

СПбГУ

ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ПРОЦЕССОВ УПРАВЛЕНИЯ
КАФЕДРА КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ И МНОГОПРОЦЕССОРНЫХ СИСТЕМ

**АВТОМАТИЧЕСКОЕ ТЕСТИРОВАНИЕ
HTML-СТРАНИЦЫ САЙТА ПОЧТЫ РОССИИ**

Выполнил студент:
Корчевнюк Т.С.
группа: 305

Преподаватель:
Щеголева Н.Л.

Санкт-Петербург
2021 год

Цель тестирования

На основе технического задания научиться писать автоматические тесты, проводить их, документировать результаты и описывать рекомендации по исправлению ошибок. Тесты разрабатывались с целью проверить реакцию программы на различные вводимые данные.

Ход тестирования

Тестирование состоит из нескольких частей. Для автоматического тестирования были выбраны однообразные и времязатратные тесты: тестирование правильности валидации для текстовых форм ввода данных.

Предварительная подготовка и примечания

- Тестирование проводится при помощи приложения PyCharm, на языке Python и с использованием библиотеки Selenium.
- Всего на странице 4 вида полей форм: текстовые, цифровые, с раскрывающимся списком и совмещающие в себе текстовые и с раскрывающимся списком.
- В качестве успешного исхода теста (Тест пройден) служит вывод программой "Passed", а в качестве неуспешного исхода теста (Тест не пройден) служит вывод программой "Failed".

Результаты тестирования

Тест 1	Тестирование поля "Адрес"
Тест 1.1	Ввод пустой строки.
Прохождение теста	Тест пройден
Комментарий	
Тест 1.2	Ввод данных без выбора из всплывающего окна.
Прохождение теста	Тест пройден
Комментарий	
Тест 1.3	Ввод специальных символов
Прохождение теста	Тест пройден
Комментарий	
Тест 1.4	Ввод данных корректно.
Прохождение теста	Тест пройден
Комментарий	
Тест 2	Тестирование поля "Дата и время приезда курьера"
Тест 2.1	Состояние поля с заполненным полем "Адрес"
Прохождение теста	Тест пройден
Комментарий	
Тест 2.2	Состояние поля с пустым полем "Адрес"
Прохождение теста	Тест пройден
Комментарий	
Тест 3	Тестирование поля "Имя и фамилия"
Тест 3.1	Ввод пустой строки
Прохождение теста	Тест пройден
Комментарий	
Тест 3.2	Ввод случайного набора букв, который не является словом
Прохождение теста	Тест не пройден
Комментарий	Пользователь может допустить ошибки в имени и фамилии или отправить просто любой набор букв и заявка будет принята.
Тест 3.3	Ввод специальных символов
Прохождение теста	Тест пройден
Комментарий	
Тест 3.4	Ввод на латинице
Прохождение теста	Тест не пройден
Комментарий	Пользователь может вводить данные на латинице не смотря на то, что сайт только русскоязычный
Тест 3.5	Ввод более двух слов
Прохождение теста	Тест не пройден

Комментарий	Пользователь может ввести лишнюю информацию.
Тест 3.6	Ввод данных корректно
Прохождение теста	Тест пройден
Комментарий	
Тест 4	Тестирование поля "Телефон отправителя"
Тест 4.1	Ввод слова
Прохождение теста	Тест пройден
Комментарий	
Тест 4.2	Ввод цифр меньшей длины, чем необходимо
Прохождение теста	Тест пройден
Комментарий	
Тест 4.3	Ввод специальных символов
Прохождение теста	Тест пройден
Комментарий	
Тест 4.4	Ввод латиницы
Прохождение теста	Тест пройден
Комментарий	
Тест 4.5	Ввод цифр большей длины чем нужно
Прохождение теста	Тест пройден
Комментарий	
Тест 4.6	Ввод номера телефона, начиная с 8
Прохождение теста	Тест пройден
Комментарий	
Тест 4.7	Ввод данные корректно
Прохождение теста	Тест пройден
Комментарий	
Тест 5	Тестирование поля "Номер договора"
Тест 5.1	Ввод цифр длины > 30
Прохождение теста	Тест не пройден
Комментарий	Нет ограничений на длину
Тест 5.2	Ввод специальных символов
Прохождение теста	Тест пройден
Комментарий	
Тест 5.3	Ввод случайного набора букв
Прохождение теста	Тест не пройден
Комментарий	Пользователь может вводить буквы в числовом поле
Тест 6	Тестирование полей "Наименование организации"и "Комментарий"
Тест 6.1	Поле "Наименование организации": ввод специальных символов

Прохождение теста	Тест пройден
Комментарий	
Тест 6.2	Поле "Комментарий": ввод специальных символов
Прохождение теста	Тест пройден
Комментарий	
Тест 7	Тестирование поля "Вес"
Тест 7.1	Ввод числа длины >5
Прохождение теста	Тест пройден
Комментарий	
Тест 7.2	Ввод латиницы
Прохождение теста	Тест пройден
Комментарий	
Тест 7.3	Ввод специальных символов
Прохождение теста	Тест пройден
Комментарий	
Тест 7.4	Ввод числа > 31500
Прохождение теста	Тест пройден
Комментарий	
Тест 7.5	Ввод корректного значения
Прохождение теста	Тест пройден
Комментарий	

