

ПРО ОНОВЛЕННЯ “ІНФОРМАЦІЙНОЇ СИСТЕМИ ПЕРЕВІРКИ ЗНАНЬ У МЕДИЧНІЙ ОСВІТІ”

А. В. Семенець, С. Б. Чеканов

*ДВНЗ “Тернопільський державний медичний університет імені І. Я. Горбачевського
МОЗ України”*

ON UPDATE OF THE “INFORMATION SYSTEM OF KNOWLEDGE ASSESSMENT IN MEDICAL EDUCATION”

A. V. Semenets, S. B. Chekanov

I. Horbachevsky Ternopil State Medical University

Розглянуто деякі аспекти інформатизації галузі медичної освіти. Показано необхідність оновлення програмного забезпечення ІСПЗМО при внесенні змін у методику проведення семестрового іспиту. Розроблено методику оновлення ПЗ ІСПЗМО, що полягає в створенні окремої підсистеми обробки результатів оцінювання тестових робіт студентів та усної співбесіди. Наведено приклад реалізації вказаної підсистеми у складі ІСПЗМО, як веб-додатка – ІС CSE. Показано практичну методику реалізації вказаного веб-додатка на мові програмування PHP з використанням відкритого веб-фреймворку Laravel.

Some aspects of informatization of the medical education field are considered. The key points of knowledge assessment methodology in TSMU are shown. An oral discuss as a part of semester exam knowledge assessment is introduced. The requirement of the ISKAME software update after the exam methodology changes is confirmed. Benefits of usage of information systems combined with applications of the cloud technologies in the medical education process are presented. The ISKAME software update methodology is proposed. This methodology is based on the separated ISKAME subsystem CSE development. The CSE main features are described. The CSE information model development and implementation is presented. Advantages of usage of migrations for the CSE database deployment are shown. The CSE implementation as a web-application using Laravel framework is offered. The controllers for the CSE knowledge assessment algorithms implementation are developed. The approach of the integration of the CSE application with information systems used in TSMU and Google Apps for Education cloud service is offered.

Вступ. Процес інформатизації галузі освіти загалом та медичної освіти, зокрема, передбачає зростання числа різноманітних інформаційних систем (ІС), що перебувають в експлуатації. Своєчасне оновлення, рефакторинг та підтримка програмного забезпечення (ПЗ) даних ІС – обов’язкова умова забезпечення ефективного, надійного та безпечного їх використання.

Основна частина.

1. Інформаційне забезпечення проведення семестрового іспиту в ТДМУ

Протягом останнього десятиліття в навчальний процес ТДМУ запроваджено велику кількість нових методик та підходів до підготовки майбутніх спеціалістів. Більша частина вказаних методик ґрунтується на використанні цілого ряду інформаційних технологій та ресурсів [1, 2]. Зокрема, починаючи з 2006–2007 н. р., іспити в ТДМУ проводяться лише у формі бланкового тестового конт-

ролю [2]. При підготовці іспиту використовується розроблена А. В. Семенцем “Інформаційна система перевірки знань в медичній освіті” (ІСПЗМО), що являє собою клієнт-серверний Windows-додаток [3]. Обробка результатів автоматизованого оцінювання письмових тестових робіт студентів здійснювалася з використанням окремого додатка, розробленого С. Б. Чекановим.

У зв’язку з запровадженнями засад кредитно-трансферної організації навчального процесу у 2015–2016 н. р. було здійснено подальше реформування методики проведення тестового іспиту з метою об’єктивізації результатів. При цьому було запроваджено усну співбесіду як складову частину іспиту. Вказані організаційно-методичні зміни викликали необхідність оновлення ПЗ обробки результатів автоматизованого оцінювання тестових робіт студентів з метою врахування їх оцінки на усній співбесіді.

Метою даної роботи є представлення досвіду авторів щодо редизайну та оновлення підсистеми автоматизованої обробки результатів оцінювання тестових робіт студентів (CSE) у складі ІСПЗМО.

