Scientific
Research
Publishing

Web Development of Spatial Content Management System through the Use of Free and Open-Source Technologies. Case Study in Rural Areas

Ioannis Pispidikis, Efi Dimopoulou

School of Rural and Surveying Engineer, National Technical University of Athens, Athens, Greece
Email: pispidikisj@yahoo.gr, efi@survey.ntua.gr

Abstract

The rapid development of the technology of the web systems and GIS, in conjunction with the world economic crisis, formed the content for the promotion and development of free systems and open-source technologies. At the same time, the tendency toward the standardization of data, metadata and services, with the aim of creating common "Language" for the reading and the dissemination of information available, is a basic research area in the global scientific community in this field. The development of WebGIS systems, taking advantage of the free technology, also contributes to finding more economical solutions, where the use of such systems is more directly accessible. The aim of this research work is, through the analysis of technologies for the Internet, and also the architecture of the WebGIS systems, to investigate the possibilities and to develop the appropriate free technologies, so as to design and implement a spatial content management system for the web. Search with the use of the latter, is the best response to the needs and visualization application maps, with scope in rural areas. In addition, reference is made to the existing content management systems, which provide both processing spatial data, and easily create a WebGIS application.

Keywords

WebGIS, Open Source Technologies, Free Systems

1. Introduction

Geographical Information Systems (GIS) are information systems based on data management with spatial and

descriptive information. They are designed to support the collection, management, processing, analysis, modeling and imaging data referred in space and change over time. The main role of systems is to offer users powerful tools for solving complex spatial problems [1]. The Internet, even though created for the needs of the USA army, now forms an integral part of society. The widespread access to the Internet and the interactive content of the World Wide Web (WWW), have made this service a powerful means of exchange and management of information. Many applications in various fields have restructured and developed via the Internet [2]. The growth of the Internet affects the GIS in three different areas, such as the access of GIS data, the dissemination of spatial information and the modeling and processing of GIS [3]. The access and transfer of geographical data over the Internet is the first step in implementing a truly useful GIS system, where users will be able to work dynamically with the geographical data, without the need to install any specialized software. Independence from the latter was achieved through the establishment of appropriate services (OGC services), provided by the map servers. In addition, the technology AJAX (Asynchronous JavaScript and XML), which was conceived by Jesse James Garret, has strengthened the capacity of online systems, significantly reducing the burden on the Server. A major advantage over the use of this technology is the fastest response of the interface, to provide for the possibility for real-time applications [4]. Therefore, the evolution of the Internet and the development of technologies that support the creation of online systems, have led to WebGIS applications where through standardization of data, metadata and services, the exchange and analysis of geographical information are easy, direct, economical and efficient. In addition, the need to create more economical solutions resulted in the development of open source technologies at all levels of architecture of a WebGIS system. In order to easily exploit these technologies that do not require specialized knowledge, Content Management Systems (CMS) were developed. Example systems with processing and analysis of spatial data are the OpenGeo Suite, which is a commercial product but implemented by free technologies, the Cartaro, which provides geospatial functions and Web services in CMS Drupal, the Map Guide OpenSource and the GeoNode.

Based on the fact that the basic architecture of a WebGIS system consists of three basic levels (Client-Server-Data Server), reference is made to the technologies used for the implementation of each one of them. In particular, the level of client utilized the AJAX technology, where through the JSON format data are received and sent to the Server, asynchronously in the background, without the need to renew the application [5]. In addition, it was based on the ExtJS Framework, considered to be the most suitable for the development of applications on the Internet [6]. Finally, web mapping capabilities were provided through the OpenLayers library. At the level of the server the web programming language PHP was used with web server, the Apache HTTP server and map server—the GeoServer. The latter is integrated with the auxiliary tool GeoWebCache, through which the response of the system becomes faster and more efficient and reduces the workload of the GeoServer [7]. At the level of the Data Server the database PostgreSQL was used and, specifically, the PostGIS, so as to have geographical capabilities [8].

The content management system that was created provides the ability to manage issues relating to processing user requests and maps. In addition, it supports the organization and transfer of spatial data in the database and the GeoServer, as well as the processing in real-time via WFS-T (Web Feature Service-Transaction) which supports the GeoServer. The aim of the system is to be used to resolve bureaucratic and time-consuming problems in supplying land distributions and land reform maps in rural areas. In addition, it covers the requirements of both the administrator and the user, while its installation is direct and easy to use.

2. Architecture of WebGIS

By the term architecture, we mean all mechanisms, libraries and software to be used in a geographical information system to be complete. A WebGIS is composed of three basic levels; the first level is the client, while the second level consists of a team of servers and software, operating on the same or different server. The software called server-side software actually activates the use of server-side web programming languages (Java, PHP, Python, etc.). As server group we mean the web server and map server, and finally, the third level consists of the data server, including the database.

2.1. Client

The client is a browser which has each user on the computer and is necessary for browsing the web, providing communication between the user and the web server. The communication with the user comprises the web page

and asynchronous data exchange via the client-side web programming languages (JavaScript). When a user enters an address, the client receives the request and via the HTTP protocol sends it to the corresponding web server where this website is stored. The web server responds to the request by providing the web page requests.

For the asynchronous exchange of data between the client and the user, scripts are used written in the client-side web programming languages. The scripts are mostly functions of JavaScript, which are carried out and modify accordingly the web page with the implementation of an event. An example event is the click of a button (onclick), whose syntax is shown below.

```
<input name= "example" type= "button" onClick= "function()
" value= "event" />
```

2.1.1. AJAX (Asynchronous JavaScript and XML)

AJAX is used to describe modern technologies, techniques and methods, which are not necessarily related to the components of the AJAX (Asynchronous JavaScript and XML). The basic idea is to no longer need to renew the entire page, in order to send data to the server. In particular, the whole process is asynchronously processed in the background, while renewing only the part of the page that needs updating. The operation of the AJAX is based on the JavaScript XML Http Request [9].

