

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

Е.В. Муромская

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.

*Таблица 1.1 – Тестовые проверки для различных видов тестирования*

<b>Объект тестирования: рюкзак</b>		
<b>Вид тестирования</b>	<b>Краткое определение вида тестирования</b>	<b>Тестовые проверки</b>
Functional Testing	– тестирование, основанное на сравнительном анализе спецификации и функциональности компонента или системы.	<ol style="list-style-type: none"> <li>1. Рюкзак имеет две лямки, для того чтобы носить за спиной</li> <li>2. Рюкзак имеет отсек для хранения вещей</li> <li>3. Рюкзак имеет замок для закрытие отсека</li> </ol>
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.	<ol style="list-style-type: none"> <li>1. Проверить, что на рюкзаке нет острых, царапающих или режущих выступов</li> <li>2. Проверить, чтобы пустой рюкзак не превышал вес в 5кг, чтобы не навредить спине</li> </ol>
Security Testing	– тестирование с целью оценить защищенность программного продукта от внешних воздействий (от проникновений).	<ol style="list-style-type: none"> <li>1. Проверить наличие кодового замка на рюкзаке для хранения ценных вещей</li> <li>2. Проверить возможность бесшумного раскрытия рюкзака, когда он находится за спиной человека</li> </ol>

Compatibility Testing	–проверка работоспособности приложения в различных средах (браузеры и их версии, операционные системы, их типы, версии и разрядность)	1. Проверить не соскальзывают ли лямки рюкзака с ветровки, шерстяной кофты, куртки, майки
GUI Testing	–тестирование, выполняемое путем взаимодействия с системой через графический интерфейс пользователя.	<ol style="list-style-type: none"> <li>1. Проверить на рюкзаке правильность написания фирмы производителя</li> <li>2. Проверка правильности дополнительных надписей на рюкзаке</li> <li>3. Проверка наличие бирки на рюкзаке с правильно написанной инструкцией</li> </ol>
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.	<ol style="list-style-type: none"> <li>1. Проверка рюкзака на удобное распределение нагрузки на спину</li> <li>2. Проверка рюкзака на удобное расположение лямок</li> <li>3. Проверка рюкзака на удобное распределение карманов</li> <li>4. Проверка рюкзака на достаточный объем отсека для вещей</li> </ol>
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"> <li>1. Проверить на наличие в рюкзаке петли для взятия его одной рукой</li> <li>2. Проверить наличие выступов на кодовом замке</li> </ol>
Internationalization Testing	– тестирование адаптации продукта к языковым и культурным особенностям целого ряда регионов, в которых потенциально может использоваться продукт.	<ol style="list-style-type: none"> <li>1. Проверить наличие надписи на брошюре на различных языках</li> <li>2. Проверить на наличие оскорбляющих</li> </ol>

		надписей на иностранных языках
Performance Testing	– процесс тестирования с целью определения производительности программного продукта.	<ol style="list-style-type: none"> <li>1. Проверить целостность рюкзака после заполнения отсека вещами весом в 6кг</li> <li>2. Проверить работу замка после 1000 раскрытий и закрытий подряд</li> <li>3. Проверить исправность лямок при нагрузке в 60Н</li> </ol>
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	<ol style="list-style-type: none"> <li>1. Проверить целостность рюкзака после заполнения отсека вещами весом в 12кг</li> <li>2. Проверить работу замка после 100000 раскрытий и закрытий подряд</li> <li>3. Проверить исправность лямок при нагрузке в 120Н</li> </ol>
Negative Testing	- тестирование, в рамках которого применяются сценарии, которые соответствуют внештатному поведению тестируемой системы.	<ol style="list-style-type: none"> <li>1. Просмотр целостности содержимого рюкзака после опускания его в воду</li> <li>2. Проверка работы замка при застревании ткани</li> <li>3. Проверка работы кодового замка при неправильной комбинации цифр</li> </ol>
Black Box Testing	– тестирование системы без знания внутренней структуры и компонентов системы	<ol style="list-style-type: none"> <li>1. Проверка открытия рюкзака без использования замка</li> </ol>
Automated Testing	– набор техник, подходов и инструментальных средств, позволяющий исключить человека из выполнения некоторых задач в процессе тестирования.	<ol style="list-style-type: none"> <li>1. Проверить открытие и закрытие замков с помощью робота</li> <li>2. Проверить исправность рюкзака после того, как он был попушен под пресс</li> </ol>

Unit/Component Testing	– тестируются отдельные части (модули) системы.	1. Проверка лямок/отсека/петельки рюкзака 2. Проверка кодового замка 3. Проверка фликера для рюкзака
Integration Testing	–тестируется взаимодействие между отдельными модулями.	1. Проверка держания фликера на рюкзаке 2. Проверка крепления кодового замка к замку рюкзака

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

build №1: Smoke + NFTAT

Smoke – проверка первых трех модулей и определения возможности дальнейшего тестирования. Если дальнейшее тестирование возможно, надо проверить каждый модуль до уровня АТ.

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

Build №2: Smoke + DV + NFTAT + RTMAT.

Smoke тестирование проверяет работу основных функций, после необходимо провести верификацию исправленных багов, сделать регрессионное тестирование старых функциональностей до уровня МАТ и проверить новый модуль до уровня АТ.

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.

Build№3: Smoke + DV + RTMAT и тестирование локализации с поддержкой английского языка. Проверка всех старых функциональностей, регрессионное тестирование по уровню МАТ и проверка исправленности багов.

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

Build№4: Smoke + DV + RTMAT и тестирование производительности в 2000+ пользователей.

**Выводы:** существует огромное количество видов тестирования, которые обеспечивают проверку различных аспектов продукта (нефункциональных и

функциональных). Планирование тестирования зависит от входных данных при поставке модулей продукта.