FRONT PAGE

PROJECT FINAL REPORT

Grant Agreement number: 286533

Project acronym: GEEWHEZ

Project title: manaGEmEnt and leisure middleWare for tHemE parks and Zoo

Funding Scheme: Research for the benefit of specific groups

Period covered: from 01/10/2011 to 30/09/2013

Name of the scientific representative of the project's co-ordinator[1], Title and Organisation: Claudio Di Capua, Executive Director of Zoomarine Italia SpA

Tel: +39 06 91533

Fax: +39 06 91533 294

E-mail: cdicapua@zoomarine.eu

Project website Errore. Il segnalibro non è definito. **address: www.geewhez.eu**

[1] Usually the contact person of the coordinator as specified in Art. 8.1. of the Grant Agreement.

Table of Contents:

List of Figures

List of tables:

1 Executive summary

GEEWHEZ is a multipurpose distributed platform specifically tailored over the specific requests of organisations like zoos and parks. The personnel hired by those organisations are people with skills in biology, chemistry and only few of them are administrative and no ones are engineers or computer scientists.

Zoos and Parks, even if not composed by engineers needs to face the same issues concerning other SMEs composed by computer skilled personnel and they have to use tools like ERP, information delivery system, security systems and other control tools that requires particular know how about such tools.

GEEWHEZ consortium developed a system able to simplify those activities reducing the complexity of platform like SAP and providing simple interfaces in order to manage, by a single point of access, the entire zoo/park.

Developing a middleware able to collect information from different sources like water monitoring, surveillance, information delivery and management systems, GEEWHEZ platform could collect all the information of the park/zoo and send them to the interested users and also merge them in order to provide new information e.g. people moving inside the dolphins' area produce an alert of the system but, checking the presence of those persons with the calendar of the ERP, GEEWHEZ will not send the alert because those persons are personnel of the zoo that have to work with the animals.

GEEWHEZ is a new system able to treat the different type of information of SME like zoos and parks publishing them and improving them with other data.

2 Project context and objectives

Currently a lot of solutions are available to manage and promote SMEs business. These tools, despite being complete and complex are not usable by everybody. If an organisation wants to install and use them it needs to provide, to its employees, particular training sessions able to explain them how they can work with those tools and how they can adapt themselves to them. This is generally translated into a huge effort in charge of the organization itself and in charge of the people involved in this training that have, in any case, to assure their normal activities.

If the organization that wants to implement such solution is a micro enterprise and it does not have people able to do such activities, especially if those people are completely extraneous to the nature of the solution, it generally implies the outsourcing of the activity to another organization that can offer the service and its maintenance.

In both cases the organization has to spend money to solve the issue and in both cases the organization will not have a complete outlook of the problem, e.g. the causes and the effects, but only that there is a problem, where it comes from and that it has to be solved. It happens because an organization is formed by different departments that are not correlated among them even if they are strictly connected. GEEWHEZ develops an innovative system able to share data among the several departments forming the organization, specializing each interface for each department but at the same time providing the overall information able to manage and promptly solve problems affecting the overall infrastructure.

The main objectives of GEEWHEZ project are:

- **Create specialised solutions**. Each GEEWHEZ solution has been designed and realised in order to work independently from each others in order to assure an inner security and safety level that it is not related to the one requested by the others. This choice has been adopted in particular to avoid any safety level violation related to the water monitoring system

- **Promoting data correlation among different add-ons**. Even if each single add-on may work as an island, the middleware, assures the cooperation among the add-ons able to produce new results and new evaluations inside the zoo/park

Albeit nowadays the “divide et imperat” paradigm is applied in many occasions to simplify problem solving in SMEs, with GEEWHEZ we moved into a more “social” representation of a SME as a facilitator in the decision making process when several areas of the SME are involved. By “social” we want to underline the possibility to share information among the different departments forming a SME; but this information sharing mechanisms can only be efficiently implemented if people are provided with easy-to-use tools relying on the expertise of each user.

The GEEWHEZ platform is formed by 4 different specialised solutions:

- **Video surveillance system**

  The video surveillance system monitors the access to different areas of the park and controls the security of the park. Applications such as the detection of the flow of visitors and the monitoring of waiting time of visitors at certain areas are important capabilities of the video surveillance system.

  Another important application of this system is the wildlife monitoring, such as the control of the flow and occupancy of animals in a given enclosure (to obtain data about their behaviour) and the detection of escaping animals or intrusion at their enclosures.

A video surveillance system developed in this way, not only a security system but also an animal behaviour monitoring system can provide health information on the species living in the park/zoo. Nowadays to check how animals behave during the day and the night zookeepers periodically check the enclosure, now GEEWHEZ will allow to continuous control the animal even if humans are not present and the system will report if during the controlling session strange behaviours have been detected

- **Water monitoring system**

  In zoos and parks hosting marine animals the pools of those animals have to be continuously checked and monitored in order to assure the safety of the animals; to measure the parameters affecting the life of those animals each pool is generally equipped with a system of filters and tanks placed under the pools. Normally the analysis are manually implemented collecting the water in phials and then sent to a laboratory to make the analysis. To optimize the analysis on the water of the pool and to increase the efficiency of the system, on each tank of the filter room the GEEWHEZ consortium mounted a panel already equipped with all the tools necessary to collect the water and make the analysis. In this way the system has to be only properly programmed, indicating the time and the type of the analysis, and it performs the measures as indicated

- **Leisure system**

  The leisure system is the add-on of GEEWHEZ platform in charge of data delivery both for visitors and zookeepers. The GEEWHEZ leisure system assures a peer to peer communication between the server and the overall smartphones of both users. The peer to peer communication has been assured by the integration of the OpenFire server inside the GEEWHEZ infrastructure, it assures that no entities in the network act as a master and no one acts as a slave but all the elements are considered as peers.

  Completely merged with the middleware the leisure system use the data published by the other add-ons to allow zookeeper and visitors to be informed on the status of the zoo/park

- **ERP**

  The GEEWHEZ ERP is a combination of inter-related modules centralizing business data and procedures, capturing the day-to-day business information exchanged with customers, providers, financial institutions, public administrations and other stakeholders by the use of documents (either in electronic or paper formats) and offering an easy to use, user friendly interface with reporting, analysis and planning capabilities to better manage the Zoo's related activities. The implementation of the different functionalities for the GEEWHEZ ERP add-on capture information from the relevant processes, use of applications and related document types that are used by the participating SMEs in the GEEWHEZ project in order to handle purchases and sales, inventories, customer relationships, accounting, management of animal and plant collection and personnel related management issues. Each of these functionalities gives origin to a different module inside the ERP system. The GEEWHEZ ERP is developed on top of the GEEWHEZ middleware using all its major characteristics and tools such as user navigation forms, reporting capabilities, periodic and on demand task execution, XML communication messaging and XML to database mapping

Each of those solution is a single system able to work independently from the other but, at the same time, they can merge all together in order to provide a new level of information to the user of the platform. This data correlation is possible thanks to the GEEWHEZ middleware and thanks to the several plug ins developed inside each add-on that allow the publishing and the collection of data.

- **Middleware**:
  The middleware, based on servlet technology, concentrates in a single infrastructure the common functionalities shared by the add-ons. The middleware is in charge of the interface with the plug-ins, it allows the customization of the basic management software functionalities and it supports them with the user interface. It also represents the extensible API, used as a starting point for all the software extensions. A key feature in the middleware development has been the meta-application approach adopted. The middleware is a kind of meta-application, an application itself, with its own “meta-database” and its own logic, but its scope is to support the underlying add-ons’ applications.

GEEWHEZ is a platform providing a unique point of access to the entire park of services supplied.

3 S&T results/foregrounds

3.1 TECHNICAL APPROACH

The GEEWHEZ project starting from an internal analysis of SMEs like parks and zoos and understanding the heterogeneity of the several solutions in them, proposed and implemented a way to uniform all the systems in a way able to offer to zookeepers, and visitors, the possibility to work one time on the data and having the possibility to check all of them where ever and when ever. The main innovation of this solution is the possibility to have, in a unique platform, the entire SME infrastructure by means of a software layer, a middleware, able to collect and publish data coming from the several parks' zoos' tool and keeping them available in line with the skills of the users.

3.1.1 Technical concept

As far as, managing each department of the enterprise, i.e. administrative, financial, research, market and so on, current solutions treat each department independently from each other and, in general, not really scaled on the specific needs of the SME. Moreover these platforms, refer to Figure 1, are not so easy to use by the involved personnel leading to extensive training needs; in addition to extensive external vendor assistance when a problem occurs.

Figure 1: The resources necessary in a park/zoo

In contrast with this approach GEEWHEZ creates a holistic system and not islands, leading to a "unique" platform, accessible through a single point of access and able to share information among the several management systems.

3.1.2 System architecture

GEEWHEZ architecture is based on a common shared software layer, the GEEWHEZ middleware, where each tool installed in the park is connected to. The presence of the middleware allows each tool of the park, hereafter called add-on, to save and share information with the middleware and, passing through it, with the overall add-ons deployed in the park/zoo.

GEEWHEZ is a distributed platform where each different add-on is deployed somewhere in the park and exploiting a network infrastructure it can allow an implementation of a dialogue among each module by means of common interfaces, common data stores and common procedures published by the GEEWHEZ middleware.

Hereafter we detailed each element forming the GEEWHEZ platform.

3.2 SYSTEM DESIGN AND PROTOTYPING PLATFORMS

3.2.1 Middleware

The goal of GEEWHEZ is to provide a software solution that integrates the add-ons in a unique solution. More specifically, the add-ons will integrate:

- a plug-in, that is a set of hardware and software components that collects, processes and makes available to the system data coming from the park infrastructure
- an application, that consists of the software required to make the data provided by the plug-in persistent and available in all the needed forms

The middleware is a servlet (that is a java program running on an application server) that concentrates in a single infrastructure the common functionalities shared by the add-ons. The middleware is in charge of the interface with the plug-ins, it allows the customization of the basic management software functionalities and supports them with the user interface. It also represents the extensible API, used as starting point for all the software extensions. It comes out that the middleware has to serve two different types of users:

1. Add-on developers, they are in charge of the middleware customization and, possibly, extends it, if needed for their add-on
2. Final user, the zookeepers, that will interact with the user interface to use the overall platform

In order to serve users like add-on developers and zookeepers, a key feature in the middleware development has been the meta-application approach adopted. In other words, the middleware is a kind of meta-application: an application itself, with its own "meta-database" (that, as we will see below we call settings database, in contraposition to the information database), its own logic, etc., but its scope is to support the underlying add-ons' applications. The process can be summarized as indicated in Figure 2: the add-on developer uploads the settings related to a specific application form or report, these settings are persisted in the settings database and retrieved by the middleware to dynamically generate a customized specific user interface.

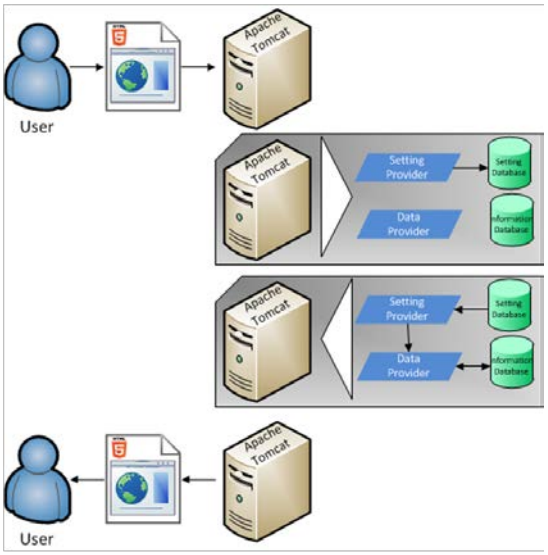

Figure 2: Meta-application

For what concerns the software tools, the choices have been oriented towards open source solutions, both to have a possible full access to the sources but also to maximize profits and margins obtainable by the sales of the platform. Of course, by the nature of GEEWHEZ a crucial role for the middleware functioning is played by the DBMS. PostgreSQL has been chosen as it is considered one of the most stable open source DBMS, it offers relational and transactional integrity and support for extension programming language.

