

2. *Informatsionno-analiticheskiy doklad «Teploenergetika i tsentralizovannoe teplosnabzhenie Rossii v 2015–2016 gody»* [Information and analytical report “Heat and Power and Centralized Heat Supply of Russia in 2015–2016”]. Ministerstvo energetiki Rossiyskoy Federatsii [Ministry of Energy of the Russian Federation]. Moscow, 2018.
3. Nurmeev B. K. *Metody regulirovaniya vybrosov v atmosferu produktov sgoraniya organicheskogo topliva ot statsionarnykh energeticheskikh istochnikov* [Methods for controlling emissions of fossil fuels from stationary energy sources into the atmosphere]. Saint Petersburg, 2006.
4. Pleshanov K. A. *Razrabotka i issledovanie sposoba szhiganiya topliv s umerennym kontroliruemym himicheskim nedozhogom* [Development and research of methods for burning fuel with a moderate controlled chemical burnout]. Moscow, 2010.
5. Roslyakov P. V., Ionkin I. L., Egorova L. E. *Kontroliruemyy khimicheskii nedozhog – effektivnyy metod snizheniya vybrosov oksidov azota* [Controlled chemical Incineration – an effective method to reduce nitrogen oxide emissions]. Available at: http://www.rosteplo.ru/Tech_stat/stat_shablon.php?id=2041 (accessed 26.05.2019).
6. Edeleva Olga A., Stennikov Valery A. Optimization of energy sources structure to minimize environment pollution. *E3S Web of conferences. Green Energy and Smart Grids*, 2018, no. 69, pp. 1–7.
7. Li J., Zhang X., Yang W. and Blasiak W. Effects of Flue Gas Internal Recirculation on NO_x and SO_x Emissions in a Co-Firing Boiler. *International Journal of Clean Coal and Energy*, 2013, vol. 2, no. 2, pp. 13–21. DOI: 10.4236/ijcce.2013.22002.
8. Nakomcic-Smaragdakis B., Z. Cepic, M. Cepic, T. Stajic. Data analysis of the flue gas emissions in the thermal-power plant firing fuel oil and natural gas. *Int. J. Environ. Sci. Technol.*, 2014, no. 11, pp. 269–280.
9. Roslyakov P. V., Ionkin I. L., and Pleshanov K. A. Reducing Harmful Emissions Discharged into the Atmosphere from Operating Boilers by Applying a Combination of Low-Cost Technological Measures. *Thermal Engineering*, 2010, vol. 57, no. 12, pp. 1052–1057.
10. Sudhakara Reddy A. V., Damodar Reddy M., Krishna Reddy Feeder Y. V. Reconfiguration of Distribution Systems for Loss Reduction and Emissions Reduction using MVO Algorithm. *Majlesi Journal of Electrical Engineering*, June 2018, vol. 12, no. 2, pp. 1–8.
11. Yousef S. H. Najjar. Gaseous Pollutants Formation and Their Harmful Effects on Health and Environment. *Ashdin Publishing Innovative Energy Policies*, 2011, vol. 1, Article ID E101203. 9 p. DOI:10.4303/iep/E101203.

УДК 004.822:338.482.22

DOI 10.21672/2074-1707.2019.47.3.017-027

ПЕРСОНАЛИЗАЦИЯ ИНТЕРФЕЙСОВ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ НА ОСНОВЕ ПАТТЕРНОВ ИНТЕРФЕЙСОВ ДЛЯ ЛЮДЕЙ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ*

Статья поступила в редакцию 26.08.2019, в окончательной версии – 02.09.2019.

Поцелуйко Анастасия Сергеевна, Волгоградский государственный технический университет, 400005, Российская Федерация, г. Волгоград, пр. Ленина, 28,
аспирант, e-mail: anastasiyasergeevnap@gmail.com

Кравец Алла Григорьевна, Волгоградский государственный технический университет, 400005, Российская Федерация, г. Волгоград, пр. Ленина, 28,
доктор технических наук, профессор, e-mail: agk@gde.ru

Кульцова Марина Борисовна, Волгоградский государственный технический университет, 400005, Российская Федерация, г. Волгоград, пр. Ленина, 28,
кандидат технических наук, доцент, e-mail: marina.kultsova@mail.com

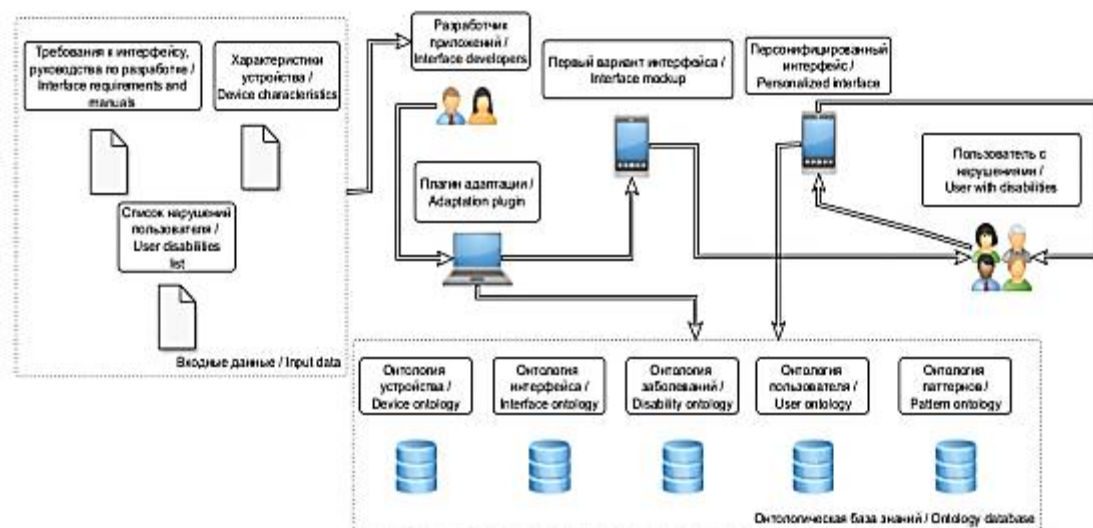
Статья посвящена проблеме персонализации интерфейса для людей с ограниченными возможностями. Пользователи с особыми потребностями испытывают неудобства при использовании приложений, которые могли бы помочь им преодолеть часть социальных барьеров в повседневной жизни. Процесс персонализации интерфейса для таких пользователей не может быть выполнен в один этап, так как необходимо учитывать информацию, собираемую при взаимодействии пользователя с мобильным приложением. Данная работа является продолжением исследований по разработке адаптивных пользовательских интерфейсов и посвящена исследованию существующих подходов к сбору и анализу информации о поведении пользователей и понятию контекстной информации, а также выработке рекомендаций по адаптации интерфейса. В статье описан механизм адаптации интерфейса на основе онтологического представления паттернов интерфейса, знаний (накопленной информации) о конечном пользователе и его поведе-

*Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 18-47-343001 «Разработка двухэтапного метода адаптации интерфейсов мобильных приложений для людей с ограниченными возможностями».

нии. Авторами был разработан набор правил адаптации интерфейса, который реализован в онтологической базе знаний. В статье описана модель онтологии, а также приведены некоторые примеры онтологического представления паттернов интерфейса.

Ключевые слова: ассистивные технологии, адаптивный пользовательский интерфейс, онтологическая модель пользователя, паттерны интерфейсов, поведение пользователей, пользователи с ограниченными возможностями

Графическая аннотация (Graphical annotation)



PERSONALIZATION OF MOBILE APPLICATION INTERFACES ON THE BASIS OF INTERFACE PATTERNS FOR PEOPLE WITH SPECIAL NEEDS

The article was received by the editorial board on 26.08.2019, in the final version – 02.09.2019.

Potseluko Anastasia S., Volgograd State Technical University, 28 Lenin Ave., Volgograd, 400005, Russian Federation,
postgraduate, e-mail: anastasiyasergeevnap@gmail.com

Kravets Alla G., Volgograd State Technical University, 28 Lenin Ave., Volgograd, 400005, Russian Federation,

Doct. Sci. (Engineering), Professor, e-mail: agk@gde.ru

Kultsova Marina B., Volgograd State Technical University, 28 Lenin Ave., Volgograd, 400005, Russian Federation,

Cand. Sci. (Engineering), Associate Professor, e-mail: marina.kultsova@mail.com

This article is devoted to the problem of personalization of the interface for people with disabilities. Users with special needs do not feel comfortable using applications that could help them to overcome some of the social barriers in everyday life. The process of personalizing the interface for specific users cannot be performed in one step; we suggest to take into account the information which is collected during user interaction with a mobile application. This work is a continuation of the research on the development of adaptive user interfaces. Much attention was paid to the study of existing approaches to the collection and analysis of information about user behavior and the concept of contextual information, as well as recommendations for adapting the interface. An improved interface adaptation mechanism was developed and described based on the ontological representation of interface patterns and knowledge of the end user and his behavior. A set of adaptation rules was developed and implemented in the ontology knowledge base. In the article, we described an improved ontology model and some examples of the ontological representation of interface patterns.

Keywords: assistive technologies, adaptive user interface, ontological user model, interface patterns, user behavior, users with special needs

Введение. Проблема социализации людей с особыми потребностями играет ключевую роль в современном мире, и особенно в сфере информационных технологий. По данным статистики организации WorldBank [17], каждый пятый человек в мире имеет некоторую степень инвалидности. Качество жизни многих людей зависит от удобства использования и доступности веб-сайтов, мобильных и настольных приложений, электронных устройств.

Для решения данной проблемы организация W3C создала руководство [19], содержащее набор рекомендаций по разработке интерфейсов для людей с ограниченными возможностями. Однако проблема остается нерешенной до сих пор, так как появляются новые устройства, типы приложений и новые типы взаимодействия человека с компьютером. Поэтому разработчики программного обеспечения стараются сделать все приложения более удобными для всех категорий пользователей, а их интерфейсы – более адаптивными и персонализированными. Самая распространенная проблема адаптации приложений для мобильных пользователей – проблема с соотношением размеров дисплеев и отображаемой информации на них, например, изображения или текст могут быть чересчур мелкими для лиц с нарушениями зрения.

Крупные IT-организации, такие как Google [9], Chrome, Android и YouTube, уже внедрили функционал для людей с ограниченными возможностями в свои продукты. А такие ведущие компании по разработке программного обеспечения, как Microsoft [15], Apple [6] и Android [4], разработали свои собственные рекомендации в отношении того, как сделать приложения более доступными. Кроме того, каждая из этих компаний предоставила в своих программных продуктах возможность персонализировать интерфейс. Например, выбрать альтернативу аудиофайлам, выполнять перевод голоса в текст, изменять контрастность цветовой гаммы приложений и т.д.

Вопросы адаптации интерфейсов для лиц с ограниченными возможностями по зрению решаются и создателями значительной доли российских и зарубежных сайтов – чаще всего с использованием специальных кнопок перенастройки интерфейса для слабовидящих людей [19].

К сожалению, не все разработчики программного обеспечения следуют рекомендациям при создании программных продуктов. Поэтому основная цель нашего проекта – облегчить разработку приложений для людей с ограниченными возможностями путем реализации набора соответствующих рекомендаций непосредственно в программной среде разработки.

Современное состояние проблемы персонализации интерфейсов. Лидеры в сфере разработки программного обеспечения в настоящее время предоставляют широкий набор инструментов для разработчиков интерфейсов, позволяющих создавать доступные приложения для пользователей с ограниченными возможностями.

Компания Apple включила собственные инструменты для обеспечения доступности приложений непосредственно в операционную систему, причем все настройки доступности приложений могут быть изменены через стандартные контекстные меню. Для разработчиков программных решений под платформу iOS Apple предлагает свое инструментальное средство UIKit [5] для улучшения доступности интерфейсов приложений. Основная цель UIKit – обеспечить возможность персонализировать интерфейс за счет настроек для людей с ограниченными возможностями. На данный момент поддерживаются следующие виды ограничений: нарушения зрения (слепота, дальтонизм, проблемы с фокусировкой), нарушения слуха (глухота, частичная потеря слуха), физические нарушения и нарушения моторики (трудности с удержанием устройства, трудности с нажатием на экран или на кнопки), психические нарушения (проблемы с запоминанием этапов выполнения задачи, с восприятием самой задачи).

