Розроблено структуру серверної частини системи обліку пасажиропотоку громадського транспорту, алгоритм функціонування та спеціалізоване програмне забезпечення. Розроблено і реалізовано інформаційне та технічне забезпечення, що забезпечує низьку ціну проектного рішення та є оптимальним рішенням з широкими функціональними можливостями. Наведено результати параметрів обліку пасажиропотоку громадського транспорту, отриманих з використанням розробленої системи

Ключові слова: "розумне" місто, система обліку пасажиропотоку громадського транспорту, інформаційна та програмна моделі

Разработана структура серверной части системы учета пассажиропотока общественного транспорта, алгоритм функционировании и специализированное программное обеспечение. Разработано и реализовано информационное и техническое обеспечение, обеспечивающее низкую цену проектного решения и являющееся оптимальным решением с широкими функциональными возможностями. Приведены результаты параметров учета пассажиропотока общественного транспорта, полученных с использованием разработанной системы

Ключевые слова: "умный" город, система учета пассажиропотока общественного транспорта, информационная и программная модели

UDC 004.02; 004.942
DOI: 10.15587/1729-4061.2017.92831

DEVELOPMENT OF MODELS AND MEANS OF THE SERVER PART OF THE SYSTEM FOR PASSENGER TRAFFIC REGISTRATION OF PUBLIC TRANSPORT IN THE "SMART" CITY

O. Boreiko
Lecturer*
E-mail: bor@tneu.edu.ua
V. Teslyuk
Doctor of Technical Sciences, Professor**
E-mail: vtesliuk@polynet.lviv.ua
A. Zelinskyy
PhD**
E-mail: andriy.zelinskyy@gmail.com
O. Berezsky
Doctor of Technical Sciences, Professor*
E-mail: ob@tneu.edu.ua
*Department of computer engineering
Ternopil National Economic University
Lvivska str., 11, Ternopil, Ukraine, 46020
**Department of computer aided design
Lviv Polytechnic National University
S. Bandery str., 12, Lviv, Ukraine, 79013

1. Introduction

Development of modern society, problems with energy sources and tough conditions of competition for markets, require that manufacturers of products develop intelligent systems using modern element base. One of such areas that is developing dynamically is the technology of "smart" city [1], which, on the one hand, enables significant saving of energy and improving the level of management in the city, and, on the other hand, help to improve the level of comfort and quality of services for the consumers.

A "smart" city combines aspects of technological infrastructure and "public technologies". New methods with the usage of smartphones, growth in popularity of online transactions, low cost of equipment and P2P-technologies create prerequisites for the collaboration of municipal authorities and people in the city for better use of resources, data collection and making effective decisions [2].

In the structural aspect, "smart" city is a system of interacting systems. The technology of "smart" city is composed of high-speed optical, sensor, cable and wireless networks [3].

Today, one of the main technologies to organize "smart" cities is the technology of the Internet of things (IoT). At present, IoT can be considered a key trend of world economy of the nearest decade [4]. An important role in the development of technologies for the Internet of things belongs to the intelligent solutions in the area of machine-to-machine communications (M2M). This concept implies the integration of communication equipment with a variety of devices [5].

Regarding the construction of automated control systems, "smart" city can be considered as a combination of several vertical markets, such as transportation [3], medicine, industrial automation, education and energy, security and logistics [6]. All developments in these fields can be applied to build an effective "smart" city.

One of the interesting examples of using technologies of machine-to-machine communications M2M is the system for passenger traffic registration of public transport [3]. Such systems are built based on the use of specialized equipment, such as controllers with appropriated sensors for collecting and processing information, positioning systems, etc. Systems for registration of passenger traffic allow control of travel payment, determining transportation load on the routes and optimizing the work of public transport depending on the days of the week [7].

Therefore, a technology of the "smart" city is a complex system. Accordingly, the implementation of the system of "smart" city is associated with a number of technical problems [8], namely: a problem with the implementation of a number of smart functions, the need to ensure low-cost of technical system, provision of the required reliability of technical solution [9], provision of scalability of the designed system, etc.

Therefore, design and implementation of the system for passenger traffic registration of public transport (SPTRPT) [3] as part of the "smart" city system is an important task at present.

2. Literature review and problem statement

The basis of operation of the system for passenger traffic registration of public transport in the "smart" city system is a client – server interaction. This feature implies establishing and implementation of certain requirements that are set to both client and server part of the system to ensure proper and reliable functioning of the registration system of passenger traffic as a whole.

The systems designed for the registration of passenger traffic by the criterion of the side, where the process of calculation actually occurs, can be divided into two types [10].

In the first case [11], data collection and processing occurs at the client side of the system with specific result, a number, sent to the server. The server in this case serves only to store data from clients and to display comprehensible statistics and analytical information for a particular client, or groups of clients.

In the second case [12], only the collection and preliminary processing of incoming data is conducted at the client side. Data are structured and sent to the server in a "raw" form. Then the process of calculation takes place at the server side. Thus, the server side has a larger number of functions and, accordingly, requirements to the reliability and efficiency of its work are more strict.

