Clim EMPOWER

CLIMEMPOWER APPLICATION ARCHITECTURE

ClimEmpower Work Package 3, D3.1

Project ClimEmpower: User Driven Climate Applications Empowering Regional Resilience

Work package 3, Deliverable D3.1

Date: 01.10.2024

Project information	
Project name:	**Project ClimEmpower: User Driven Climate Applications Empowering Regional Resilience**
GA No.	**101112728**
Start date:	**01/09/2023**
Duration:	**36 months**
Coordinator:	Denis Havlik, Scientist AIT Austrian Institute of Technology Giefinggasse 4, 1210 Vienna, Austria
	Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or CINEA. Neither the European Union nor the granting authority can be held responsible for them.
Deliverable details	
Deliverable name	**D3.1 ClimEmpower application architecture**
Description:	This Deliverable is part of WP3 User-centric CC resilience enhancing services work package and explains the envisaged implementation architecture of ClimEmpower applications. It describes the core services that are needed to fulfil the promised functionalities deriving from requirements set by the DoA as well as WP1. The Implementation architecture will allow for rapid prototyping of climate adaptation planning applications, which will serve the individual needs of the different regions in the project.
Version:	Final
Due date:	31/08/2024
Submission:	01/10/2024
Dissemination level:	**Public**
Lead:	Kaleta, Patrick – AIT
Author(s):	Kaleta, P., Kutschera, P. and Havlik, D. (AIT), Austria Leonidis, D. (CERTH), Greece Warde, A. and Hawkes, J. (ECMWF), Germany/UK Molina Lopez, P. (AQUA), Spain Murano, I. (PLENIVS), Italy

Revision history

Date	Version	Contributor/Reviewer	Description
03/06/2024	First Draft	Denis Havlik & Peter Kutschera (AIT)	Drafted the ToC
10/08/2024	Second Draft	Patrick Kaleta (AIT)	Updated section 3, 4 and 5.
29/08/2024	Third draft	Patrick Kaleta (AIT), Adam Warde (ECMWF), Dimitrios Leonidis (CERTH), Ivan Murano (PLENIVS)	Integrated Task-leaders' inputs, finalised the sections 6 and 7, Executive summary.
18/09/2024	Final draft	Patrick Kaleta (AIT)	Consolidated all inputs from task leaders and the feedback gathered on plenary meeting on 05/09/2024. Performed grammar check and formatting.
27/09/2024	Final review	Patricia Molina López (AQUA) Manuel Argamasilla Ruiz (CET)	Internal review of the documents including comments and suggestions to the authors. …
01/10/2024	Final deliverable	Denis Havlik (AIT) → EC	Finalization and upload to Participants portal

Legal Disclaimer

All information in this document is provided "as is" and no guarantee or warranty is given that the information is fit for any particular purpose. The user, therefore, uses the information at its sole risk and liability. For the avoidance of all doubts, the European Commission and CINEA has no liability in respect of this document, which is merely representing the authors' view.

Table of Contents

List of Figures

List of Tables

List of Acronyms

API	Application programming interface
CC	Climate Change
CI/CD	Continuous Integration / Continuous Deployment
CDS	(Copernicus) Climate Data Store
CLIMAAX	CLIMAAX - CLIMAte risk and vulnerability Assessment framework and toolbox (Grant agreement ID: 101093864, 2023-2026; https://cordis.europa.eu/project/id/101093864)
CoP	Community of Practitioners
DoA	Description of Action
DOI	Digital object identifier
EC	European commission
FAIR	Findable, Accessible, Interoperable and Reusable
GA	Grant Agreement
GUI	Graphical user interface
ICARIA	ICARIA - Improving ClimAte Resilience of crItical Assets (Grant agreement ID: 101093806, 2023-2025; https://cordis.europa.eu/project/id/101093806)
JSON	JavaScript Object Notation
KISS	Keep it simple, silly
KNOWING	KNOWING - Framework for defining climate mitigation pathways based on understanding and integrated assessment of climate impacts, adaptation strategies and societal transformation (Grant agreement ID: 101056841, 2022-2026; https://cordis.europa.eu/project/id/101056841)
LEAN	Project management approach that aims to minimise unproductive work through teamwork, empowerment, and continuous improvement of the workflows.
MU	Mission User
netCDF	Network Common Data Form
RA	Regional Authority
REST	Representational State Transfer
SMART	Specific, Measurable, Achievable, Relevant, Time-bound
SSO	Single Sign-on
TRL	Technology Readiness Level
URI	Uniform Resource Identifier
WP	Work Package
WPL	Work Package Leader
xarray	Open source project and Python package that introduces labels in the form of dimensions, coordinates, and attributes on top of raw NumPy-like arrays,

Glossary

Backend	In software development, the backend refers to the server-side infrastructure that handles data services, logic, and communication, enabling an application to function properly
Climate impacts	The consequences of realized risks on natural and human systems, where risks result from the interactions of climate-related hazards (including extreme weather and climate events), exposure, and vulnerability. Impacts generally refer to effects on lives; livelihoods; health and well-being; ecosystems and species; economic, social and cultural assets; services (including ecosystem services); and infrastructure (based on IPCC, 2018)
CoverageJSON	Standardized format for publishing spatiotemporal data to the Web. The primary design goals are simplicity, machine and human readability and efficiency. (https://www.ogc.org/standard/coveragejson)
C3S	The Copernicus Climate Change Service is an information service provided by the Copernicus Earth Observation Programme of the European Union. (https://www.copernicus.eu/en/about-copernicus)
Docker	Is an open-source platform allowing developers to build and run applications in an encapsulated (containerized) environment, which makes deploying these applications in various environments easier. (https://www.docker.com/)
Frontend	In software development, a frontend describes the (usually) graphical interface that a user sees when interacting with an application
GitLab	GitLab is an open-source code repository, allowing for collaborative software development. (https://about.gitlab.com/)
GitHub	Cloud based platform for collaborative code development. (https://github.com)
Jupyter Notebook	Jupyter Notebook is an open-source software, used for interactive, web-based, computational documents. (https://jupyter.org/)
netCDF	Set of software libraries and machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data. It is also a community standard for sharing scientific data. (https://www.unidata.ucar.edu/software/netcdf/)
xarray	Open source project and Python package that introduces labels in the form of dimensions, coordinates, and attributes on top of raw NumPy-like arrays (https://xarray.dev/)

Executive summary

This report is a deliverable D3.1 – ClimEmpower Application Architecture of the ClimEmpower Horizon Europe project (GA number 101112728). The project's goal is to enhance climate resilience of five South-European regions facing high climate change risks, through combination of capacity building, knowledge transfer and development of data-driven decision support tools that aid regional stakeholders in planning climate change adaptation strategies.

This deliverable outlines the envisioned **ClimEmpower application and system architecture,** which aims to facilitate these objectives by providing a **flexible and scalable framework that maximizes reuse of existing data, algorithms and software and facilitates transferability potential across ClimEmpower case study to other European regions.**

The architecture is modular and designed with openness, transparency, and reusability in mind, ensuring that it can be tailored to different regional contexts without requiring extensive technical expertise. Key aspects of ClimEmpower architecture are:

- **Modular Components**: Independent, configurable modules allow customization for each region's needs.

- **Data Integration**: The architecture supports accessing and transforming open climate data from sources such as the Copernicus C3S Climate Data Store, alongside local datasets.

- **Visualization**: The system offers specialized visualization components to present complex data in user-friendly formats tailored to various expertise levels.

- **Incremental Application Design:** architecture allows building of usable applications with different levels of complexity, for different types of users - from simple but efficient data access services, over developer-friendly algorithm implementations facilitating rapid prototyping, development and validation of algorithms, to single-page mashup/dashboard applications and multi-page applications supporting the user's strategic decision-making workflows for climate adaptation.

The development of frontend applications will follow an agile, iterative process, closely coordinated with stakeholders from each region to ensure tools are tailored to their needs. Frequent feedback loops and co-design meetings and/or workshops will guide the development, enhancing user engagement and gradually familiarizing the stakeholders with the data, algorithms, visualisation possibilities and decision support possibilities in early stages of the development. This collaborative development approach will reduce the risk of mis-developments, make the final tools more usable/practical and also reduce the need for extensive training later.

The architecture is supported by modern deployment technologies such as GitLab for version control and CI/CD pipelines, and Docker for containerization. These tools ensure consistent, reliable deployment across different environments. Comprehensive documentation will be put in place to ensure smooth operation by the regions participating in this project and allow for reuse in other regions.

