

КЛАСИФІКАЦІЯ НЕТЕХНІЧНИХ КОМПЕТЕНТНОСТЕЙ МАЙБУТНІХ ІНЖЕНЕРІВ-ПРОГРАМІСТІВ

Валерій Концедайло

Житомирський державний університет імені Івана Франка

Анотація:

У статті розглянуто сутність поняття «нетехнічні компетентності майбутніх інженерів-програмістів», а також наведено класифікації нетехнічних компетентностей з виділенням їх ключових характеристик. Доведено, що типовий освіті інженерів-програмістів бракує практичного опанування процесів розробки програмного забезпечення. Це здебільшого пов'язано з тим, що студенти під час лекцій лише ознайомлюються з відповідними теоріями щодо процесів розробки програмного забезпечення, але мають доволі обмежені можливості практичного втілення цих теорій у життя у відповідних дисциплінах. Тому фахівцям доводиться набувати навичок і досвіду, працюючи в реальних проектах, де наслідки неадекватного планування або неправильні рішення можуть призвести до провалу всього проекту або втрати значного прибутку. З'ясовано, що когнітивні й міжособистісні компетентності, на яких ґрунтуються досвідченість і професіоналізм розробників програмного забезпечення, слугують вимогами, що висуваються до компетентного інженера-програміста.

Ключові слова:

компетентність; нетехнічні компетентності; інформаційно-комунікаційні технології; інженер-програміст.

Анотация:

Концедайло Валерий. Классификация нетехнических компетентностей будущих инженеров-программистов. В статье рассмотрена сущность понятия «нетехнические компетентности будущих инженеров-программистов», а также приведена классификация нетехнических компетентностей с выделением их ключевых характеристик. Доказано, что типичному образованию инженеров-программистов не хватает практического освоения процессов разработки программного обеспечения. Это в большинстве случаев связано с тем, что студенты на лекциях только знакомятся с соответствующими теориями относительно процессов разработки программного обеспечения, однако имеют довольно ограниченные возможности практического воплощения этих теорий в жизнь в соответствующих дисциплинах. Поэтому специалистам приходится развивать свои навыки и набираться опыта, работая в реальных проектах, где последствия неадекватного планирования или неправильные решения могут привести к провалу всего проекта или к потере значительной прибыли. Установлено, что когнитивные и межличностные компетентности, на которых основываются опыт и профессионализм разработчиков программного обеспечения, служат требованиями, которые ставятся к компетентному инженеру-программисту.

Ключевые слова:

компетентность; нетехнические компетентности; информационно-коммуникационные технологии; инженер-программист.

Resume:

Kontsedailo Valerii. Classification of non-technical competencies of future software engineers. The article describes the essence of the concept of "non-technical competencies of software engineers", presents the classification of non-technical competencies and highlights their key characteristics. It was found that the typical software engineering education lacks a practical treatment of the processes of software engineering – students are presented with relevant process theory in lectures, but have only limited opportunity to put these concepts into practice in an associated class project. As a result, new professionals need to develop their experience by working in real projects, where the effects of an inadequate plan or a bad decision can lead to project failure or the loss of significant profit. It was found that the cognitive and interpersonal competencies, which underpin software professionals and technical proficiency, are being recognized as requirements for a competent software developer.

Key words:

competence; non-technical competencies; information and communications technologies; software engineer.

Постановка проблеми. В умовах розбудови системи педагогічної освіти та впровадження інноваційних технологій у процес навчання особливого значення набуває проблема якості та ефективності освітнього процесу. За останні декілька років спостерігається зростання попиту на ринку праці на високомотивованих і висококваліфікованих інженерів-програмістів. Це пов'язано з неабияким поширенням новітніх технологій, зокрема Інтернету, мобільних і хмарних, у всьому світі. На жаль, цей попит незадоволений, що призводить до різноманітних проблем під час працевлаштування студентів і випускників цієї спеціальності.

За даними звітного видання CHAOS, у 2015 р. лише 29% усіх проектів у сфері інформаційних технологій (ІТ) були успішно завершені, тобто були закінчені вчасно, у рамках бюджету й з усіма необхідними можливостями та функціями [1]. Водночас ІВМ у своєму дослідженні про причини провалу ІТ-проектів

дійшли висновку, що 54% ІТ-проектів зазнали невдачі через невдале управління проектами [2].

Успішний інженер-програміст повинен володіти широким спектром навичок і талантів. Керівники проектів знають, як важко знайти, мотивувати й утримати таких фахівців. Зі свого боку, педагоги стикаються з подібними, а можливо, й більш складними проблемами: як підготувати висококваліфікованих інженерів-програмістів? чого саме навчати інженерів-програмістів? Адже технології навчання, освітні програми й вимоги до навчального процесу змінюються. Оскільки різноманітні інформаційні технології швидко поширюються на всі аспекти сучасного суспільства, то підготовка кваліфікованих інженерів-програмістів перетворюється на більш складну проблему, що потребує нагального розв'язання [3].

Аналіз останніх досліджень і публікацій. Питання формування нетехнічних компетентностей у майбутніх інженерів-

програмістів тією чи іншою мірою розглядають у своїх роботах Р. Атал (R. Atal) [8], К. Бедсе (K. Bedse) [4], Л. Бендер (L. Bender) [6], Г. Валія (G. Walia) [6], Д. Віл (D. Veal) [9], М. Джазаєрі (M. Jazayeri) [3], А. Кальдерон (A. Calderón) [2], С. Карунасекера (S. Karunasekera) [4], С. Колфілд (C. Caulfield) [9], С. Мей (S. Maj) [9], Дж. Мюнх (J. Münch) [6], Е. Наварро (E. Navarro) [7], К. Нігард (K. Nygard) [6], М. Пагельс (M. Pagels) [6], М. Руїс (M. Ruiz) [2], А. Сурека (A. Sureka) [8], Дж. Ся (J. Xia) [9], Ф. Фагерхольм (F. Fagerholm) [6], А. Фахім (A. Faheem) [5] та ін.

Формулювання цілей статті. Метою статті є аналіз сутності категорії «нетехнічні компетентності майбутніх інженерів-програмістів», а також класифікації нетехнічних компетентностей з виділенням ключових характеристик кожної.

