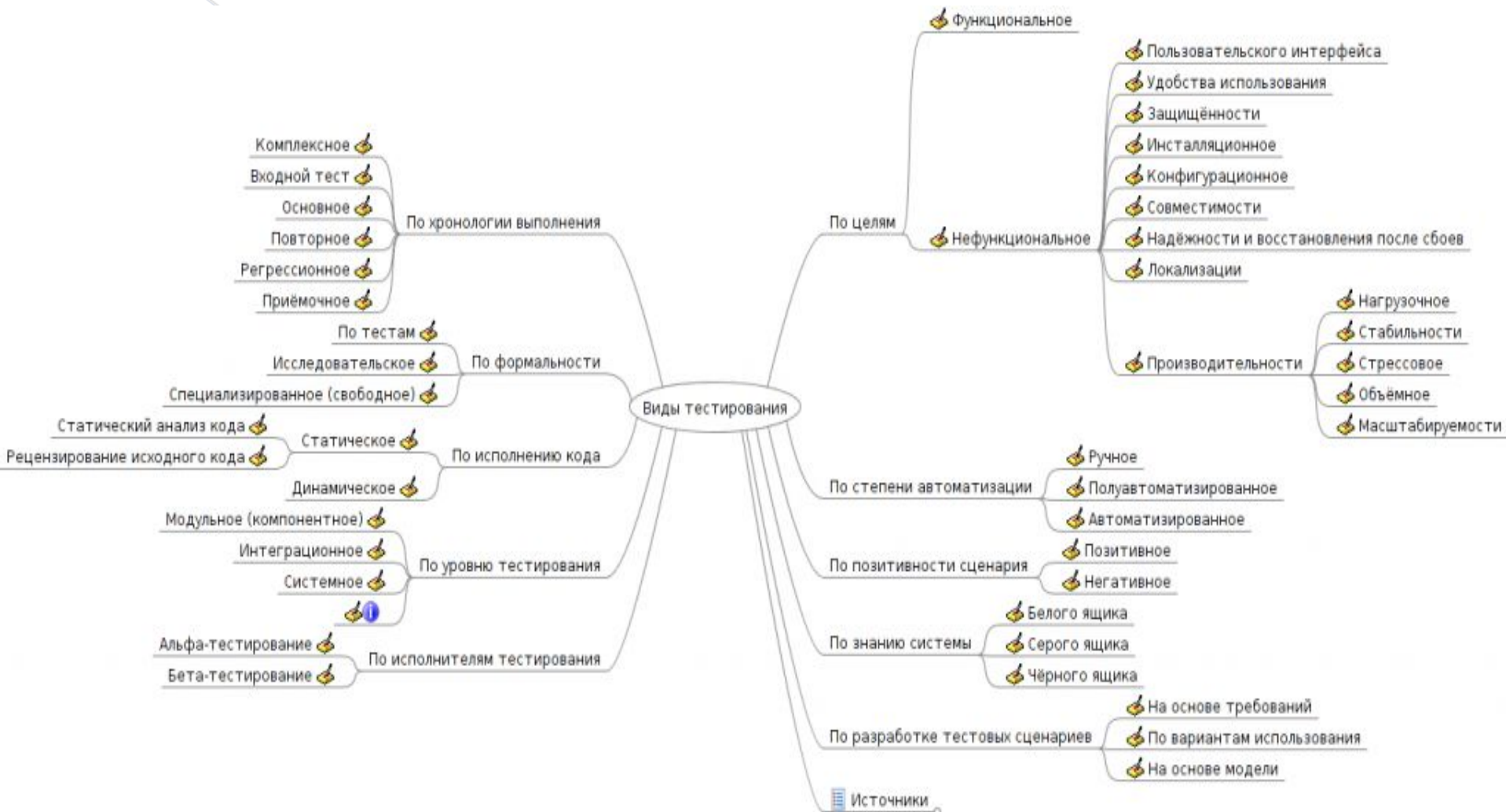


Программа курса “Введение в тестирование ПО” (20 часов)

Октябрь - Ноябрь, 2017



- Виды тестирования



3. Статическое тестирование

- Статическое тестирование



- Статическое тестирование

Статическое тестирование заключается в изучении свойств программы, **не выполняя ее код.**

Оно может быть использовано как для нахождения дефектов еще на этапах проектирования и моделирования, так и проверки исходного кода программы (code review) на предмет использованных конструкций, алгоритмов, отношений модулей и компонентов программы и т. п. Во время статического тестирования можно получить информацию о свойствах программы: ее безопасность, возможность поддержки, надежность и эффективность.

- Процесс ревью, использование инструментальных средств для статического анализа

С помощью **code review** на раннем этапе могут быть выявлены ошибки в коде продукта. Как правило code review производится самими разработчиками.

Code review - инженерная практика в терминах гибкой методологии разработки. Это анализ (инспекция) кода с целью выявить ошибки, недочеты, расхождения в стиле написания кода, в соответствии написанного кода и поставленной задачи.

