

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ СЕМЕНА КУЗНЕЦЯ

Методології розроблення сучасних веб-ресурсів та сервісів
(назва навчальної дисципліни)

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ
до проведення підсумкового контролю
з навчальної дисципліни
підготовки докторів філософії
зі спеціальності 122 «Комп'ютерні науки
та інформаційні технології»

2016 рік

РОЗРОБЛЕНО ТА ВНЕСЕНО: Харківський національний економічний університет імені Семена Кузнеця, кафедра Інформаційних систем, протокол №11 від 05.04.2016.

РОЗРОБНИКИ: д.т.н., проф., Алексієв В.О.

1. ЗАГАЛЬНІ ВІДОМОСТІ

Підсумковий контроль успішності навчання здобувачів з дисципліни «Методології розроблення сучасних веб-ресурсів та сервісів» та рівня сформованості у них компетентностей, які підтримуються даною навчальною дисципліною проводиться у формі семестрового диференційованого заліку відповідно до графіку навчального процесу спеціальності.

Семестровий диференційований залік – форма оцінки підсумкового засвоєння здобувачами теоретичного та практичного матеріалу з окремої навчальної дисципліни, що проводиться як контрольний захід.

Завданням диференційованого заліку є перевірка розуміння здобувачем програмного матеріалу в цілому, логіки та взаємозв'язків між окремими розділами, здатності творчого використання накопичених знань, вміння формулювати своє ставлення до певної проблеми навчальної дисципліни тощо. В умовах реалізації компетентнісного підходу екзамен оцінює рівень засвоєння здобувачем компетентностей, що передбачені кваліфікаційними вимогами.

Семестровий залік охоплює програму дисципліни і передбачає визначення рівня знань та ступеня опанування здобувачами компетентностей. Типовий приклад білету з заліку наведено у розділі «Завдання для підсумкового контролю успішності навчання».

Кожен білет складається з 3 завдань, які передбачають вирішення типових професійних завдань фахівця на робочому місці та дозволяють діагностувати рівень теоретичної підготовки здобувача і рівень його компетентності з навчальної дисципліни.

Екзаменаційний білет включає 1 стереотипне, 1 діагностичне та 1 евристичне завдання, які оцінюються відповідно до Тимчасового положення "Про порядок оцінювання результатів навчання студентів за накопичувальною бально-рейтинговою системою" ХНЕУ ім. С. Кузнеця.

Під час семестрового контролю у формі диференційованого заліку підсумкова кількість балів з навчальної дисципліни (максимум – 100 балів) визначається як сума (проста) балів за результати успішності здобувача при поточному контролі.

Здобувач отримує право на виконання завдань підсумкового контролю (допуск до диференційованого заліку), якщо кількість балів, одержаних за результатами перевірки успішності під час поточного контролю відповідно до змістового модуля впродовж семестру, в сумі досягла 60 балів.

2. ЗАВДАННЯ ДЛЯ ПІДСУМКОВОГО КОНТРОЛЮ УСПІШНОСТІ НАВЧАННЯ

2.1. Зразок білету з диференційованого заліку

Харківський національний економічний університет імені Семена Кузнеця

Освітній рівень доктор філософії

Спеціальність 122 «Комп'ютерні науки та інформаційні технології»
(шифр і назва спеціальності)

Навчальна дисципліна «*Методології розроблення сучасних веб-ресурсів та сервісів*»

ЕКЗАМЕНАЦІЙНИЙ БІЛЕТ № 1

Завдання 1. Дайте розширене визначення терміну «веб-сервіс». Які особливості мають відповідні веб-рішення?

Завдання 2. У середовищі хмарного сервісу Cloud9 розгорніть проект, що застосовує технології мікрофреймворку Slim. Побудуйте каркас веб-сервісу, який будується на технології REST (Для формування первинного каркасу веб-рішення можна застосовувати відкриті проекти та репозиторій GitHub).

Завдання 3. Назвіть для яких сценаріїв розроблення веб-ресурсів чи сервісів доцільно застосовувати монолітну архітектуру веб-додатку?

