Rapporti Tecnici INAF
INAF Technical Reports

Number	247
Publication Year	2023
Acceptance in OA@INAF	2023-02-06T16:08:22Z
Title	SLab Web Data Manager
Authors	CARRARO, Francesco
Affiliation of first author	IAPS Roma
Handle	http://hdl.handle.net/20.500.12386/33199; https://doi.org/10.20371/INAF/TechRep/247

SLab Web Data Manager

(Part of SLab HYDRA)

Francesco Carraro

1. Overview

1.1) What is SLab Web Data Manager?

The *SLab Web Data Manager* is a tool developed for helping activities in IAPS SLab structure. The web app has the main goal of allowing researchers in retrieving and analyzing spectra saved during the acquisition sessions.

In this user manual it will be explained how the web application works as web API for *SLab Data Manager* app, for managing users' registration, and the describes the beginning of the work for letting users to search and visualize raw and processed spectra into the *SLab database* (At the time this manual is being written, this section of the web app is in initial development phase).

1.2) SLab HYDRA (HYperspectral Data Reduction and Analysis)

SLab Web Data Manager is part of a set of informatic tools consisting of in:

- *SLab Data Manager*: UWP app for processing, analyzing and saving data ([INAF handle](#))
- *SLab Web Data Manager*: web app that is being discussed in this manual
- *SLab DB*: the centralized database containing all the info about registered users and data.

These tools build up an integrated environment for easing and speeding up as much as possible the reduction and the analysis of spectra acquired during activities carried on @ IAPS SLab.

2) The role of the web app

Since the beginning of the *SLab HYDRA* project, the team agreed on the decision of developing both a desktop app and a web app. The main reasons behind that idea were essentially two: the first was the technical requisite of having a [web API](#) working as interface between the *SLab Data Manager* app and the *SLab DB.* One of the best practices in developing a database is avoiding, when not strictly necessary, exposing it on the web. This ensure a better protection and prevent a huge number of attacks from malicious bot; the second reason was the will of creating a cross-platform tool in order to make it simpler for scientists to use it. Nothing is more cross-platform than web tools.

Figure 1 Web app concept

As stated at the beginning of this chapter, in the modern conception of a web service, developing a service that serves only its own front-end makes no sense. So, developing services that work for both mobile/desktop apps and web apps is a common practice.

In this framework, the *SLab Web Data Manager* has been designed with the dual role of

1. working as a back-end for all features provided by the desktop app
2. providing its own front-end of the same services and offering extra capabilities.

At the time this manual is being written, anyway, there's a pronounced difference between desktop app and web app capabilities.

The desktop app provides a set of features for checking whether spectra already exist on cloud or not, processing and then saving/deleting them from the database, suggesting values for sample mineral composition while lacks support for searching already saved spectra.

The web app interface provides a basic management of users, supports search of suggestions related to sample mineral composition, searching for spectra and visualizing them feature is available but is under development. There's still no support for processing spectra.

N.B. All desktop app capabilities here described rely on web app services.

3) Login

3.1) Why login is required?

SLab Web Data Manager, so far, allows users to save some information about spectra into the *SLab DB* and, in the next updates, capability of processing spectra is going to be added. When any info is saved into the DB, an ID related to the user is recorded. This means the user must be logged into the system and an ID has already been created. Said this, inserting valid credential is required for starting using the app.

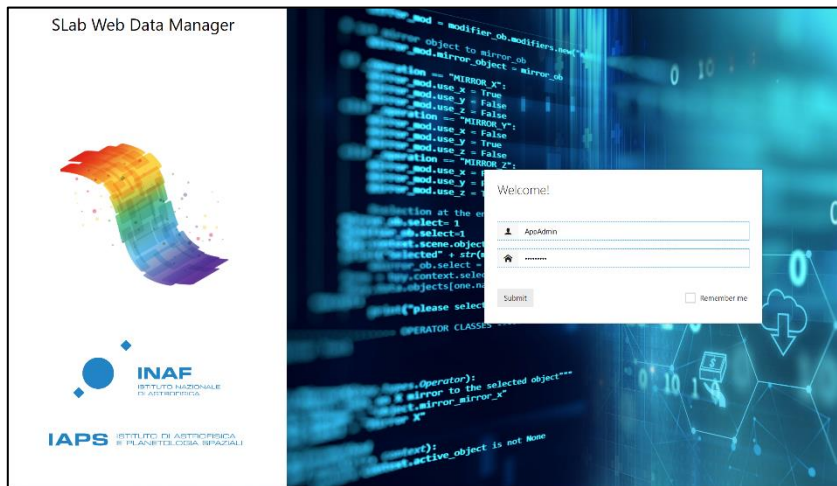Following, the provisory login screen is shown.

