Лабораторная работа №5

“ТЕСТИРОВАНИЕ ПРИЛОЖЕНИЯ С СОСТАВЛЕНИЕМ ПЛАНА ТЕСТИРОВАНИЯ И ОТЧЕТА О ТЕСТИРОВАНИИ”

**Цель работы:** изучить классификацию видов тестирования, практически закрепить эти знания путем генерации тестов различных видов, научиться планировать тестовые активности в зависимости от специфики поставляемой на тестирование функциональности.

**Тест-план для графического редактора «Рисовалка»:**

1. **Цель:** Целью составления данного тест плана является описание процесса тестирования графического редактора Paint. Данный документ позволяет получить представление о плановых работах, сроках.
2. **Области подвергаемые тестированию:**

* Установка приложения
* Обновление приложения
* Удаление приложения
* Запуск
* Запуск от имени администратора
* Загрузка файлов различных форматов (bmp, gif, png, jpeg)
* Создание файла (создание графического изображения, ввод текста)
* Масштабирование
* Инструменты и функции меню
* Сохранение файла
* Проверка готовности к печати

**3. Области не подвергаемые тестированию:**

* Использование библиотек графических процедур

**4. Тестовая стратегия и подходы:**

Будут применяться следующие виды тестирования: позитивное тестирование, негативное тестирование, функциональное тестирование, тестирование интерфейса, тестирование настольного приложения, тестирование удобства использования.

Планируется два этапа проведения процесса тестирования. Первый этап заключается в составлении чек-листа, составлении тест-плана, а также проведении функциональных тестов. Второй этап будет посвящен детализации функционального чек-листа и детальному прогону функциональных тестов с выявлением и описанием дефектов. Таким образом, достигается максимальная детализация глубины тестирования, что, в свою очередь, позволяет более точно определить затрачиваемые ресурсы.

На первом этапе будут применяться виды тестирования для всей системы, то есть функциональное тестирование и тестирование пользовательского интерфейса при которых будут определяться и конфигурироваться тестовые среды.

К началу второго этапа будет сформирован чек-лист по функциональному тестированию. На втором этапе будут производиться тестирование для отдельных областей: негативное и позитивное тестирования.

**5.** **Критерии начала и окончания тестирования:**

**Тестирование может быть начато, если выполнены следующие условия:**

* Готова и утверждена необходимая документация;
* Тестируемый функционал окончен и готов для передачи в тестирование.

**Тестирование окончено, если выполнены следующие условия:**

* Все найденные дефекты задокументированы.

**6. Ресурсы:**

* ***Аппаратные ресурсы:***

|  |  |
| --- | --- |
| **Устройства** | ASUS TUF Gaming F15 FX506LI-HN012  ASUS X509FB-BQ259T  ASUS VivoBook X571LI\_X571LI  ACER WIN-7SDG9ARNK3I  ASUS ROG Strix GA15 G15DK-R5600X0770  Lenovo RedAurochInc-1  ASUS TUF Gaming A15 FA506 |
| **Разрешение экранов** | 1920x1080 |

* ***Временные ресурсы:***

3 недели

* ***Программные ресурсы:***

|  |  |
| --- | --- |
| **Операционные системы** | Windows10, Windows11 |

* ***Человеческие ресурсы:***

|  |  |
| --- | --- |
| **TeamLead** | 1 специалист |
| **Специалист по тестированию ПО** | 2 специалиста |
| **Тестировщик ПО** | 5 специалистов |

* ***Финансовые ресурсы:***

2000$

**7. Расписание**

|  |  |  |
| --- | --- | --- |
| Время | Задачи | Решение/Описание |
| 2 часа | Провести интервью с заказчиком и принять заказ | Проводится интервью с заказчиком, где выявляются его требования |
| 4 часа | Составить тест-план | Встреча с командой, определение времени и средств, при которых возможна реализация продукта |
| 2 часа | Протестировать работоспособность | Тестирование основной части приложения |
| 1 час | Составить отчёт по выполненной работе | Описание выполненной работы |
| 2 часа | Протестировать интерфейс | Тестирование пользовательского интерфейса |
| 1 час | Составить отчёт по выполненной работе | Описание выполненной работы |
| 30 минут | Связаться с заказчиком и выполнить правки, если таковые требуются | Связь с заказчиком и уточнение требований, если они имеются, то внести правки в тест план |
| 1 час | Провести финальный тест графического редактора | Тестирование всего функционала графического редактора |
| 1 час | Составить финальный отчёт | Описание проделанной работы за все время тестирования |
| 30 минут | Провести проверку отчёта | Проверка правильности выполнения отчета |
| 1 час | Провести совещание | Проводится совещание команды для финального обсуждения заказа перед сдачей |
| 1 час | Сдать заказ | Проверка клиентом правильности выполнения заказа, если его все устраивает, то происходит оплата с организатором |

