

ДОДАТОК А

Перелік посилань відповідно до наукових досліджень кафедри

2. Несміян Ю.Ю., Широкопетлева М.С. Про підхід до створення моделі користувача в адаптивній системі навчання /«Східно Європейський журнал передових технологій», 2010, с. 51-54.

12. The methods of adaptation in computer-based training systems Shubin, I., Karmanenko, O., Gorbach, T., Umyarov, K. 2015 Information Technologies in Innovation Business Conference, ITIB 2015 - Proceedings, 2015, с. 64-67, 7355054

13. Шубин И. Ю. Информационные технологии моделирования адаптивных систем обучения // Информационные системы и технологии : материалы 3-й Междунар. науч.- техн. конф. ИСТ-2014, 15-21 сент. 2014 г., Харьков : тез. докл. – Х. : НТМТ, 2014. – С. 93-94.

14. Mining methods for adaptation metrics in e-learning Shubin, I. , Skovorodnikova, V. , Kozyriev, A. , Pitiukova, M. CEUR Workshop Proceedings, 2019, 2362

21. Бондаренко М. Ф. Комп'ютерна дискретна математика: підручник / Бондаренко М. Ф., Білоус Н. В., Руткас А. Г. –Харків, «Компанія СМІТ», 2004 - 485с.

23. Нестеренко О.А., Широкопетлева М.С., Черепанова Ю.Ю. Применение индивидуализации процесса обучения в компьютерных обучающих средах / // Восточно-Европейский журнал передовых технологий. – 2005. – № 5/2 (17). – С. 99-103.

26. Широкопетлева М.С. О подсистеме адаптации интерфейса для лексикографического украино-русско-английского словаря с учетом индивидуальных психофизиологических особенностей личности // Вестник Нац. техн. ун-та "ХПИ" : сб. науч. тр. Темат. вып. : Новые решения в современных технологиях. – Харьков : НТУ "ХПИ". – 2012. – № 68 (974). – С. 118-122.

28. Про підхід до проектування системи тестування знань з мови SQL / М.С. Широкопетлева, Ю.Ю. Черепанова // Вісник Національного технічного університету "ХПІ". Серія : Механіко-технологічні системи та комплекси. - 2016. - № 17. - С. 50-55.

30. Бондарев В.М., Черепанова Ю.Ю. Мережева підтримка навчання програмуванню // V Міжнародна науково-практична конференція "Інформаційні технології та комп'ютерна інженерія", 27–29 травня 2015 р. – Івано-Франківськ: Супрун В. П., 2015. С. 49–50.

31. Apukhtin V., Shirokopetleva M., Skovorodnikova V., The Relevance of Using Message Brokers in Robust Enterprise Applications. 2019 International Scientific-Practical Conference on Problems of Infocommunications Science and Technology, PIC S and T 2019 - Proceedings, pp. 305-309.

32. Kuzochkina A., Shirokopetleva M., Dudar Z., Analyzing and Comparison of NoSQL DBMS. International Scientific and Practical Conference «Problems of Infocommunications. Science and Technology» (PIC S&T`2018), October 9-12, 2018. Proceedings 8632133, pp. 560-565.

ДОДАТОК Б

Апробація результатів роботи

MONOGRAFIA POKONFERENCYJNA

**ПРИМЕНЕНИЕ ПОДСИСТЕМЫ АДАПТАЦИИ
ИНТЕРФЕЙСА ДЛЯ ВЕБ-СИСТЕМ**

Широкопетлева М.С.

Харьковский национальный университет радиоэлектроники

Ключевые слова: веб-система, интерфейс, модель пользователя, адаптация интерфейса

Key words: web-system, interface, user model, interface adaptation

В существующих компьютерных системах с обратной связью модель пользователя, в основном, рассматривается как модель ошибок, что лишь частично повышает эффективность взаимодействия с системой по сравнению с системами без обратной связи.

В обучающих системах используются блоки начального, промежуточного и итогового контроля, причем, начальный контроль предназначен для выяснения уровня знаний обучаемого и ограничения представляемого материала, а промежуточный контроль – для выдачи рекомендаций по повторению ранее усвоенного материала или возвращения к предыдущим урокам (в зависимости от сценариев обучения).

