



D1.5 Final industrialization and support report

WP1 – CO-Creation

Deliverable Lead: ATOS

Dissemination Level: Public

Deliverable due date: 31/05/2020

Actual submission date: 07/09/2020

Updated on 14.09.2020

Version 1.1



Document Control Page	
Title	D1.5 Industrialization and support
Creator	Mario Nunez (ATOS), Daniel Roderer (ATOS)
Description	<p>Report on the achievements of the T1.4 work in the last six months of the project. This report will include the summary of the bug and issue reports that addressed in the final project months and of the known bugs and issues that are left open at the project end.</p> <p>This report is an updated version of the original D1.5 document. Main differences are: (1) extended and improved section 2.2.4.3 and newly added section 2.2.4.5. In addition, minor updates have been made to fix some English errors.</p>
Publisher	CLARITY Consortium
Contributors	Mario Nunez (ATOS), Miguel Angel Esbri (ATOS), Daniel Roderer (ATOS), Pascal Dihé (CISMET), Denis Havlik (AIT)
Creation date	16/11/2019
Type	Text
Language	en-GB
Rights	Copyright "CLARITY Consortium"
Audience	<input checked="" type="checkbox"/> Public <input type="checkbox"/> Confidential <input type="checkbox"/> Classified
Status	<input type="checkbox"/> In Progress <input type="checkbox"/> For Review <input type="checkbox"/> For Approval <input checked="" type="checkbox"/> Approved

Disclaimer

The text, figures and tables in this report can be reused under a provision of the Creative Commons Attribution 4.0 International License. Logos and other trademarks are not covered by this license.

The content of the publication herein is the sole responsibility of the publishers and it does not necessarily represent the views expressed by the European Commission or its services.

While the information contained in the documents is believed to be accurate, the authors(s) or any other participant in the CLARITY consortium make no warranty of any kind with regard to this material including, but not limited to the implied warranties of merchantability and fitness for a particular purpose.

Neither the CLARITY Consortium nor any of its members, their officers, employees or agents shall be responsible or liable in negligence or otherwise howsoever in respect of any inaccuracy or omission herein.

Without derogating from the generality of the foregoing neither the CLARITY Consortium nor any of its members, their officers, employees or agents shall be liable for any direct or indirect or consequential loss or damage caused by or arising from any information advice or inaccuracy or omission herein.

Figure 1: CLARITY Disclaimer

Table of Contents

CLARITY Project Overview	6
Abbreviations and Glossary.....	7
Executive Summary	8
1 Introduction	9
1.1 Purpose of this document	9
1.2 Intended audience.....	9
1.3 Document structure	9
2 Development and production environment	10
2.1 Initial objective	10
2.2 Development and integration environment configuration	10
2.2.1 CSIS Development System	10
2.2.2 Service Monitoring.....	12
2.2.3 CSIS Production System	20
2.2.4 Other services	21
2.2.5 Synchronisation between the Development and Production environments	28
3 CSIS Technical Validation	30
3.1 Unit tests	30
3.1.1 Jenkins Cl.....	31
3.2 Integration Tests.....	34
3.3 User Acceptance Tests.....	34
4 Software licensing	35
4.1 Proposed licensing header for source code files	36
5 Reporting on the technical work performed in the last months concerning CSIS and Marketplace.....	37
5.1 New features implemented.....	37
5.1.1 Adaptation options in the CSIS workflow	37
5.1.2 Local effects calculation for pluvial floods and new impact indices	38
5.1.3 Economic impact of floods on roads, residential and non-residential buildings - Updates in the Data Packages	38
5.2 CSIS Management Project	39
5.2.1 Known bugs and issues closed.....	39
5.2.2 Known bugs and issues left open.....	41
6 Conclusions	42
7 References.....	43
8 Annexes.....	44
8.1 Multi Criteria Decision Analysis Tool	44
8.1.1 Requested functionality.....	44
8.1.2 Exploitation Requirements assessment.....	44
8.1.3 Technology support	45

List of Figures

Figure 1: CSIS Architecture diagram.....	11
Figure 2: Monitoring table.....	13
Figure 3: Scenario Analysis component.....	14
Figure 4: Indicator bar chart diagrams.....	15
Figure 5: Map component.....	16
Figure 6: Basic map.....	17
Figure 7:Synchronised Maps.....	18
Figure 8: Study Area Map.....	19
Figure 9: CSIS production environment architecture.....	20
Figure 10: Tests performed.....	30
Figure 11: New pipeline creation.....	31

Figure 12: Pipeline configuration	32
Figure 13: GitHub webhook configuration.....	33
Figure 14: Integration test report.....	34
Figure 15: Overview of T1.4 issues and bugs	37
Figure 16: Multi Criteria Decision Analysis Tool Technology Support.....	45

List of Tables

Table 1: CLARITY abbreviations.	7
Table 2: Known bugs and issues closed.....	40
Table 3: Known bugs and issues left open.	41

CLARITY Project Overview

Urban areas and traffic infrastructure linking such areas are highly vulnerable to climate change. Smart use of existing climate intelligence can increase urban resilience and generate added value for businesses and society at large. Based on the results of FP7 climate change, future internet and crisis preparedness projects (SUDPLAN, ENVIROFI, CRISMA) with an average TRL of 4-5 and following an agile and user-centred design process, end-users, purveyors and providers of climate intelligence will co-create an integrated Climate Services Information System (CSIS) to integrate resilience into urban infrastructure.

As a result, CLARITY will provide an operational eco-system of cloud-based climate services to calculate and present the expected effects of CC-induced and -amplified hazards at the level of risk, vulnerability and impact functions. CLARITY will offer what-If decision support functions to investigate the effects of adaptation measures and risk reduction options in the specific project context and allow the comparison of alternative strategies. Four demonstration cases will showcase CLARITY climate services in different climatic, regional, infrastructure and hazard contexts in Italy, Sweden, Austria and Spain; focusing on the planning and implementation of urban infrastructure development projects.

CLARITY will provide the practical means to include the effects of CC hazards and possible adaptation and risk management strategies into planning and implementation of such projects, focusing on increasing CC resilience. Decision makers involved in these projects will be empowered to perform climate proof and adaptive planning of adaptation and risk reduction options.

Abbreviations and Glossary

A complete glossary of all CLARITY terms and abbreviations can be found in the public document “CLARITY Glossary” available at <https://cat.clarity-h2020.eu/glossary/main>.

Table 1: CLARITY abbreviations.

Abbreviation/ Acronym	DEFINITION
CA	Consortium Agreement
CRM	Continuous Risk Management
DC	Demonstration Case
DPA	Data Protection Agency
DoA	Description of the Actions (Annex 1 to the Grant Agreement)
EC	European Commission
EM	Exploitation Manager
GA	General Assembly
IPR	Intellectual Property Rights
QA	Quality Assurance
QAP	Quality Assurance Plan
SQA	Software Quality Assurance
SQAP	Software Quality Assurance Plan
TM	Scientific & Technical Manager
TL	Task Leader
TOC	Table of Content
WP	Work Package
WPL	Work Package Leader
CSIS	Climate Services Information System
CID	Continuous Integration and Delivery
API	Application Programming Interface
OS	Operative System
VMM	Virtual Machine Monitor
VM	Virtual Machine
YAML	Ain't Mark-up Language
IT	Information Technologies
EGI	European Grid Infrastructure

Executive Summary

The objective of WP1 is to involve practitioners, suppliers, purveyors and technology providers, scientists and potential end users (customers) in the climate service co-creation and deliver the CLARITY CSIS software and workflows in support of the climate-resilience planning.

Task “T1.4 Industrialization and Support” is responsible for providing the necessary repositories for hosting of the CLARITY software code, data and services as well as a set of tools to support the i) development of industrial-quality code (including issues and bug tracking) and ii) the deployment of the resulting components and services in the testing and production environments from which the final solution will be accessible by testers and final end-users respectively. In the last six months of the project, this task has been the only active development task responsible for assuring that the remaining development efforts are spent on issue and bug fixing.

This report presents the achievements of the task T1.4 work in terms of:

- Development infrastructure setup
- Testing and production environments, including all the resulting services and components deployed that comprehend final CLARITY solution
- A summary of the bug and issue reports that were addressed in the final project months and of the known bugs and issues that are left open at the project end.

1 Introduction

The introduction chapter defines the purpose and intended audience of deliverable “D1.5 Final industrialization and support report” and briefly explains its structure.

1.1 Purpose of this document

Task “T1.4 Industrialization and Support” is responsible for providing the necessary repositories for hosting of the CLARITY software code, data and services as well as a set of tools to support the i) development of industrial-quality code (including issues and bug tracking) and ii) the deployment of the resulting components and services in the testing and production environments from which the final solution will be accessible by testers and final end-users respectively. In the last six months of the project, this task has been the only active development task responsible for assuring that the remaining development efforts are spent on issue and bug fixing. This report presents the achievements of the task T1.4 work in terms of:

- Development infrastructure setup
- Testing and production environments, including all the resulting services and components deployed that comprehend final CLARITY solution
- A summary of the bug and issue reports that were addressed in the final project months and of the known bugs and issues that are left open at the project end.

1.2 Intended audience

The target readers of this document are mainly the CLARITY technical partners in charge of the Climate Service integration and development in WP1 “Co-Creation”, WP2 “Demonstration & Validation”, WP3 “Science Support” and WP4 “Technology Support”. In addition, this report is of interest for business developers and potential users of the CLARITY technical results.

1.3 Document structure

The structure of the document and the relationships between the different chapters is as follows:

Chapter 1 (this chapter) introduces the document and explains the overall purpose of this document and its relation to other work packages and deliverables.

Chapter 2 presents the Development and Production environment setups, including the services and components deployed on them.

Chapter 3 describes how the technical validation of the CSIS prototype is performed, i.e., unitary, integration and user acceptance tests.

Chapter 4 presents the partner(s) responsible for the each of components and their licenses.

Chapter 5 explains the technical work carried out during the six last months (from M33 until M39) of the project concerning the CSIS and the Marketplace, with focus on the new features added or existing ones that were improved.

Chapter 6 provides the conclusions.

Chapter 7 lists the references and bibliography used in this document.

Chapter 8 provides an annex providing further details with regards to the Multi-criteria Decision Analysis Tool included in the CSIS prototype.

2 Development and production environment

2.1 Initial objective

The purpose of this task is to provide a continuous delivery platform allowing every consortium partner to rely on a software engineering practice already consolidated.

Continuous Integration and Delivery (CID) can prevent and reduce errors by analysing health of the source code, since code analysis and tests provided by a CID environment can reveal bugs in a more effective way ensuring a higher quality final product. This feature gives more confidence to developer team on providing new functionalities and makes development much more comfortable, it even provides tools to generate API-like documentation for all committed code, making much easier to improve and reuse source code or software components.

Also, repetitive manual processes are slow and prone to human errors and CID is able to automatize software integration and delivery. So that CID helps to avoid such practices and saves a considerable amount of effort along the way.

One more relevant aspect of CID allows releasing deployable software at any point in time, so CID avoid delays on delivery of stable versions of the product. Indeed, CID can mitigate risks by deploying directly on production environment so that prevent typical problems about rising errors when moving from test to production environment.

In conclusion being considered a good practice can be the backbone of a healthy development environment for CLARITY CSIS.

2.2 Development and integration environment configuration

Management Project for coordinating the co-creation process of the CLARITY Climate Services Information System (CSIS). High-level tasks (issues) are defined in this repository and then broken down into concrete implementable issues in different sub repositories and project boards.

The CLARTIY CSIS as such consists of a set of docker-containers and is deployed as production and development of two virtual serves managed by Austrian Institute Of Technology¹.

2.2.1 CSIS Development System

The CSIS Development System is deployed at virtual server `csis-dev.ait.ac.at` and is reachable under the URL csis-dev.myclimateservice.eu. Each subfolder contains a `docker-compose.yml` file to control the respective service groups (Figure 1). For each service groups, a respective GitHub repository exists in organisation `clarity-h2020`. It contains a `readme.md` file that provides more information regarding implementation, configuration, backups, upgrading, etc.

