

5. Теджасві Р. Платформа Windows Azure / Р. Теджасві, Т. Гвідічі. – Київ: ДМК Пресс, 2012. – 635 с. – (Networking & Cloud Computing).

УДК 37.018.43

ОСОБЛИВОСТІ ОРГАНІЗАЦІЇ ГРУПОВОЇ ВЗАЄМОДІЇ У ДИСТАНЦІЙНИХ КУРСАХ, ПОВ'ЯЗАНИХ З ПРОГРАМУВАННЯМ

Євчук О.В.

*Івано-Франківський національний технічний університет нафти і газу
76018, м. Івано-Франківськ, вул. Карпатська, 15, ktsu@iung.edu.ua*

***Анотація.** Розглянуто способи взаємодії між учасниками дистанційних курсів, пов'язаних з програмуванням. Запропоновано варіанти використання сучасних програмних та організаційних засобів при роботі над завданнями курсу в залежності від типу дистанційного курсу.*

***Abstract.** Interaction methods between e-learning course participants have been considered. Variants of using modern software and organization means are offered for working on course tasks depending on e-learning course type.*

Вступ. Взаємодія між учасниками дистанційного курсу є важливим аспектом його організації, що впливає, з одного боку, на структуру курсу, типи завдань та способи їх перевірки, а з другого – на ефективність засвоєння матеріалу учасниками курсу. Дослідження, виконані в останні десятиліття, підтверджують, що співпраця при навчанні за умови правильної організації суттєво покращує рівень навчальних досягнень, сприяє пошуку більш креативних рішень та збільшує ступінь довготривалого запам'ятовування вивченого матеріалу [1].

Сучасна індустрія розробки програмного забезпечення характеризується значною кількістю великих і складних проектів, що потребують узгодженої роботи у команді та організації ефективної взаємодії між її учасниками. Навіть на невеликих проектах досить часто активно застосовуються сучасні гнучкі методології розробки (Agile) [2], які на перше місце висувають саме співпрацю при роботі над проектами.

Враховуючи, що більшість успішних випускників ІТ-спеціальностей знаходять роботу саме у сфері розробки програмного забезпечення, доцільно при створенні дистанційних курсів, пов'язаних з отриманням відповідних компетенцій, брати до уваги важливість формування також і правильного способу взаємодії при роботі над спільними задачами. Дана проблема має два аспекти: з одного боку, використання відповідних програмних інструментів, що дозволяють ефективно організувати спільну розробку програмного коду (системи контролю версій, використання можливостей сучасних інтегрованих середовищ розробки), з іншого – використання сучасних методів організації роботи над

проектом (Scrum, парне програмування, керована тестами розробка (test-driven development), перегляд коду (code review), неперервна інтеграція тощо).

У традиційних курсах рідко виникає необхідність у використанні подібних інструментів та методів, оскільки основна увага звертається на засвоєння базових понять, розуміння концепцій та розвиток системного мислення у відповідній прикладній сфері. Крім того, більшість завдань у подібних курсах зазвичай не потребують складнішої структури проекту, аніж «один проект – один файл з вихідним кодом», а типовий хід розробки – це повна реалізація задачі за одну ітерацію (можливо, з кількома додатковими ітераціями у процесі відлагоджування), без необхідності тривалої взаємодії з викладачем та іншими учасниками курсу та без явного розбиття на етапи аналізу вимог, розробки структури, кодування та тестування, як це прийнято при виконанні реальних складних проектів. Те саме значною мірою стосується курсів, де програмування використовується лише як засіб реалізації досліджуваних методів математичного моделювання, цифрової обробки сигналів, ефективного кодування тощо. У таких випадках групова взаємодія у дистанційному курсі може зводитись до спілкування у форумах та чатах. Крім того, можна впроваджувати елементи перегляду коду (code review), коли учасник курсу оцінює код іншого учасника на предмет структурованості, зрозумілості, правильного форматування тощо. Приклад організації спільної роботи викладача із студентами в курсі програмування за допомогою спеціально розробленого програмного середовища Virtual Debug Laboratory наведено в [3]. Це онлайн-середовище, розроблене з використанням хмарних сервісів, дозволяє виконувати, зокрема, спільне відлагодження коду – один із найкращих способів отримання навиків розуміння ходу виконання програми. Для студентів, що лише починають вивчати програмування, такий підхід дає можливість якомога раніше навчитись правильно застосовувати засоби відлагодження, які надаються сучасними середовищами розробки, а на подальших етапах – ефективно знаходити і виправляти помилки за участі викладача або інших учасників.

