**Исламов Радмир ПИН-33**

**Тестирования сайта Strategium.ru**

Strategium.ru – русскоязычный форум посвященный Стратегиям.

**План тестирования**

|  |  |
| --- | --- |
| Группа | Пин-33 |
| Дата | 08.12.2023 |
| Версия | 1.0.0 |
| Автор | Исламов Р.Р. |

**Содержание**

1. Введение

1.1. Основная информация

1.2. Цель

2. Область тестирования веб-сайта

3. Тест-план и стратегия тестирования

3.1. Функциональное тестирование

3.2. Процедура тестирования

3.3. Отчеты об ошибках

4. Ресурсы

4.1. Инструменты

4.2. Список браузеров

4.3. Список устройств

5. Критерии качества

6. Риски процесса тестирования

7. Результаты

**1. Введение**

1.1. Основная информация

Документ описывает методы и подходы к тестированию, которые будут

использоваться тестировщиками отдела тестирования, для тестирования сайта и приложения.

План тестирования может использоваться как тестировщиками, так и менеджерами, разработчиками.

Объект тестирования — это деятельность, направленная на проверку работоспособности функций сайта в браузерах Chrome, Yandex Browser, Microsoft Edge и в операционных системах Windows и Linux Fedora.

1.2. Цель

Тест-план проекта преследует следующие цели:

● Определить существующую информацию о проекте и программных компонентах, подлежащих тестированию.

● Описать стратегии тестирования, которые будут использоваться.

● Определить необходимые ресурсы для проведения работ по тестированию.

● Привести результаты тестирования.

Результаты будут представлены в виде отчетов. Все найденные ошибки будут отслеживаться с помощью баг-трекинговойсистемы.

2. Область тестирования сайта

В объем работ по тестированию сайта входит регрессионное тестирование нового функционала и реализованных ранее компонентов и функций:

Цель функционального тестирования состоит в том, чтобы убедиться, что весь программный продукт работает в соответствии с требованиями, и в приложении не появляется существенных ошибок. Функциональное тестирование является наиболее существенной частью тестирования программного обеспечения, включающее в себя проверку различных аспектов системы. Программный продукт должен пройти все запланированные тесты. Только в этом случае можно быть уверенным в его качестве.

**2. Область тестирования сайта**

В объем работ по тестированию сайта входит регрессионное тестирование нового

функционала:

**3. Тест-план и стратегия тестирования**

3.1 Функциональное тестирование

Цель функционального тестирования состоит в том, чтобы убедиться, что весь программный продукт работает в соответствии с требованиями, и в приложении не появляется существенных ошибок. Функциональное тестирование является наиболее существенной частью тестирования программного обеспечения, включающее в себя проверку различных аспектов системы. Программный продукт должен пройти все запланированные тесты. Только в этом случае можно быть уверенным в его качестве

3.2. Процедура тестирования

Процедура тестирования предполагает следующие пункты:

• Сообщения об обнаруженных ошибках программного обеспечения.

Необходимо проверить различные аспекты тестируемого программного обеспечения, для

этого требуется выполнение различных типов тестирования.

Основные типы тестирования, которые должны быть выполнены:

• Функциональное Тестирование.

• Тестирование пользовательского интерфейса.

• Юзабилити-тестирование.

• Регрессионное тестирование.

• Автоматическое тестирование API

• Автоматическое тестирование фронта - Snapshot тесты.

В рамках тест-плана не будут выполняться виды тестирования:

• Тестирование безопасности.

• Тестирование нагрузки

3.3. Сообщения об ошибках

Отчеты об ошибках создаются для того, чтобы предоставить команде разработчиков и

руководителю проекта исчерпывающую информацию об обнаруженных ошибках.

Степень серьезности ошибок можно разделить на четыре категории:

● Блокирующие дефекты - это сбой всейп рограммной системы или критической

подсистемы, тестирование не может быть выполнено после возникновения дефекта.

●Критические дефекты — ошибки, которые также приводят к выходу из строя всейили

части системы, но существуют некоторые альтернативы обработки, которые позволяют

продолжить работу системы.

● Дефекты среднего приоритета - не приводят к сбою, но приводят к тому, что система

показывает неправильные, неполные или противоречивые результаты.

