PERSONALIZED QR CODES APPLICATION

DANIEL POP AND CAMELIA-M. PINTEA

Abstract. The current paper introduces new software to easily encoding diverse data with *Quick Response Codes* (*QR codes*). The user friendly tool includes also a database with personal *QR codes*. The management of the database is facilitated by a specific software design.

I.   INTRODUCTION

The information is one of the most valuable assets today. The daily life increasingly interacts more often with technology designed to facilitate and to improve living. Sharing information is an integral part of modern society. Billions of emails are sent daily and money circulates easily in cyberspace. Considering this, it is important to understand the cryptography and data security in networks [5,7] in this era of information. The data transfer via the Internet must be protected [2,9].

The way people process, store and encrypt information today is changing faster. Currently coded information is everywhere: bar-code on all commercial products, invoices, certificates, in media and of course in the environment which played a crucial role in technological development: the Internet. One of these modern means of coding is the *Quick Response Code* (*QR code*). It is a range of standards-dimensional bar coding form.

QR codes render information (messages, links, etc.), encoded using a cryptographic algorithm facilitating user access with own smart-phones, tablets and other mobile devices, to the references hidden behind code. Given the popularity enjoyed by these codes in countries like Japan, USA, Canada, Germany etc. the current project is a web application to facilitate generating and managing *QR codes* for the Romanian user.

The current application *Personalized QR App (PQRA)* brings together a small part of the vast multitude of information on cryptology, cryptography and the use of databases in computer applications. The web application is developed by combining various standards and languages as *HTML, CSS, PHP, MySQL* and *JavaScript* [10]. It is an application based on a user-name and password. An user within its account has the possibility to generate, edit and upload *QR codes* of various products as for example receipts or certificates. Thus the user can manage its own database with personal QR codes. The friendly interface facilitates user access to update the database and data encryption. The success of this era depends on the quality and evolution of cryptology as it provides us with locks and keys to the most valuable information [3].

II. QR CODES - PRELIMINARIES

Composed by joining the Greek words "kryptos" (hidden) and "logos" (science) the word "cryptology" means the discipline which studies the writings of the secret processes of encryption and decryption of messages. Based on the Greek words "kryptos" and "graphein" (write) the word "cryptography" literally means "secret writing", "hidden message" [1]. *Quick Response Code (QR codes)* and codes with"quick response" is a range of coding standards bars dimensional (2D barcode) known as code matrix (matrix barcode).

A *QR code* consists of black modules like, square points and square dots, arranged in a square grid placed on a white background which can be read by a device image such as a camera and processed using Reed-Solomon algorithm error correction until the image can be interpreted properly. The data is extracted from the modules present in both horizontal components and the vertical image [4].

The *QR* standard code was created by Japanese company Denso Wave Inc., a Toyota company in 1994. The code was developed to facilitate the management and monitoring of motor vehicles and spare parts in the vehicle industry by creating a means of labeling that could be deciphered (read) quickly; from here is the name"quick response code". Today QR codes are found in many fields such as advertising, interior design, architecture, art, and

fashion. Today has a very high popularity and it is use also in marketing, on authentication within a web page, to encode useful information etc. In the year 2000 the specifications of the QR codes have been approved by *International Organization for Standardization (ISO)* as one of its international standards [4].
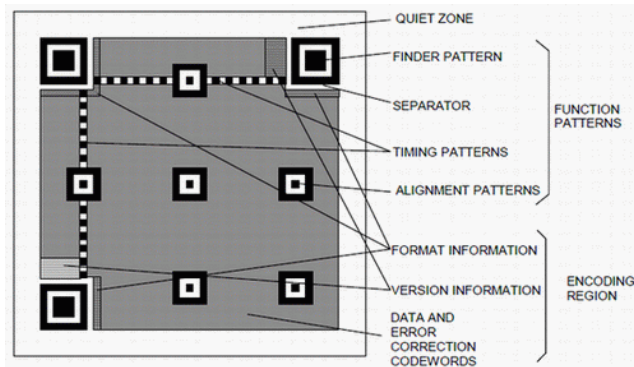

FIG. 1. The general structure of a *Quick Response Code* (*QR code*) as in [6].

