Analysis And Design Of Client Application Of Intelligent Measuring System For Measurement Of Liquid Fuels And Data Transmission

Ekaterina Antoniva Gospodinova
Electrical engineering, automation and information technology
Technical University - Faculty of Engineering and Pedagogy
Sliwen, Bulgaria
ekaterina_gospodinova@abv.bg

Abstract— **The development and exploration of new approaches and measurement methods in modern automated level measurement systems is a current issue and a major impetus for human communication. Rising requirements have led to complex applications and software servicing that is difficult to maintain. Here I present a system for measuring liquid fuels and data transmission.**

Keywords—Information systems, software, databases, client applications, algorithm, level meter system

I.   INTRODUCTION

The level meter systems are installed in gas stations. They show fuel tank information: leakage data, temperature, product density, product volume, water level. They allow remote reading and fuel control. It is necessary to introduce modern ways of accomplishing the activities. A current trend is the construction of such a system for automated measurements, development and design of modern architecture. One of the main components of software quality is reliability. This is the ability of the system to perform its tasks according to the specification on which it was created. There are external and internal factors that determine the quality. External factors such as speed, performance, lightweight interface are those designed for users of the system. Internal factors such as readability, modularity, and expandability are only visible to those who develop and maintain the product. External factors and internal factors are dependent. The most important external factors are:

- correctness and fulfillment of the tasks according to the specifics;

- ability to deal with errors and unexpected state

- ability to develop and adapt the product to changes in the specifics;

- functionality and capabilities.

II.   DESIGNING A CLIENT APPLICATION FOR AN AUTOMATED SYSTEM

Designed app include: application software installed on the computer and provided at the remote location to the client, software server, database, and communication link. The software for the client application was written on C ++, C # and MySQL.
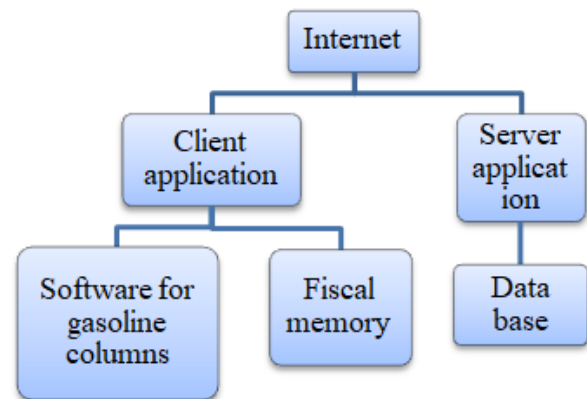

Fig. 1 Software scheme in level meter system

Tools and technologies used in the • Platform - the programming languages used to create the application are JavaScript, HTML and MySQL. These languages allow flexibility and multi-platform usage. Due to the fact that JavaScript codes are executed on a virtual machine like any modern computer supports, the application can be run on any operating system.
• Developing environment used - to make the code more efficient and faster, an integrated development environment and C # language are used.

The internal design structure of the client application is shown in Figure 2. At its creation, the program is divided into parts. As can be seen from the chart, the app consists of several separate systems that work largely independently of one another. They only share information about filling the petrol stations and the tank.
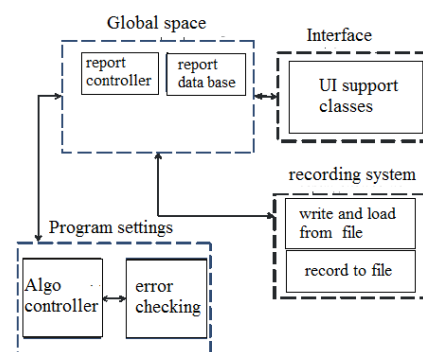

Fig. 2 Internal client-application structure

The computer system, designed for the sale of liquid fuels, gas and other goods at gas and petrol stations is based on a personal computer, an electronic system with fiscal memory and a central register device.

The program created was designed and built using a quick reference help system that facilitates the operator's work. Service personnel requirements are minimal computer skills. Programming, deployment, and decommissioning is performed by a customer service organization, with which the client has signed a maintenance contract. Data transmission is via a modem using a GPRS connection with a mobile operator selected by the customer. The sale of liquid fuels and gas is automatically recorded.