Conversely to what initially planned no client side components is needed in GEEWHEZ as all the needed functionalities can be provided by a single web interface. For this reason:

- the middleware has been developed as a Java servlet and the use of Apache Tomcat as application server and java servlet container has been required
- the client side application consists of a HTML5 page (available via compliant browsers), equipped with a JavaScript application, based on jQuery

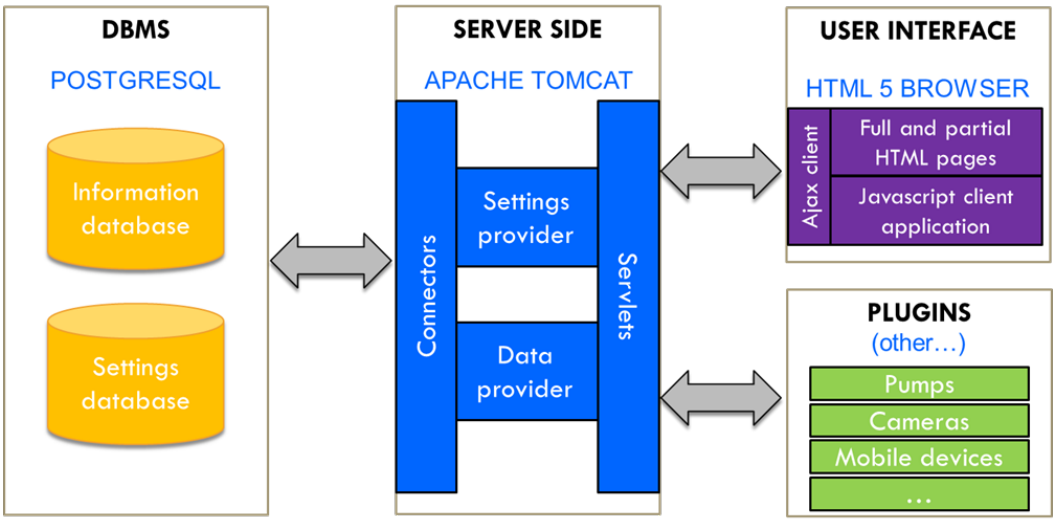

Figure 3: System architecture

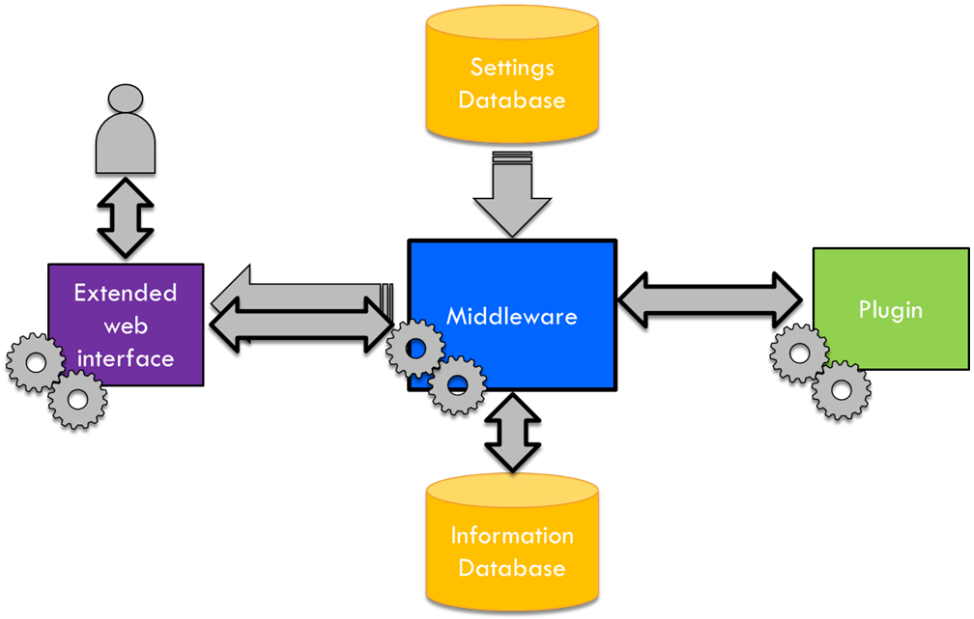

Figure 4: Add-on functioning

The communication between server side application and DBMS is realized by jdbc connections that are pooled by our java classes to keep it efficient. Once the user interface has been downloaded and displayed in the browser, the communication between client and server side is provided by AJAX calls that connects to the middleware.

The middleware offers to the developers the functionalities to create their own add-on applications. The add-on developer can use these functionalities to feed the system with the settings needed to build forms, reports, etc.. The add-on's settings are stored in the settings database and used by the middleware to "know" how to make it works and more precisely how to interface with the plug-ins and how to build the user interface.

One the most relevant results obtained during the middleware development is the distributed dataset. The basic concept beyond it is related to the scenario where the middleware needs to behave as API from which to start for the extensions by add-on developer. This is necessary when flexibility is needed to overcome the basic functionalities offered by the customization capabilities of the middleware. When it happens, the risk to lose uniformity and stability, can be avoided with Java classes, as they fully respect OOP inheritance power, but not in JavaScript procedures that are intrinsically weak and don't support robust mechanism to control what inheritors can do. For this reason, it has been built a dataset, that is a data container, that automatically replicates the information among user interface (client side) and middleware (server side), using event based algorithms.

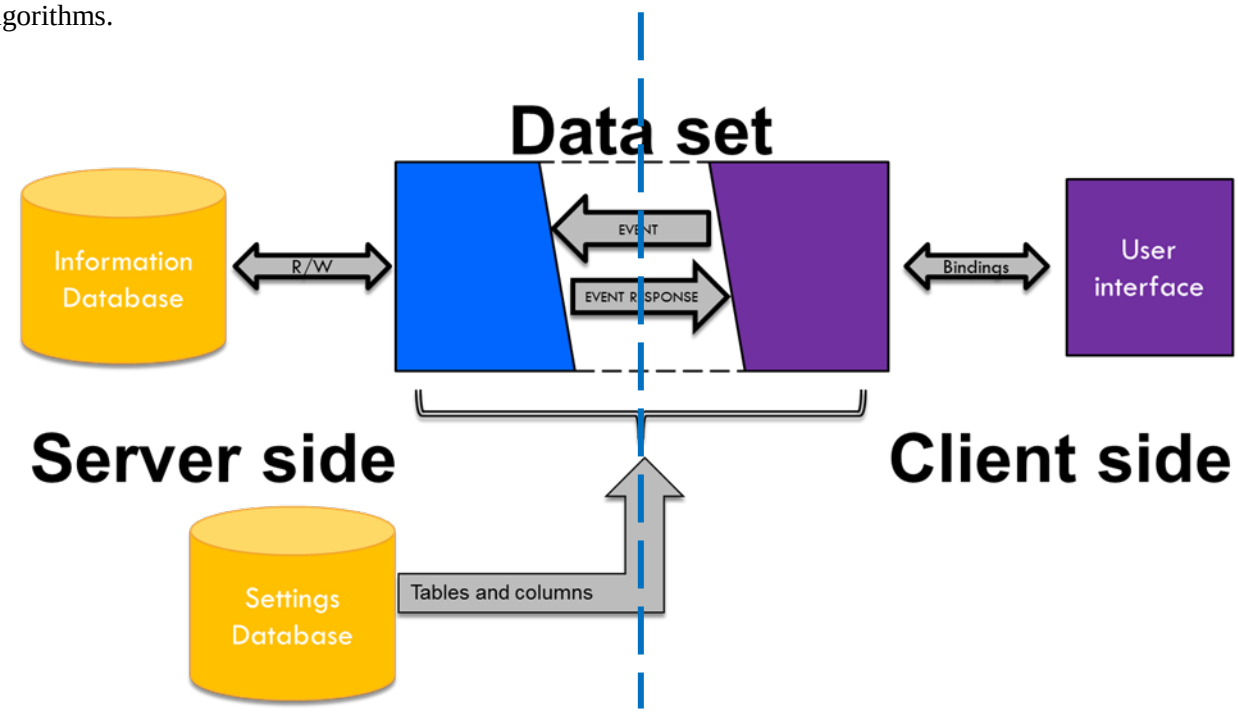

Figure 5: The distributed dataset

This data replication is performed in real time and makes all the information provided by the users (commonly available only in the client side) ready to be used by the server side business logic java classes. In this way all the needed extension can be done starting from the Java API and we have been able to lock up all the client side software without losing anything in flexibility.

The distributed dataset is also responsible for the virtualization of data transfer. Indeed, the information transferred from the database to the middleware and from the middleware to the user interface is restricted to what is currently displayed by the final user. This saves time and resources when the database increases and makes the response time practically independent from the amount of stored records.

The provided way to interface with the plugin can be summarized as indicated in Figure 6.

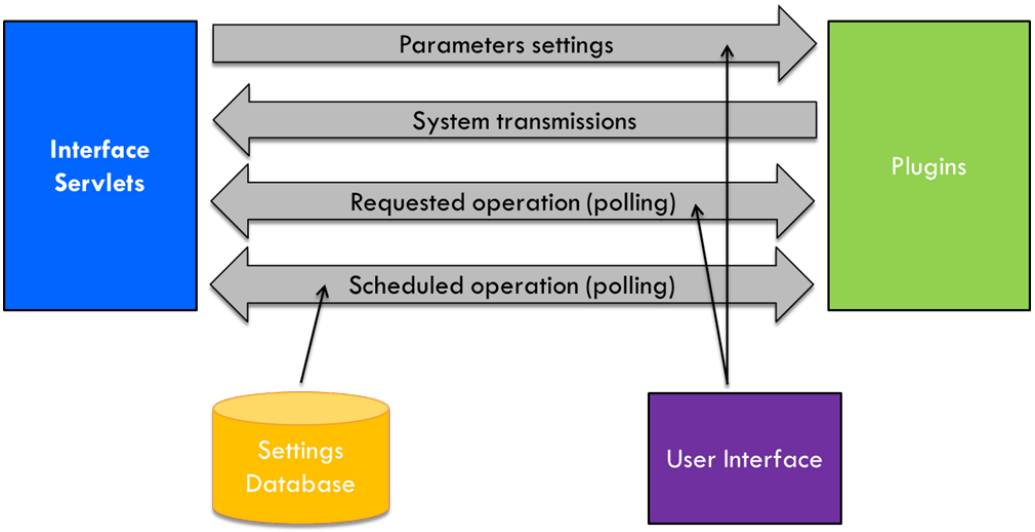

Figure 6: Plug-ins interface

The working parameter related to a plug-in can be sent by the final users to middleware and hence to the plug-in itself, using the user interface. On the other hand, the plug-in can regularly transmit operating information to the middleware, that will store it to make it available in reports and forms. Beyond the regular transmissions, the information can also be polled by the middleware (e.g. basing on a scheduled task) or directly by a final user (zookeeper).

3.2.2 Video surveillance system

The video surveillance system monitors the access to different areas of the park and controls the security of the park. Applications such as the detection of the flow of visitors and the monitoring of long waiting time of visitors at certain areas are important capabilities of the video surveillance system. Another important application of this system is the wildlife monitoring, such as the control of the flow and occupancy of animals in a given enclosure (to obtain data about their behaviour) and the detection of escaping animals or intrusion at their enclosures.

The surveillance system consists of a number of cameras and processing units (CPUs) that interact between each other. Each camera will be installed together with its associated optics inside an IP66 rated case. This set-up will also contain a LED illumination with a sensor, a fan and a heater with a controller card and a temperature probe. The communication between every camera and the processing unit will be implemented by a Frame grabber through a GigaBit Ethernet interface. The camera-processing unit connection is indicated in Figure 7.