Main advantage of using AJAX is that only the necessary data are sent and received and therefore the traffic of the server is reduced, since only part of the requested page is renewed. Finally, another advantage of its use is the fast response of the interface, as well as the possibility to provide online applications [4].

The exchange of data via AJAX is carried out using various formats. The most commonly used are summarized below:
− HTML

The most common format for the exchange of data via AJAX; it makes it easy and simple to update section of the page, by simply assigning the received data to the inner HTML attribute of a web page element.
− XML

It consist the main data-exchange format for which the techniques of AJAX were conceived.
− JSON (JavaScript Object Notation)

This format started to gain ground in recent years. Compared with the XML, for the same data, less space is required, while it is easier to parse the data.

More formats of data transfer are also used, such as the JavaScript and CSV (Comma Separated Value) [4].

2.1.2. DOM (Document Object Model)

The Dom is an independent platform and language convention, which provides a structured representation of the XML and (X)HTML document and specifies how this can be accessed from any programming language, so that it is possible to change the structure, the content and the style of the document. The structure of the document is represented as a tree composed of elements (element nodes), text (text node), properties (attribute node) and comments (comment nodes). The root of the tree is a document node, while the nodes are governed by hierarchy relations. The first element node of the tree is called root node and each element except the root has exactly one parent. The elements that have the same parent are siblings and descendants of the parent element. In the following hierarchy example (**Figure 1**), the head and body elements are siblings, descendants of the HTML, while their children are the elements with which are put together [10].

On the basis of the hierarchy of the DOM tree, the various scripts choose the elements on which they want to act. The selection methods are presented in **Table 1**.

Finally, a standard feature that is achieved through the DOM and in particular the document node, is the creation and deletion of data from the website, as well as the modification of the characteristics of existing data with the use of client-side programming languages, such as JavaScript [10].

2.2. Web Server and Application Server

The Web server is the software that responds to requests from the client via the HTTP protocol. It is designed to respond effectively to requests of a large number of clients and send static files. The request is sent via HTTP on
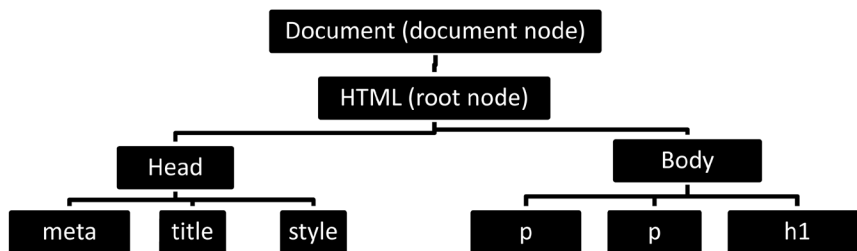
Figure 1. Example of DOM tree.

Table 1. Methods of data selection.

Selection Modes	Selection Method
Id	Document. Get Element by Id ("Contents")
Tag	Document. Get Element by Tag Name ("p")
Name	Document. Get Element by Name (Name)
Class	Document. Get Element by Class Name (Class Name)

the Web server, who is looking in the files of the hard drive of the server, if this file exists. A file will not be found in the server when not spelled correctly or when the request is not given with the correct path. If the Web browser is not configured to display a particular file type, then the user is asked to save the file on his local drive.

The most popular Web servers that are in use today are the Apache HTTP server and the Internet Information Services (IIS). The Apache runs on all modern operating systems, including Windows, Linux, Mac OS X and Unix. It is released under the license of Apache software, as open source software. It is serviced by an open source community and supervised by the Apache Software Foundation [11].

The weakness of the web server to manage and return only static documents comes to cover the Application server. It manages both the dynamic content of web pages through the supported script engine, and the simultaneous requests from users. Therefore, the Web server expects from the application server to return the result of the dynamic content and it in turn to do the job of returning the final static file to the client. Through the application server it becomes possible to communicate with the database and with other servers, such as the Map server.

2.3. Map Server

The Map server is a type Application server with manageability, processing and visualization of spatial data. The main feature of Map servers is the acquisition of spatial data from a spatial database and their dissemination on the Web, by using appropriate geospatial standards and services. A Map server may be Web server, so called web-mapping server or installed on a web server that has the appropriate server-side programming languages to support [7].

The most popular web-mapping servers used are the GeoServer, the MapServer and the ArcGIS server. The first two are open source web-mapping servers, while the third is commercial.

2.3.1. Web Map Service (WMS)

Through the WMS service, georeferenced images are available online only for viewing and not for further data processing. The client sends a request to the map server, and, based on the parameters of the request, the map server generates the final image and answers the request. The image produced by the WMS is the final product of the request and is provided by vector or normalized data or by a combination thereof.

The request for a WMS standard must be either in the form Get Capabilities or Get Map. The first one is used for the metadata of spatial data in XML format document. The second returns as result a georeferenced image. Such a request is considered as complete only when it includes information concerning the request, the name of the layer as recorded on the map server, the style of the layer, the reference system, the search limits and ulti-

mately the size and format of the final image created.

The Get Feature Info is optional request for searching information on the elements of a map produced by the WMS service on the basis of the pixel (i, j) value of the image [12].

2.3.2. Web Feature Service (WFS)

The WFS service is used for the direct use of vector data, returning the actual geometry and characteristics of the latter. For the description of spatial data the language GML (Geographic Markup Language), which is an extension of XML, is used.

