

Documentación Hackathon

Gobierno de Datos y analítica

Para el cumplimiento de este reto se utilizó diferentes herramientas y tecnologías las cuales estarán descritas en este documento, tales como, Python y algunas de sus librerías, una base de datos MySQL a la cual se le conecta directamente un Dashboard realizado en la herramienta Power BI.

Contenido

Conexión al archivo de inicio.....	3
Conexión y descargue de archivos:	3
Transformación:	4
Carga de datos:.....	4
MySQL Workbench.....	5
Conexión.....	5
Creación de las tablas.....	6
Carga de datos:.....	7
Conexión Power BI a MySQL	9
Punto de conexión	9
Conexión de tablas	10
Desarrollo de Dashboard	11
Nivel 1.....	11
Nivel 2.....	12
Nivel 3.....	12
Nivel 4.....	13

Conexión al archivo de inicio

Se tiene el archivo **blob-credentials**, este fue compartido por las personas del reto y en el se encuentran las credenciales necesarias para el descargue de los archivos con extensión .PARQUET, estos dos archivos contienen información de dos tablas necesarias para iniciar el proceso de transformación de algunas columnas y cargue a una base de datos final, para ello se hizo el siguiente código:

Conexión y descargue de archivos:

Se tiene el siguiente paso a paso:

```

import pandas as pd
from azure.storage.blob import BlobServiceClient
from sqlalchemy import create_engine
import sqlalchemy
from datetime import datetime

# Datos de conexión
account_url = "https://hackathonperiferiait.blob.core.windows.net"
container_name = "hackathon-data-source"
sas_token = "sp=r1&st=2024-09-17T22:37:26Z&se=2024-10-04T06:37:26Z&spr=https&sv=2022-11-02&sr=c&sig=CYqWUat230DmfqyGFxs32vGff5LkkPFN8WsEIwFF"

# Crear un cliente de BlobService
blob_service_client = BlobServiceClient(account_url=account_url, credential=sas_token)

# Obtener el cliente del contenedor
container_client = blob_service_client.get_container_client(container_name)

# Listar los blobs (archivos) en el contenedor
blobs_list = container_client.list_blobs()

# Descargar cada blob (archivo)
for blob in blobs_list:
    # Crear un cliente de blob para cada archivo
    blob_client = container_client.get_blob_client(blob)

    # Descargar el contenido
    with open(blob.name, "wb") as download_file:
        download_file.write(blob_client.download_blob().readall())

    print(f"Descargado: {blob.name}")

# Leer los archivos .parquet
df_job_applications = pd.read_parquet("Job_Applications.parquet")
df_jobs = pd.read_parquet("Jobs.parquet")

```

Transformación:

Se realizó algunos procesos de transformación en algunas columnas específicas:

```

# Definir el límite de caracteres
MAX_CHARS = 32768

# Truncar la columna 'anonymized_resume' antes de insertar
df_job_applications['anonymized_resume'] = df_job_applications['anonymized_resume'].apply(lambda x: x[:MAX_CHARS] if isinstance(x, str) else x)

df_job_applications['job-app-created-at'] = pd.to_datetime(df_job_applications['job-app-created-at'])
df_job_applications['job-app-rejected-at'] = pd.to_datetime(df_job_applications['job-app-rejected-at'])

df_job_applications['job-app-created-at'] = df_job_applications['job-app-created-at'].dt.strftime("%d/%m/%Y %H:%M:%S")
df_job_applications['job-app-rejected-at'] = df_job_applications['job-app-rejected-at'].dt.strftime("%d/%m/%Y %H:%M:%S")

```

Carga de datos:

Como último paso, se realizó la carga de estas dos tablas a una base de datos de MySQL Workbench en localhost:

```
# Conexión a MySQL
user = 'root'          # Reemplaza con tu usuario
password = 'Camilo12*' # Reemplaza con tu contraseña
host = 'localhost'     # Por ejemplo, 'localhost' o la IP de tu servidor MySQL
database = 'sol_periferia' # El nombre de tu base de datos

# Crear la conexión
engine = create_engine(f'mysql+mysqlconnector://{user}:{password}@{host}/{database}')

# Cargar los DataFrames en MySQL
df_job_applications.to_sql('job_applications', con=engine, if_exists='replace', index=False)
df_jobs.to_sql('jobs', con=engine, if_exists='replace', index=False)

print("Datos cargados en MySQL con éxito.")
```

Datos cargados en MySQL con éxito.

