



[DOI 10.28925/2663-4023.2023.19.619](https://doi.org/10.28925/2663-4023.2023.19.619)

УДК 004.4:316.77

Трофименко Олена Григорівна

кандидат технічних наук, доцент, доцент кафедри інформаційних технологій,
Національний університет "Одеська юридична академія", м. Одеса, Україна,
ORCID ID: 0000-0001-7626-0886
trofymenko@onu.edu.ua

Савельєва Оксана Степанівна

доктор технічних наук, професор, професор кафедри інтегрованих технологій управління
Національний університет "Одеська політехніка", м. Одеса, Україна,
ORCID ID: 0000-0002-0453-4777
saveleva@op.edu.ua

Прокоп Юлія Віталіївна

кандидат історичних наук, доцент, доцент кафедри інженерії програмного забезпечення
Національний університет "Одеська політехніка", м. Одеса, Україна,
ORCID ID: 0000-0002-6608-3668
prokop.y.v@op.edu.ua

Логінова Наталія Іванівна

кандидат педагогічних наук, доцент, завідувачка кафедри інформаційних технологій,
Національний університет "Одеська юридична академія", м. Одеса, Україна,
ORCID ID: 0000-0002-9475-6188
loginova@onu.edu.ua

Дика Анастасія Іванівна

асистент кафедри інформаційних технологій,
Національний університет «Одеська юридична академія», Одеса, Україна,
ORCID ID: 0000-0002-4196-8734
dyka.anastasiia@gmail.com

SOFT SKILLS ДЛЯ РОЗРОБНИКІВ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Анотація. Нині сформувався розуміння важливості поведінкових компетентностей для ІТ-фахівців. Аналіз публікацій підтверджує зростання потреб і зацікавленості розробників програмного забезпечення у розвитку не лише технічних навичок, а й соціальних (поведінкових) компетентностей для ефективного та дієвого виконання професійної діяльності. Під час співбесід з рекрутами ІТ-компанії намагаються з'ясувати не лише рівень hard skills, а і наявність набору soft skills. На відміну від hard skills, soft skills є спільною мовою емпатії, співпраці та добробуту людей на робочому місці. Їх варто розглядати цілісно, адже вони можуть змінюватися, відповідно до змін у житті людини. Вони формують стійкість і водночас гнучкість команди, що врешті решт впливає на успішність компанії. Кожна конкретна компанія сама визначає свою модель soft skills як набір компетентностей, необхідних для успішного виконання роботи в ній з метою досягнення її стратегічних цілей. Наразі не існує чітко визначеного, усталеного, загальноприйнятого переліку soft skills для розробників програмного забезпечення. У роботі проведено аналіз численних публікацій від ІТ-компаній та сформовано перелік soft skills, найбільш важливих для розробників програмного забезпечення з урахуванням специфіки комунікативних навичок відповідної галузі. Комунікація, робота в команді, управління часом, визнання помилок, емпатія, емоційний інтелект, терпіння, адаптивність, креативність, лідерство, розв'язання проблем і навички роботи з людьми є основними навичками soft skills, які потрібно розвинути розробнику програмного забезпечення, щоб успішно розв'язувати свої професійні завдання в команді, значно покращити продуктивність та підвищити загальне задоволення від своєї роботи. Також у статті досліджено ключові проблеми при формуванні відповідних soft skills та способи їх подолання. Врахування важливості та розвитку соціальних компетентностей працівників сприяє їх спільному залученню до роботи, ефективній взаємодії та зацікавленості



у досягненні гарних результатів, високій якості й продуктивності для довгострокової стійкості компанії у глобальному професійному середовищі.

Ключові слова: комунікація; розробники програмного забезпечення; соціальна компетентність; soft skills; hard skills; робота в команді.

ВСТУП

На сьогодні IT-компанії дійшли до розуміння цінності й важливості soft skills для IT-спеціалістів на рівні, співставному з професійними компетентностями (hard skills). Soft skills – це, по суті, набори (моделі) компетентностей поведінкових реакцій, які дають змогу фахівцю бути ефективнішим, у порівнянні з іншими працівниками з подібним рівнем hard skill, але без таких компетентностей. Часто різниця між хорошим розробником і чудовим розробником значною мірою залежить від навичок спілкування. Не рідкими є ситуації, коли на роботу в IT-компанію беруть за hard skills, а звільняють через відсутність soft skills. Річ у тім, що професійні компетентності зазвичай формуються швидше і меншими зусиллями. Із соціальними (поведінковими) компетентностями все складніше – їхнє формування відбувається повільніше і з більшими зусиллями. Наприклад, формування компетентності лідерства або організованості та орієнтації на результат – задача нетривіальна і далеко не всім під силу. Тому, якщо стоїть вибір між зарозумілим висококваліфікованим фахівцем або професіоналом середньої руки, але з розвиненими soft skills, то на роботу візьмуть, скоріше за все, саме другого, адже навчити знанням технічної частини простіше, аніж намагатися змінити усталені егоїстичні поведінкові життєві принципи людини.

Постановка проблеми. Зважаючи на зазначене, формування і розвиток відповідних soft skills при підготовці фахівців IT-галузі є вельми затребуваними, а дослідження, спрямовані на виявлення ефективних підходів щодо формування необхідного набору соціальних компетентностей, є актуальними.

