ISSN No:-2456-2165

The Influence of Multi Platform Space on the Formation of a Programming Languages Stack in the Competence-Based Approach to Computing Training at Universities

Gregory Korotenko
Doctor of Engineering
Department Of Geoinformation Systems
Dnipro University of Technology
Dnipro, Ukraine

Leonid Korotenko
Associate Professor
Department of Software Engineering
Dnipro University of Technology
Dnipro, Ukraine

Abstract:- At the present time, there exist a significant gap between educational attainment acquired by students at Universities and the trends of modern information technologies development, as well as an increase in the number of levels of computing platforms and their ecosystems. The analysis of the most highly paid jobs, based on survey of Stack Overflow developers, has shown that those in most demand are : DevOps specialists, Data scientists, Back-end developers, Full-Stack developers, Embedded developers. Training programs, however, continue to be improved in five key educational areas: Computer Science (CS), Computer Engineering (CE), Software Engineering (SE), Information Systems (IS), Information Technology (IT). The level of future specialists' competence is constantly improving through improvement of university programs developed under the aegis of organizations, i.e. the Association for Computing Machinery (ACM) and the Institute of Electrical and Electronics Engineers (IEEE). That is why it is important to explore the opportunities for improvement of modern students' competency level through the development of curricula based on the formation of a programming languages stack, necessary for their further integration into the processes of society informatization.

Keywords:- *IT platforms, profiles, training courses, programming language stack.*

I. INTRODUCTION

The ever-expanding scope of hardware and computer technology application in all spheres of human activity has led to a shortage of labour resources for the newly created specialties and jobs, and also has set a target to increase the required skill level of the personnel required. The US government circles have initiated a research and development of program called STEM (Science, Technology, Engineering, Mathematics) in this regard. This also has reflected the fact that, for example, the period from 2000 to 2010 in the United States has demonstrated the growth of jobs associated with the new directions of science and technology development to be three times higher than the growth rate of jobs not related to STEM [1].

Many reports and programs prepared by specialists summarize that the need for labour resources of this level will grow constantly. According to *STEMconnector.org*, the need for 8.65 million employees with expertise in system innovation involving STEM elements is foreseen [2]. For example, 1.7 million jobs have been created in just one field of Cloud computing during the period between 2011-2015. The US Bureau of Labour Statistics reports the structure of activities for STEM application to be as follows [2]:

- Computing – 71%;
- Traditional engineering – 16 %;
- Physical sciences – 7%;
- Life sciences or biological sciences – 4%;
- Mathematics – 2%.

Accordingly, computing is a critically important and integral component of modern scientific, technical and industrial technologies [3, 4].

In the early stages of the emergence and development of computer technology, the computing platform or digital platform meant "... an environment in which a piece of *software* runs; this can be a hardware or an operating system (OS), a web browser and related application programming interfaces or other basic software, if they enable the *program code* to be executed. Accordingly, the computing platform is a functionality that provides an ability to run *computer programs*" [5]. However, the unprecedented technological development of information and communication processing means for their delivery has significantly extended the meaning of this term, including for business enterprises. Currently, the concept of an information technology (IT) platform is very broad and covers almost all computer developments, ranging from numerous, including mobile, operating systems and ending with the Internet infrastructure [6]. A variety of platforms are becoming increasingly important for innovation and the creation of intellectual property in many industries and everyday life. At the same time, the idea of what the term platform means in the field of IT application and, accordingly, computing, continues to be refined. A systematic review of literary sources, including 132 titles taken from materials of large information systems, journals, conferences and business publications has been carried out

ISSN No:-2456-2165

in order to form the conceptual framework for understanding of this term, and in order to define the concept of IT platform in the work [6]. The conceptual definition of an IT platform in the study has been formulated as follows: "The IT platform is a technological base for the interaction of software add-ons (standards) based, in their turn on the standards and allowing to carry out transactions between stakeholders within a platform-oriented ecosystem."

