**Basic User Guide**

**Introduction**

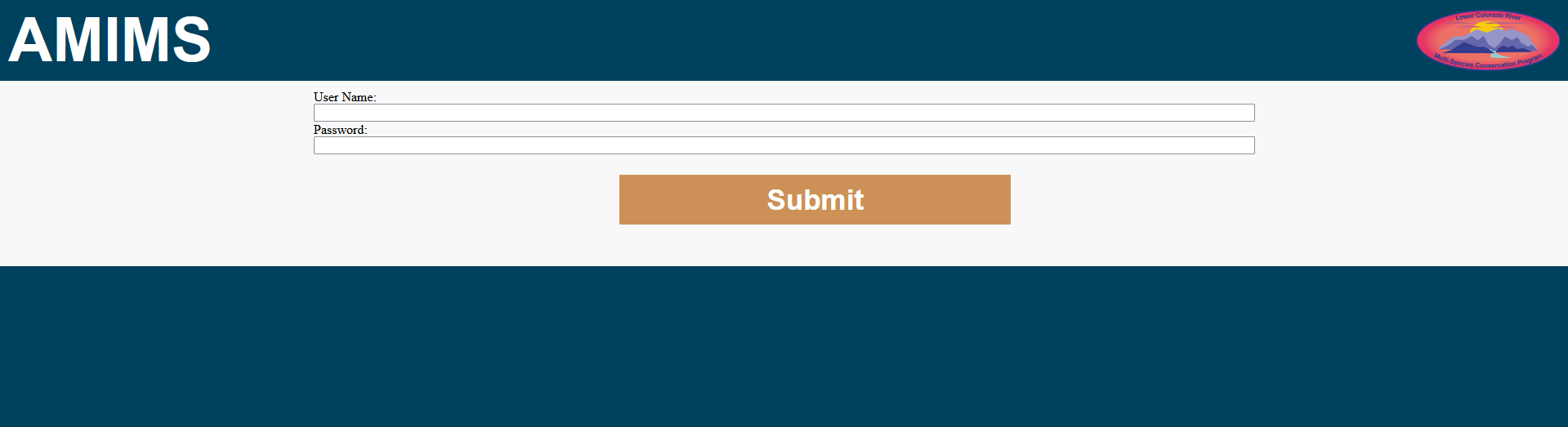
The Adaptive Management Information Management System (AMIMS) is a web application for the management of Multi-Species Conservation Program Adaptive Management Plans. The following guide will show how to navigate the website and use its features.

**Landing Page**

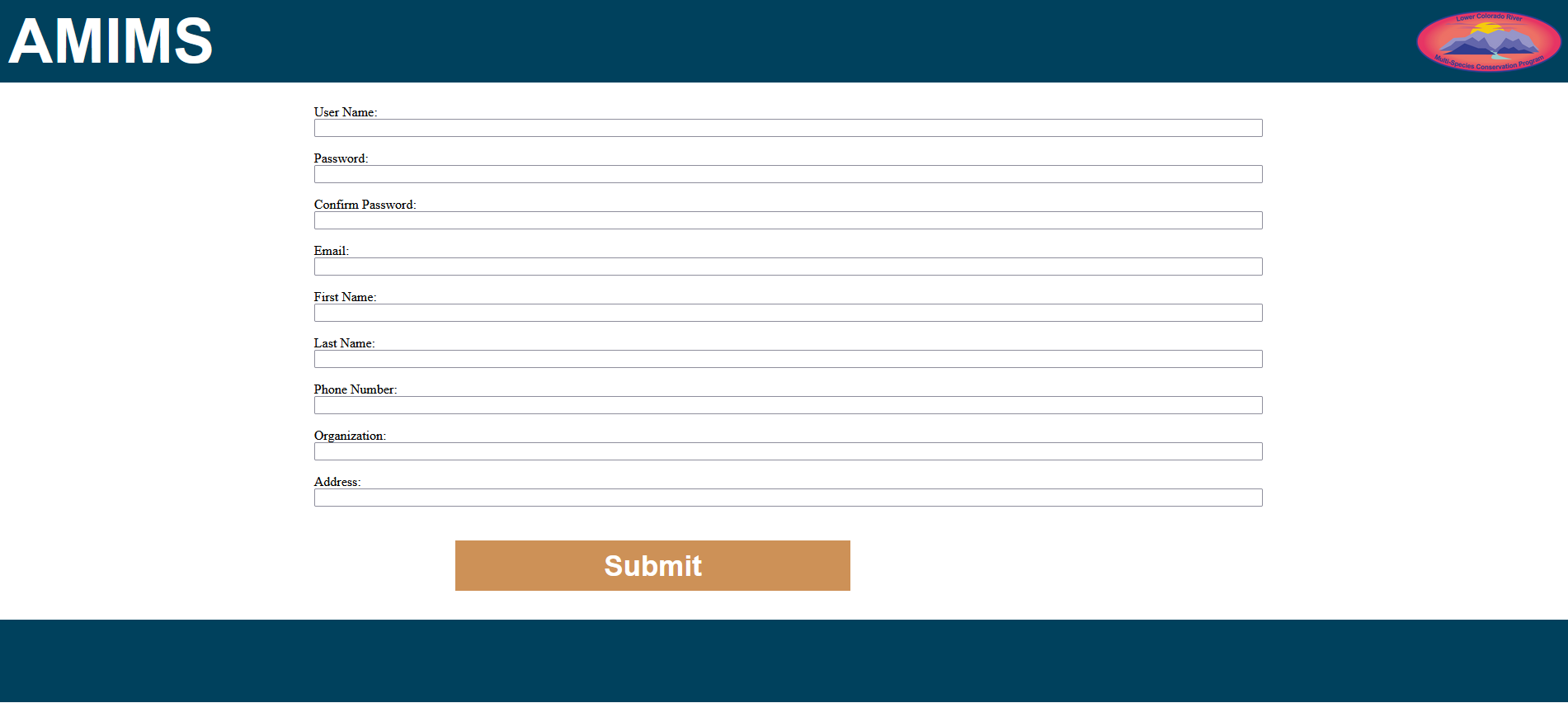
When navigating to AMIMS the first thing you see is the landing page.

Fig. 1: AMIMS Landing page

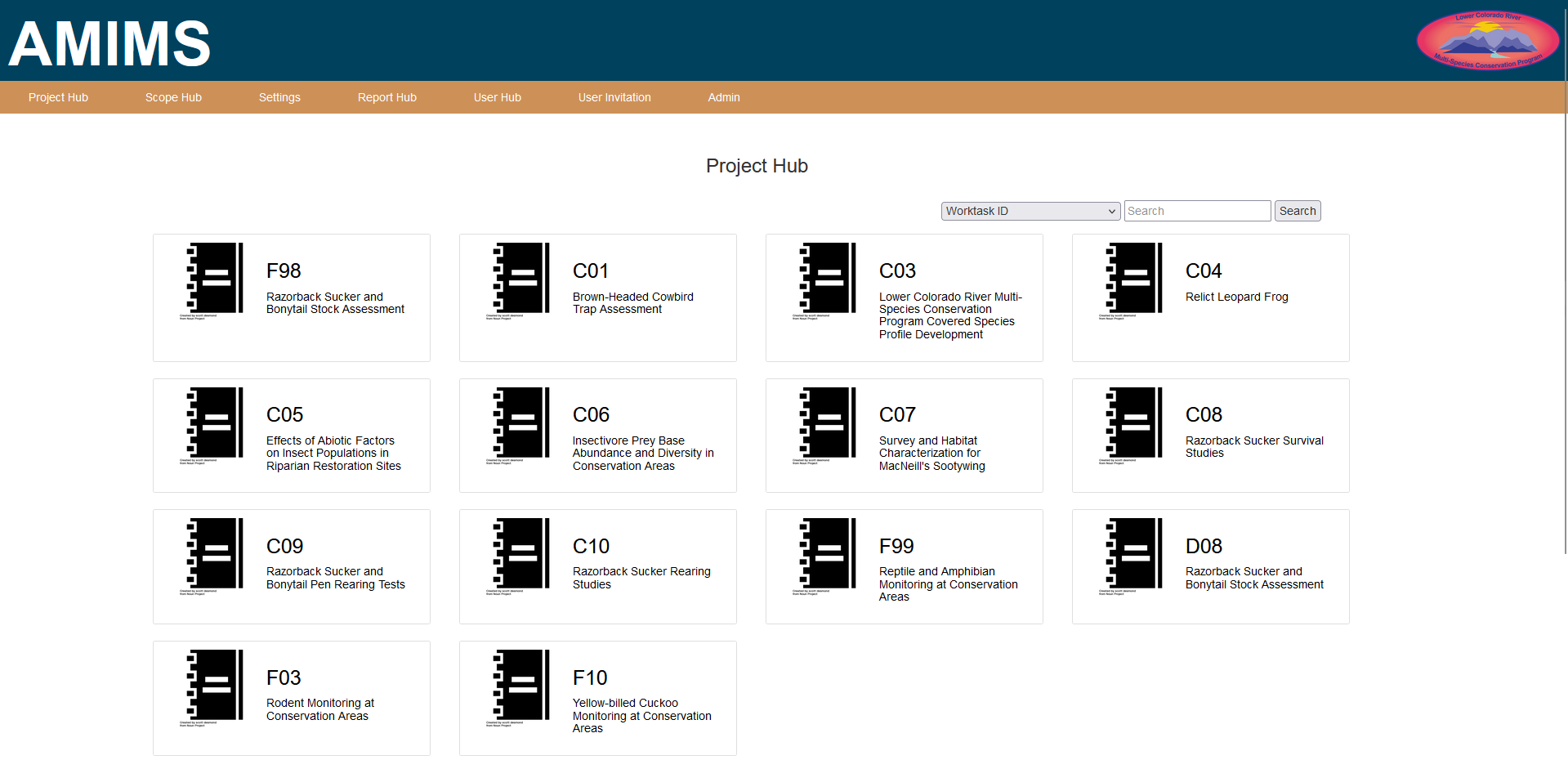
From here the login button will direct the user to one of two pages depending on their account status. Existing users will be directed to the login page where they can enter their username and password to be directed to the AMIMS Hub.

Fig. 2: AMIMS Login page

New users will be directed to the user registration page. This page allows new users to complete the creation of their profiles before being directed to the AMIMS Hub.

Fig. 3: AMIMS User Creation page

**Project Hub**

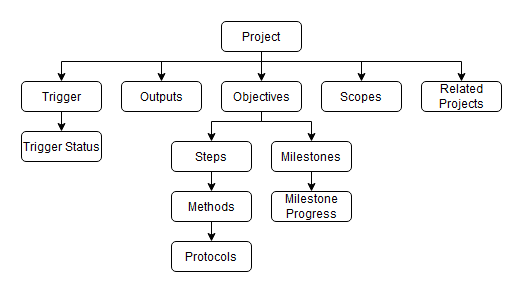
Fig. 4: AMIMS Project Hub

The Project Hub is where projects are accessed and created. All of the Projects in AMIMS have a card in the hub. Projects are accessed by clicking on their respective Project card. Contributor level users and Admins can also create projects by clicking on the “Add New Project” button. Pressing this button will create a blank project and redirect you to the new Project page. Only Admins have the authority to delete Projects, so be sure to only create projects when necessary.

The Hub also has search functionality. From the search bar select a field from the drop down and enter search parameters into the text box. The search will filter the project page to only include Projects that contain the search text in the selected field.

**Projects**

The Project is the fundamental component of AMIMS, all major components interact with or support Projects. Projects are organized to be analogous to MSCP worktasks. All project objects start with a text form that contains the information for the object. Underneath this is a section that contains all of the child objects that the project object can have. The work task hierarchy is as follows.

Fig. 5: Project hierarchy diagram

After an object has been created it is available to be edited. In order to prevent accidental changes object pages are locked by default. To unlock a page click on the lock icon at the top of the page. The unlocking the page also enables the various “Add Object” buttons.