Аналіз останніх досліджень і публікацій. Як показав аналіз наукових досліджень, нині зазначеною проблемою переймаються здебільшого самі IT-компанії, а викладачі та науковці вищих навчальних закладів, які готують IT-фахівців, не приділяють достатньої уваги вивченню набору поведінкових компетентностей, необхідних для подальшої успішної співпраці в колективі розробників програмного забезпечення. На тому, що розвитку навичок soft skills приділяється недостатня увага на ранніх академічних курсах з програмної інженерії у вищій школі, наголошено у роботі [1]. Автори статті [2] розглядають питання важливості комунікативних навичок при розробці програмного забезпечення. У дослідженні [3] проаналізовано опитування 35 практиків з програмної інженерії в Уругваї, що дозволило з'ясувати важливість soft skills для лідера та інших учасників розробки програмного забезпечення. Зазначається [4], що приблизно 80% людей зазнають невдачі в роботі не через відсутність технічних навичок, а через нездатність налагодити стосунки у спілкуванні з колегами. У статті [5] досліджено зв'язок між soft skills і hard skills студентів-математиків. Дослідження [6] присвячено вивченню впливу організаційного навчання на вдосконалення hard skills, soft skills та інновацій на результативність і кар'єрний ріст фахівця загалом. Але ж кожна професія має свою специфіку, а тому набори soft skills і проблеми щодо їх формування для продуктивної роботи в колективі розробників програмного забезпечення мають свої особливості, які потребують вивчення.



Метою статті є висвітлення специфіки й формування переліку soft skills, важливих для ІТ-фахівця, дослідження ключових проблем при формуванні відповідних soft skills та способи їх подолання.

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Завдяки розумінню важливості наявності поведінкових компетентностей для ІТ-фахівців і спрямованості на їх формування ще під час навчання, нині розробляються стандарти вищої освіти, формуються та регулярно переглядаються освітні програми, які узгоджуються різними стейкхолдерами: роботодавцями, викладачами та здобувачами вищої освіти відповідно до сучасних, мінливих вимог і викликів ІТ-галузі. Нині вітчизняна ІТ-освіта кардинально змінює усталені підходи до академічної вищої освіти, успадковані з минулого, і відходить від обмеженої мети формування знань та навичок роботи з тими чи іншими програмними засобами у бік формування набору фахових загальних і спеціальних компетентностей та програмних результатів навчання. Прикладами таких поведінкових компетентностей є здатність генерувати нові ідеї (креативність), працювати в команді, бути критичним і самокритичним, приймати обґрунтовані рішення, оцінювати та забезпечувати якість виконуваних робіт тощо. До речі, для опису компетентності некоректно застосовувати терміни «знання» чи «вміння», адже далеко не кожна особистість може через свій досвід сформувати в собі поведінку лідера. Тут доречно казати саме про здатність або відсутність такої здатності до певної соціальної компетентності.

Під час співбесід з рекрутами ІТ-компанії намагаються з'ясувати не лише рівень hard skills, а і наявність набору soft skills. Кожна конкретна компанія сама визначає свою модель soft skills як набір компетентностей, необхідних для успішного виконання роботи в ній з метою досягнення її стратегічних цілей. Переважно така сформована модель є загальною для всіх позицій компанії, щоб не провокувати корпоративні негаразди й конфлікти між її структурними підрозділами.

Набір соціальних компетентностей, важливих для ІТ. Наразі не існує чітко визначеного, усталеного, загальноприйнятого переліку soft skills для розробників програмного забезпечення. Загалом поширено [7] – [11] бажаними soft skills для ІТ-компаній є:

1) Усна та письмова комунікація – важлива компетентність, яка стосується здатності слухати й чути ідеї інших, обговорювати, презентувати свої ідеї, вміти домовлятися. Для інженера-програміста важливо вміти спілкуватися, обговорювати свою роботу з членами команди, керівниками проєктів та зацікавленими сторонами, оскільки важливо повідомляти та узгоджувати свій прогрес у проєкті. Розробники програмного забезпечення часто можуть проводити години, працюючи наодинці над завданням. Але гарні комунікативні навички все одно необхідні, оскільки команда розробки потребує узгодженої та продуктивної співпраці над різними складовими програмного продукту. Комунікація є невіддільною складовою розробки програмного забезпечення. Обмін ідеями та надання зворотного зв'язку є природною частиною командної роботи, але щоб досягти успіху в цій сфері, важливо вміти чітко формулювати свої ідеї та ставитися до своїх колег з повагою. Два професіонали з протилежними ідеями повинні створювати не суперечку, а скоріше дискусію, а для цього потрібен емоційний інтелект. Спілкування в цій сфері вимагає відкритого мислення та бажання вислухати. Найкращим рішенням може бути синтез пропозицій від кількох людей, а для цього



важливо вміти слухати, доносити свої ідеї та вміти домовлятися. Спілкування може бути особливо складним, якщо додати фактор віддаленої роботи. Ефективна комунікація означає, що всі члени команди розробки програмного продукту розуміють і розділяють відповідальність щодо термінів, вимог до проекту, очікувань тощо. Ключ до ефективної комунікації – вміти слухати. Розробники програмного забезпечення часто опиняються в ситуації, коли тільки вони знають, як щось виправити або чому чужа ідея є нездійсненною. З іншого боку, важливою є навичка уважно, з повагою вислухати співрозмовника, налагодити зоровий контакт і вміти пояснити свою точку зору так, щоб усі її зрозуміли.

Корисним при навчанні та формуванні соціальної компетентності комунікації є [7]:

– *міжособистісне спілкування* – спосіб комунікації сам на сам, коли розробники програмного забезпечення коригують свій стиль спілкування залежно від того, з ким вони розмовляють. Наприклад, розробники можуть використовувати більш технічну мову під час розмови з колегою, а під час розмови з клієнтами – простіші й зрозумілі для останніх терміни;

– *активне слухання* є способом взаємодії на робочому місці, коли розробникам програмного забезпечення необхідно слухати, щоб розуміти та відповідати належним чином. Активне слухання означає зосередження на тому, що говорить доповідач, і використання невербальних сигналів, наприклад кивання, щоб показати, що ви розумієте його слова;

– *публічні виступи* – спосіб комунікації й презентації розроблених технологічних продуктів. Для навички публічних виступів важливою є чітка, довірлива вимова, яку легко сприйматиме аудиторія. Під час виступу можливе використання сучасних візуальних засобів, щоб зацікавити аудиторію.

