

РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ПО СОЗДАНИЮ ОБУЧАЮЩИХ ПРОГРАММ ПРИ ПОМОЩИ TDD

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Алексейчик Д. О.

В статье рассматривается разработка веб-приложения по созданию обучающих программ при помощи TDD. Описана логическая составляющая такого подхода, целесообразность его использования.

TDD, *test-driven development* или процесс разработки через тестирование — это методология разработки программного обеспечения или веб-приложения, которая основывается на повторении коротких циклов разработки: изначально пишется тест, который покрывает желаемое изменение, затем пишется программный код, который реализует желаемое поведение системы и позволит пройти написанный тест, а затем проводится рефакторинг написанного кода с постоянной проверкой прохождения всех тестов.

Тестирование ПО — это процедура, которая позволяет подтвердить или опровергнуть работоспособность кода и корректность его работы. При тестировании веб-приложению передаются входные данные и запрашивается выполнение некой команды, после чего производится проверка полученных результатов на соответствие эталону, если результат соответствует ожидаемому — тест считается пройденным. Эта процедура может быть автоматизирована, в этом случае проверка работоспособности и правильности работы приложения в сравнении с ручным тестированием осуществляется гораздо быстрее, полноценнее и фактически чаще.

Методика разработки через тестирование является автоматическим тестированием разрабатываемого веб-приложения путем написания модульных, интеграционных и функциональных тестов, определяющих требования к коду непосредственно перед написанием этого самого кода. Сначала пишется тест, который проверяет корректность работы еще ненаписанного программного кода. Этот тест, разумеется, не проходит. После этого разработчик веб-приложения пишет код, который выполняет действия, требуемые для прохождения теста. После того, как тест успешно пройден, по необходимости осуществляется рефакторинг (доработка и переработка) написанного кода, причём в этом случае рефакторинг осуществляется уже под контролем прохождения тестов, что проще и надёжнее.

Цикл разработки по TDD:

- добавить тест для новой (еще не реализованной) функциональности веб-приложения;
- запустить все тесты и убедиться, что новый тест не проходит;
- написать код, который реализует новую логику;
- запустить тесты и убедиться, что они все прошли успешно;
- заняться рефакторингом и оптимизацией;
- перезапустить тесты и убедиться, что они все ещё проходят успешно;
- повторить цикл.

Эта методология позволяет добиться создания пригодного для автоматического тестирования веб-приложения и очень хорошего покрытия кода тестами, так как техническое задание переводится на язык автоматических тестов, то есть всё, что веб-приложение должно делать, проверяется. Также TDD часто упрощает программную реализацию: так как исключается избыточность — если компонент проходит тест, то он считается готовым. Если же существующие тесты проходят, но работает компонент не так, как ожидается, то это значит, что тесты пока не отражают всех требований и это повод добавить новые тесты.

В веб-приложениях, разрабатываемых таким образом, обычно очень хорошо распределяется ответственность между компонентами, а выполняемые сложные процедуры декомпозированы на множество простых. Стабильность работы веб-приложения, разработанного через тестирование, также выше за счёт того, что все основные функциональные возможности программы покрыты тестами и их работоспособность постоянно проверяется. Таким образом, веб-приложение по созданию обучающих программ будет разрабатываться и улучшаться, используя технологию TDD, чтобы веб-приложение было легко изменяемым и поддерживаемым.

Список используемых источников:

1. К.Бек Экстремальное программирование. Разработка через тестирование – 2000 г.
2. К. Сэм, Ф. Джек, Н. Кек Тестирование программного обеспечения – 2001 г.