FRAMEWORK FOR CREATING SINGLE INFORMATION WINDOWS

Volodymyr Kazymyr, Maryna Kharchenko

***Abstract**: We consider a software development and support of single information window, which is a web-based software system. Describe the architecture of the system. Attention is focused on the use of the formal apparatus of control E-nets for modeling single information windows. Describes the hierarchical structure of the model. Show the description of the framework. Consider an example of creating single information window for providing administrative services to the licensing of universities.*

***Keywords**: framework, single information window, control E-nets, imbedded models.*

***ACM Classification Keywords**: D.2.2 Design Tools and Techniques – Programmer workbench, Petri nets.*

Introduction

The cause of the principle of "single information window" can be the rapid development of the bureaucratization of the interaction of individuals or legal entities with public institutions that provide administrative services. Created in the time to improve the efficiency of such interaction bureaucratic apparatus began to hinder him.

The provision of any administrative service can be divided into three stages: 1) the submission of an application and the required documents by the client in a public institution, 2) consideration of the application and its processing, 3) the issuance to a client the drawn up administrative act. Until such time as will be issued an administrative act going a complex process of moving and handling of documents in a number of authorities, which are often geographically spread out and not linked to any information channel. This leads to significant time costs associated with not treating the client's request (useful work), but with the movement of documents. In such a way, arises the need to develop specific tools of interaction between clients and public institutions on the one hand, and between the authorities of state institutions on the other.

The authors propose a solution to this problem, which is to create a single information windows and the means of their development. The basic idea of a single information window is that for administrative services a natural or legal person (the client) does not need to apply for and bring the required documents to the state agency, coming to them personally. Enough to use the document feeder informational portal, which is accessible on the Internet. After a certain law time, client will be informed of the result of the request. Moving documents within the same institution carried out by means of the portal. Thus, the term "single information window" can be defined as an electronic form of interaction between a client who applied for the administrative services and the organization that the service it provides, as well as authorities in the organization. To implement such an interaction authors propose Internet-based software system, which is the framework of creation of single information windows.

Architecture of the framework

The framework is designed to solve such problems: 1) the reduction of the term of development and modification of single information windows, 2) reduction of the term of processing client's requests for administrative services, 3) automating of the process of providing administrative services.

Fig. 1 shows the architecture of the developed framework.

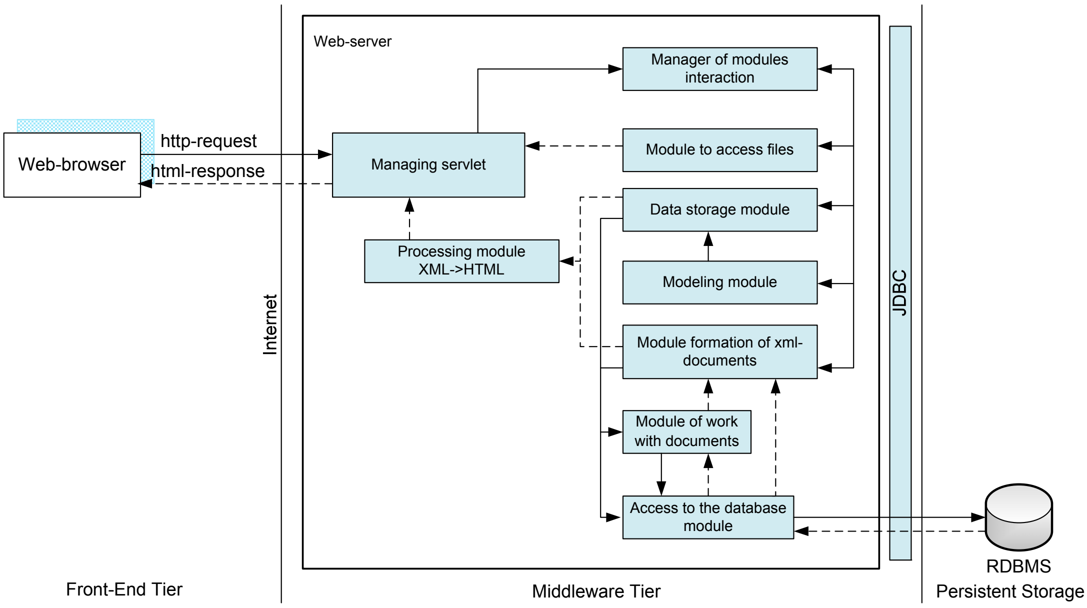


Fig. 1. Architecture of the framework

Architecture of the framework consists of three layers: the presentation layer (Front-End Tier), the business logic layer (Middleware Tier) and the layer of data retention (Persistent Storage).