2) Аналітичне та критичне мислення і навички розв'язання проблем – здатність адаптуватися, критично осмислювати й аналізувати ситуацію, приймати поінформовані рішення, тим самим розв'язувати проблеми. При чому це стосується не лише проблем безпосередньої програмної розробки, а й різних життєвих ситуацій у колективі. Також сюди відносять бажання навчатися і логічне мислення. В цілому ця соціальна компетентність є найважливішою для розробки програмного забезпечення. Зокрема, саме вона впливає на те, як розробник розв'язує складні проблеми та наскільки він мислить нестандартно. Може здатися, що розв'язання проблем вимагає лише поєднання *hard skills* і досвіду. Але вдаль розв'язання проблем і креативність є життєво важливими *soft skills*. Тому ІТ-компанії шукають розробників, які розглядають кожну проблему всередині та поза кодуванням як високотехнічний алгоритм із багатьма розгалуженнями для можливого успіху [8]. І при цьому розробник має залишити своє самолюбство та заохочувати інших розробників висловлювати свої ідеї. В реальності проблеми не повинні й часто не розв'язуються однією людиною. Статистика показує, що групи, які пройшли певне навчання *soft skills* щодо розв'язання проблем, створюють на 350% більше ідей, ніж групи, які цього не робили, а ідеї, які вони створюють, на 400% оригінальніші [9]. Ключем до розв'язання проблеми є творчість. Найкращий спосіб покращити креативність – не заохочувати її, а усунути будь-які перешкоди. Це означає запобігати песимізму та егоїзму, а також дозволяти людям робити помилки. Отже, для розробника програмного забезпечення вкрай важливо мотивувати та заохочувати себе бути більш творчими, але, що найважливіше, не знеохочувати інших. Коли розробники програмного забезпечення усувають неполадки своїх продуктів, щоб переконатися, що вони працюють належним чином, навички розв'язання проблем дозволяють їм виявляти та виправляти технічні помилки. Програмісти зазвичай аналізують свою роботу та



розробляють корисні рішення. Наприклад, якщо програма працює не так, як передбачалося, розробники можуть переглянути код, щоб визначити джерело проблеми та встановити програму для підвищення продуктивності програмного забезпечення.

3) Командна робота та співпраця – здатність працювати з людьми, поважати офісну політику та відмінності членів команди (культурні, етнологічні тощо), що важливо для сучасного глобального суспільства з культурним та етнічним розмаїттям. Розробка програмного забезпечення – це свого роду командний вид спорту. Навіть, якщо розробник працює сам, йому все одно доводиться спілкуватися з дизайнерами, керівниками проєктів, клієнтами тощо. Але робота в команді – це більше, ніж просто відправлення роботи в репозиторій. Це залучення інших навичок, як-от: управління часом, комунікація та емпатія, щоб уся команда могла рухатися як добре змащена машина. Незалежно від того, чи це розробники інтерфейсу, бекенда, повного стека чи DevOps, кожна окрема роль є важливою частиною складової пазла компанії, тому важливо ладнати. Понад 85% керівників і співробітників компаній стверджують, що головною причиною провалу проєкту була відсутність командної роботи, а понад 50% співробітників вважають, що середовище для співпраці та хороша мережа підтримки професіоналів змусили їх залишитися в компанії [9]. Командна робота не означає погодження всіх з усіма і про все. Насправді різні думки можуть бути тим, що робить компанії справді успішними. Ідея командної роботи полягає в тому, щоб знайти баланс між використанням своїх soft та hard skills і зробити проєкт максимально приємним й успішним для кожного члена команди. Завдяки згуртованій командній роботі розробники програмного забезпечення можуть продуктивно працювати зі своїми колегами для виконання поставлених завдань. Розробка та тестування програмного забезпечення може потребувати досвіду кількох професіоналів, наприклад інженера-програміста або дизайнера інтерфейсу користувача (UI), а колеги можуть мати різні стилі роботи. Розробники програмного забезпечення, які заохочені до співпраці, можуть налагодити позитивні стосунки на робочому місці зі своїми колегами та менеджерами, а також можуть вчитися у своїх колег.

4) Управління часом (тайм-менеджмент) – важлива навичка, коли мова йде про розробку програмного забезпечення. Працювати під тиском і все одно дотримуватись дедлайнів – це практично життя кожного успішного розробника програмного забезпечення. Адже керівники груп хочуть мати точні оцінки того, коли робота буде завершена. Розробники програмного забезпечення не тільки повинні звітувати перед керівниками проєктів, бізнес-менеджерами та іншими офіційними партнерами, але вони також відповідають за своєчасне доставлення кінцевого програмного продукту клієнтові. Тому дуже важливо мати стратегію управління часом. Ключовим для пошуку розробника з хорошими навичками тайм-менеджменту є звернення уваги на те, наскільки точними є оцінки потенційного розробника. Управління часом може включати створення розкладу для всього процесу розробки, де розробники резервують дні для виконання своїх завдань і доповідають один одному про їх вчасне завершення. Для кожного у команді розробників важливо знайти відповідь на запитання: «Чи розумно я витрачаю свій час?» і переконатися, що завдання, які виконуються, справді мають значення. Дослідження показують, що середній працівник витрачає 80% робочого дня на виконання неоцінених завдань [9]. З іншого боку, через багатозадачність і складність якісно виконувати водночас багато різних задач розробники швидко вигорають психологічно і фізично. Тому для ефективного розподілу та використання часу варто залучати ефективні техніки виконання завдань. Наприклад, поділ робочого часу на сегменти по 25 хвилин і 5-хвилинну перерву між ними, рішення з низьким вмістом коду

та повторне використання наявного коду є чудовими способами для розробників програмного забезпечення заощадити час і зосередитися на одному завданні за раз. Це виділить час для роботи над більш інноваційними та цікавими завданнями.

