

Wrocław, 02.05.2018r



**Wyższa Szkoła Bankowa  
we Wrocławiu**

| Od 20 lat wyznaczamy nowe kierunki

# PROJEKT

Opracowała:

Agnieszka Rytwińska

Kierunek: Tester oprogramowania

Grupa 2,

I. Założenia projektu:

Dodanie ogłoszenia o pracę na stronie internetowej <https://www.praca.pl> poprzez wypełnienie i wystanie formularza.

II. Opis środowiska do testowania:

Testy są przeprowadzane na systemie Windows 10 Pro, pod przeglądarką Chrome (wersja 66.0.3359.139 (64-bitowa). Python 2.7.15, pip 10.0.1, Selenium 3.11.0, edytor kodu: PchyCharm wer.2017.2.4

III. Warunki wstępne

1. W przeglądarce otwarta jest strona <http://praca.pl>
2. Użytkownik nie jest zalogowany

IV. Przypadki testowe:

**TC1: WPROWADZENIE DANYCH DO WSZYSTKICH PÓL W FORMULARZU I WYSŁANIE OGŁOSZENIA DO PUBLIKACJI**

Kroki:

1. W przeglądarce wejdź na stronę <https://www.praca.pl>
2. Z górnego menu poprzez kliknięcie wybierz opcję DODAJ OGŁOSZENIE
3. Wypełnij pola z sekcji pierwszej – podaj dane pracodawcy
  - a) wprowadź nazwę firmy - *Arkos Sp. z o.o.*
  - b) wprowadź adres email - *arkos@wp.pl*
  - c) wybierz wielkość zatrudnienia – *od 21 do 51 prac.*
  - d) wprowadź adres *www.arkos.pl*
  - e) wybierz branżę – *Agencje zatrudnienia*
4. Wypełnij sekcję drugą – wpisz treść ogłoszenia
  - a) wprowadź stanowisko - *Tester oprogramowania*
  - b) wprowadź nr. Ref - *123/05/2018*
  - c) wybierz kraj – *Polska*
  - d) zaznacz checkbox – *dolnośląskie*
  - e) wprowadź miasto *Wrocław*
  - f) wprowadź opis stanowiska
  - g) wprowadź wymagania
  - h) wprowadź tekst w pole oferujemy
  - i) wprowadź informacje o firmie
  - j) wprowadź formę kontaktu
  - k) wybierz wynagrodzenie od – *4000*
  - l) wybierz wynagrodzenie do – *8000*
5. Wypełnij sekcję trzecią – wskaż miejsce pracy
  - a) wprowadź miasto - *Wrocław*
  - b) wprowadź ulicę - *Kłodnicka*
  - c) wprowadź numer budynku - *136*
  - d) wprowadź numer lokalu - *4*
  - e) kliknij na przycisk USTAW MAPĘ
  - f) pozostaw pusty checkbox przy opcji NIE CHCĘ WSKAZYWAĆ MIEJSCA PRACY
6. Wypełnij sekcję czwartą – dane kontaktowe
  - a) wprowadź imię - *Elwira*
  - b) wprowadź nazwisko - *Kucińska*
  - c) wprowadź adres email *elwi@wp.pl*
  - d) wprowadź telefon - *662356098*
  - e) wprowadź stanowisko/dział – *specjalista ds. kadr*
  - f) checkbox adres korespondencyjny pozostaw pusty

7. Kliknij przycisk DALEJ
8. Wypełnij sekcję piątą – wybierz rodzaj ogłoszenia
  - a) wybierz OGŁOSZENIE TYPU STANDARD
  - b) zaznacz checkbox OFERTA TYGODNIA
  - c) zaznacz checkbox SUPER OFERTA
  - d) zaznacz checkbox ODŚWIEŻANIE
  - e) pozostaw niezaznaczony checkbox LOGOTYP
9. W sekcji szóstej wybierz szablon ogłoszenia w kolorze niebieskim
10. Wypełnij sekcję siódmą – dane do faktury
  - a) Wprowadź nazwę firmy - *Arkos S.A.*
  - b) Wprowadź NIP - *9170982465*
  - c) Wprowadź kod pocztowy - *54-134*
  - d) Wprowadź miasto- *Warszawa*
  - e) Wprowadź adres - *ul. Kwiatowa 17*
  - f) Zaznacz checkbox AKCEPTUJE REGULAMIN
  - g) Odznacz checkbox przy WYRAZAM ZGODĘ NA OTRZYMYWANIE POCZTĄ ELEKTRONICZNĄ INFORMACJI HANDLOWYCH
11. Kliknij przycisk DODAJ OGŁOSZENIE