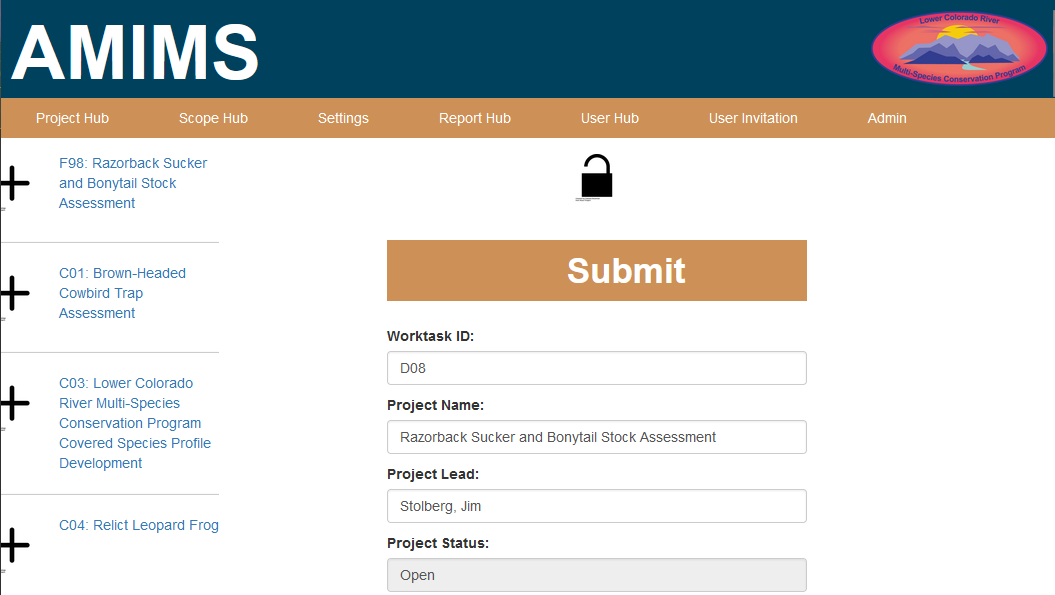


Fig. 6: An AMIMS Project with the fields unlocked

Once the page is unlocked the text boxes are free to edit. To create a new sub-object press the relevant “Add Object” button. This will create a single empty object of the selected type, and redirect you to the new empty object.

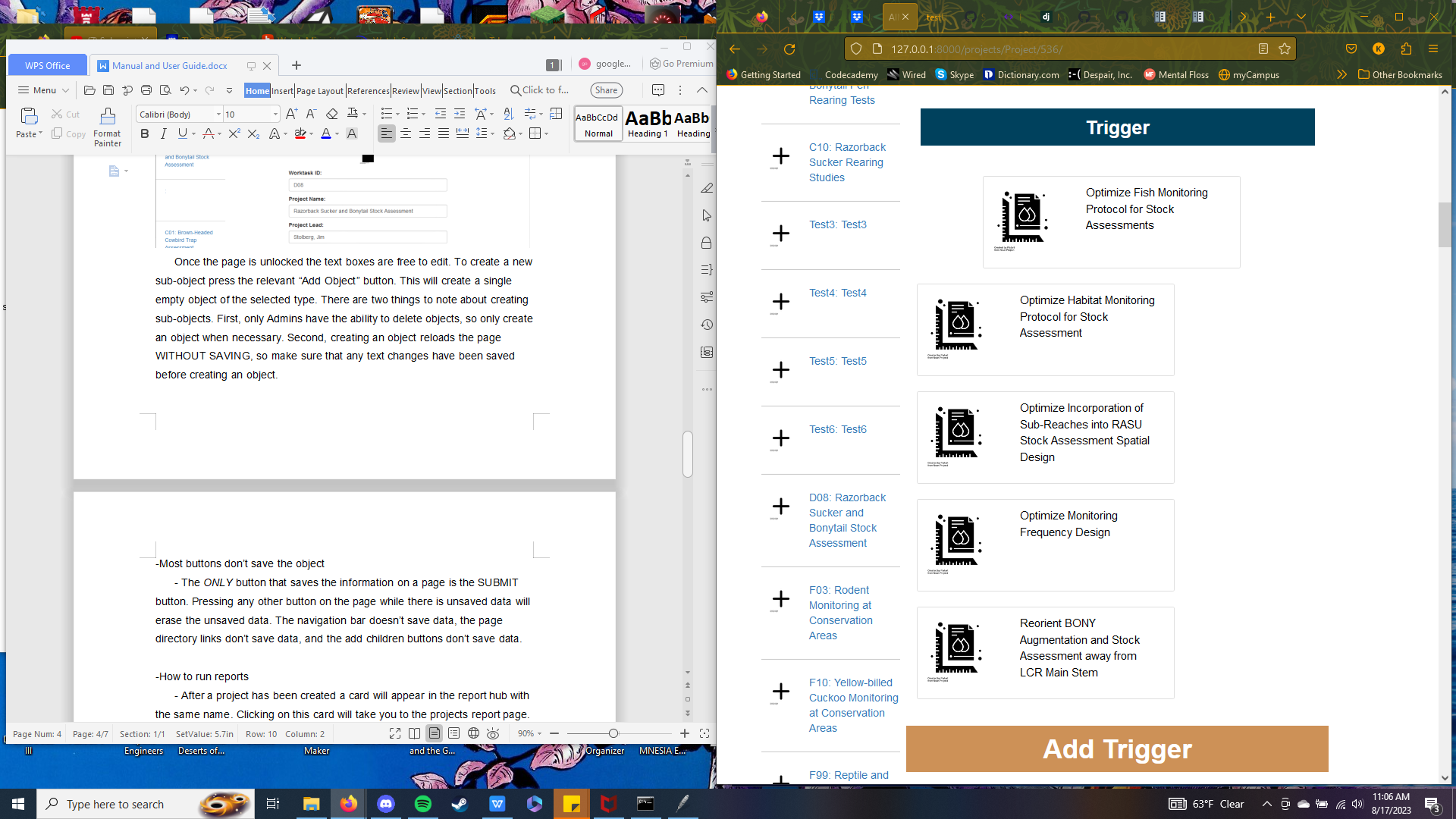


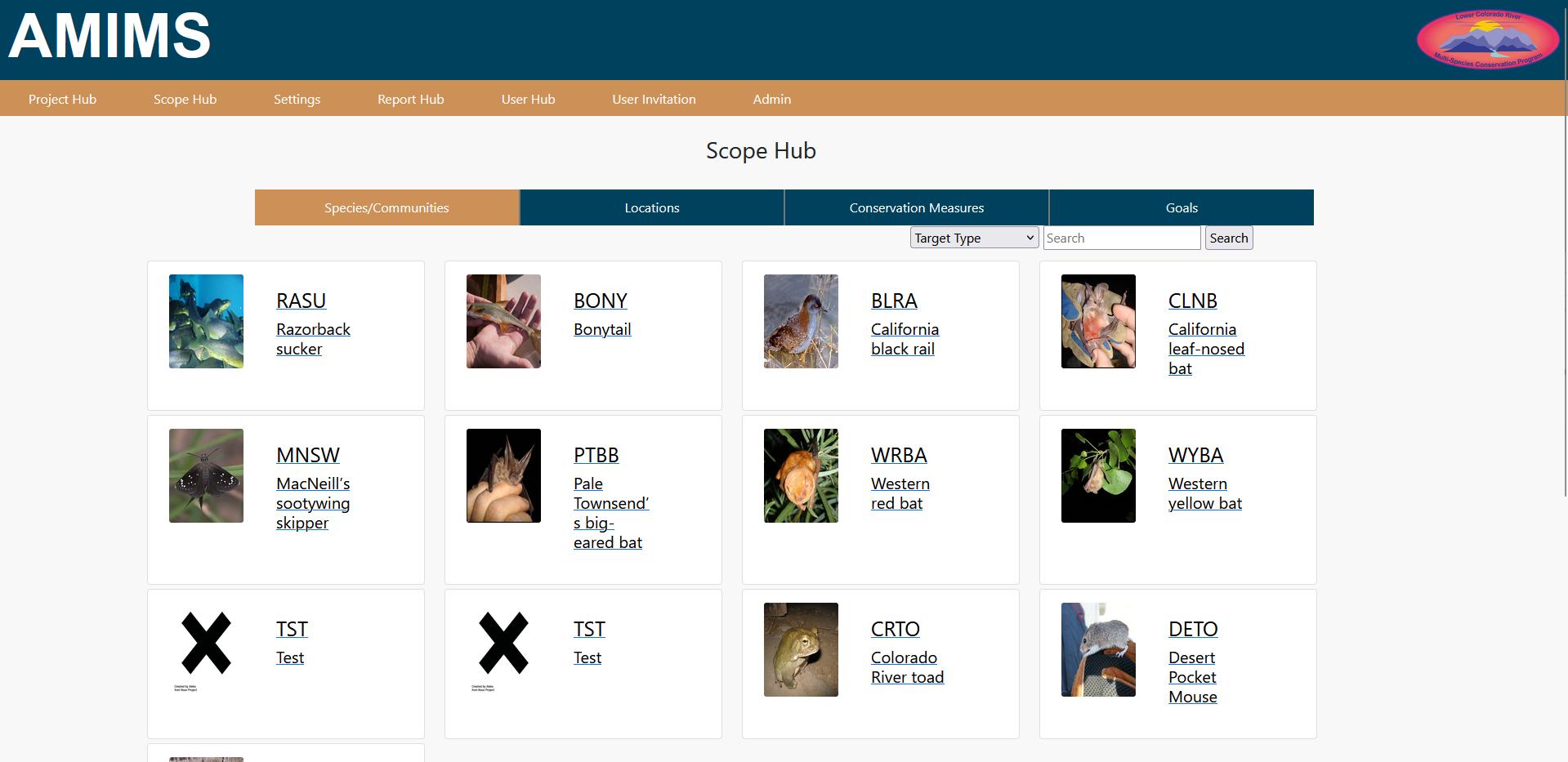
Fig. 7: An “Add Object” button

There are two things to note about creating sub-objects. First, only Admins have the ability to delete objects, so only create an object when necessary. Second, creating an object reloads the page WITHOUT SAVING, so make sure that any text changes have been saved before creating an object.

The *ONLY* button that saves the information on a page is the SUBMIT button. Pressing any other button on the page while there is unsaved data will erase the unsaved data. The navigation bar doesn’t save data, the page directory links don’t save data, and the add children buttons don’t save data.

**Scope Hub**

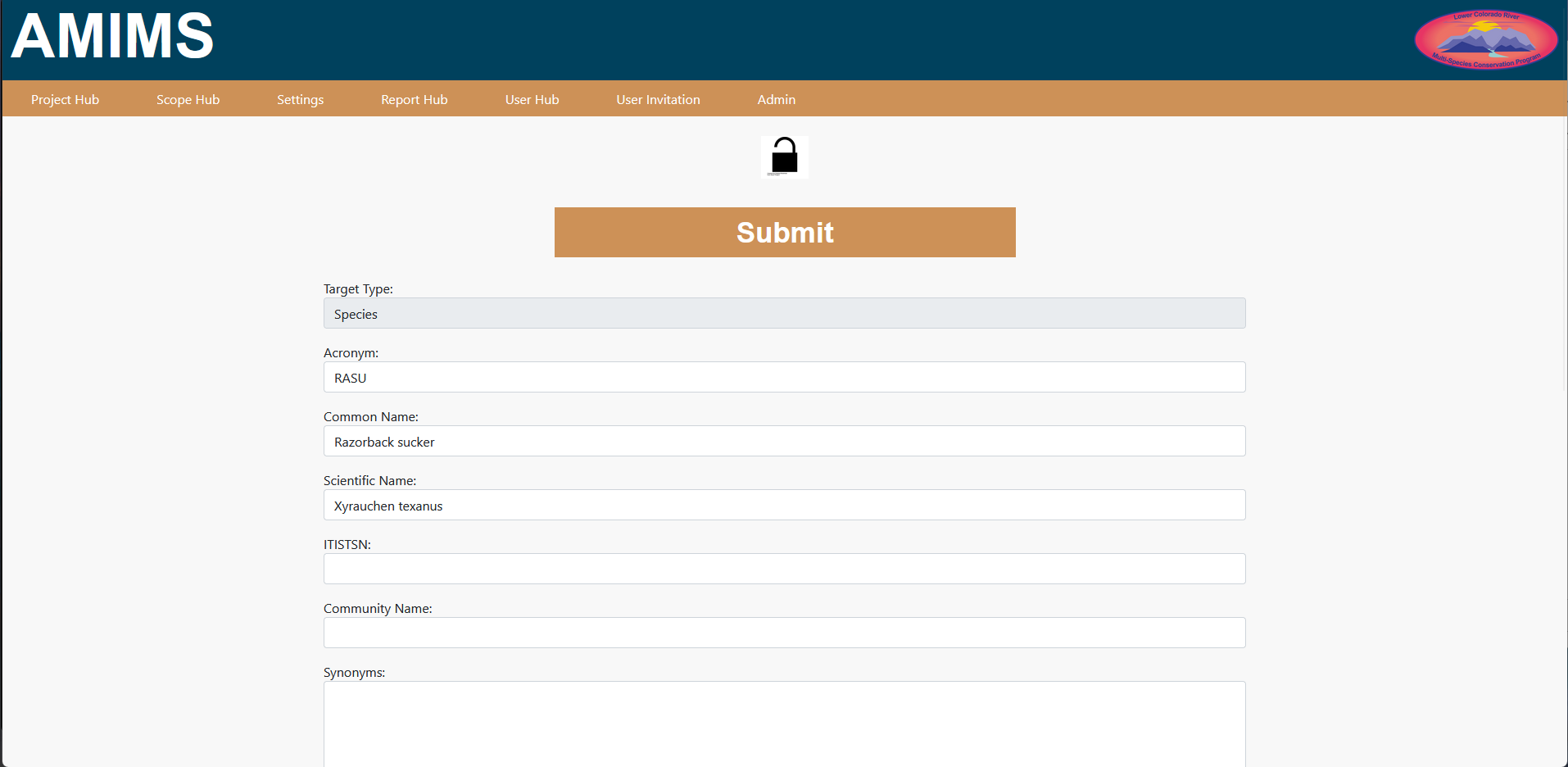
The Scope Hub is similar to the Project Hub in that it is where scopes are organized and created. Like the Project Hub clicking on a Scope card will direct you to the associated Scopes page and clicking one of the four “Add New” Scope buttons will create a relevant blank scope. Unlike the Project Hub the Scope Hub is separated into four parts, one for each type of Scope, Species/Communities, Locations, Conservation Measures, and Goals. At the top of the Scope Hub page, underneath the page title, is a navigation bar that lists the four types of Scopes. Clicking on one of these will filter the Hub to only include Scopes of that type, i.e. clicking on Species/Communities will filter the Hub to show only Species and Community Scopes.

