7Puentes
WEB DATA EXTRACTION & MACHINE LEARNING

Humai

Informe del proyecto integrador
Desarrollo para Web Scraping

EQUIPO 1

INTEGRANTES

Tomás Battistella

Adriano Canavero

Agustina Chrzanowski

Noelia Ledesma

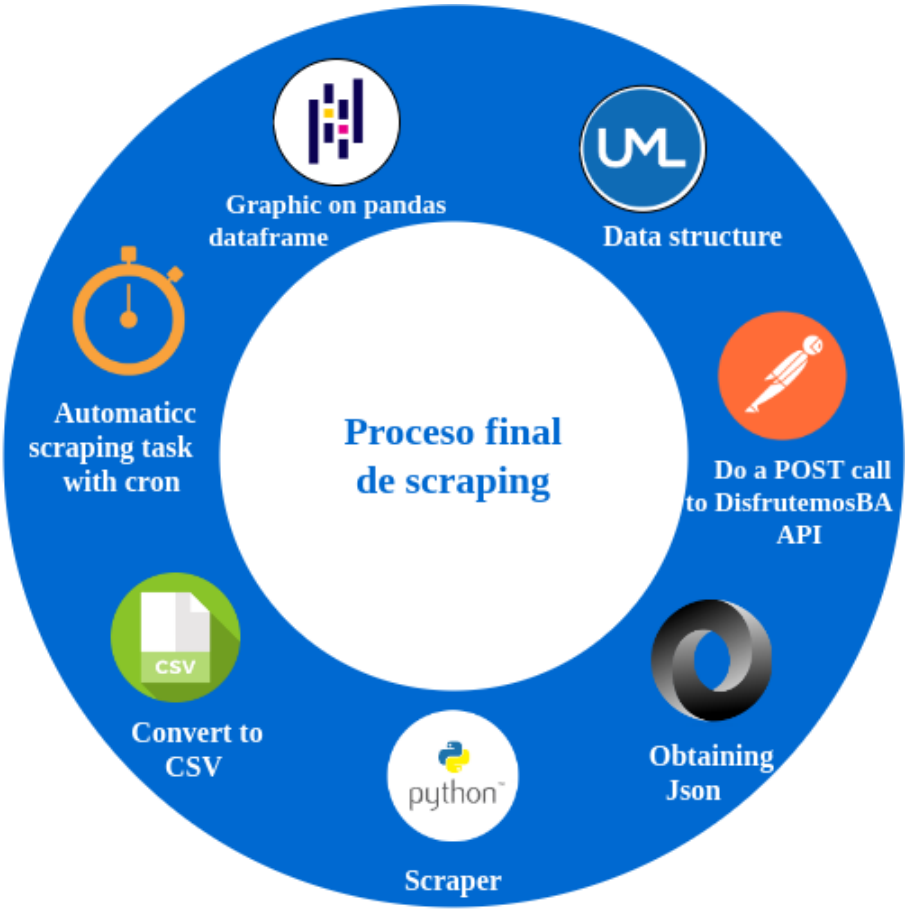
INTRODUCCIÓN

El presente informe se propone presentar las características del proyecto de scraping sobre la página web https://disfrutemosba.buenosaires.gob.ar/ del Gobierno de la ciudad de Buenos Aires, llevado a cabo en el contexto del trabajo final integrador del curso de webscraping dictado por el intituto Humai y siendo los asistentes a dichas clases integrantes del equipo de web data extraction de la empresa 7puentes.

OBJETIVOS GENERALES

Obtener automáticamente los datos de la totalidad de las actividades culturales que se realizan durante el día en la ciudad de Buenos Aires y representarlos de forma gráfica con un dashboard.

