation.

UNIWeb acts as a neutral party - it is not a platform where researchers are ranked. Some systems are designed to help you find and download papers. UNIWeb is designed to help you

find people within your institution. The purpose of UNIWeb is to promote multidisciplinary research in which natural connections can seamlessly grow. The software keeps invalid users out - professors must be invited by the university, students by their supervisors - and provides a private, institution-centric system for trusted users.

REST File

Solution Overview – Holding Area

Overview

The UniWeb system is a new system for managing CV data for professors in a centralized repository. Some of the data that will be stored in this system is currently being held in other systems. Our goal is to allow users to control the flow of data from those systems into the CV system, to ensure that only the correct data is moved into the system. The primary user objective is to make managing their electronic CV easier, by leveraging existing data.

More specifically, the Holding Area project is designed to meet the following business needs.

- A logged in user should be able to see a list of their own data that has been updated in various systems (see Integration section below) that can be mapped into the UniWeb system.
- A logged in user should be able to approve a change from that list and have it updated in UniWeb.

As this is a new project, it is intended to follow the recently discussed and adopted best practices and standards, in order to ensure that new systems are robust and offer quality to our users.

Integration

What follows is a list of systems that are planned to be integrated with the UniWeb system through the holding area, along with their expected update frequency.

1. Stipends will provide information on courses taught for UGME and PGME (weekly).
2. Mainport – Royal College portal will provide information on learning activities (monthly).
3. Qtips will provide academic work experience information based on appointments, re-appointments and cross-appointments (monthly).
4. Entrada (CPD and BAF) will provide information on courses taught and taken (monthly).
5. Academic promotions will provide information on academic work experience based on promotions (yearly).
6. Faculty department websites (Psychiatry, Surgery, Internal Medicine, and Obstetrics-Gynecology) will provide information on supervising students and event participation (weekly).
7. Physician Annual Review will be integrated into this system (to be defined better).

UniWeb Integration

The team has confirmed that writing to UniWeb does work as documented. However, there remain several challenges around the integration that must be addressed.

Of these, the most notable one is writing an update to a data set. This type of data is basically a list of items. An example of this would be updating a course that is taught or a work experience item. Since there are multiple systems providing similar types of data, and since that data is in flux, it is likely that we will need to handle updating a specific item at some point. The UniWeb API allows this to happen by clearing the entire data set and then rewriting it all. There is no method for targeting a specific item within a data set. This is likely to lead to concurrency conflicts in the future, causing data loss within UniWeb; as a trivial example, imagine two concurrent API calls that want to read the data set, delete the set, and rewriting all the items back with one change. Should the delete from one be called before the read from the other, the second call will have no item to update and may write back a blank set. Data set updating is therefore not thread-safe.

The next challenge is also surrounding data sets, and duplicate data between systems. In at least one case (academic work experience), information is pulled from two different systems (Stipends and Promotions). The ideal system would be able to detect that these are the same items and not create multiple objects for them. However, this poses significant challenges as UniWeb does not provide primary keys for existing data entries nor any deduplication facility of its own.

Solution Overview

Requirements Overview

Any acceptable solution will have to address the following broad requirements:

1. Allow users to be authenticated into the system.
2. Integrate with multiple different services to pull data.
    a. Find the proper user ID for the current user for the given system.
    b. Access a service API to pull certain types of information about that user.
    c. Monitor for changes in those types of information regularly (no more than weekly).
3. Allow users to control what data is pushed to UniWeb.
    a. Browse the list of changed data points.
    b. Accept them to be pushed into UniWeb.
4. Integrate with UniWeb to write data.
    a. Map data pulled from different services into the UniWeb API.
    b. Deduplicate and unify data from different services.
    c. Detect items that already exist within UniWeb, and update them instead of creating a new item.
    d. Write creates or updates to UniWeb in a thread-safe manner.

Open Questions

The following questions should be addressed in scoping the project:

Business

1. How do users login? What authentication source can we use?
    a. Do we need to limit which users have access to this tool?
2. How likely is it that data will need to be changed / de-duplicated? – block1: None block2: Maybe
3. Can users interact with change requests in any other way? For example:
    a. Can they ignore them? If so, should they be able to see what they have ignored?
    b. Can they cancel them if they haven't been processed yet (if there is a delay)?
    c. Can they request we revert them if they have been?
4. How much of a delay between accepting a change request and seeing it in UniWeb is acceptable?
5. How do we de-duplicate data between services and with existing data in UniWeb? → Leave it to user.
    a. How much control will/should the user have in making these decisions?
    b. How do we handle conflicting field updates?
6. What constitutes a change?
    a. Is it a change in source data? A change in what would be written to UniWeb?

Technical

1. Which integration points require additional services to be built?
2. Does data need to be transformed between the source services and UniWeb (e.g. date formats)? no
3. Do we have sufficient information available to associate all of the accounts together? YES
4. Is the data mapping too complex to do in configuration? DATA MAPS ARE AVAILABLE
5. Should we be queuing requests to UniWeb? Clarify

Proposed Components

RESTful Framework

As will be seen in the other components, new RESTful services will be developed as part of this project. Since there are few good examples of such services already written, it would be best if best practices were outlined for these services before we started. These best practices would address the following concerns:

• Intra-service calls • Code framework • Resource naming • Meta data distribution	• Security (HTTPS only) • Authentication / Authorization • Logging / Auditing	• Deployment • Monitoring • Testing

Identity Provider

In order to map data between services, it is essential to know how user IDs map between systems. In order to resolve this problem permanently, a micro-service should be built that allows a system to map one user ID into another. As this system will have to manage a user set already, providing user authentication and authorization through the same application makes sense. The broad requirements for this system would be:

- Authenticate that a user is who they say they are, and provide a token for clients to continue to use.
- Given a token, verify that the token is still valid.
- Look up an alternative user ID for a given user (or token).
- Manage the alternative user IDs for users.
- Look up whether or not a user (or token) has access to a given system or element.
- Manage which users have access to which systems.

Note that, as we are dealing with semi-sensitive information (user IDs from different systems), we should make sure there is appropriate security in place to prevent unauthorized requests.

