Models and Software for Intelligent Web-Based Testing System in Mathematics

Andrey Chukhray and Elena Yashina

National Aerospace University «Kharkiv Aviation Institute», Chkalova 17, 61070 Kharkov, Ukraine

Abstract

In this paper computer models and software for intelligent web-based testing system in mathematics are described. The analysis of computer web-based testing systems in mathematics is carried out, Parameterized mathematical models for different tasks, including algebra and geometry, are synthesized, Architecture and software of intelligent web-based testing system are presented. Knowledges and skills evaluation model on the base of fuzzy logic is developed. Now system is working in two modes: a mode of training and a control mode.

Keywords

Intelligent tutoring systems, e-learning, educational software, mathematical models

1. Introduction

In the past few years e-learning experiences true boom. The driving force of it is continuous computer potential growth: increase of processor power and memory capacity, improvement of I/O facilities, perfection of network technologies, boost of efficiency of modern software development tools. The Covid-19 pandemic gave a powerful impetus to the computer-aided learning growth.

Artificial intelligence techniques allow to individualize and personalize the educational process [1, 2] and improve feedback from students [3]. At the same time scientific theory of artificial intelligence for e-learning is developing not so fast as developers would like. Artificial intelligence techniques are successfully used in teaching computer and mathematical sciences [4]. But today it is not possible yet to replace a good teacher with a computer tutor system. There is insuperable complexity of synthesis of models and methods for adaptation to mental work features, repertoire of knowledge and skills of each learner [5].

Therefore efforts of many developers aim at more utilitarian tasks where considerable progress is achieved at interactive intelligent tutoring system, such as evaluation of mathematical expressions [6], models constructing [7], solving algebra [8] and geometry [9] problems, online and web-based tutoring systems [10, 11], web-based test systems [12].

Within the framework of the Ukrainian project of external independent evaluation before authors there was a task of intelligent web-based testing system creation. The purpose of it was increase of effectiveness of school graduates preparation to external independent evaluation in mathematics by means of availability assurance of tests examples through the Internet, introduction of interactive modes of training and control for improvement contents of tasks and technologies of answers filling understanding, economy of expenses for manufacture and distribution of test tasks paper collections,

As analysis of world experience of interactive testing systems in mathematics making has shown the most creative way is development of parameterized mathematical models for automatic generation unique tasks, right answers and wrong but plausible answers for closed tests.

International Workshop of IT-professionals on Artificial Intelligence (ProfIT AI 2021), September 20-21, 2021, Kharkiv, Ukraine
EMAIL: achukhray@gmail.com (A. 1); o.yashina@khai.edu (A. 2);
ORCID: 0000-0002-8075-3664 (A. 1); 0000-0003-2459-1151 (A. 2);
© 2021 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

CEUR-WS.org/Vol-3003/paper1.pdf

2. Task definition

This article is dedicated to solving the problems.

1. To synthesize parameterized mathematical models for automatic generation unique tasks, right answers and wrong but plausible answers for closed tests.
2. On the base of fuzzy logic to develop knowledges and skills evaluation model.
3. To create algorithms and the software for intelligent web-based testing system in mathematics which must work in two modes: a mode of training and a control mode.

3. Task solution

3.1. Parameterized mathematical models

We described 26 different mathematical tasks, including algebra and geometry ones, were parametrical. There were 12 tasks of closed type and 14 tasks of open type.

Let's consider some examples of parameterization.

3.1.1. Example I

To solve equation

$$\sqrt{x+1}\sqrt{x-2}\sqrt{x-5}=0. \tag{1}$$

In general we have

$$\sqrt{x+a_1}\sqrt{x+a_2}\ldots\sqrt{x+a_n}=0. \tag{2}$$

The admitted region is

$$\begin{cases} x+a_1\geq 0\\ x+a_2\geq 0\\ \ldots\\ x+a_n\geq 0\end{cases} \tag{3}$$

Hence it follows

$$x\geq -min\{a_1,a_2,\ldots,a_n\} \tag{4}$$

and

$$x=-min\{a_1,a_2,\ldots,a_n\}. \tag{5}$$

The model of the task consists of:

1) n ∈ {1,2,. . . , 10}is randomly chosen;
1) set $\{a_1,a_2,\ldots,a_n\},-100\leq a_i\leq 100, i\neq j\rightarrow a_i\neq a_j$ is randomly chosen;
2) variants of answer are generated by the following way:
   a. equation has no roots;
   b. $x=-min\{a_1,a_2,\ldots,a_n\}$;
   c. $x=-max\{a_1,a_2,\ldots,a_n\}$;
   d. $x=\{-a_1,-a_2,\ldots,-a_n\}$
3) variants of answer are arranged randomly;
4) right answer is compared with answer of graduate.

3.1.2. Example II

The medial line of trapezium in **Ошибка! Источник ссылки не найден.** is equal 7. Height is $\frac{15\sqrt{3}}{7}$.

