

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования
Кафедра инженерной психологии и эргономики
Дисциплина: Тестирование и оценка программного обеспечения

Лабораторная работа №1

**ВИДЫ ТЕСТИРОВАНИЯ. ПЛАНИРОВАНИЕ
ТЕСТИРОВАНИЯ**

Выполнил: студент группы 910901

Стасевич В.Г.

Проверил: Кабариха В.А.

Минск 2022

Объект тестирования: фоторамка		
Вид тестирования	Краткое определение вида тестирования	Тестовые проверки
Functional Testing	Сравнительный анализ спецификации и функциональности	<ul style="list-style-type: none"> • Вешается ли на стену • Ставится на стол • Засунуть/достать фотографию • Пробовать на прочность
Safety Testing	Защита внешнего мира от объекта	<ul style="list-style-type: none"> • Можно ли порезаться о стекло в рамке? • Можно ли загнать занозу?
Security Testing	Защита объекта от внешнего мира	<ul style="list-style-type: none"> • Можно ли сломать? • Может ли поцарапаться стекло? • Можно ли разбить/сжечь?
Compatibility Testing	Проверка работоспособности в различных средах	<ul style="list-style-type: none"> • Как воздействует с температурой? • Как ведет себя под водой ?
GUI Testing	Проверка стиля, формы и т.д.	<ul style="list-style-type: none"> • Проверка шрифта на читабельность • Дизайн • Наличие логотипа фирмы • Цвет рамки соответствует картинке/фотографии • Размер/форма
Usability Testing	Проверка использования	<ul style="list-style-type: none"> • Удобно ли держать в руке • Вешать на стену/ставить на стол • Хорошо ли рамка держит картинку • Поменять фотографию • Читать текст • Положить в сумку

		<ul style="list-style-type: none"> • Остается ли краска от рамки • Все ли отгибается хорошо • Удобно ли ставить на поверхность без повреждений
Accessibility Testing	Проверка легкости использования пользователями с ограниченными возможностями	<ul style="list-style-type: none"> • Удобно ли использовать людям с ограничениями по зрению? • С проблемами координации движений?
Internationalization Testing	Проверка адаптации продукта к различным языковым и культурным особенностям	<ul style="list-style-type: none"> • Как используется в разных странах? • Допустим ли заданный дизайн в различных странах?
Performance Testing	Проверка производительности	<ul style="list-style-type: none"> • Долговечность
Stress Testing	Проверка на граничных значениях или в состоянии ограниченных ресурсов	<ul style="list-style-type: none"> • Ударить • Потрясти • Уронить • Попытаться согнуть • Прочность веревки/гвоздя • Влажность • Пожароустойчивость
Negative Testing	Тестирование на заведомо отрицательный результат	<ul style="list-style-type: none"> • Сжечь • Заморозить • Оставить в воде на большой промежуток времени
Black Box Testing	Тестирование без знания внутренней структуры и компонентов	<ul style="list-style-type: none"> • Понятно ли как пользоваться интуитивно?
Automated Testing	Тестирование без участия человека	<ul style="list-style-type: none"> • Сделать робота, научить его менять фотографию

		и вешать на стену/ставить на стол
Unit/Component Testing	Тестирование отдельных компонентов	<ul style="list-style-type: none"> • Отдельно проверяем рамку • Отдельно проверить стекло • Отдельно проверяем картинку/фотографию
Integration Testing	Тестирование взаимосвязи между отдельными модулями	<ul style="list-style-type: none"> • Проверить качество проклейки между деталями

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

Smoke + NFT_{AT}(1,2,3)

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

Smoke + NFT_{AT}(4) + DV + RT_{МАТ}

5. 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

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

Smoke + PT + VT