5) *Визнання помилок, підзвітність.* Значною частиною професійного розвитку в будь-якій галузі є визнання своїх помилок і просування вперед з новим досвідом, здобутим від виправлення помилки. Помилки є невіддільною частиною робочого процесу. Насправді можливість робити помилки сприяє появі нових та інноваційних ідей. Усвідомлення і визнання помилок дозволяє покращити робочий процес. Коли розробник бере на себе відповідальність за свої помилки, решта команди може скористатися цим прикладом, щоб пізніше виявити подібні помилки. Але інколи помилки можуть мати доволі серйозні наслідки, і, якщо їх не виявити й не виправити, вони можуть коштувати компанії дуже дорого і вплинути на репутацію. Тому життєво важливо створити робоче середовище, яке забезпечує зворотний зв'язок. Це означає публічно хвалити людей за виконану ними роботу, приватно висловлювати конструктивну критику та дозволяти людям робити те саме з вами. Причина цього полягає в тому, що коли люди отримують позитивний відгук, вони, швидше за все, визнають помилки. Люди ухиляються від відповідальності, коли відчувають, що їх недооцінюють і знецінюють.

6) *Емпатія, емоційний інтелект.* Наявність емпатії та емоційного інтелекту означає здатність поставити себе на чиесь місце і подумати: «Чи я хочу, щоб до мене ставилися так?». Хоча програмісти проводять багато часу з комп'ютерами, вони все одно повинні мати базове співчуття, адже навички інженера з програмного забезпечення починаються і закінчуються основами людської взаємодії. А емоційний інтелект є фундаментальним для будь-якої людської взаємодії.

Статистика показує, що понад 35% керівників компаній вважають, що хороший емоційний інтелект невдовзі стане обов'язковою умовою на робочому місці [9]. На посаді розробника програмного забезпечення емоційний інтелект означає тип реакції на складні ситуації на робочому місці. Завдяки високому емоційному інтелекту розробник зможе зберігати спокій у непередбачених обставинах, наприклад, якщо перевитрата бюджету або зміна вимог клієнтів можуть завадити успіху проекту [7]. Також працівник зможе наполегливо долати професійні виклики та зосереджуватися на досягненні організаційних цілей.

Інженери-програмісти повинні мати базову емпатію. Емпатія визначає здатність людини розуміти або відчувати те, що переживає інша людина в будь-який момент часу. Часто саме через брак емпатії в колективі виникають проблеми. Важливо розуміти, що відчувають колеги поруч. Крім команди розробників, співчуття поширюється на розуміння точки зору бізнес-аналітиків чи інженерів із забезпечення якості. Головне тут – терпляче і доброзичливо пояснити свою точку зору, без обвинувачень та принижень інших. До того, розробники також відповідають і за розуміння кінцевих користувачів. Щоб краще співпереживати та розуміти своїх користувачів, багато розробників використовують дизайнерське мислення. Важливо знати й цінувати навички та слабкі сторони кожного і, що більш важливо, бути достатньо усвідомленим, щоб знати свої власні.

7) *Терпіння.* Оскільки процес розробки програмного забезпечення може бути тривалим і складним, з моментами відчаю та розчарування, саме терпіння може допомогти розробникам підтримувати рівень продуктивності та продовжувати створювати якісний програмний продукт. Ця навичка особливо важлива для інженерів програмного забезпечення, коли справа доходить до виявлення помилок у коді, коли



доводиться переглядати код рядок за рядком і ніяк не вдається точно визначити проблему. Цей процес втомлює і вимагає лазерного фокусування на крихітних деталях, при цьому формується та накопичується дрібне розчарування. Терпіння – це нелегка для формування навичка, тому професіонали галузі цінують кваліфікованих розробників та тестувальників, які розвили її. Зазвичай найкращі рішення люди приймають у виважено нейтральному психологічному стані. Спокійна доброзичлива поведінка сприяє оптимальній розробці програмного забезпечення. Терпіння також може знадобитися і для взаємостосунків на робочому місці, наприклад, при навчанні нового персоналу або при очікуванні нового обладнання від постачальників. Формування поведінкової компетентності залежить від здатності виявляти співчуття. Саме такі розробники при виявленні проблеми, спокійно розв'язують її та рухаються далі.

8) Адаптивність означає здатність швидкої адаптації до нових обставин. Для розробників програмного забезпечення потреби користувачів і технології стрімко змінюються з часом, вимагаючи нових підходів у дизайні. Щодня розробляються та застосовуються нові інструменти та засоби розробки. Розробник має бути в курсі останніх тенденцій, які потенційно можуть оптимізувати кінцевий програмний продукт. Професіонали, здатні швидко адаптуватися, можуть змінювати свої підходи до розробки та покращувати продуктивність.

Пристосовуваність породжує самостійність. Але насправді більшість людей бояться пробувати щось нове, навіть якщо воно може бути корисним. Але найбільш кваліфіковані розробники, попри вагання, прагнуть використовувати ті методології, засоби та підходи, які покращать процес розробки й сам програмний продукт [8]. Для цього потрібні високий рівень самосвідомості, постійне навчання у своїй повсякденній роботі та відсутність страху змін. Також важливо напрацювати впевненість, щоб не зламатися у разі конструктивної критики. Саме такі енергійні ентузіасти, які впевнені у тому, що добре знають, доброзичливо і з повагою до інших висловлюють свою думку, мають найбільші шанси на професійний розвиток, оскільки вони не дозволяють своїй невпевненості стримувати їх.