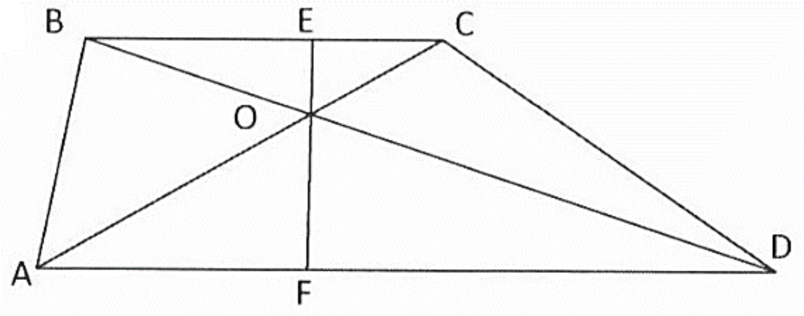


Figure 1: Graphical task statement

Angle between diagonal is equal 120°. To find product gets of diagonals of base is equal L and vertex angle is equal 120° BOC and AOD are similar triangles. Hence it follows

$$\frac{AD}{FO} = \frac{BC}{EO}. \tag{6}$$

Also we know that

$$S = \frac{AD + BC}{2} * (EO + FO) = \frac{1}{2} * AC * BD * sin(120°). \tag{7}$$

Therefore

$$AD, FO \leq \frac{AD}{2\sqrt{3}}, EO < FO \tag{8}$$

are generated. BC is calculated as

$$AD * \frac{EO}{FO}. \tag{9}$$

Then desired value is

$$\mathrm{AC} * \mathrm{BD} = \frac{\mathrm{AD} + \mathrm{BC}}{2} * \frac{1}{\sqrt{3}} * \mathrm{EF}. \tag{10}$$

3.1.3. Example III

To solve equation

$$2^{x+2} - 2^{x} = 96. \tag{11}$$

In general we have

$$(-1)^{b_1} a^{x+c_1} + (-1)^{b_2} a^{x+c_2} + \ldots + (-1)^{b_n} a^{x+c_n} = P. \tag{12}$$

The model of the task consists of following actions:

1) $n \in \{2,3,\ldots,10\}$ is randomly chosen;
2) a,$x \in \{2,3,\ldots,10\}$ are randomly chosen;
3) $c_i \in \{0,1,3,\ldots,10\}, i = \overline{1,n}$ are randomly chosen;
4) $b_i \in \{0,1\}, i = \overline{1,n}$ are randomly chosen;
5) P is calculated:
6) right answer is compared with answer of graduate.

3.1.4. Example IV

Areas of bounds of rectangular parallelepiped are equal S_1, S_2, S_3. To find volume of parallelepiped.
The model of the task includes:
1) generation random $x, y, z \in \{2,3,\ldots,10\}$;
2) calculation $S_1 = xy; S_2 = zy; S_3 = xz$;
3) check graduate answer against x ·y ·z.

3.2. Knowledges and skills evaluation model

To create evaluation model fuzzy approach was chosen because of its closeness to teacher qualitative reasoning.

Let us consider input linguistic variables:

U_I=”*Tasks solving speed*”,

described by the term set

$T(U_1)$={*Slow, Medium, Fast*}

as in [2];

U_2=”*The number of errors*” ,

$T(U_2)$={*Zero, Very small, Average, Large, Very large*};

and

U_3=”*Help reference*”,

$T(U_3)$={*Without help, With small help, With help, With great help*}.

For example, membership functions for $T(U_3)$ are presented in Figure 2.

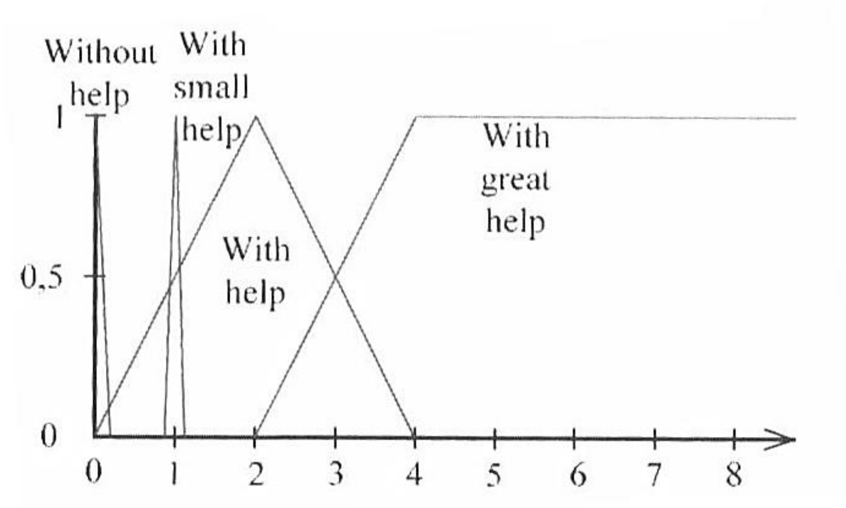


Figure 2: Membership functions for input variable Help reference

Let

$Y\ =\ "Knowledges\ and\ skills"$

be an output linguistic variable with term set

$T(Y)$={ *Excellent, Good, Satisfactory, Bad* }

T(Y)={Excellent, Good, Satisfactory, Bad}.

