КРАТКИЕ СООБЩЕНИЯ

MODELS AND METHODS OF COMPUTER – AIDED DESIGN OF THE USER INTERFACE OF SOFTWARE SYSTEMS

N.S. Vaytsel[1], *O.A. Bubareva*[1]
[1]Biysk Technological Institute, Biysk, Russian Federation
E-mails: vaitsel.nikita@yandex.ru, angel@bti.secna.ru

The article deals with the design, mathematical modeling and development of the prototype of the interface of the information system of the audit fund. In this paper, the study of the design of the user interface, methods for creating interfaces, user interaction types, a list of the functionality of the system, a mathematical model, nondeterministic finite automaton (NFA) was built user interaction with system elements and a prototype of the system have been created. According to the authors, this system will automate and simplify the process of managing the audit fund, joint use of the NFA and one of the methods of creating interfaces, namely, ontology will allow describing domain concepts, edit domain concepts, generate databases and knowledge bases, broadcast from one language to another, get rid of information redundancy, reduce time for database development as well as simplify the process of scheduling the university.

Keywords: user interface design; mathematical model; information system prototype classroom fund.

Introduction

The user interface is an important element of any programme on which the user's efficiency with the system depends. In view of the increasing complexity of information systems and user interfaces, special tools are being developed to support the development of user interfaces. The sources [1–4] provided theoretical bases for designing user interfaces.

In the given studies a concept of the user interface, methods to design user interfaces, and interface design tools supporting the design, modification and automatic generation of the code of all interface components were considered in detail. However, they require the developer to study specialized declarative languages to describe interface components, or they are based on formalisms, for example, UML (Unified Modelling Language) which reduces the intellectual support of the developer. Some tools do not satisfy the requirement of openness, therefore, when the IS (information system) enters the market. Its modification to the available requirements is already required. This article is aimed at studying methods of designing user interfaces and use henceforward the results obtained in designing a prototype of the real-time system interface in the field of automated control of classroom facilities at the Biysk Technological Institute.

Formulation of the Problem

Thus, the tasks to be solved within the article can be determined:
1) study the design process and methods for making user interfaces;

2) determine the requirements for the developed classroom facilities system being developed;

3) to develop a prototype of the interface of the classroom facilities information system.

1. Theory

1.1. The Process of Designing a User Interface

In many well-known methods to design interfaces there are tools for the systematic and effective reform of knowledge collected in the course of studies into a detailed specification of the interface. One of the reasons is the designers are disconnected from the research cycle and as a result they have to rely on someone else's notions about users' actions and wishes.

Another reason is the users behaviour in a form that is suitable for forming a product definition is fixed by several approaches [1, 2]. Most of the methods provide information at the task level, not users' purposes. The information of this type does not have any benefit to the definition of a common infrastructure that depicts what the product does, how it satisfies the users' requirements and what it is. This information is best suitable for making layouts, modelling the workflow and converting functions into user interface components [1, 2]. To overcome this gap, it is necessary to create user models, define requirements for the user interface and convert them into a single concept of interaction. To do this it is necessary to use goal-oriented design which allows eliminating the existing gap between the researches of users' auditorium and designing in the course of development combining both new and previously known approaches [1, 2].

A targeted design includes methods of modelling users, basic sets of templates and their design principles for interaction between the user and the interface, marketing a study, scenario-based design and also ethnography. On the one hand it allows creating solutions that meet users' needs and goals, and on the other hand business requirements and technological limitations. This process can be divided into six stages.

The first stage is a research. The next stage of the study is modelling. After that the requirements are worked out. Further, a common infrastructure is defined. Tracking and detailing are the last [1, 2].

1.2. Methods for Making User Interfaces

When making user interfaces it is possible to consider two different methods [3]:

1) in the manual approach the developer writes a code that processes users' events and is responsible for generating elements of the user interface. This approach is not always fast but it gives maximum control over the interface and processing events.

2) when using a graphic editor it becomes possible to depict the interface and watch how it will look. This method is much more obvious, however, it is almost always less flexible.

