

Царьов В.М.

доцент кафедри бізнес-економіки та підприємництва
Київський національний економічний університет
імені Вадима Гетьмана

м. Київ, Україна

e-mail: vitaliytsarov2015@gmail.com

ORCID: 0000-0003-1180-9449

ДЕЯКІ АСПЕКТИ МОДИФІКАЦІЇ ФРЕЙМВОРКІВ ПРІОРИТЕЗАЦІЇ ЗАВДАНЬ У ПРОДУКТОВОМУ ІТ

Tsarev Vitaliy

PhD in Department of Business Economics and Entrepreneurship

KNEU named after Vadym Hetman

Beresteyskiy avenue, 54/1, Kyiv, Ukraine

e-mail: vitaliytsarov2015@gmail.com

ORCID: 0000-0003-1180-9449

MODIFICATION OF TASKS PRIORITIZATION FRAMEWORKS IN PRODUCT IT

Анотація. У сучасній економіці продукти та послуги інформаційних технологій стали не тільки невід'ємною частиною будь-якого бізнесу, а й її основним продуктом. Особливості розробки інформаційних продуктів пов'язані у першу чергу із великим впливом такого фактору як інтелектуальний капітал та управління персоналом, а також із постійною зміною вимог користувачів та клієнтів до продуктів. Це все зумовлює те, що ІТ-професіонали стикаються з великою кількістю завдань, які потрібно виконати, що вимагає постійної пріоритезації робіт. У статті розглядаються різні підходи до пріоритезації завдань, які допомагають командам розробників та менеджерам ефективно управляти своїм завданнями та досягати поставлених цілей. Також робиться спроба проаналізувати різні інструменти та програмне забезпечення, які можуть допомогти в управлінні та пріоритезації завдань. Автором наголошується, що ефективна пріоритезація завдань є критично важливим навичком для продуктових та проєктних менеджерів. Для доведення цієї гіпотези автор спирається на результати досліджень компанії Planview та Wrike. Крім того, у дослідженні аналізувалися наукові роботи спеціалістів у сфері стартапів та відомих продуктових менеджерів, таких як Стівен Мейберг, Мет Лімей, Райан Сінґер та інші. Приділено посилену увагу та критично розглядається широковживана в практиках продуктового менеджменту методика пріоритезації RICE. Зазначено, що основним недоліком застосування фреймворку (методики) RICE є суб'єктивізація алгоритму його розрахунку, а також нехтування факторами, які можуть суттєво вплинути на його значення. У статті пропонується новий підхід до пріоритезації завдань на основі RICE, який враховує додатковий фактор — компетенцію команди розробників. Для вирішення цього завдання автором пропонується ввести коефіцієнт, який враховує якість команди розробників. Також наводяться приклади бальної оцінки якості команди залежно від складу та досвіду її учасників. Крім того, у статті робиться спроба прикладного застосування модифікованої методики RICE для пріоритезації завдань у продуктовому ІТ. Висвітлюються особливості визначення кожної складової, врахування фактора цінності та якості команди розробників. У підсумку стаття доводить, що модифікована методика RICE з додаванням фактора якості персоналу є ефективним інструментом для пріоритезації завдань у продуктовому ІТ. Це дозволяє враховувати більше факторів та забезпечує більш точну оцінку впливу завдань на продукт та організацію в цілому.

Ключові слова: пріоритезація у IT, фреймворк пріоритезації, фреймворк RICE, управління проєктами, продуктовий менеджмент, управління командою.