Oczekiwany rezultat: Formularz zostanie przesłany. Otworzy się nowa strona <https://www.praca.pl/zamowienie/podziekowanie.html>

## **TC2: WYBRANIE OPCJI „LOGOTYP NA LIŚCIE OFERT PRACY”**

### Kroki:

1. W przeglądarce wejdź na stronę <https://www.praca.pl>
2. Z górnego menu poprzez kliknięcie wybierz opcję DODAJ OGŁOSZENIE
3. W sekcji czwartej kliknij przycisk DALEJ
4. W sekcję piątą zaznacz checkbox LOGOTYP

Oczekiwany rezultat: Otworzy się nowa sekcja „LOGO NA LIŚCIE OFERT PRACY”

## **TC3: WYSYŁANIE WYPEŁNIONEGO FORMULARZA BEZ ZAZNACZONEJ OPCJI „AKCEPTACJA REGULAMINU”.**

### Kroki:

1. W przeglądarce wejdź na stronę <https://www.praca.pl>
2. Z górnego menu poprzez kliknięcie wybierz opcję DODAJ OGŁOSZENIE
3. Wypełnij pola z sekcji pierwszej – podaj dane pracodawcy
  - f) wprowadź nazwę firmy - *Arkos Sp. z o.o.*
  - g) wprowadź adres email - *arkos@wp.pl*
  - h) wybierz branżę – *Agencje zatrudnienia*
4. Wypełnij sekcję drugą – wpisz treść ogłoszenia
  - m) wprowadź stanowisko - *Tester oprogramowania*
  - n) wybierz kraj – *Polska*
  - o) zaznacz checkbox – *dolnośląskie*
  - p) wprowadź miasto *Wrocław*
  - q) wprowadź wymagania
  - r) wprowadź informacje o firmie
5. Wypełnij sekcję trzecią – wskaż miejsce pracy
  - g) wprowadź miasto - *Wrocław*
6. Wypełnij sekcję czwartą – dane kontaktowe
  - g) wprowadź imię - *Elwira*
  - h) wprowadź nazwisko - *Kucińska*

- i) wprowadź adres email *elwi@wp.pl*
- j) wprowadź telefon - *662356098*
- k) checkbox adres korespondencyjny pozostaw pusty
- 7. Kliknij przycisk DALEJ
- 8. Wypełnij sekcję siódmą – dane do faktury
  - h) Wprowadź nazwę firmy - *Arkos S.A.*
  - i) Wprowadź NIP – *9170982465*
  - j) Wprowadź kod pocztowy - *54-134*
  - k) Wprowadź miasto- *Warszawa*
  - l) Wprowadź adres - *ul. Kwiatowa 17*
- 9. Kliknij przycisk DODAJ OGŁOSZENIE

Oczekiwany rezultat: Użytkownik otrzymuje komunikat „To pole jest wymagane”. Formularz nie zostaje przesłany.