Затверджено на засіданні

кафедри Інформаційних систем ХНЕУ ім. С. Кузнеця.

Протокол № ___ від " ___ " _____ 20___ р.

Зав. кафедри _____ Викладач _____.
(підпис) (підпис)

2.2. Перелік завдань для підсумкового контролю

Стереотипні завдання:

1. Дайте розширене визначення терміну «веб-сервіс». Які особливості мають відповідні веб-рішення?

2. Дайте розширене визначення терміну «веб-ресурс». Які особливості мають відповідні веб-рішення?

4. Поясніть яким чином виконують класифікацію сучасних веб-орієнтованих проектів та систем?

5. Розкажіть про роль та призначення маршрутизатору у корпоративних мережах. В чому перевага мереж на базі комутаторів?
6. Поясніть особливості застосування технологій хмарних обчислень у інфраструктурі сучасного підприємства.
7. Яким чином застосовують ресурси хмарних обчислень для рішення науково-технічних завдань та розробки веб-ресурсів та сервісів?
8. Наведіть приклади впровадження рішень на базі технологій хмарних обчислень у інфраструктуру дослідних організацій і виконання науково-дослідних та науково-технічних проектів.
9. Наведіть приклади загально доступних ресурсів хмарних обчислень, які можна застосовувати для навчання відповідним технологіям та виконати оцінку якості відповідних сервісів. Які висновки можна зробити щодо доцільності застосування цих ресурсів у комерційній розробці?
10. Наведіть приклади етапів розроблення веб-ресурсу. Поясніть основні програмні рішення та паттерни програмування, що застосовуються у відповідних проектах.
11. Наведіть приклади етапів розроблення веб-сервісу. Поясніть основні програмні рішення та паттерни програмування, що застосовуються у відповідних проектах.
12. Дайте розширене визначення терміну «життєвий цикл інформаційної системи». Проілюструйте відповідь прикладами.
13. Назвіть основні методології щодо організації процесів розроблення веб-ресурсів та сервісів.
14. Надайте розгорнуту характеристику основних можливостей сучасних інструментальних засобів управління проектами.
15. Порівняйте гнучкі (Agile) методології виконання проектів щодо розроблення програмних систем: Scrum та Kanban.
16. Наведіть структурну та функціональну схеми сучасного програмно-апаратного рішення для побудови платформи веб-ресурсу.
17. Наведіть структурну та функціональну схеми сучасного програмно-апаратного рішення для побудови платформи веб-сервісу.
18. Наведіть типову архітектуру серверного рішення, що масштабується, на базі технології Microsoft Windows Server.
19. Наведіть типову архітектуру серверного рішення, що масштабується, на базі технологій Linux-систем.
20. Поясніть патерн програмування MVC. Наведіть макет програми за технологією MVC на мові C#.
21. Поясніть патерн програмування MVC. Наведіть макет програми за технологією MVC на мові PHP.

22. Наведіть приклад застосування PHP-фреймворку для програмування REST-сервісів.
23. Поясніть призначення та технологічні властивості REST-технологій.
24. Поясніть основні патерни програмування. Як вони застосовуються при розробленні веб-ресурсів та сервісів?
25. Яким чином виконують інтеграцію компонент веб-ресурсів? Наведіть типові приклади.
26. Поясніть технологію TDD (розроблення скрізь тестування). Наведіть переваги та недоліки відповідного підходу.
27. Для чого розробляють модульні тести (Unit testes)? Наведіть приклади.
28. Наведіть характеристики сучасних систем управління базами даних, що найчастіше застосовуються у рішеннях для побудови веб-ресурсів та сервісів.
29. Поясніть сопливості технології «Code first», що застосовується у методології реалізованій засобами Microsoft Visual Studio.
30. Поясніть які переваги надають нереляційні бази даних (NoSQL-рішення) порівняно із традиційним підходом застосування реляційних моделей. Як це впливає на імплементацію архітектури веб-ресурсу чи сервісу?

