Задание подготовлено в рамках проекта АНО «Лаборатория модернизации образовательных ресурсов» «Кадровый и учебно-методический ресурс формирования общих компетенций обучающихся по программам СПО», который реализуется с использованием гранта Президента Российской Федерации на развитие гражданского общества, предоставленного Фондом президентских грантов.

**Разработчики**

Лысенко Ирина Владимировна, ГАПОУ «Тольяттинский индустриально-педагогический колледж»

Мясин Дмитрий Николаевич, ГАПОУ СО «Тольяттинский индустриально-педагогический колледж»

**Назначение задания**

Целеполагание и планирование. Уровень II

Специальность 09.02.07 Информационные системы и программирование

Специализация: Разработчик web и мультимедийных приложений

ПМ 01. МДК.01.02 Поддержка и тестирование программных модулей

Тема: Отладка и тестирование программного обеспечения

**Комментарии**

Задание предлагается на этапе освоения нового материала, когда обучающиеся еще не знакомы с форматом теста для проведения ручного тестирования. Требования к формату вводятся по итогам обсуждения результатов выполнения задания.

Вы - постоянный участник форума IT-специалистов. Сегодня вы прочитали на форуме сообщение «форумчанина», который ранее не раз давал вам дельные советы по разработке сайтов, и решили помочь.

Прочитайте сообщение (источник 1). Изучите общие требования к процедуре тестирования (источники 2 и 3), ознакомьтесь с чек-листом проверки форм сайтов на время (источник 4).

**Составьте план ручного тестирования формы авторизации.**

Заполните бланк. Отчеркивайте каждый критерий в отдельную строку. Нумеруйте этапы с использованием круглой скобки, а критерии – с использованием точки

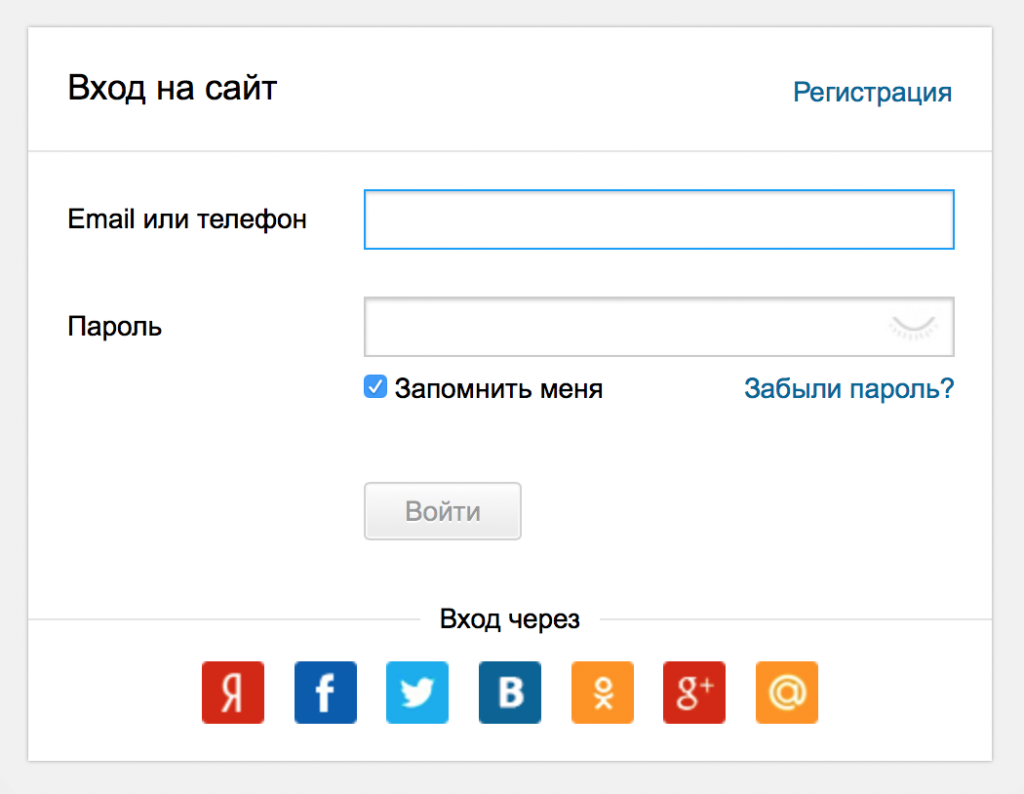
*Бланк*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Этап тестирования формы | Критерий, проверяемый при тестировании | Время на проверку (сек) | Не выполнение - блокировка формы | Инструменты |
| 1) |  |  |  |  |  |
| 1. |  |  |  |  |  |
|  |  |  |  |  |  |

Итого на проверку теста затрачено времени: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ минут.