Abstract. Information technology products and services have become not only an integral part of any business, but also its main product in modern economy. The peculiarities of the development of information products are primarily associated with the great influence of such factors as intellectual capital and personnel management, as well as with the constant change in user requirements. All this leads to the fact that IT professionals are faced with a large number of tasks that need to be performed, which requires constant prioritization of work. The article discusses various approaches to prioritizing tasks that help teams of developers and managers effectively manage their tasks and achieve their goals. An attempt is also made to analyze various tools and software that can help in task management and prioritization. The author emphasizes that effective task prioritization is a critical skill for product and project managers. To prove this hypothesis, the author relies on the results of research by Planview and Wrike companies. In addition, the research analyzed the scientific works of specialists in the field of startups and well-known product managers, such as: Steven Mayberg, Matt Limey, Ryan Singer, and others. In particular, the author pays more attention and critically examines the RICE prioritization framework, which is widely used in product management practices. It is noted that the main disadvantage of using the RICE framework (method) is the subjectivization of its calculation algorithm, as well as the neglect of factors that can significantly affect its value. The article proposes a new approach to task prioritization based on RICE, which takes into account an additional factor — the competence of the development team. To solve this problem, the author suggests introducing a score that takes into account the quality of the development team. Also, there are examples of point-based assessment of team quality depending on the composition and experience of its members. In addition, the article attempts to apply the modified RICE methodology for prioritizing tasks in product IT. The specifics of defining each component, taking into account the value factor and the quality of the development team are highlighted. In conclusion, the article proves that the modified RICE technique with the addition of the staff quality factor is an effective tool for prioritizing tasks in product IT. It allows taking into account more factors and provides a more accurate assessment of the impact of tasks on the product and the organization as a whole

Key words: prioritization in IT, prioritization framework, RICE framework, project management, product management, team management.

JEL codes: M54, L23, L25.

Постановка проблеми. Продуктові IT-компанії працюють у ринковому середовищі, яке характеризується швидкою зміною технологій та постійним сталим зростанням попиту на цифрові продукти та послуги. Це можуть бути як стартапи, малі та середні компанії, або великі технологічні корпорації. У такому середовищі користувачі стають все вимогливішими та швидше адаптуються до нових технологій та продуктів.

Саме тому продуктові IT-компанії повинні бути гнучкими, швидко реагувати на зміни в ринкових умовах та потребах користувачів. Компанії постійно оновлюють свої продукти та послуги, додають нові функціональні можливості та забезпечують якість своїх продуктів та сервісів. Для успішної роботи у ринковому середовищі продуктові IT-компанії також повинні мати досвідчену та компетентну команду, яка зможе впроваджувати нові ідеї та інновації у свої продукти та процеси. Висока конкуренція за кваліфіковані ресурси, та зростання вимог користувачів (клієнтів) вимагає від продуктових IT-компаній постійного пошуку нових функцій продукту. Через це, в умовах обмежених ресурсів

та часу (особливо на стадіях імплементації та розвитку продуктів), більшість стартапів стикається з проблемою пріоритезації виконуваних робіт.

Аналіз досліджень і публікацій. Дослідженням питання пріоритезації у ІТ займалися багато вчених, підприємців та експертів у царині продуктового менеджменту та проєктного управління. Серед них: спеціаліст з продуктового менеджменту Стівен Мейберг (Stephen J. Mayeux), Мет Лімей, продуктовий менеджер у компанії Basecamp Райан Сінгер (Ryan Singer), автор методики GTD Девід Аллен (David Allen), Ерік Піс (Eric Ries) та ін.

Зазначені, а також інші автори у своїх роботах висвітлюють широкий спектр тем, пов'язаних з продуктовим менеджментом, включаючи планування та пріоритезацію продуктів, аналіз ринку та конкурентів, комунікації із зацікавленими сторонами, тощо.

ІТ-продукти можуть зазнавати змін на будь-якій стадії життєвого циклу, залежно від різних факторів, таких як потреби користувачів, нові технології, ринкові умови, зміни в законодавстві, бізнес-стратегії компанії та інші. Проте, найбільше змін в ІТ-продуктах відбувається на початковій стадії життєвого циклу, тобто під час розробки та випробувань продукту. На цій стадії зазвичай виявляються помилки, недоліки, усунення яких потребує внесення змін у продукт. Окрім того, на цій стадії можуть з'являтися нові ідеї та можливості для покращення продукту.

Також, зміни в ІТ-продуктах можуть відбуватися на стадії експлуатації, коли користувачі надсилають зворотній зв'язок та пропозиції щодо поліпшення продукту. Успішні продуктові компанії постійно вдосконалюють свої продукти та сервіси, враховуючи потреби своїх користувачів та зміни на ринку, тому зміни в ІТ-продуктах є постійним процесом.

