**ЛАБОРАТОРНАЯ РАБОТА 9 (2 часа)**

**Тема:** Тестирование интерфейса пользователя средствами инструментальной среды разработки

**Цель:** изучения процесса тестирования интерфейса пользователя различными методами.

**ТЕОРЕТИЧЕСКИЕ ПОЛОЖЕНИЯ**

**Графический интерфейс пользователя** (Graphical userinterface, GUI) – разновидность интерфейса обеспечивающее взаимодействие через графические элементы (меню, кнопки, значки, списки и т. п.).

Варианты реализации GUI:

– интерфейс настольного приложения;

– мобильный интерфейс;

– веб-интерфейс.

**Цели тестирования:**

– обеспечение качественного взаимодействия с пользователем;

– выявление ошибок функциональности;

– выявление необработанных исключений при взаимодействии с интерфейсом;

– выявление потери или искажения данных, передаваемых через элементы интерфейса;

– выявление ошибки в интерфейсе (несоответствие проектной документации, отсутствие элементов интерфейса).

**Особенности тестирования:**

– тест планы в виде сценариев работы пользователя;

– сценарии на естественном языке или в виде скриптов;

– выполнение тестов в ручном режиме или с помощью эмулятора;

– анализ экранных форм и видимых элементов, а не внутренних переменных.

*Покрытие* – участие интерфейсных элементов в тестах.

Найденный дефект: несоответствие реального поведения требованиям или проблемы в требованиях.

Проблема отчета об ошибке: расплывчатость формулировок.

**Типы требований к UI**

– требования к внешнему виду пользовательского интерфейса и формам взаимодействия с пользователем;

– требования к размещению элементов управления на экранных формах;

– требования к содержанию и оформлению выводимых сообщений;

– требования к форматам ввода;

– требования по доступу к внутренней функциональности системы при помощи пользовательского интерфейса;

– требования к реакции системы на ввод пользователя;

– требования к времени отклика на команды пользователя.

**Функциональное тестирование UI:**

– анализ требований к пользовательскому интерфейсу;

– разработка тест-требований и тест-планов для проверки пользовательского интерфейса;

– выполнение тестовых примеров и сбор информации о выполнении тестов;

– определение полноты покрытия пользовательского интерфейса требованиями;

– составление отчетов о проблемах в случае несовпадения поведения системы и требований либо в случае отсутствия требований на отдельные интерфейсные элементы.

**Оценка покрытия UI.** Функциональное покрытие – покрытие требований к пользовательскому интерфейсу. Структурное покрытие – для обеспечения полного структурного покрытия каждый интерфейсный элемент должен быть использован в тестовых примерах хотя бы один раз. Структурное покрытие с учетом состояния элементов интерфейса и внутреннего состояния системы – поведение некоторых интерфейсных элементов может изменяться в зависимости от внутреннего состояния системы. Каждое такое различимое поведение интерфейсного элемента должно быть проверено.

**Задание 1 – Тестирование веб-сайта**

**Технология выполнения:**

1. Протестировать сайт <http://shop.bugred.ru/>
2. Составить Тест-план:

* Название (Название, версия)
* Объект тестирования (что тестируем, что не тестируем)
* Принцип работы (краткое описание того, как работает объект)
* Виды тестирования:
* функциональное тестирование (проверка форм, ссылок, поисковой строки и др.);
* тестирование безопасности (целостность, доступность, конфиденциальность);
* тестирование на совместимость (проверка на разных устройствах и ОС);
* Usabiliti тестирование (простота эксплуатации, удобство навигации, тестирование контента);
* Тестирование сайта на продуктивность (определение работоспособности сайта, нагрузки).
* Порядок тестирования (что за чем, начало и конец тестирования)
* Окружение (где тестируем)

1. Оформить список проверок - Чек-лист:

Приложение делится на модули и составляются быстрые и простые проверки. В результате Pass или Fail.

* Декомпозиция
* Составление чек-листа

1. Составить Тест-кейсы на основной функциональности сайта:

* Корзина
* Регистрация
* Вход
* ...