1 ClimEmpower summary

ClimEmpower is a Horizon Europe collaborative research project dedicated to addressing the ongoing Climate Crisis (CC) in Europe by empowering the regional stakeholders in some of the most vulnerable European regions (Figure 1).
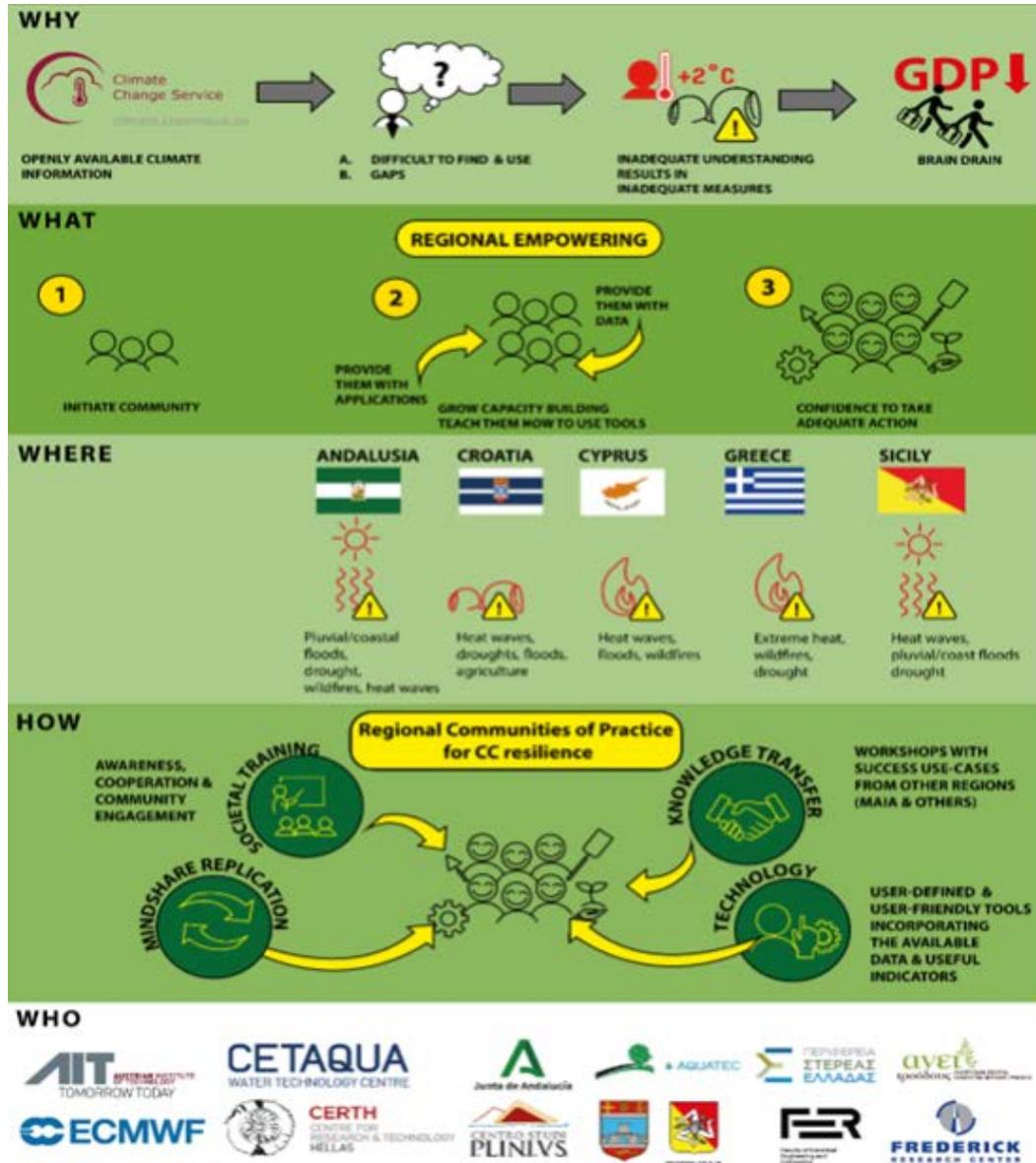

Figure 1: ClimEmpower at a glance: why, what, where, how and who.

1.1 Project Context

Climate risks results from a combination of hazard, exposure, and vulnerability. Addressing all three aspects is crucial for the effective increase of regional resilience. However, exposure, vulnerability, and related aspects, such as adaptive capacity, strongly depend on available knowledge and climate literacy. Consequently, the global climate crisis frequently has a higher impact on socioeconomically vulnerable regions, thanks to a higher human and economic potential for addressing the issue in more affluent regions. To maximize its impact, ClimEmpower has therefore chosen to address the EU regions featuring a combination of high potential CC impacts and low and/or stagnant regional GDP/capita. This is mainly the case for regions in South and Southeast Europe (Figure 2).
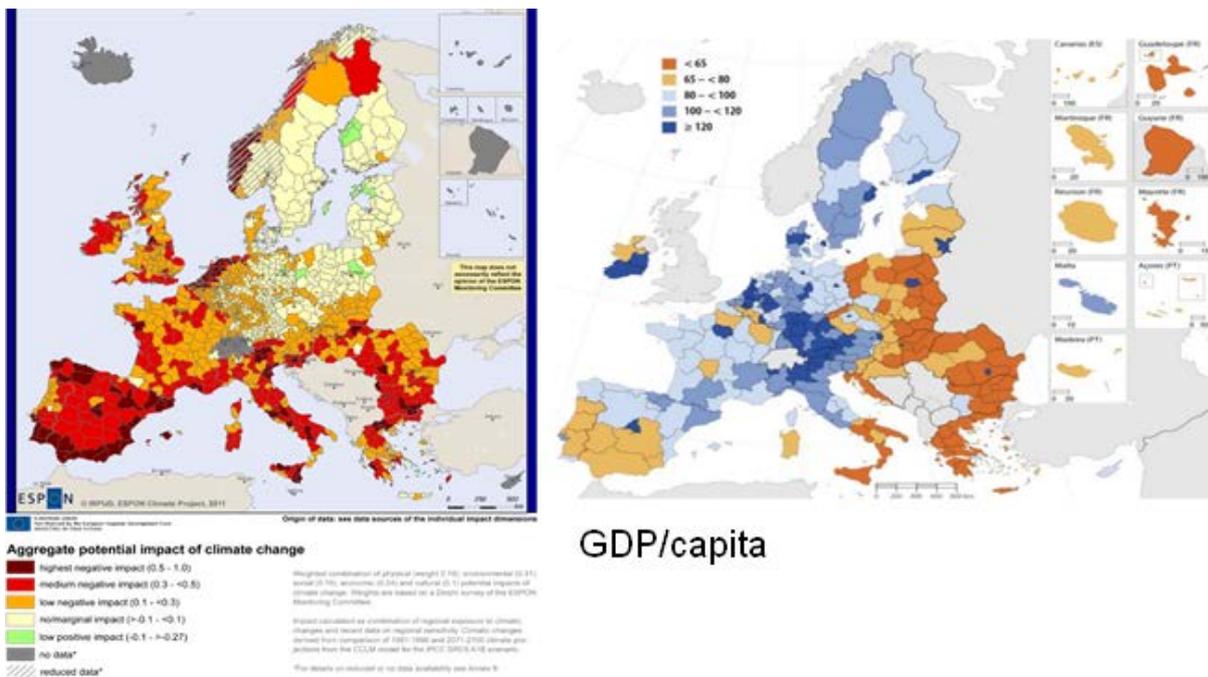
Figure 2: left: aggregated potential impact of climate change (https://www.espon.eu/climate-2012); right. GDP/capita (based on https://ec.europa.eu/eurostat/web/products-eurostat-news/-/ddn-20210303-1)

The context the project addresses is thus one of an ongoing global warming, high regional vulnerability, and low coping capacity of the participating regions.

The **overarching strategic objective of ClimEmpower** is to empower the Regional Authorities (RAs) and other Mission Users (MUs) in five EU-regions featuring a combination of exceptionally high climate hazards and exceptionally low coping capacity. This will be achieved by improving their collective understanding of the Climate Change (CC) hazards, risks and resilient development pathways and supporting their knowledge-based regional planning and development through provision of relevant data, knowledge, and user-defined and user-friendly decision support applications.

1.2 Project Objectives

To achieve this overarching goal, **ClimEmpower has identified six SMART[1] Strategic Objectives** (**SO**), each one related to one or several work packages.

SO1 Understand regional background, challenges, and expectation (WP1, societal)

SO2 Addressing the gaps in availability and usability of CC data and services (WP2 and WP4, scientific)

SO3 Identification, definition, estimating, and communication of climate impact/resilience indicators suitable for local end-users (WP2 and WP4, scientific)

SO4 Simplify access to CC data and development of end user applications (WP3, technological)

SO5 Empower the regions to activate and enhance their potential for addressing the climate change challenge. (WP4, societal)

SO6 Ensure the use and impact of the ClimEmpower outputs (WP4 and WP5, scientific and societal)

The SOs have also been classified according to different categories: societal, contributing to improved dialogue, awareness, cooperation and community engagement as highlighted by the European Climate Pact (SO1, SO5); scientific, corresponding to research activities for advances beyond the state-of-the-art (SO2, SO3); technological, suggesting and/or developing novel solutions, integrating state-of-the art and digital advances (SO4); and outreach, aimed at sharing ClimEmpower results to a broader scientific and non-scientific audience, including additional regions and communities, to maximize project impact (SO6).

> ClimEmpower's key ambition is to **prove beyond doubt that CC-resilience should, and can, be an integral part of regional development everywhere in EU and beyond it**. That is, we anticipate that the regional stakeholders will recognise that CC-resilient development pathways offer multiple benefits to them, including but not limited to higher quality of life and reviving economy, and that these can be understood using available data, tools, and services. Second key ambition of the project is to **help the regions address the CC resilience in key community systems addressed in five ClimEmpower trials**.
>
> Underlying philosophy of the project is to "**help the regions to help themselves**". This will be achieved through various mechanisms, including co-creation and mediation of the regional "**Communities of Practice**", provision of the **Climate Change -resilience training materials**, as well as in provision and training in use of the user-centric data and services – including those that have already been made available through previous research projects and EU initiatives.

---

[1] Specific (related to WPs), Measurable (by relevant KPIs), Achievable (the WPs in which they will be achieved are listed), Realistic (since they are referred and explained in the methodology section), and Timebound (each KPI is related to a deliverable and a month of achievement).

2 Introduction

2.1 Deliverable summary

This Deliverable is part of **WP3 – User-centric CC-resilience enhancing services** – and describes the envisioned application architecture of ClimEmpower. It details the primary features of the applications that will be developed during the project and how this development and deployment process will look like.

2.2 End-users for ClimEmpower services

In ClimEmpower, users will come from diverse background with various experience and different levels of domain knowledge and technical expertise. These users can generally be categorized into three main types:

1) **Decision-makers** will primarily include regional policymakers, and authorities responsible for climate adaptation strategies. They require simplified, high-level tools that offer clear insights, risk assessments, and decision support services to aid in developing and implementing climate resilience plans. Applications for this user type will consist of a combination of single-page **dashboard/mashup** applications and **multi-page DSS** tools implementing some of the user's decision workflows.