Fig. 8: AMIMS Scope Hub

Also like the Project Hub each scope has its own search bar. Like with the Project Hub, to search a section of the Scope Hub select a field from the drop down and enter search text into the search box. The Scopes will then be filtered based on the search criteria.

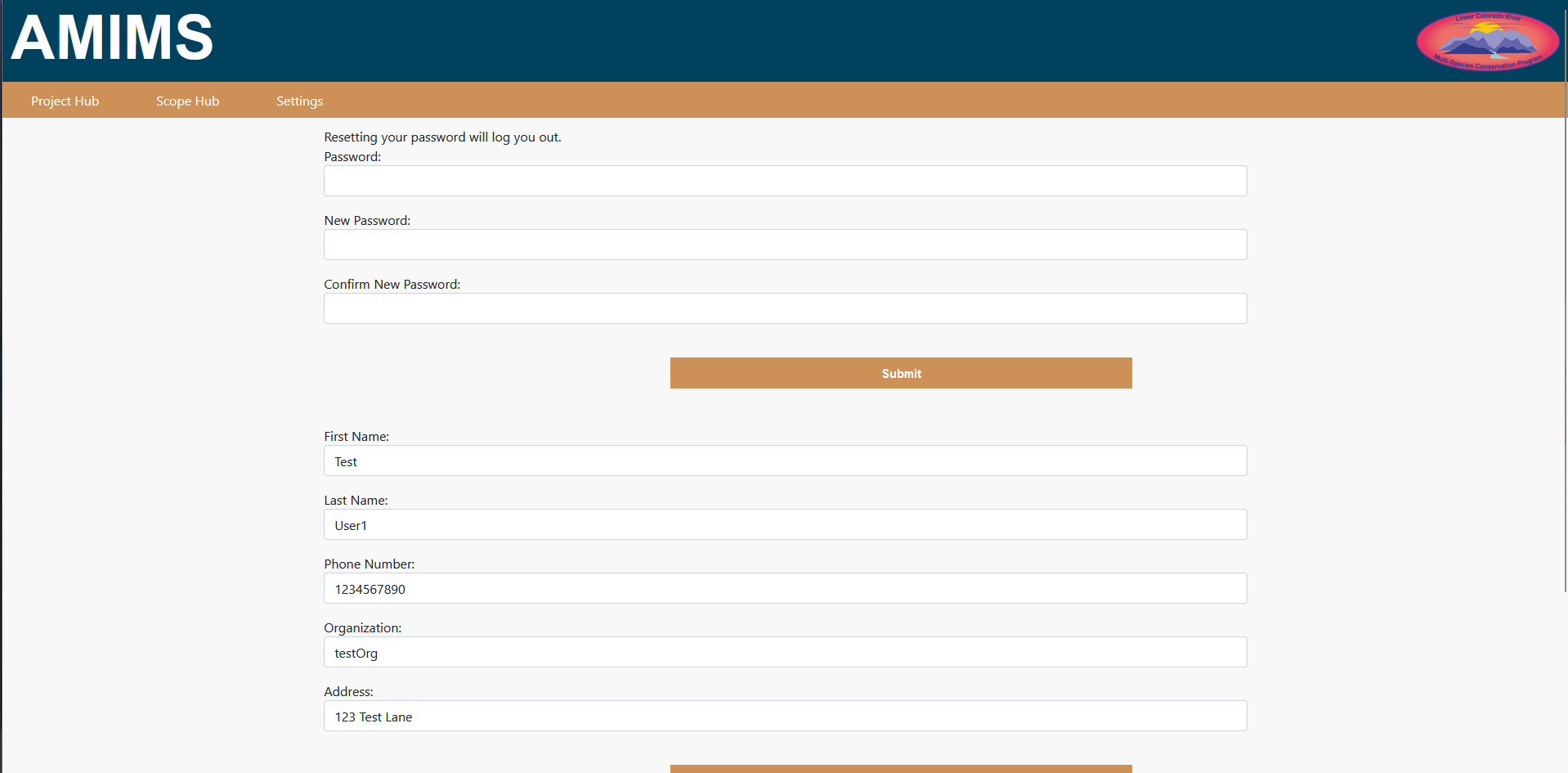
**Scopes**

Scopes are [Something] that can be referenced by multiple Projects. Scopes are formatted to be analogous to the various MSCP scopes. The text fields contain the editable information for each scope. As with Projects in order to edit the scope click the lock icon to change the fields into their editable form. Also, as an additional reminder, navigating away from a page before submitting the data will delete the unsaved data.

Fig. 9: An AMIMS Scope with its fields unlocked

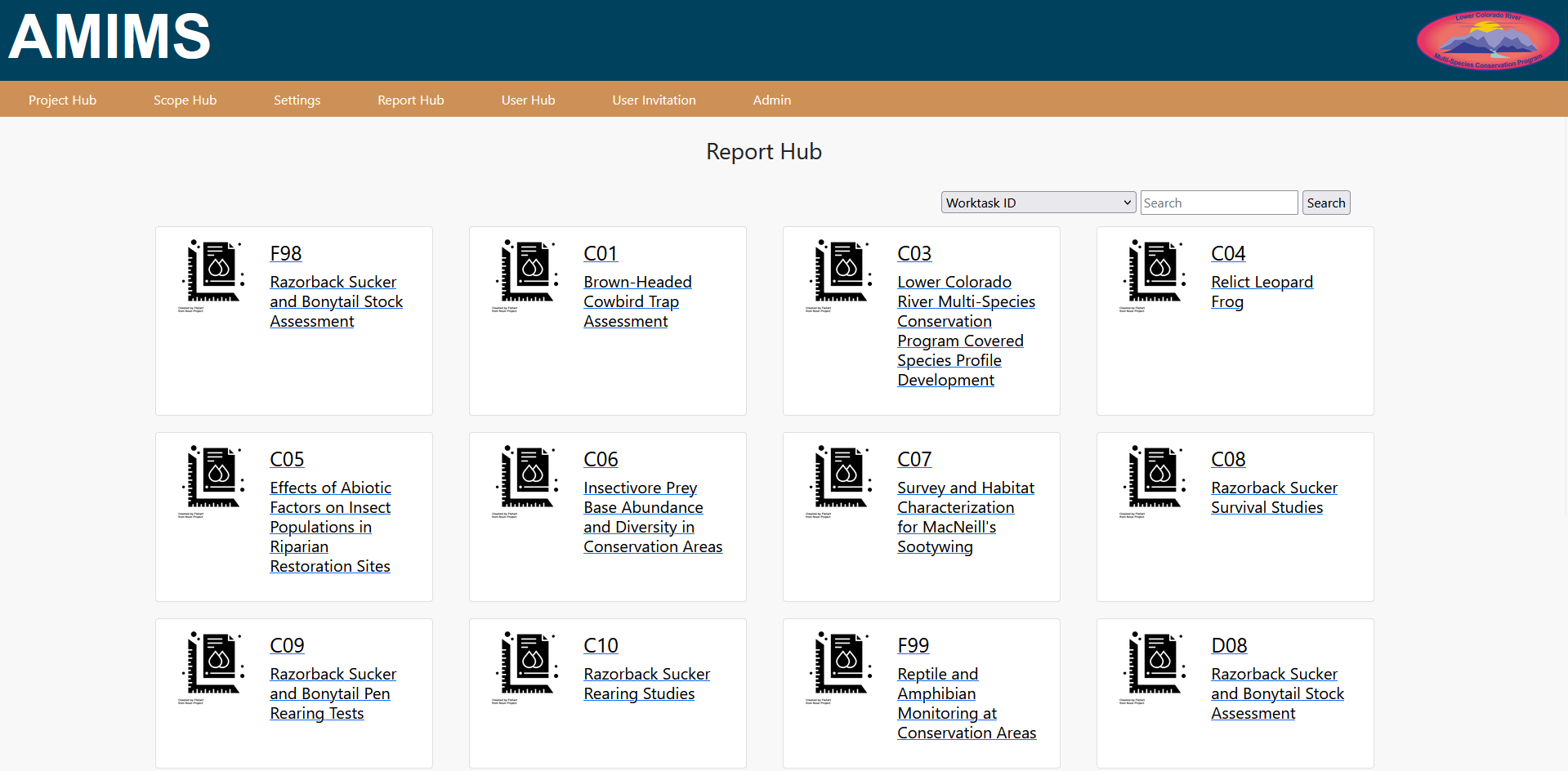
**Settings**

The Settings page is where a user can update their profile information and change their password. In order for a user to update their information simply change the information present in the setting form and press the submit button. To change a password enter the existing password into the first field and then double enter the new password into the other two fields. When the form is submitted the user will be logged out and will need to log in with the new password.

Fig. 10: AMIMS User Settings page

**Report Hub**

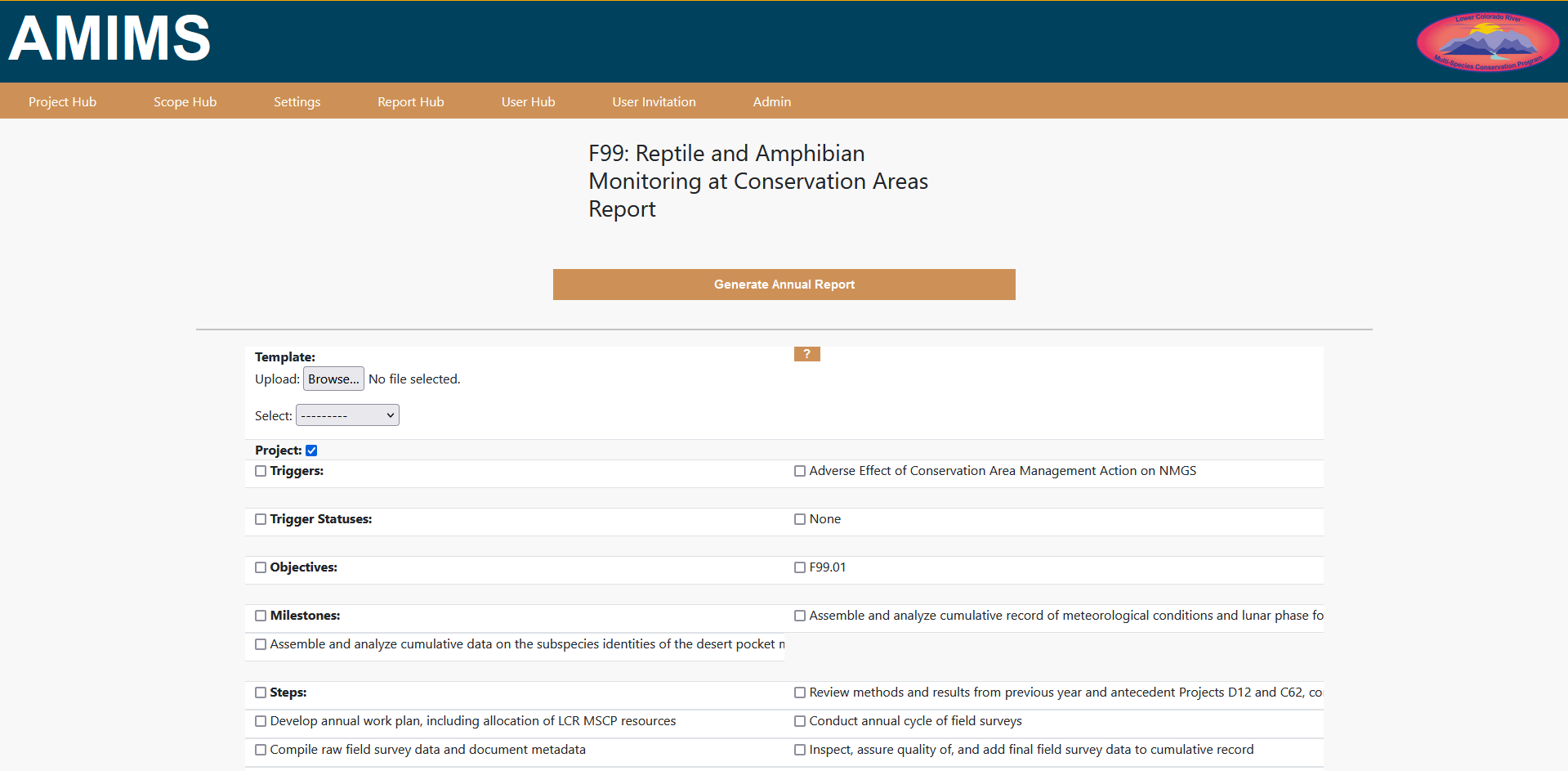
Reports are one of the main features of AMIMS and allow Admins and Contributors to format Project data into readable text. The Report Hub is a parallel to the Project Hub, for every project in the Project Hub a Report Hub card is also created. These cards direct to the Report Generation page for its respective Project. The Report Hub also has search functionality that functions identically to the search in the Project Hub.