Виклад основного матеріалу дослідження. Відомо, що до професії інженерів-програмістів висуваються певні посадові вимоги. Так, в оприлюдненій версії документа «Guide to the Software Engineering Body of Knowledge, Version 3.0» [10], підготовленого міжнародним комітетом Software Engineering Coordinating Committee разом з міжнародною спільнотою IEEE Computer Society, для об'єднання знань з інженерії програмного забезпечення зазначено 15 галузей знань, якими повинні володіти інженери-програмісти-практики та за якими повинна проводитись професійна підготовка майбутніх інженерів-програмістів:

- вимоги до програмного забезпечення;
- проектування програмного забезпечення;
- конструювання програмного забезпечення;
- тестування програмного забезпечення;
- супровід програмного забезпечення;
- управління конфігурацією програмного забезпечення;
- управління розробкою програмного забезпечення;
- методології розробки програмного забезпечення;
- моделі та методи розробки програмного забезпечення;
- якість програмного забезпечення;
- професійна практика розробки програмного забезпечення;
- економіка програмної інженерії;
- основи обчислювання;
- основи математики;
- основи інженерії.

З наведеного вище переліку можна виокремити такі «нетехнічні» галузі знань:

- вимоги до програмного забезпечення;
- управління розробкою програмного забезпечення;

– методології розробки програмного забезпечення;

– моделі та методи розробки програмного забезпечення.

Відповідно до них професійні компетентності інженерів-програмістів поділяються на дві широкі категорії – «технічні компетентності» й «нетехнічні компетентності». До технічних компетентностей належать технічні знання, здібності, навички та досвід, які працівник повинен мати, щоб виконувати завдання, пов'язані з безпосередньою роботою. Отже, від кандидата можуть вимагати знання про парадигму об'єктно-орієнтованого програмування й навички роботи з базами даних тощо. Нетехнічні ж компетентності сягають своїм корінням у психологію, людську поведінку, соціологію й належать до більш широкого діапазону характеристик, пов'язаних з типами особистості, зі здатністю до соціальної взаємодії, спілкування, лідерства й з особистими звичками. Ці компетентності є суттєвими, бо відповідають за те, як спеціалісти думають, сприймають і реагують у робочому середовищі відповідно до обставин, обмежень і вимог. У середовищі командної розробки програмного забезпечення володіння нетехнічними компетентностями й використання їх також підвищує шанси людини на успіх, кар'єрне зростання, а також позитивно впливають на досягнення спільної мети проекту [5].

Підкреслимо, що на важливості нетехнічних компетентностей у галузі програмної інженерії, зокрема на набутті їх у процесі підготовки майбутніх інженерів-програмістів, наголошують як представники індустрії розробки програмного забезпечення, так і викладачі та науковці. Дж. Беклунд (G. Backlund) і Дж. Сьоннессон (J. Sjunnesson) твердять, що технічні компетентності інженерів-програмістів являють собою лише малу частину компетентностей, необхідних для того, щоб бути успішним на практиці [11]. Проте виявляється, що майбутні інженери-програмісти не володіють такими компетентностями, як, наприклад, взаєморозуміння, співчуття й самокритика, що є важливими для професійного співробітництва [11]. І. Тонг (I. F. Tong) зауважує, що роботодавці воліють наймати випускників університетів відповідних спеціальностей, які володіють як технічними, так і нетехнічними компетентностями, зокрема такими, як міжособистісне спілкування, планування, управління людьми та командами [12]. Ч. Ворачек (C. R. Woratschek) і Т. Ленокс (T. L. Lenox) виявили, що іноді нетехнічні компетентності, такі, як комунікація, мотивація, розв'язання проблем, управління часом, професійна етика, а також схильність

до навчання, є більш важливими, ніж технічні навички [13]. П. Груба (P. Gruba) й Р. Аль-Махмуд (R. Al-Mahmood) з'ясували, що студенти інженерних спеціальностей, професорсько-викладацький склад, а також представник промисловості цілком погоджуються, що розвиток комунікативних навичок і компетентностей має важливе значення для професійного успіху [14].

На сьогодні багато університетів працюють над упровадженням нетехнічних компетентностей у галузь програмної інженерії [15].

Важливо також, на нашу думку, навести визначення терміна «компетентність». Так, А. Хуторської розрізняє такі поняття як «компетенція» і «компетентність» і у своїх роботах подає такі визначення цих термінів: «Компетенція – це сукупність взаємопов'язаних якостей особистості (знань, умінь, навичок, способів діяльності), які задаються щодо певного кола предметів і процесів і є необхідними для якісної продуктивної діяльності в цій сфері» [16]. Компетентність же «характеризує оволодіння людиною відповідною компетенцією на базі досвіду, що включає її особисте ставлення до компетенції та предмета діяльності» [16].

У контексті нашого дослідження слід навести визначення нетехнічних компетентностей. Отже, **нетехнічні компетентності** – це такі компетентності, що безпосередньо не пов'язані з контролем за обладнанням і технічною майстерністю. Вони охоплюють такі аспекти поведінки, як когнітивні компетентності (наприклад, ситуаційна обізнаність, ухвалення рішень, управління помилками тощо) і компетентності міжособистісного спілкування (див. рис. 1).

Розглянемо більш детально всі види нетехнічних компетентностей.

Так, у словнику компетентностей Гарвардського університету наведено універсальні нетехнічні компетентності, що необхідні інженерам-програмістам, а також спеціалістам інших галузей для ефективного виконання їхньої роботи, покладених на них функцій і завдань [17; 18]. Розглянемо когнітивні компетентності інженерів-програмістів:

- **Адаптивність** – фахівець швидко адаптується до змін і з легкістю розглядає нові підходи до будь-якої сфери діяльності. Ключові характеристики інженерів-програмістів у межах цієї компетентності:

- виявляє гнучкість і відкритість до нових ідей, підтримує інших у процесі змін;
- ідентифікує й розуміє зміни в робочому процесі, а також розпізнає підстави для таких змін;
- ефективно мислить і діє в стресовій ситуації;
- шукає нові способи розв'язання проблем, а також нові можливості в умовах зміни наявних обставин і/або виникнення нових;
- змінює робочі стратегії відповідно до нових даних;
- організовує графік роботи й прогнозовані результати відповідно до потреб;
- долає перешкоди задля досягнення результату;
- визначає способи й можливості включення нових методів до робочого процесу.

- **Безперервне навчання**

- **Безперервне навчання й саморозвиток** – спеціаліст прагне постійно досліджувати нові перспективи, підходи й моделі поведінки, а також вживати заходи для оцінки й підвищення своєї продуктивності. Ключові характеристики інженерів-програмістів у межах цієї компетентності:

- демонструє відкрите, зацікавлене й неупереджене ставлення до думок інших людей;
- з повагою ставиться до думки іншої людини, навіть якщо вона відміна від власної;
- визначає складні завдання й шукає можливості для особистісного зростання;
- шукає сфери для власного розвитку, отримуючи відповідні знання від інших людей або з різних інформаційних джерел (таких, як професійні організації, публікації тощо);
- бере участь у відповідних навчальних заходах, що задовольняють потреби в саморозвитку;
- ставить перед собою конкретні цілі задля досягнення бажаних результатів, що виправдовують або перевершують очікування;
- застосовує на практиці, у своєму робочому процесі нові знання у сфері технологій і бізнесу.

