

УДК 378.147:005.94:004.03

**Гарко Ірина Ігорівна**

кандидат фізико-математичних наук,  
доцент кафедри прикладних інформаційних систем факультету інформаційних технологій  
Київського національного університету імені Тараса Шевченка, м. Київ, Україна  
ORCID ID 0000-0003-0671-6336  
*garko.iryana@gmail.com*

**Пирог Микола Володимирович**

асистент кафедри прикладних інформаційних систем факультету інформаційних технологій  
Київський національний університет імені Тараса Шевченка, м. Київ, Україна  
ORCID ID 0000-0003-2588-6066  
*mykola.pyroh@yahoo.com*

**Міронова Вікторія Леонідівна**

кандидат технічних наук, доцент,  
доцентка кафедри прикладних інформаційних систем факультету інформаційних технологій  
Київський національний університет імені Тараса Шевченка, м. Київ, Україна  
ORCID ID 0000-0002-0878-0967  
*vicky.mironova@gmail.com*

## ЗАСТОСУВАННЯ AGILE-МЕТОДОЛОГІЇ У НАВЧАННІ АЛГОРИТМІЗАЦІЇ І ОСНОВ ПРОГРАМУВАННЯ СТУДЕНТІВ СПЕЦІАЛЬНОСТІ КОМП'ЮТЕРНІ НАУКИ

**Анотація.** У представленій статті розглядається результат імплементації гнучкої методології розробки програмного забезпечення – Agile – у процес навчання студентів закладу вищої освіти. Нами представлено підходи до модернізації освітнього процесу, розроблене навчально-методичне забезпечення та підходи до викладання у вищій школі на прикладі навчальної дисципліни «Алгоритмізація і основи програмування» для студентів спеціальності 122 «Комп'ютерні науки» освітньої програми «Прикладне програмування». Проаналізовано широкий спектр публікацій, що окреслюють проблематику розроблення нових методологій викладання, зокрема проблем підготовки фахівців технічних спеціальностей у реаліях трансформації інформаційного суспільства та інформатизації навчального процесу. Проведено аналіз ринку інформаційних технологій та запити провідних компаній на ринку України, а також досліджено рефлексії випускників щодо ефективності навчання в сучасних закладах вищої освіти. У статті описано результати кількарічної поступової адаптації технології Agile до навчального процесу, її введення в навчальний процес та результати апробації розробленої методології. Розглянуто теоретичні основи та переваги Agile-методології над класичною схемою навчання, що визначається нами як «пряма» або «каскадна». З метою ефективного дослідження впливу гнучких методологій на освітній процес представлено підходи до формування експериментальних та контрольних академічних груп студентів. Наведено динаміку виконання лабораторних робіт та порівняння підсумкових результатів навчання академічних груп. Розглянуто вплив Agile-методології на успішність академічних груп та проведено статистичне порівняння з класичним підходом до викладання. Визначено напрями подальших досліджень теми імплементації Agile-методології в освітній процес з огляду на актуальність питання трансформації освіти в Україні та професійної підготовки фахівців інформаційно-технічної галузі.

**Ключові слова:** Agile-методологія; Scrum; освіта; інформаційні технології; підготовка фахівців; професійна підготовка.

## 1. ВСТУП

**Постановка проблеми.** Стрімкий розвиток інформаційних технологій (далі – ІТ) та їх імплементація та впровадження в усі сфери життя суспільства ставить перед вищою освітою питання про адаптивність у підходах до підготовки майбутніх ІТ-фахівців. Розглядаючи проблематику підготовки фахівців сфери ІТ, варто відзначити складнощі в традиційних, усталених методах та методологіях навчання в закладах вищої освіти [1], [2]. На нашу думку, основний аспект даної проблеми полягає саме в прямолінійності та чіткій, послідовній схемі викладання та трактування матеріалів навчальних курсів, що нівелює можливості до адаптивного підходу у викладанні. Тож за час підготовки студента викладений матеріал та методи опанування необхідних навичок застарівають. Для розуміння необхідності адаптації до таких змін ми рекомендуємо оцінити періодичність, з якою виходять оновлення стандартів з мов програмування. Розглянемо мови, що використовуються під час вивчення дисципліни «Алгоритмізація і основи програмування» та в цілому на перших двох курсах. Після аналізу документації та історії мов програмування матимемо наступні результати: 3 роки – для мови C++, 1-3 роки – для мови Java, та від пів року до 3 років – для мови C#.

Для розв'язання даної проблеми пропонуємо впровадження в навчальний процес адаптивного підходу на основі Agile-методології [3], що має сприяти розвитку у майбутніх ІТ-спеціалістів навичок до швидкого опанування нового матеріалу та гнучкості на швидкоплинному ринку ІТ-індустрії. Згідно підходу Agile, програмне забезпечення створюється інкрементально з самого початку проєкту, водночас пріоритетні завдання вирішують у межах коротких ітерацій, розбиваючи весь проєкт на невеликі робочі шматочки.

**Аналіз останніх досліджень і публікацій.** Дослідженню та розробленню методологій у навчальному процесі присвячено роботи як багатьох педагогів, так і спеціалістів галузі ІТ, оскільки питання якісної підготовки відповідних кадрів гостро поставлено в Україні [4], [5]. Як особливо актуальні та ґрунтовні праці можна відзначити праці І.О. Бардус [6], М.О. Дудки [7] та П.М. Малежика [8]. Окрім робіт фахових науковців та педагогів проаналізовано публікації підприємців та потенційних роботодавців [5], [9], а також безпосередньо студентства з рефлексіями щодо питання реалізації навчального процесу [10], [11].

Вищезгадані роботи висвітлюють проблемні аспекти професійної підготовки фахівців у галузі ІТ, певні шляхи їх вирішення та підходи до організації навчального процесу в закладах вищої освіти. Проте більшість з них вказують на необхідність комплексного підходу до вивчення фундаментальних дисциплін, що не сприяє гнучкості навчального процесу та адаптивності майбутнього спеціаліста.

Світовий досвід також вказує на те, що питанням використання Agile-методології в навчальному процесі присвячено багато дослідницьких робіт. Зокрема в роботі [12] автори стверджують, що навчання та розробка програмного забезпечення є більш ефективними, коли реакція на зміни розглядається як частина процесу, без непохитного дотримання певного розкладу. Водночас навчальна програма складається з різних за обсягом проєктів (від малих до великих), і найбільша увага приділяється не вивченню теоретичного матеріалу, а конкретно здобуттю досвіду студентами під час здійснення спільної роботи над цими проєктами. Крім того, у даній роботі детально описано, у який спосіб можна кожен з принципів Agile перенести в освітній процес.

У роботі [13] представлено перші результати реального впровадження Agile-методології в навчальний процес у межах курсу з розробки програмного забезпечення, що були отримані на основі даних за три навчальних роки.

Варто також відзначити роботу [14], у якій описано методології, що розширюють Agile та водночас зберігають її принципи, а саме – Scrum та eXtreme Programming. У даній роботі, окрім безпосереднього опису наведених вище методологій, розглянута специфіка їх впровадження в освітній контекст.

Водночас, варто зазначити, що Agile-методологія знаходить своє застосування не лише при вивченні курсів, безпосередньо пов'язаних з розробкою програмного забезпечення, а є ефективною також при вивченні інших дисциплін, зокрема дискретної математики [15].

