

Конюхов С.Л.,
*старший викладач кафедри інформатики і кібернетики,
Мелітопольський державний педагогічний університет
імені Богдана Хмельницького*

ЗАСТОСУВАННЯ ПРОЕКТНОГО ПІДХОДУ У ПРОЦЕСІ ПРОФЕСІЙНОЇ ПІДГОТОВКИ ПРОГРАМІСТІВ У ВНЗ

Постановка проблеми. ІТ-індустрія – найбільша галузь світової економіки, у якій працюють і від продуктів якої залежать мільйони спеціалістів. До якості і надійності програмних продуктів сьогодні висуваються високі вимоги, але проблема їхнього забезпечення наразі остаточно не вирішена [7, с. 319]. Серед причин такої ситуації дослідники називають недостатній рівень підготовленості програмістів до реальної практики діяльності підприємств галузі [2, с. 73].

Наразі існує розрив між рівнем підготовки випускників вишів і вимогами з боку роботодавців, які зауважують, що випускникам вишів бракує умінь формувати мету програмного проекту, виконувати колективну роботу і підтримувати комунікації, своєчасно завершувати розробку і впровадження програмних продуктів [1, с. 3].

Як зазначають дослідники, сучасний програміст повинен не лише знати теоретичні основи процесу розробки програмного забезпечення (ПЗ), володіти різними мовами програмування, методами конструювання алгоритмів, проектування і розробки складних програмних систем, а й вміти використовувати методи і засоби, прийняті у великих, територіально розподілених і неоднорідних за складом колективах розробників [2, с. 73].

Отже, професійна підготовка фахівців для галузі програмування є актуальною проблемою, розв'язання якої сприятиме підвищенню рівня інформатизації українського суспільства, а також збільшенню українського сегмента на світовому ринку програмного забезпечення [5, с. 87].

Одним із шляхів вирішення завдання щодо підвищення якості професійної підготовки програмістів у ВНЗ, формування у них базового комплексу компетентностей, необхідного для успішного початку діяльності за фахом, є застосування проектного підходу.

Аналіз актуальних досліджень. Теоретико-методологічні засади професійної підготовки ІТ-фахівців обґрунтовані у працях Р.М. Горбатюка, Л.В. Гришко, М.Г. Коляди, В.В. Осадчого, З.С. Сейдаметової, М.І. Лазарева й ін.; особливості побудови процесу навчання програмістів на основі компетентнісного підходу висвітлені у дослідженнях М.О. Вінника, Ф.С. Ільясової, В.С. Круглика, В.Є. Седова й ін.

Теоретико-методологічні засади впровадження проектної технології у навчальний процес представлені у роботах Н.В. Борисової, Т.В. Качеровської, О.М. Пехоти, Г.М. Романової, С.О. Сисоєвої й інших вітчизняних дослідників. Напрями і способи використання проектного підходу в якості методу навчання майбутніх програмістів у ВНЗ досліджували А.О. Афонін [2], О.Ф. Баранюк [1], С.О. Борозенний [2], В.В. Бублик [2], М.О. Вінник, В.С. Круглик [10], С.О. Лещук [4], В.Є. Седов [8], А.М. Стрюк [9] й ін.

Разом із тим, залишається актуальним завдання щодо розробки нових способів застосування проектного підходу у процесі професійної підготовки майбутніх програмістів у ВНЗ, зорієнтованих на формування у них компетентностей з об'єктно-орієнтованої методології розробки ПЗ (об'єктно-орієнтованого аналізу, проектування і програмування).

Мета статті. Обґрунтувати доцільність і методику використання проектного підходу у процесі вивчення об'єктно-орієнтованої методології розробки програмного забезпечення.

Виклад основного матеріалу. Проектний підхід передбачає організацію самостійної діяльності учнів, розв'язання ними певної проблеми з використанням різних методів і засобів, створення і презентацію «продукту» (розв'язку теоретичної проблеми або результату, готового для впровадження) [6, с. 150].