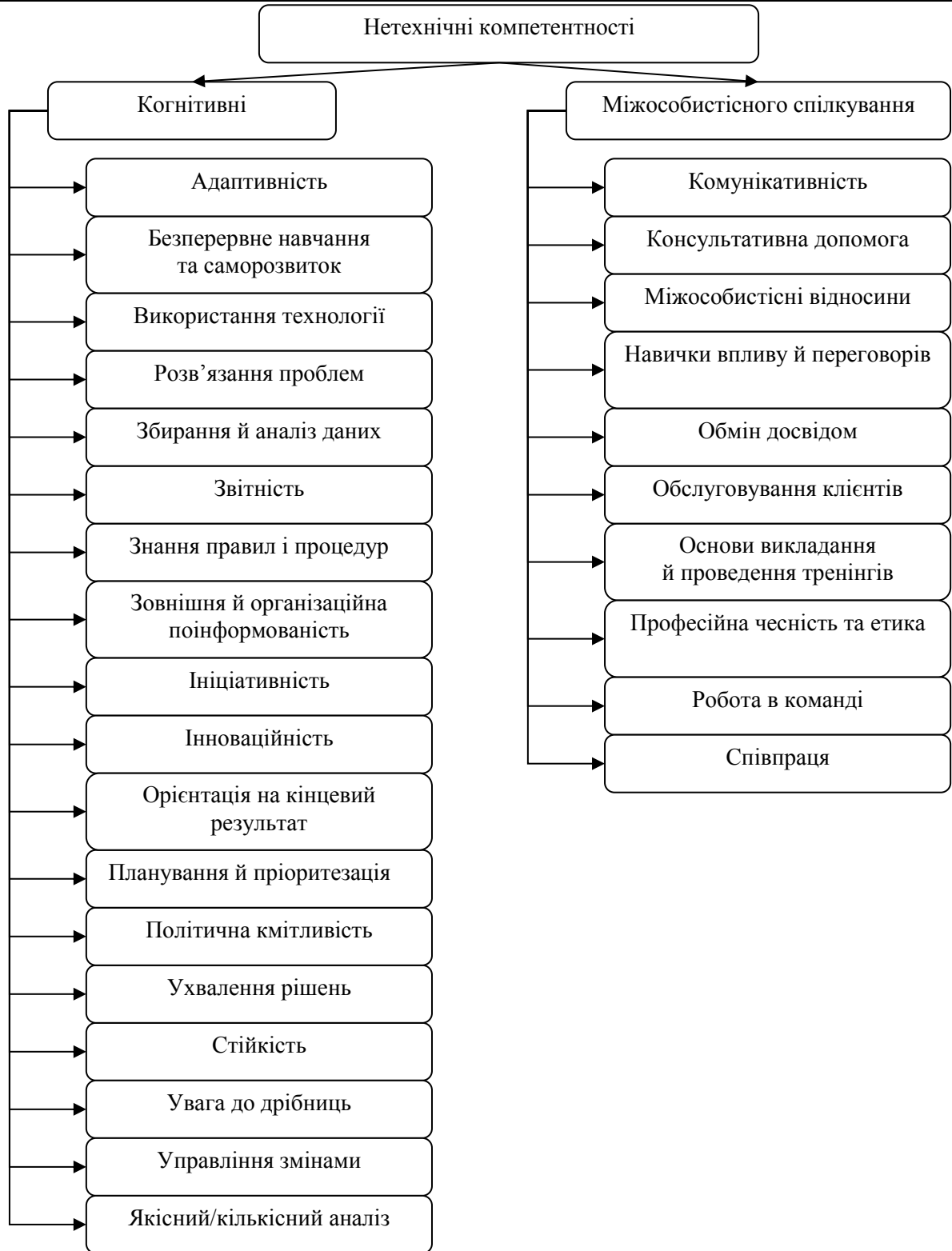


Рис. 1. Класифікація нетехнічних компетентностей інженерів-програмістів відповідно до очікуваних аспектів поведінки

- **Використання технології** – фахівець прагне використовувати нові технології для оптимізації особистої й організаційної продуктивності. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - уміло використовує сучасні технології та ресурси для поліпшення робочого процесу;
 - демонструє навички використання онлайн інструментів, бізнес-додатків та інших корисних матеріалів у відповідній сфері діяльності;
 - розуміє, як застосовувати необхідні програми та пошукові системи в роботі;
 - активно користується наявними технологіями та офісним

- у статкуванням, а також стежить за їх оновленням;
- використовує веб-ресурси для пошуку необхідних знань і відомостей;
 - уводить, поновлює й дістає необхідні дані з програм на ПК (наприклад, з наявних баз даних);
 - підтримує ведення електронної та паперової адміністративної документації в загальному доступі як резервних копій, а також для відстеження змін у завданнях команди;
 - створює звіти в рамках повсякденних завдань і у відповідь на особливі запити за допомогою різних інформаційних систем.
- **Розв'язання проблем** – спеціаліст виявляє проблеми, а також використовує логіку, здоровий глузд і наявні дані для оцінки альтернативних рішень і рекомендує відповідні рішення для досягнення поставленої мети та результату. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - визначає й оцінює проблеми, а також їхні чинники для виявлення їх потенційних наслідків;
 - пропонує розв'язання проблем, урахувавши політичні, організаційні й особистісні фактори;
 - визначає варіанти виконання завдань, а також оцінює сильні та слабкі сторони кожного з варіантів;
 - керується логікою для організації й аналізу даних і знань;
 - ретельно досліджує проблеми, не квапиться з висновками й не висловлює думки, що спираються лише на припущення й неповні дані;
 - звертається до внутрішніх нормативних документів за потреби виявити проблему, а також надає рекомендації щодо її розв'язання шляхом комунікації, допомоги, навчання тощо;
 - співпрацює з іншими колегами, спеціалістами конкретної сфери й керівниками вищих ланок у питаннях інтерпретації внутрішньої політики, надання рекомендацій і розв'язання інших проблем.
 - **Збирання й аналіз даних** – фахівець шукає, збирає та синтезує дані й знання, здобуті з різних джерел, і неупереджено доходить висновків або формулює мету чи судження, що слугують базою для ухвалення рішень і формування стратегій. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - аналізує й досліджує актуальні відомості, здобуті в різних форматах і з різних джерел;
 - аналізує дані для того, щоб дійти висновків і встановити причинно-наслідкові зв'язки на підтримку рішень керівництва, а також для надання рекомендацій щодо курсу дій і розв'язання організаційних питань;
 - документує джерела даних і знань та перевіряє достовірність даних для виявлення невідповідностей;
 - проводить спеціальні й стандартні аналізи відповідно до потреб клієнтів, включаючи в аналіз відповідні дані поточних проектів;
 - збирає та інтерпретує необхідні дані з використанням різних методів збирання даних;
 - визначає питання, проблеми й можливості, а також починає діяти, коли це необхідно;
 - відстежує тренди, тенденції та взаємозалежності в здобутих даних;
 - використовує зовнішні мережі для забезпечення інтеграції всіх наявних і відповідних знань в аналізі та інтерпретації даних;
 - влаштовує ділові зустрічі з клієнтами, щоб зрозуміти їхні потреби й проблеми, для належного використання даних, аналітики та звітності;
 - готує відомості в разі виникнення перевірки або за потреби передачі даних;
 - проводить аналіз трендів і проєкцій відповідно до стратегічних потреб;
 - шукає додаткові ресурси в разі виявлення прогалів і розбіжностей у даних.
 - **Звітність** – спеціаліст бере на себе відповідальність за успішне досягнення поставлених цілей і результатів, встановлюючи високі стандарти роботи для себе та інших. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - бере на себе відповідальність за позитивні й негативні результати роботи;

