

УДК 657:330.3:005.342

ЗАСТОСУВАННЯ АКТИВНИХ МЕТОДІВ НАВЧАННЯ ПІД ЧАС ОЦІНЮВАННЯ ЯКОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Скорін Юрій

Харківський національний економічний університет ім. Семена Кузнеця

Сьогодення характеризується тим, що під час здійснення певних освітніх процесів, які потребують того, щоб фахівців з галузі розробки та тестування програмного забезпечення було краще підготовлено. Використання активних форм навчання під час вивчення фахових дисциплін, особливо таких, як технології тестування програмного забезпечення, має певні причини. Важливо, щоб здобувачі не тільки отримували знання, але й вміли їх застосовувати в реальних ситуаціях. Існують різноманітні форми навчання, які допомагають активізувати навчальний процес: ділові ігри, імітація роботи з програмним забезпеченням, круглі столи тощо. Вони стимулюють активну взаємодію між здобувачами та викладачами. З метою значної активізації навчання у вищому навчальному закладі важливо знати особливості навчання, які вимагають передачі студентам нових навичок, вмінь і знань у пізнавальній діяльності [2]. Характерні ознаки активного навчання [1]: стимулювання інтересу до вивчення матеріалу та вміння застосовувати набуті знання під час виконання практичних завдань; активне участь студентів у навчальному процесі, що дає змогу розвивати їхні пізнавальні та творчі здібності під час розв'язування завдань; самостійне розв'язання проблем, підвищення емоційної мотивації здобувачів до вивчення обговорюваних питань; інтерактивність, тобто постійна взаємодія між учнями, вільне обговорення різноманітних способів вирішення ситуацій.

Активні методи навчання допомагають вирішити такі навчальні завдання: зробити процес навчання під керівним впливом викладача; забезпечити вплив на активну участь як підготовлених, так і непідготовлених студентів; постійно контролювати вивчення матеріалу та його якість [2].

Використання активних методів змінює процес навчання — він стає не лише цікавішим, але й ефективнішим. Пізнавальна активність зростає, розвивається критичне мислення, формуються здатність приймати творчі рішення під час підготовки видання, його маркетингу та інших процесів. Під час застосування активних методів навчання виникає, так звана, примусова активність студентів, коли вони не можуть залишатися пасивними, активно взаємодіють із викладачем. Це підвищує мотивацію, емоційність та творчість. Викладач постійно взаємодіє з учнями, допомагає формувати їх професійні, інтелектуальні уміння та навички. Активні методи навчання дозволяють розвивати такі здібності: здатність адаптуватися в групах, чітко висловлювати свої думки, формулювати ідеї, проекти, передбачати наслідки вибору; уміння встановлювати особисті контакти, обмінюватися інформацією, ефективно керувати своєю діяльністю та часом, брати на себе відповідальність за результати групи, приймати нестандартні рішення [2]. Для вирішення вказаних завдань під час організації самостійної роботи студентів доцільно застосовувати підходи, які використовуються в промисловій розробці програмного забезпечення, а сьогодні вони пов'язані з поняттям "тестування програмних продуктів". У такому підході застосовуються активні форми навчання, при цьому імітується процес розроблення програмного забезпечення, який використовується в ІТ-галузі. Тобто, навчальний процес наближається до реальних умов ринку праці [2]. Щоб описати цю модель процесу навчання, потрібно ввести кілька понять.

1. Учасник процесу навчання (учасниками є здобувач і викладач).
2. Роль – це є сукупність дій і функцій, які виконуються учасниками. У процесі навчання, коли використовуються методи програмної інженерії, здобувач і викладач можуть заходити в різні ролі. Сама і та ж роль може виконуватися як здобувачем, так і викладачем.
3. Робочий продукт – це є результат роботи учасника в певній ролі.

Оскільки головним об'єктом у даній моделі є завдання, пов'язане з розробленням програмного продукту, який створюється разом, потрібно ввести такі ролі учасників даного спільного навчального процесу.

1. Автор, його завданням є – поставити завдання для виконання здобувачем.

2. Модератор, його завданням є – призначити завдання виконавцям, контролювати процес виконання завдання та прийняти остаточне рішення про його виконання.

3. Кодувальник, його завданням є – написати програмний код для вирішення завдання, яке було призначене йому.

4. Інспектор, його завданням є – аналіз коду програми з метою перевірки її правильності та якості написання, а також відповідності використаних програмних рішень умовам завдання. Інспектор може виписати один із трьох вердиктів: а) прийняти завдання без зауважень – це означає, що інспектор повністю зрозумів всі програмні рішення, код написано правильно, текст програми відповідає всім необхідним коментарям; б) прийняти завдання з урахуванням зауважень – це означає, що програма загалом правильно написана, але у неї є деякі незрозумілі або некоректні місця. Інспектор формулює свої зауваження, а потім перевіряє завдання знову після їх усунення кодувальником. Процес усунення зауважень триває, доки інспектор не видає вердикт "прийняти без зауважень"; в) відхилити завдання – це означає, що програма не виконує потрібних функцій або кількість зауважень є такою, що її не можна прийняти без суттєвих змін.