III. BASIC CLIENT APPLICATION ALGORITHM

The program created runs several algorithms. The solution resembles the coloring of a graph. However, there are several significant differences that increase the level of difficulty. All methods are written specifically for the task.

First, the user is required to enter identification, device type, and charging method. Depending on the device, the function can be switched to gas station or tank. If it is a tank, the charged quantity is accounted and passed to reports. If it`s gasoline consoles, a choice is made from a preentered list box, the corresponding settings, and finally reports are again reached.

The tools and technologies of object-oriented programming used are the basis for designing software and software systems. Basic programming languages are supplemented with object support capabilities.

Object-oriented programming means are used, which is a paradigm where a programming system is modeled as a set of objects that interact with one another. Each object is capable of receiving messages, processing data,
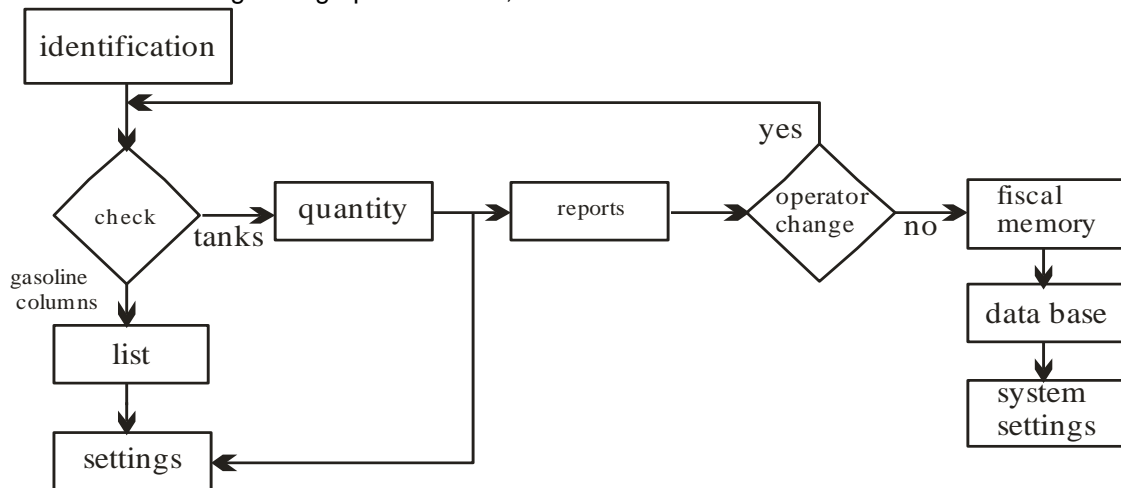

Fig. 3 Program Algorithm

and sending messages to other objects. The classes in this appendix are defined according to the specific needs. The application is based on Object Oriented Programming and contains both classes and objects as well as data structures. An algorithm has been implemented to study consumer habits and make autonomous decisions. The idea of this algorithm is to monitor when a command is executed and, on the basis of these observations, make suggestions to the user for its automatic execution. For example, if the program notices that the same event occurs every day in a given period, it will offer the user automatic execution. The first step of the algorithm is to record on what date and how many hours each command is being executed. Only when the database becomes large enough, the program makes its conclusions. Once enough data is collected, the information is sorted in ascending order of the hours and minutes of execution of a command. After sorting, the list stores the distance between each two elements and the indexes of the left and right elements.

IV. PROTECTION

An important part of the program is to provide access protection. To do so, a dialog box is required asking the user to enter a name and a password.

The algorithm shown in fig.4 starts with a check of the password entered by the operator. If it is correct, it can continue to work by giving access to the necessary menus. If the password is incorrect, it is possible to re-enter, and if an error is made again, it starts checking the issued documents and the system access. An AES standard with a Rijndael algorithm and a key length of 128 bits is used. By using encryption, the data transfer is secured so that the user's name and password cannot be seen in pure form. The encryption and decryption keys are unique and are kept on the client side of the program and once more on a server.