2) **Technical experts** (data scientists, climate specialists, software developers, modellers) will need access to detailed climate data, advanced analytical tools, and technical documentation. There focus will be less on graphical interfaces and more on working directly with the provided data services (access and processing services) through APIs. For this user type the developed **Jupyter Notebooks** will serve as the primary applications.

3) The **general public** will benefit from educational materials, interactive visualizations, and simplified data representations, helping them understand climate risks and adaptation options in their region. This will be achieved mainly using single-page **dashboard/mashup** applications.

2.3 Results and expected impacts

This deliverable, on one side, serves the tool developers by defining the general procedures for the development & deployment process and defines the crucial features that need to be addressed by the developed applications. On the other hand, the end-users of the envisioned applications can use this deliverable as a first guide in assessing the relevancy of the proposed applications and get an overview of the next steps taken by WP3.

The application architecture not only provides an overview of the different components and services that need to be implemented but also details the upcoming development process step by step, ensuring that both developers and end-users have a better understanding of the upcoming tasks and outcomes. In a first step, simple application prototypes will be constructed based on Jupyter Notebooks from the CLIMAAX project. These first application prototypes, which are primarily oriented towards more technically experienced users and expert users (climate data scientists and developers), will serve as a starting point for development and testing the impact calculation and other algorithms, and as the basis for further discussions between developers and end-users to specify the needs and requirements of the regions. In phase 2, the algorithms implemented in the notebooks will be transformed into data/indicator services, and more user-friendly GUIs, catering more specifically to the requirements posed by the

regions developed in the form of dashboard-like, single-page mashup applications. The final step (phase 3, described in more detail in section 6.1) will entail the construction of more complex, multi-step applications from these single-page elements. These applications will support the regions in implementing their strategic decision-making workflows for climate adaptation.

The architecture and development process established in this deliverable will help ensure that SO4 – Simplify access to CC data and development of end user applications – can be successfully achieved within the project duration. By developing the set of frontend components in three steps, we can assure that user feedback can be incorporated in time for the next phase, which reduces the risk of developing unnecessary functionalities and overlooking missing features. Moreover, the three different levels of frontend GUIs can target the various user types identified. Climate data scientists will be able to continue working with the Jupyter Notebook-based applications developed in phase 1, while decision makers, planners or other users might choose to either work with single-page dashboard applications or more sophisticated workflow-supporting DSS applications. The three development phases for the GUIs are explained in more detail in section 6.1.

2.4 Relation to other work

This deliverable is the first of three in WP3 and builds the foundation for **D3.2 – Reusable libraries and services for climate feature extraction, data** – which is scheduled for release in M24 (August 2025), and **D3.3 – HTML5 framework and user-centric applications for climate resilience** – which is scheduled for M33 (May 2026).

The deliverable relies on results gathered by WP1, namely from the deliverable **D1.2 – ClimEmpower Scenarios** – which collected, analysed, and summarized the different requirements posed by the five regions to the tools that WP3 is tasked with developing.

Furthermore, this deliverable is input to WP4/task 4.3 "Co-design of tools, services and trials for increased regional resilience". Based on the information in this deliverable, the WP4 team will be able to make initial assessments regarding the definitions of the trial scenarios and start the intensive co-creation process with the five case study regions to co-design the applications and the ways of testing and evaluating them

2.5 Data, security, and ethics

No (new) data sets relevant for ClimEmpower Data Management Plan were used or produced in preparation of this deliverable.

2.5.1 Data interoperability

This deliverable addresses the issue of data interoperability through provision of the data access API and library that is introduced in sections 4.1 and 5.1 of the document. This service primarily aims to simplify the task of extracting complex features from a datacube, such as polygon regions or spatio-temporal paths. In addition, it also provides a set of python modules that allows users to retrieve, interpolate, regird, plot and map data from ECMWF and other earth science dataset sources

Primary output types for the data access library are shapefiles and CoverageJSON, but a conversion kit will also be provided to support other formats, such as xarray and netCDF. Moreover, the GRIB data format, which is the standard output format for most existing ECMWF services and commonly used in the meteorological and climate services to store historical and forecast data is also supported.

In terms of the API, the service will provide both an Environmental Data Retrieval (EDR) REST interface and a simpler REST API for developers that aren't familiar with the OGC EDR standard. EDR API will allow us to easily combine the data access service with existing tools and applications that already

support the OGC EDR standard and also simplify the use of the new service by geospatial community who may not be familiar with meteorological datasets, but have used EDR in other applications.

2.5.2 Data accessibility and reuse

Primary data sources used in the preparation of the D3.1 deliverable consist of the ClimEmpower GA and the deliverable D1.2 produced by WP1 – Regional CC-resilience background, challenges, and expectations. Both sources were used to assess the requirements placed upon the applications developed by WP3.

Table 1: Data used in preparation of ClimEmpower deliverable D3.1

Data set name	Format	Size	Owner & re-use conditions	Potential Utility within and outside	Unique ID
ClimEmpower Grant Agreement	PDF	5.2MB	Sensitive	• Grant Agreement is a legally binding contract between the consortium and the EC. • Document is available from SyGMa	• GA number 101112728 • On SyGMa
Deliverable D1.2 ClimEmpower Scenarios	PDF	5MB	Public	• Document is available on Projects' document repository, in "Project Outputs" directory. • Document will be made available on CORDIS after the M18 (Feb 2025) project review	• On Teams (Project Outputs) • CORDIS link N/A yet

2.5.3 Security and Ethics

The work performed in this deliverable isn't considered sensitive in terms of ethics or security.

3 Background and requirements

ClimEmpower aims to bolster climate change resilience in five South-European regions, which are highly vulnerable to climate change and simultaneously possess low adaptive capacity individually. By fostering regional CoPs and co-creating region-specific resilient development strategies, the project will leverage user-driven climate applications, capacity building, and best practices from other European projects. And while the project itself focuses on these five selected regions, the overall goal for the envisioned ClimEmpower services should be to aim for a region-independent and flexible solution, giving other regions the opportunity to benefit from the project's outcomes. To achieve this the proposed system architecture of the comprising ClimEmpower applications needs to be flexible and adaptable, allowing for multiple options for deployment and offering a clear and well-documented structure for the developed tools and services, as well as provide interfaces for connecting already existing legacy systems used by the different regional administrations.

3.1 Requirements derived from GA

Based on the vision laid out in the grant agreement, the primary goal for the ClimEmpower services is defined as strategic objective number 4 (SO4), which aims to simplify the access to climate change data through easy-to-use end user applications. To achieve this, SO4 calls for methods and tools to:

4) **Access CC data** gathered from both within and outside of the project itself.

5) **Transforming that data** by either structuring, enriching, or simplifying it for further usage.

6) **Visualizing the transformed data** in various ways depending on different user groups.

7) Offer on top of that analytical tools suitable for **aiding decision support.**

8) **Modularity and reusability**

The grant agreement mentions a "CC-resilience toolbox", a kit of smaller, individual application components that can be combined to build the ClimEmpower service for a specified region. These individual components need to provide dedicated interfaces to allow for information transfer between components and should ideally minimize the need for adaptations on the level of application source code. Instead of this the adaptation to specific regional requirements should be configurable on a higher level, lowering the required technical skills on the end user side.

As a prime source for open climate data the Copernicus C3S Climate Data Store (C3S CDS) was identified in the grant agreement and a considerable amount of effort allocated to developing necessary tools to access and process this data and make it available to the other components developed by ClimEmpower. Hence, **data access and transformation capabilities** have been identified as one of the key aspects to be considered in the design for the ClimEmpower application architecture.

Another key focus that needs to be reflected in the architecture are the visualization capabilities of the toolbox. Potentially large amounts of complex data need to be presented to users with different levels of knowledge in the context of climate. This is further complicated by the fact that only a small part of the available data will be of interest and relevance for each of the ClimEmpower trial regions and their CoPs. To accommodate such requirements, the

GUI components responsible for visualizing the data need to be built in a modular way, where each component will handle a very specific form of data visualization. These **micro frontend components** can then be joined together to form the final GUI of the ClimEmpower service for a regional CoP. To allow developers working on these components in an agile way, enabling them to adapt to changing user requirements fast, **low-code and no-code environments** should be used when possible.

To lower the hurdle for others to reuse the developed components, these components should be made **open source**, provide detailed and extensive **documentation**, and be designed with **reusability** in mind from the start. Open sourcing the components will not only foster transparency and collaboration but also encourage community contributions that can enhance the functionality and robustness of the tools. Comprehensive documentation will ensure that users and developers can easily understand, deploy, and modify the components to suit their specific needs. Furthermore, designing with reusability in mind involves adhering to standardized coding practices, **modular design** principles, and providing clear **APIs for integration**. This approach will facilitate the adoption of ClimEmpower tools across different regions and projects, promoting widespread use and adaptation. Additionally, offering tutorials and other training material can provide practical insights into the application of these tools, thereby lowering the technical barriers and empowering a broader range of stakeholders to engage in climate resilience efforts.

3.2 Requirements derived from WP1 / D1.2

Work package 1 – **Regional CC-resilience background, challenges, and expectations** – and especially deliverable D1.2 **ClimEmpower scenarios** delved deeper into the needs and expectations of the different stakeholders, which lead to further requirements posed on the application architecture.

One important point that emerged from the discussions with the regions is the importance of data availability and data management. While some regions offer large amounts of openly available data, as well as private datasets, other regions were lacking behind in this regard. The system architecture must consider the possibility of data not being available in certain regions and offer possible solutions around this problem (for example by offering information and/or algorithms for bridging such data gaps).