**8. Роли и ответственность:**

|  |  |
| --- | --- |
| **ФИО** | **Роль** |
| Ясюкевич Полина | TeamLead |
| Богдан Екатерина | Специалист по тестированию ПО |
| Войтехович Анастасия | Специалист по тестированию ПО |
| Найбич Павел | Тестировщик ПО |
| Швед Матвей | Тестировщик ПО |
| Булохов Егор | Тестировщик ПО |
| Ермолович Рада | Тестировщик ПО |
| Шваб Вероника | Тестировщик ПО |

**9.** **Риски и ограничения:**

* Ввиду ограниченности ресурсов, не можем гарантировать корректное отображение графического редактора, корректную работу с изображением и добавлением уже существующих на мобильных устройствах,
* Человеческий фактор.

**10. Документация:**

После окончания тестирования планируется наличие таких документов:

- тест-план;

- чек-лист;

- баг-репорт

**Главные выводы:** Конечным итогом проведения тестирования должен стать оформленный конечный результат процесса тестирования с описанными дефектами, а также рекомендациями по улучшению продукта с точки зрения конечного пользователя.

**Контрольные вопросы:**

1. **Какие существуют типы тестов по покрытию? Дайте характеристику каждому.**

Типы тестов по покрытию (по глубине):

**Smoke test** – тестирование системы для определения корректной работы базовых функций программы в целом, без углубления в детали. При проведении теста определяется пригодность сборки для дальнейшего тестирования.

**Minimal Acceptance Test (MAT, Positive test**) - тестирование системы или ее части только на валидных данных (валидные данные – это данные, которые необходимо использовать для корректной работы модуля/функции).

**Acceptance Test (AT) -** полное тестирование системы или ее части как на корректных, так и на некорректных данных/сценариях. Вид теста, подтверждающий, что приложение может использоваться по назначению при любых условиях.

1. **Какие существуют тестовые активности? Дайте характеристику каждому.**

Тестовые активности (типы тестов по покрытию (по ширине)):

**Defect Validation** – проверка результата исправления дефектов. Включает в себя проверку на воспроизводимость дефектов, которые были исправлены в новой сборке продукта, а также проверку того, что исправление не повлияло на ранее работавшую функциональность

**New Feature Test (NFT, AT of NF)** – определение качества поставленной на тестирование новой функциональности, которая ранее не тестировалась. Данный тип тестирования включает в себя: проведение полного теста (АТ) непосредственно новой функциональности; тестирование новой функциональности на соответствие документации; проверку всевозможных взаимодействий ранее реализованной функциональности с новыми модулями и функциями.

**Regression testing (регрессионное тестирование**) – проводится с целью оценки качества ранее реализованной функциональности. Включает в себя проверку стабильности ранее реализованной функциональности после внесения изменений, например, добавления новой функциональности, исправление дефектов, оптимизация кода, разворачивание приложения на новом окружении.

1. **Какие существуют типы тестов знанию кода? Дайте характеристику каждому.**

**Черный ящик** – тестирование системы, функциональное или нефункциональное, без знания внутренней структуры и компонентов системы. У тестировщика нет доступа к внутренней структуре и коду приложения либо в процессе тестирования он не обращается к ним.

**Белый ящик** – тестирование, основанное на анализе внутренней структуры компонентов или системы. У тестировщика есть доступ к внутренней структуре и коду приложения.

**Серый ящик** – комбинация методов белого и черного ящика, состоящая в том, что к части кода архитектуры у тестировщика есть, а к части кода – нет.

1. **Какие существуют типы тестов по изолированности компонентов? Дайте характеристику каждому.**

**Unit/component (модульное)** – тестирование отдельных компонентов (модулей) программного обеспечения.

**Integration (интеграционное)** – тестируется взаимодействие между интегрированными компонентами или системами.

**System (системное)** – тестируется работоспособность системы на соответствие установленным требованиям.

1. **Какие существуют типы тестов по месту и времени проведения? Дайте характеристику каждому.**

**User Acceptance Testing (UAT) (приемочное тестирование**) – формальное тестирование по отношению к потребностям, требованиям и бизнес процессам пользователя, проводимое с целью определения соответствия системы критериям приёмки и дать возможность пользователям заказчикам или иным авторизованным лицам определить, принимать систему.

**Alpha Testing (альфа–тестирование)** – моделируемое или действительное функциональное тестирование, выполняется в организации, разрабатывающей продукт, но не проектной командой (это может быть независимая команда тестировщиков, потенциальные пользователи, заказчики). Альфа тестирование часто применяется к коробочному программному обеспечению в качестве внутреннего приемочного тестирования.

**Beta Testing (бета–тестирование)** – эксплуатационное тестирование потенциальными или существующими клиентами/заказчиками на внешней стороне (в среде, где продукт будет использоваться) никак связанными с разработчиками, с целью определения действительно ли компонент или система удовлетворяет требованиям клиента/заказчика и вписывается в бизнес–процессы. Бета–тестирование часто проводится как форма внешнего приемочного тестирования готового программного обеспечения для того, чтобы получить отзывы рынка.

