

ДОДАТОК А

Код управляющего модуля программы

```
from browser import EdaPlayground
import openpyxl
import time
import sys
import os
from latin import latinizator

chro_param = 0

if (len(sys.argv) == 2 and sys.argv[1] == "help"):
    print(f"{sys.argv[0]} \"login\" \"password\" \"path to chrome
driver\" \"path to download projects\" \"path to input excel file\"
[visible]")
    sys.exit(0)
elif (len(sys.argv) >= 6 and isinstance(os.sys.argv[1], str)
      and isinstance(os.sys.argv[2], str)
      and isinstance(os.sys.argv[3], str)
      and isinstance(os.sys.argv[4], str)
      and isinstance(os.sys.argv[5], str)):
    script = sys.argv[0]
    eda_login = sys.argv[1]
    eda_password = sys.argv[2]
    cro_drv_path = sys.argv[3]
    dl_path = sys.argv[4]
    xl_path = sys.argv[5]
    if len(sys.argv) > 6 and isinstance(os.sys.argv[6], str) and sys.argv[6]
    == "visible":
        chro_param = 1
else:
    print("Invalid input parameters. Enter \"help\" for more information.")
    sys.exit(0)

xl_path_res = xl_path.replace(".xls", "_result.xls")

eda_playground = EdaPlayground(eda_login, eda_password, dl_path,
cro_drv_path, chro_param)
eda_playground.google_login()

eda_playground.browser_hide()

in_book = openpyxl.load_workbook(filename = xl_path, read_only = True)
sheet = in_book.active

if os.path.exists(xl_path_res):
    xl = openpyxl.load_workbook(filename = xl_path_res)
    xls = xl.active
    xls.delete_cols(1, 3)
    xl.save(xl_path_res)
else:
```

```

res_sheet = openpyxl.Workbook()
res_sheet.save(xl_path_res)

res_book = openpyxl.load_workbook(filename = xl_path_res)
res_sheet = res_book.active

res_sheet.cell(row=1, column=1).value="Surname"
res_sheet.cell(row=1, column=2).value="Variant"
res_sheet.cell(row=1, column=3).value="Path"
res_sheet.cell(row=1, column=4).value="Result"

i = 2
while i <= sheet.max_row:

    surname = latinizator(sheet.cell(row=i, column=1).value)
    uri = sheet.cell(row=i, column=2).value
    var = sheet.cell(row=i, column=3).value

    res_sheet.cell(row=i, column=1).value=surname
    res_sheet.cell(row=i, column=2).value=var

    if not (isinstance(uri, str) and isinstance(var, int)):
        i += 1
        continue

    download_path, download_result = eda_playground.download_project(uri,
f"{surname}_var{var}")

    if download_result is not None:
        res_sheet.cell(row=i, column=3).value = download_path
    else:
        res_sheet.cell(row=i, column=3).value = "--->"

    res_sheet.cell(row=i, column=4).value = download_result

    print(f"{surname}\t\tvar_{var}\t\t{res_sheet.cell(row=i,
column=4).value}")

    i += 1

eda_playground.browser_quite()

res_book.save(xl_path_res)
res_book.close()

in_book.close()

