***ЗАДАНИЕ №5***

**Тестирование программного обеспечения: разработка тестов**

*Цель работы:* разработать рабочую тестовую документацию для тестирования web-приложения.

# Теоретические сведения

## 1.1 Рабочая тестовая документация

Создание тестовой документации значительно улучшает качество продукта за счет более тесного сотрудничества, уточнения деталей при разработке плана тестирования и документации. После завершения тестирования наличие тестовой документации позволяет проверить, насколько успешно были проведены все этапы тестирования.

Существует несколько разновидностей рабочей тестовой документации

(таблица 1):

1. Check List
2. Acceptance Sheet
3. Test Survey
4. Test Cases

***Check List* –** высокоуровневый список проверок, набор правил и критериев, по которым проводится тестирование приложения. Описывает основные проверки для типовой функциональности.

***Acceptance Sheet* -** документ, который содержит подробный перечень всех модулей и функций приложения, а также результаты всех тестов данных функций. Как правило, содержит статистику по наиболее важным показателям каждой сборки, определяющим ее качество.

***Test Survey* -** документ, который содержит подробный перечень всех модулей и функций приложения, конкретные проверки для них, а также результаты всех тестов. В некоторых случаях для проверок может быть указан ожидаемый результат. Как правило, содержит статистику по наиболее важным показателям каждой сборки, определяющим ее качество.

***Test Cases* (набор тест-кейсов)** **-** набор входных значений, предусловий выполнения, ожидаемых результатов и постусловий выполнения, разработанный для определенной цели или тестового условия, таких как выполнение определенного пути программы или же для проверки соответствия определенному требованию.

Основной фактор выбора тестовой документации – сложность бизнеслогики проекта. Кроме того, определяющими факторами могут быть сроки проекта, размер команды и объем проекта.

На одном проекте могут комбинироваться несколько типов тестовой документации. Например, для всего проекта составлен Test Survey, но для наиболее сложных частей составлены тест-кейсы.

**ТАБЛИЦА 1 – ВИДЫ РАБОЧЕЙ ТЕСТОВОЙ ДОКУМЕНТАЦИИ**

|  |  |  |  |
| --- | --- | --- | --- |
| **Тип**  **документации** | **Что описываем** | **Когда используем** | **Пример** |
| ***Checklist*** | Основные проверки | Для типовой функциональности | Протестировать форму входа в почту |
| ***Acceptance Sheet*** | Части функциональности, подлежащие проверке. | Небольшие, простые по бизнес-логике проекты  Часто выпоняемые тесты (Smoke test) | Форма входа на сайт |
| ***Test Survey*** | Конкретные проверки в рамках отдельных кусков функциональности.  Может содержать ожидае-мый результат. | Средние или большие проекты, с понятной бизнес-логикой | Форма входа на сайт:  - Корректные данные  - Неверное имя пользователя  - Неверный пароль... |
| ***Test Cases*** | Пошаговое описание, инст-рукции по тестированию.  Всегда содержит ожидаемый результат. | Большие и долгосрочные проекты, требующие глубоких знаний в предметной области | Форма входа на сайт:  1. Откройте форму входа  2. Введите имя пользователя test1  3. Введите пароль test1  4. Нажмите кнопку «Войти»  Ожидаемый результат: пользователь переходит на домашнюю страницу |

## 1.2 Тестирование web-приложения

Web-приложениями будем называть любые приложения, предоставляющие web-интерфейс. В настоящее время такие приложения получают все большее распространение: системы управления предприятиями и драйверы сетевых принтеров, интернет-магазины и коммутаторы связи – это только небольшая часть приложений, обладающих web-интерфейсом. В отличие от обычного графического пользовательского интерфейса web-интерфейс отображается не самим приложением, а стандартизированным посредником – web-браузером. web-браузер берет на себя все взаимодействие с пользователем и обращается к web-приложению только в случае необходимости.

Web-приложения в первую очередь характеризуются тем, что их пользовательский интерфейс имеет стандартизированную архитектуру, в которой:

1. для взаимодействия с пользователем используется web-браузер;
2. взаимодействие с пользователем четко разделяется на этапы, в течение которых браузер работает с одним описанием интерфейса;
3. эти этапы разделяются однозначно выделяемыми обращениями от браузера к приложению;
4. для описания интерфейса применяется стандартное представление (HTML);
5. коммуникации между браузером и приложением осуществляются по стандартному протоколу (HTTP).

