

## **ПРОЕКТИРОВАНИЕ КОМПЬЮТЕРНЫХ СИСТЕМ С ИНТЕЛЛЕКТУАЛЬНОЙ ДИАГНОСТИЧЕСКОЙ ИНФРАСТРУКТУРОЙ**

### **Введение**

В связи с развитием Internet, ростом численности электронных коммерческих приложений, систем поиска большое внимание в области разработки средств вычислительной техники уделяется проблеме «персонализации» компьютерных систем различного рода, что выдвигает ряд требований к разработке интерфейса. В идеале интерфейс – это не только посредник, помощник, но и средство, контролирующее уровень нагрузки пользователя и его эмоциональное состояние. Именно такие требования к разработке интерфейса являются первостепенными [1].

Интерфейсы, реализующие данные требования, создаваемые с использованием методов искусственного интеллекта, в частности генетических алгоритмов, моделей Маркова, сетей Байеса и др., называются интеллектуальными. Согласно [2, 3] они представляют совокупность программных и аппаратных средств, обеспечивающих пользователя, не имеющего специальной подготовки в области вычислительной техники, возможностью применения компьютера для решения задач, возникающих в области профессиональной деятельности, либо полностью без посредников-программистов, либо с незначительной помощью. Отличительными особенностями таких интерфейсов является наличие знаний о предметной области, системной модели мира и пользователе [4 – 7].

### **Постановка задачи**

Проблеме моделирования пользовательского интерфейса посвящено большое количество публикаций, ориентированных на создание поисковых систем для Internet, рекомендующих и обучающих систем и др. Такие системы сложны и довольно дорогостоящи. Отечественные исследования проводятся, в основном, в учебных заведениях для создания специализированных систем. Особенность таких систем – существенные затраты времени пользователя на выполнение многочисленных тестовых заданий при создании модели студента с целью повышения успеваемости студентов и усвоения лекционного материала.

Для систем массового использования возникает необходимость при сохранении тех же качественных показателей существенно упростить сбор информации о пользователе и минимизировать личные затраты времени пользователя на этот сбор. В связи с этим проблема построения интерфейса, ориентированного на широкого пользователя является чрезвычайно актуальной. В работе предлагается один из подходов к решению данной проблемы.

### **Обобщенный алгоритм создания модели пользователя**

Предлагаемый подход основан на использовании мета-онтологии построения модели пользователя, что позволило выделить необходимые понятия и взаимосвязи между ними, и с учетом вышеуказанных требований создать модель пользователя. Предлагаемая интерфейсная система, включающая эту модель, обеспечивает за счет адаптации эффективную работу пользователя с системой, подстраивая систему под уровень владения пользователем компьютером с учетом его эмоционального и физиологического состояния особенно в начальный момент взаимодействия с компьютером. Обобщенный алгоритм создания такой модели показан на рис. 1.

В блоке входа в систему происходит распознавание пользователя: впервые он обращается к системе или же он постоянно с ней работает. Если он известен системе, то после прохожде-

ния теста в блоке тестирования эмоционального состояния пользователя ему разрешается работать с системой, предварительно настроив систему, компьютер для данного пользователя через блок анализа и выделения ресурсов в соответствии с его моделью. Если он впервые обращается к системе, то интерфейс создает модель этого пользователя на основе полученных от него сведений.