The proposed system is developed with a focus on the second type, that is, the process of calculation takes place at the server side.

First, at the client side, a video recording of passenger traffic is captured by videocameras placed over the front and rear doors of the vehicle. After that, photo registration of passengers with the right to free travel takes place [13], GPS- coordinates and other system information are defined. All processes occur by the interaction between the controller, based on the single board mini-computer Raspberry Pi [14], and peripheral devices at the client side [15].

The next stages are preliminary processing of data from the periphery, structuring, packages formation and sending the packages to the server (WEB and FTP server) [16] for their storage and further processing.

3. The purpose and objectives of the study

The purpose of present study is the development of models and tools of the server side of the system for automated registration of passenger traffic of public transport, which is characterized by a wide functionality and low price.

To achieve this purpose, it is necessary to solve the following tasks:

– to develop a structure of the server side of the system for passenger traffic registration of public transport, which should be based on module principle;

– to build a functioning algorithm and develop specialized software for the server part of SPTRPT, which is based on the use of object-oriented approach;

– to develop an information model that will ensure reliable data exhange between a client and the server of the developed system;

– to build a technical support for the server part of the system for passenger traffic registration of public transport that provides for a low cost of technical solution.

4. Structure of the system for passenger traffic registration in the "smart" city

A client-server interaction is the basis of the developed structural scheme of the system for passenger traffic registration of public transport in the "smart" city, which is shown in Fig. 1. A controller that is installed in a vehicle and receives data from sensors and the driver is related to the client part. The server side consists of hardware and software implementation of the server, as well as operators that manage the process of processing data coming in from the controllers (clients).

Data collection and processing in the system for passenger traffic registration of public transport proceed in several stages.

At the first stage, each controller, installed in a unit of public transport vehicle, receives data from the periphery (GPS-module, cameras). At the same stage, controller receives data from the driver. Next, the driver activates the camera by pushing an appropriate button for the registration of passengers with the right for a free travel [17].

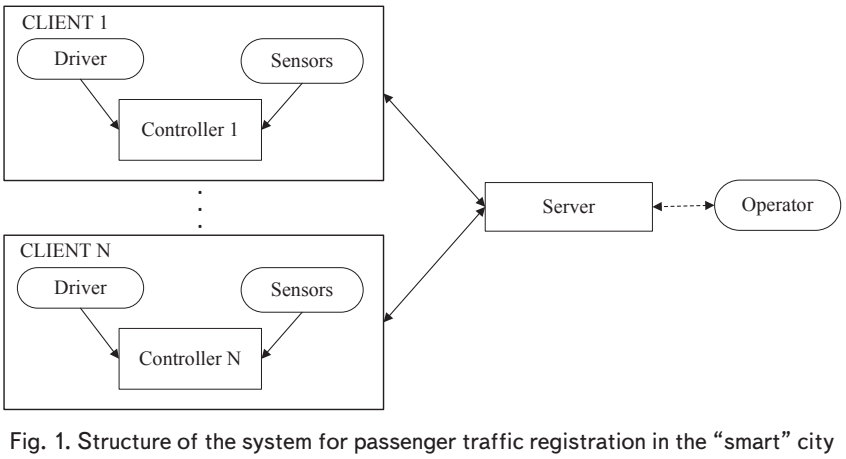

Fig. 1. Structure of the system for passenger traffic registration in the "smart" city

The next stage is the transfer by controller of received data to be processed by server. At the server side, all data are structured and stored for their further processing by the system and by operators, if necessary.

The final stage is the data processing with consequent representation of detailed and accurate statistics and analytics on the passenger traffic dynamics in public transport over the appropriate period.

In the process of implementation of the system for passenger traffic registration of public transport in the "smart" city, a modular principle is employed that makes it possible to effectively organize the improvement and modification of the designed system.

5. Information model of the system for passenger traffic registration of public transport of the "smart" city

The designed server part of the system includes a hardware server with software realizations of database server, a server for saving files and the Web server. Processing of data, received from clients, takes place at the server side. Operators deal with incomplete or ambiguous information.

The information model developed aims to describe information processes, options and system properties for passenger traffic registration of public transport of the "smart" city (Fig. 2).

The model is presented in the form of a diagram that shows all basic composition elements of the system, interaction between them and describes data flow and stages of interpretation of them into complete information analytics.