Additional study of the definition elements clarifies its constituent parts [6]:

- Technological base - a basic set of applications and hardware that implements data stream processing technologies for the functioning of the main services of the platform;
- Software add-ons - software, software customization codes, modules and applications;
- Platform-oriented ecosystem being an integrated complex of the platform itself and the modules specific to this platform.

Thus, it appears that the platforms not only create constantly developing ecosystems around themselves, but also provide active interaction between the expanding spectrum of more and more platforms and their ecosystems.

Taking into account worldwide trends, in the year 2013, one of the world's leading companies for researching trends in the IT industry, IDC, introduced the "*third platform*" term. This word formation, according to the opinion of the authors and a number of other specialists, was supposed to describe a conglomerate of the progress driving forces in the development of modern digital business, based on the following elements [7, 8, 9]:

- Mobile devices;
- Cloud Services;
- Social Networks;
- Analytical procedures applied to Big Data.

According to the classification, the next (fourth) platform included:

- Artificial Intelligence, AI;.
- Internet-of-Things (IoT) and some other elements.
- Further events have shown a steady trend towards the development of this direction, as a result of which additional names appeared in this phenomena at the junction of directions.
- SMAC (social, mobile, analytics and cloud) [10];
- Nexus of Forces: Social, Mobile, Cloud and Information [11].

The general view of the "four platforms" by IDC, Gartner, and many others specialists, which we call "conceptual" includes the first mainframe line, which originated in the 1950s, and continues to be used up to present day. The second conceptual platform is the client-server architecture, which began its development in the 1980s, the third is the SMAC platform, and finally, the fourth is AI + IoT. In turn, the lower level platforms included in the conceptual ecosystems will be called "technological". According to sources [12-17], the total number of "technological" platforms at the moment exceeds 400 units (Fig. 1).

