



This project is co-financed by the European Union

Grant Agreement No.: 824603

Call: H2020-SwafS-2018-1

Type of action: RIA

Starting date: 1/02/2019



Lifecycle-aware citizen science templates Version 2

Coordinator: Esteban Gonzalez, Oscar Corcho
Quality reviewer: Neal Reeves

Deliverable nature	Technical report
Dissemination level	Public
Work package and Task	WP4 T1.2
Contractual delivery date	31/01/2022
Actual delivery date	

Authors

Author name	Organization	E-Mail
Esteban Gonzalez	UPM	egonzalez@fi.upm.es
Óscar Corcho	UPM	ocorcho@fi.upm.es
Rafael González	UCM	rgonzalez@ucm.es

Abstract	This document describes the different templates that are going to be developed in ACTION for helping pilots to export/use external platforms. Also, a new tool to create data workflows will be described. Finally, a guideline illustrates the process to create a workflow in ACTION.
Keywords	Citizen science platforms, data management

Disclaimer

The information, documentation and figures available in this deliverable, is written by the ACTION project consortium under EC grant agreement 824603 and does not necessarily reflect the views of the European Commission. The European Commission is not liable for any use that may be made of the information contained herein.



This deliverable is licensed under a Creative Commons Attribution 4.0 International License

How to quote this document

Gonzalez, E, Corcho, O. (2020). *Lifecycle-aware citizen science templates*

TABLE OF CONTENTS

EXECUTIVE SUMMARY	4
1 INTRODUCTION	5
2 EXTERNAL PLATFORMS	6
2.1 Epicollect	6
2.2 Zooniverse	8
3 TEMPLATES	10
3.1 Introduction	10
3.2 Data Management Plan templates	10
3.3 Data Management Workflow templates	11
3.3.1 Common Workflow Language	12
3.3.2 Generic Workflow description	15
3.3.3 Apache Airflow: A platform to execute workflows	18
3.3.4 ASSET: A tool to describe workflow	19
3.4 Presentation template	19
4 ASSET Tool	21
5 CONCLUSIONS	28
ANNEX I: Guideline to create workflows in ACTION	29

EXECUTIVE SUMMARY

In the context of our project, templates are mechanisms to build citizen science projects following the *good practices* that will be analyzed in ACTION, in particular in WP5. These templates can be either tools (software) or guidelines to facilitate the use and the integration of tools in Citizen Science projects, covering the steps of their lifecycle.

After an analysis made in deliverable D4.1 (Conceptual Architecture) about citizen science projects, and our first version of this deliverable, we discover that many of the projects suffered a lack of data management and policies. Due to these facts, we decided to focus this version of the deliverables in the creation of templates to define and to execute workflows

A Workflow is a sequence of tasks that can be executed. The concept of workflow in computer systems allows the automation of tasks, reducing human intervention in the process. In fact, one of the biggest problems in CS projects is that they are no longer maintained once funds are run out. It always depends on a project's participants to download data from one place, create new tasks and publish the results.

To help CS projects, ACTION has deployed a platform to execute workflows (Apache Airflow), and adapted a tool (ASSET) to design specific workflows for CS projects. These workflows use templates already designed to manage data from external platforms such as Epicollect5, Zooniverse and Zenodo. This allows to connect the different resources identified in the previous version of the deliverable.

1 INTRODUCTION

This deliverable covers the work done regarding the objective O4.1 to create a set of templates to create projects in external platforms such as Zooniverse, PyBossa and SciStarter. As we analyzed in the previous version of the document, templates in PyBossa were not feasible because this platform is no longer maintained. Also, the SciStarter provides a generic template to register Citizen Science projects in its platform so there is no need to provide extra functionality in this aspect to new CS projects

Therefore in this version, we focus our efforts in the platform Zooniverse and in a set of tools to support CS projects in data management tasks.

In section 2, an introduction to Epicollect5 and Zooniverse is included..

In section 3, the methodology and the templates for data management in CS are introduced

In section 4, a new tool ASSET, to create sketches of workflows is described.

In Annex I, a guideline to create a workflow based on our templates and in our suite is provided.

2 EXTERNAL PLATFORMS

The initial purpose of the templates is to *export* citizen science projects based on pollution domain to the main citizen science platforms. In this way, we can raise the participation and outreach of the project.

In this section, we are going to explain the different options that the external platforms offer, and how we can interact with them.

2.1 Epicollect

Epicollect5¹ is a platform to create customized online forms to collect data. This data can be downloaded in CSV or JSON format and visualized on a table or on a map. The platform has an API to get the entries with some limitations such as 60 requests per minute for entries and 1000 entries per request. The application is designed for Android mobile phones and it only works in mobiles with these operative systems.

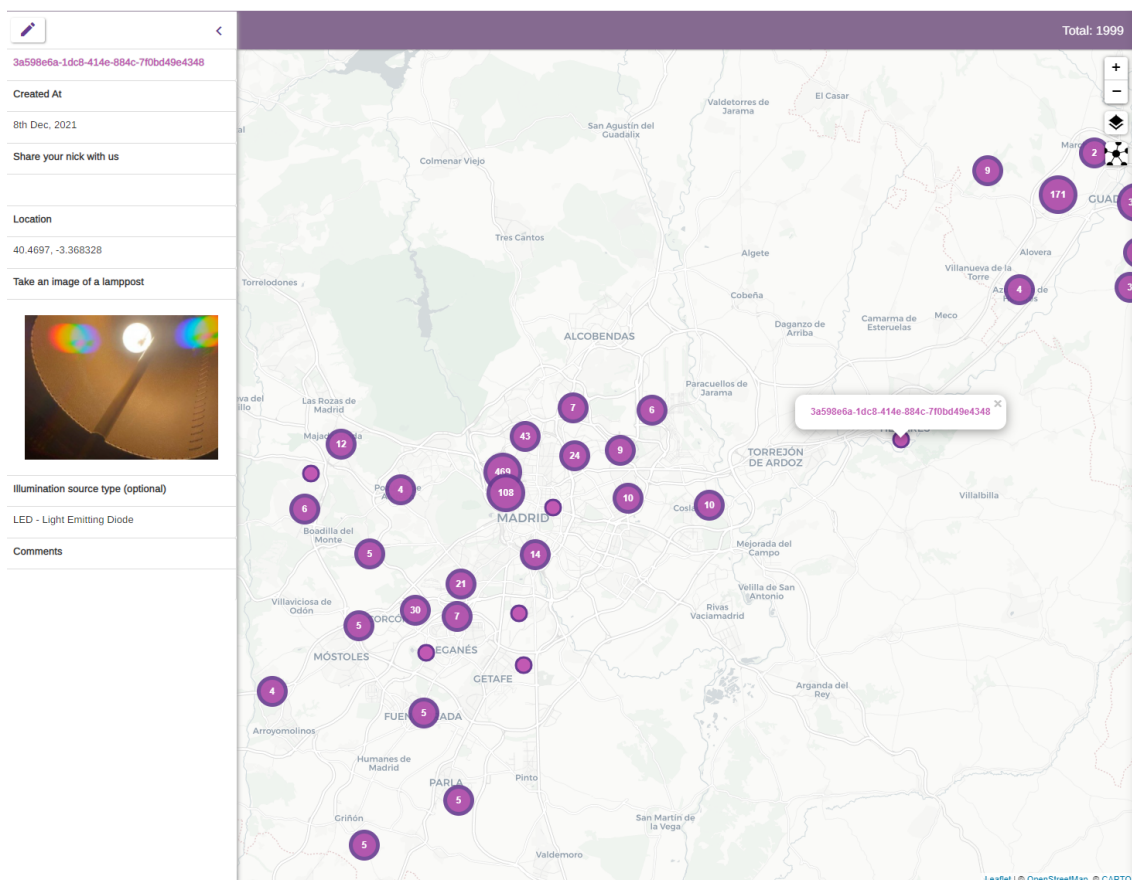


Fig 1. Epicollect5 map

¹ <https://five.epicollect.net/>

The application has a graphical form builder to easily create a form based on basic components. These include fields to manage texts, numbers, phones, dates, time, text boxes and elements to add radio buttons and checkboxes. In the case of date and time, this information is completed automatically with the current date and time, although the user has the possibility to change it manually. This facilitates the work of the user introducing new entries. Time is stored in the system in UTC format.

