CLIENT-SERVER SYSTEM FOR DATA FROM WEB PAGES

Artur Britvin[1], Jawad Hammad Alrawashdeh[2], Rostyslav Tkachuck[3]

[1,3]*Lviv Polytechnic National University, 12, Bandera Str, Lviv, 79013, Ukraine*
[2]*Majmaah University, VC78+QMQ, Industrial Area, Al Majma'ah 15341, Saudi Arabia*
[3]*Lviv State University of Life Safety, 35, Kleparivska Str., Lviv, 79007, Ukraine*
Author's e-mail*: britvin.artur.mki.2020@lpnu.ua*

*Abstract***: An overview of the basic principles and approaches for extracting information and processing information from web pages has been conducted. A methodology for developing a client-server system based on a tool for automation of work in Selenium web browsers based on the analyzed information about data parsing has been created. A third-party API as a user interface to simplify and speed up system development has been used. User access without downloading additional software has been enabled. Data from web pages have been received and processed. Development has been based on this methodology of its own client-server system, which is used to parse and collect the information presented on web pages. Analysis of cloud technology services for further deployment of data collection system from web pages has been carried out. Assessment and analysis of the viability of the system in an autonomous state have been deployed in the cloud service during long-term operation.**

*Index Terms***: web page analysis, big data, client-server model, Selenium, Python.**

INTRODUCTION

With the introduction of big data in our modern business model, the need to address, analyze and process data is becoming increasingly important for companies in all industries. As data collection increases, they need to be read and understood. Just as natural languages require translation for effective communication between people, computer languages and language programming also require such processes. This is where parsing data begins. In the simplest data analysis, unstructured and sometimes unreadable data is transformed into structured and easy to read one. Whether you work in a team of company developers or take responsibility for customers, such as marketing roles, understanding data and how they are transformed, it is important for long-term business success [1].

Data parsing is the process of taking data in one format and converting it to another format. You will find parsers that are used everywhere. They are in compilers when we need to parse computer code and generate machine code. This process is very rapid in today's world, important for today's digital transition. Therefore, data collection and processing, in the current realities, play an important role in all services.

Data parsing includes lexical and parsing. In general, this can be characterized by the process of analyzing a string of characters in a language that conforms to the rules of formal grammar. Analysis in terms of data analysis extends this definition to a two-step process that shows a software parser that can be read, analyzed, or converted. The result is a more structured format.

An important aspect of suitability for purposes is the structure of the data in which they are present. Often the structure itself is not suitable for data needs. For example, a data acquisition system does not have fields for each piece of information with different uses, which leads to user detours, such as entering some individual pieces of information into a single free text field or using incorrect fields for information that has no obvious place of information. Abo-words need to be flipped to a new system with a different data structure. It can also be serviced if you want to remove duplicate data, and it is difficult to delete them due to the data structure.

In addition, the data structure may be justified, but its use is insufficiently controlled or may be erroneous. For example: Users did not hesitate to select all the necessary information that causes problems, such as entering contacts using "data cheats" instead of real names in the name fields. Or the application displays the fields in an illogical order, which leads to the fact that users enter the wrong fields [2].

FORMULATION OF THE PROBLEM

A key factor in the study is the awareness of the need to optimize the algorithm, create an uninterrupted system with automatic recovery, and minimize the chance of data loss. One approach to developing similar software and hardware is to use information designed for consumption by ordinary users, this nuance dramatically changes the algorithm, which must be complex enough to understand the structure of pages written for people.

ANALYSIS OF RECENT RESEARCH AND PUBLICATIONS

Encrypted search and clustering in data ironing were demonstrated in March 2014 at the American Society for Engineering Education. Gautam Sivach, who studied data parsing at the Massachusetts Institute of Technology's Computer Science and Artificial Intelligence Laboratory, and Amir Esmailpur of the ENT Research Group explored key features of data parsing as clusters and their relationships. They focused on security of data parsing and term orientation on the availability of different types of data in encrypted form on the cloud interface, providing raw definitions and real-time examples within the technology. In addition, they proposed an approach to identify coding techniques and advance an accelerated search for ciphertext, which will improve the security of data parsing.

