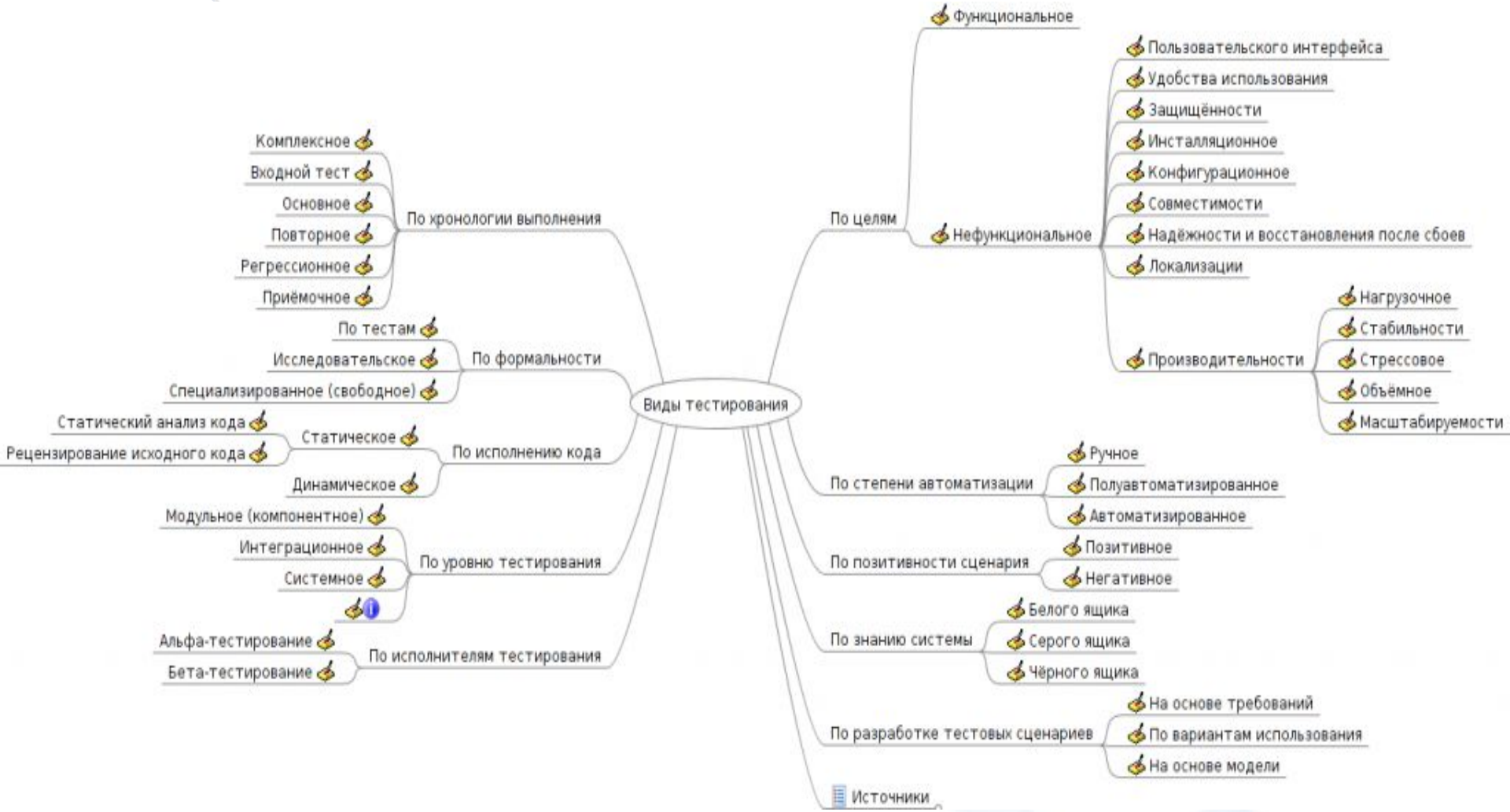
Программа курса “Введение в тестирование ПО” (20 часов)

Октябрь - Ноябрь, 2017

- Виды тестирования



4. Динамическое тестирование

- Динамическое тестирование

- Обзор техник тестирования

Черный ящик (blackbox).

- Обзор техник тестирования

Черный ящик (blackbox).

Используя этот метод, тестировщику не нужно знать внутреннее устройство программы. Объектами тестирования в этом случае являются потоки входных и выходных данных. Это позволяет определять «правильность» работы ПО в соответствии с функциональными требованиями к продукту. Таким образом, критериями тестирования черным ящиком являются:

- Тестирование функций программы.
- Тестирование потока входных данных.
- Тестирование потока выходных данных.
- Тестирование области допустимых значений.
- Тестирование длины набора данных.
- Тестирование порядка входных данных.

- Обзор техник тестирования

Белый ящик (whitebox).

- Обзор техник тестирования

Белый ящик (whitebox).

Работая этим методом, разработчик (тестирование белым ящиком, в основном, осуществляется разработчиком, а не тестировщиком, т.к. необходимо знание внутреннего устройства программы, принципов разработки, программирования и т.д.) проверяет внутреннюю структуру ПО. Объектами тестирования в этом случае являются данные, полученные путем анализа логики программы. Критериями тестирования белым ящиком являются:

- Покрытие операторов.
- Покрытия решений и условий.
- Покрытие комбинаций условий.

