



## Deliverable D3.1 – Mock-up of the Toolbox as input for the request for proposals (WP4)

### WP3 – Regional CRA prototypes and operationalization

Version 1.1 | September 2024

(Grant Agreement 101093864)

HORIZON-MISS-2021-CLIMA-02-01 - Development of climate change risk assessments in European regions and communities based on a transparent and harmonised Climate Risk Assessment approach

## Document history

Deliverable Title	Mock-up of the Toolbox as input for the request for proposals WP4
Brief Description	The deliverable describes the components of the prototype of the CLIMAAX toolbox, following the specifications outlined in Deliverable D2.1. Furthermore, it provides a first toolbox user guideline and outlines the plans for future developments.
WP number	WP3
Lead Beneficiary	European Centre For Medium-range Weather Forecasts – ECMWF
Author(s)	Milana Vučković, Fredrik Wetterhall – ECMWF Lena Reimann – IVM Ted Buskop – Deltares
Deliverable Due Date	30/09/2023
Actual Delivery Date	30/11/2024
Nature of the Deliverable	P – Prototype; D – Demonstrator
Dissemination Level	PU - Public

Date	Ver.	Contributors	Comment
15/09/2023	0.1	Milana Vuckovic, Fredrik Wetterhall	First version ready for internal review
22/09/2023	0.1rev	Bart van den Hurk, Maurizio Mazzoleni, Lena Reimann, Ted Buskop	First internal review
28/09/2023	0.9	Milana Vuckovic, Fredrik Wetterhall	Reviewed version ready
29/09/2023	1.0	First finished version ready	Reviewed by project management
30/11/2024	1.1	Fredrik Wetterhall	Revision of rejected deliverable

## Table of contents

Document history .....	2
Table of contents .....	3
Acronym list .....	5
List of figures .....	6
Executive Summary .....	7
1 Introduction .....	8
2 CLIMAAX toolbox functionality .....	10
2.1 The user journey of a CRA .....	10
2.2 Initial stage of the toolbox implementation and guiding principles .....	11
3 Examples and usage guidelines .....	12
3.1 One example of a user journey .....	12
3.1.1 Setting up the environment .....	12
3.1.2 Access the needed data .....	13
3.1.3 Calculate the hazard .....	13
3.1.4 Calculate the risk .....	14
3.1.5 Calculate the future risk due to climate change .....	16
3.1.6 Extra resources to aid in the user journey .....	16
4 Toolbox technical architecture .....	17
4.1 Implementing the Jupyter environment .....	17
4.2 Workflows GitHub Repositories .....	20
4.2.1 Workflow notebook file .....	21
4.2.2 Computational environment file .....	21
4.2.3 Readme file and licence .....	22
4.2.4 Licence .....	22
4.2.5 Additional files .....	22
4.3 Jupyter Book description .....	22
4.3.1 Climate Risk Assessment book structure .....	23
4.4 Adding new workflows and peer-review process .....	24
4.4.1 CRA book GitHub repository structure .....	25
4.4.2 Automatic Building and Deploying using GitHub Actions .....	25
4.4.3 CRA book and workflows repositories connection .....	26
4.5 JupyterHub Integration .....	27

4.6	User authentication.....	28
4.7	Binder integration.....	29
4.8	Interactive dashboards using Voila .....	30
5	Future developments and conclusions.....	33
5.1	Collaboration .....	33
5.2	Technical improvements .....	33
5.3	Content development.....	33
5.4	Planned developments for the Toolbox Version 2 (M16) .....	33
5.5	Conclusions.....	34
6	Glossary.....	35
7	References .....	37

## Acronym list

API – Application Program Interface

CADS – Climate and Atmosphere Data Store

CI – Continuous integration

CMIP6 - WCRP Coupled Model Intercomparison Project (Phase 6)

CORDEX – Coordinated Regional Climate Downscaling EXperiment

CRA – Climate Risk Assessment

DOI – Digital Object Identifier

ECMWF – European Centre for Medium-Range Weather Forecasts

EU – European Union`

FAIR – Findable, accessible, interoperable and reusable

IPCC - Intergovernmental Panel on Climate Change

JRC – Joint Research Centre of the European Commission

OGC – Open Geospatial Consortium

RDH – The Joint Research Centre's Risk data hub

WCRP – World Climate Research Programme

## List of figures

Figure 1. Conceptual workflow of a CRA using the CLIMAAX toolbox. ....	10
Figure 2. Screenshot of instructions how to load the libraries needed for the specific workflow .....	12
Figure 3. Example of a workflow in the Climate Risk Assessment book - view of the data description with links to the data origins .....	13
Figure 4. Example of a cut-out of a specific region of a land-use map. ....	14
Figure 5. Vulnerability curves for flood damages for different land covers.....	15
Figure 6. Flood risk in Zilina expressed losses in EUR due to a flood event of 200-year return period.....	15
Figure 7. Future climate projects of the CMIP6 global climate models for three different scenarios.....	16
Figure 8. Mapping of the technologies used on the toolbox onto the user story of the CRA bakery. ....	17
Figure 9. Schematic representation of the CLIMAAX toolbox elements. The components “book”, “binder”, “hub” and “voila” are described in the main text. The main figure denotes the Jupyter workflows, also known as Jupyter notebooks.....	18
Figure 10. Structure of the CLIMAAX GitHub organisation space with (currently) six hazard workflows and one generic CRA code workflow (labelled “crabook”). ....	20
Figure 11. Climate Risk Assessment book entry page accessible from <a href="https://toolbox.climaax.eu">https://toolbox.climaax.eu</a> .....	23
Figure 12. Workflow notebook rendered in CRA book with notebook structure on the right and book structure on the left. ....	24
Figure 13. Links to repositories with Jupyter notebook workflows inside the CRA book repository .....	27
Figure 14. View of user's space in JupyterHub with the folders containing notebooks material.....	29
Figure 15. Schematic image of the different ways the workflow notebooks are used: They are hosted on GitHub and imported in the JupyterBook, they are automatically copied to the JupyterHub, and can be executed on Binder or downloaded for work on local computer .....	31
Figure 16. Links to different downloading or executing options available for each workflow notebook .....	32

## Executive Summary

This document identifies and describes all the CLIMAAX toolbox components which will be used for the climate risk assessment by the pilot project partners and participants of the Open Call. The main design target of this work is to create a modular and inclusive architecture which can efficiently integrate with the climate risk assessment framework defined in WP1 and provide the tools for the technical implementation of climate risk assessments by the participants. The core content of this deliverable is a demonstration of the mock-up of the Toolbox, which will serve as an input for the call for proposals designed in WP4. The document starts with an overview of the main components of the toolbox and guiding principles. Then we explain in an example a user journey of using the toolbox in a climate risk assessment. The deliverable then describes in detail the CLIMAAX toolbox functionality and technical architecture and its components. Finally, the document concludes with the description of plans for further developments.

The skeleton of the implemented CRA CLIMAAX toolbox that this document is based on is visible here: <https://toolbox.climaax.eu/>. The Jupyter notebooks are available on the CLIMAAX GitHub: <https://github.com/orgs/CLIMAAX/dashboard>. Note that the CRA toolbox is currently under intensive development and we do not foresee to provide a frozen version until we have delivered version 1 of the toolbox in December 2023.

## 1 Introduction

In 2021, the European Commission introduced a mission-oriented research and innovation (R&I) programme to tackle significant climate change challenges confronted by the EU and the global community. This approach fosters targeted efforts involving regional and local authorities supplementing the conventional approaches. Climate adaptation at regional scales is a key challenge, given its conceptual and practical obstacles like financial constraints, limited development capacity and scarcity of well-tested resources for effective climate risk assessment. To address these challenges CLIMAAX received funding to promote the execution of regional Climate Risk Assessments (CRA) feeding into disaster risk management and climate adaptation strategies, contributing to climate literacy and risk-informed decision-making in a wide range of European regions and communities<sup>1</sup>.

To achieve its goal, the CLIMAAX project will create a cohesive framework for regional CRAs, and a practice-oriented toolbox supporting their implementation. The framework and practical tools will encourage standardized best practices and enable subnational authorities to investigate climate-related risks and identify adaptation opportunities. This approach aims to foster a culture of climate risk intelligence and catalyse climate, adaptation, and resilience services ecosystems.

This deliverable is a description of an early demonstrator version (a “mock-up”) of the CLIMAAX toolbox, derived from specifications defined in deliverable D2.1 documenting the practice of disaster and climate risk assessments within pilot regions. The mock-up version of the CLIMAAX toolbox is being developed as part of task **T3.1: Co-development of the Operational Toolbox facility**, as a collaboration of several project partners. Four main requirements of the toolbox were determined:

- Flexible tool to combine available reference data and tools with own data and custom risk assessment procedures based on open code and technical guidelines,
- Offer a range of regional risk assessment approaches across Europe, including utilization of high spatial resolution information,
- Assess risks posed by a variety of climate hazards under present and future conditions,
- Ability to analyse different exposed elements (e.g., population, infrastructure, ecosystems, buildings) and their vulnerability under present and future conditions.

The toolbox is designed to be an adjustable and open online resource to conduct climate risk assessments. It contains tools and guidance on how to extract information and conduct analysis from a range of included hazard, vulnerability and exposure data, embedded in flexible yet state-of-the-art climate risk assessment open-source tools and procedures (so-called ‘risk workflows’). It is intended to be useful for multiple environmental and societal domains and levels of experience.

Around the toolbox CLIMAAX will build and connect a community of users and contributors, not only from pilot partners and beneficiaries from the open call, but any scientist or enthusiast from the climate risk assessment community. This is achieved by organising periodic “show-and-tell” sessions for the participants and establishing a review process for contributing external risk workflows to the collection of available

---

<sup>1</sup> <https://www.climaax.eu/>



procedures, putting open science into practice towards collaborative, reusable and transparent climate risk assessment tools.

**Section 2** of this deliverable presents the **overview of the concepts** behind the CLIMAAX toolbox, outlining the goals and specific use cases of the toolbox. It also gives a high-level overview of the toolbox components.

**Section 3** provides **practical examples, usage guidelines and best practices** for users to maximize its added value. It includes practical examples and step-by-step guides for running specific risk workflows.

**Section 4** provides the details of the **toolbox technical architecture**. It contains detailed technical descriptions of all the toolbox components and the connections between them.

The deliverable finishes with the description of the known limitations of the current mock-up version and **plans for future development and conclusions** in **Section 5**.

## 2 CLIMAAX toolbox functionality

This section gives an overview of the CLIMAAX toolbox as a response to the requirements described in deliverable D2.1 “Specifications for toolbox methods”. In the comprehensive analysis of requirements to conduct a climate risk assessment collected from the five CLIMAAX pilot sites a substantial variety in different regional contexts was noted (e.g., climate zones, legislation, available resources), and correspondingly also the needs varied across the pilot regions. Current risk assessment methods in the pilot sites range from simple hazard-based risk indices to comprehensive analysis of expected losses supported by high resolution local hazard, exposure and vulnerability information.

Despite the large variability in climate hazards, all sites seek help to communicate current and future climate risks and associated technical guidance for the climate risk assessment. On top of this, we anticipate that many of the participants in the open call will similarly lack the necessary information, data and experience to identify and prioritise climate risks, assess key vulnerabilities, and develop strategies to manage them. Therefore, one of the toolbox’s primary aims is to be an educational resource for the regions contributing to an overall elevated level of climate literacy.