- чітко окреслює вимоги до делегованих обов'язків, зокрема й стосовно ухвалення рішень, необхідних дій і результатів;
 - демонструє розуміння зв'язків між власними обов'язками й загальним проектом, програмою, організаційними цілями та потребами;
 - зосереджується сам і фокусує інших на досягненні результатів роботи;
 - допомагає співробітникам і підтримує їх у роботі для досягнення успіху організації;
 - оцінює ризики, що можуть стати на заваді успіху організації, і вживає заходи для їх зменшення;
 - бере особисту участь у досягненні успіху організації.
- **Знання правил і процедур** – фахівець розуміється на документації й застосовує знання про статuti, положення, правила й процедури. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - підтримує належний рівень знань відповідних законодавчих актів, положень, правил і процедур у відповідних галузях;
 - гарантує виконання роботи відповідно до законодавчих актів, положень, правил і процедур у належні терміни;
 - надає рекомендації та роз'яснення зацікавленим сторонам щодо законодавчих актів, положень, правил і процедур.
 - **Зовнішня й організаційна поінформованість** – спеціаліст визначає внутрішні та зовнішні тенденції (наприклад, економічні, політичні, соціальні тощо) і розуміє, як вони впливають на роботу організації. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - ураховує можливості організації, її місію, бачення, цінності та стратегічні цілі у власному робочому процесі;
 - демонструє розуміння взаємозв'язків між зарубіжними та внутрішніми установами і організаціями;
 - демонструє розуміння функцій і обов'язків організації;
 - знає, до кого звернутися в разі виникнення питань щодо технічних аспектів робочого процесу;
 - визначає можливості розвитку організації, базуючись на унікальних культурних потребах;
 - обізнаний у формальних і неформальних звітних зв'язках і відносинах;
 - знає, як і коли передавати на розгляд керівництву питання, що потребують нагального розв'язання;
 - усвідомлює можливості й обмеження певної організації;
 - цікавиться новинами й відомостями про політичні та соціальні тенденції, що можуть вплинути на роботу організації;
 - обережно й розсудливо підходить до розв'язання делікатних питань.
- **Ініціативність** – фахівець виявляє наявні можливості та проблеми, а також активно діє в напрямі досягнення вигоди й розв'язання проблем. Ключові характеристики інженерів-програмістів у цієї компетентності:
 - активно шукає й виявляє можливості для досягнення цілей;
 - підтримує належний рівень трудової активності для досягнення поставлених цілей;
 - усвідомлює мету й цінність своєї роботи, а також відчуває відповідальність за неї;
 - не втрачає можливостей, що сприяють досягненню мети;
 - уміє працювати самостійно й без детальних вказівок;
 - виходить за межі вимог своєї роботи, шукаючи способи покращення роботи всієї організації.
 - **Інноваційність** – спеціаліст розробляє нові ідеї, а також застосовує різноманітні, зокрема й новітні, рішення для їх поліпшення. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - створює умови для роботи, що стимулюють творче мислення й упровадження інноваційних ідей;
 - шукає можливості для розвитку й покращення бізнесу;
 - досліджує різні ідеї, методики й альтернативні підходи до досягнення результатів;
 - упроваджує нові концепції та стратегії, що значно поліпшують спосіб ведення бізнесу;
 - оцінює, ураховує потенційні ризики та керує ними для досягнення цілей;

- пропонує альтернативні шляхи для виявлення й розв'язання проблем; не обмежується загальноприйнятими підходами;
- поєднує окремі унікальні ідеї й встановлює зв'язки між різнорідними ідеями; розглядає ідеї та ситуації під різними кутами зору; заохочує колективне обговорення різних підходів і стратегій.
- **Орієнтація на кінцевий результат** – фахівець фокусується на бажаному результаті, ставить складні стимулювальні цілі й досягає їх. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - визначає та зустрічається з відповідними сторонами задля вироблення загального розуміння цілей проекту та бажаних результатів;
 - розробляє й використовує метрики та індикатори досягнення поставленої мети й результатів;
 - використовує відповідні засоби й методи для того, щоб проекти залишалися в рамках поставлених цілей і в рамках бюджету;
 - демонструє гнучкий підхід у роботі для мінімізації ресурсів, що використовуються, у процесі досягнення поставлених цілей;
 - ефективно залучає співробітників до процесу досягнення цілей шляхом виявлення їхніх сильних сторін і відповідного розподілу завдань між ними.
- **Планування й пріоритизація** – спеціаліст планує й організовує трудову діяльність; здатний керувати декількома завданнями одночасно. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - передбачає проблеми й пом'якшує їхні наслідки;
 - оцінює відносну важливість діяльності та завдань; регулює пріоритети, коли це необхідно;
 - контактує з клієнтами й узгоджує з ними дії, якщо виникають проблеми або конфлікти, які можуть вплинути на вчасне завершення робіт;
 - контактує з іншими зацікавленими сторонами й узгоджує з ними дії за наявності численних конкурентних вимог для забезпечення вчасного завершення роботи;
- визначає необхідну послідовність дій та ефективний обсяг ресурсів, необхідних для досягнення короткострокових і довгострокових цілей;
- реалізує й використовує стратегічні плани на щоденній основі;
- підтримує високий рівень внутрішньої енергії та прихильності для виконання декількох завдань і пріоритетів одночасно, а також використовує наявні ресурси для досягнення «більшого» з найменшими витратами, при цьому не втрачаючи фокус;
- веде переговори щодо коригування термінів і/або обсягів робіт, якщо це необхідно; у разі потреби консультиється з керівником для визначення пріоритетів;
- організовує роботу, встановлює пріоритети й визначає потреби в ресурсах;
- проактивно інформує зацікавлені сторони про прогрес у виконанні роботи, а також про завершення роботи;
- визначає, бере до уваги й регулює взаємозалежності дій і ресурсів;
- бере на себе зобов'язання, встановлює та підтримує високі стандарти якості роботи й оперативності в наданні адміністративних послуг; легко переорієнтовує пріоритети для реагування на невідкладні й мінливі вимоги;
- використовує переваги наявних ресурсів (співробітників, відділів, процесів та інструментів) для ефективного виконання роботи;
- ефективно використовує час, а також запобігає виникненню проблем, що відволікають або негативно впливають на успішне завершення роботи.
- **Політична кмітливість** – фахівець проявляє впевненість і професійну дипломатію, і водночас налагоджує ефективні відносини з людьми на всіх рівнях, усередині організації та поза нею. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - демонструє розуміння ролі, обов'язків і взаємозв'язків усередині своєї організації;
 - налагоджує й підтримує професійні відносини;

