Intellectual Support of Control System Human-Machine Interface Designers

Zhanna Kaminska [0000-0003-4029-1538]

Dept. of Software Tools, Zaporizhzhia National Technical University, Zaporizhzhia 69063, Ukraine
kamzhana@gmail.com

Abstract. The problem of intellectual support of human-machine interface (HMI) designers for Process Control Systems is considered. A method for solving this problem in the form of a set of models, technologies and knowledge, forming a new information technology for Control System HMI design is proposed. Software in the form of an Expert System that implements the proposed technology is developed. The use of an Expert System in the HMI design process will improve the efficiency, quality, reliability and safety of Control System operator's activity.

Keywords: expert system, SCADA system, man-machine interface, mnemonic scheme, ergonomic design, intellectual support

1 Introduction

The introduction of new information technologies in production processes, improvement of automation equipment leads to the fact that the human-operator becomes the "narrowest" link in modern Control Systems. It was shown [1] through operational experience of modern Control Systems that human mistakes or erroneous behavior cause 42% abnormal situations in industrial systems used. Accidents of 45% for nuclear plants, 60% - aviation crashes, 80% - sea disasters and 90% - car accidents [2, 3] happened due to human psychophysiological capabilities limitations and faults. In fact, Cochran and Bullemer [4] have estimated that abnormal situations cost the petrochemical industry $10B (US) annually in preventable losses. The Chemical Manufacturers Association estimates that 80 to 85% of human errors in chemical production result from poor design of the working environment [5]. European Agency for Health and Safety at Work, besides poor design of HMI, notes [6] also such reasons as multi-factorial risks (combined effects of poor ergonomic design, poor work organisation, mental and emotional demands) and complexity of new technologies, new work processes and HMI leading to increased mental and emotional strain. This demonstrates the relevance and importance of research in the domain of ergonomic design of Control Systems operator's HMI.

For today Automatic Control System development is mostly powered by Supervisory Control And Data Acquisition (SCADA) technology with wide variety of special component libraries as well as graphic editors used. However, they include no modules or add-ons to provide intellectual support for HMI ergonomic design. Hereinafter HMI determines Control System HMI developed by SCADA technology. The main purpose of our research is intellectual support of Control System HMI designers.

2 Literature review

The use of intelligent systems as a component of SCADA systems or in conjunction with them is known [e.g. 7–9]. Such integration provides intelligent support of operator in the analyzing data process, diagnosing the state of technological equipment, forecasting the development of situations and making appropriate decisions in normal and emergency situations.

Operator is often remote from objects of observation and control (OOC) and performs remote control using the information model (IM) in modern Control Systems. There are many types of IM. A special case of IM is the HMI of the operator developed by SCADA technology.

There are a lot of techniques for ergonomically driven the IM design process for Control System operators [e.g. 10]. Their disadvantage is the insufficient formalization of the design stages. The analysis [11], as well as further research, showed the following:

- software is being developed for displaying operational dispatch information on the Internet, as well as archived data [12]. A language for describing mnemonic diagrams was developed based on the XML standard. The mnemonic description file contains information about the appearance of the mnemonic diagram and how to react to various events or interactive actions of the user. The disadvantage of this approach is the rejection of the presentation of mnemonic diagrams in the SVG format. As a result, the possibility of drawing mnemonic diagrams through graphic editors developed by such companies as COREL DRAW, VISIO etc. is lost;
- basic requirements for the organization of HMI for process engineers of NPPs with VVER-1000 reactors and a set of HMI elements that is sufficient for organizing the control and diagnostics of a power unit is defined [13];
- software development of the system for improving the quality of interfaces of computer simulation systems is under development [14]. A new method has been developed to automate the HMI quality assessment process. The method is based on analyzing the factors that most affect the quality of HMI computer-based training systems on mnemonic diagrams;
- currently there are domestic standards that regulate the requirements for ergonomic design of such types of IM as a mnemonic diagram [15, 16]. The closest to the problem under consideration is the international standard ANSI/ISA-101.01-2015 [17]. This standard addresses HMI for equipment and automated processes. If the standard, recommended practices and methodology are followed, the result should enable the users to be more effective yielding improved safety, quality, productivity and reliability. The standards ISO-9241, ISO-10075, ISO-13407, ISO-14915 provides

recommendations for ergonomic human-system interaction design, operator's mental workload, and human-centered design of the HMI, respectively.

There are also many HMI design guides for designing of task-based interface (Ecological Interface) (e.g. [18, 19]), human-centred design (e.g. [20]) and HMI for Process Control Applications (e.g. [21]), but they give only general guidelines for interfaces designing.

On the basis of the foregoing, we can conclude that there are practically no work in the Control System HMI design automation field. It should also be noted that the design and configuration tasks of the HMI can significantly increase the cost and complexity of the Control System design project. According to [22], designers spend $10,000 or more to develop each page of HMI graphics for their process operations. Therefore, the actual task is to develop a method of intellectual support for the design of the Control System HMI.

