

Министерство образования Республики Беларусь
Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования

Кафедра ИПиЭ

Дисциплина: Тестирование, оценка ПО

Отчет
по Лабораторной работе № 1
на тему «Виды тестирования. Планирование
тестирования»

Студент гр. 910902

Шаповалова Я.В.

Проверил

Кабариха В.А.

Минск 2022

Цель: изучить классификацию видов тестирования, разработать проверки для различных видов тестирования, научиться планировать тестовые активности в зависимости от особенностей поставляемой на тестирование функциональности.

Ход работы

1. Для данного пункта лабораторной работы в качестве объекта для тестирования будет выбран светильник (настольная лампа). Составим таблицу тестовых проверок для различных видов тестирования (табл.1.1).

Таблица 1.1

Объект тестирования: светильник (настольная лампа)		
Виды тестирования	Краткое определение вида тестирования	Тестовые проверки
Functional Testing	Тестирование, основанное на сравнительном анализе спецификации и функциональности объекта	<ol style="list-style-type: none"> 1. Включение/выключение света 2. Увеличение/уменьшение яркости 3. Изменение цвета света 4. Проверка на подвижность, устойчивость
Safety Testing	Тестирование оговоренным образом оставаться в рамках приемлемого риска причинения вреда здоровью, собственности или окружающей среде	<ol style="list-style-type: none"> 1. Продолжительное время светить на часть тела, а после проверить тот участок на наличие ожогов, облучения, сухости кожи тп. 2. Проверить на сколько быстро разлагается каждая деталь нашего объекта 3. Оставить гореть свет максимально продолжительное время и проверить, станет ли объект портиться (плавиться, дымиться, издавать неприятные запахи)
Security Testing	Тестирование с целью оценить защищённость программного продукта от внешних воздействий	<ol style="list-style-type: none"> 1. Определить возможную степень нажатия на кнопки 2. Определить через какое время кнопка перестанет реагировать 3. Ронять на предмет вещи

		<p>разной тяжести</p> <ol style="list-style-type: none"> 4. Ронять наш предмет 5. Проверка на устойчивость 6. Тянуть провод
Compatibility Testing	Проверка работоспособности объекта в различных средах	<ol style="list-style-type: none"> 1. Опустить объект в воду 2. Проверить работоспособность при высоких/низких температурах 3. Поместить объект в среду с очень высоким/низким давлением
GUI Testing	Тестирование, выполняемое путем взаимодействия с системой через графический интерфейс пользователя	<ol style="list-style-type: none"> 1. Проверить, совпадают ли характеристики с реальностью
Usability Testing	Тестирование с целью определения степени понятности, легкости в изучении и использовании, привлекательности продукта для пользователя при условии использования в заданных условиях эксплуатации	<ol style="list-style-type: none"> 1. Проверить насколько чувствительные кнопки на светильнике 2. Насколько яркий свет 3. Насколько устойчивая лампа
Accessibility Testing	Тестирование, которое определяет степень легкости, с которой пользователи с ограниченными способностями могут использовать систему или ее компоненты	<ol style="list-style-type: none"> 1. Можно ли её согнуть одной рукой 2. Насколько длинный провод и можно ли с лампой удобно передвигаться по комнате 3. Сможет ли слепой человек включить свет не видя кнопки
Internationalization Testing	Тестирование адаптации продукта к языковым и культурным особенностям целого ряда регионов, в которых	<ol style="list-style-type: none"> 1. Узнать совпадают ли значки вкл/выкл в других странах 2. Узнать где будут продавать наш товар и сделать инструкцию на соответствующих