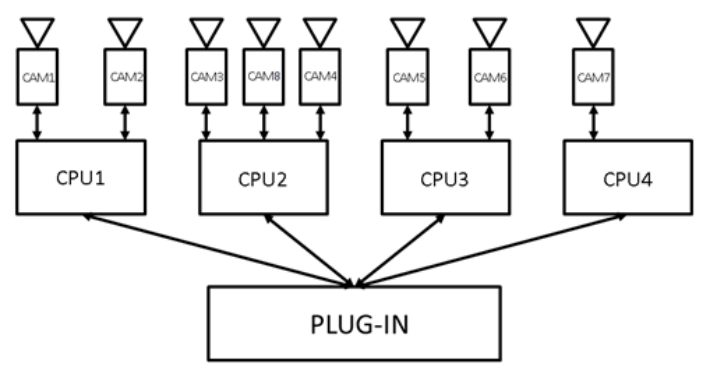

Figure 7: Cameras-Processing Units interconnection

As indicated in Figure 7, one, two or three cameras are connected to the same CPU which in turn are connected to the network to inject the information in a sequential manner to the middleware, according to 10 frame per second, sending this information within an XML file.

In Zoomarine GEEWHEZ installation, there are 8 cameras and due to their location in the park, several cameras are connected in such a way that each pair is sharing one processing unit and one of the cameras uses one computer. Therefore, there are eight cameras and four CPUs as shown in Figure 8.

Figure 8: Cameras' and CPUs' locations in Zoomarine

The video surveillance add-on executes several cases simultaneously for each camera set-up. This depends on the specifications of the system and it is case specific. For instance, a camera pointing to an animal enclosure might be executing case 3 (detection of intrusion), case 4 (flow of animals) and case 5 (escape of animals).

The video surveillance system add-on performs the following operations:

- **Use Case I - Counting of people at specified areas (queue estimation)**
  The aim of this use case is to detect and calculate the waiting time of visitors and the occupancy at certain areas, for instance, in queues. In this case, the system detects people agglomeration, calculates the occupancy at a given area and estimates the number of people in a given time. If the amount of people estimated is over a defined value, which is related to the assessed area capacity, an alarm will be generated and delivered to the middleware.
  Figure 9 shows a flow diagram that describes the implementation of the use case I. This diagram contains the different modules involved in the implementation of the mentioned case. The first column on the left shows the main method. The second and the third column show a detailed split of the "Process Sub Image" and "Remove Illumination" methods.

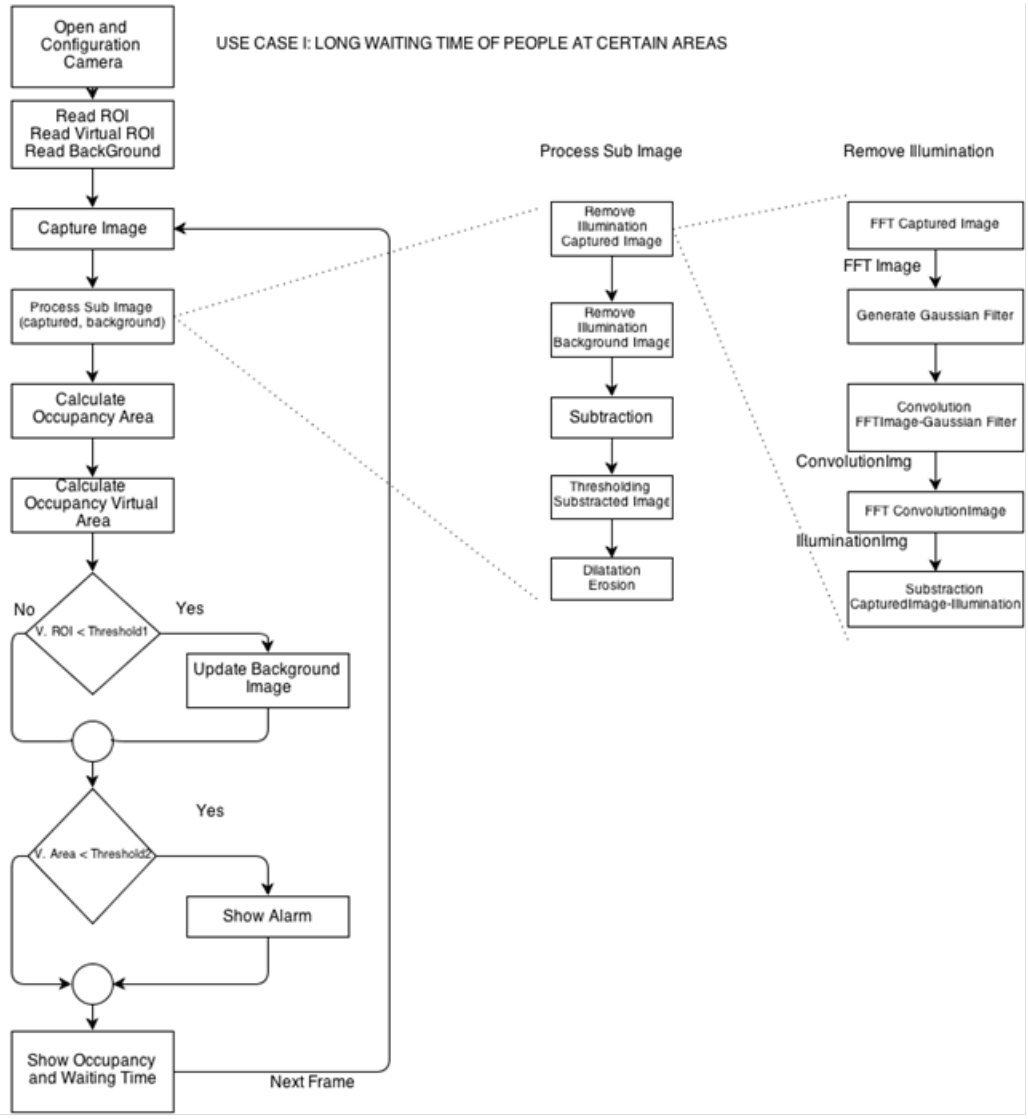

Figure 9: Image processing flow diagram Use Case I

- **Use case II - Movement pattern of visitors at certain areas**
  The goal of this use case is to obtain information of the visitors flow and to estimate the amount of people in the assessed area. The surveillance system add-on detects the most important motion flows of people in that area. Even if the is no unusual activity, the surveillance system obtains information and send it to the park authorities.
  Figure 6 shows the flow diagram of the implemented use case II. The first column on the left shows the main method. The other columns show the most relevant functions: “Tracking” and “Group Tracking”.

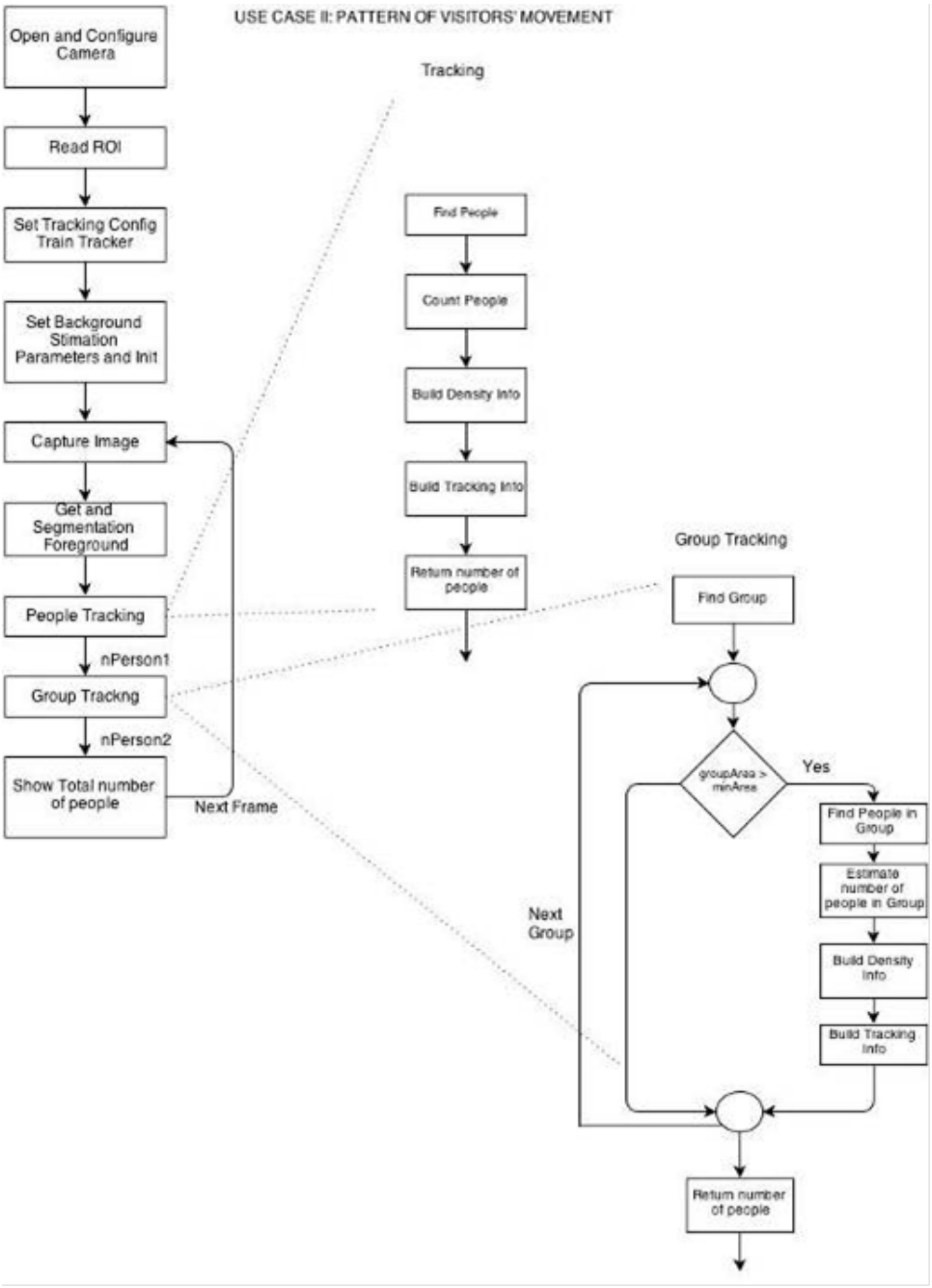

Figure 10: Image processing flow diagram Use Case II

- **Use case III - Detection of intrusions at specified areas**

  The aim of this use case is to detect intrusions in a specific area. So to achieve the main goal of this use case, people are detected in an inspection area and then, if there is an intrusion, an alarm signal is generated. There is a special situation, for instance, when for a given period of time, no activity is detected by the surveillance system add-on in a given area. This can be due to the park being closed or because the link with the camera is broken.

  Figure 11 shows the flow diagram of the implemented use case III. The column on the left contains the function main. On the right column, the most relevant function, “People Tracking”, is described.