***Источник 1***

**Товарищи! Спасите, а то меня уволят!**

Нужна помощь в решении следующей задачи: устроился работать тестировщиком, а опыта работы в этой области нет. Дали первое задание, вроде бы оно очень простое:  
Есть форма авторизации на сайте: 

**\*\*\*\*\*\*\*\***

Введите данные на английском языке без знаков препинания.  
Проверьте нажатие CaрsLk

Haas1989@mail.ru

Нужно описать, как протестировать эту форму на все возможные ошибки.

**«Mega.ru»**

Но кроме того, что нужно ввести логин и правильный/ неправильный пароль и посмотреть, произошел ли вход, я ничего больше придумать не могу.

***Источник 2***

**Этапы тестирования интерфейсов веб-сайтов**

*Дмитрий Нарыжный*

Тестирование сайта - один из важных жизненных этапов, после которого, предоставляется заказчику готовый проект без ошибок, с хорошей читабельностью, воспринимаемой легкостью, удобством и надежностью. Мы хотели бы рассказать о том какие этапы проходит сайт перед выходом в продакшн.

Тестирование - это отклонение фактического результата от ожидаемого, другими словами - это процесс поиска багов (ошибок).

Основные правила тестирования веб-сайтов - это шаги, которые показывают пользователю, насколько удобен и логичен будет проект, насколько просто и возможно найти ту или иную информацию. Хорошо ли воспринимается человеческому взгляду и правильно ли работает весь функционал данного сайта, который был поставлен по ТЗ или брифу - это основные показатели для тестировщика.

**Как тестировать сайт?**

Тестирование может происходить самыми разными способами, однако не стоит забывать о самом процессе и стратегии тестирования. От него зависит последовательность ваших действий. На сегодняшний день, специалисты по тестированию веб-сайтов применяют такие виды как:

* Тестирования удобства пользования (юзабилити),
* Тестирование интерфейса пользователя и корректности (UI testing),
* Функциональное тестирование и тестирование производительности,
* Тестирование безопасности.

***Источник 3***

**Процедура тестирования форм ввода**

В настоящее время практически в каждом программном продукте интерфейс (внешний вид и способ общения с пользователем) выполнен в виде форм с полями для ввода/вывода определенной информации. Это могут быть формы отправки комментариев и отзывов, форма обратной связи, платежные данные, анкеты и многие другие. Они должны соответствовать определенным требованиям по внешнему виду и юзабилити, проверки корректности ввода данных, функциональным возможностям. От этого зависит скорость решения задачи пользователя и общее впечатление от сервиса. Форма считается хорошей, если по мере ее заполнения у пользователей не возникает никаких вопросов и задержек.

Предусмотрены основные этапы проверки критериев программных продуктов с формами:

1. Удобство интерфейса (юзабилити);
2. Проверка корректности ввода данных;
3. Функциональность формы;
4. Защита данных.

Начинают проверку с не фатальных ошибок (они приводят к блокировке формы), а с проверки интерфейса, потому что его проверить легче всего - обычно во время визуального осмотра. При создании форм необходимо обратить внимание на то, понятно ли для чего предназначена форма и зачем ее нужно заполнять? Прежде всего нужно убедиться, что макет формы легко понять, поскольку сложная форма или форма с большим количеством полей будет вызывать у пользователей нежелание ее заполнять. Форма получит более высокий уровень конверсии, если у нее будет меньше полей и они будут хорошо продуманными и легко заполняемыми.

Каждая форма должна запрашивать у пользователя только необходимый минимум информации, пользователь должен всегда понимать, каких конкретно данных от него ждут и зачем. Поэтому в сложных формах все неочевидные поля нужно снабжать соответствующими комментариями. Также все поля могут иметь текстовый заполнитель (placeholder), который содержит подсказку что именно нужно указывать в поле и исчезает при вводе первого символа. Если текстовые поля содержат заполнители, то они должны быть центрированы по вертикали. Если форма содержит поля для указания персональной информации, например, номера телефона или адреса, необходимо добавлять пояснение о том, как эта информация будет использована.

Для удобства работы с формами очень важен диалоговый интерфейс с пользователем - появление предупреждений, диалоговых окон, вопросов, предложений о помощи, если у пользователя возникают проблемы, предупреждения, сообщения об успешном/не успешном выполнении операции и т.д. Также нужно предусмотреть, если не работает мышь, можно ли заполнять форму, переходя с помощью клавиш клавиатуры - tab и др.