Another point which was mentioned by several regions was the need for training and educating personnel in the usage of the provided tools. Training and educational measures are crucial for ensuring the effective utilization of the ClimEmpower tools and services. Personnel involved in planning and decision-making processes require comprehensive training to understand and operate the technological solutions, interpret climate data accurately, and make informed decisions. While ClimEmpower training efforts are coordinated through T4.1 "Provide educational materials" WP4 task, the educational materials related to ClimEmpower architecture, services, tools and applications will have to be provided by WP3 partners. Moreover, part of the "educational needs" raised by the regions is for the applications raising public awareness about climate hazards and adaptation strategies, as a way to strengthen community resilience and engagement.

Summarizing key conclusions from D1.2 **ClimEmpower scenarios,** it can be said that while these regions share many of the hazards, the sectors most vulnerable to those hazards, and therefore also the impacts and the adaptation measures differ from region to region. This finding further stresses the necessity for the developed solutions to be easily adaptable and configurable to the specific characteristics of the different regions. Focusing on the needs stated by the five regions, the ClimEmpower tools must consider:

- The regions' desire to have applications capable of **addressing multiple hazards, impact and mitigation pathways**.

- The functionalities expected by all 5 regions – **risk mapping, data filtering** and **prioritization of adaptation measures.**

- Help structuring a "**Regional Climate Framework**".

- Offering support for regions to **learn from previous extreme climate events and prepare and recover** from such events in the future.

The system architecture for ClimEmpower must be designed with **modularity and configurability** at its core, allowing for easy adaptation to specific regional needs without requiring extensive technical skills. Robust **data processing and visualization** capabilities are essential to handle complex data sets and present them in an easy-to-understand manner for various user groups. To promote reuse, transparency, and community collaboration, all components should be **open source** and accompanied by **comprehensive documentation**. Utilizing low-code and no-code environments will enable agile development and rapid adaptation to the changing user requirements. Finally, the architecture must include decision support tools to aid in strategic climate adaptation planning and support informed decision-making processes.

4 Functional architecture

Based on the requirements, the ClimEmpower system architecture needs to be modular, flexible, and user-centric, ensuring that it can easily adapt to the diverse needs of different regions and stakeholders while being accessible to users with varying levels of technical expertise. The key features to be achieved by the architecture are:

1) **Modularity and Configurability** – The architecture should be composed of independent, modular components or microservices (**Component-based Design**) that can be easily configured and reassembled to meet the specific requirements of each region. This approach allows for the customization of tools without altering the core codebase, facilitating the adaptation of the system to different regional contexts.

2) **Data Integration and Processing** – The architecture must include robust data integration capabilities (for commonly used data formats and standards) to access, structure, enrich, and simplify climate data from multiple sources, including the Copernicus C3S Climate Data Store (C3S CDS), regional data repositories, and stakeholder-provided datasets (**Interoperability**). These data access services should ensure the seamless flow of data across components, supporting accurate and timely analysis. The system should be capable of integrating both open and licensed datasets, as well as governmental and private data, ensuring comprehensive coverage and usability of relevant climate information.

3) **Advanced Visualization Tools** – The architecture should feature specialized, modular GUI components tailored for different types of data visualization (e.g., risk maps, impact assessments, forecasting models). These components can be combined to create user interfaces that cater to the specific needs (**Modular Visualization Components**) and expertise levels of diverse user groups, from policymakers to the general public. Emphasis on intuitive, user-friendly design is crucial to ensure that the visualized data is accessible and understandable, even for non-experts.

4) **Decision Support Systems** – The architecture should incorporate analytical tools that assist users in making informed decisions about climate adaptation strategies. These tools should be capable of processing complex data and providing useful insights tailored to regional needs.

5) **Training and Educational Support** – The system should include built-in training and educational resources, such as interactive tutorials, user guides, and case studies, to ensure that personnel and the general public can effectively utilize the tools and understand the implications of the data provided. The architecture should facilitate **knowledge transfer** by linking to external resources, and enabling peer-to-peer learning among different regions and stakeholders.

6) **Open Source and Reusability** – All components should be developed as open source, encouraging collaboration, transparency, and the reusability of tools in other projects or regions. Extensive and detailed documentation should be provided to support the reuse of components, making it easier for other regions or projects to adopt and adapt the ClimEmpower tools.

Following those requirements, the project partners have designed the initial version of the implementation architecture, as presented in Figure 3. The various GUI components are part

of the ClimEmpower Frontend, which will be fed with data and controlled by the services of the ClimEmpower Backend. This backend also facilitates the connections to already existing legacy systems and services. The individual components and services are further examined in the subsequent sections.
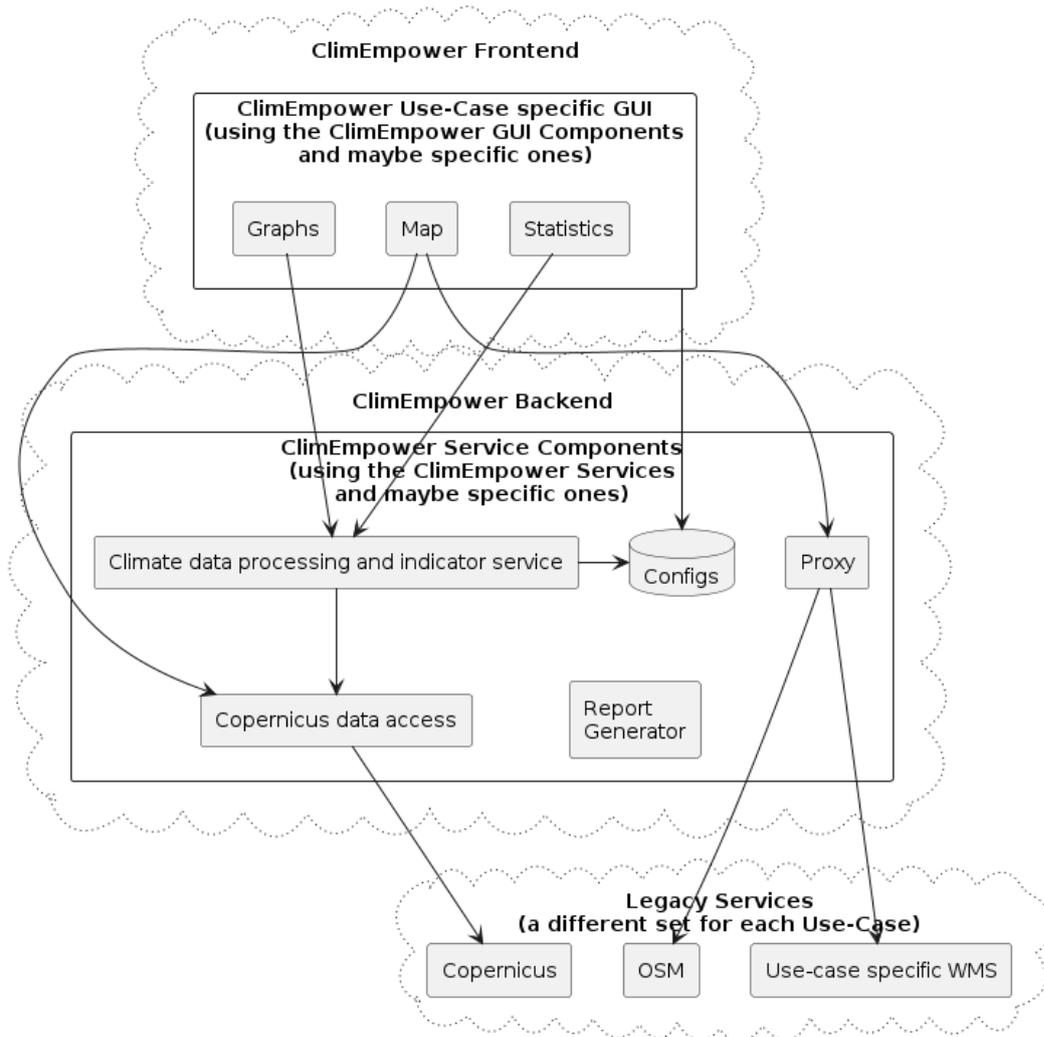

Figure 3: ClimEmpower architecture – component view

4.1 Data services

4.1.1 Data Access

Data access is integral to the ClimEmpower project. Without data tailored to users' needs the goals of the project become much more difficult to attain if not impossible. We intend to provide data that is easy to access, understandable and tailored to the region's needs. These data access services will also be supported and maintained post the ClimEmpower project by ECMWF.

Main users of the data services are the data scientists, modellers, and the service developers.

ClimEmpower addresses these challenges in a number of ways. For a start, the project will implement a new way for accessing the Copernicus data store (CDS, https://cds.climate.copernicus.eu/), which contains numerous datasets such as ERA5, a reanalysis for the global climate and weather for the past eight decades, as well as many other useful datasets that can be used by the regions in this project to estimate hazard and risk assessment at a more granular level. While a data access API for these datasets

already exists, the new API developed in T3.1 "Develop feature extraction library and service for Copernicus data" WP3 task will be more user friendly and allow the users to easily retrieve the part of the data that is relevant for their region and use case, rather than downloading entire data set and extracting required parts of it locally. This will lower the transfer bandwidth required for retrieving the data and allow us to utilise Copernicus data services more efficiently, while increasing the speed of service development and lowering their complexity and processing needs.

This functionality will be made available both as a python library and as a REST API that users can request data from if they wish to more easily integrate this data access into their workflows.

CDS and the ClimEmpower python/REST API service are both developed and hosted by ECMWF and part of the European Copernicus program. The code for the libraries developed within ClimEmpower will also be published as open source with ongoing maintenance from ECMWF and Copernicus.

The library will be modular and flexible enough that other datasets or data sources can be added in the future as backends with minimal need to change the underlying code.

4.1.2 Data Formats

Data formats are integral to data services. If users do not receive a format they can work with, they will not use the data service. The existing formats provided by the CDS are specific to the meteorology community and while they are useful in that context users in ClimEmpower may not be familiar with these data formats, so we intend to use an open and user-friendly format.