Діагностичні завдання:

31. У середовищі хмарного сервісу Cloud9 розгорніть проект, що застосовує технології мікрофреймворку Slim. Побудуйте каркас веб-сервісу, який будується на технології REST (Для формування первинного каркасу веб-рішення можна застосовувати відкриті проекти та репозиторій GitHub).
32. У середовищі Microsoft Visual Studio розгорніть проект за технологією ASP.NET MVC та сформуєте базовий каркас веб-ресурсу.
33. Застосуйте середовище Microsoft Visual Studio Team Services для рішення завдань супроводу проекту побудови веб-ресурсу.
34. У середовищі хмарного сервісу Cloud9 розгорніть додаток на мові PHP за технологією MVC.
35. Розробіть каркас проекту у середовищі Microsoft Visual Studio та опублікуйте веб-рішення у хмарному сервісі Azure.
36. Виконайте комплекс дій щодо розгортання системи управління змістом веб-ресурсу WordPress у системі хмарних обчислень Red Hat OpenShift.
37. Виконайте розгортання веб-сервісу чи ресурсу на мові PHP у системі хмарних обчислень Red Hat OpenShift.
38. Виконати розроблення веб-інтерфейсу ресурсу із застосуванням фреймворку Bootstrap та технологій JavaScript.

39. Розгорнути у середовищі віртуальних машин систему OpenShift (локальна установка). Поясніть для яких сценаріїв доцільно застосовувати відповідні рішення.

40. Розгорніть у середовищі віртуальних машин систему неперервної інтеграції Jenkins. Поясніть її призначення та наведіть перелік базового функціоналу.

Евристичні завдання:

41. Назвіть для яких сценаріїв розроблення веб-ресурсів чи сервісів доцільно застосовувати монолітну архітектуру веб-додатку?

42. Назвіть для яких сценаріїв розроблення веб-ресурсів чи сервісів доцільно застосовувати мікросервісну архітектуру веб-додатку?

43. Поясніть основні концепції розроблення веб-рішень із застосуванням підходу на основі мікросервісів.

44. Наведіть переваги та недоліки архітектури веб-рішень, що будуються на основі MVC-каркасу веб-додатку.

45. Розкажіть які основні етапи присутні при вирішенні завдань розроблення складних веб-ресурсів та сервісів.

46. Як, на вашу думку, впливає вибір технологій та засобів проектування на ефективність виконання науково-технічних розробок?

47. Наведіть переваги та недоліки застосування Scrum-методології щодо виконання проекту із розробки веб-ресурсу чи сервісу. Чи є відмінності застосування цієї методології для розроблення ве-ресурсу порівняно до веб-сервісу?

48. Наведіть переваги та недоліки застосування Kanban-методології щодо виконання проекту із розробки веб-ресурсу чи сервісу. Чи є відмінності застосування цієї методології для розроблення ве-ресурсу порівняно до веб-сервісу?

49. Наведіть типовий технологічний процес, що не залучає методологію гнучкого (Agile) управління проектом розроблення програмного продукту.

50. Виконайте порівняння підходів щодо розроблення веб-ресурсів та сервісів на технологічних стеках мов C# та PHP.

3. СИСТЕМА ОЦІНЮВАННЯ УСПІШНОСТІ НАВЧАННЯ ПІД ЧАС ПРОВЕДЕННЯ ПІДСУМКОВОГО КОНТРОЛЮ

Результат семестрового диференційованого заліку оцінюється в балах і проставляється у відповідній графі екзаменаційної "Відомості обліку успішності".

Мінімально можлива кількість балів, які здобувач може отримати за результатами проведення підсумкового контролю – 15.

Максимально можлива кількість балів, які здобувач може отримати за результатами проведення підсумкового контролю – 40.

Виконання кожного завдання підсумкового контролю успішності здобувача оцінюється відповідно до Тимчасового положення "Про порядок оцінювання результатів навчання студентів за накопичувальною бально-рейтинговою системою" ХНЕУ ім. С. Кузнеця (табл. 1).

