# ЛАБОРАТОРНАЯ РАБОТА №5

# Тема лабораторной работы: ручное тестирование

**Общие сведения по работе**

Работа предусматривает использование ранее созданной тестовой документации в тест-кейсах и тест-сьютах для контроля реализованной функциональности в программной системе. Ручное выполнение тест- кейсов может быть выполнено в зависимости от стадии разработки программного обеспечения (необходимая функциональность должна быть реализована) и является наиболее простым способом реализации процедуры тестирования как таковой. Другие подходы в виде реализации модульных тестов, реализации автотестов могут быть более трудоемкими и менее гибкими.

**Методические рекомендации и материалы**

Ручное функциональное тестирования подразумевает выполнение тестовых сценариев тестировщиком на специально подготовленном стенде. Порядок проведения ручного тестирования описывается следующими шагами:

1. Анализ исходных документов с требованиями и описанием тестовых случаев.
2. Изучение тест-плана для уточнения сроков, объемов, видов тестирования, ответственных лиц, проводящих тестирование.
3. Выполнение требуемых тест-кейсов с учетом условий выполнения, шагов сценария, эталонных результатов. Ручное выполнение подразумевает взаимодействие тестировщика с элементами пользовательского интерфейса тестируемой программной системы. Эталон необходимо сравнивать с полученным в программной системе результатом.
4. После завершения очередного прогона тест-кейса в отчете по тестированию фиксируются сведения о времени, идентификаторе соответствующего тест-кейса и общем результате: успешно или неуспешно. В случае неуспешного выполнения необходимо описать возникший дефект: на основании шагов и условий тестового сценария нужно описать расхождение в полученном и ожидаемом результате, прикрепить соответствующие дополнительные материалы (скриншоты, видео, логи, дампы) для лучшего понимания разработчиком возникшей проблемы и ускорения поиска путей ее решения.

Ручное выполнение тест-кейсов может быть проведено с использованием соответствующего программного обеспечения, в котором как хранятся сами тест-кейсы, так и организуется создание тест-планов и есть возможность поэлементного просмотра списка тест-кейсов в режиме выполнения.

Описание дефектов производится в соответствии со следующей структурой:

* Краткое описание. Приводится текстовое описание дефекта. Главная цель – пояснить, уточнить особенности дефекта, которые не могут раскрыть остальные поля.
* Серьезность. Отражает объем функциональности, которая не может быть использована из-за возникшего дефекта. Градация из значений – блокирующая (blocker), критическая (critical), значительная (major), незначительная (minor), тривиальная (trivial), предложение к улучшению (enhancement) – позволяет сделать вывод о том, насколько дефект затрагивает работу программной системы и может ли тестирование быть выполнено далее.
* Приоритет является атрибутом, позволяющим управлять очередностью исправления дефектов.
* Шаги к воспроизведению. Строго определенная последовательность действий при взаимодействии с программной системой, приводящая к состоянию, описанному данным дефектом.
* Результат описывает полученное в ходе взаимодействия с системой состояние.
* Ожидаемый результат – некоторое описание эталонного состояния системы, которое ожидалось после выполнения описанных шагов.

Кроме представленных атрибутов описания дефекта можно дополнительно прикладывать материалы, позволяющие точнее воспроизводить дефект и быстрее понимать, в чем состоит проблема.

**Задания к лабораторной работе**

1. Сформировать по ранее разработанной тестовой документации тест- план, включить в него некоторое подмножество тест-кейсов.
2. Выполнить тест-кейсы, представленные в тест-плане.
3. В отчет по лабораторной работе включить:
   1. Цель работы.
   2. Тест-план.
   3. Отчет по тестированию.
   4. Список выявленных дефектов.
   5. Выводы по работе.
   6. Список использованных источников.
4. Оформить и защитить отчет.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Автор | Приоритет | Заголовок | Список тест-кейсов |
| 1 | User | 1 | Модуль пользователя | 1.1Авторизация  1.2 Редактирование профиля |
| 2 | User | 1 | Модуль администратора | 2.1Код админа  2.2 Фамилия  2.3 Код должности |
| 3 | User | 1 | Модуль оборудования | 3.1 Код оборудования  3.2 Код назначения оборудования  3.3 Название оборудования |
| 4 | User | 1 | Модуль отчетности | 4.1Оформление отчета  4.2 Оформление отчета  4.3 Оформление отчета |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Описание  (Тип) | Предусловия | Шаги | Ожидаемый результат |  |  |  |
| 1.1 | Модуль пользователя | 1. Пользователь находится в базе данных 2. Пользователь был ранее зарегистрирован в системе | 1. Ввести в поля «Логин» и   «Пароль» логин и пароль пользователя   1. Нажать кнопку «Войти» | 1. Пользователь перенаправляется в программу  Да |  |  |
| 1.2 | Модуль администратора | 1. Администратор зашел в базу | 1. Ввести в поля «Логин» и   «Пароль» логин и пароль пользователя  Нажать кнопку «Войти» | 1.Администратор на странице базы данных получит  Данные о пользователях и оборудовании  Да |  |  |
| 1.3 | Модуль оборудования | 1. Оборудование зарегистрировано. | 1. Ввести в обязательные поля «Код оборудования», «Код назначения»,   «Название оборудования», Нажать кнопку «Поиск» | 1. Пользователь на странице базы данных получит Данные о дисциплинах  Да |  |  |
| 1.4 | Модуль отчетности | 1. Пользовать зашел на страницу отчетности базы данных. | 1.Ввести в обязательные поля «Отчет», «Поиск» | 1. Пользователь на странице базы данных получит  Сформированный  Да |  |  |

**Контрольные вопросы**

1. Какие виды тестирования можно выполнять в ручном режиме?

1.Тестирование белого ящика (White box)

2.Тестирование черного ящика (Black box)

3.Тестирование серого ящика

4.Тестирование дыма

5.Кроссбраузерное тестирование

6.Приемочное тестирование

7.Бета-тестирование

8.Исследовательское тестирование

1. Какая документация участвует в ручном тестировании?

Формализация тестирования: тест-план, формат баг-репортов, отчётность Для того, чтобы подготовиться к функциональному тестированию QA-инженер составляет тест-план. Это — документация, которая потребуется ему при тестировании продукта, список действий, которые ему нужно будет совершить.

1. Может ли существовать программная система, свободная от дефектов?
2. Каковы требования к описанию дефекта?

- Краткое описание

-Серьезность

1. Каков жизненный цикл дефекта?

Жизненный цикл дефекта – это стадии, которые проходит ошибка с начала своего существования и до ее полного разрешения. Чтобы было проще воспринимать, жизненный цикл рисуют схематично, где отображаются все статусы и действия, которые эти статусы и сменяют. Каждый этап работы с дефектом обозначается как «Статус»