БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ • BULGARIAN ACADEMY OF SCIENCES

ПРОБЛЕМИ НА ТЕХНИЧЕСКАТА КИБЕРНЕТИКА И РОБОТИКАТА, **59**
PROBLEMS OF ENGINEERING CYBERNETICS AND ROBOTICS, **59**

София • 2008 • Sofia

On an Approach to the Design of Information Systems

Nikolay Dokev

New Bulgarian University, 1618 Sofia and Institute of Information Technologies, 1113 Sofia
E-mail: niki@iinf.bas.bg

1. Introduction

The topic of the present paper is the design and implementation of an information system for hotel rooms booking, applying the new vector standard SVG (Scalable Vector Graphics) of the W3C Consortium.

The main purpose of the system is to provide the users with a flexible interface for booking hotels, found in the available database. The hotel presentation must be accompanied by appropriate photos and a map of the town, where the hotel is located

The Web application, proposed with the purpose to design a reservation system with scalable vector graphics, employs client-server technology.

Some of the existing browsers and/or applications created with any of the existing technologies for the design of clients' applications, such as JavaScript, DHTML, Java applets and ActiveX controls, etc., are used.

The server creates a separate document depending on the data, provided by the browser and the users' actions in these applications. The representatives of server technologies are PHP, CGI, JSP (Java Server Pages), Servlets, ASP.

2. The SVG vector standard

With the process of Internet information systems globalization, demand arises more often among the users for improving the quality of information presentation. Until recently only the presentation of text and raster images, in which information is stored for each pixel, was standardized. This way of storing the image data has its shortcoming – the result from the compression sometimes requires very large storage capacity. The representatives of raster graphics are: JPEG format (Joint Photographic Expert Group), GIF (Graphics Interchange Format), PNG (Portable Network Graphics), Bitmap, TIF (Tag Image File), etc. The JPEG, GIF and PNG formats are the most wide-spread on the Internet. Thanks to the good data compression, they reduce the required storage memory to a minimum.

The very promising standard SVG was created and recommended by W3C (World Wide Web Consortium) at the end of 1998. The SVG format is based on XML (eXtensible Markup Language). SVG extends Internet graphics, enabling the creation of pure structured descriptions of vector graphics for Web, for geographic maps, with the combined use of raster-vector. In order to create SVG graphics, there exist some basic and complex forms, as well as backgrounds with color blending and many filters for additional effects. The graphics can be animated and also interactive, with the use of some scripting languages, like JavaScript. SVG completely supports the DOM Level 2 specification. DOM stands for Document Object Model. The purpose of the model is to present a document in a hierarchical structure. Thanks to the use of identical DOM API functions, the same syntax of JavaScript is used in SVG documents, as in HTML.

Due to the fact that SVG is text-based, SVG graphics can be authored even with the help of an ordinary text editor, like Notepad. The software market offers many commercial applications for SVG graphics development, which aid the work of the designers. One such product is WebDraw by Jasc Software (**http://www.jasc.com/**). It supports all standard shapes, backgrounds, filters and animation of elements by defining a time line. Another product is SVG Studio by PCX Software (**http://www.pcxsoftware.com/**). Since SVG finds wide application in cartography, a lot of specialized applications for maps creation in this format have also been developed.

For SVG support, plugins are still used by many browsers. Adobe is recognized as the standard vendor of SVG plugins as well. The Adobe SVG-Viewer has been installed on more than 50 million computers on the global network. It is preferred because of the numerous functions it provides, namely: Zoom in and Zoom out, animation stopping (Pause), searching for a given text in a SVG image (Find) and also displaying source code (View Source). Other plugins are IBM SVG View and Csiro SVG Toolkit. SVG is natively supported by big vector packages, such as Adobe Illustrator, Corel Draw and Macromedia Freehand. One document, containing SVG, could be created and stored in Corel Draw, then opened in Illustrator and at last published in Macromedia Freehand. The graphics remains the same.

The SVG vector standard has its child branches. One of them is Mobile SVG, with the following main areas: GPS – mobile navigation devices, multimedia messaging, industrial applications and others.

Table 1 presents a short overview and a comparison between Flash and SVG.

Table 1

Features	SWF / Flash	SVG
Official W3C standard	No	Yes
XML based	No	Yes
Data accessible to read and edit	No (binary)	Yes (text)
Size	Small	Small, allows archiving
Cascading Stylesheets support	No	Yes
Indexing by search engines	No	Yes
Validation	No	Yes, since it is XML based
Needs plugin	Yes	Yes, but browser integration is being worked on
Searchable documents	No	Yes
Server-side generation by scripting languages	No	Yes (Perl, PHP, Python etc.)