**Метою статті** є аналіз результатів впровадження Agile-методології у навчання студентів технічних спеціальностей у закладах вищої освіти та порівняння з класичною методологією викладання. А також представлено розвиток розробленої методології та її вплив на навчальний процес на прикладі вивчення дисципліни «Алгоритмізація і основи програмування» для студентів освітньої програми «Прикладне програмування» спеціальності 122 «Комп'ютерні науки».

## 2. ТЕОРЕТИЧНІ ОСНОВИ ДОСЛІДЖЕННЯ

У попередніх роботах [1], [2] було досліджено проблему каскадності в побудові навчальних програм та підходів до викладання дисциплін згідно з такими програмами. Враховуючи стрімкі зміни на ринку ІТ та тенденції щодо змін у попиті на знання мов програмування, нами була розроблена більш адаптивна робоча навчальна програма дисципліни «Алгоритмізація і основи програмування» для студентів спеціальності 122 «Комп'ютерні науки» освітньої програми «Прикладне програмування». Імплементуючи в навчальний процес засади Agile Education [16] та власні напрацювання, нами було розроблено методику викладання згаданої дисципліни, що, не порушуючи структуру та норми закладені в Київському національному університеті імені Тараса Шевченка [17], дозволяла б зробити процес навчання більш адаптивним.

Модель послідовного викладання матеріалу вже віджила своє в епоху доступу до інформації, і завданням викладачів стає не примітивна вчитка матеріалу, а формування уявлення про гнучкість процесів, що у світлі розробки програмного забезпечення називають Agile-принципами. Згідно з Agile-методологією та прагненням авторів наблизити виконання лабораторних робіт студентами до робочих реалій, що існують та розвиваються в компаніях ІТ сфери, було переглянуто класичну схему представлення та виконання лабораторних робіт у вигляді каскаду (рис. 1) на користь більш розгалуженої схеми (рис. 2), що ґрунтується на принципах гнучкої методології розроблення програмних продуктів: «необхідно зробити, потрібно зробити, можна зробити» (must do, should do, could do).

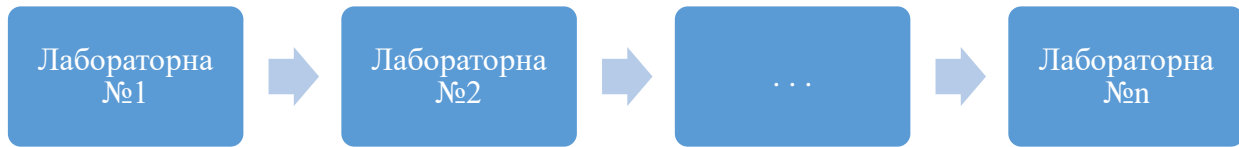


Рис. 1 Каскадна схема

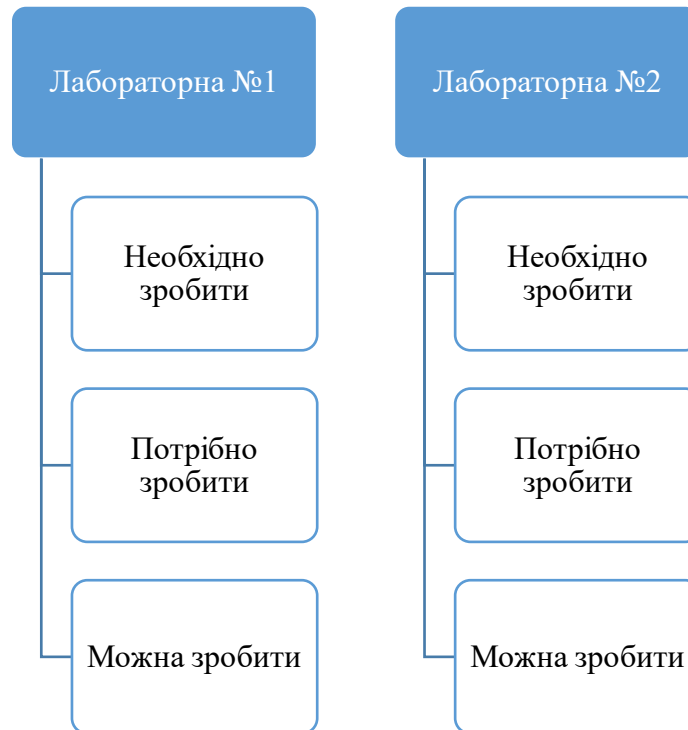


Рис. 2. Розгалужена схема

Відмінністю між цими підходами є відмова від звичайного переходу від попередньої до наступної роботи шляхом ділення її на задачі, що дають можливість студентам виконувати їх чітко усвідомлюючи «ступені заглиблення» у тему. Концепція «необхідно зробити, потрібно зробити, можна зробити» або ж пріоритизація MoSCoW [18] використовується при розробці програмного забезпечення на підприємствах для розбиття функціоналу на різнорівневі складові для вироблення підходів до виділення зіставних частин проекту від «важливого» до «можливого», що на практиці також можна застосувати і до лабораторних робіт студентів. Так, частина «необхідно зробити» є безпосереднім завданням, яке згідно з пройденим за тією чи іншою темою матеріалом, що є наріжним для успішної здачі лабораторної роботи. Частина «потрібно зробити» – ускладнений рівень завдання, що передбачає додаткові умови до функціоналу програми в межах пройдені теми безпосередньої роботи та попередніх робіт, а остання частина «можна зробити» – завдання, що спрямоване на активне залучення відведених на

самостійну роботу годин, це спонукає студента до вивчення можливостей покращення коду програми, його рефакторингу та роботи в напрямку чистоти коду.

Згідно з «Положенням про організацію освітнього процесу» [17] на самостійну роботу припадає від 50 до 67 відсотків часу, відведеного на вивчення дисципліни. Для ефективного використання цього часу з метою якісної підготовки майбутніх фахівців нами було розроблено комплексні завдання, що мають на меті не тільки поглибити знання студентів, а також дати розуміння про самоорганізацію груп під час виконання комплексних робіт. Відповідно до Agile Education, для виконання комплексного завдання формуються групи студентів від 3 до 5 осіб, що розподіляють між собою ролі в групі та будують взаємозв'язки відповідно до методології eduScrum [16]. У цьому контексті викладачу відводиться роль консультанта-координатора, що визначає вимоги до функціоналу комплексної роботи, спрямовує студентські групи в пошуку ними інформації та допомагає розвивати навички взаємодії, взаємодопомоги та взаємонавчання членів групи.

Оскільки дисципліна вивчається на першому курсі ми не розглядаємо четверту частину пріоритизації MoSCoW – цього не буде (won't be), адже вважаємо за можливе на такому ранньому етапі навчання додавати завдання, що потребують розуміння принципів управління ризиками. Після отримання комплексної роботи студенти починають опановувати принципи Agile, створювати план проєктування та розроблення свого програмного продукту (рис. 3). Студенти в межах своїх груп з'ясовують, що необхідно зробити, як це можна зробити та якими засобами можна досягнути поставленої мети, і беруть на себе відповідальність, за вирішення всіх труднощів, які виникають у процесі виконання роботи. Звичайно, викладач також є частиною команди і працює спільно зі студентами, допомагаючи їм на кожному етапі. Водночас викладачу як консультанту-координатору потрібно вибудовувати спільну продуктивну роботу зі студентами кожної групи. Надзвичайно важливими в цьому процесі є координаційні зустрічі, під час яких обговорюється прогрес по кожному зі спринтів та всі проблеми, що виникають під час виконання поставлених задач.