The presentation layer is responsible for the visual display (user interface). Its purpose is to allow the user access to functions and data systems through visual components.

The business logic layer implements all the functionality of the framework. It consist of several modules forming the software core:

- managing servlet accepts http-request from the user’s web-browser, resolves the request into object form and sends it to the manager of modules interaction;
- manager of modules interaction analyzes the incoming request and passes control to the module that it can handle;
- in the module to access files are implemented functions of loading, uploading and delete files in the format Doc, Rtf, Rar, Zip. These files include statements that delivers the client of single information window, the archive of the required documents, the file of an administrative act with the decision, etc;
- data storage module stores all data in database;
- the modeling module is designed to allow the user to create and maintain a model, which will realize the process of providing a specific administrative service of single information window;
- module of work with documents processes requests for interpretation of the stored models by extracting from the database required data and matching them with the model. The model can be interpreted by parts. The result of the interpretation is a xml-document;
- in the module formation of xml-document the xml-files, created by the module of work with documents, are rebuilt among themselves according to the model hierarchy. The result of it is ready xml-file;
- processing module XML->HTML is a XSLT-processor that converts xml-documents in their html-description and forming at their base elements of the user’s interface framework;
- access to the database module implements the functions of save, edit and delete data.

The layer of data storage represented by relational database comprised of such tables:

- documents – keeps the elements of the model and the user interface in the xml format;
- attributes – store attributes model elements and entities system: the element type (model, action, subaction, title, department, executor, user, client, etc.), time of creation and modification time, owner's ID;
- links – the table of links between model elements and user interface elements. Records of the table provides to create the instance of a process model providing a service or user interface from records of documents table;
- reminder – the template of message are stored, which are required to send users the developing of single information window;
- right – access rights of users to the system.

All the elements of a process model of the service described by the corresponding templates in xml format and stored in the database. In the construction of a formalized process model templates are interconnected and form a ready xml-file model that also saves in the database.

Simulation of the process of providing administrative service

Any process of providing administrative services can be represented as a sequential movement of documents filed by the client and its processing by the employees of a governmental agency. This sequence is strictly regulated and is registered in the state regulatory base. For its information support necessary to formalize such a description, presenting it in terms of the functioning of a single information window. In the quality of the formal apparatus was chosen the formal apparatus of control E-nets [Kazymyr, 2005]. This choice is due to the fact that the whole process of providing administrative service can be represented as a directed graph of discrete states, and its dynamics – in the form of processes of moving documents in the network [Kazymyr, Kharchenko, 2011].

Control E-net consists of a certain way (in accordance with a logical sequence, which is described in the standard documentation) by linked E-net's transitions. Each transition implements the stay of the label circulating through the network in varying position. The transitions are fixed by the actions specified in the regulations. Tag in the network corresponds to the document, and position – is an executor (official or client) that handles this document. At the modeling stage the model parameters of the process are determined, the attributes labels (documents) and are installed their initial values. A formal model of service delivery on the licensing by higher education institutions looks like this, as shown in Fig. 2.