Существующие информационные системы имеют один или несколько типов представления информации, которые «жестко» привязаны к предоставляемому материалу, т.е. вид представления информации зависит от предпочтений разработчика, заполняющего базу знаний системы и его нельзя изменять в зависимости от предпочтений (или возможностей) пользователей.

Целью технологии адаптивного представления является адаптация

представления учебного материала с учетом преобладающего типа восприятия информации пользователем, при этом знания и другая информация хранятся в модели пользователя [1]. В системе с адаптивным представлением страницы адаптивно генерируются на основе содержимого учебного материала и модели пользователя для каждого пользователя. Например, при применении техники адаптивного представления хорошо подготовленный пользователь будет получать более детализированную и углубленную информацию, а новичок получит больше дополнительных пояснений [1]. Адаптивное представление очень важно для веб-систем и мобильных приложений.

Программная система, которая учитывает индивидуальные особенности пользователя и адаптируется к его предпочтениям, делает процесс взаимодействия с системой максимально быстрым и эффективным.

Модель пользователя в системе представлена набором характеристик личности [2]. Модель включает в себя описание психотипа личности и оптимальных характеристик обучающего материала для данного типа личности.

Психотип личности может определяться объективными и субъективными характеристиками. Выделим их.

Объективные характеристики:

- пол;
- возраст;
- образование.

Субъективные характеристики:

- преобладающий тип интеллекта;
- тип характера – по степени активности личности;
- взаимоотношение с внешним миром – интроверсия и экстраверсия.

Объективные характеристики могут быть получены при заполнении регистрационных форм.

Субъективные характеристики могут быть получены из результатов тестов при создании стартовой модели, или же если пользователь откажется от тестов, при уточнении характеристик предоставляемого материала.

Для определения психотипа могут быть использованы следующие тесты:

- преобладающий тип интеллекта – тест Амтхауэра (TSI);
- тип темперамента – Lushera;
- взаимоотношение с внешним миром – Опросники Айзенка (EPI).

Шаги взаимодействия с системой подробно описаны в [3]:

- знакомство с системой – пользователь знакомится с возможностями и особенностями системы;

- регистрация в системе (начинается создание модели обучаемого) – включает в себя набор тестов необходимых для определения стартовых параметров модели;

- определение психотипа – включает в себя определение объективных и субъективных характеристик личности;

- завершение формирования стартовой модели обучаемого;

- процесс обучения – в процессе обучения уточняются параметры модели (путем определения взаимосвязей между характеристиками предоставленного материала и успешностью прохождения контрольных заданий);

- завершение обучения – пользователь покидает систему. Модель заносится в хранилище данных для последующих статистических исследований.

Список литературы

1. Adaptive Navigation Support in Educational Hypermedia on the World Wide Web. / Brusilovsky P., Eklund J., Schwarz E., 1997, p. 278-285.
2. Материалы 6-й Международной научно-практической конференции «Эффективные инструменты современных наук -2010», 27 апреля – 05 мая 2010. Том 20. Современные информационные технологии. Прага, изд. дом «Образование и наука», 2010 г. – с. 82-84.
3. Несміян Ю.Ю., Широкопетлева М.С. Про підхід до створення моделі користувача в адаптивній системі навчання // «Східно Європейський журнал перетових технологій», 2010, с. 51-54.

ДОДАТОК В

Слайди презентації

ХНУРЕ
Кафедра Програмної інженерії
Атестаційна робота магістра

1

Дослідження методів побудови індивідуальної траєкторії навчання в системах дистанційної освіти

- Виконала: студентка групи ПЗСзм-18-1 Широкопетлева М.С.
- Керівник: професор, к.т.н. Дудар З.В.

2

Мета дослідження

- Метою роботи є висвітлення особливостей різних методів побудови моделі користувача яка є базовою для побудови траєкторії навчання
- Об'єктом дослідження є процес побудови траєкторії навчання в системах дистанційної освіти.

