

ИСПОЛЬЗОВАНИЕ МЕТОДОЛОГИИ SCRUM В ПРОЦЕССЕ ОБУЧЕНИЯ

А.А. Турейко, Е.Н. Унучек

Белорусский государственный университет информатики и радиоэлектроники, Минск, Беларусь, kanc@bsuir.by

Abstract. This article shows how Scrum methodology of software development was used in the process of training developers in IT companies. It proposes how this experience can be applied and adapted to the education in the universities. The main goal is to describe the new approach to training the students of IT-specialties and prove its effectiveness.

В последнее время гибкие методологии разработки информационных систем активно применяются на практике в большинстве компаний, занимающихся разработкой программного обеспечения (ПО). Более 80% проектов выполняются по методологии Scrum, входящей в группу гибких методологий разработки ПО.

Часто IT-компании принимают молодых специалистов без опыта работы на курсы по программированию с возможностью дальнейшего трудоустройства. И, как правило, программа курсов включает следующие этапы [1]:

1. Обучение, включающее теоретическую и практическую подготовку (в группе).
2. Выполнение контрольного проекта (самостоятельно).

По результатам выполнения контрольного проекта с учетом активности обучающихся (в дальнейшем, стажёров) на протяжении всего периода проведения курсов принимается решение о принятии на работу на испытательный срок.

Однако при использовании данного подхода к подготовке и отбору начинающих разработчиков сложно определить, обладает ли стажёр навыками командной работы и, как следствие, не возникнут ли у него сложности при необходимости работать в команде.

С целью минимизации данного риска предлагается модифицировать второй этап обучения, а именно заменить самостоятельное выполнение контрольного проекта на командное. Суть контрольного проекта заключается в том, что формируется несколько команд стажёров. Каждая команда должна разработать программный продукт, используя в команде методологию разработки Scrum, которая предполагает, что успех проекта основан на командной работе и тесном сотрудничестве.

Scrum – это набор принципов, на которых строится процесс разработки, позволяющий в жёстко фиксированные небольшие промежутки времени предоставлять конечному пользователю работающее ПО с добавленными возможностями, для которых определён наибольший приоритет [2].

Базовые принципы методологии Scrum [3]:

1. Участники разработки выполняют одну из трёх определяемых методологией ролей: владелец продукта (product owner), руководитель команды разработчиков (Scrum master) и члены команды (team).
2. Процесс разработки разбивается на итерации или спринты (небольшие промежутки времени продолжительностью от 2 до 4 недель).
3. Результатом спринта является готовый продукт, который можно передавать заказчику.
4. Команда сама решает, кто и как будет выполнять те или иные действия для достижения поставленных целей (принцип самоорганизации).
5. Оптимальное количество людей в команде должно быть 5 – 9 человек.

6. Наличие единого приоритезированного списка требований к продукту (backlog, бэклог), который постоянно пересматривается и дополняется в процессе разработки.

7. Планирование спринта или определение фиксированного объёма задач на ближайшую итерацию.

8. Проведение ежедневных митингов для того, чтобы все члены команды знали, кто и чем занимается в проекте.

9. Проведение обзора и ретроспективы спринта, в ходе которых команда демонстрирует заказчику разработанный продукт, а также обсуждает поставленные задачи, препятствия, принятые решения, нерешённые проблемы.

10. Прозрачность и визуализация проектных данных.

На рисунке 1 представлена схема процесса разработки ПО по методологии Scrum:

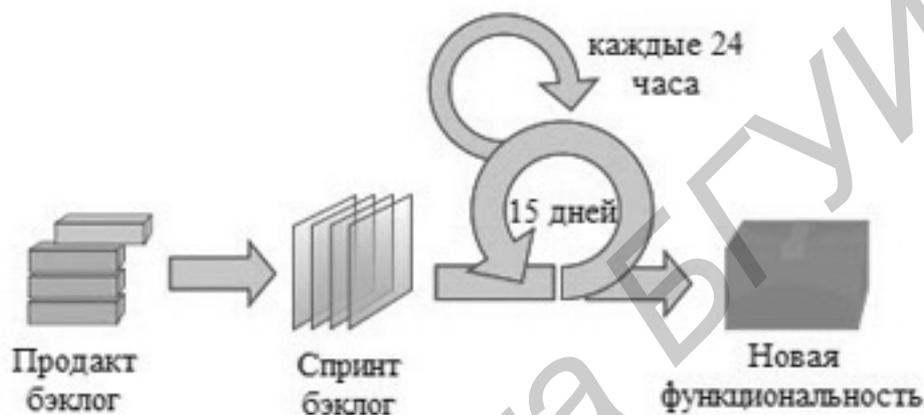


Рисунок 1 – Процесс разработки ПО по методологии Scrum

Для использования данной методологии в процессе разработки контрольного проекта, её необходимо адаптировать следующим образом: роль владельца продукта, который передаёт свои требования стажёрам-разработчикам, должен выполнять сотрудник фирмы. Продолжительность учебного процесса должна составить не меньше месяца (2 спринта по 2 недели).