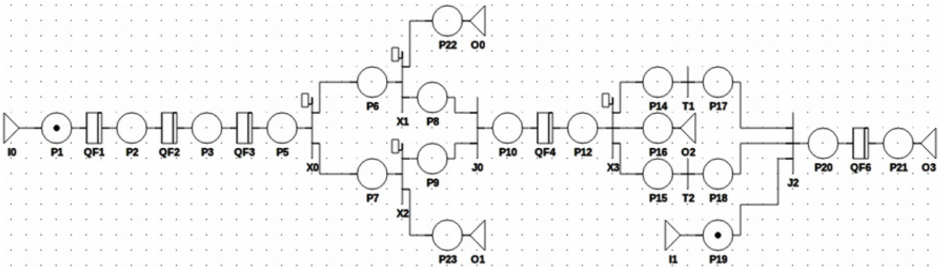


Fig.2. A formal model of university licensing

We describe the process of work of single information window represented by this model. Authorized representative in the matters of licensing of the university loads the required documents into the system, which are the licensed business. Immediately after loading (P1), the system automatically initiates sending an artifact of the case (card) and messages to the Office of the Ministry of Education and Science (QF1). Clerk registers case, assigns it a unique identifier (registration number) that is added to the attributes of the case and reported to the university (P2), and sends a card with a listed serial number to director of department of scientific work and licensing (QF2). After receiving information about a registered case, director of the department defines the executor and adds the information about him in the card case (P3). The system analyzes the input data and sends information about the case to the specified executor (X0). The term of the implementation of mentioned actions should not exceed 3 days from the filing of the case by the client.

The next step is the consideration of the case by the executer for compliance with the requirements (P6, P7). The term review is 15 days from the date of registration of the case. The result of examination takes the form of a preliminary decision that the same as the file of the case is loaded into the repository. In the case of a negative decision the case is returned to the university on revision. If the decision is positive, there is a modification of the attributes of its card (X1, X2) and is formed by a draft order to create an expert commission (P10). The draft order is loaded by executer in the system, and the reference to it is stored in the card of the case. The card is sent to the Ministry of Education and Science (QF4). After the signing of it in the Ministry (P12) the order is sent to the university and the Department of Licensing (X3), which is the basis for the start of the work of expert committee. After completion of the work at the university commission of experts is preparing a decision that should be loaded into the system. Reference to the decision automatically recorded in the card of the case (P19) after which it is sent for review to the State Accreditation Commission (J2). For two months, the Commission reviews the submitted case (P20), its representative forms decision to grant (or refusal to issue) a license to the university, establishes a specific attribute in a card of the case and informs the university about the result of their work (QF6).

Thus, the correspondence of position in the model the executer of the process and performed by him actions. Transitions correspond to the functions of the system. The label is represented by an artifact (card) with a history of processing of license case. It contains the following attributes: type of document (license case), date of receipt of the case by the licensing body; email address of authorized representative of the university, registration number of the case, date of registration, executer, working with the case, the mark of a preliminary decision (positive or negative) the approval mark of the order to create an expert commission, the date of the acception of the decision of the State Accreditation Commission, the decision of the State Accreditation Commission, a reference to the licensed case, a reference for a preliminary decision, the reference to an order to set up an expert committee, a reference to the decision of the expert committee, a reference to the decision of the State accreditation Commission.

From the description it is clear that the model has a hierarchical structure (Fig. 3).

The above structure is applicable to modeling of any process of provision of administrative service. To formalize it authors proposed to use xml language.

One of the most popular technologies used to represent the structured hierarchical data is XML [Evjen, 2007]. XML language has a hierarchical structure that is very well suited to describe the multi-level models. The model of hierarchical structure shown in Fig. 3 in the language xml looks like this:

```
<model>
        <name/>
        <author/>
        <alias>
                <doctype><![CDATA[link to /root/admin/doctype]]></doctype>
        </alias>
        <defination/>
        <config/>
```

```
    <action>
        <subaction>
            <action/>
        </subaction>
        <responsible>
            <user><![CDATA[link to /root/admin/deprtment/user]]></user>
            <customer><![CDATA[link to /root/admin/deprtment/customer]]></customer>
            <department><![CDATA[link to /root/admin/deprtment]]></department>
        </responsible>
    </action>
</model>
```

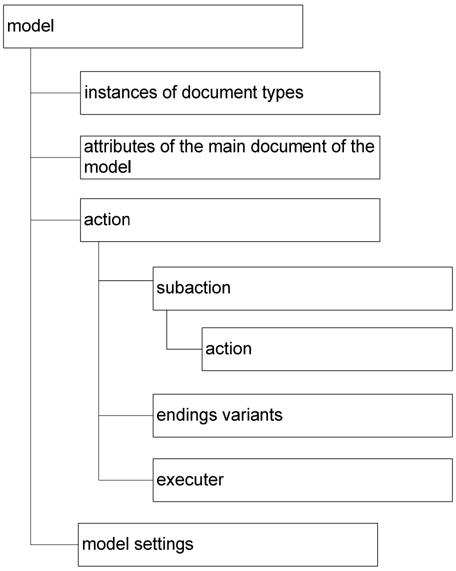

model
instances of document types
attributes of the main document of the model
action
subaction
action
endings variants
executer
model settings

Fig. 3. The hierarchical structure of the model of the process of providing administrative service

Tags <name/> and <author/> carry proprietary information of the model (name, author).