In March 2012, the White House announced the National Data Parsing Initiative, which consisted of six federal departments and agencies that allocated more than $ 200 million to data parsing research projects.

The initiative included a $ 10 million National Science Foundation Expedition in Computing for five years for AMPLab at the University of California, Berkeley. AMPLab has also received funding from DARPA and more than a dozen industry sponsors and is using big data to attack a wide range of issues, from congestion forecasting to cancer control.

The White House data parsing initiative also involved a commitment from the Department of Energy to provide $ 25 million in funding over five years to establish the Institute for Scale Data Management, Analysis and Visualization under the direction of Lawrence Berkeley Laboratory. The SDAV Institute seeks to combine the experience of six national laboratories and seven universities to develop new tools to help scientists manage and visualize data on the department's supercomputers [3].

In May 2012, the US state of Massachusetts announced the Massachusetts Mass Media Initiative, which provides funding from the state government and private companies for various research institutions. The Massachusetts Institute of Technology is home to the Intel Data Science and Technology Center of the Massachusetts Institute of Technology and Artificial Intelligence, which combines government, corporate and institutional funding and research.

The European Commission funds the two-year Big Data public-private forum through its Seventh Framework Program to engage companies, researchers and other stakeholders in big data discussions. The project aims to define a strategy in terms of research and innovation, aimed at supporting the actions of the European Commission for the successful implementation of the big data economy. The results of this project will be used as baseline for Horizon 2020, their next framework program.

In March 2014, the British government announced the establishment of the Alan Turing Institute, named after the pioneer of computers and code switchers, which will focus on new ways to collect and analyze large data sets.

At the Canadian Open Data Inspiration Day (CODE) at the University of Waterloo, Stratford University, participants demonstrated how the use of data visualization can improve the understanding and appeal of big datasets and bring their history to the world.

Computing Social Sciences - Anyone can use the application programming interfaces (APIs) provided by big data owners, such as Google and Twitter, to research the social and behavioral sciences. Often these APIs are provided for free. Tobias Price et al. used Google Trends data to show that Internet users from countries with higher GDP per capita are more likely to search for information about the future than information about the past. The findings suggest that there may be a link between online behavior and real-world economic performance. The authors of the study examined the Google search logs, made by the ratio of search volume [4].

BASIC CONCEPTS OF WEB PARSING

Web parsing is a method of obtaining a large amount of publicly available data from websites. It automates data collection and converts the collected data into formats of your choice, such as HTML, CSV, Excel, JSON, txt. This process primarily consists of 3 parts:
- HTML page analysis
- Data extraction
- Data storage

The most important thing in data parsing is programming. Because of this, many companies need to hire experienced developers to crawl websites. Whereas for those who do not have a large budget and do not have coding skills, tools for scraping on the Internet will come in handy. Both parsing programming languages and the use of web parsing tools have some common advantages. Among the main advantages of data parsing are:

- Data extraction is automated. Copying and pasting data manually is absolutely painful. In fact, it's simply impossible to copy / paste large amounts of data when you need to regularly extract from millions of web pages. Parsing programs can automatically retrieve data with zero human factor.

- Speed. When work is automated, data is collected at high speed. Tasks that used to take months can now be completed in minutes.

- The information collected is much more accurate. Another advantage of a web scraper is that it greatly improves the accuracy of data retrieval as it eliminates human errors in the process.

- This is a cost-effective method. A common myth about scraping a network is that people need to either learn to code on their own or hire professionals, and both require a lot of time and money. The opposite is true: encoding is not required to remove websites, as there are dozens of scraper tools and services available on the market. In addition, it is an affordable solution for companies with a limited budget. Some web scraper tools offer free small volume recovery plans, and the market price for high volume data extraction does not exceed $ 100 per month.

