Knowledge-Based System for Web Interface Design

Maxim A. Bakaev and Tatiana V. Avdeenko

***Abstract*—The paper suggests the development of a knowledge-based system (KBS) to support web interface design via organizing somehow dispersed HCI-related knowledge and providing it on a context basis. The frame-based ontology was created with Protégé-Frames editor and CLIPS inference engine to serve as the core for the KBS, producing web design guidelines per business goals of a web-development (e-commerce) project and its target user group. The more theoretical knowledge, such as HCI laws and principles, was also embraced, and tag-based classification was used in the ontology to relate the guidelines to the design context, thus enhancing the explanatory component of the KBS. The results of the experimental testing suggest that the website developed with KBS's support compares favorably with existing analogues from e-commerce industry, attaining task success rate of 85.9% and receiving mildly positive user subjective impression ratings. The proposed KBS may be used not only as an expert system or in training inexperienced IT specialists, but also by skilled web and interface designers to improve the quality of their work by more wholly considering the attributes and requirements of a target user group.**

***Index Terms*—Human-computer interaction, knowledge-based systems, ontology, web interface design.**

I. Introduction

It was estimated that about 50% of all programming code produced when information systems and software applications are built is devoted to user interface [1]. Though usability and user interaction quality are recognized as major factors for software to succeed, web usability still remains at a relatively low level, despite some improvement noted lately. Such simple metric as "success rate" for e-commerce websites in the USA reached 81% in 2009 [2], but for user groups with special needs the web interaction quality leaves much to be desired: e.g. it was reported [3] that success rate for elder users with websites was almost 1.5 times lower than for younger group.

This situation is caused, to a significant degree, by relatively poorly organized knowledge in the fields of human-computer interaction (HCI), interface design and usability engineering: models and practical guidelines often lack strictly objective and quantitative nature or may even contradict each other. Besides, getting hold of findings and recommendations relevant to a particular interface design project context may be a separate work-intensive task, as existing software development support tools do offer capabilities to build interaction models (e.g. with UML) or interface prototypes, but they neither provide business goal- or target user- dependent guidelines, nor are able to evaluate the result of the design (more detailed review is available in [4], pp. 241-243). Enhanced HCI knowledge organization to support interface and web design activities, complementing recognized and effective usability engineering methods (such as usability testing, etc.), could alleviate requirements towards software developers' skills, decrease work effort needed for design as well as the probability of costly design errors.

In the field of artificial intelligence (AI), there are technologies developed for knowledge representation and their use has been reported in such diverse areas as biology and medicine, linguistics and education, business and management, and even e-business modeling [5]. We believe that HCI and web interface design domains are suitable for AI methods application as well, as this field matches the premises for knowledge-based systems (KBS) development and application, some of which are [6], pp. 36:

- Lack of specialists in the field, or a necessity to use several specialists to solve a relatively common task, as neither of them possesses required knowledge alone.
- Tasks that require exhaustive analysis of a complex context, taking time and leading to decreased productivity.
- High disparity between best and worst specialists' work results. High competitive advantage for companies that solve the tasks better.

The effective organization of practical guidelines and design patterns (which are seemingly equivalent for the purposes of knowledge engineering) is recognized as an important item in the research agenda for HCI. Further, it is argued that the relevant approach should focus not on design automation, but on the "intelligent management, organisation and retrieval" of knowledge [7], pp. 87. It was suggested that such organization should include [8]:

- *Taxonomy*, so that users (designers) could find materials they need from a large corpus.
- *Relationships* between guidelines and supporting knowledge, allowing in particular to easily move between levels of the problem.
- *Evaluation* of guidelines applicability through assessing the quality of results (design solutions).
- *Generation* of new knowledge (probably as well as updating existing one).

However, it is currently noted that there is not enough evidence that existing guidelines or design pattern organization systems have significant positive effect in practical interaction design [7, p.86]. Considering all of the above, we would argue that a knowledge-based system in

Manuscript received September 23, 2012; revised November 25, 2012. This work was supported in part by the Ministry of Education and Science of Russian Federation under Grant TP- 8.536.2011.

The authors are with the Economical Informatics Department, Novosibirsk State Technical University, 20 K.Marx prosp., Novosibirsk, Russia (e-mail: maxis81@gmail.com, tavdeenko@mail.ru).