```

ДОДАТОК Б

Код модуля роботи з браузером

```
from selenium import webdriver
from selenium.common.exceptions import NoSuchElementException
from bs4 import BeautifulSoup
import os
from os import listdir
from os.path import isfile, join
import time
from zipfile import ZipFile

class EdaPlayground:

    def __init__(self, login, password, dl_path, cro_drv_path, chro_param):
        self.login = login
        self.password = password
        self.download_path = dl_path.replace('/', '\\')
        self.chro_param = chro_param

        chrome_options = webdriver.ChromeOptions()
        prefs = {"download.default_directory" : self.download_path}
        chrome_options.add_experimental_option("prefs",prefs)
        chrome_options.add_argument("--start-maximized")

        self.cro_driver = webdriver.Chrome(chrome_options=chrome_options,
executable_path=cro_drv_path)

    def google_login(self):
        self.cro_driver.get("https://www.edaplayground.com/login")
        self.cro_driver.find_element_by_id('google_login').click()

        if not self.__wait_id_element(20, "identifierId"):
            print("Some issue durin logining")
            return

self.cro_driver.find_element_by_id("identifierId").send_keys(self.login)

self.cro_driver.find_element_by_css_selector("button[jsname='LgbsSe']").click()

        time.sleep(1.5)

self.cro_driver.find_element_by_css_selector("input[jsname='YPqjbf']").send_k
eys(self.password)

self.cro_driver.find_element_by_css_selector("button[jsname='LgbsSe']").click()
()
```

```

time.sleep(5)

def download_project(self, url, file_name):

    self.cro_driver.get(url)

    if not
self.cro_driver.find_element_by_id('resultZip').is_displayed():
        self.cro_driver.find_element_by_css_selector("div[data-
title='ToolsSimulators']").click()

        if not self.cro_driver.find_element_by_id('resultZip').is_selected():
            self.cro_driver.find_element_by_id('resultZip').click()

        if not
self.cro_driver.find_element_by_id('runButton').is_displayed():

self.cro_driver.find_element_by_css_selector("button[class='navbar-
toggle']").click()

        downloads_before = os.listdir(self.download_path)

        self.cro_driver.find_element_by_id('runButton').click()

        if not self.__wait_css_selector(20, "span[class='EXIT']"):
            return None, "Unable to download"

        downloads_after = os.listdir(self.download_path)

        for el in downloads_before:
            downloads_after.remove(el)

        zip_path = self.download_path + "\\\" + downloads_after[0]

        res = "Ok"

        if
self.cro_driver.find_element_by_id("results").get_attribute("innerHTML").find
("Compile success 0 Errors") < 0:
            res = "Compiled with errors"

        extract_path = self.download_path + "\\\" + file_name

        with ZipFile(file=zip_path, mode="r") as zip_ref:
            zip_ref.extractall(extract_path)

        os.remove(zip_path)

        return extract_path, res

def browser_hide(self):
    if self.chro_param == 0:
        self.cro_driver.set_window_position(-10000,0)
#chrome_options.add_argument('window-size=1920x1080')

```

```

        self.cro_driver.set_window_size(1920,
1080)#chrome_options.add_argument("--window-position=10000,10000")

    def browser_quite(self):
        self.cro_driver.quit()

    def __wait_css_selector(self, timeout, str_to_find):
        sleep = timeout
        while sleep:
            try:
                element =
self.cro_driver.find_element_by_css_selector(str_to_find)
            except NoSuchElementException:
                time.sleep(1)
                sleep -= 1
            else:
                if not element.is_displayed():
                    time.sleep(1)
                    sleep -= 1
                else:
                    break

        if sleep:
            return True
        else:
            return False

    def __wait_id_element(self, timeout, str_to_find):
        sleep = timeout
        while sleep:
            try:
                element = self.cro_driver.find_element_by_id(str_to_find)
            except NoSuchElementException:
                time.sleep(1)
                sleep -= 1
            else:
                if not element.is_displayed():
                    time.sleep(1)
                    sleep -= 1
                else:
                    break

        if sleep:
            return True
        else:
            return False

```

ДОДАТОК В

Графічний матеріал атестаційної роботи

Харківський національний університет радіоелектроніки
Кафедра АПОТ

Атестаційна робота

Система автоматизованого формування
бази проектів для верифікації

Другий
(магістерський)
рівень

Виконав ст. гр. СКСм19-1 Лавров А.О.
Керівник проф. каф. АПОТ Хаханова І.В.

Актуальність роботи

2

Необхідність автоматизації формування бази проектів зумовлена тим, що процес верифікації потребує багато часу. Іноді співвідношення часу верифікації до розробки складає 3:1. Автоматизація дозволяє досягти прискорення процесів верифікації, а також покращити якість за рахунок усунення людського фактору.