3 Information technology of Control System HMI design

The base of the information technology for designing the Control System HMI were suggested by the author in [23] and require further development as part of this work.

The object-oriented approach, which simplifies the implementation of projects in SCADA-tools, is applied for modeling the subject domain "HMI design". The conceptual model of the subject domain "HMI design", made in the UML notation, is presented in Fig. 1.

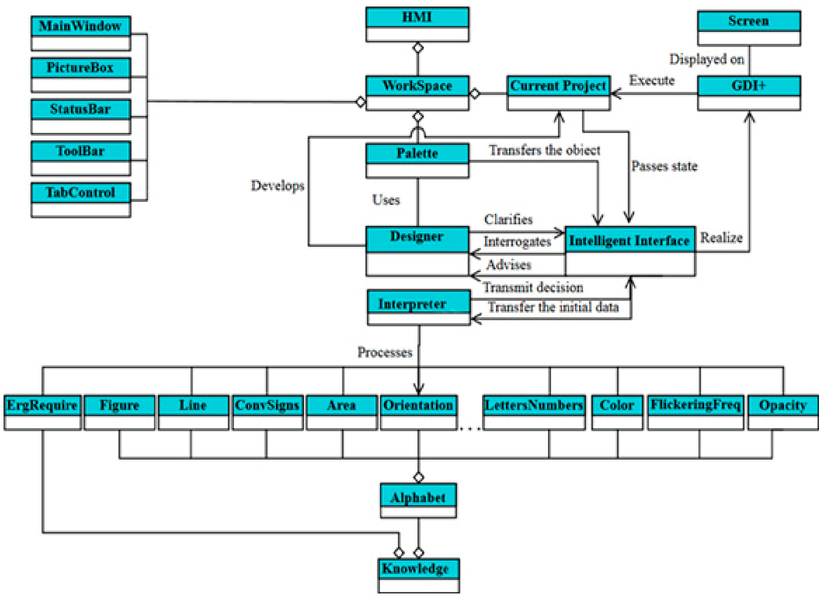

Fig. 1. The conceptual model of the subject domain "HMI design"

The internal hierarchical structure of the subject domain reveals the aggregate association. The nature of the other relations between classes in the model is set using the named directional associations.

According to Fig. 1, the designer (the class "Designer"), using a set of standard interface elements ("Palette"), is developing the project HMI ("Current Project") step by step. The current HMI project is carried out in the workspace ("WorkSpace"), which includes a set of tools, templates, and methods and means of using them within the project. The final state of the current project (the product of the designer's labor) is the Control System HMI ("HMI").

The designer fills the workspace with objects that are implemented using various program classes. The class "WorkSpace" is a container for the following classes: MainWindow (the main window of the HMI development tool); PictureBox (represents a Windows picture box control for displaying an image); StatusBar (represents a Windows status bar control); ToolBar (a toolbar that is a container for a group of commands or controls); TabControl (represents a control that contains multiple items that share the same space on the screen).

Intellectual support of the designer is carried out using certain Expert System (ES). A typical ES consists of the following main components: a inference engine (interpreter); working memory (data base) intended for storage of initial and intermediate data of a current problem; knowledge base; knowledge acquisition component; explanatory component; dialog component. To display them in the conceptual model (Fig. 1), the following classes are introduced:

– "Interpreter" for inference engine;

– "Knowledge" for knowledge and data base (KB);

– "Intelligent Interface" (subsystem "Intelligent Interface") for the remaining components.

As a result of the interaction of the designer with the subsystem "Intelligent Interface" is his intellectual support in the form of recommendations generated by ES.

ES provides intellectual support to the designer by generating advice during the design process. The process of advice inference is as follows. The input of the subsystem "Intelligent Interface" receives information: about the current state of the project (from the "Current Project"); about the typical interface element chosen by the designer (from the "Palette"), as well as the results of the dialogue with the designer (from the "Designer").

This subsystem transfers to the "Interpreter" the results of processing input information in the form of some initial judgments (prerequisites) that describe the current problem design task TS_i. The "Interpreter", using the initial data and knowledge from the KB of expert system ("Knowledge"), forms a sequence of rules (advice to the designer), which lead to the solution of the problem design task TS_i.

KB of *N-th* Man-Machine System (MMS) is represented as the tuple [24]:

$$KB(N) = \langle KBC(N), KBO(N) \rangle, \tag{1}$$

where $KBC(N)$ – KB about *N-th* MMS components; $KBO(N)$ – KB about *N-th* MMS organization, i.e. information about the relations of the MMS components.