In tag <alias> defines copies of documents that will be moved in the model.

In <defination/> tag defines the structure of the main document (set of documents) model, which is supplied by the client.

<config/> tag defines the initial configuration of the model: the name for the user, the initial stage, the attributes available to the client upon completion of the model.

In the tag <action> steps are illustrated done by stakeholders. Each action may consist of several sub-actions <subaction> (they can still be called a variant of development of the process) that define the logic of the process. Each subaction have a link to the next action. If the subaction is the latest in the model, this link is not present.

All actions in the model are fixed to executors or responsible (tag <responsible/>). The client, individual employee or department of state establishment can be an executor.

The shown structure will have models of all the processes of administrative services provision. The difference will be only the contents of tags.

Description of the framework

Designed framework of creation of single information windows is an Internet-oriented software system that allows to develop and maintain the functioning of the single information windows for the provision of administrative services. For its realization has been used a three-tier client-server architecture. Within this architecture the sphere of the functioning of the system consists of a DB server, application server and client's web-applications.

The application server receives and processes requests from users of the system. In order to receive data from the server by client's web-application, it is necessary that the user has logged on. To login you need to enter username and password. This information is sent to the server, is checked, and if the user is already registered, he gets the access to the system.

Access rights to the system are limited to four roles: the superuser (root), administrator, client and executer. For each role assigned separate functions of management and work with the framework, as well as working with a single information window, developed with its help. Consideration of the functions of the framework is convenient to carry on roles.

Superuser (root)

Functions of superuser focused on work with administrators: its terms of reference include the creation and deletion of their accounts.

Administrator

The administrator is a main user of the framework. Functions that available to administrator are: building and managing process models of administrative services (creating a single information window); search function for users, documents, models, attributes of models, stages, executers, etc; managing of active users (administrator can complete a particular user session by removing him from the system); exchange messages with the users of the system; register new clients of the single information window (verifies the data and adds the client to the system).

Client

After registration and confirmation profiles by administration client can log in the system using its login and password. Having successfully passed the authorization, it gets access to functions.

As an administrator. client can view and change its personal data. Also client can send message to administrator and receive system messages with information about processing the submitted case; apply for the provision of administrative services with the necessary package of documents (case); view the condition of the active cases and a history of processed archive cases.

Executer

Executors – are employees of state agencies that handle client requests and decide on the issuance of an administrative act. Functions that available to executer are: login in the system, send message to the system users, processing the cases. The functions for processing cases depends on the settings of the model.

It should be noted that the user Executor and the Client have been working with a created single information window.

An example of using a framework to create an single information window

Creating a single information window by the framework tools does not required a lot of time. Consider it an example of the process of providing administrative services to the licensing of university. Before building model of the process involves two preparatory stages.

First, to identify all types of documents which will be used in the model. For example, the document type "Case" is the type of main document supplied by the client. Submission of a document of this type triggers the start of the fulfilling of process. Types of the "Order", "Resolution", "Protocol" are optional and their copies of documents are generated from the process of implementing government agency already while it is running.

Second, determine the structure of the organization, which provides administrative services – departments (sub-departments), employees and positions held by employees.

Preparations and the construction of the model is performed by the administrator of framework using the "Settings" menu (Fig. 4).

By clicking the "Document Types" you can view the existing document types of single information window, add a new one or delete the old one (Fig. 5).

To add, edit and delete posts, use the button "Post".

The button "Departments" is intended to form the organizational structure of the institution. After clicking it opens the editor window (Fig. 6), which adds information about departments, sub-departments and their employees. In the future, the construction of a model from the list of employees will be assigned the performers of stages of model.

After the preparatory stages of the process model is developed. We describe the actions to be taken at that time.

To create a new model, you can press the "New Model" and fill in the Model name, Description and Author (Fig. 7).