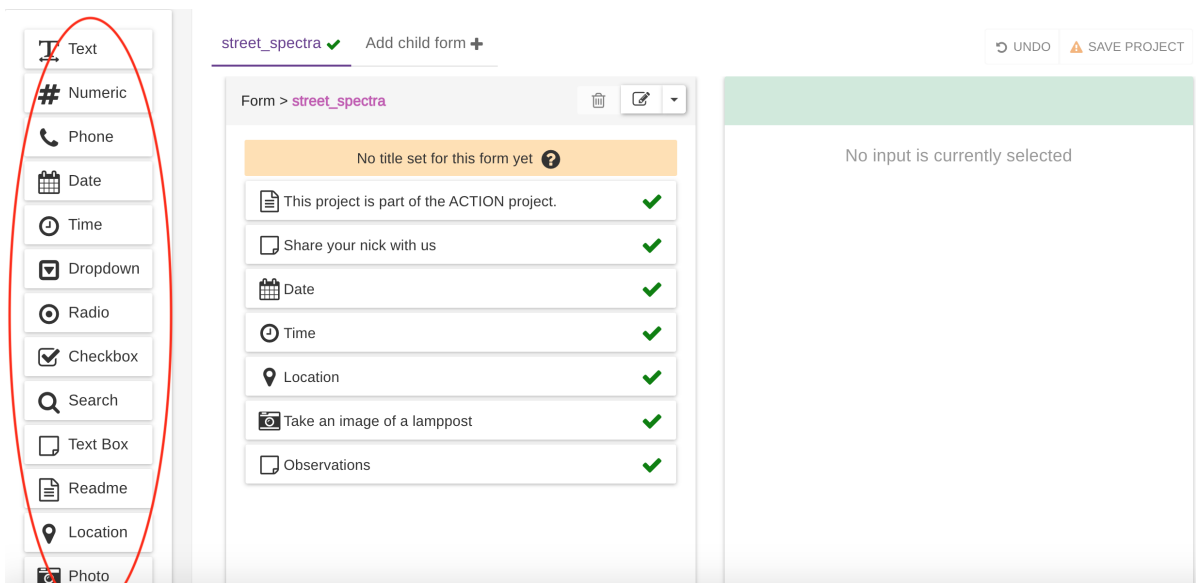


Fig 2. Epicollect5 form

Epicollect5 is also designed to add media data such as images, video or audio. Citizens can add images of street light spectra, audio of the noises in their city or videos of animals. This allows CS designers to enrich their data collected. These media files can be added from the mobile gallery or taken in the moment of form fulfillment. Another interesting field is the location. The application takes advantage of the capacity of a mobile to get the location from the GPS and include it in the device. Therefore, this field includes the coordinates of the observation automatically in the form, although it can be done manually if needed.

To prevent the privacy of the citizens, the IP or any other personal information of the user is not shown in the observations, except the position of the citizens. It is not a problem because the identification of the user is not recorded.

Epicollect5 provides a mechanism for building smart forms. Smart forms can be formed by multiple subforms that can be accessed depending on some particular conditions. For example, if a citizen responds with a specific value to a field, the flow jumps to another subform with specific fields for this answer.

Citizens can work with Epicollect5 in offline mode. It means that they can collect data without internet connection in their mobiles.. Once they are at home, they can send the observations from there. This can be very useful in cases where there is no internet coverage in remote places.

To create a project on this platform, a registered user is needed. Any user can add new observations, even if they are not registered as a user in the platform. Projects can have multiple administrators (can modify the project) and curators (can modify the entries of the project).

In relation with the templates, in the previous deliverable, we prepared a guide for creating Epicollect5 projects. An example of the street spectra form was used for this purpose. Plus, in Epicollect5 it is possible to export/import forms, so a template of an example of the project was provided to pilots.

In the context of ACTION webinars, we organized one to show pilots the different tools to create CS projects like Epicollect5 and Zooniverse. As a result, two pilots adopted this tool to collect data.

Walk Up Aniène: <https://five.epicollect.net/project/walk-up-aniene>

Open Soil Atlas: <https://five.epicollect.net/project/open-soil-atlas-action-2021>

2.2 Zooniverse

Zooniverse² is one of the most popular citizen science platforms. In this platform, citizens can create their own citizen science projects and feed them with tasks (built over images mainly). The platform is highly configurable and customizable³. In this analysis, we will focus on those parts that we will be filled in through the templates.

There are three main categories of parameters:

- **Definition of the project:** In this area, the project leader defines the objective and a brief description of the project. Also, more parameters such as introduction text, research question, tutorials and logos can be defined.
- **Workflow:** In this area, the coordinator or an authorized user can design the tasks that will be presented to the users. These tasks can be based on four types: questions, drawings, surveys and transcriptions. These tasks can be combined forming a workflow.
- **Subject Sets:** Once the workflow has been defined, this will be fed with the different tasks (mainly with images).

Definition of the project

In this part, the project leader defines information related to the description of the project. This information is used to create the project on the platform. This information can be used in the future

² <http://www.zooniverse.org>

³ <https://help.zooniverse.org/getting-started/>

Lifecycle-aware citizen science templates

to feed some templates/tools, for example, to describe the metadata needed to integrate its results in external repositories such as Zenodo or Github or to create websites for the projects.

We can highlight the following fields:

- Name
- Description
- Introduction
- Tags
- Social links

Workflow

A workflow is a sequence of tasks that the volunteers execute to report observations. In this section, the project coordinator has to design the workflows configuring some fields. This includes:

- Title of the workflow
- Subject sets
- Tutorials associated to the workflows
- An option for the volunteers to see classification summaries
- An option for the volunteers to pan and zoom on the image
- An option to show a set of images instead of only one image
- The limit of classification per subject

Besides these configuration fields, other fields can be shown up depending on the type of the task selected such as questions, drawings, surveys and texts.

In the case of **questions**, the designer can define a question and a set of possible answers that volunteers have to answer from. This question is related to an image that was previously uploaded in the subject sets section. The designer can add the option for multiple answers..

In the case of **drawing**, volunteers can draw different shapes such as circles, polygons, ellipses onto the image presented. The object is to create a region of interest for other tasks. This information is included as a result

In the case of **surveys**, volunteers identify objects in the image(s) by filtering by their visible characteristics, and then they answer some questions previously defined.

In the case of **texts**, volunteers write in a free field of text the transcription of a text that they are seeing in an image.

We will see in section 3 how this approach can be exported to Zooniverse.

Subject Sets

These are the resources (images) that can be classified by the volunteers. This information can be manually uploaded to the platform or automatically through the API or the panoptes application. Zooniverse has several clients developed in the most popular languages, to interact with its system. It allows external systems to build and configure a project, as well as feed the project with new tasks and images. It is the way that our templates will interact with Zooniverse.

3 TEMPLATES

3.1 Introduction

In the previous section, we have seen that the platform Epicollect5 is a good choice to create projects for collecting data, including media files. Also, Zooniverse is an ideal choice for classifying data. In the previous version of the deliverable (D4.2), ACTION wrote a guideline to create projects in both systems. This has been used by other pilots to collect their data.

In our periodical meeting with our pilots, we detected the necessity to create templates for data management as most of them are composed by amateur scientists. Therefore, we focused our efforts in this version in creating a system to help them in this purpose.

Templates are divided into the following categories: i) Data Management Plan, ii) Data Workflow and iii) Presentation.

3.2 Data Management Plan templates

In modern projects, especially In H2020, a Data Management Plan has been delivered in order to describe the data that you are going to collect and how you are going to manage it. For example, in ACTION, a deliverable⁴ has been sent to explain how the pilots are going to deal with their data.

In a previous version of the deliverable (D4.2), a new tool was developed to help CS projects to generate DMPs. This tool was based on a questionnaire that explored the requirements of the users. Questions were expressed in a language more understandable for non-expert users. Based on the answers of this questionnaire, a DMP was generated in DOCX and PDF format. This DMP could be used as a template for creating the final version of the DMP. In fact, this template has been used by our pilots to generate their DMPs.

During the last year, a new tool to create DMPs was released by OpenAire, Argos. Argos is an online tool to manage the lifecycle of DMPs. It includes a lot of documentation to support users and services to integrate external systems like orcid. DMPs can be exported in PDF format, following the template of H2020 projects, and can be deposited on Zenodo.

⁴ D3.1 Data Management Plan

For this purpose, no new features were added to the DMP creator tool. In our interviews with the pilots, we detected that some questions of a DMP were difficult to answer for them. In concrete, those questions related with the license to use and how to assure the quality of the data.