Then rules can be expressed in the Mamdani form:

If U_I is *Fast* and U_2 is *Zero* and U_3 is *Without help* then Y is *Excellent*.

If U_I is *Fast* and U_2 is *Very small* and U_3 is *With help* then Y is *Good*.

…

If U_I is *Slow* and U_2 is *Very large* and U_3 is *With great* help then Y is *Bad*.

4. Software

The main task of this work is intelligent computer system for testing knowledge and skills in mathematics software development. We have collected and analyzed the system requirements. Based on this the architecture of system and main components software was developed.

4.1. System architecture

As a result of the analysis of the system requirements, the main components and the connections between them were identified. The system architecture model is shown in Figure 3.

The system consists of the following levels:

- the database level, which contains the database for accounting information about users (registration, test results);
- the business logic level, which includes the APP Server;
- the communication level, that is, the level that carries out network interaction with the user;
- presentation layer, which includes tools for development and presentation of data, such as ASP .NET, WinForms, as well as directly displaying Web sites, that is, various browsers;
- user level.

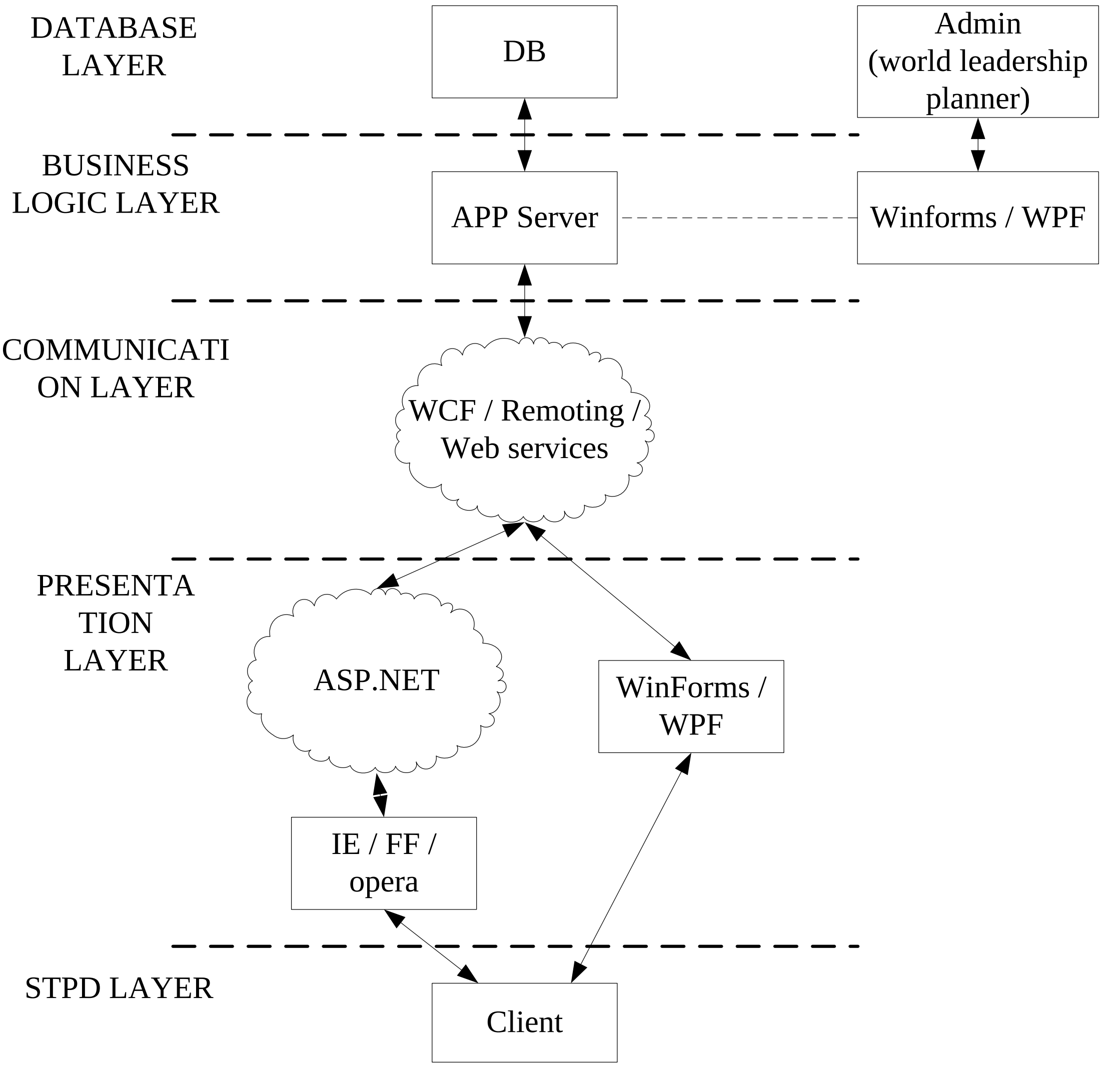


Figure 3: The architecture of an intelligent computer system for testing knowledge and skills in mathematics

Components for generating problem conditions and answer options for them are stored in separate library DLL files.