2. Розробка ІС CSE як підсистеми у складі ІСПЗМО

2.1. Оновлення інформаційної моделі ІСПЗМО

Вимоги до проведення семестрового іспиту в ТДМУ (станом на 2015–2016 н. р.) передбачають запровадження усної співбесіди як складової частини іспиту. Співвідношення балів ЄКТС між тестуванням та співбесідою встановлено як 60/20. Співбесіда проходить в той же день, що і тестування, однак проводять її викладачі кафедр, за місцем їх фактичного розміщення.

Таким чином, виникає необхідність:

- оперативного введення даних про результати співбесіди студентів;
- встановлення остаточної оцінки за іспит;
- формування необхідної документації та аналізу даних.

Очевидно, що вказана задача не потребує внесення змін до концептуальної моделі ІСПЗМО, яка показана в роботі [4]. Необхідні зміни функціонально-модульної структури ПЗ ІСПЗМО представлені на рисунку 1.

Таким чином, процес оновлення ІСПЗМО полягає в розробці ІС CSE, як окремої підсистеми ІСПЗМО, що забезпечить реалізацію необхідного функціоналу (див. виділення на рис. 1).

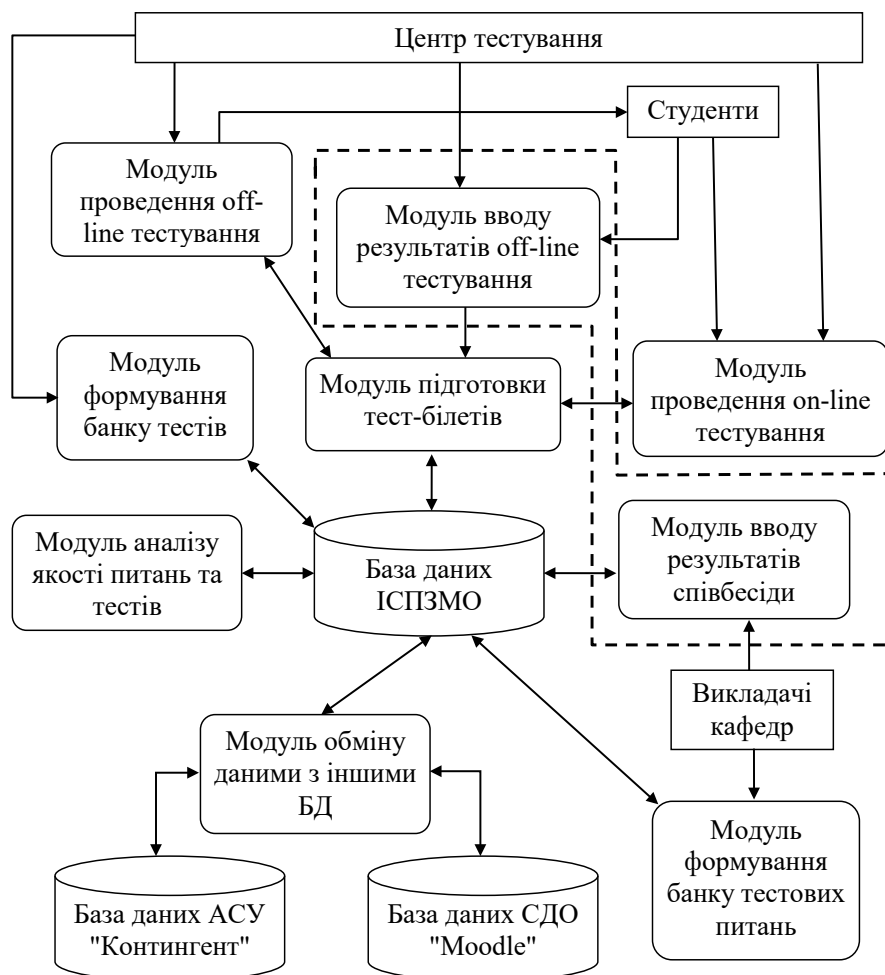


Рис. 1. Функціонально-модульна структура програмного забезпечення ІСПЗМО після оновлення.