Ознайомлення із найбільш актуальними засобами і технологіями розробки частіше відбувається на старших курсах, коли розглядаються конкретні фреймворки та технології програмування (способи вирішення конкретних прикладних задач, за складністю наближених до реальних). В таких курсах доцільно формувати комплексні завдання, що повинні виконуватись командою студентів на протязі тривалого часу (кілька тижнів чи навіть місяців). В цих випадках зручним є застосування систем контролю версій (Git, SVN тощо), що дозволяють зберігати проект на віддаленому сервері, узгоджено вносити зміни різними учасниками проекту, аналізувати код, написаний іншими учасниками, а для викладача – легко відслідковувати динаміку роботи над проектом, активність окремих учасників та якість їх роботи. Приклад безкоштовної системи для розміщення програмних проектів – GitLab (Community Edition) [4], сайт та система керування репозиторіями програмного коду для Git, де можна створювати закриті репозиторії з різними додатковими можливостями (багтрекер, логування часу, вікі-довідник тощо).

Що стосується застосування методик на зразок Scrum, Kanban та подібних технологій ітеративної розробки програмного забезпечення, то необхідність їх у університетських курсах, особливо дистанційних, не є однозначною. Дані технології спрямовані на отримання максимально

робочого прототипу програми за мінімальний проміжок часу, із постійним плануванням коротких циклів розробки, на кожному із яких додається невелика кількість завершених функціональних можливостей. Важливу роль тут відіграють планування із врахуванням пріоритетності задач та постійна робота із замовником – суто організаційні заходи, що дозволяють підвищити продуктивність на реальних проєктах, однак не є необхідними в навчальних проєктах. Разом з тим є доцільним ознайомлення з такими методиками і їх практичне застосування в рамках окремих дисциплін, що вивчають методи управління та групової взаємодії у програмних проєктах.

Висновки. У дистанційних курсах, пов'язаних з програмуванням, доцільно використовувати програмні засоби та елементи організаційних методик, актуальні для сучасної програмної індустрії.

Використані літературні джерела

1. Ruth C. Clark, Richard E. Mayer. e-Learning and the Science of Instruction: Proven Guidelines for Consumers and Designers of Multimedia Learning. - John Wiley & Sons, 2016. – 528pp.
2. Agile-маніфест розробки програмного забезпечення / [Електронний ресурс] <http://agilemanifesto.org/iso/uk/manifesto.html>
3. Q. Ding, S. Cao. RECT: A Cloud-Based Learning Tool for Graduate Software Engineering Practice Courses With Remote Tutor Support / IEEE Access, vol. 5, 2017, pp. 2262-2271
4. Features | GitLab / [Електронний ресурс] <https://about.gitlab.com/features/>

УДК 622.276.53:621.67(047)

СУЧАСНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ЯК ОСНОВА ВИБОРУ ДІАГНОСТИЧНИХ ОЗНАК СТАНУ ШГНУ

Заміховський Л.М., Романюк А.Б.

*Івано-Франківський національний технічний університет нафти і газу
76019, м. Івано-Франківськ, вул. Карпатська, 15, delete@meta.ua*

Анотація. Проведено аналіз методів діагностування ШГНУ і можливостей використання сучасних інформаційних технологій для вибору діагностичних ознак їх стану з точки зору використання як функціональної складової системи автоматичного управління процесом періодичної експлуатації нафтових свердловин.

Abstract. The analysis methods of diagnosing Sucker-Rod Pump and utilization of modern information technology to select the diagnostic signs of their condition in terms of use as a functional component of the automatic process control of the periodic operation of oil wells.