HISEA DELIVERABLE 5.3

SERVICE TOOLS PROTOTYPE I

WORK PACKAGE NUMBER: 5

WORK PACKAGE TITLE: SERVICE IMPLEMENTATION

HISea Project Information	
Project full title	High Resolution Copernicus-Based Information Services at Sea for Ports and Aquaculture
Project acronym	HiSea
Grant agreement number	821934
Project coordinator	Dr. Ghada El Serafy
Project start date and duration	1st January, 2019, 30 months
Project website	https://hiseaproject.com/

Deliverable Information	
Work package number	5
Work package title	Service Implementation
Deliverable number	5.3
Deliverable title	Service Tools Prototype I
Description	This deliverable reports on the development of different tools that are meant to be used by the service users in order to tailor customized derived products adapted to their needs. At the most basic level, a toolset of statistical methods will be provided to extract means, limits, trends, patterns, etc. in order to deliver uniform, easy to use forecast information both to generic users and premium services alike. On top of this layer, a customizable engine will allow the combination of these outcomes under the form of more complex events, facilitating the composition of triggering rules for notifications and reports on meaningful events for the users. There will be two deliverables in total. This will be the first one.
Lead beneficiary	Ascora

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 821934

Lead Author(s)	Danny Pape
Contributor(s)	Daniel Wegmann
Revision number	5
Revision Date	28.02.2020
Status (Final (F), Draft (D), Revised Draft (RV))	F
Dissemination level (Public (PU), Restricted to other program participants (PP), Restricted to a group specified by the consortium (RE), Confidential for consortium members only (CO))	PU

Document History			
Revision	**Date**	**Modification**	**Author**
0.1	24.02.2020	Create Document Structure	Danny Pape
0.2	24.02.2020	Content for Section 3 and 4	Danny Pape
0.3	25.02.2020	Content for Section 1, 5 and 6	Danny Pape
0.4	25.02.2020	Internal Review	Daniel Wegmann
0.5	28.2.2020	Confirmation by management	Brache Ehrman

Approvals				
	Name	**Organisation**	**Date**	**Signature (initials)**
Coordinator	Ghada El Serafy	Deltares	28 Feb 2020	GES
WP Leaders	Danny Pape	Ascora	28 Feb 2020	DP

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 821934

Executive Summary

The deliverable 5.3 Service Tools Prototype I describes the HiSea Platform background components. The latest do not include any Graphical User Interface but are services working in the background to provide valuable input to the demonstrators. Except for the access to live data, all expected services are implemented and provided as docker containers. Live data could not be realized yet, because the DIAS functionalities have not met the expectations.

Until this stage the following backend components are implemented:

- A **Download Module** to configure recurring download jobs that provide the HiSea platform with actual data. This module consists of the following subcomponents:
    - The **downloader logic** for the authentication, authorization, connection to other sources, and download of data;
    - The **API** to provide the backend logic publicly;
    - The **database** to keep meta information permanent accessible;
- A **Timeseries service** to extract valuable information from NETCDF files;
- **Grid Service** to provide Web Map Services to HiSea demonstrators;
- **Alarm Forward Service** to communicate any triggered alarm to mobile demonstrators.

The next steps have been identified and are closely linked to the results of D3.5 Data Processing Algorithms. Mainly the pre- and postprocessing services must be adjusted in order to be compliant with Copernicus data and the DIAS environment.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 821934

Table of Contents

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 821934

1 Scope

This deliverable is ultimately about services that work in the background and are visually intangible to the user. The HiSea platform makes use of different data sources such as Copernicus Marine Environment Monitoring Data, local measurements and remotely sensing acquired data. In order for the platform to be able to properly “communicate” with these different data sources, several components are set up. The platform can be seen as a complex box able to acquire data , process input data and deliver data output. The data input cover all aspects of receiving and feeding data into the system from sources, such as Copernicus. This data then need to be processed and each source can provide data in different structures. It must then be harmonized, enriched or transformed into a HiSea compliant data format. Only then the data can be processed in the platform and finally presented to the user. GUI components are not handled in this deliverable. These can be found in the Deliverable 5.1 Web and Mobile Clients Prototype I.

