

СИСТЕМА КОНТРОЛЯ РАБОЧЕГО ПРОЦЕССА ПО МЕТОДОЛОГИИ SCRUM

ВВЕДЕНИЕ

Среди общераспространенных проблем процесса разработки программного обеспечения встречаются следующие:

- изменение требований непосредственно в процессе разработки;
- нечеткое распределение ответственности за выполняемую работу и ее результат;
- наличие непрерывного потока мелких, «быстрых», наваливающихся требований, отвлекающих разработчиков и менеджеров от основного направления работ.

Для решения задачи успешной организации процесса разработки ПО была создана гибкая методология разработки ПО.

I. ГИБКАЯ МЕТОДОЛОГИЯ РАЗРАБОТКИ

Гибкая методология разработки (англ. Agile software development) – это набор принципов и правил, в рамках которого осуществляется разработка ПО.

Большинство гибких методологий нацелено на минимизацию рисков путём сведения разработки к серии коротких циклов, называемых итерациями, которые обычно длятся одну-две недели. Каждая итерация сама по себе выглядит как программный проект в миниатюре и включает все задачи, необходимые для выдачи мини-прироста по функциональности: планирование, анализ требований, проектирование, кодирование, тестирование и документирование. Хотя отдельная итерация, как правило, недостаточна для выпуска новой версии продукта, подразумевается, что гибкий программный проект готов к выпуску в конце каждой итерации. По окончании каждой итерации команда выполняет переоценку приоритетов разработки.

Agile-методы делают упор на непосредственное общение лицом к лицу. Большинство agile-команд расположены в одном офисе, иногда называемом *bullpen*. Как минимум она включает и «заказчиков» (англ. *product owner*). Офис может также включать тестировщиков, дизайнеров интерфейса, технических писателей и менеджеров. Основным результатом работы по agile-методологии является работающий программный продукт. Расценивая именно работающий

программный продукт в качестве единственного показателя работы команды проекта за конечный период времени, создатели концепции agile сформулировали следующие ценности и принципы методологии:

- личности и их взаимодействия важнее, чем процессы и инструменты;
- работающее программное обеспечение важнее, чем полная документация;
- сотрудничество с заказчиком важнее, чем контрактные обязательства;
- реакция на изменения важнее, чем следование плану.

II. МЕТОДОЛОГИЯ РАЗРАБОТКИ SCRUM

Существуют методологии, которые придерживаются ценностей и принципов, заявленных в Agile Manifesto. Одной из наиболее распространенных является методология разработки Scrum, которую, пожалуй, можно считать набором конкретных практик, используемых в процессе разработки ПО. Термин Scrum ассимилирован в русский язык без перевода.

Scrum (в переводе с англ. «митинг», «свалка», «драка») чётко делает акцент на качественном контроле процесса разработки [1].

Scrum – это набор принципов, на которых строится процесс разработки, позволяющий в жёстко фиксированные небольшие промежутки времени (спринты от 2 до 4 недель) предоставлять конечному пользователю работающее ПО с добавленными возможностями, для которых определён наибольший приоритет. Требуемый к реализации функционал в очередном спринте определяется до его начала на этапе планирования и не может изменяться на всём протяжении спринта. При этом строго фиксированная небольшая длительность спринта придаёт процессу разработки предсказуемость и гибкость. Scrum является одним из наиболее общераспространенных «последователей» гибкой методологии разработки ПО.

1. Cohn, M. Succeeding with Agile: Software Development Using Scrum / M. Cohn // Addison-Wesley Professional. – 2009. – 504 с.

Бобров Денис Игоревич, магистрант кафедры информационных технологий автоматизированных систем БГУИР, bobrov.denis.at@gmail.com.

Научный руководитель: Ломако Александр Викторович, декан факультета заочного обучения БГУИР, кандидат технических наук, доцент, lavlot@bsuir.by.