Таким образом, тестирование белым ящиком позволяет определять «правильность» работы ПО с точки зрения технических решений.

- Обзор техник тестирования

На основе этих методов существует также тестирование «серым ящиком» (greybox). При работе этим методом подразумевается, что тестировщик имеет доступ к внутреннему устройству программы, но тестирование производит с точки зрения конечного пользователя.

Суть этих методом не сложная, но эффективность тестирования с помощью каждого из них требует хороших знаний и навыков.

- Обзор техник тестирования

Дымовое тестирование или Smoke Testing

Оно применяется для поверхностной проверки всех модулей приложения на предмет работоспособности и наличия быстро находимых критических и блокирующих дефектов. Подвидом дымового тестирования являются Build Verification Testing или Acceptance Testing, выполняемые на функциональном уровне командой тестирования, по результатам которого делается вывод о том, принимается или нет установленная версия программного обеспечения в тестирование, эксплуатацию или на поставку заказчику.

Для облегчения работы, экономии времени и людских ресурсов рекомендуется внедрить автоматизацию тестовых сценариев для дымового тестирования.

- Обзор техник тестирования

Регрессионное тестирование или Regression Testing

Это вид тестирования направленный на проверку изменений, сделанных в приложении или окружающей среде (починка дефекта, слияние кода, миграция на другую операционную систему, базу данных, веб сервер или сервер приложения), для подтверждения того факта, что существующая ранее функциональность работает как и прежде

- Обзор техник тестирования

Как правило, **для регрессионного тестирования используются тест кейсы, написанные на ранних стадиях разработки и тестирования**. Это дает гарантию того, что изменения в новой версии приложения не повредили уже существующую функциональность. Рекомендуется делать автоматизацию регрессионных тестов, для ускорения последующего процесса тестирования и обнаружения дефектов на ранних стадиях разработки программного обеспечения.

Сам по себе термин "Регрессионное тестирование", в зависимости от контекста использования может иметь разный смысл.

Регрессия багов (**Bug regression**) - попытка доказать, что исправленная ошибка на самом деле не исправлена

Регрессия старых багов (**Old bugs regression**) - попытка доказать, что недавнее изменение кода или данных сломало исправление старых ошибок, т.е. старые баги стали снова воспроизводиться.

Регрессия побочного эффекта (**Side effect regression**) - попытка доказать, что недавнее изменение кода или данных сломало другие части разрабатываемого приложения

- Обзор техник тестирования

Тестирование безопасности или Security and Access Control Testing

Тестирование безопасности - это стратегия тестирования, используемая для проверки безопасности системы, а также для анализа рисков, связанных с обеспечением целостного подхода к защите приложения, атак хакеров, вирусов, несанкционированного доступа к конфиденциальным данным.

Принципы безопасности программного обеспечения

Общая стратегия безопасности основывается на трех основных принципах:

- конфиденциальность
- целостность
- доступность

- Обзор техник тестирования

Конфиденциальность

Конфиденциальность - это сокрытие определенных ресурсов или информации. Под конфиденциальностью можно понимать ограничение доступа к ресурсу некоторой категории пользователей, или другими словами, при каких условиях пользователь авторизован получить доступ к данному ресурсу.

- Обзор техник тестирования

Целостность

Существует два основных критерия при определении понятия целостности:

- **Доверие**. Ожидается, что ресурс будет изменен только соответствующим способом определенной группой пользователей.
- **Повреждение и восстановление**. В случае когда данные повреждаются или неправильно меняются авторизованным или не авторизованным пользователем, вы должны определить на сколько важной является процедура восстановления данных.

- Обзор техник тестирования

Доступность

Доступность представляет собой требования о том, что ресурсы должны быть доступны авторизованному пользователю, внутреннему объекту или устройству. Как правило, чем более критичен ресурс тем выше уровень доступности должен быть.

- Обзор техник тестирования

Виды уязвимостей
В настоящее время наиболее распространенными видами **уязвимости в безопасности программного обеспечения** являются:

XSS (Cross-Site Scripting) - это вид уязвимости программного обеспечения (Web приложений), при которой, на генерированной сервером странице, выполняются вредоносные скрипты, с целью атаки клиента.

- Обзор техник тестирования

XSRF / CSRF (Request Forgery) - это вид уязвимости, позволяющий использоваться недостатки HTTP протокола, при этом злоумышленники работают по следующей схеме: ссылка на вредоносный сайт установливается на странице, пользующейся доверием у пользователя, при переходе по вредоносной ссылке выполняется скрипт, сохраняющий личные данные пользователя (пароли, платежные данные и т.д.), либо отправляющий СПАМ сообщения от лица пользователя, либо изменяет доступ к учетной записи пользователя, для получения полного контроля над ней.

- Обзор техник тестирования

Code injections (SQL, PHP, ASP и т.д.) - это вид уязвимости, при котором становится возможно осуществить запуск исполняемого кода с целью получения доступа к системным ресурсам, несанкционированного доступа к данным либо выведения системы из строя.

Server-Side Includes (SSI) Injection - это вид уязвимости, использующий вставку серверных команд в HTML код или запуск их напрямую с сервера.

Authorization Bypass - это вид уязвимости, при котором возможно получить несанкционированный доступ к учетной записи или документам другого пользователя

В о п р о с ы ?