

*Гломозда Дмитро Костянтинович*, асистент кафедри інформатики Національного університету „Києво-Могилянська Академія”.

Адреса для листування: 03037, Київ, вул. Максима Кривоноса, д. 14, кв. 7, e-mail: [glomozda@voliacable.com](mailto:glomozda@voliacable.com)

## ПРОТОТИП КОЛАБОРАТИВНОЇ СИСТЕМИ ДИСТАНЦІЙНОЇ ОСВІТИ ДЛЯ ВИЩОГО НАВЧАЛЬНОГО ЗАКЛАДУ

### **Вступ**

В роботі продовжується розгляд питань, пов'язаних з побудовою моделі програмної системи підтримки дистанційної взаємодії в масштабах Інтернет (далі ПСПДВІ), що ґрунтується на технології рівневого контролю [1, 2]. В попередніх роботах [3, 4] була запропонована модель ПСПДВІ мовою мереж Петрі та описана багатоагентна система, що моделювала роботу складових такої системи. Мета цієї роботи — використати одержані результати для побудови прототипу координаційної системи, що пов'язує автоматизовану систему управління навчальним закладом (далі АСУНЗ) з системою керування навчанням (далі СКН).

### **Використані технології**

До складу колаборативної системи дистанційної освіти для вищого навчального закладу входять АСУНЗ, СКН та система забезпечення координації роботи перших двох складових.

В якості СКН було взято використовуване в НаУКМА середовище Moodle, розгорнуте в середовищі WampServer 2.0. В якості АСУНЗ — систему MAMS [5]. Третя складова — наша власна

EPCS (Education Process Coordination System — система координації навчального процесу), яка координує роботу користувачів Moodle та MAMS, запобігаючи виникненню конфліктних ситуацій. Прикладом такої конфліктної ситуації може бути випадок, коли вчитель виставляє учням оцінки за підсумковий тест з курсу, а методист кафедри запитує підсумкову іспитово-залікову відомість для цього курсу. Якщо їх дії не узгоджувати, методист кафедри може отримати відомість із заниженими оцінками, що призведе до неприємностей.

В якості мови програмування для написання EPSCS було обрано PHP, а в якості платформи розгортання — WampServer 2.0. До складу останнього входять інтерпретатор мови PHP з великим набором розширень, в тому числі php\_mysql та php\_mssql для роботи з базами даних під управлінням СКБД MySQL та Microsoft SQL Server, та СКБД MySQL з програмою phpMyAdmin — веб-застосуванням для адміністрування баз даних MySQL, необхідне для взаємодії з середовищем MOODLE.

### **Опис системи**

Програмна частина системи EPSCS являє собою веб-застосування, написане мовою PHP з використанням JavaScript, яке складається з трьох модулів: модуль входу в систему та реєстрації нових користувачів, модуль викладача та модуль методиста кафедри. Модулі взаємодіють між собою через базу даних „epcs”. Вона складається з двох таблиць, „users” (користувачі) та „active\_tasks” (активні задачі). В таблиці „users” зберігаються дані про зареєстрованих користувачів: логіни, паролі, тип користувача (адміністратор, методист чи викладач), імена в системах MAMS та Moodle, а також прапорець, який показує, чи перебуває даний

користувач зараз у системі. В таблиці „active\_tasks” зберігаються відомості про задачі, які в даний момент виконуються в системі, а саме тип задачі, ім'я користувача-виконавця та ідентифікаційний номер задіяного при цьому ресурсу.

**Модуль входу та реєстрації нових користувачів** включає в себе такі програмні файли:

а) index.html — стартова сторінка системи з полями, куди зареєстровані користувачі вводять свої імена та паролі. Система не передбачає гостьового доступу до своїх функцій, тож нові користувачі мають зареєструватись, натиснувши на кнопку „New user”;

б) newuser.php — програма, що запускається, коли користувач натискає кнопку „New user”. Вона пропонує користувачу ввести ім'я, під яким він хоче бачити себе в системі; свою роль (методист чи викладач); імена, під якими він зареєстрований в системах Moodle та MAMS, а також бажаний пароль (з підтвердженням). Після заповнення всіх полів слід натиснути на кнопку „Register”, що запускає програму newuser\_check.php;

в) newuser\_check.php — програма перевірки правильності введених користувачем при реєстрації параметрів. Спершу перевіряється ідентичність двох введених екземплярів паролів. Потім за допомогою SQL-запитів до баз даних Moodle та MAMS перевіряється, чи зареєстровані такі користувачі в цих системах (а в разі, якщо користувач хоче зареєструватися як викладач, чи є він викладачем хоча б одного курсу в Moodle), і чи є вже користувач з таким іменем в EPCS. Крім того для запобігання створенню дублюючих один одного облікових записів в базі даних EPCS міститься інформація про те, які імена користувачів Moodle та MAMS вже були використані, і якщо новий користувач назвався вже

зайнятим іменем, система його про це повідомить і попросить ввести інше ім'я. Якщо ж усі перевірки пройшли вдало, система повідомляє користувачу, що він може входити і працювати в системі;

г) login.php — програма обробки введених користувачами імен та паролів. За допомогою SQL-запиту програма звертається до бази даних „erps” і перевіряє, чи існує такий користувач і чи збігається його пароль із уведеним. Якщо всі перевірки проходять вдало, програма запитує з бази даних роль користувача в системі та завантажує відповідний модуль. При цьому ім'я користувача передається через рядок браузера, щоб у подальшому його можна було використовувати в запитах до баз даних.