3. Design of the application

The goals of the information system being developed can be defined as follows:

- the application must store information about hotels with a possibility to provide the user with a reservation form for the respective hotel;
- it must allow filtering of the information, available in the database, according to certain criteria;
- the application must provide a possibility for registration and viewing of the already registered clients;
- it must allow adding new information and updating the stored information;
- the application must enable the use of the SVG vector standard for Internet graphics when presenting the photographic material about a hotel, as well as country and city maps;
- the application must be multi-platform.

The following requirements are defined to achieve correct functioning of the system:

- a large number of users must be able to use simultaneously the application.

Requirements on behalf of the users;

- any user, who is not registered, must be able to access the hotels information, and be able to register;
- the registered users must be recognized by the system by means of their user’s account and password;
- the users must be given the possibility to view the information available in the database, filtering it by country and city;
- the users must be given the possibility to search for a hotel by its geographic location (country, city), category or name;
- to facilitate the users in finding a hotel by marking its location on a city map;

- to provide a reservation form for booking a room (rooms) after selecting the desired hotel.

Requirements on behalf of the administrator:

- to be able to view all the reservations entered, the new or already expired ones being flagged as such;
- to be able to change the status of any reservation to approved one or to delete it from the database;
- to enter a new hotel or delete an existing one from the database;
- to be given access to the personal information of the users already registered in the system and to be able to delete any user from the system.

Taking into account the purposes and the requirements set above, the Web application “Hotel Guide” has been developed. The application ensures the necessary access for registration and recognizes the already registered visitors. It enables the users to view the content of the database, filtered by city and country or to search for a hotel by its name.

After the selection of a hotel among those available in the database, detailed information is displayed about that hotel and it is possible to fill in a room(s) reservation form for it. In case any user is not registered or recognized by the system, the filling of the form is disallowed and he/she is re-directed to registration or authentication. An opportunity is offered for the visualization of the hotel location on a map of the town, where it is situated.

From the viewpoint of the administrator, the system allows the review of all, or only of the new reservations with options of approval and confirmation of the reservations, requiring additional verification through the user's e-mail, or deleting and canceling the reservation. If there are any expired reservations, the administrator is notified to delete them from the database. It is possible to view the information stored about the hotels, to add a new hotel or delete an existing one. Access is also given to the personal information of the already registered users with the option to delete any user from the system.

3.1. Choice of the software tools for implementing the application

The developed “Hotel Guide” information system has a three-layer architecture comprising the following layers:

- First layer – in our case this is the client (the user’s Web browser). It can also be a Java applet, Java application or platform-independent client program.
- Second layer – this is the logic part of the application, namely the web-server and the scripts executed by it. The task of the second layer is to ensure communication between the client and the database, receiving the clients’ queries towards the database and after the necessary processing of the corresponding data sending the information back in an appropriate for presentation form.
- Third layer – this is the database server. It stores all the data, necessary to represent the system content.

Fig. 1 shows the three-layer architecture of the system.

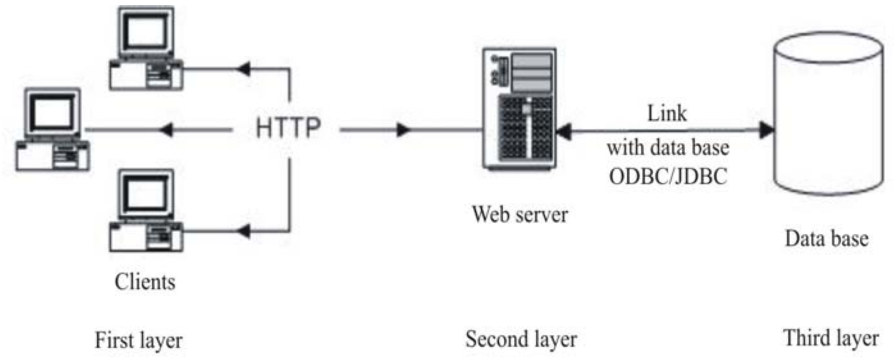


Fig. 1. The three-layer architecture of the system

The present Web application is implemented under Microsoft Windows XP operating system and Internet Explorer 6.0 browser, but the application can also be used with other browsers – Netscape, Mozilla Firefox, Opera or any other, because it is browser-independent.