● Дефекты низкого приоритеты - это небольшие ошибки, которые не влияют на

функциональность: опечатки, грамматические ошибки, неправильная терминология и т.д.

Информация, указанная в каждом отчете об ошибке:

Каждый баг-репорт содержит следующую информацию о дефекте:

● Название баг-репорта.

●Краткое описание, представляющее собой̆ краткое описание проблемы.

● Предусловия.

● Шаги для воспроизведения ошибки;

● Ожидаемый результат.

● Фактический результат.

● Дополнительная информация о дефекте в виде прикрепленных скриншотов или

видеозаписей.

● Браузер, в котором проводились тесты.

4. Список браузеров

Chrome, Firefox, Edge. Используются последние версии.

5. Критерии качества

Продукт должен работать в соответствии с требованиями и техническим заданием (если таковые имеются).

Продукт не должен содержать критических и блокирующих дефектов в окончательной версии проекта.

6. Риски процесса тестирования

Следующие проблемы могут повлиять на результаты тестирования:

● Изменения и модификации программного продукта, которые не были запланированы и не обсуждались заранее с командой тестирования.

● Изменения в требованиях к программному обеспечению, которые не были предварительно обсуждены с командой тестирования.

●Задержки в исправлении ошибок.

7. Результаты

**1. Тестирование функциональности сайта**

1.1. Проверка работы основных функций сайта, таких как:

Регистрация нового пользователя

Авторизация пользователя

Создание новой темы на форуме

Отправка сообщения в теме на форуме

Поиск по форуму

1.2. Проверка работы функций сайта на разных устройствах и разных браузерах.

**2. Тестирование совместимости сайта**

2.1. Проверка работы сайта на разных операционных системах.

2.2. Проверка работы сайта на разных версиях браузеров.

2.3. Проверка работы сайта на разных разрешениях экранов.

**Тестовый сценарий**

Открыть страницу <https://www.strategium.ru/forum/>

Нажать войти

Указать данные

Нажать войти

Поиск несуществующей информации

Поиск существующей информации

Все данные верные

Пустые поля

Данные неверные

Буквы и спецсимволы

Все данные верные

Только верный логин

Неверный логин, спецсимволы

Только верная почта

Неверный

Только верный пароль

Неверный

Регистрация множества пользователей

Множественная авторизация

Находим поле поиска, вводим данные, нажимаем на поиск, получаем результат

Нажимаем кнопку "войти", указываем данные, смотрим результаты проверки

Нажимаем кнопку "регистрация", указываем данные, смотрим результаты проверки

**Тестовые кейсы**

Ручные тесты:

|  |  |  |
| --- | --- | --- |
| Проверка открытия сайта | Открытие сайта | Открываем сайт |
| Поиск | Поиск несуществующей информации | Находим поле поиска, вводим данные, нажимаем на поиск, получаем результат |
| Поиск | Поиск существующей информации | Находим поле поиска, вводим данные, нажимаем на поиск, получаем результат |
| Авторизация | Все данные верные | Нажимаем кнопку "войти", указываем данные, смотрим результаты проверки |
| Авторизация | Пустые поля | Нажимаем кнопку "войти", указываем данные, смотрим результаты проверки |
| Авторизация | Данные неверные | Нажимаем кнопку "войти", указываем данные, смотрим результаты проверки |
| Авторизация | Буквы и спецсимволы | Нажимаем кнопку "войти", указываем данные, смотрим результаты проверки |
| Регистрация | Все данные верные | Нажимаем кнопку "регистрация", указываем данные, смотрим результаты проверки |
| Регистрация | Только верный логин | Нажимаем кнопку "регистрация", указываем данные, смотрим результаты проверки |
| Регистрация | Неверный логин, спецсимволы | Нажимаем кнопку "регистрация", указываем данные, смотрим результаты проверки |
| Регистрация | Только верная почта | Нажимаем кнопку "регистрация", указываем данные, смотрим результаты проверки |
| Регистрация | Неверный | Нажимаем кнопку "регистрация", указываем данные, смотрим результаты проверки |
| Регистрация | Только верный пароль | Нажимаем кнопку "регистрация", указываем данные, смотрим результаты проверки |
| Регистрация | Неверный | Нажимаем кнопку "регистрация", указываем данные, смотрим результаты проверки |
| Нагрузочный | Регистрация множества пользователей | Нажимаем кнопку "регистрация", указываем данные, смотрим результаты проверки |
| Нагрузочный | Множественная авторизация | Нажимаем кнопку "войти", указываем данные, смотрим результаты проверки |

