

ОБЗОР МЕТОДОЛОГИИ SCRUM

Абуова К.С., Варламов О.С.

Евразийский национальный университет им. Л.Н.Гумилева, Астана

Научный руководитель – Кинтонова А.Ж.

Одной из актуальных задач в сфере разработки программных продуктов является завершение проекта в планированные сроки. Самое оптимальное решение этой проблемы – это командная работа, при которой проект разбивается на модули и распределяется между членами команды - разработчиками. Важным фактором является целостность всего проекта и контроля его реализацией. Например, увольнение одного из разработчиков приводит к потере целых фрагментов проекта[1].

Другой не менее важный вопрос – наличие методики реализации проекта. Современные средства разработки позволяют решить эту проблему. Для успешного проекта необходимо постоянное согласование действий. Можно использовать почту, различные средства связи, но эффективнее использовать специальный инструментарий.

Scrum - методология управления разработкой информационных систем для гибкой разработки программного обеспечения. Scrum — это набор принципов, на которых строится процесс разработки, позволяющий в жёстко фиксированные небольшие промежутки времени (от 2 до 4 недель) предоставлять конечному пользователю работающее ПО с новыми возможностями, для которых определён наибольший приоритет. Scrum чётко делает акцент на качественном контроле процесса разработки. Одна из причин ее популярности – простота [2].

Общими особенностями гибких методологий являются:

- Ориентированность на людей – как разработчиков, так и заказчиков. Считается, что умение собрать в проектной команде «правильных» людей определяет успех или неудачу проекта в значительно большей степени, чем любые процессы или технологии.

- Использование устных обсуждений вместо формальных спецификаций везде, где это возможно. Обсуждения должны быть главным способом коммуникации как с заказчиком, так и внутри проектной команды.

- Итеративная разработка с возможно более короткой (в разумных пределах) продолжительностью итерации, при этом в результате каждой итерации выпускается полноценная работающая версия продукта.

- Ожидание изменений – в гибком процессе проектная команда не пытается зафиксировать требования в начале проекта и затем следовать жестко определенному плану. Изменения могут быть сделаны на сколь угодно позднем этапе проекта[3].

Принципиальное отличие методологии Scrum от других методологий командной работы над проектом – это независимое формирование работы каждого члена команды в процессе разработки.

Роли

В методологии Scrum всего три роли:

1. Scrum Master
2. Product Owner
3. Team.

1 Скрам Мастер (Scrum Master) - самая важная роль в методологии. Скрам Мастер отвечает за успех Scrum в проекте. По сути, Скрам Мастер является интерфейсом между менеджментом и командой. Как правило, эту роль в проекте играет менеджер проекта или тимлид. Важно подчеркнуть, что Скрам Мастер не раздает задачи членам команды. Команда является самоорганизующейся и самоуправляемой.

Основные обязанности Скрам Мастера таковы:

- Создает атмосферу доверия
- Участвует в митингах в качестве фасилитатора
- Устраняет препятствия
- Делает проблемы и открытые вопросы видимыми
- Отвечает за соблюдение практик и процесса в команде

Скрам Мастер ведет DailyScrumMeeting и отслеживает прогресс команды при помощи SprintBacklog, отмечая статус всех задач в спринте. ScrumMaster может также помогать ProductOwner создавать Backlog для команды.

2 *ProductOwner* - это человек, отвечающий за разработку продукта. Как правило, это productmanager для продуктовой разработки, менеджер проекта для внутренней разработки и представитель заказчика для заказной разработки. ProductOwner - это единая точка принятия окончательных решений для команды в проекте, именно поэтому это всегда один человек, а не группа или комитет.

Обязанности ProductOwner таковы:

- Отвечает за формирование productvision
- Управляет ROI
- Управляет ожиданиями заказчиков и всех заинтересованных лиц
- Координирует и приоритизирует Productbacklog
- Предоставляет понятные и тестируемые требования команде
- Взаимодействует с командой и заказчиком
- Отвечает за приемку кода в конце каждой итерации

ProductOwner ставит задачи команде, но он не вправе ставить задачи конкретному члену проектной команды в течении спринта.

3 *Команда (Team)* берет на себя обязательства по выполнению объема работ на спринт перед ProductOwner. Работа команды оценивается как работа единой группы. В Scrum вклад отдельных членов проектной команды не оценивается, так как это разваливает самоорганизацию команды.

Обязанности команды таковы:

- Отвечает за оценку элементов баклога
- Принимает решение по дизайну и имплементации
- Разрабатывает софт и предоставляет его заказчику
- Отслеживает собственный прогресс (вместе со Скрам Мастером).
- Отвечает за результат перед ProductOwner

Размер команды ограничивается размером группы людей, способных эффективно взаимодействовать лицом к лицу. Типичные размер команды - 7 плюс минус 2.

Команда в Scrum кроссфункциональна. В нее входят люди с различными навыками - разработчики, аналитики, тестировщики. Нет заранее определенных и поделенных ролей в команде, ограничивающих область действий членов команды. Команда состоит из инженеров, которые вносят свой вклад в общий успех проекта в соответствии со своими способностями и проектной необходимостью. Команда самоорганизуется для выполнения конкретных задач в проекте, что позволяет ей гибко реагировать на любые возможные задачи.

Для облегчения коммуникаций команда должна находиться в одном месте (colocated). Предпочтительно размещать команду не в кубиках, а в одной общей комнате для того, чтобы уменьшить препятствия для свободного общения. Команде необходимо предоставить все необходимое для комфортной работы, обеспечить досками и флипчартами, предоставить все необходимые инструменты и среду для работы.