The functionality of the toolbox is following a principle of “structured flexibility”, which means that the user will initially follow a formalised structure of the CRA workflow. These initial workflows will contain the hazard, sensitivity and vulnerability data to make initial CRA assessments albeit on a coarser resolution. However, the users will be able to modify the workflow to fit their needs, add their own data and save the workflow for further analysis and share this with the wider community.

### 2.1 The user journey of a CRA

The conceptual idea of the toolbox is depicted in Figure 1. The user is here envisioned as a baker (1) who goes to the bakery shop (2) where they can browse through the available ready-made workflows for specific risk assessments. The user then chooses a specific recipe (3) and brings it home to the kitchen (4). There the user can try out the recipe, using the available data that is supplied, bring their own data into the recipe, tailor the data as they see fit and modify the recipe to provide a new workflow resulting in a climate risk assessment specific for their needs (5). The result can be an updated risk map, data that is needed for further investigation or dashboard displaying a range of outcomes.

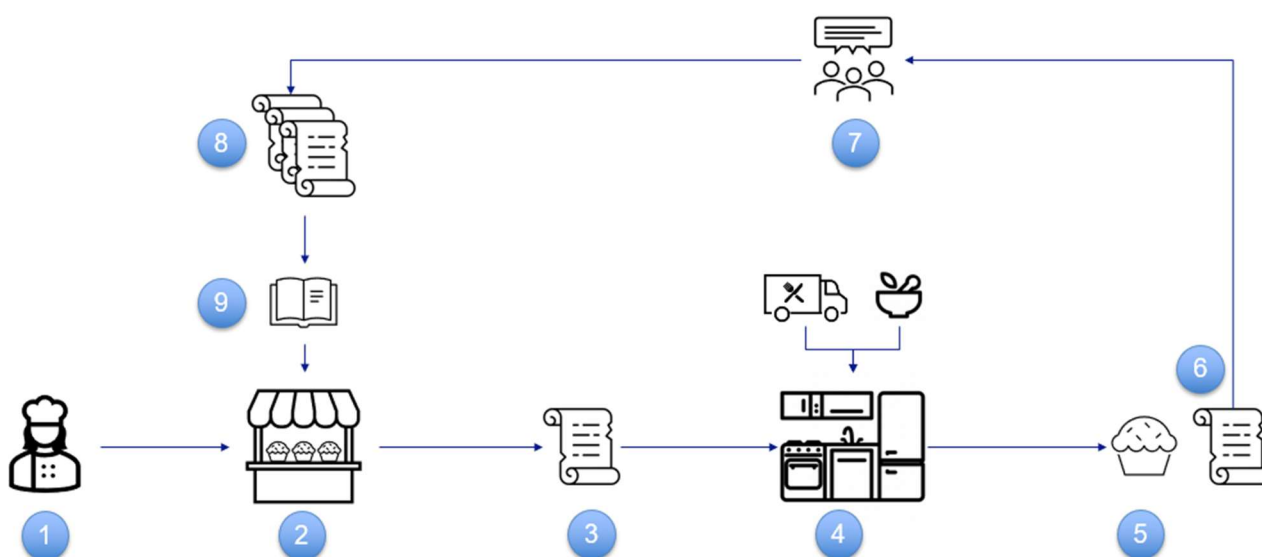


Figure 1. Conceptual workflow of a CRA using the CLIMAAX toolbox.

The next step is to take the new recipe and share it with the wider community (7). This then becomes part of the collection of new recipes (8) all presented in the common book and made available to the users in the bakery. An example of a ready-made workflow is presented in section 3. Behind all this is lies the technical infrastructure to make this possible, and this will be presented in section 4. The user will also be guided through the workflow with tutorials, videos and a FAQ to make the user journey understandable and workable. Online help sessions and a user service will provide support and update the workflows continuously.

## 2.2 Initial stage of the toolbox implementation and guiding principles

During the first conceptualisation of the toolbox in the CLIMAAX proposal phase, it was envisioned to consist of four main elements:

1. A **wiki** to serve as user guide with full description of the tools involved,
2. **Jupyter notebook** templates and examples of the workflows of case studies,
3. **Access points** to the **models, data, and tools** needed for data manipulation and visualisation as part of the climate risk assessment,
4. **Access to computational and storage** resources.

In the initial stages of the CLIMAAX project implementation some elements of this toolbox vision were changed, retaining the overall concept. The wiki is now part of the JupyterBook, not a standalone feature, to guarantee an integrated toolbox set-up. The core of the toolbox still consists of Jupyter notebook examples and workflows. The access points to data and tools are implemented as planned, but the access point to supporting models is yet to be realised designed. The toolbox will provide access to storage and computational resources if needed and asked for by the users.

Additionally, a fifth element will be added to the upcoming version of the toolbox. This will be a set of interactive hazard dashboards, based on the collection of Jupyter notebook workflows, which will hide the programming code from the users. These dashboards will serve users interested in displays without extensive technical details to respond to the need for comprehensive risk communication. To be an effective resource for regions advanced in conducting climate risk assessments, the CLIMAAX toolbox is built on FAIR principles for scientific software development (Barker et al., 2022) as well as data (Wilkinson, 2016), defined below:

- **F:** Software and its associated metadata is easy to find for both humans and machines,
- **A:** Software, and its metadata is accessible via standardised protocols,
- **I:** Software interoperates with other software by exchanging data and/or metadata, and/or through interaction via application programming interfaces (APIs), described through standards,
- **R:** Software is both usable (can be executed) and reusable (can be understood, modified, built upon, or incorporated into other software).

## 3 Examples and usage guidelines

Climate Risk Assessment book is not intended to be read from the beginning to the end. It is also not intended to transfer the knowledge in one direction, from the toolbox to the user. In addition to building the toolbox, our goal is to build a CRA community and to have the toolbox be populated with the example workflow from the real users. Note that this example assumes that the user is familiar with some of the technical aspects of the Jupyter functionality. However, you can read this section and skip the technical details, all which are explained in detail in Section 4.

### 3.1 One example of a user journey

A user who wants to do a specific climate risk assessment for their area can start from one of the workflows in the workflow collection in the CRA book. The example to illustrate the of the toolbox is a user who wants to create a map of expected flood damages over the Zilina pilot on Slovakia for different levels of flood extent, expressed as return periods. In each workflow they can read basic information about the data that is used. The basic workflow would be:

0. Setup the environment
1. Access the needed data
2. Calculate or assess the return period (hazard)
3. Combine the hazard data with exposure and vulnerability data to assess the risk
4. Assess the future climate risk

#### 3.1.1 Setting up the environment

The first step in the process is to setup the environment to run the notebooks, which consists of a number of tasks, such as load the needed code libraries and setting up the file structure. Each step in this process is described in detail in the notebook (Figure 2).

#### Load libraries

In this notebook we will use the following Python libraries:

- [os](#) - To create directories and work with files
- [pooch](#) - To download and unzip the data
- [rasterio](#) - To access and explore geospatial raster data in GeoTIFF format
- [xarray](#) - To process the data and prepare it for damage calculation
- [rioxarray](#) - Rasterio xarray extension - to make it easier to use GeoTIFF data with xarray
- [damagescanner](#) - To calculate the damage maps
- [cartopy](#) - To plot the maps
- [matplotlib](#) - For plotting as well

```
import os
import pooch

import pandas as pd
import rasterio
from pathlib import Path
import rioxarray as rxr
import xarray as xr

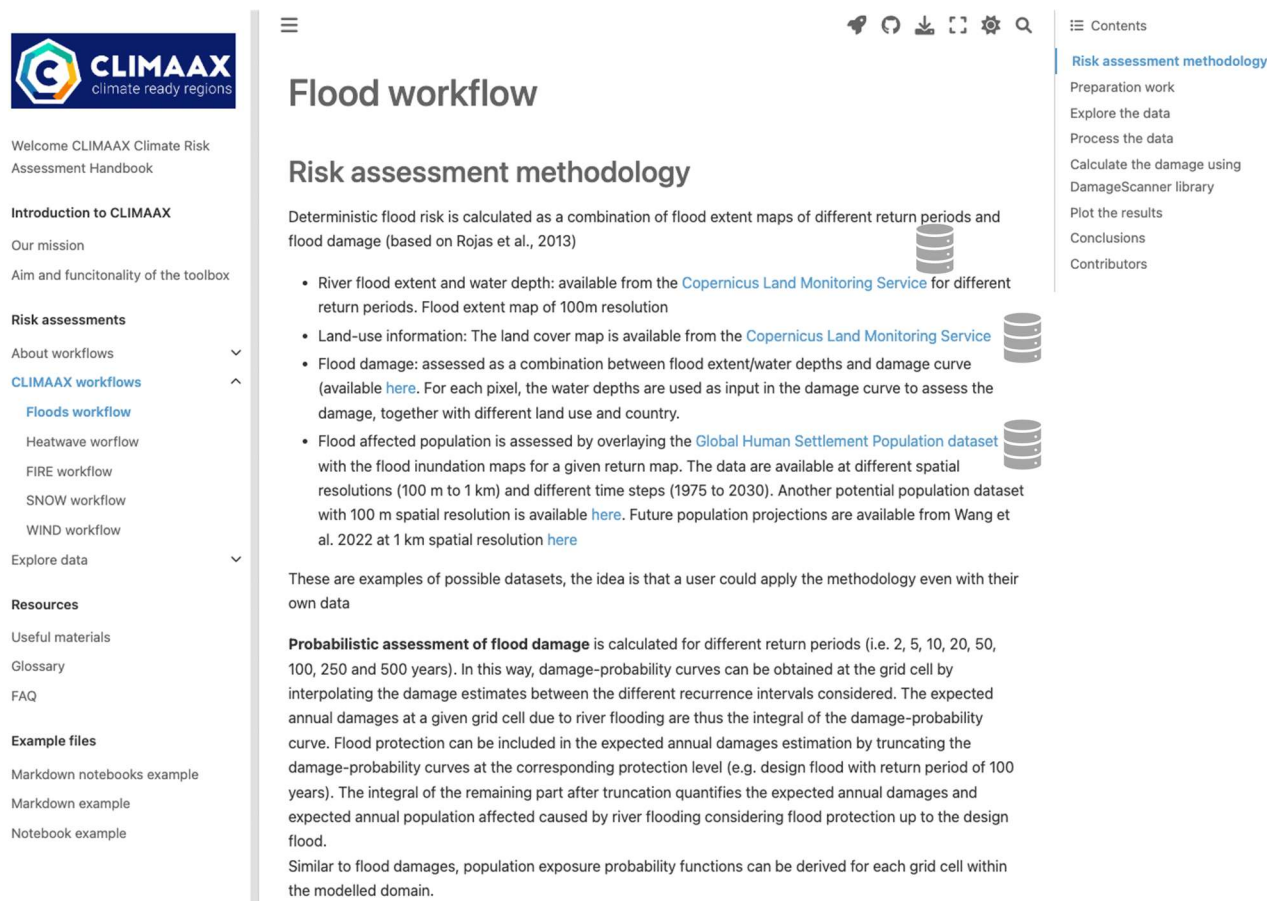
from damagescanner.core import RasterScanner
from damagescanner.plot import damagemap_raster

import matplotlib.pyplot as plt
import cartopy.crs as ccrs
import cartopy.feature as cfeature
import cartopy
```

Figure 2. Screenshot of instructions how to load the libraries needed for the specific workflow

### 3.1.2 Access the needed data

The next step is to access and explore the data. The links to the origins of the datasets are provided for each of the datasets used. This is illustrated in Figure 3. In this particular case the user has access to hazard data in terms of the flood extent, land use information, flood damage curves and affected population. If new to the data, the users will be able to explore each data set in depth with data notebooks in the “Explore data” section. Common queries and questions will be explained within this section. Similarly, the links to the documentation of each library used in the workflow are provided, so users can learn more about them if they want to expand the workflow or better understand the code in the workflow.