Методика досліджень. Деякі дослідження показують, що зміни в ІТ-продуктах можуть бути досить значними. Наприклад, згідно з дослідженням, проведеним компанією Planview, більшість продуктів зазнають значних змін протягом року. У середньому 41 % змін відбувається на етапі планування, 34 % — на етапі розробки та 25 % — на етапі тестування та випуску продукту. Разом з тим 70 % організацій повідомляють, що їхня найбільша складність у впровадженні управління продуктами полягає у визначенні пріоритетів та розподілі ресурсів [1].

Дослідження компанії Wrike показало, що менше ніж половина (44 %) компаній забезпечують продуктами від 80 до 100 % очікуваних функцій. Це означає, що більшість продуктів потребують покращень та змін, для того щоб задовольнити потреби користувачів та відповідати ринковим вимогам.

Разом з тим згідно із зазначеним дослідженням компанії Planview 70 % організацій повідомляють, що їхня найбільша складність у впровадженні управління продуктами полягає у визначенні пріоритетів та розподілі ресурсів.

Виділення невирішених раніше частин загальної проблеми. Швидкий розвиток продуктового ІТ, а особливо практик управління продуктам та командами зумовив появу великої кількості методик та фреймворків щодо управління завданнями — зокрема пріоритезацією. Для цього були застосовані та адаптовані методики, які розроблялися для традиційних (нецифрових продуктів). Разом з тим проблема полягає у особливостях розробки саме інформаційних продуктів та середовищі функціонування продуктової компанії, які, на нашу думку, не є повністю враховані у існуючих фреймворках пріоритезації.

Метою статі є визначення шляхів та алгоритмів подальшого розвитку та методології фреймворків пріоритезації завдань, які б дозволили обирати найефективніші варіанти необхідних продуктових функцій в умовах обмежених ресурсів та швидких змін всередині та зовні компанії

Викладення основного матеріалу. Все зазначене означає, що пріоритезація є дуже важливою у продуктовому менеджменті, оскільки дозволяє команді продукту зосередитися на найважливіших завданнях та досягнути мети продукту. Актуальність пріоритезації полягає у таких аспектах:

1. оптимізація ресурсів: пріоритезація допомагає використовувати ресурси (час, людські ресурси, бюджет) найефективніше, спрямовуючи їх на завдання, які мають найбільший вплив на мету продукту;

2. забезпечення успішності продукту: пріоритезація дає змогу зосередитися на завданнях, які допоможуть досягти мети продукту та забезпечити його успіх на ринку. Відправним пунктом для пріоритезації завдань є стратегія продукту та потреби користувачів;

3. адаптація до змін: завдяки пріоритезації продуктова команда швидко адаптується до змін у потребах користувачів, ринкових умов та технологіях;

4. ефективна комунікація: команда продукту чітко розуміє, які завдання мають високий пріоритет. Це також допомагає уникнути розбіжностей у пріоритетах серед стейкхолдерів;

5. контроль якості: пріоритезація забезпечує контроль якості продукту, оскільки завдання з високим пріоритетом є найважливішими для досягнення мети продукту.

Разом з тим пріоритезація завдань у продуктовому IT ускладнюється через різні потреби інтересів, обмеженість ресурсів, ринкову непередбачуваність та недостатність інформації, недостатнє врахування стратегії. Різні стейкхолдери можуть мати різні потреби та інтереси, які не завжди легко поєднати. Це може призводити до конфліктів та ускладнювати процес пріоритезації завдань.

Обмеженість ресурсів, таких як час, людські ресурси та бюджет, може призвести до виникнення проблем у визначенні того, яким завданням потрібно приділити першочергову увагу. Ринкова непередбачуваність, зміна побажань користувачів або технологій також можуть внести зміни в пріоритети завдань, тому важливо бути готовим до змін та швидко адаптуватись, а недостатня інформація про користувачів, ринок або технології може призвести до неправильної оцінки пріоритету завдань та невдалих рішень. Крім того, пріоритезація завдань повинна ґрунтуватися на стратегії компанії та місії продукту. Якщо стратегія нечітко визначена або не врахована, то це може вплинути на правильність пріоритетів.

Відбір найпріоритетніших функцій продукту зазвичай здійснюється на основі різноманітних методик, методологій та підходів (фреймворків). Такі підходи та методики в основному базуються на якісних та кількісних показниках, які включають рівень очікування щодо впливу функції продукту на споживачів, різного типу ризику, обсяг необхідних робіт тощо.