- Процесс ревью, использование инструментальных средств для статического анализа

К очевидным плюсам этой практики можно отнести:

- Улучшается качество кода
- Находятся «глупые» ошибки (опечатки) в реализации
- Повышается степень совместного владения кодом
- Код приводится к единому стилю написания
- Хорошо подходит для обучения «новичков», быстро набирается навык, происходит выравнивание опыта, обмен знаниями

- Процесс ревью, использование инструментальных средств для статического анализа

Примерами ошибок, которые потенциально можно выявить с помощью автоматического статического тестирования, могут быть:

- утечки ресурсов (утечки памяти, неосвобождаемые файловые дескрипторы и т.д.)
- возможность переполнения буфера (buffer overflows)
- ситуации частичной (неполной) обработки ошибок

- Процесс ревью, использование инструментальных средств для статического анализа

Конкретные примеры инструментальных средств статического тестирования для Java: Checkstyle, FindBugs, PMD (Programming Mistake Detector), для Javascript: JSLint.

4. Динамическое тестирование

- Динамическое тестирование



- **Динамическое тестирование**

Динамическое тестирование производится путем **запуска** продукта и проверки его функционала. Проверка осуществляется с помощью ручного или автоматического выполнения **заранее подготовленного набора тестов.**

Динамическое тестирование помогает определить свойства программы по результатам ее выполнения – соответствие функциональным требованиям, работа в разных средах, разных версий, локализация, нагрузка, работа с входными и выходными данными, – типов динамического тестирования очень много. Поэтому, в большинстве случаев, говоря слово «тестирование» подразумевается именно динамическое.

- Процесс разработки тестов

Тест план (Test plan)



Тест дизайн (Test design)



Тестовый случай (Test case)

- Процесс разработки тестов

Тест план (Test Plan) - это документ, описывающий весь объем работ по тестированию, начиная с описания объекта, стратегии, расписания, критериев начала и окончания тестирования, до необходимого в процессе работы оборудования, специальных знаний, а также оценки рисков с вариантами их разрешения.

- Процесс разработки тестов

Хороший тест план должен как минимум отвечать на следующие вопросы:

Что надо тестировать?

описание объекта тестирования: системы, приложения, оборудование

Что будете тестировать?

список функций и описание тестируемой системы и её компонент в отдельности

Как будете тестировать?

стратегия тестирования, а именно: виды тестирования и их применение по отношению к тестируемому объекту

Когда будете тестировать?

последовательность проведения работ: подготовка (Test Preparation), тестирование (Testing), анализ результатов (Test Result Analysis) в разрезе запланированных фаз разработки

Критерии начала тестирования:

готовность тестовой платформы (тестового стенда)

законченность разработки требуемого функционала

наличие всей необходимой документации

...

Критерии окончания тестирования:

результаты тестирования удовлетворяют критериям качества продукта

требования к количеству открытых багов выполнены

выдержка определенного периода без изменения исходного кода приложения **Code Freeze** (CF)

выдержка определенного периода без открытия новых багов **Zero Bug Bounce** (ZBB)

- Процесс разработки тестов

Тест дизайн – это этап процесса тестирования ПО, на котором проектируются и создаются тестовые случаи (тест кейсы), в соответствии с определёнными ранее критериями качества и целями тестирования.

План работы над тест дизайном

- анализ имеющихся проектных артефактов: документация (спецификации, требования, планы), модели, исполняемый код и т.д.
- написание спецификации по тест дизайну (Test Design Specification)
- проектирование и создание тестовых случаев (Test Cases)

- Процесс разработки тестов

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

Под тест кейсом понимается структура вида:

Action > Expected Result > Test Result

Action	Expected Result	Test Result (passed/failed/blocked)
Open page "login"	Login page is opened	Passed

- Процесс разработки тестов

Виды Тестовых Случаев

Тест кейсы разделяются по ожидаемому результату на **позитивные и негативные:**

Позитивный тест кейс использует только корректные данные и проверяет, что приложение правильно выполнило вызываемую функцию.

Негативный тест кейс оперирует как корректными так и некорректными данными (минимум 1 некорректный параметр) и ставит целью проверку исключительных ситуаций (срабатывание валидаторов), а также проверяет, что вызываемая приложением функция не выполняется при срабатывании валидатора.

- Процесс разработки тестов

Структура Тестовых Случаев (Test Case Structure)

Каждый тест кейс должен иметь 3 части:

PreConditions	Список действий, которые приводят систему к состоянию пригодному для проведения основной проверки. Либо список условий, выполнение которых говорит о том, что система находится в пригодном для проведения основного теста состоянии.
Test Case Description	Список действий, переводящих систему из одного состояния в другое, для получения результата, на основании которого можно сделать вывод о удовлетворении реализации, поставленным требованиям
PostConditions	Список действий, переводящих систему в первоначальное состояние (состояние до проведения теста - initial state)

- Процесс разработки тестов

Вывод

Для того чтобы команда тестирования работала сплоченно и не отвлекалась по вопросам оформления тест кейсов, у всех должен быть **единый шаблон или подход к их написанию.**

Вопросы?

