An interactive Web-Based database of antibiotic resistance in bacteria in aquaculture systems: structure.

[1]M. Giacomini, [1]R. Michelini, [2]S. Bertone, [1]F. Raffo, [1]C. Ruggiero

[1]Dept. of Communication Computer and System Science – University of Genova
Via Opera Pia 13 16145 Genova Italy
Mauro.Giacomini@dist.unige.it
[2]RI.L.A.B. srl Via Guerrazzi 24/12 16146 Genova Italy.

Abstract

The internet is the fastest growing tool and arguably the most popular method of accessing public domain information. Web-based applications will not only enable people to take advantage of the wide geographic coverage and low cost of the internet, but will also save time, money and training in the future as transition to a browser-based user interface standard continues. The presented application has been designed in order to collect data coming from aquaculture samples. It is part of the European founded project ASIARESIST (ICA4-CT-2001-10028). This project aims to isolate bacteria resistant to antibiotics in fish water and sediment samples coming from Malaysia, Thailand and Vietnam. We present the technical choices and specific implementation of the database and its interface. The final result of this system will be a steadily growing database monitoring antibiotic trends and dynamics across the industry and the environment of the south-east Asia region.

Keywords:

Database; Active Server Pages; Aquaculture; Data access; Flash; Internet; Antibiotics

1. Introduction

The internet is the fastest growing tool and arguably the most popular method of accessing public domain information. This New World has become an interactive environment where server, client, various devices and users are interacting at different levels, sharing software and business services.
One application area with great promise is web access to data. Using the internet brings the benefit of fast and intelligent data access and data storing to a whole new audience, opening up new possibilities for data sharing. Web-based applications will not only enable people to take advantage of the wide geographic coverage and low cost of the internet, but will also save time, money and training in the future as transition to a browser-based user interface standard continues. When choosing an architecture for a data storing and retrieval application, the designer has to face a number of issues that really guide or focus the development process. A client/server side application should be easy to develop, scale, maintain, deploy and use. A primary consideration in planning is identifying who will use the Web application: the audience. The Web application's audience can be defined by identifying those who have access to the Web application and identifying the type of Web browser they use. The capabilities of the Web browser also shape the plans for using client and server script in the Web application. Using server script and Active Server Pages (ASP), browser- independent pages can be easily generated. The server processes the server script and then sends HTML to the browser. In contrast, if the type of browser that the user

has is known, client scripts can be used to generate the page and minimize the load on the server.
Whatever implementation strategy is chosen for the client/server application, security must be considered. Data should leave the client and reach the server being safely protected from hacker's access. Commonly used methods to identify a user are: cookies, user-id and password or IP numbers.
This web-based application has been designed in order to collect data coming from aquaculture samples. It is part of the European founded project ASIARESIST (Hazard analysis of antibiotic resistance associated with Asian aquacultural environments - ICA4-CT-2001-10028 - INCO-DEV Program). ASIARESIST aims to isolate bacteria resistant to antibiotics in fish water and sediment samples coming from Malaysia, Thailand and Vietnam.

2. Materials and Methods

This web based application has been designed to collect data coming from biological sampling on fishes and their environment coming from three different countries: Malaysia, Thailand and Vietnam.
The main points we focused on were:

- Programming languages
- Database structure
- Controlling data access
- User interface

2.1 Client – server architecture

This architecture is based on the interaction of two computers to perform a task. The client requests the information needed to the server, which sends it back to the client. The Internet is based on a client – server model. In that case the server is a Web Server. It is a computer containing all the web pages of a web site. The client is a web browser, which receives the HTML pages sent by the server and shows it on the user screen [1].
In our application the web server resides in the Bioengineering and Medical Informatics laboratory of DIST (MedInfo - DIST - University of Genova) while all the other partners play the part of the clients connecting to it every time they have new samples to store into the database. To avoid problems related to client scripts (different type of browser and different versions of browser software) we used only server-side scripting implemented using Active Server Pages.

2.2 Active Server Pages

Active Server Pages (ASP) enables Hypertext Markup Language (HTML) authors and Web developers to mix HTML and inline scripting using any authoring tool. The scripts can reference components running on the local server – or any other server – to access databases, applications or process information. When a browser requests an ".asp" file, the web page is processed by the server and it is returned to the client as plain HTML (fig.1).
ASP is a server side scripting language, which exists only on the server and cannot be visible to the clients. For this reason ASP provides a browser independent approach to application design [2].

Active Server Pages program code can be written using different scripting languages such as VBScript, Jscript or PerlScript [3].
We chose to use VBScript as its syntax is similar to Visual Basic language and our group was already familiar with that language.