Так, наприклад, фреймворк Risky Bet — це метод, запропонований Раяном Сінгером та компанією Basecamp у рамках методології Shape Up, який допомагає оцінити ризики і визначити, наскільки великий ризик несе компанія відносно нового продукту або функції, яку вони збираються розробити. Фреймворк Risky Bet базується на двох параметрах: значущості продукту або функції, не-

визначеності. Значущість продукту або функції оцінюється тим, наскільки важливою є функція для компанії. Вона може бути визначена через прямий вплив на бізнес, потребу користувачів або стратегічні цілі компанії. Під невизначеністю розуміють рівень точності знань у команди про те, як потрібно розробляти продукт або функцію. Якщо команда знає точно, що потрібно розробити, то це значить, що невизначеність низька, а якщо команда не знає, як потрібно розробляти продукт або функцію, то це значить, що невизначеність висока.

На основі цих параметрів фреймворк Risky Bet розділяє проєкти на три категорії:

1. легкі ставки (Small Bets): проєкти з низькою значущістю та низькою невизначеністю, які можна запустити швидко та без значних витрат;
2. ставки середнього рівня (Bets): проєкти з високою значущістю та середньою невизначеністю, які потребують більш ретельної підготовки та уваги;
3. ризиковані ставки (Risky Bets): проєкти з високою значущістю та високою невизначеністю, які потребують багато ресурсів та мають значний ризик невдачі, але водночас можуть мати великий вплив на бізнес компанії.

Крім того, у продуктовому менеджменті є кілька інструментів для визначення пріоритетів, які можуть використовуватися залежно від потреб і особливостей компанії (табл. 1).

Таблиця 1

Інструмент пріоритезації рішень	Опис
Канбан-дошка	Дозволяє організувати завдання за пріоритетом і розташувати їх у порядку виконання. Кожне завдання представлена картою, яку можна переміщувати по колонках «В очікуванні», «В роботі» та «Готово»
SWOT-аналіз	За допомогою SWOT-аналізу можна визначити, які продукти і функції найкраще відповідають потребам ринку та конкурентним перевагам
Lean Startup	Дозволяє побудувати мінімальний життєздатний продукт (MVP) і отримати зворотний зв'язок від користувачів до його доведення до фінальної версії
Метод вагованих оцінок	Кожен критерій отримує вагу, яка відображає його важливість відносно інших критеріїв
Матриця Ейзенхауера	Класифікує завдання за двома критеріями — важливість та терміновість. Завдання можуть бути розташовані у чотирьох квадрантах: «Важливо і терміново», «Важливо, але не терміново», «Не важливо, але терміново» та «Не важливо і не терміново»
Метод MoSCoW	Використовується для класифікації завдань за чотирма категоріями: «Must have», «Should have», «Could have» та «Won't have». Інструмент допомагає визначити, які функції або функціональність продукту є найбільш важливими та повинні бути включені до першої черги
Value / Effort	Спосіб пріоритезації, який оцінює функції за критеріями вкладу кожної функції та зусиль на її реалізацію
RICE	Використовується для прийняття рішень щодо вибору функцій або функціональності продукту для розробки. Він оцінює функції за чотирма критеріями: Reach, Impact, Confidence та Effort
ICE	Інструмент для прийняття рішень у продуктовому менеджменті, що допомагає оцінити потенційний вплив (Impact), рівень впевненості в розробці функції (Confidence) та рівень легкості реалізації (Ease)

Найбільшої популярності у продуктових ІТ-компаніях набули методології RICE та ICE, які базуються на розрахунку комплексного показника із використанням схожих кількісних та якісних параметрів.

Методологія ICE (Impact, Confidence, Ease) — це інструмент, що покликаний оцінити потенційний вплив (Impact), рівень впевненості в розробці функції (Confidence) та рівень легкості реалізації (Ease). Метод визначення пріоритетів ICE був розроблений Шоном Еллісом та спочатку був призначений для пріоритизації експериментів зростання. Пізніше ICE почали використовувати і для пріоритизації функцій продукту (фіч).

