

УДК 004.738.5

ВИКОРИСТАННЯ МЕНЕДЖМЕНТУ ПРОЕКТІВ ІЗ ІТЕРАТИВНИМ ЦИКЛОМ РОЗРОБКИ

Микола ПАВЛЕНЧИК

магістрант

Науковий керівник: **Володимир ХОЛЯВКА**

канд. екон. наук, доцент кафедри

економіки та менеджменту

Львівський державний університети фізичної культури
імені Івана Боберського (Україна)

За досить короткий проміжок часу сфера інформаційних технологій в Україні зі звичайної малозначної галузі перетворилась на один з основних рушіїв української економіки. Цьому сприяло багато факторів, а зокрема пандемія COVID-19 та повномасштабна війна. Враховуючи всі ризики та обмеження, що зумовлені зазначеними подіями, люди активно почали шукати можливості заробітку в умовах ізоляції, повітряних тривог, відсутності електроенергії та інших обмежень [1-3]. Відтак, ІТ-індустрія збагатилася великою кількістю молодих спеціалістів із низьким рівнем досвіду. У той же час компанії, представлені на цьому ринку, зіштовхнулися із проблемою управління людськими ресурсами в командах. Менш досвідчені працівники повинні бути навчені та під ретельним наглядом. Тому цей контроль повинен здійснювати більш кваліфікований і досвідчений персонал. Все це призводить до збільшення термінів реалізації проекту, а отже, до підвищення витрат на розробку, що не завжди прийнятно з точки зору власника продукту. Більш того, часто на етапі активної розробки проекту власник продукту хоче якомога частіше бачити результати роботи. Тому досить поширеним є виконання вимог щодо демонстрації результатів розвитку принаймні щотижня.

У цих реаліях класичні методології менеджменту проектів, такі як Scrum, Kanban, Agile, Waterfall та інші не є достатньо ефективними й потребують різних модифікацій та адаптацій. Розглянемо детальніше найбільш

розповсюджені із них: Scrum та Kanban. Кожна із зазначених технологій майже ніколи не використовується для менеджменту проектів у оригінальному вигляді [4]. Для зазначених умов Scrum володіє наступними недоліками:

1. Орієнтований на відсутність у команді менеджера та залучення всієї команди до планування спринта й менеджменту завдань. Очевидно, що це достатньо неефективний підхід за умови великої кількості малодосвідчених працівників у команді.

2. Передбачає повне залучення команди та значну витрату часу на такі процеси, як планування спринта та ретроспектива. За умов значного обмеження часу та необхідності щотижневої демонстрації результатів просто неможливо забезпечити ефективну розробку витрачаючи значний час на ці процеси.

3. Не передбачено навчання персоналу. Кваліфікований персонал і ефективний менеджмент - ключ до успіху для будь-якого проекту. Якщо команда складається із значної кількості малодосвідчених працівників передача знань всередині команди грає важливу роль для успішного виконання завдань [5].

Кожна компанія повинна винайти власний підхід до управління проектами. Тому було б корисно розробити стандартизований підхід для вимог, описаних вище. Найкращим способом зробити це було б поєднати частини існуючих методологій, які найкраще відповідають описаним умовам, і модифікувати їх за необхідності.

Перш за все, необхідно вирішити проблеми з недосвідченими членами команди та комунікаційними процесами всередині команди. Очевидно, що найшвидший шлях підготовки молодих спеціалістів – це запровадження менторських програм. Обмін знаннями в команді призведе до підвищення продуктивності, якості та зниження витрат на розробку в довгостроковій перспективі. Крім того, це зменшить витрати на адаптацію нового співробітника у разі зміни колективу.

За цих умов ми пропонуємо наступний підхід: кожному члену команди буде призначена посада відповідно до рівня його досвіду, а між членами команди буде чітка ієрархія спілкування. Буде 3 рівні: молодший, середній і старший. Зараз компанії зазвичай використовують цю градацію для команд, тому реалізувати таку структуру не складе труднощів. Для цієї класифікації найпростіше буде використовувати досвід роботи. Це можна зробити так: Junior (0-3 роки досвіду), Intermediate (3-5 років досвіду), Senior (5+ років досвіду). Кожен старший співробітник буде наставляти 1-2 середніх співробітників. При цьому кожен співробітник рівня Middle буде наставником для 2-3 співробітників рівня Junior. У цьому випадку кожен менш досвідчений член команди зможе при необхідності звернутися за допомогою до наставника. Крім того, таке тісне спілкування підвищить ефективність тренувальних процесів членів і допоможе побудувати міцні та дружні стосунки між наставниками та їх вихованцями, що дуже важливо для спільного командного духу. Є ще одна практика, яка може значно підвищити ефективність описаного підходу: перегляд коду. Будь-які зміни необхідно переглянути, перш ніж вони будуть об'єднані з основним кодом і запущені будь-які процеси забезпечення якості.

Підсумовуючи, зазначимо, що в сучасних реаліях ІТ-індустрії існує необхідність модифікації існуючих методологій управління проектами або розробки нових, які забезпечать ефективну ітераційну розробку в умовах обмежених людських і фінансових ресурсів, а також в умовах обмеженості часу. Ці практики можуть бути використані в майбутньому для розробки нової стандартизованої методології управління проектами.

Список використаних джерел:

1. Pavlenchuk, N., Horbonos, F., Pavlenchuk, A., Skrynkovskyu, R., Pawlowski, G. Increasing the competitiveness of enterprises based on the use of marketing management tools. *Agricultural and Resource Economics: International Scientific E-Journal*, 2021. 7(3), 77–89. doi: <https://doi.org/10.51599/are.2021.07.03.05>

2. Pavlenchyk, N., Pavlenchyk, A., Skrynkovsky, R., Tsyuh, S. The influence of management creativity on the optimality of management decisions over time: An innovative aspect. *Journal of Eastern European and Central Asian Research (JEECAR)*, 2023. T10 №3. C 498-514. DOI: 10.15549/jeecar.v10i3.1318

3. Skrynkovsky, R., Pavlenchyk, N., Tsyuh, S., Zanevskyy, I., Pavlenchyk, A.. Economic-mathematical model of enterprise profit maximization in the system of sustainable development values. *Agricultural and Resource Economics: International Scientific E-Journal*, 2023. 8(4), 188–214. doi: <https://doi.org/10.51599/are.2022.08.04.09>

4. Corona E., Pani F.E. *A Review of Lean-Kanban Approaches in the Software Development*. 2013.

5. State of Kanban Report” Kanban University. Accessed October 2022. <https://kanban.university/state-of-kanban/> (2022).