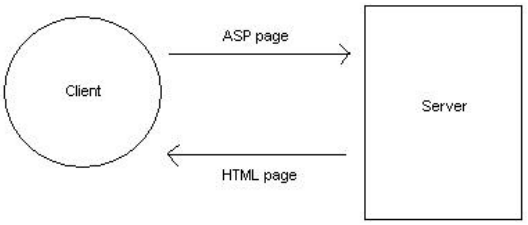

Fig. 1: ASP/HTML process

2.3 Microsoft Access Database

Microsoft Access has built a tradition of innovation by making historically difficult database technology accessible to business users in general. Whether users are connected by a LAN, the Internet, or not at all, Microsoft Access ensures that the benefits of using a database can be quickly realized. With its integrated technologies, Microsoft Access is designed to make it easy for all users to find answers, share timely information, and build faster solutions.
At the same time, Microsoft Access has a powerful database engine and a robust programming language, making it suitable for many types of complex database applications.
We organised a powerful relational database ready to host data coming from the project.

2.4 User recognition

There are three main ways of recognising a user connected to a web site:

- Cookies
- User-id and password
- IP address

Cookies are text files of small sizes that are sent by the Web server and stored on the computer of the user. These files can contain various information such as a user id or a password and they can be read on certain events to retrieve user's information. The main problems related with cookies are that the web server identifies a computer and not a user and that the user can delete the cookie from the hard disk becoming unidentifiable during the next access.
The IP address is a number given by the provider of the internet connection to the computer that is connected to the network. IP addresses can be static or dynamic. An IP number is static when it does not change in time, that means that the computer has always the same IP address. On the other hand the IP number is dynamic when the provider gives the computer a different number each time the computer enters the network. Once again IP number does

not identify the user but only the computer connected to the server and if the computer has a dynamic IP address it cannot be recognised at all.
The best way to recognise a user and to limit access to the web site is to assign each user a "user ID" and a password. The server will keep a copy of them in a database or in a text file and will allow the user to enter the web site only if the one inserted matches with the stored one. At first we decided to avoid unexpected access to the web server by opening the access only to the IP numbers of the project's partners. We then discovered that in most cases their IP number was assigned dynamically and many of the Asian partners could not access the database. For this reason we decided to use once again the database for storing a user ID and a password for each of them.

2.5 User interface

A key factor in developing a web application is its purpose. After having clearly specified the Web application's purpose the features and functions that define the visitor's experience can be easily determined. One of the most common things one wants to do in a dynamic Web application is to interact with a user; this is commonly accomplished using the forms and functions provided within HTML. Tags are inserted into the HTML text to provide controls on the Web page. Using forms is an effective way of getting information from a user and passing it to another page for further interaction or to a database for recording purposes [4].
Our database has been designed to host data describing:

- Sampling site
- Fish farm
- Sampling period
- Sample description

Each partner should insert data of samples coming from three different sites in his home country and belonging to three different types of sample:water, sediment and organism.
As the amount of data is high and it is easy to get confused, we thought it would have been better to add some visual help to our partners by the use of Flash integrated with ASP and HTML.

3. Results

The aim of the Italian participation in the Asiaresist project is to build a Web based application made of an HTML graphical interface and a database residing on a web server for an easy insertion and maintenance of data coming from aquaculture samples.
The first part of the project is concerned with obtaining a collection of antibiotic resistant bacterial isolates derived from aquaculture sites of economic importance in Malaysia, Thailand and Vietnam. Each bacteria is then classified on the base of its country of origin, the farm in which the sample has been taken, and the type of sampling medium in which it has been isolated. There are three types of sampling medium for each farm: water, sediment and organism. At the end of the process of isolation, each partner gets 144 isolates from the three sampling sites chosen for his country. On the basis of data insertion and according to a Standard Operating Procedure for sampling agreed among all partners, followed by us in the creation of the database, each strain is given a code, which contains information about:

- Resistance / Non resistance to a certain concentration of antibiotic
- Country of origin
- Sampling Site
- Fish farm pond
- Sampling period
- Sample type
- Colony number

All the information is stored in the relational database, together with the strain code. In order to avoid excessive manipulation of data, each partner has the right of inserting data only into the database related to his/her country but once inserted, they can only be deleted by the database administrators. The overall structure of the database is shown in fig. 2 according to the Entity-Relation notification [5].