9) Креативність. Коли професіонали створюють програмне забезпечення, вони використовують креативні стратегії для розробки технологій, які приваблюють користувачів і задовольняють клієнтів. Креативність також може допомогти технології виділитися серед конкурентів, тому розробникам програмного забезпечення важливо думати про нові підходи до процесу розробки.

Креативний програміст – велика рідкість і удача. Сама креативність не є обов'язковою компетентністю для ІТ, але дуже цінною якістю фахівця. Професіонали можуть поєднувати креативність і навички розв'язання проблем. Наприклад, у разі технічної помилки, з якою фахівець ніколи раніше не стикався, застосування креативного підходу до усунення несправності дозволить не лише дослідити проблему та звернутися по допомогу до колег, а й визначити спосіб виправлення помилки без негативного впливу на загальний прогрес розробки. Саме креатив іноді допомагає вивести проєкт на новий рівень або просто знайти коротке та витончене рішення рутинного завдання. Відомі численні приклади, коли ІТ-фахівці успішно переймали ідею з інших індустрій, грали на психології та людських почуттях, виявляли емпатію [7]. Розробники програмного забезпечення можуть використовувати поєднання креативності та здатності до адаптації, щоб залишатися продуктивними в несподіваній ситуації. Наприклад, компанія повторно складає бюджет для проєкту, що зменшує доступні кошти на обладнання. Розробники можуть творчо шукати економічно ефективне та якісне обладнання для використання у своїх проєктах, пристосовуючись до змін у бюджеті.



Ворогами креативності є мультизадачність через відволікання на різні завдання, розв'язання всіх завдань звичними способами страх помилок, песимізм і перфекціонізм. Але існують численні методики із доведеною ефективністю для розвитку креативності.

10) Лідерство – здатність вирішувати конфліктні ситуації у колективі, брати відповідальність на себе і приймати колективні рішення, делегувати завдання членам команди, пояснювати, навчати, але не повчати нових членів команди. Лідерство є важливою навичкою для розробників програмного забезпечення, які хочуть стати менеджерами або очолити команду над проектом. Проте ця компетентність важлива і для працівників не на керівних посадах, наприклад, у разі потреби допомоги у розв'язанні проблеми, з якою хтось не впорався. Щоб мати гарні лідерські навички роботи з людьми, потрібно поєднати всі soft skills, згадані вище. Але крім цього потрібен ще один важливий елемент – ставлення до людей як до особистостей. Це більше, ніж запам'ятовування їхніх імен. Йдеться про розуміння того, як кожна людина реагує на різні тригери та подразники, а також про те, як отримати найкраще від усіх членів команди. Розуміння людей у робочому колективі допомагає покращити залученість співробітників. Дослідження показують [9], що висока залученість співробітників може підвищити продуктивність і прибутковість більш ніж на 20% і зменшити кількість прогулів більш ніж на 40%. Для лідерства важливим є емоційний інтелект. Саме емоційно розумний лідер може бути впливовим лідером. Завдяки емоційному інтелекту формується позитивне професійне ставлення до своєї команди, попри будь-які ситуативні труднощі.

Важливими як для керівників, так і для всіх розробників програмного забезпечення є наявність робочої етики, здатності працювати з кількома завданнями одночасно, мотивація, наполегливість, планування, відповідальність, організованість, дотримання правил компанії, відсутність запізнь і прихвильності на роботу, і при здачі виконаних завдань. Не менш важливим є позитивний настрій, який проявляється у таких поведінкових навичках як впевненість, енергійність, ентузіазм, терплячість, доброзичливість, повага до інших.

Для зміцнення своїх soft skills розробника програмного забезпечення варто спочатку визначити сфери, які потрібно покращити, та скористатися відповідними методиками. Наприклад, для покращення управління своїм часом варто потренуватися записувати свої робочі обов'язки в планувальник і відстежувати, скільки часу було витрачено на кожне завдання. Знання конкретних здібностей, які потрібно розвинути, будь-то навичок публічних виступів або активного слухання, дозволить зосередити зусилля у потрібних напрямках. Для зміцнення навичок soft skills важливою є активна практика. Відпрацьовуючи власні здібності, корисно поміркувати про свою продуктивність при досягненні своїх цілей.

Розвиток soft skills. Розвивати власні навички комунікації важливо насамперед для самого працівника. Для цього є різноманітні психологічні методики активного слухання, вирішення та недопущення конфліктних ситуацій, розвитку креативності, ораторського мистецтва. Не завадить розвивати мову свого тіла та вчитися розпізнавати мову тіла співрозмовника. Для ефективного спілкування з колегами варто бути уважним та доброзичливим, поважати себе та оточення, навчитися планувати час, проекти та життя, працювати спокійно і розмірено, вміти слухати усміхаючись і не перебивати, навчитися отримувати насолоду від невимушених розмов (small talk), освоїти вміння жартувати, навчитися звертати увагу на приємне, корисне, гарне і, що не маловажно, говорити про це. Адже той, хто звик скаржитись, звинувачувати та засуджувати інших, грубити, принижувати та висміювати оточення або виявляти завуальований пасивно-агресивний



негативізм, змішувати робоче й особистісне, не вміє знаходити спільну мову з колегами й керівництвом, не зможе прижитися у жодному колективі та подолати невидиму стелю soft skills.