Слід врахувати, що формування сучасної інформаційної інфраструктури медичного ВНЗ передбачає застосування вільно-розповсюдженого програмного забезпечення з відкритим кодом [5, 6]. Починаючи з 2013 р., в ТДМУ триває процес

впровадження хмарних сервісів Google Apps For Education у навчальний процес, що показано в роботі [7]. Важливо також забезпечити інтеграцію з ІС “Контингент” (<http://contingent.tdmu.edu.ua/>). Вказана ІС розроблена на замовлення Міністер-

ства охорони здоров'я України (наказ № 34-Адм від 04.03.1999 р.) для автоматизації обліку руху контингенту студентів та створення відповідних наказів і звітів, а також для контролю міністерством руху контингенту студентів у всіх медичних ВНЗ України та отримання загальної статистики.

Проблема комплексної інтеграції інформаційних систем та онлайн-сервісів, що використовуються в ТДМУ, в єдине інформаційне середовище була вирішена авторами шляхом розробки IC "SysAdminka", що показано в роботі [8]. Очевидною є необхідність інтеграції новоствореної IC CSE з IC "SysAdminka".

Перелік основних об'єктів інформаційної моделі IC CSE включає:

1. XML-файл – описує інформацію про відомості успішності студентів з дисципліни. Застосовується для обміну даними з IC "Контингент".

2. XLS-файл – описує дані результатів автоматичного оцінювання письмових відповідей студентів на тестову частину іспиту.

3. Іспит (з дисципліни) – інтегрує дані сутностей "XML-файл" та "XLS-файл".

4. Відомість – загальна інформація про іспит з дисципліни.

5. Успішність – детально описує успішність студента з дисципліни в балах (поточна успішність, тестовий іспит, співбесіда тощо).

6. Користувач – інформація про користувачів системи.

Нижче буде розглянуто деякі практичні аспекти реалізації IC CSE.

2.2. *Практична реалізація IC CSE як веб-додатка*
Виходячи з рекомендацій роботи [9], IC CSE розробляється у вигляді крос-платформового веб-додатка. В якості мови програмування С. Б. Чекановим запропоновано використання мови PHP у поєднанні з наступними фреймворками:

- Twitter Bootstrap (<http://getbootstrap.com/>) – найбільш поширений на даний момент HTML, CSS та JS фреймворк для розробки веб-додатків;

- Laravel (<https://laravel.com/>) – зручний PHP-фреймворк з ефективною реалізацією MVC (Model-View-Controller) підходу до розробки додатків;

- AngularJS (<https://angularjs.org/>) – JavaScript-фреймворк від компанії Google, що забезпечує швидку розробку функціональної клієнтської частини веб-додатка.

Фреймворк Laravel надає програмістам широкий перелік інструментальних засобів розробки. Особливої уваги заслуговує наявність розвинених

засобів автоматичної генерації об'єктів РСУБД з використанням механізму міграцій. У лістингу (лістинг 1) показано приклад міграції, що реалізує створення в БД таблиць для зберігання даних сутності "Користувач" та інформації про відповідні ролі.

Лістинг 1. Файл міграції для сутності "Користувач" в IC CSE

```
<?php
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Database\Migrations\Migration;
class CreateUsersTable extends Migration
{
    public function up()
    {
        Schema::create('roles', function (Blueprint
$table) {
            $table->increments('id');
            $table->string('name');
            $table->timestamps();
            $table->softDeletes();
        });
        Schema::create('users', function (Blueprint
$table) {
            $table->increments('id');
            $table->string('name');
            $table->string('username')->unique();
            $table->string('email')->unique();
            $table->string('password', 60);
            $table->string('confirmation_code');
            $table->boolean('confirmed')->default(false);
            $table->unsignedInteger('role_id')->nullable();
            $table->foreign('role_id')->references('id')->on('roles')->onDelete('set null');
            $table->string('lang');
            $table->rememberToken();
            $table->timestamps();
            $table->softDeletes();
        });
    }
    public function down()
    {
        Schema::drop('users');
        Schema::drop('roles');
    }
}
```

В якості сховища даних для IC CSE авторами вибрано СУРБД з відкритим кодом MySQL (<http://www.mysql.com/>). Внутрішнє представлення БД, згенероване за допомогою механізму міграцій Laravel, на основі відповідних моделей сутностей, показано на рисунку (рис. 2).

