

Сопровождение требований

NX Bootcamp

Демонстрация проекта

Демонстрация - цель

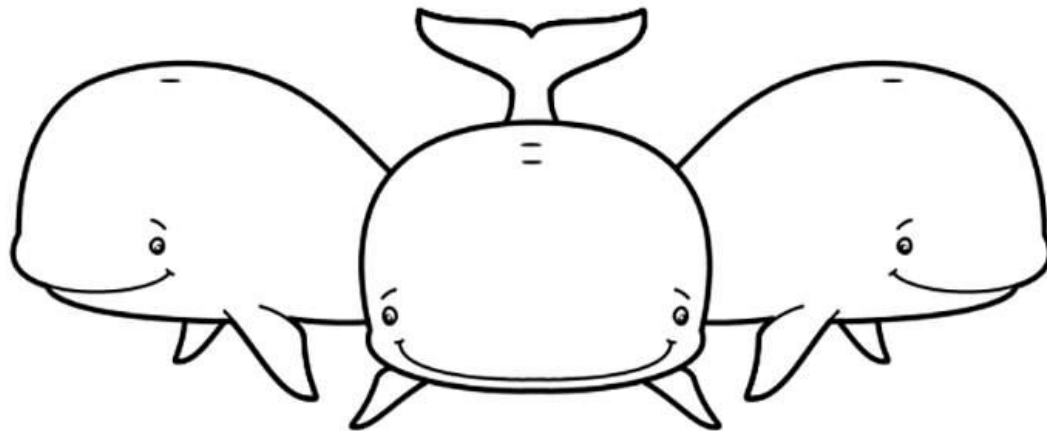
1. Убедить аудиторию
2. Получить обратную связь
3. Ознакомить с информацией



Демонстрация - ключевые составляющие

Презентация – это единство трех элементов:

1. Содержание
2. Оформление
3. Выступление



Демонстрация - содержание

1. Подготовить структуру
 - a. Проблема – Решение – Доводы
 - b. Описание – Вопрос – Обсуждение
 - c. Описание – Детали – Примеры
2. Подготовить текст выступления



Демонстрация - оформление

1. Заголовок на каждом слайде
2. Один слайд - одна мысль
3. Повествование движется только вперед
4. Единство оформления
5. Стандартные шрифты без засечек – Arial, Tahoma, Verdana
6. 2-3 цвета, для всей презентации
7. Нет спецэффектам
8. Изображения, для визуализации и создания образов



Демонстрация - оформление

Сочетания цветов



Демонстрация - выступление

1. Репетиции
2. Укладываться в тайминг
3. Проверить презентацию и оборудование перед выступлением



Внедрение проекта

Способы оповещения

1. Почтовая рассылка
2. Сообщение в месенджере
3. Персональный звонок
4. Через руководителя



Способы обучения

1. Пользовательские инструкции
2. Видеоуроки
3. Оффлайн семинары
4. Вэбинары



Способы запуска

1. Параллельный запуск
2. Глобальное переключение
3. Фичи-флаги



Сбор обратной связи

Способы получения ОС

1. Анкеты
2. Наблюдение
3. Через руководителей
4. Анализ обращений в Тех. поддержку



Анализ обратной связи

Ожидание		Реальность
Отлично	10	Отлично
Отлично	9	Хорошо
Хорошо	8	Нормально
Хорошо	7	Нормально
Нормально	6	Плохо
Нормально	5	Плохо
Плохо	4	Плохо
Плохо	3	Ужасно
Ужасно	2	Ужасно
Ужасно	1	Ужасно

Подготовка тестовых сценариев

Качество программного обеспечения

1061-1998 IEEE Standard for Software Quality Metrics Methodology

качество программного обеспечения – это степень, в которой программное обеспечение обладает требуемой комбинацией свойств.;