The knowledge base of ES consists of a set of local KB. Knowledge about components $KBC(ES)$ and organizations $KBO(ES)$ contained in the KB of Expert System is discussed further. In Fig. 1 presents two local KB "ErgRecuire" (denote it $KBEV$) and "Alphabet" ($KBAL$) such as $KBEV, KBAL \in KBO(ES)$.

The first KB "ErgRecuire" contains the following knowledge of the ergonomic support for the HMI designer: psycho-physiological characteristics of the operator; rules for developing IM and HMI; code arrangement rules; rules and recommendations for the design of instrument scales and graphs.

The second KB, "Alphabet", represents knowledge about the coding methods (alphabets) of information about OOC of the technological process (TP). Qualitative and quantitative characteristics of OOC are coding in different ways (alphabets, codes). 17 coding methods are known [25].

Fig. 1 presents the following types of alphabet: "Figure" - coding with abstract geometric figures; "Line" - type of line; "ConvSigns" - conventional symbols; "Area" - area of geometric figures; "Orientation" - orientation of the figure or line in space; "LettersNumbers" - letters, punctuation marks and numbers; "Color" - color; "FlickeringFreq" - flicker frequency, "Opacity" - opacity.

Subsystem "Intelligent Interface" received the results of the logical inference from the "Interpreter" and in accordance with them modifies the current HMI project. Thus, we get a solution to the current problem task TS_i.

As it is known, GDI+ ("GDI+" in Fig. 1) is the Windows operating system graphic device interface, which enables functions to transfer graphic objects to a monitor. Access to GDI+ functions is implemented through a set of managed classes. The interface of the managed classes associated with images on a screen ("Screen" in Fig. 1) is part of the .NET Framework.

3.1 Analysis of Control System HMI structure

An analysis of HMI of modern Control Systems made it possible to reveal their typical structures that determine knowledge about the components of HMI (i.e. $KBC(ES)$). The following typical formats for presenting information in the Control System HMI are defined: mnemonic diagrams of three levels (generalized, group and individual); indications (sets of analog and discrete information); current events associated with the change of states of technological systems and processes; alarms; graphs of analog parameters (trends); instructions (for displaying the text of operational instructions); free formats that are used to obtain reference information in the operator's dialogue mode with an intelligent SCADA-system module.

Each information format is practically implemented as a specific HMI zone on the screen. Interface elements that can be located by the designer in the zones were combined into the conceptual class "Palette". The domain model "Palette" in the form of a class diagram of the UML language is shown in Fig. 2. The diagram displays two types of relations - aggregation and generalization, that is, the R_2 relation of the type "IS a". The relation R_2, as applied to the class diagram, describes the hierarchical structure of classes and the inheritance of their properties and behavior.

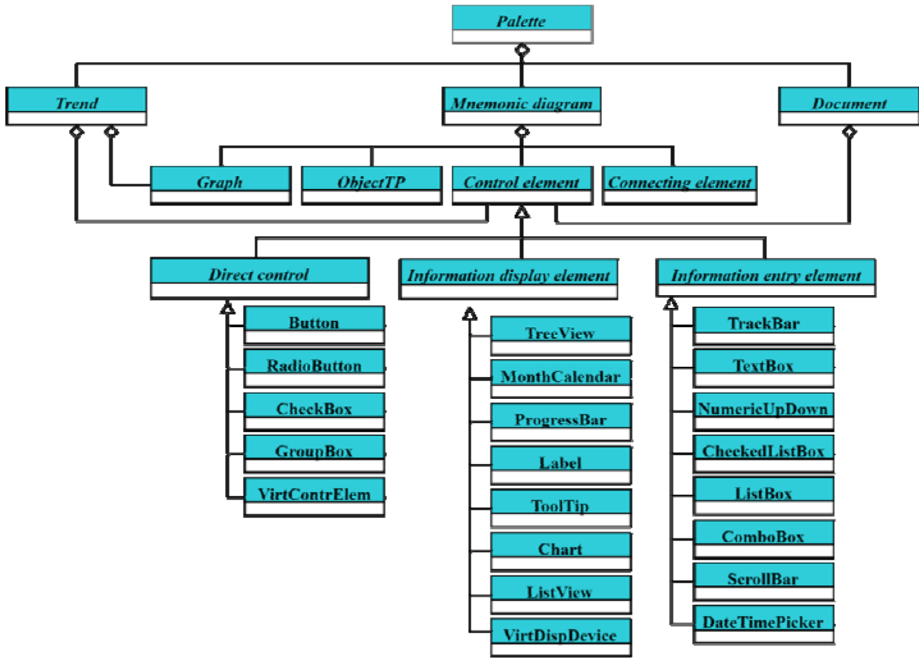

Fig. 1. Model of the subject domain "Palette"

Zones of all formats are presented in the form of three classes: "Mnemonic diagram", "Trend" and "Document" in Fig. 2. The purpose of the first two classes is obvious. The "Document" class was introduced to represent all the other formats listed above.