	потенциально может использоваться продукт	языках
Performance Testing	Процесс тестирования с целью определения производительности продукта	<ol style="list-style-type: none"> 1. Оставить свет на долгое время включённым 2. Много тыкать на кнопки, и проверить на сколько долго они остаются рабочими
Stress Testing	Вид тестирования производительности, оценивающий систему или компонент на граничных значениях рабочих нагрузок или за их пределами	<ol style="list-style-type: none"> 1. Согнуть предмет настолько, насколько это возможно 2. Охладить/нагреть до критической температуры
Negative Testing	Полное тестирование системы или ее части на некорректных данных/сценариях (Negative Test)	<ol style="list-style-type: none"> 1. Использовать настольную лампу как потолочную 2. Поджигать лампу чтобы был свет 3. Светить в глаза чтобы ослепнуть
Black Box Testing	Тестирование системы без знания внутренней структуры и компонентов системы	<ol style="list-style-type: none"> 1. Нажимает кнопку и включается свет
Automated Testing	Набор техник, подходов и инструментальных средств, позволяющий исключить человека из выполнения некоторых задач в процессе тестирования	<ol style="list-style-type: none"> 1. Машина будет большое количество раз нажимать на кнопку 2. Машина может гнуть лампу пока та не сломается 3. Машина может изменять температуру помещения где находится лампа
Unit/Component Testing	Тестируются отдельные части (модули) системы	<ol style="list-style-type: none"> 1. Нажатие кнопки 2. Подвижность лампы 3. Качество соединения шнура и лампы
Integration Testing	Тестируется взаимодействие между отдельными модулями	<ol style="list-style-type: none"> 1. Соединение шнура с лампой 2. Крепление светодиодной ленты к лампе

3. Разработаем композицию тестов для первой поставки программного обеспечения (build 1), состоящей из трех модулей (модуль 1, модуль 2, модуль 3).

Опишем модули тестирования для композиции тестов первой поставки ПО:

Модуль 1: SMOKE Testing

Модуль включает в себя минимальный набор тестов для определения пригодности сборки для дальнейшего тестирования. Оно покрывает базовые функции программного обеспечения.

Модуль 2: NFT (New Feature Test)

Модуль необходим для определения качества поставленной на тестирование новой функциональности, которая ранее не тестировалась, тестирования новой функциональности на соответствие документации, проверки всевозможных взаимодействий ранее реализованной функциональности с новыми модулями и функциями.

Модуль 3: AT (Acceptance Test)

АТ предполагает полное тестирование системы или ее части как на корректных (Positive Test), так и на некорректных данных/сценариях (Negative Test)

4. Разработаем композицию тестов для второй поставки программного обеспечения (build 2), состоящей из четырех модулей (исправлены заведенные дефекты, доставлена новая функциональность – модуль 4).

Опишем модули тестирования для композиции тестов второй поставки ПО:

Модуль 1: SMOKE Testing

Модуль 2: DV (Defect Validation)

В условии задания было указано, что в данной поставке исправлены дефекты, обнаруженные ранее. Данный этап тестирования как раз таки необходим для проверки исправления дефектов.

Модуль 3: NFTAT (New Feature Testing)

Так как был добавлен новый функционал, в билд включается его полное тестирование.

Модуль 4: RTMAT (Regression Test)

так как ранее мы уже делали полную проверку программного продукта, регрессионное тестирование.

5. Разработаем композицию тестов для третьей поставки программного обеспечения (build 3): заказчик решил расширять рынки сбыта и просит осуществить поддержку программного обеспечения на английском языке.

Для тестирования билда воспользуемся стандартной схемой DV +

RTMAT. Т.е. выполняется проверка исправления дефектов программистом (Defect Validation, DV), а также проверка работоспособности остальной функциональности после исправления дефектов на позитивных сценариях (Minimal Acceptance Test, MAT). Ввиду того, что новый функционал добавлен не был, в NFTAT необходимости нет.

Так как была добавлена поддержка ПО на английском языке необходимо добавить модуль IT (Internationalization Testing). Он необходим для тестирования адаптации продукта к языковым и культурным особенностям целого ряда регионов, в которых потенциально может использоваться продукт. Так как английский язык - язык целого ряда стран мы будем использовать именно IT а не схожее по применению LT(Localization Testing).

6. Разработаем композицию тестов для четвертой поставки программного обеспечения (build 4): заказчик хочет убедиться, что программное обеспечение выдержит нагрузку в 2000 пользователей.

Так как новый функционал добавлен не был, воспользуемся схемой DV + RTMAT.

Для того, чтобы проверить «прочность ПО», добавим модуль нагрузочного тестирования (Performance and Load Testing). Данный модуль необходим для оценки поведения ПО при определенном количестве пользователей (в данном случае 2000).

Вывод: изучила классификацию видов тестирования, разработала проверки для различных видов тестирования, научилась планировать тестовые активности в зависимости от особенностей поставляемой на тестирование функциональности.

Контрольные вопросы:

1. Что такое тестирование?

Тестирование (Testing) – процесс анализа программного средства и сопутствующей документации с целью выявления дефектов и повышения качества продукта.