#### **TC4: WPROWADZENIE BŁĘDNego ADRESU EMAIL PRACODAWCY (bez znaku @).**

##### Kroki:

- 1. W przeglądarce wejdź na stronę <https://www.praca.pl>
- 2. Z górnego menu poprzez kliknięcie wybierz opcję DODAJ OGŁOSZENIE
- 3. Wypełnij pola z sekcji pierwszej – podaj dane pracodawcy
  - i) wprowadź nazwę firmy - *Arkos Sp. z o.o.*
  - j) wprowadź adres email - *arkoswp.pl*
  - k) wybierz branżę – *Agencje zatrudnienia*
- 4. Wypełnij sekcję drugą – wpisz treść ogłoszenia
  - s) wprowadź stanowisko - *Tester oprogramowania*
  - t) wybierz kraj – *Polska*
  - u) zaznacz checkbox – *dolnośląskie*
  - v) wprowadź miasto *Wrocław*
  - w) wprowadź wymagania
  - x) wprowadź informacje o firmie
- 5. Wypełnij sekcję trzecią – wskaż miejsce pracy
  - h) wprowadź miasto - *Wrocław*
- 6. Wypełnij sekcję czwartą – dane kontaktowe
  - l) wprowadź imię - *Elwira*
  - m) wprowadź nazwisko - *Kucińska*
  - n) wprowadź adres email *elwi@wp.pl*
  - o) wprowadź telefon - *662356098*
  - p) checkbox adres korespondencyjny pozostaw pusty
- 7. Kliknij przycisk DALEJ
- 8. Wypełnij sekcję siódmą – dane do faktury
  - m) Wprowadź nazwę firmy - *Arkos S.A.*
  - n) Wprowadź NIP – *9170982465*
  - o) Wprowadź kod pocztowy - *54-134*
  - p) Wprowadź miasto- *Warszawa*
  - q) Wprowadź adres - *ul. Kwiatowa 17*
  - r) Zaznacz checkbox AKCEPTUJE REGULAMIN
- 9. Kliknij przycisk DODAJ OGŁOSZENIE

Oczekiwany rezultat: Użytkownik otrzymuje komunikat „Podany adres e-mail jest nieprawidłowy”  
Formularz nie zostaje przesłany.

```

# -*- coding: utf-8 -*-
import unittest
import time
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support.select import Select
# import ActionChains are way to automate interactions such as mouse movements, mouse button, key press, and menu
from selenium.webdriver.common.action_chains import ActionChains

# variable declaration
firm = "Arkos Sp. z o.o."
mail = "arkos@wp.pl"
incorrect_mail = "aaawp.pl"
www = "arkos.pl"
size = "3"
trade = "Agencje zatrudnienia"
workplace = "Tester oprogramowania"
number = "123/05/2018"
country = "Polska"
city = u"Wrocław"
contact_name = "Elwira"
contact_surname = u"Kucińska"
contact_email = "elwi@wp.pl"
contact_phone = "662356098"
contact_position = "specjalista ds. kadr"
specification = u"""Jako Tester, pracujący w zespole z programistami i analitykami, będziesz odpowiedzialny za:
    Przygotowywanie przypadków testowych i wykonywanie testów
    Wychwytywanie i rejestrowanie defektów podczas procesu testów
    Testy systemów podczas wdrożeń
    Przygotowanie i dbałość o dokumentację testową"""
requirements = u"""Od kandydatów oczekujemy:
    * ukończonych studiów wyższych,
    * doświadczenia w testowaniu aplikacji biznesowych,
    * doświadczenia w wykonywaniu testów manualnych, automatyzacji oraz dokumentowaniu wyników testów,
    * doświadczenia w przygotowaniu scenariuszy i danych testowych"""
offer = u"""pracę w profesjonalnym, międzynarodowym środowisku,
    udział w ciekawych projektach informatycznych dla dużych polskich oraz międzynarodowych klientów,
    możliwość zdobycia cennego doświadczenia zawodowego oraz poszerzenie wiedzy biznesowej oraz technicznej,
    dostęp do nowoczesnych technologii,
    jasno określoną ścieżkę kariery coroczna ocena wyników,
    atrakcyjne wynagrodzenie oraz pakiet socjalny"""
company = u"""Naszym klientem jest organizacja, która oferuje prace przy jednym z największych
    projektów informatycznych w kraju."""
communication = u"Prosimy o przesłanie listu motywacyjnego wraz z życiorysem na adres arkos@wp.pl "
salary_from = "4000"
salary_to = "8000"
place_city = u"Wrocław"
place_street = u"Kłodnicka"
place_building = "136"
place_flat = "4"
billing_company = "Arkos S.A."
billing_nip = "9170982465"
wrong_nip = "12"
billing_post_code = "54-134"
billing_city = "Warszawa"
billing_address = "ul. Kwiatowa 17"