In accordance with the model, for example, the following elements can be placed in the zone of the mnemonic diagram: "Graph"; "ObjektTP" (OOC of technological process); “Control element” (HMI Controls) and “Connecting element” (connectors on a mnemonic diagram).

The “Control element” class is the parent for the following classes: “Direct control” (direct control elements); "Information display elements" and " Information entry element" (information input elements). Each of these classes, in turn, has child classes. Most of them are typical elements of the interface "Button", "TreeView", "ComboBox" and others. HMI may also contain virtual elements that simulate real controls and devices. To describe them, the model has introduced the classes “VirtContrElem” and “VirtDispDevice” respectively.

3.2 Knowledge about technology to transfer graphic objects to a monitor

The *KBO*(*N*) organizational knowledge of any MMS can be represented as a complex of morphological structures. The components of these structures are the many goals of the MMS, the conditions, tasks, functions, plans, means of technology and laws of the MMS control [23].

To achieve the “HMI design” goal, the designer must solve a lots of tasks $TS = \{TS_i \mid i = 1 \div K_i\}$, the nomenclature of which is due to known methodological techniques of IM and HMI design [10, 18-21, 25]. Knowledge about plans and technologies for their implementation (*KBTS*) is a subset of *KBEV* knowledge. They can be represented as a tuple [24]:

$$KBTS = \left\langle Fn_i, MTS\,(TS_i), MTc_i, P_{mm}\,(KM_p, MTc_i) \right\rangle, \tag{2}$$

where $Fn_i = \{Fn_{ij} \mid j = 1 \div K_j\}$– set of functions Fn_{ij}, that must be implemented to solve the *i*-th task; $MTS(TS_i)$ – morphological-time structure of functions Fn_{ij} that shows a planned logical-time sequence of functions necessary for solving the TS_i task; MTc_i – set of technological ways to implement the function Fn_{ij}; $Pmm(KM_p, MTc_i)$ – the predicate "The implementation of the function Fn_{ij} is possible in a set of technological methods MTc_i with a combination of conditions KM_p".

Windows Forms graphic elements can be divided into three main categories: two-dimensional (2-D) vector graphics; drawings; typographic markup.

The managed class interface associated with transfer graphic objects to the monitor in the .NET Framework 4.5 and 4.6 consists of the following namespaces: System.Drawing; System.Drawing.Drawing2D; System.Drawing.Imaging; System.Drawing.Text; System.Drawing.Configuration and System.Drawing.Design.

The parent System.Drawing namespace contains types that support the basic graphical functions of GDI+. The basis of the GDI+ interface is the Graphics class. It directly performs the drawing of straight and curved lines, geometric figures, the output of drawings and text.

The analysis of the functional purpose of the System.Drawing namespace classes, the associated methods of the Graphics class, as well as the types of the alphabet, and the possibility of their use for coding the Windows Forms graphic elements was carried out.

We introduce four sets:

– A_1 – information transferred to a screen;

– A_2 – System.Drawing namespace;

– A_3 – Graphics class methods;

– A_4 – types of the alphabet.

In order to determine the knowledge required by the "Intelligent Interface" subsystem to implement the transfer and coding functions of HMI elements, we introduce the tetradic relation R_3 "To transfer and code a_1, use a_2, a_3, a_4" on the $A_1 \times A_2 \times A_3 \times A_4$ Cartesian product.

The relation R_3 can be matched to a 4-place predicate. A tuple (a_1, a_2, a_3, a_4) belongs to a relation if and only if the predicate of this relation is $P(a_1, a_2, a_3, a_4)=1$.

All elements of the R_3 relation are of the same type of tuples. Tuples of the same type make it possible to consider them as analogs of rows (i) in a simple table, that is, in a table where all rows consist of the same number of cells and the same data types are located in the corresponding cells. An example of the R_3 relation representation in the form of a table is presented in the Table 1.

Table 1. An example of the R_3 relation representation

Displayed information	Classes Namespace System.Drawing	Class methods Graphics	Type of alphabet								
			Geometric figures	Symbols	Area of figures	Orientation in space	Hatching	Alphanumeric	Color, brightness	Flicker frequency	Opacity
2-D vector graphics	Brush	FillClosed Curve							1		
		FillEllipse					1		1		
		FillPolygon					1		1		
		FillRectangle					1		1		
	Pen	DrawArc	1			1			1		
		DrawClosed Curve	1	1	1	1			1		
		DrawEllipse	1		1	1			1		
		DrawLine				1			1		
		DrawPolygon	1		1	1			1		
Raster images	Image	DrawImage	1	1	1	1	1	1	1	1	1
	Icon	DrawIcon		1	1				1		
Characters and Strings	StringFormat	DrawString						1			

The number of columns j=4 in the Table 1 is determined by the number of sets of Cartesian products. Table 1 for the convenience of presenting information (reducing the number of rows) has the following features:

– the 4-th column is presented in the form of 18 elements of the set A_4 (in the example of Table 1, nine alphabets are represented);

– tuples with common elements from j=1÷3 columns and various elements of A_4 set are combined in one line;

– the following condition must be met for the x_{ij} elements of the table

$$x_{ij} = \begin{cases} 1, if R_3 = true, \\ 0, if R_3 = false \end{cases} \tag{3}$$

Such codes as letters, punctuation, and numbers are represented by one "Alphanumeric" alphabetic in Table 1. Due to the fact that the computer implementation of the "Color" and "Brightness" codes is associated with the same RGB color model, in Table 1 they are presented in one column.