The screenshot shows the CLIMAAX web application interface. On the left is a sidebar with the CLIMAAX logo and navigation menu. The main content area is titled 'Flood workflow' and 'Risk assessment methodology'. It contains a detailed description of deterministic flood risk calculation, listing three data sources: Copernicus Land Monitoring Service for flood extent, Copernicus Land Monitoring Service for land-use information, and Global Human Settlement Population dataset for population. It also includes a section for probabilistic assessment of flood damage and a note about population exposure probability functions.

Figure 3. Example of a workflow in the Climate Risk Assessment book - view of the data description with links to the data origins

In this example the user is using pre-existing data, but the idea is that the user adds their own data here, alternatively accesses other available data sets and help and user guide for how to do that will be developed.

### 3.1.3 Calculate the hazard

The user now has its data and can calculate the hazard. This step will require modification for the user as they will have to decide the level of hazard they are interested in and which particular variable. IN this example the return period of flood extent is the target variable, and this has here been pre-computed by the JRC and provided as a dataset. However, the user can here create their own type of hazard, for example using a different method to provide the expected hazard for example through a different hydrological model. We will provide basic tools for this type of operations, for example how to calculate return periods and provide information how to interact with model output from other sources. This step also contains information on how to manipulate the data, such as cutting out regions, interpolation and averaging in space and time. The

user can run snippets of the code to see in real-time the results and modify the code, for example the geographical boundaries of a cutout of a map (Figure 4).

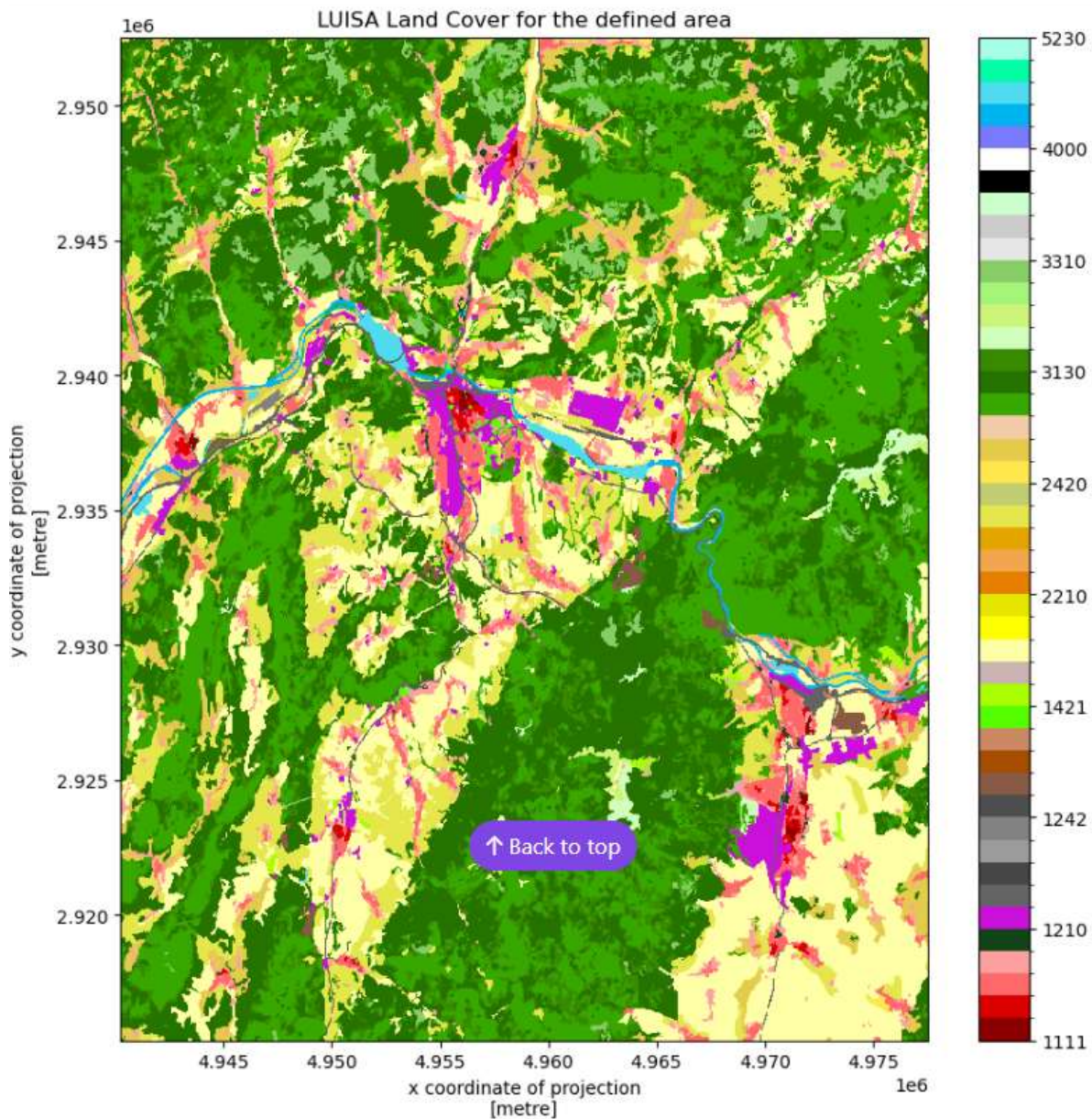
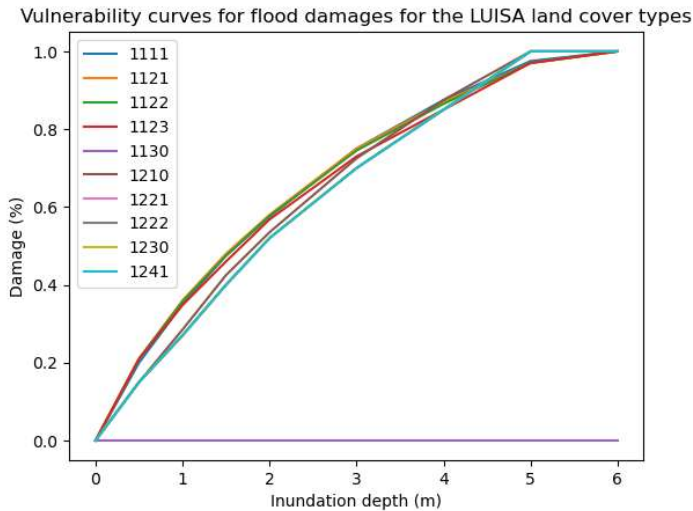


Figure 4. Example of a cut-out of a specific region of a land-use map.

### 3.1.4 Calculate the risk

Once the hazard is calculated, the next step is to provide the risk assessment. Using the hazard maps, these can now be combined with the vulnerability and exposure to create the risk information. The user will here be able to fully explore the risk from several angles, for example looking at different exposure (buildings, infrastructure etc) and vulnerability (population, economic value) factors for their particular interest. Figure 5 show an example of vulnerability curves for flood damages related to land cover types.



```
LUISA_info_damage_curve.loc[maxdam.index[0], building_types].values
```

```
array([0.5, 0.5, 0.0], dtype=object)
```

Figure 5. Vulnerability curves for flood damages for different land covers.

The risk assessment output in this example is in the form of a risk map, showing which areas are most affected in terms of monetary losses expressed EUR (Figure 6). The user can then see clearly which areas are most affected and where actions are needed to build resilience.

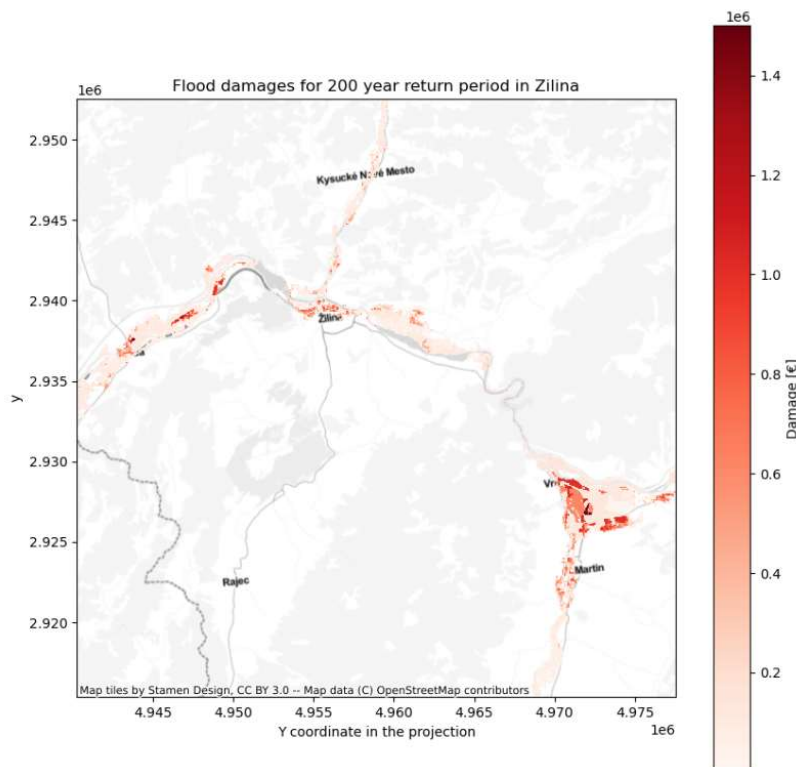


Figure 6. Flood risk in Zilina expressed losses in EUR due to a flood event of 200-year return period.

### 3.1.5 Calculate the future risk due to climate change

The last step in the CRQA is to assess the future risk due to climate change. The toolbox will provide tools on how to handle the climate data and provide guidance in selecting potential future scenarios. The main data for this will be the global climate models that are run as part of the IPCC CMIP6 scenarios (Figure 7). These climate model scenarios are quite coarse, and other downscaled datasets of higher resolution are envisioned to also be used, such as the CORDEX projections over Europe. These functionalities are currently under development, and the main source of the climate change projects will come from the Copernicus climate and atmospheric data store (CADS).

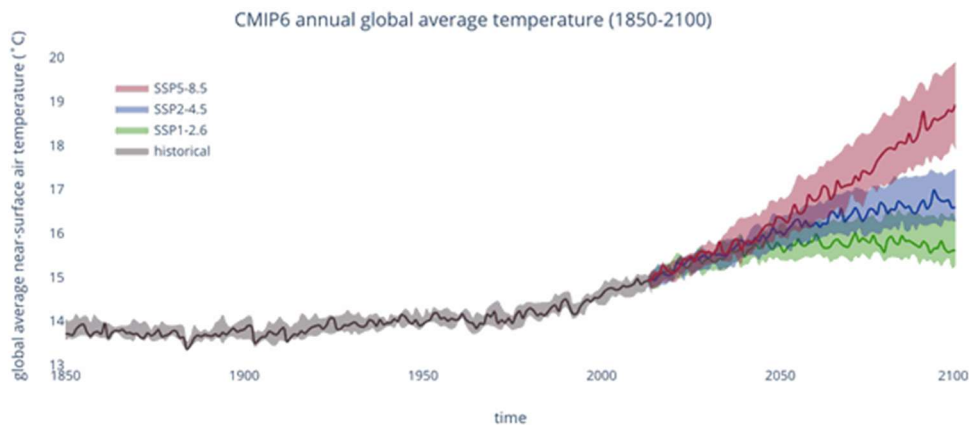


Figure 7. Future climate projects of the CMIP6 global climate models for three different scenarios.

