Напишите в свободной форме ответ на указанные вопросы. За каждый ответ можно получить 4 балла.

Пронумерованные ответы отправьте в ваш репозиторий на github и сообщите мне об этом на почту ([draftpodium@gmail.com](mailto:draftpodium@gmail.com)), либо в скайп.

1. Опишите своими словами, как вы понимаете уровни тестирования. Что необходимо указать в ответе - какие бывают виды, кто чаще всего выполняет тесты на конкретном уровне, приведите пример одного вида тестирования, которое можно выполнять на каждом из уровней.
   1. Виды:

Функциональные виды тестирования:

* Функциональное тестирование;
* Тестирование безопасности;
* Тестирование взаимодействия;

Нефункциональные виды тестирования:

Все виды тестирования производительности:

* нагрузочное тестирование;
* стрессовое тестирование;
* тестирование стабильности или надежности;

Тестирование установки;

Тестирование удобства пользования;

Тестирование на отказ и восстановление;

Конфигурационное тестирование;

Связанные с изменениями виды тестирования:

* Дымовое тестирование;
* Регрессионное тестирование;
* Повторное тестирование;
* Тестирование сборки;
* Проверка согласованности/исправности.
  1. Учитывая, что согласно лекции основных уровней 4:

Модульное тестирование – тесты на этом уровне выполняет разработчик,

Интеграционное тестирование - тесты на этом уровне выполняет разработчик и тестировщики (тест-аналитик и тест-дизайнер),

Системное тестирование - тесты на этом уровне выполняют тестировщики (аналитик и дизайнер);

Приемочное тестирование – заказчик и бизнес-аналитик при поддержке тест-аналитика. 2 последние роли можно объединить.

* 1. Вид тестирования, которое можно выполнять на каждом из уровней – регрессионное.

1. Что такое регрессионное тестирование? В каких случаях его необходимо выполнять? Какие особенности есть у этого вида тестирования?
   1. регрессионное тестирование - это выборочное тестирование, позволяющее убедиться, что изменения не вызвали нежелательных побочных эффектов, или что измененная система по-прежнему соответствует требованиям.
   2. Его выполняют чтобы проверить, что изменения, сделанные в приложении или окружающей среде (починка дефекта, слияние кода, миграция на другую операционную систему, базу данных, веб сервер или сервер приложения) не повлияли негативно на ранее разработанную (существующую) функциональность.
   3. Особенности: можно использовать ранее разработанные тест-сценарии, которые при возможности можно автоматизировать; может занять много времени, однако можно оптимизировать с помощью автоматизации, коллективного подхода и оценки рисков.
2. Что такое юнит-тесты? В чем их преимущества? Почему в большинстве случаев нельзя ограничиться только юнит-тестами?
   1. Модульное тестирование, или юнит-тестирование — процесс в программировании, позволяющий проверить на корректность отдельные модули исходного кода программы
   2. Преимущества:

Являются низкоуровневым и фокусируется на маленькой части ПО;

Тесты пишут сами разработчики;

Тесты выполняются очень быстро, можно выполнять тесты несколько раз в минуту;

При разработке можно выполнять не все тесты, а только те, которые необходимы разработчику;

Является отличной предпосылкой к рефакторингу кода.

* 1. Нельзя ограничиться только юнит-тестами, потому что при их выполнении происходит тестирование каждого из модулей по отдельности. Это означает, что ошибки интеграции, системного уровня, функций, исполняемых в нескольких модулях, не будут определены. Кроме того, данная технология бесполезна для проведения тестов на производительность. Таким образом, модульное тестирование более эффективно при использовании в сочетании с другими методиками тестирования

1. Предположим, вам предлагают протестировать какой-то проект (программу, сайт). Чем вы будете руководствоваться, выбирая конкретные виды тестирования для этого проекта?

Я буду руководствоваться целями проекта, его назначением и основными функциями, и возможностями, наличием задокументированных требований и временем отведенным заказчиком.

1. Что такое тестирование черного ящика (Black Box Testing) и чем оно отличается от тестирования белого ящика (White box testing)? Приведите пример вида тестирования, которое невозможно при тестировании черного ящика.
   1. Отличие Black Box Testing: Мы не знаем, как устроена тестируемая система от White box testing: Нам известны все детали реализации тестируемой программы
   2. пример вида тестирования, которое невозможно при тестировании черного ящика: статический анализ кода