Information Exchange

The ideal method for managing information exchange between systems would be to write a configurable tool for doing so. By defining how data flows between systems in configuration instead of code, future additions can be pushed out quickly and seamlessly while also involving a minimum of developer time. Such configuration would have to include:

- Defining a list of RESTful resources that can be used.
    - The system should be able to pull which type of ID is needed, and be able to get that ID from the identity provider.
    - Most support will probably be for a GET of listings and then a GET of each item.
    - UniWeb Write API should be included in here.
- Defining how resources are mapped between systems.
- Defining how resource properties are mapped (and possibly transformed).
- Defining how de-duplication is done (and how any conflicts are handled).
- Defining how change is detected.

As an alternative to the above, configuration could be implemented through code so that it is more flexible (but deployment will be required for changes).

The broad interface for this system would be:

- Triggering a new check for changes on a given service (different services are scheduled differently).
- Browsing a list of changes for a given user.
- Approving a change for a given user.
- Support for other operations on data as needed (see open questions).

In addition, the best way to write to external services is through queuing (to prevent load problems on their end and responsiveness problems on our user's front end). Approving a change should be an asynchronous operation therefore, with a service to trigger pushes.

UniWeb Integration

In order to avoid the problems in UniWeb, the best option is to layer it behind a new RESTful API that hides these concerns from service users. This new API will focus on implementing the resources required for integration first:

- Members
  - Courses Taught
  - Events Participated
  - Academic Work Experience
  - Presentations
  - Student Supervision

Service Integration

For each of the other integrations, it would be ideal to have a RESTful API available so that mapping can be configured. Such a service will have to be created where it does not exist.

User Interface

Finally, a user interface will need to be constructed to allow users to monitor the changes in their data and approve them to be moved into eCV. The broad requirements for this interface will be:

- Ability to browse changes to their personal data.
- Ability to approve moving those changes along the mapping into the UniWeb system.

Work Plan (Tentative)

Please note that estimates in italics are subject to change based on the business use cases and will be refined as the project progresses. It is also assumed that service integrations already exist for the given systems. Physician Annual Review is not included as it lacks specifications at the moment.

Step	Name	Description	Owner	Estimate
1	Create business use cases	Define all the interactions that need to happen. For this project, this should include at least the following: • User authentication and authorization • Browsing of change requests • Manipulating change requests (each action from each valid states) • Reading and detecting changes from each system • Writing to UniWeb after approval of request Estimate: there will be about 25 of these, two hours each.	Edy	50 hours
2	Design micro-services framework	Can happen concurrently to initial planning work. Define how we want to go about creating a micro-services environment for Medtech.	Robert / Jim	50 hours
3	Formalize architecture	Based on business use cases, formalize the architecture of the project in a specifications document for each component. Two hours each.	Robert	*50 hours*
4	Estimation	Based on the architecture and the complexity of the requirements, an estimate will be prepared for how long the entire project will take (refining the ones prepared here).	Robert	6 hours

5	Resource Analysis	Define what the resources will look like. Estimate there will be about twenty of these, at two hours each.	Robert	*40 hours*
6	UI Prototyping	Define what the user interface will look like, using mock services.	Robert	*60 hours*
7	Identity Provider	Analyze, design, build, and test the identity provider service.	Robert	*120 hours*
8	Information Exchange	Analyze, design, build, and test the information exchange service.	Robert	*240 hours*
9	UniWeb Integration	Analyze, design, build, and test the UniWeb integration component.	Robert	*90 hours*
10	UI Integration	Integrate the user interface with the information exchange and identity provider components	Robert	*30 hours*
11	Integrate with MOCOMP	Leverage an existing REST service or provide a micro-service on top of it, and integrate it into the information exchange service.	Robert	*90 hours*
12	Integrate with Stipends	Leverage an existing REST service or provide a micro-service on top of it, and integrate it into the information exchange service.	Robert	*90 hours*
13	Integrate with Qtips	Leverage an existing REST service or provide a micro-service on top of it, and integrate it into the information exchange service.	Robert	*90 hours*
14	Integrate with Entrada (CPD and BAF)	Leverage an existing REST service or provide a micro-service on top of it, and integrate it into the information exchange service.	Robert	*90 hours*
15	Integrate with Academic Promotions	Leverage an existing REST service or provide a micro-service on top of it, and integrate it into the information exchange service.	Robert	*90 hours*
16	Integrate with Faculty department portals	Leverage an existing REST service or provide a micro-service on top of it, and integrate it into the information exchange service.	Robert	*90 hours*
17	Testing & Deployment	This will be iterative throughout development.	Robert	*42 hours*

Based on the above tentative work plan, it will be about seven weeks until we can start development (though this depends on the scope of the business cases as well). This puts us at the end of the year, meaning development can begin in January 2016. Assuming the problem is no more complex than estimated, there are about two full quarters of development. This means we will be finished delivering (except Physician Annual Review) in fall 2016 (September-October), with the first iteration shipping in early spring (May-June). A functional prototype (no real data) would probably be available for review in March. It is important to note that there are still many assumptions to validate and that this timeline could change depending on them. To summarize this approximate timeline:

Target Date	**Deliverable**
Late November 2015	Business use cases, RESTful architecture best practices
Christmas	Specifications, refined estimate
March 2016	Functional prototype demo
Spring 2016	First integration live
Fall 2016	Six integrations live (MOCOMP, Stipends, Qtips, Entrada, Academic Promotions, Faculty portals)

Conclusion

This project provides us with an excellent opportunity to move towards a robust micro-services environment with thin-client front-ends. While adapting the team to those patterns may take a bit longer than hacking it together, the result will be a robust system that can be used as an exemplar for future development while also providing an efficient and useful tool for users to control their CV material.

Jem by Nien Nong: 7/10/2015

Edy Lopez x 8324

1/9

UNIWeb API

Overview

The purpose of the API is to integrate UNIWeb with other systems within your organization. The Authenticated API provides secure read/write access to information stored by UNIWeb, and it provides a mechanism to reduce the need to duplicate data.