2. Что такое качество программного обеспечения?

Качество (Quality) – степень, с которой компонент, система или процесс соответствует зафиксированным требованиям и/или ожиданиям и нуждам пользователя или заказчика.

3. Что такое дефект?

Дефект (defect, bug, ошибка) – отклонение фактического результата от ожидаемого.

4. Назовите три условия обнаружения дефекта.

1. Знать ожидаемый результат.
2. Знать фактический результат.
3. Сравнить ожидаемый и фактический результаты.

4.

5. Какие существуют виды тестирования в зависимости от объекта тестирования? Дайте характеристику каждому.

В зависимости от объекта тестирования, тесты делятся на функциональные, пограничные и нефункциональные.

Функциональное тестирование - тестирование ПО в целях проверки реализуемости функциональных требований, то есть способности ПО в определённых условиях решать задачи, нужные пользователям.

Пограничное тестирование - тестирование ПО в различных средах.

Нефункциональное тестирование - тестирование, направленное на проверку характеристик или свойств программы.

6. Какие существуют виды функционального тестирования? Дайте характеристику каждому.

Функциональное тестирование включает в себя:

Функциональное тестирование (Functional Testing) – тестирование, основанное на сравнительном анализе спецификации и функциональности компонента или системы.

Тестирование безопасности (Safety Testing) – тестирование программного продукта с целью определить его способность при использовании оговоренным образом оставаться в рамках приемлемого риска причинения вреда здоровью, бизнесу, программам, собственности или окружающей среде.

Тестирование защищенности (Security Testing) – тестирование с целью оценить защищенность программного продукта от внешних воздействий (от проникновений).

7. Какие существуют виды нефункционального тестирования? Дайте характеристику каждому.

Тестирование требований (Requirements Testing) – проверка требований на соответствие основным атрибутам качества.

Тестирование прототипа (Prototype Testing) – метод выявления структурных, логических ошибок и ошибок проектирования на ранней стадии развития продукта до начала фактической разработки.

Тестирование пользовательского интерфейса (GUI Testing) – тестирование, выполняемое путем взаимодействия с системой через графический интерфейс пользователя .

Тестирование удобства использования (Usability Testing) – тестирование с целью определения степени понятности, легкости в изучении и использовании, привлекательности программного продукта для пользователя при условии использования в заданных условиях эксплуатации .

Тестирование доступности (Accessibility Testing) – тестирование, которое определяет степень легкости, с которой пользователи с ограниченными способностями могут использовать систему или ее компоненты.

Тестирование интернационализации (Internationalization Testing) – тестирование адаптации продукта к языковым и культурным особенностям целого ряда регионов, в которых потенциально может использоваться продукт.

Тестирование локализации (Localization Testing) – тестирование адаптации продукта к языковым и культурным особенностям конкретного региона, отличного от того, в котором разрабатывался продукт.

Тестирование производительности (Performance Testing) – процесс тестирования с целью определения производительности программного продукта.

Нагрузочное тестирование (Performance and Load Testing) – вид тестирования производительности, проводимый с целью оценки поведения компонента или системы при возрастающей нагрузке, например количестве параллельных пользователей и/или операций, а также определения какую нагрузку может выдержать компонент или система;

Объемное тестирование (Volume Testing) – позволяет получить оценку производительности при увеличении объемов данных в базе данных приложения;

Тестирование стабильности и надежности (Stability / Reliability Testing) – позволяет проверять работоспособность приложения при длительном (многочасовом) тестировании со средним уровнем нагрузки.

Стрессовое тестирование (Stress Testing) – вид тестирования производительности, оценивающий систему или компонент на граничных значениях рабочих нагрузок или за их пределами, или же в состоянии ограниченных ресурсов, таких как память или доступ к серверу.

Тестирование на отказ и восстановление (Failover and Recovery Testing) – тестирование при помощи эмуляции отказов системы или реально вызываемых отказов в управляемом окружении.

Тестирование установки (Installability Testing) и лицензирования – процесс тестирования установки программного продукта.

8. Какие существуют виды тестирования в зависимости от глубины покрытия? Дайте характеристику каждому.

Smoke Test – поверхностное тестирование для определения пригодности сборки для дальнейшего тестирования, должно покрывать базовые функции программного обеспечения; уровень качества: Acceptable / Unacceptable.