Помимо UIKit iOS предоставляет и другие средства для улучшения доступности приложений, например, VoiceOver, SwitchControl и AssistiveTouch. Все эти инструменты могут быть интегрированы в разрабатываемые приложения. В дополнение к этому iOS предоставляет возможность разработчикам использовать в своих приложениях классы и интерфейсы, отвечающие за доступность приложений. Например, класс UI Accessibility предоставляет информацию обо всех элементах пользовательского интерфейса.

Создатели операционной системы Android также разрабатывают программные средства для обеспечения доступности приложений. Настройки доступности встроены в операционную систему, начиная с Android 6.0. Эти настройки включают в себя возможность увеличивать элементы интерфейса, изменять размер шрифта, менять контрастность и включать подписи к элементам интерфейса. Последняя версия операционной системы включает в себя Android Accessibility Suite, который состоит из следующих компонентов:

1. TalkBack – озвучивает действия пользователя и предупреждает его о поступивших оповещениях или уведомлениях.
2. SelecttoSpeak – позволяет озвучить выбранный на экране текст.
3. VoiceAccess – позволяет пользователю управлять устройством с помощью голосовых команд.
4. BrailleBack – позволяет пользователю подключить дисплей Брайля к устройству через Bluetooth.
5. SoundAmplifier – позволяет фильтровать и усиливать звук в окружающей среде с использованием наушников.

Так же, как и для iOS, разработчики Android предоставляют специализированные классы для создания доступных приложений, которые содержат методы генерации событий и реагирования на них для элементов интерфейса. Чтобы быть уверенным, что приложение может использоваться людьми с ограниченными возможностями, Android создал утилиту AccessibilityScanner, которая собирает такую информацию, как дублирующиеся ссылки, недоступные ссылки, элементы без текстового сопровождения, дублирующиеся описания элементов, коэффициент контрастности для объектов менее 3.0 и многое другое.

Компания Microsoft располагает большим набором инструментов, поддерживающих разработку приложений для людей с различными нарушениями здоровья. На данный момент разработаны программные средства для поддержки людей с нарушением зрения, слуха, нарушениями работы нервной системы, физическими отклонениями и психическими нарушениями.

Microsoft предоставляет исчерпывающую документацию [15] об обеспечении доступности веб-приложений и настольных программ. Кроме того, имеется отдельный обзор общих принципов обеспечения доступности с подробным описанием. Помимо специализированных классов для разработчиков компания предоставляет новую технологию разработки доступных приложений – MicrosoftActiveAccessibility. Она основана на COM (ComponentObjectModel), что позволяет улучшить доступность приложений, написанных под операционную систему Windows. Эта технология включает в себя надежные методы предоставления информации об элементах интерфейса. Основным СОМ-интерфейсом технологии является IAccessible, который предоставляет информацию об элементах графического интерфейса пользователя. Наконец, Microsoft разработала инструмент для тестирования интерфейсов на доступность, который называется UIAccessibilityChecker. Он проверяет соответствие основных требований доступности в приложении.

Проанализировав методы обеспечения доступности, предлагаемые ведущими компаниями по разработке программных продуктов, мы можем утверждать, что сейчас доступно множество методов и инструментов для обеспечения доступности приложений. Однако необходимость их использования оставлена на усмотрение разработчиков и напрямую зависит от их опыта, квалификации и навыков, мотивации к обеспечению удобства работы пользователей с ограниченными возможностями. Важнейшим моментом в разработке доступных приложений для таких людей [7] являются вопросы адаптации и персонализации пользовательского интерфейса. Можно выделить два основных подхода к адаптации интерфейса: онтологический подход (проекты AEGIS [8] и EGOKI [16]) и подход, основанный на паттернах (MyUIProject [7]). Авторы подробно проанализировали эти подходы и соответствующие программные инструменты в работах [10, 11, 12, 13].

Несмотря на разнообразие существующих технологий для разработки доступных приложений, не существует единого универсального инструмента для создания адаптивных персонализированных интерфейсов для пользователей с ограниченными возможностями. Поэтому целью нашего проекта является создание специального инструмента InterfaceEditor, который позволяет разрабатывать адаптивные интерфейсы для кроссплатформенных доступных приложений. Чтобы обеспечить адаптивность и персонализацию интерфейса, авторы объединили подходы на основе онтологий и паттернов, что позволило реализовать этот комбинированный подход в первой версии InterfaceEditor [10].

Тестирование интерфейсов, разработанных с помощью указанного редактора интерфейсов, в группах пользователей с ограниченными возможностями выявило ряд проблем. Они были вызваны недостаточным учетом изменения контекста пользователя во время его взаимодействия с интерфейсом. Под контекстом пользователя понимается информация о его действиях в приложении, характеристиках устройства, изменениях в нарушениях здоровья. Данная статья описывает процесс разработки интерфейса и онтологической модели, которые учитывают информацию о поведении пользователя и изменение контекста пользователя.

Правила адаптации и рекомендации. В нашем проекте мы реализовали наиболее важные рекомендации по обеспечению доступности интерфейса в соответствии с W3C [19]. Основные принципы этого руководства определяют понятный, функциональный, легкий в восприятии и надежный интерфейс.

Понятный интерфейс должен соответствовать следующим требованиям.

1. Все мультимедийные данные, представленные пользователю, должны иметь текстовую альтернативу.
2. Если у интерфейса есть виджеты (элементы интерфейса) для ввода текста, то они должны сопровождаться кратким и ясным текстовым описанием.
3. Если невозможно использовать текстовую альтернативу некоторым элементам интерфейса, то текст виджета должен содержать как минимум описательную информацию об этих элементах.

4. Цвета не должны играть ключевую роль во взаимодействии пользователя с интерфейсом.
5. Пользователь должен иметь возможность изменять размер шрифта, цвет и стиль текста, а также цвет фона.

6. Коэффициент контрастности [18] между элементами интерфейса и их фоном должен быть не менее 7:1.