In order to present the data in the client, processed in the second layer, HTML language is used. The Apache Web server is selected as the most widely spread in the world. It serves more than half of the existing sites, and its market share is still increasing. Its advantage is that it is multi-platform. The Apache Web server is particularly quick and powerful and operates very well with MySQL database and the PHP programming language, selected for the application.

In order to improve the security, SSL (Secure Sockets Layer) protocol is attached to the Web server with the help of the mod-ssl module and the OpenSSL library. SSL encrypts the information exchanged between the server and the browser and enables mutual authentication of both sides. The encryption technique used is public key cryptography, hence the server sends a public key to the client, which is used to encrypt the information, so that only the server having the private key can decode it later. The client uses the public key for encrypting and transmitting to the server its own key, which uniquely identifies it.

There exist a large number of technologies to choose from on the server side for implementing the functionality of the second layer. The present work uses PHP technology. This selection is based on the following reasons:

- PHP is supported by many platforms (Linux, Windows, etc.) and different Web servers (Apache, Internet Information Server, Personal Web server and others). It is freely available on the Internet and the scripts written in PHP can be used with practically no changes on any PHP platform.
- PHP supports an API for access to a wide set of databases.

3.2. Choice of the database software

MySQL database is used to store the necessary information in our system, because it is free, easy to install and configure when utilizing PHP. Using this product is even more facilitated thanks to the PHP built-in functions for working with MySQL.

MySQL is a relational data base. It supports SQL (Structured Query Language). SQL is a standard language for queries and administration of a database. One of the advantages of the MySQL product is its platform independence. MySQL is also a quick database, as proved by numerous Benchmark tests.

4. Design of the database

The created database is intended to store the following information:

- information about a hotel – name, address, category, types of services offered by the hotel, room furnishings, room categories, number of the rooms available, price of every room, type and number of the beds;
- views of the hotel, the different types of rooms and maps in SVG format of the cities and countries, where the hotels entered in the database are situated;
- personal information about the users – name, surname, address, e-mail;
- the bookings already made in the system – check-in and check-out dates, type and number of the rooms already booked, reservation status.

The tables and fields in the database are defined as Countries, City, Hotels, Room_type, Room, Reservation and Users in a block diagram of the database (Fig. 2).

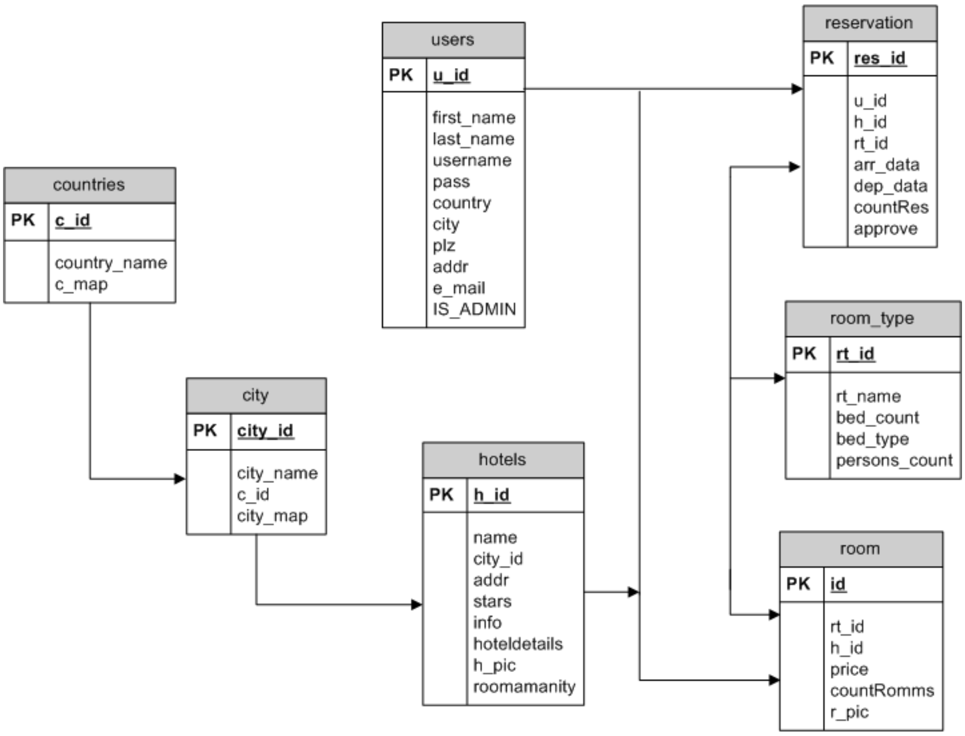


Fig. 2. Block diagram of the database

5. Algorithmic implementation of the application

The implemented project consists of two basic parts – an administrator’s module and a module, serving the clients. As already mentioned, the software tools employed in the system are PHP technology and MySQL database, using also the capabilities of SVG vector graphics standard of W3C.