1. **Какие существуют типы функциональных тестов? Дайте характеристику каждому.**

**Дымовое тестирование (smoke test, intake test, build verification test)** направлено на проверку самой главной, самой важной, самой ключевой функциональности, неработоспособность которой делает бессмысленной саму идею использования приложения (или иного объекта, подвергаемого дымовому тестированию). В связи с высокой важностью тест-кейсов на данном уровне пороговое значение метрики их прохождения часто выставляется равным 100 % или близким к 100 %.

**Тестирование критического пути (critical path test)** направлено на исследование функциональности, используемой типичными пользователями в типичной повседневной деятельности. Пороговое значение метрики успешного прохождения «теста критического пути» уже немного ниже, чем в дымовом тестировании, но всё равно достаточно высоко (как правило, порядка 70-80-90 % – в зависимости от сути проекта).

**Расширенное тестирование (extended test)** направлено на исследование всей заявленной в требованиях функциональности – даже той, которая низко проранжирована по степени важности. Пороговое значение метрики успешного прохождения расширенного тестирования существенно ниже, чем в тестировании критического пути (иногда можно увидеть даже значения в диапазоне 30-50%, т.к. подавляющее большинство найденных здесь дефектов не представляет угрозы для успешного использования приложения большинством пользователей).

1. **Какие существуют типы нефункциональных тестов? Дайте характеристику каждому.**

**Тестирование пользовательского интерфейса (GUI)** – тестирование, выполняемое путем взаимодействия с системой через графический интерфейс пользователя.

**Тестирование удобства использования (Usability Testing)** – тестирование с целью определения степени понятности, легкости в изучении и использовании, привлекательности программного продукта для пользователя при условии использования в заданных условиях эксплуатации, визуальное оформление, навигация, логичность.

**Тестирование доступности (Accessibility testing**) – тестирование, которое определяет степень легкости, с которой пользователи с ограниченными способностями могут использовать систему или ее компоненты.

**Тестирование интернационализации** – тестирование способности продукта работать в локализованных средах

**Тестирование локализации (Localization testing)** – тестирование, проводимое с целью проверить качество перевода продукта с одного языка на другой.

**Тестирование производительности или нагрузочное тестирование** – процесс тестирования с целью определения производительности программного продукта.

1. **Какие этапы составляют процесс тестирования?**
2. Изучение и анализ предмета тестирования
3. Планирование тестирования

3) Выполнение тестирования

1. **Что происходит на этапе изучения и анализа предмета тестирования?**

Изучение и анализ предмета тестирования начинается еще до утверждения спецификации и продолжается на стадии разработки (кодирования) программного обеспечения. Конечной целью этапа изучение и анализ предмета тестирования является получение ответов на два вопроса: какие функциональности предстоит протестировать, как эти функциональности работают.

**10. Что происходит на этапе планирования тестирования?**

Планирование тестирования происходит на стадии разработки (кодирования) программного обеспечения. На стадии планирования тестирования стоит задача поиска компромисса между объемом тестирования, который возможен в теории, и объемом тестирования, который возможен на практике. На данной стадии необходимо ответить на вопрос: как будем тестировать? Результатом планирования тестирования является тестовая документация.

**11. Что происходит на этапе исполнения тестирования?**

Выполнение тестирования происходит на стадии тестирования и представляет собой практический поиск дефектов с использованием тестовой документации, составленной ранее.

**12. Какие типы тестов выполняют для первой поставки программного продукта?**

Для первого билда рекомендуется проводить Smoke+AT готовой функциональности: поверхностное тестирование выполняется для определения пригодности сборки для дальнейшего тестирования; полное тестирование системы или ее части как на корректных, так и на некорректных данных/сценариях позволяет обнаружить дефекты и внести запись о них в багтрэкинговую систему.

**13. Какие типы тестов выполняют для последующих поставок программного продукта?**

Для последующих билдов композиции тестов могут быть следующими:

Если не была добавлена новая функциональность, то: DV+MAT. Т.е., выполняется проверка исправления дефектов программистом, а также проверка работоспособности остальной функциональности после исправления дефектов на позитивных сценариях.

Если была добавлена новая функциональность, то: Smoke+DV+NFT+Regression Test. В частности, выполняется поверхностное тестирование, проверка исправления дефектов программистом, тестирование новых функциональностей, проверка старых функциональностей, т.е. регрессионное тестирование.

Если была добавлена новая функциональность, то возможен также вариант: DV+NFT+Resression test.

**Вывод:** изучила классификацию видов тестирования, практически закрепила эти знания путем генерации тестов различных видов, научилась планировать тестовые активности в зависимости от специфики поставляемой на тестирование функциональности

\.