- Obtaining clean and structured data. Data collection is usually cleaned and reorganized because the data collected is not structured and ready for use. Web scraper tools convert unstructured data into structured data, and web page information is reorganized into presentable formats [5].

The scope of parsing can be reduced to two purposes: analyzing competitors to better understand how they work and to borrow some approaches from them; analysis of own site for elimination of errors, fast introduction of changes, etc.

Now let's look at what other purposes you can use parsing for.

Market research. Parsing allows to quickly assess what products and prices competitors have.

Analysis of the dynamics of change. Parsing can be performed regularly to assess how any indicators have changed. For example, prices have risen or fallen, the number of online ads or forum posts has changed.

Elimination of shortcomings on own resource. Detection of errors in meta tags, broken links, problems with redirects, duplicate elements, etc.

Collect links leading to your site. This will help evaluate the work of the link building contractor. How to check external links and what tools to do it is described in detail in the article. An example of such a report: Parsing is, an example of a parser report, link parsing, site parsing

Filling the catalog of the online store. Usually, such sites have a huge number of items and it takes a long time to make a description for all products. To simplify this process, foreign stores are often scolded and simply translate product information.

Compilation of the client base. In this case, parsing contact information, for example, users of social networks, forum participants, etc. But here it is worth remembering that the collection of information that is not publicly available is illegal.

Collection of reviews and comments on forums, social networks. Creating content that is based on data sampling. For example, the results of sports competitions, infographics on changes in prices, weather, etc.

By the way, unscrupulous people can use parsers for DDOS attacks. If you start parsing hundreds of pages of the site at the same time, you can "put" the site for a while. This, of course, is illegal - more on this will be stated below. You can protect yourself from such attacks if the server is protecte [6].

DATA EXTRACTION TOOLS FROM WEB PAGESWARE

Web scraping is a technique for collecting page data to enter that data into a search engine database. Scraping consists of two stages: systematic search and loading of web pages, and extraction of data from web pages.

You can create a search engine from scratch with the help of various modules and libraries provided by the programming language, but in the future - as the program grows - this can cause a number of problems. For example, you will need to reformat the extracted data to CSV, XML, or JSON. You may also encounter sites that require special settings and access models to analyze.

Selenium is a free (open source) automated testing platform used to test web applications in different browsers and platforms. You can use several programming languages, such as Java, C #, Python, etc. to create Selenium test scripts. Testing with a selenium testing tool is commonly referred to as selenium testing.

This tool consists of many modules, for example Selenium WebDriver, Selenium RC, Selenium Server, Selenium Grid, Selenium IDE. However, to achieve our goal, we need to use a tool called Selenium WebDriver.

The Selenium WebDriver tool is used to automate web application testing to make sure it is working properly. It supports many browsers such as Firefox, Chrome, IE and Safari. However, using Selenium WebDriver, we can automate the testing of web applications only. It does not meet the requirements for window programs. It also supports various programming languages such as C #, Java, Perl, PHP and Ruby for writing test scripts. Selenium Webdriver is platform independent, as the same code can be used in different operating systems, such as Microsoft Windows, Apple OS and Linux. It is one of the components of the selenium family, which also includes Selenium IDE, Selenium Client API, Selenium Remote Control and Selenium Grid.

Selenium WebDriver does not handle window components, but this limitation can be overcome with external tools such as AUTO IT tool, Sikuli, etc. It also has various location strategies such as ID, Name, Link Text, Partial Link Text, Class Name, CSS Selector and Xpath. It also has better support for dynamic web pages, such as Ajax, where web page elements can be changed without reloading the page itself. Using various jar files, we can also test APIs, database tests, etc. using Selenium WebDriver [7].

USING DATA EXTRACTION TOOLS FROM WEB PAGES

We will use the Python language to implement a client-server system of data parsing with minimal resources and time.

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic text input and dynamic binding, make it very attractive for rapid application development, as well as for use as a scripting language or glue to connect existing components. The simple, easy-to-learn Python syntax emphasizes readability and, therefore, reduces program maintenance costs. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and a large standard library are available in source or binary for free on all major platforms and can be freely distributed [8].

