NET4mPLASTIC PROJECT

WP3 – Act. 3.3 Study and Design of the Integrated Platform's structure associated with early warning system

D 3.3.3

EWS Software Architecture design

March, 2020 - Version 1.2

Project Acronym	NET4mPLASTIC
Project ID Number	10046722
Project Title	New Technologies for macro and Microplastic Detection and Analysis in the Adriatic Basin
Priority Axis	3
Specific objective	3.3
Work Package Number	3
Work Package Title	Preliminary activities and project implementation
Activity Number	3.3
Activity Title	Study and Design of the Integrated Platform's structure associated with early warning system
Partner in Charge	PP3
Partners involved	LP, PP1, PP2, PP3, PP4, PP7
Status	Final
Distribution	Public

CONTRIBUTING PARTNERS	PP2, PP3, PP4

Data	Vers	Prep	Resp	Appr	Rev	Comment
31.10.2019	1.0	PP2 PP3 PP4	PP3	Daniele Calore	Draft	Comment and approval
31.12.2019	1.1	PP2 PP3 PP4	PP3	Daniele Calore	First Release	Comment and approval
15.03.2020	1.2	PP2 PP3 PP4	PP3	Daniele Calore	Final	Approved

INDEX

Acronyms / Abbreviations

ACRONYM	DEFINITION
EWS	Early Warning System
MP	Microplastic
OBU	On board Unit
PP	Project Plan
PT	Project team
TC	Technical task coordinator
TGS-ML	Technical Subgroup on Marine litter, European Union expert group On marine litter
TM	Task Manager
UML	Unified Modelling Language
WP	Workpackage

1 Introduction

1.1 Background of the project

The main goal of the NET4mPLASTIC project is to achieve an efficient monitoring system for plastic and MP distribution along the Croatian and Italian coastal and marine areas in order to improve the environmental coastal and marine sea quality conditions.

According to doc R1, the Act 3.3 deals with the study and design of the EWS - Early Warning System including:

- a control centre, based on system hardware and network (Prosoft), and a EWS application (Hydra Solutions) integrated with the transport model and external systems (such as the oceanographic model - (Marche Region);
- Integrated Marine Drone, for collection of MP - microplastic, and geolocalized water indicators on the route (Hydra Solutions);
- Integrated Marine OBU, a unit to be installed on board of ships for improved MP collection with geolocalized water indicators on the route (Hydra Solutions).

The design shall be carried out with the modern system engineering approach based on UML - Unified Modelling Language (Hydra Solutions). UNITS and RERA SD will provide data for the first set up of the platform related to MP. Based on this WP, the transport model will be developed in WP4. The development of the EWS platform integrated with the transport model will be done in WP5.
The expected deliverable are the following:

- D 3.3.1 – EWS Requirements definitions based on the stakeholders and users' needs, through questionnaires and specific meeting
- D 3.3.2 – EWS Hardware Architecture and network design (central Data Centre Hardware Architecture Client/Server, Data network architecture and related communication segments)
- D 3.3.3 – EWS Software Architecture design (data modelling software, GIS applications, early warning detection software, etc.), the Relational Database to manage all collected data with related meta data, the communication Front-End for web remote access, the Data Centre Software Interfaces for users
- D 3.3.4 – EWS Hardware and other software Components Specifications design (Integrated Marine Drone and Marine OBU, with details of required components (hardware and firmware), firmware and other software components (mobile apps for managing the drones and for remote mobile activities).
- D 3.3.5 - Report and database provision with all the collected data

1.2 Purpose of the report

This document is the deliverable D.3.3.3 - EWS Software architecture design, based on the EWS requirement definition report [R2], and the deliverable D.3.3.2 - EWS Hardware Architecture and network

design, to describe the software architecute design, within the activity 3.3 of the **Net4mPlastic project - New Technologies for Macro and Microplastic Detection and Analysis in the Adriatic Basin.**

The purpose of this document is summarized as follows:

- Description of the software architecture;
- Description of each software module;
- Interface description of external modules.

Reference documentation

No	Title	Rif/Report N.	Pubblished by
[R1]	**APPLICATION FORM - NET4mPLASTIC Project - New Technologies for macro and Microplastic Detection and Analysis in the Adriatic Basin** 2014 - 2020 Interreg V-A Italy - Croatia CBC Programme Call for proposal 2017 Standard - NET4mPLASTIC Priority Axis:Environment and cultural heritage	Application ID: 10046722, dated 30/06/2017	Lead applicant: UNIVERSITY OF FERRARA
[R2]	**EWS requirements definition**	HYD001-SPE-001.0	ACT3.3 – Net4Mplastic
[R3]	**EWS hardware architecture and network design**	HYD002-SPE-001.0	ACT3.3 – Net4Mplastic

2 Software architecture

2.1 Introduction

This chapter provides an overview of the software architecture expected for the EWS system, based on the system architecture.

2.2 Functional Diagram

Basically, the general architecture of the Software involved in the EWS is depicted in the picture herebelow where the following elements and modules can be identified:

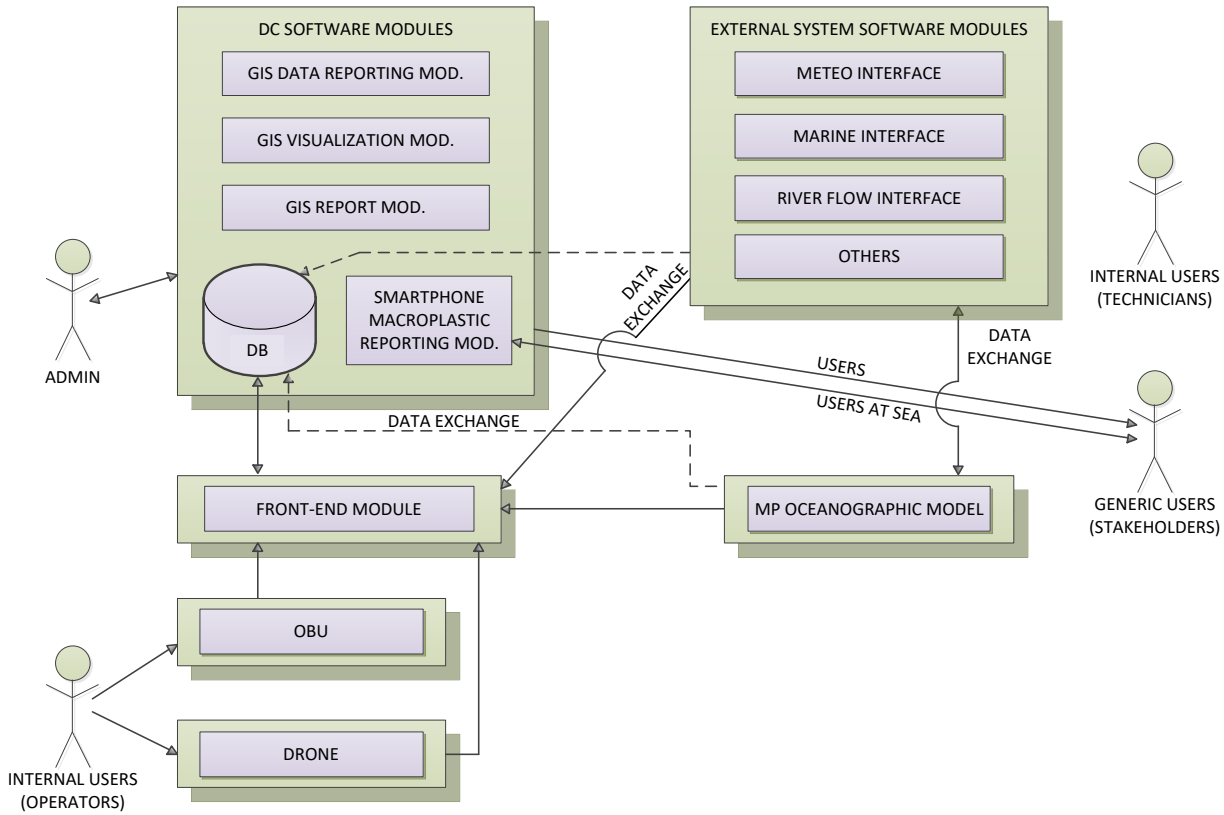

Fig. 2.1 – Software Architecture – Conceptual Deployment Diagram

- Data Centre software modules, to collect, process and store in an organized way and make available to the users all the collected data. It mainly consists of:
    - GIS Data Reporting Module;
    - GIS Data Visualization Module;
    - GIS Report Module;
    - GIS Database;

  - Smartphone App Module – Macroplastic Reporting App.

- Front-end software module, that could be both integrated in the Data Centre to acquire, decode and store some kind of data as same as it could be a portable software module, both for laptop or eventually tablet pc to acquire, decode and store/transmit from local source of data as the drone or the OBU;

- External system software modules, to collect all the necessary external source of data needed by the EWS and provide them directly to the Data Centre Database or eventually directly to the Prediction Model. It mainly consists of:

  - Meteo Interface Module;
  - Marine Interface Module;
  - River Flow Interface Module;
  - Others (such as Macroplastic data, etc).

- OBU software module, to acquire and collect all the data from the related sensor, store it on the internal OBU memory and prepare it for the automatic and/or manual transmission to the Data Centre Database;

- Drone software module, to acquire and collect all the data from the related sensor, store it on the internal Drone memory and prepare it for the automatic and/or manual transmission to the Data Centre Database.

The macro-list of software modules with their theoretic location of deployment is described in the table below.

ID	Software Module	Note
Data Centre		
DC01	GIS Data Reporting Module	
DC02	GIS Data Visualization Module	
DC03	GIS Report Module	
DC04	GIS Database	
DC05	GIS Front-End	
DC06	Smartphone Macroplastic Reporting Module (Data Centre Side)	
External System		

ES01	Meteo Interface	
ES02	Marine Interface	
ES03	River Flow Interface	
ES04	Others	
Prediction Model Environment		
PM01	MP Oceanographic Model	
Portable		
PO01	Front-end for local application/instrumentation	
PO02	Macroplastic Reporting App	
OBU		
OB01	OBU Internal Software Module	
DRONE		
DR01	Drone Internal Software Module	

Tab. 2.1 – Software Architecture – Macro Software Modules

2.3 Breakdown of Software Modules with Related Functionalities

Given the list of software modules with related location as described in the Tab. 2.1, the main functionalities to be implemented are coded and described herebelow:

Data Centre:

Module ID	Software Module Functionality	Note
DC01 – GIS Data Reporting Module		
DC01-F001	Retrieve desired data stored in the GIS Database and provide in to the user in table-view format	
DC01-F002	Provide to the user selectable query criteria such as date, time, location as same as search by parameters	
DC01-F003	Allow the user to export selected data in most pouplar format such as .xls or .csv	
DC01-F004	Keep track of the most important data of the system to be shown as always-on-display	
DC01-F005	Save for export functionality	
DC01-F006	Alarm configuration and dispatching options	To evaluate to integrate a separate software module
DC01-F007	User login/logout configuration	For administrator role, to evaluate to integrate a separate software module for administration
DC02 – GIS Data Visualization Module		
DC02-F001	Retrieve desired data stored in the GIS Database and provide in to the user in a map-view format	
DC02-F002	Provide to the user selectable query criteria such as date, time, location as same as search by parameters	
DC02-F003	Allow the user to export selected data in most pouplar format such as .jpg or .png	
DC02-F004	Keep track of the most important data of the system to be shown as always-on-display	

DC02-F005	Allow the user to perform all the typical GIS operation on the map such as zoom, group values by color, group by position, etc.	
DC02-F006	Save for export functionality	
DC02-F006	User login/logout	
DC03 – GIS Report Module		
DC03-F001	Allow the user to select data from the GIS Database, both in table-view and map-view format to generate a report (typically in .pdf format)	
DC03-F002	Report template configuration	
DC03-F003	Automatic report generation configuration	
DC03-F004	Report dispatching	
DC03-F005	User login/logout	
DC04 – GIS Database		
DC04-F001	Store received data following Relational Data Base Management System Standards	Directly accessible only to Administrator
DC04-F002	Provide Easy and Fast data retrieval	Directly accessible only to Administrator
DC04-F003	Ensure data atomicity	Directly accessible only to Administrator
DC04-F004	Ensure data consistency	Directly accessible only to Administrator
DC04-F005	Ensure data isolation	Directly accessible only to Administrator
DC04-F006	Ensure data durability	Directly accessible only to Administrator
DC04-F007	Easy commit and roll-back operations	Directly accessible only to Administrator
DC04-F008	Periodic backup and restore capability	Directly accessible only to Administrator
DC04-F009	User login/logout	Directly accessible only to Administrator
DC05 – GIS Front – End		