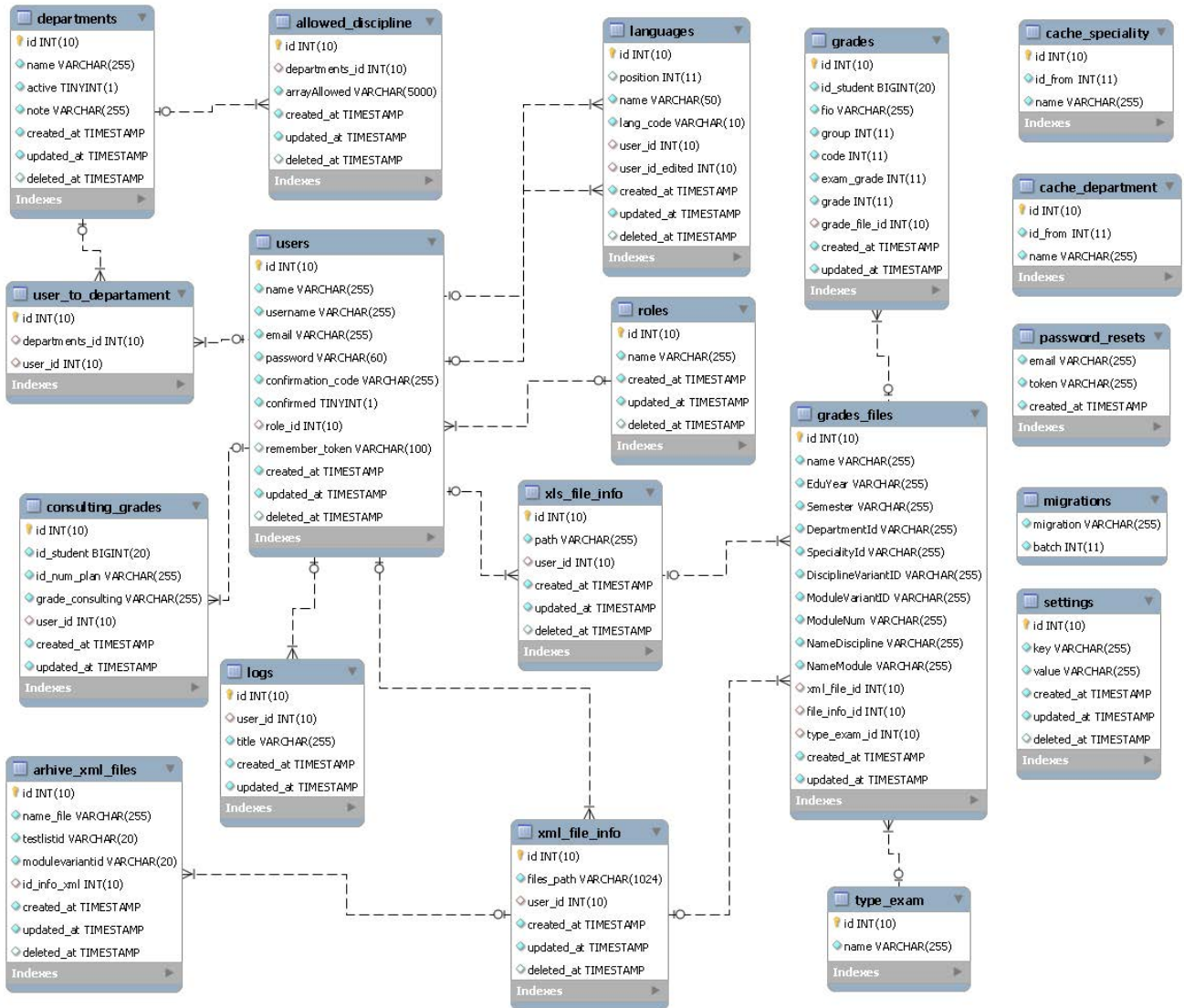


Рис. 2. Внутрішнє представлення БД IC CSE.