In order to create scripts for parsing data from web pages, we need to take a few steps to customize our project. The first step is to add a selenium web driver library to the project. Next, create an instance of Chrome with the path to the driver, which was downloaded through the websites of the appropriate browser. Next, use the driver's .get () method to load the website. We can also load the local development site, as this process is equivalent to opening the Chrome window on the local machine, entering the URL and pressing Enter. The .get () method not only starts loading the website, but also waits for it to be fully visualized before proceeding to the next step. After successfully loading the page, you can use the .title attribute to access the text title of the web page [9].

That is, in order to propagate data from a web page, we need to create a browser driver that will mimic a normal web browser. Next, you need to specify the address of the site we need, and using the commands of this library to get the necessary elements of the page.

USER INTERFACE USING A THIRD-PARTY API

A chatbot is a program that automatically responds to a person in correspondence by responding to keywords in a given scenario. For example, a customer asks in a chat: "What is your warehouse address?", And the bot responds to the words "address" and "warehouse" and instantly gives a pre-registered answer. This saves a lot of support time and allows to focus on more important tasks. Another advantage of the bot is that it works around the clock, which is beneficial for business. If the customer wants to know the availability of the product or pays for it immediately at four in the morning - the program will help him. Chatbots can be connected to almost any popular messenger or social network.

To create an easy user interface with high speed, we use the Telegram API, and build an automated user interface based on it. This solution offers two types of APIs for developers. The Bot API makes it easy to create applications that use Telegram messages for the interface. The Telegram and TDLib APIs allow to create your own custom Telegram clients.

This API allows to connect bots to the Telegram system. Telegram bots are special accounts that do not require an additional phone number. These accounts serve as an interface for code running somewhere on your server.

You don't need to know how MTProto encryption works to take advantage of this - the proxy server will handle all encryption and communication with the Telegram API. Communication with this server is via a simple HTTPS interface, which offers a simplified version of the Telegram API. In essence, Telegram bots are special accounts that do not require an additional phone number. Users can interact with bots in two ways:

- Send messages and commands to bots, open a chat with them or add them to groups.

- Send requests directly from the input field by entering @username bot and request. This allows to send content from embedded bots directly to any chat, group or channel.

Messages, commands, and requests sent by users are transmitted to software running on your servers. Our proxy server handles all encryption and communication with the Telegram API for you. You communicate with this server through a simple HTTPS interface that offers a simplified version of the Telegram API. We call this interface our Bot API [10].

USING CLOUD PLATFORMS TO DEPLOY THE SYSTEM

To deploy our system, we need to choose a suitable platform for this. Of the services available on the market and suitable for our lightweight system, we can distinguish two companies from Amazon Web Services and Google Cloud. Let's take a closer look at them.

Amazon Web Services (AWS) is a cloud platform that provides subscription services to users. The list of services includes both infrastructure solutions and ready-made platforms that together form this platform (Fig. 1).

Elastic Compute Cloud (EC2). The service provides subscribers with virtual server platforms, storage systems and a load balancer. Users can choose either a pre-configured server with a pre-installed operating system, or build it themselves. The service also allows to create your own images or use your own OS. To ensure security, subscribers can differentiate access to EC2 servers by IP addresses.

The cost of the service is paid for each hour, and some options are paid on a monthly basis. If the client plans to use EC2, then it is recommended to use the reservation function. The subscriber pays for 3 months of work at once, and the total cost becomes one and a half times lower.

*Fig. 1. Services included
in the AWS complex*

Simple Storage (S3) provides subscribers with distributed disk space with a maximum capacity of 5 TB. S3 stores files in so-called buckets, which are located at different Amazon sites.

The service maintains a permanent log of all operations on the storage and stores information in a special bucket. This allows customers to immediately receive detailed information on the latest transactions. A distinctive feature of the service is the replacement of the http protocol with BitTorrent for downloading files.