Fig. 11: AMIMS Report Hub

**Reports**

The Report page is where the two types of reports are generated Annual reports and Project reports, also called Ad Hoc reports. The first type of report, the Annual Report, is the simplest to generate and is meant to be a complete overview of the selected project. The “Generate Annual Report” button will, as the button suggests, generate an annual report. An Annual Report is a report that is the most similar to a Worktask. It contains every component of the project arranged in hierarchical order.

The second type of report are project reports and allow for more specificity in report generation. Project reports require a template. Templates can either be uploaded via the Upload option or selected from an Administrator curated list in the Select field. To create your own template follow the in page guide by clicking on the question mark button. After a template has been selected or uploaded a number of fields must be selected. The list of tick boxes contains both Object categories and the Objects themselves. Each Object can be selected individually, but ticking the category will select all of the Objects in the category. The Project Object is selected by default. Once a template has been uploaded and some Objects have been selected pressing the “Generate Project Report” button will create the project report based on your choices.

Fig. 12: AMIMS Report Generation Page

**Administrators Guide**

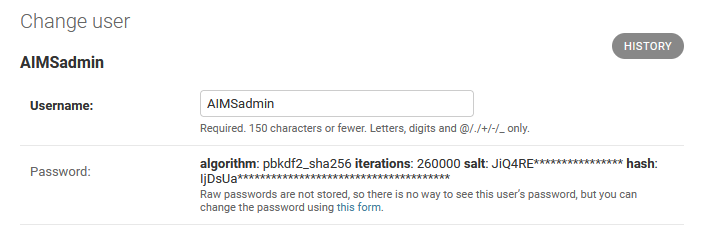
Administrators are the highest level of user in AMIMS and have access to a number of features that other users do not. The following sections involve Admin specific pages.

**How to associate a user with an AMIMS admin**

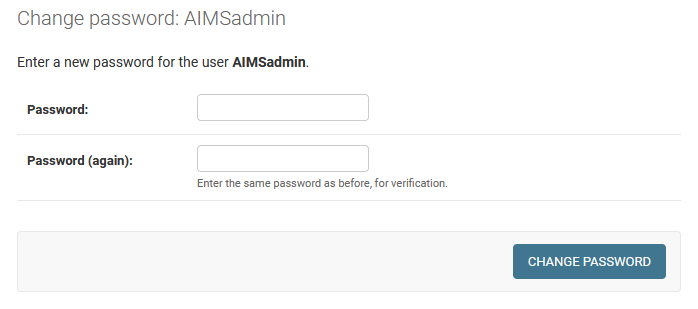
­Because the AMIMS admin is created before any other object in the system it is not associated with an AMIMS user. As using the admin like a user can cause some minor problems it is advised to use the AMIMS admin to create a second user and elevate it to Administrator status. To do this use the User Invite page to create a new user, navigate to the user on the admin page, tic the Superuser Status box, add all available groups via the ‘choose all’ button, and then save the user. This user should then have all the same permissions as the initial AMIMS admin.

**How to reset passwords using the Admin Page**

On occasion it may be nessecary to reset a users password for the m. Do do this navigate to the desired user page. Under the Password field is the following sentence: "Raw passwords are not stored, so there is no way to see this user’s password, but you can change the password using this form."

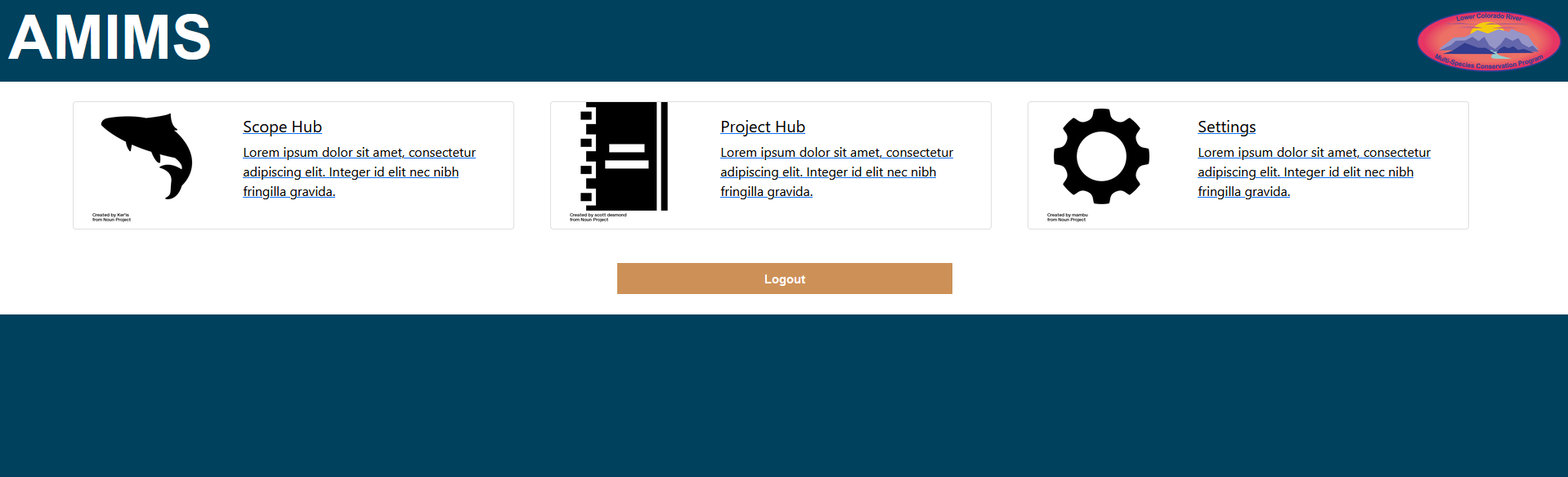
Fig. 13: The ‘Change user’ section of the AMIMS Admin user page in the Admin page

The words "this form" are a hyperlink that will take you to the password change page. Fill out the form as instructed.

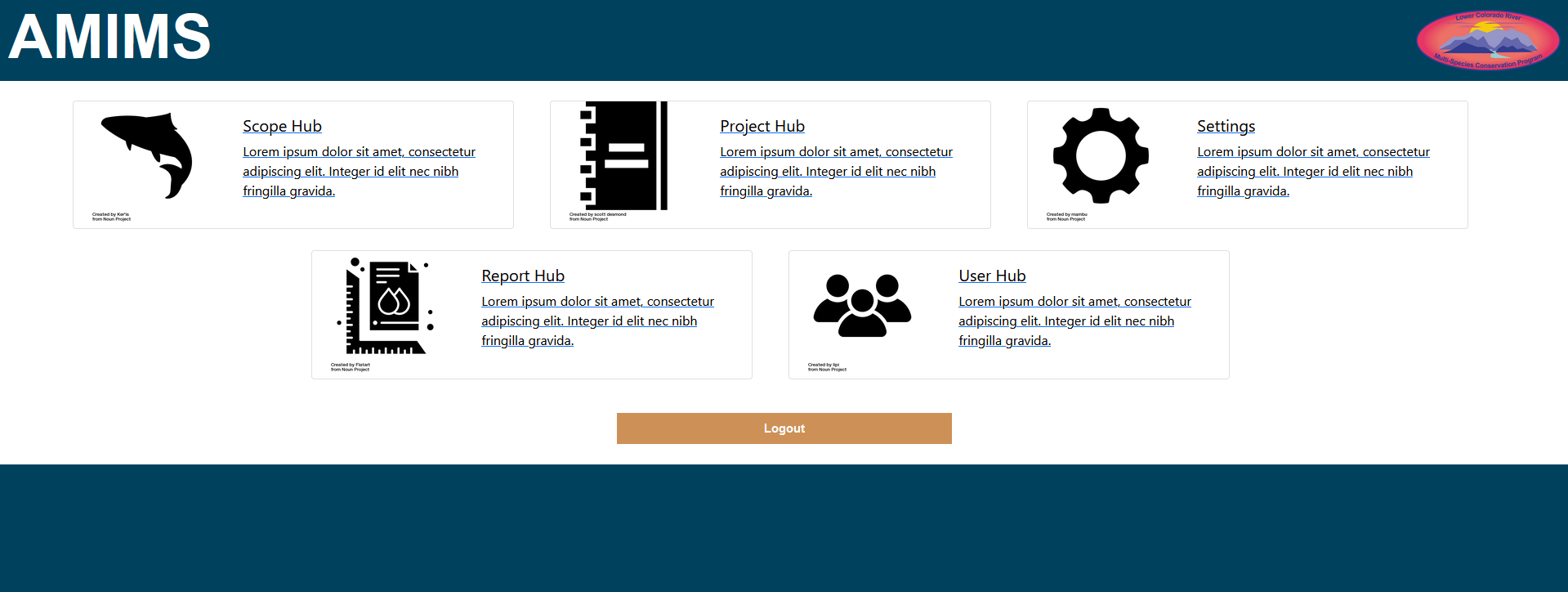
Fig. 14: The Change password form for the AIMSadmin

**User Access levels and what they allow access to**

AMIMS users are separated into three categories, each granting increasing levels of access. The lowest level users have access to the Project Hub, the Scope Hub, and the Settings page. They also have the ability to create and modify scopes, projects, and project children.

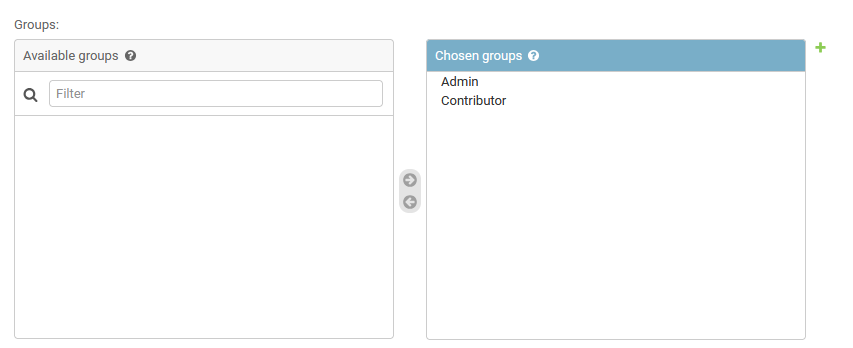
Fig. 15: AMIMS User Hub for a Basic Level User

The second level user, Contributors, have access to the Report Hub and the User Hub as well as the ability to generate reports.

 Fig. 16: AMIMS User Hub for a Contributor Level User

The highest level user, Admin users, have access to every AMIMS function. In addition to all the previously mentioned functions Admins have the ability to invite users and have access to the Django admin page.

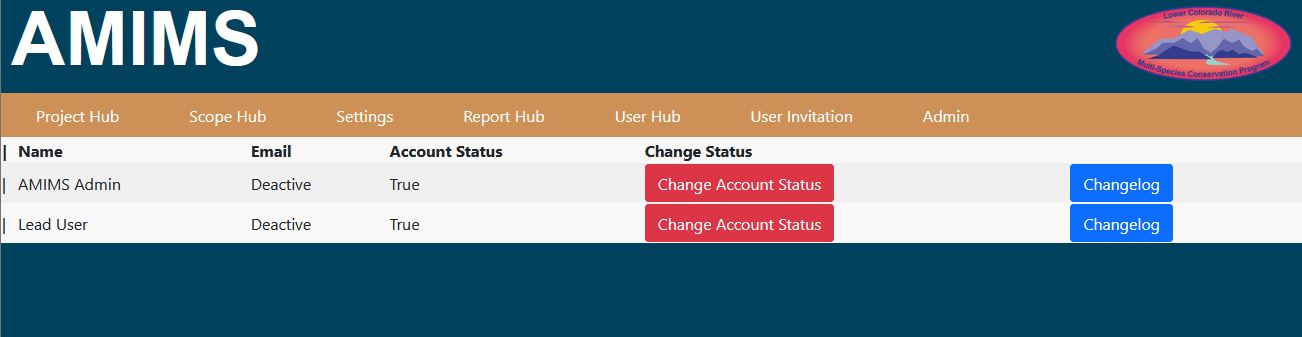
To change a user navigate to the relevant user page and to the Groups field. Select the group you want to add and click the arrow to move it to the “Chosen Groups” field.