As a guide, ACTION prepared two surveys based on Coney to help users. In these surveys, users will answer different questions presented by Coney, building a path of answers and finally, Coney will give users a possible answer. The two surveys can be found here:

- License:
<https://survey.actionproject.eu/chat/?data=U2FsdGVkX19Ldlc%2Fbc4fl2Biq3iPGKid0cL0gmSHhMo1zsCeFm18SVSXluPsg0sHnFzFoFYxd2YHjl7BA%2F23YxCoZxkSvOnHRLtwWITsRtg%3D>
- Data quality:
<https://survey.actionproject.eu/chat/?data=U2FsdGVkX1%20%2FprsbliHkFUMHxhi5gouNtZzpxe%2FloWlnLEqT1M8Up6XH7J2hM2fJGOdAsciOaSMlpHlyD5kiRE3gqi9IW3DKIRKyWemLbFI%3D>

3.3 *Data Management Workflow templates*

A DMP defines the strategy of your project's data management, but does not implement the process necessary to collect your data and to generate the results. In deliverable - D4.1 Conceptual Architecture - the steps of the participatory science lifecycle on a CS project were analyzed, generating a figure as a result. Figure 4 depicts the steps needed and includes: i) Collect data, iii) pre-process and curate data, iv) analyze data, v) interpreter data and vi) share results.

In CS projects, these steps are normally executed manually by the CS project team. Normally this does not pose a problem in projects that are designed for a specific campaign. This situation changes in projects that are based on continuous campaigns along the time. In these cases, a certain grade of automatization is required to assure the sustainability of the project.

The objective of this template is to create an environment where simple workflows can be executed in a standard environment. To achieve this objective, the following steps are necessary:

- A declarative format language to allow us to specify workflows. The Common Workflow Language⁵ is our best candidate because it is widely used (to describe workflows) and many tools support it to define and execute workflows.
- Design and implement a set of task's operations to execute a generic workflow to manage data in CS projects
- Deploy a platform to execute these workflows.
- Similar to the DMP situation, CWL is a language that is not easy for non-experts to use. So, as with the DMPs, the solution found was to create a tool to facilitate this work.

⁵ <https://www.commonwl.org/>

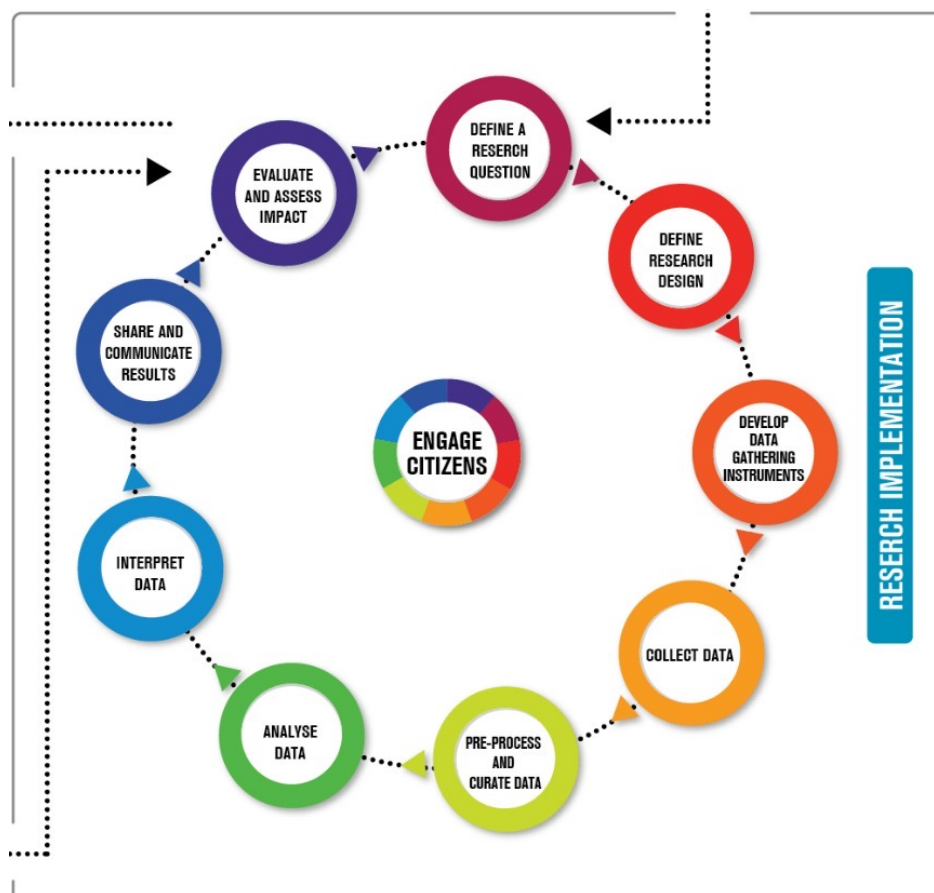


Fig 3. Participatory Science Lifecycle (data section)

3.3.1 Common Workflow Language

CWL or Common Workflow Language is a specification to describe workflows. CWL is described in JSON or YAML format. To execute a workflow, two files are required: i) A file to describe the workflow (with the extension CWL) and ii) a file with the inputs of the workflow (yml extension).

The main class to execute bash commands (scripts of Linux) is *CommandLineTool*. The following description shows how to remove the third column from a CSV file.

remove_column.cwl

```
cwlVersion: v1.0
class: CommandLineTool
baseCommand: cut
inputs:
  delimiter_flag:
    type: string
```



```

inputBinding:
  position: 1
  prefix: -d
  separate: false
remove_flag:
  type: input
inputBinding:
  position: 2
  prefix: -f
  separate: false
csv_file:
  type: File?
inputBinding:
  prefix: --file=
  separate: false
  position: 3

```

```

outputs: []

```

remove column.yml

```

delimiter_flag: ,
remove_flag: 3
csv_file:
  class: File
  path: dataset.csv
outputs: []

```

With this configuration, the CWL executor will execute the following command that remove the fourth column of the CSV file “dataset.csv”

```
>cut -d, -f4 dataset.csv
```

At this point, a configuration of a workflow step has been defined. Note that these recipes of CWL are defined in the template. The file with yml extension is the only file generated because it is the instantiation of the recipe for our project.

To build a workflow, it is necessary to connect different steps, composed by different CWL recipes. This can be done also with another description represented in a CWL file. Imagine we need to download data from epicollect (street spectra project). In this case, we need to execute a wget command (note that you need to be logged as owner to download the data collected):

```
> wget https://five.epicollect.net/api/export/entries/action-street-spectra/
```

In this case, a new CWL file will be needed to describe the step of downloading data from epicollect.

download_from_epicollect.cwl

```

cwlVersion: v1.0
class: CommandLineTool
baseCommand: wget
inputs:
  user:
    type: string
    inputBinding:
      position: 1
      prefix: --user
  password:
    type: string
    inputBinding:
      position: 2
      prefix: --password
  output_file:
    type: string
    inputBinding:
      position: 3
      prefix: -o
  url:
    type: string
    inputBinding:
      position: 4
outputs: []

```

A file with the specific configuration for a project will be generated. Remember that the description file is common to all projects.

download_from_epicollect.yml

```

user: <your_user>
password: <your_password>
output_file: example.csv
url: https://five.epicollect.net/api/exports/entries/street-spectra?{format=CSV}

```

Finally, a file for the description of the workflow will be needed. The following workflow will download data from epicollect and will remove the fourth column from the CSV file.

workflow.cwl

```

cwlVersion: v1.0
class: Workflow
steps:

```

```

download:
  run: download_from_epicollect.cwl
  out: [data_file]
clean:
  run: remove_column.cwl
in:
  src: data_file
  out: [processed_file]

```

This example shows the different features provided by this language and how a workflow can be modeled by this language. The advantage in using a specification resides in that it can be used by different platforms, making the work done more reusable and maintainable.

3.3.2 Generic Workflow description

Figure 5 depicts the structure of the templates (some of them are code snippets) generated to implement workflows in ACTION.

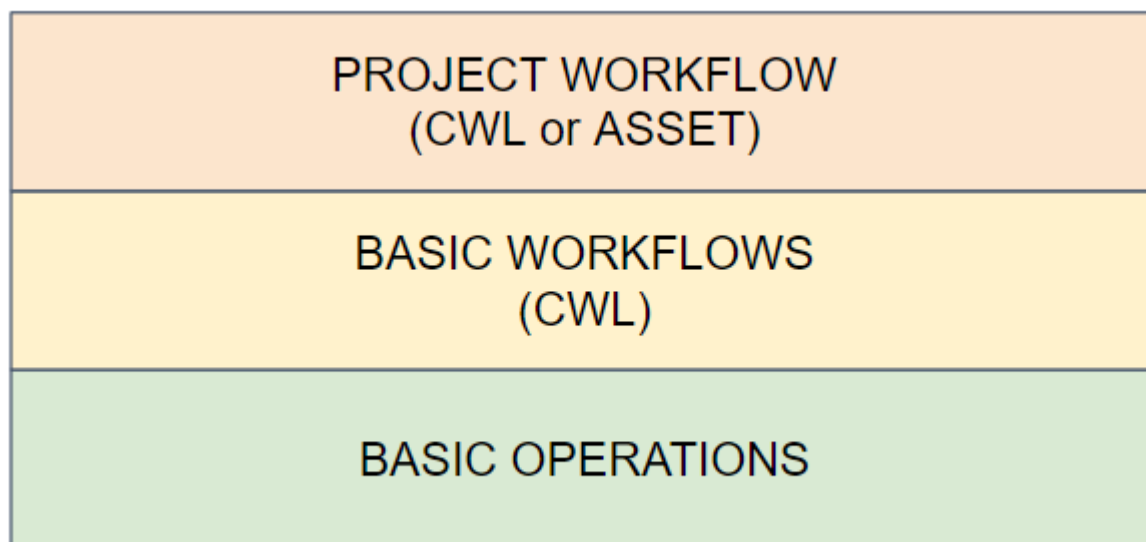


Fig 4. Generic workflow

Basic operations

Basic operations are code snippets grouped in a library to implement atomic operations to execute the workflow. The source code can be found here: <https://github.com/actionprojecteu/action-tool>.