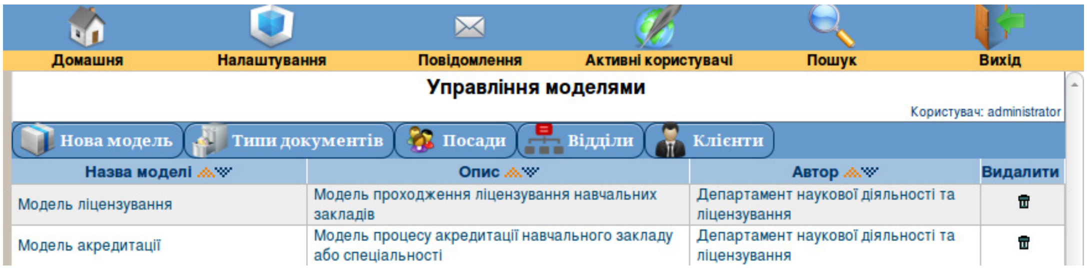
For each of the new models are instances of type documents determined at the preparatory stage (Fig. 8). For example, the licensing case (type "Case"), protocol of the State Accreditation Commission (type "Protocol"), reasoned written refusal to hold licensing (type "Resolution"). The model can be used by multiple instances of the same type of documents.

The next step is to define the attributes of the model (Fig. 9). They detail the progress and results of the process and carry informative nature to the participants (clients and executors.) Participants can view or edit them. For example the attributes of models may be the date of registration of the license case, the registration number and date of work of the expert commission in university, etc.

After performing the above actions by means of a graphical editor (Figure 10) is constructed as a graphical representation of the model [Kazymyr, Kharchenko, 2012] of the process as control E-net (call the editor is done by clicking on the appropriate model from the list). It identifies the steps, options for their completion, setting of their characteristics.

Stage is characterized by the name, the term of implementation, the type of executor, the attributes that are available for viewing and possible options for completion. Executor may be the department of institution, a specific employee or client. For example, for the stage "Checking the case" staff of institution is the executor – head of regulatory support and licensing expertise.

Next to that performed in the present stage of an option is determined by the completion of the latest. This will create a logical sequence of steps of the process model and determine the further course of the process. Option completion is characterized by attributes and documents available for editing and uploading tap by executor, as well as the next step to be executed if the current will end by this option. For example, for the stage "Checking the case" there are two possibilities of completion: "the case meets the requirements" – the transition to the next stage, no attributes, and the documents can not be edited and loaded, "the case does not meet the requirements" – the completion of the process, the attribute of editing – "cause of the completion", document – "reasoned written failure to licensing".

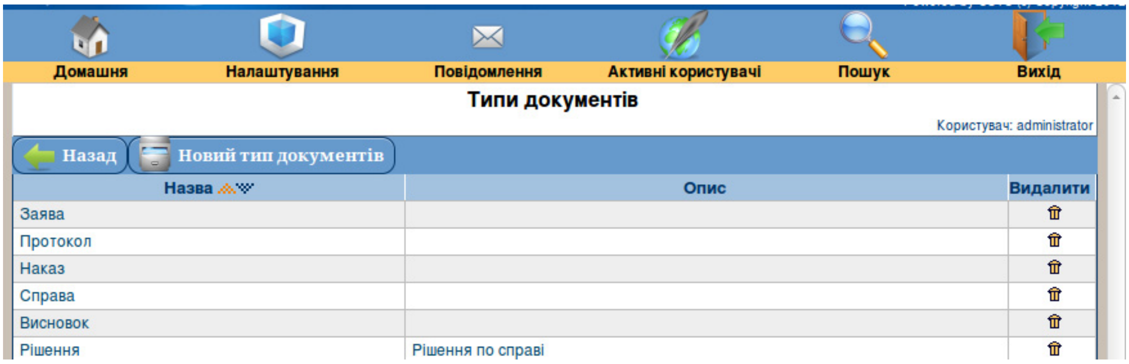


Управління моделями

Користувач: administrator

Назва моделі	Опис	Автор	Видалити
Модель ліцензування	Модель проходження ліцензування навчальних закладів	Департамент наукової діяльності та ліцензування	🗑
Модель акредитації	Модель процесу акредитації навчального закладу або спеціальності	Департамент наукової діяльності та ліцензування	🗑