Fig. 3 presents a scheme of the connections among the pages of the user interface.

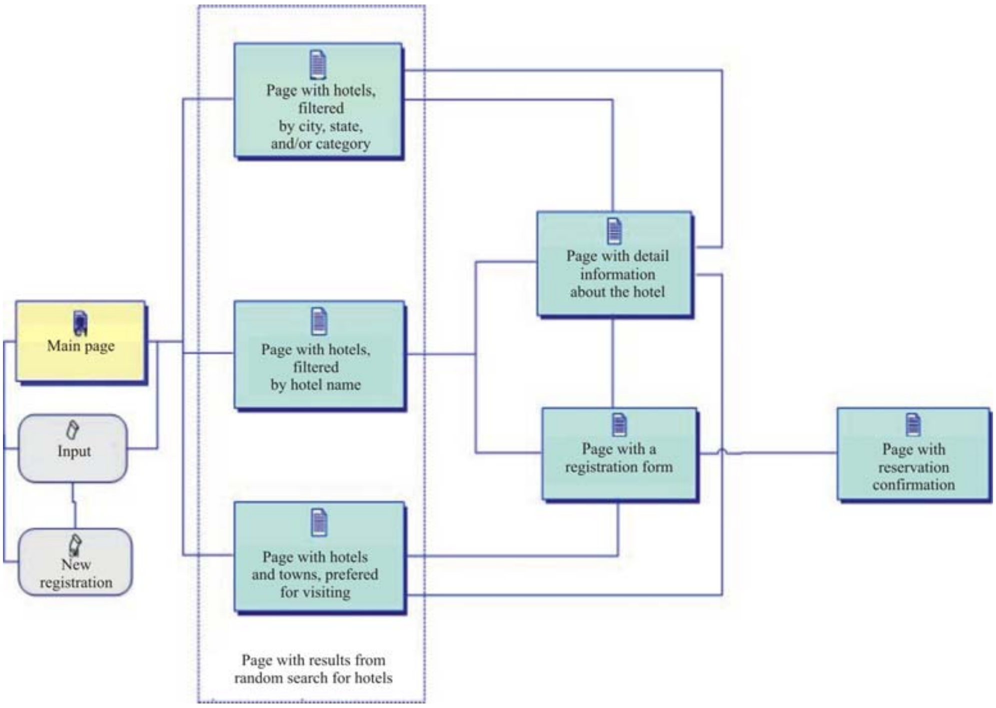

Page with hotels, filtered by city, state, and/or category
Page with detail information about the hotel
Main page
Page with hotels, filtered by hotel name
Input
Page with a registration form
Page with reservation confirmation
New registration
Page with hotels and towns, prefered for visiting
Page with results from random search for hotels

Fig. 3. A scheme of the connections among the pages of the user interface

6. Main pages during system operation

The implemented system for hotel rooms’ reservations consists of two modules: user’s and administrator’s. An user-friendly Web interface has been developed for both modules. When the system is initially loaded, the main page of the application appears. It contains two frames – upper and lower.

The upper frame contains the application banner and some basic references (links to the “Home” page and to the “About us” page, giving information about the system developers) and also a search field for hotel names. The lower frame visualizes the pages of the user and the administrator, first loading the user interface main page (Fig. 4).