Використання засобів автоматичної генерації програмного коду суттєво скорочує трудомісткість та час розробки додатка, дозволяє програмісту сконцентрувати зусилля на розробці ключового функціоналу. При реалізації IC CSE авторами розроблено більше 20 класів контролерів з використанням засобів автогенерації коду методів, зокрема:

- ArchiveController.php – містить методи керування переліком попередніх (архівних) іспитів – перегляд, редагування параметрів;

- DocumentsController.php – включає методи керування виділеним (поточним) іспитом – отримання всіх відомостей, вибірка результатів співбесіди, підготовка звітів та статистики, розсилка поштових повідомлень;

- XMLController.php – реалізує методи експорту/імпорту відомостей у вигляді XML-файла для інтеграції з IC “Контингент”;

- ExcelController.php – містить метод імпорту оцінок студентів за тестову частину іспиту у форматі файла Excel;

- TeacherSetGrade.php – включає методи встановлення та редагування оцінок за співбесіду;

- DashboardController.php – реалізує методи керування графічним оточенням користувача.

Одним з ключових в IC CSE є контролер TeacherSetGrade.php, що реалізує методи реєстрації оцінок викладачів відповідей студентів на співбесіді (див. лістинг 2). Повністю з програмним кодом, як даного модуля, так і проекту в цілому, можна

ознайомитися на сервері GitHub в репозиторії авторів роботи (<https://github.com/TDMU/CSE>).

Лістинг 2. Код контролера *TeacherSetGrade.php* на мові PHP

```

class TeacherSetGrade extends Controller
{
    public function edit($deplId,$moduleVariant)
    {
        $this->about_module = GradesFiles::
where('ModuleVariantID',$moduleVariant)->where
('DepartmentID', $deplId)->get();
        $students = Grades::select('consulting_
grades.id_student as stud_consult_grade',
        'consulting_grades.grade_consulting',
        'grades.id_student',
        'grades.fio',
        'grades.group',
        'grades.grade'
        )->
        whereIn('grade_file_id', (array)$this-
>about_module->lists('id')->toArray())
        ->leftJoin('consulting_grades',
function($join)
    {
        $join->on('consulting_grades.
id_student', '=', 'grades.id_student')
        ->where('consulting_grades.
id_num_plan', '=', $this->about_module->first()-
>ModuleVariantID);
    })
    ->orderBy('group', 'asc')
    ->orderBy('fio', 'asc')
    ->distinct()
    ->get();
        $about_module = $this->about_
module->first();
        return view('admin.consulting.edit_add',
compact('about_module', 'students'));
    }
    public function saveGrade(Request
$request, ConsultingGrades $cons)
    {
        $check = $cons->where('id_
student', $request['student'])->where('id_num_
plan', $request['modnum'])->get()->first();
        if(!is_int($request['value']) &&
$request['value']>20){
            return json_encode(['status'=>'false',
'message'=>trans("admin/modules/consulting.value
MoreThen20"), 'grade'=>(isset($check->id))?$check-
>grade_consulting:"]);
        }
        if ($check) $cons->where('id', $check-

```

```

>id)->delete();
        if($request['value']!="") {
            if ($cons->create([
                'id_student' => $request['student'],
                'id_num_plan' =>
$request['modnum'],
                'grade_consulting' =>
$request['value'],
                'user_id' => Auth::user()->id,
                'department_id' =>
$request['deplId']
            ])
            ) {
                return json_encode(['status'=>'tr
ue', 'message'=>trans("admin/modules/consulting.
ok")]);
            }
        }
        return json_encode(['status'=>'false', 'm
essage'=>trans("admin/modules/consulting.error")]);
    }
    public function clearGrade(Request
$request, ConsultingGrades $cons)
    {
        $check = $cons->where('id_
student', $request['student'])->where('id_num_
plan', $request['modnum'])->get()->first();
        if ($check) {
            $cons->where('id', $check->id)-
>delete();
            return json_encode(['status'=>'true', '
message'=>trans("admin/modules/consulting.ok")]);
        }
        return json_encode(['status'=>'false', 'm
essage'=>trans("admin/modules/consulting.error")]);
    }
    public function data()
    {
        $allowDiscepline =
UserToDepartments::where('user_id', Auth::user()-
>id)->join('allowed_discipline', 'allowed_discipline.
departments_id', '=', 'user_to_department.
departments_id')->get()->first();
        $grades = GradesFiles::select(array(
            'grades_files.EduYear',
            'grades_files.Semester',
            'cache_department.name as
nameDep',
            'cache_speciality.name as
nameSpec',
            'grades_files.NameDiscipline',
            'grades_files.NameModule',
            'type_exam.name as