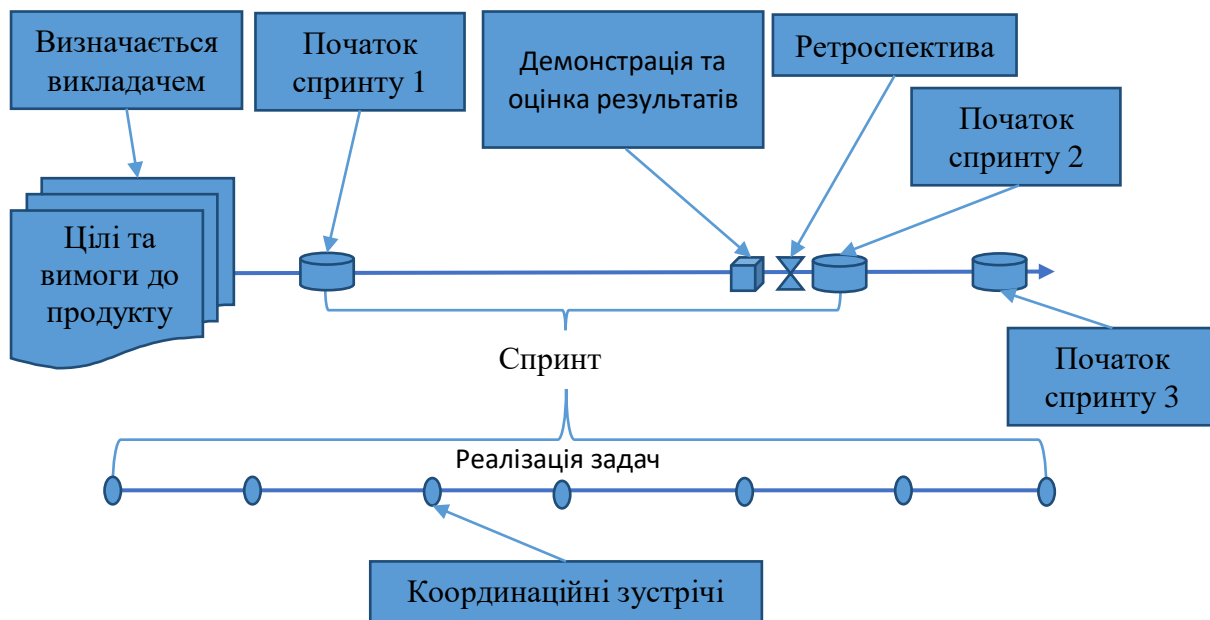


Рис. 3. План проєктування та розроблення програмного продукту комплексної роботи

### 3. МЕТОДИКА ДОСЛІДЖЕННЯ

Згідно з розробленою Agile-методологією кожна лабораторна робота розглядається як груповий проєкт. Відповідно до цих принципів план роботи над комплексним завданням розбивається на етапи (відповідно до eduScrum – спринти).

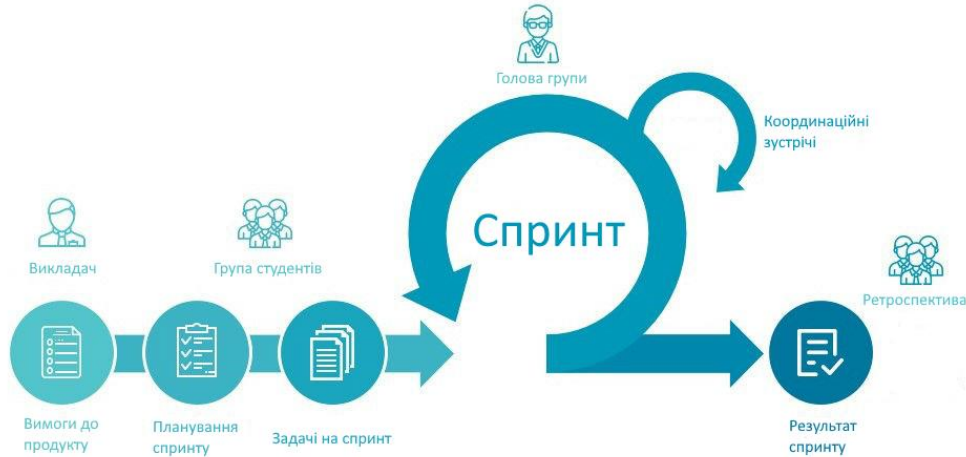


Рис. 4. Класична схема eduScrum

Перед першим етапом студенти отримують завдання та розподіляються на групи, що будуть працювати над своїми комплексними роботами. Відповідно до методології для координації своїх дій у групі передбачені спеціальні зустрічі (рис. 4.), на яких студенти обговорюють свою вже виконану роботу або визначають проблемні етапи.

Після розроблення плану студенти розпочинають роботу над цілями та вимогами до продукту, що полягає в подрібненні комплексної роботи на окремі завдання, які розподіляються між членами групи.

Студенти визначають ступінь «готовності» кожного завдання, тобто який його стан вважати таким, що відповідає цілям групи та може бути визначений як задовільний для інтеграції в продукт, а також визначають пріоритетність завдань за згаданою вище схемою «необхідно зробити, потрібно зробити, можна зробити». Результати такої роботи прийнято демонструвати у вигляді Agile-дошки (рис. 5).

До виконання	В процесі	Виконано
Задача 1	Прочитати параграф 1 методичного комплексу	Опрацювати матеріал лекцій 1 та 2
Задача 2		
Задача 3		
Тестування виконаних задач		

Рис. 5. Фрагмент Agile-дошки спринту дисципліни «Алгоритмізація і основи програмування»

За нашим задумом викладач координує діяльність студентських груп під час ретроспектив, на яких підбивають підсумки роботи групи протягом етапу. На ретроспективі студенти звітують про виконану роботу, закриті задачі, невиконані задачі, переносять на наступний етап задачі з категорії «необхідно зробити», якщо вони не були закінчені протягом поточного етапу, та відповідають на теоретичні питання викладачів.

У такий спосіб подача матеріалу заохочує студентів докладати зусиль до саморозвитку в умовах впливу зовнішніх, суб'єктивних та об'єктивних чинників на кшталт:

- зміни в академічному розкладі;
- зміни в методах та засобах навчання;
- потреба звернутись до іншої технології;
- необхідність перегляду функціоналу продукту;
- непередбачувані зміни в команді розробки.

Вище означені процеси дають нам картину розуміння навчання, що відповідає класичній схемі eduScrum (рис. 4) та розробленій Agile-методології.

Причину трансформації класичного підходу «викладач-практик» до «викладач-координатор» ми обґрунтовуємо наведеними вище підходами до трансформації самих принципів викладання і отримання інформації. Відходячи від каскадної моделі, роль «викладача-координатора» полягатиме не тільки в контролі за виконанням роботи, допомозі в опануванні навчального матеріалу, а й у спонуканні до отримання додаткової інформації, навичок та знань, адже сучасні університети мають на меті не тільки класичну схему надання освітніх послуг, а й випуск на ринок праці фахового спеціаліста з компетентностями до швидкої адаптації. Тому, коли ми говоримо про необхідність модернізації та інформатизації навчання, вважаємо за необхідне говорити не тільки про технічне забезпечення навчальних установ, а й про трансформацію методологій, що сприятимуть підготовці спеціалістів у галузі інформаційних технологій.