**Листинг**

**import** pytest  
**from** selenium.webdriver.support **import** expected\_conditions **as** EC  
**from** selenium.webdriver.common.by **import** By  
**from** selenium.webdriver.support.wait **import** WebDriverWait  
  
**from** selenium **import** webdriver  
**from** selenium.webdriver.common.keys **import** Keys  
  
  
*# Функция ожидания элементов***def** wait\_of\_element\_located(xpath, driver\_init):  
 element = WebDriverWait(driver\_init, 10).until(  
 EC.presence\_of\_element\_located(  
 (By.XPATH, xpath)  
 )  
 )  
 **return** element  
  
  
*# Вынесем инициализцию драйвера в отдельную фикстуру pytest*@pytest.fixture  
**def** driver\_init():  
 options = webdriver.ChromeOptions()  
 options.add\_experimental\_option(**"excludeSwitches"**, [**"enable-logging"**])  
 driver = webdriver.Chrome(options=options)  
 driver.get(**"https://www.strategium.ru/forum/"**)  
 **yield** driver  
 driver.close()  
  
  
  
  
*# Вынесем аутентификацию юзера в отдельную функцию***def** auth\_user(user\_name, password, driver\_init):  
 *# Поиск и ожидание элементов и присваивание к переменным.* open\_login = wait\_of\_element\_located(xpath=**'//\*[@id=\"elUserSignIn\"]'**, driver\_init=driver\_init)  
 input\_username = wait\_of\_element\_located(xpath=**'//\*[@name=\"auth\"]'**, driver\_init=driver\_init)  
 input\_password = wait\_of\_element\_located(xpath=**'//\*[@name=\"password\"]'**, driver\_init=driver\_init)  
 login\_button = wait\_of\_element\_located(xpath=**'//\*[@name=\"\_processLogin\"]'**, driver\_init=driver\_init)  
 *# Действия с формами* open\_login.send\_keys(Keys.RETURN)  
 input\_username.send\_keys(user\_name)  
 input\_password.send\_keys(password)  
 login\_button.send\_keys(Keys.RETURN)  
  
  
  
**def** reg\_user(user\_name, email, password):  
 options = webdriver.ChromeOptions()  
 options.add\_experimental\_option(**"excludeSwitches"**, [**"enable-logging"**])  
 driver = webdriver.Chrome(options=options)  
 driver.get(**"https://www.strategium.ru/forum/register/"**)  
 open\_reg = wait\_of\_element\_located(xpath=**'//\*[@id=\"elRegisterButton\"]'**, driver\_init=driver)  
 *# open\_reg = driver\_init.find\_element\_by\_id("elRegisterButton").click()* input\_username = wait\_of\_element\_located(xpath=**'//\*[@id=\"elInput\_username\"]'**, driver\_init=driver)  
 input\_email = wait\_of\_element\_located(xpath=**'//\*[@id=\"elInput\_email\_address\"]'**, driver\_init=driver)  
 input\_password = wait\_of\_element\_located(xpath=**'//\*[@id=\"elInput\_password\"]'**, driver\_init=driver)  
 input\_password\_confirm = wait\_of\_element\_located(xpath=**'//\*[@id=\"elInput\_password\_confirm\"]'**, driver\_init=driver)  
 open\_reg.click()  
 input\_username.send\_keys(user\_name)  
 input\_email.send\_keys(email)  
 input\_password.send\_keys(password)  
 input\_password\_confirm.send\_keys(password)  
 **if "Это значение не допускается." in** driver.page\_source:  
 driver.close()  
 **return True  
  
  
  
def** test\_open\_site(driver\_init):  
 **assert** (**"Strategium"**, driver\_init.title)  
  
  
**def** test\_search\_incor(driver\_init):  
 search = driver\_init.find\_element(By.NAME, **"q"**)  
 search.send\_keys(**"34234234"**)  
 search.send\_keys(Keys.RETURN)  
 **assert "По вашему запросу ничего не найдено. Попробуйте расширить критерии поиска." in** driver\_init.page\_source  
  