The UNIWeb API provides:

- An interface that allows you to control who has access to your institution's data through our API.
- A means by which to securely read and update your institution's information.
- Rich data in simple, straightforward JSON for maximum readability and reusability.
- The choice to pre-filter the requested data, to obtain just the subset of information in which you are interested.

The UNIWeb API uses Internet Engineering Task Force (IETF) open authentication standard OAuth 2.0, for authentication and authorization using the Resource Owner Password Credentials Grant protocol. In this protocol, only users with granted permission can access the API.

The following four steps are required to access a resource through API:

1. Get permission to create OAuth 2.0 Clients ✓
2. Create a OAuth 2.0 client and obtain client credentials ✓
3. Use your Oauth 2.0 client credentials to retrieve an access token.
4. Use your access token to interact with the API. Your token is valid for an hour from the time it is issued.

These steps are explained in more details below.

Setting up Authorized Clients

OL

1. Get permission to create OAuth 2.0 Clients

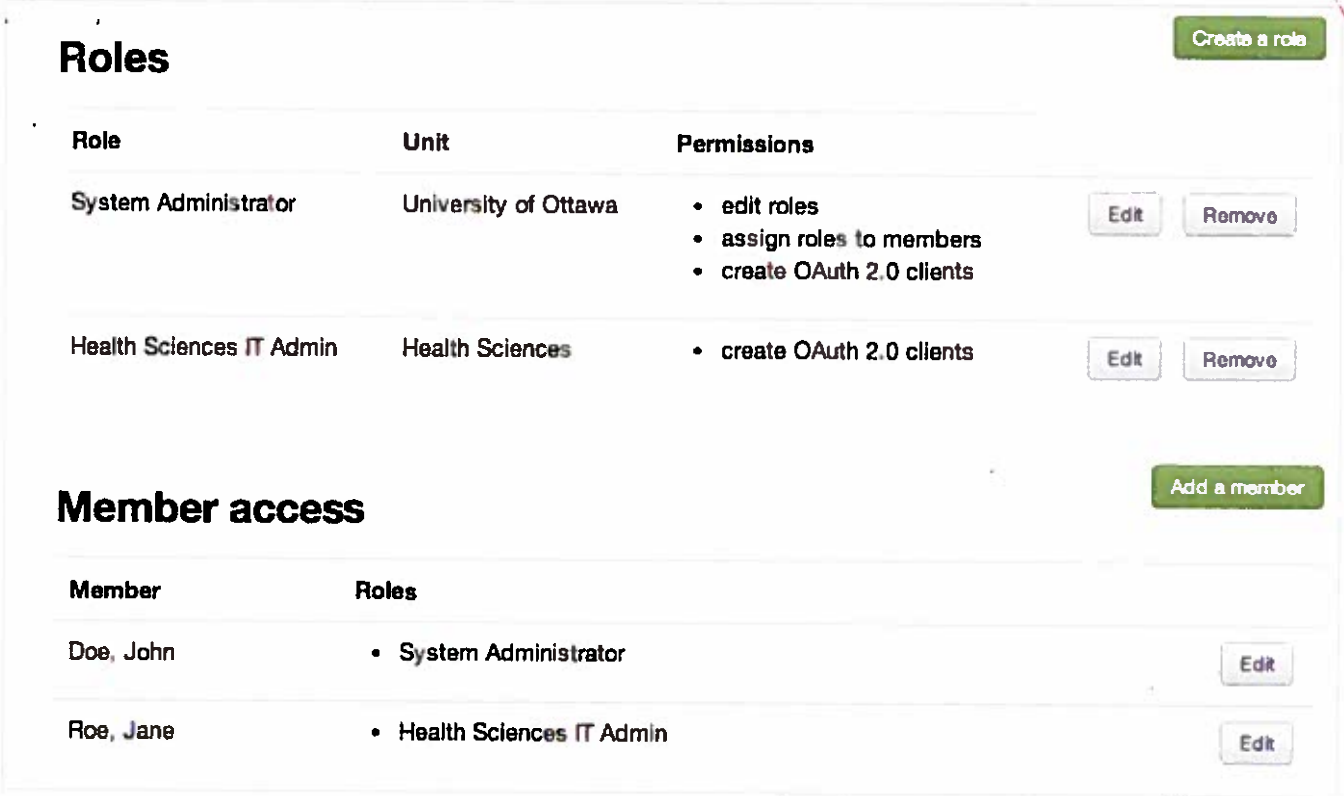
A *System Administrator*, can grant any user permission to *create OAuth 2.0 clients*. If you are not the *System Administrator* yourself, ask the *System Administrator* to give you this permission, as example below shows:

Token = is an object which represents the right to perform an operation.
(data security) is the process of substituting a sensitive data element.

JSON API spec, for how a client should request that resources be fetched or modified, and how a server should respond to those request.
https://application/vnd.api+json)

CONVENTION KEY WORDS: MUST MUST NOT REQUIRED
SHALL SHALL NOT SHOULD SHOULD NOT
RECOMMENDED MAY OPTIONAL

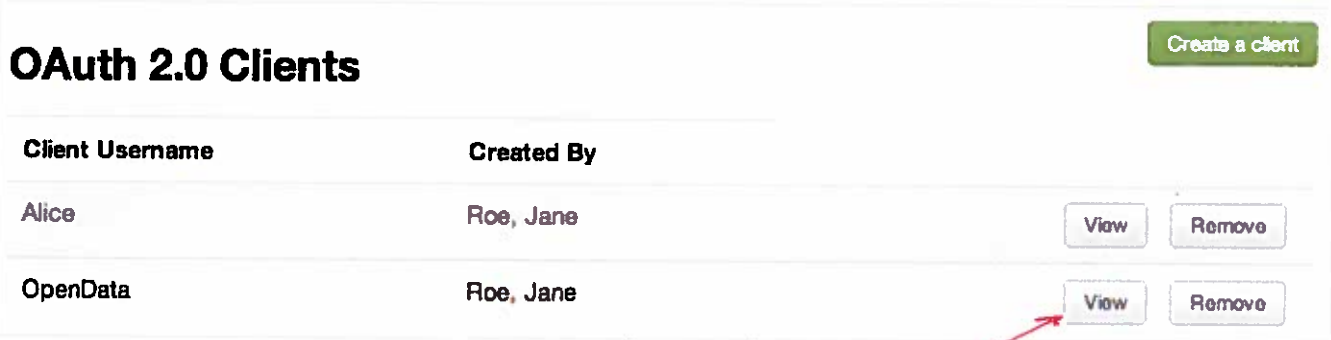
1



Example above shows *Access Control* page, accessible from *Administration* panel, to *John Doe*, the *System Administrator*. In this example, role *Health Science IT Administrator* has the permission to *create OAuth 2.0 clients* for *Health Sciences* department. *John Doe* assigns this role to *Jane Roe*. *Jane Roe* can now *create OAuth 2.0 clients*.

