

Близнюкова Ірина ОлександрівнаАспірант кафедри комп'ютерних наук та системного аналізу, orcid.org/0000-0001-7486-7983

Черкаський державний технологічний університет, Черкаси

Семко Інга БорисівнаКандидат технічних наук, доцент кафедри електротехнічних систем, orcid.org/0000-0002-6251-5830

Черкаський державний технологічний університет, Черкаси

Кійко Сергій ГеннадійовичКандидат технічних наук, Голова правління, orcid.org/0000-0003-4676-5722

ПрАТ Електрометалургійний завод «Дніпроспецсталь» ім. А. М. Кузьміна, Запоріжжя

ОГЛЯД СУЧАСНИХ МЕТОДОЛОГІЙ УПРАВЛІННЯ КОМАНДАМИ ІТ-ПРОЄКТІВ

Анотація. Застосування проєктних методів управління все більше охоплює різні сфери бізнесу з використанням різноманітних інструментаріїв. Сучасні темпи розвитку в умовах ринкової жорсткої конкуренції вимагають приділяти значну увагу ефективності команд, від якості роботи яких залежать обґрунтованість розподілення інвестицій замовника. Проведено огляд найбільш затребуваних сучасних методологій управління командами ІТ-проєктів, ефективного командування. Розкрито розуміння, що одним із шляхів забезпечення успішності проєкту є саме створення проєктної команди. Процес формування команди проєкту ускладнюється кваліфікацією та спрямованістю спеціалістів. Це вимагає в кожному конкретному випадку використовувати різні механізми, не забуваючи при цьому, цілі ІТ-проєкту. Розглянуто методології Scrum, Kanban, XP, Lean, Six Sigma. Надано основні характеристики кожного, особливості, структура та недоліки в командному управлінні. В межах гнучких методів передбачені особливості, які й роблять ці методи гнучкими: короткі спринти, відсутність формалізації функцій членів команди, звітної документації, команда взаємодіє із замовником, простота та мінімізація роботи, швидке реагування на зміни. Методи добре зарекомендували себе при роботі в умовах з високою невизначеністю. Гнучкий підхід до організації роботи допомагає швидко виводити продукт на ринок, робити продукт якіснішим, покращувати клієнтський досвід. Для ІТ-проєктів гнучкі методології краще, ніж класичні моделі, але компанії продовжують використовувати традиційну методологію, що дозволяє їм отримати успішну реалізацію проєктів лише в 50% випадках, тоді як гнучкі цей показник перевищують (понад 60% успіху).

Ключові слова: методологія; ІТ-проєкти; команда; ролі; управління проєктами

Вступ

ІТ-проєкти, безумовно, характерні для компаній з різноманітними організаційними структурами управління, кількістю працюючих, потенційного масштабу проєкта та профілем діяльності, але всі вони мають деякі загальні риси.

ІТ-проєкти найбільш ризикові, динамічні проєкти, що характеризуються значною кількістю проблем, високим рівнем напруженості та конфліктів, часто вимагають рішення нетипових завдань, і за статистикою лише третина з них завершується цілком успішно [1]. Успіх впровадження та управління ІТ-проєктами напряму залежить від команди фахівців, їх інформативності, розподілу ролей, повноважень, відповідальності.

Процес створення команди проєкту ускладнюється кваліфікацією та спрямованістю спеціалістів, участь яких протягом проєкту буде

необхідна. Створення команди ІТ-проєкту може ускладнюватися і тим фактором, що в системі управління команда не має прямого контролю над всіма спеціалістами, в роботі яких є потреба.

Ще одним із факторів, що впливає на успішність впровадження ІТ-проєктів, є відсутність ідеальної системи управління проєктами, яка б на 100% була прийнятною і керівникам, і всім членам команди [2].

Наприклад, на металургійному підприємстві «Дніпроспецсталь» для вирішення завдань планування й управління портфелем проєктів енергозбереження, а також енергоменеджменту реалізуються проєкти впровадження цілої низки автоматизованих систем, ядром при цьому виступає корпоративна інформаційна система управління на базі SAP ERP (ECC 6.0) [3].

Отже, питання формування команди є досить актуальним задля забезпечення завершення проєкту в межах терміну та бюджету.

Мета статті

Мета статті полягає в проведенні огляду сучасних механізмів управління командами ІТ-проектів для з'ясування, яка з методологій дає змогу сформувати професійну ефективно працюючу команду та виявити недоліки, які необхідно буде уникати.