#### 4. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Такий підхід до комплексних завдань було апробовано авторами у 2017/2018 та 2018/2019 навчальних роках. Даний підхід сприяє інтерпретації звичного навчального процесу як довготривалого проєкту з використанням Agile-методології [1]. Однак за два роки апробації нової методології було прийнято рішення адаптувати даний підхід до виконання лабораторних робіт.

На початку статті нами було окреслено певну адаптацію комплексу лабораторних робіт до Agile-методології. Тому у 2019/2020 навчальному році було вирішено адаптувати описану методологію групової роботи і для лабораторних завдань. Задля оцінки ефективності такого підходу в повному циклі вивчення дисципліни «Алгоритмізація і основи програмування» за наявності двох груп (44 студенти), що ділились на чотири підгрупи (по 11 студентів), у поточному навчальному році було прийнято рішення дві підгрупи навчати за Agile-методологією (експериментальні групи: ЕГ1, ЕГ2), а дві – за усталеними методологіями (контрольні групи: КГ1, КГ2). (табл. 1)

Дисципліна «Алгоритмізація і основи програмування» вивчається на першому курсі протягом двох навчальних семестрів. Експеримент проводився в першому семестрі 2019/2020 навчального року, у якому вивчаються основи програмування мовою C++. Зважаючи на те, що мова йде про студентів-першокурсників, для забезпечення

нормальності розподілу студентів на групи за рівнем початкової підготовки за дані вважалися бали, набрані ними при складанні ЗНО з математики, отримані з інформаційної бази vstup.info.

Таблиця 1

#### Розподіл студентів за групами

	Контрольна група (КГ)	Експериментальна група (ЕГ)	Всього
Перша група	11	11	22
Друга група	11	11	22
Всього	22	22	44

В Agile-підгрупах (ЕГ1 та ЕГ2) студенти були розділені на групи по 3-4 особи, що працювали над завданнями разом. Ці студенти отримували комплексні завдання, які містили пакет теоретичних питань, та, зважаючи на колективність роботи, захищали їх разом. Водночас варто зазначити, що втручання в процес роботи груп з боку викладача поступово мінімізувалося. Студенти, отримавши завдання, переважно самі регулювали взаємодії між собою на всіх етапах виконання поставленої задачі, враховуючи розподіл ролей кожного з них в цьому процесі. У підгрупах КГ1 та КГ2 лабораторні роботи виконувались за класичною «каскадною» схемою, коли кожен лабораторну роботу студенти здавали в індивідуальному порядку. За результатами іспиту було обчислено середній бал студентів за всіма підгрупами (табл. 2).

Таблиця 2

#### Середній бал студентів за групами

	Контрольна група (КГ)	Експериментальна група (ЕГ)	Всього
Перша група	71.82	80.27	76.05
Друга група	73.00	84.73	78.86
Всього	72.41	82.50	77.45

Як бачимо, студенти контрольних груп, які навчались за традиційним форматом, у середньому отримали на 10.09 балів менше в порівнянні зі студентами експериментальних підгруп. Варто також зазначити, що середній бал студентів другої групи на 2.81 більший, ніж у студентів першої групи. Подібна динаміка спостерігається і у відповідних контрольних та експериментальних підгрупах: середній бал в експериментальній підгрупі першої групи (ЕГ1) перевищує середній бал в контрольній підгрупі цієї ж групи (КГ1) на 8.45 балів, водночас для підгруп ЕГ2 та КГ2 ця різниця складає 11.73 балів. Також варто зазначити, що студенти другої групи в цілому навчаються краще та мають вищі показники.

На рисунках 6,7 представлено розподіл підсумкових балів з дисципліни для студентів усіх підгруп.