HCI, to attain greater utility and recognition, should also conform to the following principles:

- Focus not just on organization of guidelines or design patterns, but support all the relevant stages of software development process and HCI integration in the process.
- Consider target users and their attributes in defining the interaction context.
- Consider interface elements as well as its overall quality and support the usability level evaluation.
- Provide feedback on the applicability and relevance of employed knowledge (through tracking the solutions' usability and success).

II. Method

A. The Ontology Design and Implementation

One of the up-to-date tools for knowledge engineering are ontologies – formal representations of a set of concepts within a domain and the relationships between those concepts [9]. So, we have designed and built the frame-based ontology serving as the core of the KBS that supports interface and web design activities by incorporating various levels of HCI knowledge: laws, principles, and guidelines. The system provides guidelines for a designer on a project-context basis, which is defined by 1) business goals of the project and 2) attributes of target users or of a pre-defined user category.

Among the numerous available ontology editors, we chose Protégé-Frames developed by Stanford University (http://protege.stanford.edu), since it: 1) supports frame-based ontologies and provides high level of abstraction, 2) has rich integration and import/export capabilities, 3) is free and open-source, 4) is in active development, has detailed documentation and reasonably easy interface.

Initial brainstorming sessions with experts produced a list of 109 concepts related to the domain. Then, based on this list, the top-down design process was carried out for the frame-based structure of the ontology that after several iterations came to incorporating 93 classes and 101 slots. The ontology classes' hierarchy is presented on Fig. 1. *Target user*, one of the key classes in the ontology, was assigned the slots representing users' personal attributes and context of use factors as shown on Fig. 2.

The identified concepts related to HCI knowledge representation: laws, principles, and guidelines, were transformed into ontology classes and organized as shown on Fig. 3. *HCI knowledge representation class* and *HCI knowledge* are abstract classes used to better structure the other concepts. *Law*, *Principle*, and *Guideline* represent levels of HCI knowledge as mentioned earlier, while additional class *Finding* was deemed necessary to stand for facts or evidence obtained in HCI or related fields from research or practical experience. The *Source* class generally corresponds to a publication, such as a book or research paper, and *Reference* is a subsidiary class containing an instance of *Source* and a page number.

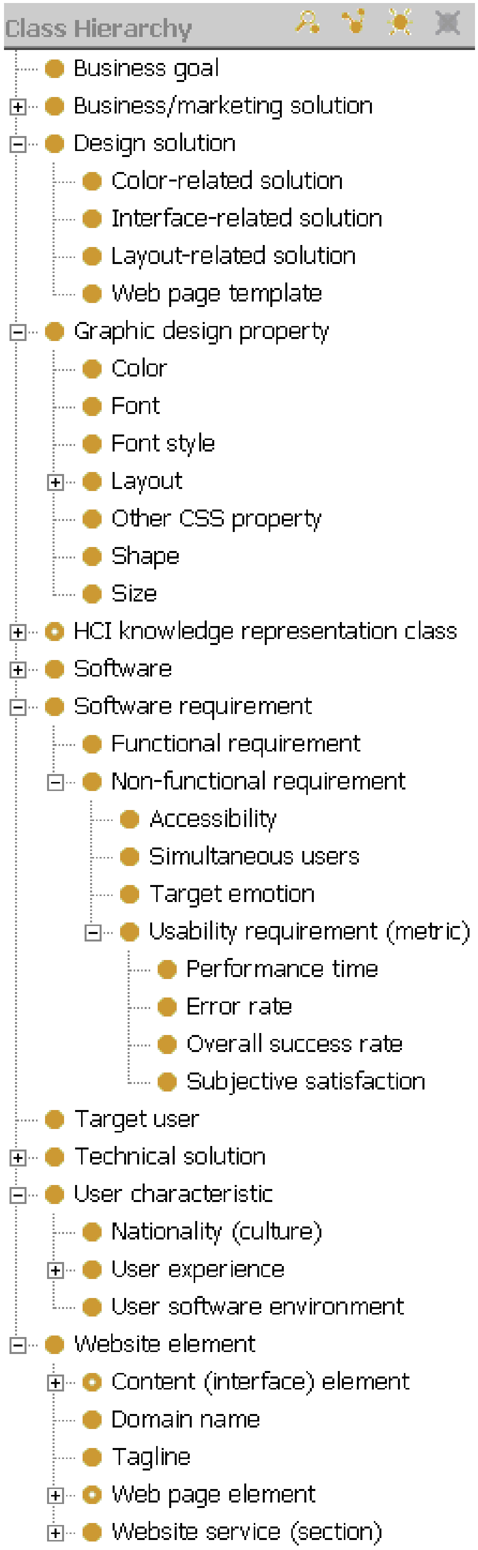

Fig. 1. The hierarchy of classes in the ontology.

