Diez Alonso, Stela
Ruiz España, Silvia

PRACTICA 1: Web Scrapping

1. Contexto. Explicar en qué contexto se ha recolectado la información. Explique por qué el sitio web elegido proporciona dicha información.

La base de datos NANDO es publicada en una página web por la Comisión Europea y presenta la información de los organismos notificados (NB) y organismos de evaluación técnica (TAB) de acuerdo con las especificaciones técnicas armonizadas (normas armonizadas hEN y documentos de evaluación técnica EAD).

Los fabricantes europeos de materiales de construcción utilizan esta página para buscar los NB disponibles en Europa que pueden realizar la validación por tercera parte de su fabricación, como requiere el reglamento de productos de construcción RPC EU 305/2011. Esta página web se actualiza frecuentemente con los nuevos organismos notificados y aquellos que dejan de estarlo de acuerdo con los requerimientos de la legislación.

La información también resulta útil para los organismos notificados, la vigilancia de mercado y las empresas constructoras que pueden, consultando estos listados, comprobar si los organismos notificados efectivamente están autorizados para realizar esta tarea para los productos que corresponda.

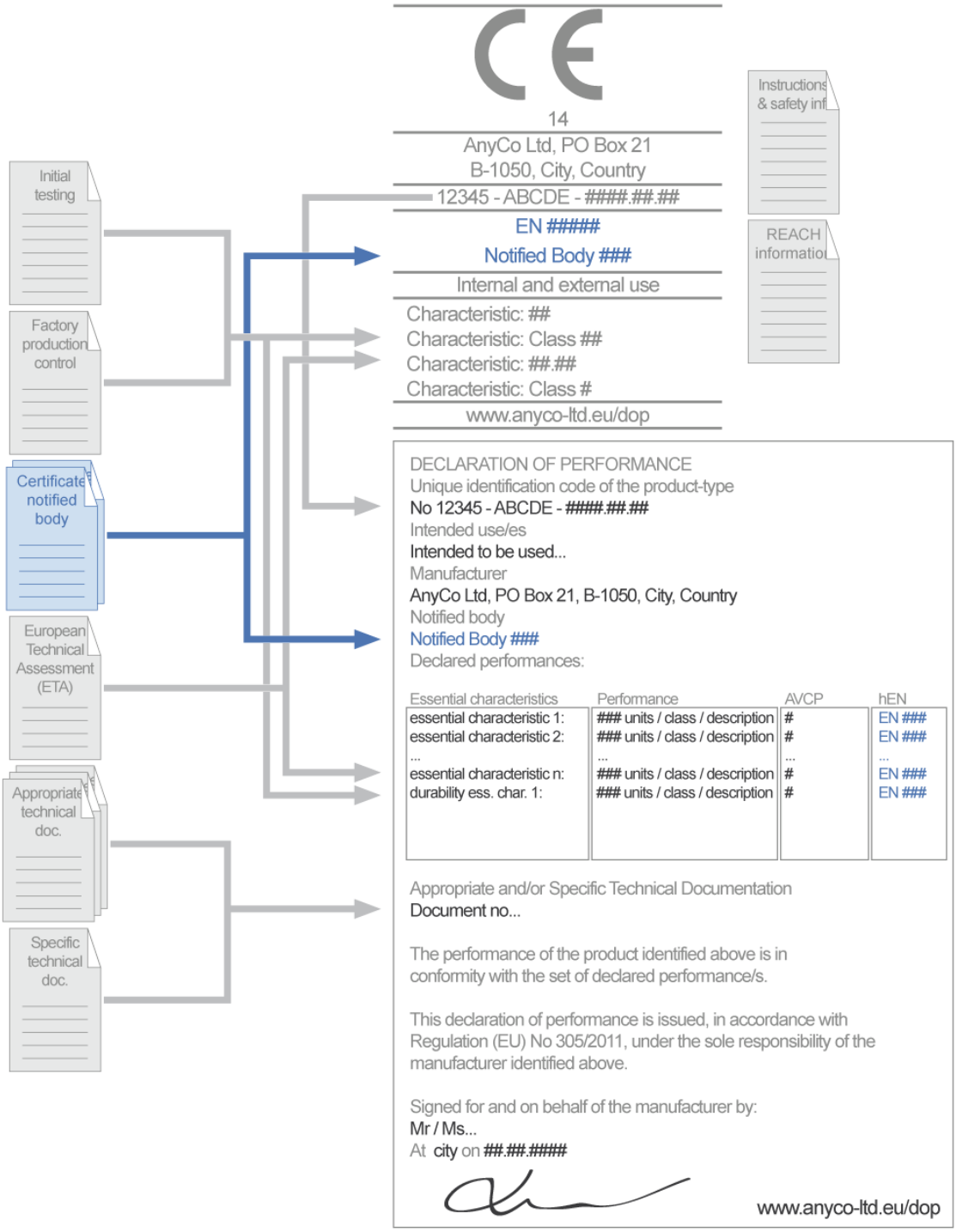
2. Definir un título para el dataset. Elegir un título que sea descriptivo.