Переважно компанії зацікавлені у розвитку соціальних компетентностей своїх працівників, визнають необхідність навчати нових співробітників відповідним поведінковим навичкам та організують регулярні зустрічі soft skills meetup. Проте за умов дефіциту кадрів компанії просто не можуть собі дозволити витратити багато грошей і часу на перевиховання однієї людини, роблячи все можливе, щоб розробник прижився у колективі. Робота – це значна частина життя людини, на ній має бути комфортно. Ніхто не ходитиме на роботу, на якій йому некомфортно. Тому у компаніях немає культу терпимості та, як показує досвід, простіше звільнити одну «погану вівцю», ніж заново збирати все «стадо».

Коли страждає продуктивність, керівництво компанії має визнати проблеми та дослідити способи їх усунення. Але прогалини в соціальних навичках не однакові. Досить не просто розвинути нові soft skills, як-от активне слухання чи то лідерство.

Ключові проблеми при формуванні soft skills для ІТ-фахівців і способи їх подолання.

1. Соціальні компетентності важко виміряти. Перевірити набуття hard skills для ІТ-студента чи то працівника доволі легко. Для цього варто лише дати відповідне завдання, яке стосується розробки того чи іншого програмного забезпечення, і перевірити його виконання. Для оцінювання цих компетентностей можна використовувати кількісні показники. Що ж стосується оцінювання рівня soft skills, то тут все складніше і неоднозначніше. Складно однозначно визначити рівень таких поведінкових компетентностей, як управління часом чи то креативність через специфічні нюанси й контекст [12].

У професійному світі існують специфічні системи оцінювання soft skills рекрутів і працівників. Переважно такі моделі оцінки містять біля десятка компетентностей, кожна з яких характеризують за рівнями. Наприклад, рівень клієнтоорієнтованості працівника може бути від максимально позитивного для спеціаліста, який з'ясовує очікування клієнта, керує ними, діє на випередження і навіть може їх прогнозувати, до негативного, якщо людина демонструє протилежну поведінку до тої, що від неї очікують, формує у клієнта оманливі сподівання у тому числі обіцяє виконання задач, які апріорі неможливо виконати в заявлені клієнтом час, бюджет та обсяг, вказує клієнтові на його помилки, приховує від нього важливу інформацію або навіть займає змагальну позицію щодо клієнта [13].

Хоча модель є єдиною для всіх посад у тій чи іншій компанії, проте вимоги до рівня тих чи інших компетентностей можуть дещо різнитися для різних позицій (керівник проєкту, аналітик, менеджер проєкту, розробник, тестувальник тощо). Так, компетентність «клієнтоорієнтованість» набагато важливіша для фахівця, який безпосередньо спілкується із замовником, аніж для члена команди на позиції розробника, наприклад, Middle Software Developer, який не має безпосередніх стосунків із клієнтами. І навпаки, для розробників критично важливою компетентністю є «командна робота», тобто вміння працювати у команді, вміти лояльно доносити свої ідеї іншим і дослухатися до думки колег, бути готовим шукати спільні рішення.

Для подолання цієї проблеми доцільно розробляти тренінги для коректного й успішного формування soft skills, зосередитись на перевірці знань на основі сценаріїв, рольових ігор та інших контекстних, якісних вимірюваннях навчання. Наприклад, можна



надати менеджерам контрольний список спостережуваних форм поведінки, на які слід звернути увагу співробітників.

2. Тренування *soft skills* вимагає довгострокову програму навчання. На відміну від набуття *hard skills*, тренування соціальних компетентностей потребує значних зусиль щодо тривалої практики формування поведінкових навичок при різних обставинах спілкування і роботи у колективі розробників програмного забезпечення.

Для досягнення стійких змін у поведінці людини потрібен час і багаторазове закріплення для формування відповідної звички. Наприклад, лідерські навички не можливо сформувавши в людині, якщо один раз пояснити йому про їхню важливість і доцільність. Справжнє формування та розвиток такої компетентності відбудеться лише після багаторазової практики, зворотного зв'язку від інших і тривалих роздумів самої людини про її місце в соціумі.

Тому для подолання цього роду проблем варто розробити довгострокову програму навчання і не очікувати, що зміни відбудуться миттєво. Наприклад, навчання протягом року підкріпити та розширити можливими стимулами для продовження участі.

3. Різні соціальні навички формуються у різних людей не однаково просто і швидко. Залежно від психологічної особистості, для деякого у колективі розвиток навичок спілкування може здатися майже неможливим, але це не так. Навіть сором'язливі люди можуть стати потужними ораторами. Але цей процес не є простим і швидким, більш того він потребує індивідуального підходу [10]. Ключовим для розвитку навичок спілкування, які не є природною частиною особистості людини, є мотивація до змін.

Для розв'язання цієї проблеми варто у програми навчання *soft skills* включати ключові мотиваційні чинники до змін, які чітко даватимуть відповідь на запитання особистості «навіщо це мені?». Якщо людина зрозуміє, що, розвиваючи *soft skill*, вона зможе досягти свої особисті цілі, вона захоче і зможе навчитися будь-чого.

4. Складність прийняття людьми рішення щодо змін, навчання та розвитку своїх *soft skills*. Для розуміння специфіки цієї проблеми можна навести аналогію: можна навчити когось ловити рибу, але не можливо навчити отримувати задоволення від риболовлі. Складні поведінкові навички сильно залежать від особистісних рис і звичок людей. Розвиток навичок налагодження стосунків вимагає розуміння себе, розуміння потреби в змінах, бажання змінюватися та розвитку нових звичок, які зміцнюють зміни.

Для подолання такого роду труднощів варто розуміти й не поспішати з навчанням *soft skills*. Люди мають самі прийняти рішення щодо змін, навчання та розвитку своїх нових звичок. Спочатку їм це може не подобатися, але як тільки вони спробують і попрактикуються, нова навичка приживеться.