To provide versatility multilevel architecture of system was developed. It consists of four levels: database level, business logic level, communication level and presentation one. Main components of business logic layer are presented in Figure 4.

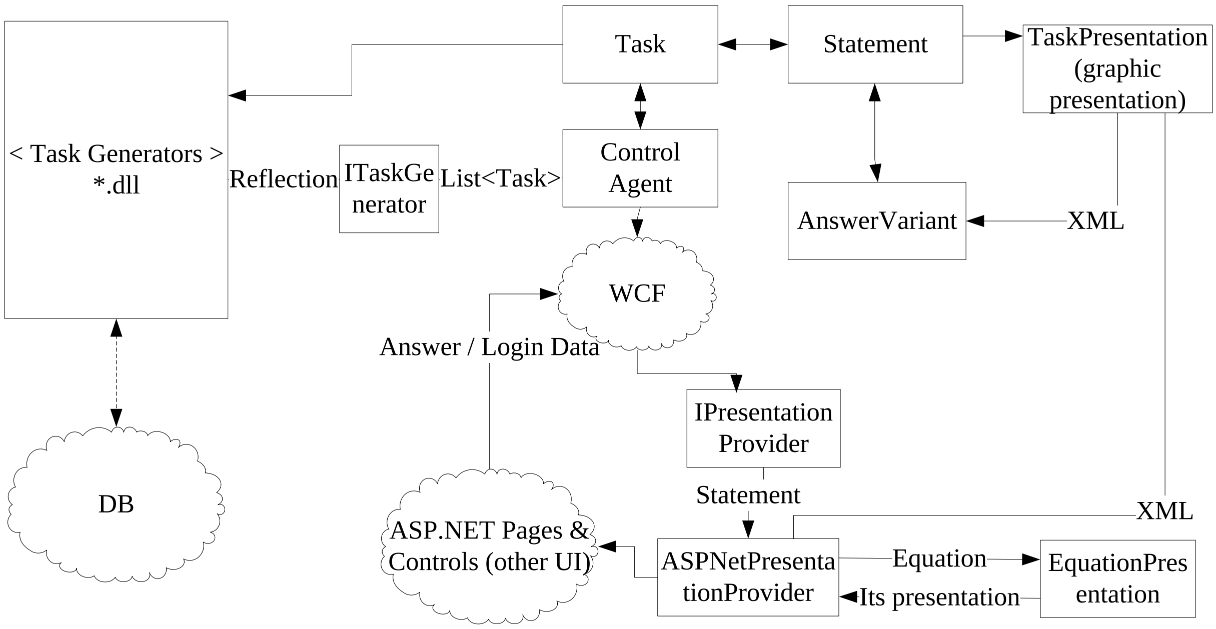


Figure 4: Components of business logic layer

After an entrance of the user in system, which happens at ASP layer WCF layer begins work. WCF represents an intermediate link witch connects parts of display and logic. WCF contacts library with tasks and through interface *ITaskGenerator* by means of reflection operation receives the list of tasks. Each element of the task list, i.e. a copy of class *Task*, contains a task condition, variants of answers and a right answer as XML-code. Control agent cuts right answer leaving it on a server and directs the remained elements (a condition and answer variants) to layer WCF. Thus, right answers in no way cannot become accessible to the user. Therefore security is provided.

The diagram of the main classes of an intelligent computer system for testing knowledge and skills in mathematics is shown in Figure 5.

The basic abstract class *Task* has two inheritors: the *EnclosedTask* class a of a closed type task; the *OpenTask* class of an open type task. Both classes override the base class's *Check()* method and have several constructors.

The *Task* class defines a *Statement* class that represents a task condition. It is associated with the *AnswerVariant* class, which generates answer options.

The Statement and *AnswerVariant* classes through the *TaskPresentation* class form the expression of the condition of the task and the options for responding to it in the form of an XML representation.

In addition to the *AnswerVariant* class, there is an abstract *RightAnswer* – correct answer class. Its inheritors are the classes of the correct answer to tasks of the private *EnclosedRightAnswer* and public *OpenedRightAnswer* types.

The *Task* class, through the *TaskPresentation* class, generates conditions and response options in the form of XML code containing tags of a specific purpose, for example, the *<expression />* tag contains an expression for building a mathematical formula, the *<plot />* tag is a directive for plotting a function graph. *ASPNetPresentationProvider* class implements transformation of task XML-code for its presentation on WEB-form. To implement tags such as a formula or graph, the *ASPNetPresentationProvider* class includes the appropriate libraries to represent them graphically.