```

```

class AddPublication(unittest.TestCase):

```

```

    def setUp(self):

```

```

        # open new session in Chrome
        self.driver = webdriver.Chrome()
        self.action = webdriver.ActionChains(self.driver)
        # go to praca.pl
        self.driver.get("https://www.praca.pl")
        self.driver.maximize_window()

```

```

    def test_add_publication(self):

```

```

        # open the form

```

```

driver = self.driver
action = self.action
self.driver.find_element_by_link_text('Dodaj ogłoszenie').click()

# ----- insert section 1 - content of the announcement -----
driver.find_element_by_name('data[Ogłoszenie][firma]').send_keys(firm)
driver.find_element_by_name('data[Ogłoszenie][email]').send_keys(mail)
select_employment = Select(driver.find_element_by_name('data[Firma][wielkosc_zatrudnienia_id]'))
select_employment.select_by_value(size)
driver.find_element_by_name('data[Firma][www]').send_keys(www)
select_trade = Select(driver.find_element_by_name('data[Ogłoszenie][branza_id]'))
select_trade.select_by_visible_text(trade)

# ----- insert section 2 - content of the announcement -----
driver.find_element_by_name('data[Ogłoszenie][tytuł]').send_keys(workplace)
driver.find_element_by_name('data[Ogłoszenie][nr_ref]').send_keys(number)
select_country = Select(driver.find_element_by_id('OgłoszeniePanstwold'))
select_country.select_by_visible_text(country)
select_province = driver.find_element_by_id('AnnouncementRegion1')
if not select_province.is_selected():
    select_province.click()
driver.find_element_by_name('data[Ogłoszenie][miasto][1]').send_keys(city)
# add text in specification, requirements, communications
text_specification = driver.find_element_by_id('opis_stanowiska_ifr')
action.click(text_specification).send_keys(specification)
text_requirements = driver.find_element_by_id('wymagania_ifr')
action.click(text_requirements).send_keys(requirements)
text_offer = driver.find_element_by_id('oferujemy_ifr')
action.click(text_offer).send_keys(offer)
text_company = driver.find_element_by_id('info_o_firmie_ifr')
action.click(text_company).send_keys(company)
text_communication = driver.find_element_by_id('forma_kontakt_u_ifr')
action.click(text_communication).send_keys(communication)
action.perform()
select_salary_from = Select(driver.find_element_by_name('data[Ogłoszenie][wyn_min]'))
select_salary_from.select_by_value(salary_from)
select_salary_to = Select(driver.find_element_by_name('data[Ogłoszenie][wyn_max]'))
select_salary_to.select_by_value(salary_to)

# ----- section 3 - workplace -----
driver.find_element_by_name('data[Workplace][city]').send_keys(place_city)
driver.find_element_by_name('data[Workplace][street]').send_keys(place_street)
driver.find_element_by_name('data[Workplace][building_number]').send_keys(place_building)
driver.find_element_by_name('data[Workplace][flat_number]').send_keys(place_flat)
# click show place
driver.find_element_by_id('map-group-sync-btn').click()
# I want to indicate a place of work
dont_show_map = driver.find_element_by_id('dontAddToMap')
if dont_show_map.is_selected():
    dont_show_map.click()

# ----- section 4 - add contact -----
driver.find_element_by_name('data[Fosoba][imie]').send_keys(contact_name)
driver.find_element_by_name('data[Fosoba][nazwisko]').send_keys(contact_surname)
driver.find_element_by_name('data[Fosoba][email]').send_keys(contact_email)
driver.find_element_by_name('data[Fosoba][telefon]').send_keys(contact_phone)
driver.find_element_by_name('data[Fosoba][stanowisko_dzial]').send_keys(contact_position)
# choosing the next button
driver.find_element_by_id('go-to-step2').click()

# ----- section 5 - type of advertisement -----
# select standardAnnouncement
type_adv = driver.find_element_by_xpath('//*[@@id="IBody"]/div/div/form/div[3]/section[1]/div[2]/div[1]')
ActionChains(driver).move_to_element(type_adv).click(driver.find_element_by_id('announcementType2')).perform()
select_offer_week = driver.find_element_by_id('Ogłoszenie-produkt_1')
if not select_offer_week.is_selected():
    select_offer_week.click()
select_super = driver.find_element_by_id('Ogłoszenie-produkt_2')
if not select_super.is_selected():
    select_super.click()
select_refresh = driver.find_element_by_id('Ogłoszenie-produkt_4')