The *HCI knowledge* class was given several slots (see Fig. 4) that would be inherited by its successors. We'd like to specially explain the two slots: *related* may be used to point to any other instances of an *HCI knowledge* sub-class that contain adjacent or interconnected knowledge, while *tag* slot is utilized to implement tag-based classification and its value (multiple) may be any classes existing in the ontology. The employed tag-based classification would allow HCI knowledge to be presented to potential ontology user on a context basis, in the process of interface or web page design.

In addition to inherited slots, *Principle* and *Guideline* classes were assigned optional *justification* slot to possibly reflect instances of higher level knowledge, i.e. of *Finding*, *Law* or *Principle* serving as a basis for the more concrete recommendation. Further, *Guideline* class (see Fig. 5) was given additional slot *applicability*, which is used for the purpose of evaluation of guidelines and stores the current estimation of the guideline's relative effect on the web interface design usability level.

Name	Cardinality	Type
age	single	Integer
context of use	multiple	String
education level	single	Symbol {none,high_school,college}
gender	single	Symbol {Male,Female}
internet access	single	Symbol {dial-up,cable,broadband,local_net
nationality (culture)	single	Instance of Nationality (culture)
software environment	multiple	Instance of User software environment
user experience	multiple	Instance of User experience

Fig. 2. Slots for the Target user class.

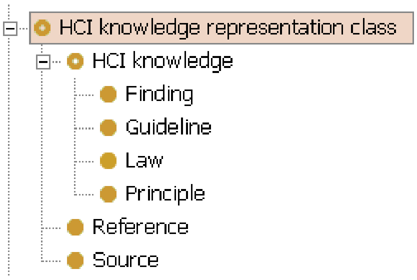

Fig. 3. HCI knowledge representation classes.

Name	Cardinality	Type
reference	multiple	Instance of Reference or Source
related	multiple	Instance of HCI knowledge
tag	multiple	Class with superclass :THING
text (english)	single	String
text (russian)	single	String

Fig. 4. Slots of the HCI knowledge class.

Name	Cardinality	Type
applicability	single	Float
justification (guideline)	multiple	Instance of Finding or Principle or Law
reference	multiple	Instance of Reference or Source
related	multiple	Instance of Guideline
tag	required multiple	Class with superclass :THING
text (english)	single	String
text (russian)	single	String

Fig. 5. Slots for the Guideline class.

B. Using the Knowledge-Based System

In accordance with the above-mentioned principles for developing the KBS in HCI, besides assuring knowledge organization, the system should support interface design-related activities as widely as possible in the software development process. Below we briefly outline how the KBS is used on the development stages most relevant for HCI (see in [4]). Since interface design is an intrinsically iterative process, the stages do not necessarily come in the described order, and the slots of the top-level class, HCI engineering task (see Fig. 6), may be filled in any order, which is one of the important advantages of frame-based model.

1) On the requirements analysis stage

Website target users are identified by designer (analyst) and are reflected in the system as instances of *Target user* class, the slots of which are linked to instances of *User attribute* sub-classes (the significance of user attributes consideration in website design is discussed, e.g., in [10]). Based on business goals, instances of *Requirement* class and its sub-classes are created with the support of the system, use cases are documented and interface quality target values are set.

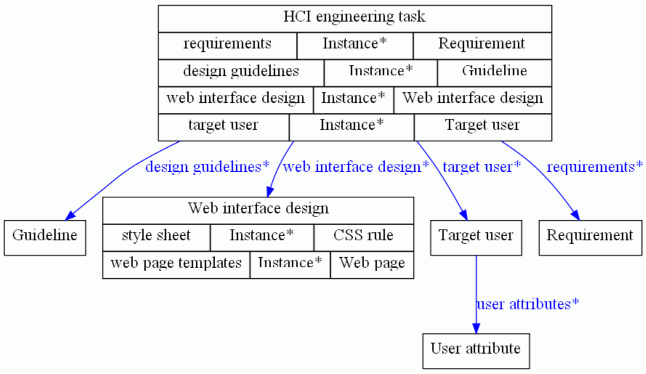

Fig. 2. HCI engineering task (top-level class in the ontology).