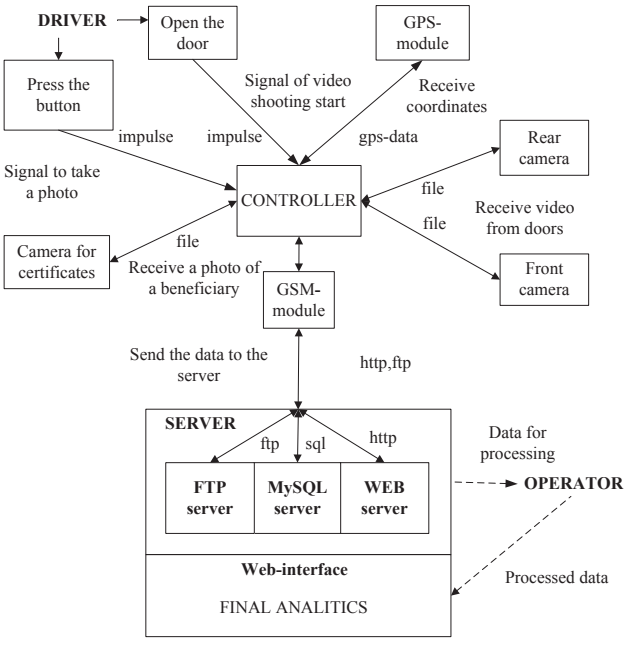

Fig. 2. Structure of data flow in the information model of the system for passenger traffic registration of public transport

In a general case, each controller–client sends data about the event. The structures of data on the event can be divided into two types. In the first case, the fact of opening the front or rear door of the vehicle and pressing a special button for registering beneficiaries can be considered as the event. Both by the fact of doors opening and by the fact of pressing the button for registering beneficiaries by the driver, a report is formed by the controller that includes a set of specific data. Such data are the following:

– specific coordinates of the place where the event took place from a GPS-module;

– exact time when the event occurred;

– the title of the video (if the door opened) or photo (after camera was activated by a button to register the beneficiaries);

– service data (front/rear door, certificates and etc.).

Multimedia data, such as video and photo in particular, belong to the second type, Thus, the data sent from the client to the server, are related to the first type. They arrive in a form of a POST-request to the Web server. Multimedia data coming in for saving at the file server are related to the second type. A simplified diagram of the developed structures of data is shown in Fig. 3.

The reports contain information about the time of the event, its location and the content itself (the information about the event is the video from doors and what exactly, or the photo from the camera of fixing beneficiaries). Reports are generated in a POST request to the Web server, and multimedia files (photos of beneficiaries, video from the doors) are sent to the FTP server. All POST-requests from controllers are stored in the database on the server and the files are converted for processing and are stored in the server's disk space. Processed statistical data are stored as XML-files [18].

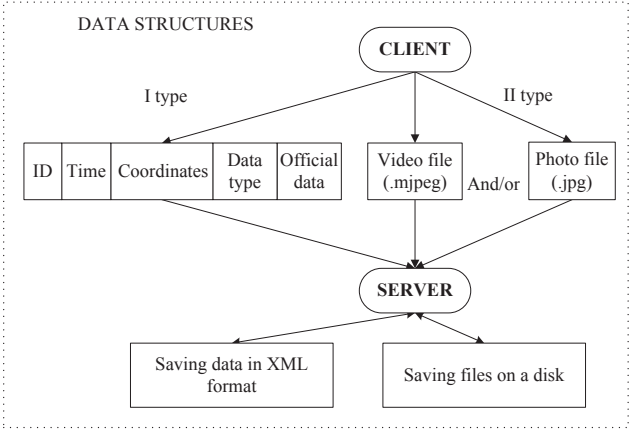

Fig. 3 Types of structures of data in the system for passenger traffic registration of public transport

Therefore, the developed structures of data of two types make it possible to control, maintain and work out effectively parameters of passenger traffic of public transport of the "smart" city. In the process of realization of the information model, list structures of data were used.

6. Working algorithm of the server part of the system

The data received from the client side and saved on the server are checked for their integrity, accuracy and correctness. When the verification is completed, some data undergo automatic processing, while the data, which is characterized by a kind of ambivalence, are sent for processing by the operator.

A fragment of flowchart of the algorithm for the procedure of processing data from clients at the server part is shown in Fig. 4.

The developed algorithm of the server part of the system for passenger traffic registration of public transport of the "smart" city includes the following steps (stages):

1. Parameters initialization – preparation of all hardware and software tools of the server to receive data from clients (software running, downloading initial data, etc.).

2. Check the availability of input data – software checks the fact of the requests implementation from clients to the server (WEB-server, FTP-server).

3. Data save – in the case of getting queries and data from clients, save them to database and disk space of server, otherwise expect requests from clients.

4. Verification of correctness and completeness of data – analysis of the data received from the clients and evaluation of the possibility of further work with them. For example, if the GPS coordinates are correct, to build a GPS-track to display the route of the vehicle on the map, otherwise skip; to send for processing data with proven reliability only, completeness and correctness.

5. Check a possibility of automatic data processing – if data received from clients meet all the stated requirements, then perform their automated processing, otherwise send to be manually handled by the operator.

6. Data processing – actually processing by the server part of the system the data received from clients in the automatic or manual (if necessary) mode.

7. Representation of results – report and compilation of all statistical and analytical information by each particular client based on the data obtained from him.