After formalization and recording of the knowledge given in Table 1 in KB, the expert system will have knowledge about the transfer graphic objects to a screen technology and selection of codes for Windows Forms graphic elements. For example, when solving the problem of transfer raster images, the "Intelligent Interface" subsystem will propose to use the Image class AND/OR Icon class of the System.Drawing space and the corresponding methods of the Graphics class. If designer chooses the Image class, all kinds of codes will be recommended as possible. If he chooses Icon class, the following types of codes will be recommended: "Symbols" AND/OR "Area of figures" AND/OR "Color" AND/OR "Brightness".

The combination of the developed models, technologies and knowledge forms a new information technology of Control System HMI design.

4 Implementation

The developed information technology has become a theoretical basis to develop an ES, which could deliver a methodology of the design and set of functions as the intellectual tool to support technical designers whose task is to create Control System HMI. The general scheme of the ES demonstrated at Fig. 3.

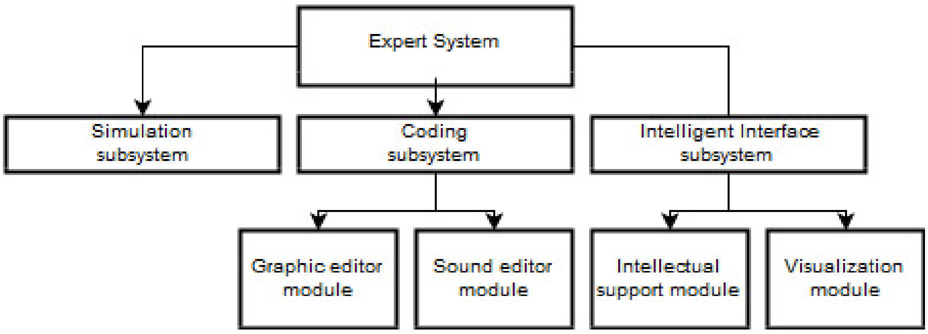

Fig. 3. The general scheme of the Expert System

The “Simulation” subsystem provides interactive extraction of expert knowledge about TP, OOC and their parameters, algorithms of the operator’s activity. The goal of TP may be to decompose into other sub-levels (tasks, procedures, functions and operations). Knowledge about the sub-levels could be formalized in the form semantic graph. The nodes of this graph are the names of the functional units F_i of different scale, and the arcs show the relation R_{FiFj} - "to implement the functional unit F_i (goal, task), it is necessary to perform the functional unit F_j (task, procedure, function, operation)". The root of the tree is the concept of "Technological process". The description of the TP, given by experts, is saved in the tptree.xml file.

The description of TP objects, given by experts, is saved in the objects.xml file. Here is a fragment of this file that describes the "Operative" state of the Compressor-01object of TP. This object has one quantitative parameter “Pressure” and one qualitative parameter “Noise”. The parameter "Pressure" should be in the range from 1,0 to 2,1 Pa in "Operative" state. The operator must observe (tag "observed") and control (tag "controled") the parameter "Pressure". The “Noise” parameter is only observed and can be “true” or “false”:

```
<objects>
<object name="Compressor-01" uuid="{550e8400-e29b-41d4-a716-
446655440000}">
    <state name="Operative" uuid="s001">
        <parameters>
            <qnParameter name="Pressure" uuid="p001-p001" union="Pa"
        controled="true" observed="true">
                <minValue>1.0</minValue>
                <maxValue>2.1</maxValue>
            </qnParameter>
            <qnParameter name="Noise" uuid="q001" union="" controled="false"
        observed="true">
                <values>
                    <value> true </value>
                    <value> false </value>
                </values>
            </qlParameter>
        </parameters>
    </state>
</object>
</objects>
```

The use of simulation at the stage prior to the development of the design specification for HMI allows: determine the probabilities of various MMS states; estimate the ability of the operator to process the incoming information necessary for decision making and draw up a list of those situations that he cannot solve due to his psycho-physiological limitations. The results obtained will allow the designer to reasonably accomplish the task allocation between the operator and the machine. In the next stages of HMI design, this subsystem, if there are several variants of “paper” HMI prototypes, ensures the selection of the optimal variant. The choice is based on such quantitative indicators of the operator's activity, such as the probability of correct execution and execution time of the control algorithms obtained as a result of a simulation experiment.