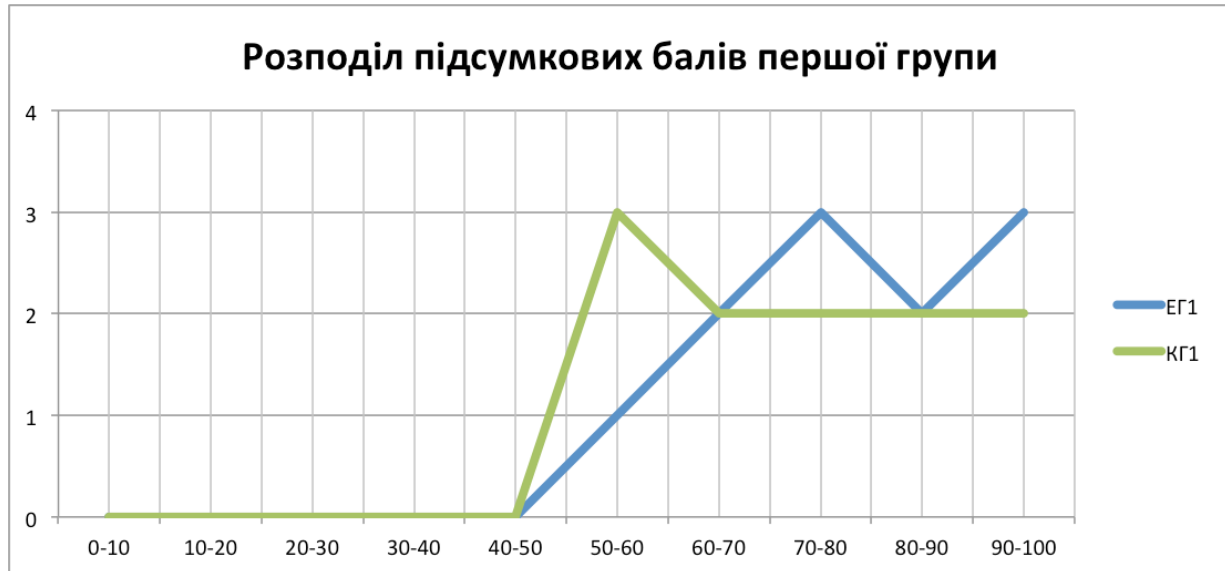


Рис. 6. Розподіл підсумкових балів першої групи за підгрупами EG1 та KG1

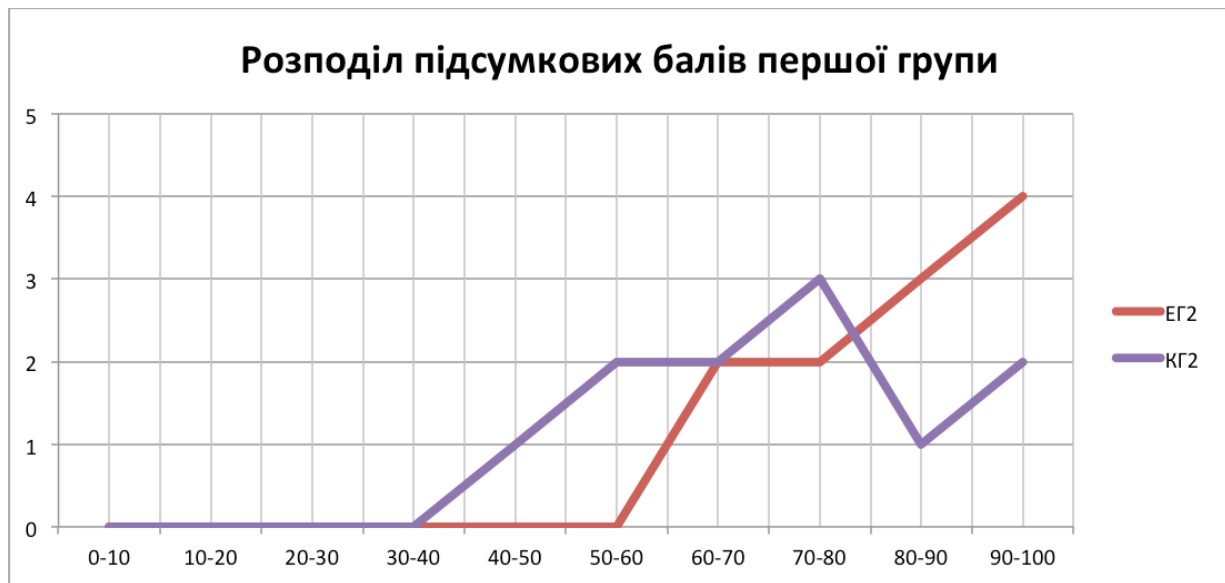


Рис. 7. Розподіл підсумкових балів другої групи за підгрупами EG2 та KG2

Вивчаючи графіки, представлені на рисунках 6 та 7, можемо помітити, що для експериментальних груп розподіл підсумкових балів з дисципліни суттєво зміщено вправо, у підгрупі EG2 максимум виникає в області 90-100 балів. Водночас у контрольних підгрупах KG1 та KG2 спостерігається однаковий відсоток студентів, що отримали підсумкову оцінку «відмінно» (90 балів і вище) – 18.18%, в експериментальних підгрупах EG1 та EG2 цей показник вищий – 27.27% і 45.45% відповідно. Разом з тим, в експериментальних підгрупах немає жодного студента з балом нижче за 60, а в контрольних підгрупах відсоток таких студентів складає: KG1 – 18.18% та KG2 – 9.09%.

Для детальнішого аналізу ефективності впровадження Agile-методології та виявлення особливостей навчання студентів за різними технологіями навчання було

розглянуто не менш важливий показник – своєчасність виконання студентами лабораторних робіт. Позитивною особливістю використання Agile-методології в експериментальних підгрупах є те, що студенти в межах мікрогруп (3-4 особи) здійснювали мотиваційний вплив один на одного, що спонукало їх виконувати всі поставлені завдання з дотриманням дедлайнів, що й відобразилось на динаміці вчасності виконання ними лабораторних робіт (рис. 8-11).