2. Create a OAuth 2.0 client and obtain client credentials

Using the UNIWeb Interface, you can create, edit, view and remove OAuth 2.0 clients. Each client has a unique username referred to as *Client ID*, and a system generated random password, referred to as *Client Secret*. Example below shows *Jane Roes*'s *OAuth 2.0 Administration* page.



In this example, *Jane Roe* has created two OAuth 2.0 clients. Clicking on the *view* button for *Alice* reveals her*Client Secret* as shown below:

In this hypothetical case, *Alice*'s *Client ID* is *Alice* and her *Client Secret* is *7740731b32440350fccd*. These credentials are used in the next step to authenticate *Alice*.

3. Authenticate and get an Access Token

With the client credentials obtained in step 3, you can authenticate to the UNIWeb *Token Endpoint*, and get an*Access Token*. An *Access Token* is valid for one hour, and it will be used in the next step to retrieve resources from UNIWeb *Resource Endpoint*.

With these pieces of information you will be allowed to make API requests. To do so, you can use one of our pre-built client libraries :

- PHP client lib
- Java client lib
- cURL client lib

or you can write your own by following the information in the API In Depth page.

4. Access information through structured requests

API requests are made by submitting JSON objects to the server. They tell the server which action, resources, sections and fields are desired and what filters to apply. In particular, the request objects can have the following properties: **action**, **content**, **id**, **filter**, and **resource**.

Example request object:

```
{
    "action": "read",
    "content": "members",
    "id": "bob@mail.ca",
    "resource": [
        "profile/biography",
        "profile/selected_degrees"
    ]
}
```

Example response for the above request:

```
{
    "profile/biography": "Bob always knew he would be a great scientist",
```

```
    "profile/selected_degrees": [
        {
            "degree_name": "PhD",
            "organization": "McGill University",
            "specialty": "Materials Engineering"
        },
        {
            "degree_name": "Engineering",
            "organization": "University of Ottawa"
        }
    ]
}
```

API Requests

Before requesting information from UNIWeb, it is necessary to understand the terminology used to identify pieces of data stored in the system. The information within a UNIWeb page is usually divided into sections, sub-sections, sub-subsections and so on. A *section* contains a list of *items*. An item within a section is made out *fields*. An API *request* is the mechanism for obtaining the *field values* of all items within a section.

Resource Paths

In UNIWeb, a resource is always associated to a type of *content*. Current content types are: 'members', 'units' and 'groups'.

To request a resource, it is necessary to provide a path to it within UNIWeb. A *request path* can be specified as a string by separating each element in the path with '/'. The path must have the following form:

`page/section/section/section/...`

`page`	The 'page' where the information is displayed within UNIWeb. For example, 'profile', 'cv' or 'graph'.
`section/...`	Sequence of section, subsection, sub-subsection,... that contain the target set of items to retrieve.

For example, the string `cv/education/degrees` refers to all the items within the section *Degrees*, which is a subsection of the *Education* section in the CV page of UNIWeb *members*.
Optionally, a request path can be specified as a JSON object. In particular, this is needed if one desires to request only a subset of the field values of an item. In this case, the *resource path* can be given as

```
{"page/section/section/section/...": ["field name A", "field name B", ...]}
```

It is also possible to encode the entire path as a JSON object. This is useful when requesting multiple sections under a common parent section or page:

```
"page": {
```

```
    "parent_section":[
      "child_section A",
      "child_section B"
    ]
}
```

The *resource path* above is equivalent to specifying two separate resource paths as strings:

```
[
  "page/parent_section/child_section A",
  "page/parent_section/child_section B"
]
```

to continue

Naming Conventions

The names of sections and fields used by the API are derived from the **English titles** of their respective sections and fields shown in the UNIWeb UI. Spaces, slashes and question marks are not allowed in resource names. In addition, resource names are always lowercased. To "normalize" a string to meet API rules, do the following:

- Lowercase the given string
- Replace the substrings " / ", "/", and " " with "_"
- Replace the substrings "?" with the empty string ""

For example, the string "Postal / Zip Code" is normalized to "postal_zip_code".