The main demand that contains a service WFS is in the form Get Capabilities, Describe Feature Type, Get Feature, Lock Feature and Transaction. Through the Get Capabilities the user receives the metadata of spatial data in XML document format.

```
Http://example.com/geoserver/wfs?
  service=wfs&
  version=1.1.0&
  request=GetCapabilities
```

With the Describe Feature Type appear in GML format, information on either a layer or for a specific element of the layer.

```
Http://example.com/geoserver/wfs?
  service=wfs&
  version=2.0.0&
  request=DescribeFeatureType&
  typenames=namespace:featuretype
```

With the Get Feature request return to the user the real vector data with their geometry and their descriptive features.

```
Http://example.com/geoserver/wfs?
  service=wfs&
  version=2.0.0&
  request=GetFeature&
  typenames=namespace:featuretype&
  featureID=feature
```

The above example is a request via HTTP for the feature of a specific Feature type layer, as created and registered into the GeoServer.

The Transaction request provides the ability to create, modify, and delete data which have been published through the WFS service. The WFS service which supports this request is called WFS-T (Web Feature Service-Transaction). Finally, the Lock Feature is used in order to protect the data from the request WFS-T [13].

2.3.3. Web Coverage Service (WCS)

The WCS concerns the WFS service for the normalized data (raster). It is a service which makes possible to access grid coverage's data online. The grid data refers to satellite imagery, digital aerial photographs, digital terrain models and phenomena that can be represented by values at each measuring point. The request forms supported by the WCS service are the Get Capabilities, the Describe Coverage and the Get Coverage. The first request returns the metadata of this spatial data. The Describe Coverage returns in XML document format the full description of coverage request. Finally, the Get Coverage returns the final request in image format.

It should be noted that there should be no confusion between the two services WMS and WCS. Via the WMS service the spatial data is visualized in raster format. The end result is an image, in which there is no possibility

for further analysis and processing. On the other hand, with the WCS, the image is a grid coverage, which the user can access via the requests supported by this service [13].

2.3.4. Web Processing Service (WPS)

The WPS service provides GIS functions to clients on the web, including access to planned in advance calculations and computational models, which are associated with spatial data [14]. It can provide simple calculations, such as the definition of a buffer zone or complex, as a model of global climate change. The WPS aims to edit both vector and raster data. It should be noted that with this service successive algorithms and procedures apply on some data. Therefore, the results will assist decision-making in complex problems. The requests that may be created by the WPS service are the Get Capabilities, the Descripe Process and the Execute. With the first request form, information about the WPS service is acquired in XML format. With the Descripe Process, information about the procedure that has been selected to run is provided. In particular, the XML format contains information relating to the name of the process, the input spatial data, the parameters and the format of the data output. Finally, through the Execute request, the procedure provides the final result of the processing of the input spatial data [14].

2.3.5. Geography Markup Language (GML)

The GML is an extension of XML language and is defined in order to express and transfer geographical features on the Internet. It is an international standard that is able to integrate not only vector features but also rasterdata and data from sensors. The possibility of integrating of the latter, implemented with the GML version 3.0, is important for its usefulness.

The GML encodes the geometric objects as elements in a document. The types of geometric objects that support the versions GML 1.0 and GML 2.0 are points, lines and polygons. The GML version 3.0 incorporated new structures, which support coverage information, normalized structures and sensors' data.

The features of GML represent physical entities, which may contain geographical and/or descriptive data (e.g. rivers, bridges and buildings). Features describing the physical entity of an element should not to be confused with the geometric objects, since the latter define a location or area. Therefore, a feature may comprise one/ more or not any geometric objects. Such a physical entity that may contain more than two geometric objects is a building described by its location and surface area.

The geometric objects are identified by coordinates, which are recorded within the following tags:

```
<gml:coordinates panel> 30.45, 54.21 < /gml:coordinates panel>
<gml:pos> coordinates panel  < /gml:pos>
<gml:posList> coordinates panel  < /gml:posList>
```

With the use of the label <gml:coordinates>, as shown in the example above, it is not possible to access the coordinates separately (e.g. 30.45), given that the contents of this label is a string. From the GML version 3.0 and then, labels <gml:pos> and <gml:posList> are used for recording coordinates in order to allow accessibility of individual coordinates via the XML DOM.

The CRS determines the reference system of the coordinates of the geometric objects of a GML document.

```
<gml:LineString gml:id= "p21 "srsName= " / .. /EPSG/ 4326 ">
<gml:posList>45.67 88.56 55.56 89.44 < /gml:posList>
 < /gml:LineString >
```

In accordance with the above, a line geometric object was created, of which the reference system is determined through srsName feature. The value of the latter is a URL of an XML document of the specific reference system based on the EPSG (European Petroleum Survey Group) codes. In particular, the code 4326 defines the WGS84, while the corresponding XML document can be accessed through the website [15].

2.3.6. KML Encoding Standard (KML)

The Keyhole Markup Language (KML) language is an XML which focuses on the visualization of geographic

information, including notes on maps and images. The geographical visualization includes, not only the representation of spatial data on the globe, but also the control of the user's navigation. Initially developed to be used with the Keyhole Earth Viewer (by the Keyhole company, funded by CIA), which in 2004 was acquired by Google. The application has been further developed known as Google Earth, which is currently very popular, with millions of users worldwide. Given this popularity, the KML has been widely spread and has nowadays been incorporated in many geographical information management systems, as a means of interconnecting with Google Earth. In 2008 the KML has been approved as an international standard by the OGC. The KML is complementary to most standards of OGC including the GML (Geography Markup Language), the WFS (Web Feature Service) and the WMS (Web Map Service). The current version (2.2) of the KML incorporates geometry elements of the GML (v2.1.2). These include points, lines, line strings, linear rings and polygons.

The basic structure of a KML is based on the general principles of the XML (Extensible Markup Language) language, which uses tags in order to determine the beginning and the end of the different properties included in the file.

```
<kml xmlns= "http://www.opengis.net/kml/2.2">
<Document>
DEFINITION OF GENERAL PROPERTIES
<Placemark>
DEFINITION OF SPATIAL AND DESCRIPTIVE DATA
< /Placemark>
< /Document>
< /kml>
```

The label: <kml xmlns= "http://www.opengis.net/kml/2.2"> is placed at the beginning of the KML and indicates that the file will follow the rules of the KML (v2.2) standard of the OGC. All data, together with the definition of the general properties, are registered within the label <Document>.