¹ <https://www.ait.ac.at/>

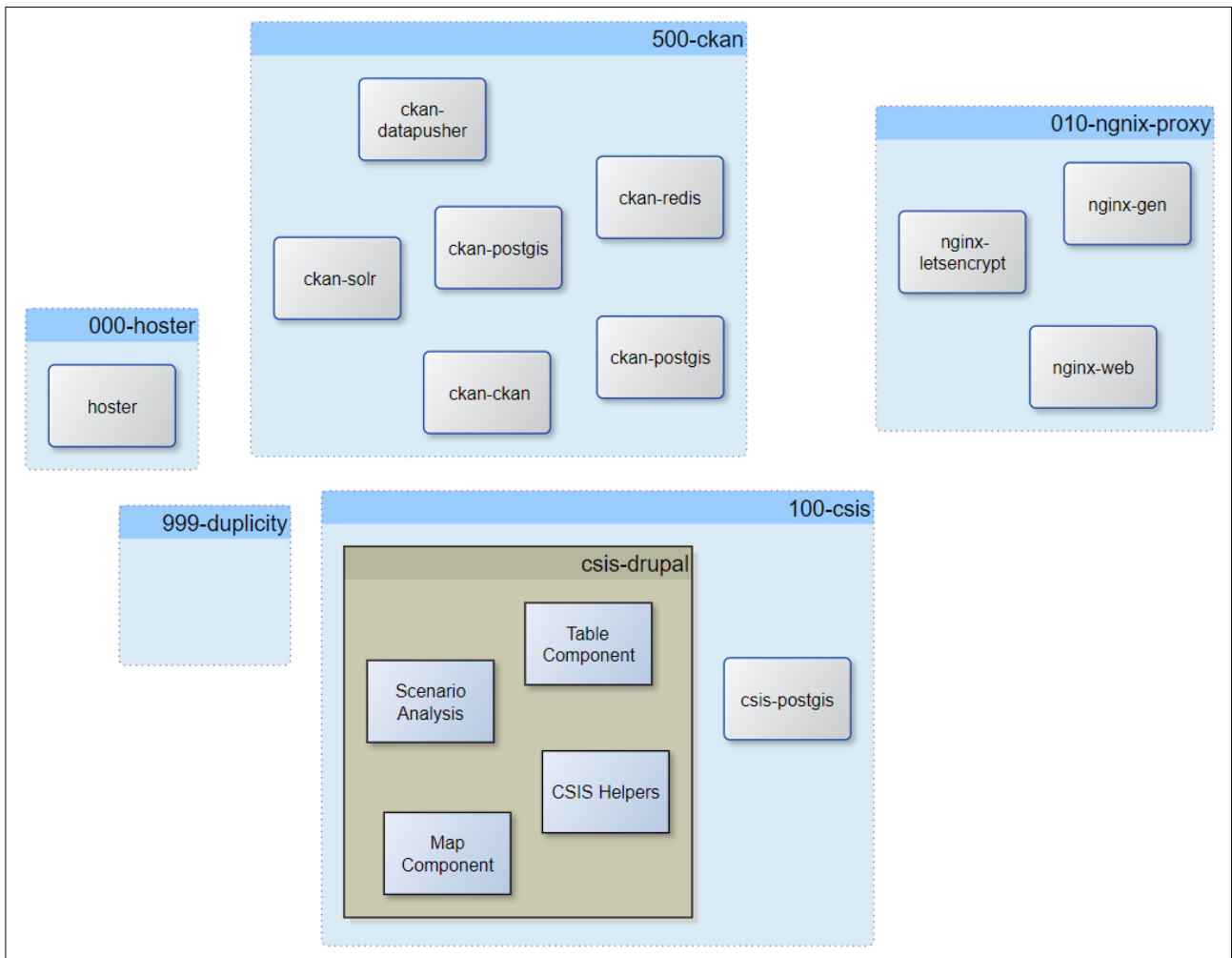


Figure 1: CSIS Architecture diagram.

2.2.1.1 Hoster

Hoster² is a file injection tool for resolving names of local Docker containers on the host. Instead of exposing ports of a container on the docker host (Localhost), can be accessed by its name directly.

Hoster is intended to run in a Docker container. The `docker.sock` is mounted to allow Hoster to listen for Docker events and automatically register containers IP. Hoster inserts into the host's `/etc/hosts` file an entry per running container and keeps the file updated with any started/stopped container.

2.2.1.2 Nginx-proxy

The nginx reverse proxy is used to expose several internal docker services of CLARITY's Climate Services Information System (CSIS) on `myclimateservice.eu` subdomains.

- **deployed at:** `/docker/010-nginx-proxy`
- **repository:** [docker-compose-letsencrypt-nginx-proxy-companion](https://github.com/clarity-h2020/docker-nginx-proxy-companion)
- **public endpoint:** csis-dev.myclimateservice.eu

² <https://github.com/clarity-h2020/docker-hoster/>

2.2.1.3 CKAN

CKAN³ is the world's leading open-source data portal platform. CKAN makes it easy to publish, share and work with data. It's a data management system that provides a powerful platform for cataloging, storing and accessing datasets with a rich front-end, full API (for both data and catalog), visualization tools and more.

CLARITY's online meta-data catalogue available at ckan.myclimateservice.eu reports on all datasets produced and used within the CLARITY H2020 project. It has been used as a "living" Data Management Plan during the project. Datasets are described by meta-data compliant to the DataCite⁴ metadata schema with CLARITY-specific extensions. The actual data is not stored in the meta-data catalogue directly but in the repositories of the original data provider (data used) and in private (non-open data produced) or public (open data produced) repositories selected or maintained by CLARITY partners. Thereby, the default repository for depositing open data produced by CLARITY is the Zenodo⁵ research data repository. The meta-data in the catalogue contains either the public URL for accessing the dataset and/or the contact details of the responsible data provider for requesting access to (non-public) data.

2.2.1.4 Duplicity

Duplicity is backup tool that is used by CLARITY's Climate Services Information System (CSIS) on CSIS Development System and CSIS Production System virtual servers to perform incremental backups.

This is a fork of wernight/docker-duplicity⁶ (which we call *upstream* in the following). We created this repo when *Duplicity* due to Python 2 deprecation moved to Python 3 but *upstream* did not follow yet.

2.2.2 Service Monitoring

The following CSIS Services are monitored with help of statping⁷, a status page and monitoring server for websites, applications and web services (Figure 10):

- CSIS Production environment
- CSIS Development environment
- EMIKAT REST API
- EMIKAT GeoServer
- ATOS GeoServer
- CLARITY CKAN

³ <http://ckan.org/>

⁴ <https://datacite.org/>

⁵ <https://zenodo.org/communities/clarity>

⁶ <https://github.com/wernight/docker-duplicity.git>

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

CSIS PROD	
CSIS (PROD) - Study 1 Summary	ONLINE
CSIS (Prod) - Json API	ONLINE
CSIS (PROD) - Solution Offers	ONLINE
CSIS (PROD) - Adaptation Options	ONLINE
CSIS DEV	
CSIS (Dev) - Json API	ONLINE
CSIS (Dev) - Summary Study 1	ONLINE
CSIS (DEV) - Adaptation Options	ONLINE

Figure 2: Monitoring table.

The statping monitoring service is deployed as docker container on a dedicated machine. It is itself again monitored by a cypress.io-based⁸ test specification that is executed on Jenkins CI. Developers are notified by email and a new issue is posted in repository csis-technical-validation when one of the monitored services fails.

2.2.2.1.1 CSIS

Currently, the following custom modules and integrated apps (Building Blocks⁹) are deployed together with the Drupal 8 system:

2.2.2.1.2 Scenario Analysis Component

This is the AngularJS (version 1.x)¹⁰ implementation of the Scenario Comparison and Analysis and the Multi-Criteria-Analysis and Decision Support Functional Building Block based on CRISMA Worldstate Analysis Widgets¹¹.

The Multi Criteria Decision Analysis Tool supports the analysis and comparison of (adaptation) scenarios regarding performance indicators that can be defined by the end user and thus leverages what-if decision support to investigate the effects of adaptation measures and risk reduction options in the specific project context, and allows the comparison of alternative strategies (Figure 4). Thereby the tool provides multi-criteria ranking functions to compare and rank different scenarios and corresponding adaptation plans according to different criteria and their relative weight and level of importance. More information can be found at Annex 8.1 (Multi Criteria Decision Analysis Tool).

⁸ <https://www.cypress.io/>

⁹ <https://github.com/clarity-h2020/csis/blob/master>

¹⁰ <https://angularjs.org/>

¹¹ <https://github.com/crismaproject/worldstate-analysis-widgets>

Indicator table

Indicators	BASELINE: BASELINE (1971-2000)	BASELINE: RCP85 (2071-2100)	ADAPTATION-01: BASELINE (1971-2000)	ADAPTATION-01: RCP85 (2071-2100)
Impact following Heat Wave Events				
Discomfort Level (Frequent)	4	5	4	5
Discomfort Level (Occasional)	5	5	5	5
Discomfort Level (Rare)	5	5	5	5
Heat Wave Impact (Frequent)	17	21	17	21
Heat Wave Impact (Occasional)	20	21	20	21
Heat Wave Impact (Rare)	20	21	20	21
Increase in Mortality Rate (Frequent)	0.262 ‰	0.856 ‰	0.262 ‰	0.856 ‰
Increase in Mortality Rate (Occasional)	0.460 ‰	1.468 ‰	0.460 ‰	1.468 ‰
Increase in Mortality Rate (Rare)	0.571 ‰	1.591 ‰	0.571 ‰	1.591 ‰
Impact following Pluvial Flood Events				
Damage Class (Frequent)	4	4	4	4
Damage Class (Occasional)	4	4	4	4
Damage Class (Rare)	4	4	4	4
Damage Probability (Frequent)	54	54	54	54
Damage Probability (Occasional)	54	54	54	54
Damage Probability (Rare)	69	69	69	69
Impact (Frequent)	43,389,722.116 €	43,389,722.116 €	43,389,722.116 €	43,389,722.116 €
Impact (Occasional)	43,389,722.116 €	43,389,722.116 €	43,389,722.116 €	43,389,722.116 €
Impact (Rare)	43,389,722.116 €	43,389,722.116 €	43,389,722.116 €	43,389,722.116 €

Figure 3: Scenario Analysis component

2.2.2.1.3 User Interface

Several UI components, in particular criteria and ranking diagrams (Figure 5), that are not needed or are not working due to the absence of *Criteria Functions* and *Decision Strategies* had to be disabled so that they do not interfere with report generation. Currently the only diagram that is enabled is the Indicator Bar Chart Diagram:



Figure 4: Indicator bar chart diagrams.

2.2.2.2 CSIS Helpers Drupal Module

This module is a collection of functions that allow other components to interact with the CSIS and expand its features. As the name suggests this module is specifically designed for the usage in the CSIS and its highly specialized *Study group*. Key features are:

- Generating variables which makes certain information about nodes and groups available as JSON.
- Taking screenshots of map and table components.
- Updating GL-step relations (needed for including/removing Twins).
- Notifying and triggering Emikat about new Studies and receiving the Emikat identifier for each individual Study.
- Notifying the TM about new Studies relevant for Transport Infrastructure.

This module can be installed as any other Drupal module via the "Extend" tab in the Backend. Installation via Composer is not supported. This module has to be placed in the appropriate modules folder in order to be found by Drupal. Initial configuration after the install is necessary in order for the module to be able to communicate with external services.

2.2.2.3 Map component

The Map Component is a reusable, flexible and highly configurable Building Block meant to be used throughout CSIS. It is an embeddable component that can be easily adapted to different parts of the common CSIS UI (Figure 6). The core functionality of this component is visualization of different maps and layers following to the EU-GL process model¹².

¹² <https://myclimateservices.eu/en/about>

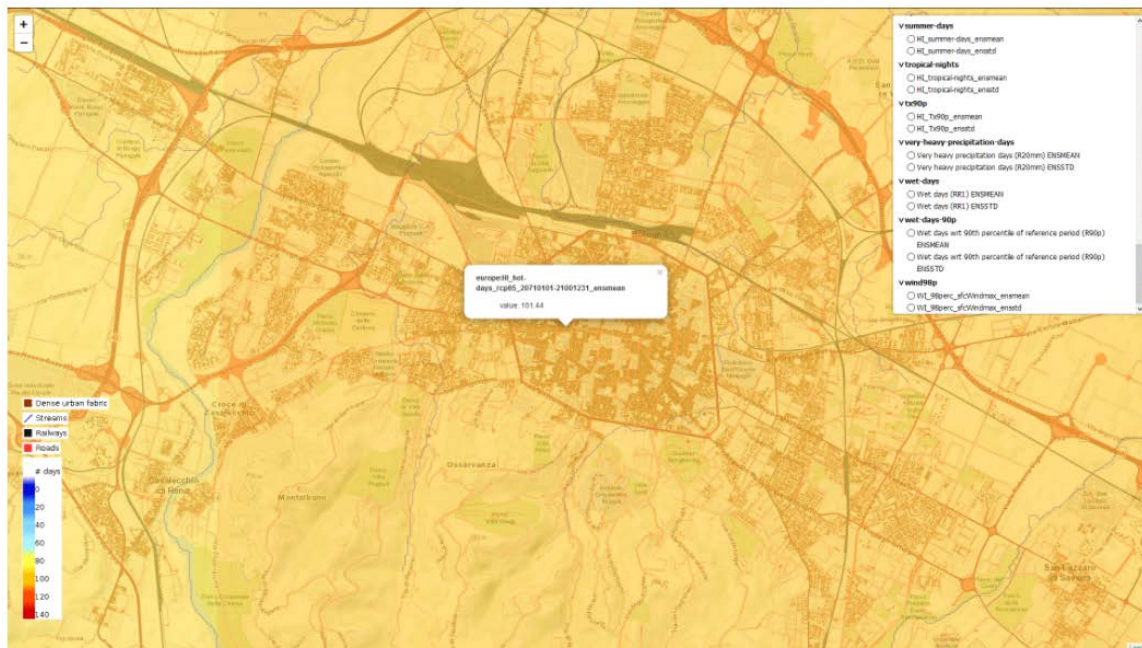


Figure 5: Map component.