Subscribers can independently choose the security policies that apply to buckets. This allows to differentiate user access rights, make the cloud completely private, etc. Payment for the service is charged monthly. The cost depends on the amount of disk space, the number of requests and outgoing traffic.

The Relational Database Service provides users with a virtual database that resides on a dedicated server. At the same time, the platform is configured and optimized for the selected database. The minimum disk space for a database is 5 TB.

Access is configured depending on the wishes of users and security policies. For example, customers can only allow connections from certain IP addresses (subnets) or for those security groups that are specified in the EC2 service.

Subscribers have access to SnapShot, replication (asynchronous and synchronous), automatic data backup and metro cluster.

Payment is made similarly to the EC2 service - hourly. If you pay three or more months in advance, then the total cost is reduced by one and a half times. Additional functions are paid separately.

Route 53 gives users the ability to set up a DNS server in the Amazon cloud. The service integrates seamlessly with other AWS services.

Payment is charged for the number of requests to the server, but there is also a free limit.

Amazon Web Services provides most cloud solutions to users at an affordable price. Reliability and resiliency are guaranteed at the six nines level. The client can flexibly configure the required list of available services, choosing only the necessary ones. Each service has its own personal limit, which significantly reduces the cost of the service [11].

The Google Cloud Platform is a collective set of cloud computing services offered by Google itself. It includes a range of computing and hosting services for storage and application development that can run on Google hardware. In short, it is a collection of public cloud services in which all tests and patches are regularly updated by Google. Its services can be easily accessed by developers, cloud administrators, or other IT professionals on the public network through any other dedicated network connection. GCP also offers services for better storage, cloud computing, big data, and more.

The main functioning of the Google cloud platform includes:

Google's compute engine, also called infrastructure as a service (IaaS), provides users with virtual machine instances to host workloads.

The Google App engine, also referred to as Platform as a Service (PaaS), offers software developers access to scalable Google hosting. Once this is done, developers can easily use the SDK to develop products powered by the Google engine.

Benefits of Google Cloud Platform: Better value than other competitors. Google will only let you pay for the time you use to compute, and the advantage is that they even offer discounts in case of long schedules or workloads. This proforma is widely used in the development of SaaS applications. It also doesn't require any upfront commitments, and you can even get a discount for monthly VM usage. This feature made it suitable for startups or platforms that are willing to spend less. When it comes to performance versus price, it is best to rely on Google's cloud platform as it is much cheaper than other instances.

Private world fiber. Imagine how good cloud integration services are if they can't handle the bulk? There is a feature that sets Google's cloud hosting platform apart from other competitors, and it is the

largest network in the world. It also comes with a private distribution backbone between all data centers. In 2016, Google announced a faster cable system that gave them access to 10 Tbps of total cable bandwidth between countries like us and Japan. We are now using the same technology for Google Cloud Platform and Google App Clients. Having this infrastructure allows users to pay for available computing resources.

Live migration through virtual machines. This is another major benefit of Google Cloud Hosting, especially how it is used for live migration of virtual machines. No other competitors, such as AWS or Azure, provide features that can be offered to users, so this is what sets GCP apart from others. This allowed engineers to resolve issues such as repairs, patching to update hardware and software without having to reboot the machine. In addition to real-time migration, Google also offers resizing persistent drives with no downtime. To actually resize a live disk, you can easily use the portal or command line interface.

Increased productivity. Even at the enterprise level of big data application services, other cloud services are grateful to Google for their excellent uptime. It can easily handle a lot of concurrent visitors without any interruption. If it is absent, sites can experience spikes, and GCP has significantly reduced them to half. Google engine service provides automatic scalability instead of constant for limited CPUs of more than 1 or 2 VPS. Google's cloud platform is making a breakthrough when it comes to dramatically increasing the number of cloud computing services.

Security concerns. No other benefit can be as satisfying as security, and Google's cloud platform offers a security feature without any inconvenience. It easily protects services like Gmail when searching or transferring [12].