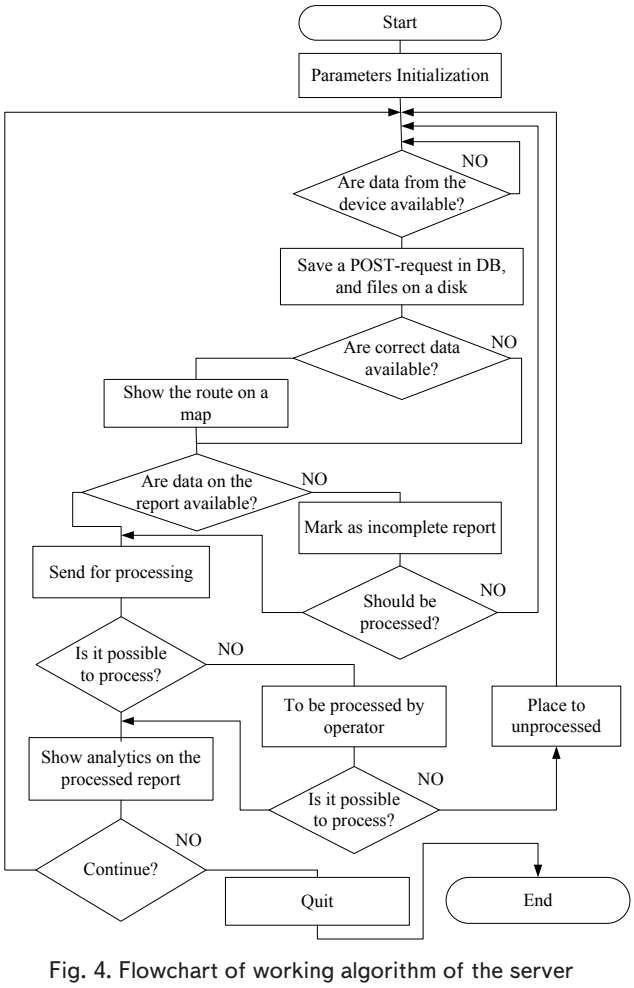

Fig. 4. Flowchart of working algorithm of the server

7. Peculiarities of software development for the server side of the system

The basis of software (SW) structure of the server part of the system is the server environment of the Ubuntu operating system [19] and software implementations of the servers (Fig. 5). Programming language of server part of the system is PHP [20]. Web-framework is Zend Framework [21].

A model of software organization of server side of the system is shown in Fig. 6. This model includes three hierarchical levels of processing and storing of data from the client side. At the first level, the system receives requests from clients that come through Internet. At the second level, all requests are sorted according to their purpose (POST requests are for the Web server, files are to FTP server). At the third level, with the help of tools of PHP programming language, we save and, if necessary, take out the data received from the client side of the system through the Web server.

PHP Hypertext PreProcessor is the script programming language, which is used for execution at the server side. PHP is created for building dynamic and interactive Internet sites. This programming language proved to be very flexible and powerful, so it gained popularity and started to be used in multi-scale projects: ranging from basic blog to large Web applications. The advantages of this language are:

– PHP relates to free software that is distributed under a special license (PHP license);

– easy to master at each stage;

– widely supported by users and developers;

– there are essential tools to maintain databases (DB);

– a lot of libraries is implemented, as well as extensions of this language;

– it offers valid resources for the development of Web sessions, software interface of extensions;

– it may be deployed on almost any server;

– it is applicable for a large number of hardware and software platforms.

Zend Framework is a Web application framework developed by Zend (USA) (the company, which carries out support and coordination of PHP project,). Zend Framework widens the PHP language while keeping its spirit, its main criterion is simplicity, it used the best techniques of object-oriented programming, a friendly license, and a well-tested code that is quickly executed. It is important that Zend Framework includes a possibility to design well-protected, reliable and modern applications of WEB 2.0 and Web services and widely available API functions from the teams of leaders in this fielde, such as Google, Yahoo!, Amazon, Flickr (USA).

As the Web server, we chose Apache server [22]. Apache server is an application of Web server with open-source software developed by the Apache Software Foundation (USA). Server software is freely distributed and licensed with open source code, which means that users can edit the basic code to adjust productivity and contribute to the development of the program.

Apache is cross-platform, and is most often used in the combination with the Linux operating system, which is also distributed with open source code. These two components, combined with MySQL [23], database and PHP script language, form a popular solution for the Web server called LAMP (complex server software).

Main advantages of Apache and LAMP server systems are:

– low cost, as well as the license for software is free;

– flexible programming through the open source code;

– improved security, because Apache was developed for the Unix-like operating systems.

Database server (SQL Server) is presented by the MySQL package. This package is well adapted for use in a Web environment, DMS (database management system).

Usually, on most hosting platforms, to execute client applications, service providers allocate not a very large amount of resources (computing, disk). That is why it is important to use efficient DMS, which is characterized by high reliability (preferably web-applications and internet sites should work under 24-hour mode without days off).

In connection with the enumerated reasons, MySQL turned to be a rock-hard standard in the area of DMS for Web, and now it is also developing capabilities for use in a variety of mission-critical business applications.