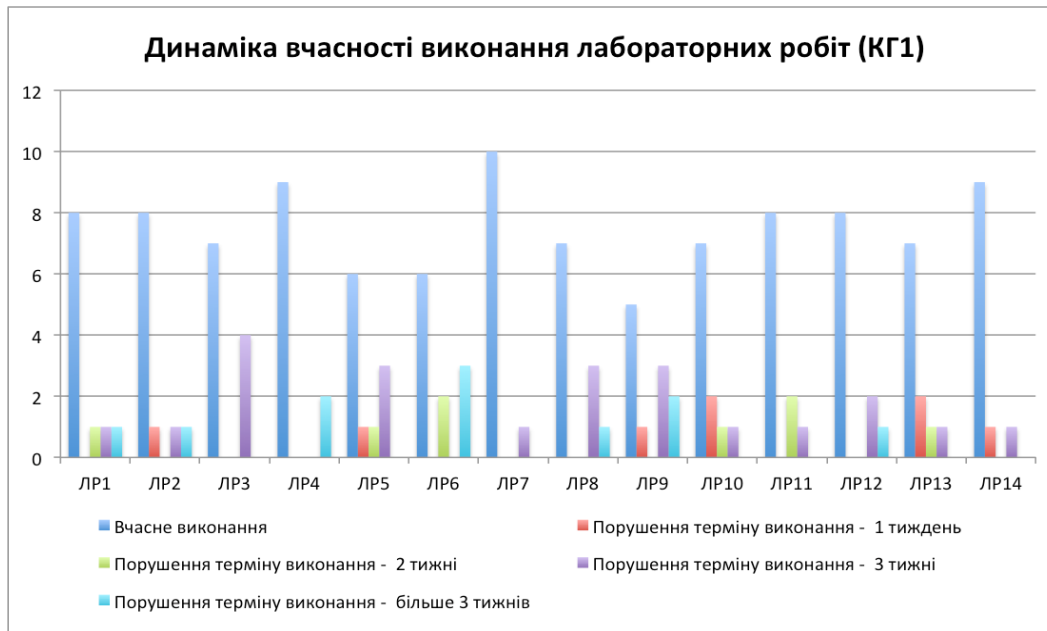


Рис. 8. Динаміка вчасності виконання лабораторних робіт студентами КГ1



Рис. 9. Динаміка вчасності виконання лабораторних робіт студентами ЕГ1

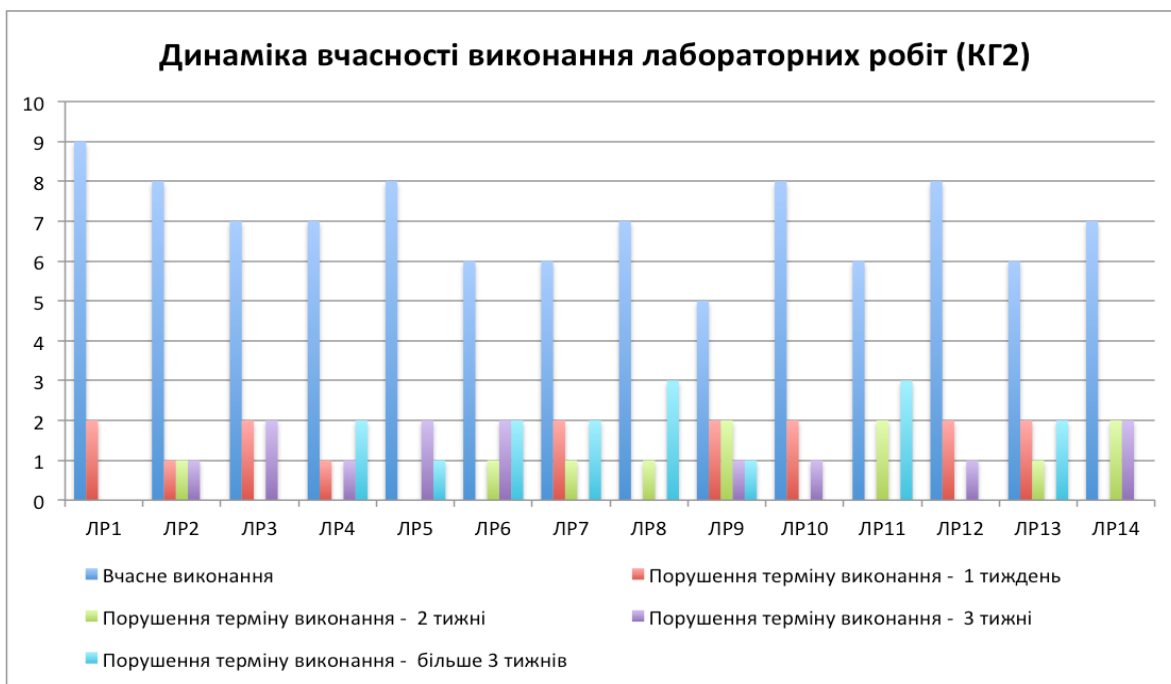


Рис. 10. Динаміка вчасності виконання лабораторних робіт студентами КГ2

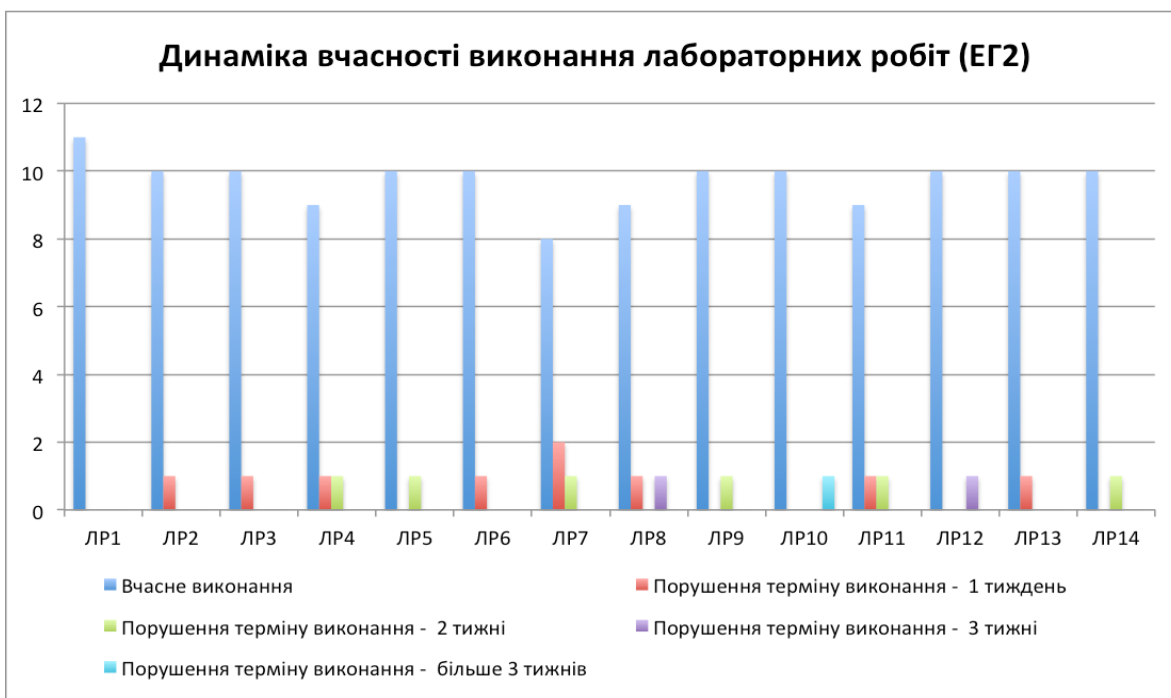


Рис. 11. Динаміка вчасності виконання лабораторних робіт студентами ЕГ2

Отже, бачимо, що відсоток студентів, які вчасно виконували лабораторні роботи в експериментальних підгрупах, коливається в межах 86-88%, тоді як цей показник для контрольних підгруп тримався на рівні 64-68%. Крім того, у контрольних підгрупах спостерігалось порушення встановлених термінів виконання лабораторних робіт на 3 і

більше тижнів у 18-21% студентів на відміну від 2-4% в експериментальних підгрупах (табл. 3).

Таблиця 3

#### Динаміка вчасності виконання лабораторних робіт за групами

	КГ1	ЕГ1	КГ2	ЕГ2
Вчасне виконання	68%	86%	64%	88%
Порушення терміну виконання – 1 тиждень	5%	8%	10%	6%
Порушення терміну виконання – 2 тижні	5%	2%	7%	4%
Порушення терміну виконання – 3 тижні	14%	3%	8%	1%
Порушення терміну виконання – більше 3 тижнів	7%	1%	10%	1%

За результатами проведеного дослідження можемо констатувати, що Agile-групи здавали матеріали з кращим дотриманням встановлених термінів виконання, а робота в групах спонукала студентів допомагати одне одному у вивченні теорії, що було умовою успішної здачі лабораторних робіт.

## 5. ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

Впровадження гнучких методологій у навчальний процес оптимізує підходи в навчанні майбутніх фахівців сфери ІТ. Запроваджені імплементації гнучкої методології розробки програмного забезпечення – Agile – у процес розробки навчально-методичного забезпечення та запропоновані підходи у викладанні у вищій школі сприяють не тільки кращому розумінню та засвоєнню студентами теоретичного та практичного навчального матеріалу. Цей підхід також допомагає студентам зрозуміти виробничі процеси в ІТ-проектах, які організовані згідно з методологією Agile, налаштувати себе працювати в рамках Scrum, дає початкове уявлення про поняття складності задачі, обов'язкових та додаткових вимог до продукту – їх знань у рамках дисципліни.

Крім того, практика організації групової навчальної роботи студентів показує ряд інших не менш важливих переваг:

- на формування необхідних компетенцій витрачається менше часу, ніж при використанні класичних традиційних методів навчання;
- спостерігається зростання обсягу та глибини розуміння навчального матеріалу;
- відбувається зростання пізнавальної активності та підвищення творчої самостійності студентів;
- прослідковується зміна характеру взаємостосунків між студентами: вони починають краще розуміти один одного і самих себе та розвивають у собі рівень самокритичності;
- допомагаючи своїм партнерам по групі, студенти вчаться будувати свою поведінку з урахуванням позиції інших людей та стають більш відповідальними.

Методології Agile і Scrum дозволяють розробити та імплементувати сучасну, адаптовану до вимог ринку праці програму навчання ІТ-спеціалістів та допомагають розвинути навички пошуку необхідної інформації та взаємодії з одногрупниками, що також сприяє подальшому навчанню та покращенню своїх «soft skills».

Впровадження Agile-методології в навчальний процес в принципі змінює ставлення студентів до навчання та стимулює їх систематичну навчальну роботу, що своєю чергою

впливає і на якість виконання роботи, і на дотримання термінів її виконання. Це дозволяє більш об'єктивно та диференційовано оцінювати результати роботи студентів та розвиває можливості особистісно-діяльнісного підходу до навчання.

Ми переконані, що впровадження гнучких методологій у навчальний процес підготовки студентів технічних спеціальностей має стати невід'ємним елементом трансформації галузі вищої освіти.