Fig. 17: The Access level groups for AMIMS

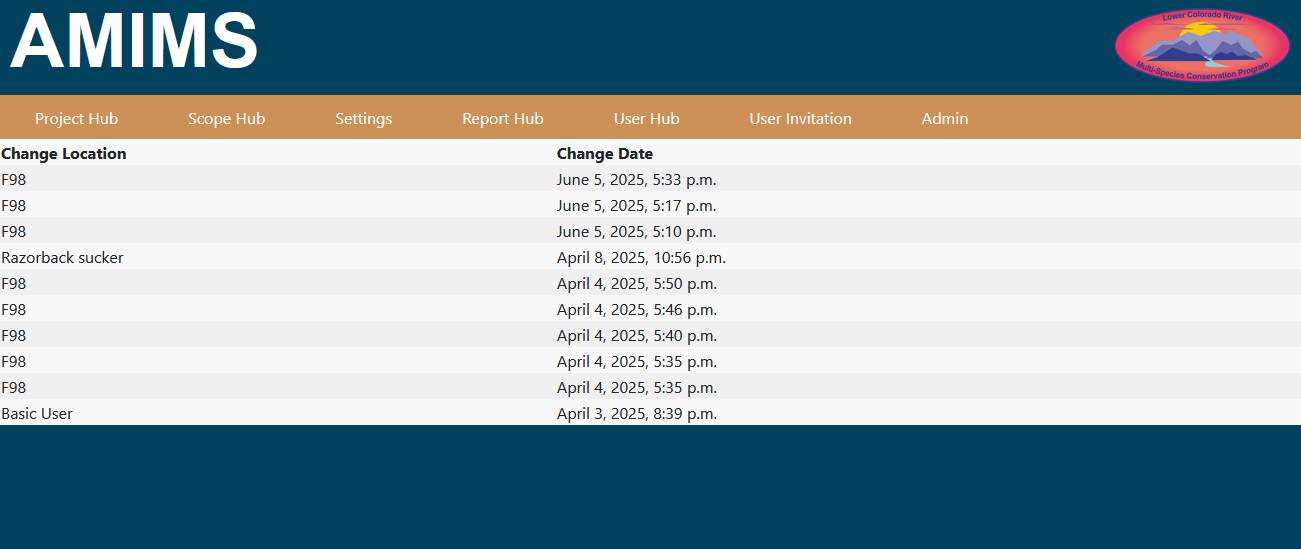
**Manual**

**User Hub**

Admin Users have access to the User Hub, a master control list for all AMIMS Contributor and Admin users. This page contains a list of all AMIMS users, their associated email addresses, and two function buttons.

Fig. 18: AMIMS User Hub

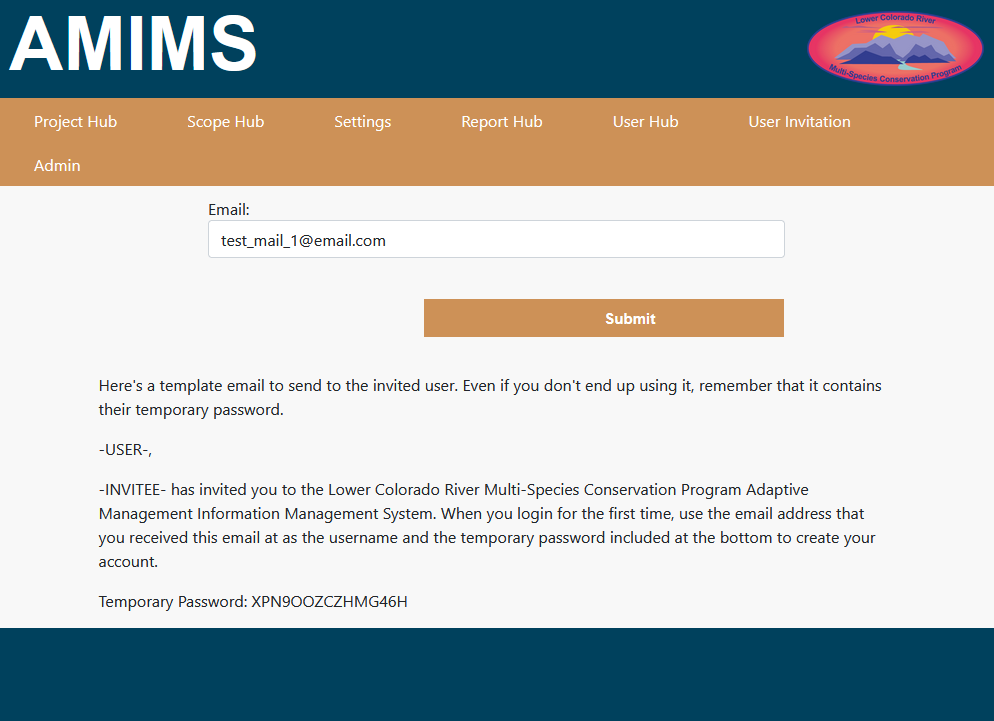
The Change Account Status button switches the user between active and inactive depending on their current account status. Inactive users are unable to login to AMIMS. The Change Log button directs admin users to the relevant users change log.

Fig. 19: A User Changelog

The change log displays all the changes made by the selected user from most to least recent.

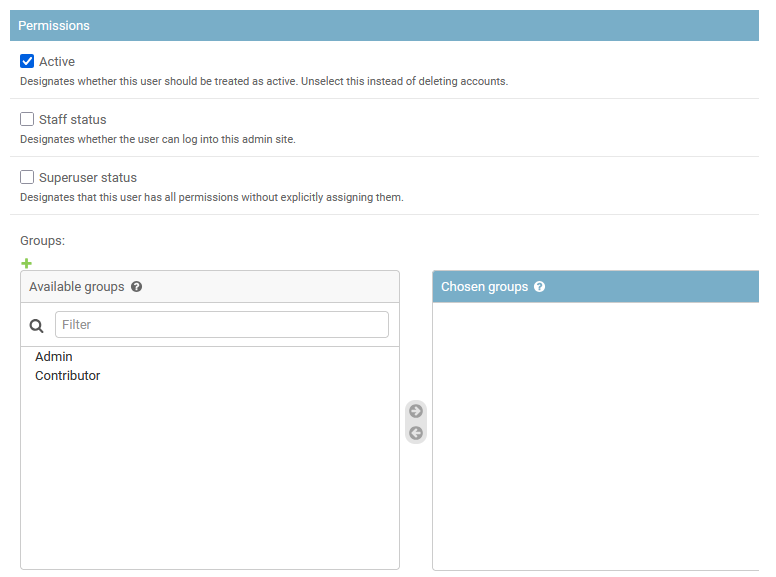
**User Invitation**

The User Invitation page is used to create new users and invite them to AMIMS. Using the User Invitation page enter the email of the new user.

Fig. 20: The User Invitation page showing the example email template

This will create AMIMS and Django user objects with the email as the username and the auto-generated password as the password. When the new user logs in with this temporary login info they will be redirected to the user creation page where they will be prompted to fill out the rest of the user object and to update their password.

Users created in this way are basic level users. To create a Contributor or Admin user a basic user needs to be promoted using the Admin page. To do this navigate to the desired user in the Admin page and find the Permissions section. Then select the groups the user is going to be added to and move them into the “Chosen Groups” with the arrow button. If the user is meant to access the Admin page also select the Staff status option.

Fig. 21: AMIMS Admin User permissions and groups

**Admin Page**

**Object philosophy**

Every Django object is meant to be a one-to-one recreation of a work task element. For instance: the Project Object has a text or character field for each field of the Project section of a work task. Each object also has, when relevant, many-to-many fields for every child and a Foreign Key for their parent.

**Reversion and Reversion-compare**

Reversion and Reversion-compare are the libraries that allow us to view historical AMIMS objects and to revert to these versions if need be. On the Admin side the functions of these libraries are accessed by the “history” button on each of the Admin object pages.

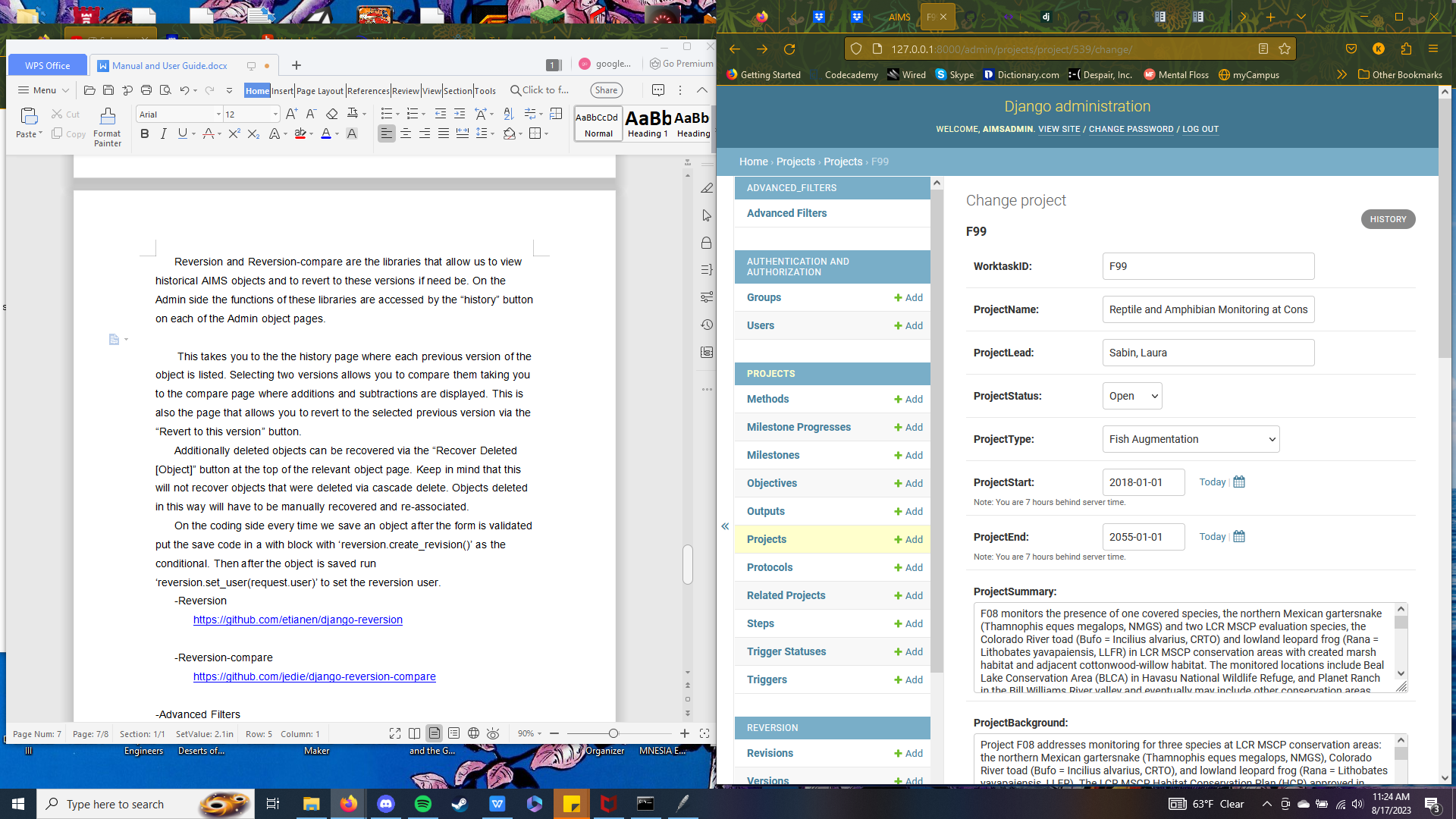


Fig. 22: A section of the admin page showing the “History” button

This takes you to the history page where each previous version of the object is listed.