Щоб здійснити відбір пріоритетних функцій за методологією ICE, для кожної функції оцінюються три параметри:

1. Impact — потенційний вплив на продукт або бізнес, який може мати функція. Оцінюється за шкалою від 1 до 10;
2. Confidence — рівень впевненості в тому, що розробка функції буде успішною. Оцінюється за шкалою від 1 до 10;
3. Ease — рівень легкості реалізації функції. Оцінюється за шкалою від 1 до 10

Після оцінки всіх функцій за кожним з параметрів, для кожної функції розраховується її бал (score), який обчислюється за формулою:

$$\text{ICE} = \text{Impact} \times \text{Confidence} \times \text{Ease}. \quad (1)$$

Чим вищим є показник ICE-, тим вищий пріоритет матиме функція (виконане завдання).

Параметр Impact показує, наскільки ідея позитивно вплине на ключовий показник, який команда намагається покращити. Confidence є оцінкою того, скільки зусиль та ресурсів потрібно для реалізації цієї ідеї. Ease — рівень впевненості в оцінках впливу та легкості реалізації.

Методологію ICE використовують у багатьох компаніях, особливо в сфері продуктового менеджменту та маркетингу, серед них Hubspot (маркетингова платформа, яка використовує ICE для прийняття рішень щодо розробки та вдосконалення своїх продуктів), Intercom (компанія, яка надає послуги з комунікації та клієнтської підтримки), LinkedIn (соціальна мережа для професійного спілкування, яка використовує ICE для визначення пріоритетів щодо функцій та розвитку продукту), Atlassian (розробляє програмне забезпечення для спільної роботи та управління проєктами), Buffer (соціальний медіаменеджер). Крім того дану методологію, або її елементи використовують і інші продуктові компанії.

Одним з головних недоліків методу ICE є його надмірна суб'єктивність, а саме:

- одна і та ж функція (фіча) може оцінюватися по-різному однією і тією ж особою у різний час. Це може вплинути на остаточний перелік пріоритетів;
- у разі якщо функції оцінюють різні люди — усі вони оцінюватимуть її по-різному;
- якщо члени команди бажають, щоб їх функції були пріоритетними, то можуть маніпулювати результатами, щоб отримати позитивний результат.

Одним із найпопулярніших підходів є використання методології RICE, яка включає розрахунок чотирьох показників: Reach, Impact, Confidence та Effort. Відмінність від методики ICE полягає у використанні кількісного показника Reach — потенційної кількості користувачів, яких буде залучено після впро-

вадження нової функції. Цей показник у методиці є єдиним, який збирається на основі дослідження кількісних даних. Impact — оцінка впливу нової функції на користувачів та продукт. Confidence — впевненість у вірогідності успішного запуску нової функції. Effort — оцінка зусиль, необхідних для впровадження нової функції. Серед відомих компаній, які використовують методику RICE у продуктовому менеджменті такі компанії, як Facebook, Dropbox, Atlassian, Intercom, Mixpanel, Uber, Zendesk та інші.

Результати розрахунку кожного з цих показників дають загальний бал RICE, який і визначає пріоритетність функцій.

Розрахунок загального балу RICE здійснюється за формулою:

$$\text{RICE} = \frac{\text{Reach} \times \text{Impact} \times \text{Confidence}}{\text{Effort}}. \quad (2)$$

Рівень охоплення (reach) вимірюється кількістю людей/подій за певний період. Цей фактор призначений для оцінки того, на яку кількість людей кожна функція (фіча) або проєкт вплине протягом певного періоду часу, і скільки користувачів (клієнтів) побачать такі зміни. При цьому важливою відмінністю та вимогою є використання реальних метрик, а не якісних показників (таких як експертна оцінка, бали тощо). Наприклад, якщо функцією продукту (подією) скористається 100 людей на місяць.

Вплив (Impact) показує, який внесок приносить обрана функція всьому продукту. Рівень впливу оцінюється за бальною системою: 3 — якщо функція матиме масовий вплив, 2 — для високого впливу, 1 — якщо функція має середній вплив, 0,5 — для низького та 0,25 для мінімального впливу. При цьому кожний продукт та/або продуктова компанія може мати свої розуміння «високого» чи «низького» рівня впливу. Цінність функції буде залежати не тільки від продукту, але й від цілей компанії. Найчастіше найбільшу цінність будуть мати функції, які збільшують обсяг продажу продуктів (конверсію), допомагають залучити нових користувачів, допомагають зберегти та повернути поточних користувачів, відрізняють продукт від продукту конкурентів.

