

Методології управління проєктами: Scrum, Kanban

Вінницький національний технічний університет

Анотація

У роботі було розглянуто методології Agile - Scrum, Kanban. Наведено їх переваги та недоліки.

Ключові слова: менеджмент, управління проєктами, гнучні методології, Scrum, Kanban, ефективність, Agile

Abstract

The methodologies of Agile - Scrum, Kanban were considered in the work. Their advantages and disadvantages are given

Keywords: management, project management, flexible methodologies, Scrum, Kanban, efficiency, Agile

Вступ

В основі Scrum, Kanban методології лежать принципи Agile. Сам Agile – це сімейство «гнучких» підходів до розробки програмного забезпечення. Такі підходи іноді називають фреймворками або agile-методологіями. Сенс Agile сформульований в Agile-маніфесті розробки ПЗ: «Люди та взаємодія важливіші за процеси та інструменти» [1]. Працюючий продукт важливіший за вичерпну документацію. Kanban та Scrum - обидві системи розроблені таким чином, щоб забезпечити швидкий і стабільний темп роботи, який можна адаптувати до мінливих потреб клієнтів.

Результати дослідження

Kanban — це концепція підтримки безперервного потоку завдань та їх ефективного виконання завдяки максимальній ефективності роботи команди. Водночас команді ніколи не дається більше роботи, ніж вона здатна виконати. Якщо просто: ви складаєте список завдань, а люди їх виконують. Kanban має два основні принципи: візуалізації роботи; обмеження кількості завдань “у процесі”. Візуалізуючи задачі визначають, які задачі необхідно виконати, документують їх на картках та розміщують на Kanban-дошці. Картки можуть бути цифровими, наприклад, у програмі для управління проєктами, або реальними фізичними картками — наліпки на дошці в офісі. Обмеження кількості завдань “work-in-process”: після організації дошки, команді потрібно спокійно працювати. Колеги будуть брати завдання в роботу, закінчувати їх, а потім переходити до наступного пункту в списку. Щоб роботи не стало стільки, скільки команда не зможе виконати, в кожному стовпці на дошці повинна бути обмежена кількість карток — тобто стільки завдань, скільки реально можна виконати за встановлений проміжок часу. Тому після досягнення цієї межі картки більше не додають, поки частина роботи не буде виконана. Такий підхід допомагає виявити вузькі місця. Якщо виникла ситуація, коли завдання ще є, але ресурсів для їх виконання вже немає, команда зосереджує зусилля саме на цьому сегменті [2].

Scrum — це популярна сучасна методологія управління проєктами. Причиною її виникнення стала неефективність методології Waterfall (каскадна модель) [3] для середовища, наповненого невизначеністю, що виникає під час сучасного процесу розробки програмного забезпечення. Виникає необхідність методики, яка дозволила би швидше випустити програмне забезпечення, для процесу тестування та збору даних про необхідні зміни під час наступної ітерації. Ідея полягає в тому, щоб розбити велику задачу на мінімальні підзадачі, які можна виконати в деякий часовий період. Тому роботу над проєктом розділяється на часові інтервали, так звані спринти [3]. Зазвичай використовують для спринтів проміжок часу 14 днів, хоча можна встановити інший: однотижневий або тритижневий. Але вони не мають бути надто короткими або довгими.

Як і в Kanban, для кожного завдання використовуються картки та створюється загальна дошка, розділена на періоди. Різниця полягає в тому, що Scrum має більше елементів планування. Протягом кожного спринту команда декілька разів зустрічається на спільне обговорення: для планування спринту, його огляду, на ретроспективну зустріч, а також на щоденні наради. Для кожного спринту під час планування встановлюється мета. І всі задачі на спринт мають бути прив’язані до цієї мети.

Під час розгляду спринту, результати роботи оцінюються і, якщо їх затверджено, «випускаються в світ». Scrum також передбачає розподілення конкретних ролей у команді. Власник продукту, який представляє замовника, забезпечує роботу колективу. Scrum-майстер контролює процес та полегшує його для команди [4].

Висновки

Розглядаючи вищеописані методології організації роботи над проектами по розробці програмного забезпечення можна виділити наступні моменти.

Принципи Kanban прості, тому їх легко застосовувати та адаптувати для будь-якої компанії. Також у Kanban можна інтегрувати інші принципи управління проектами, щоб керувати робочим процесом та постійно його розвивати, наприклад, зворотний зв'язок та щоденні зустрічі.

Основними Перевагами методології Kanban є підвищення ефективності роботи команди над проектом, шляхом налагодження взаємодії між її учасниками та висока гнучкість. Економія часу на обговорення та наради. З переваг Scrum можна виділити передбачуваність результатів роботи, швидке вдосконалення продукту, підвищення самоорганізації колективу.

Недоліки Scrum: потреба у структуруванні робочий процесу та плануванні робочого навантаження вимагає введення ще одного співробітника Scrum-мастера, збільшення кількості нарад. Методологія Scrum вимагає часових витрат на її впровадження [4]. Поряд з цим недоліками Kanban є ризики швидкого гальмування процесу розробки, вимога постійного наявності та своєчасного оновлення завдань, відсутність часових обмежень та ризики «втрати» задач з високим пріоритетом.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Agile-маніфест розробки ПЗ [Електронний ресурс]. – Режим доступу: <https://agilemanifesto.org/iso/uk/manifesto.html> – Заголовок з екрану.
2. Андерсон, Д. Канбан. Альтернативний шлях у Agile / пер. с англ. А. Карабейников – М.: «Манн, Иванов и Фербер», 2017.– 350 с.
3. Сазерленд Д. Scrum. Революційний метод управління проектами, 2-ге вид. 2021. – 288 с
4. Вольфсон, Б. Гнучке управління проектами та продуктами. – СПб.: П, 2021. – 144 с.

Людва Назарій Вікторович — студент групи ІБС-20б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: nazarliudva@gmail.com

Остапенко-Боженова Аліна Василівна — асистент кафедри Захисту інформації, Факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, email: ostapenko_bozhenova@vntu.edu.ua

Liudva Nazarii Viktorovych - student of group 1BS-20b, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: nazarliudva@gmail.com

Ostapenko-Bozhenova Alina V. — assistant in Information Protection department, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, email: ostapenko_bozhenova@vntu.edu.ua