Within the label <Placemark> the descriptive and spatial data of a geometric object are placed. The latter may be a point, a line or a polygon.

```
<Placemark>                                                    Point
<point><coordinates>........< /coordinates>< /Point>
< /Placemark>
<Placemark>
<LineString><coordinates>.......< /coordinates>< /LineString>
< /Placemark>                                                  Line
<Placemark>
<Polygon><outerBoundaryIs><LinearRing><coordinates>.......</coordinates></LinearRing><
/outerBoundaryIs>< /Polygon>
< /Placemark>                                                  Polygon
```

It should be noted that the coordinates of each geometric object are defined within the <coordinates> label [16].

2.3.7. GeoJSON

The GeoJSON [17] is an encoding format of geographical information, based on data transfer standard JSON (JavaScript Object Notation). Compared with the two previous forms, the GeoJSON is not considered by OGC as an international standard but as the result of a web developers' team. A GeoJSON may represent geometry, a feature or a collection of features. In a GeoJSON an object consists of a set of parts, which are described by a name and a value. The name is always a string, while the value may be string, number, object, table, true/false or null.

The geometrical objects which support a GeoJSON are points (Point), lines (Line String), polygons (Polygon), multiple points (multipoint), multiple lines (Multi Line String), multiple polygons (Multi Polygon) and collec-

tion of geometric objects (GEOMETRYCOLLECTION).

A feature in a GeoJSON contains a geometric object together with its properties. In particular, a feature is described by two pairs of names and values. The first is the geometry, which takes the elements of a geometric object and the second is the properties, where the feature attributes are recorded.

```
{ "type": "Feature",
 "geometry": { "type": "Point", "coordinates": (102.0, 0.5) },
 "properties": { "prop0 ": "value0 "} }
```

The reference system in a GeoJSON is determined through the "crs" object, which is described by two names. The first is the type where the EPSG value is assigned, while the second is the properties which consists the subject where various pairs of names and values are assigned. The most important is the code, which takes as value the corresponding EPSG code of the reference system. If the reference system is not defined, then the WGS 84 is considered as default. It should be noted that depending on the reference system, the description of the crs object can be found in the website:

```
"crs": {
    "type": "EPSG"
    "properties": {
      "code": 4326
    } }
```

http://spatialreference.org/.

2.4. Data Server

The Data server distributes the data in a database. Usually, in case of spatial data, the database connects with the map server and the query is executed by the server. For non-spatial data, the acquisition of data from the database can be accessed through the Application Server and in particular through the available server side programming language. For the database querying, whether spatial or non-spatial, the SQL (Structured Query Language) language is used. The most popular databases used in web applications are PostgreSQL, MySQL and Oracle. The first two are open source databases, and Oracle is a commercial product. MySQL is not suitable for spatial data, while PostgreSQL with the integration of PostGIS spatial database is considered as the most appropriate.

3. Design and System Function

Within the content of this research work, an online system was created, based on free systems and open-source technologies, consisting of three discrete subsystems. The first one is used for the easy and quick implementation of the necessary elements of the system. The second is a spatial content management system, and the third was created for web querying of information that relates to land consolidation or distribution work for a specific parcel.

The development of the application was based on the ExtJS Framework, considered to be the most suitable for the development of applications on the Internet [6]. Finally, web mapping capabilities were provided through the OpenLayers library. At the level of the server the web programming language PHP was used with web server the Apache HTTP server, map server the GeoServer and Database the Postgre SQL/PostGIS (**Figure 2**).

3.1. Methodology

The methodological approach adopted in order to implement the system consists of three stages. In the first stage the system's requirements were reported with regard to the users' management, the personal data, the GeoServer, the need for determining the competence of the application area and the data regarding the web request of the citizens. Then the appropriate tables were created in the spatial database PostGIS and exported into
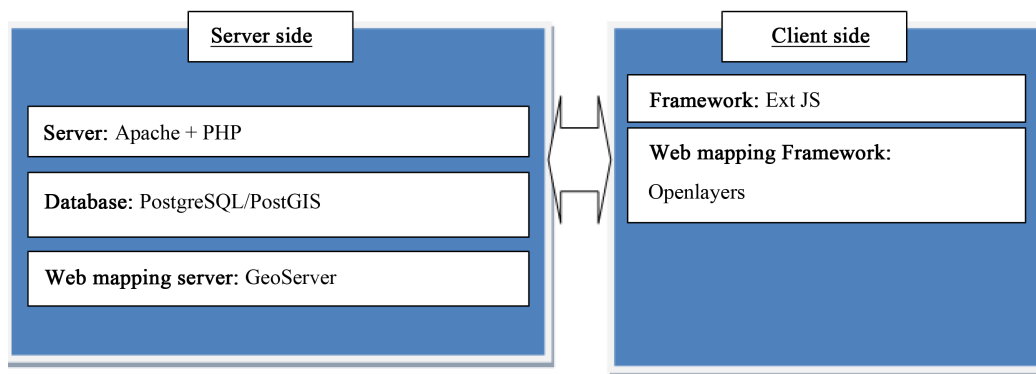
Figure 2. System architecture.

"sql" file. In addition, the most appropriate free technologies that can be utilized at the various levels of the architecture of the online system were investigated. As a result, the development of the application was based on ExtJS framework and therefore the basic language of implementation is the JavaScript. It should be noted that in this Framework the AJAX technology is integrated. PostgreSQL/PostGIS was used as database, while to get data from it, the server-side programming language PHP was used. For the publication of spatial data on the Internet the Web-mapping server GeoServer has been used, while their web cartographic representation was through the OpenLayers. Finally, the Apache HTTP server was used as server.

