

Министерство образования Республики Беларусь  
Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования

Кафедра ИПиЭ

Дисциплина: Тестирование, оценка ПО

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

Студент гр. 910903

Солтан М.В.

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 1.1

<b>Объект тестирования: точилка для карандаша</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	1. Хорошо ли она может заточить карандаш? 2а. Не сломается ли она в процессе? 2б. Не сломается ли она, если ее уронить? 3. Не сильно ли портится лезвие в процессе заточки?
Safety Testing	Тестирование ПП для определения его способности оставаться в рамках приемлемого риска причинения вреда здоровью, бизнесу, программам, собственности или окружающей среде.	1а. Не порежемся ли мы, пока будем точить карандаш? 1б. Не порежемся ли мы если случайно сожжем ее в руке?
Security Testing	Тестирование защищенности ПП от внешних воздействий	1. Сможет ли она заточить кривой карандаш? 2. Начнет ли она ржаветь, если будет периодически находиться в контакте с водой?

Compatibility Testing	Проверка работоспособности приложения в различных средах	1. Сможет ли она заточить карандаши разной толщины?
GUI Testing	Тестирование взаимодействия с системой через графический интерфейс пользователя	1. Не слишком ли точилка большая/маленькая? 2. Материал, из которого она состоит, достаточно прочный?
Usability Testing	Тестирование для определения степени понятности, легкости в изучении и использовании, привлекательности ПП для пользователя	1. Удобно ли держать ее в руках, пока точишь карандаш?
Accessibility Testing	Тестирование определяющее степень легкости, с которой пользователи с ограниченными способностями могут использовать систему	1а. Можно ли понять где передняя/задняя часть точилки в темноте? 1б. Можно ли заточить карандаш без света?
Internationalization Testing	Тестирование адаптации продукта к языковым и культурным особенностям	1. Смогут ли с помощью выпускаемой точилки заточить карандаш люди из другой страны?
Performance Testing	Тестирования с целью определения производительности ПП	1. Каков срок годности? 2. Сколько карандашей можно будет заточить ей до того, пока не затупится лезвие?
Stress Testing	Тестирование производительности,	1. Сломается ли точилка, если ее

	оценивающее систему на граничных значениях рабочих нагрузок/за их пределами/в состоянии ограниченных ресурсов	уронить? 2. Сломается ли точилка, если ее сильно сжать?
Negative Testing	Полное тестирование системы или ее части на некорректных данных/сценариях	1. Можно ли с помощью нее заточить палец? 2. Сломается ли точилка, если с помощью нее попробовать заточить ручку?
Black Box Testing	Тестирование системы без знания внутренней структуры и компонентов системы	1. Догадается ли человек, для чего точилка, если дать ее ему и ничего не объяснять?
Automated Testing	Набор техник, подходов и инструментальных средств, позволяющий исключить человека из выполнения некоторых задач в процессе тестирования	1. Заставить робота точить карандаши до того, пока не сломается точилка
Unit/Component Testing	Тестируются отдельные части (модули) системы	1. Отдельно проверить остроту лезвия и прочность корпуса
Integration Testing	Тестируется взаимодействие между отдельными модулями	1. Не отваливается ли лезвие?

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

**Smoke + NFT<sub>AT</sub>(1, 2, 3)**

#### **Smoke Testing:**

Модуль включает в себя минимальный набор тестов для определения пригодности сборки для дальнейшего тестирования. Оно покрывает базовые функции ПО.

#### **NFT (New Feature Test)**

Модуль необходим для определения качества поставленной на тестирование новой функциональности, которая ранее не тестировалась, тестирования новой функциональности на соответствие документации, проверки всевозможных взаимодействий ранее реализованной функциональности с новыми модулями и функциями.

#### **AT (Acceptance Test)**

AT предполагает полное тестирование системы или ее части как на корректных (Positive Test), так и на некорректных данных/сценариях (Negative Test).

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

**Smoke + NFT<sub>AT</sub>(4) + DV + RT<sub>mat</sub>(1, 2, 3)**

#### **DV (Defect Validation)**

В условии задания было указано, что в данной поставке исправлены дефекты, обнаруженные ранее. Данный этап тестирования необходим как раз для проверки исправления дефектов.

## **RT (Regression Test)**

RT 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, и др.

Следует обратить внимание на то, что ранее уже проводилась полная проверка программного продукта. Поэтому, в данном случае регрессионное тестирование (тестирование старых функциональностей) может быть произведено только для корректных входных значений (позитивное тестирование (MAT)).

**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**

## **ITTLT (Internationalization Testing)**

Тестирование адаптации продукта к языковым и культурным особенностям целого ряда регионов, в которых потенциально может использоваться продукт.

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

**Smoke + PLT**

## **PLT (Performance and Load Testing)**

Нагрузочное тестирование — вид тестирования производительности, проводимый с целью оценки поведения компонента или системы при

возрастающей нагрузке, например количестве параллельных пользователей и/или операций, а также определения, какую нагрузку может выдержать компонент или система. Данный модуль необходим для оценки поведения системы при 2000 пользователей.