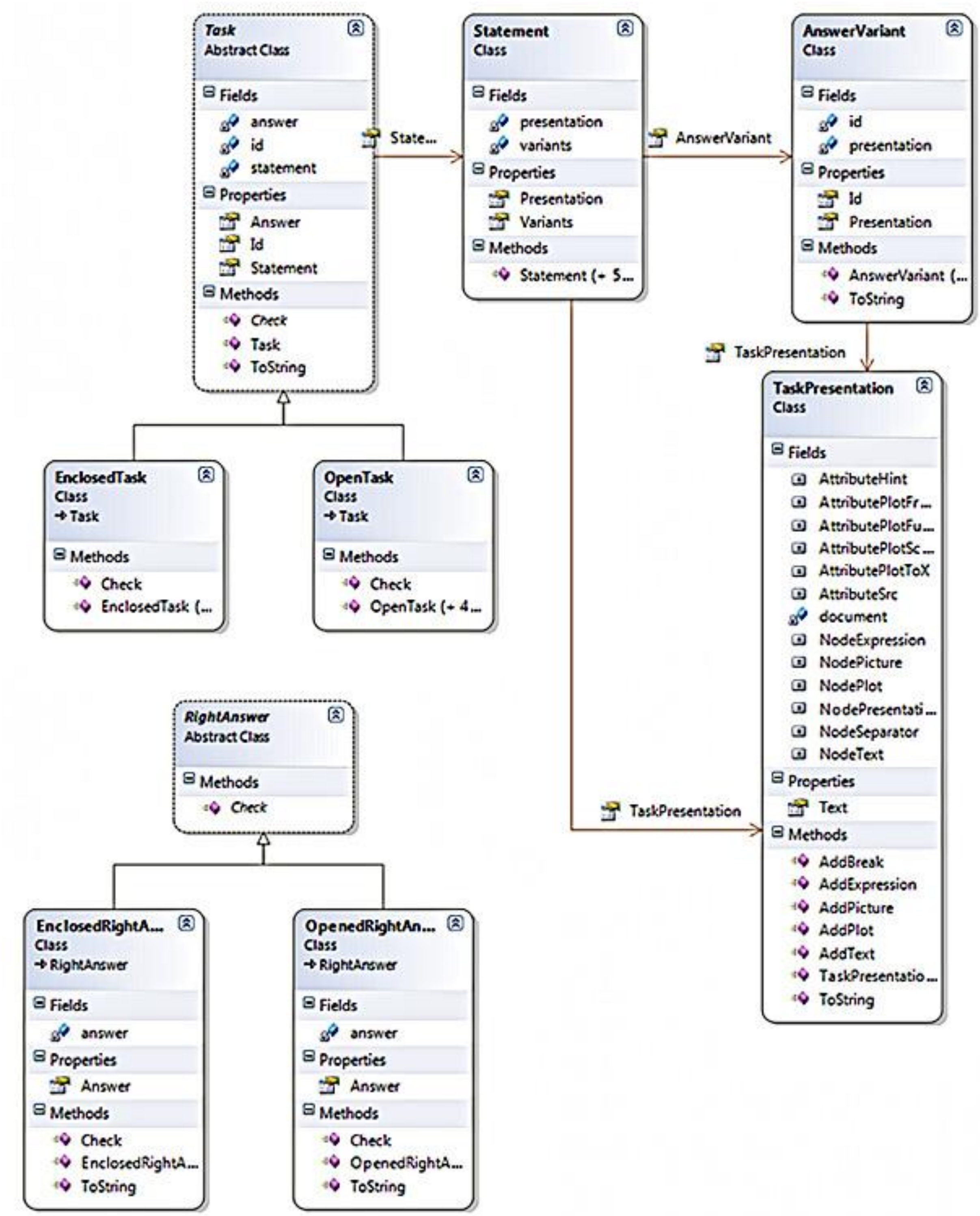


Figure 5: The diagram of the main classes of an intelligent computer system for testing knowledge and skills in mathematics

4.2. System Interface Development

The system must have an advanced user interface for visualizing formulas, graphs and other graphical objects. One of the system components is a module for constructing mathematical formulas designed for visual output of mathematical formulas of any complexity. The formula is specified as a sequence of elementary instructions that determine the location of the formula elements. After the structure of the formula is formed, the elements are arranged, the formula enters to the graphical module, which draws it on the Bimap object and displays it on the web form. The example of formula visualization is given in Figure 6.