Код тестов

```
1 import time
2 from selenium import webdriver
3 from selenium.webdriver.support.ui import WebDriverWait
4 from selenium.webdriver.common.keys import Keys
5 from selenium.webdriver.common.action_chains import ActionChains
6 from selenium.webdriver.support.select import Select
7
8
9 def line():
10     print('-----')
11
12
13 def test_address():
14     print('Тестирование поля "Адрес"')
15     adr = driver.find_element_by_name('address')
16     text_try = (" ", 'петергоф Ботаническая 66', "%^&&")
17     for i in range(len(text_try)):
18         adr.click()
19         adr.send_keys(text_try[i])
20         driver.implicitly_wait(5)
21         driver.find_element_by_id('downshift-0-input').click()
22         if "Неправильно введен адрес" in driver.page_source:
23             print("{}" : Passed\n'.format(text_try[i]))
24             adr.send_keys(Keys.CONTROL + 'A', Keys.BACKSPACE)
25         else:
26             print("{}" : Failed\n'.format(text_try[i]))
27             adr.send_keys(Keys.CONTROL + 'A', Keys.BACKSPACE)
28     adr.send_keys('петергоф Ботаническая 66')
29     adr.click()
30     driver.implicitly_wait(20)
31     pl = driver.find_element_by_xpath("//*[contains(text(), 'г Санкт-Петербург,' +
32                                     " г Петергоф, ул Ботаническая, д. 66," +
33                                     " литера А, корп. 2')])")
34     pl.click()
35     driver.find_element_by_id('downshift-0-input').click()
36     if "Неправильно введен адрес" in driver.page_source:
37         print("{}" : Failed\n'.format('Вводим данные корректно'))
38     else:
39         print("{}" : Passed\n'.format('Вводим данные корректно'))
40     line()
41
42
43 def test_vrem():
44     print('Тестирование поля "Дата и время приезда курьера"')
45     adr = driver.find_element_by_name('address')
46     adr.send_keys(Keys.CONTROL + 'A', Keys.BACKSPACE)
47     dat = driver.find_element_by_id('downshift-2-input')
48     adr.send_keys('петергоф Ботаническая 66')
49     adr.click()
50     driver.implicitly_wait(20)
51     pl = driver.find_element_by_xpath("//*[contains(text(), 'г Санкт-Петербург,' +
52                                     " г Петергоф, ул Ботаническая, д. 66," +
53                                     " литера А, корп. 2')])")
54     pl.click()
55     dat.click()
56     if not driver.find_elements_by_id('downshift-2-item-0'):
57         print("{}" : Failed\n'.format('Работа поля с введенным адресом'))
58     else:
59         print("{}" : Passed\n'.format('Работа поля с введенным адресом'))
60     adr.send_keys(Keys.CONTROL + 'A', Keys.BACKSPACE)
61     dat.click()
62     if not driver.find_elements_by_id('downshift-2-item-0'):
63         print("{}" : Passed\n'.format('Работа поля без введенного адреса'))
64     else:
65         print("{}" : Failed\n'.format('Работа поля без введенного адреса'))
66     line()
67
68
69 def test_imf():
70     print('Тестирование поля "Имя и фамилия"')
71     name = driver.find_element_by_name('name')
72     input = ('', 'ывоарпгыарпылоа', '&&$%^&*&#^%', 'Ivan', 'Иванов Иван Иванович')
73     for i in range(len(input)):
74         name.click()
75         name.send_keys(input[i])
76         driver.find_element_by_id('downshift-0-input').click()
77         if "Укажите имя и фамилию" in driver.page_source:
78             print("{}" : Passed\n'.format(input[i]))
```

```

79         name.send_keys(Keys.CONTROL + 'A', Keys.BACKSPACE)
80     else:
81         print("{} : Failed\n".format(input[i]))
82         name.send_keys(Keys.CONTROL + 'A', Keys.BACKSPACE)
83     name.click()
84     name.send_keys('Иван Иванов')
85     driver.find_element_by_id('downshift-0-input').click()
86     if "Укажите имя и фамилию" in driver.page_source:
87         print("{} : Failed\n".format('Вводим данные корректно'))
88     else:
89         print("{} : Passed\n".format('Вводим данные корректно'))
90     line()
91
92
93 def test_telef():
94     print('Тестирование поля "Телефон отправителя"')
95     tef = driver.find_element_by_name('phone')
96     input = ('Иван', '3458247', '&^$#', 'sfdhdf', '30867868628745067', '8919058735')
97     for i in range(len(input)):
98         tef.click()
99         tef.send_keys(input[i])
100        driver.find_element_by_name('documentNumber').click()
101        if "Неверный формат телефона" in driver.page_source:
102            print("{} : Passed\n".format(input[i]))
103            tef.send_keys(Keys.CONTROL + 'A', Keys.BACKSPACE)
104        else:
105            print("{} : Failed\n".format(input[i]))
106            tef.send_keys(Keys.CONTROL + 'A', Keys.BACKSPACE)
107        tef.click()
108        tef.send_keys('9190587353')
109        driver.find_element_by_name('documentNumber').click()
110        if "Неверный формат телефона" in driver.page_source:
111            print("{} : Failed\n".format('Вводим данные корректно'))
112        else:
113            print("{} : Passed\n".format('Вводим данные корректно'))
114        line()
115
116
117 def test_dog():
118     print('Тестирование поля "Номер договора"')
119     dock = driver.find_element_by_name('documentNumber')
120     dock.click()
121     input = ('1234567890987654321123456789098765432112345678',
122             '#$%^&*()(*&^$#', 'зфывапролджжэжд')
123     for i in range(len(input)):
124         dock.send_keys(input[i])
125         driver.find_element_by_name('organizationName').click()
126         if dock.get_attribute('value') == '' or \
127            dock.get_attribute('value').isdigit() == True:
128             if len(input[i]) <= 30:
129                 print("{} : Passed\n".format(input[i]))
130             else:
131                 print("{} : Failed\n".format(input[i]))
132             dock.send_keys(Keys.CONTROL + 'A', Keys.BACKSPACE)
133         else:
134             print("{} : Failed\n".format(input[i]))
135             dock.send_keys(Keys.CONTROL + 'A', Keys.BACKSPACE)
136     line()
137
138
139 def test_org():
140     list = ('Тестирование поля "Наименование организации"',
141            'Тестирование поля "Комментарий"')
142     id = ('organizationName', 'comment')
143     for i in range(len(list)):
144         print(list[i])
145         pol = driver.find_element_by_id(id[i])
146         pol.send_keys('@#$$$&*()')
147         if pol.get_attribute('value') == '':
148             print("{} : Passed\n".format('@#$$$&*()'))
149         else:
150             print("{} : Failed\n".format('@#$$$&*()'))
151     line()
152
153
154 def test-ves():
155     print('Тестирование поля "Вес"')
156     v = driver.find_element_by_name('weight')
157     v.click()