2) On the design stage

Based on the contextualized collection of *Guidelines* supplied by the system, designer makes decisions regarding the implemented web interface aspects (instances of *Design solution* class in the system). The provided guidelines are organized by means of *tag* slot as described above and are also related to higher level knowledge (*Findings, Laws, or Principles*) as well as supporting *Reference*.

The universe of *Design solutions* shapes the *Web interface design* prototype, which is reflected in the system as instances of *CSS rules* and *Web page* templates (template, in turn, is an ordered set of *HTML elements in web page*, each of which may have its own *Style properties*).

3) On the testing and evaluation stage

Generally, before *Web interface design* is finalized, its one or several prototypes undergo heuristic evaluation by experts or usability testing with real users. Thus, the system saves evaluations and performance metrics as *Interface quality metrics*, which can be compared across different prototypes, as well as related to the applied *Design solutions* and *Guidelines* to estimate their *applicability* and auto-update knowledge stored in the system.

III. Experiment

Ontology that contains instances and rules becomes knowledge base, although it's not always easy to draw this distinction. We used several sources (including [11] and [12]) to populate the ontology with web design guidelines as well as HCI laws and principles – by creating instances of *Guideline*, *Law* and *Principle* classes respectively. The inference rules were created with Prot ég é-supported CLIPS expert system language.

Then, the KBS was used in a real web development project – creating a website for the People's Faculty of Novosibirsk State Technical University (NSTU), a department that provides education for senior citizens, including computer literacy courses. The business goals were defined as "providing information about services online" and "providing a ground for the website visitors (students and alumni) to communicate". The target user was identified as "Elder user" (*age* = 60; *gender* = "*Female*"; *user experience* = IT_experience_low; education level = "*college*").

A. Experiment Description

To assess the effectiveness of the KBS application for web interface design, we ran an empirical investigation exploring whether a website developed with the system's support yields better usability for its target user group, senior users. The abovementioned website, designed and implemented for the People's Faculty of NSTU, was used in the experimentation, together with 5 other websites.

1) Websites and tasks

In total, 6 websites were used in the experiment. Websites #1, #2, #3, and #4 were third-party sites selected on the basis of being representative of small or medium e-commerce sites, or possibly having seniors as one of target user groups. Website #5 was also a real company's website, but created about 6 months before the experiment by a development team affiliated with the authors of the current paper. The team included experienced web designer and usability specialist. Finally, website #6 in the experiment was the People's Faculty website, designed based on guidelines provided by the KBS. More detailed information about the experimental websites is provided in Table I.

TABLE I: WEBSITES USED IN THE EXPERIMENT.

Website ID	Website URL	Description
#1	http://pensionerki.ru	A web forum for pensioners.
#2	http://npfraiffeisen.ru	A non-state pension fund.
#3	http://euro-kurses.ru	A business education center.
#4	http://moscow.apteka.ru	An online medical shop.
#5	http://vgroup.ru	A web development company.
#6	http://nf.assoc.nstu.ru	The People's Faculty of Novosibirsk State Technical University.

For each of the websites, one to four tasks were developed, up to total amount of 12 experimental tasks. The tasks were of two types typical for e-commerce: 6 "search" tasks, in which subjects had to look for certain information, and 6 "input" tasks that involved filling in web forms. As the experiment participants were inexperienced senior internet users, the developed tasks were relatively simple, so that the total time of the experimental session would be no more than 90 minutes.

2) Subjects

Among the recent graduates of computer literacy courses for seniors provided by the People's Faculty, 11 subjects (2 male, 9 female) were recruited for the experiment. The sampling was not random, as higher priority was given to graduates with more intense online experience, which was deemed necessary to better simulate the current and future senior population online, or even the alumni themselves in a few months from the graduation date.

The subjects' age ranged from 58 to 71 years, with the mean of 62.5 and standard deviation of 4.1. The mean self-reported time spent online by the participants was about 9 hours a week. All the subjects took part in the experiment voluntary and provided informed consent after reading through the tasks and learning the instructions.