The advantages of MySQL are the following:

– multiflow and the ability to support multiple concurrent requests;

– possibility to optimize connections by adding much data over a single cycle;

– support of records with fixed or variable length;

– ODBC driver;

– the use of a flexible system and privileges;

– flexible support, number of formats and time tags;

– fast work, easy scalability;

– possible interface with C and Perl, PHP languages;

– compatibility with ANSI SQL;

– good support of hosting services by providers;

– fast support of transactions due to InnoDB mechanism.

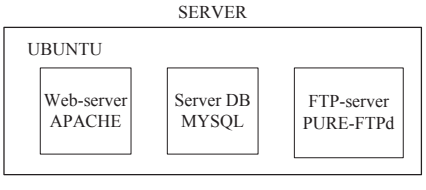

Fig. 5. Structural model of the server software

Pure-FTPd Server is used as a FTP server [24]. Pure-FTPd is a free FTP server for the UNIX family of operating systems, distributed by BSD license. Its main focus is security and simplicity of tuning.

It was compiled from the source code for Linux, OpenBSD, NetBSD, DragonFly BSD, FreeBSD, Solaris, Tru64, Darwin, IRIX and HP-UX for Android.

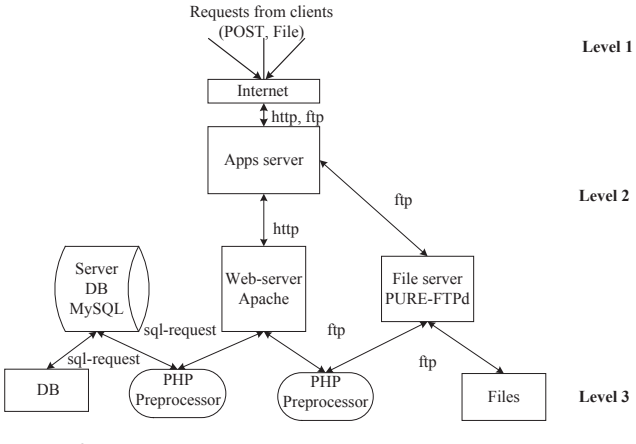

Fig. 6. Multilevel model of software organization of the server part of the system

A distinctive feature is that the server does not read the settings directly from the configuration files, and accepts them only from the command line. But a possibility of using configuration files exists.

Therefore, developed SW of the server part of the system for passenger traffic registration of public transport of the «smart» city provides desired functionality and is free.

8. Development of hardware provision of the server part of the system

Analysis of technical task, as well as the requirements to the software of the server part of the system for passenger traffic registration of public transport of the "smart" city, allowed us to draw conclusions about the requirements to hardware provision of the server part of the system. An important condition is maintaining the possibility of continuous scaling, increasing technical capabilities of the equipment for further development of the project.

The main server parameters and, accordingly, its selection criteria are:

1) CPU (central processing unit), based on which it operates;

2) TYPE and volume of RAM (random access memory);

3) volume of PAM (permanent access memory).

Thus, Intel Pentium G3260 (USA) is selected as the central processor for the server operation [25]. The main technical parameters are given in Table 1.

Table 1

Technical specifications of Intel Pentium G3260 processor

Number of cores	2
Number of threads	2
Basic clock frequency of processor	3.30 GHz
Cash-memory	3 MB
Frequency of system bus	5 GT/s DMI2
Calculated power	53 W

Random access memory is represented by the Kingston DDR3-1600 8 GB (USA) (two microchips per 4 GB) [26]. Device settings are listed in Table 2.

Table 2

Technial specifications of RAM Kingston DDR3-1600 4 GB

Volume of memory	4 Gb
Memory type	DDR3 SDRAM
Supply voltage	1.5 W
Memory frequency	1600 MHz
Effective throughput	12800 Mb/s
Memory Timing circuit	CL11

Volume of permanent access memory is 500 GB. All the main characteristics of the hardware server SPTRPT of the "smart" city are summarized in Table 3.

Thus, the developed technical provision of the server part of SPTRPT provides a low cost of technical solution, provides a possibility of continuos scaling and increasing equipment capacity in the process of the system development.

Table 3

Technical specifications of the server

Processor	Intel Pentium G3260 3.3 GHz (2 cores)
Chipset	Intel H81
Random access memory	8 Gb DDR3
Video	Intel HD Graphics
Audio	Realtek ALC662, HD, 5.1-channel
Network adapter	1x10/100/1000 Mbit
Ports	On the front panel: 2 x USB 2.0 On the back panel: 2 x PS/2 1 x VGA 1 x DVI 4 x USB 2.0 1 x RJ-45 3 x audio
PAM	500 Gb HDD
Power unit	External network adapter for 60 W
Dimensions	210×62×210 mm

9. Discussion of results: examples of application and prospects for the system development

The developed tools of the system allow mapping a route of the vehicle in real time, which, as an example, is shown in Fig. 7. At any time, it is possible to receive information about location of the selected vehicle. However, the constructed system makes it possible to receive a report by specific route, where the information on the date, start and end time of the motion, driver's name, etc. is represented (Fig. 8).