2 Planned Activities

The first goal of task 5.23: Development and integration of a set of service tools were to create the most basic level of tools and structure aiming at having a running platform to be presented to users and be further improved after the users' feedback. The platform is built with a modular structure of the system architecture and the microservice approach, in such a way that it is possible to successively expand and customize the analytical processing functionalities based on users’ feedback. A robust base architecture has been implemented and service tools developed that are ready to extend successively. Table 1 shows that almost all planned objectives have been met.

Table 1: As-is and to-be analysis

Planned Goals	Achieved Goals
Reliable access to receive data from providers.	The data input to HiSea is currently performed with a Download service that enables the system user to schedule repetitive downloads from sources. A connection to Copernicus data at DIAS provider CREODIAS was established however, it turned out that the data are very outdated and therefore not suitable for the purpose of the HiSea project.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 821934

Conversion of the raw data into HiSea compliant data.	Currently, NETCDF files can be downloaded and stored. The access to these files can be performed with a time series service.
Permanent access to downloaded data.	Meta information about the downloaded NETCDF files are stored in a PostGreSQL database. The database entries contain the necessary information, such as the type and location of the NETCDF files.
Access to live-data	It is foreseen to use DIAS providers, such as WEKEO or CREODIAS, in order to get access to time-critical-data. The platform has been tested ed both on WEKEO and CREODIAS. Unfortunately, the provided data were either too old, or the connection to the DIAS environment could not be established.
Creation of APIs to provide data to demonstrators	An essential aspect of a microservice environment is the accessibility of the developed microservices. In HiSea, it was agreed to use RESTful APIs in order to enable reliable data communication. Therefore, all data transfer objects are provided via a public RESTful API, that can be accessed from the demonstrators, in order to process and visualise the data.

3 Progress

This section lists all components that have been developed and are categorised as HiSea service tools. The entire services were also successfully demonstrated at the 1st review meeting of the project in February 2020.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 821934

3.1 Downloader

The Download module enables the HiSea platform to download data, such as NETCDF files, from different sources. It contains three subcomponents running as Docker[1] containers orchestrated via Docker Compose[2]. Each container fulfils a different task.

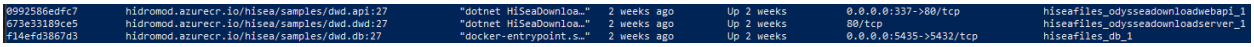

Figure 1: Downloader Module Containers

3.1.1 DWD.DWD

This subcomponent contains the complete logic to download data from user-specified sources. Internally, a set of different models are used to configure the download process. To configure the download process, the DWD.API (Section 3.1.2) must be used. After configuring a download process, this subcomponent takes care of the following:

- accessing the source including authentication and authorization,
- writing download details with enriched metadata into the PostgreSQL[3] database in DWD.DB (Section 3.1.3),
- triggering scheduled downloads automatically – depending on the configuration,
- storing files in downloaded files in a defined folder

Figure 3 lists all models that are processed internally in this component. Models that are going to be used externally to this subcomponent are extended with the abbreviation DTO (Data Transfer Object). To work with DTOs, the swagger documentation can be used to explore the properties (see Section 3.1.2). Figure 2 shows the model structure of a simple download data transfer object.

[1] https://www.docker.com/
[2] https://docs.docker.com/compose/
[3] https://www.postgresql.org/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 821934