In this case we will be providing a community standard data format (see 5.1.2) that is interoperable with existing tooling, and we will also provide our own custom tooling to convert and interpret the data in other popular formats that users are likely to be familiar with. This tooling will also provide a number of other utility functionalities such as plotting, mapping and regridding tools.

4.1.3 Data Interfaces

Ultimately users will interact with these pieces of software via interfaces, whether they are a python interface or a REST interface. In this case both will be provided allowing users easier development phases where they can use the python library to quickly experiment and test the data services and their models with this data, and once this phase is done and the operational phase begins users can then use the REST interfaces for their applications.

As this project is made up of partners from many different domains, we will provide multiple interfaces. The main interface for data access is tailored toward the meteorological community which may not be familiar to other partners. For this reason, we will also provide an EDR (Environmental Data Retrieval) API. This is an OGC community standard that should be more familiar to users from an earth observation background. In this way users can interact with the data using existing EDR services if they are familiar with them and can point to the data service endpoint.

4.1.4 Model outputs

ClimEmpower data services will also provide access to indicators and hazard and risk maps that are made available by the regions or developed within the project, to fill the regional data gaps, for example the regional risk maps addressing threats like floods, heatwaves, droughts, and more (Figure 4). The complexity of these models will vary depending on the required level of detail. For instance, hydrodynamic flood and heatwave models will be more case-specific and time-consuming, while drought models may require less effort and could be transferable to other ClimEmpower regions. It's important to note that the development of these models will be managed externally but in parallel, with the final results being shared through the service.

Figure 4: On the left, drought risk map in Marbella (Costa del Sol). Right side shows flood map in Marbella.

4.2 Processing and indicator services

Using the data available from the various services described above and those provided by each region, the processing and indicator services will construct climate indicators that can be customised by the user according to the priorities set by regional and national adaptation policies and individual projects. The necessary climate-related data as input for these climate indicators will be gathered in WP2.

Main users of processing and indicator services are the data scientists and application developers.

In particular, the ClimEmpower processing and indicator services shall facilitate construction of climate hazard indicators and macro-indicators of environmental vulnerability and socio-economic vulnerability. Macro-indicators are the indicators whose values cannot be directly measured. Their values are determined by combining the values of several simpler sub-indicators whose values are known or can be directly measured. The user will be able to express a level of priority to be given to each sub-indicator, thus influencing the final value of the macro-indicator.

This level of interaction is intended to allow the user to understand the complexity of the specific areas of interest and to make the appropriate design and policy choices to properly respond to the specific critical issues.

To further illustrate the indicator customisation procedure and the consequent generation of action-based recommendations, an example of the process is given in Figure 5 and Figure 6. Scenario A presents the default case where each sub-indicator is valued equally in its importance. This default behaviour will make it possible for users to quickly get first calculation result without the need of levering the importance of the individual sub-indicators. The possibility of fine-tuning the macro-indicators by assigning different weights to the sub-indicators is shown in scenario B. The set of applied macro-indicators in combination with selected actions, will return different lists of suggested recommendations, helping decision-making processes. Envisaged process for dveloping such sevices is presented in section 5.2.

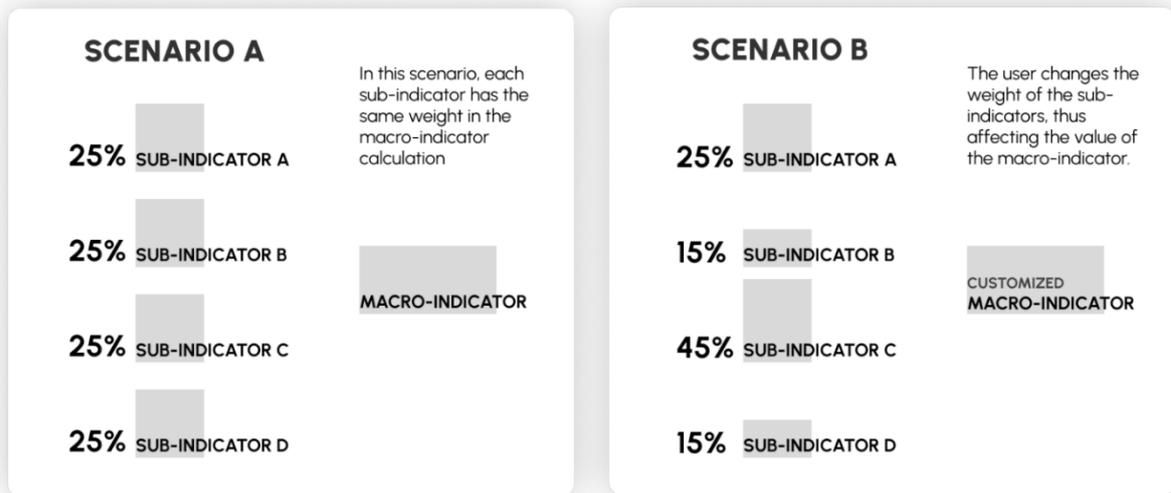**Figure 5: Left side with default scenario, on the right side a customized macro-indicator with changed weights for the sub-indicators.**
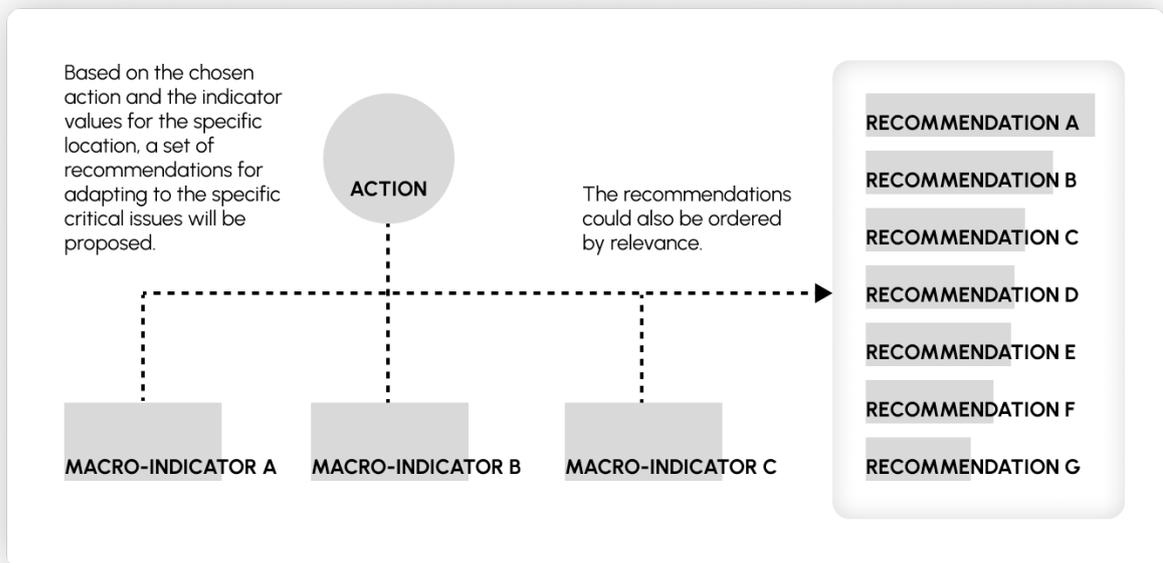

Figure 6: List of recommendations is being influenced by the set of macro-indicators.

4.3 GUI/Frontend components

4.3.1 Identifying user types

In accordance with the objectives outlined in the grant agreement and specifications, the graphical user interface of ClimEmpower will need to be built to be used effectively by different groups of users with varying levels of technical knowledge (see also 2.2 as reference).

During the planning and designing of the architecture the different user types were split into two main categories:

1) **Users with technical proficiency:** This user group consists of users such as software developers and data scientists, people responsible for creating models and services utilizing ClimEmpower Backend services.

2) **Users with limited technical knowledge**:

   a. **Decision makers, general public –** This user group mainly consists of people who simply need data visualization from services available in the ClimEmpower project to simplify decision making process.
   b. **Planners, consultants** – This group of users performs workflows with well-defined steps on a regular basis, generate reports, and possibly data manipulation.

Therefore, the graphical user interface will consist of distinct types that will serve the needs of those groups.

4.3.2 High-level graphical user interface outline

Developer applications (for data scientists and modellers)

This type of applications empowers the data scientists and modellers, to create, test and validate data processing algorithms and discuss the outcomes with end-users. They facilitate rapid prototyping, enable quick iterations and experimentation. The results of these prototypes can be efficiently communicated to end users, and to the software developers, thus bridging the gap between data and algorithms, technical implementation thereof and practical applications. By streamlining this process, the developer application type serves as a catalyst for accelerating discussions and collaboration among developers, scientists, and end-users. This enhanced communication flow not only speeds up the development cycle but also ensures that the final product aligns closely with user needs and expectations. "Developer mode" applications will be primarily used to facilitate co-creation (in close cooperation with task T4.3 of WP4), allowing the data scientists, modellers and end-users to quickly test out the algorithms and ideas of data visualisation, while also simplifying the work of the developers that need to transform these prototypes into solid data and indicator services and user-friendly GUI applications.

Single-page applications (configurable dashboard/mashup application)

The second type of the application features a highly customizable dashboard/single page mashup applications, designed to cater to the diverse needs of various user groups. This application type allows the **developers and technology-savvy end users** to tailor the application user interface to their specific preferences and requirements, but we also expect it to result in single-page applications fully resolving some of the end user needs (e.g. overview presentation and simple analysis of future risks for **policy makers, strategic level decision making** or for the **general public**).

Our capability to efficiently develop such applications will critically depends on provision of a component library offering a rich array of visualization tools, including maps, reports, and graphs, all of which seamlessly integrate with the project's available services by the T3.3 "Develop Web-GUI application

components and GUI integration environment" project task. This flexibility enables users to create personalized, data-driven dashboards that present information in the most relevant and actionable format for their particular use case, with a minimal amount of coding or no coding at all. By providing this level of customization, the application ensures that each user group can efficiently access and analyse the data most pertinent to their objectives, thereby enhancing overall productivity and decision-making processes.