The following table shows the most important basic operations developed:

export_ec5_observations	This operation downloads the observations from Epicollect5. A username, password and endpoint must be provided. Only logged-in users can download observations although all users can see them.
transform_ec5_observations	This operation transforms the observations modifying the format of the date to adapt it to our database
load_ec5_observations	This operation stores the observations in our databases through our API. This operation only makes sense if you want to use our platform. If the data is loaded in our system, a project will be able to use our visualization infrastructure based on Grafana. A username, password and a project must be provided to execute this task
clean_ec5_observations	This operation removes the temporal files created in the previous steps
manage_subject_sets	This operation analyzes if the project has enough observations to create a subject_set in Zooniverse.
check_enough_observations	This operation checks if your project in Zooniverse needs more observations to classify them or not. A username, password and endpoint of the Zooniverse must be provided. Also a threshold must be included. This variable indicates the minimum number of observations to upload a new subject_set.
download_from_action	This operation gets the observations stored in our database that have been previously downloaded from Epicollect5.
upload_subject_set	This operation uploads the observations to Zooniverse, transforming them first into tasks compatible with Zooniverse.
export_classifications	This operation gets the classifications made by citizens in Zooniverse. As other similar operations, it needs the username, password and endpoint of zooniverse
only_new_classifications	This operation returns only the new classifications in Zooniverse. For it, use the API to access the data.
preprocess_classifications	This operation removes some metadata fields returned by Zooniverse that are not relevant.
aggregate_classifications	This operation builds an aggregate value of the classifications based on the different classifications done by citizens in Zooniverse



publish_aggregate_csv	Publish in Zenodo the file with the aggregated values
publish_individual_csv	Publish in Zenodo the file with the raw data extracted from Zooniverse

Basic workflows

Once the atomic operations have been defined, CWL templates are built on the top of these operations. This templates can be found in this repository⁶. Operations are groped in more complex workflows. Our generic template has been divided into three specialized workflows that can be executed in different time framings.

Epicollect5 data collecting workflow is in charge of downloading results from the Epicollect5 project selected, transforming the data and storing them in our database using the API. Figure 6 shows the diagram in Apache Airflow of our workflow.

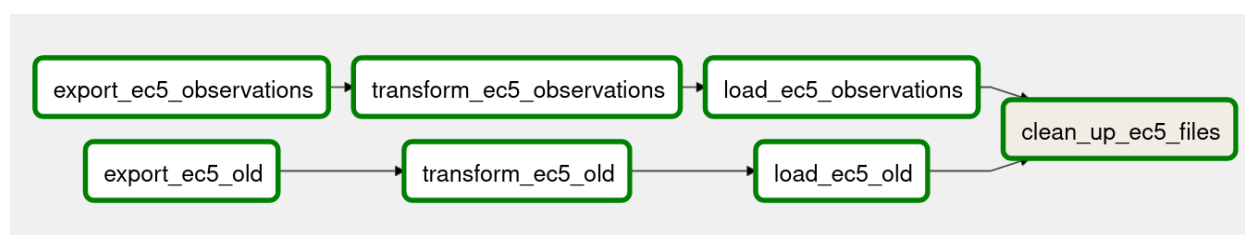


Fig 5. Epicollect5 data collecting workflow

Zooniverse feeding workflow creates tasks in Zooniverse based on the data collected in an Epicollect5 project. If a non Epicollect5 project wants to deploy this workflow, it should load the observations in our database, using our API and the format of metadata used in Epicollect5. Figure 7 shows the diagram in Apache Airflow of our workflow.

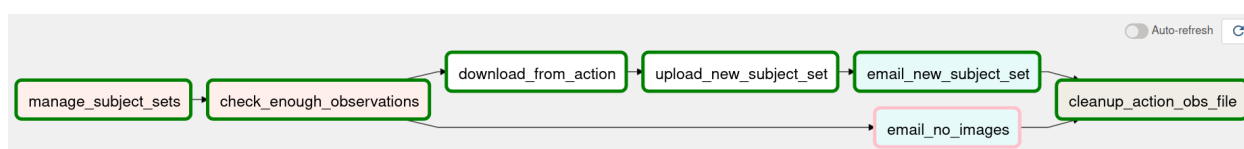


Fig 6. Zooniverse feeding diagram workflow

Aggregation workflow aggregates the classifications done by citizens in the different tasks

Project workflow

⁶ <https://github.com/actionprojecteu/workflows>

The project workflow allows defining a workflow for a project. A template of a generic workflow has been created to facilitate its customization to other projects. This workflow will be automatically divided in basic workflows and operations depending on the requirements. A diagram of this workflow can be seen in Figure 8.

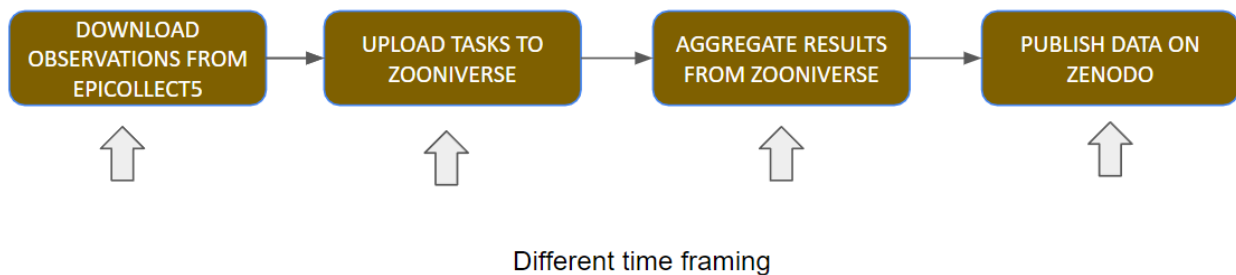


Fig 7. Generic workflow

The tasks of this workflow are the following:

- *Download observations from Epicollect5* corresponds to the workflow Epicollect5 data collecting. This task can be added or removed depending on whether the project collects data with this application or not.
- *Upload tasks to Zooniverse* corresponds to the workflow zooniverse feeding workflow.
- *Aggregate results* corresponds to the workflow aggregation.
- *Publish data on Zenodo* corresponds to an operation to publish in our Zenodo community all our results.

3.3.3 Apache Airflow: A platform to execute workflows

Once our workflow has been created, we need an environment to execute it. Apache Airflow has been the platform selected to schedule and monitor our workflows. This platform has a graphical web interface to control the execution of your workflows and to report the results of their execution via email.

As features of the platform, we can highlight the following:

- Workflows based on acyclic graphs (without loops)
- Workflows can be built using Python or Bash commands. Also, they can be built using the CWL
- Set of operators to integrate workflows with external services such as mysql, azure, email server, http requests, etc ...
- Easy to use through its graphical interface.
- It is open source.



The platform has some mechanisms to share data among workflows or to store some relevant information for them. In our case, we highlight two of them because they are used in our implementation.

- **Connections** are places to store information needed to connect our workflows with external systems. In our case, we have connections such as endpoints, users and passwords for the platforms Epicollect5, Zooniverse and ACTION API. This is sensitive information that can not be shared and it is created when the workflow is deployed. Each project has their own connections and they are not shared between them.
- **Variables** are used to store data in the platform. They are something similar to environmental variables in operative systems. We use a variable (**read_tstamp**) to store the last value read and processed by the platform.

3.3.4 ASSET: A tool to describe workflow

In the case of CS projects, normally users have no experience of workflows so the task to create one based on the template is not an easy task. Thus, we have adapted a tool named ASSET to create sketches of workflows and configure them.

