Домашнее задание 2:

1. Опишите 5 видов тестирования на примере интернет магазина

Так как у нас нет спецификации по интернет-магазину или иных разъяснений от заказчика, по идеи функциональное тестирование невозможно.

С другой стороны, ввиду распространенности интернет-магазинов мы можем предположить, что целью заказчика является создание сайта, с помощью

которого можно быстро и просто совершить покупку товара.

Исходя их этого, мы должны провести следующие виды тестирования:

**Смоук тестирование** - например, грузится ли приложение в принципе, можно ли создать аккаунт, добавить товары в корзину и оплатить их.

**Функциональное** - как быстро грузится магазин (клиент не будет ждать больше 10-30 сек.), добавление товара в корзину, регистрация, оформление доставки, оплата, отслеживание. Куда ведут основные ссылки, как работают фильтры, можно ли сделать отмену заказа, почистить корзину, возможна ли оплата картой, есть ли отслеживание после оформления заказа и т.д.

**Пользовательский сценарий (и с ним же UI тестирование)** - возможность поиска товара по каталогу и есть ли классификация товаров в каталоге, можно ли найти товар через строку поиска. Наличие раздела с описанием товара, его параметров, если одежда/обувь то размер и ссылка на таблицу размеров.Можно ли добавить товар в корзину без захода на страницу товара, можно ли оформить покупку без регистрации или зарегистрироваться по упрощенной схеме. Насколько доступно расположена информация о тех.поддержке, в случае проблемы, сможет ли покупатель быстро найти эту информацию и связаться с менеджерами поддержки. формируется ли раздел “рекомендуемые товары” или напоминание какие товары ранее заинтересовали клиента и т.д.

**Позитивное** - проверяем куда ведут ссылки, правильно ли выдается информация при поиске по ключевым словам, добавление товара в корзину, добавление второго товара в корзину увеличивается ли кол-во, можно ли добавить/убрать товары не покидая корзину,

**Негативное** - вводим заведомо некорректные данные, например, пользователь ввел при регистрации неверный символ электронной почты, что произойдет, в корректном виде ли выйдет подсказка или программа зависнет.

P.S. По ощущениям, одни и те же действия по тестированию можно отнести к разным видам.

2. Ответы на тест:

1. Существует четыре вида тестирования:

Модульное - тестирование начального уровня, отдельных модулей кода, как правило выполняется разработчиками, через автотесты.

Интеграционное - это тестирование проверят интеграцию модулей в ПО и их взаимодействие между собой.

Как правило, тесты пишут разработчики, реже тестировщики (очень опытные), тоже зачастую идет автотестирование.

Подходы разные: от простого к сложному, от сложного к простому, либо тестируется все сразу.

Системное - именно на этом этапе работают тестировщики, работа начинается когда закончено юнит и интеграционное тестирование.

Проверяются функциональные и нефункциональные требования в целом, тестируется с позиции бизнес потребности, а не отдельных частей кода.

Приемочное - тоже производится тестировщиками, но носит скорее формальный характер, основная задача этого тестирования приемка ПО заказчиком.

На каждом уровне можно выполнить функциональное тестирование, в том числе с помощью автотестов.

2. Регрессионное тестирование - проверяет, чтобы обновления не внесли ошибок в уже существующий функционал.

Проводится тестировщиками, может быть автоматизировано.

Для уменьшения объема работы тестировщиком и разработчиком определяется какие куски кода подвергались изменениям и именно их тестируют. Либо могут тестироваться наиболее вероятные места для ошибок.

3. Юнит-тесты проверяют отдельные куски кода, тесты пишутся разработчиками, преимущество в автоматизации и скорости проведения этих тестов.

Нельзя ограничиваться только модульным тестирование, по причине проверки только отдельных модулей, а не работы/готовности/соответствия ПО в целом.

4. Буду руководствоваться задачей, которую поставил Заказчик при разработке идеи ПО.

5. Тестирование белого ящика - когда есть доступ к коду. Тестирование черного ящика - когда доступа к коду нет. Соответственно, при тестировании черного ящика невозможно воспользоваться юнит-тестированием.