```

```

typeExamName',
    'grades_files.ModuleNum',
    'grades_files.ModuleVariantID',
    'grades_files.DisciplineVariantID',
    'grades_files.DepartmentId as
DepartmentId',
    ))->distinct('ModuleVariantID')
->join('cache_department','cache_department.
id_from','=', 'grades_files.DepartmentId')
->join('type_exam','grades_files.
type_exam_id', '=', 'type_exam.id') ->join('cache_
speciality','cache_speciality.id_from','=', 'grades_files.
SpecialityId');
    if(isset($allowDiscepline-
>arrayAllowed)) {
        if (in_array(Auth::user()->role_id, [5]) &&
(count(json_decode($allowDiscepline-
>arrayAllowed)) > 0)) {
            $grades-
>whereIn('DisciplineVariantID', (array)json_
decode($allowDiscepline->arrayAllowed));
        }
    }elseif(in_array(Auth::user()->role_
id,[5])){
        $grades->whereIn('DisciplineVariant
ID',null);
    }
    $grades = $grades->get();
    foreach($grades as $key=>$grade) {
        $this->about_module = GradesFiles::
where('ModuleVariantID',$grade->ModuleVariantID)-
>where('DepartmentId', $grade->DepartmentId)-
>get();
        $gradesEachStud =
Grades::select('id_student')->distinct('id_student')-
>whereIn('grade_file_id', (array)$this->about_
module->lists('id')->toArray())->get()->count();
        $consultGradeEachStud = Consulting
Grades::select('id_student')->distinct('id_student')-
>where('id_num_plan',$grade->ModuleVariantID)-
>where('department_id', $grade->DepartmentId)-
>get()->count();
        $consultGradeEachStud = ($consult
GradeEachStud==0) ? 1 : $consultGradeEachStud;
        $gradesEachStud =
($gradesEachStud==0) ? 1 : $gradesEachStud;
        $grade->percent = number_format
($consultGradeEachStud/$gradesEachStud*100,2).'%';
    }
    return Datatables::of($grades)
->edit_column('EduYear',
'{{$EduYear}}/{{$EduYear+1}}')
->edit_column('NameModule',
'{{$ModuleNum}}. {{$NameModule}}')
->add_column('percent','{{$percent}}')
->add_column('actions','@if (in_
array(Auth::user()->role_id, [4])) <a href="{{$ URL::to(
documents/\'.$DepartmentId.\'\'. $ModuleVariantID.\'
/true/getAllDocumentsDeanery\' )}}}" class="btn
btn-success btn-sm"><span class="glyphicon
glyphicon-pencil"></span> {{ trans("admin/
modules/consulting.getEmptyDoc") }} @else <a
href="{{$ URL::to(
teacher/\'.$DepartmentId.\'\'. $ModuleVariantID .
\'/edit\' )}}}" class="btn btn-
danger btn-sm"><span class="glyphicon glyphicon-
pencil"></span> {{ trans("admin/modules/consulting.
grades") }}</a><a href="{{$ URL::to(
documents/\'. $DepartmentId.\'\'. $ModuleVariantID .
\'/false/
getAllConsultingDocuments\' )}}}" class="btn btn-
success btn-sm"><span class="glyphicon glyphicon-
pencil"></span> {{ trans("admin/modules/consulting.
getEmptyDoc") }}</a>@endif)
->remove_column('ModuleVariantID')
->remove_
column('DisciplineVariantID')
->remove_column('ModuleNum')
->remove_column('DepartmentId')
->make();
    }
}

```

Додаткові переваги, що зумовили використання авторами фреймворку Laravel в процесі розробки IC CSE:

- наявність готових шаблонів (Templates) представлень (Views), що дало можливість авторам швидко створювати необхідні сторінки доступу до даних (рис. 3);