7. Размер текста может быть при необходимости увеличен пользователем до 200 %.

Функциональный интерфейс должен соответствовать следующим требованиям.

1. Интерфейс не должен содержать анимацию, которая может спровоцировать эпилептические припадки у пользователей.

2. Все элементы интерфейса должны быть доступны с помощью клавиатуры, т.е. пользователь может при необходимости работать с приложением и с помощью периферийных устройств.

3. Пользователь должен иметь возможность останавливать, приостанавливать или перезапускать мультимедийный контент.

4. Приложение должно уведомлять пользователя о любых неожиданных сбоях, предлагать сохранить данные или завершить все текущие задачи.

5. Компоненты интерфейса получают фокус в логическом порядке.

Легкий в восприятии интерфейс должен соответствовать следующим требованиям.

1. Язык интерфейса по умолчанию должен определяться приложением, однако у пользователя должна быть возможность изменить его.

2. Интерфейс не должен изменять контекст, если произошло какое-либо событие (нажатие кнопки, изменение текста для виджетов ввода).

3. Контекстная информация пользователя может быть изменена только пользователем.

4. Процесс навигации должен быть одинаков во всем приложении.

5. Единообразные компоненты интерфейса должны иметь одинаковый функционал.

6. Приложение должно предоставлять пользователю контекстно-зависимую помощь.

7. Приложение должно предупреждать пользователя о возникновении ошибки и предлагать способ ее устранения.

8. Приложение должно проверять данные, которые вводит пользователь.

Надежный интерфейс должен соответствовать основному условию – быть совместимым со всеми текущими и будущими версиями пользовательских устройств, в том числе со вспомогательными периферийными устройствами ввода и вывода.

Большинство приведенных рекомендаций уже было реализовано в первой версии редактора интерфейсов для Unity3D [10] как часть меню настроек. Например, текстовая альтернатива мультимедийному контенту, настройки шрифта и фона приложения, уровень контраста, отключение мерцающего контента и некоторые другие.

На данный момент ведется работа над расширением функциональных возможностей редактора. В новой версии будут реализованы все рекомендации, упомянутые выше. Отличие предлагаемого подхода к адаптации интерфейса от всех существующих заключается в том, что разрабатываемое программное средство включает в себя все варианты рекомендаций, представленных в W3C [19]. Кроме того, разрабатываемый плагин включает в себя базу знаний о заболеваниях пользователей, а также о том, какие ограничения на использование мобильных приложений приносит то или иное нарушение. Такая двойная адаптация позволяет максимально точно подобрать те элементы интерфейса и их характеристики, которые не будут доставлять пользователю неудобства в процессе использования мобильного приложения.

После тестирования первой версии редактора было решено включить паттерны интерфейса в онтологическую модель для обеспечения целостности знаний и упрощения расширения базы паттернов.

Правила адаптации и рекомендации. Для учета контекстной информации о пользователе и его поведении в процессе использования приложения был изменен процесс разработки интерфейсов, представленный в работе [10]. Модернизированный двухэтапный процесс разработки адаптивных персонализированных интерфейсов представлен на рисунке 1.

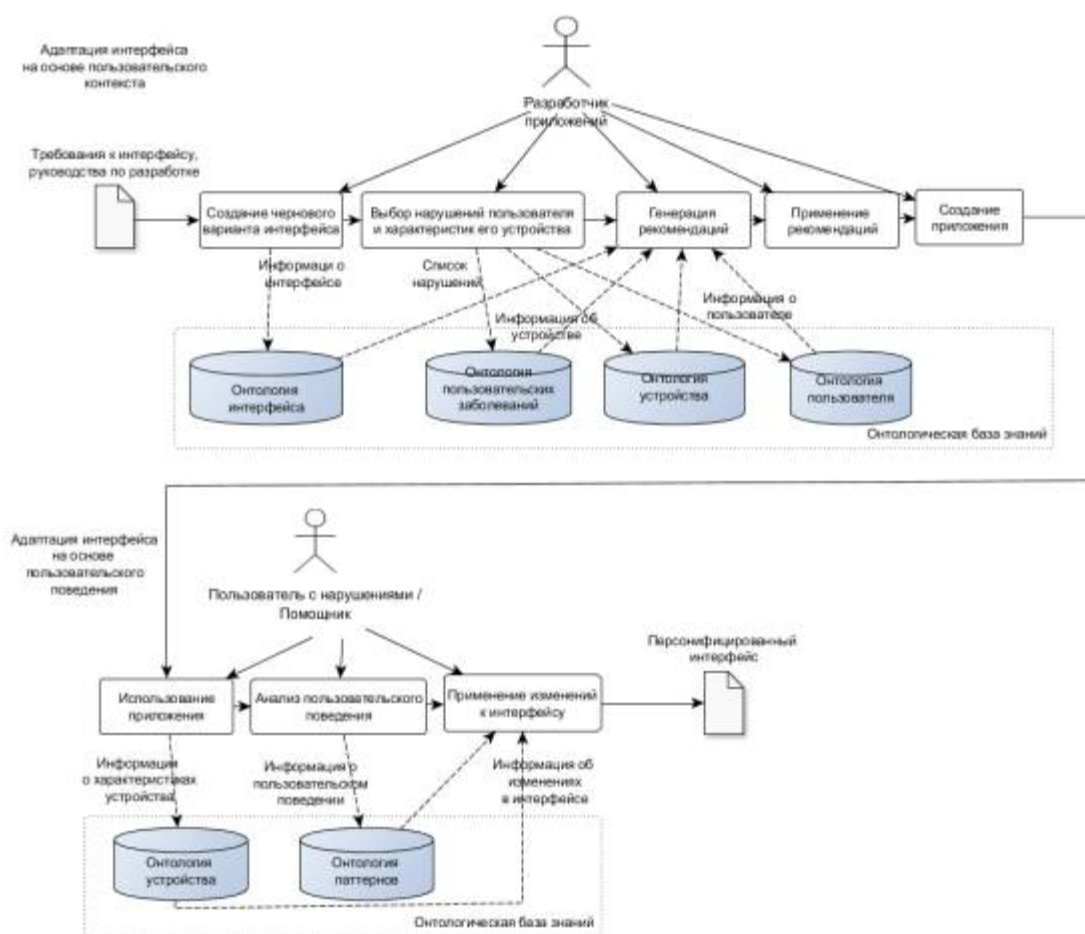


Рисунок 1 – Двухэтапный процесс адаптации интерфейсов

Два главных этапа алгоритма представлены на разных уровнях диаграммы. Первый этап выполняется разработчиком программного обеспечения и включает в себя создание прототипа интерфейса и адаптацию интерфейса в соответствии с пользовательским контекстом. Промежуточный шаг по генерации рекомендаций и проверке интерфейса на соответствие правилам выполняет система. Второй этап процесса затрагивает непосредственно пользователей приложения и состоит из использования интерфейсов и их персонализации на основе информации о пользовательском поведении и изменении контекста.

