# APÊNDICE B – Código Exemplo: Robô de Coleta WEG

### Bot Coleta WEG

from selenium import webdriver

from selenium.webdriver.firefox.options import Options

from bs4 import BeautifulSoup

import time

import html5lib

import pandas as pd

import openpyxl

import time

import os, re

## Inicializar as variáveis que irão guardar os nomes e links das notícias

news\_titles = []

news\_links = []

news\_dates = []

## Anotar o horário de coleta das informações

coletatime = time.strftime('%d.%m.%Y %Hh%Mm%Ss')

print('Coleta inciada em '+coletatime)

## Renderizar a página e receber as informações necessárias

url = "https://ri.weg.net/en/publications-and-documents/news/"

options = Options()

options.headless = True

driverPath = 'D:/Google Drive/1. UFSC/1. Graduação/20.2/Mono II/1. TCC (Mono2)/bots/geckodriver.exe'

driver = webdriver.Firefox(executable\_path=driverPath, options = options)

driver.get(url)

## Iniciar o loop para coletar informações dos 10 últimos anos

for pagenr in range(1,11):

    #Clicar no botão da próxima página (a partir do segundo loop)

    if pagenr != 1:

        xpath = "html/body/main/article/div/div/section/div[1]/ul/div[1]/div/div["+ str(pagenr+1)+ "]"

        nextYrButton = driver.find\_elements\_by\_xpath(xpath)[0]

        nextYrButton.click()

    # Dar tempo para a página renderizar

    time.sleep(5)

    # Extrair o conteudo que queremos com o beautifulsoup

    soup = BeautifulSoup(driver.page\_source, 'html5lib')

    news\_containers = soup.find\_all('div', class\_ = "document-header panel-header")

    # Preencher a lista com os titulos e links de noticias encontrados

    for container in news\_containers:

        news\_title = container.find('div', class\_ = 'document-title').a.text

        news\_titles.append(news\_title)

        news\_link = re.search("href=\"(.\*?)\"",str(container.find('div', class\_ = 'document-title'))).group(1)

        news\_links.append(news\_link)

        news\_date = container.find('div', class\_ = 'document-date').text

        news\_dates.append(news\_date)

    print('Coleta realizada com sucesso: Página '+str(pagenr))

## Fechar o browser depois de ter coletado informações de todas as páginas

driver.quit

## Criar os dataframes

weg\_news = pd.DataFrame({'date': news\_dates, 'headline': news\_titles,  'link': news\_links})

# Escrever o dataframe no Excel

planilhaPath = 'D:/Google Drive/1. UFSC/1. Graduação/20.2/Mono II/1. TCC (Mono2)/bots/Scraps/news.xlsx'

fileExists = os.path.isfile(planilhaPath)

sheetname = 'WEG\_'+coletatime

if fileExists:

    book = openpyxl.load\_workbook(planilhaPath)

    options = {}

    options['strings\_to\_formulas'] = False

    writer = pd.ExcelWriter(planilhaPath, engine='openpyxl', options = options)

    writer.book = book

    weg\_news.to\_excel(writer, sheet\_name = sheetname, index=False, header=True)

    writer.save()

else:

    weg\_news.to\_excel(planilhaPath, engine= 'openpyxl',index = False, header=True,sheet\_name= sheetname)

## Enviar confirmação de sucesso da coleta

print('Coleta realizada com sucesso. O Resultado foi salvo em '+planilhaPath)