Figure 2 Provisory login of the web app

3.2) How obtain a login

At the time this manual is being written, valid credentials can be only obtained by contacting the administrator of the *SLab HYDRA* system: francesco.carraro@inaf.it. As mentioned at the beginning of this manual, anyway, the web application is still under development and will, in the future, allow the self-registration of users by means of the ORCID ID.

N.B. Being both app and web app part of the same integrated system, as stated before, credentials are the valid everywhere.

3.3) Login management

Managing login for both desktop app and web app means being able first of all to recognize the caller who is asking for being authorized to access web services, then deal with two different ways of providing credentials.

The standard way for a web app or web site for saving credentials provided to the back end in each request implies the usage of **cookies**. "***HTTP cookies** (also called **web cookies**, **Internet cookies**, **browser cookies**, or simply **cookies**) are small blocks of data created by a web server while a user is browsing a website and placed on the user's computer or other device by the user's web browser. Cookies are placed on the device used to access a website, and more than one cookie may be placed on a user's device during a session.*" (from Wikipedia).

A widely used way for managing credentials when dealing with mobile apps, on the other hand, is using a token, called **bearer token**: "*A security token with the property that any party in possession of the token (a "bearer") can use the token in any way that any other party in possession of it can. Using a bearer token does not require a bearer to prove possession of cryptographic key material (proof-of-possession).*" (from rfc-editor.org).

Providing the server with the capability of recognizing the user who is sending the request is critical for two main reasons:

1. when a request is received, the server must be able to authenticate the user who is sending the request. This means verifying whether credentials exist or not into the database. If credential exist, a check must be executed to verify whether the login has expired or not. If credentials are not valid, the user has no more valid access to the data and incoming requests must be rejected.
2. Since each row in the database tables, related to a processed spectrum, is marked with the user ID, only data belonging to the user who is sending the request must be retrieved.

To accomplish this crucial task, each request is analyzed by first looking for credentials coming from cookies. If not found, the bearer token is searched among the request headers.

If credentials are found, then the validation process is completed as previously explained, otherwise the request is rejected.

3.4) User permissions

The *SLab Hydra* has been designed providing a wide set of features, each linked to permission, and a set of authorization levels. Each level is related to a sub-set of permissions. When a user is created in the system, an authorization level is assigned which grants a set of actions (permissions) to the user.

Currently three levels are available:

- *Admin*: all permissions are granted
- *User*: exploring the DB, processing and saving data are granted
- *Guest*: only exploring the DB and some processing levels are granted.

4) User Interface

4.1) Responsive UI

"**Responsive web design** (**RWD**) or **responsive design** is an approach to web design that aims to make web pages render well on a variety of devices and window or screen sizes from minimum to maximum display size to ensure usability and satisfaction" (from Wikipedia)