Diagrama del proceso

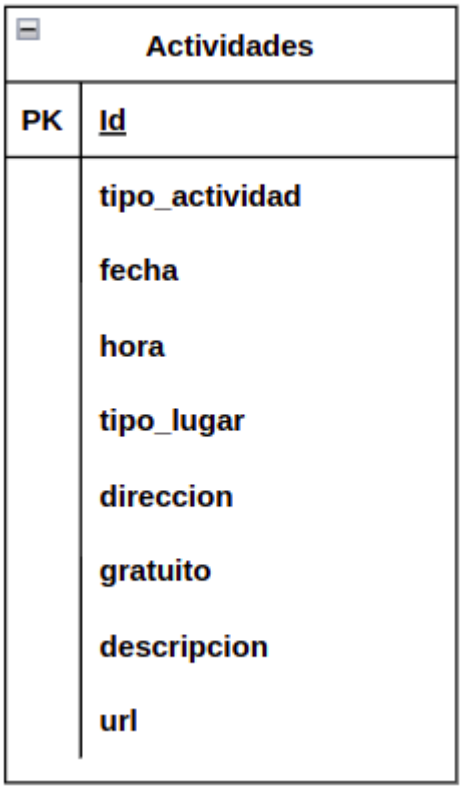


Primera etapa

Seleccionamos la página a escrapear en base a la buena accesibilidad de los datos, la variedad de campos disponibles para agrupar y visualizar mediante filtros. Antes de decantarnos por esta web descartamos otras opciones por razones de moneda atada a la variación cambiaria, baja frecuencia de cambio en los datos o porque necesitábamos hacer una normalización de los ítems a comparar y no disponíamos de tiempo suficiente para ello.

Segunda etapa

Elegimos la estructura de la tabla de datos que queríamos extraer de la fuente de datos. También elegimos el tipo de dato de cada campo en la estructura que tendría una base en SQL en caso de crearla:

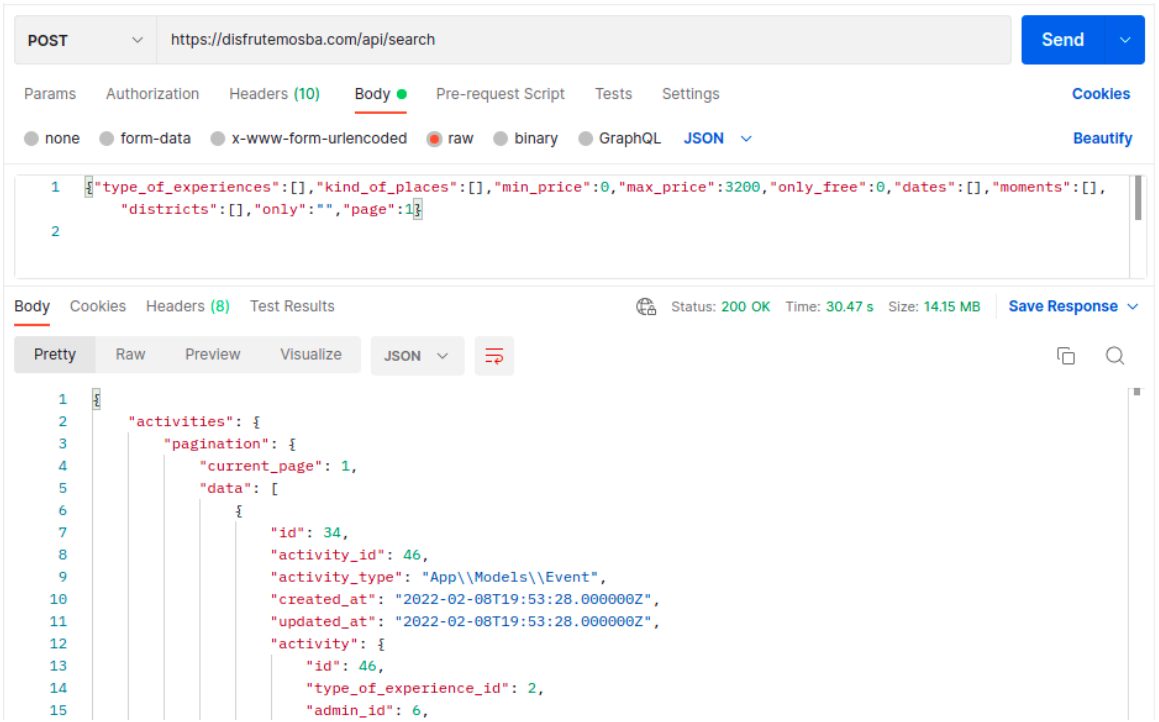


Actividades

Id (INT AI PK NN)
Tipo actividad [VARCHAR(200) NN]
Fecha [DATE]
Hora [TIME]
Tipo de lugar [VARCHAR(200) NN]
Dirección [VARCHAR(900)]
Gratuito [BINARY]
Descripcion [TEXT]
Url [VARCHAR(250)]

Tercera etapa

Se realizó la búsqueda de dónde estaban alojados los datos de las actividades. Se encontró una api en el cual se encontraba la información de cada actividad en formato JSON. Se analizaron las rutas de acceso a cada dato particular.



