

УДК 004.421.2

ПРИМЕНЕНИЕ МЕТОДОЛОГИИ RATIONAL UNIFIED PROCESS ДЛЯ РАЗРАБОТКИ ИНТЕЛЛЕКТУАЛЬНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ АВТОМАТИЗИРОВАННОЙ ТОРГОВЛИ НА ФОНДОВОЙ БИРЖЕ

Щедрина И.В., Щедрин С.С., студент гр. ПИМ-151, I курс

Научный руководитель: Сарапулова Т.В., к.т.н., доцент

Кузбасский государственный технический университет имени Т.Ф. Горбачева
г. Кемерово

Методология – это система принципов, а также совокупность идей, понятий, методов, способов и средств, определяющих стиль разработки программного обеспечения. Также, это реализация стандарта, ведь сами стандарты лишь говорят о том, что должно быть, оставляя свободу выбора и адаптации. Именно она определяет, как будет выполняться разработка [1]. На данный момент разработано много успешных методологий создания программного обеспечения, а выбор конкретной зависит факторов, связанных со спецификой самого проекта, командой разработчиков. От того, верно ли будет выбрана методология разработки очень сильно зависит качество результирующего продукта, а попытки разработки без применения одной из них почти всегда заканчиваются сорванными сроками, неудовлетворенностью заказчика и другими неприятными ситуациями.

Rational Unified Process (RUP) – методология разработки программного обеспечения, созданная компанией Rational Software. Это одна из спиральных методологий разработки программного обеспечения. В качестве языка моделирования в общей базе знаний используется язык Unified Modelling Language (UML).

Данная методология была использована для разработки интеллектуальной информационной системы для автоматизированной торговли на фондовой бирже. Выбор методологии был обусловлен следующими факторами: изначально были четко определены требования к разрабатываемой системе, рабочую версию системы необходимо было предоставить в кратчайшие сроки.

Жизненный цикл разработки продукта состоит из четырех фаз, каждая из которых включает в себя одну или несколько итераций (рис. 1):

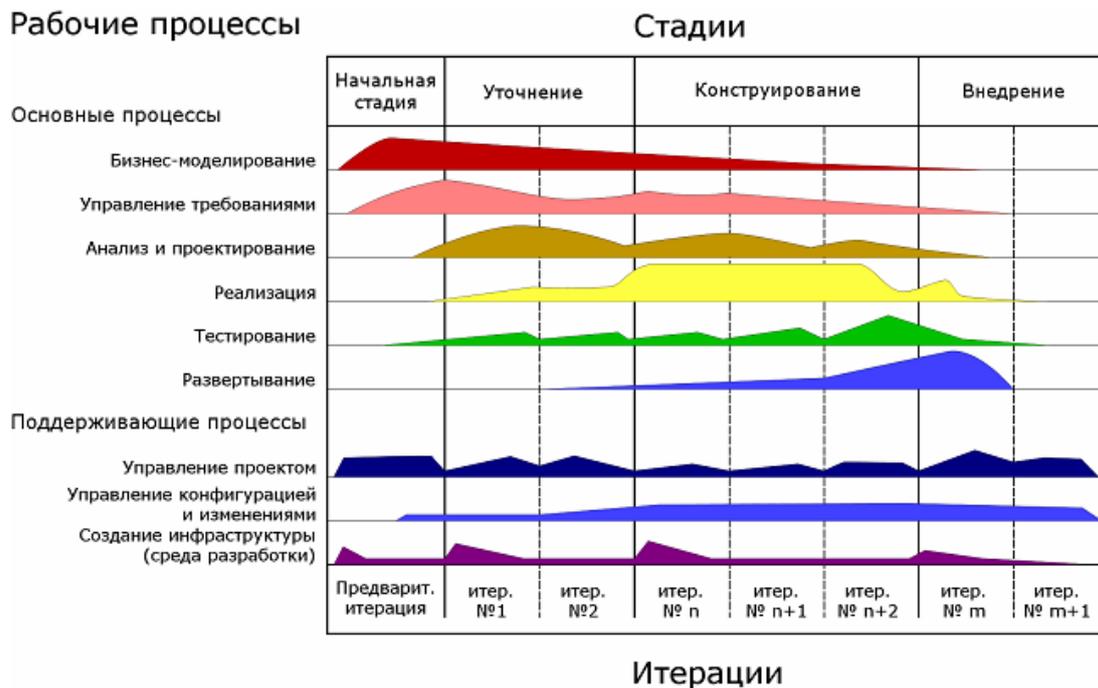


Рисунок 1 – Графическое представление процесса разработки по RUP
 Процесс разработки был разбит на четыре фазы:

1. Начало

На данном этапе были выполнены следующие задачи:

- определение требований к проекту;
- оценка рисков проекта;
- формирование плана разработки.

2. Проектирование

В рамках данной фазы были проведены работы по разработке и оценке архитектуры системы. Для облегчения итеративной разработки было решено использовать модульную архитектуру. Это позволило обеспечить расширяемость и гибкость системы. В итоге система состоит из гибкой библиотеки классов, двух настольных приложений на основе Windows Presentation Foundation, а также, консольного приложения для работы с системой без предоставления графического интерфейса.

3. Построение

Данная фаза была разбита нами на четыре основных вехи:

- альфа-версия (обучение с учителем) – для проверки эффективности алгоритма обучения;
- бета-версия (обучение без учителя) – для использования, в дальнейшем, данного функционального модуля для обучения сетей, которые мы будем использовать для анализа котировок на фондовой бирже;
- релиз-кандидат (возможность работы с несколькими ценными бумагами, оценка рисков) – для адаптации системы к работе в реальных условиях;
- финальная версия (интеграция с торговой платформой, автоматический режим работы) – для выхода в реальные условия.

Данный шаг был проделан для того, чтобы точнее оценивать текущий уровень готовности системы, а также, для разделения реализуемой функциональности на блоки.

Также, этот каждая веха была разбита на несколько итераций:

Разработка альфа-версии:

#1: Реализация общего каркаса системы

#2: Реализация создания нейронных сетей

#3: Реализация обучения нейронных сетей с учителем

Разработка бета-версии:

#4: Рефакторинг системы

#5: Реализация обучения нейронных сетей без учителя

#6: Разработка графического интерфейса

Разработка релиз-кандидата:

#7: Рефакторинг системы

#8: Реализация системы управления рисками

#9: Реализация торговли на нескольких ценных бумагах

Разработка финальной версии системы:

#10: Рефакторинг системы

#11: Интеграция с торговой платформой

4. Внедрение

В завершительной фазе необходимо выполнить тестирование системы на реальных данных, провести окончательную отладку и перейти в режим поддержки разработанной системы.

В результате использования данной методологии разработки получились следующие результаты:

- сформулирован план разработки системы;
- разработаны первые две рабочие версии системы;
- первая рабочая версия была разработана в кратчайшие сроки;
- проектные риски минимизированы;
- система получила модульную архитектуру для облегчения итерационной разработки.

Список литературы:

- 1) Методологии разработки программного обеспечения // Хабрахабр URL: <https://habrahabr.ru/sandbox/43802/>
- 2) Методологии разработки программного обеспечения // Компьютер Пресс URL: <http://compress.ru/article.aspx?id=9633>
- 3) Методология разработки программного обеспечения Rational Unified Process // DPGroup URL: <http://www.dpgrup.ru/methodology-rup.htm>