In the second stage, the web application of the system's implementation was created. The user fills up tabs with data relating to personal information and data concerning the connection to the spatial database. Then, by using this information and the "sql" file, the appropriate tables are created in the database; at the same time, a "php" file that contains the login information to the database is also created. In addition, implemented installation of GeoServer with the following plugin: geoserver-xx-SNAPSHOT-printing-plugin, in order to be able to print data in pdf.

In the third stage, the interface was created both for the spatial content management system and the web application for the online users' requests. The first application was created in a way to be connected with both the database implemented in the preceding step and the GeoServer.

The system thus created (**Figure 3**) offers the user the online data search concerning a land parcel. By implementing this application, an automated and friendly system is created both for the citizens and the surveyor engineers that request data on parcels that are included in a land consolidation or distribution project area. The WebGIS application consists of two parts; the right part shows is complemented by the citizen, while in the left part there is the geospatial map, with various tools for the correct completion of the form. At the top right of the window two more buttons provide communication and online help to the users.

The spatial content management system (**Figure 4**) consists of seven main sections for the users' management: the web-mapping server GeoServer, the online applications, the contact information and the processing of layers. Access to the system is provided via the username and the password created during the installation process.

3.2. Case Study

Poros of Evros Prefecture, located in the region of Eastern Macedonia-Thrace, was chosen as study area. Evros Prefecture is considered as a plain with only 10% mountainous, while 62% is covered by a fertile plain related to the Evros River with its streams and dense irrigation system. From the early 20th century, many land distribution and re-allotment projects took place in the study area, in order to deal with the problem of land fragmentation.

3.2.1. Materials
The choice of this particular region was based on the availability and accessibility of distribution and re-allotment maps to be used for the application procedure. The available data are in raster format, georeferenced in the Hellenic Geodetic Reference System HGRS87. The whole application process is based on raster data, although the spatial content management system created also deals with vector data.
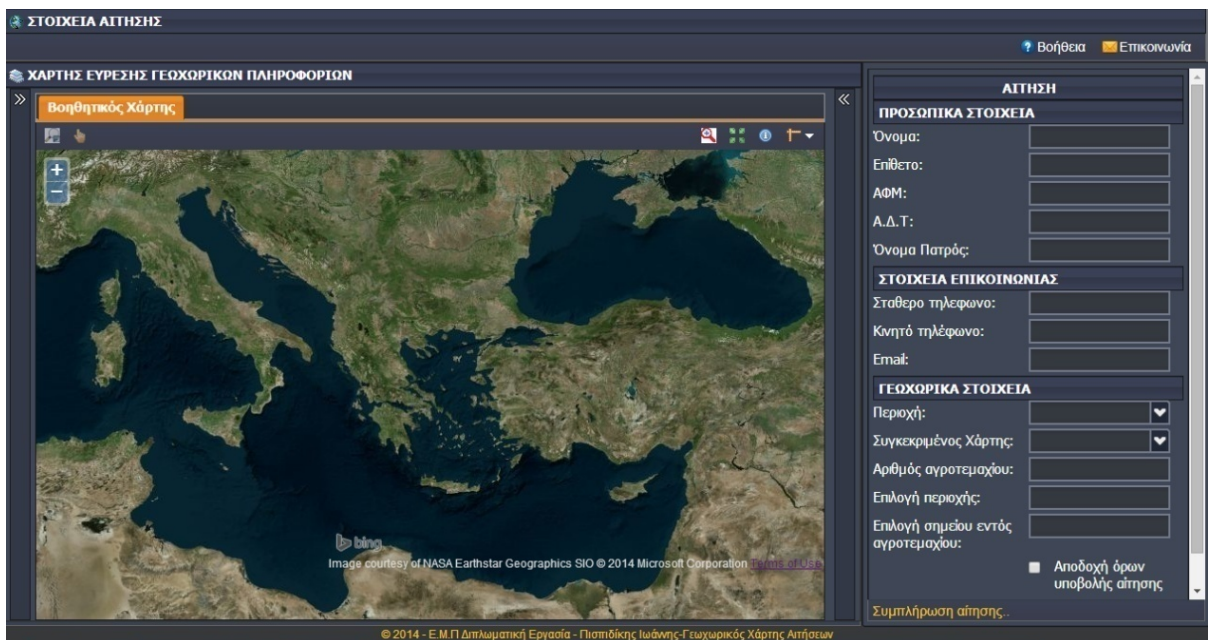
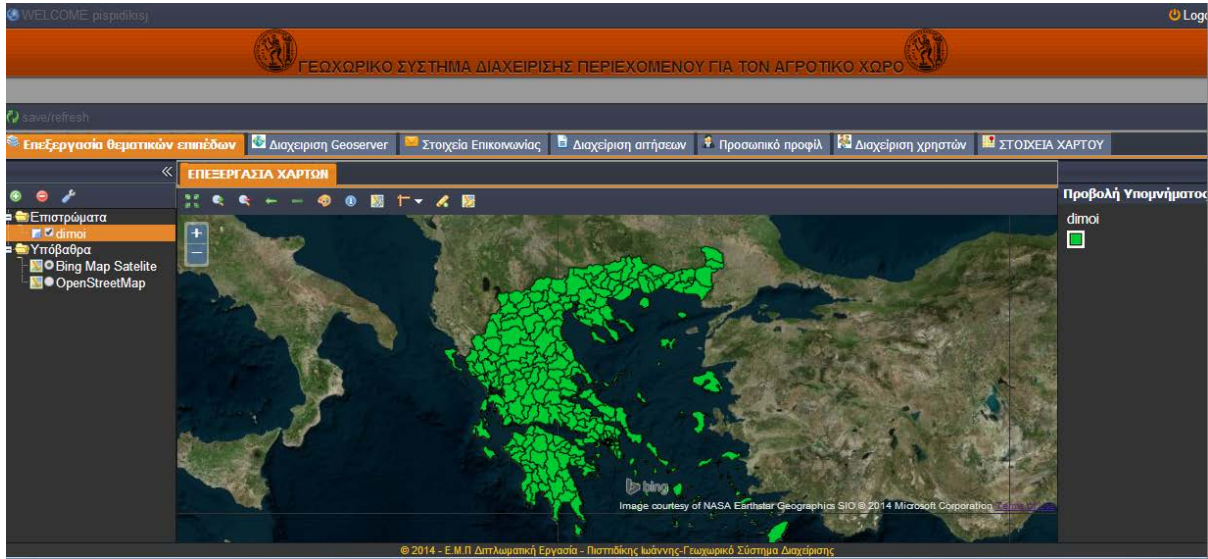**Figure 3.** Web application for the online users' requests.