Для генерации рекомендаций на первом этапе используются экспертные знания о заболеваниях пользователя, а также об устройствах пользователя, хранящиеся в онтологической базе знаний. На втором этапе используется онтологическая модель паттернов интерфейсов и устройств пользователя. Онтология паттернов содержит знания о возможных изменениях элементов интерфейсов (по рекомендациям W3C) в соответствии с изменениями пользовательского поведения, вызванными изменением контекстной информации.

Например, пользователь имеет проблемы со зрением средней тяжести. Интерфейс приложения разработан под данный вид нарушений в редакторе интерфейсов. После использования приложения в течение некоторого времени проблемы со зрением пользователя усугубились, что привело к изменению контекста. Возможна ситуация, при которой пользователь будет не способен нажимать кнопки и читать текст с дисплея. Чтобы настроить интерфейс под изменившийся контекст, пользователь или его помощник может изменить настройки интерфейса в приложении, что приведет к изменению соответствующих элементов интерфейса.

Для реализации такого процесса необходимо было пересмотреть онтологическую модель знаний, разработать онтологическую модель паттернов интерфейса и создать набор паттернов интерфейса.

Онтологическая модель предметной области. В соответствии с описанным процессом разработки интерфейса была переработана метаонтология предметной области (рис. 2) с целью включения онтологической модели паттернов интерфейса.

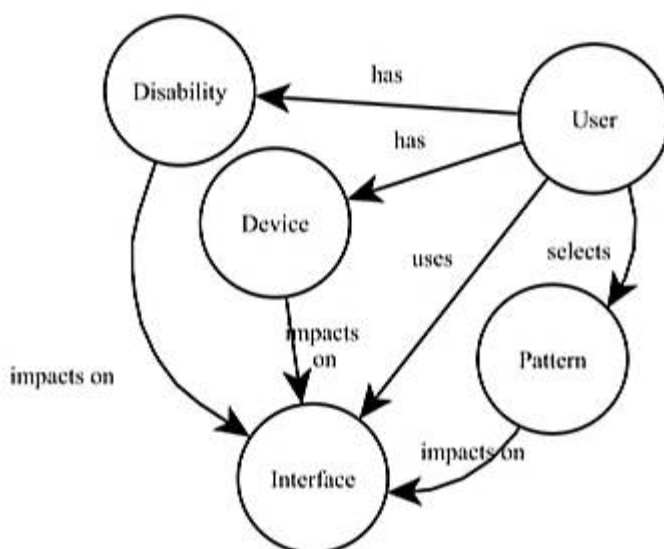


Рисунок 2 – Метаонтология предметной области

Данную модель можно представить следующим образом:

$$M = \langle O_M, C, Inst, R, I \rangle,$$

где M – метаонтология;

$O_M = \{O_{User}, O_{Dis}, O_{Int}, O_{Dev}, O_{Pat}\}$ – набор онтологий областей, O_{User} – онтология пользователя, O_{Dis} – онтология заболевания пользователя, O_{Int} – онтология интерфейса, O_{Dev} – онтология устройства пользователя, O_{Pat} – онтология паттернов интерфейса;

C – конечный набор концептов метаонтологии, $C = \emptyset$;

$Inst$ – конечный набор сущностей мета-онтологии, $Inst = \emptyset$; $R = \{has, impacts_on, uses, selects\}$ – конечный набор отношений между компонентами метаонтологии;

I – конечный набор правил интерпретации, $I = \emptyset$.

Онтологическая модель паттернов интерфейса связана с онтологией пользователя, поскольку пользователь может напрямую персонализировать свой интерфейс через меню настроек. Модель паттернов была реализована в первой версии редактора интерфейса [11], однако она не была включена в онтологическую базу знаний. Главный недостаток такого подхода – контекст мог меняться независимо от поведения пользователя, кроме того, условия замены одного виджета другим не были включены ни в какие стандарты разработки интерфейса.

Основное преимущество предложенной модифицированной модели онтологии (рис. 2) состоит в том, что модели интерфейса и паттернов интерфейса были разделены. Поэтому контекст теперь может меняться в соответствии с поведением пользователя. Все шаблоны, включенные в онтологию паттернов, соответствуют рекомендациям W3C. Кроме того, предложенная онтологическая модель может быть легко расширена путем добавления новых паттернов интерфейса.

Онтология паттернов интерфейса представлена на рисунке 3 и может быть описана следующим образом:

$$O_{Pat} = \langle C_{Pat}, Inst_{Pat}, R_{Pat}, I_{Pat} \rangle,$$

где C_{Pat} – конечный набор концептов онтологии области заболеваний;

$Inst_{Pat}$ – конечный набор сущностей онтологии заболеваний, $Inst = \emptyset$;

$R_{Pat} = \{isa, impacts_on\}$ – конечный набор отношений между компонентами онтологии заболеваний;

I_{Pat} – конечный набор правил интерпретации, $I_{Dis} = \emptyset$.