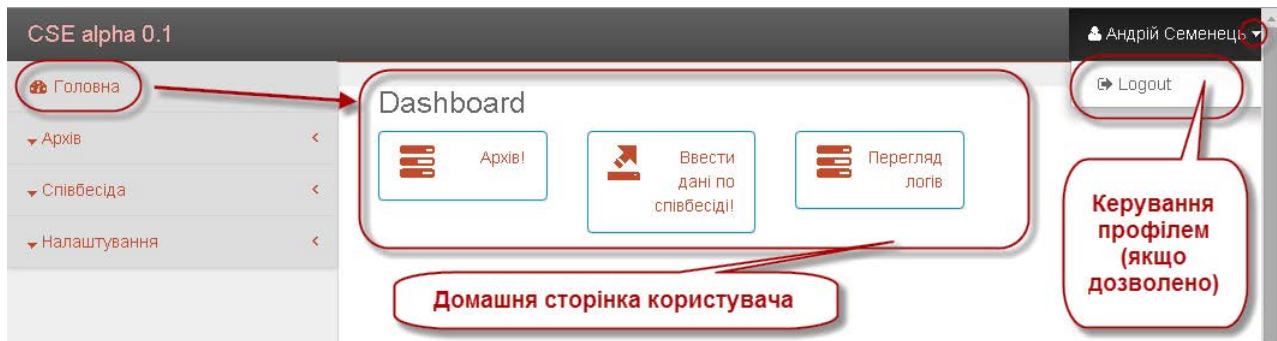
- простота інтеграції з іншими фреймворками, зокрема відмінна сумісність з вищевказаними Twitter Bootstrap та AngularJS, що дозволило авторам ефективно використати в проєкті можливості технологій HTML5, CSS3, Ajax та ін., без надмірних затрат праці і часу;

- вбудовані засоби відлагодження програмного коду та сервісні утиліти.

IC CSE розгорнуто на окремому віртуальному сервері в складі мережного кластера, що використовується в ТДМУ [5], з наступними характеристиками:

- кількість процесорних ядер / ОЗП / HDD – 1/1 Гб / 20 Гб;
- операційна система – Ubuntu Server 14.04;
- конфігурація веб-сервера – Apache 2 (2.4.16) / PHP 5.6.12 / MySQL 5.6.26.

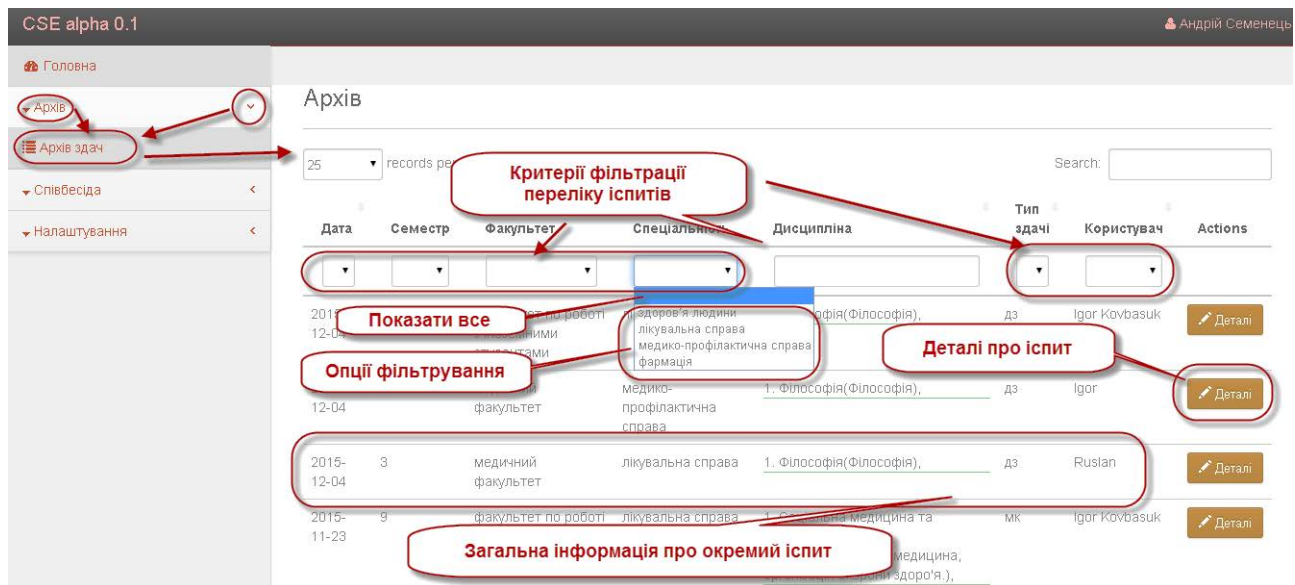
Висновки. Вчасне оновлення, реінжиніринг, рефакторинг та підтримка ПЗ IC в медичній освіті