Having considered both services in more detail, and comparing the services they provide, we can conclude that a set of services from Google is best for our system. Because the competitor's platform has many services that are redundant for the task. Price policy also plays an equally important role. Because Google provides a trial period which includes free use of services in the amount of $ 300. It allows to choose the characteristics of the virtual machine flexibly to our requirements, and get a complete picture of the use of this system without incurring additional costs, which is certainly a big plus to the choice of this platform.

Also, in favor of this solution is that Google Cloud Platform has a simple and easy interface (Fig. 2). It simplifies the configuration of all necessary system configurations.

And no less important is the ability to access the command line directly from the browser. It allows to do without the use of third-party services during system setup.
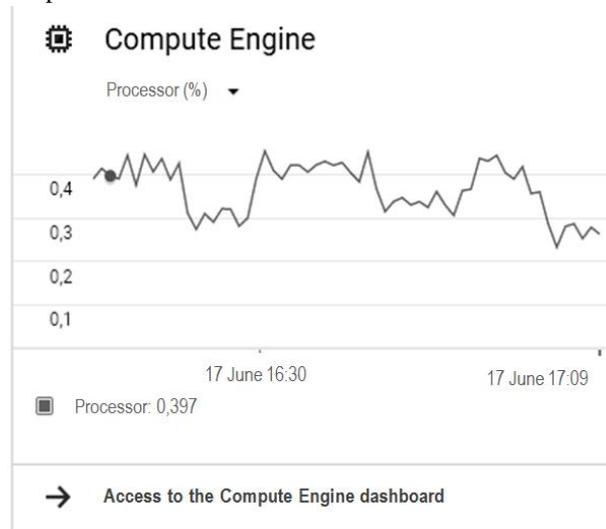

Fig. 2. Example of the Google Cloud Platform interface [12]

ANALYSIS OF THE VIABILITY AND AUTONOMY OF THE SYSTE

Having built this system using all the above tools, we got a parsing system that works as follows: the user specifies the necessary link to the site, and keywords by which to analyze the data, in the telegram client. These parameters are stored in the database. Then you plan to analyze this web page every half hour.

When analyzing and parsing a page, this system selects the desired elements of the web page according to the specified keys. Then it gets textual information from all the elements that fall under the specified keywords. The next phase is called the filtering one, during which it checks whether this page element is a link and whether the text in it matches the required filters. In the case of a positive answer, the system goes to the link and checks whether the text inside is filtered.

The next step is to package all the items that have been filtered into packets and send them to the automated user interface, where this data is displayed to the user. At each stage, the current state of the system is recorded (Fig. 3).

Considering all mentioned above, you can see that the system is quite autonomous and does not require constant supervision by the user. It allows to collect data in the background and in bulk.

Long-term viability studies have also been conducted in the background. It showed that the system worked continuously for a month. All internal errors

were handled by the system itself and, in case of critical failures, the system resumed its own work.

```
04-26 22:02 [PROXY][THREAD: 1 ] Proxy works: 46.4.96.137:8080
04-26 22:02 [PROXY][THREAD: 3 ] Proxy works: 207.154.231.213:8080
04-26 22:02 [PROXY][THREAD: 6 ] Proxy works: 180.252.181.3:80
04-26 22:02 [PROXY][THREAD: 0 ] Proxy works: 102.129.249.120:8080
04-26 22:02 [PROXY][THREAD: 0 ] Proxy works: 151.253.165.70:8080
04-26 22:02 [OK][PROXY] Found: 7
04-26 22:02 [OK][DRIVER] Driver with proxy is initialized
04-26 22:02 [OK] Publication is opened: https://www.olx.ua/obyavl
04-26 22:02 [ERROR] Can't find element by ID on: https://www.olx.
04-26 22:02 [WAIT][DRIVER] Prepearing proxy driver..
04-26 22:02 [OK][DRIVER] Driver with proxy is initialized
04-26 22:02 [ERROR] Can't find element by ID on: https://www.olx.
04-26 22:02 [WAIT][DRIVER] Prepearing proxy driver..
04-26 22:02 [OK][DRIVER] Driver with proxy is initialized
04-26 22:02 [ERROR] Can't find element by ID on: https://www.olx.
04-26 22:02 [WAIT][DRIVER] Prepearing proxy driver..
04-26 22:02 [OK][DRIVER] Driver with proxy is initialized
```

