

Лабораторная работа №25

Тестирование и отладка программного продукта

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Изучить краткие теоретические сведения по теме «Тестирование и отладка программного продукта».

Выполнить задание №1_Изучение стратегий, уровней и видов тестирования.

Таблица 1-Стратегии тестирования (По доступу к коду и архитектуре приложения)э

№	Название стратегии	Сущность (характеристика стратегии)	Преимущества	Недостатки	Кто осуществляет
1	Метод белого (прозрачного) ящика	Метод, который предполагает, что внутренняя структура / устройство/реализация системы известны тестирующему.	<ul style="list-style-type: none">• Оптимизация кода путем нахождения скрытых ошибок• Доступность структуры кода позволяет выбрать тип входных данных, необходимых для эффективного тестирования• Возможность автоматизирования тест-кейсов	<ul style="list-style-type: none">• Знание кода и внутренней структуры является необходимым условием, для проведения тестирования. Требуется квалифицированный тестировщик, что увеличивает стоимость• И почти невозможно изучить каждый кусок кода, чтобы обнаружить скрытые ошибки, что может создать проблемы, приводящие к сбою приложения	Квалифицированный тестировщик, разбирающийся в коде
2	Метод черного ящика	Стратегия тестирования, при которой не используется знание о внутреннем устройстве (коде) тестируемого объекта.	<ul style="list-style-type: none">• Простота: облегчает тестирование проектов высокого уровня и сложных приложений• Экономия ресурсов: тестеры сосредоточены на функциональности программного обеспечения• Тестовые случаи: Сосредоточение	<ul style="list-style-type: none">• В действительности выполняется избранное число тестовых сценариев, результатом чего является ограниченный охват• Отсутствие четкой спецификации затрудняет разработку тестовых сценариев• Низкая эффективность	Пользователь

			<p>внимания на функциональности программного обеспечения для облегчения быстрой разработки тестовых случаев</p> <ul style="list-style-type: none"> • Обеспечивает гибкость: специальные знания программирования не требуются 		
3	Метод серого ящика	<p>Тестирование серого ящика предусматривает частичную осведомленность о внутренних процессах. Данный метод - это комбинация двух предыдущих подходов.</p>	<ul style="list-style-type: none"> • Позволяет быстро выявить ошибки в функциональных спецификациях • Тестировщику не нужна дополнительная квалификация • Тестирование проходит «с позиции» пользователя • Составлять тест-кейсы можно сразу после подготовки спецификации 	<ul style="list-style-type: none"> • Связывание дефектов затруднено, когда тестирование серого выполняется для распределенных систем • Ограниченный доступ к внутренней структуре приводит к ограниченному доступу для обхода пути кода • Поскольку доступ к исходному коду невозможен, полное тестирование белого ящика невозможно • Тестирование серая коробка не подходит для тестирования алгоритма • Большинство тестовых случаев сложно спроектировать 	Кодер с позиции пользователя

Таблица 2 – уровни тестирования

№	Название уровня	Сущность (характеристика уровня)	Когда осуществляется	Кто осуществляет
1	Модульное (компонентное)	<p>Тестирование программы на уровне отдельно взятых модулей, функций или классов.</p> <p>Цель модульного тестирования состоит в выявлении локализованных в модуле ошибок в реализации алгоритмов, а также в определении степени готовности</p>	<p>Когда нужно проверить функциональность определенного куска кода (функцию, процедуру) за раз.</p>	Программист

		системы к переходу на следующий уровень разработки и тестирования.		
2	Интеграционное	Отдельные программные модули объединяются и тестируются в группе. Обычно интеграционное тестирование проводится после модульного тестирования и предшествует системному тестированию.	После проверки работы всех функций\процедур в приложении, когда нужно работать их взаимодействие друг с другом.	Программист
3	Системное	Уровень тестирования, который проверяет законченный и полностью интегрированный программный продукт. Целью системного теста является оценка сквозных технических характеристик системы.	Когда проверяется целостность работы приложения, готовый ПП	Программист, тестировщики
4	Приёмочное	Проверяет соответствие системы требованиям и проводится с целью: <ul style="list-style-type: none"> • определения удовлетворяет ли система приемочным критериям • вынесения решения заказчиком или другим уполномоченным лицом принимается приложение или нет 	Когда нужно удостовериться в том, что ТЗ заказчика выполнено на 100%	Независимый тестировщик
5	Выходное	Это завершающий этап тестирования , на котором проверяется готовность ПП к поставке заказчику. Данный вид тестирования проводит независимый тестировщик.	Когда требуется проверить приложение перед выходом на платформу, его взаимодействие в среде с другими приложениями	Заказчик, программист