Поэтому тестирование web-приложений обладает рядом особенностей, и при проведении самого процесса тестирования необходимо обращать внимание на следующие аспекты:

1. единство дизайна;
2. навигация;
3. функциональность;
4. совместимость с браузером;
5. совместимость с операционной системой;
6. "дружественность";
7. "работоспособность";

***Единство дизайна***. Под единством дизайна понимается не только, а точнее не столько сочетаемость цвета элементов (так как это удел дизайнера), сколько соблюдение выбранной цветовой гаммы, придающей всем страницам сайта "единство". Сюда входят цвета фона (или рисунок), ссылок (в т.ч. посещенной и активной), а также любых других элементов, расположенных на странице. Кроме того, на этом же этапе необходимо оценивать размер и вид используемого шрифта для различных уровней вложения текста (заголовки различных уровне, собственно текст, ссылки, примечания и т.п.) Здесь же имеет смысл оценивать совместимость с дизайном звуков, рисунков и анимации, а также проверять имеет ли место единство отображения при использовании других экранных расширений и глубин цвета.

***Навигация.*** Навигация предполагает тестирование перемещения по сайту, что дает представление о возможности любого пользователя легко найти необходимый раздел, независимо от способа реализации меню (текстовые ссылки, картинки, единая картинка с картой ссылок и др.). На этом же этапе оценивается логичность перемещения между формами, кнопками и другими элементами страницы при помощи TAB, курсорных клавиш и т.п.

***Функциональность.*** Общие подходы к тестированию функциональности веб-страниц аналогичны таковым при тестировании приложений. Ниже приведен примерный перечень основной функциональности веб-страниц:

* ссылки (работоспособность, открытие в том же или новом окне и
* т.п., полное отсутствие битых ссылок)
* формы (ввод текста, чисел, использование маски, работа с незаполненными полями, длина вводимых символов, корректная работа чек-боксов, комбо-боксов, radio buttons, логичность установок "по умолчанию" и т.д.).
* базы данных (поиск, добавление информации, редактирование, удаление, проверка на дублирование информации).
* доступ (различные роли и права)
* секретность (работа с паролями, передача данных, защита и т.д.)  кеширование (проверка на установку кеширования и обновления файлов)
* проверка работы с браузером (refresh, forward/back, изменение размеров окна, выбор кодировки, скроллинг, отключение флеша, скрипта)
* фреймы (загрузка страниц, скроллинг и т.п.)
* анимация (наличие, изменение размеров, загрузка и т.д.)  аудио и видео (наличие, размещение, качество и др.)
* activex
* печать (корректно ли печатаются страницы)
* загрузка (как на сайт, так и с сайта)
* экспорт/импорт данных (если есть такая ф-я)
* интеграция (например: возможность входа на сайт через социальные сети или оплата услуг через paypal account)
* рекомендации для достижения хорошего веб приложения (в некоторых частных случаях могут не соблюдаться - например, не всегда валидационные сообщения будут красного цвета):
* как внешний вид так и наполнение сайта должны быть надлежащего уровня (с сайтом должно быть приятно и удобно работать);
* все элементы должны быть выровнены, выравнивание одних и тех же элементов в различных частях одного и того же сайта должно быть унифицировано (например: выравнивание кнопок по левому краю);
* шрифт всех ярлыков должен быть одинакового цвета;
* не должно быть битых ссылок (ссылок не ведущих на необходимый контент, несуществующих ссылок);
* каждая под-страница (так называемые "child pages" или "subpages") должна открываться в новом окне;
* на каждой странице должны присутствовать удобные элементы навигации. такие как: хлебные крошки, вперед/назад, сохранить/отменить и т.д.;
* если в таблице содержится больше 10 элементов информации, то должна присутствовать "пагинация" (навигационные ссылки: следующая/предыдущая, первая/последняя и номера страниц);
* у каждой страницы наряду с заголовком должен быть подзаголовок выдержанные в одном стиле (шрифт, цвет и т.д.);
* каждый ярлык должен начинаться с заглавной буквы;
* желательно, чтобы валидационные сообщения были красного цвета;
* все обязательные для заполнения поля должны быть жирного шрифта или помечены специальным символом: \*.