Артефакты:

- *ProductBacklog* - это приоритезированный список имеющихся на данный момент бизнес-требований и технических требований к системе. Product Backlog включает в себя use cases, defects, enhancements, technologies, stories, features, issues, и т.д. Productbacklog

также включает задачи, важные для команды, например "провести тренинг", "добить всем памяти". ProductBacklog постоянно пересматривается и дополняется - в него включаются новые требования, удаляются ненужные, пересматриваются приоритеты. За ProductBacklog отвечает ProductOwner. Он также работает совместно с командой для того, чтобы получить приближенную оценку на выполнение элементов ProductBacklog для того, чтобы более точно расставлять приоритеты в соответствии с необходимым временем на выполнение.

- *SprintBacklog* содержит функциональность, выбранную Product Owner из Product Backlog. Все функции разбиты по задачам, каждая из которых оценивается командой. Каждый день команда оценивает объем работы, который нужно проделать для завершения задач.

Сумма оценок оставшейся работы может быть построена как график зависимости от времени. Такой график называется SprintBurndownchart. Он демонстрирует прогресс команды по ходу спринта.

- *Sprint*. В Scrum итерация называется Sprint. Ее длительность составляет 1 месяц (30 дней). Результатом Sprint является готовый продукт (build), который можно передавать (deliver) заказчику (по крайней мере, система должна быть готова к показу заказчику).

Короткие спринты обеспечивают быстрый feedback проектной команде от заказчика. Заказчик получает возможность гибко управлять score системы, оценивая результат спринта и предлагая улучшения к созданной функциональности. Такие улучшения попадают в ProductBacklog, приоритизируются наравне с прочими требованиями и могут быть запланированы на следующий (или на один из следующих) спринтов.

Каждый спринт представляет собой маленький "водопад". В течение спринта делаются все работы по сбору требований, дизайну, кодированию и тестированию продукта. Score спринта должен быть фиксированным. Это позволяет команде давать обязательства на тот объем работ, который должен быть сделан в спринте. Это означает, что SprintBacklog не может быть изменен никем, кроме команды.

Жизненный цикл спринта:

В начале каждого спринта проводится планирование спринта. В планировании спринта участвуют заказчики, пользователи, менеджмент, ProductOwner, Скрам Мастер и команда.

Планирование спринта состоит из двух последовательных митингов:

- *Планирование спринта, митинг первый*

Участники: команда, ProductOwner, ScrumMaster, пользователи, менеджмент.

Цель: Определить цель спринта (SprintGoal) и SprintBacklog - функциональность, которая будет разработана в течение следующего спринта для достижения цели спринта.

Артефакт: SprintBacklog

- *Планирование спринта, митинг второй*

Участники: Скрам Мастер(ScrumMaster), команда (Team)

Цель: определить, как именно будет разрабатываться определенная функциональность для того, чтобы достичь цели спринта. Для каждого элемента SprintBacklog определяется список задач и оценивается их продолжительность.

Если в ходе спринта выясняется, что команда не может успеть сделать запланированное на спринт, то Скрам Мастер, ProductOwner и команда встречаются и выясняют, как можно сократить score работ и при этом достичь цели спринта.

Остановка спринта (Sprint Abnormal Termination)

Остановка спринта производится в исключительных ситуациях. Спринт может быть остановлен до того, как закончатся отведенные 30 дней. Спринт может остановить команда, если понимает, что не может достичь цели спринта в отведенное время. Спринт может остановить ProductOwner, если необходимость в достижении цели спринта исчезла.

После остановки спринта проводится митинг с командой, где обсуждаются причины остановки спринта. После этого начинается новый спринт: производится его планирование и стартуются работы.

DailyScrumMeeting

Этот митинг проходит каждое утро в начале дня. Он предназначен для того, чтобы все члены команды знали, кто и чем занимается в проекте. Длительность этого митинга строго ограничена и не должна превышать 15 минут. Цель митинга - поделиться информацией. Он не предназначен для решения проблем в проекте. Все требующие специального обсуждения вопросы должны быть вынесены за пределы митинга. Скрам митинг проводит Скрам Мастер. Он по кругу задает вопросы каждому члену команды:

- Что сделано вчера?
- Что будет сделано сегодня?
- С какими проблемами столкнулся?

Скрам Мастер собирает все открытые для обсуждения вопросы в виде ActionItems, например в формате что/кто/когда.

Команда демонстрирует инкремент продукта, созданный за последний спринт. ProductOwner, менеджмент, заказчики, пользователи, в свою очередь, его оценивают. Команда рассказывает о поставленных задачах, о том как они были решены, какие препятствия были у них на пути, какие были приняты решения, какие проблемы остались нерешенными. На основании ревью принимающая сторона может сделать выводы о том, как должна дальше развиваться система. Участники митинга делают выводы о том, как шел процесс в команде и предлагает решения по его улучшению.

Скрам Мастер отвечает за организацию и проведение этого митинга. Команда помогает ему составить адженду и распланировать кто и в какой последовательности что представляет. Подготовка к митингу не должна занимать у команды много времени (правило - не более двух часов). В частности, именно поэтому запрещается использовать презентации в PowerPoint. Подготовка к митингу не должна занимать у команды более 2-х часов [4].

Любой долгосрочный проект, связанный с разработкой программного обеспечения, разрастается до необъятных размеров, становясь трудно управляемым и трудно прогнозируемым. Описанная методология позволяет повысить эффективность командной работы в планировании, мониторинге и реализации проекта. Эта методология является актуальной в сфере разработки программных продуктов.

Литература

1. Сергеева И. Методология разработки сложных программных систем.
2. <http://ru.wikipedia.org>
3. Гибкие процессы разработки ПО., cdtgos.googlecode.com
4. Асхат Уразбаев. Обзор методологии Scrum., <http://agilerussia.ru/methodologies>