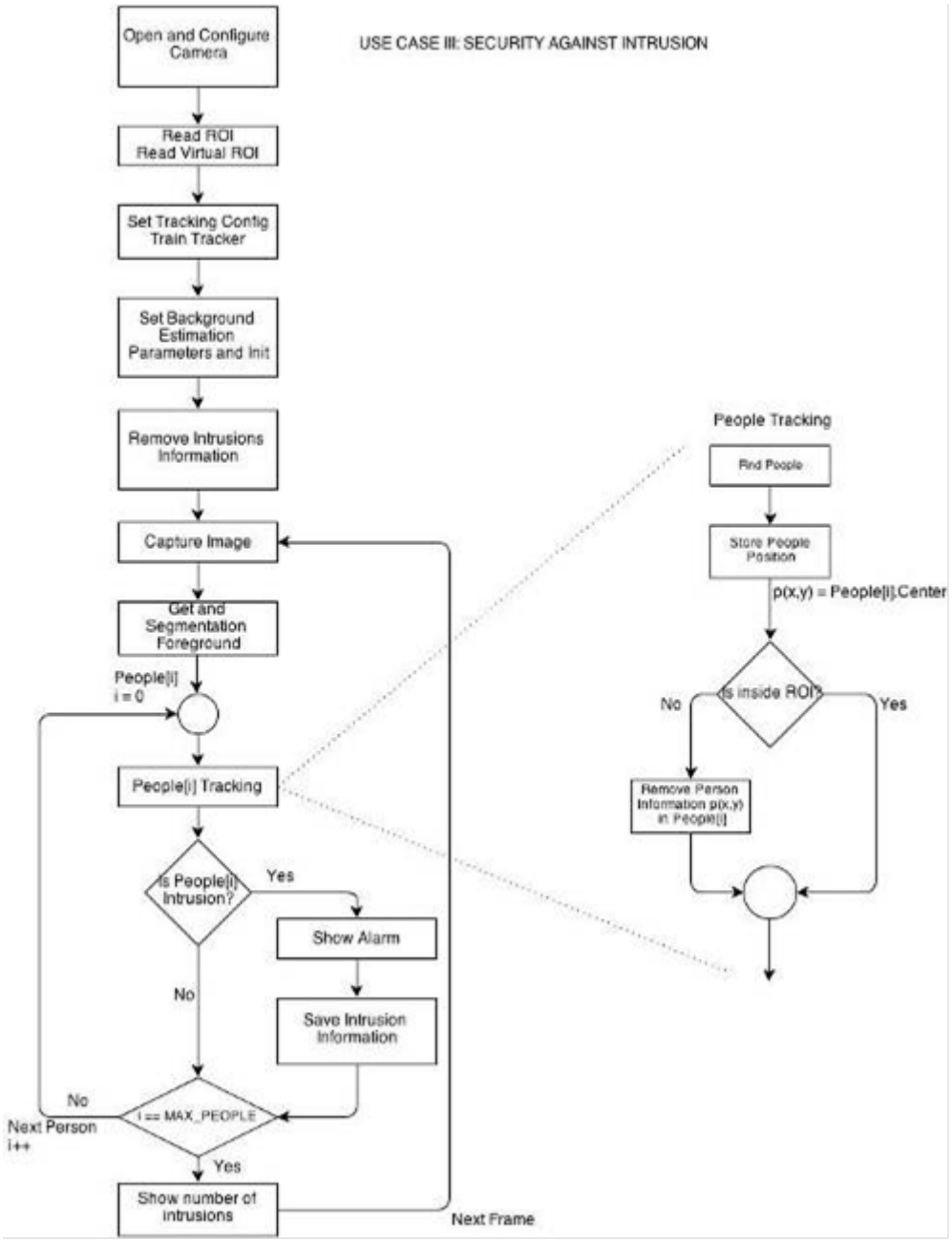

Figure 11: Image processing flow diagram Use Case III

- **Use case IV - People tracking in a given area of interest**
  The aim of this use case is to control the flow and occupancy of animals at their enclosure in order to obtain data about their behaviour. The surveillance system tracks the amount of animals and their movements into an enclosure, generating a density map where it is shown the movement pattern of animals with different colours. Each colour represents the number of times an animal passed through a specific area.
  In Figure 12 we reported a flow diagram setting out the software implementation of use case IV.

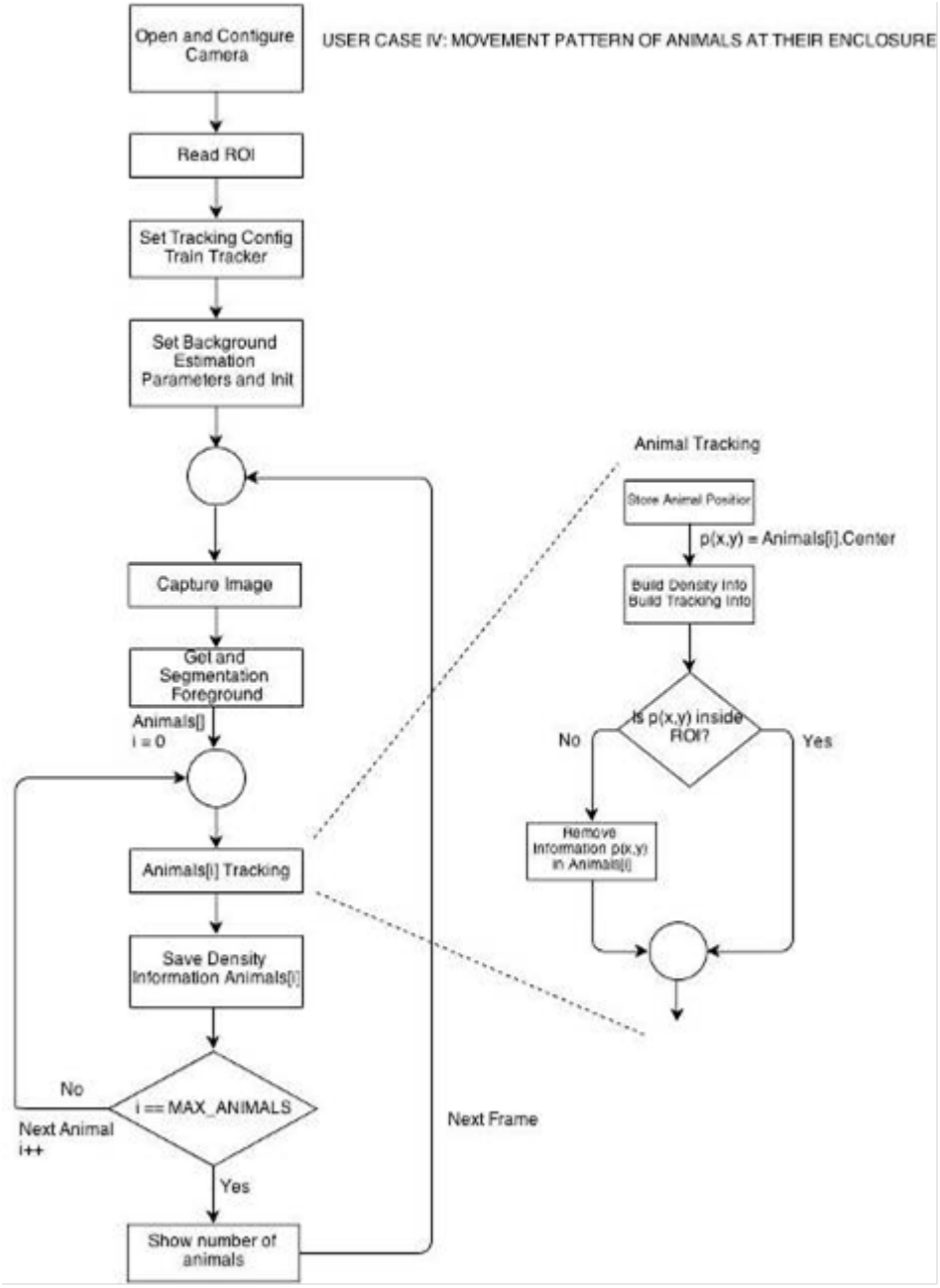

Figure 12: Image Processing flow diagram Use Case IV

- **Use case V - Animal tracking at their enclosures + Use case VI - Detection of escaping animals** The aim of these use cases is to ensure the safety in the park. The surveillance add-on detects situations in which an animal is close to the fence or if there is any animal escape. If any of the previous situations happens, an animal escape alarm is generated and sent to the middleware. In Figure 13 reported a flow diagram setting out the software implementation of use case V and use case VI.

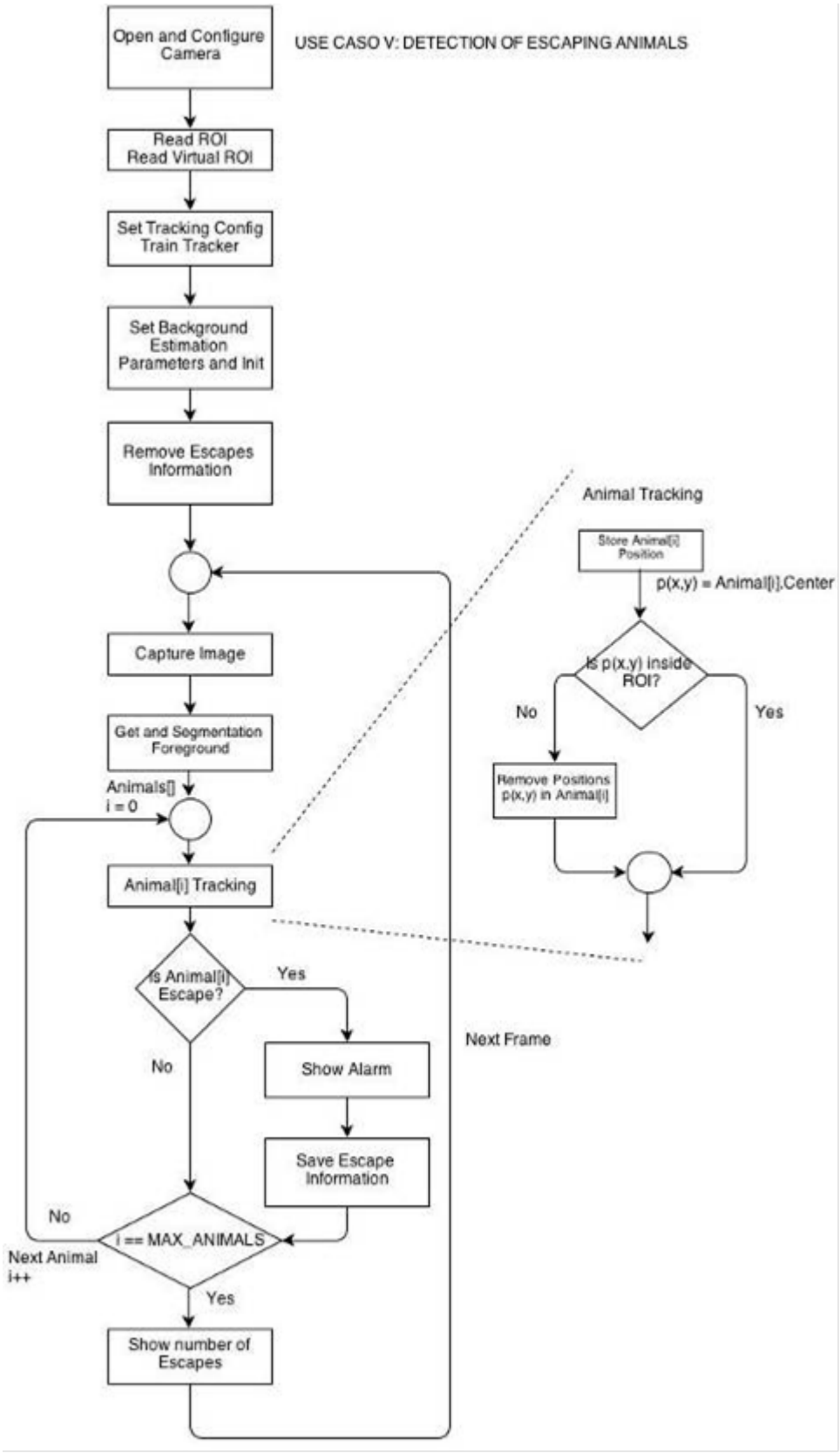

Figure 13: Image processing flow diagram Use Case V + Use Case VI

- Send and receive data to and from the middleware

  The Plug-In implements the interconnection of the add-on and the middleware by XML files. The functionalities of the Plug-In are the following:

- Delivery of the data from the middleware to the add-on. The Plug-in parses the received files from the middleware and delivers them to the appropriate camera system.
  The middleware delivers an XML containing all the information related to the cameras to be enabled and the information to set them up. This XML file is received by the Plug-in, which in turn parses it, splits it in different XML files that are then delivered to the camera systems. These files contain the information required to get the camera system enabled and ready to start capturing and processing images

- Delivery of the data from the different camera systems to the middleware. The Plug-in links the data received from the different cameras systems and delivers a file to the middleware. The images requested by the cameras are sent as the rest of information to the middleware, through the Plug-In, which delivers a unique XML file. The Plug-In creates a thread that continuously listens and waits for XMLs files from the CPUs.

In order to make up this interconnection, the Plug-In manages two XML files queues as shown in Figure 14.