Виклад основного матеріалу

Теоретичні основи командної роботи висвітлені в роботах М. Амстронга, Джона С. Максвелла, Річарда Л. Дафта, також дослідженнями у сфері управління командами проекту займалися такі вітчизняні вчені, як: С. Д. Бушуєв, І. Ю. Лебедева, С. В. Леонова, В. О. Москаленко та ін.

Команда проекту – група осіб, яка підтримує керівника проекту у виконанні проекту для досягнення цілей проекту [4]. Команда проекту створюється на період реалізації проекту та поєднує людей таким чином, що робота набуває синергічного ефекту. Особливості роботи в команді допускають, що кожен окремо взятий член команди не обов'язково повинен мати повний набір навичок та досвіду для виконання проекту. В індивідуальних особливостях виконавців є причина створення команд, де сильні сторони кожного співробітника доповнюють слабкі сторони колег. Обмеженість ресурсів може компенсуватися за рахунок професіоналізму та ефективності сумісних дій усіх членів [5].

Життєвий цикл команди послідовно проходить декілька етапів, які змінюють один одний:

- формування команди (постановка цілей, розподілення ролей в команді);
- притирання (осмислення цілей, визначення спільного вектора руху);
- нормалізація (досягнення цілей внаслідок компромісів та налагодження комунікації);
- функціонування (збільшення продуктивності за рахунок оптимізації процесів розробки і самоорганізації членів команди);
- розформування команди (отримання результатів, завершення проекту).

Провідні компанії спочатку визначаються з методологією впровадження проекту, а вже потім створюють ідеальну команду.

В управлінні проектами значну роль відіграють застосовані підходи до того чи іншого процесу: жорсткі (орієнтовані на задачу), гнучкі (орієнтовані на команду) та незалежні команди з високим ступенем незалежності [6].

Методи, за якими працюють команди ІТ-проектів різнопланові, серед них розрізняють: Scrum, Kanban, XP, Lean та інші.

Scrum – каркас розробки, з використанням якого людина може вирішувати продуктивно проблеми,

при цьому виробляючи продукти високої значущості [7].

Переважає більшість практиків, вважають, що Scrum – це гнучка методологія Agile, філософія розроблення програмного забезпечення (ПЗ), сукупність методів розроблення і управління, гармонічно поєднаних між собою, в основі яких лежать принципи виробничої ітеративності, що в свою чергою, допомагають замовнику частіше отримувати певний робочий функціонал в рамках розроблення проекту. Відповідно до методології Scrum [8] всі члени команди проекту повинні бути взаємно замінні і рівнозначні.

Якщо команді не достатньо знань для виконання проекту, то якість проекту може бути під загрозою. Однак метод характеризується відсутністю жорстко заданого розподілу ролей та обов'язків у команді. Принцип «всі відповідальні за все» працює далеко не завжди.

Крім того, в Scrum добре реалізований і сам процес збирання функції та її розподіл по ітераціях. Як і багато інших гнучких методологій, Scrum вимагає наявності представника замовника або представника ринку, але замовник не завжди може визначити, що саме йому потрібно, а часта зміна вимог лише уповільнює роботу [9].

Підсумовуючи, можна сказати, що Scrum – це набір принципів, на яких будується процес розроблення, який дає змогу в короткі проміжки часу (спринти) надавати кінцевому користувачеві робоче програмне забезпечення з доданими можливостями, для яких було визначено найбільший пріоритет. Потрібний функціонал для реалізації в черговому спринті визначається до його початку на етапі планування і не може змінюватися протягом всього спринта. При цьому жорстко фіксований не великий термін спринта додає процесу розроблення гнучкості [10].

Класичний Scrum містить три ролі в команді, що відображено на рис. 1.

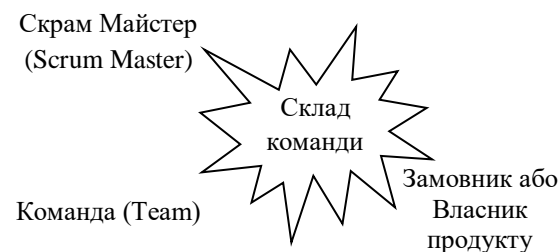


Рисунок 1 - Склад команди в методології Scrum

Скрам Майстер, найважливіша роль (менеджер проекту) в методології, є з'єднуючою ланкою між замовником і командою. Основні обов'язки пов'язані зі створенням робочої атмосфери довіри, ліквідації перепонів, дотримання практик і процесу в команді.

