

## УНИФИЦИРОВАННЫЙ ПРОЦЕСС РАЗРАБОТКИ И ЭКСТРЕМАЛЬНОЕ ПРОГРАММИРОВАНИЕ

*БНТУ, Минск*

*Научный руководитель Дробыш А.А.*

Унифицированный процесс Rational (Rational Unified Process, RUP) и экстремальное программирование (Extreme Programming, XP). Оба они являются примерами итеративных процессов, но построены на основе различных предположений о природе разработки программного обеспечения и, соответственно, достаточно сильно отличаются.

RUP является примером так называемого «тяжелого» процесса, детально описанного и предполагающего поддержку собственно разработки исходного кода ПО большим количеством вспомогательных действий. Примерами подобных действий могут быть разработка планов, технических заданий и др. Основная цель такого процесса – отделить успешные практики разработки и сопровождения ПО от конкретных людей, умеющих их применять. Многочисленные вспомогательные действия дают надежду сделать возможным успешное решение задач по конструированию и поддержке сложных систем с помощью имеющихся работников, не обязательно являющихся суперпрофессионалами.

Для достижения этого выполняется иерархическое пошаговое детальное описание предпринимаемых в той или иной ситуации действий. В ходе проекта создается много промежуточных документов, позволяющих разработчикам последовательно разбивать стоящие перед ними задачи на более простые. Эти же документы служат для проверки правильности решений, принимаемых на каждом шаге, а также отслеживания общего хода работ и уточнения оценок ресурсов.

Экстремальное программирование, наоборот, представляет так называемые «живые» методы разработки, называемые также «легкими» процессами. Они заостряют внимание на использовании хороших разработчиков, а не хорошо отлаженных процессов разработки. Живые методы избегают фиксации четких схем

действий, чтобы обеспечить большую гибкость в каждом конкретном проекте, а также выступают против разработки дополнительных документов, которые не вносят непосредственного вклада в получение готовой работающей программы.

Унифицированный процесс Rational является довольно сложной, детально проработанной итеративной моделью жизненного цикла ПО. RUP основан на трех ключевых идеях:

- Весь ход работ направляется итоговыми целями проекта, выраженными в виде вариантов использования (use cases) – сценариев взаимодействия результирующей программной системы с пользователями или другими системами, при выполнении которых пользователи получают значимые для них результаты и услуги. Разработка начинается с выделения вариантов использования и на каждом шаге контролируется степенью приближения к их реализации.

- Основным решением, принимаемым в ходе проекта, является архитектура результирующей программной системы. Архитектура устанавливает набор компонентов, из которых будет построено ПО, ответственность каждого из компонентов (то есть решаемые им подзадачи в рамках общих задач системы), четко определяет интерфейсы, через которые они могут взаимодействовать, а также способы взаимодействия компонентов друг с другом.

Архитектура является одновременно основой для получения качественного ПО и базой для планирования работ и оценок проекта в терминах времени и ресурсов, необходимых для достижения желаемых результатов. Она оформляется в виде набора графических моделей на языке UML.

- Основой процесса разработки являются планируемые и управляемые итерации, объем которых определяется на основе архитектуры.

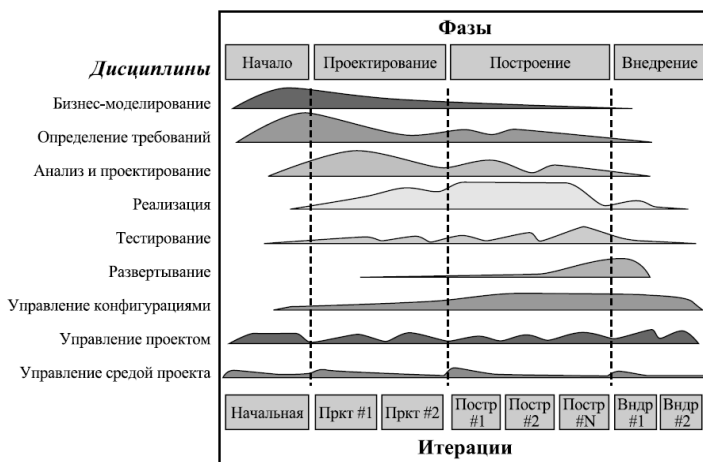
RUP выделяет в жизненном цикле 4 основные фазы, в рамках каждой из которых возможно проведение нескольких итераций. Кроме того, разработка системы может пройти через несколько циклов, включающих все 4 фазы.

1) Фаза начала проекта (Inserption). Основная цель этой фазы – достичь компромисса между всеми заинтересованными лицами относительно задач проекта и выделяемых на него ресурсов.

2) Фаза проектирования (Elaboration). Основная цель этой фазы – на базе основных требований разработать стабильную базовую архитектуру продукта, которая в дальнейшем используется как основа разработки системы.

3) Фаза построения (Construction). Основная цель этой фазы – детальное прояснение требований и разработка системы, удовлетворяющей им, на основе архитектуры, которая была спроектирована ранее.

4) Фаза внедрения (Transition). Цель этой фазы – сделать систему полностью доступной конечным пользователям. На этой стадии происходит развертывание системы в ее рабочей среде, бета-тестирование, подгонка мелких деталей под нужды пользователей.



### Работы в проектах

Наиболее важные с точки зрения RUP артефакты проекта – это модели, описывающие различные аспекты будущей системы. Большинство моделей представляют собой наборы диаграмм UML.

Основные используемые виды моделей: модель вариантов использования (use-case model); модель анализа (Analysis Model); модель проектирования (Design Model); модель реализации

(Implementation Model); модель развертывания (Deployment Model); модель тестирования (Test Model Или Test Suite); моделирование предметной области (Бизнес-Моделирование, Business Modeling); определение требований (Requirements); анализ и проектирование (Analysis And Design); реализация (Implementation); тестирование (Test); развертывание (Deployment); управление конфигурациями и изменениями (Configuration And Change Management); управление проектом (Project Management); управление средой проекта (Environment). Первые пять дисциплин считаются рабочими, а остальные – поддерживающими. Распределение объемов работ по дисциплинам в ходе проекта выглядит, согласно руководству по RUP (рисунок).

УДК 37.026.6

Воронцова А.Н.

## **АКТИВНАЯ ОЦЕНКА РЕЗУЛЬТАТОВ УЧЕБНОЙ ДЕЯТЕЛЬНОСТИ СТУДЕНТОВ**

*БНТУ, Минск*

*Научный руководитель Канашевич Т.Н.*

Активная оценка – это не метод выставления отметок, а методика, которая состоит из различных техник, помогающих преподавателю стимулировать учебную деятельность студентов. Эта методика создает благоприятную атмосферу для повышения качества знаний обучающихся, помогает проводить занятия педагогу так, чтобы студентам было интересно учиться. Использование педагогами элементов активной оценки на занятиях, способствует повышению эффективности обучения, что в настоящее время особенно актуально.

Основными особенностями активной оценки являются:

1. Планирование целей на «языке студента». Формулировка целей должна быть понятна обучающимся, они имеют право знать, зачем они пришли на занятие. По возможности, цели должны быть интересными и практикоориентированными.

Хорошо, если цели занятия на «языке студента» записаны на доске и преподаватель и студент держат их в центре своего внимания.