2.2.2.3.1 Implementation

Specialised maps for EU-GL Steps EE, HC, HC-LE, RA/IA and AAOP are implemented as separate components. The main components are:

2.2.2.3.1.1 Basic Map

This is the basic class of all specialized map classes. It implements the common way to extract the layers from the CSIS Study. In fact, it is a wrapper component for a component that controls an actual mapping component implementation (e.g a react-leaflet map). In principle it transforms the CSIS Drupal JSON format into a simple internal layer definition format (Figure 7) that is understood by a JSX Component that renders map.

The component processes the query parameters and, with help of `csis-helpers-js`¹³, prepares *Background layers*, *Overlay Layers* and respective *Layer Groups* based on Data Package Resources either from a single resource (query parameter `resource_uuid`) or from a set of resources contained in a Data Package associated with a Study and retrieved from the CSIS Drupal AP.

Thereby, *Template Resources*¹⁴ are expanded into several layers according to the scenario definition (time_period, emissions_scenarios and event_frequency) provided via query parameters. Only those resources are considered that are associated with a specific Taxonomy term, in general a term from the EU-GL Taxonomy like Hazard Characterization. The taxonomy as well as the specific term are either specified via query parameters.

The categorization of layers into different *Layer Groups* (Backgrounds, CLARITY Backgrounds, etc.), which is a feature of `react-leaflet-grouped-layer-control`, is also controlled via taxonomy terms.

¹³ This is a *JavaScript helpers'* module that supports external AJAX Apps like the Map Component or the Table Component that are embedded as *External iFrame* into the CLARITY CSIS Drupal Site to communicate with the CSIS Drupal API and the AIT EMIKAT API.

¹⁴ *Template Resources* are Data Package Resources that contain one or more references (service paths) with parametrizable URLs. This means, URLs of such resources contain variables that are replaced by actual values that are either taken from a *selected scenario* in CSIS or from a *permutation* of all possible values that have been added as tags to the resource. Those two usages of *Template Resources* are called *scenario-based resource parametrization* and *parameter-based resource expansion*.

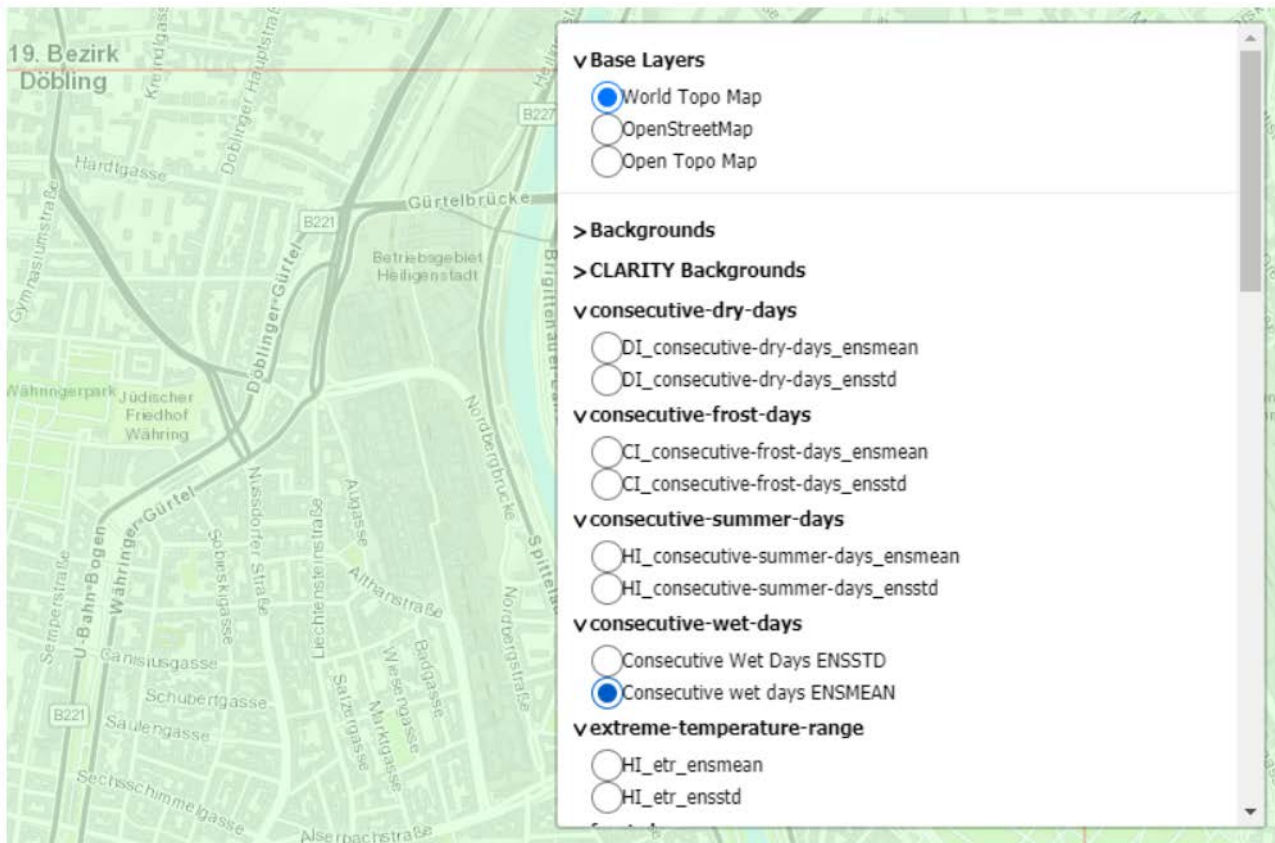


Figure 6: Basic map.

2.2.2.3.1.2 Leaflet Map

A common map component based on ReactLeaflet¹⁵ v2.7 that understand the JSON layer definition format created by a Basic Map. It is mainly responsible for creating a WMS tile layer and WMS layer components from the JSON objects received via props overlays and *exclusive groups*. "Exclusive Groups" are a feature of *react-leaflet-grouped-layer-control*¹⁶.

2.2.2.3.1.3 Generic Map (Synchronised Maps)

Generic Map extends Basic Map and is actual *base class* of all specialised map classes (Figure 8). Unless the Simple Table Component (see section 2.3.1.8), the Map Component does not use functional components but classes. In addition to Basic Map, it adds the possibility to show two synchronised components side-by-side with help of Leaflet.Sync¹⁷:

¹⁵ <https://react-leaflet.js.org/>

¹⁶ <https://github.com/clarity-h2020/react-leaflet-grouped-layer-control>

¹⁷ <https://github.com/jieter/Leaflet.Sync>

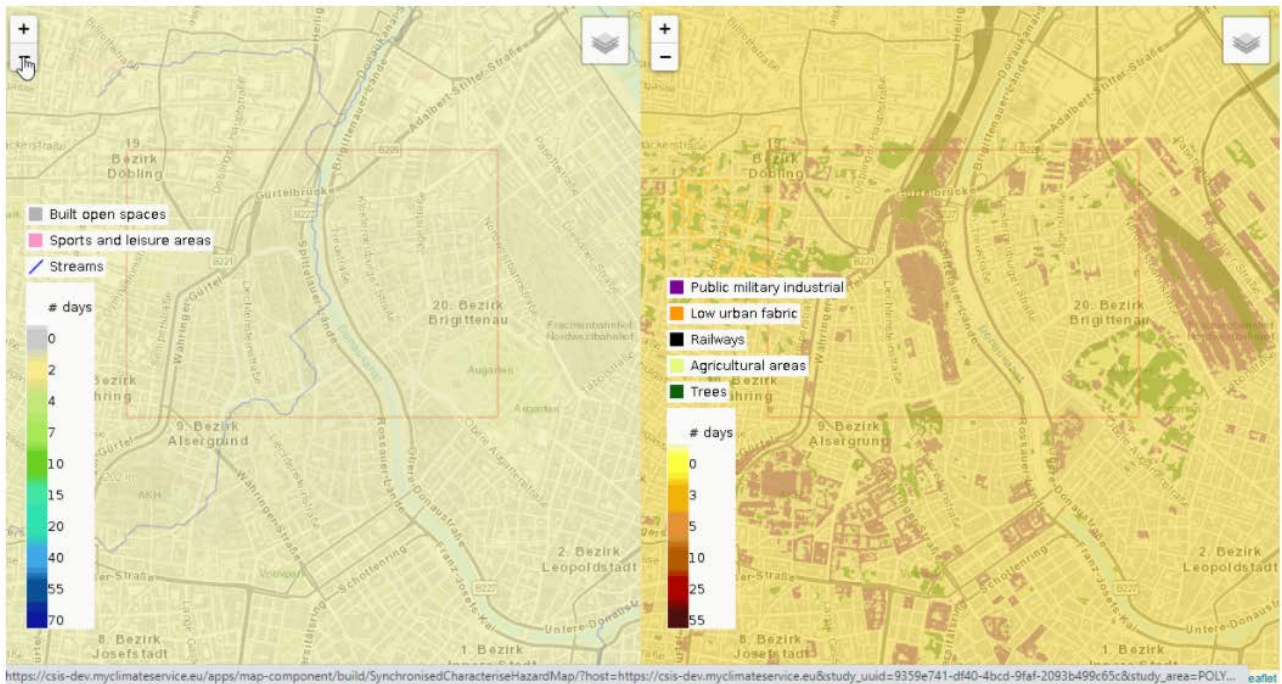


Figure 7: Synchronised Maps.

2.2.2.3.1.4 Specialized Maps

For convenience, specialized Maps currently offer just EU-GL-specific pre-sets of the query parameters and props, respectively, accepted by Basic Map. The following specialized maps are currently available:

- Characterise Hazard Map
- Exposure Map
- Hazard-Local Effects Map
- Risk and Impact Map

2.2.2.3.1.5 Study Area Map

Study Area Map is used by Drupal to show and edit the study area (Figure 9). As the functionality is quite different from the other maps and since it requires more interaction with the Drupal API, it is not derived from Basic Map. Since it is integrated into CSIS Drupal with help of React Mount Node¹⁸.