- ураховує корпоративну культуру під час ухвалення рішень; розуміє вплив і наслідки ухвалених рішень;
 - визначає ситуації, що потребують втручання вищого керівництва, і попереджає відповідних посадових осіб про такі ситуації;
 - ставить з розумінням до оточення й поводить відповідно до ситуації;
 - ураховує організаційні й політичні аспекти та діє відповідно до них;
 - ураховує політичне середовище, пріоритети в галузі управління, ролі й обов'язки персоналу, а також зовнішні фактори, що впливають на організацію;
 - працює над досягненням цілей і забезпеченням позитивних результатів у своїй роботі;
 - дотримується корпоративних пріоритетів, культурних норм і «неписаних» правил для досягнення успіху на всіх рівнях організації;
 - з повагою ставить до позицій і думок інших людей щодо влади й політики, а також адаптує до них особистісні підходи.
- **Ухвалення рішень** – спеціаліст набуває необхідних знань, умінь і навичок, визначає ключові питання й наслідки ухвалення обґрунтованих та об'єктивних рішень. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - підходить до виконання складних завдань чи розв'язання проблеми через аналіз їх окремих складників;
 - розробляє різні альтернативні шляхи досягнення цілі, розв'язання проблем і/або уникнення ризиків і обирає найкращий з них;
 - активно шукає розбіжності й невідповідності в наявних даних;
 - ураховує витрати, вигоди, ризики та шанси на успіх у процесі ухвалення рішень;
 - формулює логічні висновки з урахуванням фактів в умовах складної та неоднозначної ситуації;
 - ухвалює обґрунтовані, вчасні та ефективні рішення, беручи до уваги лише необхідні дані, і консультує інших за потреби.
 - **Стійкість** – фахівець зберігає високу продуктивність і самоконтроль під тиском життєвих і професійних негараздів. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - контролює власні емоції й поводить стримано, незважаючи на робоче середовище;
 - закінчує роботу у визначені терміни, незважаючи на непередбачувані обставини;
 - визначає стратегії, що забезпечують належний рівень якості виконання робіт, незалежно від робочого середовища;
 - змінює пріоритетність виконання робіт, якщо висуваються додаткові вимоги.
 - **Увага до дрібниць** – спеціаліст гарантує повноту й точність даних і знань; стежить за тим, щоб домовленості й зобов'язання були виконані. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - дотримується в робочому процесі тих кроків, що передбачені стандартними оперативними процедурами;
 - ознайомлюється з робочими матеріалами й перевіряє їх точність і зрозумілість;
 - забезпечує контроль за якістю виконання робіт.
 - **Управління змінами** – фахівець розуміє необхідність змін і допомагає планувати їх і пристосуватись до них настільки творчо й позитивно, наскільки це можливо. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - виявляє гнучкість і відкритість до нових ідей, підтримує інших у процесі змін;
 - шукає шляхи використання наявних технологій для досягнення цілей і покращення результатів власної роботи й роботи організації загалом;
 - пропонує використовувати в роботі нові підходи, методи й технології;
 - використовує найбільш сучасні ресурси та технології у власній роботі й у роботі організації;
 - оцінює готовність організації та її працівників до змін перед тим, як запроваджувати інновації в робочий процес;
 - визначає ймовірні труднощі, що можуть виникнути під час реформування робочих процесів і принципів керівництва;
 - ураховує загальну структуру, закономірності та цикли в робочому процесі організації та її систем, а також використовує методики аналізу та оцінки для визначення

показників і стандартів виконання робіт;

- оцінює досягнення місії та цілей організації, оцінює результати впровадження змін у робочі процеси, а також зміни в поведінці співробітників, результати їхнього навчання та їхні реакції/відгуки; формулює відповідні висновки.

- **Якісний/Кількісний аналіз** – спеціаліст аналізує й оцінює наявні дані для управління й досягнення результатів. Ключові характеристики інженерів-програмістів у межах цієї компетентності:

- збирає та інтерпретує дані з різних джерел, а також виявляє тенденції, що доступні через звичайні і/або альтернативні джерела;
- аналізує наявні дані для того, щоб виконати необхідні порівняння й сформулювати висновки;
- використовує визначені моделі аналізу даних, а також відповідні інструменти для якісного й кількісного аналізу даних;
- визначає причинно-наслідкові зв'язки для розв'язання складних проблем.

До компетентностей міжособистісного спілкування інженерів-програмістів належать:

- **Комунікативність** – фахівець забезпечує чітку та ефективну комунікацію, а також бере на себе відповідальність за розуміння інших. Ключові характеристики інженерів-програмістів у межах цієї компетентності:

- чітко і ясно передає знання та дані в усній формі;
- ставить доречні запитання;
- чітко формулює й висловлює ідеї в усній і письмовій формі;
- уважно слухає, ставить запитання для уточнення почутого, підсумовує почуте для перевірки розуміння;
- забезпечує комунікацію за потреби обговорення питань щодо роботи, керівництва або особистісних ситуацій;
- використовує аналогії, візуальні ефекти та інші методи для того, щоб адаптувати комунікацію до конкретної аудиторії;
- визначає й використовує ефективні способи та методи комунікації (презентації, електронні листи, соціальні медіа);
- використовує навички презентації даних і знань, аналізу ідей і думок

у чіткій, ясній, лаконічній і переконливій формі, урахувавши при цьому особливості аудиторії.