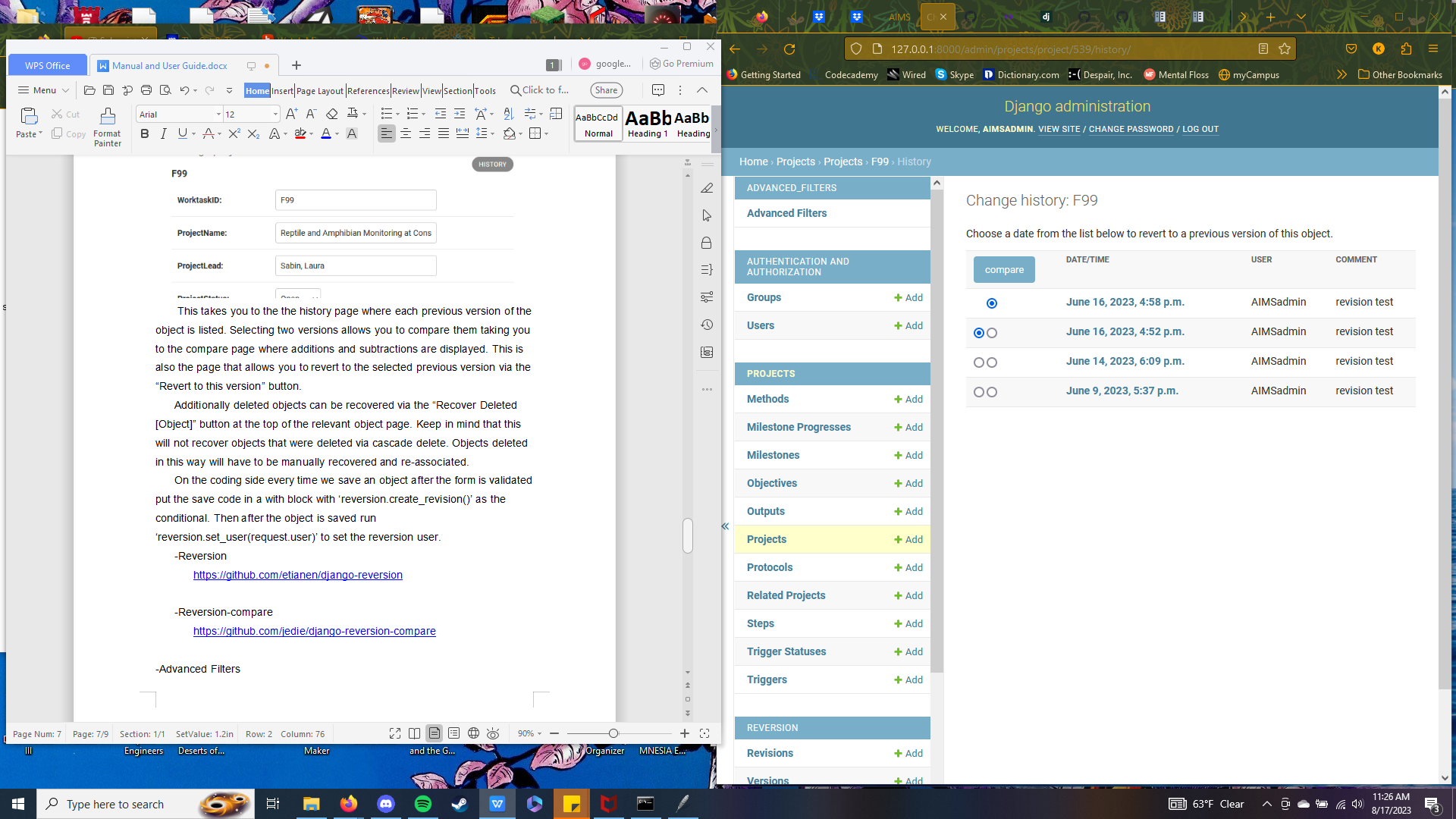


Fig. 23 The Admin change history page

Selecting two versions allows you to compare them taking you to the compare page where additions and subtractions are displayed.

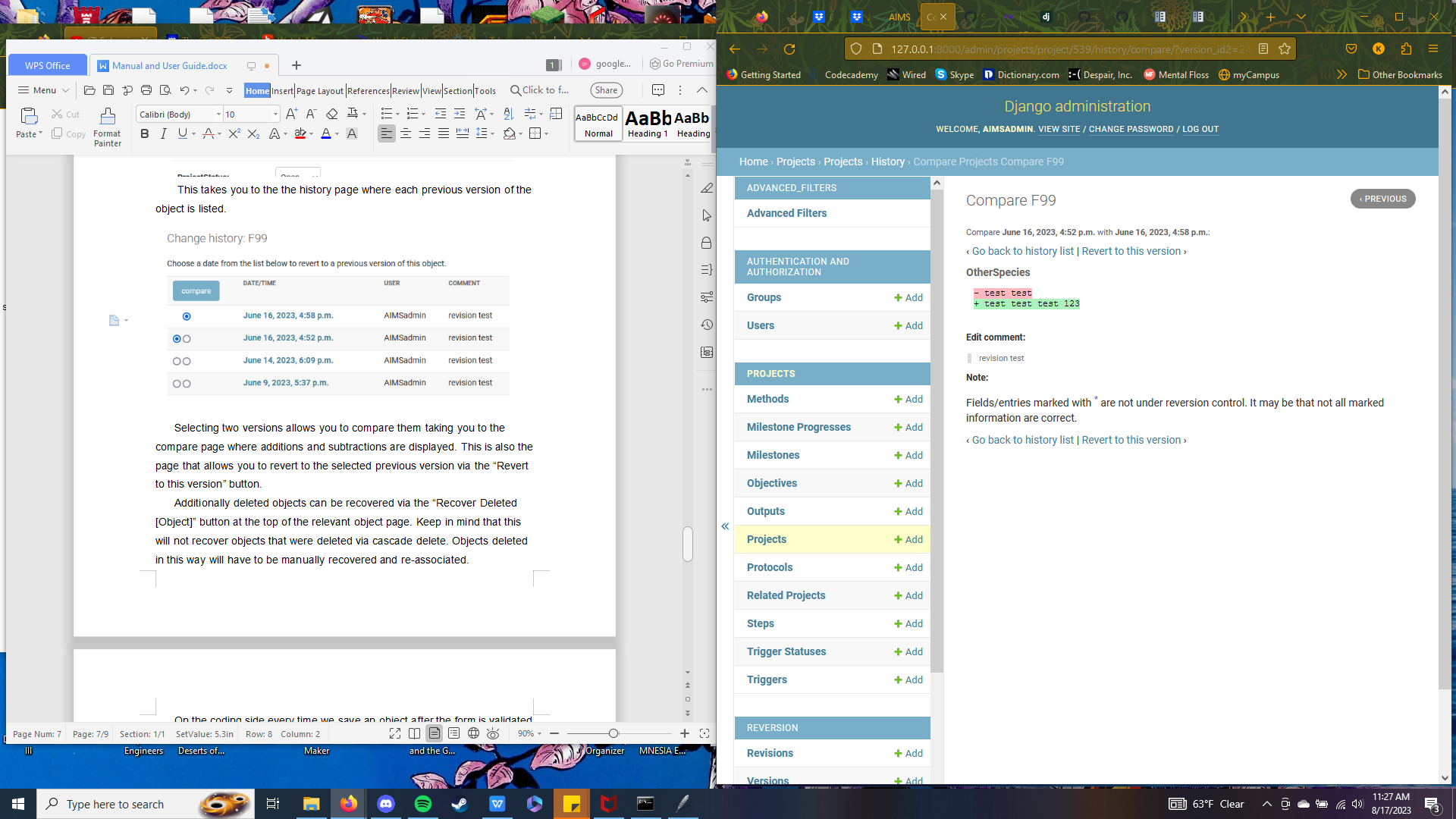


Fig. 24: A comparison page for two versions of the F99 project

This is also the page that allows you to revert to the selected previous version via the “Revert to this version” button.

Additionally deleted objects can be recovered via the “Recover Deleted [Object]” button at the top of the relevant object page.