3) Procedure

The experimental design and settings were quite typical for a user testing session, but relatively high number of websites was involved and the subjects' online experience often differed considerably. Thus, some of the participants did not attempt to perform all the tasks, presented to them in random order, during the experimental session time.

For each of the attempted tasks, success rate was measured by the instructor, and 0 was assigned for completely failed tasks, 0.3 – for tasks involving major errors possibly requiring support from the instructor, 0.7 – for tasks involving minor errors possibly requiring encouragement from the instructor, and 1 – for successfully completed tasks (similar approach to success rate measurement was proposed in [13]). After completing all the tasks with all the websites, the subjects were also asked to evaluate their overall impression of the websites by ranking them on a scale from 1 (worst) to 5 (best). Thus, independent variables in the experiment were website ID and group (third-party website, expert designer's website or KBS-developed website), as well as task ID. Dependent variables were task success rates and user subjective evaluations of the websites.

All the subjects used the same software environment, 1024×768 screen resolution, and accessed the experimental websites with Mozilla Firefox 3.6.3 browser.

B. Results

1) Success rate

In total, 106 tasks were performed by the participants and the overall mean success rate was 63.4% (see Table II). The mean success rate for search-related tasks was 67.2%, while for input-related tasks – 59.4%. The mean success rate for the control group of websites (#1, #2, #3 and #4) was 40.8%, while for the website developed with the KBS (#6) it ran up to 85.9%. For the website developed by the team including expert designer and usability specialist (#5), the success rate was even higher, 86.4%.

TABLE II: WEBSITES USED IN THE EXPERIMENT

Website ID	Task ID	Task type	Task attempts	Success rate
#1	1	input	10	28.0%
	2	input	9	17.8%
#2	3	search	10	20.0%
#3	4	search	9	82.2%
#4	5	search	8	27.5%
	6	input	7	80.0%
#5	7	search	7	77.1%
	8	input	7	95.7%
#6	9	search	10	93.0%
	10	search	10	100.0%
	11	input	10	78.0%
	12	input	9	71.1%
			106	**63.4%**

2) Subjective evaluations

Some of the senior participants refused to rank their overall impression of the experimental websites, blaming the lack of experience in judging websites, or assigned all-positive grades, which were not included in the analysis: all in all, the subjective evaluations were gathered from 7 subjects. Table III shows mean values and standard deviations for the evaluations included in the analysis.

TABLE III. SUBJECTS' SUBJECTIVE EVALUATIONS OF WEBSITES.

Website ID	Mean evaluation (SD)	
#1	3.86 (0.69)	3.88 (0.91)
#2	3.29 (1.25)	
#3	4.29 (0.76)	
#4	4.20 (0.45)	
#5	4.33 (0.52)	
#6	4.50 (0.84)	

ANOVA was used to test whether subjective evaluations were significantly different for different websites. The difference was found not to be significant ($F_{5,32}$=1.96; p=.111), however post-hoc analysis showed that evaluations for the website #2 significantly (p=.03) differed from ones obtained for websites #3, #5, and #6.

IV. CONCLUSIONS

Despite the wide recognition of the importance of web usability and the availability of cost-effective usability engineering methods, HCI integration in software development process remains far from universal. At the same time, there's lack of substantial evidence for practical utility of existing web design support systems that help to organize knowledge in HCI domain. With consideration of possible shortcomings of existing solutions [7], we proposed the development of the knowledge-based system (KBS) that covers all relevant stages of web development process and considers the context of the interaction (primarily defined by target users' attributes) as well as includes the capability of evaluating the applicability of employed knowledge.

The KBS was built on a specially developed frame-based ontology which incorporated about 200 classes and slots and was implemented in Protégé-Frames ontology editor. The knowledge base was populated with guidelines from various sources and used in a real project – creating a website of the People's Faculty of NSTU. To access the KBS applicability, the website was then experimentally tested with the target user group, elder people, together with another 5 sites.

The overall success rate in the testing, 63.4%, is in line with the data obtained for senior users in other experiments. For example, Jacob Nielsen reported a corresponding success rate of 52.9% in 2002 [3], but web usability has obviously manifested certain improvement since then. Input-related tasks were predictably harder for seniors than search-related ones, producing success rates of 59.4% and 67.2% respectively.