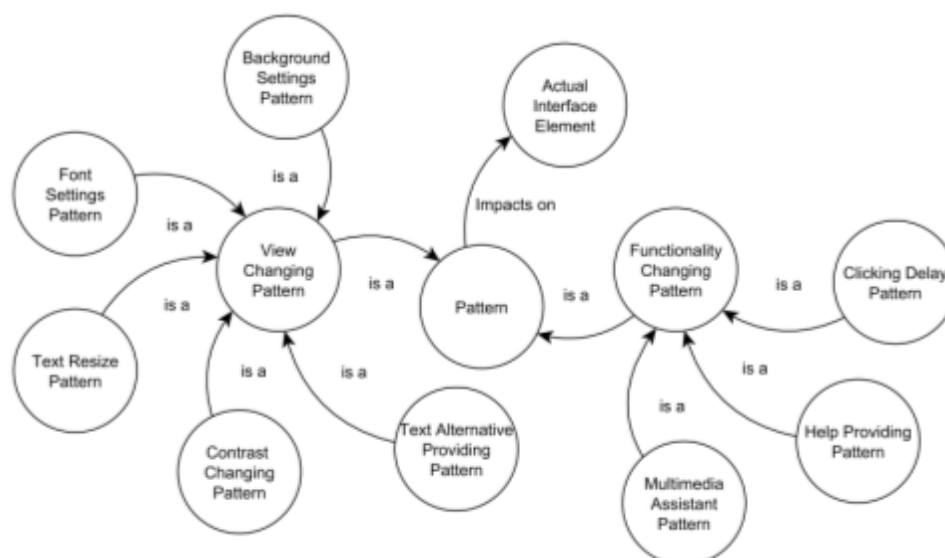


Рисунок 3 – Онтологическая модель паттернов интерфейса

Все паттерны интерфейса разделены на две группы: паттерны, которые изменяют внешний вид элемента интерфейса, и паттерны, которые изменяют функциональность элемента интерфейса. Каждый паттерн связан с фактическим элементом интерфейса и имеет доступ к его свойствам. Все паттерны в онтологии были представлены с помощью Jena-правил [3]. Примеры паттернов изменения шрифта представлены в таблице 1.

Таблица 1 – Паттерн изменения размера шрифта

Название	Паттерн изменения размера шрифта
Проблема	Пользователь может испытывать затруднения с чтением или вводом текста установленного размера
Решение	Позволить увеличивать размер шрифта до 200 %
Реализация	Получить каждый текстовый компонент каждого элемента интерфейса и увеличить размер шрифта

Онтологическое представление этого паттерна показано на рисунке 4: интерфейс содержит два текстовых элемента интерфейса. Каждый текстовый элемент имеет текущий размер шрифта. Каждый экземпляр текстового элемента отслеживается с помощью паттерна изменения размера шрифта текста. Он содержит текущий размер шрифта, а также два ограничения на размер шрифта: размер шрифта не может быть меньше 8 и больше 32 пунктов в соответствии с решением из таблицы 1.

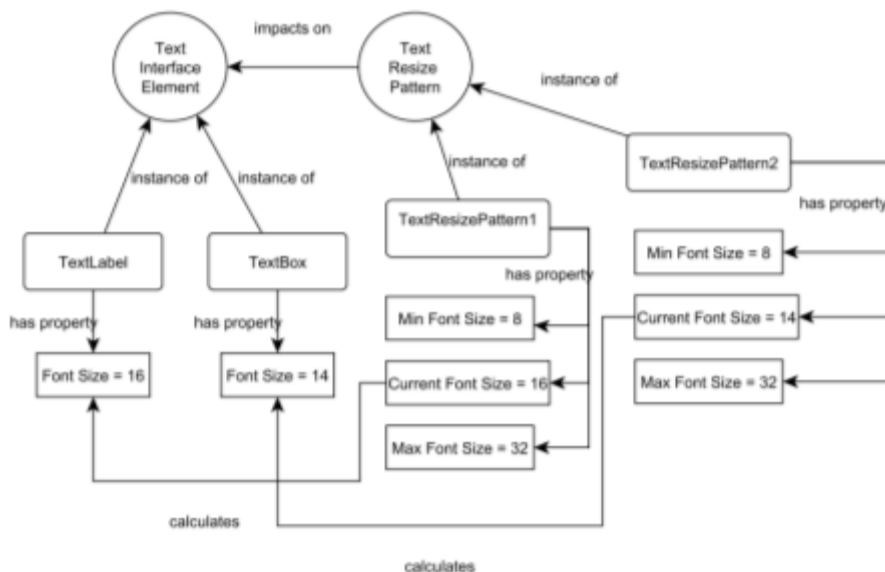


Рисунок 4 – Онтологическое представление паттерна изменения размера шрифта

Соответствующее Jena-правило представлено ниже.

User:hasNecessityToIncreaseText (?user, VisionDisability) ^ User:hasInterface (?user, ?interface) ^ Interface:hasElement(?interface, ?element) ^ Element:type(?element, ActualTextContentElement) → Element (?element, hasTextSize ?newTextSizeValue),

где ?user, ?interface, ?element, ?newTextSizeValue – переменные Jena;

hasNecessityToIncreaseText, hasInterface, hasElement, hasTextSize – отношения между концептами онтологической модели;

VisionDisability, ActualTextContentElement – сущность онтологической модели интерфейса.

Примеры паттернов помощи работы с мультимедиа представлены в таблице 2.

Таблица 2 – Паттерн помощи работы с мультимедиа

Название	Паттерн помощи работы с мультимедиа
Проблема	Пользователь хочет воспроизвести, приостановить или остановить воспроизведение мультимедийного контента
Решение	Добавить кнопки «Стоп», «Перезапуск» и «Воспроизведение» всем мультимедийным элементам интерфейса
Реализация	Получить все мультимедийные элементы интерфейса и активировать панель с кнопками управления содержимым элементов

Онтологическое представление паттерна из таблицы 2 показано на рисунке 5: интерфейс содержит два элемента мультимедийного типа: видео- и аудиоэлемент. Каждый виджет имеет свойство для активации панели с кнопками «Стоп», «Перезапуск» и «Воспроизведение». Каждый экземпляр мультимедийного виджета отслеживается и может быть изменен с помощью паттерна.