This tool generates a description file of the project workflow. This file will be uploaded to a specific directory in the Nextcloud. This directory will be created by ACTION if a project wants to use the workflows. This task must be done manually in the project set up process.

A background process deployed in the server of Airflow, transforms this file into a set of CWL files (based on the templates) that will be deployed in the Airflow Platform. Figure 9 describes the architecture of the process.

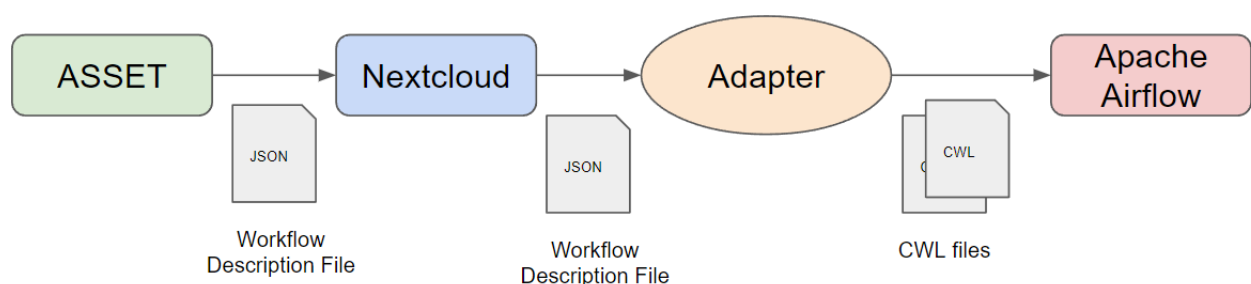


Fig 8. Generic workflow

This tool will be described in section 4 and an example about how to create a workflow in ACTION can be found in Annex I.

3.4 Presentation template

Lifecycle-aware citizen science templates

In this new version, presentation templates are integrated in the Open Data Portal. In this portal, the outcomes of each pilot can be seen previously categorized in the portal of ACTION (see D4.6 Open Data Portal). The information is provided by the research objects created for each pilot (see D4.8 Research Object catalogue). Figure 10 shows the final result of a research object in our portal.

Resources

DATASETS

Noise Maps ACTION pilot data 2020

The Noise Maps project focused on deploying a citizen science process in the Barcelona neighborhoods of Sagrada Família and the Raval to address the challenge of noise pollution. The sound data was generated between May and September 2020.

© CC-BY-4.0

AUDIO

Raval May2020

Collection of ambient urban outdoors audios from Raval

© Not Specified

SOFTWARE

AudioMoth-Firmware-SPL

This repository contains an AudioMoth firmware adaptation to calculate the Sound Pressure Level (SPL). This is based on the 1.3.0 version of AudioMoth firmware (published on AudioMoth-Project and AudioMoth-Firmware-Basic). We include the SPL library (src/spl.c and inc/spl.h) that implement all the functions related to the SPL estimation.

© MIT License

Training corpus

These sounds are used to train the data processing algorithm

© Private

PRESENTATIONS

Data visualization with Grafana

This presentation will help ACTION pilots to create their own dashboards

© CC-BY-4.0

ADDITIONAL INFORMATION

Noise Maps Dashboards

Dashboards created in Grafana to visualize data

© Not Specified

Fig 9. Presentation template

4 ASSET Tool

ASSET⁷ is a tool to create a sketching interface for scientists to quickly draft their thinking process in their day-to-day activities.

The original tool can sketch workflows composed of a large variety of tasks and tools that can be described. One of the objectives in ACTION is to adapt these tasks and tools to the specific environment that is used in CS projects. Therefore, a new version of this tool has been created to adapt the tasks to the generic workflow defined in the templates of our project. The tool can be found in the following link: <https://asset.actionproject.eu/>

Figure 11 shows the different areas of the tool.

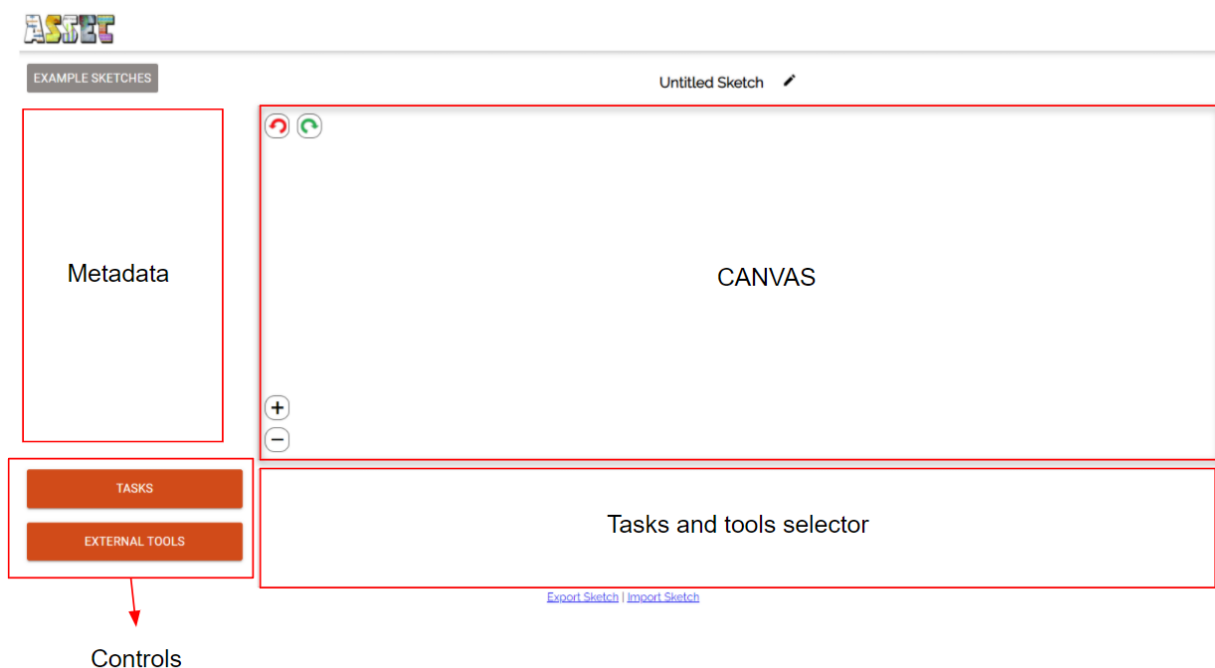


Fig 10. ASSET areas

There are four main areas in ASSET.

- Canvas where the users will add tasks and their relations to build a workflow. Users will drag these tasks from the *Tasks and tools selector area* and drop it in the canvas
- Metadata where the metadata associated with each task will be displayed and modified. Some relevant metadata are periodicity of the workflow, user and password of external tools, etc ...

⁷ <https://github.com/KnowledgeCaptureAndDiscovery/ASSET>

Lifecycle-aware citizen science templates

- Controls to display the available tasks and external platforms in the *Tasks and tools selector* area.
- Tasks and tools selector where the user will be able to select the available tasks and external platforms.

Our adaptation of ASSET provides ten different digital objects as can be seen in Figure 22. We have decided to simplify the tasks facing users, hiding the complexity from them. Thus, creating workflows is easier for them, but this sacrifices the diversity of workflows that can be created.

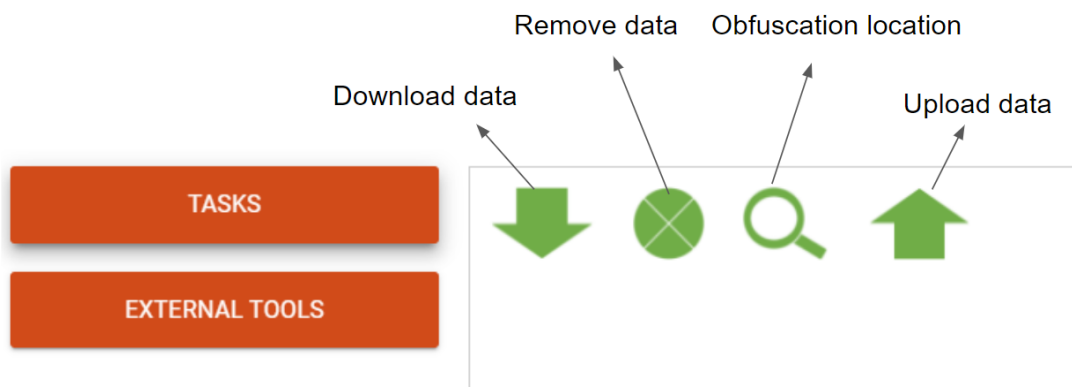


Fig 12. Digital objects in ASSET

Available tasks are:

- Download data. This task can be used to download data from Epicollect5 or Zooniverse. It will depend on the external tool selected.
- Remove data. This task is used to remove some fields that we do not want to be published. For example, if there is any kind of personal information.
- Obfuscation location. This task is used to obfuscate the location of an observation. It is based on an algorithm that removes some decimal points from the latitude and longitude coordinates.
- Upload data. This task can be used to upload data to external platforms like Zooniverse or Zenodo, depending on the external tool selected.

