**Домашнее задание к занятию 1.1. «Введение в тестирование ПО. Методы и виды тестирования»**

Прочтите описание ситуаций, в которых работает тестировщик. Подумайте, какой вид или уровень тестирования стоит применить в каждой из ситуаций. Возможно применять несколько из них.

Для выполнения домашнего задания достаточно написать номер ситуации и название (-я) вида или уровня. Желательно дописать краткое пояснение, почему вы выбрали именно этот вариант. Названия были рассмотрены на занятии.

**Список ситуаций:**

1. Наша команда разработки делает новый сайт, на котором мы сможем смотреть фильмы. Но пока еще дизайнеры не решили, как будет выглядеть сам сайт и разработчик сделал только сам видео-плеер, который потом будет использоваться на сайте. Какой вид или уровень тестирования мы станем применять, если хотим протестировать отдельно этот просмотрщик?

**Ответ: Компонентный или модульный уровень (Component testing)**

1. Команда разработки систем оплаты вместе с новым провайдером интернета сделала новую фишку - при покупке месяца интернета пользователь получает полгода подписки в нашем сервисе в подарок. Наша задача - применить подходящий вид или уровень тестирования для данной ситуации.

**Ответ: Интеграционный уровень (Integration testing)**

1. Наш банк решил сделать приложение для детей, которые учатся пользоваться банковскими картами. Мы целый год делали приложение, старались и вот, наконец, сборка готова, интеграционные тесты пройдены, все, казалось бы, работает и мы почти готовы к настоящему первому релизу в прод. Но перед этим осталось провести самую большую серьезную проверку. Что же мы применим?

**Ответ: Системный уровень (System testing)**

1. Наша компания делает экстеншн для браузера, который помогает переводить слова на странице. Но некоторые пользователи почему-то жалуются, что при обновлении версии экстеншна со старой на новую, он перестает переводить. Что же применить?

**Ответ: Тестирование установки**

1. Дизайнеры решили, что наш дизайн устарел и вот, прошло пара месяцев и мы зарелизили совершенно новый дизайн управления банковскими переводами. Но после этого в техническую поддержку начали писать разозленные пользователи: говорят, что невозможно найти кнопку создания нового перевода, а если и получается найти, ткнуть в нее совершенно невозможно. Но как так, все было по тз! Придется протестировать сервис нам еще раз с этим видом тестирования, чтобы понять, почему юзерам неудобно.

**Ответ: Тестирование пользовательского интерфейса (GUI Testing)**

1. Наш сервис предоставляет пользователям возможность хранить свои фотографии в облаке. Но мы знаем, что наши сервера находятся в Америке, а значит, что их могут вдруг отключить и включить. Наша же задача - сделать так, чтобы пользователь не потерял свои данные в случае отключения или переноса сервера. Какое тестирование применим тут?

**Ответ: Тестирование на отказ и восстановление**

1. Представим, что мы делаем новостной сайт и зарабатываем на рекламе. Но наши аналитики говорят, что не все пользователи видят рекламу. После проверки нашего функционала, мы поняли, что в некоторых браузерах встроено отключение рекламы, но мы не знаем точно в каких и как она отключается так, что не видно и наши баннеры. Какое тестирование мы проведем чтобы понять, где именно не работает функционал?

**Ответ: Санитарное тестирование**

1. Разработчику поступила задача добавить события аналитики к основному функционалу программы. Перед тем, как отдать программу на проверку тестировщику, после изменений разработчик проверяет весь основной функционал на работоспособность. Какой вид тестирования выполняет в данном случае разработчик?

**Ответ: Дымовое тестирование**

1. Представим, что мы релизим наш сайт с новым функционалом. Он уже протестирован, все новинки работают, но перед тем, как выпускать наш сайт на всех пользователей, нам нужно проверить, не сломался ли старый функционал - вдруг новые изменения затронули старое?

**Ответ: Регрессионное тестирование**

1. Мы работаем в компании, которая делает мобильное приложение. Поздний вечер, все разошлись, и разработчик перед уходом прислал нам сборку на тест. Мы ставим ее на телефон, а она внезапно падает на старте. Что ж, сегодня мы уже не сможем протестировать его задачу. Какое тестирование помогло нам понять, что пора и нам уже домой?

**Ответ: Тестирование сборки**

1. Разработчик добавил новый функционал на сервере и теперь сервер умеет отвечать на три новых запроса к нему. Чтобы понять, надо ли приступать к полной проверке с негативными, позитивными и прочими сценариями, мы проверили, что сервер действительно отвечает на эти три запроса, если все сделать как в ТЗ. Какое тестирование мы провели?

**Ответ: Приемочное тестирование**

**\* Дополнительная задача более сложного уровня:**

Есть реальная ситуация из опыта лектора, на основе которой надо сделать выводы на 2-3 предложения. Идеального правильного ответа нет, важно, чтобы вы задумались над тем, как вообще можно применять то, что вы прошли на первом занятии.

Наш сервис, как и многие другие, имел форму авторизации, где можно зайти не только нашим логином и паролем, но и другими сервисами, такими как гугл, вконтакте и так далее.

Судя по регрессам, все работает, но пользователи все жалуются, что вход гуглом не работает. И тут мы понимаем, что дело в том, что если попытаться зайти на сайт не через кнопку гугла, а через кнопку “почта” и ввести там правильные логин и пароль от гугла, юзер видит ошибку, что пароль неверный.

Что из пройденного материала стоило бы использовать более продуктивно, чтобы так не подвести своих пользователей? Опишите эти виды и поясните, почему бы вы выбрали именно их. Если есть какие-то идеи помимо описанных в лекции, расскажите и их тоже. Помните, что наша цель - не только проверить усвоился ли материал, но и попробовать представить себя в типичной для специалиста по качеству ситуации.

**Ответ: Как я понял, пользователи вместо идентификационных данных нашего сервиса, вводили идентификационные данные гугла. В данном случае, я думаю, следовало использовать продуктивнее тестирование пользовательского интерфейса (GUI Testing) и тестирование удобства пользования. Почувствовать себя в роли неопытного пользователя.**