```

```

if not select_refresh.is_selected():
    select_refresh.click()
# do not publish the logotyp
select_logotyp = driver.find_element_by_id('Ogłoszenie-produkt_21')
if select_logotyp.is_selected():
    select_logotyp.click()

# -----section 6- templates-----#
# select blue templates
menu = driver.find_element_by_xpath('//*[@id="lBody"]/div/div/form/div[3]/section[2]/div[2]/div[1]/div[2]')
ActionChains(driver).move_to_element(menu).click(driver.find_element_by_id('templateColor2')).perform()

# -----section 7 - billing-----#
driver.find_element_by_name('data[Fadres][firma]').send_keys(billing_company)
driver.find_element_by_name('data[Fadres][nip]').send_keys(billing_nip)
driver.find_element_by_name('data[Fadres][kod_pocztowy]').send_keys(billing_post_code)
driver.find_element_by_name('data[Fadres][miasto]').send_keys(billing_city)
driver.find_element_by_name('data[Fadres][ulica]').send_keys(billing_address)
# acceptance of the regulations
consent_reg = driver.find_element_by_id('Consent-regulations')
if not consent_reg.is_selected():
    consent_reg.click()
# no consent for marketing
marketing = driver.find_element_by_id('Consent-marketing')
if marketing.is_selected():
    marketing.click()
# -----submit-----#
driver.find_element_by_xpath('//input[@type="submit"]').click()
message = driver.find_element_by_xpath('//meta[@content="https://www.praca.pl/zamowienie/podziekowanie.html"]')
self.assertTrue(message)

def test_select_logotyp(self):

    # open the form
    driver = self.driver
    self.driver.find_element_by_link_text('Dodaj ogłoszenie').click()
    # choosing the next button
    driver.find_element_by_id('go-to-step2').click()
    self.driver.set_page_load_timeout(3)
    # select a logotyp
    select_logotyp = driver.find_element_by_id('Ogłoszenie-produkt_21')
    if not select_logotyp.is_selected():
        select_logotyp.click()
    # the selection of the logotype opens a new section LOGO
    add_section_message = self.driver.find_element_by_xpath('//*[@id="listLogo-cont"]/div[2]/h2').text
    self.assertEqual(add_section_message, u'LOGO NA LIŚCIE OFERT PRACY')

def test_regulations_not_accepted(self):

    # open the form
    driver = self.driver
    action = self.action
    driver.find_element_by_link_text('Dodaj ogłoszenie').click()

    # ----- insert section 1 - content of the announcement-----#
    driver.find_element_by_name('data[Ogłoszenie][firma]').send_keys(firm)
    driver.find_element_by_name('data[Ogłoszenie][email]').send_keys(mail)
    select_trade = Select(driver.find_element_by_name('data[Ogłoszenie][branza_id]'))
    select_trade.select_by_visible_text(trade)

    # -----insert section 2 - content of the announcement-----#
    driver.find_element_by_name('data[Ogłoszenie][tytuł]').send_keys(workplace)
    select_country = Select(driver.find_element_by_id('OgłoszeniePanstwold'))
    select_country.select_by_visible_text(country)
    select_province = driver.find_element_by_id('AnnouncementRegion1')
    if not select_province.is_selected():
        select_province.click()
    text_requirements = driver.find_element_by_id('wymagania_ifr')
    action.click(text_requirements)
    action.send_keys(requirements)
    text_company = driver.find_element_by_id('info_o_firmie_ifr')