The extracted expert knowledge about TP, OOC and their parameters are input to the Control System HMI design. The HMI design process is performed with the participation of the “Coding” subsystem and “Intelligent Support” module of the “Intelligent Interface” subsystem. The scenario of the HMI designer interaction with "Coding" subsystem can be described by the use case diagram shown in Fig. 4. The "Intelligent Interface" subsystem is shown as an external entity.

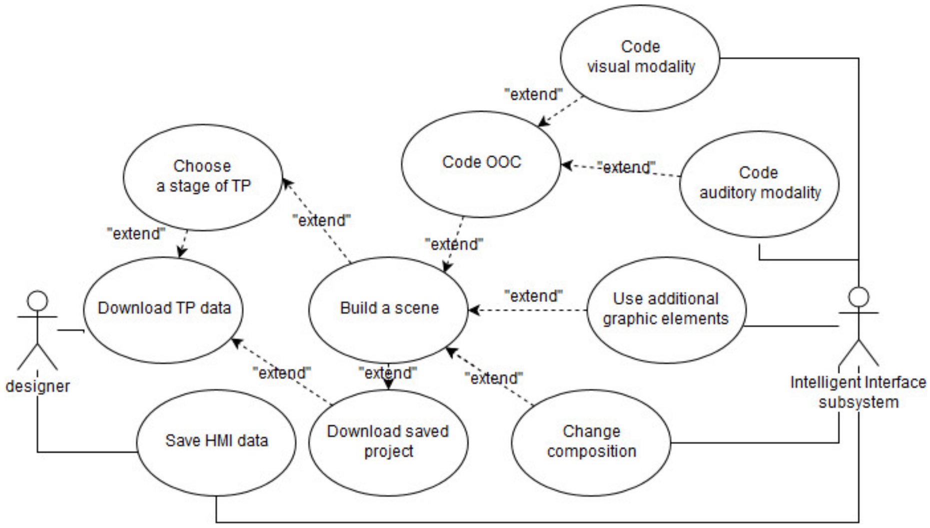

Fig. 4. The use case "HMI design" diagram

For each stage of TP a scene (video frame) is built, representing the TP mode. The video frame can be divided into zones with various formats of information presentation and filled with graphic elements. Thus, HMI consists of a set of scenes.

The process of building a scene consists in coding OOC, their states and parameters, as well as developing a scene composition with the allocation of operational space zones using various data presentation formats. The result of the expert's interaction with the intellectual support module is filling the KB with the following knowledge: requirements for the information about the OOC (see position 2 in Fig. 5); about the specifics of the operator's tasks and environment (see position 3 in Fig. 5). In accordance with the results of interaction, the HMI designer receives the ES recommendations on the choice of alphabet types for coding (see position 1 in Fig. 5). All subsequent steps of building the scene are also supported by recommendations for the designer. They are generated by the intellectual support module.

The graphic and sound editors modules are intended, respectively, for coding OOC and basic interface elements using alphabets coding visual modality (symbols, color, flicker frequency, brightness, etc.) and alphabets of auditory modality (verbal and sound signals) in accordance with ergonomic requirements.

Declarative knowledge is represented in the ES in the form of predicates (j); $P(x_1, x_2)$, where j is a name of fact (the ordinal number of the fact in the KB); P is a name of predicate; x_i, i=1, 2 is a subject constants.

Procedural knowledge is presented in the ES in the form of production rules (k); R_i: $A_i \rightarrow K_i$, where k is a name of the production (the ordinal number of the rule in the KB); R is a condition for the applicability of the production core; $A_i \rightarrow K_i$ is a production core.

After adding an element to the scene, HMI designer can apply coding to it in accordance with the received recommendations. This is done by changing the corresponding properties of the element. An example of the correspondence between the properties of the Button control and the categories of codes (alphabets) that can be applied by using these properties are listed in Table 2.