Зазначені особливості обумовлюють доцільність застосування проектної технології у вищих навчальних закладах з метою наближення умов освітнього середовища до реальних виробничих процесів у галузі інформаційних технологій. Автори роботи [2] слушно зазначають, що наявність досвіду виконання групового програмного проекту сприяє більш успішній побудові професійної кар'єри, тому таке практичне тренування можна розглядати як ключовий фактор у підготовці кваліфікованих фахівців. Ефективність впровадження проектних технологій у навчальний процес підвищується завдяки залученню промислових партнерів, що дозволяє наблизити навчальний процес до потреб практики. Така взаємодія сприяє посиленню готовності до групових проектів як студентів, так і викладачів. [2, с. 74].

Наприклад, результати впровадження проектного підходу у процесі навчання ІТ-фахівців представлені у статті С.О. Лещука [4]. Під час виконання програмного проекту для організації колективної роботи студентів була використана методологія Scrum. Окрім того, вони набували досвіду застосування принципів об'єктно-орієнтованого програмування, систем контролю версій, програмного інструментарію web-розробки, архітектурного шаблону MVC [4, с. 83-84].

Професійна підготовка програміста з об'єктно-орієнтованого програмування (ООП) включає оволодіння знаннями і навичками практичного застосування об'єктно-орієнтованого аналізу, об'єктно-орієнтованого проектування, шаблонів проектування, об'єктно-орієнтованих даних (XML), уніфікованої мови моделювання UML, методів об'єктно-орієнтованої web-розробки (включаючи мобільну), об'єктно-орієнтованих мов програмування й ін. [3, с. 15].

Спектр можливостей ООП не обмежується названими питаннями, тому перед викладачем ВНЗ постає завдання щодо застосування методів і засобів навчання, які допоможуть студентам опанувати на достатньому рівні об'єктно-орієнтованими засобами розробки.

У процесі вивчення майбутніми програмістами об'єктно-орієнтованої методології розробки ПЗ проектний підхід може використовуватися на різних етапах: 1) на початку ознайомлення з фундаментальними основами ООП; 2) у процесі вивчення об'єктно-орієнтованих засобів окремих мов програмування (C++, C#, Java, JavaScript, PHP, Delphi, Python тощо); 3) під час вивчення

методології проектування програмних систем, що передбачає застосування методів об'єктно-орієнтованого аналізу і проектування.

Перші два типи проектів є монодисциплінарними і виконуються у процесі вивчення певної дисципліни з метою формування практичних умінь у її межах. Проекти третьої групи є міждисциплінарними і мають на меті узагальнення знань з об'єктно-орієнтованої методології розробки, а також набуття досвіду колективної роботи над проектом. Міждисциплінарні проекти доцільно використовувати на останньому році підготовки програмістів у бакалавріаті, коли студенти вже мають комплекс знань і практичних умінь.

Для виконання такого проекту створюється колектив розробників, до складу якого входять керівник, кодери і тестувальники. Розподіл обов'язків між зазначеними посадами: 1) керівник розробляє об'єктну модель предметної області, визначає об'єкти і правила взаємодії між ними, визначає основні публічні методи класів, готує опис форматів вхідних і вихідних даних, розподіляє доручення між іншими членами колективу; 2) кодери розробляють програмні модулі на основі проекту, створеного керівником, виправляють знайдені помилки, виконують об'єднання модулів у готовий програмний продукт; 3) тестувальники готують набори тестів для перевірки функціонування окремих програмних модулів, а також готової програми.

Для реалізації таких програмних проектів пропонуються такі напрями розробки: створення web-додатків, створення мобільних додатків, створення розподілених desktop-додатків. Кожний із зазначених напрямів передбачає поділ команди кодерів на дві групи – розробників серверного і клієнтського програмного забезпечення. Наприклад, під час створення web-додатків серверна частина може розроблятися з використанням мови PHP, а клієнтська – JavaScript; у випадку мобільних додатків серверна частина може розроблятися з використанням мов Java (PHP, C#), а клієнтська – Java.

Організувати колективну роботу, яка великою мірою є самостійною, доцільно з використанням спеціалізованих web-орієнтованих технологій, зокрема: для проектування класів, побудови діаграм зв'язків, проектування інтерфейсу користувача, побудови сценаріїв взаємодії учасників системи (<https://www.invisionapp.com/>, <https://moqups.com/>); для керування проектом, моніторингу виконання задач та їх обговорення, відстеження помилок та їх виправлення (<http://www.redmine.org/>); для спільної роботи над кодом з можливістю централізованої перевірки і контролю версій (<https://github.com/>).