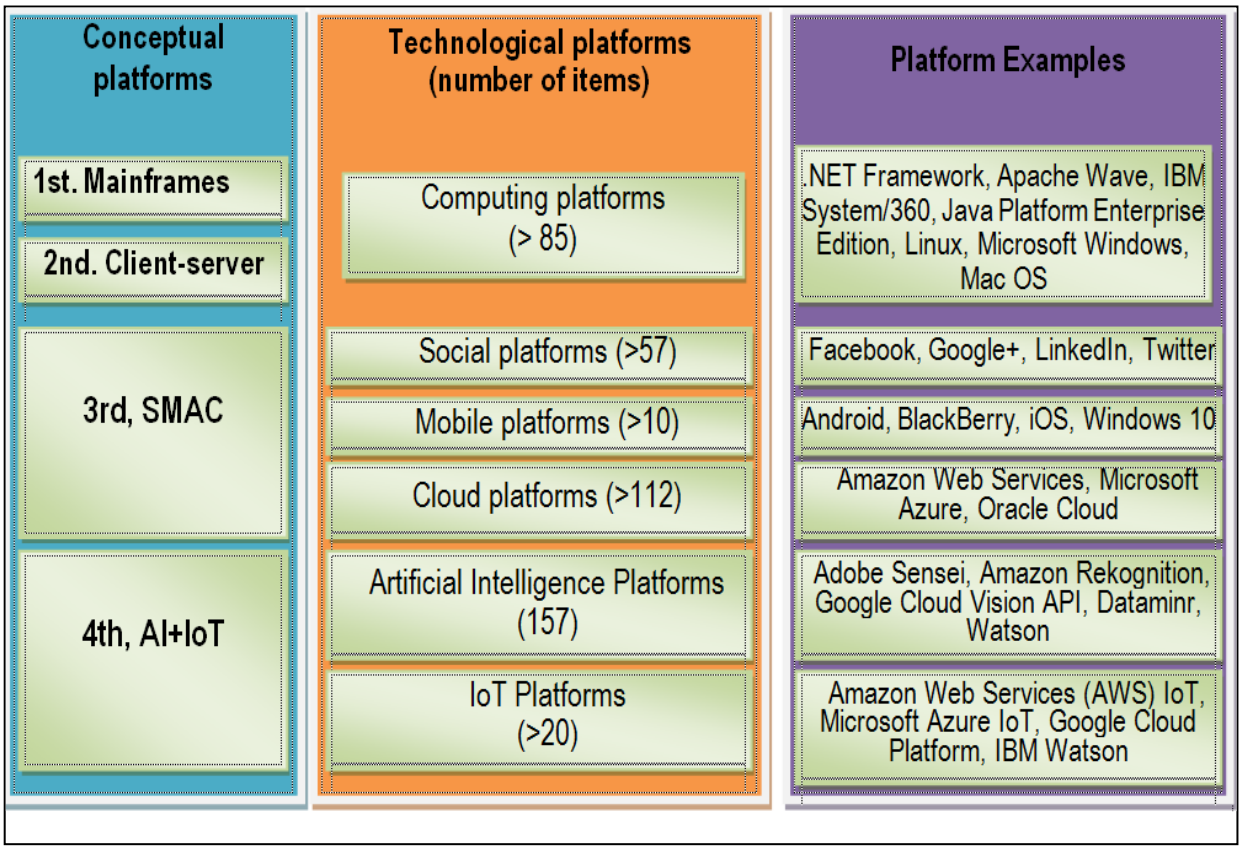

Fig 1:- The ratio of the qualitative and quantitative composition of platforms computing

ISSN No:-2456-2165

The level of abstractions forming this representation does not allow to present the other components in more details, including hardware, computing, software, technological and some other types of platforms, as well as software development platforms for the above components, frameworks, APIs and their interconnections.

One of the most important conclusions to be drawn from consideration of the interaction of the components of the four conceptual platforms, is that each new platform includes or absorbs the previous one at the level of combining their ecosystems and, accordingly, combines their functionality (Fig. 2).

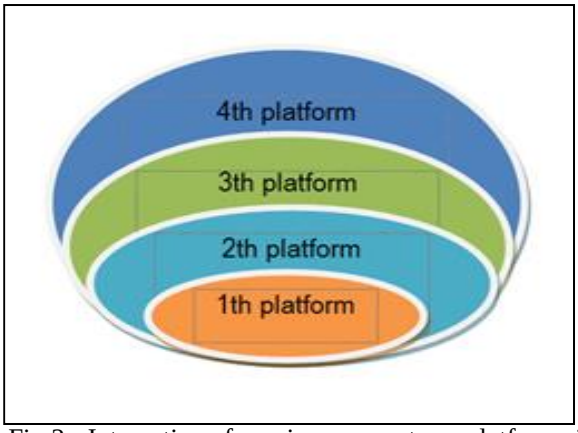

Fig 2:- Integration of previous ecosystems platforms in interaction with subsequent

Based on the above mentioned, it can be concluded that computing platforms include different levels of abstractions in computing sphere, including the architecture of computers, operating systems (OS) and run-time libraries. In other words, computer platforms are hardware-software systems that provide the basic services sets that are needed by the users to perform tasks through *computer programs* launching. Consequently, each platform, as a rule, is based on one or several software programming languages [5].

Long enough, attempts have been made to assess the influence of various factors on the quality of program projects implementation subject to an explosive increase in the number of software developments and the participating programmers therein. In particular, one of them led to a creation of the COCOMO model (COnstructive COst MOdel– model of development costs), which was published for the first time in 1981 in the book by Barry W. Boehm "The real economics of software development" [18] and was indicated as a model for making performance analysis, estimating production costs and developing scheduled plan for software development projects. It has been used to study 63 projects at the TRW aerospace company, in which Barry W. Boehm headed a Software and Technology Research Department.

The study classified projects by size, depending on the number of lines of code (from 2 to 100 thousand), as well as by software programming language (from assemblers to the high-level PL/l language). Based on obtained data processing, the author has concluded that the impact of knowledge of the programming languages knowledge on the quality and the speed of software development at all stages is more significant than the influence of experience of working on a technical platform (the values of influence coefficients are 1.43 and 1.40, respectively [19]).