Multi-page applications (workflow applications)

Final application type allows the application developers to assemble more complex applications by connecting several single-page applications with minimal amount of coding (low code/no code). For example, such application could lead the users through single-page applications featuring hazards, exposure of various elements at risk, vulnerability, risk/impact estimates and recommended adaptation measures. This application type can be used to realise decision support applications for common climate adaptation assessment and resilient urban/regional planning tasks of interest to ClimEmpower regions. Similarly to single page applications, our ability to efficiently develop such multi-page applications will critically depend on T3.3 "Develop Web-GUI application components and GUI integration environment" ensuring that single page applications can be easily combined into multi-page workflows.

5 Implementation architecture

5.1 Data services

5.1.1 Data Access

The data access services provided by ECMWF are comprised of a stack of software as seen in the table below.

Table 2: Overview of data access service stack

Service/tool name	Purpose	Further information
Polytope Feature Extraction	Polytope is a library for extracting complex data from datacubes. It provides an API for non-orthogonal access to data.	https://github.com/ecmwf/polytope
Polytope-mars	This library provides a higher-level API to allow requests for features such as time series and vertical profiles.	https://github.com/ecmwf/polytope-mars
Covjsonkit	Covjsonkit is a library for encoding and decoding coverageJSON files/objects of meteorological features such as vertical profiles and time series.	https://github.com/ecmwf/covjsonkit
Polytope-client	Polytope-client provides a REST API and python interface for access to hypercube data, stored in various data sources.	https://github.com/ecmwf/polytope-client
Earthkit	A python library providing powerful tools for speeding up weather and climate science workflows by simplifying data access, processing, analysis, visualisation and much more.	https://earthkit.readthedocs.io

As we can see from the table above the software stack for the data access services has a number of components. The primary interfaces that users will interact with are via "Earthkit" and the "Polytope - client library, the other components merely enable these two interfaces but can also be used directly if needed by the users.

The Polytope Feature Extraction (FE) library is used for extracting complex data from datacubes. It provides an API for non-orthogonal access to data, where the stencil used to extract data from the datacube can be any arbitrary *n*-dimensional polygon (called a *polytope*). This can be used to efficiently extract complex features from a datacube, such as polygon regions or spatio-temporal paths. Previously when providing data, we only returned global fields which users would have to process for their local area, however Polytope FE allows us to return arbitrary cut outs from our data cubes allowing the regions to only receive data tailored to their needs and removes the need for post processing on their end.

An easy-to-use interface for Polytope FE, polytope-mars has been developed to allow users to more easily request features that otherwise would be more difficult using Polytope FE alone. Polytope-mars provides an interface that can return features such as time series, vertical profiles and cut outs of countries of regions via shape files or geoJSON geometries. This library will be integrated with the polytope client to expose the interface.

These libraries in conjunction with ECMWF's existing software infrastructure will streamline users ability to return only the data they require in an easy to use format.

The combination of this stack will allow users a single access point to CDS datasets and allow them to request only the region they want via the interface.

The data access will be deployed and hosted by ECMWF on their own infrastructure.

5.1.2 Data Formats

The primary data format that we will be returning to users is CoverageJSON. CoverageJSON is an Open Geospatial Consortium (OGC) community standard. CoverageJSON allows users to return geospatial data in a simple, human and machine readable, web friendly format. It explicitly has types for many interesting meteorological features such as time series, vertical profiles, and trajectories.

As CoverageJSON is an OGC standard it automatically comes with interoperability for some tooling provided by the OGC such as leaflet. However, we have also developed Covjsonkit to allow users unfamiliar with CoverageJSON to easily interact and manipulate this data format. Covjsonkit is used internally with our tooling to create coverages but can also be used to convert CoverageJSON into other formats such as xarray and netCDF. Covjsonkit is also flexible written so new engines to convert CoverageJSON to other formats can be easily added if a particular format is popular with users.

GRIB is the default format returned by most ECMWF services. GRIB is commonly used in the meteorological and climate to store historical and forecast data. However, GRIB can be difficult for users unfamiliar with the format. Our data access services currently only return GRIB as global fields requiring some post processing of the data by the user. Users can still request GRIB in the form of global fields via the data interfaces provided for this project.

5.1.3 Data Interfaces

We will provide two main interfaces for accessing our data. The first is a REST API via the polytope client. Users can integrate calls to this library into their workflows for data retrieval. An open API specification will be provided to users along with the endpoints they can access hosting the datasets.

The other interface that is provided is earthkit. Earthkit is a set of python modules that allows users to retrieve, interpolate, regrid, plot and map data from ECMWF and other earth science dataset sources. Earthkit provides a python library interface for these utilities and uses polytope-mars for interpretation of the polytope request. Covjsonkit is integrated within earthkit so users can manipulate the coverageJSON returned within the same library calls.

As well as our own REST API via Polytope client we will also provide an Environmental Data Retrieval (EDR) REST interface. EDR compliments our own interface, having an overlapping feature set but allowing further interoperability with the geospatial community who may not be familiar with meteorological datasets, but have used EDR in other applications. As it is an OGC standard it also allows the use of existing tools and applications that already use EDR.

5.2 Processing and indicator services

One of the main goals of ClimEmpower is to create indicator services that can help regions understand which strategies will work best in adapting to climate change. Each region involved in the project has its own needs and critical issues but there are also critical issues shared among all regions; the goal is to create macro-indicators that can work for all regions but can also be customized based on specific local needs, if needed.

To achieve this, the indicator service that will be built will need three essential elements:

- **Sub-indicators**: these are the base on which the macro-indicators will be built. In general, each of them will have the same weight in the calculation of the final indicator but the user will have the option to assign different levels of priority to each of them, thus customizing the macro-indicator.

- **Actions**: this is the list of localized interventions of which the user will be able the obtain recommendations.

- **Recommendations**: for each action set by the user, the indicator service will provide a list of recommendations based on the location of the action itself and thus on the specific critical issues identified by the macro-indicators.

In a first step, Jupyter Notebooks will be utilized to bridge the gap between the mathematical algorithms and the implemented services. The Python-based, script-like Jupyter Notebooks can be used to implement and assess the validity of the proposed algorithms with relative ease. Hence, they will be used as a proof-of-concept and a steppingstone towards the goal of having fully implemented and user-customizable processing and indicator services integrated in the backend of the ClimEmpower toolbox.

Similar to the data services outlined in section 5.1, the final step will be to provide a REST API for communication between the processing and indicator services and other services and components within WP3. The usage of a graph database, particularly with the aim of incorporating natural language processing capabilities, is currently being investigated as part of the ongoing efforts in task T3.2.

5.3 GUI/Frontend components

The ClimEmpower GUI will be built using cutting-edge front-end technologies and libraries, ensuring a robust, scalable, and user-friendly interface. Our approach focuses on creating reusable web components that can be seamlessly integrated into the final user-centric UI, catering to both the "Developer type" application and the dashboards/mashups and DSS applications.

Key technologies under consideration include:

- **Jupyter Notebook**: Jupyter Notebook is an open-source software, used for interactive, web-based, computational documents.

- **JavaScript**: As the foundation of modern web development, JavaScript will be extensively used to create dynamic and interactive elements within the GUI.

- **HTML5 and CSS:** These core web technologies will be employed to structure content and style the interface, ensuring a visually appealing and responsive design across various devices and screen sizes.

- **TypeScript**: By utilizing TypeScript, we can enhance code quality and maintainability through static typing, making it easier to develop and scale complex components.

- **React or Angular:** These popular front-end frameworks are being evaluated for their ability to create efficient, reusable UI components. They offer robust ecosystems and tools for building complex, data-driven interfaces.

- **Tailwind CSS:** This utility-first CSS framework is under consideration for its ability to rapidly build custom designs without leaving HTML, potentially speeding up the development process.

By leveraging these technologies, we aim to create a library of reusable web components that can be easily customized and integrated into both the "developer type" application and the single-page and workflow-like applications. This approach will allow for:

- Rapid prototyping and iteration for the "developer type" application utilizing Jupyter Notebook