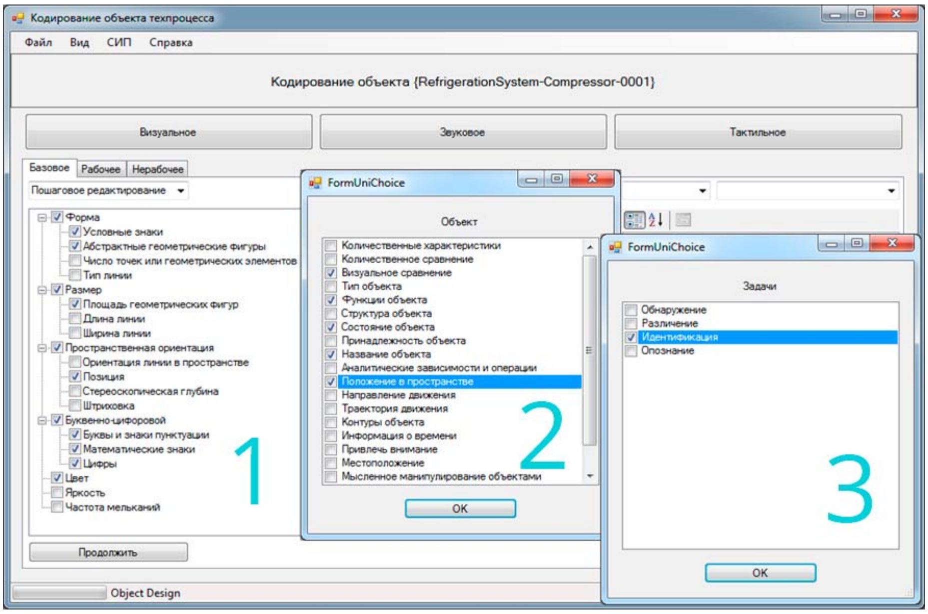

Fig. 5. An example of a expert survey and the generation of recommendations on the choice of the alphabet

Table 2. Table of correspondence of the Button control properties and the code category

Property	Code
FlatStyle	Stereo Depth
Font	Alphanumeric
FontHeight	Size
ForeColor	Color
Height	Size
Image	Geometric figures, Conventional symbols, Type of line, Number of points or geometric elements, Orientation in space, Alphanumeric
Size	Area of figure
Text	Alphanumeric
Width	Area of figure

These properties are changed and set using the element control properties panel.

The developed HMI project is saved in the .xml file format. Data from this file is input to the "Intelligent Interface" subsystem for the visualization of HMI. An example of the design results is presented in Fig. 6.

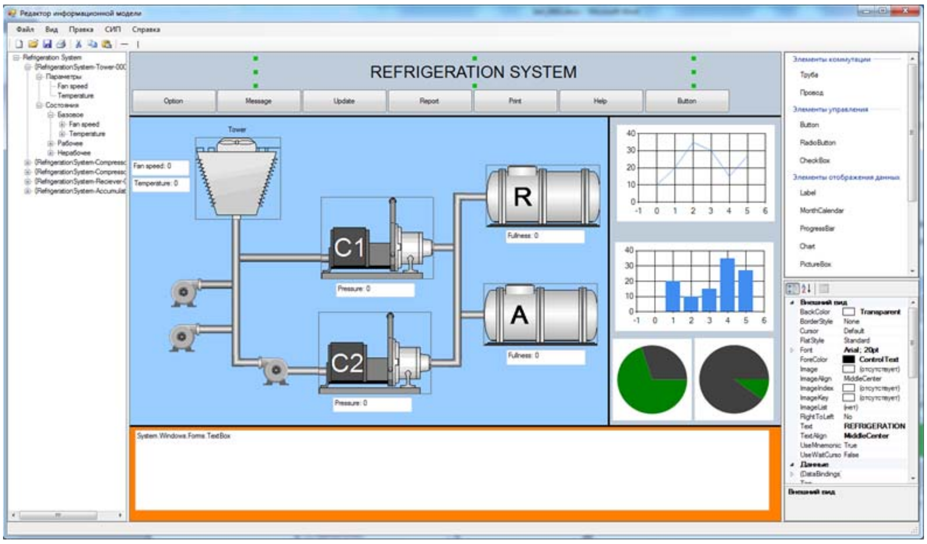

Fig. 6. An example of the HMI design results

The ES is implemented using Microsoft .Net Framework in C#. The Asp.Net, Ajax, CSS, HTML and SQL technologies were used to implement the visualization module. The WaveGenerator.dll library was used to work with the Wav sound format.

5 Conclusion

The urgent problem of information and mathematical support development is solved to automate the HMI designer activities.

The method of intellectual support for Control System HMI ergonomic design is proposed for the first time. Information technology that provides a methodology covering both design and pre-design stages of HMI design has been developed. At the pre-design stages of the HMI development, it is planned to carry out simulation modeling of TP and control processes (operator activities). The simulation results will allow more precise specify of the requirements for the operator activities and HMI at the terms of reference creation stage and to increase the validity of the task allocations between operator and machine in the early stages of HMI developing. If there are several “paper” prototypes or real HMI, the simulation will allow justifying the choice of the optimal variant based on quantitative indicators.

The practical significance of the obtained results is that the software realizing the proposed method is developed and can be used to solve practical problems of HMI design.

The experimental results of the Automated Control System operating analysis and modeling for aerated concrete production [26] have shown that the proposed method can be used to solve practical problems of HMI designing, as well as optimizing the structure and methods of TP controlling.

Acknowledgment

The work was performed as part of the project “Development of methods, models and information technology for managing complex objects under conditions of stochastic uncertainty” (number of 04918) of Zaporizhzhia National Technical University.