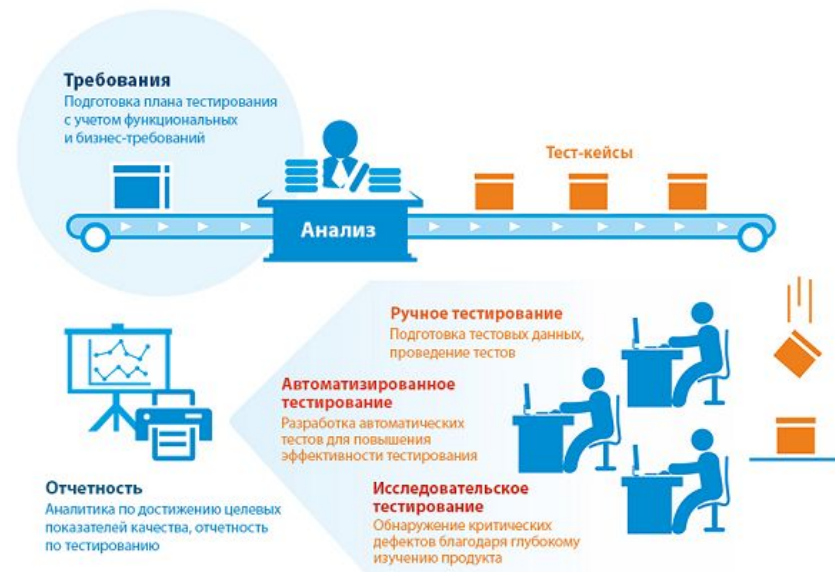
ГОСТ 28806-90 «Качество программных средств. Термины и определения»

качество программного средства – совокупность свойств программного средства (ПС), которые обуславливают его пригодность удовлетворять заданные или подразумеваемые потребности в соответствии с его назначением.

Функциональное тестирование

Функциональное тестирование - проверка того, какие функции программного продукта реализованы, и того, насколько верно они реализованы.

- Ручное тестирование
- Автоматическое тестирование
- Исследовательское тестирование



Нефункциональное тестирование

Нефункциональное – проверяет корректность работы нефункциональных требований.

- Тестирование производительности – проверяет как программный продукт работает под определенной нагрузкой. Включает в себя тестирование:
 - Нагрузочное – определяет, как программа работает под ожидаемой нагрузкой.
 - Стрессовое – определяет работоспособность программного продукта при максимальной нагрузке.
 - Тестирование стабильности – проверяет, выдержит ли программный продукт длительную нагрузку.
 - Конфигурационное – определяет, как программный продукт будет работать в условиях смены конфигурации (платформы, драйверов, компьютеров).

Нефункциональное тестирование

- Тестирование пользовательского интерфейса
- Тестирование удобства использования
- Тестирование защищенности
- Инсталляционное тестирование
- Тестирование совместимости
- Тестирование надежности
- Тестирование локализации

Техники тест дизайна

Эквивалентное Разделение (Equivalence Partitioning - EP)

Анализ Граничных Значений (Boundary Value Analysis - BVA)

Причина / Следствие (Cause/Effect - CE)

Предугадывание ошибки (Error Guessing - EG)

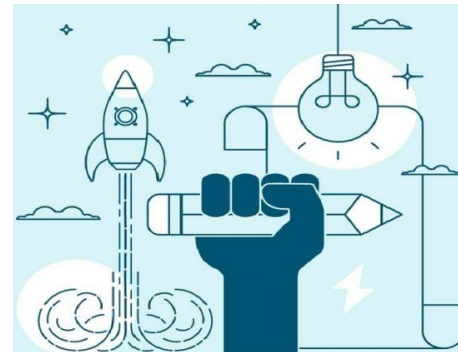
Исчерпывающее тестирование (Exhaustive Testing - ET)

Тестовый случай

Тестовый случай (Test Case) = артефакт, описывающий совокупность шагов, конкретных условий и параметров, необходимых для проверки реализации тестируемой функции или её части.

Практическое задание 024

Каждому слушателю написать минимум два тест кейса для одного UseCase из задания 018



Thank you

Join us!

VK



WhatsApp



Telegram



nxbootcamp@nexign.com