Almost twenty years after its invention the *QR code* it is not just popular but very useful and still in continuous development and improvement and will replace conventional barcodes [8]. *QR Code* is a matrix symbol consisting of a series of square modules arranged in a pattern of the same shape (Fig. 1). Currently there are forty versions of *QR codes* that depend on the size of the encoded message. Each *QR code* is made up of square modules included in a square matrix; it has two regions: region coding (information about alignment, formatting, and coding of data and error correction) and function templates (separators, templates for position detection for timing and alignment). Square modules that are represented data can have two values corresponding to the binary 1 and 0 values: dark modules and light modules. The *QR code* size was between 21 x 21 modules, in the first version and 177 x 177 modules in version 40, increasing with 4 modules on each version of the code.

III. PERSONALIZED QR APPLICATION

The popularity of QR codes is growing and become more common and used also in Romania. In Romania it is used for commercial products, promotional posters, promotional offers, official documents as insurance and bills. Based on its utility we develop a web application intended for Romanian user, to easily generate and manage *QR codes*. The introduced application is called *Personalized QR App (PQRA)*. It is developed using various standards and

languages as HTML, CSS, PHP, MySQL and JavaScript [10]. The web-based software is using authentication. The user is invited to create a personal account where he will be able to manage its own collection of QR codes, whether they were generated by the user or loaded from other sources.

Developed with the help of standards and technologies such as *HTML, CSS, PHP, MySQL* and *JavaScript* the application is built around a database with the collection of *QR codes*. The software is available to a wide audience and does not require special knowledge in the *IT* field. There exists also other similar applications but *Personalized QR App* is a simplified option including the operation of generating *QR codes* and the management of a personal database with related codes.

The designed is minimalist and modern combining standard styling with *CSS* and scripts written in JavaScript; the user experience is both enjoyable and interactive on its personal collection of *QR codes*. The relational database within the web application *Personalized QR App* is implemented using *MySQL* via *phpMyAdmin* tool. It contains information used for authentication, user-name, and password, and also information about each *QR code* collection of each user. The database, called suggestively *qr* is composed of two tables, *users* and *codes*. The table *users* includes within others the unique *id* as primary key for each user, the username and the password used for authentication; the name and the e-mail address of the user. The password is an encrypted form via *MD5* function to enhance information security. The table *codes* includes as primary the key the unique identification code assigned; the $user - id$ and the *encrypted information* encoded in the corresponding code; the *source code* with two possibilities: *generated (G)* and *loaded (I)*; the physical path to the image file including the code and the path to the image file that contains the code used to display it on the application.

Personalized QR codes application

FIG. 2. The design of the new *Personalized QR App*, an efficient software to generate and manage a personal database with *QR codes*.

The software design. The *PQRA* has the classic structure of web pages including a header, the body, and the footer. The information is for Romanian users and therefore the information is written in Romanian. The header has a personalized logo of the application and the authentication panel is used to authenticate access to personal account. The main body includes the content of each page viewed and the footer includes information about the application and development (Fig.2).

The authenticated user menu includes three possible options: *Generation*, *Upload* and *View*. The navigation menu include the main page called *Home* with a slider presentation of the current application; the *Registration* page is for the common users without a personal account; the *About* page has information about *QR codes* and code readers and *Contact* page includes the contact information.

The application uses a *php* library for generating *QR codes*. There are specific folders with images in the design application, with JavaScript script slider presentation, the *QR codes* for each user, the *php* library to generate QR codes and temporary files of codes.

The Users Guide for the Personalized QR App.

The registration. At first an user needs to create a personal account and to be authenticated on the *Personalized QR App* web-site. In the registration page the user will complete the entry form in order to create an account and benefit of the services provided by the application. The server will build a personal media folder for each unique user who will save its *QR codes*.

The authentication. Once registered, the user will be able authenticated; for now on has the possibility to choose the operations: to generate, upload or view the codes in the personal collection of codes.

FIG. 3. In the left side is the representation of the user-interface window where the generation of *QR code* is performed based on a given text; there is also the possibility to choose a specific error code and the dimension of the future code. In the right side is the representation of the generated *QR code* based on the previous specifications.