### 3.1.6 Extra resources to aid in the user journey

If the user is unfamiliar with some part of the workflow or the terminology, they can find extra help in the **Resources** section. A glossary will hold the definitions of risk assessment, climate, data and software terms. An FAQ will be populated with the questions and challenges revealed during the collaboration with the pilot regions and expanded with the frequently asked questions from the participants of the open call in the later stage of the project.

Besides Glossary and the FAQ section, a page with the useful materials will be a valuable resource. We are aware that we cannot provide examples and tutorials about all the technical topics that are needed for all climate risk assessment workflows. However, there are plenty of open-source resources, trainings, webinars and tutorials available online and we will maintain the collection of the links towards well established open repositories. The initial list consists of the links to resources about:

- FAIR principles for data and software
- Git and GitHub
- Python
- Jupyter
- Scientific Python libraries

This will be expanded with the links to new Copernicus Climate and Atmosphere Data Store (CADS) toolbox documentation and tutorials as they become available, as well as any other resource that is used in the toolbox.



## 4 Toolbox technical architecture

### 4.1 Implementing the Jupyter environment

In this section, we describe all currently available components of the toolbox architecture in detail. As a guide to how the different technologies map onto the conceptual bakery shop, Figure 8 gives an overview of how the toolbox is implemented. The technologies will be presented in detail in the sections below.

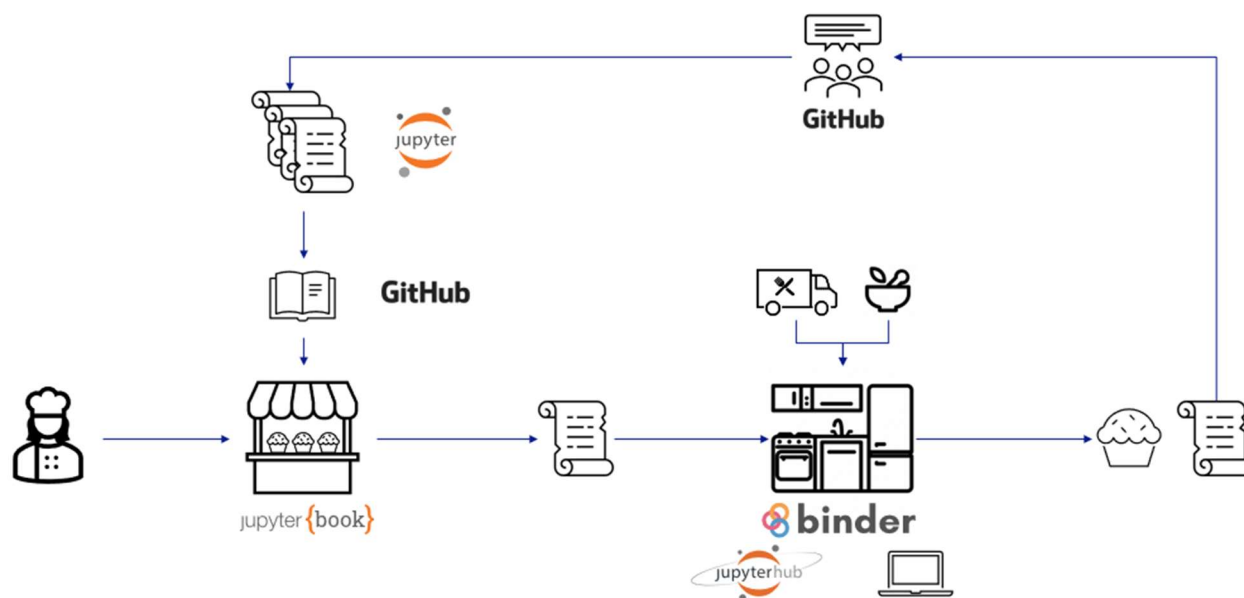


Figure 8. Mapping of the technologies used on the toolbox onto the user story of the CRA bakery.

To support the FAIR principles – particularly the need for toolbox workflows to be reused and built upon European and local data – the Jupyter ecosystem was chosen as the primary toolbox platform (Figure 9). Central in the toolbox are the climate risk assessment workflows captured in the computational Jupyter notebooks<sup>2</sup> that combine computer code, plain language descriptions, data, rich visualisations, and interactive controls.

<sup>2</sup> <https://docs.jupyter.org/en/latest/#what-is-a-notebook>

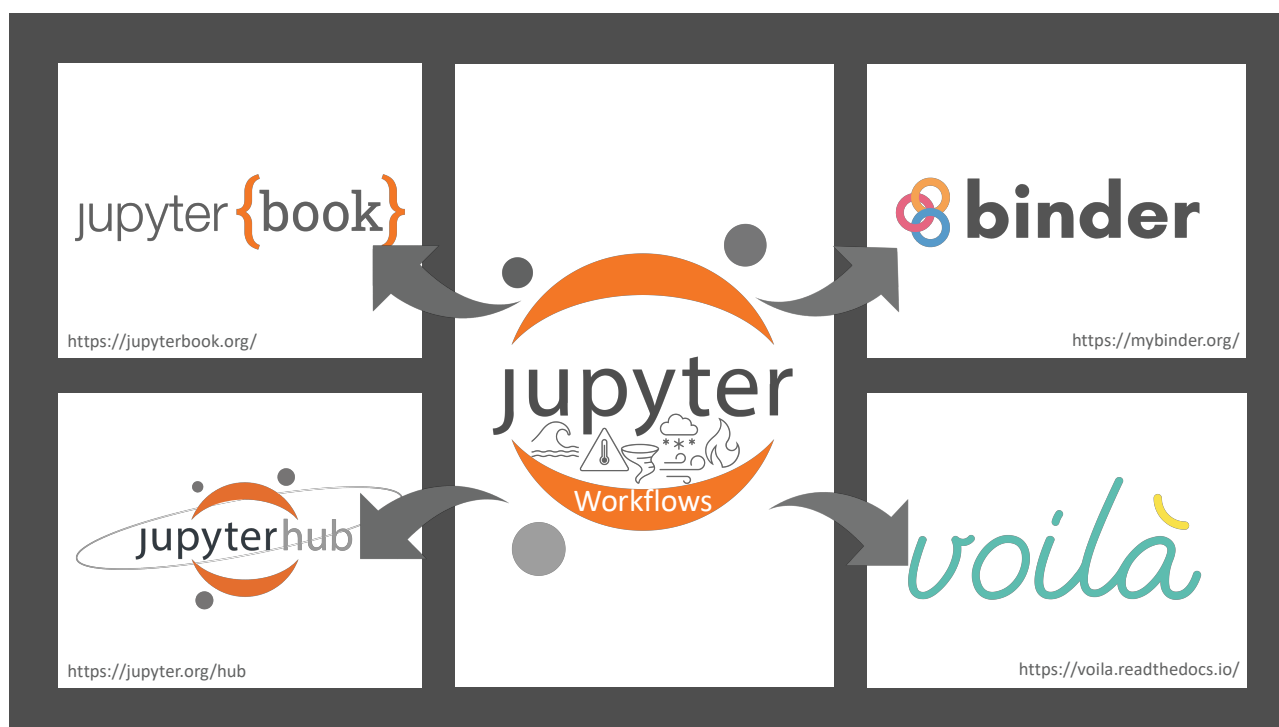


Figure 9. Schematic representation of the CLIMAAX toolbox elements. The components “book”, “binder”, “hub” and “voilà” are described in the main text. The main figure denotes the Jupyter workflows, also known as Jupyter notebooks.

Two sets of Jupyter notebooks will be in the toolbox for each of the elements of the toolbox shown in Figure 9:

- A set of workflow notebooks – each containing one or several climate risk assessments for a specific hazard,
- A set of data notebooks – each containing specific datasets used in one or more workflow notebooks.

In the mock-up version of the toolbox only the workflow notebooks are included, while a first set of the data notebooks will be ready for milestone M3.2 (“Prototype of the Operational Toolbox, including documentation”) in M12 of the project. The second toolbox component is an online tool in the form of a **Jupyter Book**. A Jupyter Book is an open-source tool used for creating interactive, web-based books and documentation from Jupyter notebooks and text files. The Jupyter Book is where all the workflow and data notebooks are presented, together with other relevant documentation needed for the climate risk assessment. The Jupyter book, in our case called **Climate Risk Assessment book**, is a **publicly available web resource** and it acts as an entry point for all the toolbox elements. In order to lower the entry barrier for the users of the toolbox, the Jupyter notebooks are available on these two platforms:

- a **Binder hub**<sup>3</sup> for public users. Public Binder hub<sup>4</sup> is an online service for building and sharing reproducible and interactive computational environments from online repositories. In this environment the users can run the toolbox and make minor changes but will not allow the users to create new workflows and save their work. This option is there to allow for a first assessment.

<sup>3</sup> <https://mybinder.org/>

<sup>4</sup> mybinder.org

- a dedicated **Jupyter hub**<sup>5</sup> available for the CLIMAAX pilot partners and participants of the open call. JupyterHub is an open-source, multi-user server for Jupyter notebooks. It is designed to provide a scalable and customizable platform for hosting Jupyter notebooks and instances in a multi-user environment. This will be a place where users can run, edit and create their customized risk workflows according to their needs without having to install anything on their own computer. The computational environment will be managed by the CLIMAAX team.
- As the last component of the toolbox, the workflow notebooks will be adapted to serve as **interactive dashboards (Voila)** for less technical users. Since the toolbox is intended as a cookbook where each workflow notebook is a separate recipe and educational material, a set of instructions on how to convert the workflow notebooks into dashboards will be provided.

In a project of the size and complexity of CLIMAAX, collaboration teams will consist of different partners in different countries jointly working on the same data or code simultaneously and building on each other's work. For this, good practices in version control are of utmost importance. Using a version control system, changes made by different people are tracked and automatically combined, saving a great deal of manual effort. Using version control from the beginning of building the toolbox also makes every step of the process transparent. To foster collaborative work on the toolbox and its documentation, and reproducibility of the workflows, **Git**<sup>6</sup> is used as the versioning system and **GitHub**<sup>7</sup> was chosen as the hosting platform. Under the umbrella of the CLIMAAX organisation page<sup>8</sup> currently seven public code repositories have been created in the mock-up version: six for each representative hazard CRA workflow notebook present, and one hosting the code for the CRA Jupyter book (Figure 10).