***Совместимость с браузером.*** Общеизвестно, что в силу конкуренции, тот или иной браузер имеет нередко даже существенные отличия в отображении одной и той же страницы. Для того, чтобы убедится, что любой пользователь сможет получить всю необходимую информацию требуется проводить тестирование Web-страниц в различных браузерах. Кроме того имеются различия и в разных версиях одного и того же браузера. Это также необходимо учитывать при тестировании.

***"Дружественность".*** Под "дружественностью" мы понимаем то, насколько прост, легок в обращении и интуитивно понятен интерфейс сайта: легка ли навигация, доступно ли меню, не используются ли раздражающие пользователя приемы, не много ли всплывающих окон, все ли ссылки являются "рабочими", все ли необходимые данные доступны для пользователя и т.д. Например, если на сайте есть файл для скачивания, то желательно, чтобы пользователь имел возможность заранее знать его размер, мог оценить время закачки.

***"Работоспособность".*** Проверка на "работоспособность" подразумевает оценку скорости загрузки как страниц сайта в целом, так и каждого элемента в отдельности. Сюда включается оценка размера используемых рисунков, html-файлов, аудио и видео файлов, адаптация их к различным типам соединений (от обычного модемного dial-up соединения, начиная с 14400, до высокоскоростных технологий).

## 1.3 Основные проверки при проведении тестирования web-приложения

В таблице 2 представлен перечень основных проверок необходимых при проведении тестирования web-приложения, где элементы с отметкой (B) – basic, т.е. являются основными проверками, которые необходимо сделать при тестировании Web-приложений; с отметкой (A) –advanced, т.е проверки, предоставляющие информацию для полноценного тестирования элементов Web-приложений.

Таблица 2 – Перечень основных проверок при тестировании web-приложений

|  |  |  |
| --- | --- | --- |
| **Название элемента** | **Functional Test** | **GUI Test** |
| **Поле (B)** | * Обязательность ввода * Обработка только пробелов * Использование пробелов в тексте (1. пробелы в начале и в конце строки должны отсекаться при сохранении, 2. пробелы внутри текста отсекаться не должны) * Минимально/максимально допустимое количество символов (осуществлять проверку на ввод большого текста без пробелов не нужно) * Формат данных (исходя из его логического назначения и требований приложения) * Формат числовых данных (если допускаются): негативные, дробные с точкой и запятой, с точкой и запятой 123.123.123,00 ) * Ввод тегов и скриптов (проверка должна осуществляться только в пользовательской части - ввод тегов в админку для проверки их применения на фронте не производится. Введенные теги должны отобразиться в том же виде, в котором они были введены) * Использование специальных символов (введенные символы должны отобразиться в том же виде, в котором они были введены, если только ввод спец. символов не запрещен требованиями приложения) * Возможность редактирования введенных значений * Корректное распределение текста по строкам (переход на новую строку автоматически) * Уникальные данные (например, уникальность логина, email'а | * Название поля (спеллинг, соответствие с от-   крытым модулем или страницей)   * Выравнивание названий полей (выравнивание по левому краю или правому краю (в зависимости от требований приложения, отступы, идентичность расстояний   между названием и полем)   * Корректное расположение текста внутри текста, длинный текст не выходит за границы поля при вводе * Унификация дизайна (цвет, шрифт, размер (высота/ширина), выравнивание полей) * Расположение вводимого текста внутри поля (унификация, выравнивание по нижнему краю   (для textarea - по верхнему), если иное не определено специфичными требованиями приложения) |
|  | пользователя)  Автоматическая постановка курсора в первое поле для ввода при открытии формы |  |
| **Кнопка (B)** | * Отсутствие вызова одного и того же действия повторно при нажатии на кнопку несколько раз * Недоступные кнопки не скрыты, а заблокированы (если кнопка недоступна в данный момент, то она не должна пропадать с формы, пользователь должен знать ее существовании; но кнопка должна быть задизейблена - некликабельна и отображается более светлой) * Нажатие на пространство между близко расположенными кнопками не должно приводить к действию * Нажатие на пространство возле кнопки не должно приводить к действию * Должно работать нажатие на всю площадь кнопки, а не только по ее названию | * Название кнопки (спеллинг, соответствие с действием) * Эффект ‘нажатия’ (вид кнопки должен изменяться при нажатии, если это не противоречит возможностям браузера) * Название хинтов (соответствие с названием кнопки, спеллинг) * Унификация дизайна   (цвет, шрифт, размер (высота/ширина), цвет подсветки, выравнивание) . Проверка осуществляется как для кнопки, как элемента, так и для названия кнопки |
| **Радио баттон (B)** | * Функциональность (включение/выключение) * Не может быть меньше 2 радиокнопок * По умолчанию одна радиокнопка должна быть включена * Не может быть включено более   1 радиокнопки   * При переходе на следующую страницу и возвращении назад выбранная радио кнопка не должна сбрасываться | * Унификация дизайна для всего приложения * Выравнивание расположения радиобаттона с соответствующим названием * Выравнивание расположений радио баттонов (по краю) |
| **Чек бокс (B)** | * Функциональность (включение/выключение) * Обязательность выбора хотя бы одного чекбокса * Наличие дополнительного чекбокса, выставляюще   го/снимающего все чекбоксы при наличии больше 10 чекбоксов | * Унификация дизайна для всего приложения * Выравнивание расположения чек бокса с соответствующим названием * Корректность отображения задисэйбленного |