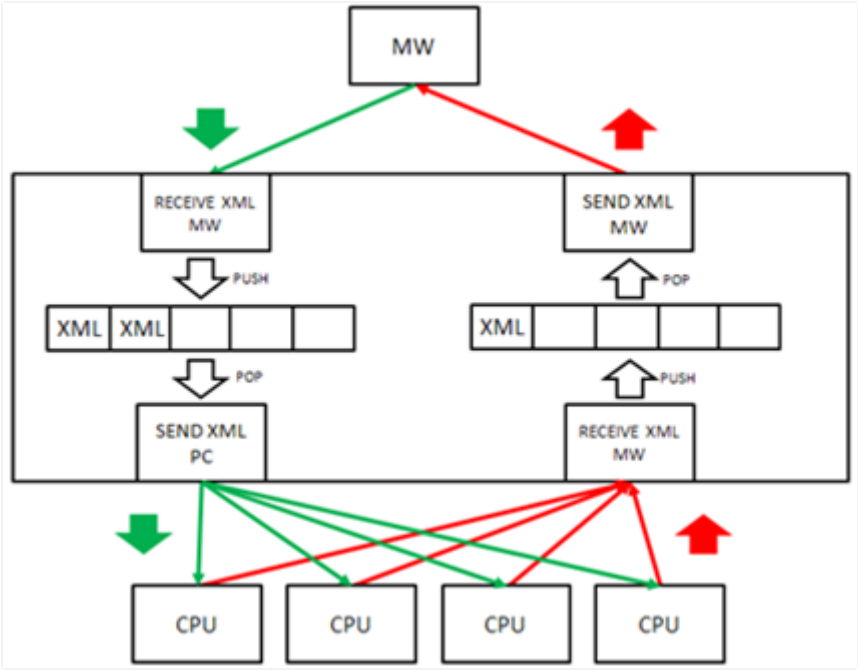

Figure 14: Plug-In structure

3.2.3 Water monitoring system

The water monitoring system add-on is based on the middleware capabilities and, more specifically, it uses the system integration API to access the instantaneous readings from the probes and store this information, together with the reading time, on the database, to make it available to GEEWHEZ.

The database structure is based on 4 tables, corresponding to the schema “Water Monitoring”:

1. **Systems**
   This table contains the “systems” available in the park. For example, in Zoomarine the table contains 3 records, one corresponding to the dolphins’ pool and 2 corresponding to the pinnipeds’ pools. For each system it is stored a unique code, the name and the IP address. In particular the IP address is used to retrieve the readings, as explained below.

2. **Devices**

   Each system, as stored in the previous table, contains several probes. For each probe the table Devices stores a unique code, the system that controls the probe (table Systems) and the name of the device. This table is updated automatically by the add-on

3. **Channels**

   Each channel represents a physical or chemical parameter. For each channel there is a record in the table Channels containing a mnemonic unique code and the unit of measure. This table is updated automatically by the add-on

4. **Measurements**

   The measurements are read each 5 minutes and they are stored (as explained below) in the table Measurements. For each record it is stored: a unique Id, the reference to System, Device and Channel, the value of the reading, the time of the reading and the range (min / max).
   The range is stored for each measurements as it can change due to possible fine tunings on the probes. This information is used to alert the administrative staff in case of anomalies and to highlight them in the reports.

The interface between water monitoring system and middleware is realized using the API contained in the System Integration module with the XML produced by the Plug-In. The XML is produced by each "system" (e.g., for Zoomarine: dolphins, pinnipeds 1 and pinnipeds 2) of the Plug-In. The XML is available in the corresponding URL (starting from the IP address of the system) after a client authenticates itself. Both the authentication and the XML reading are handled by REST calls. The XML contains real time information taken from the probes (that is the device, the channel and the measurement information). The responses to the REST calls, both for the login process and for the reading, are sent as web page and the XML must be extracted from the HTML script. Since the HTML scrip is not a valid XML, the SAX parser developed in the add-on to process the information cannot be used directly over the stream. For this reason a specific class called "Filtered Input Stream" have been developed in the add-on. This class act as a proxy and filters all the bytes read from the stream that come before or after the XML part. The XML part is recognized basing on the opening and closing tags (that are never contained in the extra HTML script).

The add-on handles the information parsed from the XML basing on the logic classes contained in the package eu.geewhez.watermonitoring.systeminterface. the information stored in the XML are coming from the master modules mounted in each pools. After understanding how the water monitor system has been integrated in the GEEWHEZ platform, exchanging messages with the middleware, it is important to understand how those information are collected from the pools. In zoos and parks with marine animals the pools of those animal has to be continuously checked and monitored in order to assure the safety of the animals living inside them; to measure the parameters affecting the life of those animals each pool is generally equipped with a system of filters and tank place under the pools, as indicated in Figure 15. To optimize the analysis on the water of the pool and to increase the efficiency of the system on each tank of the filter room, the GEEWHEZ consortium mount a panel like the one shown in Figure 16. In that panel there are sensor and pumps able to:

- Measure the pH: using sensor type PHES 112 SE having glass sensor shaft and ceramic membrane, measuring range 1…12

- Measure the redox reaction: using sensor type RHES Pt SE having glass sensor shaft and ceramic membrane, electrode material platinum

- Measure the free chlorine: using sensor type CLE3.1 CAN, a sensor for connection to a CANopen interface, measuring principle amperometric, 2 electrodes, diaphragm-covered
- Measure the total chlorine: using sensor type CTE 1 CAN, a sensor for connection to a CANopen interface, measuring principle amperometric, 2 electrodes, diaphragm-covered
- Inject chemicals in to the pools' water by means of solenoid dosing pumps controlled via CAN BUS

Figure 15: Filters room

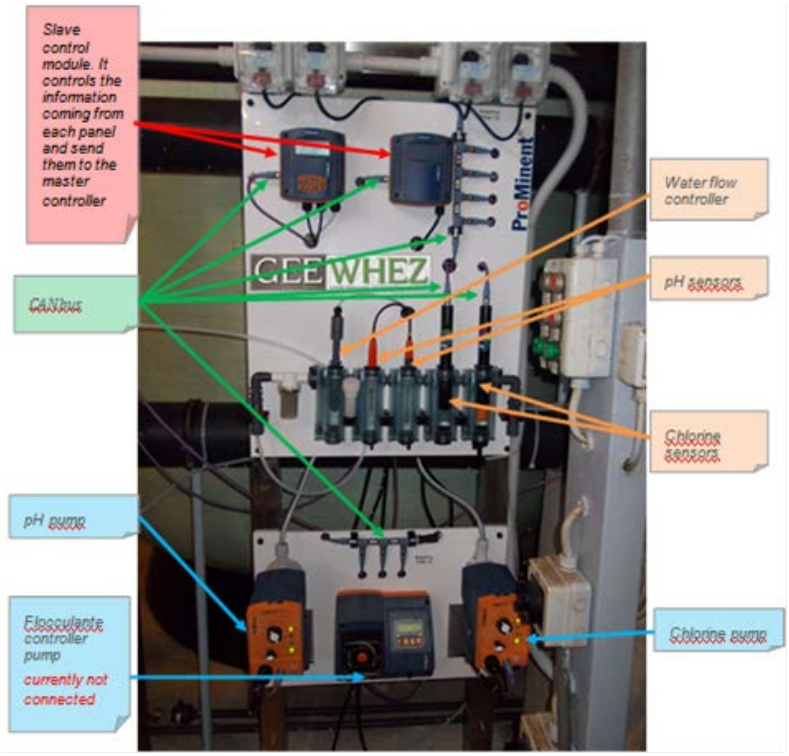

Figure 16: Water monitoring system slave panel

After installing the slave panel, Figure 16, on each tank of the dolphins' and pinnipeds' pools the aspect of the filter rooms is reported in Figure 17. Each slave panel sends the detected measure to the master modules, see Figure 18, that publish the XML data to the GEEWHEZ middleware.

Figure 17: The GEEWHEZ filter room

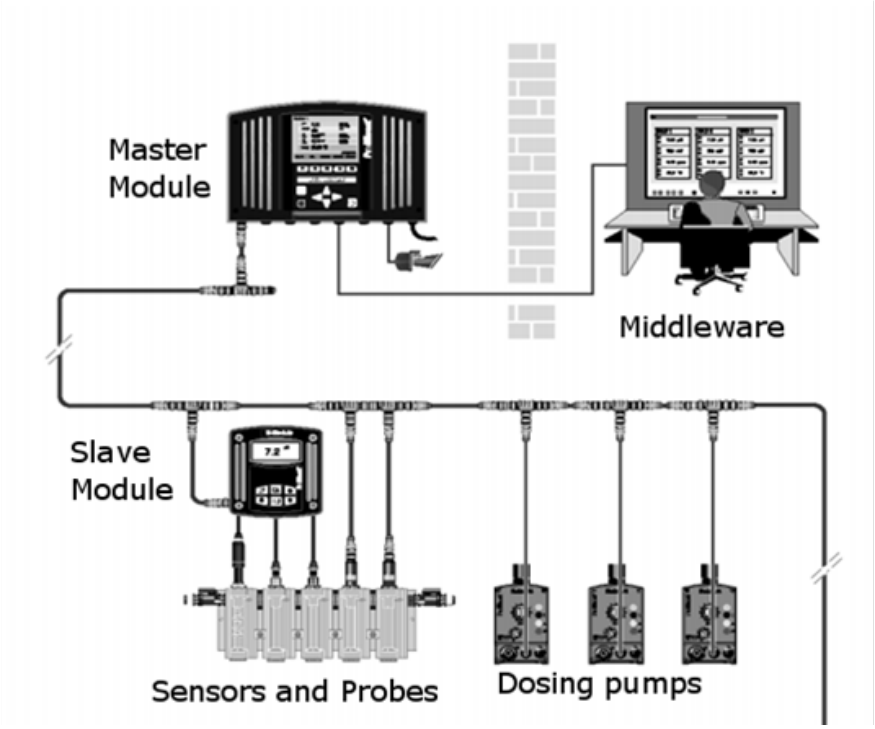

Figure 18: Water monitoring system architecture

3.2.4 Leisure system

The Leisure system is the add-on of GEEWHEZ platform in charge of delivery information to visitors and to zookeepers. The add on is formed by the following macro modules, briefly shown in Figure 19:

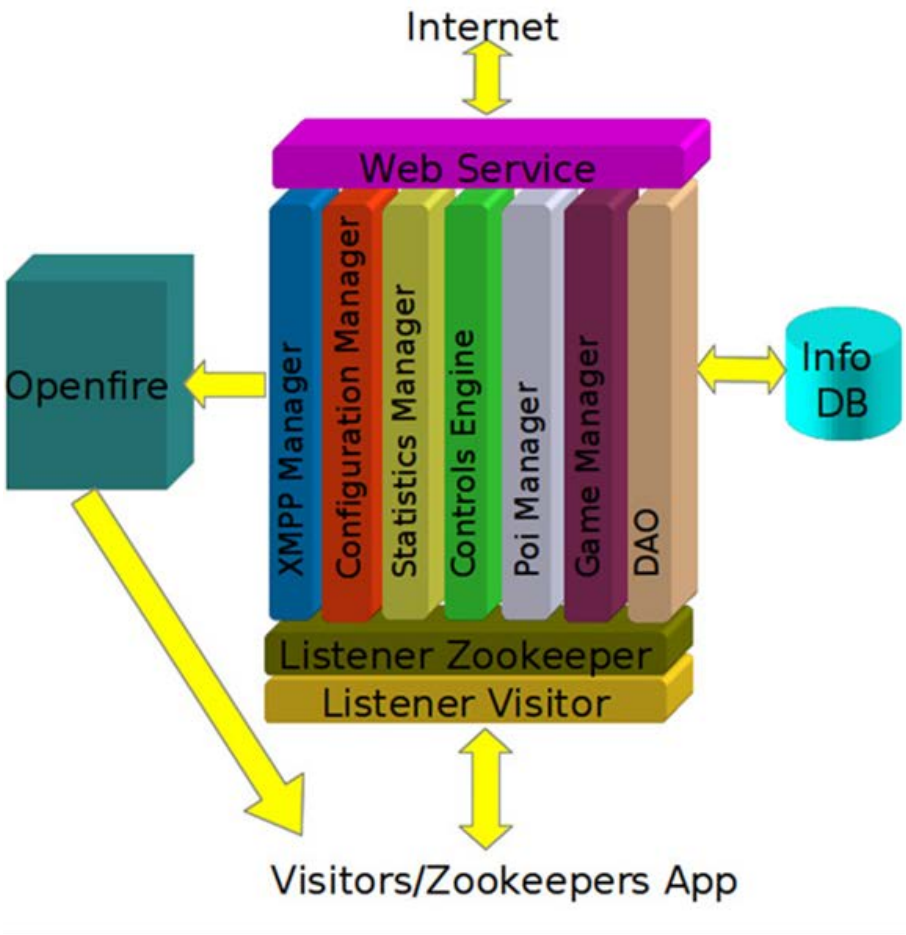

Figure 19: Leisure system architecture

- **Openfire server** (http://www.igniterealtime.org/projects/openfire/): It is a server used to simulate a P2P communication among the GEEWHEZ leisure system server and the GEEWHEZ leisure system Mobile Applications. OpenFire is a RTC server using XMPP protocol for instant messaging; it is invisible to the clients that send and receive messages as in a real P2P network. P2P network are special type of network where the hierarchy of master and slave has been lost, all the network nodes work as peers. It means that the communication can be started by the server and by the client as well