Замовник (власник продукту) відповідає за розроблення продукту. Як правило, це менеджер продукту або представник замовника, якщо це аутсорс продукт. Замовник приймає кінцеві рішення для команди в проєкті, відповідає за формування і бачення продукту, керує рентабельністю та очікуваннями замовників і всіх стейкхлдерів, координує і розставляє за пріоритетами потік задач, взаємодіє з командою і замовниками.

В методології Scrum команда є самоорганізованою та самокеруючою, бере на себе обов'язки виконання задач перед Замовником. Основні обов'язки команди відображені на рис. 2.

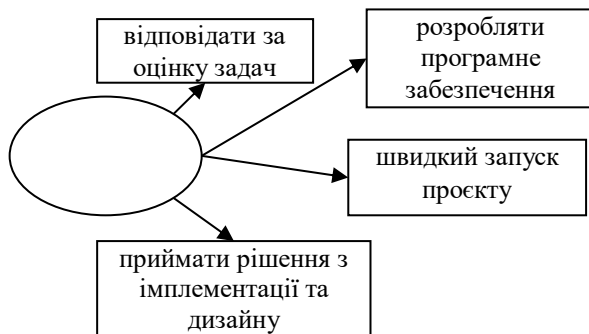


Рисунок 2 – Схема основних обов'язків команди

І як висновок, Scrum має основні особливості (рис. 3).

Scrum має і недоліки:

- простота та мінімалізм ведуть до більш жорстких вимог, що створює «конфлікт» з головним принципом клієнтоорієнтованості;
- самоорганізація та багатофункціональність команди в Scrum веде до підвищення витрат на відбір персоналу, мотивацію, навчання.

Розроблений для проєктів, в яких необхідні «швидкі перемоги» в поєднанні з толерантністю до

змін. Крім того, цей фреймворк підходить для ситуацій, коли не всі члени команди мають достатній досвід в тій сфері, в якій реалізується проєкт – постійні комунікації між членами командами дозволяють нівелювати брак досвіду або кваліфікації одних співробітників за рахунок інформації і допомоги від досвідчених колег.

Ще одна з досить поширених – методологія Kanban, яка бере свій початок на виробничих лініях компанії TOYOTA та активно запроваджує систему «ощадливого виробництва», тобто виникла ідея створення єдиного потоку (процесу) без простою від незавершених або неузгоджених завдань, відсутності паузи. Це допомогло уникнути безглузких витрат, покращити якість продукту, знизити вартість та скоротити терміни виконання. Kanban працює в командах підтримки, тобто:

- групи підтримки програмного забезпечення, де важлива швидкість реагування на зміни;
- групи тестування, що працюють окремо від груп розробки;
- служби підтримки;
- інші приклади «неосновних виробництв» [11].

Необхідно зауважити, що Kanban добре працює на стартапах, тобто там, де не має чіткого плану, але активно працюють над розробкою (рис. 4).

ІТ-компанія, яка при виробництві застосовує ітеративну методологію для відстеження оперативної ситуації всередині використовує Kanban-дошку – об'єктивний елемент гнучкої методології.

Кожен член команди отримує до неї доступ в будь-який час і бачить, на якому етапі перебуває завдання. Kanban-дошка підлаштовується під будь-який процес та застосовується в будь-якій галузі.

