

УДК 005.004.01

ИСПОЛЬЗОВАНИЕ SCRUM, KANBAN В ПРОЕКТАХ ГИБКОЙ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ РАЗЛИЧНЫХ СФЕР ДЕЯТЕЛЬНОСТИ ОРГАНИЗАЦИЙ



Ю.Н. Метельская,
студент БГУИР



П.С. Шафранович,
студент БГУИР

Белорусский государственный университет информатики и радиоэлектроники, Республика Беларусь
E-mail: metelskaya2808@gmail.com, polina.638@bk.ru

Ю.Н. Метельская

Обучается в Белорусском государственном университете информатики и радиоэлектроники по специальности информационно-коммуникационных технологий в экономике.

П.С. Шафранович

Обучается в Белорусском государственном университете информатики и радиоэлектроники по специальности информационно-коммуникационных технологий в экономике.

Аннотация. Инновации в агропромышленном комплексе представляют собой достижения науки и техники, необходимые для повышения производительности труда, продуктивности производства, эффективности существования всех отраслей сельского хозяйства. Необходимо обоснованно подходить к выбору гибкой методологии разработки программного обеспечения, а также выбора подхода к разработке в проектах для агропромышленного комплекса.

Ключевые слова: Гибкая методология, Scrum, Kanban, разработка ПО, спринт.

Введение. Сегодня отрасль агропромышленного комплекса – один из самых динамично развивающихся секторов экономики. Ускоренное развитие информационных технологий в агропромышленном комплексе набирает обороты. Информационные технологии крайне необходимы в сельском хозяйстве – начиная от управления трактором с помощью спутниковых технологий и заканчивая автоматизацией крупных перерабатывающих предприятий. Спектр работ для сферы информационных технологий просто огромный.

Цель. Целью работы является обоснование гибкой методологии разработки программного обеспечения, а также выбора подхода к разработке в различных проектах.

Материал и методика исследования. В ходе исследования были использованы методы анализа, синтеза и другие. При написании работы использовались статистические сборники, исследования подходов гибкой разработки программного обеспечения.

Результаты исследования. В условиях постоянных изменений внешней и внутренней среды необходимо уметь на них ориентироваться. А также не забывать об обратной связи от заказчиков и пользователей. Это заставляет разработчиков экспериментировать и искать новые решения, не ограничивая себя жесткими рамками и стандартами. К отдельным agile-подходам относятся Scrum и Kanban. Обе считаются гибкими и итеративными.

Метод Scrum – это один из вариантов гибкой разработки программного обеспечения.

Scrum – это «подход структуры». Над каждым проектом работает универсальная команда специалистов, к которой присоединяется еще два человека: владелец продукта и Scrum-мастер. Первый соединяет команду с заказчиком и следит за развитием проекта. Это не

формальный руководитель команды, а скорее куратор. Второй помогает первому организовать бизнес-процесс: проводит общие собрания, решает бытовые проблемы, мотивирует команду и следит за соблюдением Scrum-подхода.

Процесс веб-разработки начинается с планирования спринта. Происходит встреча всей команды и формируется список задач для backlog спринта. На данном этапе оценивается по времени каждая задача и обсуждается взаимодействие между членами команды. Количество задач бэклога не ограничено, но время их выполнения должно строго соответствовать выделенному времени для спринта. Это одна из самых важных встреч, так как формирует основу всего спринта.

В процессе разработки каждый день происходит короткая встреча команды. Основная цель – поделится опытом и оценить, как проходит процесс разработки. Scrum-мастер контролирует процесс встречи. После завершения спринта проходит обзор выполненной работы. Завершенный модуль показывается владельцу продукта или клиенту. После обзора проходит встреча команды для обсуждения оптимизации работы. Именно здесь можно обсудить организационные проблемы, чтобы в будущих спринтах это учесть и улучшить процесс разработки.

Метод Kanban позволяет визуально структурировать процесс разработки. По сути, это огромная доска, на которой задания перемещаются между секциями.

Kanban – это «подход баланса». Его задача – сбалансировать разных специалистов внутри команды и избежать ситуации, когда дизайнеры работают сутками, а разработчики жалуются на отсутствие новых задач.

Вся команда едина: в Kanban нет ролей владельца продукта и Scrum-мастера. Бизнес-процесс делится не на универсальные спринты, а на стадии выполнения конкретных задач. Доска может состоять из множества столбцов. Чаще всего используются 4 основных столбца:

–«To do». Для каждого задания создается своя карточка. За каждой карточкой закрепляется разработчик, тот человек, который должен выполнить это задание.

–«In progress». Карточка перемещается в эту стадию, когда разработчик начинает работать над своим заданием. Когда задание завершено – карточка перемещается в следующую секцию.

–«Need testing». При попадании задания в этот столбец, происходит процесс тестирования. Если тестирование прошло успешно, карточка перемещается в секцию «Done». Если возникли какие-либо ошибки, к карточке прикрепляется комментарий с описанием проблемы и возвращается обратно в столбец «To do».

