

## **РАЗРАБОТКА НОВОГО ПОДХОДА К ВЕДЕНИЮ ПРОЕКТА ОСНОВАННОГО НА КОМБИНАЦИИ РАЗЛИЧНЫХ МЕТОДОЛОГИЙ**

**Брюханов О.В.,**

**научный руководитель канд. пед. наук, доцент Виденин С.А.**

*Сибирский федеральный университет*

Современные компании ведущие IT аутсорсинговую деятельность при ведении проектов сталкиваются с такими проблемами как плохое взаимодействие с заказчиком, медленная обратная связь, частое изменение требований к продукту и изменение сроков выполнения задач, сроков сдачи проекта. При этом методологии ведения проектов, которые используют в компаниях, не позволяют быстро и гибко реагировать на изменения технического задания, в частности такая методология как экстремальное программирование не подходит из-за отсутствия постоянного общения с заказчиком. В связи с этим в работе аутсорсинговой компании возникла необходимость разработки нового подхода к ведению проекта основанного на комбинации различных методологий, которая будет включать в себя необходимые, в данных условиях, качества и правила.

Рассмотрим существующие фундаментальные методологии ведения IT-проектов, которые стали основой для разработанной нами комбинаторной вариации методик – Scrum и Kanban, для компании работающей в условиях IT аутсорсинга.

Kanban - система организации производства и снабжения, позволяющая реализовать принцип «точно в срок». Система Kanban была разработана и впервые в мире реализована фирмой «Toyota». В 1959 году эта фирма начала эксперименты с системой Kanban и в 1962 году запустила процесс перевода всего производства на этот принцип. Появление термина Kanban связано с перечислением стандартных операций: мастера участков перечисляли выполняемые работы на бумаге и вывешивали их на видном месте рядом с такими же списками мастеров других участков. Постоянное использование философии «точно в срок» позволяет раскрыть не обнаруженные до сих пор дефекты. Так как запасы продукции и деталей могут скрывать проблемы на производстве, то при их уменьшении ежедневный контроль выявит, к примеру, неисправности или простои.

В методологии Kanban классически используется три правила:

### 1. Визуализируйте производство

— Разделение работы на задачи, каждую задачу можно оформлять на карточке и помещать на стену или доску.

— Использование именованных столбцов, чтобы показать положение задачи в производстве.

2. Ограничение WIP (work in progress или работу, выполняемую одновременно) на каждом этапе производства.

3. Измерение время цикла (среднее время на выполнение одной задачи) и оптимизирование процесса, чтобы уменьшить это время.

Еще одна из современных методологий это Scrum - методология управления проектами, активно применяющаяся при разработке информационных систем, а так же для гибкой разработки программного обеспечения. Scrum это набор принципов, на которых строится процесс разработки, позволяющий в жёстко фиксированные и небольшие по времени итерации, называемые спринтами, предоставлять конечному пользователю работающее ПО с новыми возможностями, для которых определён наибольший приоритет. Возможности ПО к реализации в очередном спринте

определяются в начале спринта на этапе планирования и не могут изменяться на всём его протяжении. При этом строго фиксированная небольшая длительность спринта придаёт процессу разработки предсказуемость и гибкость.

В методологии Scrum всего три роли:

– Scrum Master

Scrum Master - самая важная роль в методологии. Scrum Master отвечает за успех Scrum в проекте. Scrum Master является интерфейсом между менеджментом и командой.

– Product Owner

Product Owner - это человек, отвечающий за разработку продукта. Как правило, это product manager для продуктовой разработки, менеджер проекта для внутренней разработки и представитель заказчика для заказной разработки.

– Team

Команда в Scrum кросс-функциональна. В нее входят люди с различными навыками: разработчики, аналитики, тестировщики. Нет заранее определенных и поделенных ролей в команде, ограничивающих область действий членов команды. Размер команды ограничивается размером группы людей, способных эффективно взаимодействовать лицом к лицу.

В Scrum итерация называется Sprint. Каждый спринт представляет собой маленький "водопад"- модель процесса разработки программного обеспечения в которой процесс разработки выглядит как поток, последовательно проходящий фазы анализа требований, проектирования, реализации, тестирования, интеграции и поддержки. В течение спринта делаются все работы по сбору требований, дизайну, кодированию и тестированию продукта. Его длительность составляет 1 месяц. Результатом Sprint является готовый продукт, который можно передавать заказчику (по крайней мере, система должна быть готова к показу заказчику). Короткие спринты обеспечивают быструю обратную связь (feedback) проектной команды с заказчиком. Заказчик получает возможность гибко управлять разработкой системы, оценивая результат спринта и предлагая улучшения к созданной функциональности. Такие улучшения попадают в очередь на разработку, приоритизируются наравне с прочими требованиями и могут быть запланированы на следующий (или на один из следующих) спринтов.