JSON object

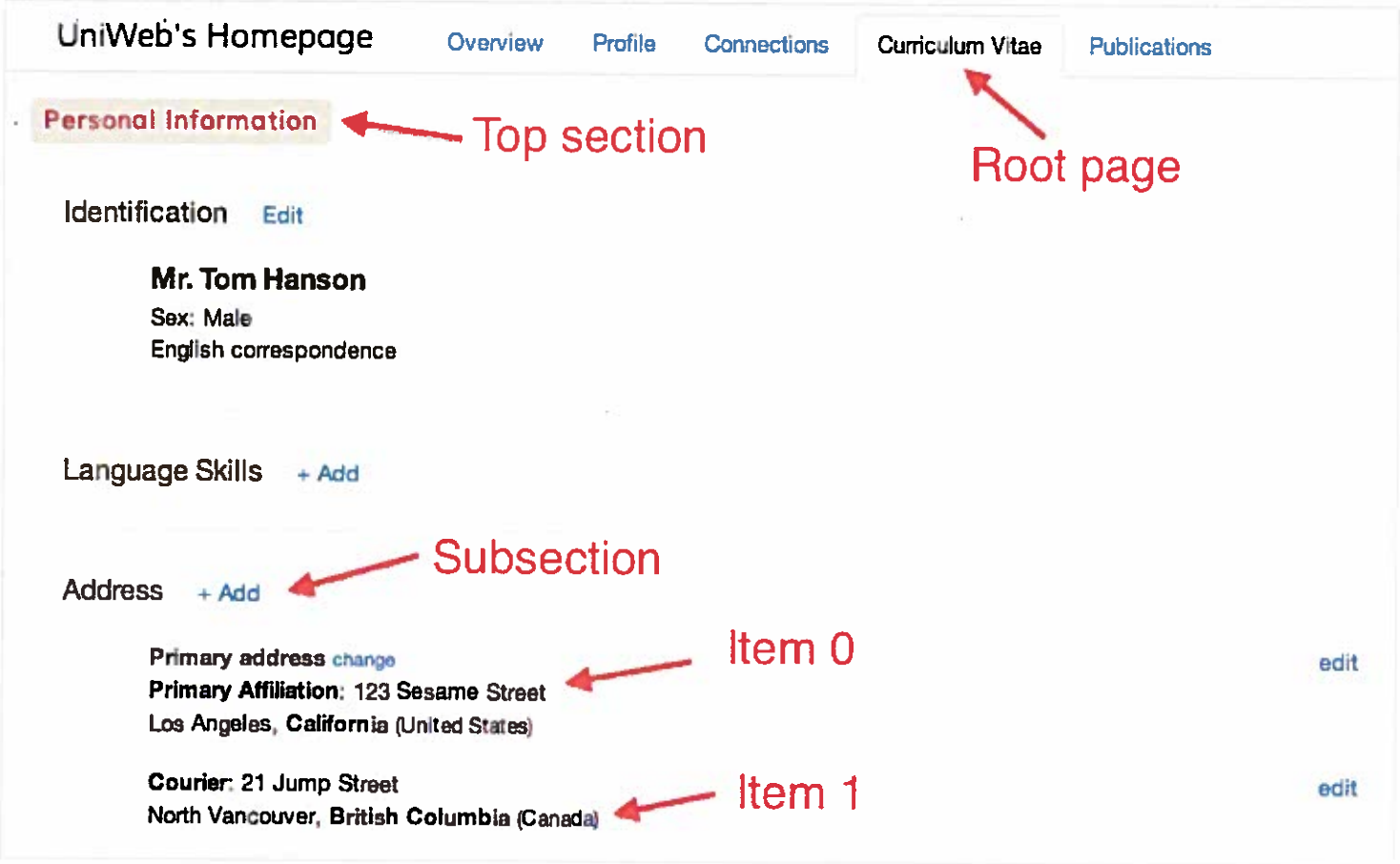
Section Names

The names of sections in resource paths must: (1) correspond to the sections names shown in the UNIWeb UI, and (2) be normalized according to the API naming rules described above. For example, the path to the Address resource in a CV is written as

`"cv/personal_information/address"`

To request objects to UniWeb they must have the following properties:

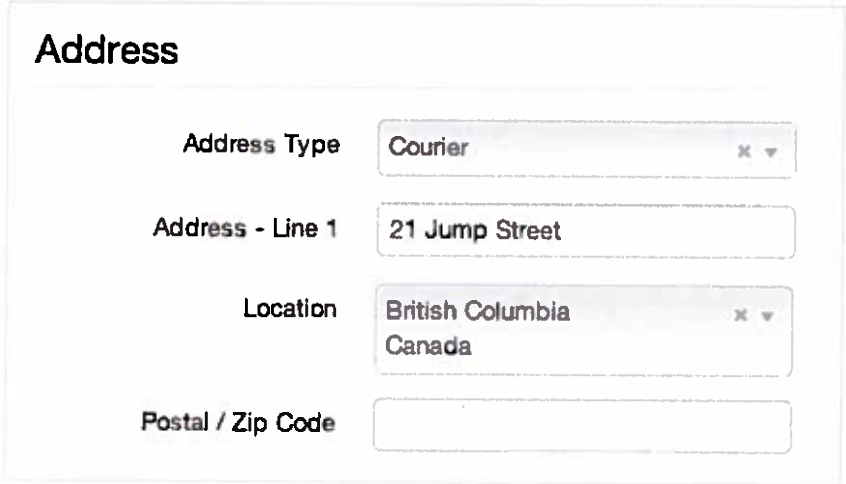
"Read"	ACTION	(required)
"members"	CONTENT	(required)
"bobpmail.cc"	ID	OPT.
	FILTER	OPT
"profile/Biography"	RESOURCE	(required)
"profile/selected_degrees"		



Fields Names

The names of fields in resource paths must: (1) correspond to the field labels in the UNIWeb UI, and (2) be normalized according to the API naming rules described above. For example, the fields shown below in the Address section can be requested as

```
["address_type", "address_-_line_1", "city", "postal_zip_code"]
```



Tip

In the response to an API request, a section item is encoded as a map of field names to field values.

Stucturing Requests

API requests are given as JSON objects with one or more of the following properties.

`token`	required	The hashed value returned by the authorization server.
`action`	required	A string value specifying the desired action to take. Show options
`content`	required	Selects the type of content to retrieve. Show options
`id`	optional	The ID of one particular content to request. Show example
`resource`	required	One or more paths to the requested resources. Show options
`filter`	optional	An object value with filering settings. Show options
`index_by`	optional	Selects how the response indexed the resources in the answer. Show options
`language`	optional	Responses use the default institution's language unless specified otherwise. Show options

Tip

When requesting items of a section, always expect to receive an array of items even if the target section cannot have multiple items. For example, the `Identification` section in a CV can only have a single item given that it lacks a +Add link (i.e., it has an Edit link instead). Regardless, the response to the resource `"cv/personal_information/identification"` will be an array of items of length one.