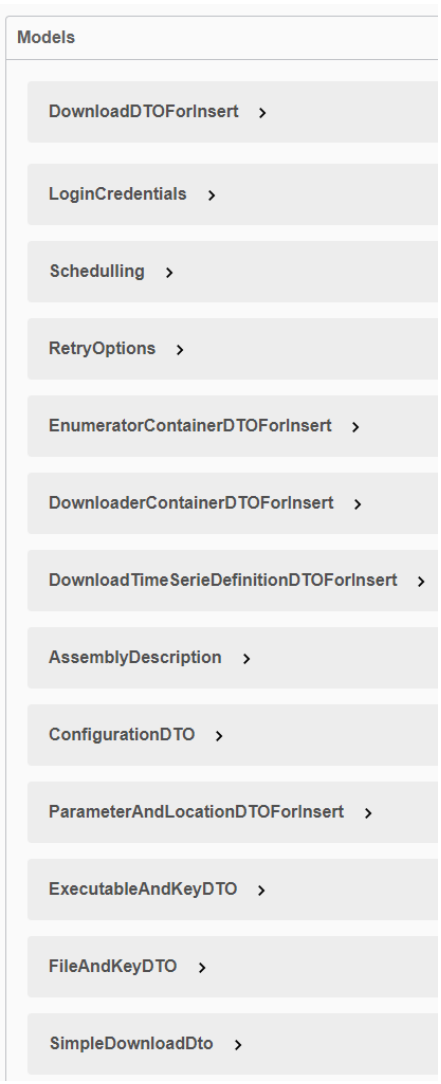

Figure 3: DWD.DWD Models

Figure 2: DWD.DWD Model Example

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 821934

3.1.2 DWD.API

This subcomponent provides public access to the functionality of DWD.DWD (see Section 3.1.1). The API is split into 3 categories. The first category “downloads” (Figure 4) provides different options to manage the downloads in general. The RESTful API provides three GET endpoints to retrieve information about finished downloads. A detailed response can be seen in Figure 5 There, the id can later be used within the Timeseries service, to extract information out of the downloaded files. The information retrieved from the GET endpoints is requested from the DWD.DB subcomponent (see Section 3.1.3). The first step to start a download is to create a download job by using the POST endpoints. The PUT endpoint is to update an existing download job, and the DELETE endpoint removes a download job.

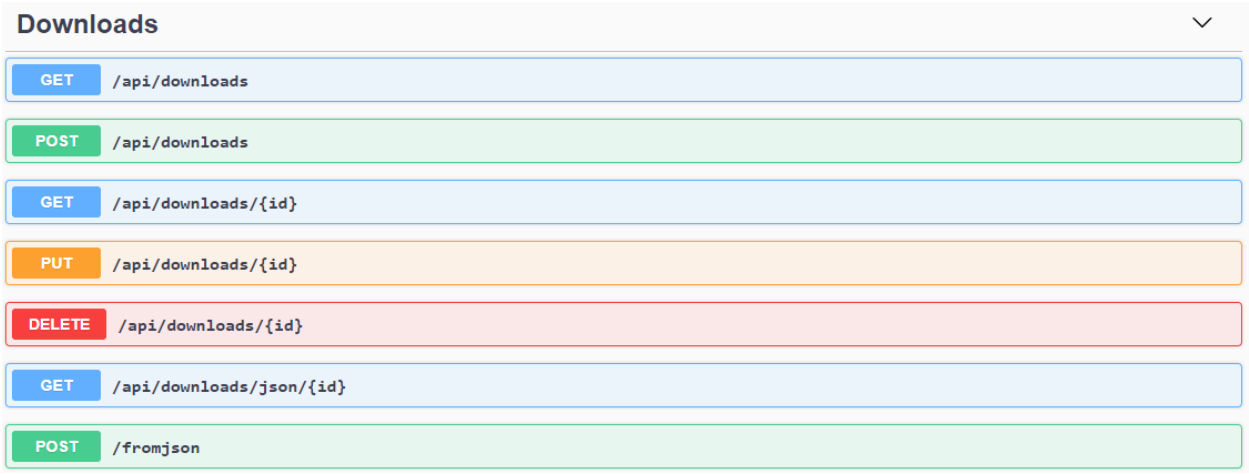

Figure 4: Swagger DWD.API Downloads

Response body


```
[
  {
    "id": 1,
    "name": "Download 6d5c057c-d0d0-4471-bdda-cab5678e03cf",
    "schedullingEnabled": true,
    "lastExecutionStartDate": "2020-02-05T10:34:07.256085",
    "lastExecutionEndDate": "2020-02-05T10:37:48.772528",
    "nextExecutionStart": "2020-03-01T00:00:00Z",
    "executionResult": 1
  }
]
```


Figure 5: GET Download Response

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 821934

After a download job is created, the next step is to schedule a download job (Figure 6). Four different PUT endpoints enable the user/developer to:

- **Execute** the download immediately;
- **Start** the recurring download in a defined interval (default settings 30 days);
- **stop** the scheduled plan of the download job. The download will not be triggered anymore;
- and **reschedule** the download job with different settings.

