

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Марков Н.Г., Заворин А.С., Мирошниченко Е. А., Заворин А.С., Тайлашева Т.С., Петровская Т.С., Кудинов А.В., Юнак А.Л., Воронова Г.А. Аналитический обзор «Системная инженерия и задачи инженерной подготовки в ТПУ». – Томск: ТПУ, 2012. – 37 с.

МЕТОДОЛОГИЯ ПОДДЕРЖКИ ПОЗНАВАТЕЛЬНОЙ АКТИВНОСТИ СТУДЕНТОВ ПРИ РЕАЛИЗАЦИИ ТВОРЧЕСКИХ ПРОЕКТОВ

Ботыгин И.А.

Национальный исследовательский Томский политехнический университет

Россия, г. Томск, пр. Ленина, 30, 634050

E-mail: bia@tpu.ru

METHODOLOGY SUPPORTING COGNITIVE ACTIVITY IN STUDENTS CREATIVE PROJECTS

Botygin I.A.

National Research Tomsk Polytechnic University

Russia, Tomsk, Lenin str., 30, 634050

E-mail: bia@tpu.ru

***Annotation.** Scrum-described software development methodology small student teams within the discipline of "Creative project" in "Computer Science and Engineering." It is shown that the iterative technology flexible software development projects based on the principles of system analysis and design international standards, ideally support the structure and content of the set active practice of students that will be formed from them in the process of studying this discipline.*

***Аннотация.** Описывается Scrum-методология разработки программного обеспечения небольшими студенческими командами в рамках дисциплины «Творческий проект» по направлению «Информатика и вычислительная техника». Показано, что гибкая итеративная технология разработки программных проектов на основе принципов системного анализа и международных стандартов проектирования идеально поддерживают структуру и содержание набора активностей студентов (практик), которые будут сформированы у них в процессе изучения этой дисциплины.*

В образовательной программе подготовки бакалавров по направлению 230100 – «Информатика и вычислительная техника», начиная с плана приема 2012 года, введена учебная дисциплина «Творческий проект». Дисциплина изучается три семестра без аудиторных занятий. Специфика направления подготовки «Информатика и вычислительная техника» практически сразу predetermined форму, методику и способы проведения занятий по указанной дисциплине, в основе которых лежит методология разработки программного обеспечения сложных информационных систем. Безусловно, спектр методологий проектирования и разработки программного обеспечения очень обширный. Не претендуя на всеобъемлющий обзор и анализ подобных методологий, отметим, что здесь универсальных решений

нет и при реализации каждого крупного проекта команда разработчиков пользуется, как правило, некоторым симбиозом CASE-технологий, функционал которых учитывает требования и особенности конкретных автоматизируемых бизнес-процессов и помогает контролировать ход выполнения проекта.

В настоящей работе описывается использование при выполнении заданий творческих проектов одной из технологий гибкой разработки (Agile software development) программного обеспечения – Scrum-технологии [1-4]. Обобщенно командный подход к управлению проектами в Scrum-методологии можно представить бесконечным итеративным циклом работ по достижению требуемого функционала системы и ее модернизации (рис.).

Сразу отметим, что Scrum-методология идеально подходит для небольших по численности творческих коллективов. Временные творческие группы студентов априори не превышают 3-х – 5-ти человек, что является оптимальным количеством участников в Scrum-командах. Главным действующим лицом – заказчиком (Product Owner) в Scrum-команде в нашем случае является руководитель проекта. Он же выполняет и функции так называемого посредника (Scrum Master) между заказчиком и членами Scrum-команды. В учебных творческих проектах именно преподавателю приходится сглаживать отношения и между самими членами команды. Основная функция Product Owner – грамотно и профессионально формировать список требований по реализации программного продукта на текущую Scrum-итерацию (Sprint). Сложность подлежащего реализации функционала определяет и срок выпуска очередного релиза. Выдержать жесткий фиксированный интервал Sprint-итераций, допустим, равный одной неделе, в студенческих учебных творческих проектах не представляется возможным ввиду загруженности студентов учебными поручениями по аудиторным дисциплинам. По мнению автора, оценка длительности Scrum-итерации (planning poker) – чрезвычайно трудная задача. Необходимо обеспечить одновременно и напряженность, и комфорт в работе. Тем более, что материальных стимулов при выполнении учебных творческих проектов у студентов нет.



Рис. Итерационный цикл Scrum-методологии

Разработка программного кода и, что очень важно, его тестирование осуществляется членами scrum-команды самостоятельно. Самостоятельность – условная, так как стимулируется и поощряется помощь друг другу. Задачи творческих проектов, в нашем случае, формируются таким образом, что для их реализации достаточно информационно-вычислительных возможностей, имеющихся в личном пользовании студентов.

Чрезвычайно важным этапом в Scrum-методологии является так называемый Scrum-митинг, когда участники по истечении Sprint-периода устраивают коллективное обсуждение достигнутых результатов. Заметим, что Scrum-митинг – это не круглый стол конференции. Это скорее – рабочие диалоги, в ходе которых обсуждаются не только итоги и возникшие новые проблемы, но и формируются авторские предложения (Product Backlog) на очередную Scrum-итерацию.

Демонстрация результатов спринта – это не мультимедийная презентация разработанного программного продукта. Демонстрация результатов спринта – это средства внешнего пользовательского и внутреннего служебного интерфейса, обеспечивающего взаимодействие с разрабатываемой информационной системой.

Таким образом, если кратко, Scrum – это практики активного рабочего общения студентов в процессе разработки программного обеспечения, в которых документирование и формальное управление минимизированы (сведены к нулю). При этом, повышается не только личный профессионализм участников команды, но и нарабатывается умение работать в команде с использованием современных гибких методологий проектирования и разработки программного обеспечения информационных систем.

СПИСОК ЛИТЕРАТУРЫ

1. Create software that delivers more value // Scrum.org. 2014. URL: <https://www.scrum.org/> (дата обращения: 20.02.2014).
2. What is Scrum? // Scrum Alliance, Inc. 2014. URL: <http://www.scrumalliance.org/> (дата обращения: 20.02.2014).
3. Agile Russia // AgileRussia [2007-2013]. URL: <http://agilerussia.ru/> (дата обращения: 20.02.2014).
4. Scrum // Mountain Goat Software. [1998-2012]. URL: <http://www.mountaingoatsoftware.com/agile/scrum> (дата обращения: 20.02.2014).