Для начала необходимо проверить то, что все обязательные поля в формах отмечены звездочкой. Также должно быть указано, что поля, отмеченные звездочкой, являются обязательными. Это облегчает пользователям понимание того, какие поля они должны заполнить, а какие являются необязательными. Если же все поля обязательны для заполнения, рядом с их названиями звездочки не выводятся.

В случае, если пользователь ввел некоторые данные некорректно или прошел на другую страницу, а потом опять вернулся к заполнению формы, форма должна быть показана заново и при этом все введенные данные (за исключением паролей и полей CAPTCHA) должны сохранять свои значения.

В формах должна быть встроена проверка заполнения обязательных и необязательных полей. Существует 2 типа валидации: inline-валидация и валидации при отправке формы.

Inline-валидация - это автоматическая проверка введенных данных в режиме реального времени. Если пользователь заполнил информацию правильно, рядом со строкой ввода может появляться подтверждение этого в виде зеленой галочки. Если что-то не так - сведение об ошибке. Такой способ проверки заметно улучшает конверсию и экономит время пользователя.

Валидация при отправке формы осуществляется только после того, когда форма отправлена, т.е. в случае наличия каких-либо проблем с одним или несколькими полями формы, пользователь получит обратную связь только после отправки формы.

Очень часто в формах допускают ошибки - не проверяют заполнение поля - тогда пользователь может не заполнить какие то важные поля, но разработчик формы забыл проверить этот факт и форма отправила неполные данные. Выхода здесь два - или предусмотреть автозаполнение по умолчанию (если это возможно - типовыми данными, данными, сохраненными на этом компьютере с прошлой сессии работы в форме и т.д.).

Если форма заполнена не полностью, то кнопка отправить должна быть неактивна. Ошибки ввода данных в поля логин и пароль приводят к фатальным ошибкам работы формы. Если форма заполнена не валидными данными и/или не полностью - кнопка «Отправить» должна быть неактивна и неверно заполненные поля должны показывать подсказку, иначе в базу отправятся невалидные данные, которые она не сможет сохранить.

Формы должны проверять ввод значений в некоторых полях, например, адрес электронной почты или номер телефона. При обнаружении ошибки курсор следует автоматически переместить в поле с ошибкой и в удобном для восприятия месте необходимо указать причину неудачи отправки данных. Все неверно заполненные поля должны быть визуально выделены и рядом с полем указывается суть ошибки заполнения, а также подсказка по правильному заполнению. Часто для полей задают ограничения по длине символов, ограничения на ввод некоторых символов (нельзя вводить «» и т.д.), требования к обязательному наличию каких то комбинаций символов (в вашем сообщении должны присутствовать большие и маленькие буквы, цифры), требования к языку ввода и т.д.

Выпадающий список - элемент графического интерфейса пользователя, который позволяет выбрать одно из нескольких значений параметра. Как у любого элемента формы, у выпадающего списка должны быть название и подсказка (placeholder). Следует также проверить наличие варианта по умолчанию: вариант по умолчанию должен быть только в том случае, если есть уверенность, что около 90% пользователей выберет именно его. Иначе необходимо предусмотреть это поле незаполненным. Как правило, использование выпадающих списков оправдывает себя, если количество значений параметра от 5 до 15. Если вариантов менее 5, то необходимо использовать радиокнопки, если вариантов больше 15, то следует или сократить их количество, или заменить выпадающий список на текстовое поле.

Наведение на кнопку отправки данных и клик по ней должен сопровождаться простейшей анимацией этой кнопки (например, увеличение, вдавливание, изменение цвета, размера, тени и т.д.).

Как только форма успешно отправлена, следует убедиться, что пользователю отображается соответствующее сообщение.

Нужно удостовериться, что некоторые функциональности не доступны неавторизованному пользователю.

Очень часто разработчики форм забывают проверять - а есть ли уже такой пользователь в системе. Очень часто бывают сбои при смене ФИО, логина, пароля, привязанных телефона, электронной почты и т.д. Очень часто при полном удалении пользователя из базы данных его логин все еще хранится в базе, что приводит к сокращению списка сводных логинов и не позволяет новому пользователю выбрать логин, который уже никому не принадлежит. Все это приводит к фатальным ошибкам формы.