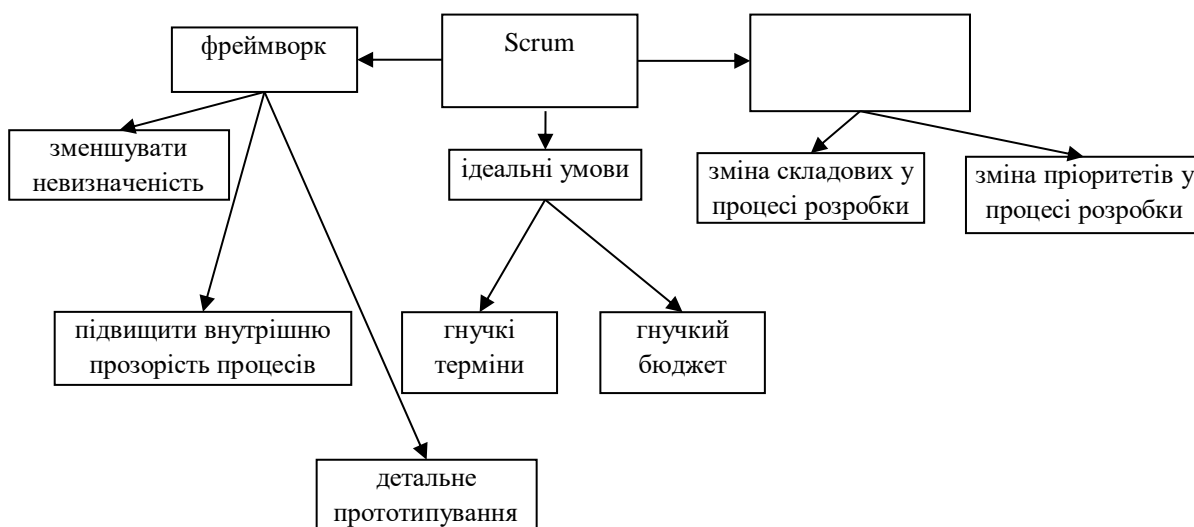


Рисунок 3 – Схема основних особливостей методології Scrum

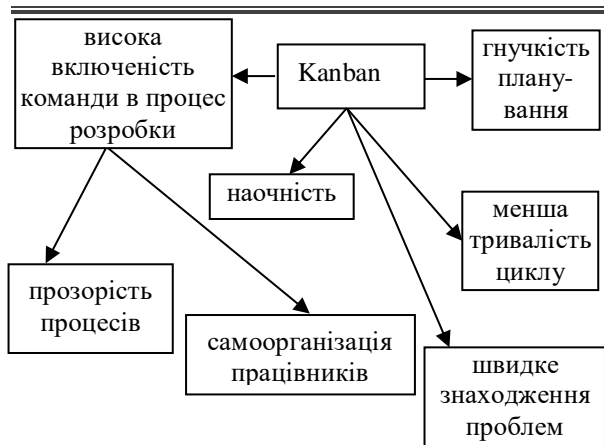


Рисунок 4 – Схема основних особливостей методології Kanban

У кожного проекту є план процесу робіт. Спочатку його аналізують та поділяють дошку на стовпці-етапи, наприклад, для процесу створення ІТ-проєкту: To do – показує кількість завдань, необхідних для виконання; In progress – завдання, які вирішуються в даний момент; Ready for deploy – завдання, які вже виконані, але не представлені в тестовому середовищі; QA – завдання в процесі тестування; PM approving – готові завдання проходять перевірку проєктним менеджером; Done – виконані (завершені) завдання поточного спринту.

Процеси в спринті відбуваються таким чином: взявшись до роботи, програміст перетягує завдання з колонки «To do» в «In progress». В цей момент часу він має тільки одну задачу, концентрується на її ефективному вирішенні, так як паралельне виконання завдань не допускається. Після закінчення своєї частини роботи, він перетягує її в колонку «Ready for deploy» [10].

І як висновок, Kanban має основні особливості.

Kanban має і свої недоліки:

- команди мають бути чисельністю не більше п’яти осіб;

– не призначений для довгострокового планування.

Добре зарекомендував Kanban в стартапах, які не мають чіткого плану, але активно працюють над розробкою. Точний розрахунок навантаження на команду, правильна розстановка обмежень і концентрація на постійне поліпшення – все це допомагає економити ресурси і укладатись в дедлайни та бюджет.

У таблиці подана порівняльна характеристика методологій Scrum і Kanban.

Методологія Lean використовує концепції зменшення витрат та бережливого виробництва, допомагає знаходити проблеми в процесі з подальшим вирішенням їх за допомогою різних практик та інструментів з постійним вдосконаленням на всіх етапах реалізації. Основні принципи концепції бережливого виробництва:

- визначити цінність конкретного продукту;
- визначити потік створення цінності для цього продукту;
- забезпечити безперервне (протягом всього потоку) створення цінності продукту;
- дати змогу споживачеві витягати продукт;
- прагнути досконалості [12].