|  |  |  |
| --- | --- | --- |
|  | * При переходе на следующую страницу и возвращении назад выбранный чекбокс не должен сбрасываться * Отсутствие функции сортировки таблицы по полю, которое содержит только чекбоксы | чек бокса |
| **Поле со списком/выпада**  **ющий спи-**  **сок (B)** | * Сортировка по алфавиту или по смыслу * В случае, если значения выходят за границы списка, и нет возможности увеличения размера списка, то необходимо отображение хинтов (всплывающих подсказок) * Выбор пункта списка по нажатии соответствующей первой буквы на клавиатуре * Возможность выбора нескольких значений для поля со списком * Возможность введения значений вручную (если это позволяет приложение) * Возможность выбора значния из списка указателем мыши либо стрелками с клавиатуры | * Спеллинг значений * Подсветка при выборе каждого из значений, при выборе нескольких значений одновременно * Унификация дизайна   (цвет, шрифт, размер (высота/ширина), цвет подсветки, выравнивание). Проверка осуществляется как для поля, как элемента, так и для значений, и их названий |
| **Меню (B)** | * Осуществление соответствующего перехода при выборе пункта меню * Визуальное различие в момент работы на определенной вкладке   (подсветка, подчеркивание) | * Подсветка таба при наведении курсора * Изменение курсора при наведении * Эффект ‘нажатия’, если это не противоречит возможностям браузера * Если работа в данный момент в выбранной вкладке, то в меню она отличается визуально, при этом является некликабельной * Совпадение названий в случае, если меню дублируется в нескольких местах |
| **Окно (B)** | * Возможность изменения размера окна браузера. Несколько спо- | * Появление скролла при уменьшении (изменении) |

|  |  |  |
| --- | --- | --- |
|  | собов сменить разрешение окна браузера при тестировании:   * Сменить разрешение самого экрана. * Воспользоваться виртуалкой. * Воспользоваться расширениями для браузеров. * Возможность изменения масштаба страницы. | размера окна браузера   * Сохранение расположения элементов при уменьшении (изменении) окна браузера, при изменении масштаба. Если дефект воспроизводится только при смененном масштабе, то в описании дефекта одним из шагов можно указать «увеличить/уменьшить масштаб до …%» * Соответствие названия окна в зависимости от назначения страницы   (например, название окна должно быть Profile, если пользователь находится на странице профиля)   * Спеллинг, синтаксис названий * Унификация названий |
| **Скрол-**  **линг (B)** | * Отсутствие скролла в случае, если текст вмещается на странице без прокрутки. * Соответствующее изменения текста при использовании скролла. * Возможность изменения положения скролла при помощи мыши, кнопок Page up/down,   Home/End. | Унификация видов и типов скроллов на всех страницах (если есть кастомный скролл, он должен быть применен на всех идентичных формах) |
| **Ссылка (B)** | * Функционирование ссылки (должен осуществиться переход на соответствующую страницу) * При наведении указателя мыши отображается подсказка (желательно) * Форматы ссылок и префиксов * Срабатывание ссылки только при клике на саму ссылку, а не на пустую область возле нее | * Унификация стилей (в соответствие с дизайном сайта) * Расположение ссылок (в соответствие с дизайном сайта). Например, расположение всех ссылок слева или справа от элементов * Названия (унификация, идентичность названий ссылок одинакового |