MySQL Workbench

Para este ejercicio se decidió alojar las tablas en una base de datos de MySQL como localhost, se realizó la creación de las tablas y los campos que contienen los archivos, esto para garantizar un alojamiento y dominio de la información:

Conexión

Se realiza la creación de conexiones las cuales nos servirán posteriormente para conectarnos a la base de datos creada para generar visualizaciones:

Jobs

Navigator

SCHEMAS

Filter objects

sol_periferia

Tables

job_applications

jobs

Views

Stored Procedures

Functions

sys

Query 1 job_applications jobs job_applications - Table jobs - Table

Table Name: jobs

Charset/Collation: utf8mb4 utf8mb4_0900_ai_ci

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
job-id	BIGINT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
job-internal	TINYINT(1)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
job-status	TEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
job-title	TEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
job-remote-status	TEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
job-employment-type	TEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
job-currency	TEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
department-id	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
role-id	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
requisition-id	BIGINT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
role-name	TEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
department-name	TEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
requisition-job-title	TEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
requisition-job-description	TEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
requisition-country	TEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
requisition-salary-time-unit	TEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
requisition-number-of-open...	BIGINT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
requisition-technologies	TEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
requisition-experience	TEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Carga de datos:

En este punto ya se puede realizar la carga de los datos mediante el código de Python indicado anteriormente, ya que se encuentran creadas las tablas y los campos específicos:

Job_application

Navigator

SCHEMAS

Filter objects

sol_periferia

Tables

job_applications

jobs

Views

Stored Procedures

Functions

sys

Query 1 job_applications jobs job_applications - Table jobs - Table

1 • SELECT * FROM sol_periferia.job_applications

Result Grid

Filter Rows: Exports: Wrap Cell Contents: Fetch rows:

job-app-id	job-app-created-at	job-app-rejected-at	job-app-sourced-application	candidate-id	job-id	reject-reason-id	candidate-connected	candidate-internal	candidate-referred	candidate-sourced	candidate-unsubscrib
105172933	01/08/2024 00:27:10	12/08/2024 09:02:04	0	75277289	4785672	918805	1	0	0	0	0
105172943	01/08/2024 00:27:25	01/08/2024 08:29:21	0	75277289	478308	882625	1	0	0	0	0
105170773	01/08/2024 00:29:36	02/08/2024 16:49:17	0	91154025	4784527	897254	0	0	0	0	0
105182504	01/08/2024 01:46:02	01/08/2024 08:09:53	0	91157990	4780778	897254	0	0	0	0	0
105183932	01/08/2024 02:02:24	01/08/2024 08:09:38	0	82445460	4780778	882625	1	0	0	0	0
105191120	01/08/2024 02:59:33	01/08/2024 09:37:04	0	91163899	4785711	882625	0	0	0	0	0
105222608	01/08/2024 06:30:53	05/08/2024 08:45:26	0	82068416	4784527	882625	1	0	0	0	0
105222912	01/08/2024 06:33:01	12/08/2024 09:02:04	0	91187967	4785672	918805	1	0	0	0	0
105228248	01/08/2024 07:06:51	02/08/2024 13:27:50	0	91191916	4785724	882625	0	0	0	0	0
105228474	01/08/2024 07:08:25	01/08/2024 11:32:25	0	91192092	4770984	882632	1	0	0	0	0
105229017	01/08/2024 07:12:12	01/08/2024 11:32:25	0	91192496	4770984	882632	1	0	0	0	0
105229089	01/08/2024 07:13:28	12/08/2024 09:02:04	0	91192633	4785672	918805	0	0	0	0	0
105231197	01/08/2024 07:26:58	02/08/2024 16:47:35	0	91194174	4784527	882625	1	0	0	0	0
105232649	01/08/2024 07:36:43	05/08/2024 17:13:13	0	91195284	4785256	901966	1	0	0	0	0
105234029	01/08/2024 07:45:57	06/08/2024 11:07:34	0	91196337	4788288	882630	1	0	0	0	0
105235151	01/08/2024 07:53:06	12/08/2024 09:02:04	0	91197206	4785672	918805	0	0	1	0	0
105235483	01/08/2024 07:55:08	12/08/2024 09:02:04	0	91197465	4785672	918805	0	0	0	0	0
105236202	01/08/2024 07:59:58	06/08/2024 11:07:34	0	83416424	4788288	882630	1	0	0	0	0
105236892	01/08/2024 08:04:43	05/08/2024 09:34:28	0	91198550	4771013	882622	0	0	0	0	0
105237147	01/08/2024 08:06:36	01/08/2024 08:50:19	0	91198743	4780778	882625	0	0	0	0	0
105238379	01/08/2024 08:15:36	05/08/2024 14:26:51	0	91199462	4749168	882632	1	0	0	0	0
105239489	01/08/2024 08:22:33	01/08/2024 10:55:51	0	67036261	4789206	882625	1	0	0	0	0
105240826	01/08/2024 08:31:13	01/08/2024 10:39:08	0	50947029	4770805	882625	1	1	0	0	1
105241560	01/08/2024 08:36:22	05/08/2024 08:45:26	0	77952161	4784527	882625	1	0	0	0	0
105241765	01/08/2024 08:37:51	05/08/2024 14:50:06	0	82436773	4791425	918805	1	1	0	0	0