LMS та MOOC

3



MOOC

4

Простота інсталяції	обов'язкова реєстрація	обов'язкова реєстрація	обов'язкова реєстрація	реєстрац.	реєстрац.	реєстрац.	реєстрац.
Вартість	безкошт., окрім книг та сертифікату	безкошт.	Безкошт. тільки 6 курсів	безкошт.	безкошт.	безкошт.	безкошт.
Документація	+	+	+	+	-	-	-
Підтримка мов	англ., рос. та інші. + субтитри англ., рос.	англійська, китайська, французька, гінді, іспанська	англ. + субтитри	наявність багатомовності	Китайська, англійська та ін.	наявність багатомовності	наявність багатомовності
Середня тривалість онлайн курсу	6-10 тижнів, 1-2 години відеолекцій на тиждень	3-5 міс.	1 міс.	навчальні курси відсутні, є навчальні відео	короткострокові	1-1.5 міс.	1-2 міс.
Платформи	Онлайн	Онлайн	Онлайн, ios	ios	Онлайн	ios, Android	Онлайн
Особливості	Найбільша, визнається світовими ЗВО	Використовується як платформа для інших систем		мікрокурси	Пошук та рекомендація курсів		

<https://sj.npu.edu.ua/index.php/kosny/article/view/29/30>

Узагальнена структура системи навчання



Моделі користувача

- Оверлейна
- Модель помилок

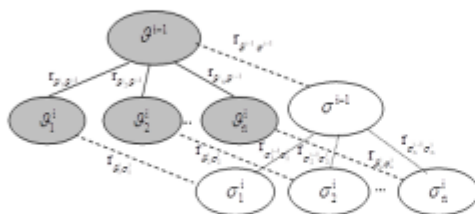
Постановка задачі

- Підвищення рівня сприйняття навчального матеріалу за рахунок подання матеріалу у формі, максимально «зручною» для сприйняття конкретною людиною:
 - розробка моделі понятійних знань для довільної проблемної області на основі класифікацій;
 - зміна розробленої класифікації з урахуванням індивідуальних особливостей учня (введення асоціативних зв'язків);
 - визначення виду представлення інформації окремому користувачеві на основі його психотипу;
- Розробка методів визначення доступності навчального матеріалу на підставі моделі користувача та моделі навчального контенту.

Функції програмної системи навчання

- можливість ведення навчальних блоків (курсів):
 - формування тем та підтем;
 - формування відповідних тестів;
 - визначення взаємозв'язку між темами та критеріїв засвоєння матеріалу;
- підтримка моделі користувачі з урахуванням психофізіологічних особливостей користувача:
 - визначення базової моделі на підставі психологічних тестів (опціонально);
 - визначення рівня засвоєння матеріалу;
 - корегування моделі користувача (опціонально);
- формування послідовності вивчення матеріалу з урахуванням моделі користувача:
 - визначення базової послідовності;
 - визначення допустимих для вивчення тем для конкретного користувача;
 - визначення допустимих тестових завдань;
- підтримка адаптивного інтерфейсу для представлення навчального матеріалу з урахуванням особливостей користувача.

Таксономічна класифікація з врахуванням істотних властивостей об'єкта



Зв'язок між видовим та родовим поняттями

$$\exists g^i, g^{i-1} \in \theta : (v^i < v^{i-1}, \omega^{i-1} \subset \omega^i) \Rightarrow \exists r_{g^i g^{i-1}} \in R_g$$

Наявність зв'язку «об'єкт-властивість»

$$\exists g_n^i (v_n^i, \omega_n^i, \sigma_n^i), g_n^i \in \theta : \Rightarrow \exists r_{g_n^i \sigma_n^i} \in R_{\sigma}$$

Зв'язок між родовими та видовими властивостями понять

$$\exists g_n^i(\sigma_n^i), g_n^{i-1}(\sigma_n^{i-1}) \in \theta, r_{g_n^i \sigma_n^{i-1}} \in R_g : \Rightarrow \exists r_{\sigma_n^i \sigma_n^{i-1}} \in R_\sigma$$

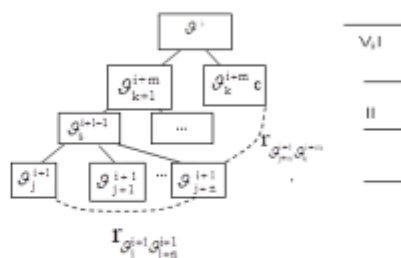
Зв'язок між видовим та родовим поняттями для багатоаспектної класифікації

$$\exists g^i, g^{i-1} \in \theta : (v^i \leq v^{i-1}) \Rightarrow \exists r_{g^i g^{i-1}} \in R_m$$