¹⁸ https://www.drupal.org/project/reactjs_mount

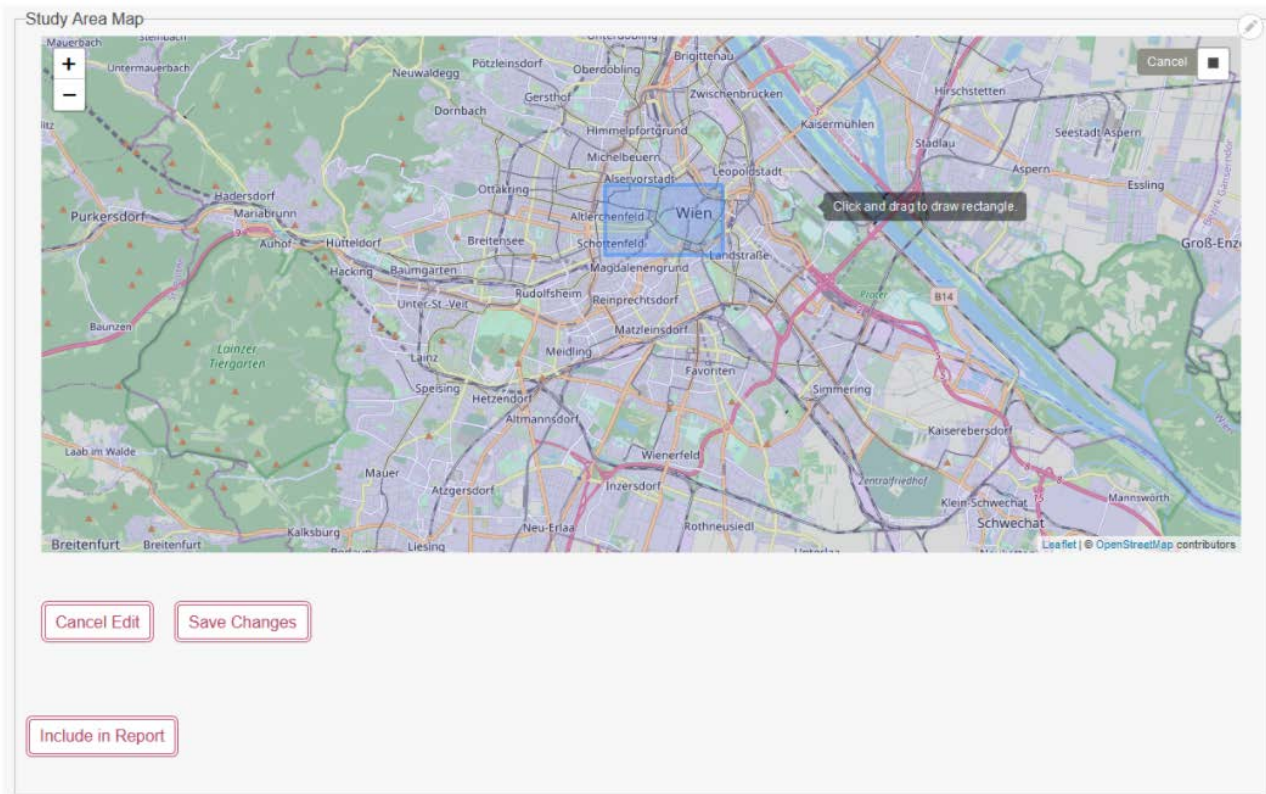


Figure 8: Study Area Map.

2.2.2.4 Simple table component

The *Simple Table Component* visualises tabular data directly received from EMIKAT API. It visualises the data “as-is”, which means, it does not perform any aggregation in contrast to the now deprecated *Table Component* which used to access the now deprecated *table-state-rest-api*. Specialised tables for EU-GL Steps EE, HC, HC-LE, RA/IA and AAOP are implemented as separate components. The main components are:

- **Generic Emikat client:** A Generic EMIKAT Client that communicates with the EMIKAT API and accepts as *render prop* an arbitrary JSX Component that is able to understand the EMIKAT Data Format and visualises the data retrieved by the client component.
- **Download button:** The Download Button is a component similar to the Generic Emikat Client, except that it provides an UI for selecting the format of the data retrieved from EMIKAT API and forces the browser to download the data instead of forwarding it to a separate render component.
- **Generic Emikat table:** A Generic Table Component based on *ReactTable*¹⁹ v6.0 that understand the JSON format of the EMIKAT API. It uses *csis-helpers-js* to translate the proprietary EMIKAT JSON format into a tabular format understood by *ReactTable*. It is the 'base class' of more specific tables like the Exposure Table. Specific tables can provide their own functions as props that control which columns are shown and how the data is formatted.

¹⁹ <https://www.npmjs.com/package/react-table>

- **Parameter selection component:** The Parameter Selection Component can wrap an Emikat Client Component and a Download Button Component as *render props* and add a user interface for selection parameters related to a *scenario*. The Parameter Selection Component thereby changes the respective Emikat Parameters props of the wrapped components. These props are used to parameterise the API template URL. More information on EMIKAT API parameters can be found in EMIKATHelpers.js²⁰.
- **Specialised tables:** tables that implement some custom formatting of row values and custom selection of columns (not all of them are interesting for end users) are:
 - Characterise hazard table
 - Hazard Local Effects table
 - Exposure table
 - Risk and Impact table
 - Adaptation Options Appraisal table

2.2.3 CSIS Production System

The CSIS Production System is deployed at virtual server csis.ait.ac.at and is reachable under the URL csis.myclimateservice.eu. Synchronisation between deployment and production system is described in Synchronisation between the Development and Production environments (see section 2.3.4). The architecture is described in the following diagram:

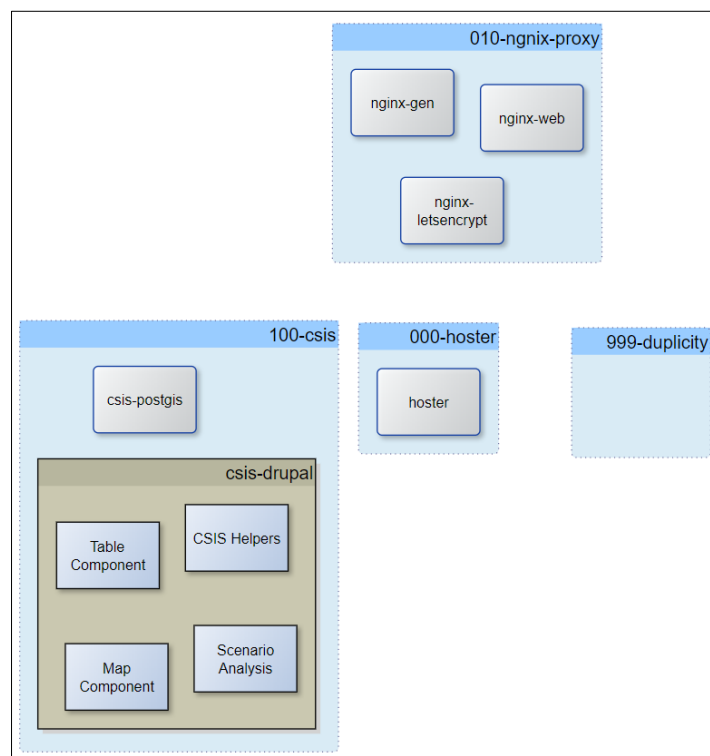


Figure 9: CSIS production environment architecture.

²⁰ <https://github.com/clarity-h2020/csis-helpers-js/blob/dev/src/lib/EMIKATHelpers.js>

For most service groups that are deployed at the production virtual server, separate development and production branches exist in the respective GitHub repository.

2.2.3.1 Hoster

For more information about the Hoster see section 2.2.1.1.

2.2.3.2 Ngnix-proxy

For more information about the proxy see section 2.2.1.2.

The **public endpoint** is csis.myclimateservice.eu. The production configuration is maintained in branch csis.ait.ac.at.

2.2.3.3 CSIS

For more information about the CSIS see section 2.2.1.3.

The **public endpoint** is csis.myclimateservice.eu. The actual production and development Drupal site configuration is stored in the separate **private** clarity-csis-drupal repository. The custom modules and integrated apps are usually deployed from the master branch.

2.2.3.4 Duplicity

For more information about the Duplicity system see section 2.2.1.4.

The production configuration is maintained in branch csis.ait.ac.at.

2.2.4 Other services

The CSIS relies on several external services and application that are not deployed as docker containers on csis.ait.ac.at or csis-dev.ait.ac.at virtual servers.

2.2.4.1 Myclimateservices profile service

The myclimateservices profile service is the central place for the administration of user and organisation data. This is the page on which the user registers for all other services of myclimateservices.eu (CSIS, events, marketplace) and maintains his profile and the profile of his organisation(s). profile.myclimateservices.eu is also the point of Single Sign On (SSO) for all services of myclimateservices.eu. The single pages will only receive the data of the user's profile, which are needed for that service.

Implementation

The profile service is implemented as Drupal 8 instance. To provide the Single Sign On functionality the contributed Drupal 8 CAS server module is used. So, the other services implemented in Drupal 8 can use the Drupal CAS module to participate on the Single Sign On. The parts of the user profile needed for a specific Service can be exposed to the service.

- as CAS attributes during the SSO process or
- as a REST endpoint at <https://myclimateservices.eu>

Deployment

The profile service is deployed as composer project. The only requirement is a LAMP system (Linux Apache Mysql PHP web server) with installed php package manager composer. Please prepare a website in the web servers' configuration. You will need access credentials to the composer repository of myclimateservices.eu

Installing the code base

Only code dedicated to the profile service is located in the repository. All other code is managed by composer packages. To get the whole code for running the Drupal website:

```
cd /path/to/website
git clone https://gitlab.com/smart-cities-consulting/mcs-profile.git ./
composer install
composer drupal:scaffold
```

Install Drupal

Create a database for the System:

```
mysql -u root -p
CREATE DATABASE <databasename>;
GRANT ALL PRIVILEGES on <databasename>.* to "<dbuser>"@"localhost" IDENTIFIED BY
"<password>";
```

Open a web browser and visit the URL of the page. Follow the install wizard of Drupal 8 and enter your database credentials when asked

Import Drupals configuration

```
cd /path/to/website
vim web/sites/default/setting.php -> $config_directories['sync'] = '../config/sync';
vendor/bin/drush cim
vendor/bin/drush cr
```

- **repository:** <https://gitlab.com/smart-cities-consulting/mcs-profile>
- **public endpoint:** profile.myclimateservices.eu

2.2.4.2 AIT EMIKAT

Emikat is used by the CSIS to perform a wide range of different calculations for the Studies created in the CSIS. Communication between Emikat and CSIS is established by a flow of individual REST calls between the endpoints of the two systems, which were created specifically for that purpose.

General procedure

1. CSIS needs to trigger EMIKAT (by sending some initial information and receiving an Emikat's internal ID of the Study, which is later used to identify the Study in consecutive REST calls.) once the user has provided sufficient data for the Study and executed the Study calculation request.
 2. Emikat requests additional data it needs from the Drupal REST endpoints, which were provided in the initial trigger request from the CSIS. The CSIS then pulls information about the status of the current calculation progress in regular intervals until all calculations are completed.
 3. During the calculation the user cannot request a second triggering of the calculation. (Actually, they can. It's not a good idea because the first calculation will be finalized before the second one starts.)
- **repository:** <https://github.com/clarity-h2020/emikat> for basic documentation but the code is at the AIT-internal repository (EMIKAT is not Open Source)
 - **public endpoints:** <https://service.emikat.at>

Three types of EMIKAT – related endpoints exist.

- First, there are endpoints of the https://csis.myclimateservice.eu/rest/emikat/* form. These are CSIS REST interfaces that make it simpler for EMIKAT to get the necessary information about study, data package, resources, adaptation strategies and adaptation options from CSIS (see <https://github.com/clarity-h2020/emikat>).
- Second, there are REST endpoints provided by EMIKAT, with the https://service.emikat.at/EmiKatTst/api/*/* form. These endpoints provide access to results of the EMIKAT data calculations, e.g. “population exposure”, “hazard local effect” of heatwaves and pluvial floods, impact indices and effects of applying the adaptation options, in tabular form and as averages over the project area.
- Finally, there are also endpoints of the https://service.emikat.at/geoserver/clarity/wms?* type. These provide access to results of EMIKAT calculations in map form.

A complete list of endpoints of the type 2 and 3 is provided at [https://github.com/clarity-h2020/csis/wiki/Services-endpoints-\(used-by-CSIS\)](https://github.com/clarity-h2020/csis/wiki/Services-endpoints-(used-by-CSIS)).

2.2.4.3 ATOS GeoServer

GeoServer is a Java-based software server that allows users to view and edit geospatial data. Using open standards set forth by the Open Geospatial Consortium (OGC)²¹, GeoServer allows for great flexibility in map creation and data sharing²². This sever provides all the layers needed by the CSIS for creating the maps showed to the users.

An instance of Geoserver has been deployed in the Atos server using Docker Compose. Docker Compose allows virtualizing the applications as a microservices. The purpose of that is to be able to migrate/redeploy all those microservices in another server if needed without losing any data/configuration made previously for each container.

The Geoserver provides a plugin (Import plugin) which eases the migration of all the raster data stored previously in the MeteoGrid Geoserver. Thank you to this plugin, more than four hundred of layers has been migrated from this server.

²¹ <http://www.opengeospatial.org/>

²² <http://geoserver.org/about/>

For publishing vector data demanded by the CSIS an instance of PostgreSQL²³ database (v.12) has been deployed as well. The database stores all the vector data generated in the project as Input Layers (Heat waves and Pluvial flood layers) for more than five hundred cities in Europe. To manage and interact with the spatial data a specific extension has been installed in the database called PostGIS²⁴ (v.2.4).

To keep the configuration and the data of the services deployed in the server (Geoserver and PostgreSQL) two volumes has been mapped to the server file system.

The last service deployed is a reverse proxy (Nginx²⁵), this service allows to configure a DNS for accessing the other applications. The proxy also provides the possibility of using SSL certificates. The certificate has been generated by Let's Encrypt²⁶ technology.