According to a study done by Timothy C. Lethbridge in 1998 and based on a survey of more than 200 experienced programmers [20], the "Specific programming languages" (3.8) and the "Data structures" (3.6) became the highest rated subject areas (the first and the second place with the corresponding number of points) among 75 training courses held in almost all US higher education institutions.

It may therefore be assumed that the study of specific programming languages is one of the most important components of training specialists at Universities for their subsequent work with software systems and products.

However, the issue of the programming languages stack studied at Higher Educational Institutions and Universities remains open, i.e. the sets of specific programming languages recommended for learning, taking into account the presence and active interaction of the four computing platforms.

II. A REVIEW OF THE LITERATURE

The study of the programming languages stack formation process, from the point of view of the study of them, on the basis of the corresponding curricula at Universities and based on their usage in the structure of existing and newly created computing platforms (IT platforms), can be represented as follows (Fig. 3).

The most active organizations that are constantly expanding the boundaries of the Body of Knowledge (BOK) in computing areas of study, and, accordingly, improving the competencies in this area, are the organizational structures of the United States: the Association for Computing Machinery (ACM) and the Institute of Electrical and Electronics Engineers (IEEE) [3]. Their training areas differentiation, consistent with the nature of the IT specialists' activities in various areas, has defined the following five basic profiles (referred to computing disciplines in CC 2005 [4]): Computer Science (CS) [21]; Computer Engineering (CE) [22], Software Engineering (SE) [23], Information Systems (IS) [24], Information Technology (IT) [25].

Each of these profiles has been designed to combine training courses to prepare students and provide them with the relevant competencies in the field of computing [26]. Therefore, the introduction of unified requirements for the integrated competence of IT specialists graduated from Universities, as laid out in professional and educational standards of education of all countries, is of great importance [2].

A review of the fundamental documents on the five main areas of computing education [21] has been carried out to study the features of courses involving the study of programming languages and those recommended for learning by University curricula.

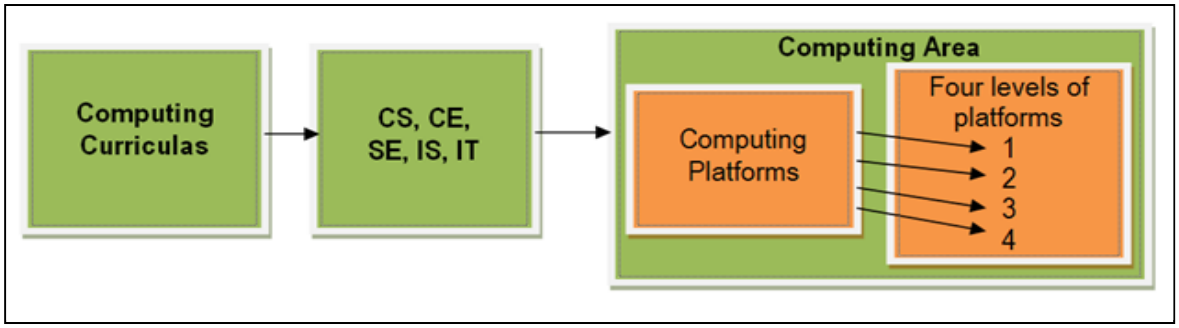

Fig 3:- Connection of curricula and their courses (disciplines) with computing platforms

The following points have been clarified at this stage.

