|  |  |  |
| --- | --- | --- |
| Дата | Содержание выполняемой работы | Подпись руководите ля |
| 25.05.2022,  26.05.2022 | **Практическая работа №25-26**  **Тема:** Тестирование программной системы.  **Наименование работы:** Формирование тестов. Проведение тестирования программной системы.  **Цель:** Закрепление умений и навыков по верификации и валидации программного средства, тестирование программного средства.  **Задание 1.**  Прочитал и законспектировал в дневник-отчет теоретический материал. (Определения верификации и тестирования; характеристики качества ПО, задачи верификации в рамках жизненного цикла ПО).  Тестирование программного обеспечения — проверка соответствия реальных и ожидаемых результатов поведения программы, проводимая на конечном наборе тестов, выбранном определённым образом.  Верификация — это процесс оценки системы, чтобы понять, удовлетворяют ли результаты текущего этапа разработки условиям, которые были сформулированы в его начале.  Такой процесс формальной проверки или верификации может доказать, что дефекты отсутствуют, с точки зрения используемого метода. (То есть нет никакой возможности точно установить или гарантировать отсутствие дефектов в программном продукте с учётом человеческого фактора, присутствующего на всех этапах жизненного цикла ПО).  Существует множество подходов к решению задачи тестирования и верификации ПО, но эффективное тестирование сложных программных продуктов — это процесс в высшей степени творческий, не сводящийся к следованию строгим и чётким процедурам или созданию таковых.  С точки зрения ISO 9126, Качество (программных средств) можно определить как совокупную характеристику исследуемого ПО с учётом следующих составляющих:  • Надёжность  • Сопровождаемость  • Практичность  • Эффективность  • Мобильность  • Функциональность  **Задание 2.**  Изучил и законспектировал методы тестирования: черного и белого ящика. Проанализировал и записал в дневник-отчет достоинства и недостатки каждого метода. Сделал вывод о применимости данных методов для тестирования вашего ПП.  В терминологии профессионалов тестирования (программного и некоторого аппаратного обеспечения), фразы «тестирование белого ящика» и «тестирование чёрного ящика» относятся к тому, имеет ли разработчик тестов доступ к исходному коду тестируемого ПО, или же тестирование выполняется через пользовательский интерфейс либо прикладной программный интерфейс, предоставленный тестируемым модулем.  При тестировании белого ящика разработчик теста имеет доступ к исходному коду программ и может писать код, который связан с библиотеками тестируемого ПО. Это типично для юнит-тестирования, при котором тестируются только отдельные части системы. Оно обеспечивает то, что компоненты конструкции — работоспособны и устойчивы, до определённой степени. При тестировании белого ящика используются метрики покрытия кода.  При тестировании чёрного ящика тестировщик имеет доступ к ПО только через те же интерфейсы, что и заказчик или пользователь, либо через внешние интерфейсы, позволяющие другому компьютеру либо другому процессу подключиться к системе для тестирования. Например, тестирующий модуль может виртуально нажимать клавиши или кнопки мыши в тестируемой программе с помощью механизма взаимодействия процессов, с уверенностью в том, все ли идёт правильно, что эти события вызывают тот же отклик, что и реальные нажатия клавиш и кнопок мыши. Как правило, тестирование чёрного ящика ведётся с использованием спецификаций или иных документов, описывающих требования к системе. Как правило, в данном виде тестирования критерий покрытия складывается из покрытия структуры входных данных, покрытия требований и покрытия модели (в тестировании на основе моделей).  Для моего программного продукта я считаю важным тестирование обоими способами, так как это позволит найти больше недостатков или багов и выявить их до выпуска в эксплуатацию.  **Задание 3.**  Просмотрел презентацию Разработка тестов. Законспектировал в дневник-отчетов основные моменты по разработке тестов. Просмотрел файл Пример разработки тестов. Используя шаблон для разработки тестов, составил тестовые сценарии функционального тестирования для своего приложения и протестировал его. Оформил в Приложении 25.1. |  |