DC05-F001	Retrieve and parse all the different kind of data received, both manually and automatically, decode and store it in the GIS Database	
DC05-F002	Retrieve data from the GIS Database, to be encapsulated in the required format and transmit/export it when necessary.	
DC05-F003	First data validation	
DC06 – Smarpthone Macroplastic Reporting Module		
DC06-F001	Listen for data to be received from the Smartphone app module	
DC06-F002	Decode received data and store it on the GIS Database	
DC06-F003	Interface with other existent systems	To be evaluated

External System:

Module ID	**Software Module Functionality**	**Note**
ES01 – Meteo Interface		
ES01-F001	Acquire meteo data such as wind speed, wind direction, pressure, humidity, temperature and solar radiation (could be from sensors, websites, data sources, etc)	
ES01-F002	Encapsulate acquired data in the designed data format, to be acquired (via automatic transmission and/or manually) by DC05 – GIS Front-End software module	
ES01-F003	Discard corrupted or incomplete data (or mark it bad, incomplete, etc)	
ES01-F004	Keep log of the performed operation	
ES02 – Marine Interface		
ES02-F001	Acquire marine data such as significant wave height, direction and period, current profiles in terms of depth, current speed and current direction, water quality data in terms of conductivity, temperature, depth, dissolved oxygen and turbidity (could be from sensors, websites, data sources, etc.)	
ES02-F002	Encapsulate acquired data in the designed data format, to be acquired (via automatic transmission and/or manually) by DC05 – GIS Front-End software module	

ES02-F003	Discard corrupted or incomplete data (or mark it bad, incomplete, etc)	
ES02-F004	Keep log of the performed operation	
ES03 – River Flow Interface		
ES03-F001	Acquire river flow data (could be from sensors, websites, data sources, etc)	
ES03-F002	Encapsulate acquired data in the designed data format, to be acquired (via automatic transmission and/or manually) by DC05 – GIS Front-End software module	
ES03-F003	Discard corrupted or incomplete data (or mark it bad, incomplete, etc)	
ES03-F004	Keep log of the performed operation	
ES04 - Others		
ES04-F001	Acquire other kind of necessary data, such as bathymetry (could be from sensors, websites, data sources, etc)	To be detailed
ES04-F002	Encapsulate acquired data in the designed data format, to be acquired (via automatic transmission and/or manually) by DC05 – GIS Front-End software module	
ES04-F003	Discard corrupted or incomplete data (or mark it bad, incomplete, etc)	
ES04-F004	Keep log of the performed operation	

Prediction Model Environment: To be detailed by Marche Region

Portable:

Module ID	**Software Module Functionality**	**Note**
PO01 – Front-end for local application/instrumentation		
PO01-F001	Manually retrieve and parse all the different kind of data received, such as Drone stored data or OBU stored data and prepare/export it for the DC05-GIS Front-End software module	
PO01-F002	First data validation	
PO02 – Macroplastic Reporting App		
PO02-F001	Take picture and notes of the area of interest	

PO02-F002	Georeference the picture acquired, connect and send the data to DC06 Smarpthone Macroplastic Reporting Module	
PO02-F003	User login/logout to univocally identify the reporter	

OBU:

Module ID	**Software Module Functionality**	**Note**
OB01 – OBU Internal software module		
OB01-F001	Verify sensor integrity and acquire the diagnostic data of the unit	
OB01-F002	Acquire, store and sort the data from the sensor payload of the OBU	
OB01-F003	Validate the integrity of the data	
OB01-F004	Encapsulate data in the designed format and prepare it for the automatic transmission or manual download	

Drone:

Module ID	**Software Module Functionality**	**Note**
DR01 – Drone Internal software module		
DR01-F001	Verify sensor integrity and acquire the diagnostic data of the unit	
DR01-F002	Acquire, store and sort the data from the sensor payload of the Drone	
DR01-F003	Validate the integrity of the data	
DR01-F004	Encapsulate data in the designed format and prepare it for the automatic transmission or manual download	
DR01-F005	Route and mission program configuration	

3 Data Centre Software Modules

3.1 Introduction

This chapter provides an overview of the Data Centre Software Modules functionalities, based on the system architecture.

3.2 Web GIS platform data reporting module

As described in par. 2.3 of this document, Web GIS platform data reporting module shall satisfy and implement the macro functional requirements briefly reported herebelow:

DC01-F001	Retrieve desired data stored in the GIS Database and provide in to the user in table-view format
DC01-F002	Provide to the user selectable query criteria such as date, time, location as same as search by parameters
DC01-F003	Allow the user to export selected data in most pouplar format such as .xls or .csv
DC01-F004	Keep track of the most important data of the system to be shown as always-on-display
DC01-F005	Save for export functionality
DC01-F006	Alarm configuration and dispatching options
DC01-F007	User login/logout configuration

An procedural example of the required functionalities is depicted below, where it is shown an ideal software module that allow the user to select a certain time span, then select on another view a set of parameter to then exectute the query and visualize the output in a table format, allowing the user to eventually export the data in the most common formats:

1. The user shall choose the desired time interval of the data to be analyzed by picking the date and the hour on the date selection fields:

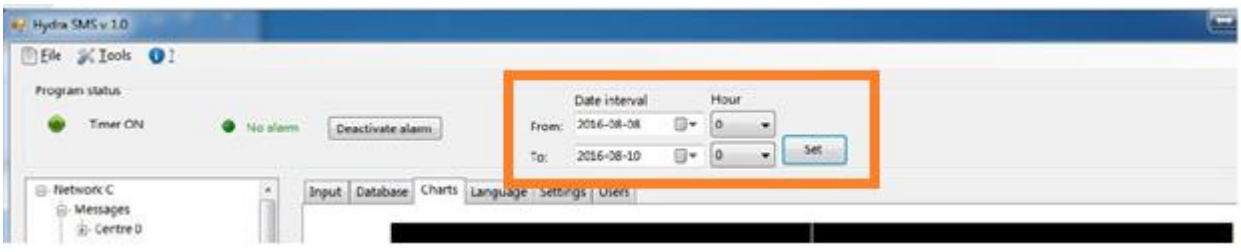

Fig. 3.1 – Date Selection

2. The user shall select on a tree-view or equivalent the parameter(s) to visualize:

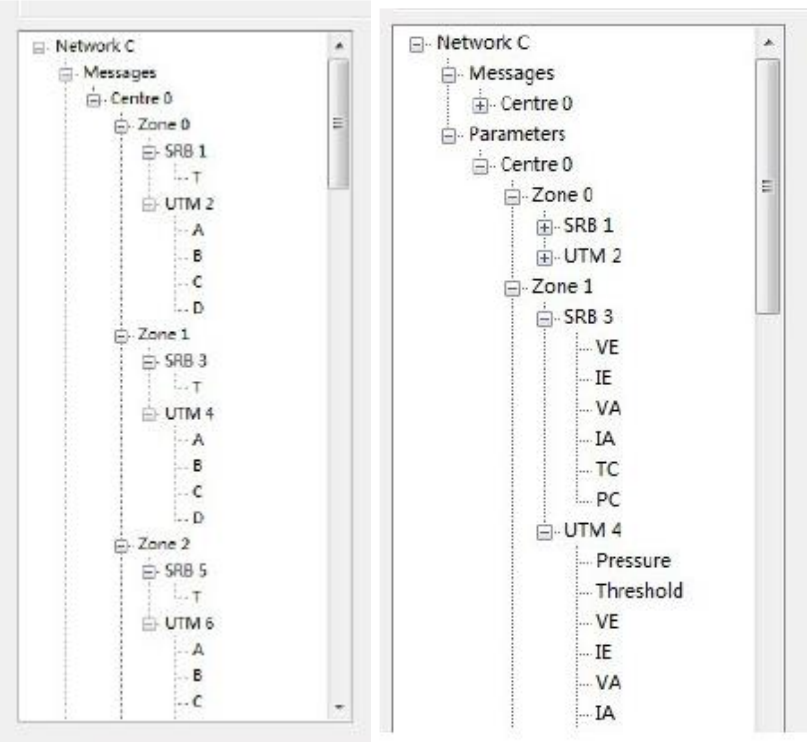

Fig. 3.2 – Different example of parameters selection

3 The user shall execute the query and the result shall appear in a table-view format

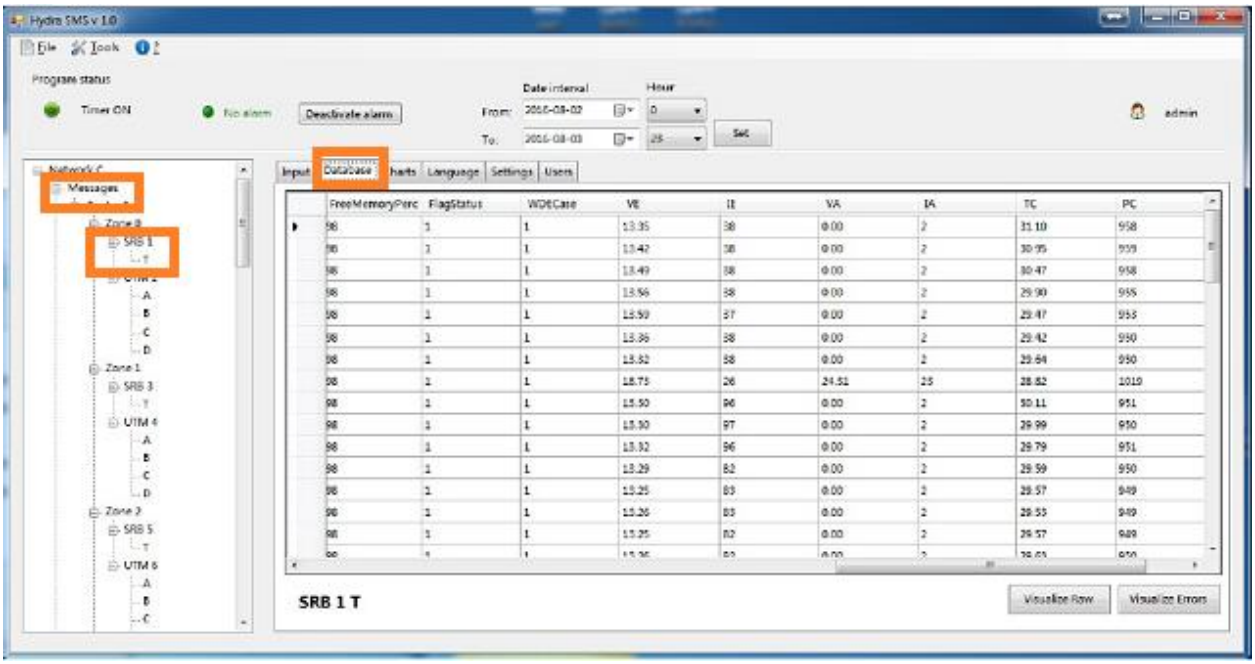

Fig 3.3 – Example of Table-view mode

3.3 Web GIS platform visualization module

As described in par. 2.3 of this document, Web GIS platform visualization module shall satisfy and implement the macro functional requirements briefly reported herebelow:

DC02-F001	Retrieve desired data stored in the GIS Database and provide in to the user in a map-view format
DC02-F002	Provide to the user selectable query criteria such as date, time, location as same as search by parameters
DC02-F003	Allow the user to export selected data in most pouplar format such as .jpg or .png
DC02-F004	Keep track of the most important data of the system to be shown as always-on-display
DC02-F005	Allow the user to perform all the typical GIS operation on the map such as zoom, group values by color, group by position, etc.
DC02-F006	Save for export functionality
DC02-F006	User login/logout

The steps required to achieve the required functionalities shall be similar to the ones described in the previous paragraph. The output in this case consist of a map (cartography) showing/plotting a series of data. By using the last state of the art technology, this software module will be implemented by the

integration of tools such as Google Earth® and Google Maps® API, as they can easily provide benefits such as:

1. Online cartography: the cartography is stored on external Google server and it is periodically updated by Google, without the needing of download unnecessary amount of data and/ore waste local storage to store it;
2. GIS functionalities integrated: it allows the user to easily integrate common GIS functionalities such as concentration views, rasterized views based on data concentration, etc;
3. Computational power: it is not required as all the elaboration to be done on the cartography are on charge of Google – server side. The interaction is depicted in the schema below.
4. Cost effective.