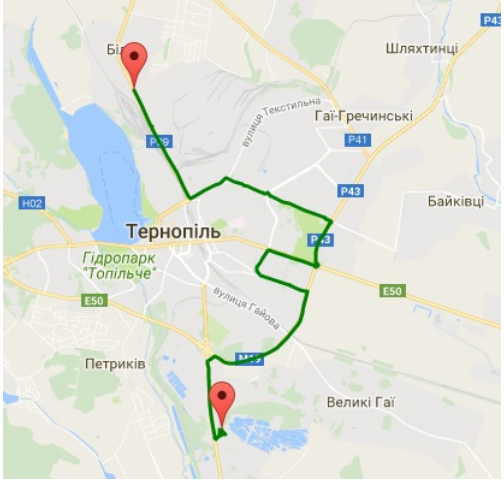

Fig. 7. Example of representation of the route of vehicle motion

An example of the menu with statistics of the route is shown in Fig. 9. Accordingly, the reports generated using a software part of the system provide a full picture of passenger traffic parameters of each of the routes and their total indicators.

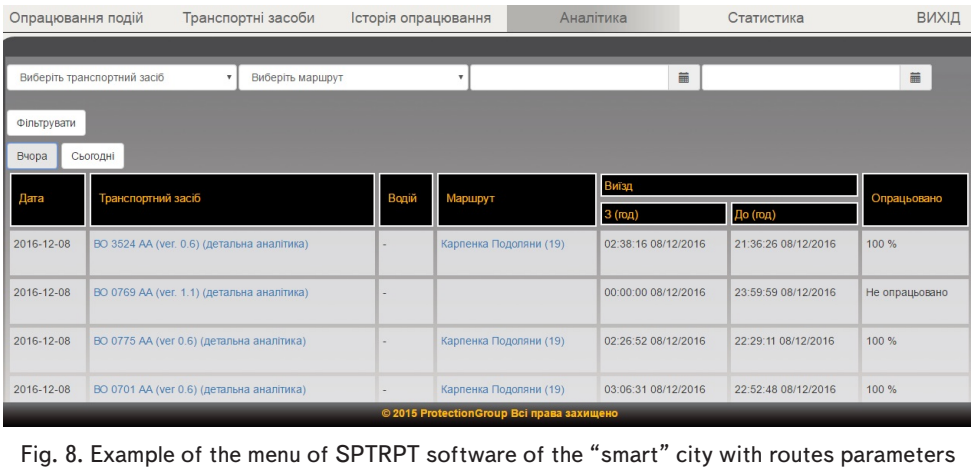

Fig. 8. Example of the menu of SPTRPT software of the "smart" city with routes parameters

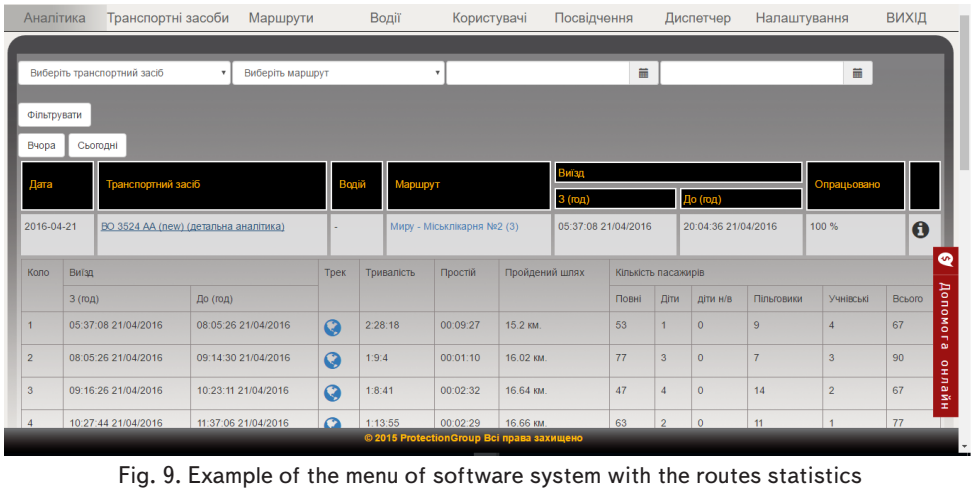

Fig. 9. Example of the menu of software system with the routes statistics

The goven generated reports with the use of the developed system display a full picture of the passenger traffic route of the vehicle. The developed registration system of passenger traffic of public transport was tested at the ATP "Mens-Auto" and "Etalon" in the city of Ternopil (Ukraine).

Regarding further development of the system, it might include the optimization of its work by two major directions – software and hardware.