Публікація

Розробка автоматизованої системи тестування великого обсягу проектів,
XXIV Міжнародний молодіжний форум «Радіоелектроніка та молодь у XXI столітті»

Мета роботи

3

Метою атестаційної роботи є розробка системи автоматизованого формування бази проектів для верифікації.

Для досягнення мети були поставлені такі задачі:

- аналіз предметної області
- огляд предмету атестаційної роботи
- огляд існуючих методів автоматизації роботи з сайтами
- огляд й вибір засобів розробки
- розробка алгоритму роботи системи
- реалізація системи

Аналіз існуючих програмних засобів

- WinHTTrack
- ParttGetPriceProgram
- VistaTask

4

Методи взаємодії з веб сервісом

5

- HTTP запити
- API
- Автоматизована навігація веб-сторінкою

Бібліотеки для роботи з веб сайтами

	Python	Perl
HTTP запити	pandas, Requests, urllib / urllib2	LWP, AnyEvent::HTTP
API	-	-
Автоматизована навігація веб сторінкою	Selenium	WWW::Mechanize::Chrome, Win32::GuiTest

Методи зберігання бази проектів

6

- Бази даних
- Хмарні сховища
- Локальні сховища

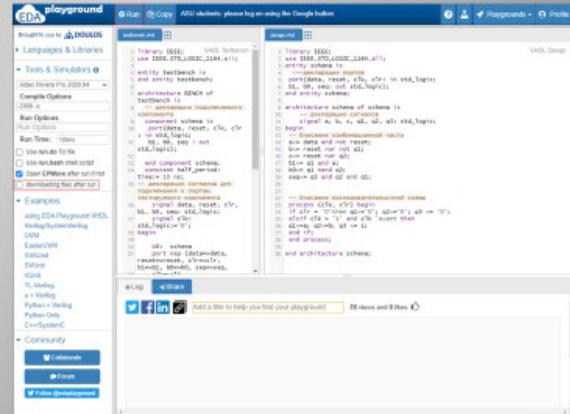
Симулятори:

- Synopsys VCS
- Cadence Incisive
- Aldec Riviera-PRO
- Incisive Specman Elite
- Та інші

Мови розробки:

- SystemVerilog/Verilog
- VHDL
- Python 2.7 + Verilog/SV
- Python 2.7 only
- C++ only
- Perl
- Csh

Вигляд відкритого проекту у EDA Playground



Алгоритм роботи

Алгоритм роботи програми



Алгоритм роботи з веб сторінкою



Опис розробленої програми

9

Формат використання програми:

"login" "password" "path to chrome driver" "path to download projects" "path to input excel file" [visible]

Мова на якій розроблена програма - Python 3.7

Використані бібліотеки:

- time
- sys
- os
- selenium
- bs4
- zipfile

Опис розробленої програми

10

Вхідні дані

	A	B	C
1	Surname	URI	Variant
2	Петров	https://www.edaplayground.com/x/UFGs	1
3	Иванов	https://www.edaplayground.com/x/bV3w	2
4	Ирюнов	https://www.edaplayground.com/x/trdv	3
5	Сидоров Иван	https://www.edaplayground.com/x/QXYS	4
6	Сидоров Василий	https://www.edaplayground.com/x/7YvR	5
7	Пеньков	https://www.edaplayground.com/x/bkTt	6
8	Дроздов	https://www.edaplayground.com/x/5Mtp	7
9	Галкин	https://www.edaplayground.com/x/9eFp	8
10	Голубев	https://www.edaplayground.com/x/kGAx	9
11	Воронова	https://www.edaplayground.com/x/PzQ	10
12	Синицин	https://www.edaplayground.com/x/dkHb	11
13	Полугаев	https://www.edaplayground.com/x/udYU	12
14	Скворцов	https://www.edaplayground.com/x/6hWP	13
15	Воробьев	https://www.edaplayground.com/x/c2bh	14
16	Дятлов	http://edaplayground.com/x/h48c	15
17	Чижова	https://www.edaplayground.com/x/BdJf	16
18			