SLab Web Data Manager adhere to responsive design principles for being usable on the wide range of screen resolutions of modern devices. Following, in Figure 3 an example of the same interface is shown respectively as visualized on a desktop computer (or wide screen device) and on a mobile device.
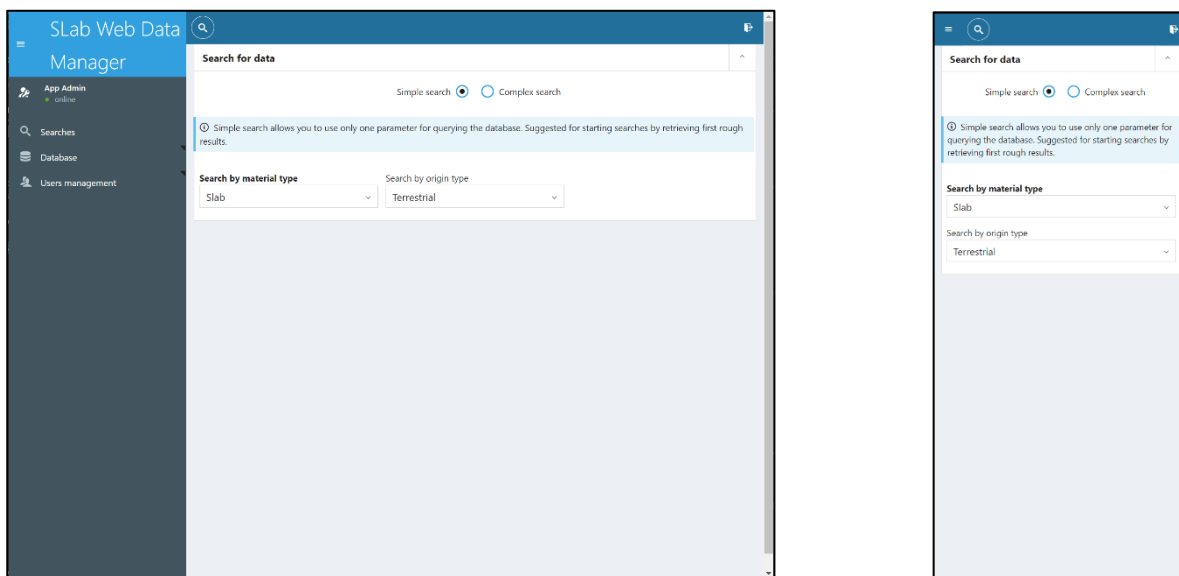

Figure 3 Search page: desktop version on the left, mobile version on the right.

4.2) Dynamic menu

As stated in 3.4) User permissions each user is assigned a role that corresponds to a subset of permissions among the whole list of available ones. When using the web app, the assigned list of permissions is *transformed* into the left-side menu. The menu is the main instrument for navigating available pages. This means that users having different levels of authorization, thus having a different subset of permissions, have different menu.

To better explain this feature, let's consider the menu shown in Figure 3, in the left-side image. This is the menu available for the app admin, who is granted with all existing permissions. If to connected user were assigned *user* level, only the *Searches* item would be visible.

At the time this document is being written, the left-side menu is the only section affected by permissions. Anyway, when other main features will be added, such as capability of processing spectra or capability of saving/downloading spectra, other UI components (i.e.: buttons) will be available depending on authorization level of current user.

5) Pages

The *SLab Web Data Manager* hosts several pages, each aimed to let user accomplishing a set of coherent tasks, i.e.: managing users, managing available minerals used for saving mineral samples composition, etc. Excluding the left-side menu, which has *its own life*, two main components are available for interacting with user, visible in Figure 3:

1. The *top bar* (the horizontal blue bar containing the buttons)
2. The *main body* (the white/grey section)

The *top bar* is the place where the main actions buttons are usually placed. This makes the bar a dynamic section where buttons are added/removed depending on the current page. Also, for some pages, different set of buttons are available depending on the current sub task. An example of this behavior is demonstrated in 5.1.2) Exploring results, where searches are explained.

The *main body* is the place where the content is shown, and the most part of interactions happen.

The above-mentioned structure has been used throughout all the pages, to create a coherent user experience.

5.1) Searches page

The default entry point for the *SLab Web Data Manager* for *user* and *guest* roles is the *Searches* page, shown in Figure 3.

This page is meant to be the place where users search for spectra by using some kinds of filters, retrieve and visualize them and, if allowed by their authorization level, execute some processing and analysis.

5.1.1) Searching for spectra

Two levels of search are available:

1. Simple search
2. Complex search

The *Simple search* has been imagined as the starting point for new searches, a way to explore the dataset of available spectra during the initial phase of the work. The *Complex search* has been conceived to be the instrument for refining searches by adding filters and complete a fine tuning of previous rough results.