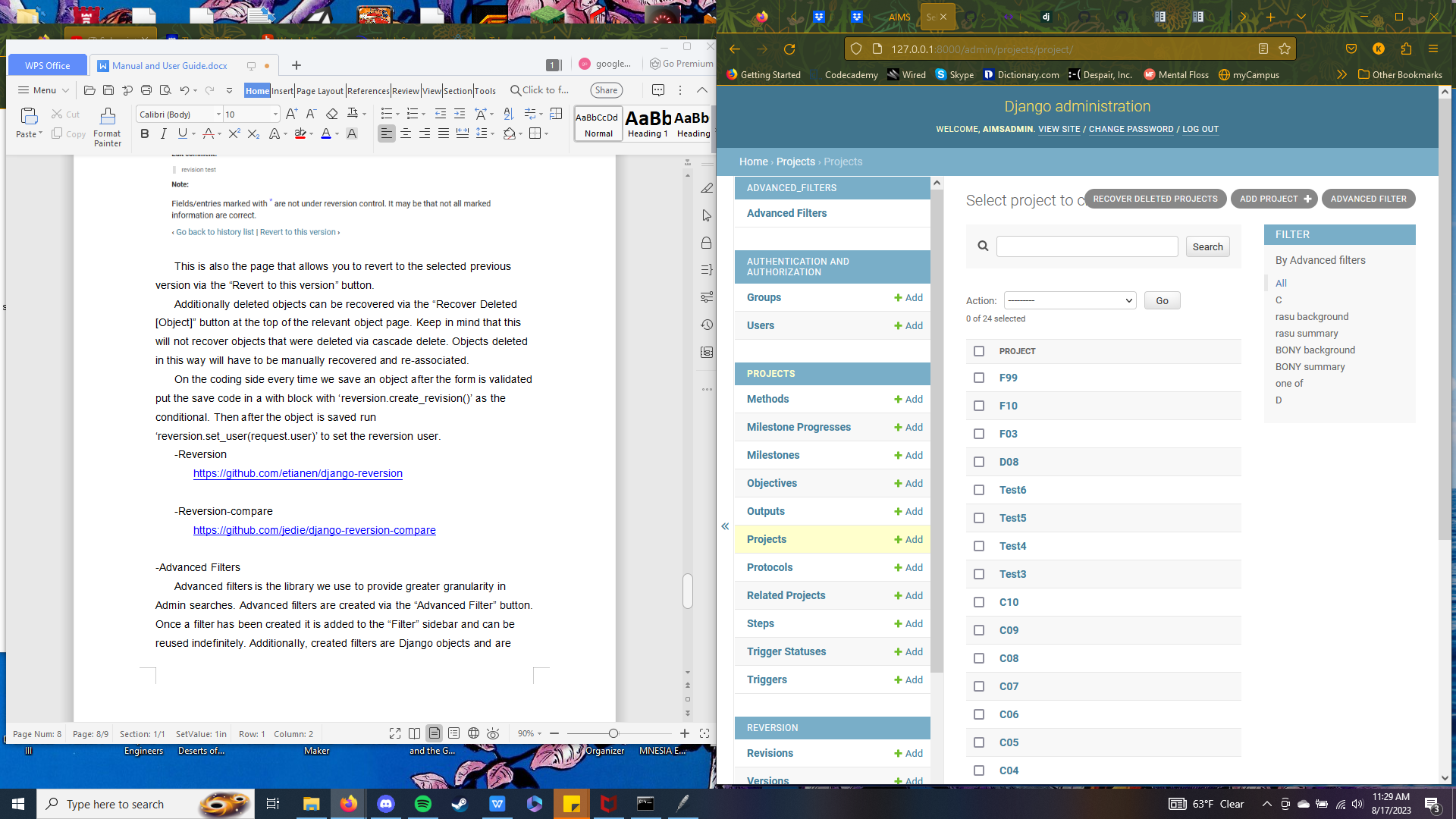


Fig. 25: AMIMS Admin Project page showing the “Recover Deleted Projects” button

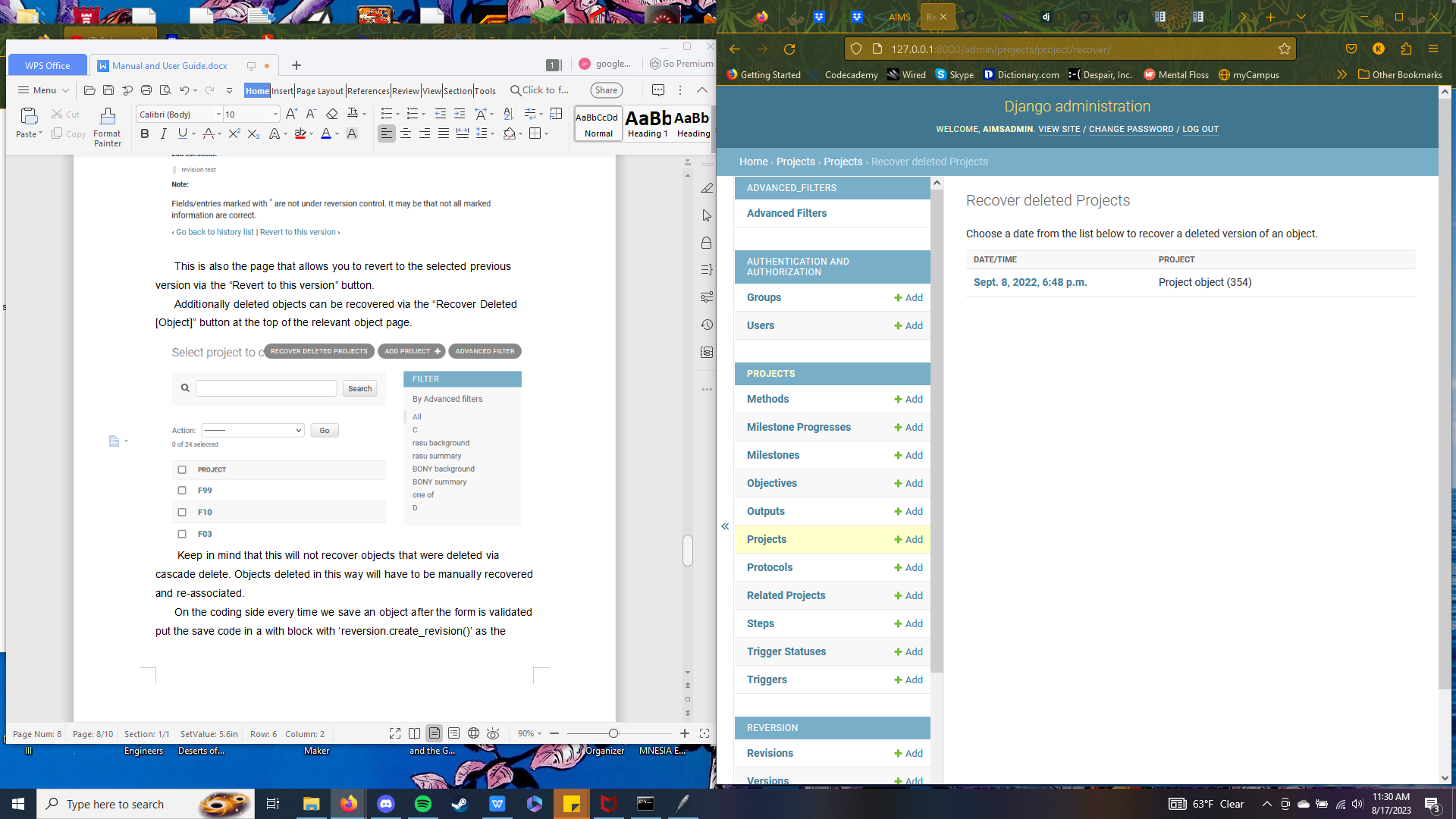


Fig. 26: The AMIMS Admin Project recovery page

Keep in mind that this will not recover objects that were deleted via cascade delete. Objects deleted in this way will have to be manually recovered and re-associated.

On the coding side every time we save an object after the form is validated put the save code in a with block with ‘reversion.create\_revision()’ as the conditional. Then after the object is saved run ‘reversion.set\_user(request.user)’ to set the reversion user.

**Reversion**

<https://github.com/etianen/django-reversion>

**Reversion-compare**

<https://github.com/jedie/django-reversion-compare>

**Advanced Filters**

Advanced filters is the library we use to provide greater granularity in Admin searches. Advanced filters are created via the “Advanced Filter” button.

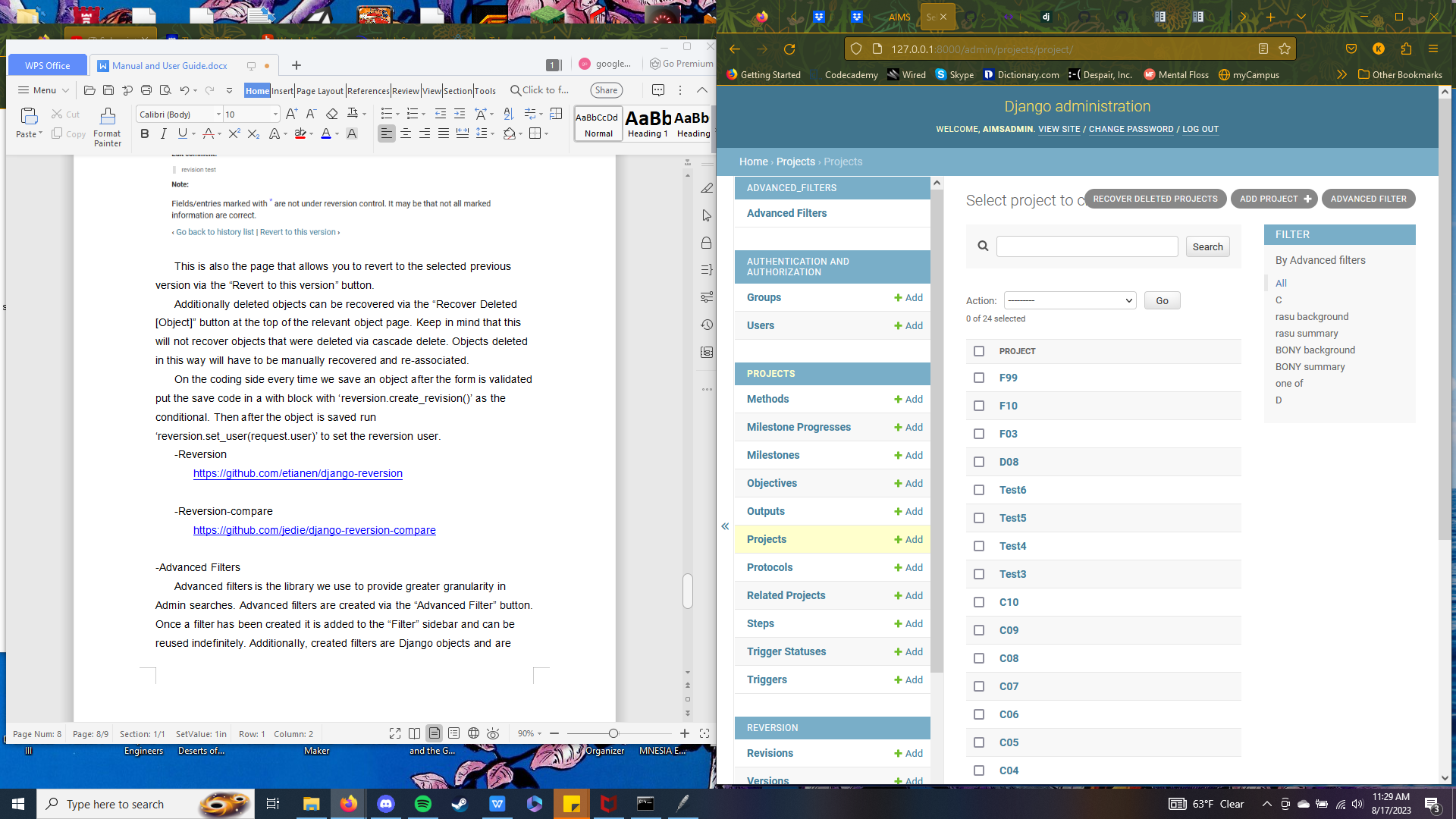


Fig. 27: The AMIMS Admin Project page showing the “Advanced Filter” button

Once a filter has been created it is added to the “Filter” sidebar and can be reused indefinitely.

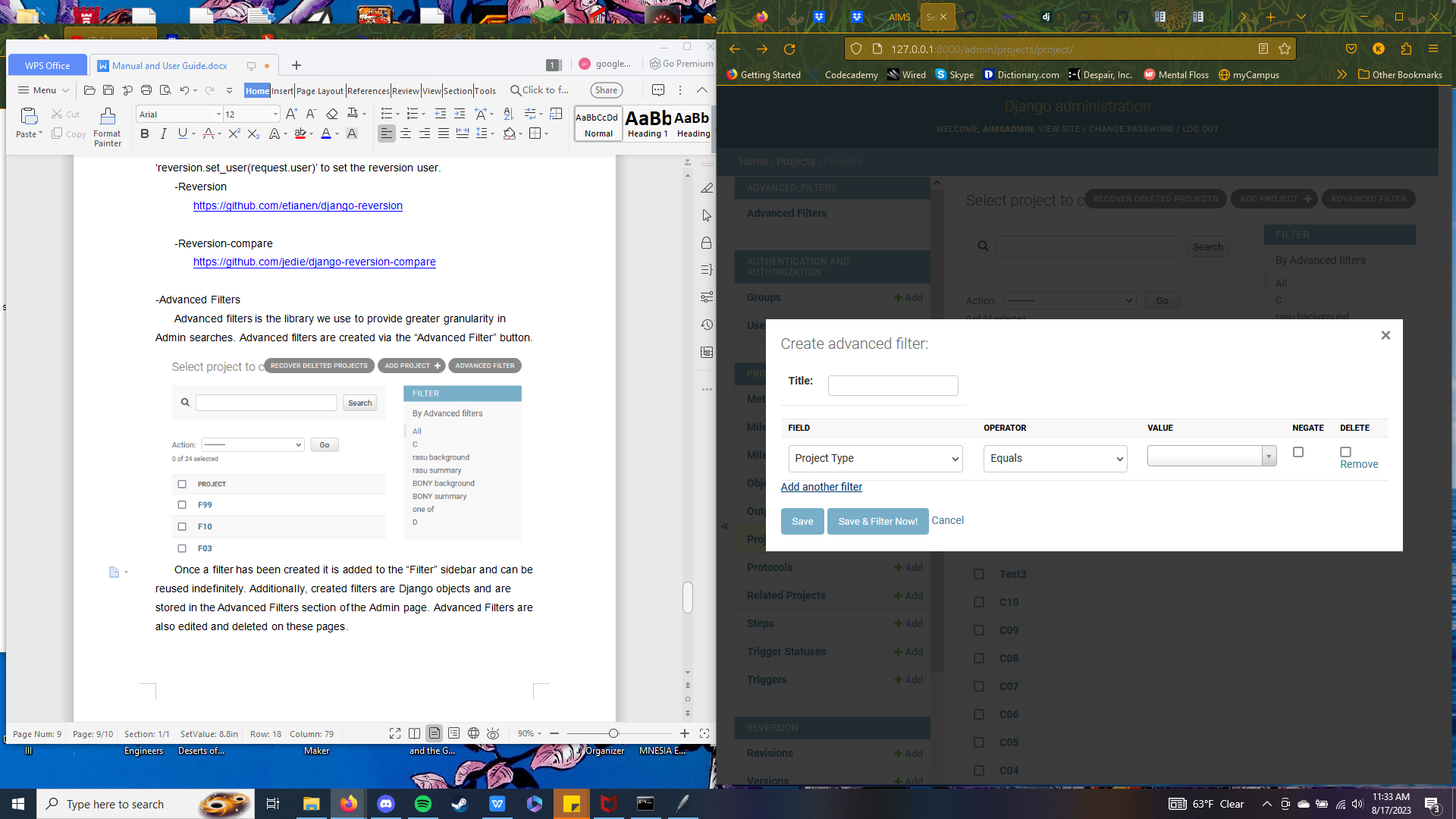


Fig. 28: The Advance Filter form

Additionally, created filters are Django objects and are stored in the Advanced Filters section of the Admin page. Advanced Filters are also edited and deleted on these pages.

On the coding side Advanced Filters are added in a similar way to search fields. The difference is that Advanced Filters are a tuple with the object field on the left and the verbose name on the right.

https://github.com/modlinltd/django-advanced-filters

**Answering Questions Using the AMIMS Admin Page**

The AMIMS Admin page has a number of powerful tools that allow Admins to answer advanced questions about the AMIMS database. The following sections explain multiple ways to answer these questions.

**Answering Questions Via the Search Bar**

The search bar searches every field of the selected AMIMS object. Searches are case insensitive. For example, Searching “Colorado River” in the Project object will return every project that contains the words “Colorado River” in any field.

**Answering Questions Via Advanced Filters**

Advanced filters are more complex than the search bar but offer much more granularity. To create an advanced filter click on the advanced filter button at the top right of the admin page. This will bring up a pop up that contains the form for creating an advanced filter. Each advanced filter must have a title and at least one filter, filters are added with the ‘Add another filter’ button at the bottom left of the pop up. Each filter has five fields; field, operator, value, negate, and delete. Negate and delete are optional, but the other three are required. Field refers to the field to be searched in the AMIMS object, e.g. the Project object contains the fields Worktask ID, Project Status, Conservation Measure Code, etc. Field also contains the OR operator which we will discuss later. The operator drop-down holds a set of logical operators that preform the main action of the search. The value field will contain the actual search value. This field is acts like a text box, but is in fact a semi-searchable drop down. If you would like to search for a value not contained in the generated list simply type into the box and click on your typed entry. Please note that values are case sensitive. The negate field is a tick box and will invert the logic of the filter. For example, if the filter is set to search for all projects whose worktask IDs begin with ‘C’, the negate option will change it to search all projects whose worktask IDs do not begin with ‘C’. Finally, the delete field is another tic box that will delete the filter after it has been used. Normally advanced filters are saved so that they can be reused later, but selecting this option will prevent this from occurring.