Administration Schemas

Information

Table: job_applications

Columns:

- job-app-id: bigint
- job-app-created-at: text
- job-app-rejected-at: text
- job-app-sourced-application: tinyint(1)
- application: application

Jobs

Query 1 job_applications jobs job_applications - Table jobs - Table

1 • SELECT * FROM sol_periferia.jobs;

Result Grid

job-id	job-internal	job-status	job-title	job-remote-status	job-employment-type	job-currency	department-id	role-id	requestion-id	role-name	department-name	rec
4785672	0	open	Analista de calidad	none	none	COP	162209	400083	214106	Soporte de Aplicativos	Soporte	ANU
4785308	0	open	Analista Funcional	none	fully	COP	162004	399501	210484	Analista de procesos	Requerimientos	Ana
4784527	0	open	Desarrollador/a Frontend React	none	none	COP	160862	517399	213697	React	Desarrollo de Software	Des
4780778	0	open	Scrum Master	none	fully	COP	163055	402054	212906	Scrum Master	Agile	Scr
4785711	0	open	Desarrollador/a FullStack	none	none	COP	160862	399489	214032	Fullstack	Desarrollo de Software	Des
4785724	0	open	Líder Técnico	none	none	COP	160862	399486	214123	Java	Desarrollo de Software	Líde
4770964	0	open	Gestor/a de personal	none	none	COP	160852	0000	212214	0000	Recursos Humanos	GES
4782356	0	open	Especialista en soporte de infraestructura	none	none	COP	162788	402057	211302	Ingeniero de Infraestructura	Infraestructura	Esp
4788288	0	open	Desarrollador/a SQL Junior	hybrid	none	COP	160862	0000	214146	0000	Desarrollo de Software	Des
4771013	0	open	Desarrollador/a Full Stack AI/MS	none	none	PEN	160862	399489	211003	Fullstack	Desarrollo de Software	Des
4749168	0	open	Analista de bienestar y SST	none	none	COP	160852	399492	210620	Analista de bienestar	Recursos Humanos	Ana
4789206	0	open	Arquitecto de Soluciones	hybrid	none	0000	160870	399482	214152	Arquitecto de Soluciones	Arquitectura de Softw...	Arg
4770805	0	open	Delivery Manager	hybrid	fully	PEN	160869	519721	211844	Analista de proyectos	Gestión de Proyectos	Del
4791425	0	open	Gerente de Proyectos	hybrid	none	COP	160869	519720	214153	Gerente de proyectos	Gestión de Proyectos	Ger
4749015	0	open	Desarrollador/a Nodejs	none	none	PEN	160862	657931	209302	Node.js	Desarrollo de Software	Des
4656976	0	open	Automatizador/a de pruebas Mobile	none	none	COP	160868	399495	205030	Automatizador de pruebas	Testing	AUT
4777635	0	open	Desarrollador/a Backend	none	none	COP	160862	399486	212919	Java	Desarrollo de Software	DES
4767158	0	open	Nuevas oportunidades Periferia IT Group	none	none	0000	0000	0000	212221	0000		Nue
4785150	0	open	Arquitecto/a de software	hybrid	none	COP	160870	399484	213688	Arquitecto de Software	Arquitectura de Softw...	ARC
4777621	0	open	Desarrollador/a Tibco	none	none	COP	160862	0000	212922	0000	Desarrollo de Software	Des
4629635	0	open	Analista Funcional	hybrid	none	COP	162004	399500	201215	Analista de requerimientos	Requerimientos	Ana
4765719	0	open	Desarrollador/a AS-400	hybrid	fully	COP	160862	402038	211742	AS400	Desarrollo de Software	AS4
4739520	0	open	Especialista Java/.Net	hybrid	fully	PEN	160862	399486	209163	Java	Desarrollo de Software	Esp
4577503	0	open	Líder Técnico	none	none	COP	160862	399499	197781	Líder de Desarrollo	Desarrollo de Software	Líde
4784879	0	open	Desarrollador FullStack	none	none	COP	160862	399489	214018	Fullstack	Desarrollo de Software	Des