The interaction schema to integrate Google API is depicted herebelow:

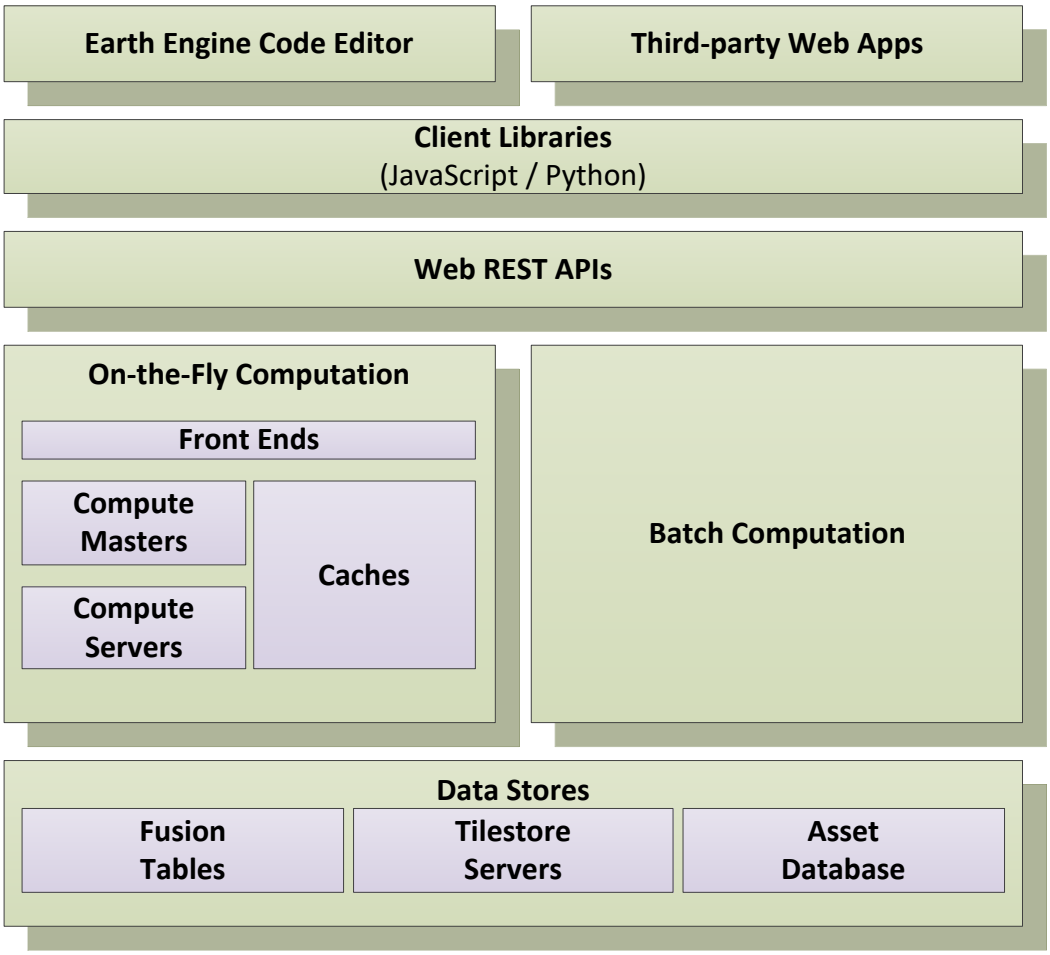

Fig 3.4 – Example of Interaction

The first three layers starting from above consist of integrating the API locally and prepare the queries by fetching data from local GIS Database and encapsulate it in the API required format, to then pass it to

Google Serves (last two layers) that will provide the interactive map and information after the data computation.

Some typical output examples are illustrated in the following pictures:

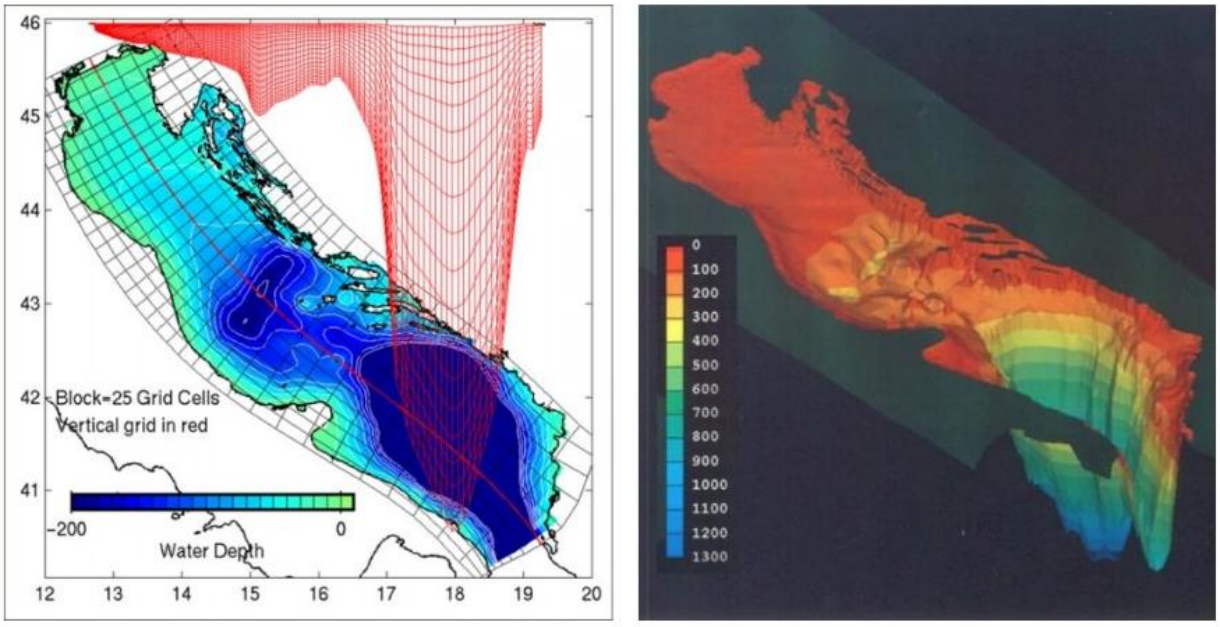

Fig 3.5 – Example of Raster View (2D and 3D) Bathymetry data

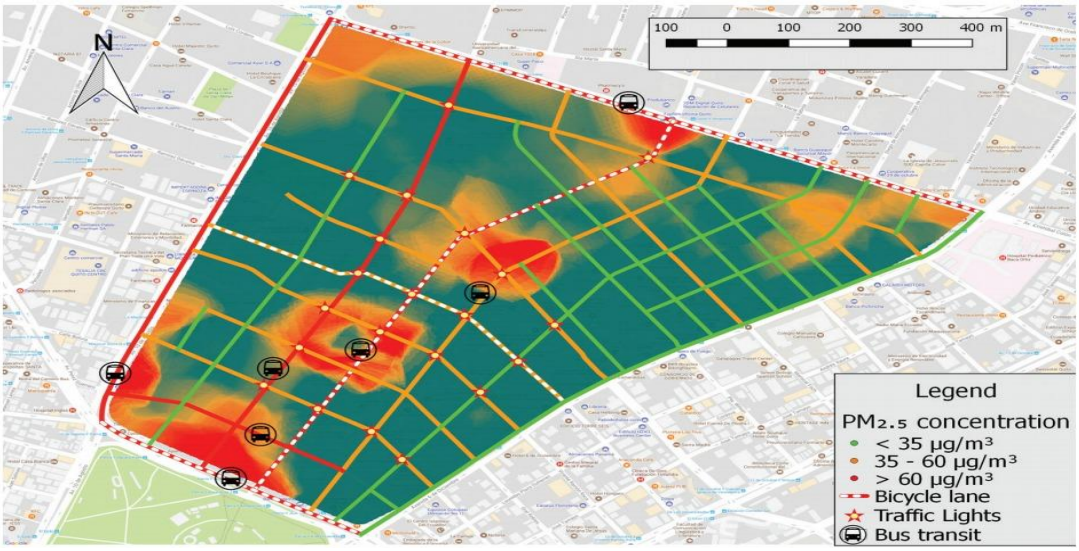

Fig 3.6 – Example of Raster View Pollution Concentration

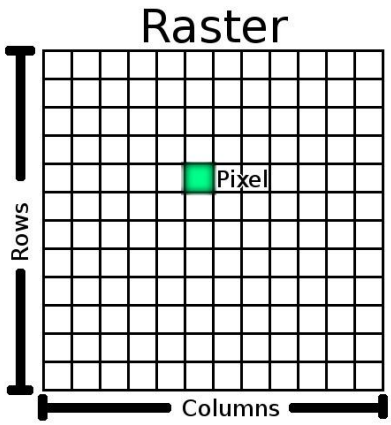

Fig. 3.7 – Raster Concept, where at every pixel is associated a color based on the range of desired data

3.4 Web GIS platform report module

As described in par. 2.3 of this document, Web GIS platform report module shall satisfy and implement the macro functional requirements briefly reported herebelow:

DC03-F001	Allow the user to select data from the GIS Database, both in table-view and map-view format to generate a report (typically in .pdf format)
DC03-F002	Report template configuration
DC03-F003	Automatic report generation configuration
DC03-F004	Report dispatching
DC03-F005	User login/logout

The steps needed to achieve the functionalities consist of creating a reporting selection criteria screen where the user can configure the desired data to be reported, both in manual way or in an automatic – periodic way. The report can be eventually automatically dispatched.

3.5 Database module

As described in par. 2.3 of this document, Database module shall satisfy and implement the macro functional requirements briefly reported herebelow:

DC04-F001	Store received data following Relational Data Base Management System Standards
DC04-F002	Provide Easy and Fast data retrieval

DC04-F003	Ensure data atomicity
DC04-F004	Ensure data consistency
DC04-F005	Ensure data isolation
DC04-F006	Ensure data durability
DC04-F007	Easy commit and roll-back operations
DC04-F008	Periodic backup and restore capability
DC04-F009	User login/logout

Those functionalities are achieved by integrating the well-known Relational Database Management System. Detailed information is provided within the related document D.3.3.5 Net4mPlastic project - EWS report and database provision.

3.6 Front-end module

As described in par. 2.3 of this document, Front-end module shall satisfy and implement the macro functional requirements briefly reported herebelow:

DC05-F001	Retrieve and parse all the different kind of data received, both manually and automatically, decode and store it in the GIS Database
DC05-F002	Retrieve data from the GIS Database, to be encapsulated in the required format and transmit/export it when necessary.
DC05-F003	First data validation

Fron-end software module shall be the agent responsible of acquire and parse all the data coming from the external environment and converging to the main GIS Database. At the actual state the role of the front-end is:

- Acquire data (automatically or manually provided) from the Drone;
- Acquire data (automatically or manually provided) from the OBU;
- Acquire data (automatically or manually provided) from the External Interfaces;
- Acquire data (automatically or manually provided) by the Oceanographic Module.

This is achieved by:

- Coding all the data formats;
- Integrating the data formats parsing schema in the front-end;
- Periodically poll for new data received in a particular folder, scan, parse and decode it by then preparing the SQL instruction to populate the GIS Database table with the information.

An event log of all the information passing thru the front-end software shall be kept and ready for the technicians to be visualized. Herebelow is depicted an example of Front-end side Error Log.

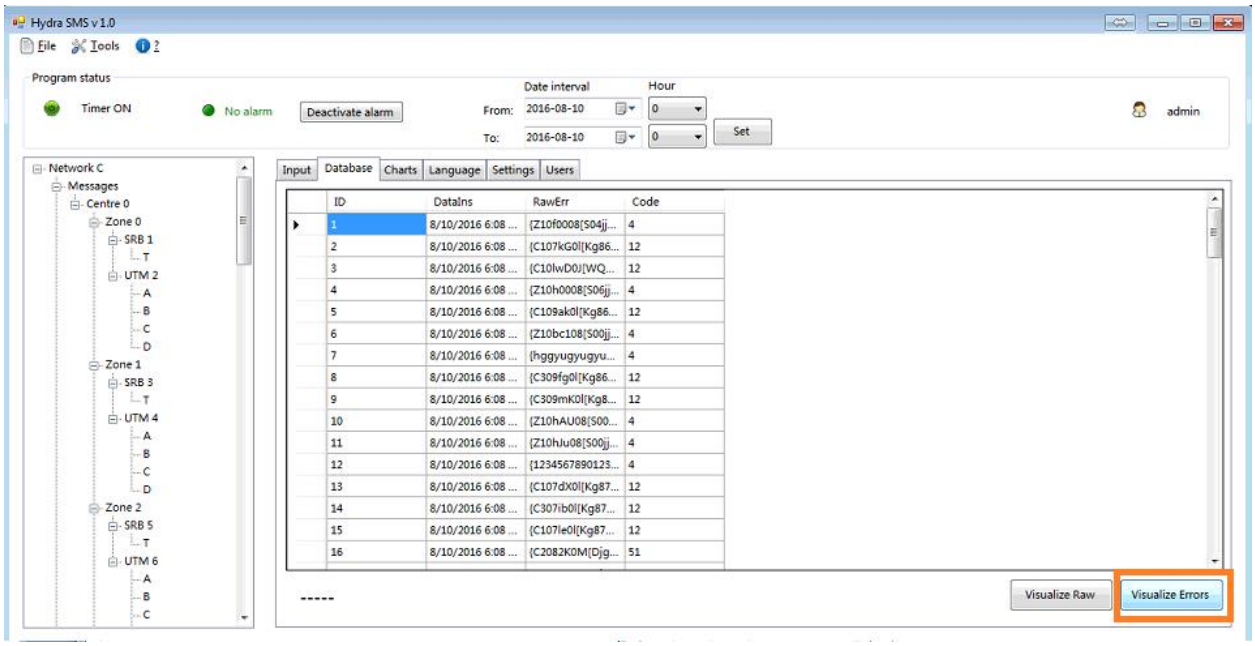

Fig. 3.8 – Error Log

3.7 Smartphone software application for Macroplastic data reporting

As described in par. 2.3 of this document, Smartphone software application for microplastic data reporting module shall satisfy and implement the macro functional requirements briefly reported herebelow:

DC06-F001	Listen for data to be received from the Smartphone app module
DC06-F002	Decode received data and store it on the GIS Database
DC06-F003	Interface with other existent systems

This software module can run as service and consist of a listener for microplastic data reporting app. Its scope is self-explanatory by the functionalities required.

4 Instrumented Units Software Modules

4.1 Introduction

This chapter provides an overview of the Instrumented Units Software Modules functionalities, based on the system architecture.

4.2 Drone Software Modules

As described in par. 2.3 of this document, Drone software module shall satisfy and implement the macro functional requirements briefly reported herebelow:

DR01-F001	Verify sensor integrity and acquire the diagnostic data of the unit
DR01-F002	Acquire, store and sort the data from the sensor payload of the Drone
DR01-F003	Validate the integrity of the data
DR01-F004	Encapsulate data in the designed format and prepare it for the automatic transmission or manual download
DR01-F005	Route and mission program configuration

The Drone Software Module shall produce the acquired data in the following meta-data string format:

{SoM[TIMESTAMP<Date,Time>,POSITION<Latitude,Longitude>,DIAGNOSTIC<V,A,T>,CTDPROBE<C,T,D,DO,Tu>,ADCP<Speed_01,Direction_01:Speed_50,Direction_50>,MPSENSORBINARY<ProprietaryData>,CRC]EoM}

Where:

Meta Value	**Sub Value**	**Description**
SoM	Start Of Message (including an identifier – Drone)	
TIMESTAMP	Date	Current date of sampling
	Time	Current time of sampling
POSITION	Latitude	Aquired GPS Latitude
	Longitude	Acquired GPS Longitude
DIAGNOSTIC	V	Internal battery voltage
	A	Current sink
	T	Internal electronic housing temperature

CTDPROBE	C	Acquired Conductivity from CTD Probe
	T	Acquired Temperature from CTD Probe
	D	Acquired Pressure from CTD Probe
	DO	Acquired Dissolved Oxygen from CTD Probe
	To	Acquired Turbidity from CTD Probe
ADCP	Speed_01 : 50	Acquired Sea Current Speed, 50 values, one each cell of depth
	Direction_01 : 50	Acquired Sea Current Direction, 50 values, one each cell of depth
MPSENSORBINARY	ProprietaryData	Binary File acquired from the innovative microplastic sensor, to be post-processed in the Data Centre
CRC	Cyclic Redundancy Check	
EoM	End of Message	

4.3 Onboard Units Software Modules

As described in par. 2.3 of this document, OBU software module shall satisfy and implement the macro functional requirements briefly reported herebelow:

OB01-F001	Verify sensor integrity and acquire the diagnostic data of the unit
OB01-F002	Acquire, store and sort the data from the sensor payload of the OBU
OB01-F003	Validate the integrity of the data
OB01-F004	Encapsulate data in the designed format and prepare it for the automatic transmission or manual download

The OBU Software Module shall produce the acquired data in the following meta-data string format:

{SoM[TIMESTAMP<Date,Time>,POSITION<Latitude,Longitude>,DIAGNOSTIC<V,A,T>,CTDPROBE<C,T,D,DO,Tu>,ADCP<Speed_01,Direction_01:Speed_50,Direction_50>,MPSENSORBINARY<ProprietaryData>,CRC]EoM}

Where:

Meta Value	Sub Value	Description
SoM	Start Of Message (Including an identifier - OBU)	

TIMESTAMP	Date	Current date of sampling
	Time	Current time of sampling
POSITION	Latitude	Aquired GPS Latitude
	Longitude	Acquired GPS Longitude
DIAGNOSTIC	V	Internal battery voltage
	A	Current sink
	T	Internal electronic housing temperature
CTDPROBE	C	Acquired Conductivity from CTD Probe
	T	Acquired Temperature from CTD Probe
	D	Acquired Pressure from CTD Probe
	DO	Acquired Dissolved Oxygen from CTD Probe
	To	Acquired Turbidity from CTD Probe
ADCP	Speed_01 : 50	Acquired Sea Current Speed, 50 values, one each cell of depth
	Direction_01 : 50	Acquired Sea Current Direction, 50 values, one each cell of depth
MPSENSORBINARY	ProprietaryData	Binary File acquired from the innovative microplastic sensor, to be post-processed in the Data Centre
CRC	Cyclic Redundancy Check	
EoM	End of Message	

5 External Systems Software Modules

5.1 Introduction

This chapter provides an overview of the External System Software Modules functionalities, based on the system architecture.

5.2 Meteo System

As described in par. 2.3 of this document, Meteo system software module shall satisfy and implement the macro functional requirements briefly reported herebelow:

ES01-F001	Acquire meteo data such as wind speed, wind direction, pressure, humidity, temperature and solar radiation (Mainly from Marche Region Model, could be from sensors, websites, data sources, etc if available)
ES01-F002	Encapsulate acquired data in the designed data format, to be acquired (via automatic transmission and/or manually) by DC05 – GIS Front-End software module
ES01-F003	Discard corrupted or incomplete data (or mark it bad, incomplete, etc)
ES01-F004	Keep log of the performed operation

The Meteo System Software Module shall produce the acquired data in the following meta-data string format:

{SoM[TIMESTAMP<Date,Time>,POSITION<Latitude,Longitude>,METEO<Ws,Wd,P,H,T,Sr>,CRC]EoM}

Where:

Meta Value	Sub Value	Description
SoM	Start Of Message (Including an identifier – Meteo System)	
TIMESTAMP	Date	Current date of sampling
	Time	Current time of sampling
POSITION	Latitude	Aquired GPS Latitude
	Longitude	Acquired GPS Longitude
METEO	Ws	Acquired Wind Speed
	Wd	Acquired Wind Direction
	P	Acquired Pressure