Figure 4. Display of the spatial content management system.

3.2.2. Management System for Data Preparation

Having implemented the application, the administrator of the system makes the necessary updates by defining the limits of access, by starting up the GeoServer and by uploading the available maps for viewing (**Figure 5**). New users that are in charge for the processing of the citizens' application request may be added.

3.2.3. Application of Maps' Request

At first, the citizen can search for a specific land parcel by using the map application, which has as background satellite imagery provided by Bing (**Figure 6**).

It should be noted that, if digitization of maps was carried out, then it would be easier for the user to search information concerning specific land parcels based on pre-set queries in the database. In particular, by selecting a point within the parcel, the spatial database would be automatically queried for information regarding the specific
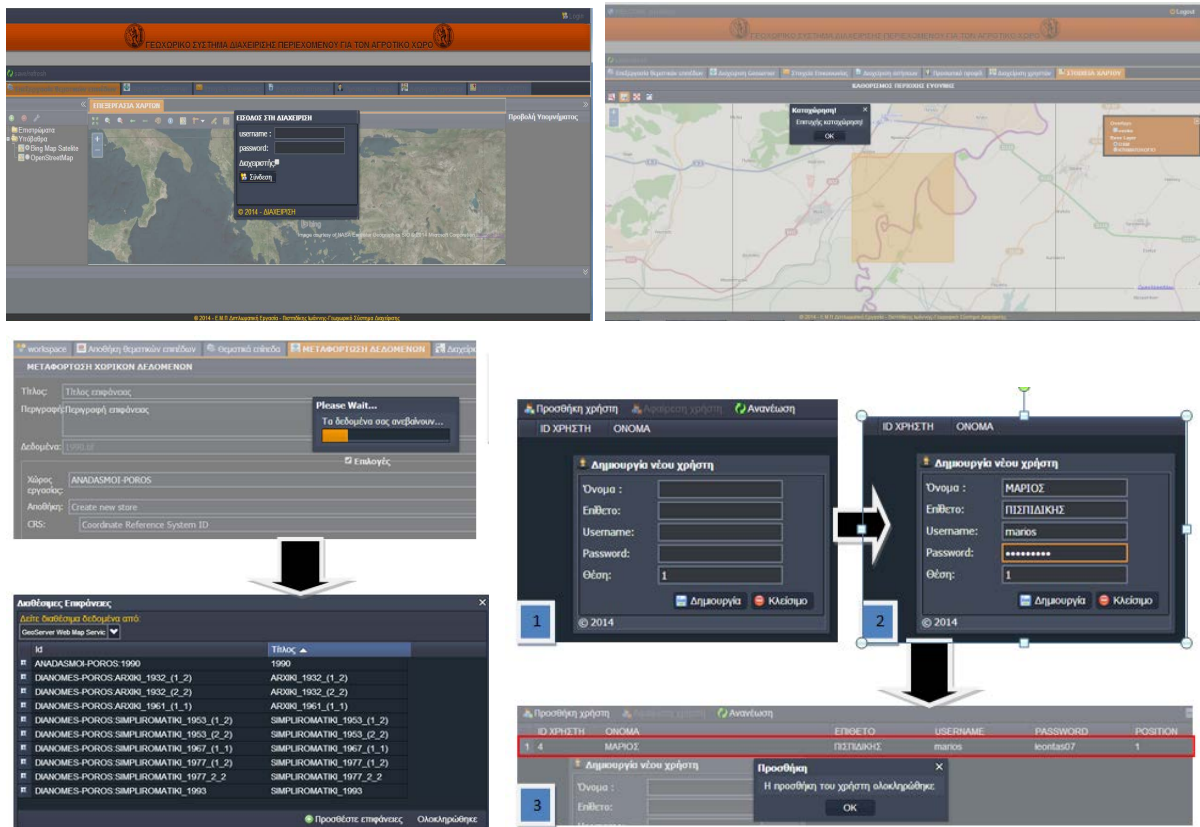
Figure 5. Initial update of the system (examples: login, specify area zone, data upload, creation of new user).

Figure 6. Example of a land parcel search.

parcel, thus, easing the citizen's search.

In this example the land parcel queried is located in an area of Poros where land consolidation occurred in 1990. The user by using tools (such as "transparent"), can easily detect the identification number of the land parcel which is 220. Then, the user completes the application. Personal information is filled up first along with contact details, and then, the geospatial data are added (**Figure 7**). After finalizing the completion of the application, the user has to accept the terms of validity required. Then, by activating the "Send request" button, the user can submit the final application. The completeness of the submitted application is checked, and the information is either transferred to the database or, in case of missing data, the user gets a relative notice.