Table: jobs

Columns:

- job-id bigint
- job-internal tryint(1)
- job-status text
- job-title text
- job-remote-

Conexión Power BI a MySQL

Power BI permite la conexión directa para bases de datos de MySQL, esto permite tener una conexión directa con las tablas necesarias para la construcción de los visuales:

Punto de conexión

Se utiliza el método de conexión que tiene el aplicativo de visualización de MySQL:

Obtener datos

mys

Todo

Base de datos

Base de datos MySQL

Importa datos de una base de datos MySQL.

Conectores certificados | Aplicaciones de plantilla

Conectar Cancelar

Base de datos MySQL

Servidor

localhost

Base de datos

sol_periferia

▸ Opciones avanzadas

Aceptar Cancelar

Conexión de tablas

Pasado este punto y luego de ingresar las credenciales pertinentes, ya podemos cargar al aplicativo las tablas necesarias para la realización de visuales:

Navegador

Opciones de presentación ▾

localhost: sol_periferia [2]

☒ sol_periferia.job_applications

☒ sol_periferia.jobs

sol_periferia.jobs

job-id	job-internal	job-status	job-title
4785672	FALSE	open	Analista de calidad
4758308	FALSE	open	Analista Funcional
4784527	FALSE	open	Desarrollador/a Frontend React
4780778	FALSE	open	Scrum Master
4785711	FALSE	open	Desarrollador/a FullStack
4785724	FALSE	open	Líder Técnico
4770984	FALSE	open	Gestor/a de personal
4758256	FALSE	open	Especialista en soporte de infraestructura
4788288	FALSE	open	Desarrollador/a SQL Junior
4771013	FALSE	open	Desarrollador/a Full Stack AWS
4749168	FALSE	open	Analista de bienestar y SST
4789206	FALSE	open	Arquitecto de Soluciones
4770805	FALSE	open	Delivery Manager
4791425	FALSE	open	Gerente de Proyectos
4749015	FALSE	open	Desarrollador/a Nodejs
4696976	FALSE	open	Automatizador/a de pruebas Mobile
4777635	FALSE	open	Desarrollador/a Backend
4767158	FALSE	open	Nuevas oportunidades Periferia IT Group
4785150	FALSE	open	Arquitecto/a de software

Los datos de la vista previa se han truncado debido a límites de tamaño.