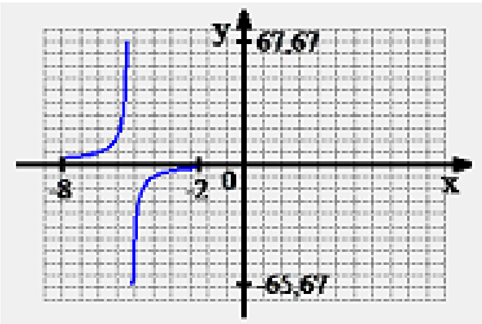
$$-\mathrm{tg}\left(2-\frac{\cos\left[\left(40+4e^{\sin(y)}\right)\cdot 42+i^{2}\right]}{-22}\right)$$

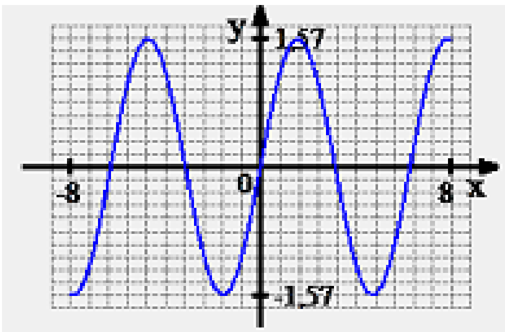
The formula is given as a sequence of instructions

+2;+40;+4e;powy;powsin();st[3]^st[5];st[2]+st[6];();st[8]*42;pow2;
i^st[10];st[9]+st[11];cos[];st[13]/-22;2-st[14];tan();-st[16];

Figure 6: The example of formula visualization

Also, a module for dynamic plotting of various mathematical functions has been implemented. Examples of dynamic output of function graph are shown in the Figure 7.





func[(x-3)/(x+5)],bound[-8;-2],dim[x;y],invP[-5]   func[(PI/2)*SINx],bound[-8;8],dim[x;y]

Figure 7: The example of function graph output

Screenshots of the system are presented in figures 8 – 10.