Fig. 4. "Settings" menu

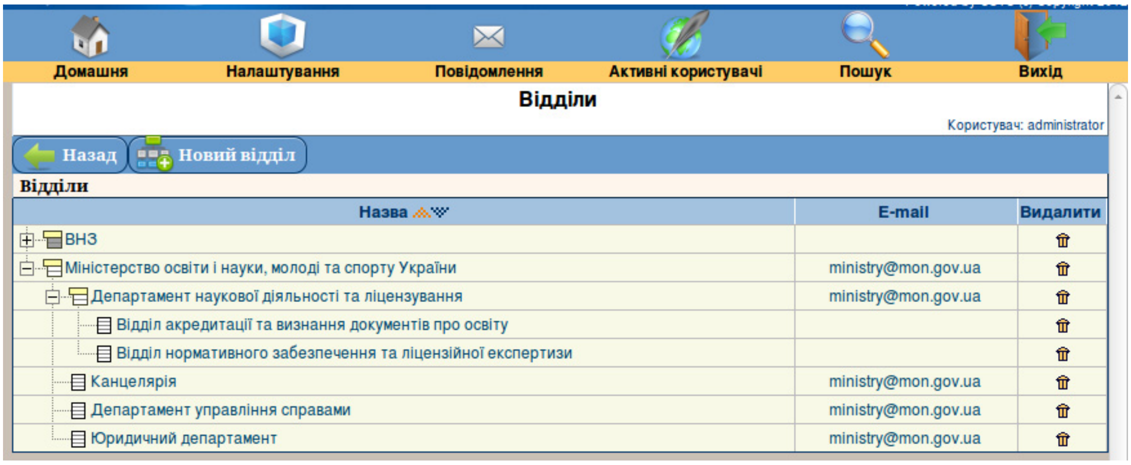


Типи документів

Користувач: administrator

Назва	Опис	Видалити
Заява		🗑
Протокол		🗑
Наказ		🗑
Справа		🗑
Висновок		🗑
Рішення	Рішення по справі	🗑

Fig. 5. Page of the control of the types of documents



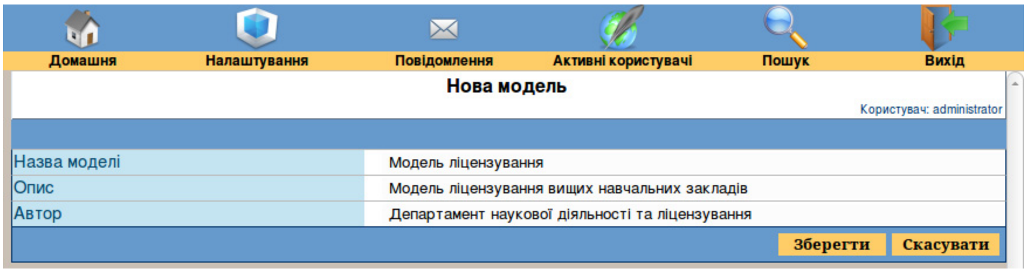
Відділи

Користувач: administrator

Відділи

Назва	E-mail	Видалити
ВНЗ		🗑
Міністерство освіти і науки, молоді та спорту України	ministry@mon.gov.ua	🗑
Департамент наукової діяльності та ліцензування	ministry@mon.gov.ua	🗑
Відділ акредитації та визнання документів про освіту		🗑
Відділ нормативного забезпечення та ліцензійної експертизи		🗑
Канцелярія	ministry@mon.gov.ua	🗑
Департамент управління справами	ministry@mon.gov.ua	🗑
Юридичний департамент	ministry@mon.gov.ua	🗑