---

<sup>5</sup> <https://jupyter.org/hub>

<sup>6</sup> <https://git-scm.com/>

<sup>7</sup> <https://github.com/>

<sup>8</sup> The CLIMAAX GitHub organisation is set up at <https://github.com/CLIMAAX>

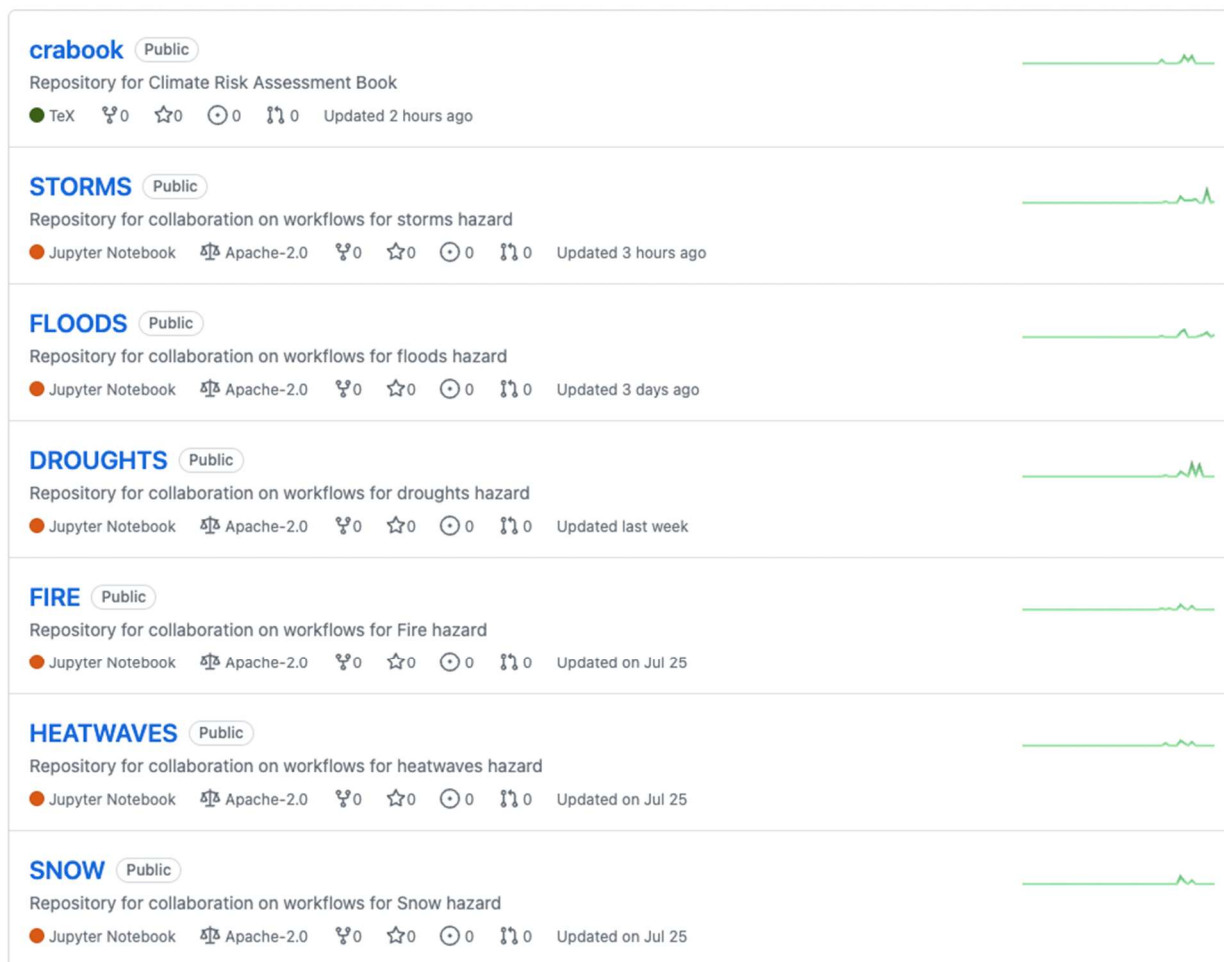


Figure 10. Structure of the CLIMAAX GitHub organisation space with (currently) six hazard workflows and one generic CRA code workflow (labelled “crabook”).

It is envisioned to have additional repositories to host the set of data notebooks, one for each data group: hazard, vulnerability and exposure data. Additional repositories will be added to accommodate currently missing hazards (e.g., landslides, coastal flooding). The hazard-based workflow repositories are the backbone of the toolbox. They are imported in the Climate Risk Assessment book and are available directly in the JupyterHub which allows for direct calculations. Furthermore, they will be used to create the interactive dashboards.

## 4.2 Workflows GitHub Repositories

Each hazard repository is named after the hazard theme, giving the following initial six hazard workflows:

DROUGHTS – <https://github.com/CLIMAAX/DROUGHTS>

FLOODS - <https://github.com/CLIMAAX/FLOODS>

HEATWAVES - <https://github.com/CLIMAAX/HEATWAVES>

FIRE - <https://github.com/CLIMAAX/FIRE>

WINDSTORMS - <https://github.com/CLIMAAX/STORMS>

SNOW - <https://github.com/CLIMAAX/SNOW>

Each repository has the same structure, containing 5 elements:

1. The `workflow` (`workflow.ipynb`) – the Jupyter notebook,
2. The `environment` (`environment.yml`) – configuration file containing all the necessary libraries for creating the computational environment that is used in the notebook.
3. `README.md` – with detailed instructions on how to run the notebook on Binder or locally, and how to contribute to the code repository,
4. `LICENCE` – with the information about the licence,
5. Additional files – configuration files with the needed data for calculations.

#### 4.2.1 Workflow notebook file

The climate risk assessment workflows are presented in Jupyter notebook files. Each workflow (Jupyter notebook) has a similar content structure:

1. Binder link – At the beginning of each workflow a Binder link is embedded, for easy connection with the public MyBinder service. Here the workflows can be run online,
2. Risk assessment methodology – description of the workflow which includes descriptions of the datasets used and the links to the origins of the data,
3. Preparation work – explanation of libraries needed for the workflow as well as creating the directory structure for downloading the data,
4. Download data – descriptions on where the data can be found and how these can be downloaded,
5. Explore the data – descriptions and code on how to open and inspect the datasets,
6. Process the data – descriptions and code for necessary processing of the data and calculating indices or damages,
7. Plot – plotting the results, including explanations of the code in case of complex plots,
8. Conclusions, describing the outcome of the CRA,
9. Contributions – authors' names and institutions,
10. Appendix – any additional material, for example, step by step plotting or processing explanations.

#### 4.2.2 Computational environment file

Working with many Python libraries requires managing dependencies between them. Over time these dependencies can change, as the libraries evolve. To ensure reproducibility of the workflows, we use a package, dependency and environment manager Conda. This manager does not only support Python, but many other programming languages, such as Ruby, R, Scala, Java, JavaScript, C/C++, Fortran and more.

Using Conda, isolated computational environments can be created (called “Conda environments”), to keep the packages for each project from conflicting with each other. The list of all the libraries installed inside the environment can be created and preserved in a so-called “yaml” file, which is a human-readable file. This file can be shared and used to recreate the environment on another machine, increasing the reproducibility of the work. Each GitHub repository contains an `environment.yml` file specific for the workflow. It is useful in many ways:

- Users can create the Conda environment automatically from this file
- Users can manually install all the libraries from the list in their python or Conda environments,
- Binder service uses this file to automatically create the computational environment for the users.

This functionality is particularly important for the collaboration and the reproducibility aspect of the workflows, because it ensures that the workflows remain functional even on different platforms.

#### 4.2.3 Readme file and licence

Each repository has detailed instructions on how to use it. These are preserved in the `README.md` file. This file also contains the link to the notebook in MyBinder service for easy access.

#### 4.2.4 Licence

Each repository has a licence attached to it. All the workflows are released under Apache License 2.0<sup>9</sup>. This is a permissive open-source software license that allows developers to use, modify, and distribute software under certain conditions. The license disclaims warranties and limits the liability of the original software authors and contributors.

#### 4.2.5 Additional files

It is anticipated that some of the future repositories will have additional files, such as small configuration files, for instance for calculating the damage curves. We will make every effort to avoid unclear and unstructured file collections and only use the data that is properly documented in respected data repositories.

### 4.3 Jupyter Book description

The main entry point of the toolbox is the Climate Risk Assessment book (CRA book) webpage, accessible on the web address <https://toolbox.climaax.eu>. Figure 11 provides a screenshot of the entry page.

---

<sup>9</sup> <http://www.apache.org/licenses/LICENSE-2.0>



Welcome CLIMAAX Climate Risk Assessment Handbook

Introduction to CLIMAAX

- Our mission
- Aim and functionality of the toolbox

Risk assessments

- About workflows
- CLIMAAX workflows
- Explore data

Resources

- Useful materials
- Glossary
- FAQ

Example files

- Markdown notebooks example
- Markdown example
- Notebook example



Contents

In a nutshell

# Welcome CLIMAAX Climate Risk Assessment Handbook

CLIMate risk and vulnerability Assessment framework and toolbox (CLIMAAX) is a 4-year Horizon Europe project that will provide financial, analytical, and practical support to improve regional climate and emergency risk management plans. CLIMAAX is designed to contribute to the harmonization and consolidation of the practice of climate risk assessment, leaving a legacy for upcoming European initiatives. The project started in January 2023 and runs until December 2026.

## In a nutshell

The European landscape of disaster risk management and climate adaptation is far from uniform.

CLIMAAX builds upon existing risk assessment frameworks, methods and tools, and promotes the use of datasets and service platforms for local and regional scale deployment. It will develop a robust and coordinated framework of consistent, harmonised and comparable risk assessments.

The project brings the existing tools and services beyond state-of-the-art by prioritizing the further development of accessibility, guidance, tuning to local contexts, interpretation and uptake by representative Disaster Risk Management and Civil Protection authorities.

CLIMAAX is designed to significantly contribute to the harmonization and consolidation of the practice of climate risk assessment (CRA), leaving a substantial legacy for upcoming European initiatives. The project will deliver:

1. A standardized CRA framework CRA built on current community experience and best-practices
2. A Toolbox with data, models and utilities to provide access to European and global open data archives integrated with local data and procedures
3. Five European pilot regional CRAs to shape the framework and toolset
4. Financial support for at least 50 regions to execute a context specific CRA
5. CRA guidance material and online helpdesk for other European regions
6. A proposal to upscale results into the future operationalization of the regional CRA support function

Next >  
Our mission

Figure 11. Climate Risk Assessment book entry page accessible from <https://toolbox.climaax.eu>

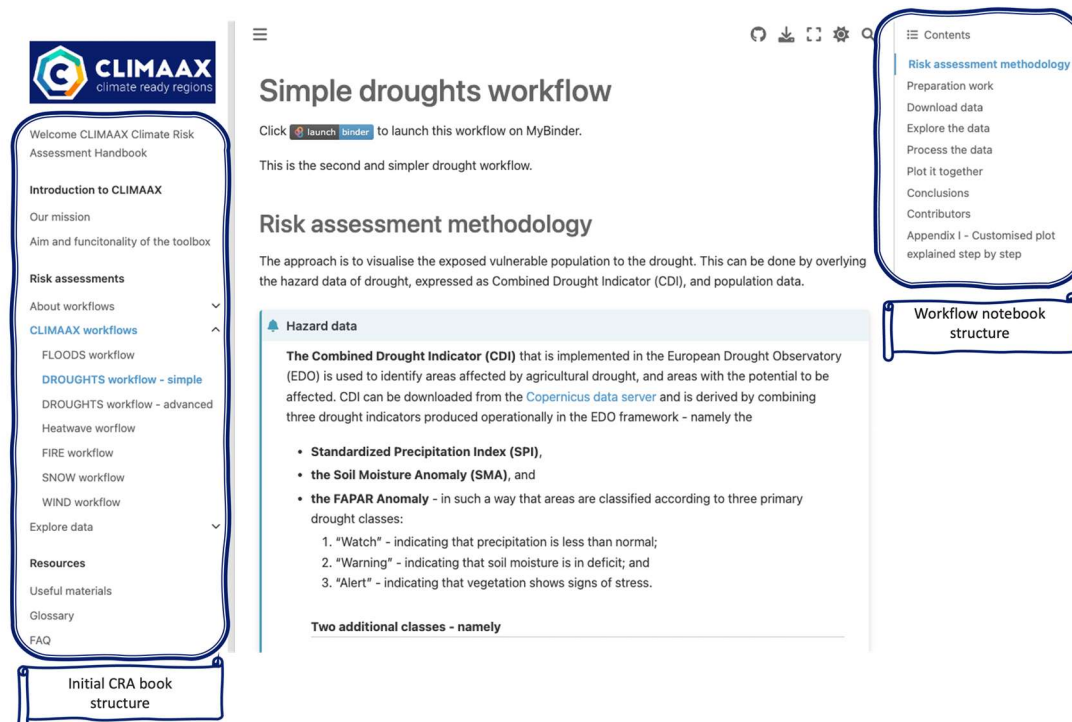
The CRA book serves as documentation for the climate risk assessment framework as well as the documentation for technical parts of the toolbox, including workflow and data notebooks. The goal of the CRA book is to provide all information that is needed to reproduce or extend the example regional climate risk assessments, for instance by adding their local data. Users can explore the resources on GitHub, re-use all the material and decide to contribute the updated CRA book to the CLIMAAX toolbox. The CRA book is set up for easy navigation to the specific workflow users wish to reproduce for their region, or to resources containing in depth material about a specific topic (such as detailed explanations about the datasets used, or some technical or framework guidance).