Organismos notificados, especificaciones horizontales y normas armonizadas para la evaluación por terceras partes de materiales de construcción de acuerdo con el RPC - EU 305/2011

3. Descripción del dataset. Desarrollar una descripción breve del conjunto de datos que se ha extraído (es necesario que esta descripción tenga sentido con el título elegido).

Los datos extraídos nos proporcionan una lista de las especificaciones horizontales y las normas armonizadas incluyendo sus títulos, así como de los organismos notificados para cada una de ellas y que por tanto están autorizados a realizar las tareas de validación por tercera parte que requiere el reglamento de productos de construcción RPC EU 305/2011. El listado además proporciona datos detallados de los organismos notificados

4. Representación gráfica. Presentar una imagen o esquema que identifique el dataset visualmente



5. Contenido. Explicar los campos que incluye el dataset, el periodo de tiempo de los datos y cómo se ha recogido.

- Campos:
    - Código, corresponde el código de la especificación horizontal o de la norma armonizada
    - Descripción, corresponde el título de la especificación horizontal o de la norma armonizada. Hay que destacar que la mayoría de las especificaciones horizontales no incluyen una descripción y simplemente repiten el código
    - Empresa, nombre del NB.
    - NB, código del NB.
    - País, país donde está situado el NB.
    - Teléfono, teléfono de contacto del NB.
    - Email, email de contacto del NB.
    - Web, web del NB.

- Periodo de tiempo de los datos

La lista de los NB es una foto de las organizaciones que actualmente están autorizadas a realizar la verificación de los productos de construcción. Por lo tanto, no hay un periodo exacto de validez de los datos, ya que la lista es dinámica y sufre modificaciones cuando una organización adquiere o pierde la autorización para realizar la validación.

Este script será lanzado semanalmente para asegurarnos de tener la última versión de la lista.

- Cómo se ha recogido

En la url inicial:

https://ec.europa.eu/growth/tools-databases/nando/index.cfm?fuseaction=cp.hs&cpr=Y#hzs

hemos iniciado un bucle que recorre la lista de las especificaciones horizontales y las normas armonizadas.

De aquí obtenemos los datos relativos a la norma armonizada o la especificación horizontal, como son el código y la descripción.

Para cada uno de los elementos de esa lista, hemos entrado en el enlace que vincula la especificación horizontal o la norma. Por ejemplo, en el primer caso TS 1187:

https://ec.europa.eu/growth/tools-databases/nando/index.cfm?fuseaction=cp.nb_hzs&hs_id=152097

En esta segunda url, hemos generado otro bucle que recorre la lista de los NB autorizados para dicha norma armonizada o especificación horizontal. Entrando en cada uno de los enlaces de cada NB. Por ejemplo, continuando con el primer caso TS 1187 y el primer NB que aparece NB 0370:

https://ec.europa.eu/growth/tools-databases/nando/index.cfm?fuseaction=cp.nb&refe_cd=EPOS_43692

Donde obtenemos los datos relativos a ese NB para añadir a nuestra base de datos (empresa, NB, país…)

Una vez recogidos todos los datos, los introducimos en el CSV.

6. Agradecimientos. Presentar al propietario del conjunto de datos. Es necesario incluir citas de investigación o análisis anteriores (si los hay).

La página web es

https://ec.europa.eu/growth/tools-databases/nando/index.cfm?fuseaction=help.main

El marcado CE para productos de construcción está detallado en esta guía publicada por la Comisión Europea:

https://ec.europa.eu/growth/content/ce-marking-construction-products-step-step-guide-now-available-all-eu-languages-0_en

El propietario de esta información es la Comisión Europea, el desarrollo se implementó en 2006 y esta es la guía de referencia para el uso de la página web que fue consultada durante el desarrollo:

https://ec.europa.eu/growth/tools-databases/nando/index.cfm?fuseaction=help.main

La guía es aplicable a todas las regulaciones y directivas europeas que requieren la designación de organismos notificados, pero en nuestro estudio sólo hemos extraído los datos correspondientes al reglamento de productos de construcción. El mismo proceso de extracción sería extrapolable a otras legislaciones con ciertas modificaciones porque hay implementaciones específicas para cada ley.