- Highly customizable and interactive visualizations in the dashboards/mashups

- Consistent design language across different parts of the application

- Improved performance and user experience

- Easier maintenance and updates to the GUI over time

The use of these modern technologies will enable us to create a flexible, efficient, and user-centric interface that meets the diverse needs of ClimEmpower's user groups while maintaining high standards of performance and scalability. Figure 7 shows just such a concept, where a dashboard-like application can be created and changed with a simple drag-and-drop behaviour by selecting the desired components from a list.
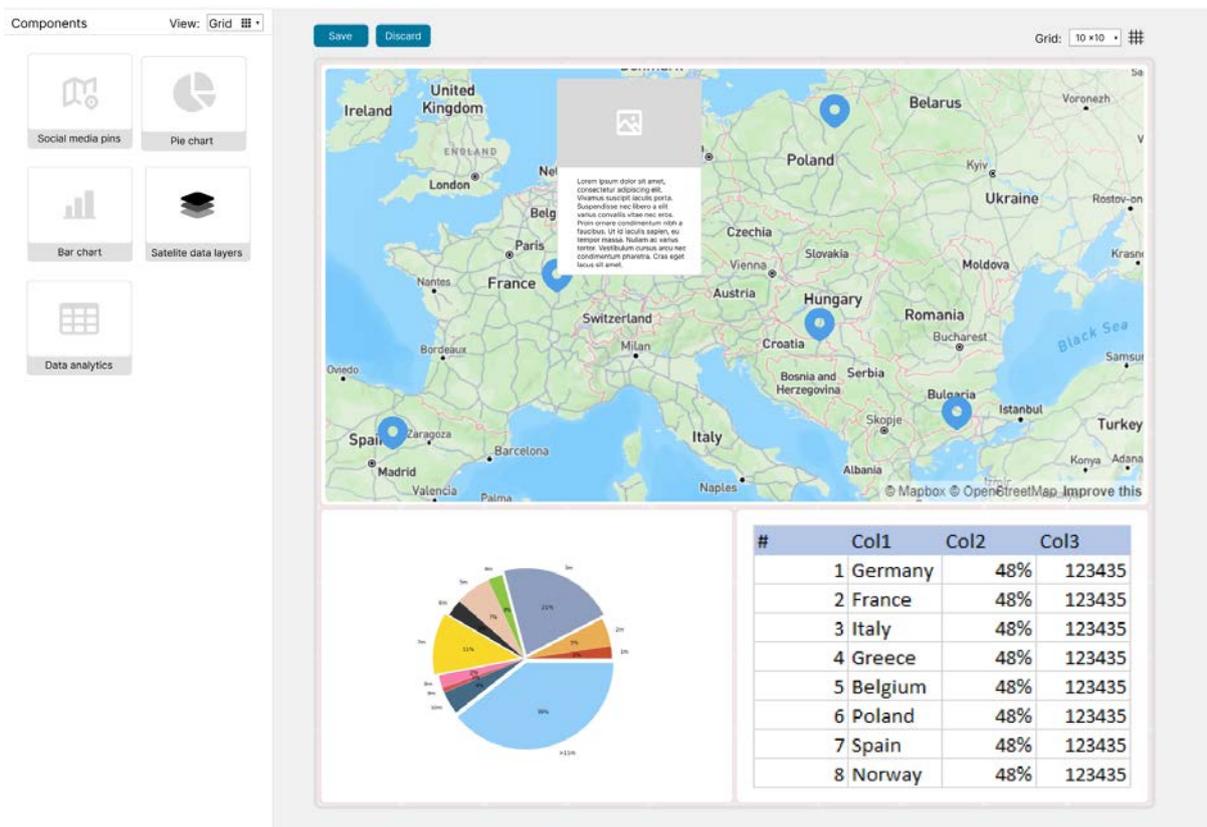

Figure 7: Early mock-up of a ClimEmpower user dashboard creation.

6 Development and Deployment

Task T3.4 – **Development of user-centric applications** – not only includes the creation of individual applications based on the components developed in the previous tasks of **WP3**, but also entails the deployment process for these components. The flexible system architecture described earlier supports multiple deployment strategies, ensuring that the applications can be tailored to the specific needs and constraints of each region. This flexibility allows for a centralized hosting model, where applications are hosted and maintained by one of the project partners, who would then also be responsible for the operation of the deployed services.

Alternatively, the architecture also supports on-site deployment within the regions themselves, where the regions supply the necessary IT infrastructure (web servers and storage space) themselves on which the applications and services will be hosted. This approach can be particularly beneficial for regions with specific data sovereignty concerns or where local data processing is essential to meet legal or regulatory requirements. Moreover, a hybrid, decentralized deployment model is also viable, wherein different applications or components are hosted by various partners and regions.

The architecture is designed to handle these diverse deployment scenarios seamlessly. Regardless of the deployment model chosen, the deployed components will be able to communicate and exchange data effectively, provided that the necessary interface endpoints are properly defined and configured during deployment. This ensures that even in a distributed environment, the system functions as a cohesive whole, enabling consistent data flow and interoperability between components. By accommodating various deployment strategies, the architecture ensures that ClimEmpower can meet the diverse technical and organizational needs of all participating regions and stakeholders.

6.1 Development Process

Task T3.4, focused on the development of front-end applications, will closely collaborate with task T4.3 – **Co-design of tools, services, and trials for increased regional resilience** – to ensure that the applications are tailored to the needs of the end-users. This collaboration is key in order to maintain a user-centric approach throughout the development process. The co-design strategy emphasizes active participation from the end-users, ensuring that their feedback is constantly integrated into the development cycle. This approach not only keeps the development process aligned with the needs and expectations of the target audiences but also enhances the overall relevance and usability of the final applications.

To implement this co-design approach effectively, the tool development will follow an agile and iterative development process. This might involve conducting several meetings, workshops, and/or feedback sessions between the tool developers and regional stakeholders. These interactions will take place multiple times throughout the project, allowing for continuous refinement of the applications based on timely end-user input. The agile approach ensures that any issues or suggestions raised by the stakeholders can be addressed promptly, leading to a more responsive and adaptive development process.

As already mentioned in section 2.2, development of the ClimEmpower frontend applications will be realized in different phases, each focusing on a different "*layer*" of GUIs. In a first step, readily available CLIMAAX Jupyter notebooks will be utilized and tailored to some of the regional needs, in order to facilitate further discussion with the end-users, which will be helpful in guiding the direction for the development of the remaining frontend applications in phase 2 and 3, namely the dashboard-like single-page applications and the workflow-like DSS applications. To briefly summarize the envisioned development phases (see also Figure 8 for reference):

1) **Phase 1** – Rapid prototyping utilizing Jupyter Notebooks, which will serve as a basis for further discussions with end-users and developers.

2) **Phase 2** – Building on the outcomes of the first phase, dedicated GUI components will be developed and used to create first dashboard-like mashup applications.

3) **Phase 3** – Assembling of multi-step, workflow-based DSS applications stemming from components and apps created during the second phase.
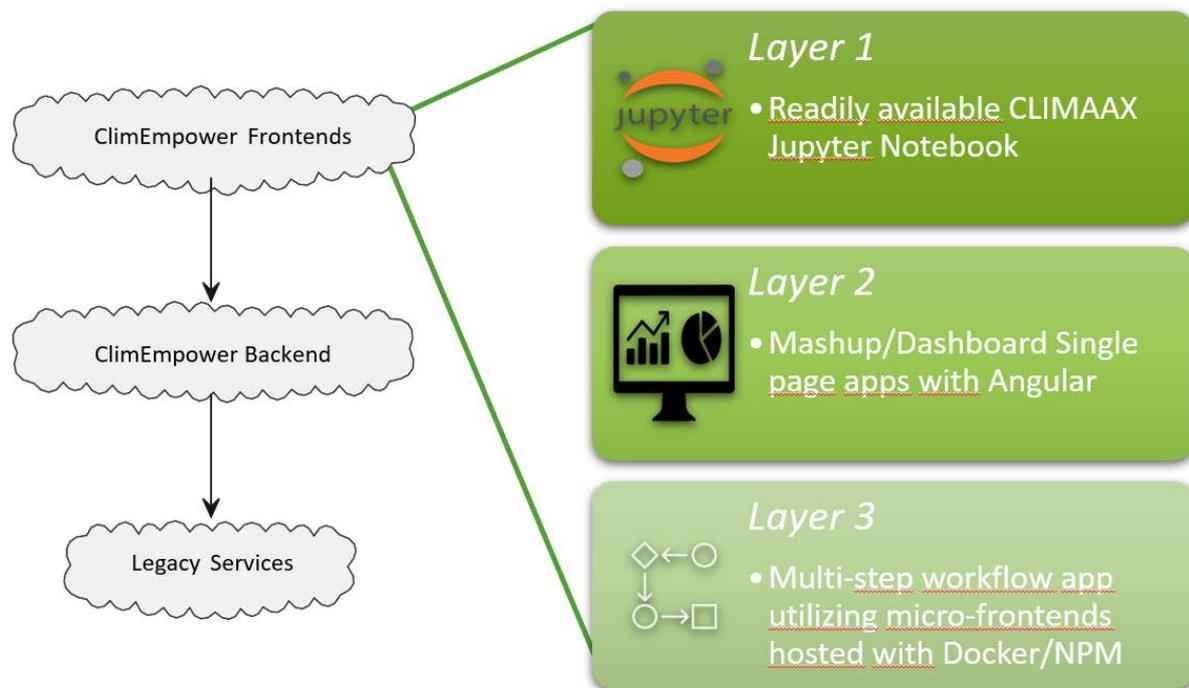

Figure 8: High-level ClimEmpower architecture focusing on the different frontend types.

Efficient communication and coordination between tasks T3.4 and T4.3 will be crucial for the success of this process. The development teams must maintain a strong connection with the regional stakeholders to ensure that the applications are developed according to their specific needs and use cases. This ongoing dialogue will also serve to familiarize stakeholders with the tools as they are being developed, reducing the need for extensive training once the applications are finalized.

6.2 Rapid service and application prototyping

Typical ClimEmpower applications will connect hazards, exposure, and vulnerability of specific elements at risk to indicate the related risks or impacts of future crisis events such as extreme heat or precipitation on these elements at risk, and ideally also indicate the impacts of potential adaptation measures. To develop such applications, the scientists, developers, and end-users working in the project will have to answer various questions such as:

- What do the users expect from the applications (WP1, WP4)

- What kind of indicators the users are interested in? (WP1, WP4)

- What input data can be used to calculate these indicators? (WP2)

- How is this data algorithmically connected to calculate the necessary indicators (WP2)

- Do the results make sense? Are they useful for decision making (WP2, WP4)

- How should the data and indicators be presented in applications (WP4)

- What kinds of data manipulation should the application allow users to do (WP4)?

The work package indicators in parentheses are a reminder that this process has already started in the scope of WP1 and continues in the scope of WP2 and WP4, with WP3 aiming to support these cross-WP efforts through provision of the rapidly prototyped "proof-of-concept" applications.

Jupyter Notebooks developed in the CLIMAAX project will play a crucial role in this process. CLIMAAX has already developed a set of sample notebooks demonstrating how to estimate risk and hazard using Copernicus data. Our idea is to use these notebooks as a starting point for developing the ClimEmpower applications, in the following way:

1) Choose the notebooks that already implement some of the impact assessment algorithms the ClimEmpower regions are interested in. For example, Figure 9 demonstrates one of the CLIMAAX notebooks that is of great interest to the Osijek-Baranja region.