- The body of knowledge (BoK) in a Computer Engineering Curricula (as revised in 2016) [22] does not contain instructions for specific programming languages or operating systems (p. 27). However, the description of courses indicates the study of assembly language (assembly language programming), the two languages for the description of hardware, such as the hardware description language (HDL) and Verilog, as well as SQL.
- The Computer Science Curricula (as revised in 2013) [21], in particular, indicates that specific languages and technology platforms change over time: therefore, graduates shall understand that they must continue learning and adapt their skills throughout their career (p. 25). Further, the Programming Paradigm and Choice of Language Section on the page 42 has pointed out that the spectrum is expanding and more efficient dynamic (Python, JavaScript) or more manageable (such as Java) programming languages are emerging. The course "Digital Logic and Digital Systems" offers the hardware description language HDL, Verilog, as well as assembly language to be learned. The C ++ language has been mentioned in the Defensive Programming course (p. 105). The optional course "Query Languages" offers SQL, XPATH, SPARQL for learning (p. 116). The optional course "Platform-Based Development" offers Objective C and HTML5 training, and the course "Web Platforms" offers HTML5, JavaScript, PHP, CSS (p. 143). The optional course "Game Platforms" has C ++, Java, Lua, Python indicated for study (p. 144).
- The Software Engineering Curricula (as revised in 2014) [23] states that although software developers currently continue to perform many of the same activities as in the 1960s, but they do so in a completely different context (p. 13). Not only languages and programming tools used for these activities are different and, as a rule, they are much more powerful; but these actions can also be taken within a much wider range of organizational models. Also, the information of the page 44 indicates that software developers should study several programming languages in detail. The titles of programming languages are not specified in the document itself.
- The Information Systems Curricula (as revised in 2010) [24] refers only the study of the SQL languages: DDL, DML and DCL (p. 41)
- The Computing Discipline Curricula and Information Technology Curricula (as revised in 2017) [25], state that programming competence is not a precondition for good achievements of learning many IT programs. Consequently, the educational level (degree) in the IT industry is quite different from the educational levels in computing or computer science spheres. Additional courses offer the study of R and Python statistical programming languages (p. 103). In Appendix C.4.5. to the "Introduction to Computer Systems" course, the study of the C language is assumed. The Appendix C.6.3. (describing the courses of the Computer Science Department of China) supposes the study of the Java language for two of the courses; and the name of the language is included into the name of the disciplines (JAVA EE Architecture & Application Development, Advanced JAVA Programming). The "iOS Mobile Application Development" course is supposed to provide the Objective-C language study, and the "Cross-platform Script Development Technology" course is supposed to provide the JavaScript language study. The Appendix D.5 lists the Geneva Information Technology Institute Program where the following languages are indicated for study: HTML, CSS, C ++, VB.NET, SQL, PHP and Java.

Accordingly, the issue on recommendations for the study of the specific programming languages in the main discipline programs of all the five documents remain open.

III. RESEARCH METHODS

This study was designed as a research with document analysis methods.

IV. RESEARCH RESULTS

At the same time, the rapid emergence of all new information technologies and platforms commits Universities to include not only theoretical, but also practical materials into the content of new training courses for the formation of relevant competencies. Then there arises a question about availability of the stack of programming languages being able to provide the best acquaintance with the subject area on the one hand, and to meet the requirements of the labour market on the other hand.

A Table 1 has been prepared showing the ten most popular programming languages based on the sources [27-30], which are the most well-known web resources and assess the so-called "popularity" of programming languages, and, in fact, the level of demand for and use of these powerful tools.

Rank	TIOBE	Stack-overflow'	GitHub	Croud-Sources	IEEE Spectrum	Rating
1	Java	JavaScript	JavaScript	JavaScript	Python	Java – 2,6
2	C	SQL	Python	TypeScript	C	C – 6,0
3	C++	Java	Java	Java	Java	C++ – 6,2
4	Python	C#	Rust	Python	C++	Python – 3,2
5	C#	Python	PHP	Rust	C#	JavaScript- 3,2
6	JavaScript	PHP	C++	PHP	R	C# – 6,4
7	Visual Basic .NET	C++	CSS	Elixir	JavaScript	
8	R	C	C#	C	PHP	
9	PHP	TypeScript	Go	Ruby on Rails	Go	PHP – 6,8
10	MATLAB	Go	C	C#	Swift	

Table 1:- The best programming languages in 2017 according to various sources

The data in Table 1 have been used for the languages present in all five ratings to sum up the numbers of occupied places; and the results have been divided by their total number. According to the results of the assessment, the first three places have been taken, respectively, by: Java, Python and JavaScript, followed by C, C ++ and C # (Fig. 4).