Методологія Lean може змінювати ієрархію в структурі організації – формується «община» багатопрофільних співробітників. Команда використовується на всі 100%: кожен може запропонувати ідею з подальшою її реалізацією та кожен відповідальний за комплексний результат. Така гнучка система дає змогу миттєво реагувати та вносити зміни.

Методологія передбачає постійне вдосконалення процесу через призму недопущення непотрібних витрат, виправлення попередніх помилок в управлінні (муда); нерівномірності та невідповідності навантаження (мура); недоцільності, складності в роботі команди (мурі).

Таблиця – Порівняльна характеристика методологій Scrum та Kanban

Характеристика	Kanban	Scrum
Темп	Повторювані спринти фіксованої тривалості	Безперервний процес
Випуск релізу	В кінці кожного спринту після схвалення проєктним менеджером (власником товару)	Потік триває без перерв або на розсуд команди
Ролі	Власник продукту, Scrum-майстер, команда розробників	Команда під керівництвом проєктного менеджера; в деяких випадках залучаються тренери по agile kanban
Головні показники	Швидкість команди	Провідний час
Прийнятність змін	У ході спринту зміни небажані, так як можуть призвести до неправильної оцінки задач	Зміни можуть трапитися в будь-який момент

На рис. 5 надана схема основних особливостей методології Lean.

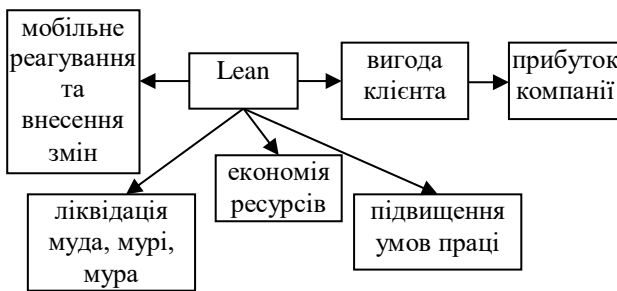


Рисунок 5 – Схема основних особливостей методології Lean

Методологія Lean – це зміни, філософія постійного вдосконалення, взаємодія на всіх рівнях та пряма комунікація.

Недоліки:

- постійний контроль управління з метою виявлення потенційних проблем та підтримки ефективності виробництва;

- на відміну від Scrum, не пропонує чіткого робочого процесу для реалізації частин проєкту, що веде до збільшення строку виконання проєкту;

- модифікація може призвести до нерівномірності навантаження на команду, що своєю чергою веде до невдоволення співробітників.

Методологія XP – гнучка методологія розроблення програмного забезпечення, має свої інструменти, ролі. Команда включає всіх учасників ІТ-проєкту. Для успішної реалізації проєкту команда програмістів повинна спільно працювати із замовником, який визначає функціональність на наступну ітерацію та вносить зміни щодо вимог продукту.

Переваги екстремального програмування такі:

- замовник отримує саме той продукт, який йому потрібен;

- команда швидко вносить зміни в код, додає нову функціональність;

- код працює за рахунок постійного тестування та інтеграції;

- швидкі темпи розроблення за рахунок парного програмування;

- зменшення ризиків та витрат на розроблення.

Методологія XP не завжди спрацьовує, тому що успішність залежить від замовника та професійного рівня програмістів. Крім того, складно передбачити час на реалізацію проєкту, відсутність документації (важлива складова великих проєктів).

Six Sigma схожа на Kanban, але має етапи реалізації задачі (планування, визначення цілей,

тестування якості). Система легко адаптується до вимог компанії та команди. Концепція Six Sigma має чітку схему для реалізації проєкту та постійного покращення процесів. Аналіз визначених цілей дає змогу отримувати кількісні дані для більш глибокого розуміння суті та вимог проєкту, в результаті чого отримують якісні рішення.

Six Sigma підходить для складних ІТ-проєктів.

Недоліки:

- плутанина в пріоритетах на різних етапах проєкту;

- прагнення зробити ще краще призводить до демотивації команди;

- якщо проєкт одноразовий, то витрати на аналіз та оптимізацію процесів марні [13].

Гнучкі методології в управлінні ІТ-проєктами спрямовані на зменшення рутинної роботи та проєктної документації, маючи на меті збільшення ключових показників ефективності команди за рахунок надійності процесів на всіх етапах проєкту та дотримання встановлених термінів.

Суть методологій криється в розподілі задач проєкту на ітерації з детальним плануванням та чітко прописаним часом реалізації. Прийняття рішень членами команди залежить від проміжних та прозорих результатів проєкту, а отже, постійно відбувається контроль за статусом робіт проєкту, змінами, проблемами [14].