With a classical approach to designing the user interface on the base of users' requirements, layouts of the planned interface are formed in the form of prototypes presenting the interface in a graphical form. The description of the interface behaviour is a part of the prototype that appears either as a result of the interaction between the user and the product or as a result of emulating product behaviour [3, 4]. Using specifications the visual image of the product is created by designers, and the developers themselves

Вестник ЮУрГУ. Серия «Математическое моделирование
и программирование» (Вестник ЮУрГУ ММП). 2019. Т. 12, № 1. С. 122–128

123

are engaged in the implementation. Each participant of the design has a personal area of competence and responsibility.

This approach is successfully used in situations when interaction models based on a fixed set of control components have been set for a long time, for example, conventional WEB applications or classical applications. But this approach does not work well for modern Ajax (Asynchronous JavaScript and XML) and RIA (Rich Internet Application). This problem is solved using specialized interface description languages and development environments. This solution allows separating the application business logic from the interface appearance. Special interface description languages describe the appearance of elements in such a way that the interface is easily brought into effect in any editor [3, 4]. Further, classifications for user interface development tools will be considered.

1.3. Types of Interaction with the User

The separation of interface design from the application is the main concept of the user interface management system. Nowadays the concept of separate interface design is fixed in the definition of UIMS or is its main property.

Managing the interaction between the user and the application programme, as well as facilitating the development, maintenance of the interface and facilitation are the main functions of UIMS. Interaction with the user of the application programme and the interface is determined by its behaviour. Three types of interaction are usually distinguished: the dialogue is caused by the application programme, by the user or both of them.

Application programme initiative. This kind of management means that the programme requests some data from the user if it needs them. When it is necessary to enter the data that the system requires, then the user is engaged in the process.

User's initiative. This kind of management means that the user assumes the initiative, and such an opportunity is implemented in the interface.

Mixed management initiative. In this type of interaction the two approaches mentioned above are combined. Nowadays there are a lot of tools for developing user interfaces that support different ways of implementing them. But there is no standard classification of the proposed toolkit. It makes difficult for the user to choose a particular tool and also to compare them with each other [4].

1.4. Mathematical Interface Model

To construct a mathematical model of the system, it is necessary to describe the sets contained in the information system of the audit fund. This system contains four sets. The first set of the system is the authorized users of the system, therefore, many different users of the system can be represented as a set:

$$P = \{AP_1, AP_2, AP_3\}, \tag{1}$$

where AP_1 are users with access level for normal viewing, AP_2 are users with access level with the ability to make changes, AP_3 are users with administrator access level. The second set of the system is a set of interfaces:

$$I = \{Z_1, Z_2, Z_3\}, \tag{2}$$

where Z_1 is the interface for users with a normal user access level, Z_2 is the interface for a user with an access level with the possibility of making changes, Z_3 is the interface for a user with an administrator access level.

The third set is the set of actions produced by the user:

$$D = \{d_i j| = 1, 2, ..., j\}, \tag{3}$$

where $d_i j$ are actions performed by users, j is number of user actions.

The fourth set is the set of interface elements:

$$E = \{e_i j| = 1, 2, ..., j\}, \tag{4}$$

where e_{ij} is the user interface element, j is the number of interface elements. For two sets P and I there is a function P_I intended for displaying the interface to the user, this function has the form:

$$P_I : P\{AP_n\} \to I\{Z_n\}, \tag{5}$$

where AP_n is an authorized user of a system with a certain access level, Z_n is the user interface, depending on AP_n.

For sets D and E, there is a function that defines the actions performed by the user on the interface element:

$$D_E : D\{d_n\} \to E\{e_n\}. \tag{6}$$

There is also a function for the sets E and I that determines which element to display on the user interface:

$$E_I : E\{e_n\} \to I\{Z_n\}, \tag{7}$$

where Z_n is the user interface. As a result, the information system is:

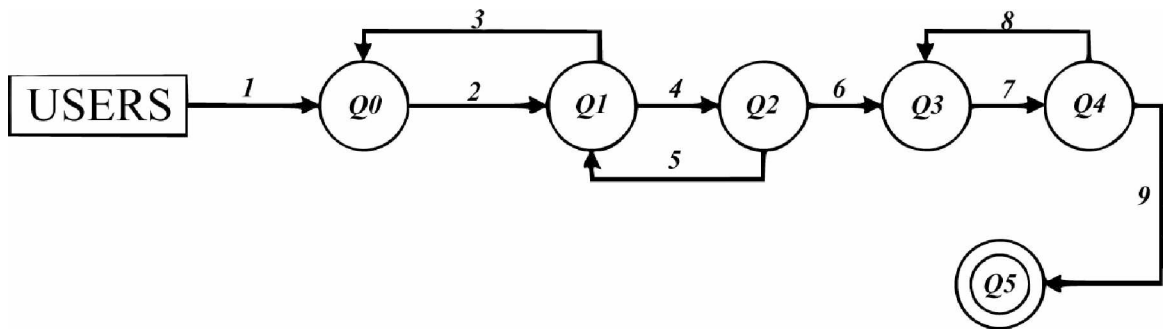$$AF = \langle P, I, D, E, P_I, D_E, E_I \rangle. \tag{8}$$

Also this system can be represented as a nondeterministic finite state machine. Consider a model in which the process of interaction of the user with the system is presented in the form of three elements of the system:

$$E_i = [\{P\}, \{D\}, \{E\}], \tag{9}$$

where E is the user interface element, P is the user set, D is the set of actions. Consider the implementation of FCM using the example of the interface element Information on the employment of the audience. The implemented NFA is presented in Figure.

Machine States: Q0 is selection page of the Institute building, Q1 is drop-down list of building selection, Q2 is choice of the building's audience, Q3 is information about the audience, Q4 is date selection, Q5 is final state (information about employment is displayed). Machine Transitions: 1 is the user pressed the "Select building" button, 2 is the user chooses the building, 3 is the user made a mistake when choosing a building, 4 is the user has correctly chosen the building and is given the opportunity to choose the audience of the building, 5 is the user made a mistake when choosing an audience, 6 is the user has correctly chosen the audience and he displays information about the audience, 7 is the user selects a date, 8 is the user made a mistake when selecting a date, 9 is the user has correctly chosen the date and displays information about the audience's employment.

According to the constructed model, it can be concluded that the interface can be represented in the form of an automation-graph model, and each element is represented in the form of an automation, this approach can predict the complexity of the interface.

Вестник ЮУрГУ. Серия «Математическое моделирование
и программирование» (Вестник ЮУрГУ ММП). 2019. Т. 12, № 1. С. 122–128

125

Building NFA information on employment of the audience

2. Prototype Development

2.1. Predesign Analysis

Before starting the interface prototype design for the information system of the classroom facilities, it is necessary to draw up a specification where the purpose of the system, the main purposes of the users' role, the functions of the system and an application field are described. Information system of the classroom facilities is a system that allows seeing the passport of a classroom, its occupancy at the moment, its location on the map of a building of the higher educational establishment (HEE).

The purpose of the system is accounting of the current condition of the facilities of the educational process as well as the analysis of the classroom facilities, control of deviations between actual indicators and planned ones. Users of the system are HEE teachers. The system should provide automation of the following processes:
1) accounting of classrooms with data on their equipment and condition;
2) prompt decision-making in operation and maintenance of the premises;
3) accounting of the classroom occupancy in the course of the educational process.
The following subsection provides information on the list of system functionality and requirements for the system.

2.2. Requirements Gathering

The functionality list is a list containing the actions that the user can perform in the system. The list provides an opportunity to set more accurately functional requirements, an accurate estimate of the project terms, planning and accounting of labour costs. The user can:
1) look through information on the number of classrooms of different types, their location and occupancy, main and additional equipment of each classroom;
2) delete non interesting blocks of information;
3) look through the classroom occupancy by dates.
In addition to the list of functionality, the system requirements were formulated:
1) preventing duplication of information and implementing the principle of single input;
2) providing adaptation of the system to new operating conditions while maintaining its operability;
3) possibility of relatively easy introduction of individual elements into the system.
This system uses the Oracle database that will allow this system being heterogeneous in the future. This is necessary to do to have a possibility to create a WEB interface and

it will also allow connecting the information system of the classroom facilities with the existing schedule system.