```

```

158 input = ('123456', 'dfgskjh', '@#$$%^&*(', '45000')
159 for i in range(len(input)):
160     v.send_keys(input[i])
161     driver.find_element_by_name('organizationName').click()
162     if "Вес не соответствует допустимому" in driver.page_source or \
163         "Укажите вес" in driver.page_source:
164         print("{} : Passed\n".format(input[i]))
165         v.send_keys(Keys.CONTROL + 'A', Keys.BACKSPACE)
166     else:
167         if len(input[i]) >= 5:
168             print("{} : Passed\n".format(input[i]))
169         else:
170             print("{} : Failed\n".format(input[i]))
171             v.send_keys(Keys.CONTROL + 'A', Keys.BACKSPACE)
172     v.send_keys('50')
173     driver.find_element_by_name('organizationName').click()
174     if "Вес не соответствует допустимому" in driver.page_source or \
175         "Укажите вес" in driver.page_source:
176         print("{} : Failed\n".format('Вводим данные корректно'))
177     else:
178         print("{} : Passed\n".format('Вводим данные корректно'))
179     line()
180
181
182 if __name__ == "__main__":
183     driver = webdriver.Chrome()
184     driver.set_window_size(1000, 700)
185     driver.implicitly_wait(20)
186     driver.get('https://www.pochta.ru/courier')
187     ActionChains(driver).move_by_offset(300, 500).click().perform()
188     driver.implicitly_wait(10)
189     start_time = time.time()
190     test_address() # тестирование поля "Адрес"
191     test_vrem()    # тестирование поля "Дата и время приезда курьера"
192     test_imf()     # тестирование поля "Имя и фамилия отправителя"
193     test_telef()   # тестирование поля "Телефон отправителя"
194     test_dog()     # тестирование поля "Номер договора"
195     test_org()     # тестирование полей "Наименование организации" и "Комментарий"
196     test_ves()     # тестирование поля "Вес отправления"
197     driver.close()
198     print("--- %s seconds ---" % (time.time() - start_time))

```

Вывод

Использование автоматического тестирования позволяет заметно ускорить процессы проверки и улучшить результаты. Например, на ручное тестирование, вместе с заполнением отчета ушло два с половиной часа. При автоматическом тестировании у меня ушел час на изучение методов и написание тестов, 50 секунд на выполнение тестов и 20 минут на заполнение отчета. В итоге, время тестирования заметно уменьшилось. Стоит отметить, что на написание новых тестов будет тратиться еще меньше времени.

Также нельзя оставить без внимания то, что автоматические тесты, позволяют выявлять ошибки лучше, поскольку из-за отсутствия человеческого фактора (при выполнении однотипных задач) невозможно что-то пропустить или забыть.