

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

Отчёт по лабораторной работе №1
Виды тестирования. Планирование тестирования

Выполнил:

Пранович А.Г.

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.

Таблица 1 – Тестовые проверки для различных видов тестирования

Объект тестирования: фитнес-браслет		
Вид тестирования	Краткое определение вида тестирования	Тестовые проверки
Functional Testing	Сравнительный анализ спецификации и функциональности компонента или системы	<ul style="list-style-type: none"> – считает количество шагов? – показывает дата-время? – хранит промежуточные данные? – отображает их? – можно зарядить?
Safety Testing	Определение способности при использовании оговоренным образом оставаться в рамках приемлемого риска причинения вреда здоровью, бизнесу, программам, собственности или окружающей среде	<ul style="list-style-type: none"> – не сожмёт руку? – не повлияет на внутреннюю работу организма?
Security Testing	Оценка защищённости программного продукта от внешних воздействий	<ul style="list-style-type: none"> – не сломается при работе? – не перестанет заряжаться?
Compatibility Testing	Проверка работоспособности приложения в различных средах	<ul style="list-style-type: none"> – не откажет в воде? – повлияет ли изменение температуры?
GUI Testing	Проверка взаимодействия с системой через графический интерфейс пользователя	<ul style="list-style-type: none"> – какого цвета? – какой формы? – есть ли логотип?
Usability Testing	Определение степени понятности, лёгкости в изучении и использовании, привлекательности программного продукта для пользователя при условии использования в заданных условиях эксплуатации	<ul style="list-style-type: none"> – удобно носить на руке? – удобная навигация?

Продолжение таблицы 1

Accessibility Testing	Определение степени лёгкости, с которой пользователи с ограниченными способностями могут использовать систему или её компоненты	<ul style="list-style-type: none"> – можно ли носить на руке при наличии каких-либо проблем? – смогут ли пользоваться дальномерами?
Internationalization Testing	Проверка адаптации продукта к языковым и культурным особенностям целого ряда регионов	<ul style="list-style-type: none"> – поддерживается ли формат данных текущим регионом? – поддерживается ли кодировка?
Performance Testing	Определение производительности программного продукта	<ul style="list-style-type: none"> – как много хранит данных? – сколько раз можно зарядить? – насколько хватает батареи?
Stress Testing	Оценка производительности системы или компонента на граничных значениях рабочих нагрузок, или за их пределами, или в состоянии ограниченных ресурсов	<ul style="list-style-type: none"> – залить водой. – постучать по столу. – уронить на пол.
Negative Testing	Проверка устойчивости системы к негативному воздействию	<ul style="list-style-type: none"> – подать большое напряжение. – сломать. – намочить и высушить.
Black Box Testing	Проверка системы без знания внутренней структуры и компонентов системы	<ul style="list-style-type: none"> – неизвестно как устроен. – пройти несколько метров — шаги посчитаются? – нажать на дисплей — информация отобразится?
Automated Testing	Проверка с помощью набора техник, подходов и инструментальных средств, позволяющий исключить человека	<ul style="list-style-type: none"> – сделать робота, научить его ходить, чтобы считать шаги. – автоматическая беспроводная зарядка.
Unit/Component Testing	Проверка отдельных частей (модулей) системы	<ul style="list-style-type: none"> – отдельно замеряем шаги капсулой. – отдельно носим ремешок на руке.

Продолжение таблицы 1

Integration Testing	Проверка взаимодействия между отдельными модулями	– собрали — капсула не вываливается из ремешка? – а в процессе работы?
---------------------	---	---

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

SMOKE + NFT_{АТ}(1,2,3)

Поверхностное тестирование (Smoke Test) выполняется для определения пригодности сборки для дальнейшего тестирования; полное тестирование системы или её части как на корректных, так и на некорректных данных/сценариях (Acceptance Test, АТ) позволяет обнаружить дефекты и внести запись о них в багтрекинг-систему.

Разработать композицию тестов для второй поставки программного обеспечения (build 2): исправлены заведённые дефекты, доставлена новая функциональность — модуль 4.

SMOKE + DV + NFT_{АТ}(4) + RT_{МАТ}(1,2,3)

Поверхностное тестирование (Smoke Test), проверка исправления дефектов программистом (Defect Validation, DF), тестирование новых функциональностей (New Feature Testing, NFT), проверка старых функциональностей, т. е. регрессионное тестирование (Regression Testing, RT).

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

SMOKE + IT + LT

Тестирование продукта на обеспечение безбарьерной локализации (Internalization Testing, IT), тестирование локализации (Localization Testing, LT).

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

SMOKE + PLT

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

Вывод: в данной лабораторной работе я познакомился с процессом тестирования, выяснил для чего оно необходимо, узнал о различных видах тестирования, а также применил их к выбранному объекту.