	H	Acquired Humidity
	T	Acquired Temperature
	Sr	Acquired Solar Radiation
CRC	Cyclic Redundancy Check	
EoM	End of Message	

5.3 Marine System

As described in par. 2.3 of this document, Marine software module shall satisfy and implement the macro functional requirements briefly reported herebelow:

ES02-F001	Acquire marine data such as significant wave height, direction and period, current profiles in terms of depth, current speed and current direction, water quality data in terms of conductivity, temperature, depth, dissolved oxygen and turbidity (Mainly from Marche Region Model, could be from sensors, websites, data sources, etc. if available)
ES02-F002	Encapsulate acquired data in the designed data format, to be acquired (via automatic transmission and/or manually) by DC05 – GIS Front-End software module
ES02-F003	Discard corrupted or incomplete data (or mark it bad, incomplete, etc)
ES02-F004	Keep log of the performed operation

The Marine System Software Module shall produce the acquired data in the following meta-data string format:

{SoM[TIMESTAMP<Date,Time>,POSITION<Latitude,Longitude>,WAVE<Wsh,Wd,Wp>,CTDPROBE<C,T,D,DO,Tu>,ADCP<Speed,Direction>, CRC]EoM}

Where:

Meta Value	**Sub Value**	**Description**
SoM	Start Of Message (Including an identifier – Marine System)	
TIMESTAMP	Date	Current date of sampling
	Time	Current time of sampling
POSITION (if available, eventually one for each instrument)	Latitude	Aquired GPS Latitude
	Longitude	Acquired GPS Longitude
WAVE (if available)	Wsh	Acquired Significant Wave Height

	Wd	Acquired Wave Direction
	Wp	Acquired Wave Period
CTDPROBE (if available)	C	Acquired Conductivity from CTD Probe
	T	Acquired Temperature from CTD Probe
	D	Acquired Pressure from CTD Probe
	DO	Acquired Dissolved Oxygen from CTD Probe
	To	Acquired Turbidity from CTD Probe
ADCP (if available, number of cells can vary)	Speed	Acquired Sea Current Cell Speed
	Direction	Acquired Sea Current Cell Direction
CRC	Cyclic Redundancy Check	
EoM	End of Message	

5.4 River flow System

As described in par. 2.3 of this document, River flow software module shall satisfy and implement the macro functional requirements briefly reported herebelow:

ES03-F001	Acquire river flow data (Mainly from Marche Region Model, but could be from sensors, websites, data sources, etc. if available)
ES03-F002	Encapsulate acquired data in the designed data format, to be acquired (via automatic transmission and/or manually) by DC05 – GIS Front-End software module
ES03-F003	Discard corrupted or incomplete data (or mark it bad, incomplete, etc)
ES03-F004	Keep log of the performed operation

The River Flow System Software Module shall produce the acquired data in the following meta-data string format:

{SoM[TIMESTAMP<Date,Time>,POSITION<Latitude,Longitude>,FLOW<f>, CRC]EoM}

Where:

Meta Value	Sub Value	Description
SoM	Start Of Message (Including an identifier – Marine System)	
TIMESTAMP	Date	Current date of sampling
	Time	Current time of sampling
POSITION (if available, eventually one for each instrument)	Latitude	Aquired GPS Latitude
	Longitude	Acquired GPS Longitude
FLOW	F	Acquired River Flow
CRC	Cyclic Redundancy Check	
EoM	End of Message	

5.5 Other System

As described in par. 2.3 of this document a general "Other Sensor" software module shall satisfy and implement the macro functional requirements briefly reported herebelow:

ES04-F001	Acquire other kind of necessary data, such as bathymetry (could be from sensors, websites, data sources, etc)
ES04-F002	Encapsulate acquired data in the designed data format, to be acquired (via automatic transmission and/or manually) by DC05 – GIS Front-End software module
ES04-F003	Discard corrupted or incomplete data (or mark it bad, incomplete, etc)
ES04-F004	Keep log of the performed operation

The following meta-data string format represent a possible encapsulation of bathymetry data:

{SoM[TIMESTAMP<Date,Time>,BATHYMETRY<XYZ>, CRC]EoM}

Where:

Meta Value	Sub Value	Description
SoM	Start Of Message (Including an identifier – Marine System)	
TIMESTAMP	Date	Current date of sampling
	Time	Current time of sampling

BATHYMETRY	XYZ	Encapsulated typical XYZ ASCII bathymetry format
CRC	Cyclic Redundancy Check	
EoM	End of Message	

6 External System Interface specification

6.1 Introduction

This chapter provides an overview on the interface specification, based on the system architecture.

6.2 MP Transport module interface specification

MP Transport software module shall recover the necessary data from the website where all the information retrieved by the phone app will be sent and stored on. MP Transport software module interface shall be handled in the following way:

- MP Portable application shall forward retrieved data to a dedicated website and/or to the main Server DBMS;
- GIS Front-End software module shall acquire the data from the dedicated website if not directly stored in the Server DBMS;
- Information shall be displayed on the GIS Data Visualization Module.

6.3 Meteo System interface specification

Meteo System software module shall recover the necessary data from the Marche Region Model and it shall be handled in the following way:

- Marche Region Model will periodically produce a set of decimated Meteo data to be made available thru datafile and/or its proprietary DBMS, by using an interchange dedicated table to be shared with Meteo System software module;
- Meteo System software module shall periodically acquire the data and produce the standardized output format described in Chapter 5, to be acquired by GIS Front-end software module that will store the decoded data in the Data Centre DBMS.

6.4 Marine System interface specification

Marine System software module shall recover the necessary data from the Marche Region Model and it shall be handled in the following way:

- Marche Region Model will periodically produce a set of decimated Marine data to be made available thru datafile and/or its proprietary DBMS, by using an interchange dedicated table to be shared with Marine System software module;
- Marine System software module shall periodically acquire the data and produce the standardized output format described in Chapter 5, to be acquired by GIS Front-end software module that will store the decoded data in the Data Centre DBMS.

6.5 River flow System interface specification

River flow System software module shall recover the necessary data from the Marche Region Model and it shall be handled in the following way:

- Marche Region Model will periodically produce a set of decimated River flow data to be made available thru datafile and/or its proprietary DBMS, by using an interchange dedicated table to be shared with River flow System software module;
- River flow System software module shall periodically acquire the data and produce the standardized output format described in Chapter 5, to be acquired by GIS Front-end software module that will store the decoded data in the Data Centre DBMS.