5. Тестер, його завданням є – розробити тест, який дозволить перевірити якість написання програми шляхом пошуку помилок, а також сама перевірка. Робота тестера починається після того, як інспектор видає вердикт "прийняти без зауважень". У цьому разі тестер не аналізує код, він лише запускає програму на тестових сценаріях, які відповідають умовам завдання.

Робочими елементами цієї моделі навчального процесу є: у автора – визначення завдання; у модератора – оцінка роботи виконавця; у

кодувальника – написання програми (текст); у інспектора – зауваження та рекомендації щодо вдосконалення коду; у тестера – сценарій тестування.

Послідовність виконання роботи з програмою: автор формулює завдання, після чого передає його кодувальнику; модератор визначає ролі; кодувальник пише код програми і передає його інспектору; інспектор аналізує код, якщо виявляється помилка, повідомляє про це кодувальнику, і вони разом виправляють помилку до того, як інспектор вирішить, що код прийнятний без зауважень; після цього код передається тестеру, який розробляє сценарій тестування і перевіряє якість програми; модератор оцінює якість програми і дійшов рішення щодо результатів роботи. У табл. 1 наведено приклад розподілу ролей у традиційній формі навчального процесу.

Таблиця 1 – Розподіл ролей між учасниками освітнього процесу за умови традиційної форми навчання

Роль	Учасник	Функція
Автор	Вчитель	Формулювання завдання
Ведучий	Вчитель	Розподіл завдань між здобувачами (розподіл завдань)
Кодер	Здобувач	Написання програмного коду
Інспектор	Вчитель	Перевірка програмного коду, оцінювання робіт здобувачів
Тестер	Вчитель	Запуск програми та її тестування, оцінка роботи здобувача

Отже, у традиційному варіанті, коли роль автора, модератора, інспектора та тестера виконує викладач, а студент виконує лише роль кодувальника. Використання такого підходу сьогодні відповідно до навчання студентів, які навчаються на напрямках, які не пов'язані безпосередньо з інформаційними технологіями. У табл. 2 наведено розподіл ролей для варіанта технології навчання, який використовує методи програмної інженерії.

Таблиця 2 – Розподіл ролей для варіанта технології навчання, який використовує методи програмної інженерії

Роль	Учасник	Функція
Автор	Вчитель	Формулювання завдання
Ведучий	Вчитель	Розподіл завдань між здобувачами (розподіл завдань)
Кодер	Здобувач 1	Написання програмного коду
Інспектор	Здобувач 2	Перевірка програмного коду, оцінювання робіт здобувачів
Тестер	Здобувач 3	Запуск програми та її тестування, оцінка роботи здобувача

У варіанті, що наведено в табл. 2, викладач виконує роль автора і модератора, а ролями кодувальника, інспектора та тестера виступають здобувачі. Викладач формулює завдання на певну тему, розподіляє ролі серед студентів, контролює процес їх роботи і оцінює результат. Добувач-кодувальник пише програмний код і передає його добувачеві-інспектору. Інспектор аналізує код, формулює відповідні зауваження й (якщо це потрібно) повертає його кодувальнику для вдосконалення. Якщо зауважень немає, код передається добувачеві-тестеру. Тестер створює сценарій тестування й перевіряє програму згідно з ним, формулюючи результат — "пройдено" або "не пройдено" для кожного пункту. Після завершення роботи тестера викладач виставляє оцінки кожному учаснику. Використання цієї технології під час вивчення навчальної дисципліни, особливо під час організації самостійної роботи здобувачів, дає можливість підвищити ефективність навчального процесу, допомагає здобувачам отримувати навички у тестуванні програмного забезпечення. Крім того, це сприяє зростанню інтересу здобувачів до вивчення означеної дисципліни, ознайомлює їх з способами колективної роботи, які застосовуються у професійній діяльності, та забезпечує прозорість процесу оцінювання [2]. Активні методи навчання саме в цьому допомагають здійснити особистий і професійний зростання студентів й досягти високих результатів у галузі тестування програмного забезпечення.

Література:

1. Активна методика навчання у підготовці фахівців. URL: <http://www.irbis-nbuv.gov.ua>.
2. Скорін Ю.І. Технології тестування програмних продуктів: методичні рекомендації для самостійної роботи студентів спеціальності 122 «Комп'ютерні науки та інформаційні технології» першого (бакалаврського) рівня/ Ю.І. Скорін. – Харків: ХНЕУ. С. Кузнеця, 2017. – 35 с.