**Example 1**

In this example we will use the search bar to find every project object that contains the phrase “Colorado River” in any field as described earlier.

1. Enter the search phrase into the search bar and click the Search button or press the Enter key. Note that the phrase does not need to be case consistent.

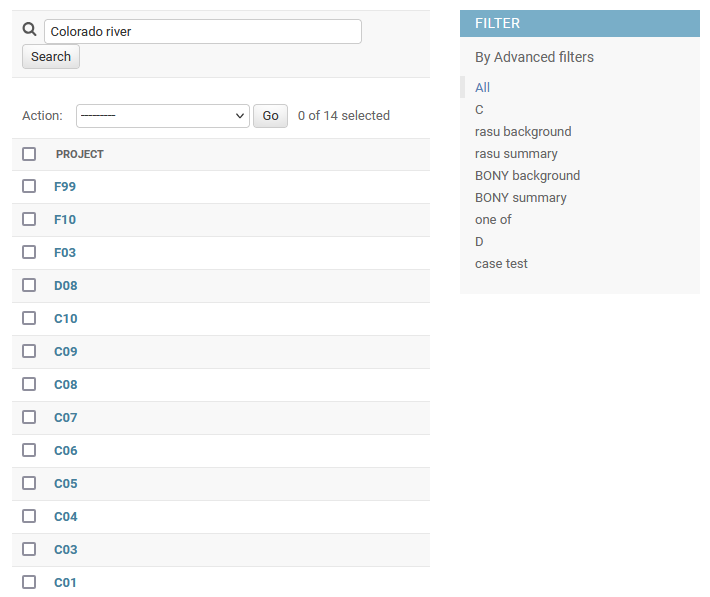


Fig. 29: AMIMS Admin Project page with “Colorado River” in the search bar

2. You can now see that we have a filtered selection of projects that contain our search phrase.

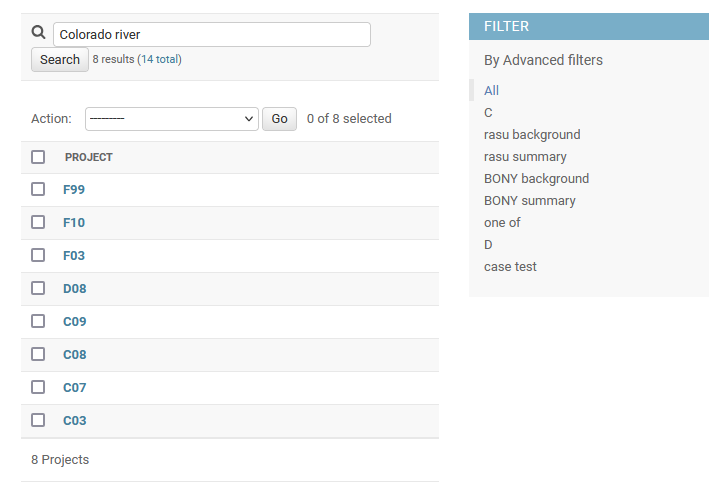
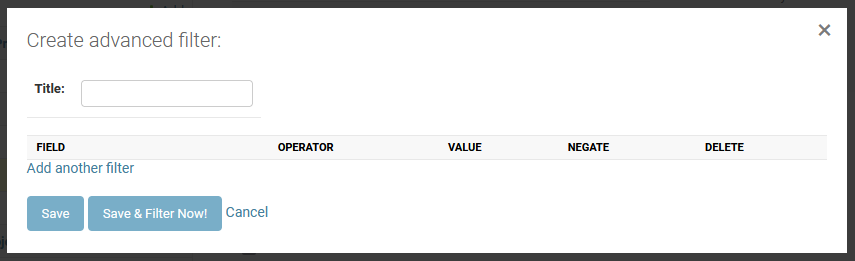


Fig. 30: AMIMS Admin Project page showing filtered Projects

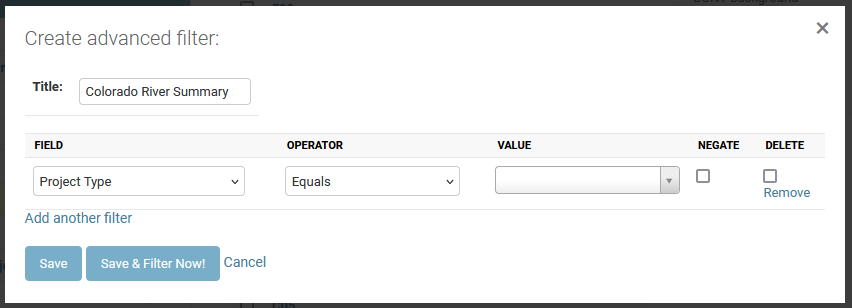
**Example 2**

In this example we will use advanced filters to search for projects that contain the phrase “Colorado River” but only in the project summary field.

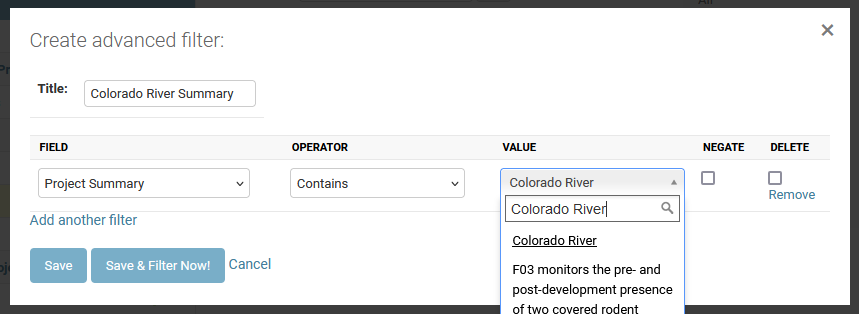
1. Click on the Advanced Filter button at the top right of the page to bring up the advanced filter form.

 Fig. 31: A blank advanced filter form

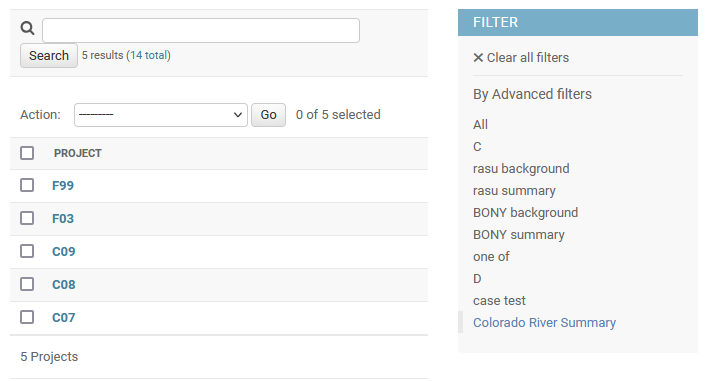
2. Type the name of the filter into the “Title” box, we’re going to name our filter “Colorado River Summary”. Then click on “Add another filter” to add an empty filter.

 Fig. 32: An advance filter with an empty field

3. In the filter change the field to “Project Summary”, the operator to “Contains”, and type “Colorado River” into value. Remember to click on “Colorado River” rather than pressing the enter key. Please also note that the value field is case sensitive. Once we’ve done this click on the “Save & Filter Now!” to preform our search.

 Fig. 33: An advanced filter field showing “Project Type Contains Colorado River”

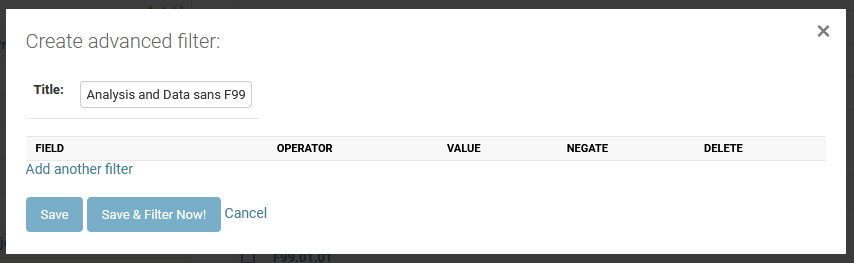
4. You can now see that we have a filtered list of projects that mention “Colorado River” in the project summary. Also, notice that in the “Filter” side bar our filter is selected. If we wished to preform this search again all we would need to do is select “Colorado River Summary” from this list.

 Fig. 34: The AMIMS Admin Project page with the “Colorado River Summary” filter selected

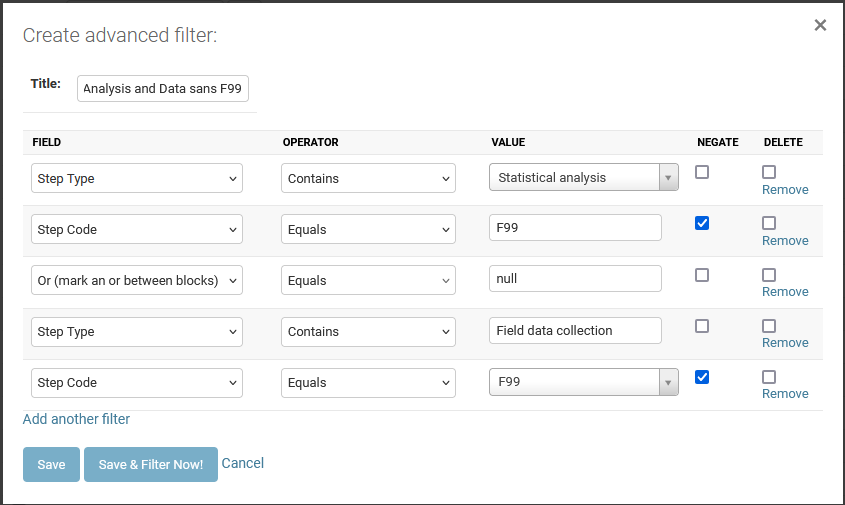
**Example 3**

In this example we are going to use advanced filters to preform a much more complex search. We are going to filter step objects by types that include “Statistical analysis” or “Field data collection” but aren’t in our example worktask F99.

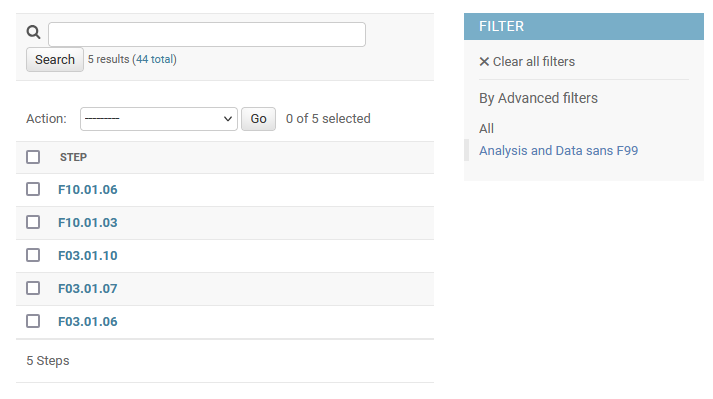
1. Like in example 2 start by creating an advanced filter and naming it. We’re going to name ours “Analysis and Data sans F99”.

 Fig. 35: A blank advanced filter titled “Analysis and Data sans F99”

2. To filter types by “Statistical analysis” or “Field data collection” we’re going to need five filters. Our first two filters will be “Step Type Contains Statistical Analysis” and “Step Code Contains F99” with the negate box ticked to invert the filter. Our third filter is the OR operator. This operator is found at the bottom of the field drop-down and allows us to separate blocks of filters. Filters four and five are similar to one and two but for “Field data collection”.

 Fig. 36: The advanced filter with the fields filled out

3. We now have a filtered list that only contains steps that match our search terms.

 Fig. 37: The AMIMS Admin Step page with the “Analysis and Data sans F99” filter selected

**Managing Static Files**

Several features of AMIMS use uploaded or “static” files. Uploaded pictures are saved in the ‘media’ folder. To delete a media file the server administrator will need to navigate to the media folder within the AMIMS directory and delete it from there. Media files can be freely deleted from the folder, but trying to access the deleted file from the associated Scope will throw an error.

**Using the Report Importer**

The Report importer is a pair of standalone scripts for importing completed Worktasks. The full importer is used to batch translate Worktasks into AMIMS. The program points to a folder of plain text files that it reads to generate the AMIMS objects. The importer relies on each AMIMS Worktask being identically formatted as it parses each file based on section headers. As such it’s imperative that each file have identically formatted and ordered headers in order for the importer to read them properly. The most common failure point in this program is the incorrect formatting of a file.

**Using the mini importer**

The mini importer is essentially a paired down version of the full importer. Everything that is true about the full importer is also true about this importer. However, this importer is only for “Mini” files, files with only Project level objects.