- **Консультативна допомога** – спеціаліст забезпечує відповідний консультативний супровід, зворотній зв'язок і необхідні ресурси. Ключові характеристики інженерів-програмістів у межах цієї компетентності:

- консулює керівників і співробітників щодо набуття відповідних знань, умінь і навичок, а також пояснює основні вимоги до їх документації;
- надає допомогу керівникам у процесі підготовки відповідних обґрунтувань, а також під час підготовки термінових замовлень і заяв;
- розробляє специфікації для замовлення товарів і послуг;
- оцінює якість послуг продавців і постачальників;
- роз'яснює керівництву вимоги й обмеження щодо документації та трудового розпорядку;
- допомагає оформляти й подавати необхідні документи;
- забезпечує точне й вчасне керівництво;
- надає консультації керівникам щодо вимог і дій, необхідних для наймання контрактних працівників;
- забезпечує консультативний супровід за допомогою перевірки, проведення переговорів, оцінки пропозицій і процесу їх внесення;
- забезпечує даними й знаннями у разі виявлення проблем, а також рекомендації рішень;
- запитує дані, щоб зрозуміти проблеми, очікування й потреби клієнтів у відповідній галузі;
- слугує консультантом і ресурсом для поліпшення процесу, дизайну, функціональності й управління;
- бере участь у плануванні, підготовці, організації та управлінні даними й матеріалами для підтримки керівництва й персоналу.

- **Міжособистісні відносини** – фахівець започатковує й підтримує ефективні відносини з іншими людьми; добре ставиться до людей іншого походження в різноманітних ситуаціях; виявляє розуміння, увічливість, тактовність, співпереживання, турботу й люб'язність.

Ключові характеристики інженерів-програмістів у межах цієї компетентності:

- ставиться до людей відкрито, по-дружньому й професійно;
 - працює й співпрацює так, щоб отримати підтримку й прихильність з боку інших у процесі виконання поставлених завдань;
 - обговорює предмет завдання в конструктивній манері з персоналом на всіх рівнях;
 - налагоджує співробітництво, співпрацю й комунікацію для досягнення консенсусу й виконання завдань;
 - демонструє навички дипломатії шляхом обережного й тактовного звернення до інших під час обговорення болючих питань;
 - формує середовище, що сприяє відкритим і прозорим зв'язкам на всіх рівнях;
 - звертає увагу на те, що відчувають інші, точно інтерпретує ці відчуття, ґрунтуючись на виборі слів, тону голосу, виразі обличчя та інших видах невербальної поведінки;
 - зростає професійно для підтримки іміджу й зміцнення довіри до себе.
- **Навички впливу й переговорів** – спеціаліст використовує відповідні техніки переконання для отримання підтримки та співпраці з боку зацікавлених сторін, керівництва, колег, підлеглих та інших сторін для досягнення бажаного курсу організаційних дій відповідно до стратегічних цілей і завдань організації. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - демонструє здатність впливати на інших (за потреби);
 - допомагає персоналу й зацікавленим сторонам бути обізнаними з цілями й завданнями та усвідомлювати їх, а також спрямовує рішення в напрямі досягнення бажаних цілей, водночас підтримуючи ефективні відносини й досягаючи консенсусу;
 - використовує всі доступні дані й знання як належить для проведення переговорів і ухвалення рішень для досягнення поставлених цілей.
 - **Обмін досвідом** – фахівець використовує формальні, неформальні й систематичні методи поширення даних і знань між зацікавленими сторонами, що веде до підвищення організаційної ефективності. Ключові характеристики

інженерів-програмістів у межах цієї компетентності:

- створює стандартні процедури та інші керівні вказівки, що сприяють безперервній діяльності, а також організаційній стійкості;
 - визначає нові підходи до організації процесів і процедур для підвищення ефективності й продуктивності зазначених процесів і процедур, а також ділиться такими знаннями й навичками з іншими співробітниками для забезпечення максимальної ефективності;
 - знає й розуміє організаційні та програмні цілі, а також шукає творчі шляхи для покращення процесу досягнення мети, обмінюючись цими знаннями й досвідом з іншими сторонами й співробітниками.
- **Обслуговування клієнтів** – спеціаліст демонструє прихильність до сфери надання публічних послуг; задовольняє запити внутрішніх і зовнішніх клієнтів; самостійно несе відповідальність за досягнення якісних результатів. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - встановлює й підтримує довіру з боку громадськості, партнерів, клієнтів і співробітників;
 - працює з клієнтами для оцінки їхніх потреб і задоволення/перевершення їхніх вимог і очікувань;
 - справляє позитивне враження під час спілкування з клієнтами й співробітниками;
 - окреслює можливості для покращення обслуговування клієнтів і підвищення їхнього задоволення;
 - розуміє клієнтів і обирає проактивний підхід для ефективного й вчасного задоволення їхніх потреб;
 - заохочує клієнтів і заручається їхньою підтримкою як партнерів у наданні послуг;
 - використовує спеціальні знання для допомоги клієнтам у розв'язанні їхніх проблем;
 - уміло використовує «нетехнічну» мову для забезпечення ефективної взаємодії з клієнтами;
 - визнає, що зворотній зв'язок з клієнтами надає можливості підвищення ефективності надання послуг, а також ідентифікує