  
**def** test\_search\_cor(driver\_init):  
 search = driver\_init.find\_element(By.NAME, **"q"**)  
 search.send\_keys(**"Европа"**)  
 search.send\_keys(Keys.RETURN)  
 **assert "Европа" in** driver\_init.page\_source  
  
  
**def** test\_auth\_user(driver\_init):  
 auth\_user(**"TheRad445"**, **"070903Rad"**, driver\_init=driver\_init)  
 *# user\_name\_id = wait\_of\_element\_located(xpath='//\*[@class=\"ipsType\_dark ipsType\_large ipsType\_bold\"]/a',driver\_init=driver\_init)* user\_name\_id = driver\_init.find\_element(By.XPATH, **"//a[@title=\"Перейти в свой профиль\"]"**)  
 **assert** (user\_name\_id, **True**)  
  
**def** test\_auth\_user\_incor(driver\_init):  
 auth\_user(**"TheRad45343"**, **"3423423Radmir"**, driver\_init=driver\_init)  
 *# user\_name\_id = wait\_of\_element\_located(xpath='//\*[@class=\"ipsType\_dark ipsType\_large ipsType\_bold\"]/a',driver\_init=driver\_init)  
 # user\_name\_id = driver\_init.find\_element(By.XPATH, "//a[@title=\"Перейти в свой профиль\"]")* **assert "ввели корректное значение" in** driver\_init.page\_source  
 *# assert (user\_name\_id, True)***def** test\_reg\_user\_correct():  
 **assert** (reg\_user(**"TheRad45343"**, **"islamov.radmir2016@yandex.ru"**, **"3423423Radmir"**), **False**)  
  
  
**def** test\_reg\_user\_name\_cor():  
 **assert** (reg\_user(**"TheRad45343"**, **""**, **""**), **True**)  
  
  
**def** test\_reg\_user\_name\_incor():  
 **assert** (reg\_user(**"@user"**, **""**, **""**), **True**)  
  
  
**def** test\_reg\_user\_email\_cor():  
 **assert** (reg\_user(**""**, **"islamov.radmir2016@yandex.ru"**, **""**), **True**)  
  
  
**def** test\_reg\_user\_email\_incor():  
 **assert** (reg\_user(**""**, **"islamov.radmir2016yandexru"**, **""**), **True**)  
  
  
**def** test\_reg\_user\_password\_cor():  
 **assert** (reg\_user(**""**, **""**, **"34234324Raddfdf"**), **True**)  
  
**def** test\_reg\_user\_password\_incor():  
 **assert** (reg\_user(**""**, **""**, **"!!!@#@!@#!"**), **True**)  
  
  
**def** test\_reg\_user\_multiple():  
 **for** i **in** range(5):  
 **assert** (reg\_user(**"TheRad45343"**, **"islamov.radmir2016@yandex.ru"**, **"3423423Radmir"**), **False**)  
  
  
  
**def** test\_reg\_user\_password\_co():  
 **assert** (reg\_user(**""**, **""**, **"34234324Raddfdf"**), **True**)  
  
**def** test\_reg\_user\_password\_incr():  
 **assert** (reg\_user(**""**, **""**, **"!!!@#@!@#!"**), **True**)  
  
  
**def** test\_reg\_re\_multiple():  
 **for** i **in** range(2):  
 **assert** (reg\_user(**"TheRad45343"**, **"islamov.radmir2016@yandex.ru"**, **"3423423Radmir"**), **False**)  
  
**if** \_\_name\_\_ == **'\_\_main\_\_'**:  
 test\_open\_site(driver\_init=driver\_init)  
 test\_search\_cor(driver\_init=driver\_init)  
 test\_search\_incor(driver\_init=driver\_init)  
 test\_auth\_user(driver\_init=driver\_init)  
 test\_auth\_user\_incor(driver\_init=driver\_init)  
 test\_reg\_user\_password\_co()  
 test\_reg\_re\_multiple()  
 test\_reg\_user\_password\_incr()  
 test\_reg\_user\_correct()  
 test\_reg\_user\_name\_cor()  
 test\_reg\_user\_name\_incor()  
 test\_reg\_user\_email\_cor()  
 test\_reg\_user\_email\_incor()  
 test\_reg\_user\_password\_cor()  
 test\_reg\_user\_password\_incor()  
 test\_reg\_user\_multiple()