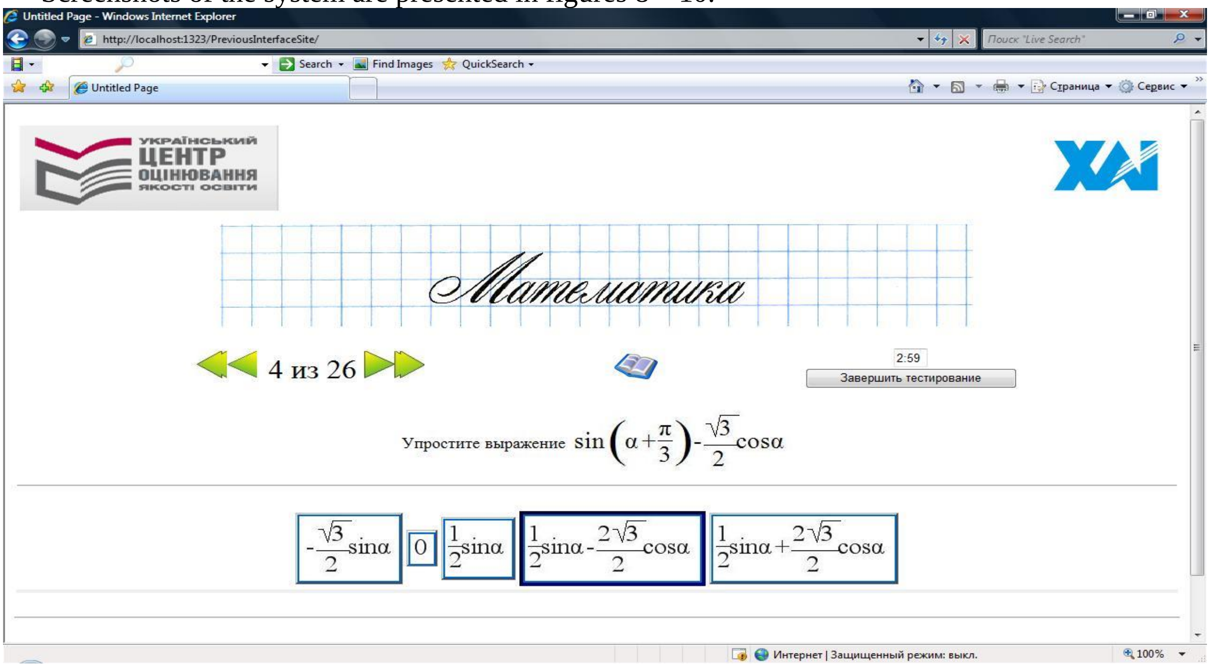


Figure 8: Screenshot of the system for trigonometric task

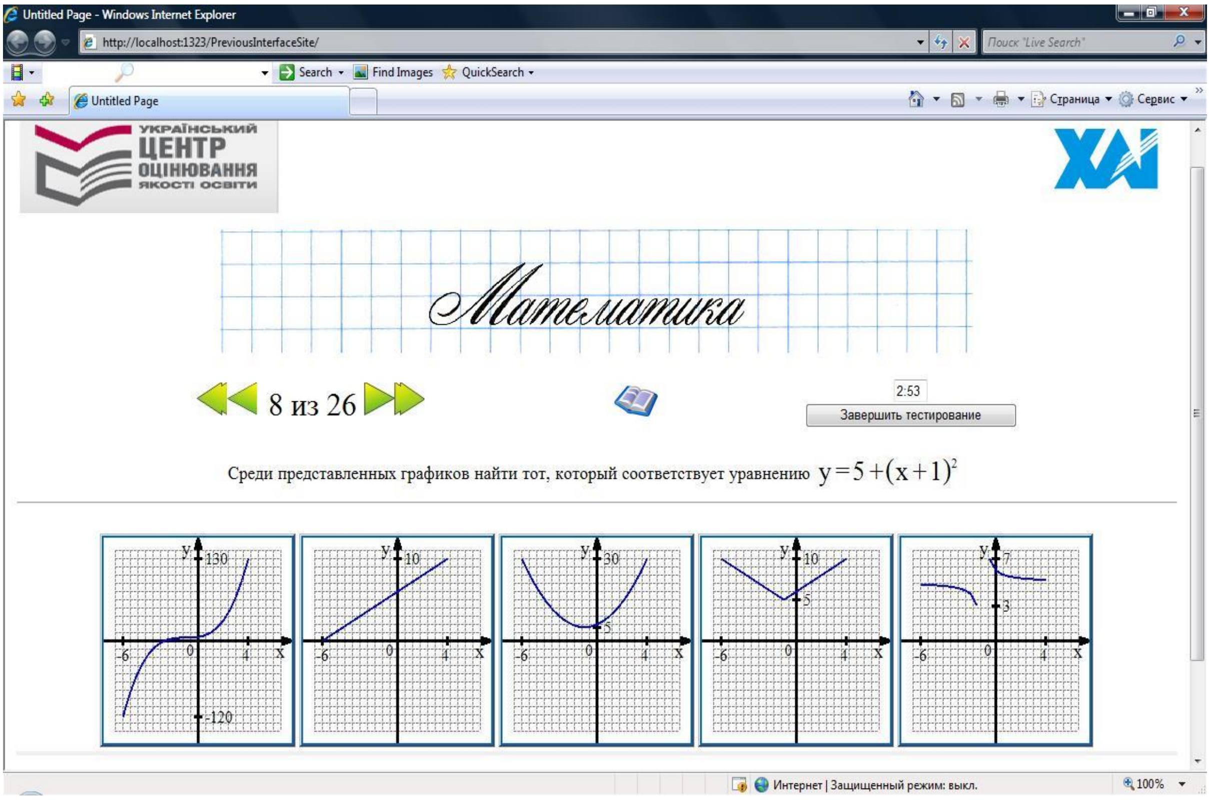


Figure 9: Screenshot of the system for algebraic task with graphical elements

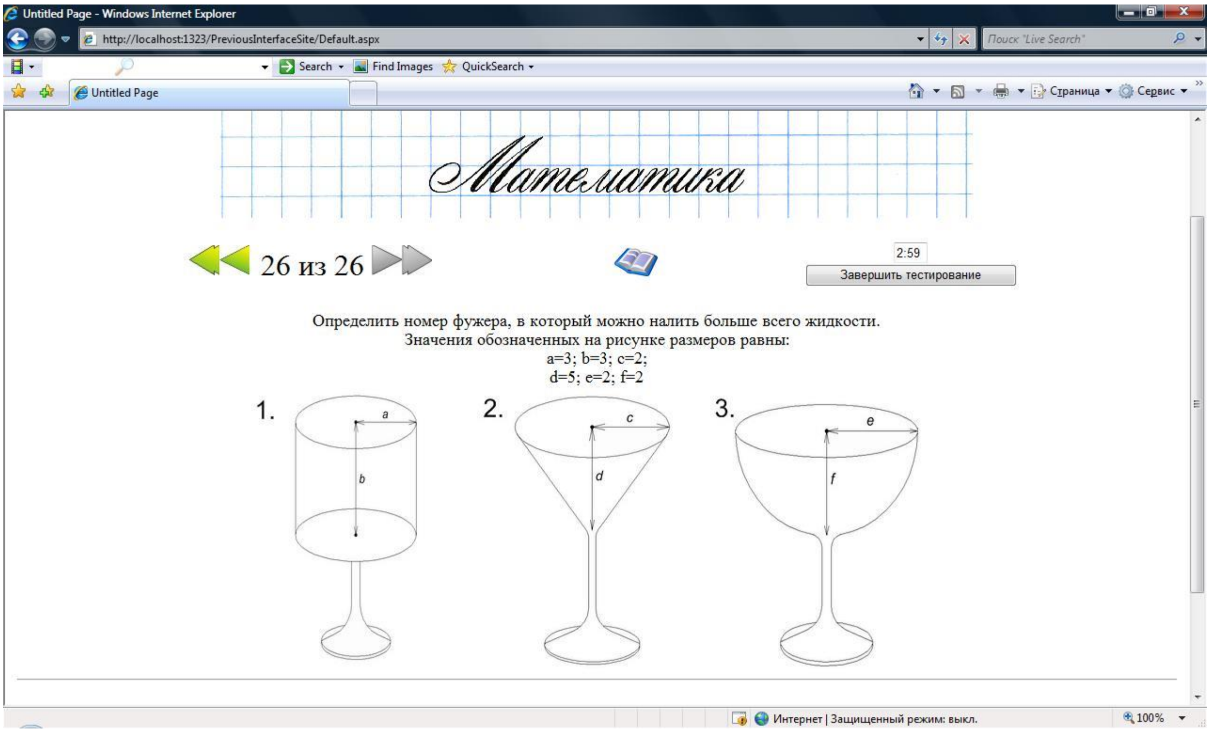


Figure 10: Screenshot of the system for stereometric task

Through interface IPresentationProvider the task in the form of a XML-code arrives to object ASPNetPresentationProvider where its analysis and split are executed. The task can consist of the text and pictures with formulas and graphs. In case of formulas object EquationPresentation provides creation of a demanded picture. Similar technology is used for creation plots of functions. Then controls arrive to ASP.NET Pages where they are displayed. Now system is being passed beta testing in site of Kharkov regional centre of education quality evaluation.

5. Summary

Parameterized mathematical models, knowledges and skills evaluation model and software for intelligent web-based testing system in mathematics are developed. The system has multilevel architecture and flexible interface to provide versatility and security. The next step may be creation of diagnostic models to adapt system to mental work features, knowledges and skills of each graduate [13].

This system joined in single complex meant for learners testing (http://zno-kharkiv.org.ua/cimt/). In the perspective, it is planned to integrate the math web-tests system with other software tutoring products [14].

6. References

[1] Nkambou, Roger, Riichiro Mizoguchi, and Jacqueline Bourdeau, eds. *Advances in intelligent tutoring systems*. Vol. 308. Springer Science & Business Media, 2010.

[2] Muangprathub, Jirapond, Veera Boonjing, and Kosin Chamnongthai. "Learning recommendation with formal concept analysis for intelligent tutoring system." *Heliyon* 6.10 (2020): e05227. doi: 10.1016/j.heliyon.2020.e05227.

[3] Putnam, Vanessa, and Cristina Conati. "Exploring the Need for Explainable Artificial Intelligence (XAI) in Intelligent Tutoring Systems (ITS)." *IUI Workshops*. Vol. 19. 2019.

[4] Abu-Naser, Samy S. "Predicting learners performance using artificial neural networks in linear programming intelligent tutoring system." (2012).