### 4.3.1 Climate Risk Assessment book structure

The CRA book has three main sections (Figure 12):

- **Introduction** – with information about the CLIMAAX project and the climate risk assessment framework. In the future operational version, this section will be updated with adequate guidance on navigation through the CLIMAAX CRA Framework,

- **Risk assessments** – contains the two sets of Jupyter notebooks: the workflow notebooks and the data notebooks,
- **Resources** – besides a glossary and FAQ, this section also provides links to several potentially relevant resources currently not included in the toolbox. These resources provide additional guidance for users who are new to conducting climate risk assessments and working with climate and environmental data, including tutorials on Python, Jupyter, Git, GitHub, FAIR principles for data and software etc.



The screenshot displays the CLIMAAX interface. On the left, a sidebar shows the 'Initial CRA book structure' with sections like 'Welcome CLIMAAX Climate Risk Assessment Handbook', 'Introduction to CLIMAAX', 'Risk assessments', and 'Resources'. The main content area shows a 'Simple droughts workflow' notebook. The right sidebar shows the 'Workflow notebook structure' with sections like 'Risk assessment methodology', 'Preparation work', 'Download data', etc. The notebook content includes a 'Hazard data' section with a list of drought indicators: Standardized Precipitation Index (SPI), Soil Moisture Anomaly (SMA), and FAPAR Anomaly.

Figure 12. Workflow notebook rendered in CRA book with notebook structure on the right and book structure on the left.

To serve also unexperienced users of the Jupyter ecosystem an **extra section** of the book is present in this mock-up toolbox version with example files containing example code. This section will be removed and replaced by the section about collaboration with information on how to contribute to the Jupyter Book to create new climate risk assessment workflows.

#### 4.4 Adding new workflows and peer-review process

As the CLIMAAX project evolves and grows, we will start welcoming and encouraging contributions from pilot partners and open call participants in the later stage of the project. These new contributions might be in the form of feedback about the current toolbox, but also in the form of additional workflows or chapters in the CRA book. We will create a series of templates to guide diverse types of content development while maintaining a consistent format of the content throughout the toolbox elements. Also, instructions about the peer review process of new contributions will be added. The peer-review process will allow the user to present the work done to create the new workflow and its functionalities and suggest this to be include in the CRA book. The project developers will then review and revise the code and decide whether it will be presented as a new workflow or merged into existing ones.



#### 4.4.1 CRA book GitHub repository structure

The foundation of the Climate Risk Assessment (CRA) resources lies within the GitHub repository hosted under the CLIMAAX organization. This collaborative repository, accessible at <https://github.com/CLIMAAX/crabook>, serves as a central hub for the CRA book's source code and the version history. The repository's structure enables effective collaboration, making it a dynamic hub for the ongoing evolution of the project. The repository's setup is created by using the Cookiecutter Jupyter Book template<sup>10</sup>, a tool designed for creating simple Jupyter book websites. This template simplifies the setup procedure through a series of documented steps available in the Cookiecutter Jupyter Book repository. Although the Jupyter Book template includes the option for initial Continuous Integration and Deployment (CI/CD) workflow integration, a custom CI system has been set up through a customized GitHub Actions workflow.

The main elements of the CRA book repository are:

- **crabook** – this folder houses all materials, essential for the generation of HTML webpages during the building process,
- **.github/workflows** – folder with the files for automatic building and deploying of the material,
- **Requirements.txt** – with the list of Python libraries necessary to update and build the CRA book,
- **README.md** – with the technical instructions for contributors to the CRA books,
- **.gitignore** – with the list of files for local work that are not uploaded to the git repository,
- **.gitmodules** – with the list of git submodules imported in the crabook repository. These submodules are the repositories containing the CRA workflow notebooks.

#### 4.4.2 Automatic Building and Deploying using GitHub Actions

As part of the CLIMAAX project's commitment to advancing climate risk assessment tools, an efficient and automated process for building and deploying resources is essential. GitHub Actions, a powerful automation tool integrated with the project's version control platform, enables the seamless creation and distribution of the CRA book. This workflow is instrumental in ensuring that the most up-to-date content and insights are readily accessible to stakeholders with seamlessly.

##### 4.4.2.1 Workflow Overview

To ensure seamless accessibility and distribution of the CRA book, a GitHub Actions<sup>11</sup> workflow is set up. This GitHub actions workflow automates the book's construction and its deployment to the GitHub Pages platform, which serves as the dedicated home for the CRA book website. This streamlined process expedites updates, guarantees consistency, and provides users with the most current insights and resources. The workflow, named 'build-deploy-book'<sup>12</sup> is triggered automatically upon changes to the main branch or can be manually initiated as required. This flexibility ensures that the climate risk assessment materials are always current and accessible to users.

The workflow comprises two essential jobs: 'build' and 'deploy':

---

<sup>10</sup> <https://github.com/executablebooks/cookiecutter-jupyter-book>

<sup>11</sup> <https://github.com/actions>

<sup>12</sup> Code for the GitHub action available at: <https://github.com/CLIMAAX/crabook/blob/main/.github/workflows/build-deploy-book.yml>

- **Job 1: Building the Jupyter Book** – Upon initiation, the '**build**' job ensures that all dependencies are up to date by installing the required Python packages from the '`requirements.txt`' file. Subsequently, the job compiles the climate risk assessment resources contained within the '`crabook`' directory. This compilation generates a folder with a set of HTML files), enhancing the accessibility and usability of the content,
- **Job 2: Deploying to GitHub Pages** – Once the Jupyter Book is successfully compiled, but only if the changes were made on the main branch, the '**deploy**' job is executed. It leverages the GitHub Pages feature to seamlessly publish the compiled Jupyter Book to a publicly accessible URL. This URL becomes the focal point for users to access and explore the climate risk assessment resources.

#### 4.4.3 CRA book and workflows repositories connection

The work on the CRA book and Jupyter notebooks needs different computational environments. Additionally, not all the users who will use the notebooks will also be collaborators on the CRA book. Therefore, the CRA book and the Jupyter notebooks are organised into separate GitHub repositories (Figure 13). This separation enables several advantages:

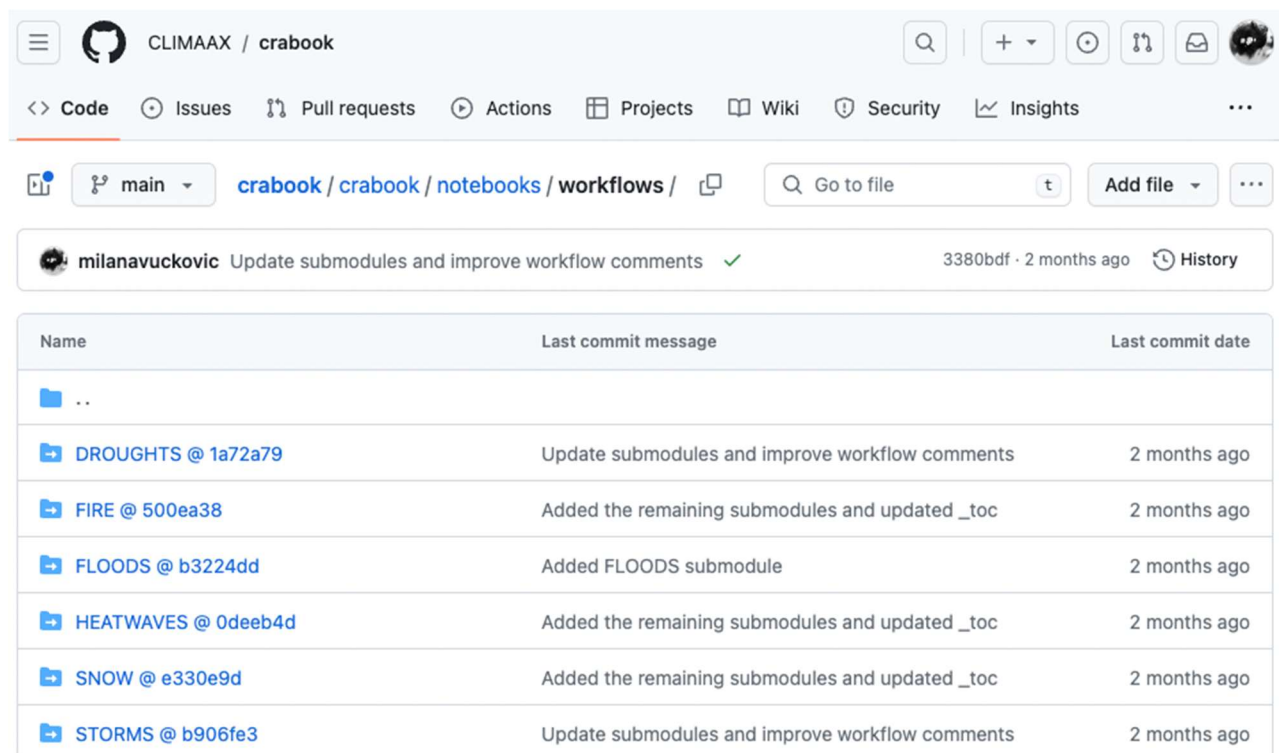
- Different teams collaborating on different hazard workflow repositories have to install smaller Conda or Python environments, leading to less possibilities for problems with conflict dependencies that come from big and complicated environments,
- With each workflow having its own repository it is possible to have independent Binder computational environments where the user can use just the workflow they need and not the entire CRA book,
- Hazard workflow repositories are updated independently from the CRA book, enabling shorter release cycles and easier updates and collaboration.

To integrate the repositories with Jupyter notebooks with the CRA book, we are using Git Submodules tool<sup>13</sup>. This tool enables inclusion of another Git repository within our main Git repository as a subdirectory. This allows us to track and reference an external repository's code in the main hub. Submodules are like "nested"

---

<sup>13</sup> <https://git-scm.com/book/en/v2/Git-Tools-Submodules>

Git repositories within the main Git repository. They also allow us to specify a specific code update or version of the external repository to use in the project.