The developed website attained the success rate of 85.9%, more than doubling the one for the control group's websites (40.8%) and almost catching up with the one for a website developed by human experts (86.4%). Although the users' subjective impression of the developed website was not significantly better than of the others, the results suggest reasonable feasibility of the developed knowledge-based system to support web design activities.

REFERENCES

[1] S. Douglas, M. Tremaine, L. Leventhal, C. Wills, and B. Manaris, "Incorporating human-computer interaction into the undergraduate computer science curriculum," in *Proc. 33th SIGCSE Technical Symposium on Computer Science Education*, pp. 211-212, 2002.

[2] J. Nielsen. (April 16, 2009). IA task failures remain costly. Alertbox, [Online]. Available: http://useit.com/alertbox/ia-failures.html.

[3] J. Nielsen. (April 28, 2002).Usability for senior citizens. Alertbox. [Online]. Available: http://useit.com/alertbox/seniors.html.

[4] M. Bakaev and T. Avdeenko, "Ontology to support web design activities in e-commerce software development process," in *Proc. IASTED Automation, Control, and Information Technology (ACIT'2010)*, pp. 241-248, June 2010.

[5] A. Osterwalder and Y. Pigneur, "An e-business model ontology for modeling e-business," in *15th Bled Electronic Commerce Conference e-Reality: Constructing the e-Economy*, Bled, Slovenia, 2002.

[6] T. A. Gavrilova and V. F. Khoroshevsky, *Knowledge Bases of Intellectual Systems // Bazi Znanii Intellektualnih Sistem.* Piter, St. Petersburg, Russia, 2000.

[7] A. Dearden and J. Finlay. "Pattern languages in HCI: A critical review," *Human-Computer Interaction*, vol. 21, no. 1, pp. 49-102, 2006.

[8] S. Fincher and P. Windsor, "Why patterns are not enough: Some suggestions concerning an organising principle for patterns of UI design," *CHI'2000 Workshop on Pattern Languages for Interaction Design: Building Momentum*, 2000.

[9] T. R. Gruber. "A translation approach to portable ontologies," in *Knowledge Acquisition*, vol. 5, no. 2, pp. 199-220, 1993.

[10] M. Bakaev and T. Avdeenko, "Rationalizing HCI integration in e-commerce software development," in *Proc. DST-RFBR Sponsored Indo-Russian Joint Workshop on Computational Intelligence and Modern Heuristics in Automation and Robotics*. Sept. 2010, pp. 144-152.

[11] K. P. Coyne and J. Nielsen, *Web Usability for Senior Citizens*. The Nielsen Norman Group, Fremont, CA, USA, 2002.

[12] M. Bakaev, "Design strategies improving e-business accessibility for elder users," *Master Thesis*, Graduate School of Digital Design, Kyungsung University, Busan, South Korea. Feb 2006.

[13] J. Nielsen. (February 18, 2001). Success Rate: The Simplest Usability Metric. Alertbox. [Online]. Available: http://www.useit.com/alertbox/20010218.html.

Maxim A. Bakaev was born in Novosibirsk, Russia. In 2003 he graduated from Novosibirsk State Technical University (NSTU), Department of Economical Informatics, Master of information Systems in Economics. He is also Master of Digital Design, Graduate School of Digital Design Kyungsung University, 2007. He works now as an Assistant Professor of Economical Informatics Department in NSTU. He is the author of more than 15 scientific papers. His research interests include Human-Computer Interaction, Interface Design and Usability, Web Design, Knowledge Engineering, Universal Accessibility.

Tatiana V. Avdeenko was born in Novosibirsk, Russia. In 1985 she graduated from Novosibirsk Electrical Engineering Institute (now Novosibirsk State Technical University - NSTU), Department of Applied Mathematics, Diploma with honor. In 1991 she received a degree of Candidate of Science in Engineering, in 2004 – a degree of Doctor of Science in Engineering, in 1996 - Certificate of Associate Professor of Program Systems and Databases, in 2007 - Certificate of Full Professor of Program Systems and Databases in NSTU. She works now as a Head of Economical Informatics Department in NSTU. She is the author of more than 90 scientific papers including 1 monograph and 5 textbooks. Her research interests include Mathematical Modelling, Knowledge Engineering, Data and Knowledge Bases, Intellectual Support of Program Systems Development. Prof. Avdeenko is a Member of a Council on Defending Doctoral and Candidate of Science Theses. She has The Honour Letter from Ministry of Education and Science of Russian Federation.