- **Info DB**: it is the database available inside the middleware, here as already indicated in the middleware section, where all the GEEWHEZ data are stored

- **Zookeeper listener and the related App**: It is the interface between GEEWHEZ leisure system server and GEEWHEZ leisure system zookeeper App. This component receives the messages from the device, manage them and send back the reply to the App. The listener does not implement a business logic because it is only a system interfaces, we moved all business functionalities into Controls Engine

- **Visitors listener and the related App**: Like the zookeeper listener also the visitor listener is an interface between GEEWHEZ leisure system server and GEEWHEZ leisure system visitor App, the only difference is the type of information exchanged over this interface. Here we can find data concerning the attraction of the park, not their management; the availabilities of games not the management information of the park or the security issues

- **Business logic unit**:
    - ***Configuration Manager*** *– It retrieves periodically system parameters from DB and publish them in order to speed up the access to DBs information and make them much more easy to access in respect to a direct DB access*
    - ***Statistics Manager*** *- It collects and stores the GEEWHEZ Leisure System statistics like: How many visitors entered in the park, How many groups have been created by visitors, How many visitors is close to a specific PoI*
    - ***PoI Manager*** *- It polls the DB PoI table and save the PoI data in a cache structure*
    - ***Game Manager*** *- It is in charge of periodically checking the changes in the game available in GEEWHEZ platform. If any changes will happen it updates games cache structure*
    - ***Controls Engine*** *- It represents the business logic classes of all GEEWHEZ Leisure System*
    - ***DAO*** *- It allows the interaction with the DB in order to implement SELECT, UPDATE and INSERT queries. GeewhezDao, the main class, offers the following useful methods:*
        - *getConnection – it returns a valid and active JDBC connection to whom invokes it*
        - *commit - it executes commit method on JDBC connection and it releases resources*
        - *rollback - it executes rollback method on JDBC connection and it releases resources*
        - *executeQuery – it is a generic method to execute a SELECT and return a generic data structure to generalize the output*

      All these methods allow us to centralise all the connection and to avoid a bad connection management

The leisure system is completely integrated in the rest of the GEEWHEZ platform allowing a complete cooperation among the different modules composing the solution. Accessing the GEEWHEZ platform by means of its web interface, Figure 20; clicking on the leisure add-on and exploding its menu, the zookeeper can manage the several information available in the park/zoo. Visitors and zookeepers can access them using their mobile phones wherever they are in the park.

The ZooBotanico leisure system solution has been installed in the zoo also with a complete wireless infrastructure that assuring to the visitors of the zoo a wireless access to internet. In Figure 21 we reported the final Wi.Fi. deploy of the installed network.

Figure 20: GEEWHEZ web interface

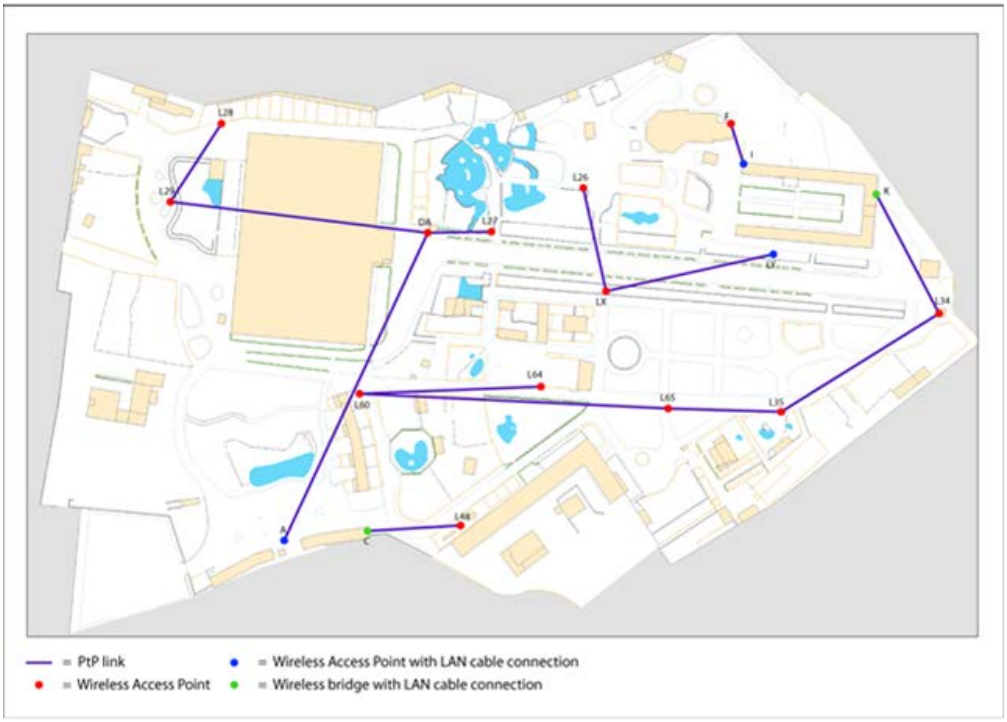

Figure 21: ZooBotanico GEEWHEZ WiFi network

3.2.5 ERP

The main components characterising the GEEWHEZ ERP are:

- Web Client
- Web Server
- Database Server
- POS (Point of Sales)

The Web Server (based on the GEEWHEZ middleware) communicates with both users and applications. Users make use of a javascript enabled web browser to navigate through the different functionalities of the GEEWHEZ ERP. External applications such as the POS System communicate with the middleware based web server via the task oriented open web services provided by the middleware infrastructure. In addition, the Web Server collects, manages and updates data from ERP database. A database connector is also used in order to synchronize data coming from the external applications using an XML to database model approach. The following diagram illustrates the top level components and their relationships.

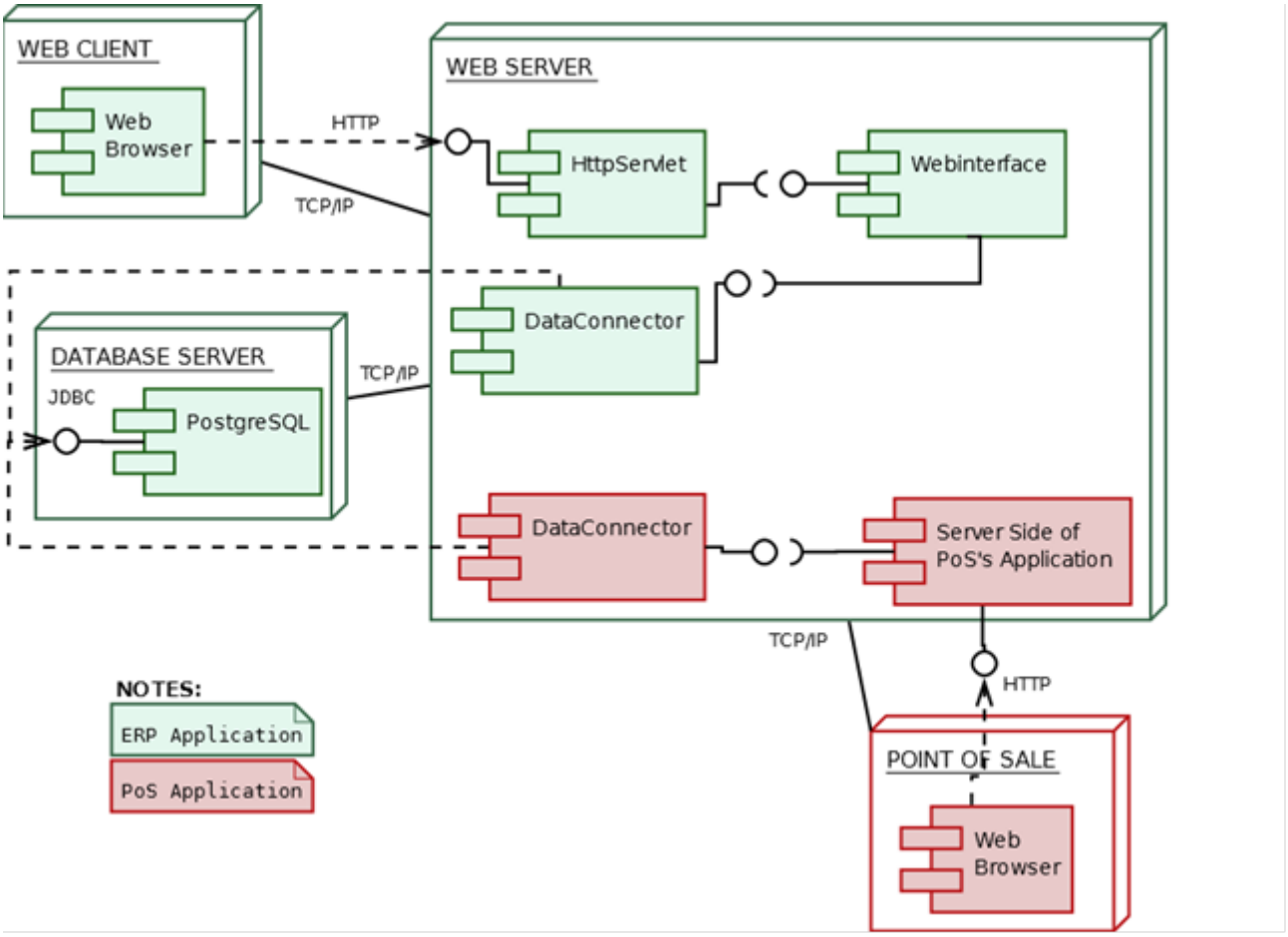

Figure 22: ERP top Level Architecture

In order to show the information exchanged and main functionalities provided by the different interfaces that should connect the top-level elements identified in the ERP System architecture, high level use cases have been defined. Each high level use case includes a subset of functional use cases so that all functionalities of the system will be covered.

- **Web Client**
  It is related with the information an user is able to see on his computer via a web browser and it is connected to web server component in order to ask for Database Server stored information

- **Web Server**
  It is the middleware based part of the GEEWHEZ platform. Its main functions are to process client's request and handle the connection with the database

- **Database Server**
  This component stores the ERP system modules necessary information, such as Warehouse, Sales and

Financial related data. It communicates with Web Server component in order to send/receive data. The database server implements tables, views, triggers and data consistency and extended data access logic with makes it possible to automate the different aspects of the GEEWHEZ ERP such as automatic accounting book registrations

- **Point of Sale**
  External Component with an application that runs in a separate way from the middleware application. The POS presents in a user friendly touch oriented user interface the different products or services that are on sold at each particular shop or restaurant. The product information is synchronized from the ERP database using an open web service communication mechanism as provided by the middleware. The logic to calculate and apply promotions and to generate invoices and keep track of each journal item is also implemented. Integration with the warehouse module to keep track of remaining products and detect low level of stocks is also provided

The top level architecture in Figure 22 has two main sets of components: the ERP System ones and, the remote ones, which are external and are utilized by ERP System via middleware provided communication mechanisms. The ERP System has three major components:

- Warehouse Module
- Sales Module
- Financial Module

The remote components are:

- Point of Sales (POS) systems

The following diagram shows all the details of the ERP System architecture and the relationships between its components.