- необхідність упровадження відповідних змін у наданні послуг;
- буде та підтримує ефективні робочі відносини з партнерами та клієнтами;
 - збалансовує інтереси різних типів клієнтів;
 - встановлює та підтримує високі стандарти якості роботи й оперативності надання послуг, бере на себе відповідні зобов'язання.
- **Основи викладання й проведення тренінгів** – фахівець визначає функціональні потреби; планує, розробляє, постачає навчальні програми та тренінги й керує ними. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - шукає, розробляє й проводить корисні та доречні тренінги для підвищення продуктивності праці співробітників і партнерів;
 - проводить збір інформації щодо зворотного зв'язку (рівня задоволеності), як результату проведених тренінгів, і формулює відповідні висновки.
 - **Професійна чесність та етика** – спеціаліст виявляє вихованість і сприяє підтримці бажаної поведінки відповідно до наявних норм організації та суспільства. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - ознайомлюється з необхідними нормами та правилами, що є чинними у відповідній галузі і/або організації;
 - дотримується норм і правил послідовно й неупереджено;
 - утримується від поведінки, що може спричинити конфлікт інтересів;
 - зберігає й не розголошує конфіденційні дані;
 - забезпечує необхідний захист і делікатне поведіння з конфіденційними даними;
 - ставиться з розумінням до оточення й поводить відповідно до ситуації;
 - дотримується професійних та етичних норм;
 - інформує менеджера про виявлені випадки неетичної поведінки;
 - зберігає самовладання в будь-якій ситуації.
 - **Робота в команді** – фахівець працює разом з іншими людьми і сприяє досягненню спільних поставлених цілей. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - бере активну участь у командних завданнях і проєктах;
 - визначає сильні сторони свого професійного досвіду й охоче надає консультації робочим групам для досягнення максимальних результатів;
 - залучає до спільної роботи інших членів команди;
 - докладає зусилля для створення атмосфери, що сприяє розвитку ефективної співпраці між членами команди;
 - ураховує відгуки колег і бере до уваги їхні пропозиції для досягнення колективних цілей.
 - **Співпраця** – спеціаліст працює спільно з іншими всередині й за межами організації для досягнення цілей, що ведуть до створення й підтримки взаємовигідних партнерських відносин, максимально ефективного використання знань і досягнення відповідних результатів. Ключові характеристики інженерів-програмістів у межах цієї компетентності:
 - визначає, організовує та підтримує взаємовигідні партнерські відносини з зацікавленими сторонами як усередині організації, так і поза її межами;
 - підтримує позитивні й продуктивні відносини;
 - співпрацює з іншими для обміну знаннями й досягнення цілей;
 - працює над досягненням цілей, що можуть принести користь команді, шляхом поширення ідей та активній участі у командних заходах;
 - сприяє створенню середовища, що стимулює обмін знаннями й активну участь у командних заходах;
 - сприяє досягненню консенсусу шляхом усунення розбіжностей у думках;
 - позитивно й конструктивно розв'язує конфлікти й непорозуміння;
 - визначає ситуації, що потребують сторонньої юридичної консультації;
 - співпрацює з зацікавленими сторонами для аналізу та покращення адміністративних процесів;
 - координує зусилля з зацікавленими сторонами для забезпечення обміну знаннями й даними.
- Так, когнітивні й міжособистісні компетентності, на яких ґрунтуються досвідченість і професіоналізм розробників

програмного забезпечення, слугують вимогами, що висувуються до компетентного інженера-програміста. Одним з головних чинників, що стимулює попит на професійні нетехнічні компетентності, є вимога використовувати й підтримувати гнучкі методології розробки програмного забезпечення. Так, наприклад, методологія розробки програмного забезпечення Agile заснована на інкрементальному й прозорому розвитку проєктів. Цей розвиток здійснюється на основі співпраці між самоорганізованими, крос-функціональними командами розробки програмного забезпечення. Agile команди багато в чому залежать від ефективної комунікації, узяті на себе відповідальності, ініціативи, управління часом, дипломатії та керівництва [6].

Наголосимо, що процес розробки програмного забезпечення є командною діяльністю, і успіх проєкту програмного забезпечення залежить від ефективної роботи команди проєкту загалом. Так, Мішель Вест зауважує, що робота в команді не означає автоматичного успіху. Звичайне об'єднання людей не гарантує, що вони будуть функціонувати як ефективна команда [19].

Важливість навчання майбутніх інженерів-програмістів управління проєктами програмного забезпечення завжди визнається такими організаціями, як Асоціація обчислювальної техніки (АОТ) та IEEE-Computer Society, і підтримується ними в спільній навчальній програмі. Ці організації не тільки підкреслюють вагомість цього питання, а й наголошують на необхідності навчання управління проєктами програмного забезпечення в повному обсязі саме на практиці, де учні зможуть перевірити свої знання й навички в реальних сценаріях. Таким чином, за допомогою такого практичного навчання майбутні інженери-програмісти зможуть пройти професійну практику саме в процесі навчання, а не після його завершення [2].

Отже, беручи до уваги важливість навчання в галузі управління проєктами програмного забезпечення, науковці (Д. Гейст, М. Майерс та І. Ібрагім) дійшли висновку, що дисципліни, пов'язані з управлінням проєктами програмного забезпечення, як і раніше, викладаються в основному в теоретичному форматі і, як наслідок, студенти не виявляють особливого інтересу до них [20].

На відміну від інших галузей, зокрема таких, як медицина або авіонавтика, майбутні інженери програмісти починають своє професійне життя з серйозним браком реальних практичних навичок. У результаті фахівці повинні розвивати свої навички й набувати досвіду, працюючи в реальних проєктах, де наслідки неадекватного планування або неправильні рішення можуть призвести до провалу всього проєкту або втрати значного прибутку [2].

Типовій освіті інженерів-програмістів бракує практичного опанування процесів розробки програмного забезпечення. Зазвичай, студенти під час лекцій лише ознайомлюються з відповідними теоріями щодо процесів розробки програмного забезпечення, але мають доволі обмежені можливості практичного втілення цих теорій у життя у відповідних дисциплінах [7].

Висновки. Отже, у статті розглянуто сутність категорії «нетехнічні компетентності майбутніх інженерів-програмістів». Наведено класифікацію нетехнічних компетентностей інженерів-програмістів, уточнено ключові характеристики кожної з них. З огляду на викладений у статті матеріал, одним з перспективних напрямів подальшого дослідження, вважаємо аналіз проблеми формування нетехнічних компетентностей майбутніх інженерів-програмістів, висвітлений у зарубіжній і вітчизняній науковій літературі.

Список використаних джерел

1. Shane Hastie. Standish Group 2015 Chaos Report-Q&A with Jennifer Lynch / Shane Hastie, Wojewoda Stéphane. – Retrieved 1. 15. – 2016.
2. Alejandro Calderón. Bringing Real-life Practice in Software Project Management Training Through a Simulation-based Serious Game / Alejandro Calderón, Mercedes Ruiz. – CSEDU (2). – 2014.
3. Mehdi Jazayeri. The education of a software engineer / Mehdi Jazayeri. – Proceedings of the 19th IEEE international conference on Automated software engineering. – IEEE Computer Society, 2004.
4. Shanika Karunasekera. Preparing software engineering graduates for an industry career / Shanika Karunasekera, Kunal Bedse. – 20th Conference on Software Engineering Education & Training (CSEET'07) IEEE, 2007.
5. Faheem Ahmed. Software Requirements Engineer: An Empirical Study about Non-Technical Skills / Faheem Ahmed. – Journal of Software 7.2. – 2012.