Критерії виділення ПОНЯТЬ

- Повноти $\forall \vartheta^i: (|\cup \vartheta^i| = |\cup^{i-1} \vartheta^i| \cup \vartheta^i) \cup |\cup^i \vartheta^i| = |\cup^{i-1} \vartheta^i| \cup \vartheta^i$
- Відсутність перетинань $\forall \vartheta^i: \vartheta_j^i \cap \vartheta_k^i = \emptyset (j \neq k)$
- Несуперечливості $\forall \vartheta^i: \vartheta_j^i \neq \neg \vartheta_k^i$
- Цілісності $\cup \vartheta_j^i (\forall \vartheta^i) = \vartheta^{i-1}_p$

Введення асоціативних зв'язків на таксономічній класифікації



Характеристична функція для відношень R_α для обліку значимості асоціативних відношень

$$\mu_{R_{\alpha}(\vartheta_l^{i-1}, \vartheta_m^{i-1})} = \begin{cases} 0, & \text{якщо } (l \neq 1 \vee m \neq 1) \vee (l = 0 \vee m = 0); \\ \frac{1}{1 + (l - m)^2}, & \text{якщо } l \neq 0 \wedge m \neq 0 \\ 0.7, & \text{якщо } (|l + m| > 0 \wedge l \neq 1 \wedge m \neq 1). \end{cases}$$

Модель інтелекту користувача

$$\Psi(k_j * (f_i)) = \sum_{i=1}^3 f_i k_i, (i=1...3, f_1 = f_j, f_2 = f_i, f_3 = f_j)$$

- динаміка зміни особливостей особистості $\kappa_i^2 = \kappa_i^1 * (1 - \Delta(f_i)), \Delta(f_i) = f_i^2 - f_i^1$
- Обвідна значень функції $f_j^j, (j \in R)$ показує динаміку змін здібностей учня в процесі навчання

- Вплив розвитку складових інтелекту на успішність по циклам дисциплін

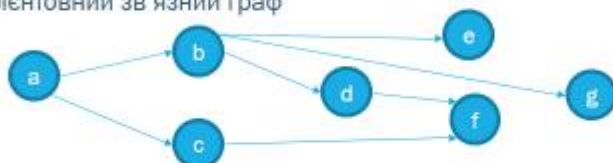


- Вплив ступеня активності особистості. Значення за замовчуванням (без визначення типу)

$$\delta_{sound}^{int} = 0.5 \quad \delta_{str}^{int} = 0.4 \quad \delta_{partial}^{int} = 0.2; \quad \delta_{verbal}^{int} = 0.2$$

Навчальний контент

- Орієнтовний зв'язний граф



- Використання алгоритмів топологічного сортування:

a, b, e, d, f, c, g

a, c, f, b, d, e, g

Підсистема тестування

- Критерій повноти: $S = \cup R_i$
- Критерій відсутності перетинань: $\forall i, j: R_i \cap R_j = \emptyset$
- Тестова множина: $S = \cup Q_i^l$
де Q_i^l – l -те питання тестового набору, яке входить до складу $R_i, l \geq 10$.

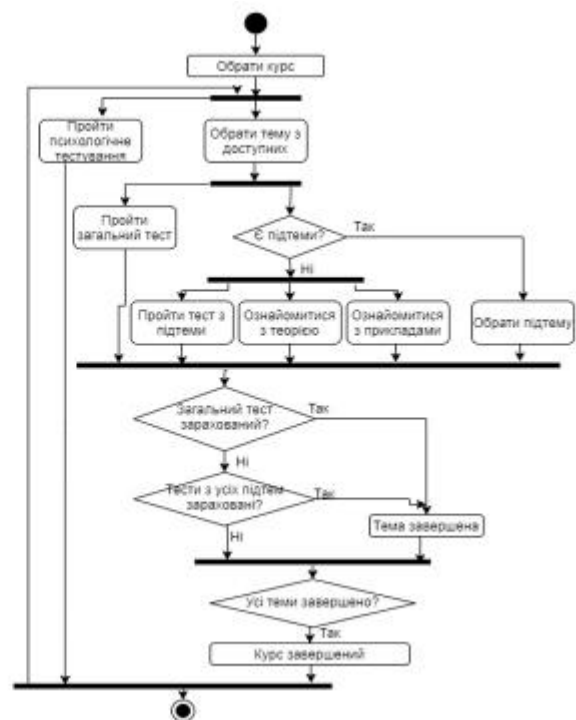
- Фрагмент визначення рівня складності для тестів з мови SQL