It is possible to change the PHP programming language to the Python language, which is better organized syntactically and structurally [27]. This in its turn will lead to applying one of the most popular and effective Python Web-frameworks. For example, such frameworks as Django, Flask [28, 29], etc. might be used. Collection of data on the work of the server part of the system and their detailed analysis is an important stage in the operation of the developed system. Such data may point to certain shortcomings in the process of increasing the number of clients. This in its turn may lead to debating the question on changing the Web-server. The use an alternative Web server, for example, Ngnix could be a solution to this problem [30].

Regarding development of hardware provision for SPTRPT of the "smart" city, it implies scaling of hardware capacity. Such scaling can be achieved by increasing the random access memory and permanent access memory. There is also a possibility of a more serious optimization in the form of using the hardware, which is based on the work of the more advanced central processor.

10. Conclusions

1. We developed a structure of the server part of the system for passenger traffic registration of city public transport. This structure is based on a modular principle, which provides a simple and fast replacement of the particular module in the case of its failure. As a result, increased reliability of the system as a whole is achieved, as well as the smooth scaling and increase of the developed system in the future.

2. The algorithm of functioning of the server part of SPTRPT is built. Its characteristic features are the use of a systemic approach to the implementation of incoming data processing and automation of work of the human-machine system. Thanks to these features, the capability of complete verification of accuracy of processed incoming data becomes possible, as well as clear and understandable representation of results of the calculation of passenger traffic parameters. The application of automated processing of incoming data made it possible to improve efficiency of the system by covering up the set of cases of incompleteness and/or damage of incoming data. We developed specialized software of the server part of the system for passenger traffic registration of public transport. The SW implements all the above features of the device and is based on the use of modern approaches to object-oriented programming, including the use of Web-frameworks.

3. We developed information software that enables secure and reliable data exchange between a client and a server of the system. Informational support includes a range of modern technologies and protocols. These technologies include video data collection using IP cameras, data transfer with the help of 3G, storing them in a relational DB and on disk space of FTP server, data processing using list structures of data and saving statistics in the form of XML-files. Accordingly, the developed information software is based on the application of modern protocols for the collection, transmission, processing and storage of data (TCP/IP, MySQL Client/Server Protocol, HTTP, FTP, etc.). Thus, the employed technologies and protocols allow implementing effectively the transfer and processing of data and meet all the stated system requirements.

4. Technical support of the server part for SPTRPT is developed. Using modern approaches to SW development and optimization of DB, we managed to reduce the load on CPU and RAM of the server. Due to this, it became possible to carry out the selection of optimal, in terms of price-quality, hardware server. Emphasis is also placed on the modularity of hardware of the server part for easy subsequent scaling. Accordingly, the technical support developed provides low cost of technical solution and is based on the use of low-cost elements that ensure reliable operation of the system in full.

References

1. Zhuhadar, L. The next wave of innovation – Review of smart cities intelligent operation systems [Text] / L. Zhuhadar, E. Thrasher, S. Marklin, P. O. de Pablos // Computers in Human Behavior. – 2017. – Vol. 66. – P. 273–281. doi: 10.1016/j.chb.2016.09.030
2. Byun, J. Smart City Implementation Models Based on IoT Technology [Text] / J. Byun, S. Kim, J. Sa, S. Kim, Y.-T. Shin, J.-B. Kim // Advanced Science and Technology Letters. – 2016. – Vol. 129. – P. 209–212. doi: 10.14257/astl.2016.129.41
3. Mazur, V. Planning of routes based on distribution of passenger flows in time and space [Text] / V. Mazur // 2016 XII International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH). – 2016. doi: 10.1109/memstech.2016.7507541
4. Gauer, A. Smart city architecture and its applications based on IoT [Text] / A. Gauer, B. Scotney, G. Parr, S. McClean // Procedia Computer Science. – 2015. – Vol. 52. – P. 1089–1094. doi: 10.1016/j.procs.2015.05.122
5. Park, Y. Analysis on Smart City service technology with IoT [Text] / Y. Park, S. Rue // Korea institute of information Technology Review. – 2015. – Vol. 13, Issue 2. – P. 31–37.
6. Nowicka, K. Smart City Logistics on Cloud Computing Model [Text] / K. Nowicka // Procedia – Social and Behavioral Sciences. – 2014. – Vol. 151. – P. 266–281. doi: 10.1016/j.sbspro.2014.10.025
7. Boreiko, O. Structural model of passenger counting and public transport tracking system of smart city [Text] / O. Boreiko, V. Teslyuk // 2016 XII International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH). – 2016. doi: 10.1109/memstech.2016.7507533
8. Stefanovych, T. The reliability model for failure cause analysis of pressure vessel protective fittings with taking into account load-sharing effect between valves [Text] / T. Stefanovych, S. Shcherbovskykh, P. Drozdziel // Diagnostyka. – 2015. – Vol. 16, Issue 4. – P. 17–24.