Гнучкі методології розробки програмного забезпечення вимагають універсальності та мультифункціональності кожного працівника команди проєкту [7].

За даними аналітичної компанії Ambysoft для ІТ-проєктів гнучкі методології кращі за класичні моделі, але компанії продовжують використовувати традиційну методологію, що допомагає їм отримати успішну реалізацію проєктів лише в 50% випадків, тоді як гнучкі методології цей показник перевищують (понад 60% успіху) [15].

Висновки

Успішність реалізації ІТ-проєкту залежить від доволі багатьох складових, але завжди є можливість зробити певні кроки назустріч успіху, головний з яких – це правильне формування команди. Злагоджена та ефективна робота команди, дотримання правила пріоритетів, відповідні підходи управління проєктами забезпечує умови успішної реалізації проєкту.

Проведений огляд гнучких методологій демонструє широке коло вибору технологій для створення команди та ефективної командної роботи.

Список літератури

- 1 Новахацька Д. В. Особливості та проблеми реалізації ІТ-проектів в Україні // Вісник ЧДТУ. – № 2. 2016. – С. 72–77.
- 2 Laing Samantha and Hryvs Karen. Growing agile: a coach's guide to training Scrum.
- 3 Кійко С. Г. Предиктивна адаптація при управлінні портфелем проектів енергозбереження на металургійному підприємстві / С. Г. Кійко // Наука і техніка Повітряних Сил Збройних Сил України. – 2020. – № 4(41). – С. 133–144. <https://doi.org/10.30748/ntps.2020.41.16>
- 4 Руководство к своду знаний по управлению проектом (Руководство РМВОК) : Шестое издание Newtown Square, PA: Project Management Institute. 2017. – 762 с.
- 5 Чернов С. К., Левит А. А. Основы эффективной работы проектной команды // Радиоэлектрон. і комп'ют. системи. – № 3. 2007. – С. 128–133. – Библиогр.: 8 назв. – рус
- 6 Гулюк Н. В. Принципы успешного управления проектами // Бизнес-образование в экономике знаний, ИГУ. – № 2. 2017. – С.36–39.
- 7 Schwaber K. and J. Sutherland The Scrum Guide. The definitive Guide to Scrum: The Rules of the Game. 2017.
- 8 Schwaber, Ken; Beedle, Mike (2002). Agile software development with Scrum // Prentice Hall PTR Upper Saddle River, NJ, USA, 2001
- 9 Сазерленд Д. Scrum. Революционный метод управления проектами. М.: Манн, Иванов и Фербер. 2016. – 288 с.
- 10 Кон М. Гибкая разработка ПО. М.: Вильямс. 2011. – 576 с.
- 11 Пушкарев А. Методология Канбан: Введение. [Электронный ресурс]. – <http://m.habr.com>
- 12 Вумек Дж.П. Бережливое производство. Как избавиться от потерь и добиться процветания вашей компании. М.: Альпина Паблишер. 2011.
- 13 Guay Matthew Project Management 101: The Complete Guide to Agile, Kanban, Scrum and Beyond, 2016. [Электронный ресурс]. – <http://zapier.com/learn/ultimate-guide-to-project-management/project-management-systems/>
- 14 Андреева Р. Н., Синяева О. Ю. Scrum: гибкость в жестких рамках. Актуальные вопросы управления // Весник университета ГУУ. – № 2. 2018. – С.13–20.
- 15 Громова А. А. Специфика управления проектами в сфере информационных технологий: Материалы Афанасьевских чтений. 2016. – С.135–143.

Стаття надійшла до редколегії 03.09.2020

Blyznyukova Iryna

Postgraduate student of the Department of Computer Science and Systems Analysis, orcid.org/0000-0001-7486-7983
Cherkasy State Technological University, Cherkasy

Semko Inga

PhD (Eng.), Associate Professor of Electrical Systems, orcid.org/0000-0002-6251-5830
Cherkasy State Technological University, Cherkasy

Kiyko Sergey

PhD (Eng.), Chairman of the Board, orcid.org/0000-0003-4676-5722
PJSC Electrometallurgical Plant "Dniprospsstal" named after A.M. Kuzmina, Zaporizhia