5. Відчуття неавтентичності та фальшивості. Навички спілкування глибоко вкорінені в особистості людей і пов'язані з їхніми звичками та життєвим досвідом. Без розуміння себе та того, як досвід впливає на звички та навички спілкування, навчання може змусити учасників почуватися неавтентичними та фальшивими.

Для подолання таких складнощів найкращий спосіб – це бути частиною команди. Робота з керівниками чи іншими експертами, які можуть моделювати навички, допомагає учасникам побачити, як виглядає бажане в дії. Це також пропонує можливості для практики та отримання зворотного зв'язку. Доцільно запропонувати методики, де люди можуть навчатися через спостереження та практику протягом певного періоду, отримувати періодичний зворотний зв'язок і розмірковувати про своє навчання та саморозвиток.



Отже, для підвищення здатності IT-компанії опановувати навички спілкування, варто пам'ятати про важливість мотивації працівників шляхом моделювання, практики, зворотного зв'язку та рефлексії в діяльності, створення способів вимірювання успіху і посилення навчання з часом.

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

На відміну від *hard skills*, *soft skills* є спільною мовою емпатії, співпраці та добробуту людей на робочому місці. Їх варто розглядати цілісно, адже вони можуть змінюватися, відповідно до змін у житті людини. Вони формують стійкість і водночас гнучкість команди, що врешті решт впливає на успішність компанії. Врахування важливості та розвитку соціальних компетентностей працівників сприяє їх спільному залученню до роботи, ефективній взаємодії та зацікавленості у досягненні гарних результатів, високій якості й продуктивності для довгострокової стійкості компанії у глобальному професійному середовищі.

Комунікація, робота в команді, управління часом, визнання помилок, емпатія, емоційний інтелект, терпіння, адаптивність, креативність, лідерство, розв'язання проблем і навички роботи з людьми є основними навичками *soft skills*, які потрібно розвинути розробнику програмного забезпечення, щоб значно покращити продуктивність, успішно розв'язувати свої професійні завдання в команді й загальне задоволення від своєї роботи.

Розуміння важливості *soft skills* для IT-фахівців важливе не лише для самих потенційних працівників, а й для вищих навчальних закладів, які формують необхідний базис компетентностей та програмних результатів навчання. Адже, серед інших факторів, навички спілкування мають великий вплив на результативність. Тому важливо для всіх освітніх компонентів без винятку вивчати та використовувати сучасні ефективні методики формування та розвитку соціальних компетентностей злагодженої роботи в колективі, адже навчальні заклади відіграють вирішальну роль у розвитку компетентних випускників, які повною мірою володіють необхідними *hard* та *soft skills*.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- 1 González-Morales, D., Moreno de Antonio, L. M., Roda García, J. L. (2011). Teaching “soft” skills in Software Engineering. *IEEE Global Engineering Education Conference (EDUCON-2011)*, 630-637. <https://doi.org/10.1109/EDUCON.2011.5773204>.
- 2 Сидорова, М. Г., Байбуз, О. Г., Лапець, О. В. (2021). Аналіз необхідних комунікативних навичок при розробці програмного забезпечення. *Актуальні проблеми автоматизації та інформаційних технологій*, 25, 152-157. <https://doi.org/10.15421/432115>.
- 3 Matturo, G., Raschetti, F., Fontán, C. (2015). Soft Skills in Software Development Teams: A Survey of the Points of View of Team Leaders and Team Members. *2015 IEEE/ACM 8th International Workshop on Cooperative and Human Aspects of Software Engineering*, 101-104. <https://doi.org/10.1109/CHASE.2015.30>.
- 4 Voitenko, S., Gadasina, L., Sørensen, L. (2018). The need for soft skills for ph.d.'s in software engineering. *CEUR Workshop Proceedings*, 2256. http://ceur-ws.org/Vol-2256/SWEPHD18_paper_03.pdf.
- 5 Putri, D. (2021). Correlation of Soft Skills and Hard Skills to Learning Outcomes of Mathematics Students. *Tarbawi : Jurnal Ilmu Pendidikan*, 17, 83-93. <https://doi.org/10.32939/tarbawi.v17i1.986>.
- 6 Mulyadi, M., Wellty, M., Dwi, N., Renova, M. (2022). The Effect of Organizational Learning on Improving Hard Skills, Soft Skills, and Innovation on Performance. *Jurnal Prajaiswara*, 3(2), 126-146. <https://doi.org/10.55351/prajaiswara.v3i2.51>



- 7 11 Important Soft Skills for Software Developers. <https://www.indeed.com/career-advice/career-development/software-developer-soft-skills>
- 8 Cordenne, B. 10 Soft Skills Every Software Engineer Should Have. <https://www.trio.dev/blog/software-engineer-soft-skills>
- 9 7 Soft Skills That Make a Great Software Developer. <https://blog.filestack.com/thoughts-and-knowledge/7-soft-skills-great-software-developers/>
- 10 15 White M. J. Software Engineer Soft Skills To Advance Your Career. <https://www.springboard.com/blog/software-engineering/5-soft-skills-every-software-engineer-needs/>
- 11 Зінченко, Т. Про Soft Skills в окремо взятій європейській країні. <https://dou.ua/lenta/articles/soft-skills-eu/>.
- 12 Why Are Soft Skills So Hard? <https://trainingindustry.com/articles/leadership/why-are-soft-skills-so-hard/>
- 13 Грицак, Н. Навіщо ІТ-спеціалістам оцінювання soft skills і як це робити. <https://dou.ua/lenta/articles/assessing-soft-skills-in-it/>