Запуск програми

```
C:\Windows\System32\cmd.exe
C:\Users\kamik\OneDrive\Diplom\project>Download.py some.address@nure.ua XXXXXXXX C:\Users\kamik\Downloads\chromedriver_win32\chromedriver.exe D:\dw_test C:\Users\kamik\OneDrive\Рабочий стол\4.xlsx visible
```

Робота програми

11

Список вирішуваних задач:

- встановлення параметрів роботи і порядку запуску системи
- отримання результуючого проекту
- первічний аналіз даних
- перевірка на плагіат
- результати моделювання

Результат роботи програми

12

Результат роботи програми

```
C:\Windows\system32\cmd.exe
C:\Users\kamik\OneDrive\Diplom\project>Download.py artem.lavrov@du
rs.ua KIUKfu-16-9 C:\Users\kamik\Downloads\chromedriver_win32\chro
medriver.exe D:\dw_test D:\dw_test\4.xlsx visible

DevTools listening on ws://127.0.0.1:64244/devtools/browser/3f1166
40-b761-4372-bf98-0358d6e27760
[0160:8364:1223/052458.016:ERROR:device_event_log_impl.cc(211)] [0
5:24:50.016] USB: usb_device_handle_win.cc:1028 Failed to read des
criptor from node connection: #Безвизначена в шестнадцатичесле
на EPC0698. (0x1f)
[0160:8364:1223/052458.124:ERROR:device_event_log_impl.cc(211)] [0
5:24:50.124] USB: usb_device_handle_win.cc:1028 Failed to read des
criptor from node connection: #Безвизначена в шестнадцатичесле
на EPC0698. (0x1f)
Petrov      var_1      Ok
Ivanov      var_2      Ok
Kryukov     var_3      Ok
Sidorov_Ivan var_4      Ok
Sidorov_Vasily var_5      Ok
Pen'kov     var_6      Ok
Drozdov    var_7      Ok
Galkin      var_8      Ok
Golubev    var_9      Ok
Voronova   var_10     Ok
Sinicin    var_11     Ok
Popugaev   var_12     Ok
Skvorcov   var_13     Ok
Vorob'yov  var_14     Ok
Dyatlov    var_15     Unable to download
Chizhova   var_16     Ok
C:\Users\kamik\OneDrive\Diplom\project>
```

Вихідні данні

	A	B	C	D
1	Surname	Variant	Path	Result
2	Petrov	1	D:\dw_test\Petrov_var1	Ok
3	Ivanov	2	D:\dw_test\Ivanov_var2	Ok
4	Kryukov	3	D:\dw_test\Kryukov_var3	Ok
5	Sidorov_Ivan	4	D:\dw_test\Sidorov_Ivan_var4	Ok
6	Sidorov_Vasily	5	D:\dw_test\Sidorov_Vasily_var5	Ok
7	Pen'kov	6	D:\dw_test\Pen'kov_var6	Ok
8	Drozdov	7	D:\dw_test\Drozdov_var7	Ok
9	Galkin	8	D:\dw_test\Galkin_var8	Ok
10	Golubev	9	D:\dw_test\Golubev_var9	Ok
11	Voronova	10	D:\dw_test\Voronova_var10	Ok
12	Sinicin	11	D:\dw_test\Sinicin_var11	Ok
13	Popugaev	12	D:\dw_test\Popugaev_var12	Ok
14	Skvorcov	13	D:\dw_test\Skvorcov_var13	Ok
15	Vorob'yov	14	D:\dw_test\Vorob'yov_var14	Ok
16	Dyatlov	15		Unable to download
17	Chizhova	16	D:\dw_test\Chizhova_var16	Ok
18				

За поставленими вимогами та проведеними дослідженнями була розроблена система автоматизованого формування бази проектів.

У ході виконання роботи були проведено аналіз предметної області, зроблено огляд предмету атестаційної роботи й її актуальності, проаналізовано існуючі методи автоматизації роботи з сайтами, огляд й вибір способів зберігання бази проектів та методу автоматизації роботи з веб сервісами, розроблен алгоритму роботи системи та реалізована сама система автоматизованого формування бази проектів для верифікації.

Відомість атестаційної роботи