Fig. 6. Page of the department management of institutions

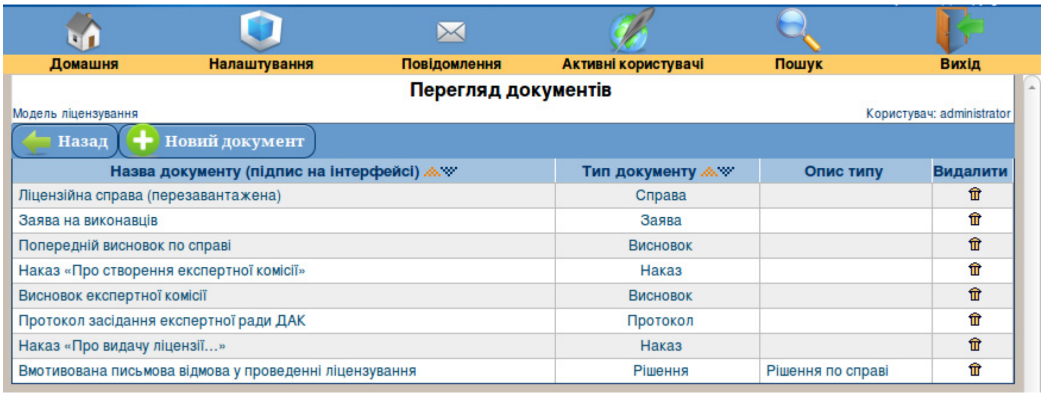


Нова модель

Користувач: administrator

Назва моделі	Модель ліцензування
Опис	Модель ліцензування вищих навчальних закладів
Автор	Департамент наукової діяльності та ліцензування

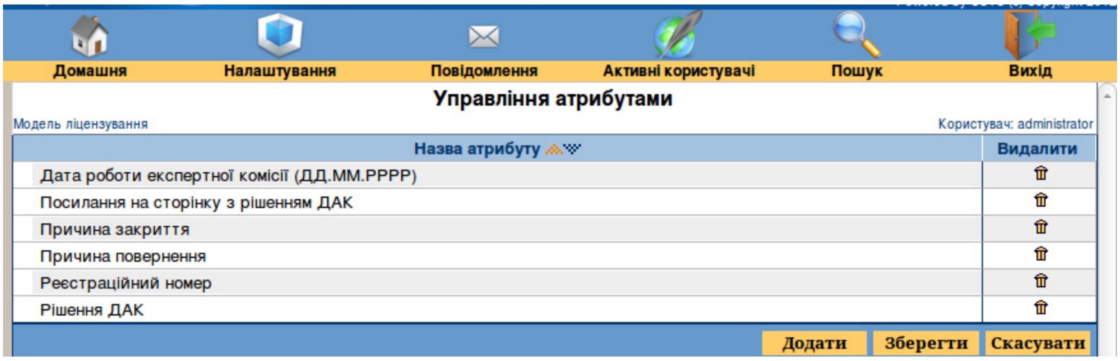
Fig. 7. Page of the creation of a new model



Перегляд документів

Назва документу (підпис на інтерфейсі)	Тип документу	Опис типу	Видалити
Ліцензійна справа (перезавантажена)	Справа		🗑
Заява на виконавців	Заява		🗑
Попередній висновок по справі	Висновок		🗑
Наказ «Про створення експертної комісії»	Наказ		🗑
Висновок експертної комісії	Висновок		🗑
Протокол засідання експертної ради ДАК	Протокол		🗑
Наказ «Про видачу ліцензії…»	Наказ		🗑
Вмотивована письмова відмова у проведенні ліцензування	Рішення	Рішення по справі	🗑

Fig 8. Page of instance management of model documents



Управління атрибутами

Назва атрибуту	Видалити
Дата роботи експертної комісії (ДД.ММ.РРРР)	🗑
Посилання на сторінку з рішенням ДАК	🗑
Причина закриття	🗑
Причина повернення	🗑
Реєстраційний номер	🗑
Рішення ДАК	🗑