Unlike the key that is a constant, the initialization vector is always randomly generated. After encryption of the provided key as an argument of the text function, it and the initialization vector are encoded using the Base64 standard. This will easily transfer the

information as text, and not binary data. Additionally, another level of protection from middle man observers is achieved through coding. After all these changes on the original text, the information is added to the generic collection NameValueCollection, in which one or more values can match a specific string. Users are stored in a MySQL database where a username is stored, MD5 password hashes, and a unique salt used for hashing.

The connection to the database has been implemented by matching the correct password and its salt. The resulting possible password is then cleared by the salt taken from the database and the resulting hash is compared with the correct password. If both hashes are the same, the entered password is correct, the 'granted' response is sent to the customer, and if the hashes are different, a 'locked' response is sent.

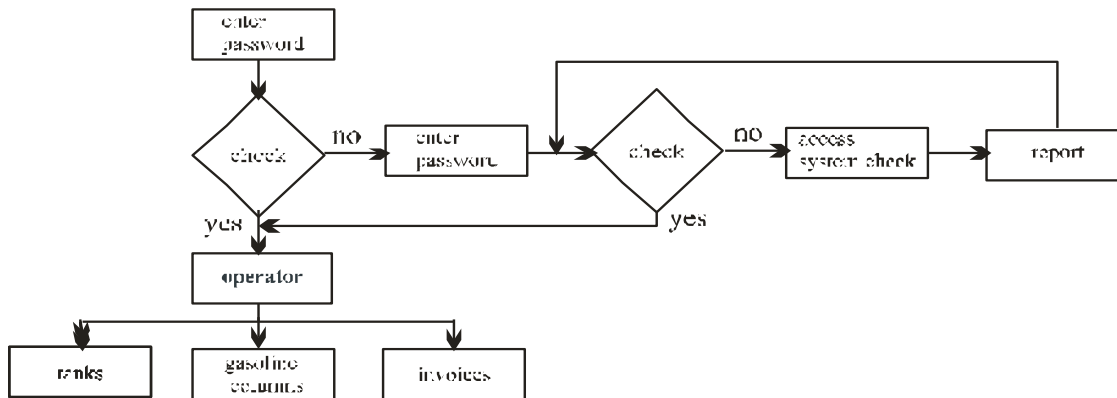The script checks the received information and executes the desired command based on it.

Fig. 4 Password Verification Algorithm

The created application is of installation type. Although the program consists of multiple files, they can be copied and placed by the user, anywhere.

V. FILE

Files or documents are created using reports. These documents can be viewed, printed and recorded for further consideration. They mainly consist of tables. They can also be written in a text format that makes it easy to transfer tables to other table-based program products.

A table is created that examines all system events that are registered. The name of the operator who performed the action, the time of the event, and the text describing it, are displayed.

VI. SETTINGS

The capabilities of the system are managed by setting the data it works with. Settings are adjusted using dialogs. In most cases, the data are presented in tables. The top row of the tables lists the names of the data in the corresponding column. Some are numbered, and the leftmost column contains numbers that cannot be changed. They change directly into the table. After editing a field, the change is applied after moving to another field or with Enter. Listed options are available for some of the fields that appear when you try to edit and can be selected with the arrows. For fields that are not edited in the table, dialogs automatically appear. When programming, the data is written directly to the database and it is not possible to go back.

VII. CONCLUSIONS

Integrating this software application into an interactive system aims to support strategic modeling and strategic management of innovative technologies, alongside with integrating the entire working process of the intelligent measurement systems. We face the challenges of storing and retrieving information in a particular situation and its protection. Open source technologies and free technologies are used. There is no need for expensive licenses for the basic software. The application is of key interest for better productivity and management, as it provides excellent review and data transmission and is built with a quick-help system that facilitates operator performance. Service personnel requirements are minimal computer skills.

REFERENCES

[1]   Klir G.J., M.J. Wierman, "Uncertainty-based information: elements of generalized information theory", 2009.

[2] S. Blake, D. Black, etc. "An Architecture for Differentiated Services ", 2016.

[3] http://www.ietf.org

[4]   Rinaudo, F., Agosto, E., Ardissone, P.: Gis and Web-Gis, Commercial and Open Source Platforms: General Rules for Cultural Heritage Documentation. In. XXI International CIPA Symposium, Athens (2007).

[5] ESPRIT Programme: Guidelines for Best Practice in User Interface for GIS. The European Commission.