Учитывая тот факт, что самоорганизация команды, во-первых, не происходит сама по себе, а во-вторых, требует времени, эксперты в данной области предварительно должны обучить стажёров основным принципам Scrum, а также следить в дальнейшем за организацией работы в командах.

По окончании работы над контрольным проектом у стажёра должны быть сформированы навыки работы в команде. Также, благодаря использованию методологии Scrum в процессе обучения, стажёры должны иметь представление о реальном процессе разработки ПО, о ролях на проекте и их взаимосвязях.

Предлагается применить данный подход в высших учебных заведениях в процессе выполнения курсового проектирования студентами ИТ специальностей. Однако в этом случае следует учитывать некоторые особенности:

1. Преподаватель (или другое заинтересованное лицо) должен будет разработать требования к системе, а также предложить архитектурное решение.

2. Необходимо будет совместно определить временные ограничения по работе над проектом (например, по часу каждый день).

3. Общение и проведение митингов на проекте могут осуществляться с помощью следующих средств: электронная почта, средства мгновенной передачи данных (Skype,

Google Talk, ICQ и другие), web-площадки для проведения встреч и совещаний (DabbleBoard, Twiddla, Vyew, GoToMeeting, Webinar.ru).

4. Также важно установить формат проведения митингов (видео-конференции, аудио-конференции, чаты).

5. Для обеспечения возможности коллективной работы студентов над проектом предлагаются следующие методы работы:

– Каждый студент должен работать над определённой частью продукта, и выполнять только задачи, относящиеся к этой части продукта. Однако применимость данного подхода ограничена и возможна лишь тогда, когда в проекте имеются атомарные непересекающиеся компоненты. Но в этом случае есть вероятность, что трудоёмкость компонентов приложения будет различной, что затруднит объективное оценивание работы студентов.

– Организовать удалённый доступ всех студентов к проекту. Популярные на данный момент средства организации совместного доступа к данным (в том числе средства управления версиями): платные TFS, Sharepoint; бесплатные Teamer, WinCvs, Dropbox, Writeboard, Google Docs.

Главной проблемой здесь будет подбор оптимального средства, которое было бы удобно всей команде.

6. Следует отметить, что такая организация работы может оказаться неприемлемой для всех форм обучения. Кроме того, студентам придётся приложить дополнительные усилия по изучению инструментов, с помощью которых будет организовано взаимодействие, в рамках выполнения курсового проекта по scrum. Однако не обязательно навязывать такую форму работы для всех студентов. Достаточно выбрать группу или несколько групп из пяти человек, желающих работать по такой методологии, чтобы члены данной группы считались командой в рамках методологии Scrum.

К основным преимуществам использования данного подхода следует отнести:

1. Подготовку студентов к работе на реальных проектах, формирование навыков работы в команде, а также ознакомление их с одной из самых распространённых на данный момент методологией Scrum.

2. Возможность выполнения реальных задач по автоматизации деятельности образовательной организации, а также постоянной поддержке и обновлению действующего ПО в качестве заданий для курсовых проектов.

Недостатки подхода:

1. Большие временные затраты и объём работ по внедрению такого подхода для преподавателя.

2. Реализация в рамках курсовых проектов задач по автоматизации деятельности образовательной организации будет ограничиваться технологиями, которые по плану необходимо применять для конкретного курсового проекта.

Таким образом, внедрение методологии Scrum в процесс обучения при условии детальной разработки заданий и правильной организации процесса, может стать инновационным подходом в образовании, который уменьшит разрыв между уровнем академической подготовки и практическими требованиями к IT-специалистам.

Литература

1. Андрей Колесов. Академическая подготовка ИТ-специалистов. – Еженедельник PC Week/RE, 2004.
2. Хенрик Книберг. Scrum и XP: заметки с передовой. – Электронные данные, 2007.
3. Майк Кон. Scrum: гибкая разработка ПО. – Вильямс, 2011.