- **public endpoint:** geoserver.myclimateservice.eu

2.2.4.4 METEOGRID Transport Application

The Transport Module aims to incorporate natural risks in the context of Climate Change into the design and management of the transport infrastructure; it is meant to provide the user with a tool that facilitates the risk assessment of a road project in the face of climate variability and change. It addresses the needs of several types of potential users who in turn will demand different climate information with a different focus.

It has been developed as an external tool that has been integrated into the CSIS. It can be accessed from within the CSIS and exchanges information with it to ensure that the users have a seamless experience when using the tool for their transport studies.

Implementation

The Transport Module architecture follows a traditional MVC model using the Django web framework as base technology along with:

- Mezzanine, an Open Source Content Management Platform using the Django framework
- PostgreSQL database
- Javascript, front-end development
- Openlayers a high-performance, feature-packed library for all your mapping needs
- The integration with the CSIS is made using an API that has been developed using the existing Django infrastructure.

Deployment

The deployment of the TM needs a Linux-based OS in which the following packages must be installed:

- python3-mapnik
- python3-psycpg2
- python-dev
- build-essential

²³ <https://www.postgresql.org/>

²⁴ <https://postgis.net/>

²⁵ <https://www.nginx.com/>

²⁶ <https://letsencrypt.org/es/>

- virtualenvwrapper
- postgis
- git
- gfortran
- nodejs

Once available, the application can be obtained from the repository on any directory that the administrator deems suitable.

```
git clone ssh://git@gitlab.meteogrid.com:20023/meteogrid/emmet.git
cd emmet
```

Then it is needed to set up the database:

```
sudo su - postgres
createuser [USERNAME]
createdb -O [USERNAME] emmet_beta
psql emmet
  create extension postgis;
exit
```

A python virtual environment is then set up:

```
mkvirtualenv --system-site-packages --python=/usr/bin/python3 emmet
workon emmet
pip install -r requirements.txt
and the javascript packages needed are installed and built using npm:

npm install
npm run-script build
```

The next step is to configure the Django application. To do that a local settings file is needed in which the information regarding the database, and other local paths must be supplied. A copy of this file can be found in the repository: `local_settings.py`. More information on this file and its options can be found in Django documentation pages: Settings

Lastly the database needs to be set up from an existing dump in which all the necessary data for the TM to work has already been set. The dump file is available in the repository as a json file and can be imported into the freshly installed system by using:

```
python manage.py loaddata < dumped_data.json
```

A web server is also needed to serve the application. The TM will work with either Apache HTTP Server or Nginx. The configuration of any of these web servers will depend on the system in which it is being installed and the best approach is to follow the documentation of the selected web server.

- **repository:** gitlab.meteogrid.com/meteogrid/emmet
- **public endpoint:** clarity.saver.red

2.2.4.5 MyClimateServices.eu marketplace

Implementation

The marketplace is a separate Drupal 8 website which can be hosted on any (virtual) server which hosts a LAMP system (Linux Apache Mysql PHP) with installed php package manager composer. It participates on the Single Sign On of profile.myclimateservices.eu. Thus, the user can switch between CSIS and Marketplace without need to login on another page again.

Code is mainly managed by composer packages. Only the custom/own code dedicated to marketplace is in the repository. Custom/own Code which is most likely usable for other pages as well(profile, myclimateservices,events.myclimateservices.eu,...) is managed using a own composer repository at <https://repository.myclimateservices.eu/>. This repository is built with satis and fetches the code from the gitlab repositories of the custom Drupal modules.

Synchronisation of needed data of users and organisations from profile.myclimateservices.eu is done by using the REST endpoints on profile created for marketplace. The synchronisation uses Drupal's built in "migrate" module to call the endpoints, which expose the users and organisations created and changed in the last 5 minutes. The necessary migrations are run by a shell script which is called by cron every 5 minutes and runs the "drush" commands for migrations

```
#!/bin/bash
cd /directory/of/marketplace

vendor/bin/drush mrs sync_organisations
vendor/bin/drush mim sync_organisations --update
vendor/bin/drush mrs sync_users_from_profile
vendor/bin/drush mim sync_users_from_profile --update
vendor/bin/drush mrs link_users_to_cas
vendor/bin/drush mim link_users_to_cas --update
```

Deployment

Installing the code base

```
cd /path/to/website
git clone https://gitlab.com/smart-cities-consulting/mcs-marketplace.git ./
composer install
composer drupal:scaffold
```

Install Drupal

Create a database for the System:

```
mysql -u root -p
CREATE DATABASE <databasename>;
GRANT ALL PRIVILEGES on <databasename>.* to "<dbuser>"@"localhost" IDENTIFIED BY "<password>";
```

Open a web browser and visit the url of the page. Follow the install wizard of Drupal 8 and enter your database credentials when asked

Import Drupals configuration

```
cd /path/to/website
vim web/sites/default/setting.php -> $config_directories['sync'] = '../config/sync';
vendor/bin/drush cim
vendor/bin/drush cr
```

Create the script and cron job to synchronise data from profile.myclimateservces.eu

Synchronisation between Dev and Prod

There is no content which has to be developed both on a development environment and the production system. So only code and configuration must be transferred from dev to prod. On the Dev system do from the directory with the composer.json:

```
vendor/bin/drush cr
vendor/bin/drush cex
git add --all
git commit
git pull
git push
```

This exports first the changes in configuration and updates then the repository with the changes Then on the production system:

```
git pull
composer install
vendor/bin/drush updb
vendor/bin/drush cim
vendor/bin/drush cr
```

This updates the codebase the composer.json and composer.lock files Then composer is told to install new packages or update old ones if the where updated on dev. Afterwards drupals databes is updated if the code changes in already activated modules have an effect on the database. The configuration is imported, and the cache is flushed to be sure all changes are applied for the user.

Update

Drupal good practices tell to update first the dev system and check if nothing will break. In the directory with the composer.json file:

```
composer update --with-dependencies
vendor/bin/drush updb
vendor/bin/drush cr
```

After that check if

- composer applied pathes
- the page is not broken

If everything on the dev system is OK sync dev to prod as described above. Sometimes the update process will require more steps like deleting directories or update a single module in two steps. In that case you have

sync to prod after ever single step or replay the update process on prod instead the to update only with this procedure

- **Repository:** <https://gitlab.com/smart-cities-consulting/mcs-marketplace>
- **public endpoint:** myclimateservice.eu

2.2.5 Synchronisation between the Development and Production environments

The synchronisation between the development and production system consists of two separate steps for synchronising content and configuration.

2.2.5.1 Importing/Exporting configuration

We're using the Configuration Split²⁷ module to manage, which configuration will be enabled on the Development or Production server only and which will be enabled on both. Configuration will always be imported/exported completely as *.yaml* files, but into different directories, which will (depending on the split settings) be used or ignored on the two environments. That way the configuration will always be completely available in our private Git repository.

2.2.5.2 Synchronisation configuration process

1. Changes will be made on Dev server (e.g. a new field or a module will be enabled/disabled)
2. View all the changes at `/admin/config/development/configuration` and:
 1. If this change should not take effect on Production, then the split needs to be configured to exclude the change.
 2. If this change should take effect on Production, the split doesn't require any changes.
3. Via terminal export the configuration using `drush config-split:export`.
4. Push configuration to private repository and pull on Production server.
5. Run composer install first if *composer.json* file changed, otherwise configuration import might fail.
6. If there are any new modules added by composer, they now need to be installed in Drupal via the "Extend" page in the Backend UI.
7. Import configuration.
8. Check if updates in DB are necessary.
9. Clear cache.

2.2.5.3 Importing/Exporting content

The Migrate module²⁸ is used for synchronization, which is part of the Drupal Core and some of its additional modules (Migrate Plus and Migrate Tools). The content is pulled from the Development environment via REST-Views, which means that Development needs to be accessible via HTTP.

²⁷ https://www.drupal.org/project/config_split

²⁸ <https://www.drupal.org/docs/8/api/migrate-api/migrate-api-overview>

The general workflow will be that certain nodes (GL-step templates, data packages and all their referenced entities) and all the taxonomies will be exported in the Development environment and imported in the Production environment. The following actions are performed:

- New elements will be created in Production.
- Existing elements will be updated.
- Elements removed in Development will also be removed in Production.
- For synchronizing only published elements will be considered.

2.2.5.4 Managing the migration configuration files

The target site (in our case the Production environment) needs for each content type the migration configuration stored in a *.yml* file (these files are stored and synchronized with the other configuration *.yml* files). To add a new migration configuration:

1. Create the *.yml* file (use existing ones for reference)
2. Add the file in the Drupal backend UI.
3. Synchronize configuration between the Development and Production environments.
4. If an existing migration needs to be changed, use the "Single item export" feature under same link and export desired migration including the migration UUID, which will be needed for the re-import later after changes to file are done

2.2.5.5 Synchronization content process

In the Docker-container of Prod run either the provided bash script to synchronize all content in the correct order or synchronize individual content types.

3 CSIS Technical Validation

This repository serves for managing all technical validation and testing related tasks of the CLARITY Climate Services Information System (CSIS).

Technical validation of integrated CSIS and its Building Blocks²⁹ is performed by means of unit tests, integration tests and user acceptance tests with the aim to increase overall software quality, stability and value proposition. Actual tools used for implementing CSIS technical validation concept include:

- Service monitoring for detecting any service disruptions and for initiating appropriate remediation measures.
- Automated unit- and integration tests based cypress.io to ensure that user interfaces as well user interactions are working as specified.
- User Acceptance Tests.

3.1 Unit tests

Automated unit tests for several apps that are built on Jenkins CI and that are deployed together with CSIS Drupal Containers are performed on Jenkins CI. These include:

- CSIS Helpers JS Module
- Map Component
- Simple Table Component
- Scenario Analysis



S	W	Name ↓	Last Success	Last Failure	Last Duration	Fav
		csis-cypress	26 min - #666	2 hr 26 min - #665	3 min 24 sec	
		csis-dev-cypress	2 hr 12 min - #126	12 min - #127	3 min 7 sec	
		csis-helpers-js	5 days 1 hr - #19	11 days - #10	1 min 37 sec	
		emikal-cypress	2 hr 26 min - #278	N/A	24 sec	
		health-check-cypress	7 hr 38 min - #105	N/A	23 sec	
		map-component	3 days 1 hr - #41	1 mo 3 days - #32	19 min	
		profiles-cypress	1 hr 20 min - #553	N/A	27 sec	
		simple-table-component	4 days 1 hr - #29	N/A	15 sec	

Icon: [S](#) [M](#) [L](#)

[Legend](#) [RSS for all](#) [RSS for failures](#) [RSS for just latest builds](#)

Figure 10: Tests performed.

If any of the unit tests fails, developers are notified by email and a new issue is posted in the respective repositories.

²⁹ <https://clarity-h2020.github.io/csis-architecture/docs/building-blocks/>

3.1.1 Jenkins CI

3.1.1.1 Jenkins pipelines

For creating the pipelines in Jenkins, the Blue Ocean plugin has been used. This plugin allows users to graphically create, visualize and diagnose Continuous Delivery (CD) Pipelines³⁰. The main features are the following³¹:

- **Sophisticated visualizations** of continuous delivery (CD) Pipelines, allowing for fast and intuitive comprehension of your Pipeline's status.
- **Pipeline editor** - makes creation of Pipelines approachable by guiding the user through an intuitive and visual process to create a Pipeline.
- **Personalization to suit the role-based** needs of each member of the team.
- **Pinpoint precision** when intervention is needed and/or issues arise. Blue Ocean shows where in the pipeline attention is needed, facilitating exception handling and increasing productivity.
- **Native integration for branch and pull requests**, enables maximum developer productivity when collaborating on code with others in GitHub and Bitbucket.