Example Requests

Simple Single Resource Read Request

The request that follows would return the public profile information of all people in the Department of Civil Engineering as JSON.

```
{
    "token": "access token",
    "action": "read",
    "content": "members",
    "filter": {
        "unit": "Civil Engineering"
    },
    "resource": "profile"
}
```

Requesting to Read Multiple Resources in a Single Request

The request that follows would return two resources belonging to the user with login name `john@smith.ca`, which include:

- the publicly available research interest tags found on his Profile
- the Degree Name, Specialization, and Thesis Title fields from his CV found under Education > Degrees

```
{
    "token": "access token",
    "action": "read",
    "content": "members",
    "language": "fr",
    "filter": {
        "unit": "McGill",
        "title": "Professor",
        "login": "john@smith.ca"
    },
    "resource": [
        "profile/research_interests",
        {
            "cv/education/degrees": [
                "degree_name",
                "specialization",
                "thesis_title"
            ]
        }
    ]
}
```

Error Messages

Errors will give information about what went wrong with a corresponding request. They will be of the following form:

```
{
    "error": {
        "message": "Error validating access token.",
        "type": "OAuthException",
        "code": 98,
        "error_subcode": 223
    }
}
```

Tip

If you make a request that would include in the response a resource that you do not have access to, the whole request will fail. For example, if you request the CV's of several researchers but some of them belong to a unit to which you do not have permissions to view, the request will fail.

UNIWeb API Grammar: Write Actions

References

CCV Schema: https://ccv-cvc-admin.ca/report/schema/doc-en.html GET IT

JSON Request

REQUEST: A **map** of string keywords to values. Mandatory keywords are: action, content and data.

```
{
    "action": ACTION,
    "content": CONTENT,
    "resources": RESOURCES
}
```

ACTION: A **string** keyword: add, or clear.

CONTENT: A **string** keyword: members, units, or groups.

RESOURCES: A **map** of CONTENT_ID to ITEMS.

```
{
    CONTENT_ID: RESOURCES,
    CONTENT_ID: RESOURCES,
    ...
}
```

CONTENT_ID: An **integer** specifying the ID of the content that owns the resource. E.g., a member ID.

ITEMS: A **map** of RESOURCE_PATH to ITEM_LIST.

RESOURCE_PATH: A **string** with the path to the resource to which the data will be written.

ITEM_LIST: An **array** of ITEM objects (for **add** ACTION only).

ITEM: A **map** of FIELD_NAME to FIELD_VALUE.

```
{
    FIELD_NAME: FIELD_VALUE,
    FIELD_NAME: FIELD_VALUE,
    ...
}
```

FIELD_NAME: A **string** with the normalized field name.

FIELD_VALUE: An **integer**, **string**, or **ITEMS**.

UNIWeb API Grammar: Info Actions

JSON Request

REQUEST: A **map** of string keywords to values. Mandatory keywords are: action, content and data.

```
{
    "action": "info",
    "content": CONTENT,
    "resources": resources
}
```

ACTION: A **string** keyword: add, or clear.

CONTENT: A **string** keyword: members, units, or groups.

RESOURCES: A **map** of CONTENT_ID to ITEMS.

```
{
    RESOURCE_PATH: RESOURCE_TYPE,
    RESOURCE_PATH: RESOURCE_TYPE,
    ...
}
```

RESOURCE_INFO: A **map** of properties that uniquely specify the target resource.

```
{
    "type": RESOURCE_TYPE,
    "name": RESOURCE_NAME
}
```

RESOURCE_TYPE: A **string** keyword: section or field.

RESOURCE_NAME: A **string** keyword: section or field.

FINAL

University of Ottawa
Faculty of Medicine

UNIWEB CV

uOttawa

L'Université canadienne
Canada's university

API FUNCTIONAL REQUIREMENTS FOR HOLDING AREA

Version #1.1
October 2015

MEDTECH SERVICE DES TECHNOLOGIES TECHNOLOGY SERVICES

Document History

Table 1: References to Other Documents

Reference Document Name
Physician annual review, September 2014
Uniweb API Grammar: Write and Info actions, July 2015
Internal documentation supports, from several dates

Table 2: Contributors

MedTech		
Name	**Position**	**eMail**
Edy Lopez	Business Analyst and Project Manager	elopez@uottawa.ca

Contributors		
Name	**Position**	**eMail**
Dr Paul Bragg	Executive Vice-dean	pbragg@uottawa.ca
Jim Cassidy	Manager Business Services	Jcassid2@uottawa.ca

Table 3: Revision History

Date	Author	Version	Change Reference
10/01	EL	1.0	First edition
10/08	EL	1.1	Add new objects in Use Case 1 & 2

Table 4: Approvals*

Dean's Office		
Name	**Position**	**Approval**
Dr Paul Bragg	Executive Vice-dean	

MedTech		
Name	**Position**	**Approval**
Jim Cassidy	Business Services Manager	-
-	-	-

* Approvals can be emails indicating acceptance of requirements

TABLE OF CONTENTS

1 EXECUTIVE OVERVIEW

Background:

The secure API at uOttawa allows to integrate UniWeb with other systems within the organization and provides the authentication mechanism to read and write information stored by UniWeb only to the users with granted permission to access the API.

The Faculty of Medicine's goal is to provide the Faculty members of this powerful tool so they can benefit of:

- A central CV platform that can access for both internal and external sources of information
- A collaborative tool which allows to work with other colleagues directly within Uniweb
- A tool that simplifies and reduces the administrative workload of the physician's professional activity.
- A tool that can change their role from adding data to keep CV updated to be a controller of which data goes in.

Current issues

Current Faculty member suffers from the following:

- Inefficient and extensively time-consuming manual processes that means duplication of efforts
- Lack of a consistent and standard way of managing the curriculum professional requirements.
- More efforts required to update the professional CV than to assure the completeness and timely report of the information required.

Proposed benefit

To achieve the full benefits of UniWeb the Faculty of Medicine supports the idea to have a central holding area within UniWeb that can integrate the Faculty member information interacting directly with other Faculty systems importing the data into the Curriculum vitae or by reading the data of the source system.

The meaningful data is hosted in different applications at the Faculty of Medicine or within external professional sources related to the medical profession.

These sources of information are identified and developed in eleven (11) Use Cases with the API functional requirements that UniWeb needs to integrate the information to the central Holding Area.