а)



б)



в)

Рис. 3. Елементи інтерфейсу користувача IC CSE:
 а) домашня сторінка користувача (user dashboard);
 б) діалог вводу балів успішності за відповідь студентів на співбесіді;
 в) сторінка керування переліком записів про іспити (архів).

є обов'язковою умовою забезпечення його ефективного, надійного та безпечного використання таких ІС.

У роботі представлено досвід авторів щодо ре-дизайну та оновлення підсистеми автоматизованої обробки результатів оцінювання тестових робіт

студентів у складі ІСПЗМО. Практичним результатом є розробка окремої ІС – CSE. Вказана ІС реалізована у вигляді веб-додатка, інтегрованого з іншими ІС, що утворюють інформаційну інфраструктуру ТДМУ. ІС CSE забезпечує такі основні функціональні можливості:

– імпорт/експорт та обробку XML-файлів з даними відомостей успішності, для інтеграції з ІС “Контингент”;

– імпорт файлів з даними обробки результатів тестового оцінювання знань студентів;

– оперативного введення даних про результати оцінювання співбесіди студентів;

– встановлення остаточної оцінки за іспит;

– проведення статистичної обробки та аналізу результатів складання іспиту;

– інтеграцію профілів та авторизацію користувачів з використанням єдиного облікового запису на основі профілю Google Apps For Education.

Список літератури

1. Ковальчук Л. Я. Впровадження в навчальний процес комп’ютерних технологій / Л. Я. Ковальчук, В. П. Марценюк // Медична інформатика та інженерія. – 2008. – № 1. – С. 14–16.

2. Марценюк В. П. Розробка і впровадження системи електронного навчання в Тернопільському державному медичному університеті імені І. Я. Горбачевського / В. П. Марценюк // Медична освіта. – 2008. – № 2. – С. 74–75.

3. Семенець А. В. Методи та програмні засоби оцінки знань в медичній освіті : дис. ... канд. техн. наук : 01.05.03 / Семенець Андрій Володимирович. – К., 2011. – 163 с.

4. Марценюк В. П. Концептуальные подходы к структуре информационной системы проверки знаний в медицинском образовании / В. П. Марценюк, А. В. Семенець // Кибернетика и вычислительная техника. – 2009. – Вып. 156. – С. 18–27.

5. Семенець А. В. Концепція побудови інформаційної інфраструктури медичного ВНЗ з використанням вільно-розповсюдженого програмного забезпечення

з відкритим кодом / А. В. Семенець, В. Ю. Ковалок // Інформаційні технології і засоби навчання. – 2014. – № 3. – С. 277–288.

6. Семенець А. В. Адаптація вільно-розповсюдженого ПЗ з відкритим кодом для підтримки навчального процесу в окремому медичному ВНЗ / А. В. Семенець // Медична інформатика та інженерія. – 2013. – № 4. – С. 57–66.

7. Семенець А. В. Застосування хмарних технологій при побудові інформаційної інфраструктури медичного ВНЗ / А. В. Семенець // Медична освіта. – 2014. – № 1. – С. 99–104.

8. Семенець А. В. Про досвід інтеграції хмарного сервісу Google Apps for Education та наявної інформаційної інфраструктури медичного ВНЗ / А. В. Семенець, В. Ю. Ковалок, С. Б. Чеканов // Медична інформатика та інженерія. – 2015. – № 1. – С. 52–60.

9. Семенець А. В. Про підходи до розробки сучасних програмних додатків в медичній освіті / А. В. Семенець, В. П. Марценюк // Медична інформатика та інженерія. – 2014. – № 4. – С. 39–52.

Отримано 12.08.16