9. Mulyak, O. Influence of software reliability models on reliability measures of software and hardware systems [Text] / O. Mulyak, V. Yakovyna, B. Volochiy // Eastern-European Journal of Enterprise Technologies. – 2015. – Vol. 4, Issue 9 (76). – P. 53–57. doi: 10.15587/1729-4061.2015.47336
10. Olaverri-Monreal, C. Intelligent Technologies for Mobility in Smart Cities [Text] / C. Olaverri-Monreal // Hiradastechnika Journal. – 2016. – Vol. 71. – P. 29–34.
11. Jamil, M. S. Smart Environment Monitoring System by Employing Wireless Sensor Networks on Vehicles for Pollution Free Smart Cities [Text] / M. S. Jamil, M. A. Jamilb, A. Mazharc, A. Ikrama, A. Ahmeda, U. Munawar // Procedia Engineering. – 2015. – Vol. 107. – P. 480–484. doi: 10.1016/j.proeng.2015.06.106
12. Gilmore, S. Validation of Automatic Vehicle Location Data in Public Transport Systems [Text] / S. Gilmore, D. Reijsbergen // Electronic Notes in Theoretical Computer Science. – 2015. – Vol. 318. – P. 31–51. doi: 10.1016/j.entcs.2015.10.018
13. Pasquale, G. D. Innovative Public Transport in Europe, Asia and Latin America: A Survey of Recent Implementations [Text] / G. Di Pasquale, A. S. dos Santos, A. G. Leal, M. Tozzi // Transportation Research Procedia. – 2016. – Vol. 14. – P. 3284–3293. doi: 10.1016/j.trpro.2016.05.276
14. Leccese, F. A Smart City Application: A Fully Controlled Street Lighting Isle Based on Raspberry-Pi Card, a ZigBee Sensor Network and WiMAX [Text] / F. Leccese, M. Cagnetti, D. Trinca // Sensors. – 2014. – Vol. 14, Issue 12. – P. 24408–24424. doi: 10.3390/s141224408
15. Kamble, K. P. Smart vehicle tracking system [Text] / K. P. Kamble // International Journal of Distributed and Parallel systems. – 2012. – Vol. 3, Issue 4. – P. 91–98. doi: 10.5121/ijdps.2012.3410
16. Bischof, S. Semantic Modelling of Smart City Data [Text] / S. Bischof, A. Karapantelakis, C.-S. Nechifor, A. Sheth, A. Mileo, P. Barnaghi // Proc. W3C Workshop on the Web of Things. – 2014. – P. 1–5. – Available at: http://www.w3.org/2014/02/wot/papers/karapantelakis.pdf
17. Boreiko, O. Developing a controller for registering passenger flow of public transport for the "smart" city system [Text] / O. Boreiko, V. Teslyuk // Eastern-European Journal of Eenterprise Technologies. – 2016. – Vol. 6, Issue 3 (84). – P. 40–46. doi: 10.15587/1729-4061.2016.84143
18. Denysyuk, P. Usage of XML for Fluidic MEMS Database Design [Text] / P. Denysyuk // 2007 International Conference on Perspective Technologies and Methods in MEMS Design. – 2007. doi: 10.1109/memstech.2007.4283450
19. Scale out with Ubuntu Server [Electronic resource]. – Ubuntu. – Available at: https://www.ubuntu.com/server
20. Kotov, D. V. PHP 5 [Text] / D. V. Kotov, A. F. Kostarev. – 2-nd ed., revised and enlarged. – Saint Petersburg: BHV-Peterburg, 2008. – 1104 p.
21. Documentation [Electronic resource]. – Available at: https://framework.zend.com/learn
22. Laurie, B. Apache: The Definitive Guide [Text] / B. Laurie, P. Laurie. – 3-rd ed. – Publisher: O'Reilly Media, 2002. – 590 p.
23. Yarger, P. MySQL and mSQL. Database for small businesses and Internet [Text] / P. Yarger. – Saint Petersburg: Symvol-Plus, 2000. – 560 p.
24. Pure-FTPd [Electronic resource]. – Available at: https://www.pureftpd.org/project/pure-ftpd/doc
25. Intel Pentium Processor G3260 (3M Cache, 3.30 GHz) [Electronic resource]. – Intel. – Available at: http://ark.intel.com/ru/products/87356/Intel-Pentium-Processor-G3260-3M-Cache-3_30-GHz
26. Servernaja pamjat' Kingston [Electronic resource]. – Kingston Technology Corporation. – Available at: http://www.kingston.com/ru/memory/server
27. Lutz, M. Programming Python. Vol. 1 [Text] / M. Lutz. – 4-th ed. – Saint Petersburg: Symvol-Plus, 2011. – 992 p.
28. Greenberg, M. Development of web applications using Flask in Python [Text] / M. Greenberg. – Moscow: DMK, 2014. – 272 p.
29. Chan, W. Django. Developing web applications in Python [Text] / W. Chan, P. Bissex, D. Forse. – Saint Petersburg: Symvol-Plus, 2015. – 456 p.
30. Welcome to NGINX Wiki's documentation! [Electronic resource]. – Available at: https://www.nginx.com/resources/wiki/