3.2.4. Management System for Processing of Applications
When the user authorized by the administrator, chooses the "Management of applications" option of the CMS, the citizens' requests are displayed (**Figure 8**).
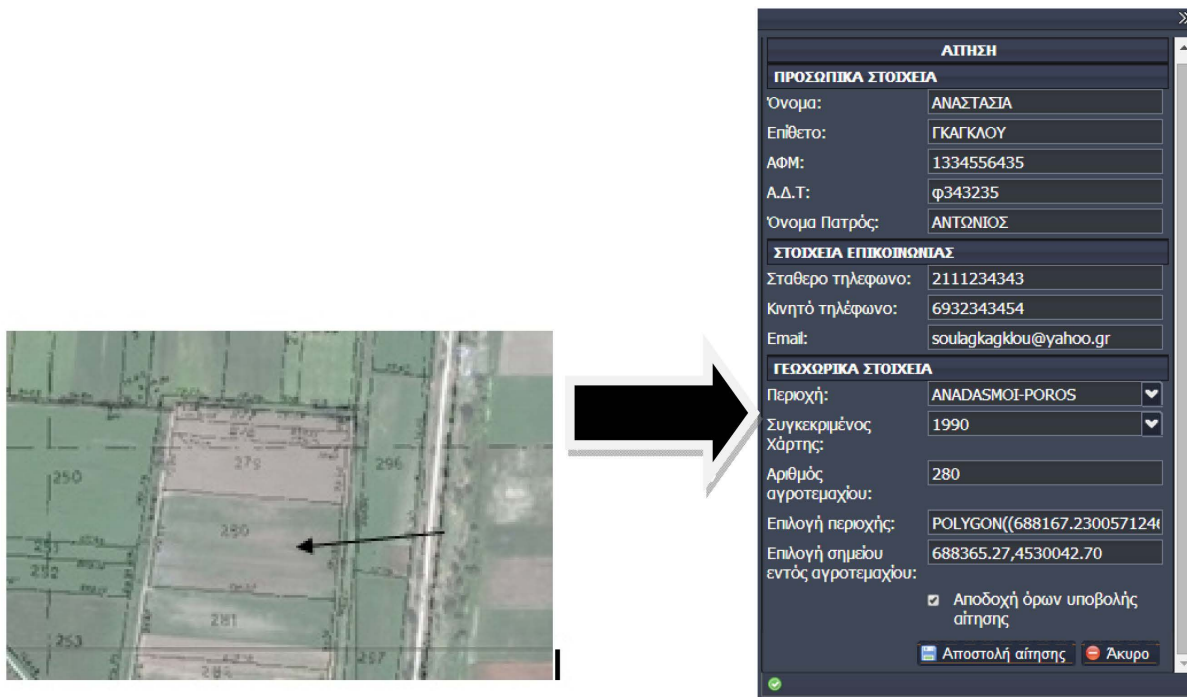
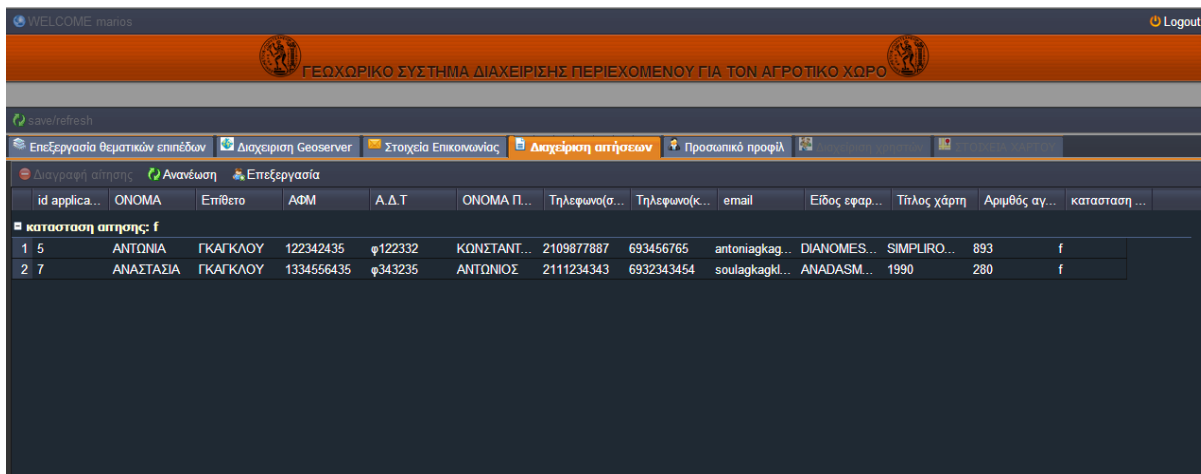**Figure 7.** Land parcel identification and completion of the application form.

Figure 8. Applications available within the spatial content management system.

Then, by choosing one of these two requests and by selecting the "Edit" button, the following WebGIS application is showed.

The user can visually select the area of the map to be extracted and choose its resolution in dpi units (**Figure 9**). Finally, the user may select the "completion of application" in order to close the relative window and update the database as well as the system that the user carried out the specific processing task.