The following image shows an example how to create a new pipeline using the Blue Ocean plugin:

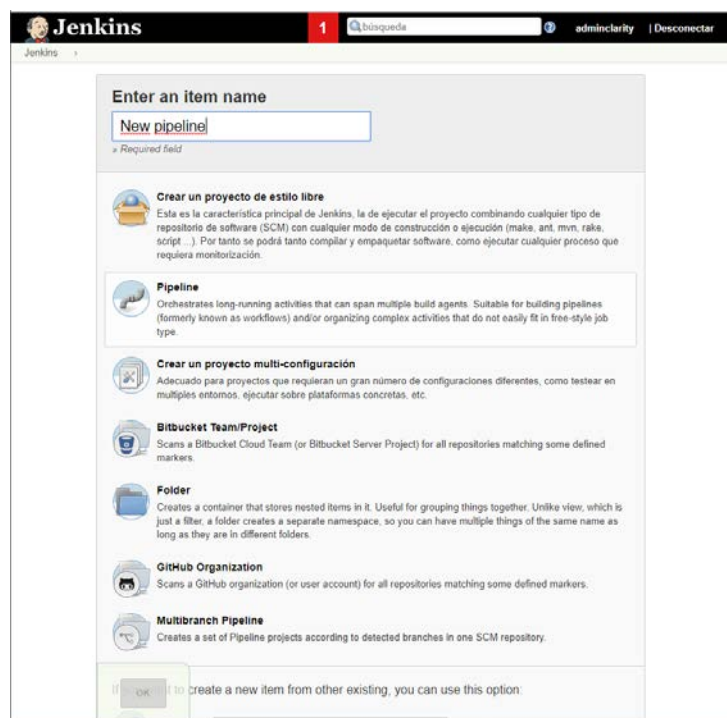


Figure 11: New pipeline creation.

³⁰ <https://www.jenkins.io/projects/blueocean/about/>

³¹ <https://www.jenkins.io/doc/book/blueocean/>

For the configuration, the “GitHub hook trigger for GIT SCM polling” in the Build Triggers section has been selected in order to trigger the test every time a push is made to the master branch in the Github repository. Some other properties have been changed in the Pipeline section as shows the image bellow:

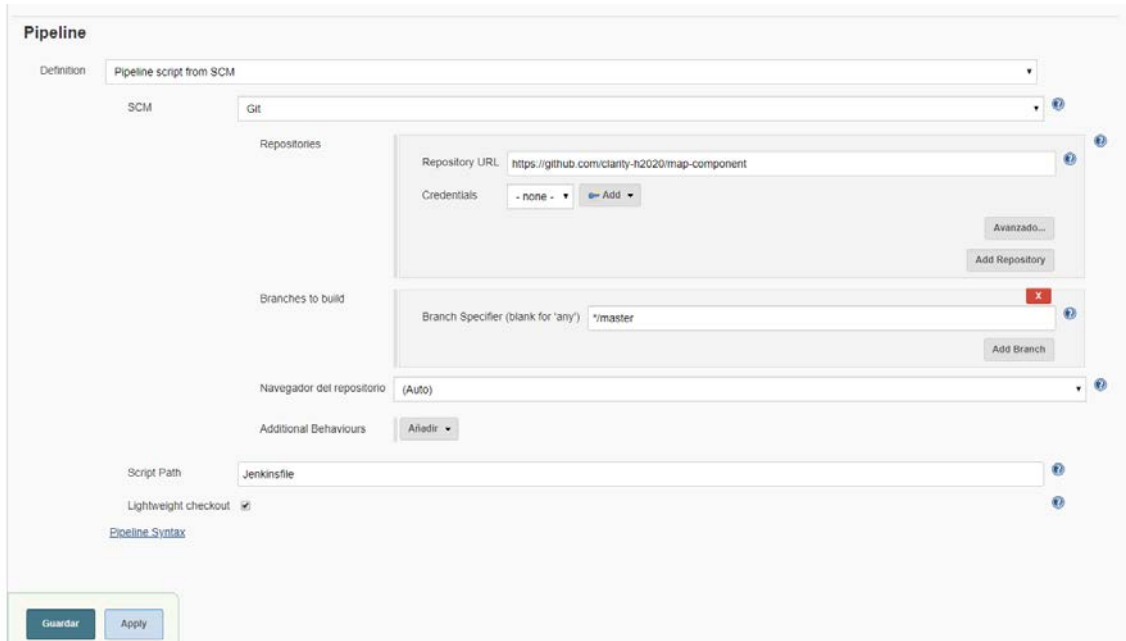


Figure 12: Pipeline configuration

Note the repository credentials is empty because a webhook will be configured in the Github.

3.1.1.2 GitHub webhooks

A webhook is an HTTP call back, an HTTP POST that occurs when something happens through a simple event-notification via HTTP POST³². GitHub webhooks in Jenkins are used to trigger the build whenever a developer commits something to the master branch. Bellow an example how to configure a webhook in the GitHub:

³² <https://dzone.com/articles/adding-a-github-webhook-in-your-jenkins-pipeline>

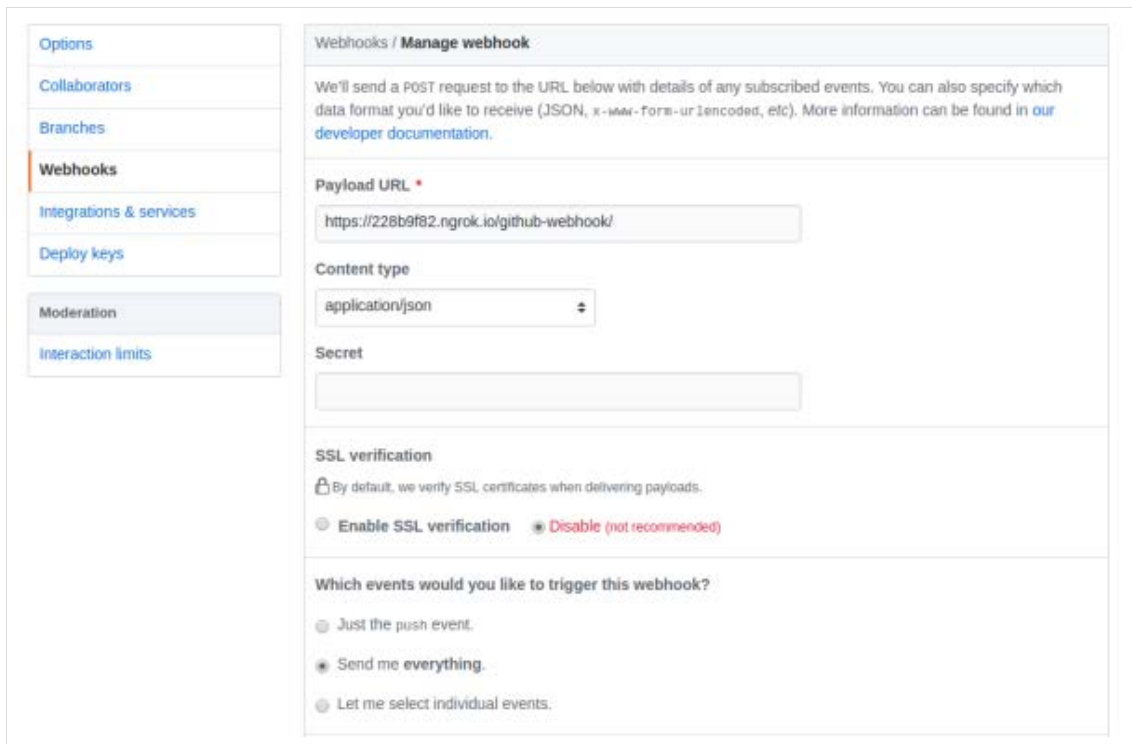


Figure 13: GitHub webhook configuration.

3.1.1.3 Jenkins file

For make all work, is necessary to add a Jenkins file to the repository. This file will automate build a Node.js and React application in Jenkins. The configuration of this file depends which frameworks, dependency manager the project is using. In this case, the pipeline is configured for working with Yarn. The Jenkins file looks like:

```

pipeline {
  agent {
    docker {
      image 'node:latest'
      args '-p 3000:3000'
    }
  }
  stages {
    stage('Install') {
      steps {
        sh 'yarn install'
      }
    }
    stage('Test') {

```

3.2 Integration Tests

UI Integration tests are performed against the CSIS development and production system as well as the myclimateservices user portal³³ and the AIT EMIKAT status page. The tests are performed with help of [cypress.io](https://www.cypress.io) and executed on Jenkins CI (Figure 11). The test specifications are maintained in repository csis-technical-validation:

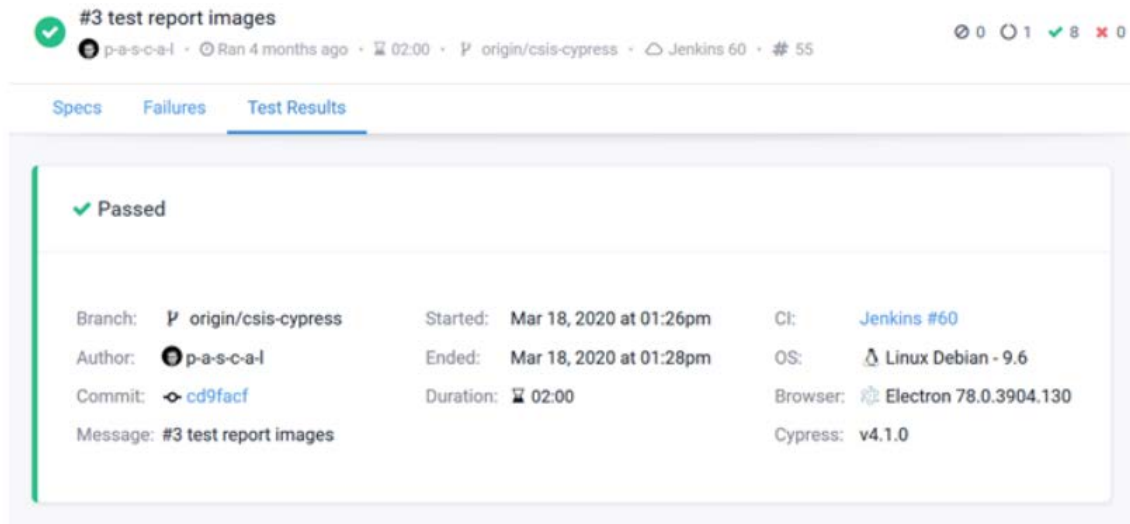


Figure 14: Integration test report.

If any of the test fails, the CI system will automatically post a new issue in the repository csis-technical-validation.

3.3 User Acceptance Tests

To ensure that CSIS, especially the novel climate screening tool³⁴, is "fit for purpose", that is, it delivers the proposed value, the CSIS Testing Team performs acceptance test following the Acceptance Test Specification³⁵. Feedback is collected with help of the GitHub platform by means of testing issues in repository csis-technical-validation.

³³ <https://profile.myclimateservices.eu/>

³⁴ <https://myclimateservices.eu/en/news/how-csis-these-videos-you-learn-how-you-participate-our-climate-tool>

³⁵ The main aim of these tests is to ensure that our "minimal viable product" is "fit for purpose", that is, it delivers the proposed value (European-level climate change impact screening service). Thereby, we want to eliminating usability issues (difficult to understand, too complicated, illogical or incoherent behaviour) and bugs (something is not working as expected or not working at all). While we want to collect also feedback for improving the system, our current focus is on making *existing* functionality work as expected rather than collecting many new feature requests for new functions.

4 Software licensing

The following table shows the main components and their license, the information has been taken from the D.5.5 Exploitation and business plan (v3).

Component	Owner	License
Marketplace	SCC	GNU General Public License version 2 (GPLv2)
CSIS for urban infrastructures	CISMET, AIT, ATOS	Open source with restrictions from proprietary license of EMIKAT.
EMIKAT connector to CSIS	AIT	Proprietary license. Public use from demonstration CSIS will be possible for free. Other commercial use will have to be discussed with owner.
User Interface Integration Platform	CISMET, AIT, ATOS	GNU Lesser General Public License v3.0, it can be used for commercial purposes by other partners
Scenario Management	CISMET, AIT	GNU Lesser General Public License v3.0, it can be used for commercial purposes by other partners
Map Component	CISMET	GNU Lesser General Public License v3.0, it can be used for commercial purposes by other partners
Scenario Analysis	CISMET	GNU Lesser General Public License v3.0, it can be used for commercial purposes by other partners
Report Generation	CISMET, AIT	GNU Lesser General Public License v3.0, it can be used for commercial purposes by other partners
Table Component	CISMET	GNU Lesser General Public License v3.0, it can be used for commercial purposes by other partners
Hazard/element at risk-specific vulnerability models	CSIS	Logic models by PLINIVS under “CC-BY Creative Commons Attribution 4.0 International”, resulting implemented software under “GNU Lesser General Public License v3.0”. C.C. confirms compatibility from CC BY-SA 4.0 to GPLv3 (https://creativecommons.org/2015/10/08/cc-by-sa-4-0-now-one-way-compatible-with-gplv3/)
Impact models for heat waves and floods	PLINIVS	Logic models by PLINIVS under “CC-BY Creative Commons Attribution 4.0 International”, resulting implemented software under “GNU Lesser General Public License v3.0”. C.C. confirms compatibility from CC BY-SA 4.0 to GPLv3 (https://creativecommons.org/2015/10/08/cc-by-sa-4-0-now-one-way-compatible-with-gplv3/)
Preparedness/resilience/adaptation models	PLINIVS	Logic models by PLINIVS under “CC-BY Creative Commons Attribution 4.0 International”, resulting implemented software under “GNU Lesser General Public License v3.0”. C.C. confirms compatibility from CC BY-SA 4.0 to GPLv3 (https://creativecommons.org/2015/10/08/cc-by-sa-4-0-now-one-way-compatible-with-gplv3/)
Tool for visualization of climate projections	MeteoGrid	Proprietary license. Public use from demonstration CSIS will be possible for free. Other commercial use should be discussed with owner.
Advanced Climate Change Screening calculation	CSIS	Proprietary license. Public use from demonstration CSIS will be possible for free. Other commercial use will have to be discussed with owner.
Tool to evaluate exposure, analyse vulnerability and assess risks	MeteoGrid	Proprietary license. Public use from demonstration CSIS will be possible for free. Other commercial use should be discussed with owner.