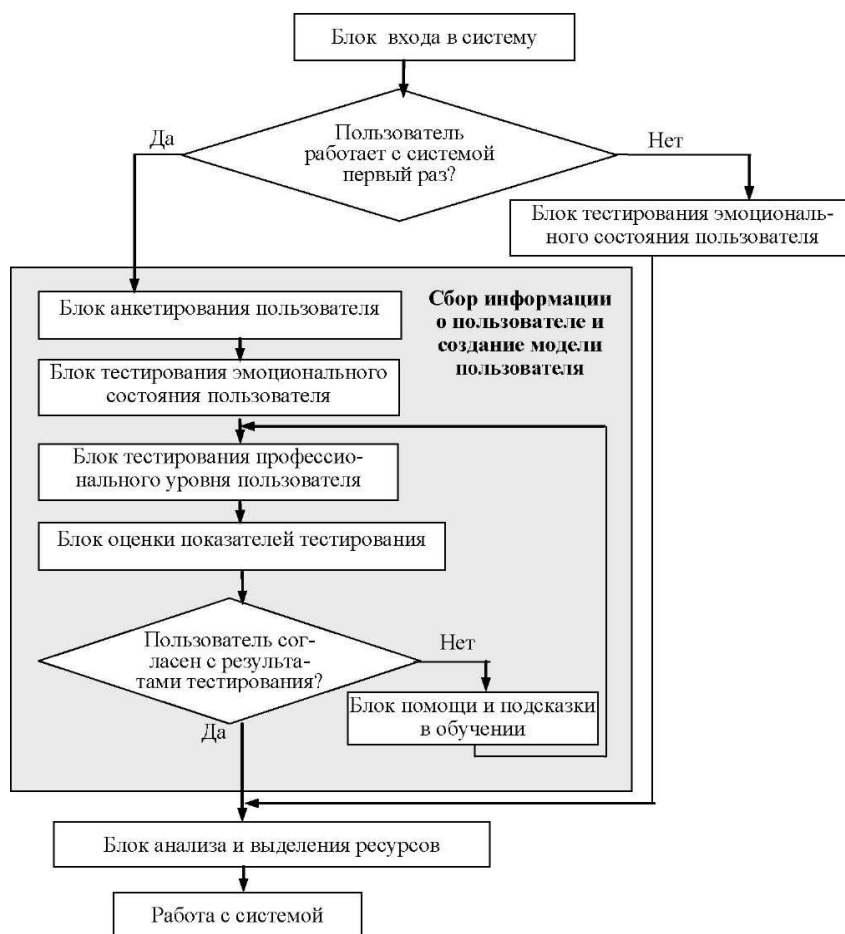


Рис. 1. Обобщенный алгоритм создания модели пользователя

Создание модели пользователя начинается в блоке анкетирования. Существует много методов сбора информации о пользователе: анкетирование, тестирование, измерение физических показателей с помощью датчиков, отслеживание событий пользовательского интерфейса. Применение того или другого метода определяется целями и назначением информационной системы. В данном случае (при требованиях минимизации времени на анкетирование пользователя) используется анкетирование и тестирование.

Пользователю предлагается анкета для заполнения, на основании результатов тестирования делаются выводы о принадлежности пользователя к одной из трех категорий в зависимости от уровня владения компьютером (новичок, пользователь, специалист).

Далее, в блоке эмоционального тестирования определяется эмоциональное и физиологическое состояние пользователя. За основу тестирования взят восьмицветовой тест Люшера. Преимущество данного теста перед другими в лаконичности, удобстве, в том, что он лишен культурно-этнических основ и не провоцирует реакций защитного характера [8 – 9]. По результатам теста делаются выводы о нахождении пользователя в одном из четырех состояний (высокая работоспособность, нормальное рабочее, низкая работоспособность, тревожное).

После того, как пользователь прошел анкетирование, тестирование и сделаны предварительные выводы о нем, блок тестирования профессионального уровня с помощью тестовых заданий осуществляет окончательную проверку знаний об уровне пользователя владением

компьютером. Тестовые задания предлагаются на основании результатов анкетирования, полученных в блоке анкетирования. Диалог между системой и пользователем ведется с учетом результатов, полученных в блоке эмоционального состояния.

Далее, в блоке оценки показателей тестирования, все показатели, полученные в блоке анкетирования, предоставляются на утверждение пользователю. Если пользователь согласен с результатами тестирования, он получает доступ к работе с системой, если же нет – предлагается пройти обучение в блоке помощи и подсказки в обучении (рис. 1).

Для пользователя, получившего доступ к системе, адаптация интерфейса происходит в блоке анализа и распределения ресурсов. В соответствии с полученными результатами тестирования, в том числе и в зависимости от поставленной задачи, устанавливается стиль и способ взаимодействия с пользователем (диалог, среда, известная пользователю). Настройка интерфейса, в том числе и рабочего стола, происходит с учетом вкусов (шрифт, цвет, графика). Контроль над интерфейсом в пределах задачи устанавливается в зависимости от уровня владения пользователем компьютером. Модель пользователя реализуется программным способом на базе агентной технологии. При этом каждый блок алгоритма (рис. 1) представляет собой отдельного агента. Рассмотрим несколько подробнее блок анкетирования. Алгоритм работы данного блока показан на рис. 2.