2. API FUNCTIONAL REQUIREMENTS FOR A HOLDING AREA IN UNIWEB

USE CASE 1: UGME TEACHING

Application	Stipends
Authentication (*)	Token | OAuth 2.0
Data Format (*)	JSON
URL source	https://domain.com/api/users/{user_id}
Parameters	User ID (required)
URL destination	https://uniweb.uottawa.ca
Data	User object with the following attributes: - Title of Activity: Ex.: Week 9 Asthma & Chronic Rhinosinusitis (Mandatory - Curriculum activity: Unit 1 (Year 1 - Department: UGME - Teaching type: Lecturer - Period 3/4/2015 – 3/4/2015
Expectations	UGME Teaching Activity must be updated into eCV Refer to the eCV section / page / field that would help - Path: Curriculum / Activities / Teaching activities / Courses taught / +Add fields - Title of activity → Course title (uniweb) - Curriculum activity → Course code (uniweb) - Department → Department (uniweb) - Teaching type → Role (uniweb) - Period → Start date / End data (uniweb)
Frequency	Data import should be done on a weekly basis
Notes	(*) UniWeb API specification

USE CASE 2: PGME TEACHING

Application	Stipends
Authentication	Token | OAuth 2.0
Data Format	JSON
URL	https://domain.com/api/users/{user_id}
Parameters	User ID (required)
URL destination	https://uniweb.uottawa.ca
Data	User object with the following attributes: - Title of Activity: Ex.: Tutorial – Failure to thrive - Curriculum activity: Pediatrics - Department: General Pediatrics - Teaching type: Tutor / instructor - Period 5/6/2015 – 5/6/2015
Expectations	PGME Teaching Activity must be updated into eCV Refer to the eCV section / page / field that would help - Path: Curriculum / Activities / Teaching activities / Courses taught / - + Add fields - Title of activity → Course title (uniweb) - Curriculum activity → Course code (uniweb) - Department → Department (uniweb) - Teaching type → Role (uniweb) - Period → Start date / End data (uniweb)
Frequency	Data import should be done on a weekly basis

USE CASE 3: MOCOMP FACULTY DEVELOPMENT TAKEN

Application	Mainport – Royal College portal
Authentication	Token | OAuth 2.0
Data Format	JSON
URL source	http://www.royalcollege.ca/portal/page/portal/rc/members/moc /{user_id}
URL destination	https://uniweb.uottawa.ca
Parameters	User ID (member required)
Data	User object with the following attributes: - Program start date: MM/DD/YEAR - Program end date: MM/DD/YEAR - Program title: Add the name of the program - Number of credits Total # number of credits granted
Expectations	Potential MOCOMP learning activity must be updated into eCV Refer to the eCV section / page / For example: - Program start date → curriculum vitae / activities / participation activities / event participation **/ start date** - Program end date → curriculum vitae / activities / participation activities / event participation **/ end date** - Program title → curriculum vitae / activities / participation activities / event participation **/ Event name** - Number of credits → curriculum vitae / activities / participation activities / event participation **/ activity description**
Frequency	Data import should be done on a monthly basis
Notes	a) The Royal College has indicated the agreement to exchange data to their holding area for physicians using the Medbiquious data standard. b) Physicians report of learning activities and hours by year is required to maintain of certification via MainPort access point.

USE CASE 4: ACADEMIC APPOINTMENT

Application	Qtips
Authentication	Token | OAuth 2.0
Data Format	JSON
URL source	*http://www.med.uottawa.ca/app/qtips/default.asp/*{user_id}
URL destination	https://uniweb.uottawa.ca
Parameters	User ID (required)
Data	User object with the following attributes: - effective date: MM/DD/YEAR - rank: Associate professor; for example - department: Surgery; for example - career path: Clinician teacher; for example - End_contract_date: MM/DD/YEAR
Expectations	Appointment should the updated in eCV Refer to the eCV section / page / field would help For "*Path*" → curriculum vitae / employment / academic work experience + Add in *Fields*: - effective date → in **start date (in uniweb)** - rank → in **academic rank (in uniweb)** - department → **in department** (uniweb) - career_path → in **position title** (section under discussion with uniweb team) - End contract date → in **end date (in uniweb)** -
Frequency	Data import should be done on a monthly basis
Notes	Object is applicable for Clinical and Non_Clinical appointments

USE CASE 5: ACADEMIC REAPPOINTMENT

Application	Qtips
Authentication	Token | OAuth 2.0
Data Format	JSON
URL source	*http://www.med.uottawa.ca/app/qtips/default.asp*{user_id}
URL destination	https://uniweb.uottawa.ca
Parameters	User ID (required)
Data	User object with the following attributes: - New_date_of_employment: MM/DD/YEAR - rank: Associate professor; for example - department: Psychiatry; for example - career path: Clinician teacher; for example - End_contract_date: MM/DD/YEAR
Expectations	Reappointment should the updated in eCV Refer to the eCV section / subsection / field For "*Path*" → curriculum vitae / employment / academic work experience + Add in *Fields*: - effective date → in **start date (in uniweb)** - rank → in **academic rank (in uniweb)** - department → in **department** (uniweb) - career_path → in **position title** (field not present in uniweb) - End contract date → in **end date (in uniweb)**
Frequency	Data import should be done on a monthly basis
Notes	1. This object is applicable to Clinical and Nonclinical appointments 2. Reappointment means that the faculty contract was renewed and has a new expiration date.

USE CASE 6: ACADEMIC CROSS-APPOINTMENT

Application	Qtips
Authentication	Token | OAuth 2.0
Data Format	JSON
URL source	*http://www.med.uottawa.ca/app/qtips/default.asp*/{user_id}
URL destination	https://uniweb.uottawa.ca
Parameters	User ID (required)
Data	User object with the following attributes: - New_date_of_employment: MM/DD/YEAR - rank: Assistance professor; for example - department: Psychiatry; for example - new career path: Clinician scientist; for example previous Clinician teacher - End_contract_date: MM/DD/YEAR -
Expectations	Cross-Appointment should the updated in eCV Refer to the eCV section / subsection / field For "*Path*" → curriculum vitae / employment / academic work experience + Add in *Fields*: - effective date → in **start date (in uniweb)** - rank → in **academic rank (in uniweb)** - department → in **department** (uniweb) - career_path → in **position title** (field not present in uniweb) - End contract date → in **end date (in uniweb)** -
Frequency	Data import should be done on a monthly basis
Notes	3. This object is applicable to Clinical and Non_Clinical appointments 4. Cross-appointment means that the faculty in the renewed contract is assigned to a different Department that former contract and has a new expiration date.