4.1 Proposed licensing header for source code files

As soon as licensing terms are defined for CLARITY software, every source code file should include some metadata describing when and who developed it as well as other basic information.

The following template could be used in this regard.

```
/**
 * CLARITY
 * Integrated Climate Adaptation Service Tools for Improving Resilience Measure Efficiency
 * @partner    ATOS
 * @author     Mario Nunez
 * @date       01/12/2017
 */
```

5 Reporting on the technical work performed in the last months concerning CSIS and Marketplace

Management Project for coordinating the Climate Service co-creation process. High-level tasks (issues) are defined in this repository and then broken down into concrete implementable issues in different sub repositories. These issues correspond to new features to implemented, open bugs and active bugs, which in turn are subdivided into “low priority”, “to do”, “in progress” and “done” depending on their priority and actual status.

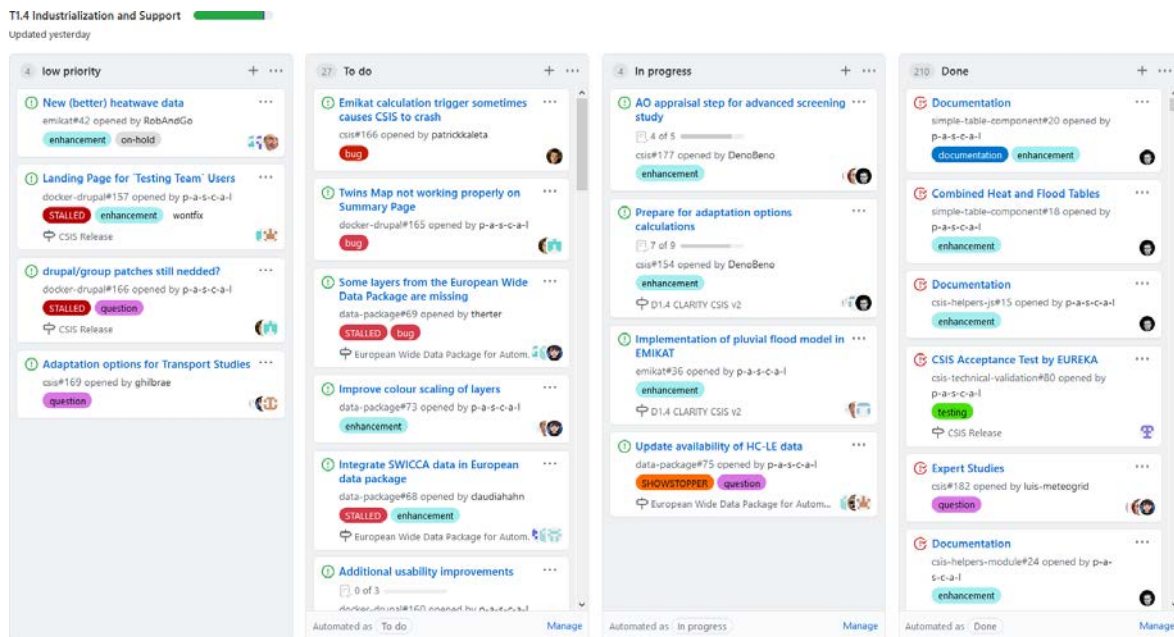


Figure 15: Overview of T1.4 issues and bugs

5.1 New features implemented

5.1.1 Adaptation options in the CSIS workflow

Depending on the selected study type, the users can select different adaptation options. Studies of the type “Advanced Screening: Urban Infrastructure” have predefined adaptation strategies, which are bundled adaptation options that all apply to a single target land category. For each available land category (buildings, build-open spaces, roads and vegetated areas) the user can select up to one adaptation strategy in the “AO Appraisal” step. The user can then decide to restart the study calculations. The selected adaptation strategies will then be applied to the study area and the resulting impact is calculated by an external service. The results of such an adapted study scenario can then be viewed and compared to the baseline scenario (where there are no adaptation strategies applied) in the final step (AO Implementation).

All study types allow users to view available adaptation options in the “Adaptation options Identification” step. Which adaptation options are shown is depending on the selected Data Package. Each data package contains a list of the adaptation options it supports. The users can include these adaptation options in their final study report and, based on that selection, view showcases of similar adaptation options in use in the “Twins” tab and include these in the report as well. Except for the study type “Advanced Screening: Urban Infrastructure”, no calculations are performed for the selected adaptation options in the remaining study types.

5.1.2 Local effects calculation for pluvial floods and new impact indices

Pluvial floods have been added as a second hazard index that is downscaled to 500m resolution on the fly in the advanced urban screening. In addition, multiple impact indices have been added for heat/population and flood/infrastructure pairs:

- Population density class (1..5 index based on population number per cell)
- Heat discomfort (1..5 index based on UTCI)
- “Heat wave impact” indicator (heat discomfort x population density class)
- Direct (hospitalization costs) and indirect (lost productivity) costs of heat waves
- Monetary value of the lives lost due to heat mortality (average “lost life” value is set to 1.6M€)
- Flood “damage class” (0..5 index indicating the level of damage for residential buildings, non-residential buildings and traffic infrastructure)

5.1.3 Economic impact of floods on roads, residential and non-residential buildings - Updates in the Data Packages

The “**European Wide Data Package**” (for advanced urban screening) has been updated through the addition of SWICCA³⁶ flood recurrence resource and new EMIKAT indices (on the fly calculation of flooding hazard, multiple new impact indices and effects of adaptation options). SWICCA dataset shows the recurrence of daily river flows corresponding to return periods of 2, 5, 10, 50, and 100 years. This data has also been made available as map layers via the ATOS Geoserver. Minor additions have also been made to the descriptions of most indices.

- The “**European DP for basic screening**” has been created by removing the dynamically generated (EMIKAT) indices from this data package.
- The “**DC1 – Naples Metropolitan Area**” data package has been updated to reflect the final expert study results and contain data on heat islands and pluvial floods as well as the updated adaptation options.
- The “**DC2- Stockholm green infrastructure**” data package has been created and updated by SMHI and contains resources of heat and comfort for a regional domain centered on Stockholm.
- The “**DC3 – Urban heat waves, urban heat islands, air ventilation in Linz, Austria**” data package has been updated to reflect the final expert study results related to heat islands in the city of Linz.
- The “**DC4 Spanish Transport Network**” data package has been updated with additional data on precipitation, floods, heat episodes as well as adaptation options. In addition, a pan-European data package for traffic applications has been partially implemented.
- **Three “RESCCUE” data packages** featuring the data produced in this project have been added, one with climatic projections for **Barcelona**, second with **Bristol** data and the third (still not finalized) with data for **Lisbon**. These data packages contain a number of resources on damage from pluvial flooding at different recurrence periods.

³⁶ <https://swicca.eu/>

5.2 CSIS Management Project

5.2.1 Known bugs and issues closed

Known bugs and issues closed	URL
Expert Studies	https://github.com/clarity-h2020/csis/issues/182
Making Studies Public	https://github.com/clarity-h2020/csis/issues/179
Create new Showcases directly on PROD rather than on DEV	https://github.com/clarity-h2020/csis/issues/178
Calculation can't be triggered on test instance of CSIS	https://github.com/clarity-h2020/csis/issues/175
Search radius isn't working in "twins"	https://github.com/clarity-h2020/csis/issues/176
New nodes aren't produced for study types that aren't published	https://github.com/clarity-h2020/csis/issues/174
Invalid web certificates?	https://github.com/clarity-h2020/csis/issues/173
Table view in "Hazard characterization" step	https://github.com/clarity-h2020/csis/issues/172
Update Texts for Transport Studies	https://github.com/clarity-h2020/csis/issues/171
Adaptation Options for Expert Studies	https://github.com/clarity-h2020/csis/issues/170
Login problems for user account (SSO not working)	https://github.com/clarity-h2020/csis/issues/168
Transport module embedded screen	https://github.com/clarity-h2020/csis/issues/167
Permissions for creating TM studies	https://github.com/clarity-h2020/csis/issues/165
New users on the CSIS	https://github.com/clarity-h2020/csis/issues/164
Perform CSIS Beta Release	https://github.com/clarity-h2020/csis/issues/162
Vulnerabilities for the simple screening	https://github.com/clarity-h2020/csis/issues/161
Paging in views broken	https://github.com/clarity-h2020/csis/issues/160
Show new cost fields in all views involving adaptation options	https://github.com/clarity-h2020/csis/issues/158
Taxonomy terms not filtered when using an Entityqueue	https://github.com/clarity-h2020/csis/issues/157
Create or clone datapackage?	https://github.com/clarity-h2020/csis/issues/156
Data model for "adaptation project"	https://github.com/clarity-h2020/csis/issues/155
Allow Emikat calculation tests to be run by the UI testing service	https://github.com/clarity-h2020/csis/issues/153
The summary page of some studies loads very long	https://github.com/clarity-h2020/csis/issues/152
Make GL-Step descriptions editable in Study	https://github.com/clarity-h2020/csis/issues/151
Issues with new study step templates in CSIS	https://github.com/clarity-h2020/csis/issues/150
Option to create an Advanced Transport Study has disappeared	https://github.com/clarity-h2020/csis/issues/149
Final PDF report for Transport Infrastructure Studies (TM Studies)	https://github.com/clarity-h2020/csis/issues/148
CSIS demo for workshop in Linz	https://github.com/clarity-h2020/csis/issues/147
Revising "Data" tab	https://github.com/clarity-h2020/csis/issues/146
Prepare CSIS BETA Release	https://github.com/clarity-h2020/csis/issues/145
Aghios Dimitrios Study Case: Add as owner	https://github.com/clarity-h2020/csis/issues/144
Change the resource description from plain text to formatted text	https://github.com/clarity-h2020/csis/issues/143
CSIS User permissions and access to tabs and creation links	https://github.com/clarity-h2020/csis/issues/142
User access to external layers protected by user/passwd	https://github.com/clarity-h2020/csis/issues/141
Improving the text content within CSIS and the data-package resources	https://github.com/clarity-h2020/csis/issues/140
Give users option to order report images in the summary	https://github.com/clarity-h2020/csis/issues/139

No impact shown for Lodz study	https://github.com/clarity-h2020/csis/issues/138
Future exposure	https://github.com/clarity-h2020/csis/issues/137
"Cannot load a NULL ID" warning	https://github.com/clarity-h2020/csis/issues/136
Provide a way to import/export/synchronize content	https://github.com/clarity-h2020/csis/issues/135
Transport Module integration description	https://github.com/clarity-h2020/csis/issues/134
Revising available GL-Steps in Study type "Basic Screening: General"	https://github.com/clarity-h2020/csis/issues/132
Adaptation options in screening (round 3)	https://github.com/clarity-h2020/csis/issues/131
Allow visitors to see (some) studies.	https://github.com/clarity-h2020/csis/issues/130
CSIS UI General Improvements	https://github.com/clarity-h2020/csis/issues/127
Integration of transport application into the CSIS	https://github.com/clarity-h2020/csis/issues/122
Enable TLS for clarity-h2020.eu	https://github.com/clarity-h2020/csis/issues/120
Please add a new study type: "transport infrastructure screening"	https://github.com/clarity-h2020/csis/issues/118
Meta issue: User/Group Permissions and roles	https://github.com/clarity-h2020/csis/issues/114
CSIS Production Cloud Infrastructure	https://github.com/clarity-h2020/csis/issues/113
Would it make sense to upload the data that we want to visualise in tables and graphs on Drupal?	https://github.com/clarity-h2020/csis/issues/110
Allow user to define more than one preset	https://github.com/clarity-h2020/csis/issues/108
CSIS Demo for European Week of Regions and Cities	https://github.com/clarity-h2020/csis/issues/105
Report bugs, things to improve	https://github.com/clarity-h2020/csis/issues/104
How to choose which "variable combinations" to use in a study?	https://github.com/clarity-h2020/csis/issues/101
Adaptation options data model for screening, round 2	https://github.com/clarity-h2020/csis/issues/100
Should we revise the "Data" tab?	https://github.com/clarity-h2020/csis/issues/102
Please install the chart_suite	https://github.com/clarity-h2020/csis/issues/98
How to relate between data packages, study types and calculation methods?	https://github.com/clarity-h2020/csis/issues/91
Allow user to skip non-existing steps	https://github.com/clarity-h2020/csis/issues/82
Create (and update) wiki page listing all services endpoints we are using	https://github.com/clarity-h2020/csis/issues/61
Integration of EEA's Urban Adaptation Map Viewer	https://github.com/clarity-h2020/csis/issues/60
Json DataPackage Import: Install necessary Drupal modules	https://github.com/clarity-h2020/csis/issues/59
GRID harmonization	https://github.com/clarity-h2020/csis/issues/42
myclimateservice.eu DNS Entries	https://github.com/clarity-h2020/csis/issues/23
Populate CKAN with metadata from datasets	https://github.com/clarity-h2020/csis/issues/11
Setup CI: Jenkins + Maven + TestNG/Junit Docker container	https://github.com/clarity-h2020/csis/issues/8

Table 2: Known bugs and issues closed.