7. Inspiración. Explique por qué es interesante este conjunto de datos y qué preguntas se pretenden responder.

El problema para los usuarios de esta página web es la limitada usabilidad de la información, por ejemplo, el filtrado de organismos notificados por país debe hacerse ordenando la lista por ese campo y cuando se accede a uno de los organismos notificados deben consultarse uno a uno. Las autoridades de vigilancia de mercado deben cruzar los datos del fabricante con el correspondiente organismo notificado y especificación horizontal o norma armonizada pero este proceso no se puede realizar automáticamente sin extraer los datos previamente.

Un sistema de consulta más flexible permitiría la automatización de procesos de verificación por las organizaciones interesadas.

Esta extracción nos permitirá detectar, por ejemplo:

- Si hay suficientes NB, o si hay escasez de ellos, en cada país para una norma determinada.
- Como se comportan los NB, es decir,
    - ¿Certifican todos más o menos el mismo número de normas o hay NB “pequeños” y “grandes”?
    - ¿En todos los países presentan ese mismo comportamiento?
- Cuantos certificadores tienen las normas armonizadas y las especificaciones horizontales
    - ¿Es igual en todos los países?

- ¿Es una distribución uniforme o existen normas que pueden ser verificadas por muchos?
- ¿Hay normas que no pueden ser verificadas en un país concreto?

8. Licencia. Seleccione una de estas licencias para su dataset y explique el motivo de su selección:

El set de datos está bajo copyright de la Unión Europea, bajo licencia: CC BY-SA 4.0 License. Podemos verificar esta licencia en la siguiente página web:

https://ec.europa.eu/info/legal-notice_es

9. Código. Adjuntar el código con el que se ha generado el dataset, preferiblemente en Python o, alternativamente, en R.

Nota: El código está incluido en el repositorio Github.

```python
# Importamos las librerías que vamos a necesitar
import re
import requests
from bs4 import BeautifulSoup
import csv
import time
import unicodedata

# Función para eliminar tildes
def elimina_tildes(cadena):
    s = ''.join((c for c in unicodedata.normalize('NFD',cadena) if unicodedata.category(c) != 'Mn'))
    return s

# Creamos el documento .csv para guardar los campos de interés
with open("Standards.csv","w", encoding="utf-8") as csvfile:
    spamwriter=csv.writer(csvfile, delimiter=';')
    # Cabecera
    spamwriter.writerow(['CODIGO','DESCRIPCION', 'EMPRESA', 'CODIGO_NB', 'PAIS_NB', 'TELEFONO_NB', 'EMAIL_NB', 'WEB
    cnt=0
    Datos=[]

    # Obtenemos los datos
    #Calculamos el tiempo de respuesta para introducir retrasos en las peticiones consecutivas
    for term in ["web scraping", "web crawling", "scrape this site"]:
        t0 = time.time()
        r = requests.get("https://ec.europa.eu/growth/tools-databases/nando/index.cfm?fuseaction=cp.hs&cpr=Y", para

    ## Calculo final de la estimación del tiempo de respuesta en segundos
    response_delay = time.time() - t0

    # Marcamos la url base para los enlaces siguientes extraidos de la web
    url_base = 'https://ec.europa.eu/growth/tools-databases/nando/'

    # Listas para guardar los campos de interés
    Codigo=[]
    Descripcion=[]
    Empresa=[]
    nb=[]
    Country=[]
    Phone=[]
    Email=[]
    Web=[]

    # Cargamos la primera página, donde se encuentran el listado de normas armonizadas.
    page = requests.get('https://ec.europa.eu/growth/tools-databases/nando/index.cfm?fuseaction=cp.hs&cpr=Y')
    soup = BeautifulSoup(page.content)

    contador=-1

    # Recorremos la lista de normas:
    for i in soup.find_all('a',{'class':'list'}):
        # Cada norma introducimos un tiempo de espera para espaciar nuestras peticiones
        # Espera de 10 veces el tiempo de respuesta
        time.sleep(10 * response_delay)

        ####################
        ### CAMPO CODIGO ###
        ####################
        codigo = i.string  # Valor a introducir en el campo: Codigo

        ##########################
        ### CAMPO DESCRIPCION ###
        ##########################
```

```
        cod_simple = re.findall('[A-Z]+ [0-9]*[-]*[0-9]*[:]*[0-9]*[/]*[+]*[A-Z]*[:]*[0-9]*', codigo)
        if cod_simple[0] == 'EN ISO':
            cod_simple = re.findall('[A-Z]+ [A-Z]* [0-9]*[-]*[0-9]*[:]*[0-9]*[/]*[+]*[A-Z]*[:]*[0-9]*', codigo)
        descripcion = soup.find_all(string=re.compile('\t'+cod_simple[0]))

        try:
            descripcion =(descripcion[0].strip())  # Valor a introducir en el campo: Descripcion
        except IndexError:
            descripcion ="---"
    #   print(descripcion)

        if contador==1:
            break
        else:
            contador=contador+1

        # Cargamos la segunda página, de esta página no estraeremos datos con los que rellenar campos
        # pero muestra un listado de los NB que pueden realizar la notificación de la norma que estamos
        # tratando.
        page2 = requests.get(url_base+i.get('href'))
        soup2 = BeautifulSoup(page2.content)

        # Recorremos el listado de NB de la norma
        for j in soup2.find_all('a',{'class':'list'}):
            #Para cada empresa introducimos un tiempo de espera para espaciar nuestras peticiones
            #Espera de 2 veces el tiempo de respuesta
            time.sleep(2 * response_delay)

            #Para cada elemento de la segunda página, es decir, para cada NB cargamos una página
            #en la que están sus datos.
            page3 = requests.get(url_base+j.get('href'))
            soup3 = BeautifulSoup(page3.content)

            ############################
            ### CAMPO EMPRESA y WEB ###
            ############################
            for k in soup3.find_all('strong'):
                empresa=(k.string) # Valor a introducir en el campo: Empresa
                empresa=elimina_tildes(empresa)
                Empresa.append(empresa)

                web=soup3.find('a',{'target':'_blank'})  # Valor a introducir en el campo: Web_NB
                if web:
                    # print(web.get('href'))
                    Web.append(web.get('href'))
                else:
                    Web.append("---")

            ###########################################
            ### CAMPOS NB, PAIS, TELEFONO, EMAIL      ###
            ###########################################
            NB = soup3.find_all(string=re.compile("Notified Body number :"))
            aux = NB[0].replace('Notified Body number :', '')
            NB = aux.strip() #Valor a introducir en el campo: Codigo_NB
            nb.append(NB)

            country = soup3.find_all(string=re.compile("Country :"))
            aux = country[0].replace('Country :', '')
            country = aux.strip() #Valor a introducir en el campo: Pais_NB
            Country.append(country)

            phone = soup3.find_all(string=re.compile("Phone :"))
            aux = phone[0].replace('Phone :', '')
            phone = aux.strip() #Valor a introducir en el campo: Telefono_NB
            Phone.append(phone)

            email = soup3.find_all(string=re.compile("Email :"))
            aux = email[0].replace('Email :', '')
            email = aux.strip() #Valor a introducir en el campo: Email_NB
            Email.append(email)

            # Almacenamos el Código y la Descripción
            Codigo.append(codigo)
            Descripcion.append(descripcion)

            ############################
            ### GUARDAMOS LOS DATOS ###
            ############################
            # Montamos los datos en una fila y lo guardamos en el csv
            Rows=[Codigo[cnt], Descripcion[cnt], Empresa[cnt], nb[cnt], Country[cnt], Phone[cnt], Email[cnt], Web[c
            Datos.append(Rows)
            # print(Datos[cnt])
            spamwriter.writerow(Datos[cnt])

            #Imprimimos un aviso cada 25 registros para saber que el proceso continua.
            if cnt%25 == 0:
                print("Guardando fila... " + str(cnt))
            cnt=cnt+1

csvfile.close()
print("Proceso finalizado. \nSe han guardado "+str(cnt)+" registros")
```

10. Dataset. Publicación del dataset en formato CSV en Zenodo con una pequeña descripción.

Titulo: Notified bodies, horizontal specifications and harmonised standards for construction materials third party assessment according to the CPR - EU 305/2011

Descripción: This dataset was extracted from NANDO, the website of the European Commission for notified bodies. It contains the list of notified bodies, horizontal specifications and harmonised standards for construction materials third party assessment according to the CPR - EU 305/2011

Enlace: https://zenodo.org/record/3741434#.XosBNnLtZEY

DOI: https://doi.org/10.5281/zenodo.3741434

11. Entrega. Presentar el trabajo con el DOI del dataset en Github.

DOI del dataset de Github: https://zenodo.org/badge/latestdoi/253556749

Enlace: https://github.com/Stela-Silvia/Standards-scraping/tree/1.0

Documento realizado por:

Diez Alonso, Stela

Ruiz España, Silvia

Contribuciones	Firma
Investigación previa	S.D.A y S.R.E.
Redacción de las respuestas	S.D.A y S.R.E.
Desarrollo código	S.D.A y S.R.E.