Министерство образования Республики Беларусь

Учреждение образования БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

ОТЧЕТ

к лабораторной работе на тему

# Виды тестирования.

**Планирование тестирования**

Студент Темиров. Б.

Руководитель В.А. Кабариха

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.

# Практическое задание:

1. Выбрать объект реального мира (например, карандаш, стол, чашка, клавиатура, сумка и др.) с целью последующей разработки тестовых проверок для него.
2. Разработать различные проверки в соответствии с классификацией видов тестирования для выбранного объекта реального мира. Результаты внести в таблицу 1.1.

|  |  |  |
| --- | --- | --- |
| **Объект тестирования**: Блокнот | | |
| **Вид тестирования** | **Краткое определение вида тестирования** | **Тестовые проверки** |
| Functional Testing | основано на сравнительном анализе спецификации и  функциональности компонента или системы | Можно ли писать? Можно ли стереть написанное?  Вырвать лист? |
| Safety Testing | определить способность программного продукта при использовании оговоренным образом оставаться в рамках приемлемого риска причинения вреда  здоровью, бизнесу, программам, собственности или окружающей среде | Можно ли порезаться листом при перелистывании?  Экологичность в порядке? |
| Security Testing | тестирование с целью оценить защищенность  программного продукта от внешних воздействий. | Защищён от попаданий воды?  Легко ли загарается? |
| Compatibility Testing | проверка  работоспособности приложения в различных средах. Виды тестирования совместимости:  кроссбраузерное тестирование, кроссплатформенное. | Можно ли использовать в космосе?  Под водой? |
| GUI Testing | тестирование, выполняемое путем взаимодействия с системой через  графический интерфейс пользователя. | В клеточку или в линеечку? В косую или прямую линейку?  Цветной блокнот? |

|  |  |  |
| --- | --- | --- |
| 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. | Обложка одноцветная или цветная?  Строки на обложке заполнены?  Если картина если да то удобно для восприятии |
| Accessibility Testing | тестирование, которое определяет степень легкости, с которой пользователи с ограниченными | Можно перелистывать одной рукой?  Писать в ней одной рукой? Люди с ограниченным возможностями могут  писать? |

|  |  |  |
| --- | --- | --- |
|  | способностями могут использовать систему или  ее компоненты. |  |
| Internationalization Testing | тестирование адаптации продукта к языковым и культурным особенностям целого ряда регионов, в которых потенциально может использоваться  продукт. | Цвет блокнота, можно ли писать с право на лево, с лево на право или вертикально? |
| Performance Testing | процесс тестирования с целью определения производительности  программного продукта. | Каково количество листов? Достаточно ли толста обложка? |
| 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. | Можно ли писать ручкой на жирном пятне?  Можно ли писать на мокром блокноте?  Писать на обложке при исписанных листах?  Можно ли стереть то что написано? |
| Negative Testing | Негативное тестирование подразумевает негативные сценарии т.е. сценарии, в которых  система/компонент что-то НЕ делает. | Закончились чистые листы блокнота намокла |
| Black Box Testing | тестирование системы без знания внутренней  структуры и компонентов системы. | Можно ли открыть  блокнот? Или надо писать на обложке? |
| Automated Testing | набор техник, подходов и инструментальных средств, позволяющий исключить человека из выполнения некоторых задач в процессе тестирования. Тест-кейсы частично или полностью  выполняет специальное инструментальное средство | Научить робота писать в блокноте. |
| Unit/Component Testing | тестируются отдельные части (модули) системы. | Насколько крепка обложка? Отдельная страница?  Везде ли можно писать? |
| Integration Testing | тестируется взаимодействие между отдельными модулями. | Останется ли след на другой странице после закрытия блокнота?  Если вырвать один лист, второй выпадет? |

1. задание:

Smoke + NFTАТ. Таким образом я проверяю работоспособность основного функционала продукта при помощи Smoke-тестирования, а затем новые возможности (NFT) очень подробно при помощи АТ-тестирования. Это лучшее сочетание видов тестирования для первой поставки ПО.

1. задание:

Smoke + DV + RTMAT + NFTАТ. В данном случае после исправления дефектов (DV) необходимо их протестировать, поэтому MAT-тестирования подходит для этого лучше всего. Затем после внедрения нового функционала (NFT) его так же необходимо протестировать при помощи АТ-тестирования.

1. задание:

Smoke + IT + LT. При поддержке нового языка (в данном случае

английского) проводится тестирование локализации и интернационализации, (IT) тестирование адаптации продукта к языковым и культурным

особенностям целого ряда регионов, Localization Testing - (LT) тестирование

адаптации продукта к языковым и культурным особенностям конкретного региона, отличного от того, в котором разрабатывался продукт.

1. задание:

Smoke + VL + PT + ST. Необходимо провести нагрузочное, объёмное и стрессовое тестирования, чтобы определить стабильность ПО при определённойнагрузке и определённой длительности данной нагрузки.

Volume Testing - (VL), позволяет получить оценку производительности при увеличении объемов данных в базе данных приложения. (PT) - Performance Testing, процесс тестирования с целью определения производительности программного продукта. (ST) - 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.