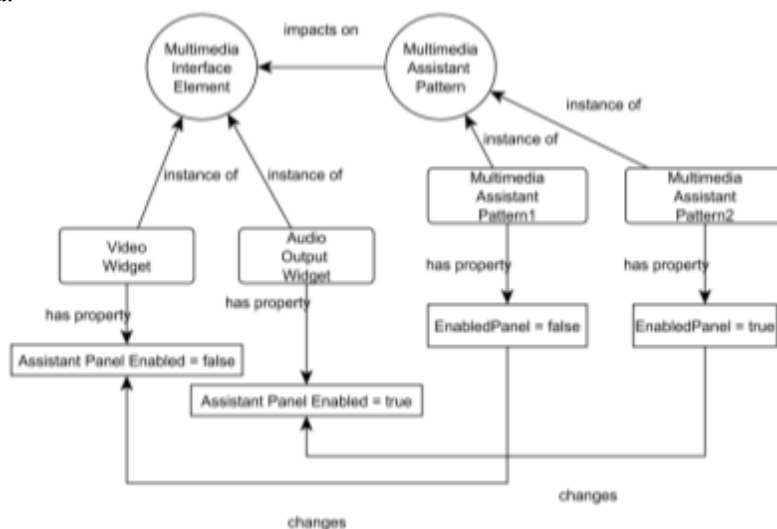


Рисунок 5 – Паттерн помощи работы с мультимедиа

Соответствующее Jena-правило представлено ниже.

User:hasInterface (?user, ?interface) ^ Interface: hasElement(?interface, ?element) ^ Element:type(?element, ActualMultimediaContentElement) → Element (?element, hasMultimediaControlButtons?buttons),

где ?user, ?interface, ?element, ?buttons – переменные Jena;

hasInterface, hasElement, hasMultimediaControlButtons – отношения между концептами онтологической модели;

ActualTextContentElement – сущность онтологической модели интерфейса.

На данный момент база паттернов интерфейса находится на стадии внедрения. Использование онтологической модели паттернов интерфейса дало ряд преимуществ, таких как простота расширения, поддержки и повторного использования онтологической модели. К настоящему моменту онтологическая база содержит реализацию 6 паттернов в виде 32 правил. Эти правила охватывают нарушения зрения и частично нарушения моторики движений пользователя. Часть онтологической базы, предназначенная для поддержки компенсации ментальных нарушений, нарушений слуха и физических нарушений будет реализована в дальнейшем.

Закключение. В данной статье проанализированы существующие подходы к разработке адаптивных пользовательских интерфейсов для людей с ограниченными возможностями. Описан модернизированный процесс разработки интерфейса, включающий в себя этап персонализации интерфейса на основе информации о поведении пользователя при взаимодействии с интерфейсом. Также в статье приведена онтологическая модель предметной области, включающая в себя онтологию паттернов интерфейса.

На данный момент авторы работают над расширением базы паттернов интерфейсов, чтобы охватить большинство возможных ситуаций, которые могут привести к изменению пользовательского контекста при взаимодействии пользователя с интерфейсом.

Для дальнейшей работы планируется создать специальную процедуру для разрешения конфликтных ситуаций, которые могут возникнуть во время персонализации интерфейса. Приложение JAVA для работы с онтологией в первой версии InterfaceEditor должно быть заменено библиотекой dll, что сделает приложение более стабильным, унифицированным и более простым для тестирования.

В конечном счете будет улучшен и дополнен двухэтапный метод адаптации интерфейсов с использованием онтологической модели предметной области, который будет включать в себя полный набор правил для вывода решений на основе списка заболеваний пользователя, характеристик его устройства, а также информации о поведении пользователя в течение процесса взаимодействия с приложением. Данный улучшенный метод ляжет в основу второй версии плагина адаптации интерфейсов для Unity 3D.

Библиографический список

1. Брумштейн Ю. М. Анализ структуры сайтов академий наук некоторых зарубежных стран / Ю. М. Брумштейн, Н. В. Васильев // Научно-техническая информация. Сер. 2. Информационные процессы и системы. – 2019. – № 1. – С. 9–24.
2. AEGIS Ontology. – Режим доступа: <http://www.aegis-project.eu>, свободный. – Заглавие с экрана. – Яз. англ. (дата обращения: 21.06.19).
3. Apache Jena official website, Reasoners and rule engines: Jena inference support. – Режим доступа: <https://jena.apache.org/documentation/inference>, свободный. – Заглавие с экрана. – Яз. рус. (дата обращения: 01.09.19).
4. Android Developers official website. – Режим доступа: <http://developer.android.com/guide/topics/ui/accessibility>, свободный. – Заглавие с экрана. – Яз. англ. (дата обращения: 30.06.19).
5. Apple official developer website, Apple Inc. – Режим доступа: <http://developer.apple.com/documentation/uikit/accessibility/>, свободный. – Заглавие с экрана. – Яз. англ. (дата обращения: 30.06.19).
6. Apple official website, Apple Inc. – Режим доступа: <http://support.apple.com/accessibility/>, свободный. – Заглавие с экрана. – Яз. англ. (дата обращения: 30.06.19).
7. Elias M. Towards an Ontology-based Representation of Accessibility Profiles for Learners / M. Elias, S. Lohmann, S. Auer // Proceedings of the Second International Workshop on Educational Knowledge Management. – November, 2016. – Vol. 1780.
8. Gamecho B. Automatic Generation of Tailored Accessible User Interfaces for Ubiquitous Services / B. Gamecho, R. Minon, A. Aizpurua, I. Cearreta, M. Arrue, N. Garay-Vitoria, J. Abaskal // IEEE Transactions on Human-Machine Systems. – October, 2015. – Vol. 45, № 5. – P. 612–623.
9. Google, Web. – Режим доступа: <http://www.google.com/intl/en/accessibility/>, свободный. – Заглавие с экрана. – Яз. англ. (дата обращения: 30.06.19).
10. Kultsova M. A Two-phase Method of User Interface Adaptation for People with Special Needs / M. Kultsova, I. Zhukova, A. Potselico, A. Skorikov, R. Romanenko // Proceedings of the CITDS2017Conference. Volgograd, Russia, September, 2017. – Volgograd, 2017.
11. Kultsova M. An Ontology-based Adaptation of User Interface for People with Special Needs / M. Kultsova, R. Romanenko, A. Anikin, A. Podelico // Proceedings of the AINL FRUCT 2016Conference. Saint-Petersburg, Russia, November, 2016. – Saint-Petersburg, 2016.
12. Kultsova M. An Ontology-based Approach to Automated Generation of Adaptive User Interface Based on User Modeling / M. Kultsova, R. Romanenko, A. Anikin, A. Podelico // Proceedings of the IISA2016 Conference. Khalkidiki, Greece, July, 2016. – Khalkidiki, 2016.
13. Kultsova M. Assistive mobile application for support of mobility and communication of people with IDD / M. Kultsova, R. Romanenko, I. Zhukova, A. Usov, N. Penskov, T. Potapova // Proceedings of the MobileHCI 16 Conference. Florence, Italy, July, 2016. – Florence, 2016.
14. Lempert L. Development of the intellectual decision-making support method for medical diagnostics in psychiatric practice / L. Lempert, A. Kravets, B. Lempert, O. Poplavskaya, N. Salnikova // Proceedings of the IISA 2018Conference. Zakynthos, Greece, July, 2018. – Zakynthos, 2018.
15. Microsoft official website. – Режим доступа: <http://www.microsoft.com/en-us/accessibility>, свободный. – Заглавие с экрана. – Яз. англ. (дата обращения: 30.06.19).
16. Peissner M. A Design Patterns Approach to Adaptive User Interfaces for Users with Special Needs