<

>

Seleccionar tablas relacionadas

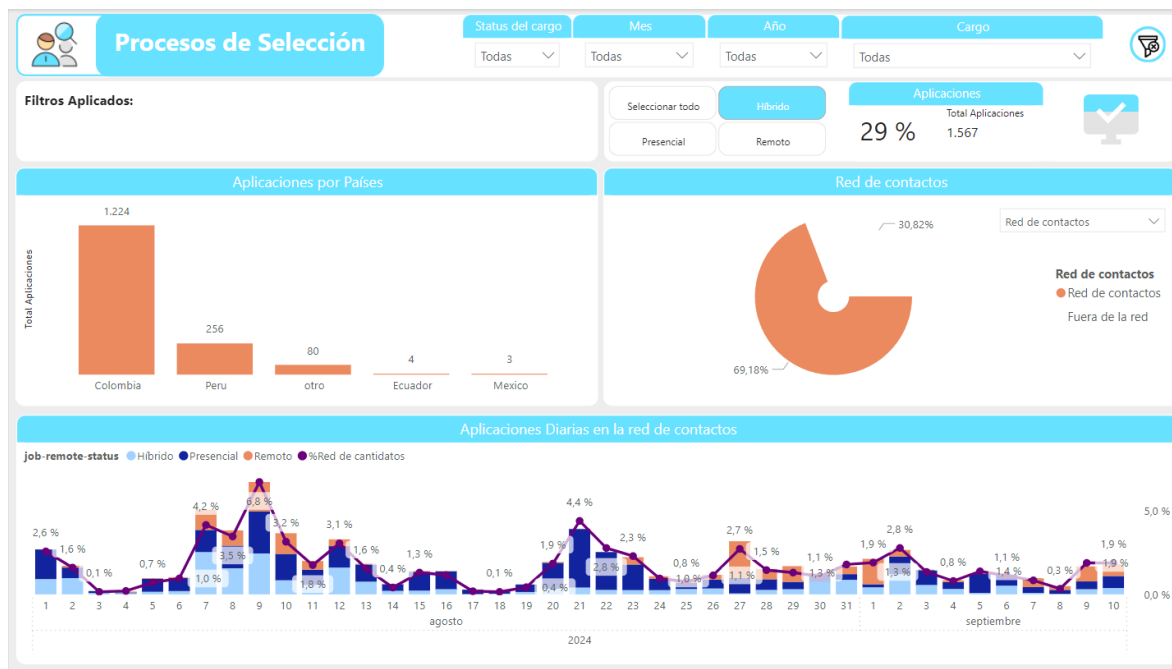
Cargar

Transformar datos

Cancelar

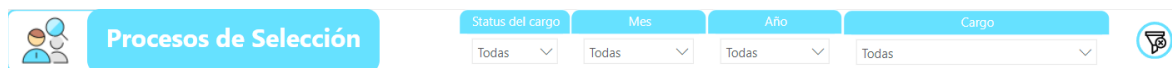
Desarrollo de Dashboard

Para la creación de este dashboard se utilizaron varios métodos de diseño, se usaron muchas facultades del aplicativo Power BI y se hizo uso de la mayoría de interacciones que este aplicativo permite, en esta parte de la documentación se van a explicar algunas de estas funcionalidades:



Nivel 1

En este nivel se observa, el ícono del negocio en conjunto con el nombre, posteriormente los filtros sugeridos para aplicar y al final esta sección se muestra un ícono de filtros, con este ícono se puede eliminar todos los filtros realizados:



Nivel 2

Para el segundo nivel se presenta una caja de “Filtros aplicados”, esta casilla muestra los filtros utilizados en el nivel anterior, no importa la cantidad de ítems que se seleccionen, siempre los va a mostrar todos sin límite:

Filtros Aplicados:

Seleccionar todo

Híbrido

Presencial

Remoto

Aplicaciones

Total Aplicaciones

29 %

1.567

Filtros Aplicados: Mes: Agosto | Año: 2024 | Cargo: Analista de Desarrollo Junior | Semillero, Analista de datos Salesforce, Analista de Aplicaciones |

Como segunda parte se encuentran 4 botones, estos dan la opción de filtrar todas las visuales entre las diferentes modalidades de trabajo como lo son: Presencial, Híbrido y remoto, el cuarto botón es para seleccionar todas las 3 opciones anteriormente indicados enseguida de esto, se observa una tarjeta con la participación de cada modalidad de trabajo y la cantidad de registros que contiene cada uno, y por último se tiene un ícono el cual se llena o se vacía acorde al filtro de modalidades de trabajo que se indique, si se seleccionan todos, la imagen estará completamente llena.

Anular toda la...