```

```

action.click(text_company)
action.send_keys(company)
action.perform()

# -----section 3 - workplace-----
driver.find_element_by_name('data[Workplace][city]').send_keys(place_city)

# -----section 4 - add contact -----
driver.find_element_by_name('data[Fosoba][imie]').send_keys(contact_name)
driver.find_element_by_name('data[Fosoba][nazwisko]').send_keys(contact_surname)
driver.find_element_by_name('data[Fosoba][email]').send_keys(contact_email)
driver.find_element_by_name("data[Fosoba][telefon]").send_keys(contact_phone)
# choosing the next button
driver.find_element_by_id('go-to-step2').click()
driver.set_page_load_timeout(5)

# -----section 7 - billing-----
driver.find_element_by_name('data[Fadres][firma]').send_keys(billing_company)
driver.find_element_by_name('data[Fadres][nip]').send_keys(billing_nip)
driver.find_element_by_name('data[Fadres][kod_pocztowy]').send_keys(billing_post_code)
driver.find_element_by_name('data[Fadres][miasto]').send_keys(billing_city)
driver.find_element_by_name('data[Fadres][ulica]').send_keys(billing_address)
# acceptance of the regulations
consent_marketing = driver.find_element_by_id('Consent-regulations')
if consent_marketing.is_selected():
    consent_marketing.click()
# -----submit-----
driver.find_element_by_xpath('//input[@type="submit"]').click()
error_message = self.driver.find_element_by_xpath('//p[@class="error"]').text
self.assertEqual(error_message, u'To pole jest wymagane!')

```

```

def test_incorrect_email(self):

```

```

# open the form
driver = self.driver
action = self.action
driver.find_element_by_link_text('Dodaj ogłoszenie').click()

# ----- insert section 1 - content of the announcement -----
driver.find_element_by_name('data[Ogłoszenie][firma]').send_keys(firm)
driver.find_element_by_name('data[Ogłoszenie][email]').send_keys(incorrect_mail)
select_trade = Select(driver.find_element_by_name('data[Ogłoszenie][branza_id]'))
select_trade.select_by_visible_text(trade)

# -----insert section 2 - content of the announcement-----
driver.find_element_by_name('data[Ogłoszenie][tytuł]').send_keys(workplace)
select_country = Select(driver.find_element_by_id('OgłoszeniePanstwold'))
select_country.select_by_visible_text(country)
select_province = driver.find_element_by_id('AnnouncementRegion1')
if not select_province.is_selected():
    select_province.click()
text_requirements = driver.find_element_by_id('wymagania_ifr')
action.click(text_requirements)
action.send_keys(requirements)
text_company = driver.find_element_by_id('info_o_firmie_ifr')
action.click(text_company)
action.send_keys(company)
action.perform()

# -----section 3 - workplace-----
driver.find_element_by_name('data[Workplace][city]').send_keys(place_city)

# -----section 4 - add contact -----
driver.find_element_by_name('data[Fosoba][imie]').send_keys(contact_name)
driver.find_element_by_name('data[Fosoba][nazwisko]').send_keys(contact_surname)
driver.find_element_by_name('data[Fosoba][email]').send_keys(contact_email)
driver.find_element_by_name("data[Fosoba][telefon]").send_keys(contact_phone)
# choosing the next button
driver.find_element_by_id('go-to-step2').click()

# -----section 7 - billing-----

```



```

driver.find_element_by_name('data[Fadres][firma]').send_keys(billing_company)
driver.find_element_by_name('data[Fadres][nip]').send_keys(billing_nip)
driver.find_element_by_name('data[Fadres][kod_pocztowy]').send_keys(billing_post_code)
driver.find_element_by_name('data[Fadres][miasto]').send_keys(billing_city)
driver.find_element_by_name('data[Fadres][ulica]').send_keys(billing_address)
consent_reg = driver.find_element_by_id('Consent-regulations')
if not consent_reg.is_selected():
    consent_reg.click()
# -----submit-----
driver.find_element_by_xpath('//input[@type="submit"]').click()
error_mes = driver.find_element_by_xpath('//*[@id="lBody"]/div/div/form/div[1]/section[1]/div[2]/div[2]/div[1]/p').text
self.assertEqual(error_mes, u'Podany adres e-mail jest nieprawidłowy')

def tearDown(self):
    self.driver.quit()

if __name__ == "__main__":
    unittest.main(verbosity=2)

```