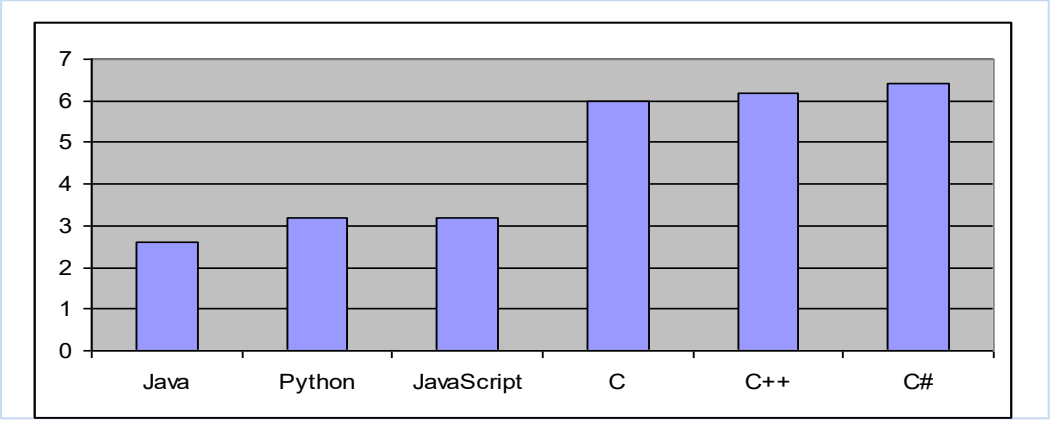

Fig 4:- Rating estimates of the most popular programming languages according to Table 1

Further consideration included areas of application of these languages in various sectors of modern hardware and technical computing applications usage (Table 2). As shown therein, Python the most sought-after language by the number of industries using it.

Programming language	Rating	Web dev.	Mobile App.	Enterprise level	Big data	Machine learning	Artificial Intelligence
Java	2,6	◆	◆	◆		◆	
Python	3,2	◆		◆	◆	◆	◆
JavaScript	3,2	◆	◆				
C++	5,8		◆	◆			
C	6,2		◆	◆			
C#	6,4	◆	◆	◆			
PHP	6,8	◆					

Table 2:- The use of programming languages in various fields of technical applications

V. CONCLUSION AND RECOMMENDATIONS

Therefore, the following conclusions should be made based on research conducted:

- the rapid development of information technologies has led to an unprecedented growth in the number of jobs in the field of STEM industry, as well as in the field of computing and, accordingly, this resulted in a shortage of trained specialists;
- the computing sphere is characterized by a rapid growth in the number of platforms: the number of conceptual platforms has reached four, and the number of technological platforms belonging to their ecosystems exceeds 400 titles;
- each new platform includes or absorbs the previous one, and the ecosystems of the previous and next platforms are actively involved in the integration and convergence process based enhancement of the range of computer tools and software components;
- it is proposed to formulate the contents of the programming language stack in all five computing disciplines taught at universities in the conditions of constantly improving hardware and computer technologies used in all spheres of human activity, an unprecedented increase in the number of software applications and a shortage of labour resources for newly created specialties and a shortage of a required level of personnel skills;
- to introduce a mandatory course of comparative analysis of the most used programming languages [31].
- at the top of the stack, it is proposed to place the programming languages necessary for mastering the necessary courses following them and supporting the necessary courses, and to place the highly specialized languages that meet the specifics of the relevant subdiscipline at the bottom of the stack;
- to recommend the study of four programming languages in all sub-disciplines of computing; they shall include: C ++, Python, JavaScript and Java, as well as language of structural representations WWW HTML5 and cascading style sheets CSS.