При проверках нужно убедиться, правильно ли работает функционал в формах при вводе данных. В первую очередь следует проверить все позитивные кейсы, которые чаще всего будут встречаться пользователям ресурса. Позитивный тестовый случай использует только корректные данные и проверяет, что приложение правильно выполнило вызываемую функцию, т.е. призван показать, что программа работает так, как и полагается, при условии, что пользователь вносит корректные данные и не выходит за рамки предусмотренного сценария поведения.

К таким кейсам можно отнести заполнение всех обязательных полей формы корректными, валидными данными, которые не должны вызвать каких-либо сбоев/ошибок. Это самый простой этап тестирования.

Что касается негативного тестирования - здесь может быть очень много различных тестов, проверок. Рассмотрим основные из них.

Если речь идет о числовых полях, значит эти поля можно проверить, введя: числа с буквами; числа с точкой; числа с запятой; числа содержащие спецсимволы; числа различной длины (как целая, так и дробная часть); пустые поля; пробелы и др.

Для текстовых полей может быть определен другой набор тестов, специфический для определенных условий. Возьмем, к примеру, поле для ввода e-mail адреса: в адресе отсутствуют обязательные символы «@» и «.»; последовательность символов «@» и «.»; количество символов «@» и «.»; только цифры в e-mail адресе; спецсимволы в e-mail адресе; допустимая длина адреса; пустое поле; пробелы.

Если заполнены не все обязательные поля, то незаполненные поля должны быть подсвечены и, желательно, сопровождаться текстовыми сообщениями.

В случае, если есть два поля: одно для ввода пароля, а второе для его подтверждения, то при вводе различных паролей должны появляться соответствующие сообщения и подсвечиваться поля.

Также при тестировании текстовых полей содержимое текстовых полей должно проверяться на наличие спецсимволов и тегов, иначе тип таких даны не совпадет с типом поля в базе данных, данные не сохранятся и пользователь не сможет войти.

Хорошей практикой считается, если текстовые поля автоматически заполняются заранее известными данными. Например, для авторизованного пользователя могут автоматически заполняться поля с именем, фамилией, номером телефона, домашним адресом, e-mail адресом (при наличии таких данных в личном кабинете пользователя).

Очень часто работа с формами происходит в присутствии других пользователей, тогда данные на форме нужно защитить - предусмотреть возможность скрывать/показывать секретную информацию при вводе данных. Хорошие формы сайта должны быть адаптированы к просмотру как на ПК, так и на мобильных устройствах - сотовых телефонах, планшетах, адаптироваться под размеры экрана.

Блокировка формы не позволяет пользователю перейти на следующую страницу до устранения проблемы на форме. Многократное введение неправильного логина или пароля слишком быстрый или слишком долгий ввод данных воспринимаются как попытка взлома, блокирует дальнейший ввод данных в форму без подтверждения личности.

Истечение срока обслуживания, удаление пользователя из базы по окончании договора или из-за нарушений навсегда блокирует профиль пользователя.

***Источник 4***

**Чек-лист проверки форм сайтов на время**

*(по результатам усредненного анализа более 100 форм сервисом «qatestlab»)*

| № | Тип теста | Среднее время (на тест каждого параметра по каждому элементу формы), секунд | Инструмент проверки |
| --- | --- | --- | --- |
|  | Проверка параметров формы, требующих визуального осмотра интерфейса и чтения сообщений\* | 2 | Клавиатура, мышь, тач-скрин |
|  | Проверка параметров формы, требующих однократного ввода данных с клавиатуры, выбора из списка, сброса данных | 10 | Клавиатура, мышь, тач-скрин |
|  | Проверка задержек, искусственное создание задержек при вводе данных, многократный вод неправильных тестовых данных | 30 | Клавиатура, мышь, , таймер |
|  | Проверка параметров, требующих, перехода на другие ресурсы | 30 | Клавиатура, мышь, тач-скрин; другие ресурсы (электронная почта; сотовый телефон-служба СМС; форма приложения, следующая за формой авториации; другой сайт, тестовый переход на который произошел во время заполнения формы; база данных пользователей с правом редактирования и др.) |

\* Проверяются путем прочтения. Вызываются при выполнении операций п.2,3,4 – отдельное время на вызов не требуется.

