

СРОКИ РАЗРАБОТКИ ПРОТИВ КАЧЕСТВА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

БНТУ, г. Минск

Научный руководитель: канд. техн. наук, доцент Дробыш А. А.

В настоящее время вопрос качества поставляемого программного обеспечения является как никогда актуальным. С ростом количества разработчиков, готовых предоставить потребителям готовую продукцию, интерес в как можно раннем выходе программного обеспечения на рынок возрастает. Это обуславливается тем, что ключевым фактором для заказчиков является как можно более ранний вывод продукции на рынок, чтобы успеть заработать на продажах программы потребителям. Поспешность при разработке позволяет уложиться в более маленькие сроки, но в конечном итоге приводит к еще большим затратам времени и средств на исправление ошибок и реинжиниринг кода.

Довольно крупные коллективы разработчиков, как правило, ошибаются уже в период формирования технического задания. Несмотря на стандартную технологию разработки программного обеспечения, заказчики часто сокращают сроки, принуждая разработчиков к отказу от процесса отладки написанного кода. Следовательно, дебаггинг поручается тестировщикам, берущимся за работу после собрания всех модулей воедино.

Еще одной ошибкой является пренебрежение анализом предоставляемых заказчиком и исполнителем требований к будущей продукции. Неполное понимание целей, которые преследует заказчик, ведет к сдаче низкосортной программы, имеющей серьёзное количество дефектов и ошибок.

Методики, оказывающие положительное воздействие на качество ПО и не создающие слишком высокую нагрузку или проблемы для сроков разработки:

1. Анализ требований:

Анализ требований до реализации программного обеспечения помогает продумать и преодолеть возможные расхождения и разногласия, которые могут возникнуть в течение разработки продукта.

2. Анализ и сквозной контроль кода:

Эта методика помогает в повышении качества кода и дизайна, а также в устранении возникших дефектов. Кроме того, она позволяет всем разработчикам быть в курсе работы друг друга, облегчает передачу работы и повышает осведомленность группы в различных аспектах разработки программного обеспечения.

3. Сессионное тестирование:

Сессионное тестирование (SBTM) – метод управления исследовательским тестированием, при котором процесс тестирования разбивается на сессии определенной длины. Таким образом, сессионное тестирование представляет собой смешение формального и инновационного тестирования, так как дает простор исследованию и интуиции.

4. Тестирование, основанное на рисках

Подход к тестированию с целью минимизирования уровня проектных рисков и информирования заинтересованных лиц о текущем состоянии рисков с начальных стадий проекта. Подразумевает под собой управление процессом тестирования, исходя из идентифицированных рисков продукта. Проводится после каждого основного релиза. Методика позволяет убедиться, что изменения не нарушили важные функции и не привели к еще более существенным рискам.

Каждая компания-разработчик желает занять свою нишу на рынке, поэтому старается поставлять максимально качественный продукт. Однако, довольно часто из-за давления со стороны руководства или заказчика, качеством приходится поступиться ради сроков разработки.