Cuarta etapa

Respondimos a la guía de desarrollo propuesto por el instituto Humai

- **Evaluar limitaciones que pueda tener el sitio a la hora de acceder a los datos.**

La limitación que puede tener nuestro sitio es que para poder traer los datos es necesario hacer una llamada a una api. Además, la otra dificultad es que los horarios en cada una de las actividades tienen formatos diferentes.

- **¿Cuándo se va a *scrapear*?**

Se va a scrapear dos veces al día en nuestro horario laboral.

- **¿Qué preprocesamiento hará falta hacer?**

Tendremos que hacer un request POST a la página para conseguir en la respuesta el DOM en formato JSON. Una vez que tengamos ese json tendremos que extraer los datos necesarios para poder parsearlos a un archivo CSV, de esta forma podremos actualizar el dashboard cuando sea necesario y además tener un registro histórico por cada vez que se corre.

- **¿Dónde y en qué formato guardaremos los datos?**

Se guardan en una carpeta en el github del proyecto y en formato CSV.

- **¿Cómo se actualizará el dashboard?**

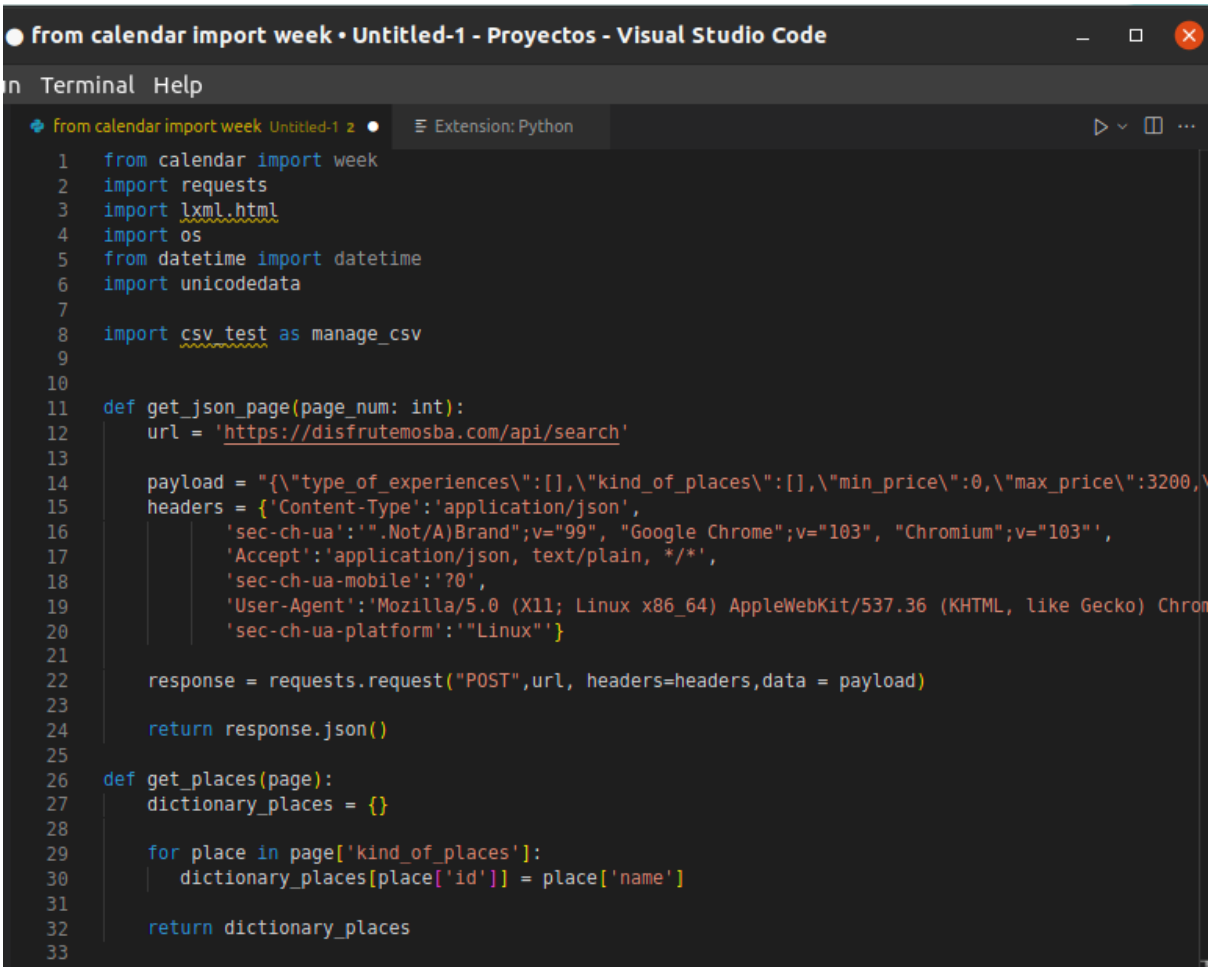
El dashboard se irá actualizando a medida que los csv se vayan cargando al sistema.