Figure 6: Swagger DWD.API Scheduling

To check if the schedule is working properly, the GET endpoints of the category "Executions" can be used to get detailed information about the download schedules. In addition to that, a Hangfire[4] Dashboard is set up, which enables to see the monitored status of a download job.

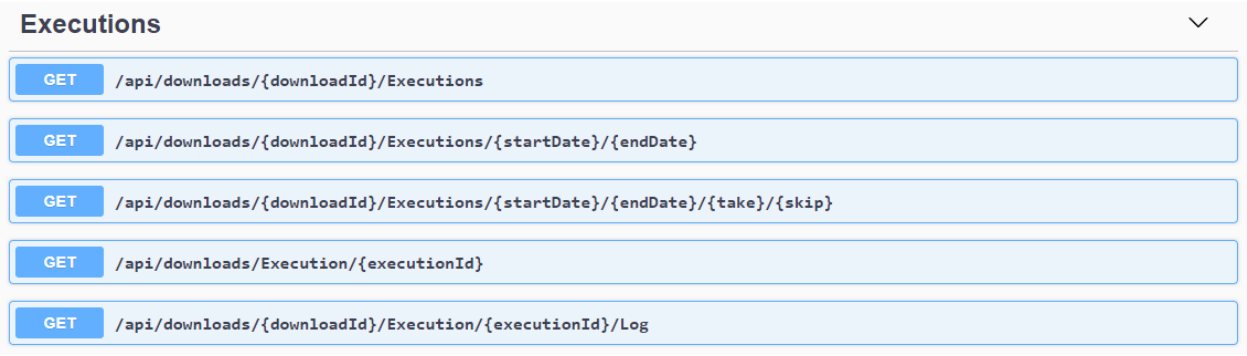

Figure 7: Swagger DWD.API Executions

[4] https://www.hangfire.io/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 821934