**Trofymenko Olena,**

PhD, Associate Professor, Associate Professor at the Department of Information Technologies of the National University "Odessa Law Academy", Odessa, Ukraine,
ORCID ID: 0000-0001-7626-0886
trofymenko@onua.edu.ua

Savielieva Oksana,

Doctor of Technical Sciences, Professor, Professor of the Department of Integrated Management Technologies of the Odessa Polytechnic National University, Odessa, Ukraine,
ORCID ID: 0000-0002-0453-4777
saveleva@op.edu.ua

Prokop Yuliia,

PhD, Associate Professor, Associate Professor at the Department of Software engineering of the Odessa Polytechnic National University, Odessa, Ukraine,
ORCID ID: 0000-0002-6608-3668
prokop.y.v@op.edu.ua

Loginova Nataliia,

PhD, Associate Professor, Head of the Department of Information Technologies of the National University "Odessa Law Academy", Odessa, Ukraine,
ORCID ID: 0000-0002-9475-6188
loginova@onua.edu.ua

Dyka Anastasiia,

Assistant of Department of Information Technologies of the National University "Odessa Academy of Law", Odessa, Ukraine,
ORCID ID: 0000-0002-4196-8734
dyka.anastasiia@gmail.com

SOFT SKILLS FOR SOFTWARE DEVELOPERS

Abstract. Nowadays, an understanding of the importance of having behavioral competencies for IT specialists has been formed. Analysis of publications confirms the growing needs and interest of software developers in the development of not only technical skills, but also social (behavioral) competencies for effective and efficient performance of professional activities. During interviews with recruits, IT companies try to find out not only the level of hard skills, but also the presence of a set of soft skills. Unlike hard skills, soft skills are the common language of empathy, cooperation, and well-being of people in the workplace. They should be considered holistically because they can change in accordance with changes in a person's life. They form the stability and at the same time flexibility of the team, which ultimately affects the success of the company. Each specific company itself defines its soft skills model as a set of competencies necessary for successful work in it to achieve its strategic goals. Currently, there is no well-defined, established, generally accepted list of soft skills for software developers. In the work, an analysis of numerous publications from the IT company was carried out and a list of soft skills, most important for software developers, considering the specifics of communication skills of the relevant industry, was formed. Communication, teamwork, time management, admitting mistakes, empathy, emotional intelligence, patience, adaptability, creativity, leadership, problem solving and interpersonal skills are the main soft skills that a software developer needs to develop to successfully solve his professional tasks in a team, significantly improve their productivity and increase their overall job satisfaction. The article also examines key difficulties in the formation of relevant soft skills and ways to overcome them. Considering the importance and development of employees' social competencies contributes to their joint involvement in work, effective interaction and interest in achieving good results, high quality and productivity for the long-term sustainability of the company in a global professional environment.

Keywords: communication; software developers; social competence; soft skills; hard skills; teamwork.

**REFERENCES (TRANSLATED AND TRANSLITERATED)**

- 1 González-Morales, D., Moreno de Antonio, L. M., Roda García, J. L. (2011). Teaching “soft” skills in Software Engineering. *IEEE Global Engineering Education Conference (EDUCON-2011)*, 630-637. <https://doi.org/10.1109/EDUCON.2011.5773204>.
- 2 Sydorova, M. G., Baibuz, O. G., Lapets, O. V. (2021). Analysis of necessary communication skills in software development. *Actual problems of automation and information technologies*, 25, 152-157. <https://doi.org/10.15421/432115>.
- 3 Matturro, G., Raschetti, F., Fontán, C. (2015). Soft Skills in Software Development Teams: A Survey of the Points of View of Team Leaders and Team Members. *2015 IEEE/ACM 8th International Workshop on Cooperative and Human Aspects of Software Engineering*, 101-104. <https://doi.org/10.1109/CHASE.2015.30>.
- 4 Voitenko, S., Gadasina, L., Sørensen, L. (2018). The need for soft skills for ph.d.'s in software engineering. *CEUR Workshop Proceedings*, 2256. http://ceur-ws.org/Vol-2256/SWEPHD18_paper_03.pdf.
- 5 Putri, D. (2021). Correlation of Soft Skills and Hard Skills to Learning Outcomes of Mathematics Students. *Tarbawi : Jurnal Ilmu Pendidikan*, 17, 83-93, <https://doi.org/10.32939/tarbawi.v17i1.986>.
- 6 Mulyadi, M., Wellty, M., Dwi, N., Renova, M. (2022). The Effect of Organizational Learning on Improving Hard Skills, Soft Skills, and Innovation on Performance. *Jurnal Prajaiswara*, 3(2), 126-146. <https://doi.org/10.55351/prajaiswara.v3i2.51>
- 7 11 Important Soft Skills for Software Developers. <https://www.indeed.com/career-advice/career-development/software-developer-soft-skills>
- 8 Cordenne, B. 10 Soft Skills Every Software Engineer Should Have. <https://www.trio.dev/blog/software-engineer-soft-skills>
- 9 7 Soft Skills That Make a Great Software Developer. <https://blog.filestack.com/thoughts-and-knowledge/7-soft-skills-great-software-developers/>
- 10 5 White M. J. Software Engineer Soft Skills To Advance Your Career. <https://www.springboard.com/blog/software-engineering/5-soft-skills-every-software-engineer-needs/>
- 11 Zinchenko, T. About Soft Skills in a single European country. <https://dou.ua/lenta/articles/soft-skills-eu/>.
- 12 Why Are Soft Skills So Hard? <https://trainingindustry.com/articles/leadership/why-are-soft-skills-so-hard/>
- 13 Hrytsak, N. Why should IT specialists evaluate soft skills and how to do it. <https://dou.ua/lenta/articles/assessing-soft-skills-in-it/>.