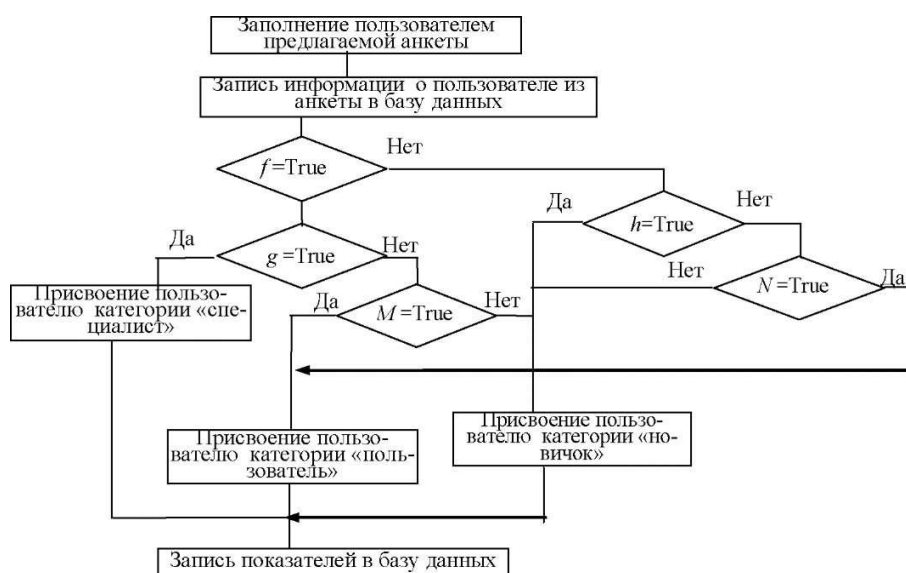


Рис. 2. Алгоритм работы блока анкетирования пользователя:  $f$  – ежедневно,  $g$  – 2-3 раза в неделю,  $h$  – редко;  $M$ ,  $N$  – параметры, определяемые по выражению 1 и 2 соответственно

Выводы о принадлежности пользователя к той или иной категории делаются на основании следующих исходных условий:

1) если пользователь использует компьютер редко (реже 1-2 раза в две недели), то он относится к категории „новичок“;

2) пользователь пользуется компьютером ежедневно для редактирования текстов (а), и/или отправки/получения электронной почты (б), и/или выполнения графических работ (с), и/или поиска в Internet (д), относится к категории «пользователь»:

$$M = (a = True) \vee (b = True) \vee (c = True) \vee (d = True); \quad (1)$$

3) если пользователь использует компьютер 2-3 раза в неделю и выполняет одновременно три работы из следующего набора: обработка текста, получение/отправка электронной почты, программирование (е), выполнение графических работ, поиск информации в Internet, он относится к категории «пользователь», в остальных случаях к категории «новичок». Указанные условия можно формально представить так:

$$N = O \vee P \vee Q \vee R \vee S \vee U \vee W \vee X \vee Y \vee Z, \quad (2)$$

где  $O = (a = True) \wedge (b = True) \wedge (e = True)$ ;  $P = (a = True) \wedge (b = True) \wedge (c = True)$ ;  
 $Q = (a = True) \wedge (b = True) \wedge (d = True)$ ;  $R = (a = True) \wedge (e = True) \wedge (c = True)$ ;  
 $S = (a = True) \wedge (e = True) \wedge (d = True)$ ;  $U = (a = True) \wedge (c = True) \wedge (d = True)$ ;  
 $W = (b = True) \wedge (e = True) \wedge (c = True)$ ;  $X = (b = True) \wedge (e = True) \wedge (d = True)$ ;  
 $Y = (b = True) \wedge (c = True) \wedge (d = True)$ ;  $Z = (e = True) \wedge (c = True) \wedge (d = True)$ ;

4) если пользователь использует компьютер ежедневно и среди прочих работ программирует, то он относится к категории «специалист».

В остальных случаях при ежедневном использовании он относится к категории „новичок“. Все другие блоки показаны на рис. 1, которые можно раскрыть аналогично.