- **¿Qué podemos hacer si falla el proceso?**

Si falla el proceso de bajada de datos se intentará nuevamente y se irá guardando los csv anteriores para tener un respaldo por si acaso.

Quinta etapa

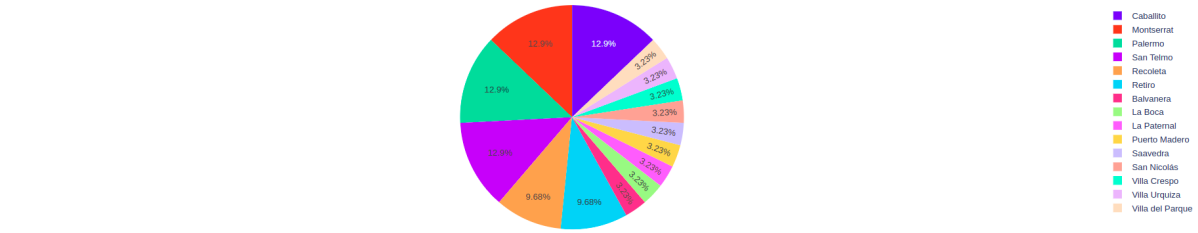
Con la información localizada se construyó el código del scraper y se extrajo un CSV de prueba con éxito. Posteriormente el código se perfeccionó para que fuera idóneo para graficar. Se utilizó pandas, plotly y dash como herramientas de visualización.




```python
import pandas as pd
import plotly.express as px
from dash import Dash, dcc, html, Input, Output
from jupyter_dash import JupyterDash
import pathlib
from datetime import date


path = str(pathlib.Path(__file__).parent.absolute())
hoy = date.today().strftime("%d-%m-%Y") # dd/mm/YY
file = path + f'/actividades_{hoy}.csv'

try:
    df = pd.read_csv(file, sep='\t')
except:
    # correr el scrapper? avisar que no corrió todavia?
    df =  pd.read_csv("/home/developer/training/proyecto_integrador_humai/data/prueba.csv", sep='\t')


df['count'] = df[df.columns[0]].count()

dftodas = df.groupby("activity_district").agg({"activity_id": "count"}).reset_index()

```

Porcentaje de actividades diario por localidad en la Ciudad de Buenos Aires

Cantidad de actividades por zona



Tipos de lugar:

Repositorio en Github del proyecto

https://github.com/Charz-a/proyecto_integrador_humai

Consideraciones finales

La experiencia fue muy satisfactoria. Nos encontramos con un desafío interesante que afrontar como equipo que nos aportó nuevos saberes no solo en lo técnico sino también en habilidades de organización y de comunicación.

Nos proponemos a futuro agregar una funcionalidad tal como comparaciones históricas de datos entre distintos días y una alerta en telegram de nuevas actividades publicadas en la web escrapeada.

Agradecemos enormemente la asistencia del instituto humai en nuestro paso por los cursos y la realización del proyecto.