# ЛАБОРАТОРНАЯ РАБОТА №2

# Тема лабораторной работы: работа с классификацией видов тестирования. Задания к лабораторной работе:

# 1. Выбрать ранее разработанный программный проект.

# 2. Провести анализ и составить отчет со следующей структурой:

# a. Цель работы.

# b. Описание программного проекта.

# c. По каждому из элементов классификации, описанному в разделе методических рекомендаций, привести список видов, типов, методов, уровней тестирования, применимых для выбранного проекта.

# d. Привести примеры дефектов, характерных для каждого вида тестирования.

# e. Выводы по работе.

# f. Список использованных источников.

# 3. Оформить и защитить отчет. Практическая часть:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № Модуля | Функциональный тип тестирования | Результат | Юзабилити | Тестирование | Метод большой ящик |
| Авторизация | применяет логин и пароль |  | Удобно вносить и обрабатывать | +надежность одномоментный вход | + |
| Добавление новых видов продуктов | Id гот. продукции НаименованиеId сырья кол-во сырья | ++++ | ++++ | +цифра;текст-текст;пусто-текст;пусто | + |
| Сырье | Id сырьяТип сырьяОстаток сырья | +++ | +++ | +цифра;текст-текст;пусто-текст;пусто | + |
| Отчет | Из БД попала цифра в отчетпотом в pdf | ++ | -форма отчета+pdf | +при формировании+до 10 сек | + |

В классификации интеграционное тестирование прошло все связи модулей. Авторизация добавления данных преподавателей дисциплины и отчет. Взаимодействие с ос windows так как не установлена программа. Системное тестирование было проверено всеми пользователями. В данном узле нет дефектов. Дефекты выявлены в добавлении данных преподавателя. Стиль отчёта не соответствует юзабилити.

**Контрольные вопросы**

1. Определяется ли качество ПО качеством программного кода?

фактор качества ПО — это нефункциональное требование к программе, которое обычно не описывается в договоре с заказчиком, но, тем не менее, является желательным требованием, повышающим качество программы.

Некоторые из факторов качества:

1. Понятность. Назначение ПО должно быть понятным, из самой программы и документации;
2. Полнота. Все необходимые части программы должны быть Представлены и полностью реализованы;
3. Краткость. Отсутствие лишней, дублирующейся информации. Повторяющиеся части кода должны быть преобразованы в вызов общей процедуры. То же касается и документации;
4. Импортируемость. Лёгкость в адаптации программы к другому окружению: другой архитектуре, платформе, операционной системе или её версии;
5. Согласованность. По всей программе и в документации должны использоваться одни и те же соглашения, форматы и обозначения;
6. Сопровождаемость. Насколько сложно изменить программу для удовлетворения новых требований. Это требование также указывает, что программа должна быть хорошо документирована, не слишком запутана, и иметь резерв роста по использованию ресурсов (память, процессор);
7. Тестируемость. Позволяет ли программа выполнить проверку приёмочных характеристик, поддерживается ли возможность измерения производительности;
8. Удобство использования. Простота и удобство использования программы. Это требование относится прежде всего к интерфейсу пользователя;
9. Надёжность. Отсутствие отказов и сбоев в работе программ, а также простота исправления дефектов и ошибок;
10. Структурированность;
11. Эффективность. Насколько рационально программа относится к ресурсам (память, процессор) при выполнении своих задач;
12. Безопасность.
13. Какие существуют виды тестирования?

Обычно тестирование подразделяется на три категории:

* 1. Функциональное тестирование Модульное (компонентное) Интеграционное Системное Регрессионное Приемочное Смоук
  2. Тестирование производительности Тестирование отказоустойчивости Нагрузочное Объемное Тестирование масштабируемости
  3. Обслуживание (регресс и обслуживание) Регрессионное Тестирование технического обслуживания

1. Какие существуют типы тестирования?

Выделяют 4 основных уровня тестирования:

1. Компонентное/модульное тестирование (Component/Unit Testing)
2. Интеграционное тестирование (Integration Testing)
3. Системное тестирование (System Testing)
4. Приемочное тестирование (Acceptance Testing)
5. Какие существуют методы тестирования?

Широко используемыми методами тестирования являются модульное тестирование, интеграционное тестирование, приемочное тестирование, и тестирование системы. Программное обеспечение подвергается этим испытаниям в определенном порядке.

1. Модульное тестирование
2. Интеграционное тестирование
3. Системное тестирование
4. Приемочные испытания
5. Какие существуют уровни тестирования?

Выделяют 4 основных уровня тестирования:

1. Компонентное/модульное тестирование (Component/Unit Testing)
2. Интеграционное тестирование (Integration Testing)
3. Системное тестирование (System Testing)
4. Приемочное тестирование (Acceptance Testing