Híbrido

Presencial

Remoto

Aplicaciones

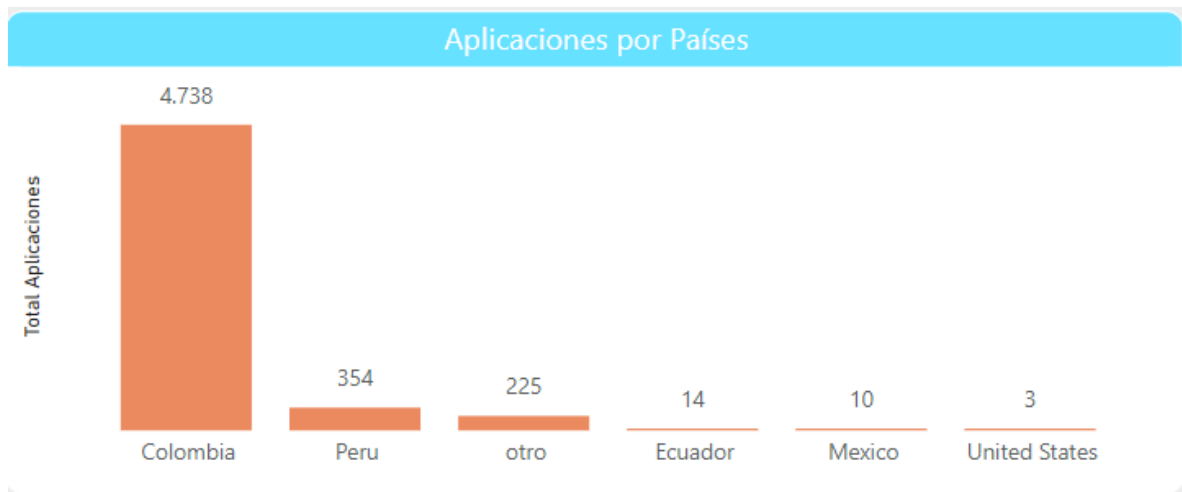
Total Aplicaciones

100 %

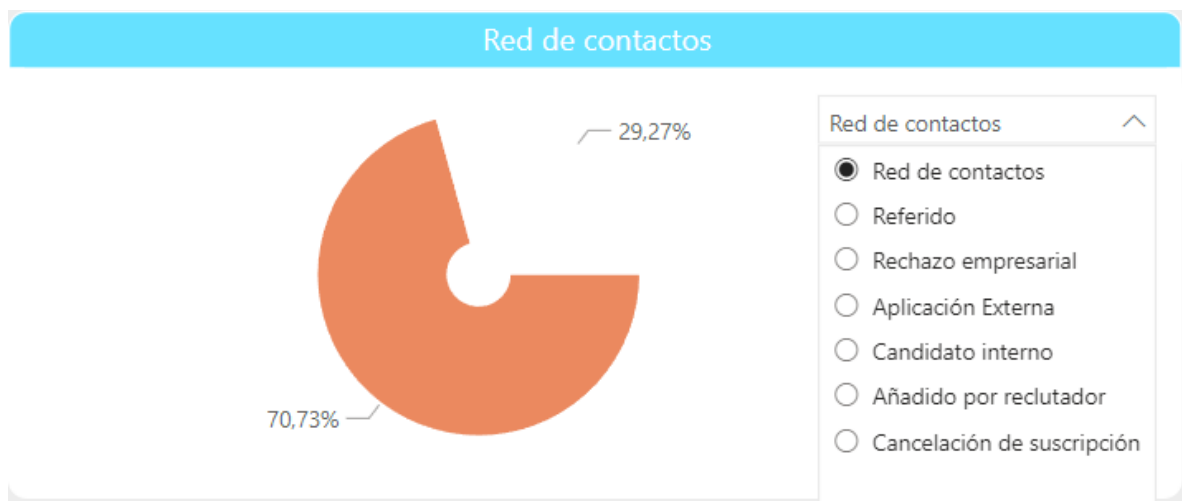
5.344

Nivel 3

A partir de este nivel se encuentran las visuales contenidas en el informe, la primera gráfica muestra la volumetría de solicitudes enviadas a cada país:



En la gráfica 2, la cual se encuentra al mismo nivel de la anterior, es un gráfico de torta compuesta por parámetros, en estos se observa la participación de diferentes ítems, entre ellos si la persona que envió la solicitud pertenece a la red de contactos, si son referidos, si fue una aplicación externa o en candidato es interno, etc.



Nivel 4

En el nivel final, se encuentra una gráfica de barras combinada con una línea de tendencia, aquí se observa la cantidad de solicitudes por día combinada con el tipo de trabajo y el % de aplicaciones que se hacen por la red de candidatos.