The available external tools are Epicollect5, Zooniverse and Zenodo.

Next, the main activities covered by ASSET will be described.

Adding an element to your workflow

To add an element to the canvas, the users must click in the tasks button and select a task

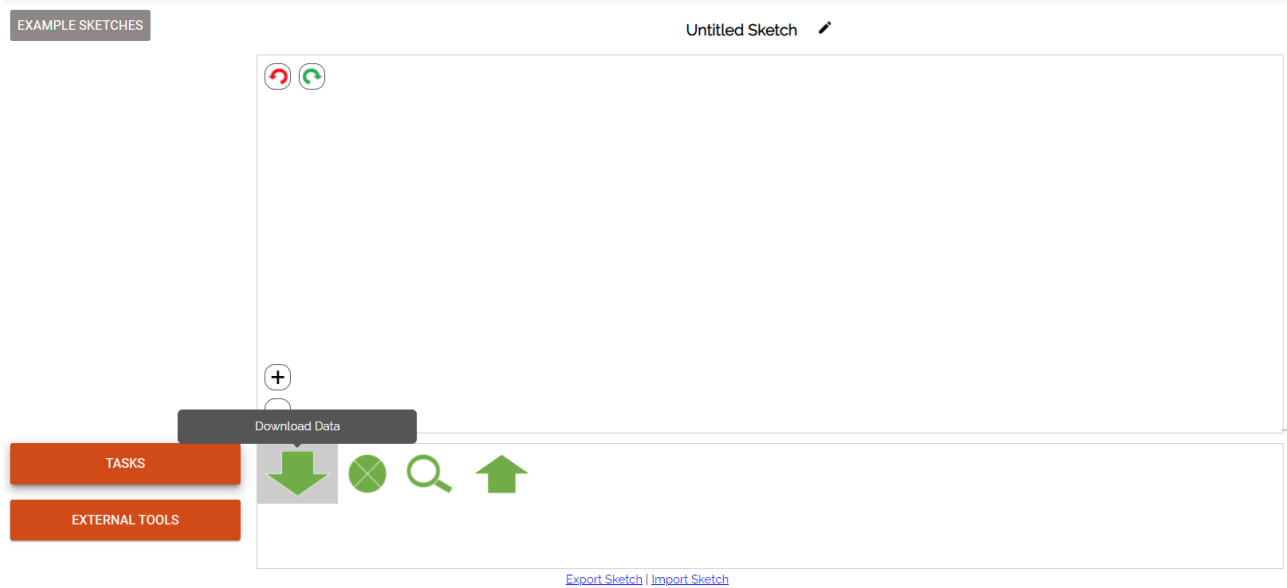


Fig 12. Select a task

Then, the user will drag the element to the canvas

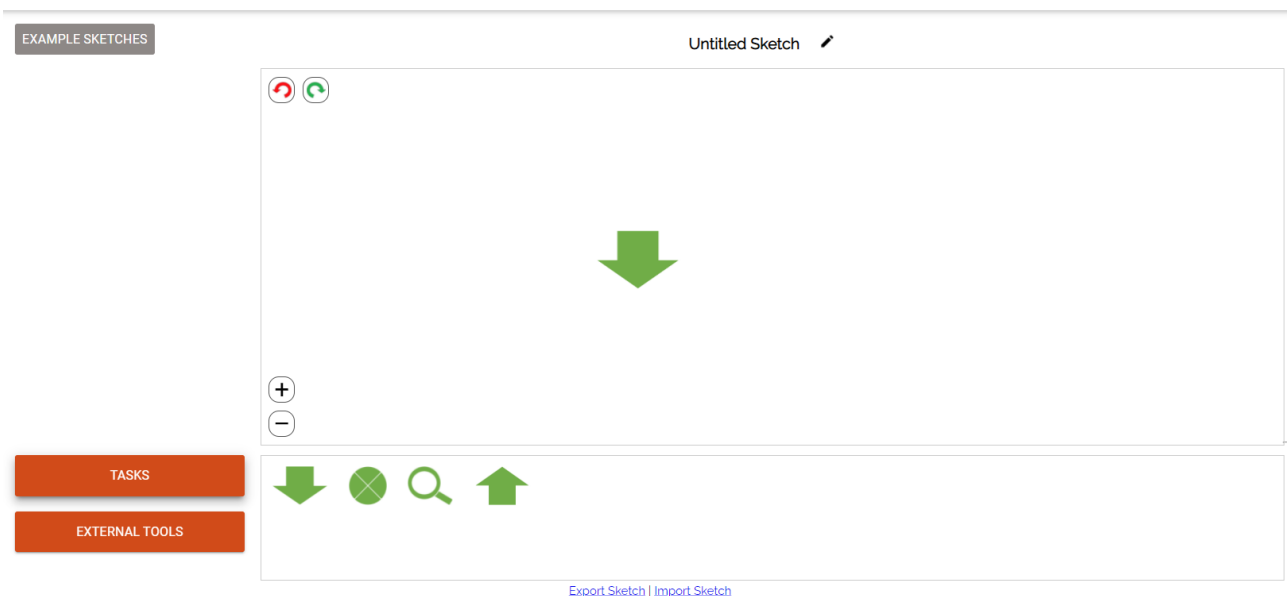


Fig 13. Add a task to the canvas

Describing an element in your sketch

Click on the element in the canvas and a table of metadata will appear in the metadata area (left side). Some metadata relevant for the workflow are the **start date** of execution (if we do not specify a date, the system will assume the date when the workflow was created) and the **periodicity** of execution (in days). Note that the tasks can have different execution periods, for

Lifecycle-aware citizen science templates

example, I can download data from Epicollect every 30 days, and download data from Zooniverse every 5 days and then publish my results every 60 days.

EXAMPLE SKETCHES

Download Data


Name	Download data	X
Description	Download data from my project	X
Author		X
Duration		X
Tools used		X
period	30	X
start	2020-01-01	X
Add property		

Select the task and then use the delete/backspace button on your keyboard to delete this task.


TASKS

EXTERNAL TOOLS

Untitled Sketch



Download data



[Export Sketch](#) | [Import Sketch](#)

Fig 14. Describe an element

Deleting an element in your sketch

Select an element (an arrow or a task in the canvas) and press the "delete" key on your keyboard.

Connecting two elements in your sketch

First, add the two elements to the canvas and double click on one of them. A pink area will display on the element. This will define the sense of the arrow.

EXAMPLE SKETCHES

Download Data


Name	Download data	X
Description	Download data from my project	X
Author		X
Duration		X
Tools used		X
period	30	X
start	2020-01-01	X
Add property		

Select the task and then use the delete/backspace button on your keyboard to delete this task.


TASKS


EXTERNAL TOOLS

Untitled Sketch



Download data





[Export Sketch](#) | [Import Sketch](#)

Fig 15. Define link between elements

Then click on the other element. An arrow will be drawn between the two elements.

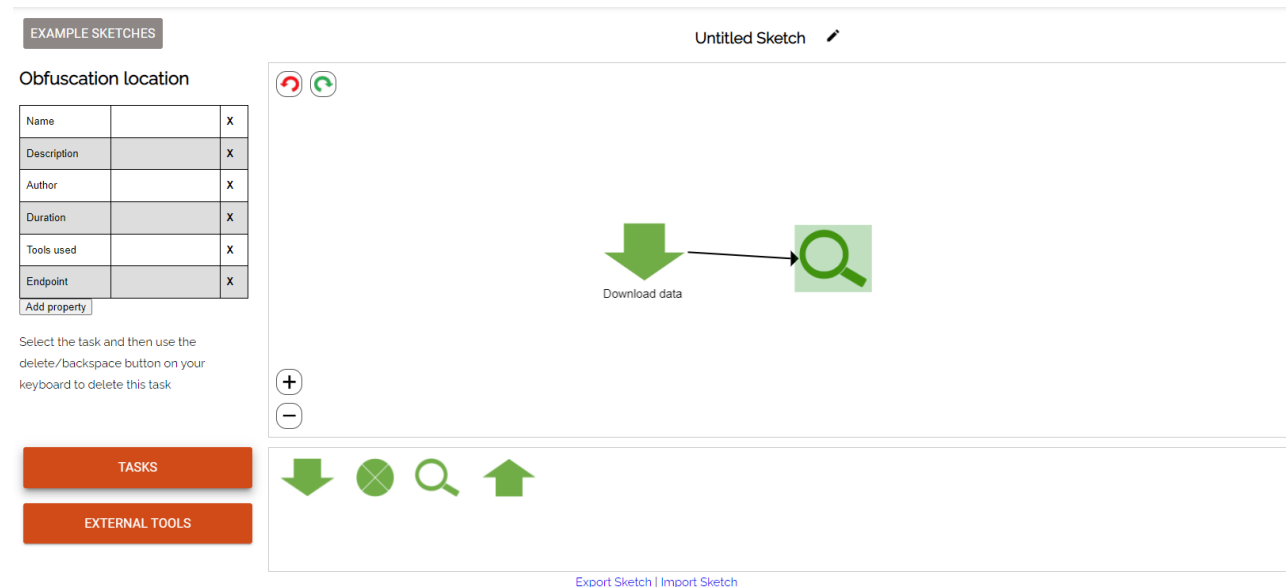


Fig 16. Link two elements

The relation can be named and described if you click on the arrow.