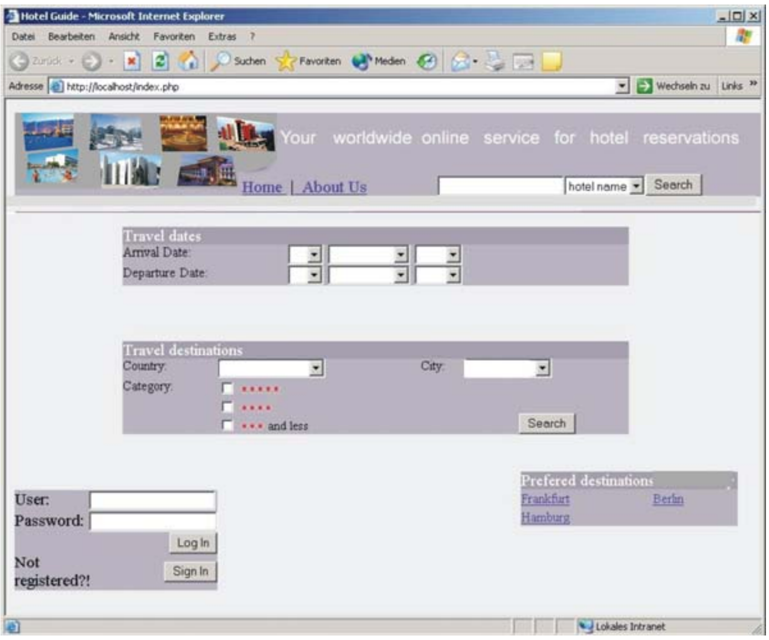
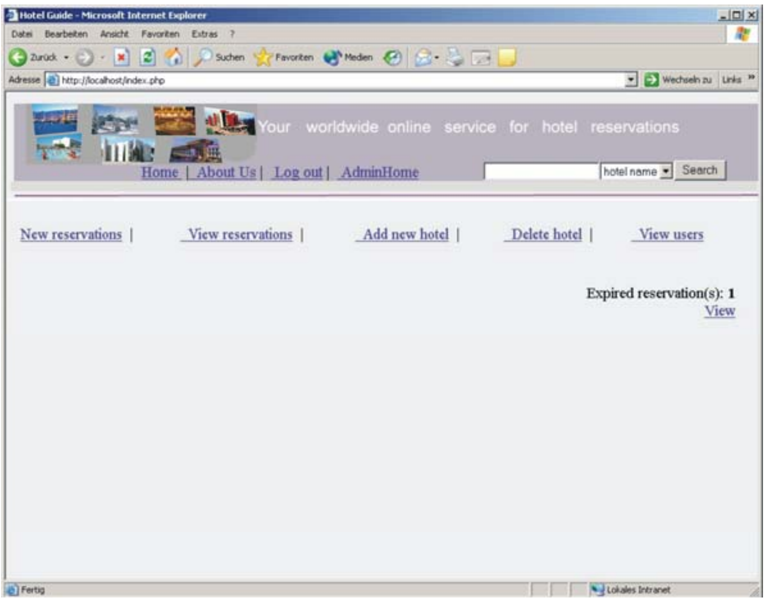


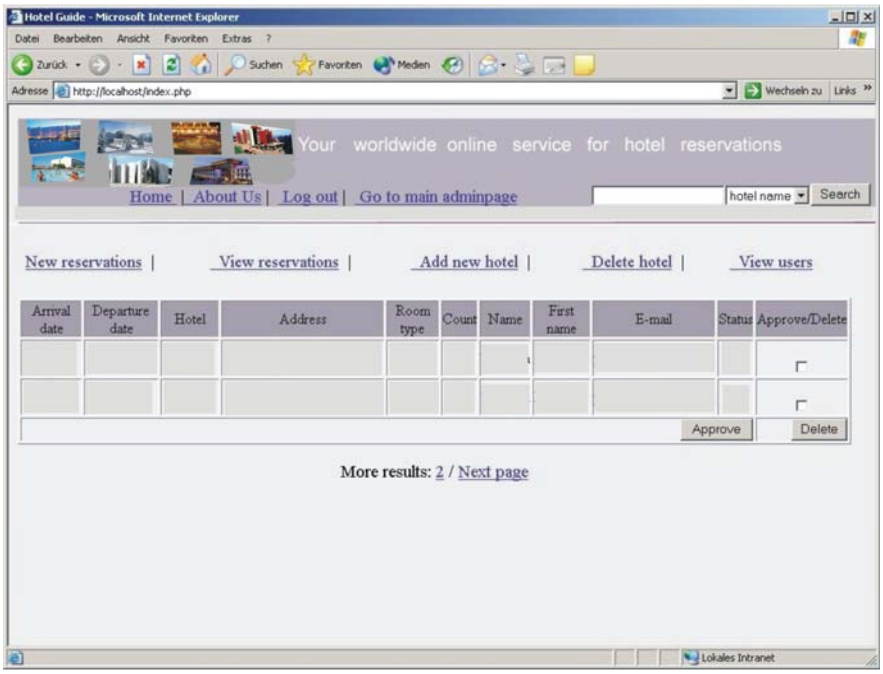
Fig. 4. Main page of the application

In order to meet the users' requirements, the Internet application needs constant supplementing and updating of the database information (Fig. 5). This is a responsibility of the system administrator. The administrator also confirms or cancels a given reservation, sending e-mail messages to the respective user (Fig. 6).