References

1. Shane, Hastie. *Standish Group 2015 Chaos Report-Q&A with Jennifer Lynch*. Retrieved 1.15. – 2016. [in English]
2. Calderón, Alejandro & Ruiz, Mercedes (2014). *Bringing Real-life Practice in Software Project Management Training Through a Simulation-based Serious Game*. CSEDU (2). [in English]
3. Mehdi, Jazayeri (2004). The education of a software engineer. *Proceedings of the 19th IEEE International Conference on Automated Software Engineering*. IEEE Computer Society. [in English]
4. Karunasekera, Shanika & Bedse, Kunal (2007). Preparing software engineering graduates for an industry career. *20th Conference on Software Engineering Education & Training (CSEET'07) IEEE*. [in English]
5. Faheem, Ahmed (2012). Software Requirements Engineer: An Empirical Study about Non-Technical Skills. *Journal of Software 7.2*. [in English]
6. Bender, Lisa (2014). *Measurement of the Non-Technical*

6. Lisa Bender. Measurement of the Non-Technical Skills of Software Professionals: An Empirical Investigation / Lisa Bender. – SEKE, 2014.
7. Emily Navarro. SimSE: A Software Engineering Simulation Environment for Software Process Education DISSERTATION / Emily Navarro. – UNIVERSITY OF CALIFORNIA, IRVINE, 2006.
8. Ritika Atal. Anukarna: A Software Engineering Simulation Game for Teaching Practical Decision Making in Peer Code Review / Ritika Atal, Ashish Sureka. – 2015.
9. Craig Caulfield. A systematic survey of games used for software engineering education / Craig Caulfield. – Modern Applied Science 5.6. – 2011.
10. Alain Abran. Guide to the software engineering body of knowledge / Alain Abran. – IEEE Computer Society 8.9. – Los Alamitos, CA, 2004.
11. Göran Backlund. Training young engineers to see / Göran Backlund, Jan Sjunnesson. – AI & society 27.4. – 2012.
12. Lee Fui Tong. Identifying essential learning skills in students' engineering education / Lee Fui Tong. – Proceedings of HERDSA, 2003.
13. Charles R. Woratschek. Information systems entry-level job skills: a survey of employers / Charles R. Woratschek, Terri L. Lenox. – Proceedings of the Information Systems Educators Conference, San Antonio. – TX. – Vol. 19. – 2002.
14. Paul Gruba. Strategies for communication skills development / Paul Gruba, Reem Al-Mahmood. – Proceedings of the Sixth Australasian Conference on Computing Education-Volume 30, Australian Computer Society, Inc., 2004.
15. Aseel Berglund. Integrating Soft Skills into Engineering Education for Increased Student Throughput and more Professional Engineers / Aseel Berglund, Fredrik Heintz. – LTHs 8 : e Pedagogiska Inspirationskonferens, Lund, Sweden, 17 december 2014. – Lunds university, 2014.
16. Хуторской А. В. Ключевые компетенции и образовательные стандарты // Интернет-журнал «Эйдос». – 2002. – 23 апреля. – Режим доступа : <http://www.eidos.ru/journal/2002/0423.htm>
17. Competency dictionary. Harvard University / Retrieved: 17.11.2016. – Режим доступа : http://campusservices.harvard.edu/system/files/documents/1865/harvard_competency_dictionary_complete.pdf
18. Competencies Dictionary. Office of Human Resources at the Global Institute of Health / Retrieved: 19.11.2016, URL : <https://hr.od.nih.gov/workingatnih/competencies/core/#non>.
19. Michel West. Effective teamwork: practical lessons from organisational research / Michel West. – EPS Blackwell, Oxford. – 2004.
20. Inurina Ibrahim. Teaching project management for it students: methods and approach / Inurina Ibrahim. – 2nd International Conference on Education and Management Technology. – 2011.
- Skills of Software Professionals: An Empirical Investigation.* SEKE. [in English]
7. Navarro, Emily (2006). *SimSE: A Software Engineering Simulation Environment for Software Process Education DISSERTATION.* UNIVERSITY OF CALIFORNIA, IRVINE. [in English]
8. Ritika, Atal & Ashish, Sureka (2015). *Anukarna: A Software Engineering Simulation Game for Teaching Practical Decision Making in Peer Code Review.* [in English]
9. Caulfield, Craig (2011). A systematic survey of games used for software engineering education. *Modern Applied Science* 5.6. [in English]
10. Abran, Alain (2004). *Guide to the software engineering body of knowledge.* IEEE Computer Society 8.9. Los Alamitos, CA. [in English]
11. Backlund, Göran & Sjunnesson, Jan (2012). *Training young engineers to see.* AI & society 27.4. [in English]
12. Fui Tong, Lee (2003). Identifying essential learning skills in students' engineering education. *Proceedings of HERDSA 2003.* [in English]
13. Woratschek, Charles R. & Lenox, Terri L. (2002). Information systems entry-level job skills: a survey of employers. *Proceedings of the Information Systems Educators Conference, San Antonio TX.* Vol. 19. [in English]
14. Gruba, Paul & Reem, Al-Mahmood (2004). Strategies for communication skills development. *Proceedings of the Sixth Australasian Conference on Computing Education.* Volume 30, Australian Computer Society, Inc. [in English]
15. Berglund, Aseel & Heintz, Fredrik (2014). Integrating Soft Skills into Engineering Education for Increased Student Throughput and more Professional Engineers. *LTHs 8: e Pedagogiska Inspirationskonferens, Lund, Sweden, 17 December 2014.* Lunds university. [in English]
16. Khutorskoy, A. V. (2002). Key competencies and educational standards. *Internet-zhurnal "Eydos".* Retrieved from: <http://www.eidos.ru/journal/2002/0423.htm> [in Russian]
17. Competency dictionary. Harvard University. Retrieved from: http://campusservices.harvard.edu/system/files/documents/1865/harvard_competency_dictionary_complete.pdf [in English]
18. Competencies Dictionary. Office of Human Resources at the Global Institute of Health. Retrieved from: URL: <https://hr.od.nih.gov/workingatnih/competencies/core/#non>. [in English]
19. West, Michel (2004). *Effective teamwork: practical lessons from organizational research.* EPS Blackwell, Oxford. [in English]
20. Ibrahim, Inurina (2011). Teaching project management for it students: methods and approach. *2nd International Conference on Education and Management Technology.* [in English]

Рецензент: Прийма С.М. – д.пед.н., доцент

Відомості про автора:

Концедайло Валерій Валерійович
me@valerykontsedaylo.com
 Житомирський державний університет
 імені Івана Франка
 вул. Велика Бердичівська, 40, м. Житомир,
 10008, Україна

Information about the author:

Kontsedailo Valerii Valeriiovych
me@valerykontsedaylo.com
 Zhytomyr Ivan Franko
 State University
 40 Velyka Berdychivska St., Zhytomyr,
 10008, Ukraine

doi: <http://dx.doi.org/10.7905/nvmdpu.v0i18.1917>

doi: <http://dx.doi.org/10.7905/nvmdpu.v0i18.1917>

Матеріал надійшов до редакції 31. 01. 2017 р.
 Прийнято до друку 28. 02. 2017 р.

Received at the editorial office 31. 01. 2017.
 Accepted for publishing 28. 02. 2017.