N.B. At the time this manual is being written, only *Simple search* is available.

5.1.2) Exploring results

Once search parameters have been set and the search is executed, a list of results is provided below the panel containing the filters. An example of results is shown in Figure 4.

Results are presented as a series of expandable rows, each showing basic information about the found spectrum. Information showed in the current version of the web app are the following ones:

- Name of the file
- Number of available spectra corresponding to the name (each spectrum corresponds to a level of processing, including the raw one)
- The kind of input raw spectrum (single acquisition or averaged)

By expanding the item, a list is presented which shows more detailed info about each spectrum.

If the spectrum is selected, by using the checkbox, it is added to the list of items ready to be shown in the charts panel. When one or more spectra are selected for being charted, a button is added in the top bar for showing the chart, see Figure 4 below.
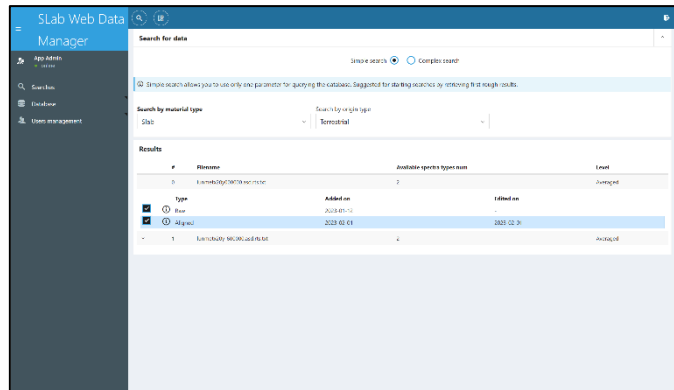

Figure 4 Example of results from a Simple search. Two spectra have been selected and the Show chart button is added on top bar

If the info icon is tapped, then a panel containing saved information about sample and measurement geometry is presented, as shown below in Figure 5. This panel, other than visualizing info, allows the user to edit information.
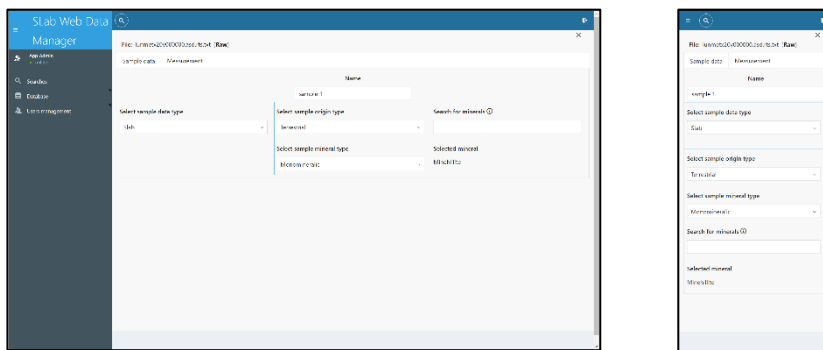

Figure 5 Example of information about the sample in both desktop and mobile version.

N.B. While designing UI of both web app and desktop app, where features overlaps (as in case of sample and measurement info panel), particular attention has been paid to make user experience as much similar as possible. Compare Figure 5 (desktop version) *and Figure 6*.
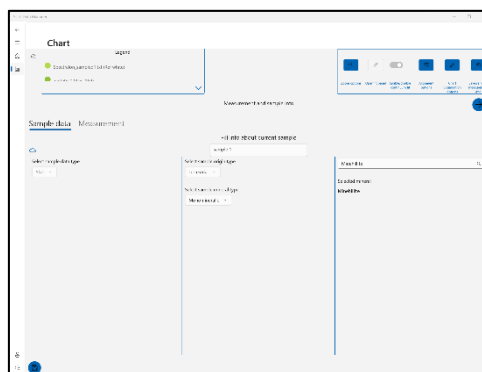

Figure 6 Sample/measurement info in desktop app

5.1.3) Charts

Charts visualization as long as the searching feature is currently under development. This means only a basic functionality is provided. Spectra selected in the previously described panel are charted as shown in