Главный показатель эффективности в Kanban – это среднее время прохождения задачи по доске. Задача прошла быстро – команда работала продуктивно и слаженно. Задача затянулась – надо анализировать: на каком этапе и почему возникли задержки и чью работу надо оптимизировать.

Kanban и Scrum имеет много общих подходов, но в тоже время у них множество различий:

–Методология Scrum жестко регламентирует по времени процесс разработки. Это заставляет команду работать упорно, но эффективно, соблюдая сроки. Каждый спринт заканчивается завершенным модулем, который можно показать клиенту. Kanban не имеет спринтов. Таким образом, в Kanban сложнее контролировать время разработки и прогнозировать завершение какого-либо модуля.

–После начала спринта Scrum методология не допускает изменений в бэклог (заданиях), так как это ломает основу всей системы. Kanban дает возможность добавлять или удалять задания на любой фазе веб-разработки.

–Scrum требует дополнительные роли/членов команды (Scrum-мастер, владелец продукта) для управления всем процессом разработки. В то время Kanban не требует таких ресурсов, так как процесс линейный и более прост в организации.

–Scrum требует время на встречи для организации спринта, ежедневные встречи, отчеты. Kanban не требует обязательных митингов. Они могут проводиться раз в неделю или раз в месяц.

Выводы. Необходимо отметить, что Scrum – гибкая методика разработки, Kanban – еще более гибкая. Если это разработка нового продукта, то на старте разработки и до релиза лучше использовать Scrum, так как он делает разработку более контролируемой по срокам. Также в Scrum много коммуникаций в команде: ребята обсуждают весь бэклог спринта перед стартом, задают вопросы авторам задачи (UX-дизайнерам, менеджерам продукта, бизнес-аналитикам), оценивают задачи сообща. Scrum помогает детально погрузить команду в суть продукта.

После релиза продукта начинается совсем другая история: начинает идти обратная связь от пользователей продукта, где необходимо быстро реагировать. Необходимо измерять и оптимизировать метрики продукта. Все увеличивает цикл разработки, начинается выполнение большого количества маленьких задач в непредсказуемой последовательности. И для этого как раз идеально подходит Kanban.

Таким образом, Kanban и Scrum имеют свои особенности. В некоторых случаях необходимо использовать Scrum, в других – Kanban. Различия в подходах отражены в Таблице 1.

Таблица 1 – Итоговая таблица сравнительного анализа методов разработки программного продукта

	Scrum	Kanban
Этап разработки	Со старта проекта до релиза продукта	После релиза продукта
Длительность проекта	От 3 месяцев	1-3 месяца
Наличие требований	Собраны перед началом разработки	Перед началом работы есть только основные требования, остальные задания формируются в процессе разработки
Наличие спецификации	Полная спецификация	Нет четкой спецификации
Затраты на планирование	Затрачивается время на планирование спринтов, составление плана работ	План создается и корректируется в процессе разработки
Контроль времени	Каждый спринт регламентируется по времени, сдается заказчику	Нет спринтов, задачи сдаются заказчику по мере их выполнения
Изменения в бэклог	Не допускаются после запуска проекта	Допускаются
Дополнительные роли в команде	Scrum-мастер, владелец продукта	Линейный процесс, не требующий трудовых ресурсов
Проведение митингов	Время на встречи для организации спринта, ежедневные встречи, отчеты	Не требует обязательных митингов

Список использованных источников

- [1]. Джефф Сазерленд. Scrum. Революционный метод управления проектами // Проекты гибкой разработки –2018. – С.10-15.
- [2]. Борис Вольфсон. Гибкие методики разработки // Agile – подход гибкой разработки. – 2015. – С.23-24.
- [3]. Зеленкова Л.В. Инновационная деятельность и инновационные технологии в агропромышленном комплексе // Сб.: Молодежная наука – гарант инновационного развития агропромышленного комплекса: материалы X Всероссийской (национальной) научно-практической конференции студентов, аспирантов, молодых ученых. – Курск. Изд-во Курск. гос. с.-х. ак., 2017. – С. 102–106.

**USING SCRUM, KANBAN IN FLEXIBLE SOFTWARE DEVELOPMENT
PROJECTS FOR VARIOUS FIELDS OF ORGANIZATIONS**

Y.N. Metelskaya
Student of BSUIR

P.S. Shafranovich
Student of BSUIR

Belarusian State University of Informatics and Radioelectronics, Republic of Belarus
E-mail: metelskaya2808@gmail.com, polina.638@bk.ru

Abstract: Innovations in the agricultural sector represent the achievements of science and technology necessary to increase labor productivity, increasing productivity, and the existence of all sectors of agriculture. It is necessary to choose a flexible software development methodology and methodology development in projects for the agro-industrial complex.

Keywords: Flexible methodology, Scrum, Kanban, software development, sprint.