2) In an interactive process with ClimEmpower trial regions, start experimenting with different ways to visualise the notebook data and discuss the ways users should be able to interact with this data and algorithms (WP4, task 4.3). This work shall also be supported through provision of mock-ups and early prototypes of the mashup/dashboard single page applications.

3) In parallel, replace the data access routines with those developed in T3.1 to simplify the notebooks and start experimenting with fine-tuning and extending the algorithms to better meet the regional expectations, and with replacing the Copernicus data with more accurate estimates for regional hazards, exposure and/or vulnerability.

4) Once the data, algorithms and visualisation needs have been agreed upon, encapsulate the notebooks code behind standard service interfaces (i.e. encapsulate the Jupyter notebooks code behind standard OGC geo-services (e.g. GeoServer or QGIS Server), or behind the data access API developed in T3.1 (feasibility of both approaches will be analyzed and then a decision will be made) and update the mashup/dashboard single-page applications to utilise these services as their data source.

5) Finally, assemble the workflow/decision-support services from individual single-page applications.

6) Where appropriate, provide added value for the users by linking the training materials (WP4, T4.1 "Provide educational materials") and/or calling some of the existing knowledge services such as the KNOWING Knowledge Base or ICARIA Portfolio of solutions) to provide additional context.
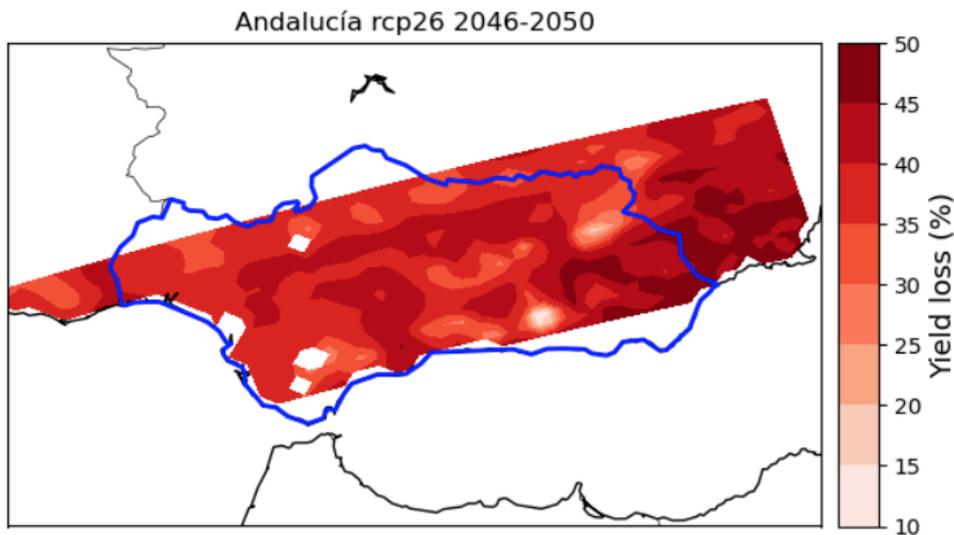
Figure 9: Example plot from CLIMAAX of the maize yield loss from precipitation deficit in Andalusia using data from the Copernicus Data Store.

Anticipated result of this process are ~TRL6 prototypic implementations of the applications that provide roughly the same functionality as the underlying Jupyter Notebooks, but in a form that is suitable for use by domain experts / CoP members with minimal IT knowledge.

6.3 Support services

Throughout the project a set of support services will be setup and maintained, with the aim to both enable and enforce a structured workflow for tool development and deployment, as well as their continuous maintenance and improvement for the duration of the project. A private GitLab repository provided by AIT (see Figure 10) will serve as the primary code repository for the tools developed by the project partners until a final decision regarding licensing and IPR condition has been made. Depending on exploitation plans, this code can be subsequently moved to a public GitHub repository.
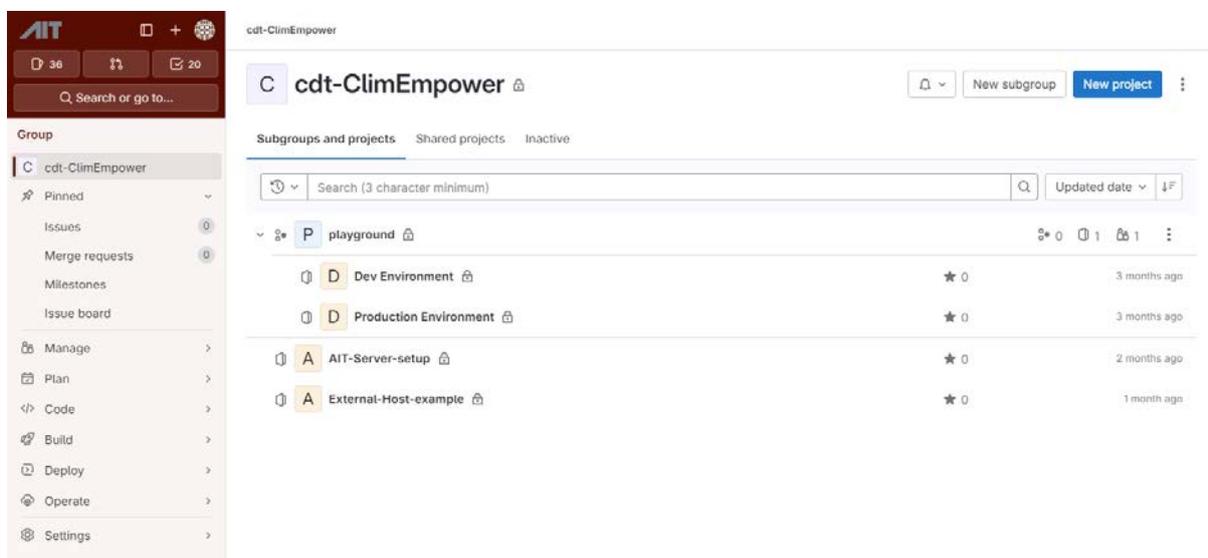

Figure 10: The private git repository for tool development provided by AIT

For the initial development phase of the individual components and applications, AIT has provided a virtual machine on its premises, where these components can be deployed and tested. For ease of deployment and managing the individual components, Docker containers shall be used whenever possible. For these containers additional services can be offered on demand, including configuration of API endpoints, management of deployed Docker containers and identity management for single-sign-on (SSO). For more details see Table 3 for reference.

Table 3: Overview of support services

Service/tool name	Purpose	Further information
Traefik	Provides API endpoints for deployed services, enabling them to be reached by other tools/components	https://doc.traefik.io/traefik/
Portainer	This service allows developers to manage and control deployed Docker containers	https://docs.portainer.io/
Keycloak	Enables identity and access management for authentication and single-sign-on capabilities	https://www.keycloak.org/documentation
GitLab	Serves as the private repository for storing, versioning and sharing of the developed code	https://about.gitlab.com/get-started/
Docker	This tool allows developers to build and deploy their applications independently from the used IT hardware through virtualization.	https://docs.docker.com/

To integrate existing legacy services used by the regions, additional backend services may be required to function as connectors between these legacy systems and the components and services provided by WP3. In some cases, the data services described in section 5.1 may facilitate these connections, but specific details regarding the legacy services operated by the regions, and their potential inclusion in the ClimEmpower project, will need to be discussed directly with the regional stakeholders.

6.4 Deployment Strategies

Applications and components will initially be deployed locally by each partner during the development phase and later tested in the test environment provided by AIT. This environment will ensure proper integration of the individual components used in the various applications developed.

During the second plenary meeting, it was agreed that the regions will eventually host the tools and services themselves. However, the envisioned architecture supports multiple deployment scenarios, allowing for a hybrid model. For example, web-based frontend applications could be hosted on-site within the regions, while backend services, interfaces, and databases may remain hosted centrally at one of the technical partners' facilities. For example, for the data access backend service provided by ECMWF, it was decided that it will be deployed and hosted by the project partner itself.

Although various deployment strategies are feasible, a definitive decision on the preferred approach should be made within the next six months. This timeline will provide clarity for the technical partners and, if necessary, give them sufficient time to prepare the appropriate technical infrastructure. Therefore, upcoming discussions with the regions will also focus on finalizing the deployment strategy to ensure smooth implementation across all stakeholders.

7 Conclusions

In conclusion, this deliverable outlines a comprehensive and adaptable system architecture for the ClimEmpower project, aimed at addressing the diverse climate challenges faced by different regions. By utilizing a flexible and modular design, WP3 ensures that the tools and services are tailored to the specific needs of each region while also remaining adaptable for potential follower regions. The emphasis on user-centric co-design and iterative development guarantees continuous refinement and alignment with stakeholder needs, ensuring the applications are both relevant and effective in supporting climate adaptation efforts.

While micro-frontends are envisioned as a core strategy for achieving flexibility in the GUI applications, the exact implementation – whether through web components, Docker containers, or NPM-managed libraries – has not yet been finalized. Each option offers unique benefits and drawbacks, and with the current understanding of the application requirements provided by the end-users, it is premature to commit to one approach. Therefore, upcoming discussions with end-users will play a critical role in further refining these requirements and guiding the decision on the most appropriate technology.

Moving forward, WP3 will focus on developing a prototype using one or more Jupyter Notebooks provided by CLIMAAX, adapting them to regional use cases, and deploying them in the test environment set up by AIT. These initial prototypes will help engage the regions in further discussions and serve as a steppingstone to define additional functionality, leading to the development of dashboard GUIs as the next phase. Ultimately, these dashboards will lead to the multi-step, workflow-like decision support tools, representing the final stage of the application development process. Additionally, the utilization and integration of tools and services provided by other projects, such as the Knowledge Base in KNOWING and the Portfolio of Solution in ICARIA will be assessed and where appropriate also implemented.

8 References

IPCC. (2018). Annex I: Glossary. In *Global Warming of 1.5°C* (S. pp. 541–562). Cambridge University Press. doi:https://doi.org/10.1017/9781009157940.008