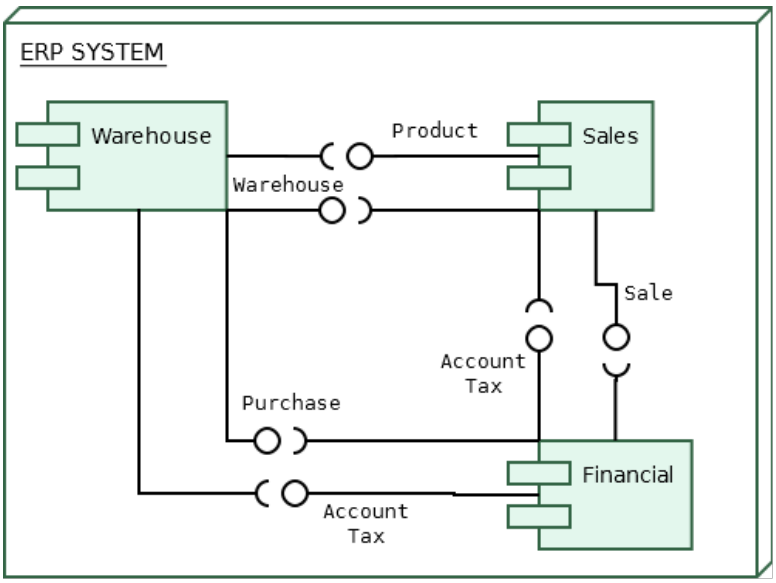

Figure 23: Second Level Architecture

- **Warehouse**: It is in charge of functionalities related to day-to-day ATP warehouse activities: management, product inventory and reporting. The Web Server module interacts with this one that provides access to the database: this interaction allows storing and retrieving from database ATP

warehouse related information. For instance, it allows users to register, consult, update and delete an ATP warehouse and register and consult existing ATP inventoried products in a given ATP warehouse. **The Sales module uses this module in order to obtain information about available ATP warehouses**

- **Sales**: This module provides functions for ATP products registering, consulting, updating and deleting; ATP stores registering, consulting, updating and deleting; ATP POS registering, consulting, updating and deleting; ATP sales registering, consulting and updating; ATP promotions registering, consulting, updating and deleting. **The Warehouse module uses this module in order to obtain information about ATP existing inventoried products**

- **Financial**: This module allows registering, consulting, updating and deletion of all kinds of ATP financial related information: the journals and their journal entries (these contain journal items), the accounts, the bank accounts, the fiscal years and the associated data, the taxes (e.g.: VAT registration), the related information about amortization book, the supplier information and all the tasks associated with the invoice and credit note management.
  The Financial module interacts with:

  - The Sales module to create journal entries and to manage the taxes related to the sales
  - The Warehouse module to create the invoices associated with the purchase orders

  In addition, the module provides a report service that creates the more usual accounting reports (e.g.: daybook, balance sheet, profit and loss, general ledger…).
  The Sales module uses this module in order to obtain information about available ATP warehouses

3.3 Demonstrations

Several demonstration activities have been arranged during the project period but the most important ones are the two organised at the end of the project inside Zoomarine and ZooBotaico Jerez.

3.3.1 Zoomarine GEEWHEZ Day

Figure 24: GEEWHEZ Day

The 4th of September 2013 Zoomarine hosted the GEEWHEZ Day (GD), an event corresponding to the official launch of the GEEWHEZ platform, with particular attention for the app dedicated to the visitors of the park. The GD has been arranged contacting several journals, and preparing totems and leaflets deployed at the entrance of the park, see Figure 24.

During the entire day the consortium partner advise visitors about the possibility to download the App, explaining to them how to download the App. During the GD in fact the App was not available using the Play Store so the visitors must take a picture of the QR-codes available in the leaflet available in the park or printed over the totems deployed inside the park.

Nowadays the App is available on the Google Play Store and here after we report the current statistics, Figure 25. Actually the App has been installed by 113 users (we have to take into account that the App has been released at the end of the summer).

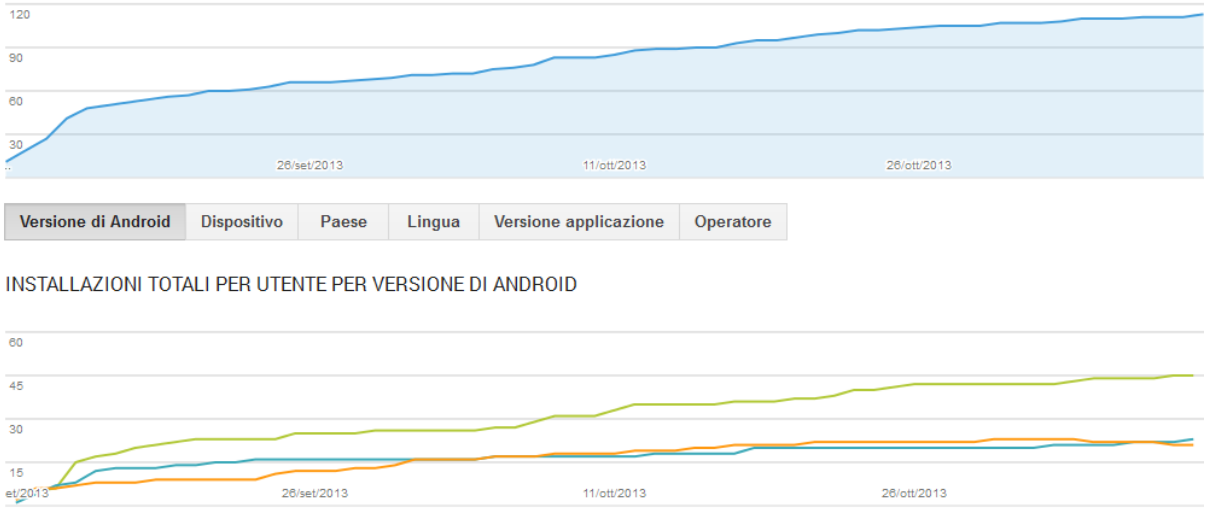

Figure 25: Google statistics

During the GD the consortium demonstrates how the different add-ons of the GEEWHEZ platform can cooperate among them in particular it was shown how using the leisure system and in particular the zookeeper App, the managers of the park can be continuously updated about the park status.

In the following figures we reported several snapshot of the cooperation among the leisure system, the water monitoring system and the video surveillance system.

In Figure 26 we reported the check of probes in the dolphins' pool. The system is also able to report an alarm and using the map of the park the zookeeper can check remotely the value simply clicking on the blinking icon on the map, if the value is a false positive the zookeeper can simply ignore it, in other case it has to go to the water monitoring system station and resolve the problem. It is the chosen methodology, the water monitoring system can only be read by the rest of the GEEWHEZ add-ons in order to do not violate its safety parameters requested to support the animals' life.

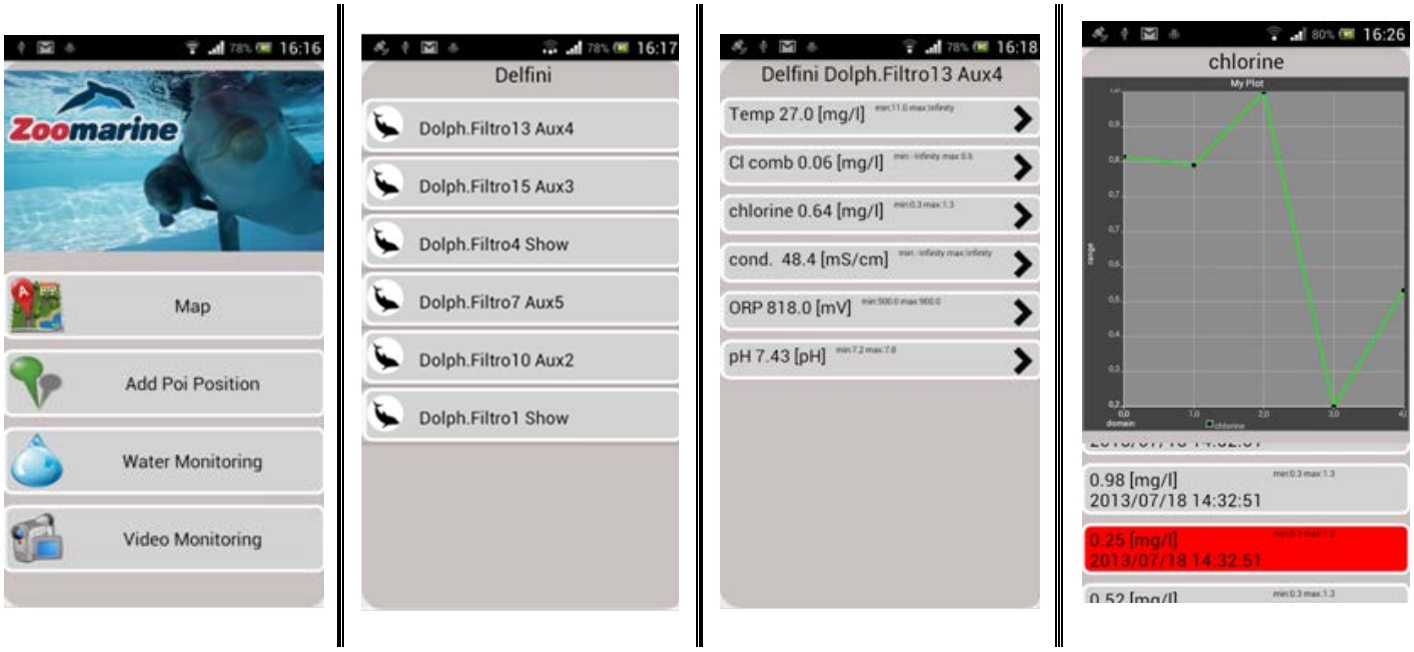

Figure 26: Leisure system and water monitoring system

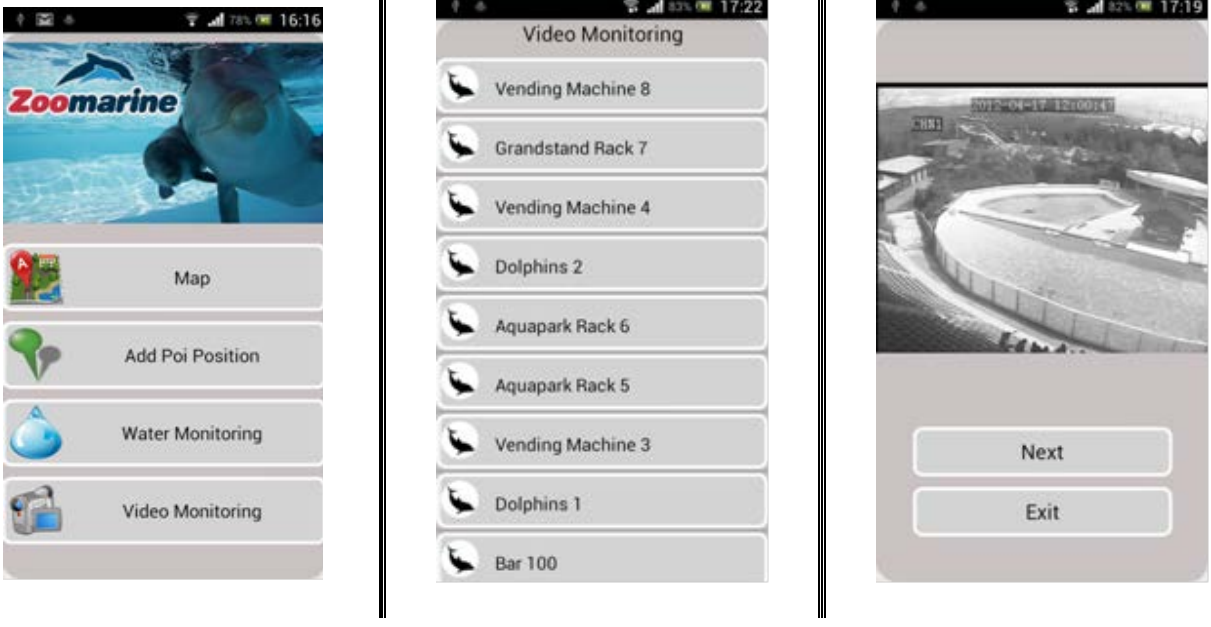

Figure 27: Leisure system and video surveillance system

Figure 27 shows the interoperability between the video surveillance system and the leisure system, in particular it shows how the zookeeper can check the status of a camera only clicking on the video surveillance menu and choosing one of the 9 cameras installed. The system can also report alarms occurred during the day showing inside the zoo map a blinking marker indicating the camera detecting the alarm. The zookeeper can click on the icon over the map and check if the detected alarm is a real alarm or a false positive.