|  |  |  |
| --- | --- | --- |
|  |  | назначения, спеллинг, соответствие с открытым модулем или страницей, вместимость названия ссылки в отведенном блоке)   * Изменение вида курсора при наведении на ссылку * Изменение вида ссылки при наведении курсора (подчеркивание) |
| **Таблица (B)** | * При появлении нескольких страниц есть кпопки Вперед, Назад, На первую, На последнюю страницу (пагинация) * Проверка сортировок (+ проверка сортировки по дефолту) * Проверка фильтрации (если есть возможность) * Апдейт значений таблицы после добавления/изменения/удаления данных * Единичное/множественное выделение нескольких значений | * Унификация дизайна для всего приложения (цвет, шрифт, размер (высота/ширина), выравнивание) * Название (соответствие с текущим модулем,   спеллинг)   * Выравнивание иконок сортировки в названии колонок * Выравнивание названий колонок, значений внутри таблицы * Корректное отображение длинных названий (соответствующие переходы на новые строки, сокращение названий (появление ..., либо сокращение по слову) * Корректное отображение данных после использования сортировки (размеры колонок и столбцов фиксированы, текст не разбивает структуру таблицы) |
| **Поп-ап (B)** | * Должен быть модальным * Корректное выделение background'а страницы (если фон страницы изменяется при появлении поп апа, то при изменении | * Cпеллинг, синтаксис текста, расположенного на поп-апе * Отображение попапа по центру страницы, ок- |

|  |  |  |
| --- | --- | --- |
|  | масштаба страницы фон должен заполнять всю страницу - размер измененного фона соответствует размеру страницы)  Фиксированное положение поп-апа (динамическое изменение положения) в случае использования скролла | на, формы   * Выравнивание текста, представленного на попапе * Корректное расположение текста на поп-апе: текст должен быть в рамках поп-апа, длинное название должно располагаться на новой строке, если иное не определено специфичными требованиями приложения |
| **Календарь (B)** | * При возможности ввода даты вручную необходимо проверить разные форматы * Проверки логичности ввода (даты в будущем, валидации возраста и т.д) | * Унификация дизайна для всего приложения (цвет, шрифт, размер (высота/ширина), выравнивание) * Отображение календаря рядом с полем * Корректное выравнивание всех элементов и ссылок в календаре |
| **Поле для загрузки файлов (B)** | * Обязательность выбора файла * Форматы * Ограничения на размер * При отсутствии изображений должен быть соответствующий thumbnail, либо картинка совсем не должна отображаться * Контроль за размером (высота/ширина), должен быть ресайз * Загрузка исполняемых файлов   (EXE, PHP, JSP  etc.). Переименованный EXE | * Унификация дизайна для всего приложения (цвет, шрифт, размер (высота/ширина), выравнивание) * Выравнивание названий загруженных файлов, самих thumbnails файлов |
| **Сообще-**  **ния (B)** | * Пользователь должен быть информирован о действиях, происходящих в системе посредством сообщений об успешном завершении операции * На необратимые действия, такие как удаление, должны быть подтверждающие сообщения * Введенные в форму данные не должны сбрасываться после появ- | * Сообщение о том, что нет соответствующих айтемов (в таблицах, при поиске, при переходе на страницы) * Спеллинг, синтаксис сообщений * Соответствие сообщений по смыслу в зависимости от выполняемого |