Таблиця 1

Шкала оцінювання: національна та ЄКТС

Сума балів за всі види навчальної діяльності	Оцінка ЄКТС	Оцінка за національною шкалою	
		для екзамену, курсового проекту (роботи), практики	для заліку
90 – 100	A	відмінно	зараховано
82 – 89	B	добре	
74 – 81	C		
64 – 73	D		
60 – 63	E	задовільно	не зараховано
35 – 59	FX	незадовільно	
1 – 34	F		

Підсумкова оцінка за складання диференційованого заліку з навчальної дисципліни «Методології розроблення сучасних веб-ресурсів та сервісів» розраховується за формулою:

$$S = Z1 + Z2 + Z3,$$

де S – підсумковий максимальний бал, який здобувач може отримати за успішне виконання усіх завдань підсумкового контролю (S=40);

Z1 – максимальна кількість балів, яку здобувач може отримати за виконання завдання 1 підсумкового контролю (Z1=10);

Z2 – максимальна кількість балів, яку здобувач може отримати за виконання завдання 2 підсумкового контролю (Z2=15);

Z3 – максимальна кількість балів, яку здобувач може отримати за виконання завдання 3 підсумкового контролю (Z3=15).

Підсумкова оцінка з навчальної дисципліни розраховується з урахуванням балів, отриманих під час диференційованого заліку, та балів, отриманих під час поточного контролю за накопичувальною системою.

Сумарний результат у балах за семестр складає: "60 і більше балів – зараховано", "59 і менше балів – не зараховано" та заноситься у залікову "Відомість обліку успішності" навчальної дисципліни.

4. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

4.1. Основна

1. Шило С.Г. Інформаційні системи та технології : навч. посіб. / С.Г. Шило, Г.В. Щербак, К.В. Огурцова. – Х. : ХНЕУ, 2013. – 219 с.
2. Ушакова, І. О. Проектування інформаційних систем : практикум / Ушакова І. О. – Х. : ХНЕУ ім. С. Кузнеця, 2015. – 234 с.
3. Пушкар О. І. Технології комп'ютерного дизайну : навч. посіб. / О. І. Пушкар. – Х. : ІНЖЕК, 2013. – 166 с.
4. Огурцов В.В. Основи веб та веб-дизайн, програмування на боці клієнта : лаборат. практикум з навч. дисципліни "Веб-технології та веб-дизайн" / В.В. Огурцов. – Х. : ХНЕУ ім. С. Кузнеця, 2015. – 207 с.
5. Глоба Л.С. Розробка інформаційних ресурсів та систем : у 2 т. / Л.С. Глоба // Київ – Т. 1 : Розподілені системи. Поняття розподіленого середовища, Зв'язок, Процеси, Іменування, Синхронізація. – 2013. – 378 с. [Електронний ресурс]. – Режим доступу: [http://www.its.kpi.ua/subjects/56/Documents/Глоба книга Том1.pdf](http://www.its.kpi.ua/subjects/56/Documents/Глоба%20книга%20Том1.pdf).
6. Blackman B. Managing Agile Open-Source Software Projects with Microsoft Visual Studio Online [Electronic resource] / Brian Blackman, Gordon Beeming, Michael Fourie, Willy-Peter Schaub. – Microsoft Press, 2015. – 157 p. – Mode of access: <https://mva.microsoft.com/ebooks>.
7. Dykstra T. Getting Started with Entity Framework 6 Code First using MVC 5 [Electronic resource] / Tom Dykstra, Rick Anderson. – Microsoft Corporation, 2014. – 292 p. – Mode of access: <http://www.asp.net/mvc/overview/getting-started/getting-started-with-ef-using-mvc/creating-an-entity-framework-data-model-for-an-asp-net-mvc-application>.
8. Lockhart J. PHP Вірний Шлях [Electronic resource] / Josh Lockhart. www.phptherightway.com. – 2014. – Mode of access: <https://iflista.github.io/php-the-right-way>.
9. Chacon S. Pro Git [Electronic resource] / Scott Chacon, Ben Straub. Apress, 2014.– 608 p. – Mode of access: <https://git-scm.com/book/uk/v2>.
10. Richardson L. RESTful Web APIs, [Electronic resource] / Leonard Richardson, Sam Ruby. O'Reilly's Open Book Project, 2007. – 448 p. – Mode of access: <http://restfulwebapis.org/rws.html>.