4. Conclusions

The design and implementation of this system were based on state-of-the-art technologies through which, the application was smoothly and fast operational at all levels. In particular, the AJAX technology was used, and through the JSON format data are downloaded from the Server asynchronously in the background without the need to renew the application. Additionally the system was based on the ExtJS Framework, considered as the
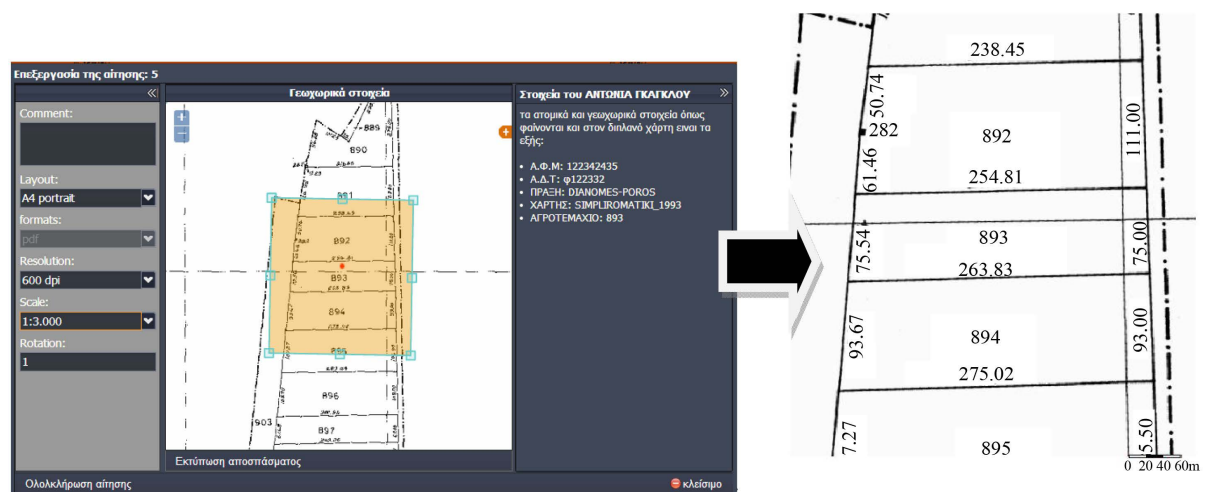
Figure 9. Generation of pdf with the extract of the selected area of the map.

most suitable for the development of web applications. The GXP library was also utilized, which includes a large number of cartographic tools for the Internet. By exploiting the GeoServer capabilities and the GeoWeb-Cache tool, the application became faster, since the response to the various spatial clients' requests was quick and efficient.

The implemented spatial content management system provides management capability in issues relating to the users' processing, the organization and transfer of spatial data in the GeoServer, as well as their online processing through the WFS-T supported by the GeoServer.

Some improvements that could be incorporated into the system in order to integrate it may include:

− Use of the GeoExplorer that provides additional possibilities to publish WebGIS applications according to users' needs.
− Further exploitation of GeoServer roles that are created to serve security functions. These roles are assigned to users/group and determine what functions are allowed in the GeoServer. Such roles are: the accessibility to various layers, the access to WFS services and the manageability of the GeoServer.
− Use of CSW (Catalog Service-Web) for database querying based on the metadata.
− Exploitation of the GeoServer capabilities, which is the available plugins, such as inspire, CSW, WPS and SQL server. The first two relate to the metadata. The WPS concerns the option of providing GIS functions including access to pre-calculated and computational models, which are associated with spatial data (raster and vector). The SQL server allows for the creation of SQL queries, based on pre-defined parameters for the specific spatial information.

Proposals for future development of the web management system described in this paper, both in theoretical and practical terms, may include:

− Further investigation of commercial and free spatial content management systems, as well as comparative survey investigation between them.
− Further investigation of commercial and free Map server with additional comparison between them.
− Economotechnical feasibility study for the creation of a web application to resolve a problem with commercial and free systems, including software cost, time and work required.
− Redesign of the database system, based on international standards, such as the LA.D.M. (Land Administration Domain Model).
− Knowledge exploitation, to investigate and implement web application for the Hellenic Cadastre.

References

[1]   Stefanakis, E. (2003) Supports Geographical Data and Geographical Information Systems. Papasotiriou Publication, Athens. (Greece)

[2]   Plewe, B. (1997) GIS Online: Information Retrieval Dialog Box, Mapping, and the Internet. Word Press, Santa Fe, New Mexico.

[3] Sotiriou, I.E. (2010) Comparison and Integration of WebGIS Software. Diploma Thesis, Geoinformatics, NTUA. http://dspace.lib.ntua.gr/handle/123456789/3179

[4] Athens University of Economics and Business (2011) AJAX: Applications on the Web.

[5] Darie, C., Brinzarea, B., Cherecheş-Toşa, F. and Bucica, M. (2006) AJAX and PHP. Building Responsive Web Application. Packt Publishing Ltd., Birmingham.

[6] Garcia, J. (2011) ExtJS in Action. Manning Publications, Stamford.

[7] OpenGeo (2012) Introduction to the OpenGeo Suite. http://presentations.opengeo.org/2012_FOSSGIS/suiteintro.pdf

[8] Kolios, N. (2009) Spatial Database Postgre SQL/Post GIS and GIS Quantum GIS. User Guide, Greek Free/Open Source Software Society (GFOSS).

[9] Woychowsky, E. (2006) Ajax: Creating Web Pages with Asynchronous JavaScript and XML. Prentice Hall, Upper Saddle River, 432 p.

[10] Marini, J. (2002) The Document Object Model: Processing Structured Documents. McGraw-Hill, New York.

[11] Apache Software Foundation (2010) Apache HTTP Server Version 2.2 Documentation. https://httpd.apache.org/docs/2.2/

[12] de la Beaujardiere, J., Ed. (2006) OpenGIS® Web Map Server Implementation Specification. Open Geospatial Consortium Inc., Wayland.

[13] Open Source Geospatial Foundation (2014) Geoserver User Manual. http://docs.geoserver.org/stable/en/user/

[14] Schut, P., Ed. (2007) OpenGIS® Web Processing Service. Open Geospatial Consortium Inc., Wayland.

[15] Portele, C., Ed. (2007) OpenGIS® Geography Markup Language (GML) Encoding Standard. Open Geospatial Consortium Inc., Wayland.

[16] Wilson, T. and Burggraf, D., Eds. (2009) OGC® KML Standard Development Best Practices. Open Geospatial Consortium Inc., Wayland.

[17] Butler, B., Daly, M., Doyle, A., Gillies, S., Schaub, T. and Schmidt, Ch. (2008) The GeoJSON Format Specification: http://geojson.org/geojson-spec.html