### Заключение

Данный подход предлагается для использования в системах массового применения, так как он обеспечивает простой и удобный алгоритм взаимодействия пользователя с системой, позволяющий не только оценить профессиональный уровень владения компьютером, но и эмоциональное состояние пользователя, что немаловажно, если пользователь работает с системой автоматизации или управления, особенно когда его ошибки могут привести к катастрофическим последствиям. Применение этого интерфейса в системах обучения, в том числе профессиональных системах образования, позволяет объективно подойти к оценке знаний пользователя и повысить его профессиональный уровень. Все это достигается благодаря тому, что предлагаемый подход основан скорее на принципах адаптации системы к пользователю, чем пользователя к системе, что является принципиально новым. Если адаптация пользователя к системе осуществляется в основном за счет обучения пользователя работе с приложением, то адаптация системы к пользователю осуществляется за счет ограничения доступа пользователя к работе с приложением в зависимости от навыков владения компьютером и его эмоционального состояния; изменения стиля и способа взаимодействия в зависимости от поставленной задачи, изменение внешнего вида интерфейса согласно вкусам пользователя. Кроме того, следует подчеркнуть, что при построении такого интерфейса используется одна из прогрессивных технологий – многоагентная технология. Именно интерфейсные агенты, обладающие автономностью действий, возможностью автоматизировать задачу в соответствии с гибким набором правил, способны адаптироваться к уровню опыта, индивидуальности и технике пользователя.

**Список литературы:** 1. *Мирошник М.А.* Методы проектирования нечетких устройств принятия решений на основе программируемых логических интегральных микросхемах / М.А. Мирошник, Корольова Я.Ю. // *Технология приборостроения*. – 2009. – №2. – С. 16–23. 2. *Мирошник М.А.* Диагностическая инфраструктура с интеллектуальными свойствами в реконфигурируемых мультипроцессорных системах / М.А. Мирошник, Корольова Я.Ю. // *Тези доповідей міжнародної науково-практичної конференції «Інформаційні технології та комп'ютерна інженерія»*, 19-21 травня Вінниця. – 2010. – С. 44–46. 3. *Мирошник М.А.* Интеллектуальные системы обработки данных в телекоммуникационных сетях / М.А. Мирошник, Загарий Г.И. // *Інформаційно-керуючі системи на залізничному транспорті*. – Алушта, 2011. – №4. – С. 146. 4. *Мирошник М.А.* Концептуальная модель диагностической инфраструктуры с интеллектуальными свойствами для телекоммуникационных систем / М.А. Мирошник, Корольова Я.Ю. // *Інформаційно-керуючі системи на залізничному транспорті*. – Харків, 2011. – №5. – С.138. 5. *Мирошник М.А.* Подход к проектированию компьютерных сетей с интеллектуальной диагностической инфраструктурой / М.А. Мирошник, Карпенко С.Г., Ковалева М.А., Панченко С.В. // *Інформаційно-керуючі системи на залізничному транспорті*. – Харків, 2011. – №6. – С. 51 – 59. 6. *Мирошник М.А.* Применение ПЛИС как аппаратной базы при проектировании встраиваемых систем // М.А. Мирошник, Корольова Я.Ю. // *Технология приборостроения*. – 2011. – №2. – С. 51 – 59. 7. *Miroshnik M.A.* Application of software complex for query processing in the database management system with a view of dispatching problem solving in Grid systems / *Miroshnik M.A. Kotukh V.G., Selevko S.N.* // *Telecommunications and radio engineering*. – 2013. – Vol.27, № 10. – P. 875-891. 8. *Мирошник М.А.* Синтез распределенных компьютерных сред на базе компьютерных сетей // *Системи обробки інформації*. – 2013. – №7 (114). – С.86-89. 9. *Miroshnik M.A.* Uses of programmable logic integrated circuits for implementations of data encryption standard and its experimental linear cryptanalysis / *Miroshnik M.A. Kovalenko M.A.* // *Інформаційно-керуючі системи на залізничному транспорті*. – 2013. – №6. – С.36 – 45.

*Українська державна академія  
 залізничного транспорту,  
 Харківський національний університет  
 городського господарства імені А.Н. Бекетова,  
 Національний технічний університет «ХПИ»*

*Поступила в редколлегию 14.02.2015*