/ M. Peissner, A. Shuller, D. Spath // Proceedings of the HCII Conference. – July, 2011. – P. 268–277.

17. The World Bank official website. – Режим доступа: <http://www.worldbank.org/en/topic#1>, свободный. – Заглавие с экрана. – Яз. англ. (дата обращения: 24.06.19).

18. UXPub official website, Ух дизайн, цвет. – Режим доступа: <https://ux.pub/cvetovoj-kontrast-v-dizajne-interfejsovp-ostandartam/>, свободный. – Заглавие с экрана. – Яз. рус. (дата обращения: 01.09.19).

19. W3C official website, World Wide Web Consortium. – Режим доступа: <http://www.w3.org/standards/webdesign/accessibility/>, свободный. – Заглавие с экрана. – Яз. англ. (дата обращения: 24.06.19).

References

1. Brumshteyn Yu. M., Vasilyev N. V. Analiz struktury saytov akademiya nauk nekotorykh zarubezhnykh stran [Structural analysis of websites of Academies of Sciences of some foreign countries]. *Nauchno-tehnicheskaya informatsiya. Series 2. Informatsionnye protsessy i sistemy* [Scientific and technical information. Ser. 2. Information processes and systems], 2019, vol. 53, no. 1, pp. 16–31.

2. AEGIS Ontology. Available at: <http://www.aegis-project.eu> (accessed 21.06.19).

3. Apache Jena official website, Reasoners and rule engines: Jena inference support. Available at: <https://jena.apache.org/documentation/inference> (accessed: 01.09.19).

4. Android Developers official website. Available at: <http://developer.android.com/guide/topics/ui/accessibility> (accessed 30.06.19).

5. Apple official developer website, Apple Inc. Available at: <http://developer.apple.com/documentation/uikit/accessibility/> (accessed 30.06.19).

6. Apple official website, Apple Inc. Available at: <http://support.apple.com/accessibility/> (accessed 30.06.19).

7. Elias M., Lohmann S., Auer S. Towards an Ontology-based Representation of Accessibility Profiles for Learners. *Proceedings of the Second International Workshop on Educational Knowledge Management*, November, 2016, vol. 1780.

8. Gamecho B., Minon R., Aizpurua A., Cearreta I., Arrue M., Garay-Vitoria N., Abaskal J. Automatic Generation of Tailored Accessible User Interfaces for Ubiquitous Services. *IEEE Transactions on Human-Machine Systems*, October, 2015, vol. 45, no. 5, pp. 612–623.

9. Google, Web. Available at: <http://www.google.com/intl/en/accessibility/> (accessed 30.06.19).

10. Kultsova M., Zhukova I., Potselucio A., Skorikov A., Romanenko R. A Two-phase Method of User Interface Adaptation for People with Special Needs. *Proceedings of the CITDS2017Conference*. Volgograd, Russia, September, 2017. Volgograd, 2017.

11. Kultsova M., Romanenko R., Anikin A., Podeluico A. An Ontology-based Adaptation of User Interface for People with Special Needs. *Proceedings of the AINL FRUCT 2016Conference*. Saint-Petersburg, Russia, November, 2016. Saint-Petersburg, 2016.

12. Kultsova M., Romanenko R., Anikin A., Podeluico A. An Ontology-based Approach to Automated Generation of Adaptive User Interface Based on User Modeling. *Proceedings of the IISA2016 Conference*. Khalkidiki, Greece, July, 2016. Khalkidiki, 2016.

13. Kultsova M., Romanenko R., Zhukova I., Usov A., Penskoj N., Potapova T. Assistive mobile application for support of mobility and communication of people with IDD. *Proceedings of the MobileHCT 16 Conference*, Florence, Italy, July, 2016. Florence, 2016.

14. Lempert L., Kravets A., Lempert B., Poplavskaya O., Salnikova N. Development of the intellectual decision-making support method for medical diagnostics in psychiatric practice. *Proceedings of the IISA 2018 Conference*. Zakynthos, Greece, July, 2018. Zakynthos, 2018.

15. Microsoft official website. Available at: <http://www.microsoft.com/en-us/accessibility> (accessed 30.06.19).

16. Peissner M., Shuller A., Spath D. A Design Patterns Approach to Adaptive User Interfaces for Users with Special Needs. *Proceedings of the HCII Conference*, July, 2011, pp. 268–277.

17. The World Bank official website. Available at: <http://www.worldbank.org/en/topic#1> (accessed 24.06.19).

18. UXPub official website. Available at: <https://ux.pub/cvetovoj-kontrast-v-dizajne-interfejsovp-ostandartam/> (accessed 01.09.19).

19. W3C official website, World Wide Web Consortium. Available at: <http://www.w3.org/standards/webdesign/accessibility/> (accessed 24.06.19).