Показник Confidence показує рівень впевненості команди в тому, що обрана функція буде мати вплив. За сутністю confidence є оберненим до ризикованості та вимірюється у відсотках.

Показник Effort (трудовитрати) оцінюються як кількість «людино-місяців», тижнів чи годин, залежно від потреб. У методиці RICE effort (зусилля) визначається на основі складності та розміру функції, а також на основі досвіду та експертної думки. Важливо зазначити, що оцінка зусиль завжди є приблизною, оскільки складність і об'єм роботи можуть змінюватися в процесі розробки продукту. Для розрахунку трудомісткості реалізації функції використовують такі підходи:

1. оцінка зусиль експертів. Підхід полягає у залученні експертів, які мають досвід у розробці програмного забезпечення, та оцінюють зусилля, необхідні для виконання функції;
2. метод Planning Poker. Метод полягає в залученні команди розробки для оцінки зусиль на основі розміру функції;
3. ретроспективний аналіз. Базується на використуванні даних про зусилля, необхідні для виконання схожих функцій, які вже були розроблені командою;

4. методологія точок функцій (Function Points). Метод використовується для вимірювання функціональної складності програмного забезпечення.

Методика RICE має декілька переваг порівняно з іншими методиками визначення пріоритетів функцій продукту:

- враховується вплив кожної функції на користувача і на бізнес, що дозволяє зосередитися на функціях, які найбільше покращують користувальницький досвід та принесуть максимальну користь бізнесу;
- методика дозволяє визначити масштаб і витрати для кожної функції, тому що вона оцінює не тільки вплив, але й розмір ринку і складність реалізації.
- методика може бути легко модифікована та адаптована до конкретних потреб продукту та бізнесу.
- визначення пріоритетів відбувається шляхом надання балів кожній функції, що дозволяє об'єктивно оцінити їх важливість;
- RICE є досить простою у використанні, тому її можуть використовувати не тільки продуктові менеджери, але й усі учасники проєкту;
- методика дозволяє швидко відрізнити відбір функцій продукту, виводячи на перший план лише ті функції, які можуть мати значний вплив на користувачів та бізнес.

Разом з тим, методика RICE містить певні недоліки, які потрібно враховувати в процесі пріоритезації завдань:

1. оцінка кожного фактора (reach, impact, confidence, effort) може бути досить суб'єктивною, що призводить до неточних результатів;
2. методика RICE вимагає точного вимірювання кожного фактора, що може виявитися складним у разі обмеження доступу до даних;
3. RICE не враховує конкурентів, що може призвести до того, що важливі функції будуть пропущені через недостатній аналіз конкурентного середовища.
4. RICE не враховує загальну стратегію компанії, що може призвести до того, що найважливіші функції для компанії будуть проігноровані;
5. відсутність оцінки ризиків. Методика RICE не враховує ризики, пов'язані з упровадженням нових функцій. Через це небезпечні (найризикованіші) функції будуть реалізовані, а безпечні — проігноровані.

Одним із напрямів модифікації методики RICE може бути її оптимізація у розрізі визначення показника Effort (зусилля). Хоча показник effort є кількісною величиною, він, як і будь-який показник трудовитрат, є комплексним явищем, і залежить від низки якісних факторів. Разом з тим компетентність команди (розробників) може вплинути на обсяг необхідних зусиль, оскільки залежно від рівня їхніх знань та досвіду може змінюватися складність реалізації конкретних функцій. Персонал з вищою компетентністю та кваліфікацією може бути ефективнішими та швидшими у вирішенні технічних проблем і розробці продукту. Це може зменшити трудовитрати (зусилля), необхідні для виконання функцій. З іншого боку, розробники з меншою компетентністю можуть потребувати додаткового часу та зусиль, щоб розібратися з технічними проблемами та виконати необхідні функції. Тому, рівень компетентності персоналу є важливим чинником, який необхідно враховувати при розрахунку зусиль (effort) у методиці RICE.