Команда розробників міждисциплінарного проекту може складатися зі студентів різних курсів. Наприклад, до розробки web-додатку можуть бути залучені студенти четвертого курсу (роль – керівник проекту), які вивчають дисципліни «Об'єктно-орієнтований аналіз і проектування» або «Проектування програмних систем»; третього курсу (роль – кодер серверної частини проекту), які вивчають дисципліну «Програмування та підтримка веб-застосувань»; другого курсу (роль – кодер клієнтської частини проекту), які вивчають дисципліну «Програмування». Така організація проектної діяльності є більш складною, але більш цікавою і корисною для студентів. У процесі навчання студент може бути учасником декількох проектних команд і пройти різні професійні етапи: від програміста-початківця з найнижчою кваліфікацією до керівника власного проекту.

Висновки та перспективи подальших досліджень. У статті висвітлені деякі

проблеми професійної підготовки майбутніх програмістів у вищих навчальних закладах, обґрунтовано доцільність застосування проектного підходу з метою підвищення якості вищої освіти, запропоновані способи використання програмних проектів для формування у студентів компетентностей з об'єктно-орієнтованої методології розробки програмних систем.

Слід відзначити, що з метою підвищення інтересу студентів до професії програміста, ознайомлення їх з реальними процесами програмної розробки, а також заохочення до активної колективної роботи доцільно організовувати міждисциплінарні проекти, у процесі виконання яких вони мають змогу спробувати різні напрями діяльності за фахом.

Подальші дослідження необхідно спрямувати на розробку методичних настанов організації, проведення і оцінювання результатів конкретних програмних проектів.

Список використаних джерел та літератури

1. Баранюк О.Ф. Проблемно-орієнтоване навчання у програмній інженерії / О.Ф. Баранюк // Наукові записки [Кіровоградського державного педагогічного університету ім. В. Винниченка]. Серія: Проблеми методики фізико-математичної і технологічної освіти. – 2016. – Випуск 9. – С. 3-10.
2. Бублик В.В. Особливості впровадження навчальної групової розробки програмних систем / В.В. Бублик, А.О. Афонін, С.О. Борозенний // Наук. зап. НаУКМА. Сер. Комп'ют. науки. – 2008. – Т. 86. – С. 73-77.
3. Вайсфельд М. Объектно-ориентированное мышление / М. Вайсфельд. – СПб.: Питер, 2014. – 304 с.
4. Лещук С.О. Окремі методичні аспекти підготовки ІТ-фахівців / С.О. Лещук // Інформаційні технології в освіті. – 2017. – № 1(30). – С. 81-96. – DOI: 10.14308/ite000621.
5. Осадчий В.В. Досвід розробки та апробації навчального курсу «Вступ до спеціальності» (інженер-програміст) / В.В. Осадчий, К.П. Осадча // Проблеми інженерно-педагогічної освіти. – 2014. – №42-43. – С. 87-93.
6. Освітні технології: Навч.-метод. посіб. / О.М. Пехота, А.З. Кіктенко, О.М. Любарська та ін.; За ред. О.М. Пехоти. – К.: А.С.К., 2001. – 256 с.
7. Поморова О.В. Сучасні проблеми оцінювання якості програмного забезпечення / О.В. Поморова, Т.О. Говорущенко // Радіоелектронні і комп'ютерні системи. – 2013. – № 5 (64). – С. 319-327.
8. Седов В.Є. Формування фахової компетентності майбутніх інженерів - програмістів в умовах магістратури: дис. ... канд. пед. наук: 13.00.04 – теорія та методика професійної освіти / Седов Віктор Євгенович. – Херсон, 2016. – 238 с.
9. Стрюк А.М. Теоретико-методичні засади комбінованого навчання системного програмування майбутніх фахівців з програмної інженерії : монографія / А.М. Стрюк // Теорія та методика електронного навчання. – Кривий Ріг: Видавничий відділ ДВНЗ «Криворізький національний університет», 2015. – Том VI. – Випуск 1 (6): спецвипуск «Монографія в журналі». – 286 с.
10. Kruhlyk V. Formation of professional competences of future engineer programmers in the process of independent educational activity / V. Kruhlyk // Інформаційні технології в освіті. – 2017. – No 2(31). – pp. 55-68. – DOI: 10.14308/ite000633.