Conclusions

In the course of the given investigation, some methods to design a user interface, types of interaction with the user, the main interface specifications were studied. A prototype of the information system interface for the classroom facilities of the Biysk Technological Institute was developed that keeps records of the current condition of the material and technical facilities of the educational process, as well as occupancy analysis of classrooms, control of deviations between actual indicators and planned ones. The following promlems were accomplished:
1) the study of the design process and methods for creating user interfaces;
2) the requirements for the developed information system are defined;
3) developed a prototype of the interface of the information system.
In future it is planned to bring into effect the information system of classroom facilities and link it with the developed prototype of the interface.

References

1. Daytel G., Teplykogo L.A., Khoduleva A.B., Shtarkman V.S. *Introduction to Operating Systems*. Boston, Addison-Wesley, 1983.

2. Loge I., Hoffman Ed.K.L., Molchanovoy T.V. *Informacionnye sistemy* [Information Systems]. Moscow, Mir, 1973. (in Russian)

3. Belousova S.A., Rogozov Yu.I. Analysis of Approaches to Creating a User Interface. *Izvestiya Southern Federal University. Engineering Sciences*, 2014, vol. 3, pp. 82–86.

4. Gribova V.V., Kleshchev A.S. Methods and Tools for Developing the User Interface: the Current State. *International Journal of Software Products and Systems*, 2001, no. 1, pp. 521–530.

МОДЕЛИ И МЕТОДЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА ПРОГРАММНЫХ СИСТЕМ

Н.С. Вайцель[1], О.А. Бубарева[1]
[1]Бийский технологический институт, г. Бийск, Российская Федерация

В статье рассматривается проектирование, математическое моделирование и разработка прототипа интерфейса информационной системы аудиторного фонда. В данной работе проведено исследование процесса проектирования пользовательского интерфейса, методов создания интерфейсов, типов взаимодействия пользователя, определен перечень функциональности системы, построена математическая модель, был построен недетерминированный конечный автомат (НКА) взаимодействия пользователя с элементами системы, а также создан прототип системы. По мнению авторов, данная система позволит автоматизировать и упростить процесс управления аудиторным фондом, совместное использование НКА и одного из методов создания интер-

фейсов, а именно онтология позволит описывать понятия предметной области, редактировать понятия предметной области, генерировать базы данных и базы знаний, транслировать с одного языка в другой, избавиться от избыточности информации, снизить время на разработку базы данных, а также упростить процесс составления расписания ВУЗа.

Ключевые слова: пользовательский интерфейс; проектирование; математическая модель; информационная система; прототип; аудиторный фонд.

Литература

1. Daytel, G. Introduction to Operating Systems / G. Daytel, L.A. Teplykogo, A.B. Khoduleva, V.S. Shtarkman. – Boston: Addison-Wesley, 1983.

2. Ложе, И. Информационные системы. Методы и средства / И. Ложе. – М.: Мир, 1973.

3. Белоусова, С.А. Анализ подходов к созданию пользовательского интерфейса / С.А. Белоусова, Ю.И. Рогозов // Известия Южного федерального университета. Технические науки. – 2014. – Т. 3. – С. 82–86.

4. Грибова, В.В. Методы и средства разработки пользовательского интерфейса: современное состояние / В.В. Грибова, А.С. Клещев // Программные продукты и системы. – 2001. – № 1. – С. 521–530.

Никита Сергеевиич Вайцель, магистрант, кафедра «Методы и средства измерений и автоматизации», Бийский технологический институт (г. Бийск, Российская Федерация), vaitsel.nikita@yandex.ru.

Олеся Александровна Бубарева, кандидат технических наук, доцент, кафедра «Методы и средства измерений и автоматизации», Бийский технологический институт (г. Бийск, Российская Федерация), angel@bti.secna.ru

128

Bulletin of the South Ural State University. Ser. Mathematical Modelling, Programming
& Computer Software (Bulletin SUSU MMCS), 2019, vol. 12, no. 1, pp. 122–128