References

1. Five contributing factors of human error. https://wonderwarepaocwest.com/news/operatr-five-contributing-factors-human-error/
2. Muratov, O., Tihonov, M. The human factor at nuclear object. http://www.proatom.ru/modules.php?name=News&file=print&sid=1555 (In Russian)
3. 2018 Self-driving safety report. General Motors. https://www.gm.com/content/dam/company/docs/us/en/gmcom/gmsafetyreport.pdf
4. Cochran, E., Bullemer, P.: Abnormal Conditions Management: Not by New Technology Alone. Proceeding of the AIChE Safety Conference, Houston, TX (1996)
5. Lorenzo, D.K.: A manager's guide to reducing human errors: Improving human performance in the chemical industry. Arlington, VA: Chemical Manufacturer's Association (1990)
6. EU-OSHA: The human-machine interface as an emerging risk. Luxembourg: Office for Official Publications of the European Communities (2009). doi: 10.2802/21813
7. The concept of the SCADA-system TRACE MODE functions development based on the expert systems of decision-making and execution technology. http://isup.ru/articles/2/241/ (In Russian)
8. Be an Expert Builder of Expert System / Gensym. http:// www.gensym.com/ product
9. Lange, T.: Intelligent SCADA Systems. In: Engineer IT. Automation and Technical.Control April 2007, pp. 26-30 (2007)
10. Munipov, V.M.: Ergonomics: human-oriented design of equipment, software and environments: Textbook. – M.: Logos, 2001. – 356 p. (In Russian) (2001)
11. Kaminska, Zh. K.: Analiz of problems of automation of process ergonomic design of information models of tekhnological processes. In: Bulletin of Zhytomyr State Technological University. Technical science. Issue 1 (52), Zhitomir, – 2010, pp.103-108 (In Ukrainian) (2010). doi: https://doi.org/10.26642/tn-2010-1(52)-103-108
12. Sukhov, S. A.: Development of a system for monitoring operational data based on WEB-technologies: Phd Theses: 05.13.11. Institute of High Energy Physics. – Protvino. – 87 p. (In Russian) (2007)
13. Poletykin, A. G.: Research and development of methods and software for information control systems of the upper block level: Doctoral Theses: 05.13.11. Institute of Control Sciences. – Moscow. – 160 p. (In Russian) (2009)
14. Sukhov, A. A.: Software systems for improving the quality of the interface of computer training systems: Phd Theses: 05.13.11. Mosk. state un-t instrument and informatics. – Moscow. – 132 p. (In Russian) (2010)
15. GOST 21480-76. Man-machine system. Mnemo-schemes. General ergonomic requirements. Introduction term 01.01.1977. – M.: Standards publishing house.– 6 p. (1976)
16. GOST 28392-89 Aircraft mnemo-schemes. General ergonomic requirements. Introduction term 01.01.1991. – M.: Standards publishing house. – 15 p. (1990)

17. ANSI/ISA-101.01-2015, Human Machine Interfaces for Process Automation Systems. Julay 2015, 64 p. https://www.isa.org/store/ansi/isa-10101-2015,-human-machine-interfaces-for-process-automation-systems/41921133 (2015)
18. Burns, C.M., Haidukiewicz, J.R.: Ecological interface design. – Boca Raton: CRC Press (2004)
19. Upton, C., Doherty, G.: Extending ecological interface design principles: a manufacturing case study. In: International Journal of Human Computer Studies. 2008. 66 (4), pp. 271–286 (2008). doi: https://doi.org/10.1016/j.ijhcs.2007.10.007
20. Innovating for people: Handbook of human-centered design methods. Pittsburgh, PA: LUMA Institute, LLC (2012)
21. Fiset, Jean-Yves: Human-machine Interface Design for Process Control Applications. ISA. USA. 171 p. (2009)
22. O'Brien, L.: ISA-101 Standard Addresses the HMI Lifecycle. https://www.arcweb.com/blog/isa-101-standard-addresses-hmi-lifecycle
23. Nazarenko, N.A., Kaminska, Zh.K.: Method of human-machine interface designing of SCADA-system. In: News SPbGETU "LETI" № 7. 2017, pp. 31–38. (In Russian) (2017)
24. Information-Control Man-Machine Systems: Research, design, test: Handbook / Under the total. ed. Gubinskogo A.I. and Evgrafova V.G. – M.: Mashinostroenie, 1993. – 528 p. (in Russian) (1993)
25. Ergonomics: principles and recommendations. Methodical manual. – M .: VNIITE. – 184 p. (in Russian) (1983)
26. Kaminska, Zh. K., Kulynych, E. M., Serdiuk, S. N.: The Automated Control System operating analysis and modeling for aerated concrete production. In: Radio Electronics, Computer Science, Control. 2018. № 4, pp. 214–225. (In Russian) (2018). doi: 10.15588/1607-3274-2018-4-21