4.2. Додаткова

1. Алексієв В. О. Застосування GRID-технології у транспортному ВНЗ : навч.-метод. посіб. / В. О. Алексієв.– Х. : ХНАДУ, 2008. – 208 с.
2. Методы и модели планирования ресурсов в GRID-системах : монографія / В. С. Пономаренко, С. В. Листровой, С. В. Минухин и др. ; Хар. нац. экон. ун-т. – Х. : ИД "ИНЖЭК", 2008. – 407 с.
3. Методи та моделі розроблення комп'ютерних систем і мереж : монографія / В. С. Пономаренко, С. В. Мінухін, С. В. Кавун та ін. – Х. : Вид. ХНЕУ, 2008. – 315 с.
4. Ньюмен С. Создание микросервисов / С. Ньюмен. – СПб.: Питер, 2016. – 304с.
5. Уоллс К. Spring в действии / К. Уоллс. – М.: ДМК Пресс, 2013. – 752 с.

6. Бэнкер К. MongoDB в действии: пер. с англ. / К. Бэнкер – М.: ДМК Пресс, 2012. – 394 с.
7. Monteiro F. Learning Single-page WebApplication Development 7 / F. Monteiro. – Packt Publishing, 2014. – 214 p.
8. Козловский П. Разработка веб-приложений с использованием AngularJS / П. Козловский, П. Б. Дарвин. – М.: ДМК Пресс, 2014. – 394 с.
9. Чакон С. Git для профессионального программиста / С. Чакон, Б. Штрауб. – СПб.: Питер, 2016. – 496 с.
10. Тарасов С. В. СУБД для программиста. Базы данных изнутри / С. В. Тарасов. – М. : СОЛОН-Пресс, 2015. – 320 с.

4.3. Ресурсы Интернет

1. Настройка среды непрерывного развертывания с помощью Jenkins [Электронный ресурс] / На Лв, Чжао Чжо, Янь Чжэ, Чэнь Сяо Лун. IBM developerWorks, 2015. – Режим доступа : <https://www.ibm.com/developerworks/ru/library/d-continuous-delivery-framework-jenkins/>.
2. Обновленный РНР: Создание виртуальных машин с помощью инструментов Vagrant и PuRHPet [Электронный ресурс] / Эли Уайт. IBM developerWorks, 2015. – Режим доступа : http://www.ibm.com/developerworks/ru/library/wa-php-renewed_4/.
3. Микрослужбы в действии: Введение в микрослужбы [Электронный ресурс] / Рик И. Осовский. IBM developerWorks, 2015. – Режим доступа : <https://www.ibm.com/developerworks/ru/library/cl-bluemix-microservices-in-action-part-1-trs>.
4. Создание и развертывание масштабируемого приложения для управления контактами в облаке [Электронный ресурс] / Викрам Васвани. IBM developerWorks, 2016. – Режим доступа : <http://www.ibm.com/developerworks/ru/library/cl-scalable-contacts-cloud1-app/index.html>.
5. Лекции Технопарка. Проектирование высоконагруженных систем [Электронный ресурс] / Блог компании Mail.Ru Group. – Режим доступа : <http://habrahabr.ru/company/mailru/blog/254843/>.
6. Распределенные базы и хранилища данных : Электронный учебник / Н. Аносова, О. Бородин, Е. Гаврилов и др. – НОУ "ИНТУИТ" [Электронный ресурс]. – Режим доступа : <http://www.intuit.ru/studies/courses/1145/214/info>.
7. Разработка безопасных облачных приложений [Электронный ресурс] / Роби Сен. IBM developerWorks, 2016. – Режим доступа : <http://www.ibm.com/developerworks/ru/library/cl-develop-secure-cloud-aware-applications/index.html>.
8. Облачные стандарты: средства взаимодействия приложений в облаке [Электронный ресурс] / Кэйн Скарлетт. IBM developerWorks, 2016. – Режим доступа : <http://www.ibm.com/developerworks/ru/library/cl-tools-to-ensure-cloud-application-interoperability/index.html>.