Підрозділ	Тема	Тематична одиниця	Рівень
SELECT	Прості запити	Форматування виведення	1
		Обмеження на виведення	1
		Спеціальні оператори, впорядкування	1
		Агрегатні операції	1
		Обмеження на групи	2
Запити на декількох таблицях	на декількох таблицях	Об'єднання таблиць	2
		Обмеження на дані з декількох таблиць	2
		Обмеження на дані та групи з декількох таблиць	3

Use-case діаграма



Діаграма діяльності студента



Головна сторінка

Сторінка курсу (викладач)

15

LMS Your choice

ГОЛОВНА КУРСИ ІНСТРУКЦІЯ ПРО ПРОГРАМУ Teacher 1

Your choice

Вітаємо в системі підтримки дистанції навчання Your Choice.

Ця система на підставі інформації курсу вподобань дозволяє самостійно обирати опанування різними дисциплінами та нестандартне представлення матеріалу.

Інструкція

LMS Your choice

ГОЛОВНА КУРСИ ІНСТРУКЦІЯ ПРО ПРОГРАМУ

КУРСИ

- ДИМАНЧНІ ІГРИ
- УПРАВЛІННЯ
- **БАЗИ ДАНИХ**
- МЕНЕДЖМЕНТ
- ПРОГРАМУВАННЯ
- ОПЕРАЦІЙНІ СИСТЕМИ

БАЗИ ДАНИХ

Дисципліна "Бази даних" передбачає ознайомлення з теорією БД, зокрема реляційних БД та вивчення мови структурованих запитів SQL.

Метою курсу є вивчення основних питань проектування та реалізації баз даних (БД), методів моделювання даних та розробки реляційних баз даних, вивчення мов запитів до БД, ознайомлення з сучасними системами управління БД (СУБД), придбання практичних навиків проектування, розробки і експлуатації баз даних та інформаційних систем, створених на їх основі.

Приклади екранних форм програмної системи

16

LMS Your choice

ГОЛОВНА КУРСИ ІНСТРУКЦІЯ ПРО ПРОГРАМУ Teacher 1

КУРСИ

- ДИМАНЧНІ ІГРИ
- УПРАВЛІННЯ
- **БАЗИ ДАНИХ**
- МЕНЕДЖМЕНТ
- ПРОГРАМУВАННЯ
- ОПЕРАЦІЙНІ СИСТЕМИ

БАЗИ ДАНИХ | РЕДАГУВАННЯ

Алгоритм нормалізації

Нормалізація – це розділення таблиць на дві або більш, таких, що володіють кращими властивостями під час включення, зміни та видалення даних.

Кожній нормальній формі відповідає деякий певний набір обмежень, і відношення знаходиться в деякій нормальній формі, якщо задовольняє властивому для неї набору обмежень. Прикладом набору обмежень є обмеження першої нормальної форми – значення всіх атрибутів відношення атомарні. Оскільки вимога першої нормальної форми є базовою вимогою класичної реляційної моделі даних, ми будемо вважати, що початковий набір відношень вже відповідає цій вимозі.

У теорії реляційних баз даних звичайно виділяється наступна послідовність нормальних форм:

- перша нормальна форма (1nf);
- друга нормальна форма (2nf);
- третя нормальна форма (3nf);
- нормальна форма Бойса-Кодда (BCNF);
- четверта нормальна форма (4nf); - п'ята нормальна форма, або нормальна форма проєкції-з'єднання (5nf або Pnf).

