

ПОДХОД К ТЕСТИРОВАНИЮ ПРОГРАММНЫХ ИНТЕРФЕЙСОВ ПРИЛОЖЕНИЙ МОБИЛЬНЫХ УСТРОЙСТВ

БНТУ, г. Минск

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

На сегодняшний день мобильные телефоны получили широкое распространение, а разработка приложений для мобильных устройств является одной из самых прибыльных сфер создания программных продуктов. Новые функциональные возможности программ вводятся в мобильные телефоны часто и быстро.

Программные интерфейсы приложений для мобильных устройств реализуются в виде стандартов, которые поддерживают производители устройств и программного обеспечения. Соответствие стандартам контролируется и базируется на тестировании интерфейсов в соответствии со спецификациями.

Основная задача статьи – обосновать подход к созданию тестов программных интерфейсов приложений для мобильных телефонов, основанный на спецификациях к интерфейсам.

Программный интерфейс приложения или интерфейс прикладного программирования – это набор функций, предоставляемый некоторой программной системой для использования ее функциональности в других прикладных программах.

Требования к данным программного интерфейса имеют декларативный характер и вносят ограничения на синтаксис и формат используемых данных. Функциональные требования в основном имеют условный характер. Они определяют функциональность (поведение) программной системы, которая должна быть создана разработчиками.

Исходя из семантики методов, предоставляемых программным интерфейсом приложения, возможны следующие варианты функционирования: ввод данных; вывод данных; переход в состояние; получение события/сообщения; отправка события/сообщения.

Объекты функционирования и взаимодействия – методы, события, диалоги, исключения, устройства ввода информации и т.п.

Техники тестирования. Классификация, описанная ниже, применяется к процессу тестирования, который базируется на спецификациях программного интерфейса приложения. Техники, базирующиеся на спецификации, разделяются на следующие виды:

1. Positive technique. Тесты строятся с ориентацией на использование тех величин, которые находятся в рамках специфицированных пределов значений.

2. Negative technique. Тесты строятся с ориентацией на использование тех величин, которые выходят за рамки специфицированных пределов значений, не соответствуют определенному формату или используют искаженные данные.

3. Boundary technique. Тесты строятся с ориентацией на использование тех величин, которые определяют предельные или близкие к предельным характеристики тестируемой системы.

4. Interaction technique. Тесты строятся с ориентацией на проверку взаимодействия между программными компонентами/модулями, имеющими связи в виде совместных процессов, ресурсов или устройств.

5. Stress technique. Техника является развитием и обобщением четырех техник, описанных выше.

При построении тестов на основе функциональных спецификаций в большинстве случаев информация о внутренней структуре системы неизвестна. Проектирование тестов происходит на основе принципа «чёрного ящика».

Создание тестовых сценариев позволяет оптимизировать процесс разработки тестов и, следовательно, процесс тестирования. При ограничениях на временные и человеческие ресурсы в процессе создания основных наборов тестов, выполняемого без анализа взаимосвязей в требованиях, могут быть пропущены проверки определенных вариантов использования системы, а некоторая последовательность действий может остаться не покрытой тестированием.

Предлагается выделить два подхода к созданию тестовых сценариев. Первый – от требований к тестам, второй – от имеющихся тестов к новым тестам.

В первом случае выполняется исследование требований, в котором прослеживаются связи между требованиями различных типов.

Во втором случае выполняется анализ плана тестирования на предмет совмещения схожих групп тестов.

Что касается анализа требований, то на данном этапе разработчик тестов для того, чтобы извлечь из требований возможные варианты использования интерфейса приложения, должен «встать на место» того, кто будет использовать API, в общем случае, разработчика приложения. Общих рекомендаций при создании тестовых приложений для API не существует. Тесты должны быть спроектированы таким образом, чтобы после их выполнения можно было с точностью определить место ошибки. Также желательно обеспечить основные и необходимые характеристики тестов: универсальность; переносимость; удобство сопровождения; расширяемость. Для того чтобы обеспечить все характеристики и иметь регулярную структуру тестового приложения, автором предлагается применять автоматный подход для разработки и программирования тестового приложения.

В работе изложены основные вопросы тестирования программного интерфейса приложения, произведен обзор требований программного интерфейса приложения и техник тестирования, описаны процедуры выбора тестов и создания тестовых сценариев, описан подход к созданию тестовых приложений для тестирования API.

Этот подход может быть использован при создании новых тестов, а также быть полезным при выявлении областей улучшения существующих тестов для программного интерфейса приложения.

УДК 621.762.4

Савва А. В., Гунько Е. А.

ДИСТАНЦИОННЫЕ ТЕХНОЛОГИИ В ОБУЧЕНИИ ИНФОРМАТИКЕ

БНТУ, г. Минск

Научный руководитель: ст. преподаватель Зуенок А. Ю.

В настоящее время дистанционное обучение находит всё более широкое применение на различных уровнях образования. Это связано с тем, что дистанционное обучение как инновационный образовательный процесс с использованием информационно-компьютерных технологий помогает школьникам реализовывать собственные образовательные цели, направленные на развитие лич-