Для вирішення поставленого завдання врахування якісного складу команди можна запропонувати додавання до показника трудовитрат коефіцієнта якості

команди. Незалежно від алгоритму розрахунку такий коефіцієнт повинен зменшувати показник зусиль (effort) у разі залученні компетентнішого персоналу, та збільшувати його у разі залучення менш кваліфікованого. Такий коефіцієнт може бути розраховано як формально, враховуючи професійні рівні розробників, які приймають участь у команді (Junior, Middle, Senior, Lead, Manager / Director), а також із використання інструментів оцінки якості складу команди.

Оцінку якості складу команди можна проводити за допомогою технічної співбесіди з учасниками команди, ретроспективного аналізу досвіду успішних проєктів, які мають учасники команди у портфоліо, Agile retrospective, тобто щотижневому процесу огляду роботи, який допомагає команді оцінити свої успіхи та проблеми і знайти шляхи для подальшого покращення. Крім того, необхідно розуміти, що компетентність та кваліфікація команди розробників може змінюватися (зростати) в процесі виконання поставлених їм завдань, тож її необхідно перевіряти періодично.

За результатами такої оцінки пропонується присвоювати кожній функції продукту коефіцієнт якості команди (quality impact). Наприклад, матриця коефіцієнтів якості команди може мати такий вигляд (табл. 2).

Таблиця 2

Рівень команди	Структура команди	Коефіцієнт якості команди
Trainee team	стажери	0,75
Junior team	спеціалісти без досвіду впроваджених проєктів, загальний досвід роботи менше від 3 років	0,9
Middle team	Загальний досвід роботи 3–5 років, досвід роботи в проєктах	1
High quality team	Загальний досвід роботи понад 5 років, успішний досвід закінчених проєктів	1,1
Professional team	Загальний досвід роботи понад 10 років, успішний досвід закінчених проєктів, ментори, product-owner	1,25

Отже, розрахунок балу SCORE буде здійснюватися за формулою

$$RICE = \frac{Reach \times Impact \times Confidence}{Effort} \times QI, \quad (3)$$

де QI — показник якості складу команди quality impact.

Висновки. Рівень якості команди не тільки безпосередньо впливає на показник трудозатрат на реалізацію функції (зусиль). Залучення до робіт розробників із низькою кваліфікацією може збільшити транзакційні витрати. Це пов'язано з тим, що низька компетентність розробників призводить до більшої кількості помилок та відхилень від вимог проєкту, що призводить до збільшення часу, необхідного для виправлення цих помилок. Крім того, розробники з низькою компетентністю можуть потребувати детальнішого та ретельнішого контролю

роботи, що також збільшує транзакційні витрати. Крім того, якість персоналу впливає на реалізацію функції продукту, що визначається показником «вплив на користувачів» (іmpact).

Запропонована шкала якості команди має лише рекомендаційний характер та є суб'єктивною оцінкою того, як продуктова ІТ компанія оцінюватиме вплив якості персоналу на реалізацію функцій і проєктів у цілому.

Література

1. Planview PPM Pro Reviews, Ratings & Features 2023. <https://www.gartner.com/reviews/market/project-portfolio-management-worldwide/vendor/planview/product/planview-ppm-pro>
2. Matt LeMay Product Management in Practice: A Real-World Guide to the Key Connective Role of the 21st Century 1st Edición
3. Ryan Singer, Jason Fried. Shape Up: Stop Running in Circles and Ship Work that Matters
4. Marty Cagan Inspired: How to Create Tech Products Customers Love (Silicon Valley Product Group) 2nd Edición
5. Eric Ries The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses Tapa dura — 13 Septiembre 2011

References

6. Planview PPM Pro Reviews, Ratings & Features 2023. <https://www.gartner.com/reviews/market/project-portfolio-management-worldwide/vendor/planview/product/planview-ppm-pro>
7. Matt LeMay Product Management in Practice: A Real-World Guide to the Key Connective Role of the 21st Century 1st Edición
8. Ryan Singer, Jason Fried. Shape Up: Stop Running in Circles and Ship Work that Matters
9. Marty Cagan Inspired: How to Create Tech Products Customers Love (Silicon Valley Product Group) 2nd Edición
10. Eric Ries The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses Tapa dura — 13 Septiembre 2011