|  |  |  |
| --- | --- | --- |
|  | ления сообщения об ошибке | действия   * Соответствие названия требуемого действия в сообщении об ошибке действию, которое пользователь должен выполнить. Например, если необходимо выбрать значение из списка, в сообщении об ошибке должно быть указано ‘Please select’, а не ‘Please enter’ * Унификация стилей   (цвет, размер) для всего приложения   * Соответствие цветов типу сообщений (красный для сообщений об ошибках, зеленый для сообщений об успешном завершении операции), если данные цвета не противоречат специфичным требованиям приложения * Соответствие названий полей в сообщениях об ошибках и сообщениях об успешном завершении операции названиям полей, форм, таблиц, кнопок, и т.д. * Соответствие порядка выведения сообщений об ошибках в соответствие с порядком расположения полей, в которых были найдены ошибки * Поле, в котором содержится ошибка должно (желательно) выделяться цветом * Невалидное значение не должно отображаться в сообщении об ошибке. |

|  |  |  |
| --- | --- | --- |
|  |  | Пример как быть не должно: "Email  2309234@@mail.ru не соответствует допустимому формату”.  Согласование числительного и связанного с ним существительного должно соблюдаться. Например, "1 день", "2 дня", "5 дней". |
| **Общие проверки (A)** | * Перед тестированием каждой новой сборки необходимо осуществить очистку кэша и кукис. Для полной очистки кэша и куки можно воспользоваться приложением Ccleaner. * 404 Error (переход по некорректному урлу должен вести на страницу с 404 ошибкой, а не просто на страницу Page cannot be found. страница 404 ошибки должна быть реализована в общем   дизайне тестируемого приложе-  ния)   * Tab order (сверху вниз слева направо). Поля, доступные для прочтения и задисэйбленные должны пропускаться * Logo должен быть ссылкой на главную страницу * Фокус на кнопке для исполнения действий (ввод данных -> нажатие Enter -> действие осуществилось) * Проверка Breadcrumbs ("Хлебные крошки" - элемент навигации, являющийся признаком удобства пользования сайтом в целом и перемещением по его структуре) * Отображение flash-элементов   при отключен-  ном/неустановленном в браузере flash-плеере (пользователю должно быть предложено скачать и | * Уменьшение / увеличение масштаба страницы (элементы должны соответственно перераспределиться с сохранением   пропорций)   * Уменьшение / увеличение размера окна браузера (элементы должны соответственно перераспределиться с сохранением пропорций, взаиморасположение должно сохраняться) |

установить последнюю версию

flash

-

плеера;

на

месте

flash

-

должно

отображаться

объекта

альтернативное изображение)



П

работоспособности

роверка

отправки email нотификаций (как

админу, так и пользователю), если

только отсутствие писем не явля-

ется спецификой проекта



работоспособности

Проверка

отключенном

при

приложения

Основная

JavaScript.

функцио-

нальность и навигация должн

а

работать. Подробное описание то-

го, как отключить JavaScript и на

что обращать внимание, находит-

ся

[здес](https://wiki.itransition.com/x/jojnAg)

[ь](https://wiki.itransition.com/x/jojnAg)

[.](https://wiki.itransition.com/x/jojnAg)



Тестирование ссылок и прав

доступа.



Всегда проверять взаимосвязь

компонентов. Обращать внимание

на то, как ведет себя один компо-

нент

при

изменении/удалении

другого. Например, при удалении

категории товара не должны уда-

ляться все товары в этой катего-

рии.

***Задание***

1. Самостоятельно найти и установить свободное программное обеспечение организации работы интернет-магазина или любое другое web-приложение, выделить наиболее простой для анализа модуль.
2. Разработать рабочую тестовую документацию (тест-кейсы) для web-приложения.
3. Оформление тест-кейсов смотреть в файле WorkBook.pdf раздел 1.4.3 «Создание и оформление тест-кейсов».

# Содержание отчета

Рабочая тестовая документация.

# Контрольные вопросы

Какие существуют разновидности рабочей тестовой документации?

1. Check List: что описывают и когда используют?
2. Acceptance Sheet: что описывают и когда используют?
3. Test Survey: что описывают и когда используют?
4. Test Cases: что описывают и когда используют?
5. Что такое web-приложение?
6. Какую архитектуру имеет web-приложение?
7. На какие особенности необходимо обращать внимание при тестировании web-приложения, охарактеризуйте эти особенности.
8. Какие основные проверки необходимо выполнять при тестировании web-приложения?