Для того чтобы определить какая из методологий больше подходит для компании работающей в условиях IT аутсорсинга было проведено сравнение методологий.

Scrum и Kanban имеют сходства:

1. Оба являются и Lean (Lean - концепция управления, основанная на постоянном стремлении к устранению всех видов потерь) и Agile (Agile это гибкая методология разработки, суть которой состоит в получении оперативной обратной связи и как следствие — безболезненной реакции на изменения в проекте (приоритетах, списке работ, в новых идеях заказчика))

2. Оба используют вытягивающие системы планирования задач - организация потоков, когда один потребитель последовательно вытягивает поставки, осуществляемые предыдущими звеньями поставщиков, включенными в общую цепочку.

3. Оба ограничивают количество незавершенных работ- количество одновременно выполняемых задач ограничены и пока одна из задач не будет выполнена нельзя приступать к выполнению следующей.

4. Оба используют прозрачность для обеспечения улучшения процесса.

5. Оба полагаются на самоорганизующиеся команды.

6. Оба требуют дробления задач на более мелкие.

7. В обоих случаях план релиза постоянно оптимизируется на основе эмпирических данных (производительности/ времени выполнения задачи).

Различия:

Таблица 1 - Таблица различий

Scrum	Kanban
Обязательны ограничения по времени.	Не подразумевает ограниченные по времени итерации.
Команда обязуется выполнить конкретный объем работы за текущую итерацию.	Обязательства по срокам выполнения задач опциональны.
Кросс-функциональные команды являются обязательными. Кросс-функциональность- это способность выполнения командой(работником) разноплановых задач.	Могут допускаться узкопрофильные команды вместе с кросс-функциональными.
Задачи должны разбиваться на более мелкие так, чтобы они были завершены в течении одного спринта.	Нет каких-либо определенных размеров задач(выполнение некоторых задач может очень сильно превышать время выполнения остальных задач).
Построение диаграмм сгорания задач(burndown диаграмм) является обязательным. Burndown диаграммы показывают количество сделанной и оставшейся работы.	Построение диаграмм необязательно.
Незавершенная работа ограничивается длиной спринта.	Незавершенная работа ограничивается явно. Как показывает практика, оптимальное количество задач равно четырем.
Нельзя добавлять новые задачи в текущую итерацию.	Позволяет добавлять задачи когда это возможно.

На основании сходств и различий, которые описаны в таблице 1, было сделано предположение, которое на данный момент завершает апробацию, что Scrum сложно работает в условиях, когда неясны требования даже на начале итерации, слишком часто возникают неожиданные задачи, требуется более быстрая скорость реакции, чем в спринте, когда нужна узкая специализация людей в команде. Kanban лишен минусов Scrum но имеет сложность обеспечения высокой согласованности между стадиями производства продукции, имеет «плавающую» оценку сроков разработки и бюджета, а также постоянно изменяющиеся параллельно корректировке требования.

Беря во внимание опыт прошлых проектов компании «АстроСофт» и на основании анализа было сделано предложение об опробовании комбинации Scrum+Kanban+Scrum на одном из проектов, где Scrum применяется на начальном и завершающем этапе, а Kanban применяется на этапе тестирования проекта. На сегодняшний момент проект завершил этап Scrum+Kanban и переведен обратно в стадию Scrum, так как заказчик проанализировал ИС и указал на недочеты, которые требуется исправить и добавил задачи по реализации нового функционала.

На основании эксперимента был сделан вывод, что применение "чистого" Scrum или Kanban в компании ведущей ИТ аутсорсинг является неэффективным решением т.к. эти методологии не могут эффективно работать в условиях частого изменения технического задания, изменения сроков выполнения задач, изменения сроков сдачи проекта, плохого взаимодействия с заказчиком и медленной обратной связи.

Проводимый нами эксперимент подтвердил наши предположения об эффективности комбинирования методологий Scrum и Kanban, объединения их плюсов: Kanban эффективно работает при плохом взаимодействии с заказчиком, Scrum благодаря спринтам позволяет четко определять сроки выполнения задач и более эффективное использование человеческих ресурсов. Объединение Scrum и Kanban позволит значительно сократить риск получения продукта, который будет не устраивать заказчика, а также срыв сроков сдачи проекта и увеличение стоимости.

#### СПИСОК ЛИТЕРАТУРЫ

1. The Productivity Press Development Team, Kanban for the Shopfloor : перевод с английского/ Болтрукевич В., Попеско И., Изд-во Москва, Институт комплексных стратегических исследований, 2007 - 126 с.

2. Книберг Х., Скарин М., Kanban и Scrum: выжимаем максимум: перевод выполнен сообществом Agile Ukraine, Изд-во InfoQ.com, 2010 - 78 с.