Hotel Guide - Microsoft Internet Explorer
Datei Bearbeiten Ansicht Favoriten Extras ?
Zurück Suchen Favoriten Medien
Adresse http://localhost/index.php
Wechseln zu Links
Your worldwide online service for hotel reservations
Home | About Us | Log out | AdminHome
hotel name
Search
New reservations |
View reservations |
Add new hotel |
Delete hotel |
View users
View
Fertig
Lokales Intranet

Expired reservation(s): 1

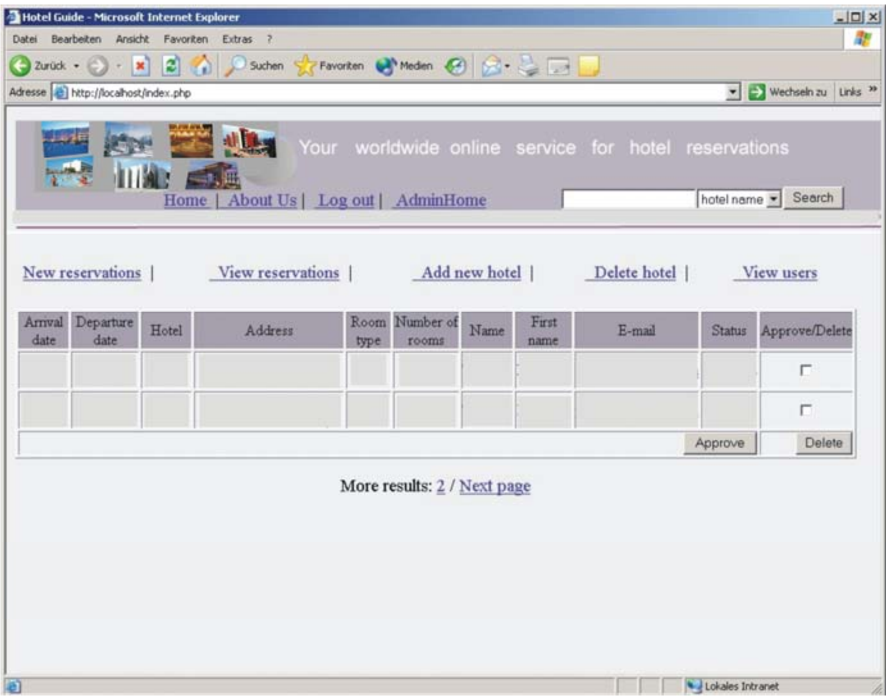
Fig. 5. Main page of the administrator's interface


Hotel Guide - Microsoft Internet Explorer
Datei Bearbeiten Ansicht Favoriten Extras ?
Zurück Suchen Favoriten Medien
Adresse http://localhost/index.php
Wechseln zu Links
Your worldwide online service for hotel reservations
Home | About Us | Log out | Go to main adminpage
hotel name
Search
New reservations | View reservations | Add new hotel | Delete hotel | View users
Approve
Delete
More results: 2 / Next page
Lokales Intranet

Arrival date	Departure date	Hotel	Address	Room type	Count	Name	First name	E-mail	Status	Approve/Delete
										□
										□

Fig. 6. Page of review of the new registrations in the system

The "View reservations" page generates a list with the reservations in the system (Fig. 7).


Hotel Guide - Microsoft Internet Explorer
Datei Bearbeiten Ansicht Favoriten Extras ?
Zurück Suchen Favoriten Medien
Adresse http://localhost/index.php
Wechseln zu Links
Your worldwide online service for hotel reservations
Home | About Us | Log out | AdminHome
hotel name
Search
New reservations | View reservations | Add new hotel | Delete hotel | View users
Approve
Delete
More results: 2 / Next page
Lokales Intranet

Arrival date	Departure date	Hotel	Address	Room type	Number of rooms	Name	First name	E-mail	Status	Approve/Delete
										□
										□

Fig. 7. Page, presenting all the reservations in the system

7. Examples of SVG images application

Two examples are given, in which different attributes of the SVG image are altered, using values of variables from PHP, defined in the code or extracted from the data base.