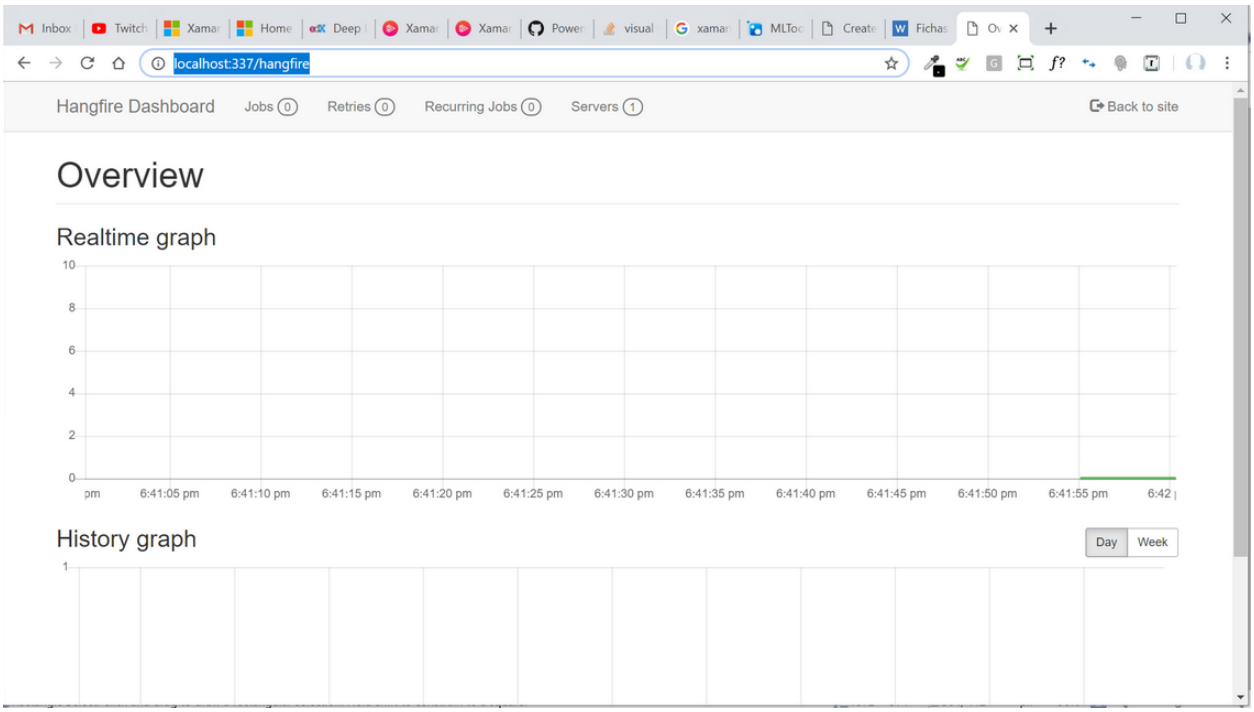

Figure 8: Hangfire Dashboard

3.1.3 DWD.DB

This subcomponent contains a PostgreSQL database to store relational data. Basically, all data used inside the download module are stored in this database, except the NETCDF files that are downloaded. These files are stored on the server in a user-defined folder. This database is not only used for the downloading component itself. It is also used to manage permanent data of the HiSea dashboard demonstrator, such as user notes or alarms.

3.2 Timeseries service

The time series service provides access via a RESTful API to previously downloaded NETCDF files. For this, two endpoints are provided publicly. The first GET endpoint to use is /api/Values/{serieID}/{propertyID}, where the relevant parameters serieID and propertyID can be retrieved beforehand from the DWD.API (Section 3.1.2). These GET endpoints extract information from NETCDF files and provide information in json format to the requesting party.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 821934

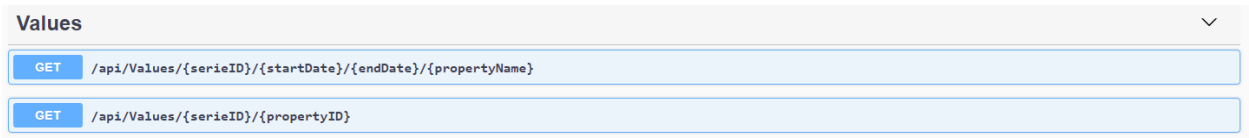

Figure 9: Swagger documentation for Timeseries API

Example Request: api/Values/nice2/ASLVZZ01

The related response contains a period in json format in which values are available.

Response:

```
{
  "startDate": "2020-02-25T10:53:41.5654195Z",
  "endDate": "2020-02-29T14:53:41.5654195Z"
}
```

This information can then be used to retrieve the sensor values enriched with a timestamp. Therefore, the subsequent request asks for sensor values:

/api/Values/nice2/2020-02-25T10%3A53%3A41.5654195Z/2020-02-29T14%3A53%3A41.5654195Z/ASLVZZ01

