

Министерство образования Республики Беларусь
Учреждения образования
**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ**

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

Дисциплина: Тестирование, оценка программного обеспечения

Отчет
по **Лабораторной работе №1**
на тему: Виды тестирования. Планирование тестирования.

Студент

гр.910902

Корчиков К.Д.

Проверил

Кабариха В.А.

Минск 2022

Цель работы: 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. Необходимо разработать тестовые проверки для бельевой пластмассовой прищепки. Составить таблицу с различными видами тестирования (таблица 1).

Таблица 1 – Виды тестирований прищепки.

Объект тестирования: прищепка бельевая пластмассовая		
Вид тестирования	Краткое определение вида тестирования	Тестовые проверки
Functional Testing	Тестирование, основанное на сравнительном анализе спецификации и функциональности компонента или системы.	Сжимается и разжимается? Держит ли бельё?
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.	Делает дырки в белье? Царапает или прищемляет пальцы?
Security Testing	Тестирование с целью оценить защищенность программного продукта от внешних воздействий (от проникновений).	Расплавится ли на солнце? Сломается ли при минусовой температуре или от влаги?
Compatibility Testing	Проверка работоспособности приложения в различных средах (браузеры и их версии, операционные системы, их типа, версии и разрядность).	Как будет вести себя на улице? Как будет вести себя на балконе?
GUI Testing	Тестирование, выполняемое путем взаимодействия с системой через графический интерфейс пользователя.	Какого цвета? Какой формы?

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.	Легко ли разжимается? Удобно ли удерживать в руках или снять с белья?
Accessibility Testing	Тестирование, которое определяет степень легкости, с которой пользователи с ограниченными способностями могут использовать систему или ее компоненты.	Разжать прищепку одной рукой. Разжать прищепку с закрытыми глазами.
Internationalization Testing	Тестирование адаптации продукта к языковым и культурным особенностям целого ряда регионов, в которых потенциально может использоваться продукт.	Прищепкой цепляют бельё или другие предметы, которые могут использоваться в других регионах.
Performance Testing	Процесс тестирования с целью определения производительности программного продукта.	Как долго прищепка способна держать бельё.
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.	Уронить. Разжать насколько это возможно.
Negative Testing	Тестирование на некорректных данных/сценариях.	Сломать. Разделить пополам.
Black Box Testing	Тестирование системы без знания внутренней структуры и компонентов системы.	Прищепить бельё прищепкой.
Automated Testing	Набор техник, подходов и инструментальных средств, позволяющий исключить человека из выполнения некоторых задач в процессе тестирования.	Сделать робота, научить цеплять прищепку к белью. Сделать робота, научить снимать прищепку с белья.

Unit/Component Testing	Тестируются отдельные части (модули) системы.	Тестируем пружину на упругость. Тестируем половинки прищепки на прочность.
Integration Testing	Тестируется взаимодействие между отдельными модулями.	Соединили половинки с пружинкой. Держится ли всё вместе?

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

Модуль 1

Smoke Test - поверхностное тестирование для определения пригодности сборки для дальнейшего тестирования, должно покрывать базовые функции программного обеспечения.

Модуль 2

NFTAT (New Feature Test) - определение качества поставленной на тестирование новой функциональности, которая ранее не тестировалась. Данный тип тестирования включает в себя: проведение полного теста (АТ) непосредственно новой функциональности; тестирование новой функциональности на соответствие документации; проверку всевозможных взаимодействий ранее реализованной функциональности с новыми модулями и функциями.

Модуль 3

AT (Acceptance Test) - полное тестирование системы или ее части как на корректных (Positive Test), так и на некорректных данных/сценариях (Negative Test); уровень качества: High / Medium / Low. Тест на этом уровне покрывает все возможные сценарии тестирования: проверку работоспособности модулей при вводе корректных значений; проверку при вводе некорректных значений; использование форматов данных отличных от тех, которые указаны в требованиях; проверку исключительных ситуаций, сообщений об ошибках; тестирование на различных комбинациях входных параметров; проверку всех

классов эквивалентности; тестирование граничных значений интервалов; сценарии не предусмотренные спецификацией и т.д.

3. Необходимо разработать композицию тестов для второй поставки программного обеспечения (build 2): исправлены заведенные дефекты, доставлена новая функциональность – модуль 4.

Модуль 1: Smoke Test.

Модуль 2: Defect Validation - проверка результатов исправления дефектов; может включать элементы регрессионного тестирования; уровень проверки не определяется.

Модуль 3: NFTAT.

Модуль 4: Regression Testing - проводится с целью оценки качества ранее реализованной функциональности. Включает в себя проверку стабильности ранее реализованной функциональности после внесения изменений, например, добавления новой функциональности, исправление дефектов, оптимизация кода, разворачивание приложения на новом окружении. Регрессионное тестирование как правило выполняется на уровне MAT.

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

Так как, никакой новой функциональности не было добавлено, то: DV + RTMAT. Т.е., выполняется проверка исправления дефектов программистом (Defect Validation, DV), а также проверка работоспособности остальной функциональности после исправления дефектов на позитивных сценариях MAT.

Но было добавлено ПО поддержки английского языка. Поэтому необходимо добавить модуль Internationalization Testing. Он нужен для тестирования адаптации продукта к языковым и культурным особенностям целого ряда регионов, в которых потенциально может использоваться продукт.

5. Необходимо разработать 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.

Вывод: в ходе данной лабораторной работы были изучены и разработаны различные виды тестовых проверок, в данном случае, для пластмассовой бельевой прищепки. Получены навыки планирования тестовых активностей в зависимости от поставленной задачи.