In the first example the defined in PHP variables $svgwidth and $svgheight are used, in order to set the image size. This is an example of using PHP variables in an SVG image. The variables are used as follows:

```
        <rect x="0" y="0" width="<?= $svgwidth;?>" height="<?=
   $svgheight;?>"  style="fill:#eeeeee;stroke:#eeeeee"/>
```

In addition, the name of the image (of the hotel or hotel room) is given in this file and it can be processed by SVG filters for blurring. The image is contained in the variable $hotel_pic and altered depending on the user’s selection.

The program code for displaying a given hotel is the following:

hotelSVG.php

```
<?php
        header("Content-type: image/svg+xml");
        //Setting the limits of the graphic in SVG format
        $svgwidth=180;
        $svgheight=180;
        //$hotel_pic – stores the name of the file with an image of the hotel/room
        $hotel_pic="$hotel_pic".".".$extension;
 ?>
         <!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.0//EN"
        "http://www.w3.org/TR/2001/REC-SVG-20010904/DTD/svg10.dtd">
        <svg width="<?= $svgwidth;?>" height="<?= $svgheight;?>"  >
        <defs>
                <filter id="rand" >
                        <!—Setting a filter, as an input parameter the file with the
                        image -- >
                        <feImage xlink:href="<?=$hotel_pic;?>" result="i1"/>
                        <!—- A filter, causing a blurring effect -- >
                        <feGaussianBlur in="SourceGraphic" stdDeviation="20"
                        result="i2"/>
                        <!—- A filter, including the results from the processing by
                        the filters above given -- >
                        <feComposite in="i1" in2="i2" operator="in"/>
                </filter>
        </defs>
        <rect x="0" y="0" width="<?= $svgwidth;?>" height="<?= $svgheight;?>"
        style="fill:#eeeeee;stroke:#eeeeee"/>
        <circle cx="90" cy="90" r="80" style="fill:white;stroke:black;stroke-
        width:8;filter:url(#rand)"/>
        </svg>
```

The result is represented as shown in Fig. 8.

Fig. 8

In the next example the values of the attributes of SVG image are extracted from the data base in the form of an array, some alterations are made, new elements added and SVG image of the town map is generated with the hotel location, marked by X and the address, written below.

city.php

```
<?php
          header("Content-type: image/svg+xml");
          include("func.php");
          //Connection with the data base
         ConectDb();
         // Extracting of SVG code from the data base on the map of the town, where
         the selected hotel is found ;
         //extracting of the hotel name, address and category
         $query = "SELECT city.city_map,hotels.name,hotels.addr,hotels.stars
                                 FROM city,hotels
                                 WHERE hotels.h_id=$hotel_id
                                 AND hotels.city_id=city.city_id;";
         $result=mysql_query($query);
         $temparray=mysql_fetch_array($result);
         $name=$temparray["name"];
         $addr=$temparray["addr"];
         $stars=$temparray["stars"];
```

```
        $tempAddr=array();
        $addr=trim($addr);
        $razdelitel=" ";
        //splittng the string with the hotel address into parts and storing the parts in
        //in an array
        $newAddr=split($razdelitel,$addr);
        $num_elements=count($newAddr);

                for($i=0,$j=0;$i<$num_elements,$j<$num_elements;$i++,$j++)

                                {
                        //Removing the street number from the address
                                        if (!ereg("^[0-9]+",$newAddr[$i]))
                                        {
                                                $tempAddr[$j]=$newAddr[$i];

                                        }
                                        else
                                        {
                                                break;
                                        }
                                                }
                                $svg_addr=implode(" ",$tempAddr);
        $image=$temparray["city_map"];
//Alteration in the map code
        $mask1="showcity('common');";
        $repl1="showcity('$svg_addr');";
        $new_image1=ereg_replace($mask1,$repl1,$image);
        $mask2="</svg>";
        $repl2="/n";
        $new_image2=ereg_replace($mask2,$repl2,$new_image1);
        $last_image=$new_image2
                         ."<text     id='AddTextPath'     style='fill:rgb(192,0,0);font-
                         size:20;font-family:Arial;font-weight:bold'
                         display='none'>".
                        "<textPath xlink:href='#$svg_addr' startOffset='40%'>" .
                        "X        </textPath>       </text>".
                        "<text       id='AddTextLine'       style='fill:rgb(192,0,0);font-
                        size:20;font-family:Arial;font-weight:bold'   display='none'>
                        X</text>".
                       "<rect id='UnterRhamen'
                      style='fill:rgb(139,93,45);stroke:rgb(0,0,0);stroke-
                      width:1;fill-opacity:0.2
                                '/>".
                        "           <text id='vartext' style='fill:rgb(192,0,0);font-
                        size:15;font-family:Arial' display='none'>X-$name,
                        $addr</text>     <use   x='500' y='550' xlink:href='#button'
                        filter='url(#3DButon)' onclick='print();'/><text  x='505'
                        y='565'  style='fill:rgb(128,0,0);font-size:15;font-
                        family:Arial;font-style:oblique' onclick='print();'>Print
                        </text></svg>";
        echo  "$last_image";
        ?>
```