The Code Generation. The code generation option is found on the main menu; the user will be redirected to the generation code window (Fig. 3). Here the user insert the message to encode and also has the possibility to choose between the correction of errors *(L, M, Q, H)* and the size of the future QR code *1, 2, ..., 10*. The button *Generate* will generate the corresponding *QR code* from the inserted text based on the selected options: its appearance and the success message for generating or loading the image file. The function that inserts code in the database is in a database *php* file. The code is generated in a specific *php* file, accessing library generators from the corresponding *php* code folder. The resulting code is moved to the folder with the unique *identification* assigned to the current user (Fig. 3).

Uploading QR codes. The user can upload *QR codes* from different sources or personal collection. Could be used already existing image files with *QR codes*. An existing code could be photographed and load it into the database using the loading option within the application menu. This option will lead to a loading window where it is also the possibility to include a short description of the loaded code, preferably the message encoded in the code. The file is uploaded to the server and added to the personalized user database.

Personalized QR codes application

FIG.4. In the left side is the window where the user has the possibility to visualize all its personal *QR codes* by default or just the generated or the uploaded ones. In the right side is the editing window where the user could modify an existing code using another text and/or another error code and/or another dimension of the future code; both *QR codes*, initial and its modified version are also illustrated.

  Visualization option. The current option allows the user to view its collection of personal codes. By default all the codes are viewed but there is the option to view only the codes generated or the codes loaded. The codes are displayed in two columns, in the left column the code and in the other column the code description. Under each code are the corresponding operations for editing or removal (Fig. 4). The function responsible for displaying all the codes, the codes generated or loaded are similar, the only change that occurs is including the corresponding option *'source'='G'* or *'source'='I'*.

  Editing QR codes. In the editing window the user could modify the text encoded in the existing code, could re-elect the correction of errors and code size (Figure 4). The new information will be updated in the database and the old *QR code* will be replaced by the newest one. The code is subject to editing in a specific *php* file.

  Delete QR codes. The option to delete a specific code will remove the 4code from the user personal collection. A deleting alert is included to prevent deleting accidentally a *QR code*. If the user is sure that he wants to delete a particular code, the code will be removed from the database. The user will be redirected to the corresponding window interface to visualize the updated codes.

IV. CONCLUSION AND FURTHER WORK

  The current paper introduces a user-friendly *Quick Response (QR) Codes* management tool. The web application *Personalized QR App* could be optimized. Some of the possible improvements follows: introducing a QR code reader in order to include the decoding operation; the QR code reader will be loaded by the user to prevents improper collection as auto-fill the description

with the message encoded within the code; extending the degree of customization of QR code by introducing: *Design QR* or *Logo QR* through which the user can change colors code, forms or modules including small pictures inside it; expanding the range of data that can be encoded for example picture, business card or resume; the possibility to dispatch selected codes via email or *MMS*.

References

[1] M. D. Coroiu, **Criptografia in Pascal**, Ed. Zestrea, Baia-Mare (2010)

[2] K.W. Dam, H. S. Lin, **Cryptography's role in securing the information society,** National Academies Press (1996)

[3] O. Goldreich, **Modern Cryptography, Probabilistic Proofs and Pseudorandomness**, Springer (1999)

[4] ISO Standard: **Information Technology. Automatic identification and data capture techniques. Bar Code symbology**. **QR Code.** ISO-IEC-18004 (2000)

[5] A. Neagos¸ S. Motogna, **Security Analysis Regarding Cross-Site Scripting on Internet Explorer**, in Proceedings of the Fifth Balkan Conference in Informatics, Serbia 2012 (Z. Budimac, Z. et al., Eds.) (2012) 125-128

[6] **Onbarcode** http://www.onbarcode.com/qr_code/

[7] C-M. Pintea, P.C. Pop, **Sensor networks security based on sensitive robots agents. A conceptual model**, Advances in Intelligent Systems and Computing, (2013) 189:47-56

[8] QRcode, **History of QR code**, 2013 http://www.qrcode.com/en/history/

[9] E. Simon, D. Naccache, **Criptografie si securitatea informatiilor. Aplicatii,** Matrixrom Publisher (2011)

[10] . World Wide Web Consortium, **HTML & CSS** (2013) http://www.w3.org/

Trencadis, Baia-Mare, ROMANIA

Technical University Cluj-Napoca
North University Center of Baia Mare
Faculty of Sciences
Department of Mathematics and Informatics
76 Victoriei, 430122, Baia-Mare
e-mail: dr.camelia.pintea @ ieee.org