OVERVIEW OF MODERN TEAM MANAGEMENT METHODOLOGIES IT PROJECTS

Abstract. The application of project management methods is increasingly covering various areas of business using a variety of tools. The current pace of development in conditions of fierce market competition requires attention to the effectiveness of teams, the quality of which depends on the validity of the distribution of customer investment. The article reviews the most popular modern methodologies for managing IT-project teams, effective team building. It is understood that one of the ways to ensure the success of the project is to create a project team. The process of forming a project team is complicated by the qualifications and orientation of specialists. This requires the use of different mechanisms in each case, without forgetting the goals of the IT- project. The methodologies of Scrum, Kanban, XP, Lean, Six Sigma are considered. The main characteristics of each, features, structure and shortcomings in team management are given. The flexible methods include features that make these methods flexible: short sprints, lack of formalization of team members' functions, reporting documentation, the team interacts with the customer, simplicity and minimization of work, quick response to changes. The methods have proven themselves well when working in conditions of high uncertainty. A flexible approach to the organization of work allows you to quickly bring the product to market, make the product better, improve the customer experience. For IT-projects, flexible methodologies are better than traditional models, but companies continue to use traditional methodology, which allows them to successfully implement projects in only 50% of cases, while flexible, this indicator exceeds more than 60% of success.

Keywords: methodology; IT projects; team; roles; project management

References

1. Novokhatska, D. V. (2016). *Features and problems of implementation of IT-projects in Ukraine*. *Bulletin of ChSTU*, 2, 72 – 77.
2. Laing, Samantha & Hryvs, Karen. *Growing agile: a coach's guide to training Scrum*.
3. Kiyko, S. G. (2020). *Predictive adaptation in the management of the portfolio of energy saving projects at the metallurgical enterprise. Science and technology of the Air Force of the Armed Forces of Ukraine*, 4(41), 133 – 144.
4. *Guide to the Project Management Body of Knowledge (PMBOK Guide)*. (2017): Sixth Edition Newtown Square, PA: Project Management Institute, 762.
5. Chernov, S. K., Levit, A. A., (2007). *Fundamentals of effective work of the project team*. *Radioelectronic and computer systems*, 3, 128 – 133.
6. Gulyuk, N. V. (2017). *Principles of Successful Project Management*. *Business Education in the Knowledge Economy*, SU, 2, 36 – 39.
7. Schwaber, K. & Sutherland, J. (2017). *The Scrum Guide. The definitive Guide to Scrum: The Rules of the Game*.
8. Schwaber, Ken; Beedle, Mike. (2002). *Agile software development with Scrum*. Prentice Hall PTR Upper Saddle River, NJ, USA.
9. Sutherland, D. (2016). *Scrum. A revolutionary method of project management*, 288.
10. Kon, M. (2011). *Flexible software development*, 576.
11. Pushkarev, A. *Kanban Methodology: Introduction*. [electronic resource] – <http://m.habr.com>
12. Wumek, J. P. (2011). *Lean. How to get rid of losses and make your company prosperous*.
13. Guay, Matthew. (2016.). *Project Management 101: The Complete Guide to Agile, Kanban, Scrum and Beyond* [electronic resource]. – <http://zapier.com/learn/ultimate-guide-to-project-management/project-management-systems/>
14. Andreeva, R. N., Sinyaeva, O. Yu. (2018). *Scrum: flexibility within a rigid framework*. *Topical issues of management*. *Vesnik University GUU*, 2, 13 – 20.
15. Gromova, A. A. (2016). *The specifics of project management in the field of information technology*. *Materials of the Afanasiev Readings*, 135 – 143.

Посилання на публікацію

- APA Blyznuikova, I., Semko, I. & Kiyko, S. (2020). *Overview of modern team management methodologies in IT-projects. Management of Development of Complex Systems*, 43, 60 – 66; [dx.doi.org\10.32347/2412-9933.2020.43.60-66](https://doi.org/10.32347/2412-9933.2020.43.60-66).
- ДСТУ Близнюкова І. О. Огляд сучасних методологій управління командами ІТ-проектів. [Текст] / І. О. Близнюкова, І. Б. Семко, С. Г. Кійко // *Управління розвитком складних систем*. – 2020. – № 43. – С. 60 – 66; [dx.doi.org\10.32347/2412-9933.2020.43.60-66](https://doi.org/10.32347/2412-9933.2020.43.60-66).