The result from this code is shown in Fig. 9.

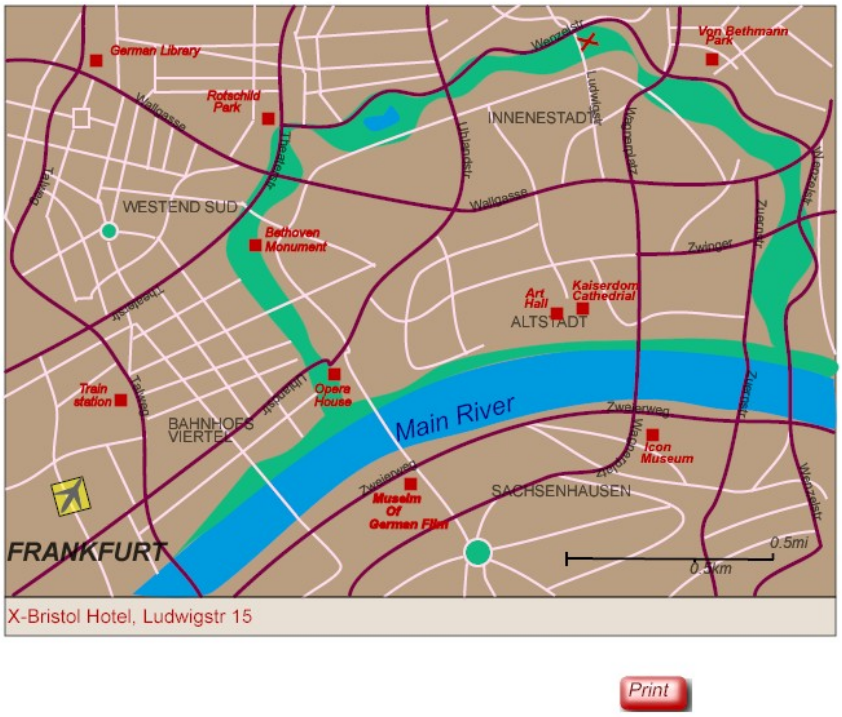


Fig. 9

8. Conclusion

The current paper discusses an information system for hotel room reservations, providing an user-friendly interface to database operations. The presentation of each hotel includes some photos of the hotel and its premises, as well as a map of the city, where it is located, using the capabilities of the SVG vector standard. The system allows updating and supplementing of the stored information as well as future expansion of its functionality with the purpose of satisfying the clients' requirements. The same approach, with some slight modifications, could be applied to information systems in other areas.

References

1. Kastaneto, Hesus, Harish Ravat. Professional Programming with PHP. SoftPress Publishing House, 2001 (in Bulgarian).
2. Kofler, Michael. MySQL – Introduction, Programming, Application. Addison Welsey Publishing House, 2001 (in Bulgarian).
3. Adam, Alexander. SVG – Practical Handbook. Franzis Publishing House, 2002 (in Bulgarian).
4. Documentation and handbooks for programming with PHP.
   www.php.net
5. Documentation for MySQL.
   www.myslq.com
6. Specification for SVG.
   www.w3.org/Graphics/SVG

Об одном подходе к разработке информационных систем

Николай Докев

Нов болгарский университет, 1618 София; Институт информационных технологий,
1113 София
E-mail: niki@iinf.bas.bg

(Резюме)

В работе показан подход к разработке информационных систем, используя векторный стандарт SVG (Scalable Vector Graphics). При помощи этого стандарта проектирана и построена информационная система для резервации отельских комнат. Подход можно успешно применять и в системах для изображения и резервирования комнат в домах отдыха, в санаториумах, общежитиях, а также и в дистанционных покупках или в аренде аппартаментов.