

Черномырдина Анна Сергеевна

студентка

ФГАОУ ВО «Самарский национальный исследовательский

университет им. академика С.П. Королева»

г. Самара, Самарская область

МЕТОДОЛОГИЯ AGILE SOFTWARE DEVELOPMENT

Аннотация: в данной статье описывается технология Agile software development, рассматриваются ее достоинства и недостатки, а также релевантность её использования.

Ключевые слова: компьютерный код, программирование, информатика, Agile software development.

При создании программного обеспечения всегда необходимо учитывать ряд факторов, таких как обеспечение хорошей защиты проекта, равновесие качества конечного продукта и скорости его создания, бюджетная сторона вопроса и различные риски, связанные с внедрением и поддержанием конечного продукта. Именно поэтому разработка программного обеспечения начинается с определения методологии проектирования продукта. В выборе методологии необходимо руководствоваться рядом аспектов: как данный подход зарекомендовал себя в широком использовании, насколько хорошо и быстро может выполняться работа, насколько сильно он подвержен рискам. Долгое время стандартной методологией являлась каскадная система «Водопад» (англ. waterfall), описанная ещё в конце 50-х годов. Но я хочу рассказать о более прогрессивной гибкой методологии разработки (англ. Agile software development).

Agile software development

Итак, приступим. Сам термин появился в начале 2000-х, когда в штате Юта был издан «Agile-манифест разработки программного обеспечения».

Данная методология состоит из циклических «спринтов», каждый из которых включает в себя планирование, разработку, тестирование, демонстрацию.

Заказчик должен постоянно принимать участие в проектах и иметь соответствующую квалификацию.

Спринт – это так называемый список ожиданий и требований заказчика. Очередь задач с четко заданным приоритетом (Backlog) поступает команде разработчиков. Чтобы избежать перегрузки команды, то есть, когда список реализуемых за конечное время фич превышает пропускную способность команды, что ведет к некачественной работе и перескакивания с задачи на задачу, используют различные стратегии планирования. Ежедневно проводятся встречи проектной команды Scrum, где контролируется ход работы, очищается и рационализируется Backlog.

Спринт имеет сравнительно небольшую продолжительность – от одной до четырех недель. В конце каждого спринта заказчик видит достигнутый результат, вносит правки и коррективы.

Agile-разработка следует принципу минимального жизнеспособного продукта. В начале разработки нет задачи сразу получить сложный законченный продукт. Первостепенной задачей является получение минимального жизнеспособного продукта, который можно показать пользователям, проверить, какие доработки необходимы, а какие функции излишни. Благодаря этому принципу детальность задания теряет свою необходимость.

Также к преимуществам можно отнести, что наращивание функционала происходит постепенно, что снижает риск крупной ошибки. Так же начальная версия продукта может быть запущена в максимально кратчайшие сроки. С финансовой точки зрения, цена продукта окажется ниже ввиду отсутствия рисков, всегда закладываемых подрядчиком.

Конечно, данная технология имеет ряд недостатков, так как не предполагает далеко идущих планов, а внесение заказчиком корректировок на каждом этапе может существенно противоречить архитектуре уже созданного продукта.

Однако, несмотря на эти недостатки, agile-разработка уже твердо стоит на ногах и очень популярна среди компаний-разработчиков. Потихоньку отводя на

второй план классическую waterfall-разработку, agile имеет широкие перспективы развития на современном рынке.

Таким образом, можно сделать вывод, Agile – очень интересная методика разработки, включающая в себя серую группу принципов и подходов для уже существующих и только разрабатывающихся методов. По моему мнению, гибкие методы agile применимы не только в ИТ сфере, но и к более широкому кругу задач. Её очень удобно использовать в задачах с неопределенностью конечного результата, критичной стоимостью и сроками разработки. Agile-разработка способна в кратчайшие сроки создавать качественные и работоспособные продукты, удовлетворяющие как заказчика, так и пользователей, причем эти продукты динамически развиваются в соответствии с теми или иными требованиями.

Список литературы

1. Agile Alliance Основополагающие принципы Agile-манифеста» [Электронный ресурс]. – Режим доступа: <http://agilemanifesto.org/iso/ru/principles.html>
2. Agile Alliance. What is Agile Software Development? [Электронный ресурс]. – Режим доступа: <https://www.agilealliance.org/agile101/>