Name	Last commit message	Last commit date
..		
<a href="#">DROUGHTS @ 1a72a79</a>	Update submodules and improve workflow comments	2 months ago
<a href="#">FIRE @ 500ea38</a>	Added the remaining submodules and updated _toc	2 months ago
<a href="#">FLOODS @ b3224dd</a>	Added FLOODS submodule	2 months ago
<a href="#">HEATWAVES @ 0deeb4d</a>	Added the remaining submodules and updated _toc	2 months ago
<a href="#">SNOW @ e330e9d</a>	Added the remaining submodules and updated _toc	2 months ago
<a href="#">STORMS @ b906fe3</a>	Update submodules and improve workflow comments	2 months ago

Figure 13. Links to repositories with Jupyter notebook workflows inside the CRA book repository

This means that we can continually update the repositories with the Jupyter notebooks, but the CRA book will only be showing the frozen versions at a specific branch, commit or release. This setup simplifies the collaboration, as teams that are working on different workflow notebooks don't have to depend on each other.

## 4.5 JupyterHub Integration

In the CLIMAAX project five regional pilot partners will have the opportunity to test the toolbox first and influence its co-development. After this initial design phase participants in the open call will be the next group of users. To facilitate this user base, JupyterHub has been set up at ECMWF infrastructure on European Weather Cloud<sup>14</sup>, the cloud-based collaboration platform for meteorological application development.

JupyterHub is designed to provide a scalable and customisable platform for hosting Jupyter notebooks and JupyterLab instances in a multi-user environment. It allows multiple users to access Jupyter notebooks on a single server or cluster of servers. Each user gets an assigned environment for running the Jupyter notebooks, preventing conflicts between multiple users' Python environments, packages and running notebooks. Users

<sup>14</sup> <https://www.europeanweather.cloud/>

cannot see or access each other's files or sessions unless sharing is explicitly enabled. While sharing will not be enabled at the beginning of use, this is something we will explore in the later stages of the toolbox.

JupyterHub can spawn and manage Jupyterlab and Jupyter notebook instances for users on-demand. Work from each user will be set-up in a so-called Docker container, a self-contained unit that packages an application and its dependencies, ensuring consistency, portability and isolation. Containers are lightweight and share the host operating system's kernel, making them efficient in terms of resource utilisation.

Since JupyterHub is highly customizable, it allows us to tailor the environment for specific needs and to evolve this environment as these needs change. This in practice means that we will begin with only one computational Python environment made available for all users and managed by ECMWF, without the possibility for the users to install and manage their own environments. We are aware that this might not be ideal for the advanced users, so we will revise this decision once we know more about the users' needs.

Besides the specific environment, we will also explore the possibility to enable custom Jupyter extensions, themes and other settings towards the needs of the toolbox users. This includes, but is not limited to:

- git extension – for the git integration, allowing the interactive management and visualisation of Git repositories, tracking changes and committing code directly from the interface,
- table of content – generates the table of contents for quick navigation through long notebooks,
- spellchecker – Jupyter notebooks don't have spellcheck by default and this extension might be very useful to help catch typos and spelling errors in the documentation,

For the purpose of the mock-up, only small machines are available to users; It is anticipated that in the operational version a profile with 8vCPU, 8-16GB RAM and 250GB Storage will be available. This will be revised as we gather the feedback from the first users.

#### 4.6 User authentication

The CLIMAAX JupyterHub is integrated with the ECMWF authentication system, meaning that users will create an ECMWF account and after the approval process, they will have assigned a policy, giving them the access to the JupyterHub as well as to JIRA ticket system for reporting issues through ECMWF Service Desk service. A guide to how to setup an ECMWF account will be provided with the toolbox.

Upon first logging into the account a workspace is created for the users. Currently the contents of the GitHub repositories are copied into the users' workspace. The example of this set up is presented in the Figure 14. This allows the users to quickly start exploring the workflows even if they are unfamiliar with git. They are able to modify the content of the folders, copy the files and create new folders and files. They are not able to upload the content back to the original repositories. For users to be able to contribute to the example notebooks a review process will need to be established.

Upon every major update of the repositories, the updated versions will be copied into users' workspace, overwriting their changes. The users will be notified about this beforehand. They will also be advised not to keep important work in these directories, but to make copies and start from these.

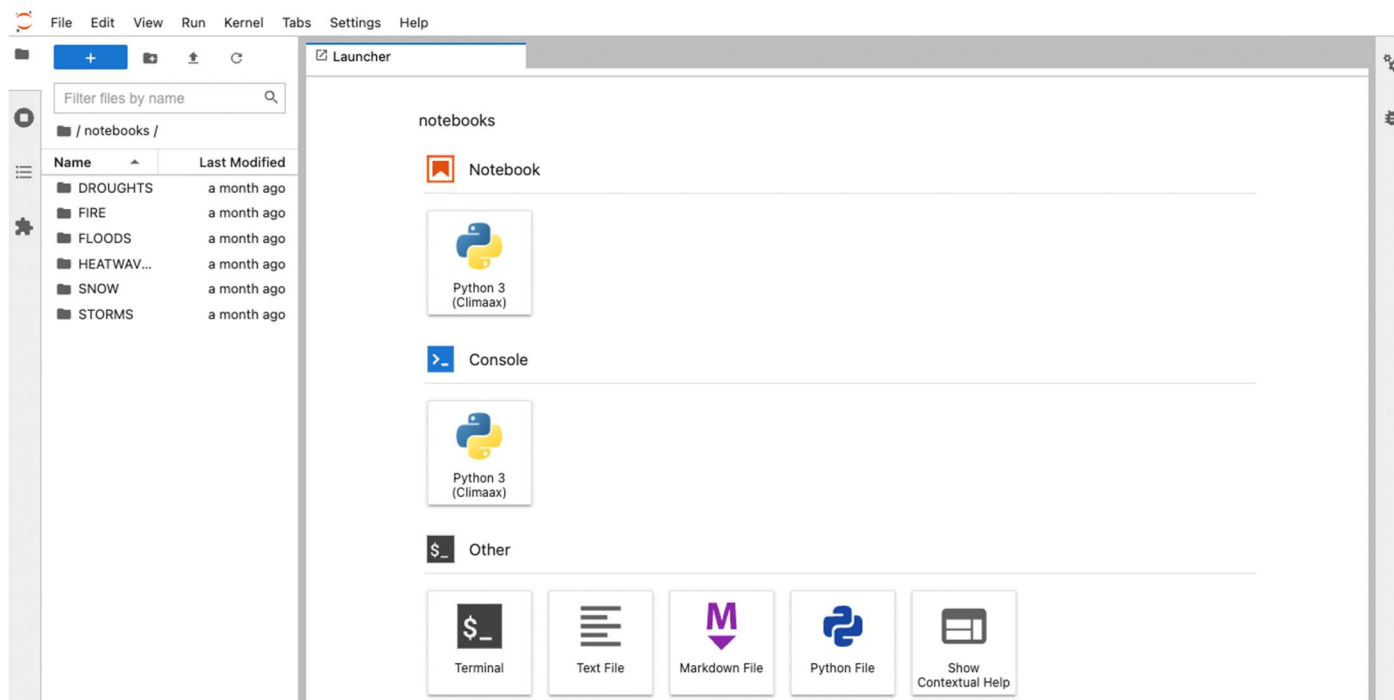


Figure 14. View of user's space in JupyterHub with the folders containing notebooks material

## 4.7 Binder integration

To lower the entry requirements for the use of the toolbox, the binder environment provides an easy-access service where there is no need to install anything on a local computer, to be a CLIMAAX partner, nor to register a user account anywhere. This is enabled by using the MyBinder<sup>15</sup> public online service<sup>16</sup>.

MyBinder is a service that allows users to create and share reproducible and interactive computational environments from online repositories. It is powered by BinderHub<sup>17</sup>, a customised and enhanced JupyterHub that runs on one of the clouds of the BinderHub Federation<sup>18</sup>. This open-source tool and open service relies on the public support of the organisations and volunteer work of individuals. The main benefits of MyBinder in the CLIMAAX projects are:

- **Reproducibility:** MyBinder ensures that our workflow notebooks are reproducible, meaning that anyone can run them in the same computational environment, defined within the repository alongside the notebooks, and get the same results. This is crucial for CLIMAAX, as we need to ensure that our results are reproducible, and the work is reusable.
- **Collaboration:** MyBinder makes it easy to collaborate on Jupyter notebooks. Notebooks are shared in GitHub and others can run them in a reproducible environment.
- **Demonstrations:** MyBinder is an excellent way to demonstrate the work to others. It takes away the technical burden of installing libraries to run the demonstration. We can create a Binder link for the

<sup>15</sup> <https://mybinder.readthedocs.io/en/latest/>

<sup>16</sup> <https://mybinder.org/>

<sup>17</sup> <https://binderhub.readthedocs.io/en/latest/>

<sup>18</sup> <https://mybinder.readthedocs.io/en/latest/about/federation.html>

notebooks and share it with the participants of the demo or workshop. This will allow them to run the notebooks in a reproducible environment and see how they work first hand.

One drawback of the MyBinder service is its limited resources. It relies on donated resources, and possibility of insufficient compute resources can be a threat to the service. If Binder service proves to be used by many CLIMAAX partners and participants in the open call, and essential for the collaboration in the project, ECMWF will consider making additional resources available by implementing our own CLIMAAX Binder service, by hosting BinderHub on our European Weather Cloud.

#### 4.8 Interactive dashboards using Voila

From the start of the discussions with CLIMAAX pilot cases it has been clear that we will need to support two distinct groups of stakeholders: participants with technical knowledge that expect to manipulate toolbox parameters in great detail, and participants with less or without technical knowledge who are primarily inspecting precalculated outcomes allowing minor adjustments.

While traditionally an extensive technical knowledge was needed in order to perform a climate risk assessment, it is possible to create a workspace where a decision maker can interact with the data and manipulate the maps without editing the code itself. One way this can be done is using interactive Voilà dashboards<sup>19</sup>.

Voilà is a Jupyter notebook extension that enables conversion of Jupyter notebooks into interactive web applications or dashboards. It's a powerful tool for sharing and presenting the data, visualizations, and code in a user-friendly and interactive format, making it easier to share and present the work with a broader audience, particularly those who are not familiar with Jupyter notebooks or coding. Voilà allows the creation of interactive elements within the dashboard. Users can adjust parameters, select options, or manipulate visualizations directly in the dashboard interface. Voilà dashboards can be implemented as standalone web applications, but also quickly started from MyBinder service.

While we will aim to have a prototype of some interactive dashboard in M12 for the M3.2 Prototype of the Operational Toolbox, the main work on creating interactive dashboard from workflow Jupyter notebooks will be done in the first months of 2024 to be ready for the second version of the toolbox in M16. Besides providing interactive dashboards, we will provide the instructions and best practices on how to structure the underlying Jupyter Notebooks, to facilitate the creation of intuitive user-friendly Dashboards. This can be promoted by for instance setting requirements on elements of the workflow that need to be explicitly calculated and made available to the dashboard.

---

<sup>19</sup> <https://voila.readthedocs.io/en/stable/index.html>

Besides reading the materials on CRA book, users can execute the code in several different ways (see Figure 15):

- By visiting the workflow notebooks individual GitHub repository, cloning it, and using the instructions on how to use it,
- By downloading the individual workflow notebook and executing code on their local machine,
- By executing the workflow notebook on MyBinder service using the 'Binder badge',
- Registered users can use each workflow on JupyterHub as well.