**Модуль викладача** включає засоби, якими вчителі послуговуються в своїй повсякденній роботі. До його складу входять файли:

а) teacher.php — основна робоча сторінка вчителя, на якій містяться його робочі інструменти, зокрема засіб для виставлення студентам оцінок за підсумковий тест. Містить два випадających списки для вибору курсу та групи, при чому перелік курсів обмежений лише тими, які даний користувач саме викладає. Для того, щоб список груп формувався в залежності від того, який курс обрано, використовується вставка мовою JavaScript, яка перевантажує сторінку кожного разу, коли змінюється обраний курс. Після того, як викладач обрав курс і групу, програма надсилає до бази даних Moodle запит, за результатами якого будується форма введення оцінок;

б) enter\_grades.php — програма обробки та запису введених викладачем оцінок до бази даних. Вона зчитує надіслану попередньою формою інформацію з асоціативного масиву `_REQUEST` і перевіряє її

на відповідність таким умовам: а) вона має бути числом; б) вона має бути не меншою за мінімально можливу оцінку і не більшою за максимально можливу оцінку;

Якщо ці умови виконуються, програма заносить оцінку до бази даних та відповідним чином переобраховує сумарну кількість набраних студентом балів. За результатами роботи програми користувачу надається звіт, з якого він може дізнатися, чи були оцінки виставлені і чи не було якихось помилок чи то з уведеними даними, чи з самою базою даних.

**Модуль методиста кафедри** включає інструменти, якими методисти кафедр послуговуються в своїй повсякденній роботі. До його складу входять файли:

а) `dep_methodist.php` — як і аналогічний йому `teacher.php`, містить робочі інструменти методиста кафедри. На разі вони представлені засобом складання заліково-іспитових відомостей для потрібних курсу та групи, які, як і на сторінці викладача, обираються за допомогою двох випадючих списків, наповнення для яких береться з бази даних середовища Moodle. Коли потрібна група обрана, з'являється кнопка „Generate”, яка запускає програму генерації відомості `gen_grades.php`;

б) `gen_grades.php` — програма генерації заліково-іспитових відомостей. Запитує з бази даних імена членів групи та їх підсумкові оцінки за даний курс. Отримані результати виводяться на екран у вигляді таблиці.

Координація дій користувачів, в даному випадку викладача і методиста, відбувається через вже згадувану таблицю „`active_tasks`” бази даних „`erps`”. Кожна задача, що потребує використання

потенційно конфліктного ресурсу, реєструється в цій таблиці за допомогою SQL-запиту. Наприклад, реєстрація процесу виставлення оцінок групі студентів виглядатиме так:

```
INSERT
INTO active_tasks
VALUES ($i, 'grades_entering', '$username', $group_id),
```

де \$i — збільшене на одиницю максимальне з наявних у таблиці значень ідентифікаційного номера задачі, який є первинним ключем і не може повторюватися (або 1, якщо таблиця порожня), „grades\_entering” (введення\_оцінок) — тип задачі, \$username — ім'я користувача (викладача), який виставляє оцінки, \$group\_id — ідентифікатор групи. Тепер користувачі, що намагатимуться здійснити яку-небудь операцію з оцінками цієї групи, натомість отримуватимуть повідомлення про те, що в даний момент часу певний викладач якраз вводить оцінки для цієї групи, і слід зачекати. Якщо ж якомусь користувачу просто знадобиться вивести на екран склад цієї групи, він зробить це без перешкод, оскільки ця дія не стосується оцінок.

Після того, як оцінки були введені і внесені до бази даних, запис про цю задачу видаляється з бази даних „ercs”. Коли користувач виходить з системи, запускається програма logout.php, до складу якої входить запит, що видаляє всі задачі, зареєстровані даним користувачем. Таким чином вдається запобігти блокуванню даних, що насправді не використовуються, але вважаються зайнятими.

## Висновки

Як показали випробування, система EPCS ефективно перешкоджає виникненню позаштатних ситуацій, пов'язаних з неузгодженим доступом користувачів до даних, що підтверджує адекватність запропонованих раніше моделей. На разі ведеться робота по інтегруванню EPCS до MAMS та Moodle в єдину систему. В перспективі планується впровадження такої системи в усьому НаУКМА.

## ЛІТЕРАТУРА

1. *Dommel H-P.* Floor control for multimedia conferencing and collaboration / H-P. Dommel, J.J. Garcia-Luna-Aceves // *Multimedia Systems.* — Vol. 5. — Springer-Verlag, 1997. — P. 23—38.
2. *Dommel H-P.* Networking Foundations for Collaborative Computing at Internet Scope / H-P. Dommel, J.J. Garcia-Luna-Aceves // *International ICSC Congress on Intelligent Systems and Applications, Symposium on Interactive and Collaborative Computing (ICC'2000).* — Wollongong (Australia), 2000.
3. *Гломозда Д.К.* Формальна модель функціонування колаборативного середовища / *Д.К. Гломозда, М.М. Глибовець* // *Тези доповідей Третьої Міжнародної конференції «Теоретичні та прикладні аспекти побудови програмних систем» (TAAPSD'2006).* — Київ (Україна). — 2006. — С. 225—230.
4. *Гломозда Д.К.* Використання програмних агентів для організації ефективного управління роботою програмної

системи підтримки мережної співпраці / *Д.К. Гломозда* // Наукові записки. Т. 86: Комп'ютерні науки / Національний університет "Києво-Могилянська Академія". — К., 2008. — С. 53—56.

5. *Глибовець М.М.* Розробка системи управління навчального закладу на прикладі НаУКМА / *М.М. Глибовець, С.А. Іващенко, О.О. Крусь* // Наукові праці: науково методичний журнал. — Серія "Комп'ютерні науки", Т. 57, Вип. 44. — Миколаїв: Вид-во МДГУ ім. Петра Могили, 2006. — С. 214-219.