3.3.2 ZooBotanico GEEWHEZ demonstration day

The last demonstration activities performed during the project was the one arranged in ZooBotanico the 20th of September. The GEEWHEZ consortium completed the installation of the platform in the ZooBotancio Jerez and during the rest of the day they presented the app to the several visitors inside the zoo, Figure 28.

Figure 28: ZooBotanico demonstration day

The consortium partners explained to the visitors how they can download the app, directly from the Play Store since the 19th of September 2013, or taking a photo of the pre-prepared QR-code and what are the main advantages they could have using the app:

- Asking for information about the enclosures
- Looking for restaurant or services
- Leaving opinions and suggestions to the park
- Receiving information about daily promotions
- Browsing over Internet

Currently the ZooBotanico Jerez App is working over ANDROID smartphones and only inside the park.

4 Socio-economic impact and societal implications

The concept and the project achievements offer different advantages for the management of SMEs like parks and zoos. At a first sight, such performance improvements will effect on economic aspects, but the technology can further serve as an enabler for new services and business models, e.g. in other SME types like B&B, museums, etc.. Such new possibilities will influence on the usage of the GEEWHEZ solution and on society.

4.1 REPERCUSSION ON BUSINESS AND SOCIETY

4.1.1 GEEWHEZ exploitation strategy

The GEEWHEZ consortium is formed by 11 partners:

- 4 SMEs – Zoomarine, Matemaitci, Gamma, Pentos
- 2 Public organisations – ZooBotanico Jerez, Stadt Nurenberg
- 5 RTDPs – Technovation, FAICO, UC3M, T-CON, Prominent

The owners of the solutions are the SMEs and with some small percentage also the 2 Public organisations. In Table 1 we reported the current ownership percentage related to each partner.

	Zoomarine	ZooBotanico Jerez	Tiergarten	Matematici	Gamma	Pentos
WMS	100%					
VSS	81%		19%			
ERP				48%	26%	26%
LS	7%	19%			59%	15%
MDW				100%		

Table 1: GEEWHEZ ownership

The entire consortium, during the project prepared a proper exploitation plan where they stressed the aspects related to the use of the GEEWHEZ solution after the end of the project. In the exploitation plan the consortium identify 2 different types of exploitation, as indicated in Figure 29.

Internal exploitation plan, it represents the direct advantages, produced by GEEWHEZ solution, to the SMEPs and OTHs in the consortium. In particular:

- **Zomarine** thanks to the:
    - Water monitoring system installation - Zoomarine increased the effectiveness of the work dedicated to the maintenance of chemical-physical and microbiological parameters for the welfare of the animals. With the GEEWHEZ solution now the procedures have been automatic and not manual

- Video surveillance system - Zoomarine can now detect intrusions in the area and thanks to the alarms sent, Zoomarine personnel is informed in real time if somebody is entering the area

- Leisure system application - Zoomarine can communicate directly with its own visitors. It means that the manager of the park can send messages, like promotions or change of schedules of the shows, to the visitors. It avoid the continuous printing of the paper version of the map of zoo or the agenda of the park. The application is also important from an educational point of view providing the opportunity to share, with the visitors, information and curiosity about zoo's animals

- ERP systems requires long personnel training session, e.g. NAVISION, SAP, etc.. Those systems previously used, have been terminated. The management software is now completely tailored on the necessity of the park and it is not a specialisation of a generic system

- **ZooBotanico Jerez** thanks to the:

  - Leisure system and to the Wi.Fi. network deployed in the park it can communicate directly to the visitors and it can provide new pack of services. The GEEWHEZ Wi.Fi. network allow the zoo to provide an access also to the internet transforming it an Internet enabled zoo, a Wi.Fi. hot spot

- **Stadt Nuremberg Tiergarten** thanks to the:

  - Video surveillance system the zoo can improve its animal monitoring activities focused on the identification of the welfare of the animals. The camera of the video surveillance system are used not only to detect intrusion or an escaping from the enclosure but also to track the movement of the animals in order to understand its condition

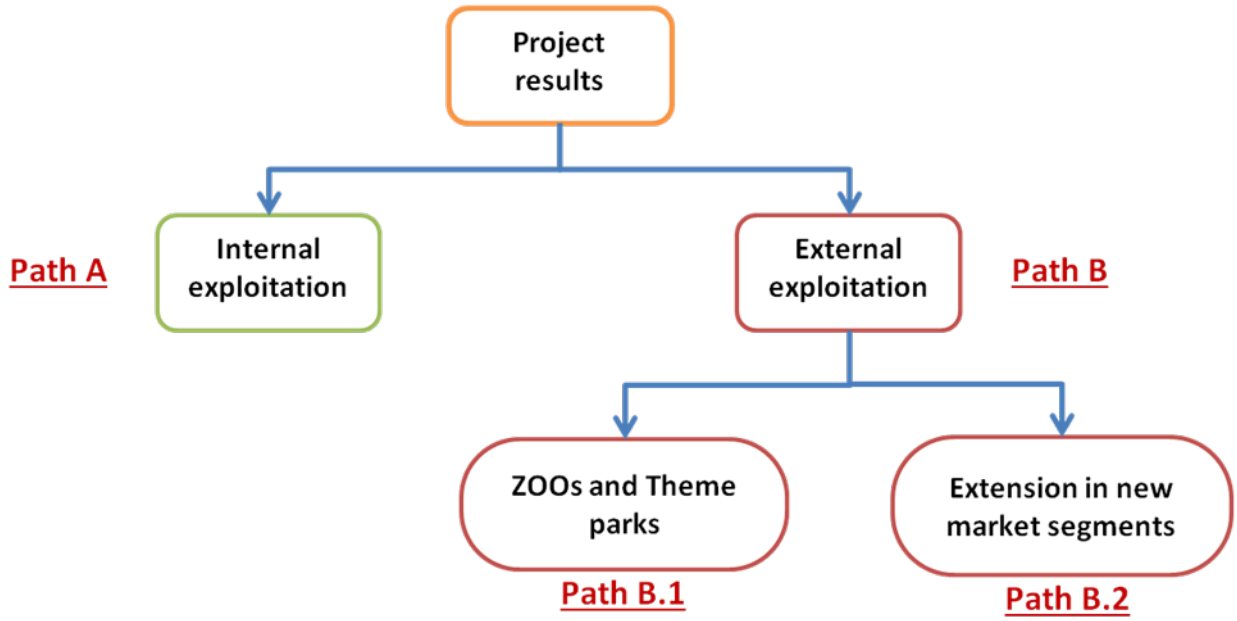

Figure 29: GEEWHEZ exploitation strategy

External exploitation is a plan to promote and commercialize GEEWHEZ results on the market following two different ways:

1. **Promotion and commercialization** of the GEEWHEZ modules toward theme parks and zoos. Actually, the consortium have a set of results completely tailored for such kind of customers that represent the target of the exploitation at this level

2. **New business lines** based mainly on the industrial development of middleware, ERP and Leisure System in other context like: museums, hotels, hostels, bed and breakfast, MICE (Meetings, Incentives, Conferencing, Exhibitions)

Matematici, Gamma and Pentos represent the three SMEs interested in the software development and in the extension of GEEWHEZ solution. As indicated before GEEWWHEZ is a specific solution for SMEs like parks and zoos; in order to obtain the maximum revenues from the activities performed so far the three SMEs are working together to promote the results of the project:

- At European level – they are trying to present an extension of GEEWHEZ project to B&B and museums

- At national level - they contacted several stakeholders that could be interested in the implementation of the solution:

    - Matematici is in contact with CINECITÀ Studios

    - Gamma is in contact with an hostel chain in Spain

4.1.2 Societal implications

Thanks to GEEWHEZ the zoos and parks will become places where people can still have fun but at the same time they can have training about animals, plants, history, science, etc.. The use of GEEWHEZ platform, and thanks to the cooperation of its several add-ons, will produce a value added information to visitors that will be aware about not only the presence of an animal but also a set of information like:

- The animal has been taken from its own environment or it is in captivity – this information can reduce the attack from those organisations fighting for the liberation of animals from zoos. It is a good action for sure but only for those animals that were not born in captivity

- The animal health status, continuously monitoring the movement of animals inside its enclosure can provide information about its health (repeating continuously the same movement can be an indicator of an animal in bad condition)

- Where the animal came from, e.g. if it is a guest of the zoo or not. Generally a guest is an animal that has been moved inside the zoo only to take care about it. After the hosting period it will be released to its own habitat

- The animal preservation, if the animal is one of the last members of its species and the reason why it could be better to stay in the zoo instead of living free

The above quoted reasons represent a set of information available on the GEEWHEZ platform that have an impact on the society explaining the importance of zoos and parks not only like amusement places but also as areas where animals can be treated and monitored.

4.1.3 Dissemination activities & PR material

As indicated during the project, a huge amount of dissemination activities have been carried on in respect to the "Interim Plan for the Use and Dissemination of the Knowledge". Among these activities the most relevant ones were the publications of three articles over accredited journals, participation in conference both on:

- ICT themes like the SDP Global Summit in 2012 where the consortium presented GEEWHEZ and how other system could interact with it
- Advancement in zoos and parks management like AIZA Conference 2013

Brochures, leaflets, posters, press kits and videos have been created especially for the demonstration events arranged at the end of the project.

Coordinator:	Claudio Di Capua
Address:	Via Casablanca, 61 00040 - Torvaianica – Roma
Phone:	+39.06.91533
Fax:	+39.06.91533 294
e-mail:	cdicapua@zoomarine.eu
Website:	www.geewhez.eu

Table 2: Logo and contact address of the coordinator

Zoomarine Italia SpA	Zoomarine	Claudio Di Capua Via Casablanca 61-00040 Pomezia Italy Phone: +39 06915331 mail: cdicapua@zoomarine.eu
PATRONATO MUNICIPAL PARQUE ZOOLOGICO DE JEREZ	ZOOBOTANICO JEREZ	Mariano Cuadrado C/ Taxdirt s/n – 11403 Jerez de la Frontera Spain Phone: +34956149784 mail: tecnicos.zoo@aytojerez.es
STADT NURNBERG	Tiergarten Nürnberg	Lorenzo von Fersen Rathausplatz 2 - 90403 Nurnberg Germany Phone: +39 06915331 mail: lvfersen@odn.de
Matematici	matematici	Massimiliano Carosi Via dei Fienili, 234 – 00049 Velletri (Rome) Italy Phone: + 390672293763 mail: carosi@matematici.com

TECHNOVATION SOLUTIONS LIMITED		Michalis Menicou Nikiforou Kato Lakatamia, 3 – 2333 Lefkosia Cyprus Phone: +357 22443433 mail: info@technovationsolutions.eu
FUNDACION ANDALUZA DE IMAGEN, COLOR Y OPTICA		Marcos Prieto Sánchez Calle Leonardo da vinci Parque cientifico y Tecnologico cartuja 93, 18 – 41092 Sevilla Spain Phone: + 34954467671 mail: mprieto@faico.org
UNIVERSIDAD CARLOS III DE MADRID		Mario Muñoz-Organero Calle Madrid, 126 – 28903 Getafe (Madrid) Spain Phone: +34916248801 mail: munozm@it.uc3m.es
T-CONNECT		Beatrice Pregarz Padriciano, 99 – 34149 Trieste Italy Phone: +39 0403755536 mail: beatrice.pregarz@t-connect.it
GAMMA SOLUTIONS		David Gómez Calle Benjamin Outram 4 Parque – 28919 Leganes Spain Phone: +34 91 525 88 00 mail: david.gomez@gammasolutions.es
PENTOS		Nikolaus Krasser Landsbergerstrasse, 6 – 80339 Munich Germany Phone: +49 8954 4937 30 mail: nikolaus.krasser@pentos.com
PROMINENT ITALIANA		David Munari Via Albrecht Duerer, 29 – 39100 Bolzano Italy Phone: +39 0471920000 mail: david.munari@prominent.it

Table 3: Project partners and contact addresses