Minimal Acceptance Test (MAT, Positive Test) – тестирование системы или ее части только на корректных данных/сценариях; уровень качества: High / Medium / Low.

Acceptance Test (AT) – полное тестирование системы или ее части как на корректных (Positive Test), так и на некорректных данных/сценариях (Negative Test).

9. Какие существуют тестовые активности? Дайте характеристику каждому.

Тестирование новых функциональностей (New Feature Test, NFT) – определение качества поставленной на тестирование новой функциональности, которая ранее не тестировалась.

Регрессионное тестирование (Regression Testing, RT) проводится с целью оценки качества ранее реализованной функциональности. Валидация дефектов (Defect Validation, DV) – проверка результатов исправления дефектов; может включать элементы регрессионного тестирования; уровень проверки не определяется.

10. Какие существуют виды тестирования в зависимости от знания кода? Дайте характеристику каждому.

Белый ящик (White Box Testing) – тестирование, основанное на анализе внутренней структуры компонентов или системы (у тестирующего есть доступ к внутренней структуре и коду приложения).

Серый ящик (Grey Box Testing) – комбинация методов белого и черного ящика, состоящая в том, что к части кода архитектуры у тестирующего есть, а к части кода – нет.

Черный ящик (Black Box Testing) – тестирование системы без знания внутренней структуры и компонентов системы (у тестирующего нет доступа к внутренней структуре и коду приложения либо в процессе тестирования он не обращается к ним).

11. Какие существуют виды тестирования в зависимости от степени автоматизации? Дайте характеристику каждому.

Ручное тестирование – такое тестирование, в котором тест-кейсы выполняются тестирующим вручную без использования средств автоматизации.

Автоматизированное тестирование (Automated Testing) – набор техник, подходов и инструментальных средств, позволяющий исключить человека из выполнения некоторых задач в процессе тестирования.

12. Какие существуют виды тестирования в зависимости от изолированности компонентов? Дайте характеристику каждому.

Модульное тестирование (Unit/Component Testing) – тестируются отдельные части (модули) системы.

Интеграционное тестирование (Integration Testing) – тестируется взаимодействие между отдельными модулями.

Системное тестирование (System Testing) – тестируется работоспособность системы в целом.

13. Какие существуют виды тестирования в зависимости от подготовленности? Дайте характеристику каждому.

Интуитивное тестирование выполняется без подготовки к тестам, без определения ожидаемых результатов,

проектирования тестовых сценариев.

Исследовательское тестирование – метод проектирования тестовых сценариев во время выполнения этих сценариев.

Тестирование по документации – тестирование по подготовленным тестовым сценариям, руководству по осуществлению тестов.

14. Какие существуют виды тестирования в зависимости от места и времени проведения? Дайте характеристику

каждому.	Приемочное тестирование (User Acceptance Testing, UAT)	– формальное тестирование
определения	соответствия системы	
критериям	приёмки	и дать
возможность пользователям, заказчикам или иным авторизованным лицам определить, принимать систему.		

Альфа-тестирование (Alpha Testing) – моделируемое или действительное функциональное тестирование, выполняется в организации, разрабатывающей продукт, но не проектной командой (это может быть независимая команда тестировщиков, потенциальные пользователи, заказчики).

Альфа тестирование часто применяется к коробочному программному обеспечению в качестве внутреннего приемочного тестирования.

Бета-тестирование (Beta Testing) – эксплуатационное тестирование потенциальными или существующими клиентами/заказчиками на внешней стороне (в среде, где продукт будет использоваться) никак связанными с разработчиками, с целью определения действительно ли компонент или система удовлетворяет требованиям клиента/заказчика и вписывается в бизнес-процессы.

15. Какие этапы составляют процесс тестирования?

Время проверки не определяется.

Процесс тестирования программного продукта включает следующие этапы:

1. Изучение и анализ предмета тестирования.
2. Планирование тестирования.
3. Исполнение тестирования.

16. Какая композиция тестов выполняется для первой поставки программного продукта?

Для первой поставки программного обеспечения рекомендуется проводить Smoke + NFTAT.

17. Какие композиции тестов выполняется для последующих поставок программного продукта?

Для второй и последующих поставок обобщенная схема композиции тестов выглядит следующим образом: (Smoke) + DV + (NFTAT) + RTMAT.