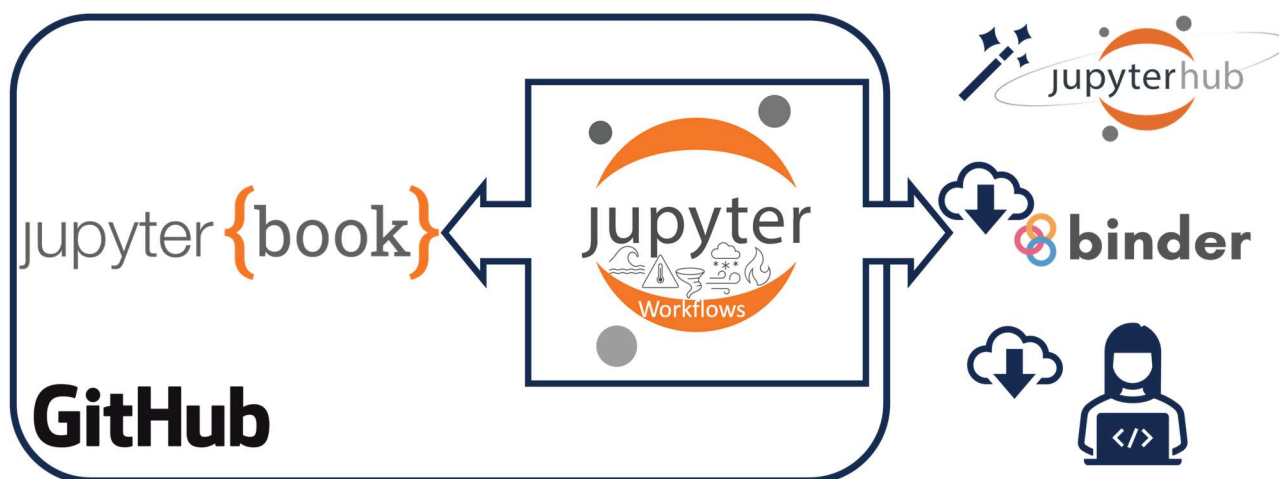


Figure 15. Schematic image of the different ways the workflow notebooks are used: They are hosted on GitHub and imported in the JupyterBook, they are automatically copied to the JupyterHub, and can be executed on Binder or downloaded for work on local computer

Links to the first three options are highlighted in the image of one of the workflows in the CRA book in Figure 16.



Welcome CLIMAAX Climate Risk Assessment Handbook

**Introduction to CLIMAAX**

 Our mission  
 Aim and functionality of the toolbox

**Risk assessments**

 About workflows  
**CLIMAAX workflows**  
 Floods workflow  
 Heatwave workflow  
 FIRE workflow  
 SNOW workflow  
 WIND workflow

Explore data

**Resources**

 Useful materials  
 Glossary  
 FAQ

**Example files**

 Markdown notebooks example  
 Markdown example  
 Notebook example

 Download    

## Flood workflow

 **Execute using Binder**

### Risk assessment methodology

Deterministic flood risk is calculated as a combination of flood extent maps of different return periods and flood damage (based on Rojas et al., 2013)

- River flood extent and water depth: available from the [Copernicus Land Monitoring Service](#) for different return periods. Flood extent map of 100m resolution
- Land-use information: The land cover map is available from the [Copernicus Land Monitoring Service](#)
- Flood damage: assessed as a combination between flood extent/water depths and damage curve (available [here](#)). For each pixel, the water depths are used as input in the damage curve to assess the damage, together with different land use and country.
- Flood affected population is assessed by overlaying the [Global Human Settlement Population dataset](#) with the flood inundation maps for a given return map. The data are available at different spatial resolutions (100 m to 1 km) and different time steps (1975 to 2030). Another potential population dataset with 100 m spatial resolution is available [here](#). Future population projections are available from Wang et al. 2022 at 1 km spatial resolution [here](#)

These are examples of possible datasets, the idea is that a user could apply the methodology even with their own data

**Probabilistic assessment of flood damage** is calculated for different return periods (i.e. 2, 5, 10, 20, 50, 100, 250 and 500 years). In this way, damage-probability curves can be obtained at the grid cell by interpolating the damage estimates between the different recurrence intervals considered. The expected annual damages at a given grid cell due to river flooding are thus the integral of the damage-probability curve. Flood protection can be included in the expected annual damages estimation by truncating the damage-probability curves at the corresponding protection level (e.g. design flood with return period of 100 years). The integral of the remaining part after truncation quantifies the expected annual damages and expected annual population affected caused by river flooding considering flood protection up to the design flood.

Similar to flood damages, population exposure probability functions can be derived for each grid cell within the modelled domain.

Contents

 Risk assessment methodology  
 Preparation work  
 Explore the data  
 Process the data  
 Calculate the damage using  
 DamageScanner library  
 Plot the results  
 Conclusions  
 Contributors

Figure 16. Links to different downloading or executing options available for each workflow notebook

Users can download and copy the workflows from the CRA book, modify them to fit their region or to develop more complex workflows.

CLIMAAX partners and the participants of the Open Call have the option to use the JupyterHub to perform the Climate Risk Assessment work in the European Weather Cloud allocated computational environment. As mentioned, in this hub every user will have all the repositories with the example workflows copied into their own space. These examples can be used as cookbooks and starting points for the climate risk assessment work. Resulting data can be stored in the hub storage or downloaded for local use.



## 5 Future developments and conclusions

### 5.1 Collaboration

The added value of the CLIMAAX toolbox is the possibility for collaboration and the possibility for the users to contribute their developments back to the toolbox. User generated workflows are preferably made available as starting point for other study areas addressing similar CRAs. For this purpose, we will organise regular “show and tell” sessions, where users can demonstrate their workflows to other users and selected workflows will be incorporated into the toolbox. We also will organise online thematic hackathons and coding drop-in sessions where participants can work together on particular shared challenges and problems. Finally, we plan to set up a platform for collaboration. We will explore several possibilities (Confluence forum, GitHub issues, Slack channel etc), and choose the most appropriate option for the community.

### 5.2 Technical improvements

For the first operational version of the CLIMAAX toolbox to be delivered in M12 (December 2023) several technical improvements are planned:

- Main branches of all the GitHub repository will be closed to deliver the first operational version and a peer review process for all the technical developments will be set up,
- A GitHub repository with the existing templates for the workflows will be set up. Currently there are six templates planned, plus a generic template
- All computational environments will be locked at versions in December 2023, and only be updated at the official update of the toolbox after extensive testing of the new versions of the libraries,
- Google analytics will be set up for the CRA book to track the usage and statistics,
- Automatic checks of validity of external links will be set up for the CRA book,
- All the workflow repositories will be indexed on Zenodo and assigned DOIs.

### 5.3 Content development

Parallel to the work on the technical improvements of the toolbox, work on the content will include:

- Notebooks for data exploration will be added,
- Glossary and Frequently Asked Questions (FAQ) sections of the CRA book will be populated,
- More links to the external resources will be added, and we will also explore the option to import external Jupyter notebooks directly in the CRA book (for example Copernicus training material for the use of data from the Climate Data Store).

### 5.4 Planned developments for the Toolbox Version 2 (M16)

For the second version of the toolbox (M16 of the project - April 2024), plans include:

- Improving existing workflows and resources based on pilot partners’ feedback,
- Adding new workflows based on pilot partners requirements,
- Updating Glossary, FAQ and links to external resources based on pilots’ feedback,
- Adding interactive dashboards based on the existing workflows as well as instructions on how to convert educational Jupyter notebook into an interactive dashboard,
- Improving the JupyterHub resources according to the real users’ needs after onboarding users from pilot partners. This includes:
  - Assigning users’ enough memory and storage to perform climate risk assessment,
  - Updating the computational environment with libraries that users need,

- Installing Jupyter extensions that enable features for advanced work.

## 5.5 Conclusions

Built on the requirements for the CLIMAAX toolbox presented in the deliverable D2.1 “Toolbox specification”, this deliverable presents the prototype serving as the baseline for the operation toolbox. The foundation has been set up, enabling collaborative work towards the functioning operational toolbox version. The toolbox set up enables comfortable learning and collaboration, removing the burden of installing libraries and conflicting dependencies. Workflow materials incorporated at the same time in the wiki web page in the form of JupyterBook and as ready to execute notebooks in the JupyterHub and MyBinder, present the basis for learning, performing climate risk assessment work and collaboration. Future results will be reported in the M12, as the Milestone M3.2 “Prototype of the Operational Toolbox”, including documentation when the first operational version of the toolbox will be released.

## 6 Glossary

**Binder** – an open-source project in the Jupyter ecosystem that allows the creation of custom, interactive, and executable computing environments from GitHub repositories containing Jupyter notebooks. It enables users to convert Jupyter notebooks into shareable web applications that others can use without needing to install any software or dependencies locally.

**BinderHub** – a customised and enhanced JupyterHub that runs on one of the clouds of the BinderHub Federation.

**Computational environment** (or computing environment) – refers to the combination of hardware, software, and network resources that collectively provide a platform for executing computer programs and performing various computational tasks. It encompasses all the elements and conditions necessary for computing to take place effectively.

**Conda environment** – a self-contained and isolated computational environment created and managed by Conda, an open-source package and environment management system.

**Continuous deployment** – a software development practice where code changes are automatically built, tested, and deployed to production environments without manual intervention. This approach ensures that every code update that passes automated tests is automatically released to users, providing rapid and seamless delivery of new features, bug fixes, and improvements.

**Continuous integration** – a software development practice where developers frequently integrate their code changes into a shared repository, and automated build and testing processes are triggered upon each integration. The primary goal of CI is to detect integration issues and defects early in the development cycle, ensuring that the codebase remains stable and reliable.

**Docker container** – a standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably from one computing environment to another. A Docker container image is a lightweight, standalone, executable package of software that includes everything needed to run an application: code, runtime, system tools, system libraries and settings.

**Git submodule** – allows users to include external repositories as dependencies in their own project, maintaining a connection to a specific version or commit of that external codebase. This helps manage complex projects by separating and version-controlling different components independently.

**JupyterBook** - an open-source tool used for creating interactive, web-based books and documentation from Jupyter notebooks and markdown files. The JupyterBook can be accessed and viewed freely online.

**JupyterHub** - a multi-user, web-based platform that enables the hosting and management of Jupyter Notebook servers for multiple users. It allows organizations, institutions, or individuals to provide a shared environment for data science and interactive computing, where users can create, run, and collaborate on Jupyter notebooks through a web interface.

**Jupyterlab** - an interactive web-based environment for data science, scientific computing, and code development. It extends the capabilities of Jupyter Notebook by offering a flexible and integrated interface that allows users to work with Jupyter notebooks, text editors, terminals, and custom interactive components within a single environment.

**Jupyter notebook** - an open-source web application that provides an interactive environment for creating and sharing documents that combine live code, narrative text, visualizations, and other rich media.

**Kubernetes** - is an open-source container orchestration platform that automates the deployment, scaling, and management of containerized applications. It provides a container-centric infrastructure where

applications are packaged as containers, and Kubernetes handles tasks like container scheduling, load balancing, self-healing, and scaling.

**Virtual python environment** – an isolated computing environment specifically created for Python packages, which allows for isolation of dependencies, libraries and packages.

**Version control** – a system that records changes to a file or set of files over time so that developers can recall a specific version later. For example, in this demonstration, we have been using a software source code as the files being version controlled, but in reality, this can be done with any type of file on a computer.<sup>20</sup>

---

<sup>20</sup> <https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control>

## 7 References

Barker, M., Chue Hong, N.P., Katz, D.S. et al. Introducing the FAIR Principles for research software. Sci Data 9, 622 (2022). <https://doi.org/10.1038/s41597-022-01710-x> , [doi:10.1038/s41597-022-01710-x](https://doi.org/10.1038/s41597-022-01710-x)

Wilkinson, M., Dumontier, M., Aalbersberg, I. et al. The FAIR Guiding Principles for scientific data management and stewardship. Sci Data 3, 160018 (2016). <https://doi.org/10.1038/sdata.2016.18>