ЗВЕРНІТЬСЯ

Інструкція

Приклади екранних форм програмної системи

17

ГОЛОВНА
КУРСИ
ІНСТРУКЦІЯ
ПРО ПРОГРАМУ
Teacher 1

ГОЛОВНА
КУРСИ
ІНСТРУКЦІЯ
ПРО ПРОГРАМУ
Teacher 1

БАЗИ ДАНИХ |

- T1 Моделі даних
 - ST11 Мережева
 - ST12 Ієрархічна
 - ST13 Об'єктно-орієнтовна
 - ST14 Реляційна
- T2 Реляційна концепція
 - ST21 Структурна частина
 - ST22 Маніпуляційна частина
- T3 Проектування схеми БД
 - ST31 Концептуальне моделювання
 - ST32 Інфологічне моделювання
 - ST33 Алгоритми нормалізації
- T4 Мова SQL
 - ST41 DML
 - SST411 SELECT
 - SST412 INSERT
 - SST413 UPDATE
 - SST414 DELETE
 - SST42 DDL

БАЗИ ДАНИХ | РЕДАГУВАННЯ

- T1 Моделі даних
 - ST11 Мережева
 - ST12 Ієрархічна
 - ST13 Об'єктно-орієнтовна
 - ST14 Реляційна
- T2 Реляційна концепція
 - ST21 Структурна частина
 - ST22 Маніпуляційна частина
- T3 Проектування схеми БД
 - ST31 Концептуальне моделювання
 - ST32 Інфологічне моделювання
 - ST33 Алгоритми нормалізації
- T4 Мова SQL
 - ST41 DML
 - SST411 SELECT
 - SST412 INSERT
 - SST413 UPDATE
 - SST414 DELETE
 - SST42 DDL

Інструкція

18

Висновки

- Представлено підхід до побудови траєкторії навчання:
 - Проаналізовані методи представлення навчального матеріалу
 - Запропонований підхід до виділення основних понять предметної галузі, критерії виділення понять
 - Продемонстрована можливість використання орґафу для організації навчального контенту
 - Запропонована модель користувача з урахуванням індивідуальних особливостей
 - Продемонстровано використання топологічного сортування для побудови траєкторії навчання
- Результатом роботи є прототип програмної системи навчання, який дозволяє формувати навчальні блоки, підтримувати модель користувача та формувати послідовність вивчення матеріалу з урахуванням моделі користувача
- Подано тези доповіді на Міжнародну науково-практичну конференцію Science, Research, Development, Technics and technology. №35

ДОДАТОК Г

Фрагмент коду програмної системи

```
using System;
using System.Collections.Generic;
using System.Linq;

namespace Graphs
{
    class DepthFirstSearch
    {
        private HashSet<Node> visited;
        private LinkedList<Node> path;
        private Node goal;
        private bool limitWasReached;
        public LinkedList<Node> DFS(Node start, Node goal)
        {
            visited = new HashSet<Node>();
            path = new LinkedList<Node>();
            this.goal = goal;
            DFS(start);
            if (path.Count > 0)
            {
                path.AddFirst(start);
            }
            return path;
        }

        private bool DFS(Node node)
        {
            node.Handler();
            if (node == goal)
            {
                return true;
            }
            visited.Add(node);
            foreach (var child in node.Children.Where(x =>
!visited.Contains(x)))
            {
                if (DFS(child))
                {
                    path.AddFirst(child);
                    return true;
                }
            }
            return false;
        }
    }
}
```

```

public LinkedList<Node> DLS(Node start, Node goal, int
limit)
{
    visited = new HashSet<Node>();
    path = new LinkedList<Node>();
    limitWasReached = true;
    this.goal = goal;
    DLS(start, limit);
    if (path.Count > 0)
    {
        path.AddFirst(start);
    }
    return path;
}

private bool DLS(Node node, int limit)
{
    node.Handler();
    if (node == goal)
    {
        return true;
    }
    if (limit == 0)
    {
        limitWasReached = false;
        return false;
    }
    visited.Add(node);
    foreach (var child in node.Children.Where(x =>
!visited.Contains(x)))
    {
        if (DLS(child, limit - 1))
        {
            path.AddFirst(child);
            return true;
        }
    }
    return false;
}

public LinkedList<Node> IDDFS(Node start, Node goal)
{
    for (int limit = 1; ; limit++)
    {
        var result = DLS(start, goal, limit);
        if (result.Count > 0 || limitWasReached)
        {
            return result;
        }
    }
}
}
}

```