### **Примеры тест-кейсов для чек-листа**

**1. Регистрация пользователя**

* Открыть страницу регистрации
* Заполнить форму регистрации
* Нажать кнопку "Зарегистрироваться"
* Проверить, что появилось сообщение об успешной регистрации
* Проверить, что пользователь был добавлен в базу данных

**2. Авторизация пользователя**

* Открыть страницу авторизации
* Ввести логин и пароль
* Нажать кнопку "Войти"
* Проверить, что пользователь был успешно авторизован
* Проверить, что отображается правильная информация о пользователе (имя, фото профиля и т.д.)

**3. Работа с профилем пользователя**

* Открыть страницу профиля
* Изменить информацию о пользователе (например, имя, фото профиля)
* Нажать кнопку "Сохранить"
* Проверить, что изменения были сохранены
* Проверить, что отображается правильная информация о пользователе

**4. Работа с контентом**

* Открыть страницу с контентом (например, статьи, видео)
* Проверить, что контент отображается корректно (например, все изображения загружаются, видео проигрывается)
* Проверить, что пользователь может ть контент в избранное или поделиться им в социальных сетях
* Проверить, что отображается правильное количество просмотров и лайков
* Проверить, что комментарии к контенту отображаются корректно

**5. Работа с функциональностью**

* Проверить работу функциональности продукта (например, поиск, фильтры, пагинация)
* Проверить, что все кнопки и ссылки работают корректно
* Проверить, что происходит корректный переход между страницами
* Проверить, что функциональность продукта соответствует описанию в требованиях

**6. Работа с различными устройствами и браузерами**

* Проверить, что продукт корректно отображается на разных устройствах (например, на компьютере, планшете и смартфоне)
* Проверить, что продукт корректно работает в различных браузерах (например, Google Chrome, Mozilla Firefox, Safari)
* Проверить, что продукт корректно работает в различных операционных системах (например, Windows, MacOS, iOS, Android)

1. Провести тестирование:

* Прогон чек-листа, тест-кейсов.
* Заведение багов.

|  |  |
| --- | --- |
| **Поле** | **Что содержит** |
| Название | Суть проблемы. Должен быть ёмким и понятным. |
| Описание | Описание действий, которые нужно совершить, чтобы дойти до бага. |
| Шаги воспроизведения | Процент выполнения шагов воспроизведения |
| Фактический результат | Что видим после того, как воспроизвели баг. |
| Ожидаемый результат | Что на самом деле хотели увидеть, как это должно работать по ТЗ. |
| Окружение | Где нашли баг. Например, ОС, браузер или тестовая среда. |
| Приоритет | Наивысший, Высокий, Обычный, Низкий |
| Серьезность | Блокирующая, Критическая, Значительная, Незначительная, Тривиальная |

**Пример заполнения баг-репорта**

1. Название: Некорректное отображение имени пользователя в профиле.

2. Описание:

* Зайти в аккаунт пользователя;
* Перейти на страницу профиля;
* Обратить внимание на поле «Имя пользователя»;
* Отметить, что вместо имени отображаются случайные символы (см. скриншот во вложении).

3.Ожидаемое поведение: Поле «Имя пользователя» должно содержать имя пользователя, введенное при регистрации.

4. Фактическое поведение: Поле «Имя пользователя» отображает непонятные символы (см. скриншот).

5. Шаги воспроизведения: 100%.

6. Приоритет: Высокий.

7. Серьезность: Средняя.

8. Окружение:

* ОС: Windows 10
* Браузер: Google Chrome версии 91.0.4472.124 (64-бит)
* Аккаунт: тестовый пользователь

1. Написать отчет о тестировании:

* Что было протестировано
* Где
* Какой результат
* Комментарии к отчету

**Задание 2 (дополнительно!) Протестировать любой сайт**

**Контрольные вопросы**

1. Что такое интерфейс?
2. Назовите цели тестирования и особенности тестирования при тестировании графического интерфейса пользователя (GUI).
3. Назовите типы требований к интерфейсу пользователя.
4. Назовите этапы функционального тестирования пользовательского интерфейса.
5. Назовите виды покрытий интерфейса пользователя (UI).