5.2.2 Known bugs and issues left open

The following table summarizes the bugs and issues that were still open at the end of August 2020. Most of the issues have been at least partially resolved, but either could not be fully resolved or the resolution wasn't fully tested.

Known bugs and issues left open	URL
Request Permissions for User	https://github.com/clarity-h2020/csis/issues/183
Advanced urban screening with EMIKAT calculation type disappeared in CSIS-dev	https://github.com/clarity-h2020/csis/issues/181
Documentation	https://github.com/clarity-h2020/csis/issues/180
AO appraisal step for advanced screening study	https://github.com/clarity-h2020/csis/issues/177
Adaptation options for Transport Studies	https://github.com/clarity-h2020/csis/issues/169
Emikat calculation trigger sometimes causes CSIS to crash	https://github.com/clarity-h2020/csis/issues/166
CSIS - Text for each step	https://github.com/clarity-h2020/csis/issues/159
Prepare for adaptation options calculations	https://github.com/clarity-h2020/csis/issues/154

Table 3: Known bugs and issues left open.

6 Conclusions

This report briefly describes the activities carried out in relation to task “T1.4 Industrialization and Support”, focusing on those aspects related to provision in the project of the proper means for facilitating the sharing, revision and testing of the source code of the components and services developed, which constitute the core elements of the CSIS and Marketplace prototypes.

To that end, task T1.4 provided the necessary repositories for hosting of the CLARITY software code, data and services as well as a set of tools to support the

- i) Improving the code quality (including issues and bug tracking) and
- ii) deployment of the resulting components and services in the Testing and Production environments from which the final solution is accessible by internal testers and final end-users respectively.

Most important improvements and bug fixes that were implemented during the last months include:

- **Integration of adaptation options** in the CSIS workflow, which now enables the user to use them in studies of the type “Advanced Screening: Urban Infrastructure” as these have a set of predefined adaptation strategies with bundled adaptation options.
- **Pluvial floods have been added as a second hazard** index that is downscaled to 500m resolution on the fly in the advanced urban screening. In addition, multiple impact indices were added for heat/population and flood/infrastructure pairs.
- Existing data packages provided by the CSIS were updated with **information about the economic impact of floods on roads, residential and non-residential buildings**.
- Various improvements of the **traffic screening service** and better integration thereof in CSIS.

Although a large part of the project effort in the final project year focused on the improvement, testing and bug fixing of the existing CSIS features, it was not possible to address all of them (as reported in section 5.2.1). However, none of these issues prevents the use of the service as is for demonstration purpose.

At the project end, CSIS is considered a feature-complete prototype that can be used to showcase the possibilities that the CSIS can offer to its end-users. Both the “advanced urban screening” and “traffic screening” services are of great interest to the stakeholders. As unique selling point, the advanced urban screening service is available more than 400 major urban areas in Europe and thousands of smaller cities in their surroundings. We are currently actively working towards designing the regional “exploitation” projects with the CLARITY demonstration regions and beyond.

7 References

- [1] M. Ángel Esbrí, M. Nuñez, D. Havlik, “D1.4 CLARITY CSIS v2,” Deliverable D1.4 of the European Project H2020-730355 Integrated Climate Adaptation Service Tools for Improving Resilience Measure Efficiency (CLARITY), 2 August 2020. [Online].
- [2] M. Ángel Esbrí, A. Geyer-Scholz, “D5.7 MyClimateService.eu Marketplace (v2),” Deliverable D5.7 of the European Project H2020-730355 Integrated Climate Adaptation Service Tools for Improving Resilience Measure Efficiency (CLARITY), 30 August 2020. [Online].
- [3] P. Dihé, “D4.2 CLARITY CSIS Architecture” Deliverable D4.2 of the European Project H2020-730355 Integrated Climate Adaptation Service Tools for Improving Resilience Measure Efficiency (CLARITY), 07 February 2019. [Online].
- [4] P. Dihé, H. Hummer, P. Kaleta, “D4.4 Technology Support Report v2” Deliverable D4.4 of the European Project H2020-730355 Integrated Climate Adaptation Service Tools for Improving Resilience Measure Efficiency (CLARITY), 02 December 2019. [Online].

8 Annexes

8.1 Multi Criteria Decision Analysis Tool

The Multi Criteria Decision Analysis Tool supports the analysis and comparison of (adaptation) scenarios regarding performance indicators that can be defined by the end user and thus leverages what-if decision support to investigate the effects of adaptation measures and risk reduction options in the specific project context, and allows the comparison of alternative strategies. Thereby the tool provides multi-criteria ranking functions to compare and rank different scenarios and corresponding adaptation plans according to different criteria and their relative weight and level of importance.

8.1.1 Requested functionality

Baseline requirements elicitation and the assessment of presently available Test Cases have yielded the following functional requirements for this Building Block:

Baseline functionality

- A highly interactive and user-friendly decision support tool.
- Supports state of the art Multi-Criteria Decision Analysis (MCDA) methods.
- Helps the decision maker and strategic planner to perform multi-criteria ranking of scenarios.
- Compares Indicators (quantity) derived (e.g. aggregated by an expert-provided Indicator Function) from complex and usually not easily comparable scenario data (e.g. outputs of impact models).
- Compares Criteria (quality) derived from Indicators by defining a level of satisfaction for each Indicator (qualification).
- Performs ranking of scenarios with respect to a user-defined decision strategy (e.g. Ordered Weighted Averages).
- Implements different weighting strategies.

Functionality requested by DC Test Cases

- From TC DC1: This Building Block should allow the following: 1. selection of the “weight” of predefined performance criteria (e.g. cost, avoided impacts, other co-benefits, etc.); 2. ranking of alternative scenarios following the selected criteria.
- From TC DC4 080: The Multi-Criteria-Analysis Decision Support Tool is used to help the user on decision making based on standardized indicators derived from impact scenario data.

8.1.2 Exploitation Requirements assessment

The assessment of the Exploitation Requirements identified the following concrete technical and functional implications on this Building Block:

- From “Establish trust in Climate Services and their providers”: Uncertainty should be quantifiable, e.g. derived from statistical properties. Then, uncertainty could be considered as an indicator that

can be transformed (normalized) to criteria. This way, uncertainty could be incorporated into Multi Criteria Analysis.

- From “Follow a multi-sectoral approach that crosses the boundary of climate sciences”: Depending on the type of project as well as the sector and fields of interest of end users, the importance of different parameters will vary. E.g. For most project managers, financial parameters/indicators and information will be of higher interest, than climatic indicators and information by itself, while end users with a focus on risk assessment in order to keep people save will value indicators about the occurrence and intensity of climate related hazards higher, than financial parameters. In order to provide pinpointed information for all relevant parties, it must be ensured that the results and provided information suits their respective needs. If there is the need to rank different scenarios, this could be e.g. done by implementing a weighing system for indicators (automatic per type of end user or end users set the weighting factors according to their preferences) for the Multi Criteria Analysis. With indicators, being part of the results provided to end users, those indicators must address all possible questions related to Climate Services end users might have. Therefore, indicators need to cover a wide variety of topics (climate, air quality, financial, social, etc.).
- From “Use, define and promote open standards for data and services”: A simple (KISS principle) standardized format for Indicators that are relevant for the Multi Criteria Decision Analysis must be defined. Technically, (impact) model output must be transformed (e.g. by an aggregation or Indicator Function) into a standardized Indicator Set so that the Indicators can easily be compared or visualized by the respective tools.

8.1.3 Technology support

Figure 13 gives an overview on the technological possibilities and the related open-source frontend and backend software components that have been selected for the Technology Support Plan.

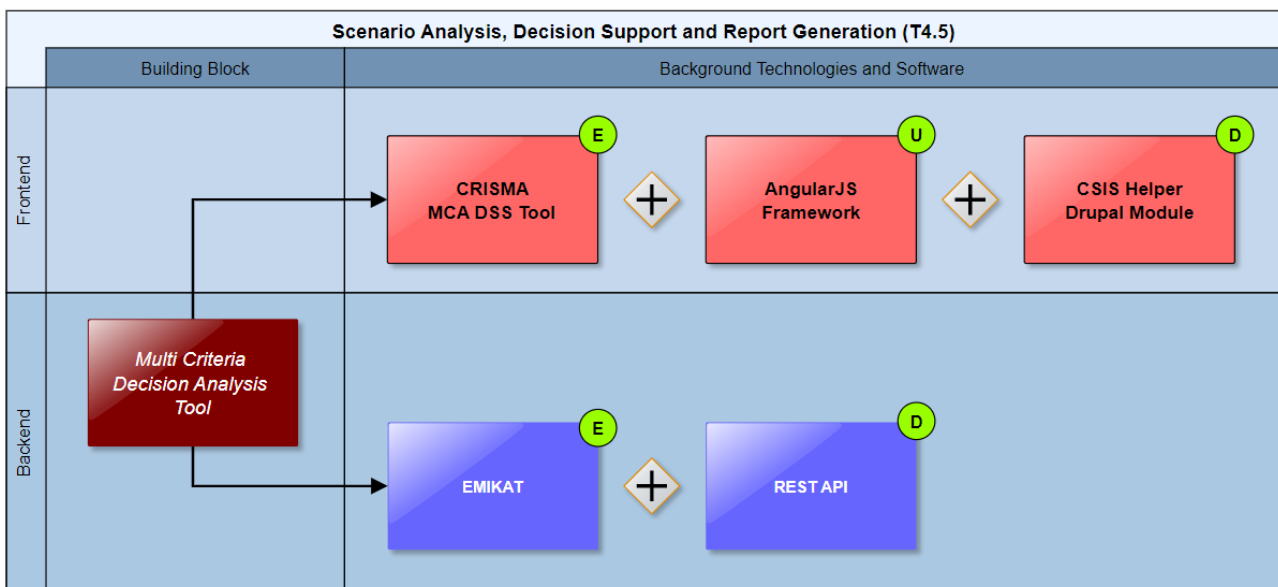


Figure 16: Multi Criteria Decision Analysis Tool Technology Support

The Multi Criteria Decision Analysis (MCDA) Tools (<https://github.com/crismaproject>) of the CRISMA project (<https://crisma-cat.ait.ac.at/>) are developed as open-source HTML5/AJAX widgets which makes them directly usable as generic ICT Climate Services. They rely on a simple standardized JSON (JavaScript Object Notation) data format which fits perfectly into CLARITY's data driven approach and, in combination with the RESTful API deployed on top of a PostgreSQL database, is also compatible with the Integration RDBMS backend of the Scenario Management and UI Integration Platform Building Blocks, respectively.

As the CRISMA tools have been implemented on basis of the AngularJS 1.0 JavaScript framework, it must be noted, that version 1.0 has been superseded by Angular 5.0 in 2017 and version 1.0 reached end of life in 2016. Although AngularJS 1.0 is still actively maintained by the community on GitHub (<https://github.com/angular/angular.js/>), no new plugins and libraries are being developed for version 1.0 and existing libraries and plugins don't receive new feature or security updates for AngularJS 1.0 in most cases.

Depending on additional functionality that is requested for the Multi Criteria Decision Analysis Tool, it might therefore be necessary to base further developments of the tool on more recent JavaScript Frameworks like Angular 6.0 or React 16.0.