Describing which external tools can be used.

Click on the external tools button. This will display the available tools.



Fig 17. Available external tools



Lifecycle-aware citizen science templates

Drag the tool from the list box and drop it on the task desired. The selected tool will be displayed under the task icon. Also, the tool will appear as metadata in the metadata area (field tools used)

EXAMPLE SKETCHES

Download Data

Name	Download data	X
Description	Download data from my project	X
Author		X
Duration		X
Tools used	EPICOLLECT5	X
period	30	X
start	2020-01-01	X
Tools used		X

Choose a Tool

Add property

TASKS

EXTERNAL TOOLS

Untitled Sketch

[Export Sketch](#) | [Import Sketch](#)

Fig 18. Add a tool to a task

Tools can be configured with some relevant information for their use. Select the tool displaying the list of Choose a Tool.

Name	Download data	X
Description	Download data from my project	X
Author		X
Duration		X
Tools used	EPICOLLECT5	X
period	30	X
start	2020-01-01	X
Tools used		X

Choose a Tool

Choose a Tool

EPICOLLECT5

Add property

TASKS

EXTERNAL TOOLS

Fig 19. Select a tool

The metadata table will be displayed on the left side of the page.

EXAMPLE SKETCHES

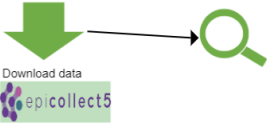
Tool: EPICOLLECT5

Name		X
Description	Download data from epicollect5	X
Author		X
Duration		X
Tools used		X
endpoint	the endpoint of your project in Epicollect5	X
username	The username of your project in Epicollect5	X
password	The password of your project in Epicollect5	X

TASKS

EXTERNAL TOOLS

Untitled Sketch





[Export Sketch](#) | [Import Sketch](#)

Fig 20. Configure a tool

Some information that the user must provide in the case of downloading data from Epicollect5 are:

i) the endpoint of the project (url), the username in Epicollect5 and iii) the password in Epicollect5

Exporting your sketch

Click on the "Export Sketch" link on the lower part of the screen. A JSON file will be downloaded. This file will be copied to your Nextcloud directory in case you want to execute your workflow. Note that this application doesn't have a backend to store sketches, so if you want to preserve your workflow, do not forget to export it every time you make a modification on it.

Importing a sketch

Click on the "Import Sketch" link on the lower part of the screen and choose a previously exported sketch from your file system.

5 CONCLUSIONS

In the previous version of the deliverable, a guideline to create CS projects in two popular platforms: Epicollect and Zooniverse was released. A practical example of Street Spectra was described. This guideline has helped the new cohort of pilots in their projects, especially Epicollect5 to collect data.

In this version, our effort has been directed to connect both platforms, adding the publication of results in Zenodo. This can be done implementing workflows. Therefore, ACTION has created/deployed: i) a set of templates to implement recurring tasks in data management (applied to CS projects), ii) a platform to schedule and to monitor workflows (Apache Airflow) and iii) a tool to create these workflows.

This template has been tested with the Street Spectra application. These workflows allow them to automate data management processes, making our projects more sustainable because the supervision of them decreases significantly.

Regarding the DMP tool creator, the new version incorporates only some surveys created by the Coney application. These surveys help users to select a license for their data, and to fill in the section of data quality assurance in a DMP. No additional features were implemented in the tool because most of the functionalities are available in the tool Argos, released the past year for OpenAire. Because of that, the tool was used to generate templates of DMPs that were used to fill in the sections of Argos. From WP4 we give support to pilots organizing some webinars and helpdesk events.

ANNEX I: Guideline to create workflows in ACTION

Lifecycle-aware citizen science templates

The objective of this guideline is to help users to create ACTION workflows with the tool ASSET with the workflow implemented in Street Spectra

First, we have to decide our workflow, the task that we want to automatize.

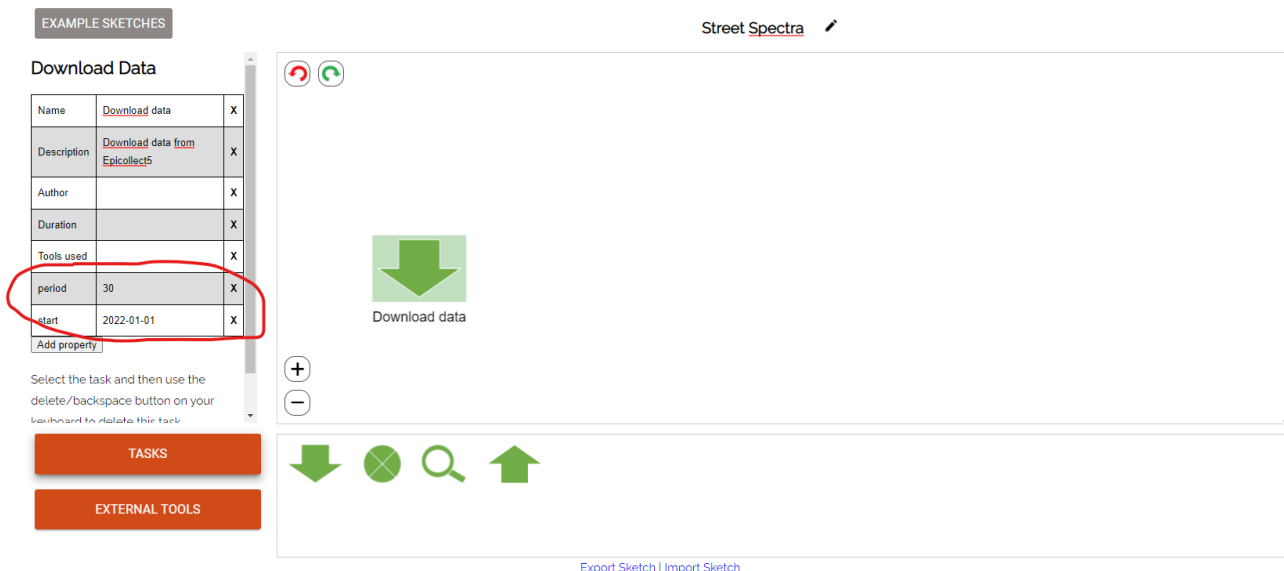
In our case, we will start from the generic workflow. The steps of our workflow are:

1. Download periodically (30 days) data from our Epicollect5 project
2. Remove the field nick from the data downloaded
3. Publish the data in Zenodo
4. Feed our project in Zooniverse with a new subject set built when there remain 100 tasks for classifying. Data will be obtained from the observations created in Epicollect5.
5. Download the classifications from Zooniverse
6. Publish the classifications in Zenodo

Once the tasks are defined, we start to add them in our canvas.

First, click on the **tasks button** and select download. Add as metadata:

- A scheduling period: 30 days
- A start date: 2022-01-01



The screenshot shows the 'Street Spectra' interface. On the left, under 'EXAMPLE SKETCHES', there is a 'Download Data' task configuration table. The table has columns for Name, Description, Author, Duration, Tools used, period, and start. The 'period' and 'start' rows are highlighted with a red circle. Below the table are buttons for 'TASKS' and 'EXTERNAL TOOLS'. The main canvas on the right shows a large green arrow pointing down with the text 'Download data' below it. At the bottom of the canvas, there are icons for adding, deleting, and searching tasks, and a bar with 'Export Sketch' and 'Import Sketch' links.

