

2.3.5.

Р.М. Хамитов, Т.В. Шорина

ФГБОУ ВО "Казанский государственный энергетический университет"
институт цифровых технологий и экономики,
кафедра информационных технологий и интеллектуальные системы,
Казань, hamitov@gmail.com, shorina.t.v@mail.ru

АСПЕКТЫ ОБЕСПЕЧЕНИЯ КАЧЕСТВА ПРОГРАММНЫХ ПРОДУКТОВ

В статье приводится набор аспектов системы обеспечения качества программных продуктов: требования к качеству документации; качеству конструкторской документации; качество кода; качество тестов; правильная стратегия тестирования; качество методов тестирования; качество тестовых примеров; качество дефектов; качество развертывания; контроль рисков; организационное техническое обучение команды разработчиков.

Ключевые слова: программное обеспечение качества; аспекты; критерии; тестирование, инженер по обеспечению качества ПО.

Целью исследования являлось выделение и систематизация наиболее важных критериев обеспечения качества программных продуктов.

В среде разработки программного обеспечения пользователи, заказчики и менеджеры проектов больше всего заинтересованы в качестве программного продукта, в том числе и как важной составной части сложных технических решений. Тестировщиков же, больше заботит качество кода, они не заботятся о качестве самого программного продукта [1]. Их цель только нахождение ошибок. В условиях постоянных изменений программного продукта обеспечение стабильного качества возможно только при переходе от каскадной к гибким моделям разработки.

Имеющим непрерывный основной цикл: Анализ требований => Рабочий проект => Кодирование => Интеграция => Тестирование => Приемка.

Каждый из этих процессов должен иметь соответствующий рабочий процесс и документ, отражающий рабочий процесс. Персонал по обеспечению качества может сформулировать разумную систему стандартов качества, чтобы направлять и оптимизировать рабочий процесс на основе прошлого успешного опыта, а также оптимизировать систему качества на практике.

Как обеспечить качество программного обеспечения. Чтобы гарантировать качество программного обеспечения необходимо гарантировать качество следующих аспектов:

1 Требования к качеству документации:

– подробное описание требований: без упущений, без двусмысленности, без дублирования, без недостижимости, без огромных рисков;

– четкие индикаторы требований: четкие предпосылки и точные индикаторы данных.

2 Качество конструкторской документации:

– четко определенные поля: отсутствие путаницы с полями, четкие источники данных и тенденции;

– понятная и беспрепятственная логика: работа ПО не заходит в тупик, а сценарии использования всесторонне рассмотрены.

– простое и понятное взаимодействие: что представляет собой каждая кнопка до ее нажатия и что изменится после ее нажатия;

– предоставление множества вспомогательных документов: блок-схем, интеллектуальных карт, интерактивных проектных образцов прототипа и т. д.

3 Качество кода:

- высококачественный дизайн фреймворка: выбор подходящего технологического стека в соответствии с реальной ситуацией в проекте, с высокой отказоустойчивостью, надежностью, масштабируемостью, интерфейсной совместимостью и производительностью;
- четкая и качественная спецификация кодирования;
- анализ кода: разработчики сами или попарно проводят пошаговые руководства по коду, учитывая, завершена ли логика, выполняется ли обработка данных и оптимальна ли производительность алгоритма;
- рефакторинг кода: оценка возможностей для улучшения кода;
- модульное тестирование: разработчики сами пишут модульные тесты, для проверки, соответствует ли метод функции требованиям, и может ли исходная функция поддерживаться с очередной итерацией версии кода.

4 Качество тестов:

4.1 Правильная стратегия тестирования:

- внутреннее тестирование: тестирование интерфейса, стресс-тестирование, тестирование развертывания, тестирование безопасности;
- внешнее тестирование: тестирование пользовательского интерфейса, тестирование производительности, тестирование совместимости, тестирование безопасности.

4.2 Качество методов тестирования:

- разработка разумных методов тестирования;
- экспертиза методов тестирования;
- улучшение и оптимизация методов тестирования.

4.3 Качество тестовых примеров:

- Методы проектирования хороших вариантов использования: разделение классов эквивалентности, метод диаграмм причинно-следственной связи, метод анализа сценариев, метод ортогонального анализа, покрытие пути, логическое покрытие и т.п.;
- обзор и анализ сценариев использования: для обеспечения широты (интерфейс, база данных, журнал) и глубины (детальности тестовых точек) тестирования;
- сопровождение тестовых примеров: заполнение недостающих и проблемных вариантов использования в процессе тестирования, журналирование ошибок и добавление их в тестовые примеры [2].

4.4 Качество дефектов:

- классификация дефектов: общая четырехуровневая классификация;
- отчеты о дефектах: система управления ошибками, теория жизненного цикла ошибок;
- отражение дефекта: обзор ошибок, разработка, чтобы избежать подобных ошибок, тестирование рассматривает аналогичные сценарии при разработке тестовых случаев.

5 Качество развертывания:

Нельзя сказать, что после обеспечения качества проектирования, разработки и тестирования качество продукта будет хорошим после того, как он будет запущен в эксплуатацию. Поскольку существует целый набор процедур для развертывания и запуска, качество процесса развертывания напрямую определяет качество запуска продукта.

- среда развертывания: соответствие аппаратной и программной среды функциональным требованиям и производительности;
- время развертывания: выбор времени, которое меньше всего влияет на пользователей;
- параметры развертывания: параметры тестовой среды и параметры рабочей среды часто отличаются, что часто требует настройки параметров;
- автоматическое развертывание: процессно-ориентированное и автоматическое развертывание значительно сократит количество человеческих ошибок.

6 Контроль рисков

- риски проекта: изменение требований, требовательная к ресурсам среда разработки, нехватка персонала, технические риски, риски безопасности;
- прогнозирование рисков: в соответствии с реальной ситуацией в проекте и собственным

опытом разработки программного обеспечения найти слабые и неопределенные связи, которые часто представляют собой высокие риски;

– управление рисками: снижение отрицательного влияния рисков на продукт.

7 Организационное техническое обучение

В конечном итоге повышение технического уровня команды может также улучшить качество программных продуктов [3].

Предложенная система критериев обеспечения качества программного обеспечения не является единственно верной и всеобъемлющей, но ее осмысление и использование в своей работе позволит будущим «тестировщикам» преодолеть кризис тестирования и подняться на ступень выше и стать специалистами по обеспечению качества программных продуктов.

Список литературы

1. Абдульмянов Т.Р., Петрова Н.К. Основные этапы разработки и тестирования высокоточных специализированных интеграторов // Вестник Казанского государственного энергетического университета. 2015. № 2 (26). С. 26-36.
2. Гараев И.А., Нарбаев И.Б., Петрова Н.К., Петрова А.С. Принципы построения тестовых примеров при разработке программного проекта // Modern Science. 2020. № 6-2. С. 236-241.
3. Шорина Т.В., Кирилова Г.И., Липатова И.А. Развитие информационной среды высшей школы в условиях реализации сетевой формы / Вестник Казанского государственного энергетического университета. 2017. № 3 (35). С. 146-152.