Fig. 3. Record of the current state of the system

CONCLUSION

This article described the main key factors in the development of an automated system for parsing information from web pages. A test program that allowed to read information and send analytical data to the user interface was developed. A key feature was the direct focus not only on working with strictly structured data, but also with information, that did not have a specific structure. In addition, this information product included self-recovery owing to unplanned failures, and the original parsing process at the moment of failure from the algorithm with previously saved information.

The conducted test studies helped to show the full effectiveness of this approach, which makes the described probable software and hardware system competitive in today's market of big data programs.

REFERENCES

[1]   Cukier, K. (2017). *Big data : a revolution that will transform how we live, work and think*. London: John Murray., 280 p.

[2]   O'neil, C. and Schutt, R. (2013). *Doing data science. Beijing*. Cambridge: O'reilly., 510 p.

[3]   Sweigart, A. (2020). Automate the boring stuff with Python : practical programming for total beginners. San Francisco, Calif. No Starch Press., 357 p.

[4]   Selenium. (n.d.). *The Selenium Browser Automation Project*. [online] Available at: https://selenium.dev/documentation/.

[5]   Espinosa-Leal, L. (2018). Special issue of Big Data Research Journal on "Big Data and Neural Networks.". *Big Data Research*, 11, pp. 120–130.

[6]   Williamson, E.P. (2017). Fetching and Parsing Data from the Web with OpenRefine. *The Programming Historian*, pp. 6–15.

[7]   Holden, G. (2016). Big Data and R&D Management. *Research-Technology Management*, 59(5), pp. 22–26. DOI:10.1080/08956308.2016.1208044

[8]   Kumar, S. and Singh, M. (2019). Big data analytics for healthcare industry: impact, applications, and tools. *Big Data Mining and Analytics*, 2(1), pp. 48–57.

[9]   Gardner, F.M. (1998). HTML Sourcebook: A Complete Guide To HTML 3.2 And HTML Extensions [Book Reviews]. *IEEE Communications Magazine*, 36(6), pp. 26–28. DOI: 10.1109/MCOM.1998.685344

[10] Varshith, K. (2020). Software Virtualization using Containers in Google Cloud Platform. *International Journal of Innovative Technology and Exploring Engineering*, 9(4), pp. 802–804.

[11] Itglobal.com. (n.d.). Amazon Web Services (AWS): *platform responsibilities*. [online] Available at: https://itglobal.com/ru-ru/company/glossary/amazon-web-services [Accessed 7 Nov. 2021].

[12] Hamed, P.K. and Preece, A.S. (2020). Google Cloud Platform Adoption for Teaching in HEIs: A Qualitative Approach. *OALib*, 07(11), pp. 1–23. DOI: 10.4236/oalib.1106819

Artur Britvin is a sixth-year computer engineering student of Lviv Polytechnic National University. Since 2020, he has been working in SoftServe Inc. His research interests include client-server systems, .Net performance, designing scalable systems on micro services basis.

Rostyslav Tkachuk – Doctor of Technical Sciences, Associate Professor. In 1993 he graduated from Ivan Franko State University of Lviv, Lviv. In 1995 he graduated from the Institute of Advanced Training and Retraining of Ivan Franko State University of Lviv, Lviv. From 1995 to 2008 - practical work in various positions at Lviv State University of Life Safety, Lviv. From 2008 to 2019 - Associate Professor of Lviv State University of Life Safety.

Since 2020 - Head of Information Security Management Department of Lviv State University of Life Safety. Research interests: information-logical and cognitive technologies of decision-making in conditions of risk; information technologies in training personnel for action in extreme conditions; information security.