Fig. 9. Page of the control the attributes of model

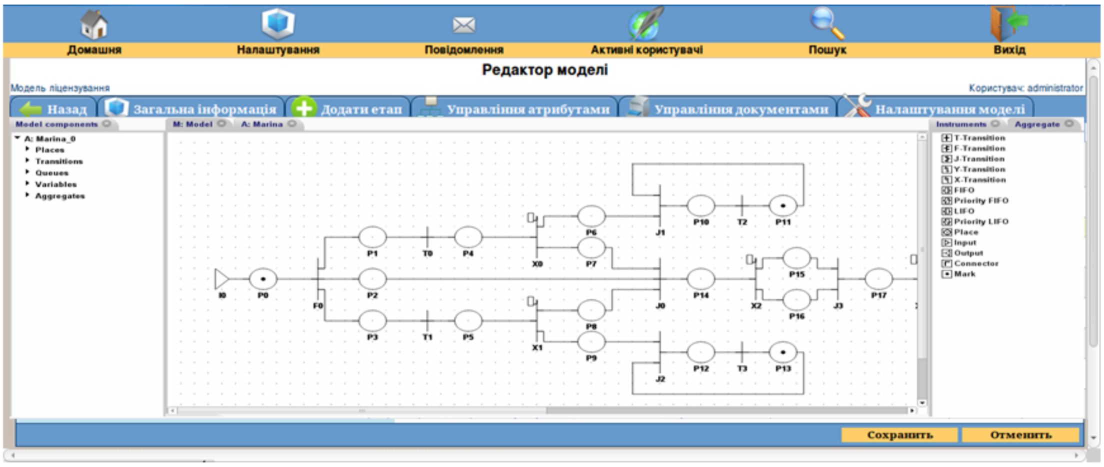


Fig. 10. Graphic editor of models

The created model is stored in the system as a format xml. Using the method proposed in [Kazymyr, 2002], on its model basis the interface forms of the stages. When a client is initialized the main document, the model becomes active and it manages progress of the process in real time.

After building the model, you must perform its initial configuration. The initial setting is to set the overall duration of the model (the number of days from the date of submission of documents by the client prior to the date of issue of an administrative act), the identification of the main document, available for viewing attributes and documents to complete the model execution. For example, setting up the model of the licensing process will be such: duration of the performance – 120 days, the initial phase – "Pre-check", attributes – "Registration number of the case", "Link to the solution," documents – the order "About the license".

Thus, by creating models for each administrative service provided by the institution, it creates a single information window.

Conclusion

The article describes the creation of a framework of single information window that allows you to design and maintain their functioning. The features of the framework are: the availability of a convenient tool for modeling the processes of administrative services, the preservation of the created models in the form of hierarchical structures with the use of the language XML; language use XSLT transformation to interpret formal models of the XML format in HTML; created by the use of models as control algorithms work single window in real-time. The framework reduces the time processing client requests for administrative services, as well as automate the process of providing it.

Special feature of the model approach is that it allows you to generate dynamically the user interface to perform the steps of the process and set the algorithm for moving model documents – the model manages the process. This approach can be used to implement single information windows of any organization that provides administrative services and can also be used to build electronic document management systems.

Acknowledgements

Authors of the article are grateful to Prof. Voloshin O. for the invitation to participate in the conference and to discuss the results.

Bibliography

[Evjen, 2007] Bill Evjen, Kent Sharkey, Thiru Thangarathinam, Michael Kay, Alessandro Vernet, Sam Ferguson. Professional XML. Wiley Publishing, Inc. Indianapolis, 2007.

[Kazymyr, 2002] V.V. Kazymyr. V.M. Miroshnichenko, A.V. Pastuhov. The method of designing of the Internet-focused information control systems based on XML standard // Problems of programming. – 2002. - №1-2. P. 326-332.

[Kazymyr, 2005] Kazymyr V.V. Model-Based Control and Intelligent Manufacturing Systems: the thesis for the degree of Doctor of Technical Sciences: 05.13.06. – Kiev, 2005.

[Kazymyr, Kharchenko, 2011] V.V. Kazymyr, M.V. Kharchenko. Modelling working procedures in the support of the quality management system // The sixth scientific-practical conference with international participation “Mathematical Modeling and Simulation of Systems. MODS'2011”. Abstracts. – Chernigov, 2011.

[Kazymyr, Kharchenko, 2012] Kazymyr V.V., Kharchenko M.V. Model-oriented information system supporting quality management of university, Chernigov, 2012.

Authors' Information

***Volodymyr Kazymyr** – Dr. Sc., Prof.; Chernihiv State Technologacal University, Shevchenko street, 95, Chernihiv-27, Ukraine, 14027; e-mail: vvkazymyr@gmail.com*
Major Fields of Scientific Research: Object-oriented programming, Model-based control, Simulation.

***Maryna Kharchenko** – postgraduate; Chernihiv State Technologacal University, Shevchenko street, 95, Chernihiv-27, Ukraine, 14027; e-mail: kharchenko.maryna@gmail.com*
Major Fields of Scientific Research: Data bases, Information systems, Model-based control.