This request results in the following response in json format:

```
[
 {
   "date": "2020-02-25T10:54:16.4448276Z",
   "value": 0
 },
 {
   "date": "2020-02-25T11:54:16.4448276Z",
   "value": 1
 },
 {...},
{
   "date": "2020-02-29T12:54:16.4448276Z",
   "value": 98
 },
 {
   "date": "2020-02-29T13:54:16.4448276Z",
   "value": 99
 }
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 821934

]

The visualisation can then be viewed at the HiSea dashboard demonstrator for example as a line chart.

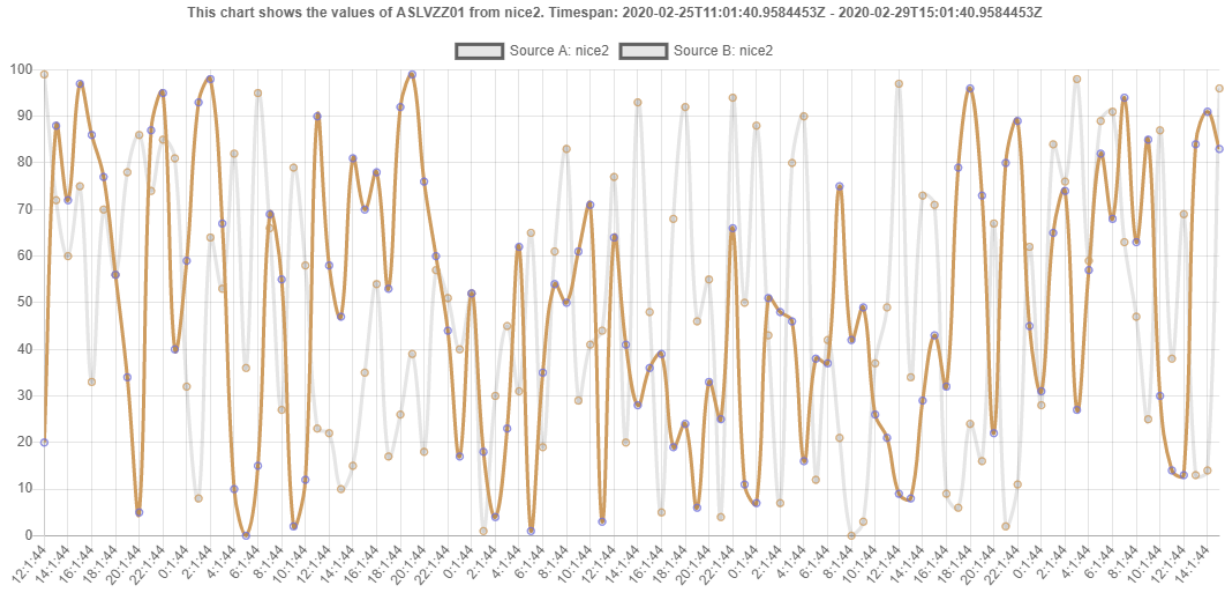

Figure 10: Example for data visualisation

3.3 Grid service

The grid service provides Web Map Services (WMS) to the HiSea demonstrators. Currently, this service provides the demonstrators with URLs to specific Web Map Services. With those WMS URLs additional map layers can be bound to a base map to look like Figure 11. For different sensor types, a selection can be made by the user. But only one selection can be made at time.

The base URL for the WMS is: http://hades.hidromod.com:8085/geoserver/ForecastLargeScale/wms?

In order to get data for specific sensor, the URL must be extended with:

- *5km_world_wp_WaterTemp* for world-wide water temperature
- *5km_world_waves_Vector* for world-wide vectors of waves
- *5km_world_waves_raster* for a world-wide raster of waves
- *5km_world_wp_Salinity* for salinity visualizations world-wide

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 821934

Figure 11: WMS Example from Grid Service

3.4 Alarm Forward Service

In the HiSea Dashboard demonstrator, alarms are set if a defined threshold is reached. In order to communicate this to mobile users a push notification has been implemented. As soon as an alarm is triggered, it will also be pushed to the mobile users.

4 Next Activities

The next activities are mainly dependent on the outcomes of D3.5 Data Processing Algorithms. D3.5 is not finished yet, but the different methodologies of pre- and postprocessing of data are the main part of the HiSea service tools. Therefore, the main task in the future for service tools is to adjust the current services in order to be compliant with Copernicus datasets and data retrieval via DIAS Providers, such as CREODIAS[5] or WEKEO[6]. Particular attention is paid to integrate existing services from the DIAS environment into the

[5] https://creodias.eu/
[6] https://www.wekeo.eu/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 821934

existing HiSea platform. With the help of these services, analyses are then created in order to generate forecasts, statistical evaluations, data comparisons, etc.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 821934