REFERENCES

[1]. STEM | Description, Development, & Facts. https://www.britannica.com/topic/STEM-education.
[2]. Smart STEM- Driven Computer Science Education - Springer Link. https:// doi.org/10.1007/978-3-319-78485-4.
[3]. Computing Curricula 2001 Computer Science. https://www.acm.org/education/curric_vols/cc2001.pdf
[4]. Computing Curricula 2005. https://www.acm.org/binaries/content/assets/education/curricula-recommendations/cc2005-march06final.pdf.
[5]. Computing platform – Wikipedia. http://en.wikipedia.org/wiki/Computing_platform.
[6]. Information Technology Platforms: Definition and Research Directions. https://www.researchgate.net/publication/303821844_Information_Technology_Platforms_Definition_and_Research_Directions.
[7]. G. M. Korotenko, L. M. Korotenko, A. T. Khar. Creation of it-oriented ontological framework for the purpose of making educational programs on the base of competencies. https://abdullahnabulsi.files.wordpress.com/2013/01/i-d-c-p-r-e-d-i-c-t-i-o-n-s-2-0-1-3-c-o-m-p-e-t-i-n-g-o-n-t-h-e-3-r-d-p-l-a-t-f-o-r-m.pdf.
[8]. EMC World 2013: EMC hails rise of third platform apps. http://www.itpro.co.uk/apps/19763/emc-world-2013-emc-hails-rise-third-platform-apps.
[9]. Third platform – Wikipedia. https://en.wikipedia.org/wiki/Third_platform.
[10]. Talking SMAC: Revisiting social, mobile, analytics and cloud. http://www.cio.com/article/2918194/cloud-computing/talking-smac-revisiting-social-mobile-analytics-and-cloud.html.
[11]. The Nexus of Forces: Social, Mobile, Cloud and Information. http://www.gartner.com/technology/research/nexus-of-forces/.
[12]. Category:Computing platforms – Wikipedia. https://en.wikipedia.org/wiki/Category:Computing_platforms.
[13]. top 20 iot platforms - Internet Of Things Wiki. https://internetofthingswiki.com/top-20-iot-platforms/634/.
[14]. Top 57 Social Media Platforms Every Marketer Should Know. https://60secondmarketer.com/blog/2010/04/09/top-52-social-media-platforms-2.
[15]. 157 Artificial Intelligence Platforms to Help You Grow Your Business. https://60secondmarketer.com/blog/2018/01/24/157-top-artificial-intelligence-platforms/.
[16]. Mobile operating system – Wikipedia. https://en.wikipedia.org/wiki/Mobile_operating_system.
[17]. Category:Cloud platforms – Wikipedia. https://en.wikipedia.org/wiki/Category:Cloud_platforms.
[18]. Barry W. Boehm. Software Engineering Economics. Englewood Cliffs, NJ : Prentice-Hall, 1981. – 767 p.
[19]. Steve McConnell. Professional Software Development. NJ: Addison Wesley Professional, 2003. – 272 p.
[20]. Timothy C. Lethbridge. Priorities for the education and training of software engineers. // The Journal of Systems and Software, 53, 2000, pp. 53-57.
[21]. Computer Science Curricula 2013 (CS2013). http://ai.stanford.edu/users/sahami/CS2013/final-draft/CS2013-final-report.pdf.
[22]. Computer Engineering Curricula 2016 (CE2016). https://www.acm.org/binaries/content/assets/education/ce2016-final-report.pdf.
[23]. SE2014: Curriculum Guidelines for Undergraduate Degree Programs. https://www.acm.org/binaries/content/assets/education/se2014.pdf.
[24]. IS 2010 ACM final. https://www.acm.org/binaries/content/assets/education/curricula-recommendations/is-2010-acm-final.pdf.

[25]. Information Technology Curricula 2017 (IT2017). https://www.acm.org/binaries/content/assets/education/curricula-recommendations/it2017.pdf.
[26]. Computer Engineering Curricula 2016. http://www.acm.org/education/education/curric_vols/CE-Final-Report.pdf.
[27]. TIOBE Index. https://www.tiobe.com/tiobe-index//.
[28]. 13 Best Programming Languages to Learn in 2017. https://usersnap.com/blog/programming-languages-2017/.
[29]. December rating of programming languages TIOBE. https://tproger.ru/news/tiobe-december-2017/.
[30]. Top Programming Languages 2017: Focus on Jobs. https://spectrum.ieee.org/computing/software/the-2017-top-programming-languages.
[31]. Robert W. Sebesta. Concepts of Programming Languages. 11-th edition. –London: Global Edition, 2016. – 788 p.