Крім вище зазначеного, запропонований підхід більш вдало поєднується з дистанційним навчанням, яке диктується нам реаліями сьогодення. З початком карантинного періоду COVID-19 у закладах вищої освіти України авторами було проведено адаптацію розробленого підходу до онлайн режиму проведення лекційних та практичних (або лабораторних) занять. Розроблені карти вивчення всієї дисципліни та зв'язків кожного з модулів надають студентам можливість під час ізоляції навіть без постійного доступу до інтернету йти самостійно по карті модуля і зрідка корегувати свою роботу згідно з вимогами команди та викладача. Agile-методологія вивчення дисциплін у дистанційному форматі буде розроблена в подальших дослідженнях і представлена разом з результатами запропонованої адаптації до дистанційного навчання в наступних статтях.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ:

- [1] В. Міронова, та М. Пирог, “AGILE методологія в освіті як основний інструмент побудови сучасних освітніх дисциплін вищої школи”, *Економіка і управління*, №4 (76), с. 124–130, 2017.
- [2] В. Міронова, М. Пирог, та І. Гарко, “Agile-методологія та інформатизація навчального процесу на прикладі проектної роботи з дисципліни «Алгоритмізація і основи програмування»”, на *VII Міжнар. наук.-техн. конф. Проблеми інформатизації*, Черкаси, 2019, с. 30.
- [3] Agile-маніфест розробки програмного забезпечення, 2001 [Електронний ресурс]. Доступно: <https://cutt.ly/SyEftwh>.
- [4] В. Семиноженко, “Сучасна освітня політика спричинить брак ІТ-спеціалістів”, *Євро Освіта*, 2014 [Електронний ресурс]. Доступно: <https://cutt.ly/Jumzx4M>.
- [5] Д. Семьонов, “Українська вища освіта — мертва. Тримайтесь від неї подалі”, *Medium*, 2016 [Електронний ресурс]. Доступно: <https://cutt.ly/Mumj2R5>.
- [6] І. Бардус, “Аналіз професійної діяльності фахівців у галузі інформаційних технологій”, *Проблеми інженерно-педагогічної освіти*, №48-49, с. 71–79, 2015.
- [7] О. Дудка, та Я. Никорак, “Проблема реформування ІТ-освіти в Україні”, на *Міжнар. наук.-практ. конф. Інформаційні технології та комп'ютерна інженерія*, Івано-Франківськ, 2016 [Електронний ресурс]. Доступно: <https://cutt.ly/uyEdx5O>.
- [8] П. Малежик, “Визначення структури готовності майбутніх ІТ-фахівців до професійної діяльності як педагогічна проблема”, *Наукові записки. Серія: педагогічні науки*, №173, с. 130–135, 2018.
- [9] EPAM+вузы: как мы сотрудничаем с университетами Украины, *habr. Блог компании EPAM*, 2018 [Електронний ресурс]. Доступно: [https://habr.com/ru/company/epam\\_systems/blog/422455/](https://habr.com/ru/company/epam_systems/blog/422455/).
- [10] Вища освіта в Україні: студентські історії, *Інформаційний портал "Перша електронна газета"*, 2017 [Електронний ресурс]. Доступно: <https://cutt.ly/Eumkdpe>.
- [11] М. Тульчинська, “Вища освіта - найдурніша інвестиція в дитину?”, *"Корреспондент"*, 2018 [Електронний ресурс]. Доступно: <https://cutt.ly/cymkUMO>.
- [12] J. Stewart, C. DeCusatis, K. Kidder, J. Massi, and K. Anne, “Evaluating Agile Principles in Active and Cooperative Learning”, in *Student-Faculty Research Day, CSIS, Pace University*, P. B3, 2009. Доступно: <https://cutt.ly/piuBBUI>.
- [13] M. Alfonso, and A. Botia, “An Iterative and Agile Process Model for Teaching Software Engineering”, in *IEEE International Conference on Software Engineering Education and Training (CSEE&T)*, 2005. Доступно: <https://cutt.ly/2iu0G2R>.
- [14] P. Salza, P. Musmarra, and F. Ferrucci, “Agile Methodologies in Education: A Review. In Agile and Lean Concepts for Teaching and Learning”, *Springer*, pp. 25–45, 2019. Доступно: <https://cutt.ly/Niu2nzk>.

- [15] S. Duvall, D. Hutchings, and M. Kleckner, “Changing Perceptions of Discrete Mathematics Through Scrum-Based Course Management Practices”, *Journal of Computing Sciences in Colleges*, no. 33(2), pp. 182–189, 2017. <https://cutt.ly/Tiu9e1Y>
- [16] Agile в школі, *Agile in Education* [Електронний ресурс]. Доступно: <https://cutt.ly/lyEfxEN>.
- [17] Положення про організацію освітнього процесу у Київському національному університеті імені Тараса Шевченка, *Київський національний університет імені Тараса Шевченка*, 2018 [Електронний ресурс]. Доступно: <https://cutt.ly/5yEhQf6>.
- [18] MoSCoW Prioritisation, *Coley Consulting* [Електронний ресурс]. Доступно: <https://cutt.ly/ryB5n4S>.

Матеріал надійшов до редакції 25.06.2020 р.

## ПРИМЕНЕНИЕ AGILE-МЕТОДОЛОГИИ В ОБУЧЕНИИ АЛГОРИТМИЗАЦИИ И ОСНОВ ПРОГРАММИРОВАНИЯ СТУДЕНТОВ СПЕЦИАЛЬНОСТИ КОМПЬЮТЕРНЫЕ НАУКИ

### Гарко Ирина Игоревна

кандидат физико-математических наук,  
доцент кафедры прикладных информационных систем факультета информационных технологий  
Киевского национального университета имени Тараса Шевченко, г. Киев, Украина  
ORCID ID 0000-0003-0671-6336  
[garko.iryana@gmail.com](mailto:garko.iryana@gmail.com)

### Пирог Николай Владимирович

ассистент кафедры прикладных информационных систем факультета информационных технологий  
Киевский национальный университет имени Тараса Шевченко, г. Киев, Украина  
ORCID ID 0000-0003-2588-6066  
[mykola.pyroh@yahoo.com](mailto:mykola.pyroh@yahoo.com)

### Миронова Виктория Леонидовна

кандидат технических наук, доцент,  
доцент кафедры прикладных информационных систем факультета информационных технологий  
Киевский национальный университет имени Тараса Шевченко, г. Киев, Украина  
ORCID ID 0000-0002-0878-0967  
[vicky.mironova@gmail.com](mailto:vicky.mironova@gmail.com)

**Аннотация.** В представленной статье рассматривается результат имплементации гибкой методологии разработки программного обеспечения – Agile – в процесс обучения студентов высших учебных заведений. Нами представлены подходы к модернизации учебного процесса, разработка учебно-методического обеспечения и подходов к преподаванию в высшей школе на примере учебной дисциплины «Алгоритмизация и основы программирования» для студентов специальности 122 «Компьютерные науки» образовательной программы «Прикладное программирование». Проанализировано широкий спектр публикаций, которые определяют проблематику разработки новых методологий преподавания, в том числе проблем подготовки специалистов технических специальностей в реалиях трансформации информационного общества и информатизации учебного процесса. Проведен анализ рынка информационных технологий и запросы ведущих компаний на рынке Украины, а также исследованы рефлексии выпускников по эффективности обучения в современных высших учебных заведениях. В статье описаны результаты многолетней постепенной адаптации технологии Agile в учебный процесс, ее введение в учебный процесс и результаты апробации разработанной методологии. Рассмотрены теоретические основы и преимущества Agile-методологии перед классической схемой обучения, определенной нами как «прямая» или «каскадная». С целью эффективного исследования влияния гибких методологий на образовательный процесс представлено подходы к формированию экспериментальных и контрольных академических групп студентов. Приведена динамика своевременности выполнения лабораторных работ и сравнения итоговых

результатов обучения академических групп. Рассмотрено влияние на успешность академических групп Agile-методологии и проведено статистическое сравнение с классическим подходом к преподаванию. Определены направления дальнейших исследований темы имплементации Agile-методологии в образовательный процесс с учетом актуальности вопроса трансформации образования в Украине и профессиональной подготовки специалистов информационно-технической области.