Таблица 3 – Виды тестирования

№	Название вида	Сущность (характеристика вида)
1	<u>По запуску кода на исполнение:</u> 1.1 Статическое 1.2 Динамическое	1.1 Тестирование без запуска кода на исполнение. Оно представляет собой процесс или технику, которые выполняются для поиска потенциальных дефектов в программном обеспечении. 1.2 Тип тестирования, который предполагает запуск программного кода. Таким образом, анализируется поведение программы во время ее работы.
2	<u>Функциональные виды</u> 2.1 Функциональное 2.2 Тестирование безопасности	2.1 Это тестирование ПО в целях проверки реализуемости функциональных требований, то есть способности ПО в определённых условиях решать задачи, нужные пользователям. 2.2 Оценка уязвимости программного обеспечения к различным атакам. 2.3 Проверяет способность приложения взаимодействовать с одним и

	2.3 Тестирование взаимодействия	более компонентами или системами и включающее в себя тестирование совместимости и интеграционное тестирование.
3	<u>Нефункциональные виды:</u> 3.1 Тестирование производительности 3.2. Тестирование установки 3.3. Тестирование удобства пользования 3.4. Тестирование на отказ и восстановление 3.5. Конфигурационное тестирование	3.1 Проводится с целью определения, как быстро работает вычислительная система или её часть под определённой нагрузкой. 3.2 Направленно на проверку успешной инсталляции и настройки, а также обновления или удаления программного обеспечения. 3.3 Направленный на установление степени удобства использования, обучаемости, понятности и привлекательности для пользователей разрабатываемого продукта в контексте заданных условий. 3.4 Проверяет тестируемый продукт с точки зрения способности противостоять и успешно восстанавливаться после возможных сбоев. 3.5 специальный вид тестирования, направленный на проверку работы программного обеспечения при различных конфигураций системы.
4	<u>Связанные с изменениями</u> 4.1. Дымовое тестирование 4.2. Регрессионное тестирование 4.3. Тестирование сборки 4.4. Санитарное тестирование или проверка согласованности/исправности	4.1 Минимальный набор тестов на явные ошибки. 4.2 Это проверка ранее протестированной программы, позволяющая убедиться, что внесенные изменения не повлекли за собой появления дефектов в той части программы, которая не менялась. 4.3 Тестирование направленное на определение соответствия, выпущенной версии, критериям качества для начала тестирования. 4.4 Узконаправленное тестирование достаточное для доказательства того, что конкретная функция работает согласно заявленным в спецификации требованиям.
5	<u>По степени автоматизации:</u> 5.1. Ручное тестирование 5.2. Автоматизированное тестирование	5.1 Производится без использования программных средств для проверки программы или сайта путём моделирования действий пользователя. 5.2 Использует программные средства для выполнения тестов и проверки результатов выполнения, что помогает сократить время тестирования и упростить его процесс.
6	<u>По времени проведения:</u> 6.1. Альфа-тестирование 6.2. Бета-тестирование	6.1 Это тип приемочных испытаний; выполняется для выявления всех возможных проблем / ошибок перед выпуском продукта для обычных пользователей или общественности. 6.2 Интенсивное использование почти готовой версии продукта (как правило, программного или аппаратного обеспечения) с целью выявления максимального числа ошибок в его работе для их последующего устранения перед окончательным выходом продукта на рынок, к массовому потребителю.

Задание 2: Тестирование программного продукта методом белого ящика.

№	Наименование функции	Описание тестового сценария с исходными данными	Ожидаемый результат
1	Переход по страницам сайта	1. Нажатие на любую ссылку в навигации сайта 2. Открытие нужной страницы	Открытие нужной страницы
2	Просмотр товаров определенной категории	1. Выбор категории товара в выпадающем списке категорий 2. Отображение товаров соответствующей категории	Отображение товаров соответствующей категории
3	Просмотр товаров определенного магазина	1. Выбор магазина 2. Отображение товаров, которые находятся в данном магазине	Отображение товаров, которые находятся в данном магазине
4	Вывод списка магазинов, где есть в наличии данный продукт	1. Выбор продукта 2. Открытие карточки продукта 3. Вывод таблицы с наличием и	Вывод таблицы с наличием и ценой в магазинах

		ценой в магазинах	
5	Регистрация пользователя	1. Заполнение данных в форму регистрации 2. Добавление пользователя в БД	Добавление пользователя в БД
6	Поиск по сайту	1. Пользователь нажимает на поле поиска вверху страницы 2. Начинает вводить строку 3. Появляются возможные варианты по введенной строке 4. Пользователь нажимает на предложенный вариант 5. Открывается карточка товара	Открывается карточка товара