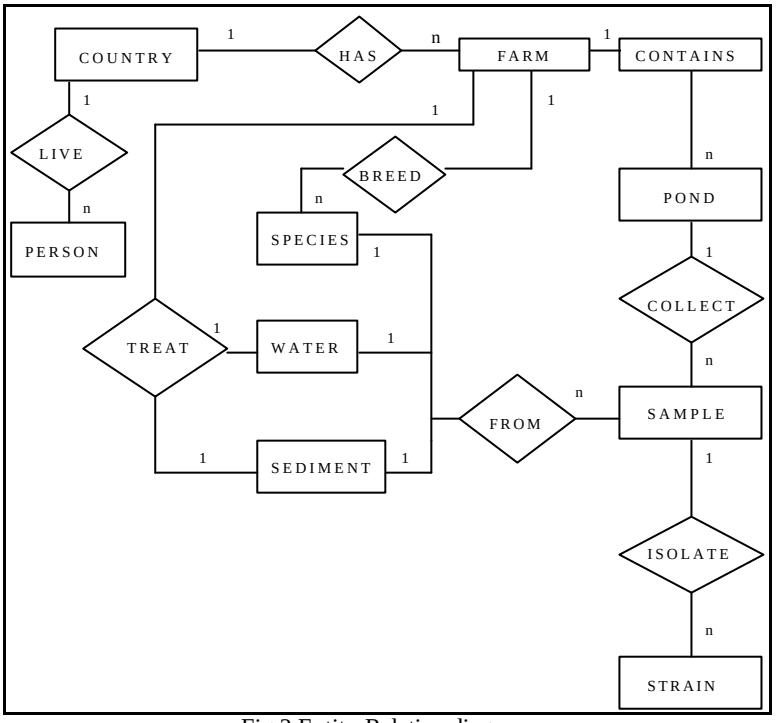

Fig.2 Entity-Relation diagram.

To obtain that and to avoid external access to data, the first page of the web site is a login page asking for a USER ID and a PASSWORD. The user is allowed to enter only if the inserted data matches with the ones in the USER table of the database. This method allows the server to recognise a user and to give him/her certain rights over data, for example the administrator can access data deletion pages, while the other users can only access data insertion pages with country restrictions. At the release the User database contains only the responsible of each country, so that they are the only authorised persons to access the database. They can authorise other persons, but only for manipulating data of his/her country. Once the bacterial strain isolation is completed, Antibiotic Susceptibility Test is

performed. This test indicates susceptibility of the organism to the antibiotic being tested by a clear zone of inhibited growth around the impregnated filter discs.
Due to the intrinsic visual nature of this testing we decided to develop a program with Macromedia Flash MX®, which resembles the dishes used in the experiments. Flash is an excellent tool for interactive application development in internet. It uses vector graphics, everything you see on your screen is displayed by a mathematical calculation, meaning that a Flash movie can stretch to fit any resolution and still look good (as long as you don't have any raster graphics such as jpegs in it), which has always been a huge barrier for HTML.
Each antibiotic (white circles) becomes a button. Once it is clicked it changes its colour to red and the user is asked to insert growth diameter and other data about the testing. When all data have been inserted and stored into database, the image will change to show the growth inhibition diameters previously inserted.

A B

Fig. 3: Antibiotic Susceptibility Test input form (A empty - B partially filled)

4. Conclusions

The present paper presents an application that joins together database, server-side scripts and HTML graphical interface into a comprehensive data storing system. The system has been designed in order to provide a user friendly interface for data insertion and a powerful relational database which, by the end of the project, will become a publicly accessible database for antibiotic resistance monitoring of environmental bacteria. This system has been designed to improve data interchange by the use of a standard communication protocol and by the setting up of common user friendly interfaces, in order to optimise the efficiency of data management. The final result of this system will be a steadily building database monitoring antibiotic trends and dynamics across the industry and the environment of the south-east Asia region.

5. Acknowledgement

This work is partially supported by the project ASIARESIST (Hazard analysis of antibiotic resistance associated with Asian aquacultural environments - ICA4-CT-2001-10028 - INCO-DEV Program), funded by European Union. We are grateful to our Asiaresist Partners in the Aquatic Animal Health Research Institute (Thailand), Can tho University (Vietnam), University of Ghent (Belgium), University Putra Malaysia (Malaysia) and University of Stirling (UK) for their agreement to this publication, and for their support.

6. References

[1] Darnell R. *HTML 4 Unleashed.* Macmillan Computer Publishing, 1997.

[2] Scott M, Atkinson J. *Active Server Pages 3.0.* Macmillan Computer Publishing, 2000.

[3] Goodman D. *JavaScript Bible,4th Edition.* The McGraw-Hill Companies, 2001.

[4] Crouch R, Amundsen M. *Database Programming with Visual InterDev 6 in 21 Days.* Sams Publishing,1999.

[5] Batini C, Ceri S, Navathe SB. *Conceptual database design.* Redwood City, CA – USA: The Benjamin/Cummings Publishing Company Inc., 1992