Name	Description	Author	Duration	Tools used	period	start
Download data	Download data from Epicollect5				30	2022-01-01

Then, click on the external tools button and add the tool Epicollect5. Add as metadata:

- endpoint: <https://five.epicollect.net/project/action-street-spectra>
- username: Epicollect5 username
- password: Epicollect5 password



Lifecycle-aware citizen science templates

EXAMPLE SKETCHES

Tool: EPICOLLECT5

Name		X
Description		X
Author		X
Duration		X
Tools used		X
endpoint	https://five.epicollect.net/project/action-street-spectra	X
username	<your_username>	X
password	<your_password>	X

Delete | Add property

Street Spectra


Download data

[Export Sketch](#) | [Import Sketch](#)

Add a task to remove the field *nick* from the dataset and link it with the download data task. Add the metadata field=*nick* to remove this field.

EXAMPLE SKETCHES

Remove Data

Name	Remove nick	X
Description		X
Author		X
Duration		X
Tools used		X
field	nick	X

Add property


Select the task and then use the delete/backspace button on your keyboard to delete this task

Street Spectra


Download data



Remove nick

[Export Sketch](#) | [Import Sketch](#)

Add the task upload to publish the data. It is not necessary to include the period of time and the start date because we apply the same values defined in the download data.

Lifecycle-aware citizen science templates

EXAMPLE SKETCHES


Upload Data

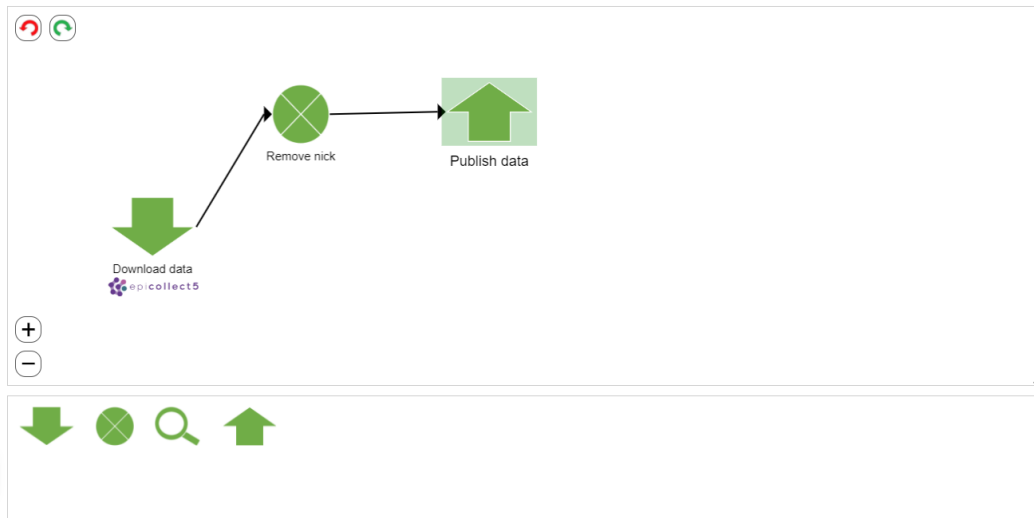
Name	Publish data	X
Description	Data generated by Zenodo	X
Author	Esteban Gonzalez	X
Duration		X
Tools used		X
Add property		

Select the task and then use the delete/backspace button on your keyboard to delete this task

TASKS

EXTERNAL TOOLS

Street Spectra 



Now, we add the tool zenodo in the publish data task. As a metadata, we add:

- Zenodo community
- Zenodo token for authentication
- License
- Funder (Grant Agreement)
- Authors
- Keywords

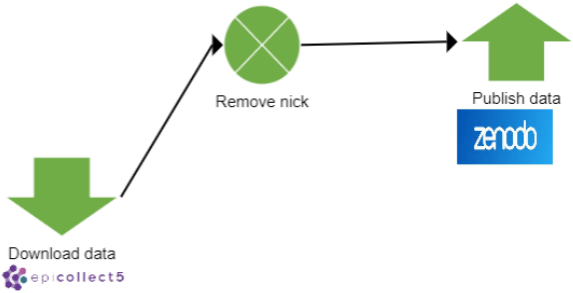
EXAMPLE SKETCHES

name	zenodo	
Description	Publish raw data in zenodo	X
Author	Esteban González	X
Duration		X
Tools used		X
keyword	light pollution	X
keyword	lampposts	X
community	street_spectra	X
token	<your token>	X
license	CC-BY-4.0	X
funder	824603	X

TASKS

EXTERNAL TOOLS

Street Spectra



epicollect5

ZOONIVERSE

zenodo

Export Sketch | Import Sketch

Now, we create a new branch in our workflow. We are going to upload new tasks into Zooniverse. Remember that these tasks are complex workflows, for example, when we download data from Epicollect5, it means the execution of operations such as downloading data, transforming data and loading data to our databases. In the case of uploading data to zooniverse, this involves the creation of tasks, the creation of new subject sets, etc. The period of execution has been established in 1 day. It means that the workflow will be executed every day.

EXAMPLE SKETCHES

Street Spectra

Upload Data

Name	<u>Upload data</u>	X
Description		X
Author		X
Duration		X
Tools used		X
<u>period</u>	1	X

Add property

Select the task and then use the delete/backspace button on your keyboard to delete this task

TASKS

EXTERNAL TOOLS



+

-

TASKS

EXTERNAL TOOLS

[Export Sketch](#) | [Import Sketch](#)

Then, we include the zooniverse tool. Add as metadata:

- **endpoint** in Zooniverse: <https://www.zooniverse.org/projects/actionprojecteu/street-spectra/>
- **username** in Zooniverse
- **password** in Zooniverse
- **remaining_tasks**: 200. This parameter is used as a threshold. When the number of tasks classified is lower than this quantity, the system creates a new subject set.
- **subject_set_size**: 200. This parameter establishes the size of the collection of tasks to be classified.

EXAMPLE SKETCHES

Street Spectra

Author	
Duration	
Tools used	
endpoint	https://www.zooniverse.org/projects/actionprojecteu/street-spectra/
username	<your_username>
password	<your_password>
remaining_tasks	50
subject_set_size	200

Delete
Add property



```

graph LR
    Z[ZOO@NIVERSE] -- Upload data --> E[epicollect5]
    E -- Download data --> R((Remove nick))
    R --> P[zenodo Publish data]
            
```

TASKS

EXTERNAL TOOLS

[Export Sketch](#) | [Import Sketch](#)

Remember to link the task Upload data with Download data from Epicollect5. The next step is to add a task to download data from Zooniverse. The objective is to publish our classifications in Zenodo.

As the previous steps, add the task download, and the tool Zooniverse. In this case, add the metadata related with the Zooniverse endpoint, username and password. Do not forget to connect this task with the previous.



Lifecycle-aware citizen science templates

EXAMPLE SKETCHES

Tool: ZOONIVERSE

Name		X
Description		X
Author		X
Duration		X
Tools used		X
endpoint	https://www.zooniverse.org/projects/actionprojecteu/street-spectra/	X
username	<your_username>	X
password	<your_password>	X

Delete Add property

TASKS

EXTERNAL TOOLS

Street Spectra

```
graph TD; A[Download data epicollect5] --> B((Remove nick)); A --> C[Upload data ZOONIVERSE]; B --> D[Publish data zenodo]; C --> E[Download data ZOONIVERSE];
```

epicollect5 ZOONIVERSE zenodo

[Export Sketch](#) | [Import Sketch](#)

Finally, we need a new task to upload our classification data to Zenodo. The application is similar to previous steps. In this case, we have to set up the period to 30 because we are interested in publishing our data every 30 days.

EXAMPLE SKETCHES

Upload Data

Name	Publish data	X
Description		X
Author		X
Duration		X
Tools used	ZENODO	X
Endpoint		X
Tools used		X

Choose a Tool

Add property

Select the task and then use the

TASKS

EXTERNAL TOOLS

Street Spectra

```
graph TD; A[Download data epicollect5] --> B((Remove nick)); A --> C[Upload data ZOONIVERSE]; B --> D[Publish data zenodo]; C --> E[Download data ZOONIVERSE]; E --> F[Publish data zenodo];
```

epicollect5 ZOONIVERSE zenodo

[Export Sketch](#) | [Import Sketch](#)

Lifecycle-aware citizen science templates

At this point, the workflow is defined. To execute it, click on the **Export Sketch** link. This action will download a JSON file with the description of the workflow. Remember to upload it in Nextcloud to execute it.

EXAMPLE SKETCHES

Street Spectra

Upload Data

Name	Publish data	X
Description		X
Author		X
Duration		X
Tools used	ZENODO	X
Endpoint		X
Tools used		X

Choose a Tool ▾

Add property

Select the task and then use the

TASKS

EXTERNAL TOOLS

Download data

epicollect5

Remove nick

Publish data

zenodo

Upload data

zoöniverse

Download data

zoöniverse

Publish data

zenodo

epicollect5

zoöniverse

zenodo

Export Sketch

Import Sketch