USE CASE 7: CPD TEACHING

Application	1. Entrada CPD owner is Office Continuous Professional Development 2. BAF_Entrada owner is Office of Francophone Affairs
Authentication	Token | OAuth 2.0
Data Format	JSON
URL source (s)	***https://eventscpd.med.uottawa.ca/{user_id}*** ***https://baf_inscriptions.med.uottawa.ca/{user_id}***
URL destination	https://uniweb.uottawa.ca
Parameters	User ID (required)
Data	User object with the following attributes: - Program date: MM/DD/YEAR - Program name: Add the head name of the program - Presentation name: Add presentation title - Type of presentation: Workshop/Keynotes/abstracts: examples
Expectations	CPD medical teaching is addressed to lifelong learning for physicians, medical educators and scientists and should be updated in eCV Refer to the eCV to Section / subsection /Items / fields / value For example: *Section / subsection path* → curriculum vitae / activities / participation activities / contribution / presentations / + Add in *Fields* - Program date → **presentation date** - Program name → **conference events name** - Presentation name → **presentation title** - - Type of presentation → **presentation type** -
Frequency	Data import should be done on a monthly basis
Notes	a. In these events the Educator can register as a leaner and participate in learning activities included in the program. Faculty can be Educator and a learner at the same time.

USE CASE 8: CPD TAKEN

Application	1. Entrada CPD owner is Office Continuous Professional Development 2. BAF_Entrada owner is Office of Francophone Affairs
Authentication	Token | OAuth 2.0
Data Format	JSON
URL source (s)	***https://eventscpd.med.uottawa.ca/{user_id}*** ***https://baf_inscriptions.med.uottawa.ca/{user_id}***
Parameters	User ID (required)
URL destination	https://uniweb.uottawa.ca
Data	User object with the following attributes: - Program start date: MM/DD/YEAR - Program end date: MM/DD/YEAR - Program title: Add the name of the program - Number of credits Total # number of credits granted - Role: learner
Expectations	Faculty medical learning addressed to lifelong learning for physicians, medical educators and scientists to be updated in eCV Refer to the eCV to Section / subsection /Items / fields / value For example: *Section / subsection path* → Curriculum vitae / activities / participation activities / event participation + Add in *Fields* For example: - Program start date → **start date** - Program end date → **end date** - Program title → **Event name** - Number of credits → **activity description** - Role → **Learner**
Frequency	Data import should be done on a monthly basis
Notes	a) In these events Learner can register as a leaner, or eventually as an Educator can participate as Educator and as a registered Learner.

USE CASE 9: ACADEMIC PROMOTION

Application	Academic Promotions
Authentication	Token | OAuth 2.0
Data Format	JSON
URL source	***http://app.med.uottawa.ca/Academic Promotions/***{user_id}
Parameters	User ID (required)
URL destination	https://uniweb.uottawa.ca
Data	User object with the following attributes: - Department: Name of the department - Division: Name of division (if applicable) - Effective_ date_employment: MM/DD/YEAR - New academic rank Associate professor, for example - Title of the position: Director, for example
Expectations	UGME Teaching Activity must be updated into eCV Refer to the eCV section / subsection / field For "*Path*" → curriculum vitae / employment / academic work experience + Add in *Fields*: - Department: → **department** - - Division → tbd? - - Effective date of employment → **start date** - - New academic rank→ **academic rank** - - Title of position → **position title** -
Frequency	Data import should be done on a yearly basis
Notes	1. Clinical and Non Clinical faculty participate of the academic promotion 1.1 Clinical rank → lecturer, assistance professor, associate professor, regular full time 1.2. Non Clinical rank → associate professor with tenure, associate professor, full professor.

USE CASE 10: DEPARTAMENTAL ACADEMIC POINTS PORTAL

Application	Faculty Department website
Authentication	Token | OAuth 2.0
Data Format	JSON
URL source (s)	***http://www.med.uottawa.ca/psychiatry/eng/{user_id}*** ***http://www.med.uottawa.ca/surgery/eng/{user_id}*** ***http://www.med.uottawa.ca/internalmedicine/residencyprogram/eng/{user_id}*** ***http://med.uottawa.ca/obs-gyne/{user_id}***
Parameters	User ID (required)
URL destination	https://uniweb.uottawa.ca
Data	User object with the following attributes: MORE OBJECTS TO ADD - Hours_supervising_ students: amount of hours of supervision - Presentation perform other schools: quantity of presentation
Expectations	Faculty part-time academic activity updated in eCV Refer to the eCV section / page / field - Hours_supervising_ students → curriculum vitae / activities / supervisory activities / student supervision - Presentation_perform_other_schools → curriculum vitae / activities / participation activities / event participation
Frequency	Data import should be done on a **Quarterly** cycle.
Notes	a) This object is applicable to Clinical faculty exclusively b) Departments participating are initially: Psychiatry / Surgery / Internal Medicine / Obstetrics. c) These four (4) departments are willing Medtech to create an academic point's portal where members can register time of academic activities that represents academic points and financial compensation.

USE CASE 11: PERFORMANCE ANNUAL REVIEW [PAR]

Application	Physician annual review
Authentication	Token | OAuth 2.0
Data Format	JSON
URL source	***https://app.med.uottawa.ca/PhysicianAnnualReview/{user_id}***
Parameters	User ID (required)
URL destination	https://uniweb.uottawa.ca
Data	User object with the attribute to read the record of the Postgrade Medical Education annual performance evaluation; *NOTE: This data needs a different type of handling, could not be relevant to Uniweb but is very relevant to the PAR as well as promotions.*
Expectations	Faculty to read evaluations for period January 1st to December 31st from several years. - It needs to be held in the holding area for review over subsequent years; - Especially if a faculty member is sub-par in performance.
Frequency	Data retrieve should be done on a yearly basis
Notes	Suggested retention of these records to be defined by Executive Vice-dean. The retention policy at the Faculty of Medicine is 8 years (minimum required).