**Ключевые слова:** Agile-методология; Scrum; образование; информационные технологии; подготовка специалистов; профессиональная подготовка.

## APPLICATION OF AGILE-METHODOLOGY IN TEACHING ALGORITHMIZATION AND PROGRAMMING BASICS FOR COMPUTER SCIENCES SPECIALITY STUDENTS

### **Iryna I. Harko**

PhD of Physical and Mathematical Sciences,  
Associate Professor at the Applied Information Systems Department Faculty of Information Technology  
Taras Shevchenko National University of Kyiv, Kyiv, Ukraine  
ORCID ID 0000-0003-0671-6336  
*garko.iryana@gmail.com*

### **Mykola V. Pyroh**

Assistant Professor at the Applied Information Systems Department Faculty of Information Technology  
Taras Shevchenko National University of Kyiv, Kyiv, Ukraine  
ORCID ID 0000-0003-2588-6066  
*mykola.pyroh@yahoo.com*

### **Victoriia L. Mironova**

PhD of Technical Sciences, Associate Professor,  
Associate Professor at the Applied Information Systems Department Faculty of Information Technology  
Taras Shevchenko National University of Kyiv, Kyiv, Ukraine  
ORCID ID 0000-0002-0878-0967  
*vicky.mironova@gmail.com*

**Abstract.** The presented article considers the result of the flexible software development methodology - Agile - implementation into the process of higher education students teaching. We present approaches to the educational process modernization, educational and methodological support development and approaches to teaching in high school on the example of the discipline "Algorithmization and basics of programming" for students in specialty 122 Computer Science in educational program "Applied Programming". Authors analyzed a wide range of publications outlining the issues of developing new teaching methodologies, including the problems of training future IT-specialists in the information society transformation realities and the educational process informatization. The analysis of the information technologies market and inquiries of the leading companies in the market of Ukraine is carried out, besides that the reflections of graduates concerning efficiency of training in modern higher education establishments are investigated. The article describes the results of several years gradual adaptation of Agile technology to the learning process, its introduction into the learning process and the results of the developed methodology approbation. The theoretical bases and advantages of Agile-methodology over the classical scheme of training, which we define as "direct" or "cascade", are considered. In order to effectively study the impact of flexible methodologies on the educational process, approaches to the formation of experimental and control academic students' groups are presented. The dynamics of laboratory work timeliness and comparison of the academic groups' final results are given. The influence of Agile-methodology on the academic groups success is considered and a statistical comparison with the classical approach to teaching is made. Also, we determined the directions of further research on the topic of Agile-methodology implementation in the educational process taking into account the urgency of the education

transformation issue in Ukraine and professional training of specialists in the field of information technology.

**Keywords:** Agile-methodology; Scrum; education; Information Technology; training of specialists; professional training.

## REFERENCES (TRANSLATED AND TRANSLITERATED)

- [1] V. Mironova, and M. Pyroh, "AGILE methodology in education as the main tool for building modern educational disciplines of higher education", *Economics and Management*, no. 4 (76), pp. 124–130, 2017. (in Ukrainian)
- [2] V. Mironova, M. Pyroh, and I. Harko, "Agile-methodology and informatization of the educational process on the example of project work in the discipline "Algorithmization and basics of programming"", in *VII International. scientific and technical conf. Problems of informatization*, Cherkasy, 2019, P. 30. (in Ukrainian)
- [3] Agile-software development manifesto, 2001 [Online]. Available: <https://cutt.ly/SyEftwh>. (in Ukrainian)
- [4] V. Seminozhenko, "Modern educational policy will cause a lack of IT specialists", *Euro Education*, 2014 [Online]. Available: <https://cutt.ly/Jymzx4M>. (in Ukrainian)
- [5] D. Semyonov, "Ukrainian higher education is dead. Stay away from it", *Medium*, 2016 [Online]. Available: <https://cutt.ly/Mymj2R5>. (in Ukrainian)
- [6] Bardus, "Analysis of professional activities of specialists in the field of information technology", *Problems of engineering and pedagogical education*, no.48-49, pp. 71–79, 2015. (in Ukrainian)
- [7] O. Dudka, and J. Nikorak, "The problem of reforming IT education in Ukraine", in *International. scientific-practical conf. Information Technologies and Computer Engineering*, Ivano-Frankivsk, 2016. [Online]. Available: <https://cutt.ly/uyEdx5O>. (in Ukrainian)
- [8] P. Malezhyk, "Determining the structure of readiness of future IT professionals for professional activities as a pedagogical problem", *Scientific notes. Series: pedagogical sciences*, no.173, pp. 130–135, 2018. (in Ukrainian)
- [9] EPAM + universities: how we cooperate with Ukrainian universities, *habr. EPAM Blog*, 2018 [Online]. Available: [https://habr.com/ru/company/epam\\_systems/blog/422455/](https://habr.com/ru/company/epam_systems/blog/422455/). (in Russian)
- [10] Higher education in Ukraine: student stories, *Information portal "First electronic newspaper"*, 2017. [Online]. Available: <https://cutt.ly/Eymkdpe>. (in Ukrainian)
- [11] M. Tulchynska, "Higher education - the stupidest investment in a child?", *"Correspondent"*, 2018 [Online]. Available: <https://cutt.ly/cymkUMO>. (in Ukrainian)
- [12] J. Stewart, C. DeCusatis, K. Kidder, J. Massi, and K. Anne, "Evaluating Agile Principles in Active and Cooperative Learning", in *Student-Faculty Research Day, CSIS, Pace University*, P. B3, 2009. Available: <https://cutt.ly/piuBBUI>. (in English)
- [13] M. Alfonso, and A. Botia, "An Iterative and Agile Process Model for Teaching Software Engineering", in *IEEE International Conference on Software Engineering Education and Training (CSEE&T)*, 2005. Available: <https://cutt.ly/2iu0G2R>. (in English)
- [14] P. Salza, P. Musmarra, and F. Ferrucci, "Agile Methodologies in Education: A Review. In Agile and Lean Concepts for Teaching and Learning", *Springer*, pp. 25–45, 2019. Available: <https://cutt.ly/Niu2nzk>. (in English)
- [15] S. Duvall, D. Hutchings, and M. Kleckner, "Changing Perceptions of Discrete Mathematics Through Scrum-Based Course Management Practices", *Journal of Computing Sciences in Colleges*, no. 33(2), pp. 182–189, 2017. Available: <https://cutt.ly/Tiu9e1Y>. (in English)
- [16] Agile in school, *Agile in Education* [Online]. Available: <https://cutt.ly/lyEfxEN>. (in Russian)
- [17] Regulations on the organization of the educational process at the Taras Shevchenko National University of Kyiv, *Taras Shevchenko National University of Kyiv*, 2018 [Online]. Available: <https://cutt.ly/5yEhQf6>. (in Ukrainian)
- [18] MoSCoW Prioritisation, *Coley Consulting* [Online]. Available: <https://cutt.ly/ryB5n4S>. (in English)