*Использованы материалы источников:*

[*https://art-lemon.com/site-test*](https://art-lemon.com/site-test)*;*

[*https://training.qatestlab.com/blog/technical-articles/test-contact-form/*](https://training.qatestlab.com/blog/technical-articles/test-contact-form/)*;*

[*https://habr.com/ru/post/268253/*](https://habr.com/ru/post/268253/)*.*

Инструмент проверки

| № | Этап тестирования формы | Критерий, проверяемый при тестировании | Время на проверку (сек.) | Не выполнение - блокировка формы | Инструменты |
| --- | --- | --- | --- | --- | --- |
| **1)** | Проверить удобство интерфейса | | | | |
|  |  | Есть имя формы, дающее понимание, что это форма авторизации? | 2 | Нет |  |
|  |  | Есть текстовый заполнитель полей, отображающий смысл? | 4 | Нет |  |
|  |  | Есть переходы по форме с помощью: - клавиатуры /мыши? | 8 | Да | Клавиатура, мышь |
|  |  | Есть подсвечивание активных полей и кнопок? | 6 | Нет | Клавиатура, мышь |
|  |  | Есть ли возможность автосохранения | 2 | Нет | Клавиатура, мышь |
|  |  | Есть ли сообщения пользователю:  *- о пустых полях ввода*  *- о неправильном вводе символов в поля*  *- о неправильной длине поля пароля*  *-о неправильном формате поля логин*  *- вопрос: вести телефон/почту для восстановления пароля,*  *- предложения войти в соц. сети ВК, ОК* | 18 | Нет | Клавиатура, мышь |
|  | Корректно отображается на мобильном устройстве (адаптирован) | 30 | Нет | Тач-скрин |
| **2)** | Проверить корректность ввода данных | | | | |
|  |  | Проверка заполнения обязательных полей (или можно без них войти) Что будет, если поля пустые (оба или одно из них)? | 20 | Да | Клавиатура |
|  | Есть ли ограничение на длину (логина, пароля) при регистрации? | 20 | Да | Клавиатура |
|  | Есть ли ограничения на допустимые символы (+-«» кириллица, заглавные буквы и т.д.)? | 20 | Да | Клавиатура |
|  | Ввод 4 комбинаций корректного/некорректного логина и пароля | 80 | Да | Клавиатура |
| **3)** | Проверить функциональность формы | | | | |
|  |  | Есть ли существенные задержки времени при распознавании введенных данных при нажатии «Войти» | 30 | Нет | Таймер |
|  | При нажатии «Войти» происходит переход на следующую форму/страницу сайта | 30 | Да | Клавиатура, мышь |
|  | Не сбрасываются уже введенные в форму данные при перезагрузке страницы | 30 | Нет | Клавиатура, мышь, кнопка обновления страницы или переход на другой сайт и обратно |
| **4)** | Проверить защиту данных | | | | |
|  |  | Предусмотрена возможность защитить ввод пароля от подсмотра (функция скрыть/показать пароль) | 2 | Нет | Мышь |
|  | Предусмотрена возможность восстановления пароля | 2 | Да | Клавиатура, мышь |
|  | Новый пароль приходит на почту, он работает в форме | 30 | Да | Электронная почта зарегистрированного пользователя сайта |
|  | Блокируется форма ввода при многократных вводах пароля (во избежание подбора пароля) | 30 | Да | Клавиатура, мышь |
|  | Блокируется форма ввода при слишком долгом вводе данных | 30 | Да | Клавиатура, мышь, таймер |
|  | Блокировка пользователя, только что удаленного из базы пользователей | 30 | Да | Клавиатура, мышь, редактируемая база данных пользователей |
| ИТОГО на проверку теста затрачено времени | | | 424 секунды=7,1 минут | | |

*Подсчет баллов*

|  |  |
| --- | --- |
| Критерии структурированы в 4 этапа | 1 балл |
| Соблюдена последовательность этапов | 1 балл |
| За полный и верный набор критериев по каждому этапу | 2 балла |
| *За набор критериев по каждому этапу, содержащий одну ошибку или пропуск* | *1 балл* |
| *Максимально* | *8 баллов* |
| За полностью и верно указанное время выполнения критериев по каждому этапу | 2 балла |
| *За время выполнения критериев по каждому этапу, указанное с одной ошибкой или пропуском* | *1 балл* |
| *Максимально* | *8 баллов* |
| За полностью и верно сделанные указания по блокировке по каждому этапу | 2 балла |
| *За указания по блокировке по каждому этапу, сделанное с одной ошибкой или пропуском* | *1 балл* |
| *Максимально* | *8 баллов* |
| За полностью и верно указанные ресурсы по каждому этапу | 2 балла |
| *За ресурсы по каждому этапу, указанные с одной ошибкой или пропуском* | *1 балл* |
| *Максимально* | *8 баллов* |
| ***Максимальный балл*** | ***32 балла*** |