[5] Hasan, Muhammad Asif, et al. "The Transition From Intelligent to Affective Tutoring System: A Review and Open Issues." IEEE Access (2020). doi: 10.1109/ACCESS.2020.3036990.

[6] Pacheco-Venegas, Nancy D., Gilberto López, and María Andrade-Aréchiga. "Conceptualization, development and implementation of a web-based system for automatic evaluation of mathematical expressions." *Computers & Education* 88 (2015): 15-28. doi: 10.1016/j.compedu.2015.03.021.

[7] VanLehn, Kurt, et al. "Learning how to construct models of dynamic systems: An initial evaluation of the dragoon intelligent tutoring system." *IEEE Transactions on Learning Technologies* 10.2 (2016): 154-167. doi: 10.1109/TLT.2016.2514422.

[8] VanLehn, Kurt, et al. "Teaching Algebraic model construction: a tutoring system, lessons learned and an evaluation." *International Journal of Artificial Intelligence in Education* 30.3 (2020): 459-480. doi: 10.1007/s40593-020-00205-3.

[9] Wang, Ke, and Zhendong Su. "Automated geometry theorem proving for human-readable proofs." *Twenty-Fourth International Joint Conference on Artificial Intelligence*. 2015.

[10] Kefalis, Chrysovalantis, and Athanasios Drigas. "Web Based and Online Applications in STEM Education." *Int. J. Eng. Pedagog.* 9.4 (2019): 76-85. doi: 10.3991/ijep.v9i4.10691.

[11] Nye, Benjamin D., et al. "SKOPE-IT (Shareable Knowledge Objects as Portable Intelligent Tutors): overlaying natural language tutoring on an adaptive learning system for mathematics." *International journal of STEM education* 5.1 (2018): 1-20. doi: 10.1186/s40594-018-0109-4

[12] Hostovecky, M., M. Misut, and K. Pribilova. "Web Based Testing in Science Education." *Innovations and Advances in Computing, Informatics, Systems Sciences, Networking and Engineering* 313 (2014): 247. doi: 10.1007/978-3-319-06773-5_33.

[13] Chukhray, A., Havrylenko, O., Mygal, V. and Mygal, G. "Models and Methods for Computer Support of Adaptive Training of Algorithmic Tasks Solution." ICTERI. 2020: 408 – 415.

[14] Chukhray, Andrey, and Olena Havrylenko. "The engineering skills training process modeling using dynamic bayesian nets." *Radioelectronic and Computer Systems* 2 (2021): 87-96. doi: 10.32620/reks.2021.2.08.